Decentralized Network Based Mobility Management: Framework, System Design and Evaluation

Dissertation
zur Erlangung des mathematisch-naturwissenschaftlichen Doktorgrades
“Doctor rerum naturalium”
der Georg-August-Universität Göttingen

vorgelegt von

Niklas Neumann

aus Itzehoe

Göttingen 2011

Referent:	Professor Dr. Xiaoming Fu
Korreferent:	Professor Dr. Otto Spaniol
Tag der mündlichen Prüfung:	16. Juni 2011

Abstract

The handling of mobile nodes in the Internet has always been challenging since the Internet Protocol (IP) by design assumes that a network interface has an identifier that stays fixed at least for the duration of a data transfer. With the recent proliferation of mobile devices that provide advanced computing capabilities and resources such as netbooks, smartphones, or Internet tablets common mobile usage of the Internet is rising constantly. Moreover, applications that where not developed with mobile devices in mind and services that generally do not do well with mobility (e.g. voice-over-IP or live streaming) can be run commonly on these devices. Providing scalable and deployable mobility support in heterogeneous IP based wireless access networks that such devices are used in remains an issue.

The more traditional approaches such as Mobile IP and Proxy Mobile IP provide centralized anchors that redirect the traffic destined for the mobile node to its current point of attachment. While especially Proxy Mobile IP sees large deployments in cellular networks its centralized approach provides deployment and scalability barriers and is not well suited for distributed and fragmented environments such as hot-spots.

This thesis proposes a network based decentralized mobility management framework which can provide mobility support without the collaboration of mobile nodes and without relying on centralized elements. The framework provides easy deployment as it only requires support from access routers and scalability as the load of managing the mobile nodes is distributed among the access routers. It is well suited for distributed and fragmented environments as necessary configuration data is made available to potential handover candidate access routers in advance.

The contributions of this thesis are the development of a framework for decentralized network based mobility management, a system design based on this framework and the evaluation of the system by means of theoretical analysis, simulation, and a prototype implementation. As the evaluations show the proposed framework can provide mobility support for generic mobile nodes with handover times that converge towards the average one-way delay between access routers while inducing a signaling overhead that linearly scales with the number of mobile nodes and the number of access routers in the system.

Acknowledgments

I would like to express my gratitude to Prof. Dr. Xiaoming Fu for his supervision of this thesis, for his support and continuous guidance and for all the valuable insights and feedback he provided. His efforts helped me to continuously develop scientifically and personally during the entire time of my studies.

I would like to thank Prof. Dr. Spaniol, Prof. Dr. Hogrefe, Prof. Dr. Grabowski, Prof. Dr. Wörgötter, and Prof. Dr. Waack for participating as members of my thesis committee.

I am deeply grateful for the countless discussions and disputes with my colleagues and peers. They provided interesting thoughts, constructive criticism, and a stimulating environment which made learning, researching, and working highly enjoyable. I would like to thank Mayutan Arumaithurai, Florian Tegeler, Fang-Chun Kuo, Jun Lei, and Ralph Lübben for being outstanding colleagues and for offering their support and friendship.

Finally, I would like to thank my friends and family for their absolute and implicit support that helped me immensely all too often during this time.

Contents

List of Tables

List of Figures

List of Abbreviations

3GPP	3rd Generation Partnership Project
AR	Access Router
ARA	Access Router Approach
AS	Autonomous System
BGP	Border Gateway Protocol
BSR	Base Station Router
BTMM	Back to My Mac
CAN	Content-Addressable Network
CAR	Current Access Router
CBB	Connexion by Boing
CCN	Content-Centric-Networking
CN	Correspondent Node
DHT	Distributed Hash Table
DMMS	Decentralized Mobility Management Service
DONA	Data-Oriented Networking Architecture
EAP	Extensible Authentication Protocol
ESSID	Extended Service Set Identifier
FMIP	Fast Handovers for Mobile IP
FPMIP	Fast Proxy Mobile IP

GMX	Global Mobile eXchange
HIP	Host Identity Protocol
HMIP	Hierarchical Mobile IP
i3	Internet Indirection Infrastructure
IETF	Internet Engineering Task Force
ILNP	Identifier-Locator Network Protocol
IMEI	International Mobile Equipment Identity
IMSI	Mobile Subscriber Identity
INL	Inter-Node Latency
INL	Inter-Node Latency
IPsec	Internet Protocol Security
LMA	Local Mobility Anchor
Loc/ID sep.	Locator/ID separation
MAC	Media Access Control
MAG	Mobile Access Gateway
MAP	Mobility Anchor Point
MIH	Media Independent Handover
MIP	Mobile IP
MN	Mobile Node
MR	Mobility Router
MSS	Mobile Support Station
NAI	Network Access Identifier
NAPT	Network Address Port Translation

NAR	New access router
NAT	Network Address Translation
NDN	Named Data Networking
OUI	Organizationally Unique Identifier
PAR	Previous access router
PEA	Protocol for Evolutionary Addressing
PMIP	Proxy Mobile IP
ROFL	Routing on Flat Labels
SAR	Session Access Router
SCTP	Stream Control Transmission Protocol
SIP	Session Initiation Protocol
SPMIP	Proxy Mobile IP with Simultaneous Bindings
SSID	Service Set Identifier
SUReq	Session Update Request
SUResp	Session Update Response
TIMIP	Terminal Independent Mobility for IP
TLS	Transport Layer Security
VIR	Virtual Id Routing
WAP	Wireless Application Protocol
WINMO	Wide-Area IP Network Mobility

1. Introduction

Mobility support for large, heterogeneous networks such as the Internet is an ongoing research topic. Numerous approaches have been proposed over the years, however, there is still no widely available mobility support for mobile users in the Internet. Existing deployments, for example in cellular networks, are limited to closed networks and generally do not allow the user to roam freely between various access networks without breaking the mobility support.

On the other hand there has been a increasing proliferation of mobile computing devices such as smartphones, web-pads, tablet computers, netbooks, or electronics build in cars, trains, and other vehicles. Such devices now commonly provide their users with advanced computing capabilities, large displays, extended run times, and a wide range of connectivity options which allow them to access the Internet through various wireless communication media. As shown in Figure 1.1 there are multiple billion mobile Internet devices estimated to be in use by the end of the decade. The Morgan Stanley Mobile Internet Report 2009 believes that more users connect to the Internet via mobile devices than via conventional desktop PCs within 5 years [101].

Mobility support has been a research topic for a long time and numerous approaches have been proposed. However, over time the mobile usage scenarios have changed drastically. These days, mobile wireless access to the Internet is widely available, using numerous technologies such as IEEE 802.11 (WLAN), 802.16 (WMAN), 802.20 (MBWA), Bluetooth, Ultra-wideband, or 3G/UMTS. In addition new generations of mobile devices offer the user access to services in a way that previously was limited to stationary computers or nomadically used laptops. This leads to an increasing number of users that make use of any service on the Internet, anytime, and anywhere while previously mobile Internet access was much more limited to certain services, for example, via specialized Wireless Application Protocol (WAP) gateways [166]. As a result of this changing conditions mobility support is still an ongoing research topic as there is still a strong need for a common mobility support solution for the Internet.

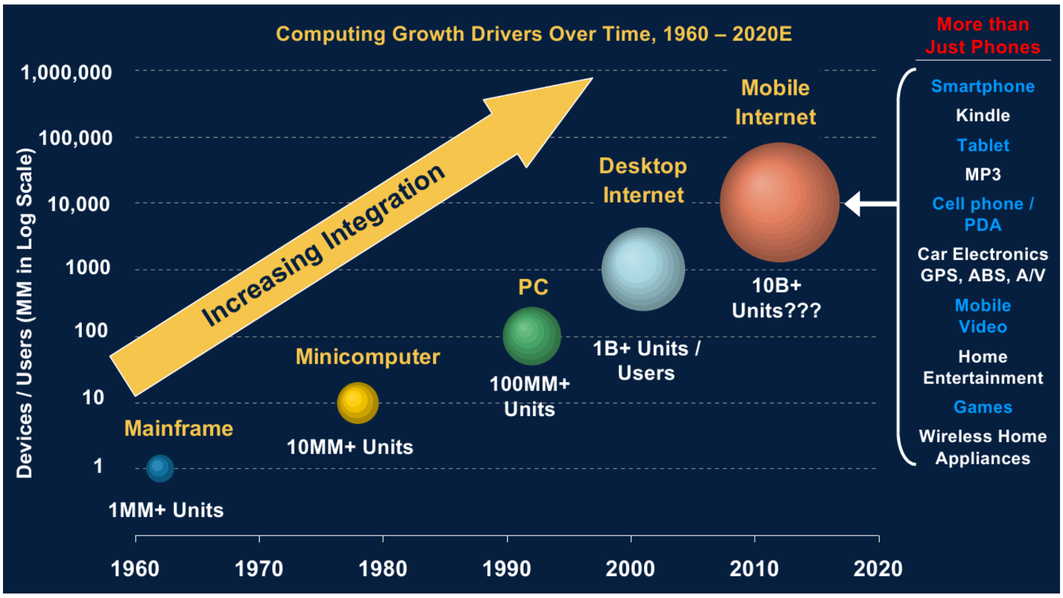


Figure 1.1.: Computing Cycle Characteristics: Tenfold increase between generations [101]

1.1. Background and Motivation

The Internet as it is used today is built around the Internet Protocol which is also commonly referred to as the "thin waist" because it is the single layer that connects upper layer protocols and lower layer protocols. When the Internet Protocol was designed, it was intended to be used over wired links between fixed nodes. Nobody was envisioning a generation of mobile devices which participate in the same Internet over wireless links while moving from place to place. This is reflected by a number of design choices in the current Internet Protocol, most notably the fact that a communication endpoint is identified by an IP address which acts as an identifier as well as a locator. Because IP addresses are topologically bound by the routing protocols to specific network locations, a node that changes its location, also changes its IP address and with it its identifier. Changing the identifier in turn breaks any existing connections that are tied to the old identifier. The fact that a change of the (IP) address breaks existing connections makes mobility support on the network layer challenging.

Supported by a convergence of mobile access devices towards platforms that provide capabilities that were only found in stationary devices not too long ago, it stands to reason that the user's attitude towards mobile Internet access also changes. Since users access a wider variety of services on the Internet over wireless technologies more often, it is important that mobile access technologies provide a user experience that is adequate to the user's expectations which might have been formed based on wired access technologies In particular it is important for mobile access technologies to support an increasing number of mobile user's which are accessing more demanding services, such as multimedia streaming and real-time interactive applications, over extended periods of time while moving. Especially applications such as Voice-over-IP or audio/video streaming are susceptible to interruptions that can be caused during handovers between different access points.

Existing approaches can be broadly classified into three categories: host based approaches, network based approaches and clean slate approaches. Clean slate approaches have inherently high deployment barriers as they are not designed with the current Internet architecture in mind. As such they must be considered long term approaches that cannot be deployed for immediate or intermediate mobility support. Host based approaches, such as Mobile IP [64], HIP [102], or i3 [150] require changes to the mobile node and depending on the approach to correspondent nodes and intermediate nodes as well. As such they have high deployment barriers especially in operator driven scenarios where the incentive to provide mobility support lies with the network operator. Moreover, Mobile IP, a well known approach to host based mobility management requires mobility signaling over the air which is prone to losses and can impact the handover-performance. It also relies on mobility anchors that are centralized components and can negatively impact the routing efficiency. Network based approaches do not require changes to the end hosts and are the only category that lies completely within the administrative control of the network operator. However, common approaches, such as Proxy Mobile IP [51] still rely on centralized mobility anchors which constitute single point of failures, bottlenecks, and can negatively impact the routing and handover performance especially in large and fragmented environments.

1.2. Decentralized Mobility Management

While numerous approaches have been proposed over the time, so far none has seen a successful, widespread deployment outside of homogeneous and closed networks. The most notable deployment of mobility support for IP based networks today is within third-generation mobile phone systems that are based on specifications by the 3rd Generation Partnership Project (3GPP).[1] However, 3GPP mobility support is based on Proxy Mo-

[1] http://www.3gpp.org/

bile IP which introduces a centralized node, the Local Mobility Anchor, that is essential for providing mobility support and as such presents a bottleneck and a single-point of failure. Moreover, it requires a mobile node's data to be routed via this centralized anchor point which interferes with the optimal routing path for the data and can introduce severe routing inefficiencies. A topologically sound placement of a centralized mobility anchor, for example as the default gateway, can alleviate such routing inefficiencies. However, finding such a topologically suitable placement becomes more and more difficult with an increasing heterogeneity of the network. Multi-homing, mobile node to mobile node communications, and the provisioning of services across foreign networks (for example via the upcoming femtocell technology[2]) are factors that effectively handicap or even prevent a topologically optimal placement. Moreover, in fragmented or very large networks the round trip time to centralized components can be significant which adds additional latency if such components need to be queried during a handover. Besides the inherent routing inefficiencies that centralized mobility anchors have, any kind of centralized system also has scalability and reliability limitations.

The issues of centralized systems can be summarized as follows:

- **Inefficient routing** Centralized mobility anchors present additional fixed-points that a mobile node's data needs to be routed through. Routing inefficiencies occur if the mobility anchor is positioned outside the regular routing path.

- **Potentially large delays** Communicating with a centralized node has potentially higher delay than communicating with a localized node. Especially in fragmented networks or very large networks the round trip time to a centralized node can be significantly higher.

- **No individual administrative control** Deployment scenarios that span multiple administrative domains are difficult to realize as administrative control over centralized components must be regulated.

- **Bad scalability** Any centralized component presents a single point of failure and a bottleneck. Moreover, the deployment barrier is raised for small deployments if centralized components must be deployed in addition to non-centralized components.

This thesis explores an approach for a decentralized mobility management framework that is designed with the decentralized and heterogeneous nature of the Internet in mind. Instead of introducing additional centralized elements to anchor and manage mobile

[2] http://www.femtoforum.org/femto/aboutfemtocells.php

nodes, the approach leverages the individual access routers that provide a mobile node with network access and extends them with functions to provide seamless, network based mobility support for any node that moves between them. In particular, the proposed approach combines the following characteristics that set it apart from other solutions:

- **Scalability and Deployability** The individual access routers themselves are primarily responsible for any operations with regards to mobility management. This makes the approach very scalable as a deployment can easily be extended by just adding more access routers. Furthermore, it provides a low deployment barrier as a small number of access routers (starting with two) can already provide mobility support without any dependency on additional devices within the network.

- **Redundancy and Fault tolerance** Any states with regards to mobility management are primarily maintained in the individual access routers. Moreover, no centralized node is involved in the routing of data for a mobile node. This makes the approach very resilient against single node failures and provides an implicit redundancy. If an access router fails a mobile node can just associate with any other available access router and will receive continued mobility support from the network.

- **Minimized impact on routing** When a mobile node initially associates with an access router, this access router dynamically becomes the mobile node's mobility anchor for the remainder of the session. This makes the approach very friendly towards the underlying network routing structure as no additional centralized anchor points are introduced that require mobile node traffic to be routed through them. Most notable it adds no routing indirections before the mobile node hands-over for the first time and it establishes the mobile node's anchor point as far towards the edges of the network as possible which allows for efficient routing strategies, especially with regards to multi-homing.

1.3. Thesis Overview

In the remainder of this thesis, first the related work will be discussed in Chapter 2. Following, Chapter 3 introduces the overall design of the proposed mobility management framework. Chapter 4 illustrates the system design and implementation based on the framework and Chapter 5 will make a number of deployment considerations for the approach. Afterwards, in Chapter 6 an in-depth evaluation of the decentralized approach is performed. Finally, Chapter 7 concludes the thesis.

2. Related Work

The original design of the Internet Protocol as the single common communication protocol of the Internet does not include considerations for host mobility (cf. [31]). As such it does not provide the means to inherently provide mobile nodes with seamless connectivity when they move between different points of attachment. Providing such mobility support on the various layers of the Internet architecture has been a long-standing research topic. This chapter will introduce related work and the state of the art in this field. Mobility approaches will be classified into three main categories: host based mobility approaches, network based mobility approaches and clean slate approaches (for other classifications see e.g. [78]).

2.1. Related Concepts

Mobility approaches can be divided into several conceptual categories. Each has its distinct properties and requirements. To assess an approach to mobility it is important to understand the underlying concept, its intended field of application, potentials and limitations. This section will introduce a number of concepts that are common to more than one particular category of mobility solutions.

Link Layer Mobility

Some common link layer technologies can inherently provide mobility support (e.g. IEEE 802.11). A mobile node that moves within any such link layer technology will not be subjected to any communication breakdowns on the network layer or above. However, there are a number of reasons that make the sole reliance on link-layer mobility unsuitable in large scale deployments. First, any link layer solution is implicitly limited to the particular link layer technology. A mobile node that employs multiple different link layer technologies, for example using multiple interfaces, cannot be supported across different technologies. Second, even within the same link layer technology mobility support is commonly limited to the same network domain. A mobile node roaming between two network domains that are not continuously interconnected by the same link layer technology cannot be supported. Third, providing mobility on the link layer means that mobility support has to be (re-)engineered into every link layer technology that supports mobile nodes if it is to be commonly available. Providing mobility support on a higher layer, for example the network layer which is common to all communication in the current Internet, can potentially lead to a more universally available solution. While there are certainly scenarios where link layer mobility is well suited and sufficient enough, for the aforementioned reasons it does not seem applicable to rely on link layer technologies to provide mobility on an Internet wide scale.

End-to-End Mobility

End-to-end mobility requires both communicating parties (the mobile node and the correspondent node) to implement mobility support but not the intermediate nodes. Typically an end-to-end mobility solution will provide the means to overcome the address change that the Internet Protocol causes when a node moves to a different network. While end-to-end solutions can effectively provide mobility support they are subject to high deployment barriers since the correspondent node has to support them. It may seem feasible to require a mobile node to support a specific protocol or technology since it will most probably utilize it. A correspondent node on the other hand that potentially only communicates with a very small fraction of mobile nodes might not be subjected to profound changes in its networking stack lightly. End-to-end approaches to mobility are most commonly found in the upper layers of the networking stack as they are easier to deploy the higher they are implemented.

Indirection Based Mobility

Approaches that are indirection based introduce an anchor or indirection point that redirects traffic to its intended destination. Indirection based mobility is commonly based on an overlay network that provides the indirection service. An overlay network provides a service infrastructure based on the underlying (IP) network. Usually the services offered by the overlay are not inherent to the underlying network. An overlay network can be efficiently used to route messages over the IP infrastructure based on service or protocol specific parameters that are not regarded by common Internet routing. support mobility. A most simplistic overlay would be composed of just the mobile node and an anchor point which serves as the communication endpoint to correspondent nodes. The mobile node would use a service-specific protocol to signal the anchor point where to forward its traffic to. Mobile IP could be regarded as a protocol that employs such a minimalistic overlay (Mobile IP will be discussed in Section 2.2.1). A more sophisticated overlay would be composed of a multitude of nodes and offer more complex naming and routing services. Overlay networks are used to provide a multitude of services in the Internet and they are also well suited to provide mobility support. The deployability of any particular approach depends on the nodes that it requires to be part of the overlay. As being a part of the overlay requires changes to a node, approaches that necessarily require the correspondent node or a given subset of routers (e.g. all edge routers) to be part of the overlay are harder to deploy than approaches that just require the mobile node to implement the overlay.

Locator/ID Separation

A fundamental problem for efficiently supporting mobility is that in current protocols the IP address is used as an identifier as well as a locator. Transport protocols such as TCP, UDP, or SCTP use the IP address to identify the endpoint of a connection. If the IP address of a node changes, the transport protocols assume that the endpoint has changed. For routing purposes the same identifier, namely the IP address, is used to route the IP packet towards its destination. This means that if a node changes its location, it also needs to change its IP address. This peculiarity is a fundamental limitation for the support of advanced network services and functionalities such as mobility and multi-homing. Based on this fundamental predicament of IP, it has been proposed to decouple a node's primary identifier (ID) from it's location identifier (Locator) (cf. [14]). Locator/ID separation (Loc/ID sep.) proposals differ in the deployment scenario and implementation details as well as the layer they are implemented on.

The essential idea behind any Locator/ID separation approach is to provide higher layer protocols with a fixed identifier that they can use to identify a communication endpoint regardless of its location. The routing on the other hand is done based on a nodes current locator which changes whenever the node changes its point of attachment. The mapping between a locator and an identifier can be done regularly when data needs to be routed instead of just when a connection is established. This allows for a mobile node to changes locators but still be able to receive data based on its identifier. Depending on the particular approach changes are required to the end hosts or the network. A set of Locator/ID Separation base protocols is currently being developed by the Internet Engineering Task Force in the Locator/ID Separation Protocol Working Group.[1] While Locator/ID separation can solve mobility implicitly, it has recently been predominantly discussed with regards to reform the routing architecture of the Internet (e.g. [63, 92]).

The line between indirection based approaches and Locator/ID separation approach is not easy to draw as both essentially introduce a mapping layer. Indirection based

[1] http://tools.ietf.org/wg/lisp/

approaches implicitly map one IP address which identifies the mobile node with regards to the particular approach (e.g. the home address in Mobile IP) to another IP address which is the current location of a mobile node (e.g. the care-of address in Mobile IP). Locator/ID separation approaches on the other hand map an explicit identifier to an explicit locator. For the context of this thesis approaches that use an explicit identifier and an explicit locator are considered Locator/ID separation approaches and approaches that do an implicit mapping between IP addresses are considered indirection based approaches.

2.2. Host Based Mobility

Host based mobility approaches involve the mobile node itself in mobility management operations and as such require changes to the mobile node. Whether changes to the mobile node are feasible or not depends on the particular deployment scenario. In small scale homogeneous deployments, such a requirement seems easier to realize than in large scale heterogeneous deployments. This section will discuss approaches to host based mobility.

2.2.1. Indirection Approaches

This section will introduce approaches that are based on an indirection concept.

Mobile IP

Mobile IP (MIP) is a well-known mobility protocol specified by the Internet Engineering Task Force (IETF) and is available in versions for IPv4 [116] and IPv6 [64]. The basic operation of Mobile IP is shown in Figure 2.1. To make changes in the point of attachment of the mobile node transparent, a mobile node is assigned a home address that it uses when communicating with corresponding nodes. The mobile node is always expected to be reachable via this home address which is topologically anchored at the home agent inside the mobile node's home network. The function of the home agent is to forward any data for the mobile node to its actual location which is registered using a care-of address with the home agent. When a mobile node roams to another network it acquires a care-of address that is valid within the visited network. In Mobile IPv4 this is done via a special router called foreign agent. The operation of the foreign agent can also be co-located within the mobile node itself. In Mobile IPv6 foreign agents are not needed and do not exists. Essentially, Mobile IP introduces a layer of indirection by

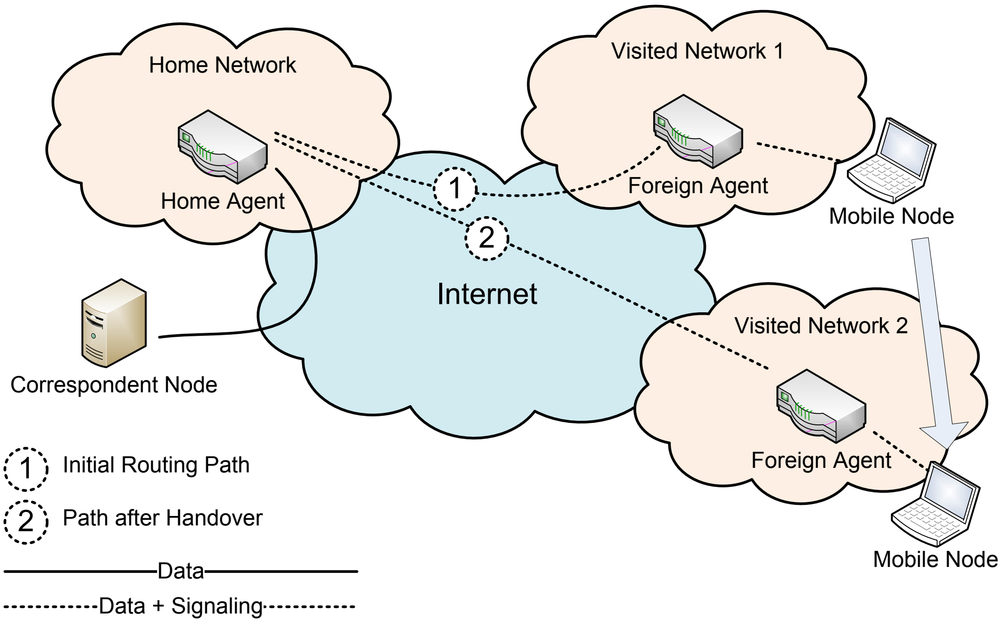


Figure 2.1.: Basic Operations of Mobile IP

providing a fixed anchor point for a mobile node through the home agent. A correspondent node can always use a mobile nodes home address to communicate with the mobile node, independently of the mobile node's actual location.

While the approach is simple and effective, it has two major drawbacks: it requires changes to the mobile node and it introduces routing inefficiencies due to triangular routing. Because Mobile IP relies on signaling messages between the mobile node and the foreign or home agent to update the binding of its current address, a mobile node needs to implement a corresponding Mobile IP stack. This is a considerable effort and raises deployment issues especially in heterogeneous environments. Triangular routing is caused by Mobile IP since the mobile node's traffic must be routed through the home agent instead of being routed directly from the correspondent node to the mobile node. This can significantly increase the latency of a connection. Moreover, this routing inefficiencies are present even on the initial routing path before the mobile node hands over for the first time. Although Mobile IPv6 allows for route optimization between the mobile node and its corresponding nodes [64, 8] this requires changes to the IP stack of the corresponding node which seems unlikely on a global scale. Additional problems of a global mobility management protocol such as Mobile IP are potentially high update latencies, high signaling overhead over wireless links, and potential infringement of location privacy [68, 73].

Despite the drawbacks mentioned in the previous section, Mobile IP is a readily available Internet standards track protocol which is well supported by the IETF. Enhancements for Mobile IP are actively developed in the Mobility EXTensions for IPv6 working group of the IETF (mext).[2] A number of approaches exist to extend Mobile IP and alleviate the aforementioned issues.

Hierarchical Mobile IP Hierarchical Mobile IPv6 Mobility Management (HMIP) [146, 26] is designed to reduce the amount of signaling between the mobile node, its correspondent nodes, and the home agent. To this end Hierarchical Mobile IP introduces a Mobility Anchor Point (MAP) which presents an additional management entity for mobile nodes. While a mobile nodes moves within the domain of the mobility anchor point a large part of the mobility signaling can be limited to message exchanges between the mobility anchor point and the mobile node. Because a mobility anchor point is topologically closer to the mobile node than the home agent, such strategy can improve handover speed. Although Hierarchical Mobile IP can offer performance advantages over Mobile IP, the major drawbacks of Mobile IP, namely changes are required to the mobile node and the introduction of triangular routing are not solved. Additionally, it introduces complexity in terms of deployability and maintenance due to the addition of mobility anchor points.

Mobile IP Fast Handovers The goal behind Mobile IPv6 Fast Handovers (FMIP[3]) [74] is to improve handover latencies due to Mobile IPv6 procedures which can be unacceptable high for real-time traffic [75]. To this end FMIP introduces predictive and reactive handovers between the previous access router (PAR) and the new access router (NAR). The previous access router is the access router the mobile node is associated with prior to a handover, and the new access router is the access router the mobile node is associated with after the handover. Specifically, the protocol provides the means to pre-establish IP connectivity for a mobile node prior to a handover which reduces the time that a mobile node does not have a valid IP configuration immediately after a handover. Moreover, data for the mobile node is forwarded from the previous access router to the new access router as long as the home agent or a correspondent node is still routing data for the mobile node to the previous access router. Without such forwarding

[2] http://tools.ietf.org/wg/mext/

[3] Mobile IPv6 Fast Handovers is a progressive update of the RFC “Fast Handovers for Mobile IPv6” [72], hence it is commonly abbreviated as FMIP.

this data would have to be discarded at the previous access router. The efficiency of FMIP is highly dependent on the ability of the mobile node to make accurate and timely handover predictions and needs to be supported by all the involved nodes (mobile node, previous access router, and next access router). FMIP also provides a reactive mode in cases where a handover could not be predicted which also seems to be able to improve performance in terms of handover latency and packet loss. However, both modes introduce additional complexity and states in the participating nodes.

Home Agent Migration In order to reduce the topological distance between a home agent, correspondent nodes, and the mobile node the concept of home agent migration has been proposed by Wakikawa et al. [163]. The approach introduces the concept of a Global Mobile eXchange (GMX) which is essentially an overlay network of multiple distributed home agents. The home agents use the overlay network to exchange information about the mobile nodes that are managed by the individual home agents. The approach employs anycast routing (e.g. [1, 66, 15]) to direct traffic from any mobile node to the closest home agent while generic routing mechanisms are used to direct traffic from any correspondent node to their closest home agent. A home agent that intercepts traffic for a mobile node leverages the information gained through the overlay to route the traffic to the actual home agent that a mobile node is associated with. SAIL [114] extends this approach and reduces some of its overhead.

Essentially, home agent migration allows to employ multiple home agents in Mobile IP instead of just one. The home agents can be placed in locations that are (topologically) close to correspondent nodes as well as to mobile nodes, for example in Internet exchange points. Traffic will always be routed to the closest home agent and from there via the overlay network to the mobile node's actual home agent. However, the approach can only reduce routing inefficiencies if a mobile node would otherwise chose a home agent that is topologically far outside the routing path between itself and its correspondent nodes. Moreover, Dynamic Home Agent Address Discovery that is part of Mobile IPv6 [64] is also providing the means for a mobile node to discover a suitable Home Agent via anycast routing and can therefore provide the benefits of dynamically assigning a home agent that is close to a mobile node's current location. While the approach can dynamically migrate a mobile node to another home agent this would assume that the mobile node has traveled a significant (topological or geographical) distance for another home agent to be better suited. In conclusion, the approach provides a solution to dynamically assign a topologically close node to provide home agent functionality every time a mobile node moves. However, it has to be seen if in realistic scenarios a mobile node can draw benefits out of this approach over a well suited choice of its initial home agent.

Other Extensions and Enhancements As Mobile IP is a well supported IETF Internet standards track protocol numerous proposals have been made to improve its performance. S-MIP [58] proposes a combination of Hierarchical Mobile IP (see above), Mobile IP Fast Handovers (see above), and a handover algorithm based on software based movement tracking techniques ([172] prior work partly by the same authors). The approach introduces a Decision Engine entity in every local mobility domain that makes the handover decisions based on tracking reports of the mobile node that are submitted by access routers. S-MIP combines two effective methods to reduce handover times (Hierarchical Mobile IP and Mobile IP Fast Handovers) but it also introduces additional complexity into the network to predict a mobile node's movement pattern.

Cao et al. propose a mailbox based scheme which associates mobile nodes that move to a foreign network with a mailbox [25]. This mailbox can receive data for the mobile node instead of the home agent. When a mobile node moves it can decide to move its mailbox as well. As correspondent nodes sent data to the mailbox instead of the home agent, the migration of the mailbox can improve routing efficiency. Marques et al. introduce an overlay network architecture that is based on Mobile IPv6 and integrates quality-of-service and authentication, authorization, accounting, and charging control per user [91].

"Chaining" anchor points is an approach to reduce the routing inefficiencies of Mobile IP [18]. The approach uses multiple anchors that form a chain to connect the mobile node to its actual home anchor. Each anchor along the chain defines the location of the mobile node with an increasing accuracy until the attachment point of the mobile node is reached. A regional mobility management for Mobile IPv6 is proposed in [104] which introduces a regional layer of access routers that handle intra-region handovers similar to hierarchical Mobile IP.

A cross-layer approach to improving TCP performance in conjunction with Mobile IPv6 is proposed in [79]. The approach proposes to introduce a monitoring element in the Mobile IP layer that notifies the TCP layer of mobility related events. Once the TCP layer is aware of the mobility it can regulate its congestion control with regards to the mobility events. For example, the approach proposes to trigger TCP fast recovery just after a handover for any outstanding TCP segments. While cross-layer optimizations make sense with regards to mobility, their implementation is anything but trivial. Furthermore it (deliberately) breaks the layered architecture of the Internet stack.

Decentralized Mobility Management Service

Le et al. proposed a Decentralized Mobility Management Service (DMMS) architecture [80] which leverages Shim6 and dynamic DNS to allow a mobile node to use a global IP address to communicate with correspondent nodes. The approach uses mobility anchors to forward data for a mobile node's global IP address to its current point of attachment. When a mobile node roams to another point of attachment it can request data to be forwarded from its previous mobility anchor via a Shim6 context. Because mobility anchors are available in every subnetwork, the architecture does not rely on any centralized node. However, the approach requires Shim6 support from the mobility anchors and the mobile node.

TCP Proxies

MSOCKS [90] is a TCP transport layer mobility solution that is based on a proxy which is placed between a mobile node and its correspondent nodes. The proxy provides an endpoint for data streams from the mobile node and from correspondent nodes and combines them to a continuous bi-directional data stream. If a connection to the mobile node breaks because it changes its point of attachment the mobile node will re-establish the connection. Because the proxy is the actual endpoint of the connection with the correspondent host this connection is not impacted and the proxy can combine the standing connection to the correspondent node with the new connection to the mobile node. I-TCP [13] is a similar approach proposing a modified TCP version on mobile hosts and intermediate proxies called Mobility Support Routers to split TCP connections between the mobile node and the proxy and the correspondent node and the proxy. Because the approaches use a proxy no modifications are required to the correspondent node. The mobile node on the other hand must support the particular approach.

Cellular IP

Cellular IP [24, 160] is an early approach to host mobility. It proposes to separate macro-mobility management (a mobile node moves between domains) and micro-mobility management (a mobile node moves within a domain) to optimize Mobile IP operations in a localized context. The approach was developed based on the issue that second- and third-generation cellular systems were built on a connection-oriented infrastructure which raises some issues about the integration the of an IP based mobility solution such as Mobile IP. Macro-mobility in Cellular IP is handled by Mobile IP while micro-mobility is handled locally by the cellular network. This architecture provides the benefit of a hierarchical mobility solution, namely shorter signaling paths and localized mobility management. Furthermore it delegates cellular-network specific tasks that have no counterpart in IP to the cellular-network components. An example would be paging which is used to query the status of a mobile node in a connection-oriented environment. Based on the convergence of cellular networks towards IP based platforms, the basic deployment assumptions of Cellular IP are no longer valid and IP based approaches such as Hierarchical Mobile IP seem better suited for deployment.

HAWAII

Similar to Cellular IP, HAWAII [128] aims at providing micro-mobility within a domain while using Mobile IP to provide macro-mobility. Intra-domain micro-mobility is provided by installing host-based forwarding entries in specific routers along the path towards the mobile node. The system adjusts the routes when the mobile node moves between access routers in the domain. This eliminates the need for a mobile node to rely on signaling to its home anchor which can potentially have large delays. The approach does require the mobile node to send HAWAII specific messages to trigger the path setup. Therefore a mobile node must support Mobile IP and additionally the HAWAII protocol.

2.2.2. Locator/ID Separation Approaches

This section will discuss approaches that are based on a Locator/ID separation concept.

Internet Indirection Infrastructure

Stoica et al. propose on overlay based approach that provides a communication paradigm based on rendezvous-points [150]. The so-called Internet Indirection Infrastructure (i3) decouples sending and receiving of data by introducing indirection points (i3 nodes) that store triggers and forward data between i3 nodes and to end-hosts. A trigger is essentially a subscription for data with a specific identifier. When a node registers a trigger the responsible i3 node will forward any data for that particular identifier to the node's current address using IP. Using this paradigm a mobile node can receive data under a constant identifier since it just needs to update its trigger to receive date under its new address. Some proposals have been made to improve i3 with regards to mobility (e.g. [173, 55]). While the approach is build on top of IP it requires support from the mobile node and the correspondent node on the application layer. Although the approach is called Internet Indirection Infrastructure, it is classified as an Locator/ID separation approach corresponding to the considerations made in Section 2.1 because it introduces a clear segregation between locators (i.e. normal IP addresses) and identifiers (i.e. i3 triggers).

Identifier-Locator Network Protocol

The Identifier-Locator Network Protocol (ILNP) [9, 10, 130, 11] is designed to work on an existing IPv6 infrastructure. It uses the 64 high-order bits of an IPv6 address as the locator and the 64 low-order bits as the identifier. It still uses the existing DNS system to resolve the mappings between the identifier and the locator. The distinct advantage of this approach is that an existing IPv6 transit network does not need to be modified. However, end-hosts still need a modified IP/ILNP stack.

Host Identity Protocol

The Host Identity Protocol (HIP) [102, 103, 54] separates the identifier and locator roles of IP addresses by introducing a new Host Identity namespace. Using host identities from this namespace instead of IP addresses decouples the transport layer from the IP layer, making it agnostic to IP address changes. Because HIP adds a layer of indirection and a separate namespace it can inherently provide mobility and multi-homing support (e.g. [111, 109]). The mobility and multihoming extensions to HIP [110] introduce a general ”LOCATOR” parameter for HIP messages to allow a host to dynamically change the IP address that it uses to receive packets while keeping the same HIP identity. This allows a node to change the IP addresses of communication endpoints without breaking connections. HIP requires end-host support.

2.2.3. End-to-End Mobility

Approaches to end-to-end mobility provide mobility support with using direct signaling between end hosts. Therefore they require modifications to mobile nodes as well as correspondent nodes. However, they typically do not require support from any intermediate node except normal traffic forwarding.

SIP Mobility

The Session Initiation Protocol (SIP) [137] is an application-layer control protocol. It is intended for establishing, modifying, and terminating multimedia sessions like Internet telephone calls, multimedia distribution, or multimedia conferences. Although SIP itself supports personal mobility (i.e. the location of the end user when starting or receiving calls does not matter) it does not support IP mobility. However, a mobile host can use SIP INVITE messages to update its location with correspondent hosts after it has changed its point of attachment [164]. Such application specific approach can reduce handover delay as no intermediate node is involved. The drawback is that application specific approaches require a high amount of implementation work.

SCTP Mobility

The Stream Control Transmission Protocol (SCTP) [148] is a reliable transport protocol designed to overcome limitations of TCP. SCTP supports multi-homing which can be extended to provide mobility support. The Dynamic Address Reconfiguration extension to SCTP [149] allows the dynamic allocation and de-allocation of IP addresses to the endpoints of a SCTP connection. This enables mobility support for SCTP (e.g. [168, 132] as a host can change the IP addresses of the endpoints of a connection (e.g. when a mobile node changes its point of attachment) without breaking the connection. However, for SCTP mobility to be available, SCTP including the Dynamic Address Reconfiguration extension needs to be employed as the transport protocol between the mobile node and the corresponding node. Subsequently, changes to both nodes are required.

TCP Mobility

TCP Migrate [145] proposes a new TCP option that allows a host to migrate an established TCP connection across a change in IP addresses. TCP-R (TCP Redirection) [47] is another TCP extension which can maintain an active TCP connection across a change in IP addresses. Similar to TCP Migrate TCP-R uses TCP options to implement the signaling of an IP address change between end hosts. Both approaches work end-to-end which means they need to be supported by the mobile node as well as the correspondent node. As they extend TCP, the approaches work on top of the existing IP infrastructure although they are naturally limited to the TCP transport protocol.

Shim6

The Shim6 protocol [112] introduces a layer 3 shim to provide multihoming and load-sharing support below transport protocols. A host can employ Shim6 to achieve redundancy or load balancing properties for its connections. While Shim6 itself does not aim to solve the problem of host mobility itself, it can be used to dynamically switch between different locator pairs[4] while a host is moving. This can be used to achieve basic mobility

[4] A locator in Shim6 terminology is a topological name for an interface or a set of interfaces.

support (e.g. [80, 136, 37, 124]). Shim6 does require changes to the mobile node as well as to corresponding nodes, or a number of intermediate nodes (e.g. proxies).

Back to My Mac

Back to My Mac (BTMM) [30] is a commercial approach to mobility support that has been deployed by Apple Inc. since 2007 with the Mac OS Leopard release. The approach requires a mobile node to dynamically update its current location in the DNS every time it changes location. Nodes that want to communicate with the mobile node establish a long-lived query to the DNS server that hosts that mobile node's location data. Using such a query the DNS server will immediately send an update to the client node if the DNS entry changes. Subsequently the client can re-establish a connection with the mobile node using its new address. While being successfully deployed on a large scale the approach uses a break and re-establish pattern that does induce long latencies during handovers. It is not clear that the level of mobility support is sufficient for constant data communications such as Voice-over-IP. The approach requires support from mobile nodes as well as correspondent nodes.

2.3. Network Based Mobility

In contrast to host based mobility approaches, network based mobility approaches do not include the mobile node in any mobility management related operations. This allows for a greater deployment flexibility as no changes are required to the mobile node. This section will discuss approaches to network based mobility.

2.3.1. Indirection Approaches

Proxy Mobile IP

Proxy Mobile IP (PMIP) is a solution for network based mobility management that is specified by the IETF for IPv6 [51] and IPv4 [162] networks. It extends Mobile IP signaling messages to allow network nodes to provide mobility management without the involvement of the mobile node. Because the network manages mobility on behalf of the mobile node, Proxy Mobile IP does not require any stack changes in the mobile node. Furthermore, as the mobile node is not involved in mobility operations there is no mobility-related signaling or tunneling that involves the interface which connects the mobile node to it's access network. In most cases this will be a wireless interface which tends to have a high power demand as well as a connection which is prone to loss. The

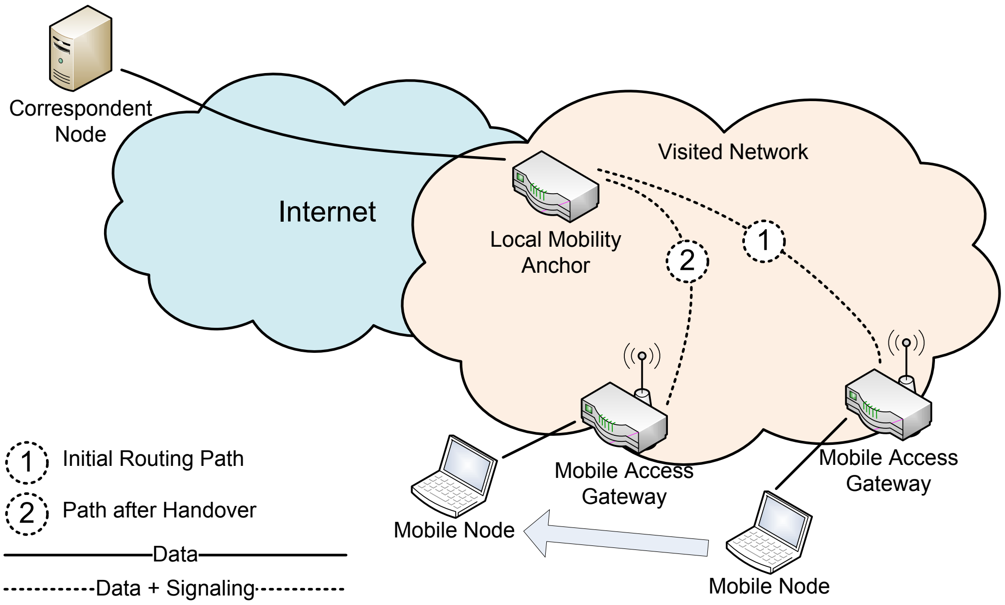


Figure 2.2.: Overview of Proxy Mobile IP

involved nodes in providing Proxy Mobile IP based mobility support to a mobile node is the Mobile Access Gateway (MAG) which connects the mobile node to the network and the Local Mobility Anchor (LMA) which provides the mobility anchor point.

Figure 2.2 shows an overview of the approach. The local mobility anchor provides a centralized anchor or indirection point for a Proxy Mobile IP Domain. It manages a mobile node's data forwarding and coordinates the mobility support with the mobile access gateways. Every time a mobile node switches between mobile access gateways the local mobility anchor provides the next mobile access gateway with an identical IP configuration for the mobile node and forwards the mobile node's data to the new point of attachment. Because every mobile access gateway provides the same set of IP configuration data to the mobile node (e.g. via DHCP) the mobility is hidden from the mobile node.

Proxy Mobile IP provides a number of advantages over Mobile IP. First, it does not require any changes to the mobile node. Second, the triangular routing problem is largely eliminated because of a natural proximity between the mobility anchor point and the mobile node. However, Proxy Mobile IP can only provide mobility support within the network domain of the local mobility anchor. That is, if the mobile node leaves this domain the mobility support breaks. Moreover, when a mobile node attaches to a new mobile access gateway, the mobile access gateway needs to signal the local mobility anchor to setup the corresponding data forwarding on behalf of the mobile node and to provide the configuration for the mobile node. If the mobile access gateway and the local mobility anchor are topologically far apart this operation can introduce noticeable handover latency for the mobile node. Enhancements for Proxy Mobile IP are actively developed in the Network-Based Mobility Extensions working group of the IETF (netext).[5] Similar to Mobile IP a number of extensions to Proxy Mobile IP have been proposed.

Simultaneous Bindings Proxy Mobile IPv6 with Simultaneous Bindings (SPMIP) [16] is an approach to reduce handover latency in Proxy Mobile IP by introducing a link layer trigger mechanism that induces a mobile node to pre-establish connectivity with the next mobile access gateway before it performs a handover. When a mobile node pre-establishes a connection, the next mobile access gateway can already start the binding procedure with the local mobility anchor which reduces the handover time once the mobile node does perform the handover. Furthermore, the approach allows the local mobility anchor to duplicate data for the mobile node and send it to the current and the next mobile access gateway simultaneously which is intended to reduce packet loss during and right after a handover. Overall, the approach is similar to Fast Handovers in Mobile IP (see Section 2.2.1) and as such relies on accurate and timely predictions of handovers that may or may not be possible in a given scenario. Another drawback is that the approach relies on two specific link layer trigger messages: One trigger message has to be processed by a mobile node and initiates a preemptive handover. The other trigger message has to be tripped by the mobile node and concludes a handover. The trigger mechanisms are not further addressed by the paper and while such mechanisms may be implemented using, for example, IEEE 802.21 Media Independent Handover (MIH) services [156] they require changes to the mobile node. Also, the simultaneous forwarding of the mobile node's traffic to two Media Access Gateways puts a strain on the network resources and requires buffering on part of the next mobile access gateway. Furthermore, it can lead to duplicate packets received by the mobile node after a handover since the next mobile access gateway does not have detailed knowledge which packets the mobile node already received via the previous gateway.

Fast Proxy Mobile IPv6 Another approach to reduce handover delays in Proxy Mobile IP is Fast Proxy Mobile IPv6 (FPMIP) [57]. It directly translates the concept behind Mobile IP Fast Handovers (see Section 2.2.1) to Proxy Mobile IP. A similar approach is

[5] http://tools.ietf.org/wg/netext/

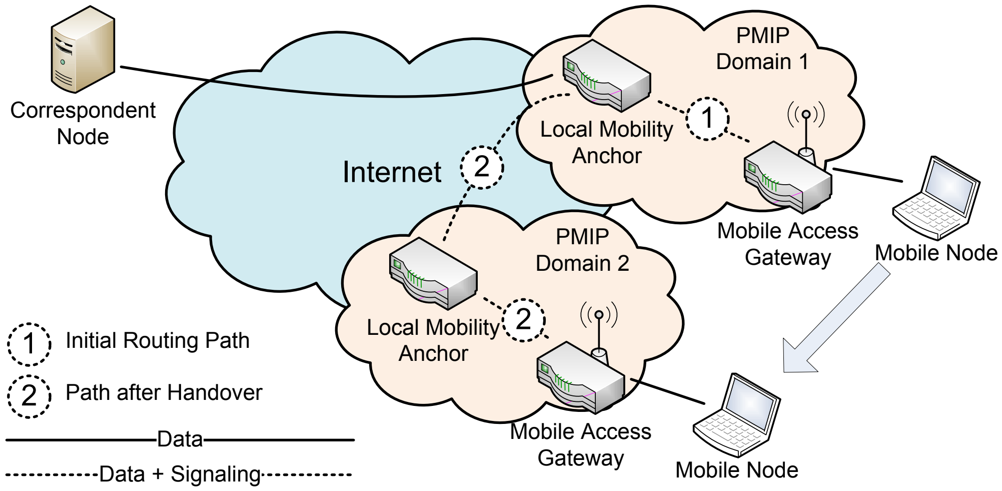


Figure 2.3.: Architecture of I-PMIP.

proposed in [81]. If a handover can be predicted prior to the mobile node performing it, the new mobile access gateway will perform any mobility related signaling and setup the data forwarding with the local mobility anchor in advance. Furthermore, downlink traffic arriving at the old mobile access gateway after the handover will be forwarded to the new mobile access gateway. Similar to Proxy Mobile IPv6 with Simultaneous Bindings (see above) the approach leverages prior knowledge of a handover to reduce signaling times when the handover actually happens. Both approaches share the same drawback that they rely on an accurate prediction of handover which, depending on the mechanisms used, might require an involvement of the mobile node.

Inter-domain extension to Proxy Mobile IP A major drawback of Proxy Mobile IP is that if a mobile user leaves the coverage of a local mobility domain, the mobility support breaks. I-PMIP [107] is an inter-domain mobility extension for Proxy Mobile IP. The approach combines inter-domain mobility with the benefits from Proxy Mobile IP based local mobility solutions. It employs a decentralized architecture as shown in Figure 2.3. As long as a mobile node moves within its initial Proxy Mobile IP domain, mobility is provided by the local Proxy Mobile IP mobility solution. When a mobile node moves between Proxy Mobile IP domains its traffic is forwarded to the new domain's local mobility anchor. Within the new domain the mobile node is also managed by the corresponding local Proxy Mobile IP mobility solution.

In summary, I-PMIP allows to interconnect multiple Proxy Mobile IP enabled mobility domains to provide a continuous mobility support for a mobile node if it moves between those domains. The I-PMIP architecture provides an implicit near-optimal placement of the anchor point for a mobile node and has a distributed character which does not introduce additional bottlenecks or single point of failures. Furthermore, I-PMIP extends the features of a network based mobility solution from Proxy Mobile IP, in particular it does not require any changes or configuration of the end host or a dedicated mobility provider for the mobile user.

I-PMIP can be characterized as a hybrid solution between centralized and decentralized mobility management as it allows to interconnect multiple (centralized) Proxy Mobile IP domains. However, it is still based on Proxy Mobile IP's centralized architecture that is build around local mobility anchors. Furthermore, handovers require information lookups of data that is potentially in another domain which can induce noticeable handover delay.

Base Station Router

The UMTS base station router (BSR) [17] is an access router that provides access to cellular network services within a limited coverage area. It is connected to the cellular network backbone via an IP network and, therefore, can be deployed everywhere where IP access is available. This includes private homes or office buildings where a cellular operator would otherwise have no access. Because the base station router can be deployed in access networks outside of the cellular network operator's own infrastructure, it is challenging to provide mobility support for them. Although the base station routers are providing network based mobility support they rely on a home agent within the cellular operator's network to provide inter base station router mobility. For inter base station router communication this introduces the same routing inefficiencies that Mobile IP has. The base station router concept is reflected in the upcoming femtocell technology.[6]

Mobile Internetworking

IP-based Protocols for Mobile Internetworking [61] is an early approach to provide network supported mobility. It introduces so-called Mobile Support Stations (MSS) in the access network that exchange data about mobiles nodes that are currently active within the cell of a MSS. A mobile node can retain the same IP address even if it switches cells. The corresponding MSS will route data for the mobile node to its current location based on the data it exchanges with the other MSS. The MSS effectively form an overlay network which provides indirection based on the fixed IP address of the mobile node.

[6] http://www.femtoforum.org/femto/aboutfemtocells.php

While the approach is essentially network based as the network provides the location management of the mobile node it requires a slight modification to the mobile node to change the routing table entry for the default gateway when it switches cells. However, as the modification is not within the networking stack but merely a system program the approach is considered somewhat of a hybrid.

Terminal Independent Mobility for IP

Terminal Independent Mobility for IP (TIMIP) [49] essentially integrates Cellular IP and HAWAII (see Section 2.2.1) with the added benefit of support of legacy mobile nodes. Like Cellular IP, TIMIP refreshes routing paths when a mobile node is detected to be idle. Like HAWAII, TIMIP uses specific routes to provide micro-mobility within a domain between the old access router and the new access router. Additionally the approach provides a node that provides Mobile IP proxy services for nodes that do not support Mobile IP. For such mobile nodes a TIMIP domain can register with a home agent on behalf of the mobile node and therefore allow a mobile node to receive data using a home address. However, when the mobile node switches TIMIP domains the mobility support breaks as the mobile node requires a different default gateway. If the mobile node does support Mobile IP, TIMIP falls back to Mobile IP for macro-mobility like Cellular IP and HAWAII.

2.3.2. Routing Based approaches

The Internet routing fabric is responsible for forwarding data from its sender to its receiver. As such it seems a natural starting point for mobility support as routing based approaches do not need to introduce an indirection layer that provides a mapping between the location and the identifier of a mobile node.

Global IP Network Mobility using Border Gateway Protocol (BGP)

Connexion by Boing (CBB) [41] is a commercial service developed by Boeing to support network mobility in their aircrafts. Each aircraft is assigned its own /24 network which is used to apply one-to-one network address translation to private IP addresses assigned to mobile nodes within the aircraft. The /24 address block assigned to an aircraft is announced via the Border Gateway Protocol (BGP) [131] by the ground station currently serving the aircraft with connectivity to the Internet. While traveling the aircraft accesses the Internet through a satellite link to a ground station. A handover is shown in Figure 2.4. When an aircraft moves from the coverage area of one ground station ("A") to the coverage area of another ground station ("B"), the IP address block is withdrawn

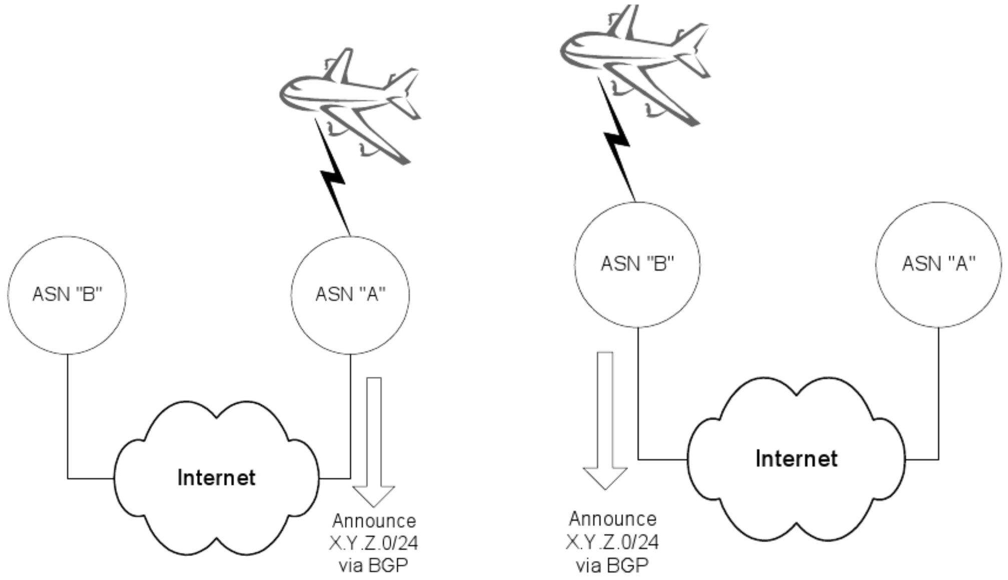


Figure 2.4.: BGP announcements in Connexion (source: [41]). Left: Before the handover. Right: After the handover.

from the BGP table in ground station “A” and is announced by the next ground station “B”.

The approach manages to reduce packet round-trip-times when compared to approaches that use a static home anchor which might be situated on a different continent. However, Connexion exploits a number of attributes that are closely associated with the deployment scenario. For instance, the link layer (Satcom) handover time is already substantial (“less than a minute”) which prevents seamless mobility and allows for generous handover latency on the mobility layer. Furthermore, the handovers are comparatively rare (handover interval “usually 4-8 hours for an average flight”) which does not raise scalability issues and alleviates the user impairment during handovers. Overall, using BGP to realize mobility support seems to be feasible in very limited deployments at best. There are major scalability problems to be expected when BGP is used for mobility support on a large scale.

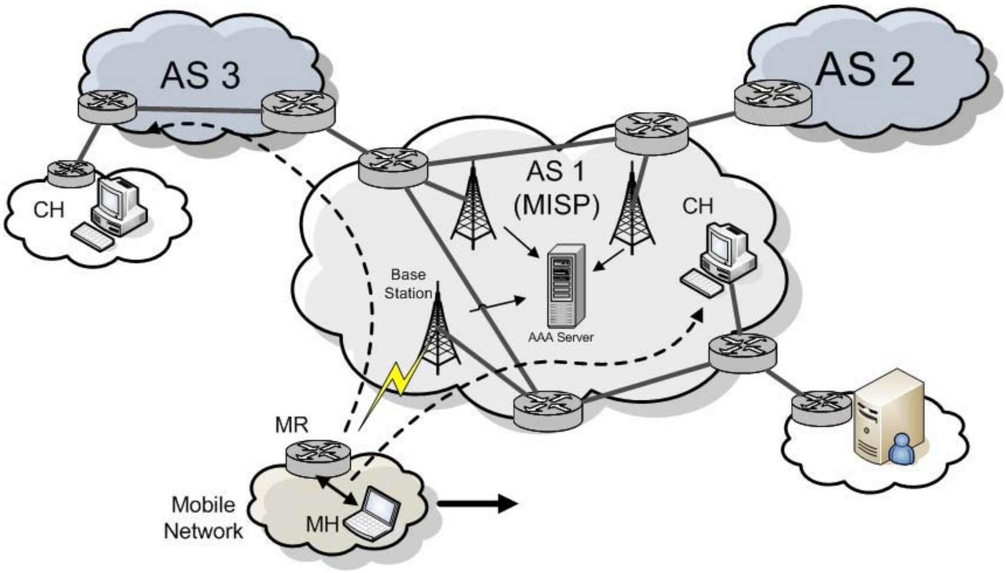


Figure 2.5.: WINMO's basic architecture (source: [59])

Wide-Area IP Network Mobility

Wide-Area IP Network Mobility (WINMO) [59] is another approach intended for network mobility and is also based on BGP. Figure 2.5 shows an overview of WINMO's basic architecture. The global network architecture depicts the current Internet structure with a set of Autonomous Systems (ASs) that are interconnected by BGP gateway routers. A mobile network has a fixed network prefix allocated by its home mobility provider; hosts within the mobile network are assigned IP addresses within this prefix. There is a mobility router (MR) inside each mobile network which connects the network to the AS that it is currently attached to.

The approach is designed with the global architecture of the Internet in mind and uses existing mechanisms (i.e. BGP). The approach tries to minimize the introduction of additional states in routers and aims at incrementally deployability. While the approach does not require modifications in the end hosts, a significant improvement of the approach does require end-host support. Namely, a secure token which is mirrored back by a corresponding host to allow routers to avoid triangular routing. In fact, it can be argued, that the results of the inter-domain evaluation of the WINMO approach are solely based on this improvement which makes the approach a lot weaker. Without this improvement WINMO suffers from triangular routing within the AS. Besides the deployability issues regarding the secure token, the decryption of the token in AS border routers in order to access the encapsulated mobility states may create high load on those systems. Another drawback of the approach is that it requires the modification of major router systems, namely the aggregation routers and, in case the secure token mechanic is deployed, the AS border routers. While it may be true that the modifications can be implemented in software only, the focus of the approach on tier-1 provider means that the aggregation routers will be among the most critical systems in the Internet. This introduces a major deployability issue. Furthermore, the additional states and operations, such as the tunnel mappings, that are introduced into these high-level systems are not explored in the original paper and may be significant.

2.4. Clean-Slate Approaches

Clean slate approaches to mobility do not consider any compatibility to existing protocols or technologies. The focus of such approaches is usually long term without regards to deployability in the current Internet environment. As such they can provide valuable insights and solutions that are not commonly considered because they are not practically applicable in current scenarios. Clean slate approaches can provide valuable and valid scientific contributions.

2.4.1. Routing Schemes

Särelä et al. propose a fast inter-domain mobility signaling protocol that is based on in-packet Bloom filters [141]. Intermediate routers collect bi-directional Bloom filters on mobility signaling messages which essentially provide a basis for source-routing of subsequent data. As subsequent data is routed based on the Bloom filters and not based on the IP address of the mobile node, the mobile node can change its point of attachment without breaking any ongoing connections.

Weak State Routing [4] is a routing scheme intended for large-scale highly dynamic networks. It uses random directional walks that are influenced by state information in intermediate nodes. Nodes only have a probable location of a destination. This weak state allows the approach to aggregate information about multiple remote locations in a geographic region. The approach was developed with large and highly dynamic ad-hoc networks in mind. Because of its probabilistic character it cannot guarantee packet delivery.

Routing on Flat Labels (ROFL) [22] proposes a routing algorithm that routes directly on host identities, thus eliminating any need for locations to be included in network-layer protocols. As the host identity is independent of any location and can stay fixed for the lifetime of a host such a routing algorithm could overcome the basic problem of changing identifiers that mobility poses to the current routing architecture. The evaluations show an average inter-domain routing stretch of 2.5 which means that the routing overhead would more than double for any host (not just mobile hosts). However, the approach can establish a basic feasibility for routing directly on host identities.

2.4.2. Locator/ID Separation

Virtual Id Routing (VIR) [85] introduces a new node identifier space which is mapped to the underlying network topology. As such it can be classified as a Locator/ID separation approach and can provide basic mobility support based on stable node identifiers. In contrast to similar schemes (e.g. [21]) the approach inserts an intermediate layer which maps the underlying topology to the node id space in a way that the node id space resembles the network topology.

"Mobile Party" [140] introduces temporary addresses that are assigned based on locality and a global identifier for each node. The temporary address changes when a node moves but its identifier stays the same. While the approach follows the general formula for Locator/ID separation (i.e. one fixed address, one variable address, and a mapping service) its implementation limits it to local (mesh) networks. It is intended as a stand-alone routing and layer 2 protocol and does not work on existing IP platforms.

Another approach is the Protocol for Evolutionary Addressing (PEA) Framework [53] which proposes a self-organizing approach to addressing and routing. In PEA nodes organize themselves as hierarchical clusters. When a node joins a network it assumes a local address within the cluster and is reachable via a global address that consists of its local address as well as all the local addresses of intermediary nodes. Since a node's global address changes when it moves, the approach requires external mechanisms to avoid or minimize disruptions at the higher layers (e.g. SCTP).

2.4.3. Data-Oriented Networking

A Data-Oriented Networking Architecture (DONA), also known as "Named Data Networking" (NDN) or "Content-Centric-Networking" (CCN) is a new communication paradigm that proposes to change the Internet architecture so that protocols communicate in a data or content centric manner rather than in a host centric manner as they do today (e.g. [76, 62, 94]). Mobility can be inherently support to a degree since data is no longer routed based on host identifiers or locations. Instead it is routed based on its name or description towards hosts that have subscribed or expressed an interest in this data. If a host moves it can re-subscribe or express another interest and the data will

Table 2.1.: Overview of *host based* mobility approaches

Approach	**Concept**	**Layer**	**Support Req.**
MIP [116, 64]	Indirection	Network	MN
HMIP [146, 26]	Indirection	Network	MN
FMIP [74]	Indirection	Network	MN
HA Migration [163, 114]	Indirection	Network	MN
MIP related [58, 172, 25, 91, 18, 104, 79]	Indirection	Network	MN
DMMS [80]	Indirection	Network	MN
MSOCKS [90]	Indirection	Transport (TCP)	MN
I-TCP [13]	Indirection	Transport (TCP)	MN
Cellular IP [24, 160]	Indirection	Network	MN
HAWAII [128]	Indirection	Network	MN
i3 [150, 173, 55]	Loc/ID sep.	Transport/App	MN, CN
ILNP [9, 10, 130, 11]	Loc/ID sep.	Transport/App	MN, CN
HIP [102, 103, 54, 111, 109, 110]	Loc/ID sep.	Intermediate	MN, CN
SIP [137, 164]	End-to-End	Application	MN, CN
SCTP [148, 149, 168, 132]	End-to-End	Transport	MN, CN
TCP Mobility [145, 47]	End-to-End	Transport	MN, CN
Shim6 [112, 136, 37, 124]	End-to-End	Shim	MN, CN
BTMM [30]	End-to-End	Application	MN, CN

be routed to its new location. However, such an architecture requires the mobile node to repeat its request for data after a handover which might introduce noticeable latency. Moreover, while such an architecture might be able to solve mobility on the network layer it cannot mask mobility on higher layers (cf. [76]).

2.5. Conclusion

Mobility support can be broadly classified depending on where it is implemented: in the end-hosts, in the network, or in the routing architecture. Table 2.1 shows an overview

Table 2.2.: Overview of *network based* mobility approaches

Approach	**Concept**	**Layer**	**Support Required**
PMIP [51, 162]	Indirection	Network	Access Network
SPMIP [16]	Indirection	Network	Access Network
FPMIP [57, 81]	Indirection	Network	Access Network
I-PMIP [107]	Indirection	Network	Access Network
BSR [17]	Indirection	Network	Access Network
Mobile Internetworking [61]	Routing/Indirection	Network	Acc. Network, MN
TIMIP [49]	Indirection	Network	Acc. Network, (MN)
CBB [41]	Routing	Network	BGP
WINMO [59]	Routing	Network	BGP

of the presented host based mobility solutions. They require support from the mobile node (MN) itself and provide mobility on the network layer (e.g. Mobile IP), on the transport layer (e.g. SCTP, MSOCKS) or on the application layer (e.g. SIP, BTMM). They can be further classified by the basic concept that they are based on. Indirection based approaches do not require support from the correspondent node (CN) as mobility is supported by an indirection that is typically implicitly provided by an overlay. Locator/ID separation approaches (Loc/ID sep.) as well as end-to-end approach require support from the mobile node as well as the correspondent node and therefore have even higher deployment barriers than indirection based approaches. Mobile IP is a very prominent approach for host mobility and is based on an indirection layer that is provided by a minimal overlay which is only composed of the mobile node and its home anchor. Because the home anchor is a centralized anchor point and topologically fixed, the approach suffers from strong routing inefficiencies. Enhancements such as home agent migration can improve Mobile IP's routing performance for the cost of additional deployment complexity. Similarly, other overlay based approaches have a high deployment complexity as they require a more extensive overlay. Moreover, as they include the mobile node, signaling messages are exchanged via a wireless interface. This makes them prone to loss which can negatively impact handover performance. Because every host based mobility solution requires changes to the mobile node they are not well suited for operator driven deployments that cannot count on the involvement of the mobile device's user or administrator.

Table 2.2 shows an overview of the presented network based mobility solutions. They

Table 2.3.: Overview of *clean slate* approaches to mobility

Approach	**Concept**	**Layer**	**Support Required**
In-packet Bloom filters [141]	Routing	Network	MN, CN, Network
Weak State Routing [4]	Routing	Network	MN, CN, Network
ROFL [22]	Routing	Network	MN, CN, Network
VIR [85]	Loc/ID sep.	Network	MN, CN, Network
Mobile Party [140]	Loc/ID sep.	Network	MN, CN, Network
PEA [53]	Loc/ID sep.	Network	MN, CN, Network
DONA/NDN/CCN [76, 62, 94]	Named Data	Network	MN, CN, Network

do not require any end host support but support mobility from within the access network or with support from the Border Gateway Protocol (BGP) [131] depending whether they are based on an indirection concept (e.g. PMIP, BSR) or routing based (CBB, WINMO). Routing based approaches are either hard to deploy as they require support from intermediate networks or the network of the correspondent node. Furthermore they have a high overhead and limited efficiency especially when they used to provide host mobility. Indirection based approaches, predominantly Mobile IP, provide a good deployability as they only require changes to the access network which is serving the mobile host. However, they commonly rely on centralized indirection points (e.g. the local mobility anchor in Proxy Mobile IP). This limits scalability and forms single points of failure for the entire network, especially when the network is multi-homed. Furthermore any signaling that includes centralized components induces large delays, especially in decentralized deployments. This negatively impacts handover times.

Table 2.3 shows an overview of the presented clean slate approaches. While they are conclusive and can potentially provide efficient mobility support they are developed without any deployability considerations for the current Internet environment. As such they commonly have very high deployment barriers and cannot be considered for short term solutions. For example all of the presented approaches require support from the mobile node, the correspondent node, and intermediate networks.

In conclusion, host based approaches are not suited for operator driven deployments as they require modifications to end hosts. Efficient routing based approaches and clean slate approaches have a very high deployment barrier as they require modifications to intermediate nodes that are outside the domain of the access provider who wants to provide mobility support. Moreover, because routing based approaches inform the whole network about changes to a mobile node's location they do not scale well in large networks or for a large number of mobile nodes. Therefore from a scalability point of view approaches to mobility that are designed outside the global routing system are preferable (cf. [170]). Clean slate approaches additionally require changes to the mobile node. Network based mobility approaches that provide mobility based on an indirection layer have a good deployability, especially in operator driven scenarios as they do not support from nodes outside the supported network. However, current approaches, predominantly Proxy Mobile IP employ centralized components that limit scalability and induce large delays in distributed scenarios. The remainder of this thesis will present a framework for mobility management that is well suited for distributed and fragmented environments because, contrary to Proxy Mobile IP, it does not employ any centralized components.

3. A Framework for Decentralized Mobility Management

As motivated in Chapter 1 this thesis aims at exploring an approach to decentralized mobility management. In contrast to more traditional approaches, like Mobile IP, a decentralized mobility management framework does not rely on any centralized components, such as dedicated mobility anchors, as they provide bottlenecks and make scalability more complex. Furthermore, dedicated mobility anchors negatively impact routing performance unless they are explicitly placed in the only possible data path for a mobile node to reach any other node. However, in a hierarchical network such as the Internet, by design, the only node that is in-path for any possible connection is the access router a mobile node is connected to. Based on this observation this thesis proposes a decentralized mobility management framework that is being implemented in its core by access routers and does not rely on any other network nodes. An early model of this framework has been published in [106]. This chapter introduces the generic framework that the decentralized mobility management solution is built on.

3.1. Design Basis

The design of the framework is based on extensive preliminary considerations about deployment scenarios, design goals, and underlying assumptions. This section will introduce these preliminary considerations as well as the design choices that are based on them.

3.1.1. Deployment Scenarios

Any respective deployment scenario has its individual requirements and conditions. Therefore, before design goals can be set and underlying assumptions can be made, the possible deployment scenarios for a decentralized mobility management solution have to be discussed. This section will highlight the particular deployment potentials in different deployment scenarios with regards to the individual demands of users, network operators, and on-site access providers.

User's Perspective The proliferation of access technologies and powerful mobile devices does not only increase the amount of data and services that is accessed, it also provides more access options for the user via various technologies. Each technology has been designed for its specific deployment scenarios. To a certain degree they complement each other rather than compete with each other. For example, the range, bandwidth, and cost of a technology are trade-offs that result in different technologies for different ranges and bandwidth requirements. here are high bandwidth, low range, and medium cost technologies (e.g. IEEE 802.11) that complement low to medium bandwidth, high range, and medium to high cost technologies (e.g. 3G/UMTS). A user wants to be able to opportunistically use the "best" technology and network for his particular scenario. Therefore, it is highly desirable for mobile users to be able to handover between different technologies and networks even during an application session. From a user's perspective a mobility solution needs to support this kind of seamless handover across different access technologies and, more importantly, between different access networks.

Network Operator's Perspective The business model of large network operators such as mobile phone network operators is to provide user's Internet access, usually within a large and continuous area. Depending on the size of the operator the covered region can be municipal, metropolitan, regional, national, or even international. Such operators operate multiple access points and users commonly roam between them. This makes mobility support an essential service for large network operators. Localized mobility solutions such as Proxy Mobile IP are designed to provided mobility in such scenarios. They use centralized anchor points that provide mobility support for a domain. It may seem feasible to position such centralized mobility anchor points in a way that round trip times for queries involving them are low and that places them on the default routing path for mobile nodes to reduce routing overhead. However when a domain becomes larger and larger, gains multiple upstream connections (multi-homing), or becomes fragmented due to a federation with another operator or because of satellite stations such as femtocells any centralized component will negatively impact routing paths and increase latencies for any queries that involves it. Therefore, from an operator's perspective a scalable mobility solution needs to support large, multi-homed, or fragmented networks.

On-Site Access Provider's Perspective With the wide availability of low-cost and license-free access technologies such as 802.11, wireless hot-spots are a common occurrence. They complement the wide area wireless networks such as cellular networks with cheap and ample bandwidth in a small coverage area. Such hot-spots can be found in any number of places such as private homes, hotels, restaurants, public buildings, libraries, companies or schools. Besides large network operators, there are small operators such as businesses that provide on-site network access as a supplemental service to their customers. Examples for such on-site access providers are coffee-shops, restaurants, or libraries. In contrast to the large network operators, on-site access providers only operate a single access point or a very small number of access points. Because of this small number of access points and the fact that network access is offered as complimentary service, mobility support is usually not specifically considered or is supported on the link layer. Therefore, from an on-site access provider's perspective a mobility solution should have low deployment barriers and ideally not require any other network equipment than the actual wireless access routers.

Inter-Domain Mobility Management

In deployment scenarios where seamless mobility support is to be provided across multiple administrative domains a decentralized mobility management can be more easily integrated due to the lack of centralized components. This pertains to management components as well as mobility anchors. Management components, for example user and mobile node databases or authentication, authorization and accounting servers, contain sensitive data of a specific administrative domain. In a distributed mobility management solutions such services can be kept separated even when mobility support is provided for multiple domains under different administrative authorities. Mobility anchors are nodes that forward traffic for mobile nodes to their current point of attachment (e.g. the home agent in Mobile IP and the local mobility anchor in Proxy Mobile IP). In a centralized mobility management solution they are commonly positioned in a topologically favorable position to minimize the impact that the traffic redirection has on the routing efficiency and with it on the traffic's delay. For example as a default gateway for the network segment that mobile nodes connect to or as an edge router of the domain that mobility support is provided for. However, in multi-domain scenarios such a placement is hard to find as there might be no common intersection of routing paths for mobile nodes that are connected to either of the domains. A decentralized mobility management solution on the other hand employs multiple mobility anchors that can be placed individually in either of the domains, edge, or access networks.

Large Networks

A decentralized mobility management solution can provide two substantial benefits to large networks: scalability and flexibility. Instead of employing a single centralized infrastructure, multiple decentralized components can be deployed in different areas of the network. This provides scalability benefits that are inherent with decentralized architectures and also allows for more flexibility and control over subsets of the network. For example in a large multi-homed network, multiple mobility anchors can be deployed close to each individual network edge instead of having a single mobility anchor in a centralized position.

Small On-Site Access Providers

A decentralized mobility management solution can also benefit small networks, for example on-site access providers such as small businesses that provide mobile network access in a very localized environment or just complimentary to their main business. In such scenarios a deployment of a complex and costly centralized infrastructure for mobility management might not be feasible. A decentralized mobility management solution on the other hand might be easier to deploy. For example the necessary services can be integrated via software in common access points that have to be deployed in any case. Providing mobility support can then be provided by simply configuring the access points to act as mobile access routers.

Off-Loading of Mobile Nodes

With increasing wireless network access by mobile users it has become more attractive for network operators to off-load users from expensive radio technologies to cheaper alternatives when the mobile user is in a corresponding coverage area. One example is off-loading mobile users from 3G cellular networks to IEEE 802.11 networks. Another example is the emerging femtocell technology (see Section 2.3.1) that enables mobile users to operate their own very small radio cell. Such a femtocell is connected to the radio operators core network over the user's broadband Internet connection. Similar, an IEEE 802.11 access point that a user is off-loaded to might be connected via a third party to the operators core network. In both cases, the new point of attachment might be topologically far away from the core network. Any centralized mobility solution will incur a delay in operations that is much larger than the delay of operations with network entities that are part of the operators own access network. In such a scenario, a decentralized mobility management solution that is designed with large delays in mind might yield better performance during handovers and normal forwarding operations.

3.1.2. Design Goals

After the envisioned deployment scenarios for the mobility management solution have been introduced in the previous section, this section will set design goals that benefit these deployment scenarios.

Minimize Bottlenecks and Single Points of Failure

Centralized components in any network present bottlenecks and single points of failure. Furthermore they can impact routing efficiency if they are positioned outside a straight routing path between two corresponding nodes and yet require traffic to be routed through them. Mobility anchors in centralized mobility support approaches such as Mobile IP or Proxy Mobile IP are commonly designed as centralized components. To overcome the limitations and drawbacks of centralized nodes, a decentralized mobility management framework must minimize the number of centralized nodes as much as possible. As a design goal, the proposed approach should not have common nodes which are permanently involved in the mobility operations of every mobile node or a large portion of the mobile nodes managed by the system.

No Changes to End Hosts

Mobility support clients are usually end devices that are managed by individual users. Therefore, any approach that requires changes to the client, for example Mobile IP, needs to be implemented by these users. Moreover, corresponding revisions of the particular operating system, protocol implementations, libraries or applications need to be implemented and made available. In case of approaches, such as HIP or SCTP based mobility, changes are even required to systems on both sides of a communication. For these reasons, solutions that require changes to clients are more complex to deploy and cannot be unilaterally implemented by a single network operator or on-site access provider. Therefore, a mobility support solution that does not require changes to the end host has deployment advantages and is especially suited for operator driven deployments. As a design goal, the proposed approach should not require any changes to end hosts to minimize implementation complexity and allow for unilateral deployments on part of network operators and on-site access providers.

Deployable within the Current Internet

Based on the deployment scenarios stated in Section 3.1.1 the approach must be deployable in the current Internet environment. In addition to the previous design goal of not requiring changes to the end host this means that also no changes must be required to any intermediate nodes that are not within the direct control of the operator deploying the mobility support solution. For example, while it is acceptable to require an operator who deploys the mobility solution to make changes to its own routing infrastructure, it is not acceptable to require changes to the routing structure of any upstream provider or of any correspondent node's operator. The rationale behind this design goal is to lower deployment barriers, especially with the deployment scenario of small on-site network operators in mind (see Section 3.1.1).

No Impact on Non-Roaming Nodes

Even when using wireless connections, user can connect to a network without roaming to another network or access point for an extended time period or even without roaming at all. Moreover, a mobile user potentially does not even know in the beginning of a session if he will roam to another network. Common approaches to mobility management can impact nodes before they even start roaming. For example Mobile IP requires a client to route all its traffic via the home agent regardless whether the node is actually roaming or not. Depending on the current whereabouts of the node and the position of the home agent this can induce noticeable latency for the mobile node. Moreover, it puts load on the home agent since it needs to forward all the client's traffic even though the client does not actually require mobility management. An ideal mobility management approach will not impact non-roaming nodes. Moreover, the impacts that non-roaming nodes have on the managing systems will be kept to a minimum. Therefore, it is a design goal of the proposed approach to be transparent to nodes as long as they are not moving away from their initial point of attachment and therefore starting to roam to another point of attachment.

Low Deployment Barriers and Inherent Scalability

Deployability and scalability are characteristics that must be addressed by any system in order to be practical in real-world scenarios. While the extend of what can be considered a still acceptable deployment barrier and the upper and lower bounds of scalability demands are highly dependent on the specific scenario it is generally favorable to keep deployment barriers low and provide good scalability. Therefore, the proposed approach must keep its deployment barrier as low as possible. In particular this means that no dedicated and specialized nodes should be introduced since this would induce additional costs in terms of hardware and maintenance. Especially small deployments might not have the necessary amount of users to redeem such additional costs. However, it is considered feasible to require upgrades to existing nodes (in particular the existing access routers) to add mobility management functionality, for example via a firmware update. Generally not acceptable for the proposed approach are mechanisms or protocols that are not compatible with the current state of the Internet infrastructure (so-called "clean slate" approaches). The approach is required to work over the current IPv4 or IPv6 infrastructure respectively.

Since no additional nodes are introduced it is required that the approach provides inherent scalability. The occurring load must be distributed among the involved systems without putting exceeding high load on any single system. However, it is acceptable that load scales with the number of mobile nodes connected to an access router. That is the more nodes are connected to a single access router the higher its load is in terms of mobility management. In cases where a single access router cannot scale over the number of present mobile nodes, additional access routers must be deployed in the same area which take over the responsibility of a portion of the mobile nodes.

Minimize Configuration Delay

Mobility management solutions commonly introduce additional network latency during handovers due to management operations that must complete before the handover can be concluded. Such management operations are setting up the forwarding of a mobile node's data traffic to its new point of attachment and providing configuration data for the mobile node. While any node usually requires dynamic configuration when connecting to a network (e.g. via DHCP or IPv6 Autoconfiguration), the procedure of providing this information is more complex in network based mobility management solutions. A principle of operations in network based mobility management is to provide a mobile node with consistent configuration data when it changes points of attachment to hide the movement from the mobile node's network layer. To keep the configuration data consistent access routers have to make sure that they present the same configuration data to a mobile as the previous access router did. This usually involves a query to some kind of authoritative entity which coordinates the configuration data. For example the local mobility anchor in Proxy Mobile IP. However, in a decentralized environment there can be considerable latency between any two nodes. As a design goal, the proposed approach must employ appropriate methods to minimize any configuration delay during handovers even in distributed deployment scenarios where an authoritative entity might be topologically far away from an access router that a mobile node hands over to.

3.1.3. Assumptions

The design of the mobility management framework is based on a number of assumptions about the environment and the scenario that it will be deployed in. This section will briefly introduce these assumptions.

Geographical Movement and Topological Distances

Roaming in communication networks happens based on geographical positions and movement. Without making any specific assumptions about how a mobile node will select available wireless communication networks at any given time a mobile node will only have a limited number of choices between wireless stations that it can connect to. This set of available stations will change, among other factors, based on the geographical movement of the node. When a mobile node moves it will eventually leave the coverage area of the radio signal of wireless stations and enter the coverage area of others. If a mobile node happens to leave the coverage are of its currently associated radio station or finds a better suited radio station for further communications a handover to another station will occur. Terrestrial radio communication technologies used for wireless Internet access typically have a maximum transmission range between hundreds of meters (e.g. IEEE 802.11) to tens of kilometers (cellular networks). Based on this observation it can be assumed that handovers will occur between wireless stations within a limited geographical area. Intuitively, a mobile node is not expected to handover between wireless stations that are geographically very far apart (hundreds of kilometers or more). Furthermore it is assumed that geographically close stations are also somewhat topologically close. While no strong correlation between geographical and topological distances is assumed, two stations that are geographically close are expected to be more likely connected on a regional or national level than on a global level. Intuitively, even if two wireless stations are in different networks, these networks are expected to be interconnected via an exchange point that allows them to communicate somewhat directly (i.e. with regional or national round trip times) without any extremely large indirections (i.e. packets routed via another continent).

Inter-Operator Relationship

The different mobility domains can be operated by different network access providers. However, in that case, it is assumed that there is some kind of business and trust relationship between those operators. The approach requires a level of trust between the different session access routers that is comparable to the level of trust that each local mobility anchor requires from it's corresponding mobile access gateways in Proxy Mobile IP. Part of such operational agreements are, for example, the conditions under which users are allowed to move between domains, authentication methods, and security associations between authoritative nodes (e.g. access routers).

Seamless Connectivity

In order to provide any kind of seamless mobility on the networking layer an underlying seamless network connection is required. If a mobile node does not have a physical or link layer connection any attempt to provide mobility support on a higher layer is in vain. The mobility framework introduced in this thesis does not consider connectivity on the link layer or below. It is assumed that a mobile node can be provided with seamless connectivity by the lower layers and that it falls to the mobility framework to manage connectivity on the network layer and above alone. If a mobile node looses its primary wireless network connection there are no mechanism provided within the mobility management framework to take any action and existing connections are subject to timeouts.

Correlation between Mobile Node Density and Access Point Capacities

An equal distribution of mobile nodes over a geographical area is not likely. For example, city blocks with family homes will have less users than city blocks with high-rises and transport hubs such as airports or bus stations will likely have a higher node density than large open areas such as parks. However, it is assumed that there is a positive correlation between the capacities of the deployed access points and the density of mobile nodes. Intuitively, an area that has a certain density of mobile nodes will likely have access points deployed that are capable of handling the average node density. For example, an area with a high mobile node density will have either multiple access points deployed or access points that have a high capacity. Some limitations and exceptions to this assumption have to be acknowledged. For example there is a technical limitation to the number of wireless nodes that can be served in a certain area as radio capacities are a limited resource. However, the assumption, is based on the general understanding that an operator of an access point will try to reach a certain level of user satisfaction. This means that once there are more mobile nodes than an access point can handle and the service quality degrades below a threshold where the users are satisfied with the service, the operator will increase capacities or reduce the number of users.

3.1.4. Design Choices

Based on the envisioned deployment scenarios depicted in Section 3.1.1, the design goals stated in Section 3.1.2 and the assumptions made in Section 3.1.3 a number of design choices have been made. These design choices will now be discussed briefly.

Overlay Based

A fundamental design choice is the selection of the basic mobility concept that the framework will be build on (some common mobility concepts have been introduced in Section 2.1). An inter-domain deployment scenario as it is envisioned in Section 3.1.1 precludes any link layer based solution. Moreover, as argued with regards to link layer mobility (see Section 2.1) mobility support on the network layer or above is much more universal and can benefit from the fact that the network layer is designed as a common denominator for all Internet communication. An end-to-end mobility approach would implicitly require changes to the end host which conflicts with the design goal not to change the end host (see Section 3.1.2). A routing based approach does not only have bad scalability for host mobility but it also requires changes to the routing infrastructure. In small deployments a service provider might not even operate its own routing infrastructure. As small on-site access providers are one of the deployment scenarios (see Section 3.1.1) a routing based approach is therefore also not feasible. Locator/ID separation approaches require either changes to the end hosts or to intermediate nodes in the edge or access networks of the mobile node as well as the correspondent node. Changes to end hosts would conflict with the design goal not to change the end host (see Section 3.1.2), as stated before any changes to intermediate nodes outside of the operator's own network would conflict with the design goal of being deployable within the current Internet environment (see Section 3.1.2). As a result of these considerations the approach will be designed as an overlay based mobility solution that provides indirection or anchor points for mobile nodes. This design choice corresponds with all the stated design goals and can be expected to provide good deployability. Moreover, it is a well known concept that has been deployed by other mobility solutions such as Mobile IP and Proxy Mobile IP.

Network Based

An overlay based mobility approach can provide host based mobility as for example Mobile IP does or network based mobility as for example proxy Mobile IP does. However, to adhere to the design goal of not changing the end host (see Section 3.1.2) only a network based approach can be considered.

Access Routers as Mobility Anchors

An indirection or anchor point based mobility approach needs anchor points that topologically anchor a mobile nodes IP address. In practical terms this means that they constitute the routable destination for an IP address for any correspondent node. Because the anchor point is fixed it can provide a stable communication point for any correspondent node. Its primary function is to forward any incoming data to the current point of attachment of the mobile node. The choice and placement of the mobile anchor points has great implications for the routing efficiency of the system. The further the anchor point is outside the direct routing path between a mobile node and its correspondent node the larger the introduced routing inefficiencies will be because the traffic is routed via the anchor point. Such routing inefficiencies put additional strain on the network infrastructure and introduce additional end-to-end latencies. Corresponding to the design goal of not having an impact on non-roaming nodes (see Section 3.1.2) the anchor point must be in path with the mobile node before it starts roaming and any potential correspondent node. Either the access router that a mobile is attached to or any default gateway, for example an edge router, can fulfill this requirement. However, in multi-homed networks a default gateway beyond the access router might not be available. For this reason and to implement the design goals of limiting the number of bottlenecks and to keep deployment barriers low (see Section 3.1.2) the proposed mobility framework will use solely the access routers as anchor points. Moreover, based on the assumption that there is a correlation between the mobile node density and access point capacities in a certain area (see Section 3.1.3), the access routers will likely scale with the number of mobile nodes which provides an inherent scalability of the framework.

3.2. Framework Architecture

Following the design choices made in the previous section the framework will provide network based mobility support that uses access routers as anchor points for mobile nodes. Because the approach is build around access routers it will be referred to as Access Router Approach (ARA) in the remainder of this thesis. Figure 3.1 illustrates the basic mode of operation of the approach. When a mobile node initially connects to an access router, this access router assumes responsibility for the mobility management of the node. An access router that has assumed responsibility for the mobility management of a mobile node is called the *session access router* with respect to that particular mobile node. Upon movement of the mobile node to another access router all of its connections are being forwarded by the session access router to the current access router. If the mobile node moves again, the session access router changes the forwarding path to the new access router, reliving the previous access router of any further responsibilities with regards to the mobile node.

Figure 3.2 shows an overview of the processes that take place when a mobile node

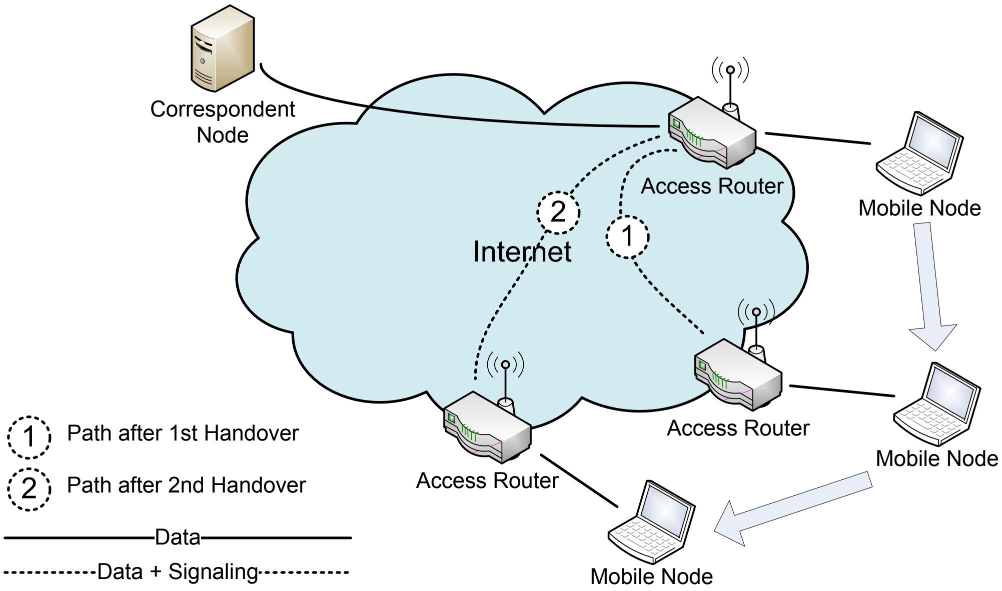


Figure 3.1.: ARA overview

(MN) attaches to an access router (AR). First, the access router must determine if the mobile node attaches initially or if the node hands over from another access router and already has a session access router. To this end the access router will lookup if a configuration already exists for the mobile node. If a configuration does exists the access can assume that it is the current access router (CAR) and will execute a handover procedure. Essentially a handover involves configuring the mobile node with the existing configuration and signaling the session access router to setup a forwarding for the mobile node's data (handovers will be explained in the following section in more detail). If a configuration does not yet exists in the system the access router can establish itself as session access router (SAR) and create a new configuration for the mobile node. The highlighted processes, namely the lookup of a mobile node's configuration from the system and signaling the session access router with the mobile node's current point of attachment, involve communications with the ARA system and therefore potentially introduce large latencies.

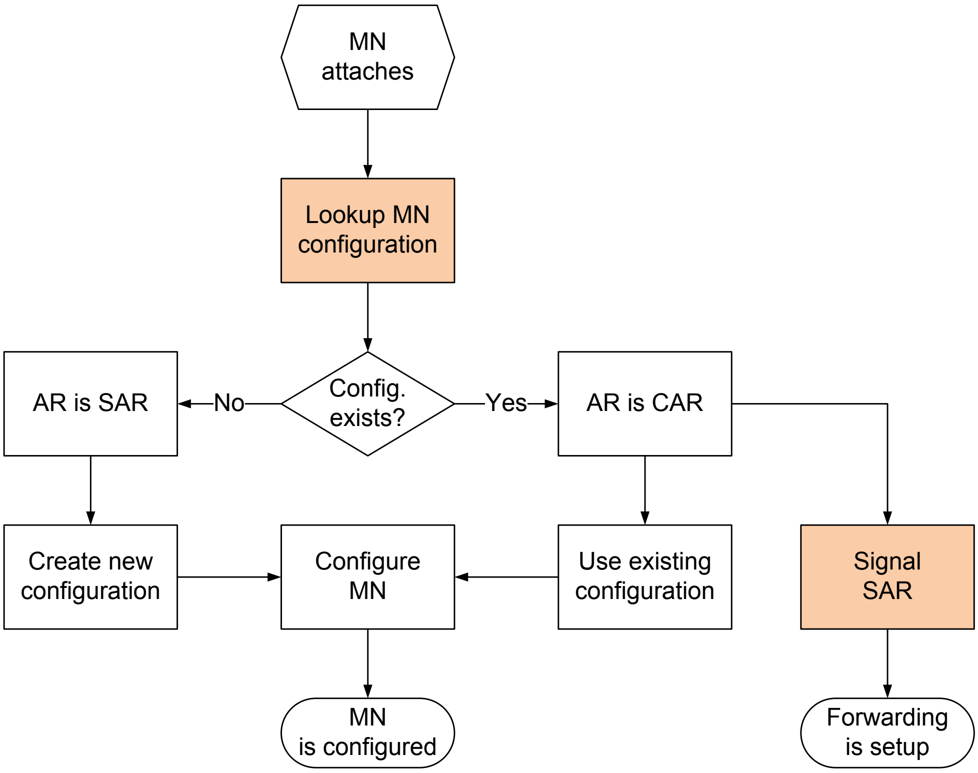


Figure 3.2.: Process overview of what happens when a mobile node (MN) attaches to an access router (AR).

3.2.1. Handovers

A fundamental operation for any mobility approach that uses anchor points for mobile nodes is the handover. When a mobile node changes its point of attachment to the network (i.e. moves from the old access router to the new access router) its binding at the anchor point has to be updated so that traffic from the corresponding node is sent to the new location of the mobile node. Data that is still forwarded to the old access router either gets discarded or needs to be forwarded to the new access router as well. Furthermore, in network based mobility management systems the new access router needs to ensure that the mobile node is configured with the same IP configuration as with the old access router in order for the movement to be transparent to the network layer.

Mobile Node Configuration

When a mobile node attaches to an access router the access router needs to determine a valid IP configuration for the mobile node. As a principle of network based mobility management a mobile node will be provided a stable IP configuration as part of the mobility support. This hides the mobility from the network layer of the mobile node and its correspondent nodes. Therefore, in order to provide the mobile node a valid IP configuration an access router must determine if the mobile node has already been assigned an IP configuration by a previous access router. If that is the case the access router must provide the mobile node with the same IP configuration otherwise the mobility support will break as the mobile node will reset its connections due to a change in IP configuration. Unless the access router already has information about a mobile node's IP configuration it must query the system for a valid IP configuration for the mobile node. If the query returns an existing IP configuration the access router will use this configuration to configure the mobile node. If the query does not return an existing IP configuration the access router must assume that the mobile node initially attaches to the network and provide an IP configuration from its own local pool. This local pool will commonly be a subnetwork that is assigned to the access router solely for this purpose. As a result a mobile node receives a local IP configuration when it initially attaches to the network just like it would without the mobility support. The exact mechanism that is being used by the access router to configure the mobile node is outside the scope of this thesis and dependent on the deployment scenario. For example, in IPv4 networks, dynamic host configuration is commonly performed using DHCP [40], while IPv6 has inherent configuration capabilities using IPv6 Stateless Address Autoconfiguration [158].

Data Forwarding

Once a mobile node has received an IP configuration it is able to send data. However, if the mobile node is not attached to its session access router it will only be able to receive data once a data forwarding has been setup from the session access router to the current access router. Since the IP address that the mobile node was configured with belongs to a subnetwork that is managed by its session access router (see previous section), normal routing mechanisms will deliver any data for this IP address to the session access router. The current access router must signal the session access router the current location of the mobile node so that the session access router can configure a data forwarding for any data that is destined for the mobile node to the current access router. Once the forwarding is established the current access router can deliver the data to the locally attached mobile node. The exact mechanism that is being used by to provide data forwarding is dependent on the deployment scenario. Simple IP in IP tunneling will suffice (e.g. [144, 115, 33, 45]). If there are security concerns the forwarding can also be based on a secure protocol such as IPsec [69, 70].

3.2.2. Global Session Cache

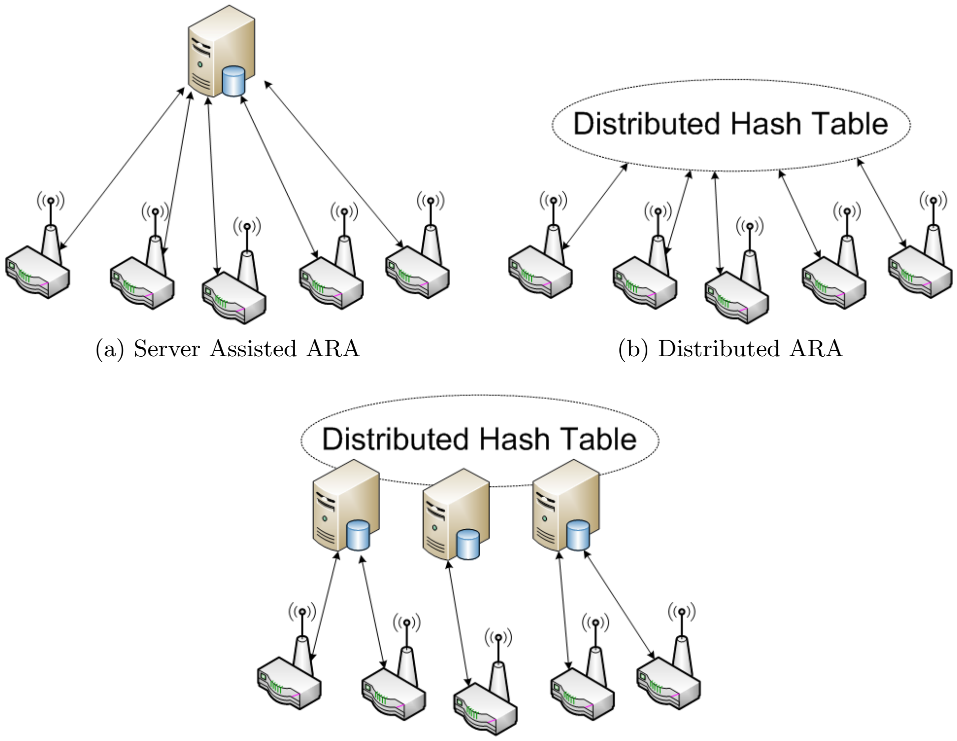
A main goal of ARA is to be easily and flexibly deployable. The actual mobility support is provided solely by the access routers which is a big step towards this goal. However, during a handover (and only during a handover) an access router needs to look up the context information of a mobile node. More specifically it needs to find the session access router for this node and query it for the mobile node's IP configuration as well as initiate the data forwarding on behalf of the mobile node. This lookup is realized using a cache that stores the mobile node to session access router associations. To implement this cache, a number of deployment models each with different characteristics can be considered.

Server Assisted ARA

A rather straight forward way is to implement the ARA cache on a dedicated node as shown in Figure 3.3a. The server can be bootstrapped into the access routers and presents a single and consistent entity to handle the complete cache. Depending on the number of participating systems in an ARA deployment this cache can be co-located within an access router or be a dedicated machine. The cache is only consulted during a handover and its operation is limited to a simple lookup and forwarding of signaling data. However, the server assisted approach still introduces a bottleneck and especially in large deployments can introduce noticeable latency in the lookup process because of long RTTs between the server and an access router. A distributed approach using data replication might alleviate some of this problems but introduces a higher deployment complexity. It must be noted that the actual forwarding of data between the mobile node and its corresponding nodes does not involve the cache in any way. The cache, no matter in which way it is deployed, is only involved in locating the session access router during a handover. Therefore, for the data forwarding it does not provide a bottleneck or a single point of failure.

Distributed ARA

A more scalable and flexible approach is to distribute the session cache over the involved access routers as illustrated by Figure 3.3b. The distributed ARA model benefits from the common attributes of distributed systems which is especially scalability. Moreover,



(a) Server Assisted ARA

(b) Distributed ARA

(c) Hybrid ARA

Figure 3.3.: ARA Deployment Models.

since the session cache is inherently co-located with the access routers there is no need to deploy any additional equipment. Therefore, distributed ARA allows very simple deployment. A new access router can be easily integrated in an existing ARA deployment just by bringing it online and bootstrapping it with the information of the distributed cache.

The cache is implemented as a distributed hash table (DHT) which is maintained in an overlay between the access routers. The DHT is using a recursive routing scheme which means that lookup requests are forwarded inside the DHT until they can be resolved. The final node which has the key-value mapping then forwards the request to the corresponding session access router. The exact DHT algorithm used for a distributed cache is flexible and can be adapted on a per-deployment basis. Promising candidates are structured peer-to-peer overlays such as Chord [151], Tapestry [171], Pastry [139], Kademlia [93], Viceroy [89], or a Content-Addressable Network (CAN) [129]. Also possible are one-hop DHTs (see [135] for an overview).

Hybrid ARA

The hybrid ARA deployment model as shown in Figure 3.3c combines server assisted ARA and distributed ARA. In this model the session cache is distributed among a number of servers which maintain the cache similar to the distributed approach. Essentially this deployment model combines the advantages of both previous approaches. The cache servers can be placed strategically to serve a crowd of access routers. A local placement strategy would for example place cache servers in certain regions or metropolises while a topological strategy would place cache nodes for example at peering points or edge networks. The goal of an optimal placement of cache servers is to reduce the round trip times between access routers and their respective cache server.

The benefit of the hybrid ARA deployment is twofold. First, because access routers and cache servers are localized, the latency between them is low. This leads to very efficient lookups for nodes that move within such localized access routers. Second, because the number of servers is rather small (compared to the total number of access routers) the lookups in the DHT are also very fast. Even in the case were the session access router of a mobile node is registered with another cache server, the result can be cached after the initial lookup to keep further lookups local.

3.2.3. Neighborhoods

As a decentralized system, an important design aspect of the ARA framework is to reduce any dependencies of an access router on other nodes. Dynamically assigning the initial access router as mobility anchor for a mobile node already distributes the function of anchoring and forwarding traffic for a mobile node in the system. Another important component of the ARA framework is the cache that contains information about the mobile nodes within a network. The global session cache is used during handovers by the new access router to obtain the relevant information about a mobile node such as its IP configuration, its session access router and its previous access router.

Section 3.2.2 discusses several possible deployment models for the global cache ranging from a dedicated server to a distributed hash table among all access routers. However, two deficiencies can be identified with regards to the global session cache: First, depending on the deployment model it impedes the distributed nature of ARA. And second, it introduces notable lookup latencies during handovers. Moreover, these two parameters are potentially diametrically opposed to each other as distributed cache approaches (e.g. based on DHTs) commonly have a higher lookup latency than simple server based ap-

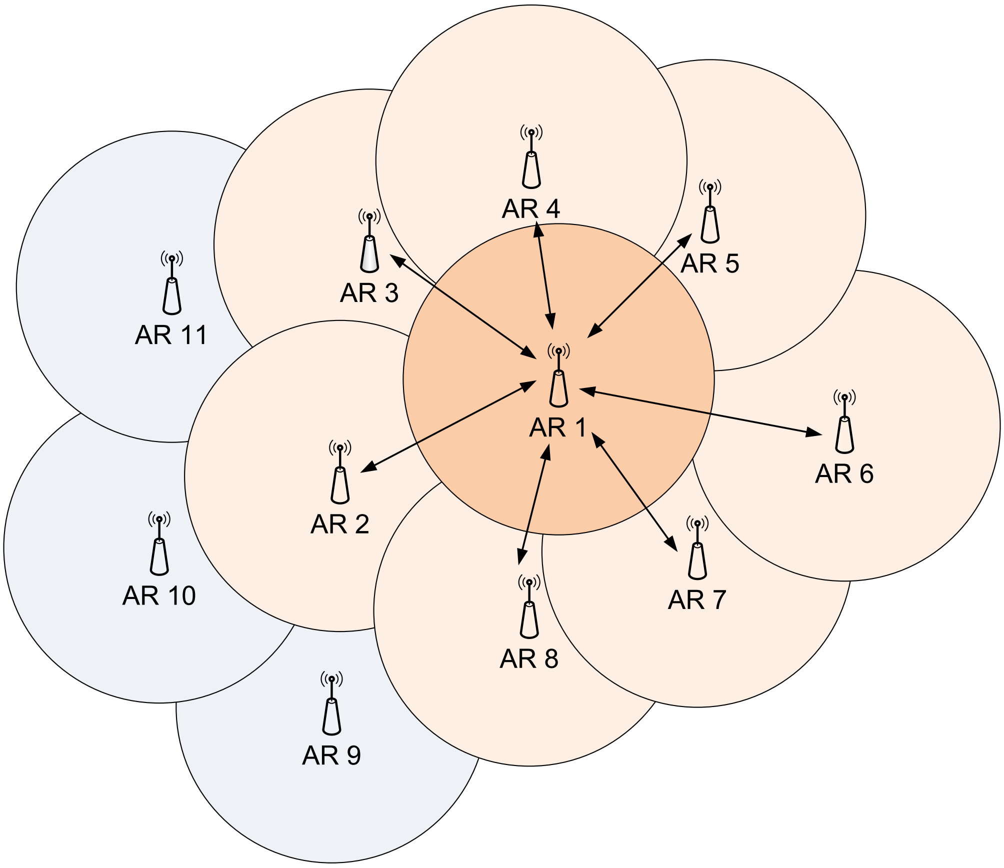


Figure 3.4.: Conceptual neighborhood of an access router

proaches. This means that the lookup latency tends to go up the more the cache is based on a distributed model. Although there are valid approaches to distributed databases based on one-hop DHTs available (see Section 2.1) that can significantly reduce the lookup latency, an optimal solution would not introduce any lookup latency at all.

To further reduce the lookup latency during handovers and to strengthen the distributed nature of the framework, ARA introduces the concept of neighborhoods. Essentially the neighborhood of an access router is the set of access routers that it can expect mobile nodes to hand in from. Figure 3.4 illustrates the concept: The neighborhood of access router AR 1 consists of access routers AR 2 to AR 8 as they have an overlapping area of service with AR 1 which would allow a mobile node to handover between them. Each of these access routers again, has its own distinct neighborhood. For example, the access routers AR 9 to AR 11 do not belong to the neighborhood of AR 1 as they are too far away for a handover to occur but they belong to the neighborhood of AR 2 (not illustrated).

Benefits

By maintaining individual neighborhoods access routers can stay in direct contact to any adjacent access routers and directly exchange information without querying the global session cache. Consider the scenario in Figure 3.4 again: Access router AR 1 and access router AR 2 are positioned close enough to each other (in a spatial sense) so that a mobile node can potentially move seamlessly between them. This makes AR 1 and AR 2 neighborhood candidates for each other since they would need to perform a handover for any mobile nodes that does move between them. By forming a neighborhood and regularly exchanging mutual information about their currently attached mobile nodes each access router learns about mobile nodes before any potential handover. This means that if a mobile node hands-in from a neighboring access router, the new access router does not need to query the global session database because the information about the mobile node is already available in its local session cache.

The main goal of ARA is to provide seamless mobility management. Therefore it is mainly concerned with mobile nodes that are moving between access routers that provide seamless coverage. In such a scenario, the neighborhood concept allows the ARA framework to transport a mobile node's information in a localized manner along the movement path of a mobile node. Every time a mobile node moves the ARA framework will proactively distribute any information regarding the mobile node to the new potential handover candidate access routers. The remaining access routers that do not require this information because a mobile node physically cannot hand in to them, do not experience any overhead because they are not in a neighborhood together with the current access router of a mobile node.

Essentially the neighborhood concept allows access routers to maintain any information pertaining to a mobile node in a truly decentralized fashion. By caching the information locally they also vastly improve the configuration lookup time. Despite the capabilities of neighborhoods, the ARA framework still maintains the global cache as a fall back in case a mobile node hands in from a previously unknown access router.

Forming Neighborhoods

Maintaining an accurate neighborhood positively impacts the configuration lookup performance directly and the overall handover performance indirectly. If an access router misses neighboring nodes in its neighborhood it will need to fall back to querying the global session cache if a node hands in from one of these nodes. This greatly impacts the configuration lookup latency and also increases the dependency on other nodes in the system (i.e. the nodes that provide the global session cache). On the other hand if an access router includes too many neighboring nodes in its neighborhood this unnecessarily increases the overall overhead. It is therefore important to have an effective selection strategy to form neighborhoods.

Static Configuration A simple approach would be to statically configure neighborhoods. However, this approach has a number of limitations. First, it might not be initially apparent which access routers actually do form a neighborhood. In such a case complex experiments would have to be conducted to identify neighborhood candidates. Second, this approach does not tend to scale well. While it might be feasible to statically configure neighborhoods in very small deployments it will be a cost a lot of time and effort in larger deployments. Third, static configuration consistently introduces manual overhead every time a neighborhood changes. Therefore static neighborhood configuration should only be considered in small non-productive deployments.

Location Based Selection A fairly straight forward way of selecting neighborhood candidates is to use location information. An access router can for example utilize GPS or IP geolocation to acquire its physical location or just have its co-ordinates statically configured. Based on this information it can use a database that includes the location of all other access routers in an ARA deployment and select neighborhood candidates based on the physical distance.

Besides introducing more overhead, dependencies, and requirements this approach also yields bad candidates because spatial proximity is a weak criteria for selecting access routers that can potentially hand-in mobile nodes. Since the propagation of radio waves is severely affected by physical objects two access routers that are closely positioned to each other can still be out of reach for a handover. For example in urban areas or buildings radio waves physically cannot propagate as well as in wide open spaces. Moreover, antenna gain, transmission power and positioning (the top of a radio mast vs. the basement) also greatly impact the actual range of wireless radios.

Overall, location based selection while a fairly straight forward approach seems not well suited to form accurate neighborhoods as locality is a bad indicator for actual handover probability. Moreover, this selection strategy would introduce additional overhead because the locations of each access router would have to be determined and maintained. As a result location based neighborhood selection is not considered in the ARA framework.

Topology Based Selection A topology based selection strategy is trying to identify neighborhood candidates based on topological proximity rather than geological one. The topological "position" of a node can be determined, for example, using network co-ordinates (see [39] for a recent survey of network coordinate systems) or simple measurements of hop counts or round trip times between two nodes.

In comparison with the location based selection strategy, the topology based selection strategy induces even more overhead since it requires network measurements. Even when using a network-coordinate system a basic set of measurements is necessary to determine a sufficient amount of parameters to determine a node's topological position. Furthermore, the topological position is an even worse indicator than the geographical position for determining neighborhood candidates. On the one hand, nodes can be topologically very close but geographically a long way away from each other (too far to allow for seamless handovers). An example is a city-wide operator who manages all of it's access routers within a single layer-2 virtual LAN. On the other hand, topologically distant nodes can be geographically close enough to allow for seamless handovers between them. Most commonly this is the case for access routers that belong to different operators but are deployed at the same location.

For the purpose of neighborhood selection in ARA, a topology based selection strategy seems to be a worse choice than a location based one. Not only does it introduce more overhead since dedicated measurements between nodes are necessary, it is also a very poor indicator for the actual handover probability between two access routers. As a result topology based neighborhood selection is also not considered in the ARA framework.

Sensing Based Selection Closer related to radio networks is a sensing based selection of neighborhood candidates. Using its own radio interface an access router can just scan all available frequencies to identify access routers that are within its own receiving range. If another access router can be identified by receiving its beacon it is a definite neighborhood candidate as its sending range overlaps with the one of the sensing access router.

A sensing based approach could even work without any kind of database since the two neighboring access routers can directly exchange information over the radio interface. In case the radio connection is volatile or has a low bandwidth it is sufficient to exchange

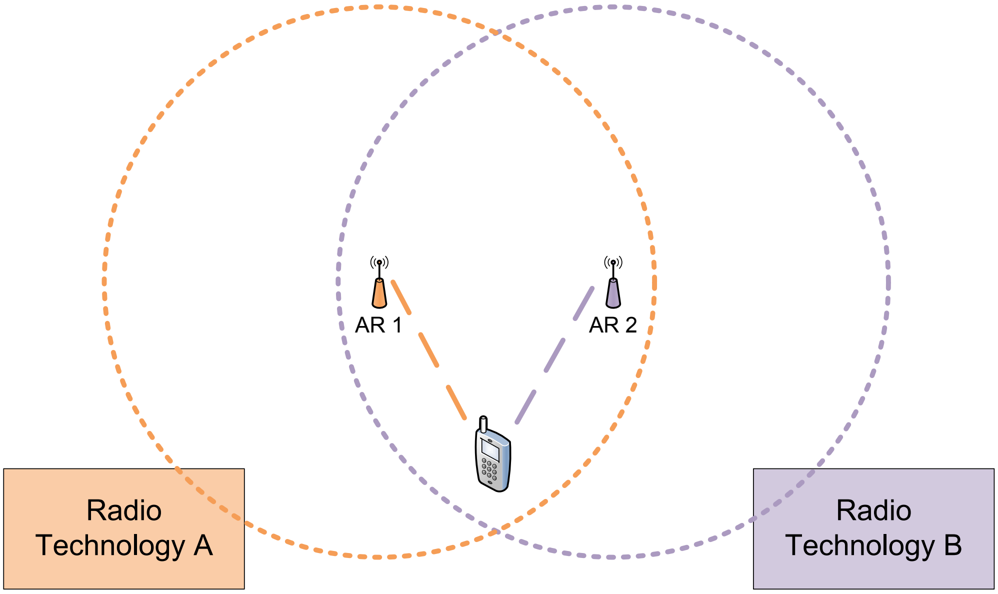


Figure 3.5.: Two access routers with different radio technologies.

connection details for other means of communication, such as IP addresses of a wired interface. Moreover, a sensing based approach also has the strong security implication that an access router cannot pretend to be in a neighborhood that it is not. If an access router does not have any overlapping service area with another access router it is simply not detected and, therefore, not included in the neighborhood.

The major drawbacks of the sensing based approach is that it only works within homogeneous radio technologies and that it generally tends to underestimate the neighborhood size. Since an access router uses its own radio to sense other access routers it can only detect access routers within the same radio technology. However, in principle, a device with multiple interfaces can also handover between access routers that do not have any radio technology in common. In such a scenario, as illustrated in Figure 3.5 it is impossible for an access router to detect a neighborhood candidate that uses a different radio technology. It is also impossible for an access router to sense a neighborhood candidate if they indeed have an overlapping service area but are outside of each others sending and receiving ranges as illustrated in Figure 3.6. While the mobile node moving within the overlapping service area can communicate with both access routers and, hence, perform a handover, the access routers cannot detect each other as their communication ranges are not large enough. As a result of these two scenarios the sensing based approach

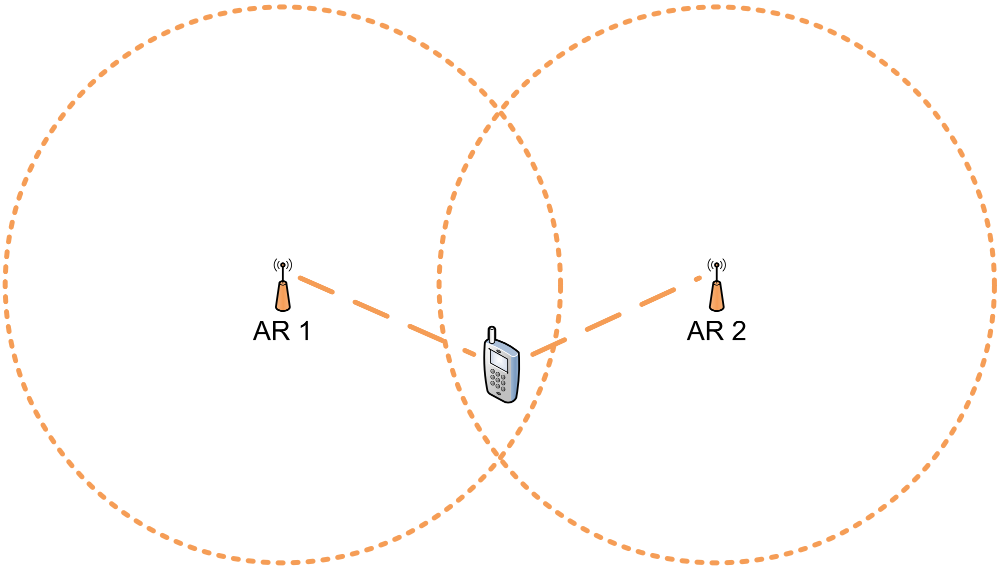


Figure 3.6.: Two access routers with overlapping service areas but outside of each others sensing range.

tends to underestimate the size of the neighborhood. Especially the scenario where two access routers have an overlapping service area but are outside of each others can be considered quite common as network providers tend to position access routers in such a manner that maximizes coverage and minimizes interference.

A sensing based approach to build neighborhoods seems a valid choice but it cannot detect access routers that use a different radio technology or are positioned outside a node's service are. As ARA, in principle, is not limited to access routers that have a common, homogeneous radio technology and deployments where access routers have an overlapping service area but are outside of each others transmission range tend to be quite common, a sensing based approach to form neighborhoods seems not optimal. As a result the ARA framework does not use sensing based neighborhood selection. However, as there is virtually no overhead associated with this approach, it is a very good candidate for a secondary selection strategy. As a matter of fact, active monitoring of the radio environment is already quite common for access routers, for example, for channel selection purposes or as part of cognitive radio applications [100]. A sensing based neighborhood selection as a secondary selection strategy can be used to effectively supplement an effective neighborhood selection process by avoiding overhead that is associated with the primary selection strategy or bootstrapping the access router with a basic neighborhood before any other selection process is started.

Self-Learning A rather simple but effective approach is to build neighborhoods based on self-learning. In this strategy an access router simply maintains a list of neighboring access routers which a mobile node has handed in from in the past. Every time a mobile node hands-in the access router determines the previous access router and compares it to its current neighborhood list. If the previous access router is not on the list it is included as a new neighboring node. Over time this allows an access router to build an accurate list of its neighborhood.

The self-learning approach has two distinct advantages: it is simple and accurate. Assuming an access router has alternative means for obtaining a mobile node's information the self-learning approach does not require additional infrastructure, devices, or a dedicated protocol. If an access router does not have information about a mobile node because the mobile node handed-in from a previously unknown access router, the access router just employs the alternative means to obtain any information it needs to perform a handover for the mobile node including information about its previous access router. This not only enables an access router to perform a handover for a mobile node despite its missing information from the local neighborhood cache, it also implicitly builds an accurate neighborhood. The self-learning strategy is very accurate because a node only becomes a neighborhood candidate after at least one mobile node has actually performed a handover between the two access routers. This by itself is a clear indication that the two access routers are neighborhood candidates.

The drawback of the self-learning approach is that it requires alternative means to obtain the information that an access router needs about a mobile node to perform a handover. This potentially makes the whole system more complex, impacts the performance, and introduces additional dependencies, overhead, and latencies. Naturally the degree to which this impacts the overall system depends on actual means that are being employed to obtain the handover-related information. However, overall the impact gets mitigated by the fact that an access router only needs to occasionally fall back on the alternative means, namely when a hand-in happens from a previously unknown neighboring access router. Moreover, this will happen more frequently at the beginning of an access router's operating phase when it's neighborhood list is still mostly empty and subsequently decrease. It can be expected, that after an initial phase the system has stabilized in the sense that the access router has learned most of its neighbors and hand-ins from unknown access routers happen much more sporadically.

One noticeable exception are mobile nodes that are initially (i.e. at the beginning of their mobility session) connecting to an access router. Because these nodes are just starting their mobility session they are unknown to every access router in the network. This also means that they are not in the local neighborhood cache and the access router will treat them like it would treat a hand-in from an unknown access router. However, while the impact on the overall system is the same for a node that hands-in from an unknown access router and for a newly attaching node, the impact on the node itself is quite different. It can be expected that a node that hands-in from an unknown access router has ongoing IP-connections that are directly impacted if an access router needs to perform an alternative lookup, for example if the alternative lookup introduces additional latency. However, in the case of a newly attaching mobile node there will be no ongoing IP-connections since the node initially attaches to the network. The impact of an alternative lookup in this case will be far less, if it is even noticeable. Intuitively, a user will barely notice if the IP-configuration of his mobile device takes a little bit (up to a couple of seconds) longer.

The self-learning strategy is the primary neighborhood selection strategy in ARA because of its simplicity and accuracy. While there are drawbacks to consider, maintaining a global cache also adds another layer of redundancy to the system. Overall the self-learning strategy of building neighborhoods together with a global cache as fall-back seems to strike a good balance of accuracy and complexity. While individual mobile nodes might be impacted during the learning phase this is deemed acceptable by ARA as long as it does not break the handover for these particular nodes altogether.

To speed up or kick start the learning phase a secondary neighborhood selection strategy such as sensing can be employed to complement the self-learning strategy. In this case the neighborhood candidates that are being selected by the secondary strategy can be directly added to the neighborhood list without waiting for a mobile node to hand-in from the candidate access router. Because an overestimation on part of the secondary strategy would directly inflate the neighborhood list a secondary neighborhood selection strategy that does not overestimate the neighborhood works best. However, in combination with an effective maintenance of the neighborhood list even an overestimating secondary selection strategy can effectively complement the self-learning neighborhood selection. Another mechanisms to increase the neighborhood learning rate that is implemented in ARA is to let an access router know when it has been added to a neighborhood. If an access router learns about a new access router in its neighborhood it sends a simple hello message to the newly learned access router to let it know that it was determined to be a part of the sending access router's neighborhood. This allows an access router which receives such a hello message to add the sending access router to its own neighborhood. As neighborhoods are usually reciprocally this allows for a bi-directional and therefore faster learning process.

Table 3.1.: List of ARA design goals.

Index	Design Goal
DG 1	Minimize Bottlenecks and Single Points of Failure
DG 2	No Changes to End Hosts
DG 3	Deployability within the Current Internet
DG 4	No Impact on Non-Roaming Nodes
DG 5	Low Deployment Barriers and Inherent Scalability
DG 6	Minimize Configuration Delay

3.3. Verifying the Design Goals

This chapter has presented the design of a framework for decentralized network based mobility management. Before the next chapter will introduce a system design based on this framework, the results of the design will be verified. Based on the design goals that were established in Section 3.1.2 the framework design will be evaluated in this section with regards to the realization of these goals. For reference purposes the design goals are listed in Table 3.1 and designated DG 1 through DG 6.

Using the outlined architecture, the Access Router Approach does not rely on any dedicated mobility anchors as other approaches commonly do. Every ARA-enabled access router can inherently provide mobility management functions to mobile nodes that are moving between ARA-enabled access routers. There is no necessary involvement or support needed of any other node in the network expect the set of two access routers that are responsible for a mobile node (the session access router and the current access router) at any given point in time. For the system of access routers and mobile nodes as a whole this means that there is no common node that provides a bottleneck or a single point of failure for every other node in the system. The architecture, therefore, satisfies the design goal to minimize bottlenecks and single points of failure (DG 1). It also satisfies the design goal of being deployable within the current Internet as no changes are required to intermediate nodes or corresponding nodes (DG 3). Furthermore the architecture contributes to the design goal of low deployment barriers and scalability as the system can be easily extended by deploying more access routers without having to consider capacity limitations of other nodes (DG 5).

By providing network based mobility support the framework implicitly satisfies the design goal of not requiring any changes to end hosts (DG 2). To satisfy the design goal of not impacting non-roaming nodes the framework dynamically assigns the first access router that a mobile node attaches to as its anchor point (DG 4). The session access router provisions a valid IP-configuration from its own IP-range which is no different than if a non-mobile node would attach to a common access router. This topologically anchors the mobile node in the network it actually resides at the point in time before it starts roaming. By design this does not introduce any routing inefficiencies in contrast to other approaches, like Mobile IP, where the mobile node is topologically anchored in the subnet of the Home Anchor. To keep the IP-configuration fixed when the mobile nodes moves, any subsequent access router must ensure that it provisions the same IP-configuration. While this is solved by the framework, a side-effect is that the mobile node keeps being topologically anchored at it's first access router. Therefore, the routing performance in terms of routing stretch degrades if the mobile nodes moves (topologically) further away from its session access router.

Finally, neighborhoods as a mechanism to distribute a mobile node's configuration data to any potential handover candidate access routers before a handover occurs contribute to the design goal of minimizing configuration delay (DG 6). After the neighborhood forming process is complete, neighborhoods should be able to reduce configuration delays during handovers close to zero as the configuration of a mobile node is already available locally. Only mobile nodes that are just starting a mobility session and associating for the first time will be subject to a network lookup delay as no valid configuration data exists yet. However, it can be assumed that this configuration phase as it does not presents a handover with ongoing connections is far less sensitive in terms of delay. In conclusion, the proposed ARA framework can satisfy all six of the established design goals.

4. System Design and Implementation

The previous chapter has introduced the conceptual system design of the ARA framework which provides network based decentralized mobility support using an overlay concept with indirection points. Based on the conceptual framework this chapter will present the more concrete system design and discuss some optional enhancements.

4.1. ARA Nodes

To provide network based mobility support, an ARA implementation needs to provide two core functionalities: 1) It needs to make changes in the point of attachment of a mobile node transparent, and 2) it needs to forward data for a mobile to its current point of attachment. As the ARA framework is decentralized these functions are assumed by individual access routers on a mobile-node-by-mobile-node basis rather than by a centralized entity. In ARA the session access router is responsible for managing a mobile node's data forwarding and the current access router is responsible for making a change in the point of attachment transparent.

4.1.1. Bootstrapping the Access Router

Depending on the complexity of an ARA deployment a number of configuration variables need to be configured on each access router. Such a configuration includes, for example, the type and parameters of the global session cache (see Section 3.2.2), link layer parameters (e.g. the Extended Service Set Identifier (ESSID) or WPA keys for IEEE 802.11), or any kind of policy an access router might need to be aware of. It can be assumed that the configuration of an access router will stay rather static as these are the basic operating parameters for an ARA deployment. Therefore, it is viable to assume a manual bootstrapping procedure. This can be for example a configuration file which is provided for all participants in an ARA deployment. Depending on the specific deployment scenario the configuration can be made available by proper means, for example a secured website.

4.1.2. System Components

An ARA node has a number of basic components which handle any data that is associated with known mobile nodes and known neighboring access routers.

Session Cache

The session cache is a register maintained by an ARA node that contains data about mobile nodes that are active within the access router's area of service or its neighborhood. The data structure is used to maintain any local and common states for a mobile node. As the session cache is a local data structure the particular data that it holds is dependent on the implementation of the ARA layer. For example, nodes with multiple wireless interfaces or nodes connected to multiple access points might want to store an additional information about which interface a mobile node is connected to. An implementation can also store timestamps used for timeouts and retransmissions in the session cache. However, certain information is required to be maintained in the session cache regardless of the particular implementation. Most notably this includes data about the ARA nodes involved in a mobile node's mobility management and data about its IP configuration:

Identifier A stable identifier that can be deducted from the communication with the mobile node. Examples for such identifiers are an IEEE 802 Media Access Control address (MAC address) [154], a Network Access Identifier (NAI) [2], a International Mobile Subscriber Identity (IMSI) [60], or a International Mobile Equipment Identity (IMEI) [44]. This identifier is used to uniquely identify a mobile node in the ARA system and must be the same in all access routers for the same mobile node.

Session Access Router The ARA node that is the session access router for the mobile node. If a mobile node associates with an ARA-enabled access router this information is used to send direct Session Update Requests to the session access router in order to initiate data forwarding of the mobile node's data to the current access router.

Current Access Router The ARA node that is the current access router for the mobile node. This information is mainly relevant for mobile nodes that are currently not associate with the access router but are active in its neighborhood. If such a mobile node roams into the service are of the access router and associates with it, the stored current access router becomes the previous access router and the information can be used to send a direct Session Update Requests to initiate temporary data forwarding. A session access router uses this information to maintain data forwarding to the access router that a mobile node is currently attached to.

IP Configuration This is the mobile node's IP configuration data as it was assigned by the session access router. It is used by any current access router that a mobile node associates with to present an identical IP configuration, therefore making the movement transparent to the network layer. The particular configuration data depends on the IP version that is used. For example, an IPv4 configuration that is disseminated via DHCP usually contains an IP address, a subnet mask, the default gateway, and optionally DNS servers. An IPv6 configuration that is disseminated via auto-configuration usually contains an IP prefix, the prefix length, and the prefix of the default gateway.

Gateway Address The IPv4 or IPv6 address of the gateway that the mobile will send its outbound traffic to. This is usually the address that the session access router has in its local network. The current access router needs to accept any traffic that is directed to this address and forward it towards its destination.

Node Type As the session cache is used to store data about different types of nodes (e.g. session nodes, visiting nodes, neighboring nodes) this flag is used to keep track of the particular type of the node.

Neighborhood List

The neighborhood list is a simple data structure that contains all known neighbors of an ARA node. If data about a locally associated node changes (e.g. a new node associates or an associated node disassociates) this information is disseminated to any node in the neighborhood list. Please refer to Section 3.2.3 for detailed information about how neighborhoods are built and how data is synchronized within them.

4.2. ARA protocol

When a mobile node attaches to an access router, this access router uses the ARA protocol to provide mobility support for the mobile node. The ARA protocol is a simple signaling protocol that is used to transfer context information (mainly the IP configuration for a mobile node). It also sets up the data forwarding between the session access router and the access router which the mobile node is currently attached to. Mainly an access router needs to find out if a session access router and an IP configuration already exists for a mobile node before responding to any configuration requests of the mobile node. One of the main design objectives of ARA is to provide network based mobility and not to require any changes to the mobile node. This means that ARA cannot rely

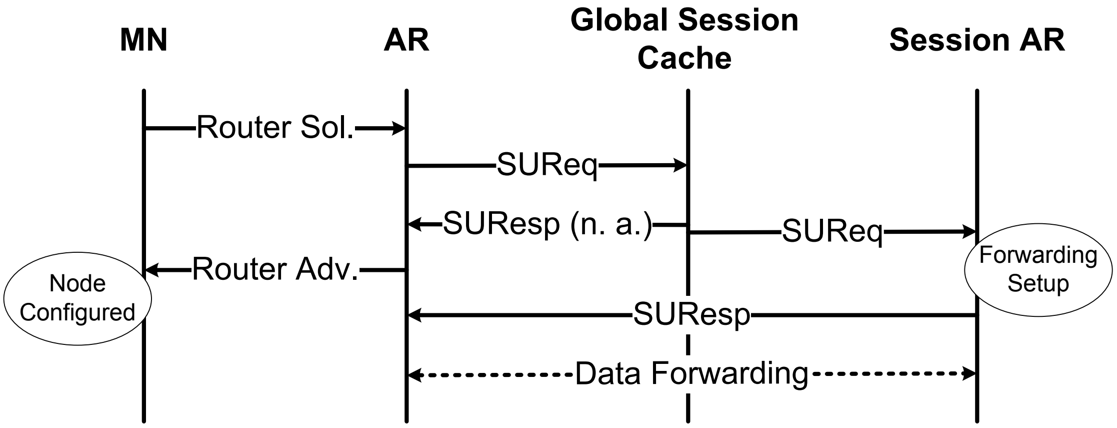


Figure 4.1.: ARA protocol flow

on any extension of the mobile node to provide a hint to the access router about any previous attachments. In order to allow an access router to locate the session router of a mobile node a global session cache is introduced which includes all the mappings between mobile nodes and their corresponding access routers. Different deployment models for the global session cache have been discussed in Section 3.2.2.

The general protocol flow in ARA when a mobile node attaches to an access router is shown in Figure 4.1. As soon as the access router can deduce the identity of the mobile node, for example by extracting its hardware address from the initial router solicitation message, it sends a Session Update Request (SUReq) to the global session cache. The request contains the identity of the mobile node, the address of the access router and a message sequence number. If the global session cache does not have a session anchor entry for the mobile node yet, it will designate the current access router as session access router and sent a Session Update Response (SUResp) back, indicating this. In case there is a valid entry found for a corresponding session access router, the global session cache will forward the Session Update Request to the session access router. To reduce query times the global session cache will also send a non-authoritative Session Update Response back to the querying session access router which contains enough information to configure the mobile node. As soon the current access router receives a Session Update Response message with a valid IP configuration it can use the IP configuration to configure the mobile node, for example by sending a router advertisement message. When the session access router receives a Session Update Request concerning a mobile node for which it is responsible for it will sent a Session Update Response to the access router from which it received the request. The response includes the IP configuration for the mobile node (e.g. an IPv6 network prefix) as well as the original message sequence number. A copy of the response is sent to the global session cache to freshen the session cache entry. Furthermore the session access router creates a data forwarding tunnel between itself and the current access router. This tunnel is used to forward any data that is intended for the mobile node to the current access router. As shown in the figure the data forwarding for the mobile node is setup by the session access router as soon as it receives the Session Update Request message. The ARA protocol defines a number of protocol messages to exchange data and signal states. ARA messages are exchanged over IP and can either be transported via UDP or TCP.

4.2.1. Session Update Request

A Session Update Request is sent from an access router that is responsible for a particular mobile node and indicates that data for this node needs to be updated. When a mobile node associates with an access router, the access router will send a Session Update Request to the global session cache or directly to the session access router of the mobile node (if known) to indicate that the mobile node has changed its point of attachment. The current access router will include the mobile node's IP configuration in the request that is valid from its current perspective. In case the node is unknown to the current access router the IP configuration will be a fresh and valid IP configuration from the local pool. In case the node is known through previous association or through neighborhood updates the IP configuration will be carried over. When a session access router receives a session update request directly from the current access router and not via the global session cache it will forward the request to the global session cache. Similar, if the global session cache receives a Session Update Request that does not originate from the session access router of a mobile node it will forward the request to the session access router. A node that receives a session update request must reply with a Session Update Response. If an access router receives a valid Session Update Request for a mobile node that it is the session access router for, it will update the data forwarding accordingly. In a soft state system a Session Update Request message will be send to the session access router for every mobile node that is associated with an access router at least once every timeout period to keep the states fresh.

A Session Update Request message contains the following data:

- The identity of the mobile node that the update request pertains to
- The address of the current access router
- The mobile node's expected prefix (IPv6) or IP address (IPv4)

- The mobile node's expected prefix length (IPv6) or subnet mask (IPv4)
- The expected gateway address that a mobile node will be using as a default

4.2.2. Session Update Response

When a Session Update Request is received the access router will respond to the request with a Session Update Response. The response will include the configuration data that is associated with the mobile node at the responding access router. If the Session Update Request was for a mobile node that does not have an IP configuration yet the configuration data will be carried over from the Session Update Request and the requesting access router will be designated session access router for the mobile node.

A Session Update Response message contains the following data:

- The identity of the mobile node that the update response pertains to
- The address of the session access router
- The address of the current access router
- The address of the previous access router that the mobile node was associated with
- The mobile node's expected prefix (IPv6) or IP address (IPv4)
- The mobile node's expected prefix length (IPv6) or subnet mask (IPv4)
- The expected gateway address that a mobile node will be using as a default
- Result type to indicate whether the requesting access router has been designated current access router or session access router
- A flag to indicate whether the response is authoritative (in case it is send from the session access router) or non-authoritative (in case the response is send from the global session cache)

4.2.3. Neighborhood Hello

The Neighborhood Hello message is sent by an access router to indicate that it wants to be added to the receiving access routers neighborhood list. Typically a Neighborhood Hello will be sent by an access router after a previous access router has been indicated by a Session Update Response message that was unknown to the receiving access router. Neighborhood Hello messages speed up the neighborhood forming process. Without

Neighborhood Hello messages neighborhoods will only be formed on part of access routers that mobile nodes handover to as they can learn the previous access router from the Session Update Response message. Using Neighborhood Hello messages the access router that a mobile node handover from can learn the identity of the next access router. This allows to establish neighborhoods in pairs instead of just unilaterally.

A Neighborhood Hello message contains the following data:

- The address of the sending access router

4.2.4. Neighborhood Update

To proactively distribute configuration information about a mobile node to potential handover candidate access routers of a mobile node Neighborhood Update messages are used. After a mobile node has successfully handed over to an access router the access router will send a Neighborhood Update message to all access routers in its neighborhood list which contains the IP configuration data that was used to configured the mobile node. In a soft state system the Neighborhood Update message will be resend for every mobile node that is still associated with an access router at least once every timeout period.

A Neighborhood Update message contains the following data:

- The identity of the mobile node that the update pertains to
- The address of the session access router
- The mobile node's expected prefix (IPv6) or IP address (IPv4)
- The mobile node's expected prefix length (IPv6) or subnet mask (IPv4)
- The expected gateway address that a mobile node will be using as a default

4.2.5. Neighborhood Purge

If an access router receives Neighborhood Update messages from an access router that it is not interested in receiving updates from it can send a Neighborhood Purge message to the corresponding access router to solicit its removal from the destination access router's neighborhood list. The Neighborhood Purge allows an access router to exercise a certain level of control over the number of access routers that it gets regular update messages from. For example, an access router could determine that the amount of updates it receives from a neighbor is grossly disproportionate to the amount of handovers from that access routers and request a purge.

A Neighborhood Purge message contains the following data:

- The address of the sending access router

4.3. State Synchronization

For every mobile node there exists a state at least in the global session cache, in its session access router, and in its current access router. Furthermore, Neighborhood Update messages are sent to any access router that is on the neighborhood list of the current access router. In a decentralized system such as ARA is important to assign clear responsibilities for state synchronization and maintenance to prevent discrepancies between the state for a mobile node that is maintained in multiple nodes. Moreover, as Neighborhood Updates essentially broadcast a mobile node's state to all neighboring access routers it must be ensured that neighborhoods themselves are maintained accurately to keep the overhead of Neighborhood Update messages to a minimum.

4.3.1. Session Cache

Maintaining session states for a mobile node is vital for its mobility support. If a mobile node's state expires its mobility support will be reset and it will be treated as a new mobile node that associates initially by its next access router. Every time a handover occurs the session access router will be queried for the mobile node's configuration data either via the global session cache or directly. If the query is forwarded via the global session cache, a non-authoritative response message will be sent by the global session cache to reduce the configuration lookup latency. Therefore it is important that a mobile node's state information is synchronized between the session access router and the global session cache. Moreover, as the current access router of a mobile node is using Neighborhood Update messages to inform neighboring access routers of a mobile node's state it needs to ensure that its session cache stays entry consistent with the session access router's data and the global session cache's data. If the state of a mobile node expires the mobile node will be treated as a new node and its mobility support will be re-initialized. In case the state expires with the session access router, it will stop forwarding data for the mobile node and might re-assign the IP configuration to the next mobile node. If the state expires in the global session cache it will not forward any Session Update Requests anymore and instead reply with a Session Update Response that indicates to the querying access router that it has been designated as the mobile node's session access router.

To ensure a consistent state for a mobile node's session cache entry on every node, one node has to be assigned the authoritative responsibility for the cache state. This could be either the global session cache, the current access router, or the session access router. The authoritative node is the only node in the system that can trigger state updates. The global session cache has a somewhat central role in the ARA framework and as such can be considered as authoritative entity for session states. However, first of all, the global session cache's significance in ARA conceptually diminishes over time as neighborhoods are formed. Second of all, as ARA is designed as a decentralized system, increasing the importance of any single entity is counterproductive. Third of all, the global session cache is the only entity that is not directly involved with the mobile node (the mobile node is neither associate with it nor does it forward data for the mobile node). Therefore it does not have any first hand indications about the actual state of a mobile node. For these reasons the global session cache is not considered a good solution as main authority over session states.

The current access router has an active association with the mobile node and as such has the best indication about the current state of a mobile node. However, first of all, this association is only a temporary as the mobile node will eventually move to another access router. Switching responsibilities for state maintenance between entities (i.e. from the current access router to the next access router once the mobile node moves onward) potentially causes problems when the switch takes too long and timeouts occur during the switch. Second of all, the current access router might be in a different administrative domain than the session access router which initially assumed operative control and accountability for the mobile node. In the interest of keeping clear accountability it seems pertinent to keep the transfer of responsibilities between administrative domains to a minimum. For these reasons the current access router is also not considered a good solution as main authority over session states.

The session access router did not only initially assume responsibility for the mobile node, it also forwards the mobile node's data continuously. Furthermore, the session access router stays fixed for a mobile node as long as the mobility session is ongoing. Therefore, in ARA the session access router is designated as the main authority responsible for the state of a mobile node. This means that the global session cache must only update its session cache entry based on dedicated Session Update Request messages send by the session access router and the current access router must only send Neighborhood Update messages once it receives a Session Update Response message from the session access router. However, as the current access router is the only entity that has an active association with a mobile node the general responsibility is somewhat split. While it is the responsibility of the session access router to update a mobile node's state it is the responsibility of the current access router to provide the session access router with continuous information about the state of a mobile node via Session Update Request messages.

4.3.2. Neighborhoods

For an access router to maintain a neighborhood is an active and ongoing process. Neighborhood maintenance means primarily to keep the neighborhood list concise and accurate. The complexity of this process depends on the churn rate of the neighboring nodes (i.e. the rate at which new neighbors become available and existing neighbors become unavailable) and the estimation accuracy of the neighborhood candidates selection strategy. The bigger the churn rate is and the less accurate a selection strategy is the more complex is the neighborhood maintenance.

Depending on the deployment scenario churn may be generated by a number of factors. Simple and less dynamic examples are new access routers being deployed or existing access routers being removed or repositioned. There are also examples that can lead to a more volatile churn rate, such as access routers regularly changing their operating state (e.g. they are being switched on or off), or a temporary change in conditions that alter the range of radio transmissions (e.g. better or worse propagation conditions caused my meteorological changes, more/less interference from external radios, or temporary physical obstacles for radio transmissions). Churn can even be inherent to a certain deployment scenario. For example, an access router that is mounted on a moving vehicle will experience considerable churn and cause minor churn for access routers that the vehicle passes by. Essentially churn leads to the neighborhood list not being static but (to a certain degree) dynamic. Besides the churn rate, the accuracy of the neighborhood selection strategy is also a contributing factor to the complexity of neighborhood maintenance. A selection strategy that largely overestimates the list of neighborhood candidates is in turn exceedingly inflating the neighborhood list while a selection strategy that tends to underestimate does not provide enough neighborhood candidates to build an adequate list.

To effectively maintain an accurate neighborhood an access router needs to regularly purge old neighbors from the neighborhood list and add new neighbors to the list. Both theses process are highly dependent on the employed neighborhood candidate selection strategy. For example, if the chosen selection strategy tends to overestimate the neighborhood candidate list, the access router might be more aggressive in purging entries from the neighborhood list. On the other hand if the chosen selection strategy tends to underestimate the neighborhood the access router might keep entries longer in the neighborhood list and perform a neighborhood selection process more often.

A general strategy to remove stale entries from the neighborhood list is to attach a timestamp to each list entry and to define a timeout value after which the entries will be removed from the list. The timestamp attached to a list entry should be updated every time a mobile node actually does hand-in from that particular neighboring access router. Similar a fixed sized list could be maintained by replacing least used entries with access routers that are newly discovered. Any specific parameters for a neighborhood maintenance strategy have potentially great impact on the performance of the ARA network and as such must be adjusted depending on the particular deployment. For example, if the algorithm removes entries from the neighborhood list too aggressively the overall handover performance will be impacted as the neighboring access routers need to query the global session cache instead of using local information when a mobile node hands in.

In order to find new neighboring access routers, the access router needs to depend on its neighborhood selection process (see Section 3.2.3). Depending on the chosen selection strategy there might be a number of parameters to adjust (e.g. how often to query the location database for nearby nodes in the location based selection strategy) but largely the access router needs to wait for the algorithm to select a neighborhood candidate and then add it to the list. As mentioned in Section 3.2.3, a secondary strategy can be chosen to complement the primary strategy for better results.

Session Cache Synchronization

The purpose of neighborhoods is to provide access routers with the configuration data of mobile nodes before they actually handover to the particular access router. In order to fulfill this purpose an access router needs to synchronize the data of a mobile with the mobile node's current access router in due time (i.e. before the mobile node initiates the handover process). To perform the synchronization with its neighborhood access routers an access router has a number of options that will be discussed below.

Push-Model Following a push-model approach an access router can subscribe to all of its neighboring access routers to receive any updates with regards to mobile nodes. Once an access router has identified a neighboring access router it will send a simple subscribe or hello message to the neighboring access router to indicate that it is interested in receiving neighborhood updates. An access router that receives such a subscribe message adds the subscribing access router to a neighborhood update list and initially send a complete set of configuration data for any currently active mobile node to the subscribing access router. After this initial synchronization the access router will push any changes that occur within its own area of responsibility (e.g. new mobile nodes arriving or current mobile nodes leaving) to any access routers on the neighborhood update list. If an access router is not interested in any updates from a particular neighboring access router anymore it will send an unsubscribe or purge message to this neighbor. If an access router receives such an unsubscribe message it will simply remove the corresponding access router from its neighborhood update list which will prevent any further updates from being send to the unsubscribing access router. An access router also needs to remove entries from the neighborhood update list if a subscribed access router is no longer reachable. This simple push model can be extended to a soft-state protocol by requiring any access router to periodically re-affirm its interest in updates. This can be done via dedicated keep-alive messages or simply by repeating the subscribe message. However, in the later case the subscribe message needs an indication for the receiving access router that a complete synchronization is not necessary and that the (re-) subscribing access router is just still interested in incremental updates. If an access router does not receive any update messages from a peer within a certain time frame it will automatically remove the particular peer from the neighborhood update list.

The benefit of a push-model is that the signaling and synchronization overhead caused by neighborhoods is minimized. Besides the initial synchronization that only occurs when an access router first subscribes for updates the only time that data messages are sent is when an actual state change happens with regards to an access router's associated mobile nodes. However, the push-model has a somewhat higher implementation complexity. It requires an access router to maintain the neighborhood update list which includes states for all entries.

Multicast Push-Model If available, IP Multicast [122, 5, 46] might be an option to further increase the efficiency of the data transfer in the push-model. Since update messages with identical content need to be sent to every peer on the neighborhood update list, multicast routing would allow an access router to simplify the synchronization process. By using multicast routing, an access router could simply send update messages to a distinct multicast group instead of maintaining a neighborhood update list. Access routers that are interested in updates can subscribe to the multicast group for as long as they want to receive the updates. However, this requires multicast routing to be available for every router in a neighborhood which might not be the case in real world deployments. Moreover, while multicast routing can reduce the implementation complexity of updates in the push-model it has an inherently higher deployment complexity with regard to the underlying network compared to networks without multicast support. The increased efficiency of multicast routing is also limited by the fact that the set of neighborhood access routers is highly individual for every access router. As explained in Section 3.2.3 every access router is forming its neighborhood only with its direct neighbors. Except in scenarios where only a small number of access routers are deployed in a way that makes them all direct neighbors of each other, access routers commonly have a unique neighborhood. To use multicast groups most efficiently, every access router would require its own multicast group. An access router would use its individual multicast group to send mobile node updates and each of its neighbors would need to subscribe to this multicast group. Depending on the neighborhood size this means that a multicast will only have a limited number of subscribers. An alternative would be to increase the number of access routers that use a multicast group to send updates. However, in this case the multicast group will also transport updates sent by peers that are not within the neighborhood of a particular access router which reduces the overall efficiency.

Another issue that needs to be solved when using multicast routing is the initial synchronization that is supposed to provide a complete state of the active mobile nodes for a neighboring access router. If the multicast group is only used to send incremental updates either an additional mechanism needs to be employed to provide an initial and complete synchronization, or access routers must do without it. The former case again increases the implementation complexity while the later case increases the number of handovers where the mobile node's information is not available in the local session cache which in turn increases handover latency. The lack of the initial synchronization will become insignificant over time as eventually any active mobile node will be captured by an incremental update. In the case of access routers with a fixed position and long operating times this makes the initial synchronization less important. However, in cases where access routers are moving (e.g. in vehicular networks or vehicle-mounted access points) or have very short operating times (e.g. access routers that are switched on and off dynamically) the time that the particular access router is part of a neighborhood can be greatly reduced. In such cases the initial synchronization is much more important to quickly populate the local session cache in order to reduce handover times. Another alternative is for an access router to use the multicast group to distribute only complete updates instead of incremental updates. However, this would greatly reduce the efficiency of the push-model. Depending on the number of active mobile nodes and their churn rate (i.e. how fast nodes are associating and disassociating with an access router) this approach might even make multicast updates less efficient than unicast ones. Finally, multicast routing also makes it harder to provide security. An access router can directly apply security checks to incoming messages (e.g. subscribe messages) and allow or deny updates based on the result of these checks. In the multicast scenario it is not that straight-forward for an access router to control who receives updates and who does not since it is sufficient to subscribe to the multicast group to receive the updates. In conclusion, multicast routing has limited potential to reduce the implementation and signaling overhead of the push-model. In certain scenarios, especially when multicast is already available in the network, it might be beneficial to use multicast routing. However, in common scenarios the benefits are most probably too limited to compensate for the drawbacks that are associated with this approach.

Pull-Model Compared to the push-model the pull-model is much simpler. In the pull model an access router simply periodically queries all its neighborhood access routers for a session cache update. For this purpose it sends a simple update request message which is answered with an update response message. This model does not require the maintenance of any states such as the neighborhood update list in the push-model. An access router interested in updates just needs to maintain a generic timer and an access router that is queried for updates does not need to maintain any per-node states at all. To increase the efficiency of the pull-model, updates can be incrementally instead of complete. While a complete update would include any active mobile nodes that are currently associated with an access router, an incremental updates only includes changes such as new nodes that have been associated with the access router and old nodes that have been disassociated since the last update request. For the purpose of incremental updates, the querying access router must include an identifier in the update request message which indicates the time since its last update. The identifier is issued by the queried access router as part of the previous update response message and is opaque to the querying access router. Depending on the implementation the identifier can be, for example, a simple local timestamp or a counter. It is not overly complex for an access router to track newly arrived nodes using the timestamps that are part of its local session cache entries. On the other hand, nodes that have been disassociated will usually be removed from the local session cache. There are two options with regards to disassociated mobile nodes and incremental updates. If they should be included in updates an access router needs to keep track of them, for example, using a corresponding flag in the local session cache or a dedicated internal list. Using this option the disassociated nodes can be included as entries that are marked as no longer valid when an access routers compiles an incremental update response message. Another option is to use soft-states for session cache entries that are received via an update response message and simply let them time out. This option requires the queried router to augment cache entries in the update response message with a time period that the corresponding entry is valid for. After the given time period the querying access router must consider the corresponding cache entry as stale and remove it. To refresh a cache entry the queried access router simply includes the corresponding entry in a following incremental update response message which refreshes the entry in the queried router as current.

The overhead of the pull-model depends significantly on the interval that access routers request cache updates. When using incremental updates the overhead is not as high as when using full updates because it is limited to signaling messages. However, in general, a larger polling interval incurs less overhead. On the other hand, the polling interval should be small enough that any mobile node that is moving straight through the service area of a neighboring node is captured by an update request before it reaches the querying access router. If a mobile node is moving faster through a neighborhood than the polling interval, it will incur a large handover delay as its configuration data is not yet locally available. Hence, an ideal polling depends on the specific deployment scenario. For example, in a scenario with very fast moving nodes (e.g. cars or trains) and a limited coverage area of each access router the polling interval needs to be much smaller than in a scenario with slow moving mobile nodes (e.g. pedestrians) and a large coverage of each access router. Assuming a vehicular network scenario where cars move with a speed of 120 km/h on a highway that is serviced by access routers with an average coverage radius of 100m it will take the car only 6 seconds (assuming it directly passes through the center) from the edge of the coverage area of a neighboring access router until it is at the outer edge of its coverage area and about to handover.

4.4. Optimizations

The ARA framework is designed to provide decentralized mobility management that is implemented by access routers in a common IP environment. Based on the design it does not require any support from any other nodes in the network such as intermediate routers, the mobile node, or correspondent nodes. However, the performance of certain functions might be optimized if this design rationale is relaxed. This section will discuss a number of optimizations with regards to routing and handovers under the assumption that other nodes in the network can be involved in mobility management functions. Any of these potential optimization are purely optional and do not intend to change the core design of the ARA framework. Arguably such optional modifications that are intended to improve the performance of the framework violate the design goal of not requiring changes to intermediate nodes or corresponding nodes. However, even such modifications would not decrease the compliance with the design goal of minimizing bottlenecks and single points of failure. First, modified routers or end hosts would not provide any additional bottlenecks as traffic is not artificially but naturally routed through them. And second, they either already are single points of failure (i.e. single, non-redundant routers) or they fail gracefully with regards to ARA (i.e. they do not perform route optimizations anymore but they do no prevent communication).

4.4.1. Routing

Per design ARA does not have any impact on a mobile node as long as it is associated with its session access router. However, when a mobile node roams away from its session access router a data forwarding is established that relays all of the mobile node's communication traffic from the session access router to the current access router. This data forwarding introduces a routing inefficiency as data is no longer routed on a straight path between the correspondent node and the mobile but instead is redirected via the session access router where the mobile node is topologically anchored (i.e. the mobile node's IP address belongs to the network that the session access router is the route destination for; see Figure 3.1 for an overview of routing paths in the ARA framework). The routing inefficiency can be measured in additional hops that the indirect routing path (i.e. routing via the session access router) is longer than the direct routing path (i.e. routing straight from the correspondent node to the mobile node). The amount of additional hops introduced is highly dependent on the specific topology that forms the basis of the ARA-enabled network and as such cannot be reliably estimated. However, as the mobile node is physically moving between access routers it stands to reason that their geographical proximity to a certain degree presents an upper bound to their topological distance. More precisely, two nodes that are located in the same geographic region are more likely connected via a regional, national, or even continental Internet exchange point (IX) than via a global one that is located on another continent. Overall the routing inefficiencies introduced by ARA are not likely to be substantial enough to effectively impair ongoing connections. Nonetheless, deviating from the optimal routing path does introduce additional overhead in terms of latency and network load. This section will discuss a number of approaches to optimize the routing of traffic destined for mobile nodes.

Reset Session Access Router in Idle Periods

Similar to the approach in [152] idle periods of the mobile node can be used to reset the mobility support and to re-assign the current access router as session access router. Idle periods mean that the mobile node has no established connections. This approach would require some sort of connection tracking by the session access router to determine the status of a mobile node's connections. If the session access router determines that no established connections exists the current access router can reset a mobile node's mobility support and assign itself as session access router. This entails providing the mobile node with a new IP configuration that belongs to a local network managed by the new session access router. Such a switch could either happen during a handover where the (old) session access router would simply signal the new access router that it has been registered as session access router or it could happen in between handovers. In the latter case a special signaling message would be sent by the session access router to the current access router triggering the reset. The drawbacks of this approach are that connection tracking is non-trivial (esp. for stateless protocols such as UDP), additional states and management operations are introduced, and such a reset would also prevent any correspondent node to establish a new connection to the mobile node under the previous IP address.

Network Address Translation for Mobile Nodes

Without changes to additional nodes such as corresponding nodes or intermediate routers traffic for a mobile node will always be routed towards the router that is responsible for the network that the mobile node's IP address is assigned from. In ARA this would be the session access router which in turn is responsible to forward the data to the current access router. An approach to optimize the routing path in a way that traffic is routed directly from a correspondent node to the current access router is to allow access routers to apply network address translation (NAT) [147] to outgoing traffic of a mobile node. By substituting the source IP addresses of a mobile node's outgoing traffic with an address that is assigned to the current access router, return traffic would be directly routed to the current access router instead of the session access router. Re-substituting the destination IP address of a mobile node's incoming traffic with the address originally assigned to the mobile node by its session access router makes the process transparent to the mobile node.

The ARA/NAT variant has the following workflow: (1) When a mobile node associates the current access router must synchronize any existing NAT mapping in addition to the mobile node's IP configuration. Just like the IP configuration this information can be previously synchronized via neighborhood updates. As before, the mobile node gets configured with the IP configuration that was originally assigned by the session access router which makes the movement between networks transparent to the mobile node's network layer. (2) In addition to initiating data forwarding with the session access router, data forwarding needs to be initiated with every access router that has an active NAT mapping for the mobile node. (3) Any source address of outgoing traffic is subject to NAT mapping. If an active mapping already exists for the particular destination it needs to be applied and the original access router that created the mapping needs to be informed that the mapping is still in use and must not expire. If no active mapping exists, a new mapping must be created using an IP address that will be routed back to the current access router. (4) Any destination address of incoming traffic is subject to (reverse) NAT-mapping according to the existing mappings. Similar to the outgoing NAT the original access router needs to be informed that the mapping is still is use and must not expire. (5) If the mobile node moves to another access router any NAT mapping that was created at the access router must be maintained until a timeout occurs. Additionally data forwarding for the existing NAT mappings must be setup once the new access router sends a corresponding request.

While this approach can effectively optimize routing of traffic between a mobile node it introduces additional complexity. First, non-session access routers must assume responsibility for any connections that are initiated while the mobile node is connected to the access router. They are responsible for these connections even after the mobile node moves to a different access router as the connections are using a (translated) IP address that is anchored with the access router. Second, NAT mappings need to be synchronized much like the IP configuration of a mobile node. However, the complete set of NAT mappings for a mobile node is much more dynamic and extensive than its IP configuration data. Another disadvantage is that the ARA/NAT variant might implicitly impose common drawbacks of network address translation such as limited reachability for nodes behind the NAT from the outside. However, first of all in ARA/NAT a mobile node still retains its original IP address that is anchored at the session access router and globally reachable. In fact, the IP address that was assigned by the session access router is the only IP address that the network layer of the mobile node is made explicitly aware of via the mobile node's configuration. Second, the degree of reachability for nodes behind NATs is based on the actual NAT mechanism that is used. While Network Address Port Translation (NAPT) can limit the reachability of mobile nodes if no explicit mapping exists Basic NAT provides a direct mapping between an two sets of IP addresses [147]. In the latter scenario, reachability of a mobile node is not restricted as both the IP address assigned by the session access router and the IP address that is used for the NAT mapping are globally reachable.

In conclusion, ARA/NAT can potentially greatly optimize the routing efficiency because every newly initiated connection is anchored at the current access router instead of the session access router. The drawbacks usually involved with NAT seem negligible in the ARA scenario as the mobile node always retains a globally reachable address. However, the approach introduces a great deployment complexity. Moreover there are open issues such as the question whether a great deal of NAT mappings can be synchronized efficiently enough during a handover.

ARA-enabled Intermediate Routers

Another approach to route optimization would be to enable intermediate routers to forward traffic directly to the current access router of a mobile node instead of the session access router. An ARA-enabled intermediate node that has knowledge about the current access router of a mobile node can directly forward the mobile node's traffic to the particular access router much like the session access router does. With regards to deployability it is not necessary to enable every intermediate node in this way. A small subset of ARA-enabled intermediate nodes, for example strategically placed in peering points, could optimize the routing path in an early stage. After an ARA-enabled router has applied the optimization once and knows the current access it can tunnel a mobile node's traffic directly to its current access router and subsequent routers do not need to give any special considerations to the traffic anymore. Such a deployment strategy would be similar to the one proposed in [163] where distributed home agents form what the authors call a Global Mobility eXchange (also see Section 2.2.1).

For an ARA-enabled router to be able to perform route optimizations there are two challenges to overcome: First, an ARA-enabled router needs to be able to identify traffic that should be optimized (i.e. traffic that is destined for a mobile node) and second, the router needs to be able to determine and track a mobile node's current access router. If those two challenges can be solved, an ARA-enabled router can tunnel traffic for a mobile node directly to its current access router much like the session access router would do. Addressing the first challenge, determining which traffic is to be subjected to router optimizations, can be done probabilistically or deterministically. Using a probabilistic selection strategy, an access router tries to perform traffic optimization on random traffic flows. If traffic optimization is successful (i.e. the destination of the flow is a mobile node) the particular traffic flow can be tracked and continuously optimized. A deterministic selection strategy on the other hand leverages flow data to determine whether a flow can be optimized. Such flow data can be source or destination addresses or other header data such as dedicated TCP or IP options. For example, an ARA-enabled router can perform traffic optimization for traffic flows that are originating from a known host that serves (largely) mobile nodes, for flows that are destined for an address out of a known network range for mobile nodes, or for flows that have a particular IP or TCP option set. Probabilistic traffic optimization seems to be more suited for scenarios where either most of the traffic is known to be susceptible to traffic optimization or where no data can be leveraged for a deterministic selection. Also, any attempt to perform traffic optimization creates overhead which is another drawback of the probabilistic selection strategy that becomes more severe with an increasing number of flows that would result in a failed optimization attempt. While deterministic flow selection can potentially yield better results it also requires specific information as the basis for the deterministic traffic selection. As already mentioned this can be information about source or destination IP addresses either of hosts that serve (predominantly) mobile nodes or of known networks ranges that are used for mobile nodes. Another possibility is to utilize specific protocol options (e.g. IP options or TCP options) to mark flows. Such protocol options could be either set by the mobile node, the corresponding node, or the ARA node that is currently handling the mobile node. An ARA-enabled router can monitor traffic flows and perform route optimization on any flow that has the particular protocol option set.

The second challenge, determining and tracking the current access router of a mobile node, is much more straight forward compared to traffic selection. Once a traffic flow has been selected for route optimization an ARA-enabled router can use the global session cache to look up the session cache entry of the mobile node much like an ARA node does when a mobile node associates. As the session cache entry includes the address of the current access router the router can setup a direct tunnel to the current access router for the corresponding flow data. When a mobile node moves to another access router while a route optimization tunnel exists, the tunnel endpoint must be adjusted to the new current access router. To notice a movement of the mobile node an ARA-enabled router can synchronize with the session access router or global session cache similar to the synchronization that is being done in neighborhoods (see Section 4.3.2 for a detailed discussion about state synchronization in neighborhoods). For example, a router could be informed via a push mechanism whenever a mobile node moves and the route optimization needs to be adjusted to a new current access router. Such a push mechanism can either be implemented via the global session cache or via the session access router.

In conclusion route optimization in intermediate routers seems promising if traffic that can be optimized with regards to ARA can be somewhat reliably identified. Because this form of route optimization is completely optional no negative impacts can be expected if the route optimization fails of even if a router that performs such route optimization fails. An issue that has to be solved is that this form of route optimization potentially introduces additional packet loss during a handover as data might be send to the previous access router before the router can update its forwarding. A solution would be to implement a data forwarding from the previous access router to the current access router for a short time after a handover.

ARA-enabled Correspondent Nodes

Similar to ARA-enabled routers, route optimization could also be performed by ARA-enabled correspondent nodes. The mechanics are the same as with ARA-enabled routers. Additionally, an ARA-enabled correspondent node might be able to leverage application specific data for flow selection. For example, if a correspondent node hosts some applications that are specifically designed for mobile users, route optimization might be applied to any flows that can be associated with these particular applications. Route optimization based on correspondent nodes is somewhat similar to the route optimization that Mobile IP provides. However, unlike in Mobile IP, the mobile node itself is not directly involved in the process. Route optimization is initiated by the correspondent node and carried out with support from the network. This approach to route optimization seems more interesting for nodes that predominantly serve mobile hosts. However, it can be argued that even in this case the deployment of an ARA-enabled intermediate router that performs route optimization on flows involving the particular host is more flexible. The same conclusions with regard to packet loss during a handover as in the previous section apply.

4.4.2. Handovers

Handovers are a critical part of any mobility management system. ARA provides a basic framework to conceptually handle handovers without the involvement of the mobile node or any other network entity. However, numerous proposals have been made with regards to handover optimizations that might be transferable to ARA.

Make-Before-Break Handovers

Many handover methods assume that a mobile node can only be connected to one access point at a time because only a single interface is available. Under this assumption a mobile node has to break an existing connection with an access point in order to establish a new connection with another access point. However, if a mobile node can connect to multiple access points at the same time it can wait until the connection with a new access point is established before breaking the connection with the old access point during a handover. This approach is called make-before-break and requires multiple interfaces or one interface that can be used to connect to multiple access points (e.g. [27, 108, 65]). Make-before-break approaches have been proposed and evaluated for multiple link layers and mobility solutions (e.g. [117, 118, 125]) and the principle is easily transferable to ARA. Assuming a mobile can be associated with multiple access routers, it just needs to maintain its connection with the previous access router until data arrives via the new access router. After a short grace period the mobile node can be certain that the handover is complete as data arriving via the new access router is a clear indication that the session access router has changed the forwarding which concludes the handover process (assuming the mobile node has been assigned a valid IP configuration already). Since in this approach a connection is maintained over multiple interfaces the mobile node needs to have some mechanism to transfer or multiplex one connection between them.

Predictive Handovers and Simulcasting

Mobile IPv6 Fast Handovers (see Section 2.2.1), Fast Proxy Mobile IPv6, and Proxy Mobile IPv6 with Simultaneous Bindings (for both see Section 2.3.1) leverage triggers from the mobile node to improve handover performance by preparing a handover procedure

Table 4.1.: Basic elements of a session cache entry and their respective data size.

Entry	**Size (B)**
Identifier (MAC address)	6
Session Access Router	16 (IPv6) + 2 (port no.)
IP Address	16 (IPv6) + 1 (prefix len.)
Gateway Address	16 (IPv6)
Sum	**57**

before a mobile node actually does handover. Furthermore, they introduce simulcasting (or bicasting) between a previous access router and the current access router to limit packet loss during a handover. Both concepts are easily applicable in ARA as well. Predicting a handover alone does not benefit ARA if a working neighborhood has been established. In that case the access router already has all the information it needs about a mobile node. While predicting a handover could provide some time to prepare a data forwarding between access routers this process is currently not considered time consuming as simple IP in IP encapsulation is suffice. However, other tunnel methods might require a tunnel setup with a bidirectional handshake in which case a predictive handover can provide the necessary time for the procedure. On the other hand, simulcasting data from the current access router to the mobile node and concurrently to the handover candidate access router has the same results in ARA as in Mobile IP or Proxy Mobile IP, namely reducing the packet loss during a handover. Simulcasting does pose some problems with regards to duplicate packets as an access router after a handover occurred has no means of determining which packets the mobile node already received. However, these issues are the same for ARA, Mobile IP, and Proxy Mobile IP. It can be assumed that simulcasting has the same advantages and disadvantages in any approach.

4.4.3. Reducing Signaling Overhead

The ARA framework uses signaling messages to exchange information about mobile nodes. Without anticipating the evaluation on signaling overhead in Chapter 6 this section will provide a theoretical discussion about the estimated overhead and present approaches to reduce the signaling overhead. Neighborhoods will be the focus of the discussion as they can be expected to contribute most of the overhead. However, most of the considerations apply to session updates as well.

Neighborhoods are used to synchronize data about active nodes and as such essentially synchronize local session cache entries. Therefore, the amount of exchanged data is based on the elements of a session cache entry. The basic elements of a session cache entry with their respective sizes in bytes are listed in Table 4.1 (see Section 4.1.2 for more details about session cache entries). The table includes all the data that is needed to provide a mobile node with a basic IPv6 configuration[1]. Although the current access router is also part of a session cache entry it is not necessary to transfer this information with every entry as it is implied during a synchronization. An access router will only send session cache updates with date of mobile nodes that it is itself the current access router for. An access router receiving session cache updates can, therefore, just record the sending access router as the current access router for any particular session cache entry that is included in the update message. The only exception are potential updates that indicate that a session cache entry is no longer valid during incremental updates (see above) in which case the corresponding local session cache entry is deleted. Also not part of neighborhood updates is the node type as only entries for neighboring nodes are exchanged which implicitly defines the node type. The data listed in Table 4.1 adds up to 57 Byte but is only a very basic set of configuration data. Depending on the scenario, supplemental configuration data such as DNS/WINS server entries, a host name, or proprietary data might also be included in session cache update messages. However, a rough estimate can be made based on this data that a single session cache update is in the order of magnitude of 60 to 200 bytes.

Besides incremental updates a number of techniques can be used to further decrease the amount of data that is transferred during a cache synchronization such as data compression, omitting data fragments that have not changed, or only sending session cache entries of mobile nodes that are likely to handover. The efficiency of lossless data compression algorithms (e.g. algorithms based on Lempel and Ziv [174], prediction by partial matching [32], or the Burrows–Wheeler transform [20]) is highly dependent on the redundancy of the source data. Session cache entries might include redundant data, since MAC addresses and IP addresses can share common prefixes. 48-bit universal LAN Media Access Control (MAC) addresses which are used in Ethernet have a three byte prefix. This Organizationally Unique Identifier (OUI) identifies the organization that provisioned it [154]. As this is usually the manufacturer or the organization that commissioned the production of the network interface card such prefixes have an inherent redundancy. Similarly, IP addresses share a common prefix if they are part of the same sub-network. This will usually be the case if mobile nodes share the same access router as they were most likely provisioned with an IP address from a common network

[1] While an IPv4 configuration is also possible with ARA, IPv6 addresses are bigger and therefore provide an upper bound.

that is managed by the access router. Furthermore, the addresses of particular session access routers and default gateways will commonly be identical between mobile nodes that share the same session access router. Based on these redundancies that might occur in session cache entries, compression can be an efficient method to reduce transferred data. However, updates with only a small number of session cache entries and especially incremental updates might lack a critical mass of redundant data. Furthermore, compression algorithms require complex arithmetic operations and memory capacity so there is also a trade-off to consider with regards to available processing power and available memory capacity. A further evaluation of the effectiveness of compression to reduce the synchronization overhead of ARA will not be conducted at this point as it is considered out of scope of this thesis.

Omitting data fragments that have not changed in an update response message is an orthogonal strategy to reduce the synchronization overhead in the push model. Instead of transferring complete cache entries, the sending access router reduces the data send to just the data elements that have actually changed. However, in ARA a cache entry is mostly static data that does not change over time. The session access router or IP configuration of a mobile node is fixed for the duration of the session and the only variable is whether a mobile node is active within a particular neighboring access router or not. Still, partial updates can be used, for example, in the scenario described above where an access router regularly re-sends an update for the same mobile node to prevent a soft-timeout of the particular entry. In this scenario sending a partial instead of a complete cache entry means that the access router limits the data that is included in the update to just the mobile node identifier and, depending on the scenario, a timestamp. Since the update is just meant to refresh the timestamp of the cache entry on the receiving access router, other cache data such as the address of the mobile node's session access router or its IP configuration can be omitted to reduce the size of the update. However, when only sending partial cache entries the sending access router must be sure that the receiving access router already has the complete cache entry available since the partial update on its own is insufficient to perform a hand-over for the mobile node. In the pull model with incremental updates this can be easily implemented by reusing the same timestamp that is used to indicate the point in time of the last update. Especially in the strategy using incremental updates without initial complete synchronization this can not be guaranteed without a sending access router keeping states for each cache entry and for each receiving access router. Using incremental updates with an initial complete synchronization on the other hand does implicitly ensure that the receiving access router has a complete cache entry. Therefore, the strategy to send partial updates is well suited for this scenario. Partial updates are not suited to reduce the synchronization overhead in the push model that uses hard states or in any of the pull models. In those scenarios, cache entries are not transferred more than once which makes partial updates useless.

Another approach to reduce the synchronization overhead is to directly reduce the number of cache entries that are considered to be included in update response messages. Normally, the local cache entry of every mobile node that is associate with an access router must be considered for synchronization. However, realistically the receiving access router only requires cache entries of mobile nodes that will be handing-in within the particular update interval. Hence, cache entries of mobile node's that will not hand-in to the receiving access router could be completely omitted to reduce the synchronization overhead. Unfortunately it can be challenging for an access router to determine the exact point in time when a mobile node is about to handover. Furthermore, in the pull model an access router cannot be sure when the querying access router will perform its next update request. Also, most of the limitations that have been discussed above with regards to the on-demand synchronization model apply as well. Theses facts make it difficult to effectively limit the number of cache entries that are considered for synchronization. Moreover, if all of the requirements are met to effectively and accurately identify mobile node's that are about to hand-over it seems to be more favorable to generally chose an on-demand push model as the basis for synchronization.

5. Deployment Considerations

For a system to be considered for deployment a number of requirements must be met. Not all of these requirements can be directly derived from the system design that was introduced in the previous chapter. Before the quantitative evaluation is presented in the next chapter, this chapter will discuss a number of qualitative considerations with regards to the deployability of ARA.

5.1. Incremental and Mixed Deployment

ARA provides a very flexible location management and forwarding framework which allows for an incremental or mixed deployment with other mobility support solutions. The ARA architecture focuses on inter access router mobility. How an access router provides mobility inside its domain is completely transparent to ARA. For example, an access router can have multiple radio interfaces for which it provides link layer mobility support. In the same sense can an access router use other mobility solutions to provide mobility support for its mobile nodes within its domain. In this case, the ARA protocol is only used when the mobile node leaves this domain.

An example for a deployment which integrates Mobile IP and Proxy Mobile IP into an existing ARA network is shown in Figure 5.1. Consider the following example: A mobile node initially attaches to access router AR_1 which becomes the nodes session access router. Now the mobile node moves to AR_2 and the ARA protocol initiates the data forwarding from AR_1 to AR_2. The mobile node continues to move into another domain which supports Proxy Mobile IP for intra-domain mobility support. The local mobility anchor of the Proxy Mobile IP domain (AR_3) is also part of the ARA network and uses the ARA protocol to initiate data forwarding from AR_1 to itself. As long as the node moves within the Proxy Mobile IP domain, the local mobility anchor will take care of the mobility support. There is no need to employ ARA. In the next step, the mobile node moves on to a domain which does not support any mobility. As a fallback the mobile node employs the Mobile IP protocol and connects to its home agent (AR_4). The home agent supports ARA and initiates the data forwarding from AR_1 to itself. It then forwards the data to the current point of attachment of the mobile node. The ability of ARA to interconnect different mobility support solutions to provide an inter-domain mobility

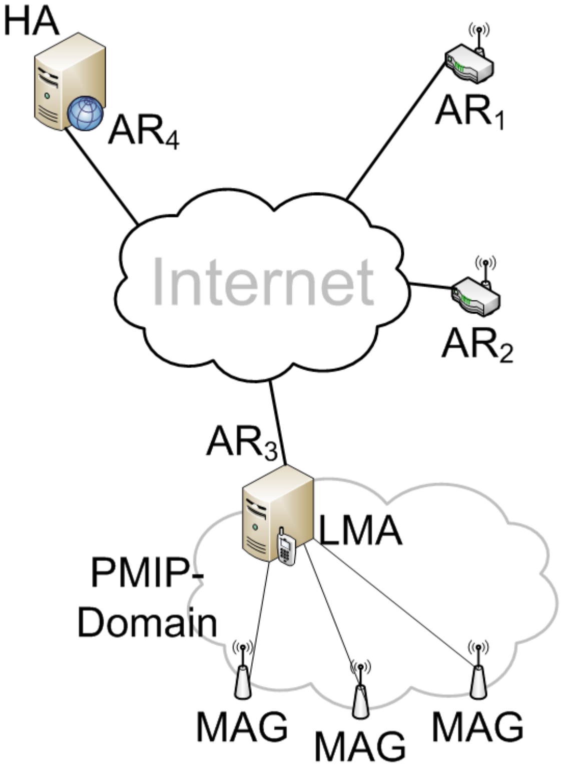


Figure 5.1.: Mixed ARA deployment with Mobile IP and Proxy Mobile IP

support makes it very versatile. It allows for mixed deployments as described above or for incremental replacements of existing deployments. In an incremental replacement strategy an existing mobility solutions can first be adapted to be embedded into an ARA network. In subsequent steps the existing system than could be replaced by ARA nodes incrementally.

5.2. Security

The introduced ARA framework provides network based mobility management functions. In particular it allows for ARA-enabled access routers to forward a mobile node's traffic from its initial access router to its current access router. This section will discuss any security considerations regarding the ARA framework itself or regarding additional risks that may be introduced by the ARA framework in networks where it is deployed.

5.2.1. Mobile Node Security

The purpose of the ARA framework is to provide mobility support for mobile nodes. As such, mobile nodes are the subject of the framework. However, as a design goal ARA does not require any changes to the mobile node's and does not actively involve the mobile nodes in any ARA-specific operations in addition to the normal operating procedures a mobile node performs in its network environment. Such normal operating procedures include, for example, associating with access routers or performing IP configuration via DHCP or IPv6 Autoconfiguration. Because ARA does not introduce any changes to mobile nodes it also does not introduce additional security risks with regards to a mobile node's protocol stack.

5.2.2. Network Access

Depending on the link layer access technology used within an ARA deployment, ARA may require certain concessions in order to be able to provide network based mobility management. This is most notably reflected in the fact that in order for network based mobility management to work and in order to prevent a mobile node to reset connections when moving between access routers, any change in network connectivity must be transparent to the mobile node. Specifically this means that certain configuration parameters must be consistent between every access router. For example, in IEEE 802.11 networks, the service set identifier (SSID) is commonly used as an identifier for a particular wireless network. If two access points do not share the same SSID a mobile node might assume they do not belong to the same network and reset any ongoing connections when associating with an access point that has a different SSID. Similar limitations can apply for other configuration parameters such as encryption keys. The principle of making changes in network attachment transparent to a mobile node also prevents switching configurations during a handover, for example encrypting data with a more secure encryption scheme in case the new access router supports it. None of these configuration parameters have any requirements of secrecy and must be known or inferable, in order for a mobile node to be able to establish network connectivity. Furthermore, any of these limitations are common for all network based mobility support solutions and are not unique or original to ARA. However, in an ARA deployment such configuration parameters can be distributed over a much larger number of nodes and administrative domains than in common scenarios which might lead to difficulties updating the configuration parameters in a timely and consistent manner. Moreover, it can introduce security risks because all of the nodes within an ARA deployment have to be operated with a commonly supported configuration set. Such a configuration might not reflect the most secure operating mode but rather a less secure operating mode that is supported by all involved access routers instead.

5.2.3. Mobile Node Authentication

Independent of basic network access an ARA deployment can employ mobile node authentication. While link layer specific authentication mechanisms can be used (e.g. IEEE 802.11i [155] or IEEE 802.1X [157]) they would limit any ARA deployment to the specific link layer. Therefore, mobile node authentication on the network layer is preferred unless the ARA deployment is specific to a certain link layer technology. For mobile node authentication on the network layer for example the Extensible Authentication Protocol (EAP) [3] can be used. More elaborate authentication architectures similar to [48] are also possible as long as they build on top of IP. However, to provide effective security the mobile node authentication has to be repeated every time the mobile node hands over to a new access router. Depending on the employed authentication method it is likely that the mobile node will need a valid IP address during the authentication procedure. Therefore, at least the IP configuration has to be completed. While the authentication procedure can be performed in parallel to setting up the data forwarding it must be ensured that the authentication procedure does not introduce additional latencies that are too long for a seamless handover. Especially authentication mechanism that involve interaction with the user such as username/password authentication cannot be employed during a handover. However, they might be employed during the initial association of a mobile node with the network as in this point of time no ongoing connections are established yet.

5.2.4. Global Session Cache

The global session cache is primarily used by ARA-enabled access routers when a mobile node hands-in from an access router that is not yet in the local neighborhood. In such a case the access router will use the global session cache to look up configuration details about the mobile node that would otherwise be transferred via neighborhood updates. As such the global session cache has a critical role during the start-up time of an access router (as neighbors are not discovered yet) and in ARA deployments where neighborhoods are very volatile (e.g. vehicular networks or vehicle-mounted access points). With an increasing period of operation it can be expected that more and more information is exchanged via neighborhoods than via the global session and as such the overall impact of attacks on the global session cache decreases. Nonetheless, a number of security threats with regards to the global session cache will be discussed in the following.

Service Failure

Depending on the deployment scenario (see Section 3.2.2) the global session cache can fail completely or partially, for example due to a denial of service attack or due to hardware failure. In case of such a failure, access routers are no longer capable of querying session data about unknown mobile nodes and queries will timeout or result in an error. For the following discussion, it is assumed that access routers are configured to fall back to creating a new IP configuration for unknown mobile nodes when the global session cache is unavailable. The reason for this assumption is that the only alternative is to not provide a mobile node with a configuration at all. This would effectively prevent the mobile node from getting any network access at all which seems to be the worst case. Providing a mobile node with a new IP configuration might break any existing connections, however, the mobile node does have basic network connectivity.

A mobile node that is unknown and therefore requires a global session cache lookup can either be a new mobile node that is initially associating with the network or it can be an existing mobile node that is handing-in from a neighboring access router. In the first case, where the mobile node is a new node, the node is unknown because no configuration has been created yet. A failure of the global session cache, therefore, can only delay the initial configuration of a mobile node as a new configuration will be created eventually. Assuming a working neighborhood, the new configuration will be shared with any neighbors and regular ARA operations will proceed even with the global session cache being unavailable. In the second case, where the mobile node is handing-in from a neighboring access router, the mobile node does have an existing configuration and a corresponding session anchor. The mobile node is unknown to the current access router because it did not receive a neighborhood update with regards to the mobile node before the mobile node associated itself. This can be the case either because the previous access router is not in its neighborhood list, hence does not provide neighborhood updates to the current access router at all, or because the mobile node was changing associations faster than the previous access router is sending neighborhood updates. However, the latter case should not happen in a working neighborhood environment that uses a suited neighborhood update mechanism (see Section 3.2.3 for a comprehensive discussion about possible neighborhood update mechanisms). Creating a new configuration for such a node which does have a previous configuration and providing the mobile node with this new configuration will effectively break any established connections on part of the mobile node.

In conclusion, a global session cache failure will only break mobility support when an existing mobile node with ongoing connections hands over to an access router that did not receive a neighborhood update for that particular mobile node before. After an initial phase of neighborhood discovery when an access router initially starts its operations and in networks that do not have a high volatility of access routers this case can be considered rare. As a matter of fact, the low dependency of access routers on the global session cache during normal operations reflects the design goal of ARA to minimize bottlenecks and single points of failure (see Section 3.1.2).

Eavesdropping

The global session cache provides access to the combined data of all session access routers and as such can provide detailed information about any active mobile node. For an unauthorized node it may be possible to obtain such information by eavesdropping on the communication of legitimate nodes with the session cache. An eavesdropper can attempt to either target the global session cache itself or a specific access router. Eavesdropping on the global session cache itself provides global information but can be inherently difficult depending on the deployment scenario. While it might be feasible to eavesdrop on a single server in a server assisted ARA deployment it is much more difficult to eavesdrop on a global session cache that is deployed via a distributed hash table (DHT) in a distributed ARA deployment. However, even in a DHT based deployment, eavesdropping is possible at least on a subset of the global session cache (e.g. just one DHT node). Eavesdropping on a specific access router, for example by an intermediate node between the access router and the global session cache, can be easier to perform but provides only localized information about mobile nodes that are currently managed by the particular access router.

Session cache information gained by eavesdropping contains data about a mobile node's current access router, its session access router, and its IP configuration. The individual information about a mobile node's current access router can be leveraged to infer the mobile node's approximate current location, for example by using geolocation techniques (e.g. [50, 167, 113, 67]).[1] In combination with the address of a mobile node's session access router, a rough movement pattern could be deduced as the session access router represents the location where the mobile node initially started its mobility

[1] Note that a mobile node's location cannot be simply inferred by leveraging the mobile node's global address since it might currently not be associated with its session access router. In such a case, an attempt to locate a mobile node based on its global address will yield false results as data for the mobile node is transparently forwarded to the mobile nodes current location. To infer a mobile node's location one must leverage the address of its current access router.

session. If an eavesdropper can obtain continuous information about a mobile node, he can combine this information to create a more fine grained movement pattern of the mobile node. Information about a mobile node's IP configuration can also be obtained by eavesdropping on communications with the global session cache. Such information can be leveraged by an attacker, for example to impersonate an access router towards the mobile node and to launch advanced attacks. A more detailed discussion on impersonation attacks will be conducted in Section 5.2.5.

Eavesdropping can primarily compromise data that is relevant to the location privacy of a mobile node. However, IP configuration data is also observable in such an attack and can be leveraged in other attacks, for example impersonation attacks. To prevent any eavesdropping on communication between an access router and the global session cache, or between different global session cache nodes in a distributed ARA deployment scenario, any communication should be encrypted. Depending on the deployment scenario the communication can be encrypted on the network layer, for example by deploying Internet Protocol Security (IPsec) [70], or on a higher layer, for example by deploying Transport Layer Security (TLS) [38].

Data Manipulation

Information from the global session cache is actively used by access routers to configure a mobile node and to initiate data forwarding with its session access router. If an attacker is able to manipulate session cache data he is indirectly able to influence data forwarding paths and mobile node configurations the next time an access router queries the cache. Data manipulation can be performed either by directly accessing the global session cache much like a legitimate access router does or by altering or discarding session update messages. For example, to perform a simple denial of service attack, an adversary could directly delete session cache entries in the global cache or discard any session update messages that are intended to register or query the session access router for a mobile node. In a more advanced attack, an adversary could register itself as the session access router for any given mobile node with the global session cache and as the current access router with the actual session access router of the particular node. This will allow the adversary to establish itself as a man-in-the-middle the next time an access router queries the global session cache and uses the information to establish a data forwarding. In a more subtle attack, an adversary could use access to the global session cache to obtain data about a mobile node. In this case the attacker would gain the same advantages an eavesdropper has (see discussion in the previous section) but with a finer control over the data he has access to.

To prevent data manipulation an ARA deployment must be able to guarantee access router authentication, authorization, message integrity and message freshness. For distributed ARA deployment scenarios DHT-specific security techniques should be deployed to secure communication within the global session cache (see [159] for a recent survey on DHT security mechanisms).

5.2.5. Impersonation

To gain access to a mobile node's communication data, an attacker might try to emulate a legitimate access router. If the mobile node can be enticed into associating with the malicious access router, the attacker can perform a man-in-the-middle attack on the mobile node's data traffic. Multiple elaborate man-in-the-middle attacks have been discussed for several common link layer technologies (e.g. GSM/UMTS [96, 95]; cf. [138, 88] for IEEE 802.11 "evil twin" wireless access points). Attacks on mobile nodes by such rogue access points are primarily a link layer security issue since the decision whether to associate with an access point or not is usually made by the link layer and numerous solutions have been proposed to identify rogue access points (e.g. [165, 56, 87, 12, 142, 6]). As ARA operates on the network layer there are only limited options to circumvent impersonation attacks from within the framework.

In the context of ARA there are three attack scenarios to consider. In the first scenario an attacker emulates a legitimate access router towards the mobile node but does not perform any actions towards the ARA network. If a mobile node wrongfully attaches to the imposter in this scenario the imposter might be able to provide basic network connectivity but it cannot provide continued mobility support. Even if the imposter could gain access to a mobile nodes configuration data via eavesdropping he would not be able to establish a data forwarding with the session access router. Therefore, any established connections break and no data is disclosed to the imposter from the ARA deployment. However, any future connections that are established or re-established after the mobile node associates with the imposter can be overheard by the imposter. As this scenario is completely outside the ARA framework there is nothing that can be done from within an ARA deployment on a technical bases. On an organizational basis the ARA deployment can enforce a secure link layer technology that provides mutual authentication between access router and mobile node. This might allow a mobile node to realize if it is connecting to an imposter.

In the second attack scenario the imposter emulates a legitimate access router towards the mobile node and the mobile node towards the ARA network. This effectively puts the imposter in the middle between the ARA network and the mobile node and allows him to overhear any data that is exchanged by the mobile node. Because the imposter is emulating the mobile node towards the ARA network the mobility support is still intact.

This kind of elaborate man-in-the-middle attack is not detectable from within the ARA network and can only be averted if strong link layer authentication mechanisms are used that prevent the imposter from emulating a mobile node towards the ARA network.

In the third attack scenario the imposter tries to gain access to the ARA network by emulating a legitimate access router. If he is successful he can act as a legitimate access router towards any mobile node. Furthermore he has full access to the mobility support for a mobile node and its data. This scenario is easily avoidable by employing strong authentication between the access routers for example by deploying Internet Protocol Security (IPsec) [70] or Transport Layer Security (TLS) [38]. However, as ARA is a distributed deployment and neighborhoods are formed autonomously it is not feasible to use pre-shared keys. As a distributed deployment ARA would require a key distribution center or a certificate authority. Both seems feasible to be introduced in an ARA deployment. A certificate authority can be provided by any of the participating operators or a third party. A key distribution center can be implemented on top of the infrastructure for the global session cache.

5.3. Accountability and Charging

As a decentralized approach to mobility management, ARA aims at being deployable in scenarios where multiple operators or providers are involved. In such multi-provider scenarios a user will commonly roam between access routers that are operated by different providers. If one or multiple providers that are involved in an ARA deployment want to charge mobile users for wireless access a common charging model has to be established and the accountability of every participant must be defined and ensured by the system. This section will discuss how accountability and charging can be provided in such an environment. Based on the assumption of inter-operator relations (see Section 3.1.3) charging models and accounting responsibilities can be agreed upon and setup outside of the ARA framework. However, their technical realization must be implementable within the ARA framework.

5.3.1. Accountability

When a mobile node is active in a network and is to be charged there needs to be a clear accountability at any time. A responsible entity must track a mobile user's usage of the network and record this data so that the user can be billed subsequently. Moreover, the user must be able to understand, agree to, and reproduce any charges. Accountability in the context of this thesis means the responsibility to ensure that a mobile node is entitled to use a service and that its usage is properly tracked and cleared with all involved parties.

In a decentralized environment like ARA accountability has to be shared between the involved entities as no central authority is available. While a mobile node is connected to an ARA enabled network, its session access router is responsible for its mobility support. Any mobility operations are coordinated and sanctioned by the session access router via Session Update Request/Response messages (cf. Section 4.3.1). As such the session access router also seems a logical choice to be responsible for any accounting pertaining to the mobile node's usage of network capacities. Besides its inherent potential to track a users network usage the session access router is per design also the initial point of attachment for a mobile node. Any authentication or authorization that might take a noticeable amount of time can be performed at this point as there are no ongoing connections yet. An example could be a simple web-based user authentication using usernames and passwords which could be tied into an authentication, authorization and accounting infrastructure such as RADIUS or Diameter (e.g. [105]). Any other time a mobile node associates with an access router there are potentially open connections that would be impacted by additional latency that is caused by an interactive user authentication or a service-specific authorization method such as an HTTP redirect to a web page with terms of use. Because any user interaction is essentially limited to the initial connection with the network which is handled by the session access router, the session access router should also be responsible for managing a user's rate. For example, the session cache entry can be extended to indicate the rate or tariff that a user has chosen or that the user has been assigned by other means.

Although a session access router can be accountable for a mobile node's overall activity in a decentralized environment there might be multiple administrative domains involved. In this case one administrative domain does not necessarily want to rely on entities in another administrative domain to assume accountability for services rendered within the own domain. For example, when proceeds are distributed based on a mobile node's online time, a domain operator might prefer authoritative statements from its own access routers about the time that a mobile node has spend roaming within his domain rather than relying on data provided by the operator of the domain that the session access router of the corresponding mobile node belongs to. Therefore, in addition to a session access router the current access router can assume accountability for services rendered within its own administrative domain.

5.3.2. Accounting

An important basis for charging a user is the accounting about its usage of a particular service. The metrics are comparable to other Internet access services, for example usage time, or downloaded and uploaded data. Standardized accounting services and protocols such as RADIUS [134, 133] or Diameter [23] can be used to collect this kind of accounting information. Since ARA is a decentralized framework, every single access router must provide such accounting information. While the information can be collected with a centralized accounting infrastructure it is also viable that every administrative domain employs its own accounting infrastructure and that items are cleared regularly among the involved domain operators.

5.3.3. Charging Users

A decentralized system like ARA can only be deployed successfully if it can provide a seamless user experience on the technical as well as on the organizational level. For scenarios where multiple operators cooperate in an ARA deployment this means in particular that a user must be admitted and charged for network usage in a transparent and consistent manner no matter where he initially connects to and no matter where he roams to. Different operators that participate in an ARA deployment can arrange different rates and conditions with their respective customers. However, subsequently the customer must be able to use the ARA deployment as a whole under these conditions. First of all, it might be impossible for a user to influence or even recognize to which domain a particular access point belongs to before associating with it. Second of all and more importantly, if it cannot be guaranteed that a user can roam between domain an important foundation for a cooperative deployment is missing. If a customer is denied network access by an access router of a domain while he has a valid contract with the operator of another domain the ARA deployment is pointless as seamless roaming is not possible. Therefore it is important for a cooperative deployment that the involved operators can agree on an organizational framework before the actual deployment is performed.

Charging models between operators and users can be individually arranged as long as every operator in an ARA deployment admits users that have a valid contract with any participating operator. Therefore any valid charging model such as volume based charging, time based charging, or flat-rate charging can be employed. How charging models can be calculated is outside the scope of this thesis. However, it can be expected that in a distributed environment inter-operator charges must be factored in as a substantial portion of an operator expenses.

5.3.4. Inter-Operator Charging

As established in the previous section a user must be able to associate with any access router in an ARA deployment. This explicitly includes access routers that are operated by a service provider that a user might not have any user contract with. However, as operators of such access routers provide resources for a user they might require a compensation from the operator that the user does have a user contract with. A number of models can be considered for such inter-operator charges for roaming users. Assuming that the operators agree to charge each other rather than providing network access as a mutual beneficial service charging can be simply done based on the access router capacities that are contributed. For example, a fee is charged from every participating operator and the overall receipts are divided among all participating operators based on the individual ration of operated access routers. More sophisticated charging models can incorporate a mobile user's consumption data such as online time or expended data volume. Such a model would require access routers to track a mobile node's usage data (see Section 5.3.2). Based on this data an operator can charge another operator for the resources that he provided to a mobile user.

6. Evaluation

To justify the system design of ARA as introduced in the previous chapter, evaluations in practical terms such as scalability and performance are necessary. While qualitative aspects such as deployability, security, or accountability have been discussed in the previous Chapter, this chapter will introduce the evaluations that have been performed with regards to quantitative aspects such as signaling overhead, handover delay, and system performance.

6.1. Methodologies

A system design can be evaluated with numerous methods that usually present a tradeoff between feasibility and validity. An extensive field test with an actual deployment can be expected to yield very good results but it also rather complex and costly to perform. Within the scope of this thesis three different methods of evaluation have been applied: theoretical analysis, simulation, and tests based on a prototype implementation. Each of these methods has its distinct setup, values and limitations.

6.1.1. Theoretical Analysis

Theoretical Analysis as an evaluation method can provide numerical estimates based on a theoretical model. As such it is better suited for initial evaluations and to provide estimates about upper and lower bounds of a problem that can be modeled theoretically. Since models are commonly based on assumptions, the explanatory power of theoretical analysis depends on the plausibility and validity of these assumptions. In this thesis theoretical analysis will be used to initially model objects of investigation based on assumptions that were made during Chapters 3, 4, and 5 as well as assumptions that will be introduced during the evaluations themselves. To strengthen the results of theoretical analysis the results will be correlated with results from the other methods, mainly from simulations.

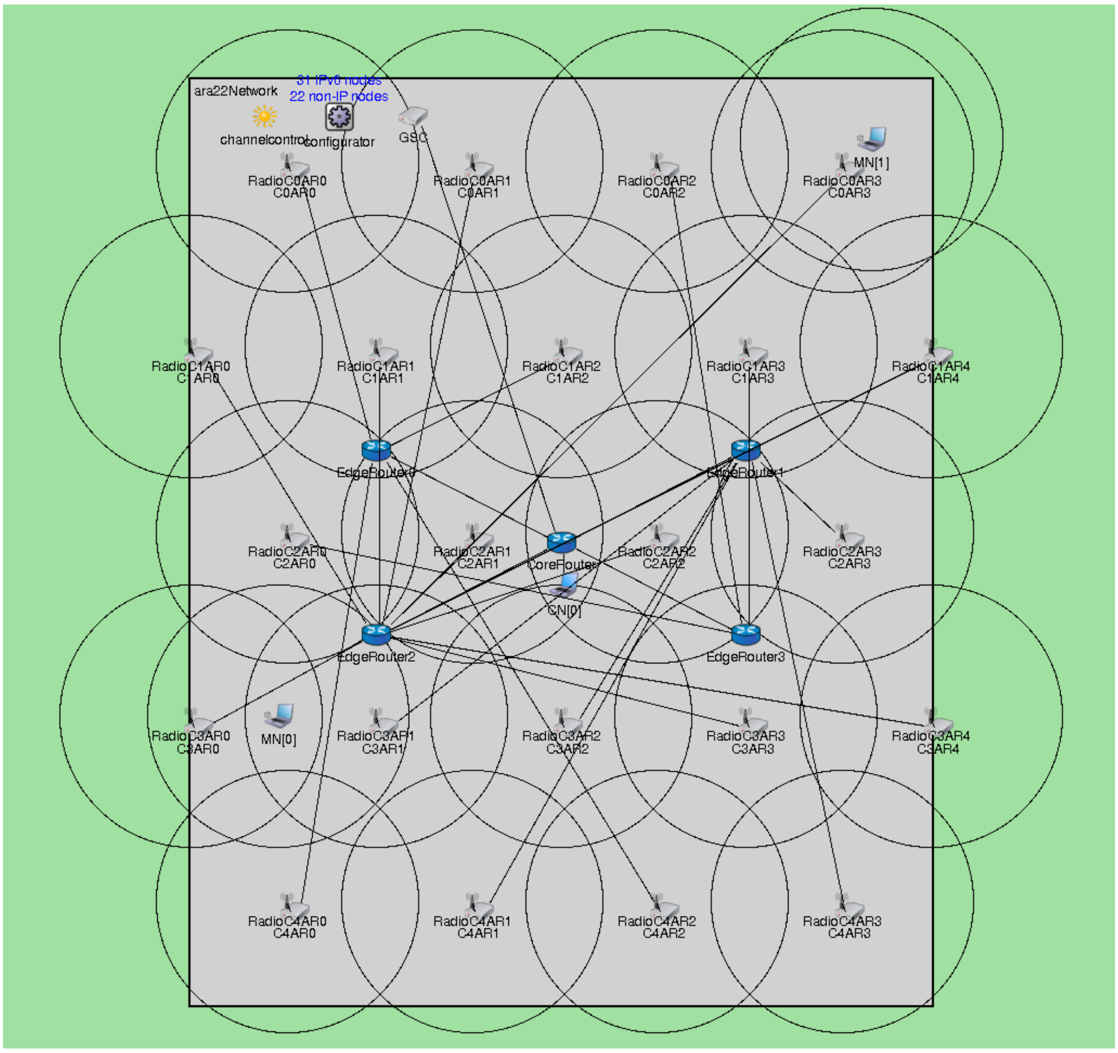


Figure 6.1.: Simulation Topology: 22 Access routers in four domains connected via a common transit network.

Table 6.1.: Simulation and evaluation parameters.

Parameter	**Value**
Number of Access Routers	22
Number of Mobile Nodes	1 to 600
Number of Domains	4
Access Routers per Domain	4/5/5/8
Intra Domain Delay	2 ms
Inter Domain Delay	20 ms
Core Traversal Delay	90 ms
Mobile Node Movement Model	Random Waypoint Mobility
Mobile Node Movement Speed	5 m/s (18 km/h)
Mobile Node Wait Time	10 sec
Simulation Run Repetitions	10

6.1.2. Simulation

The ARA framework has been implemented in the OMNet++ 4.1 simulation framework [161].[1] The implementation is based on the INET Framework, an open-source communication networks simulation package which contains models for the relevant networking protocols such as IPv6 and IEEE 802.11.[2] IPv6 was chosen as the network protocol based on its expected deployment in future networks. However, the results are largely independent of the network protocol and should be transferable to IPv4 scenarios. The simulation studies were performed based on a simple stub-topology shown in Figure 6.1 which consists of 22 access routers in four administrative domains. The domains were connected via a common transit network. The access routers were deployed in a regular pattern to provide seamless coverage within the movement area of the mobile nodes. Each access router has between three and six neighbors.

Unless otherwise noted during the discussion of the evaluation results the simulations and evaluations have been performed based on the parameters listed in Table 6.1. The topology is meant to reflect a small ARA deployment with four operators. The delay

[1] http://www.omnetpp.org/

[2] http://inet.omnetpp.org/

parameters are similar to [173] and based on empirical data from various sources.[3] The mobile node movement speed is meant to average a pedestrian and a vehicle in urban traffic. This kind of average can be used since a change of state in ARA is only triggered when a mobile node attaches to an access router but not while it is moving within the coverage area of an access router (c.f. Section 3.2). Therefore, the only direct impact of the movement speed is its influence on the handover frequency (i.e. the faster a mobile node moves, the more probably are handovers). However, the handover frequency has no direct impact on most of the evaluated metrics such as handover latency or routing efficiency. While it may have an indirect impact on quantitative metrics such as the signaling overhead (i.e. the more handovers occur, the more signaling messages need to be exchanged), the handover frequency is also influenced by other parameters, most notably the coverage area of an access router. Moreover, in such evaluations, the influence of the handover frequency will be commonly put into perspective by normalization so that its impact can be quantified and extrapolated for other parameter values. Consequently, the parameter for a mobile node's movement speed can be considered incidental and is set as an average rather than a variable. The deployment scenario is server assisted with the global session cache being hosted outside the four ARA domains but reachable with the common intra domain delay. This is meant to make the results more comparable to other deployment scenarios as every access router has the same average delay to the global session cache.

6.1.3. Prototype Implementation

To further substantiate the evaluation of the ARA framework and to gain more insights into practical problems with regards to a deployment, the framework has been implemented and deployed in a small testbed. As basis for the implementation commodity wireless access routers (Linksys WRT160N/WRT160NL) have been used which run the OpenWrt Linux distribution[4] for embedded devices. The implementation includes a modified DHCP server and an ARA daemon that runs on the access routers. No modifications have been made to the mobile nodes. The testbed topology as shown in Figure 6.2 consists of four access routers that are deployed in a common network.

[3] http://stat.qwest.net/
http://www.verizonbusiness.com/about/network/latency/
http://ipnetwork.bgtmo.ip.att.net/pws/network_delay.html
http://www.caida.org/projects/ark/statistics/
http://www-iepm.slac.stanford.edu/pinger/

[4] http://openwrt.org/

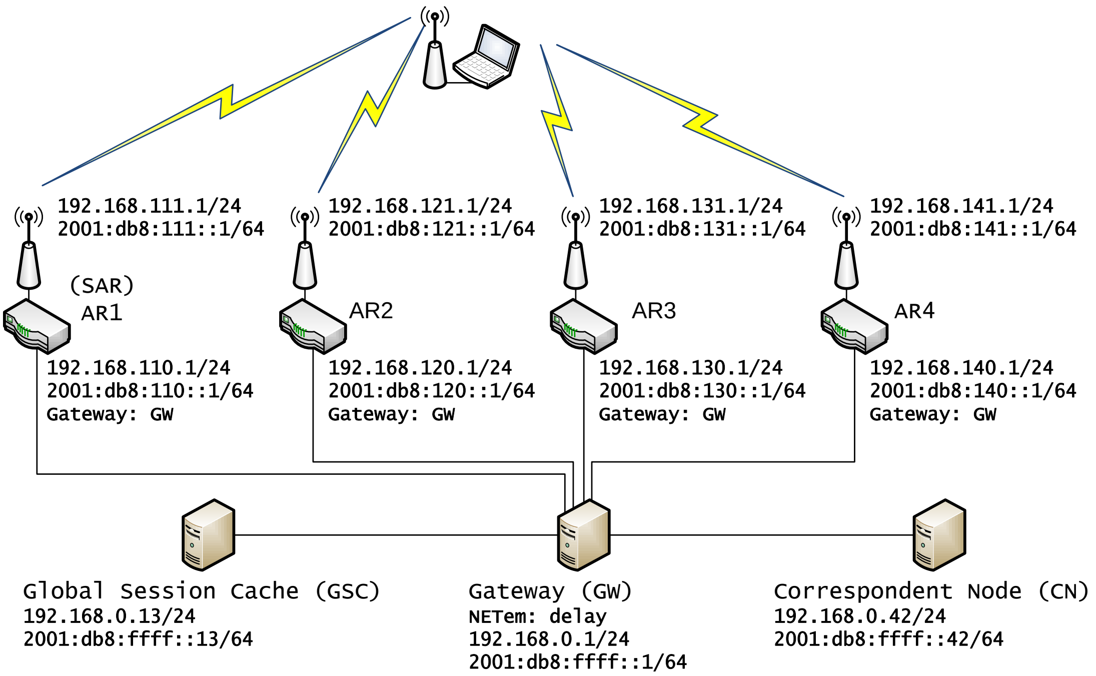


Figure 6.2.: Testbed Topology: Four access router connected via a common edge network.

6.2. Handover Latency

A critical evaluation metric for any mobility management solution is the handover latency as it determines the amount of time a mobile node is unable to communicate during a handover. For the purpose of this thesis handover latency is defined as the amount of time that is introduced by a system component during a handover in which a mobile node cannot send or receive data. It is important to note that the following evaluations only account for the handover latency that is introduced by the ARA framework. Depending on the scenario and the link layer technology in particular other components can also induce latency during a handover. For example when an IEEE 802.11 node switches access points, among other things, it needs to scan for a new candidate access point, possibly perform authentication, and finally complete a handshake to associate with the access point before any link layer connectivity is established. However, for the purpose of this thesis handover latency that is induced outside of the ARA framework is considered out of scope. Firstly, reducing handover latency, especially on the link layer, is a different object of research. Indeed, numerous contributions have been made in the fields of link layer specific handovers (e.g. [97, 126, 98, 169, 19, 35, 143, 29]). Therefore it stands to reason that a separation of these research topics has its benefits as each can be investigated on its own. Secondly and more importantly, the ARA framework operates on the network layer and is therefore link layer independent. Depending on the deployment scenario ARA can be used to provide mobility support for numerous wireless technologies all of which can have different handover methods.

In network based mobility support solutions such as the ARA framework two sources for handover latency can be identified: the configuration of the mobile node and the establishment of data forwarding from the mobility anchor. The time it takes to select a set of suitable configuration data for a mobile node determines the time delay before a mobile node can be provided an IP configuration. Without a suitable network configuration a mobile node can neither send nor receive any data. The time it takes to establish a forwarding of a mobile node's inbound data from it mobility anchor to its current point of attachment determines the time delay before a mobile node can receive data that has been sent by any of its correspondent nodes. While the configuration of a mobile node is a basic requirement for its ability to exchange data, selecting a configuration and establishing the forwarding are operations that can be executed in parallel. Depending on the approach, the entity responsible for coordinating a mobile node's configuration might be identical with a mobile node's mobility anchor which can be leveraged to unify the evaluation of the handover latency. For example in Proxy Mobile IP the entity that handles both a mobile node's configuration data and the data forwarding is the local mobility anchor. However, in the ARA framework a mobile node's configuration is looked up in the session cache while the data forwarding must be triggered by the mobile node's session access router. Therefore, the configuration lookup time and the time it takes to establish a forwarding will be evaluated separately.

6.2.1. Configuration Lookup

In contrast to host based solutions, a mobile node that is provided with network based mobility support needs to be configured with an identical IP configuration at every handover. While a host based approach can just use local configuration data a network based approach needs to make sure that the configuration data is consistent with the configuration that was used at the previous handover. To provide such consistency the particular configuration data has to be coordinated between any mobility anchor that is involved in a mobile node's handover process. In the ARA framework a mobile node's configuration data is maintained as part of the session cache. When a mobile node associates with an access router the access router has to query the session cache to

Type of cache	**Avg. Lookup Time**	**Result**
Server-assisted/Distributed (One-Hop)	$2 * INL$	40 ms
Distributed (Chord)	$(log_{10}(N_{ar}) + 1) * INL$	80 ms
Distributed (Tapestry/Pastry)	$(log_B(N_{ar}) + 1) * INL$	$\sim$ 70 ms
Distributed (One-Hop)	$2 * INL$	40 ms
Hybrid (Chord)	$(log_{10}(N_{cp}) + 2) * INL$	80 ms
Hybrid (Tapestry/Pastry)	$(log_B(N_{cp}) + 2) * INL$	$\sim$ 73 ms
Hybrid (One-Hop)	$3 * INL$	60 ms
Local Session Cache Hit	~ 0	$\sim$ 0 ms

Table 6.2.: Cache lookup times for different deployment models. Results based on 20 ms inter-node latency (INL), 128 bit hexadecimal identifiers (B=16) for DHT algorithms, 1.000 access routers (N_{ar}) for the distributed scenarios, and 100 session cache peers (N_{cp}) for the distributed scenarios.

determine if a mobile node is already managed by the ARA framework and therefore already has a configuration (see Section 3.2). If an entry for the mobile node exists in the local session cache this operation completes almost instantly as no network operation is necessary. An entry typically exists if the mobile node hands in from a neighboring access router that has advertised the mobile node's configuration via a neighborhood update or if the mobile node re-associates before the entry has been removed due to a timeout. However, if there is no entry in the local session cache the access router needs to query the global session cache which involves a network operation and can take a substantial amount of time to complete.

Theoretical Cache Lookup Times for Different Deployments

The actual lookup time of an entry in the global session cache depends on the particular deployment strategy and the network topology. Table 6.2 shows an overview over the theoretical cache lookup times for a number of deployment alternatives. INL is the average inter-node network delay between ARA nodes. The table shows results for an assumed average one way inter-node delay of 20 ms which is the average inter domain delay. This is the same value used in the simulation (see Section 6.1.2). This is an upper bound value as it essentially assumes that every node is in another domain. Figure 6.3 illustrates the expected lookup values for different latencies. Every deployment model

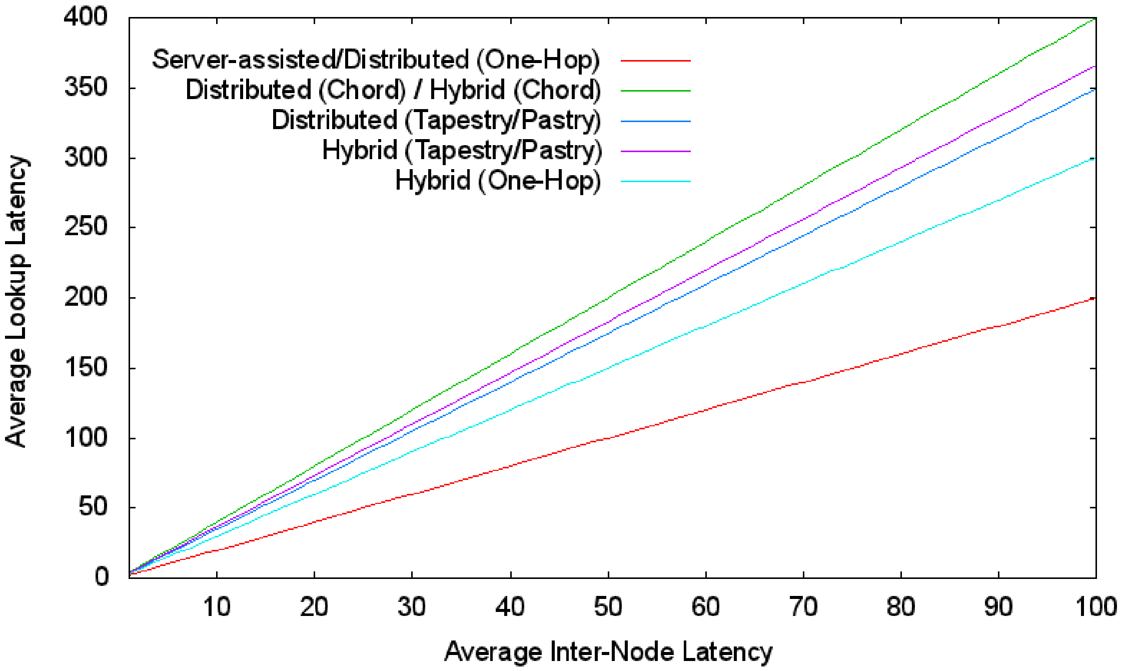


Figure 6.3.: Expected cache lookup times for different inter-node latencies.

has a linear increase of varying degree in lookup times with an increasing inter-node latency. For the server assisted ARA the lookup latency is straight forward. The access router queries the global session cache server which returns a valid configuration for the client. Therefore, the total delay is one round-trip or two times the average inter-domain delay. For distributed and hybrid deployments the lookup time depends on the selected DHT algorithm and the number of access routers that the cache is distributed over. The example includes distributed and hybrid deployments based on Chord [151] and Tapestry/Pastry [171, 139] overlays. The algorithms behind these overlay networks are well known. However, the selection is just of exemplary nature to roughly illustrate the lookup performance of distributed and hybrid deployments (please refer to [86] for a more detailed survey and comparison of peer-to-peer overlay network schemes). Moreover, the practical performance of DHTs is dependent on a number of deployment-specific and environmental parameters such as peer churn. Numerous works have been published on evaluating DHT performance under various conditions (e.g. [82, 83]) showing that the actual performance of DHTs can be highly fluctuating. On the other hand, studies also show that in a stable environment without churn the lookup performance can approximate the well known average hop count (cf. [77]). Therefore, for the scope of this theoretical analysis the average lookup performance of the DHT algorithms as established in the respective original publications will be considered sufficient.

The example shows that server-assisted and distributed deployments using a one-hop DHT algorithm can perform a global cache lookup in one round trip which is equivalent to two times the average inter domain delay and amounts to 40 ms in the topology that has been used in the simulation scenario. Other deployments scenarios and algorithms respectively perform worse. For example, for a distributed cache among 1,000 access routers using the Tapestry algorithm a lookup takes an average of 3.5 hops ($log_{16}(1.000) = 2,941$ an additional hop must be added as the result needs to be send back to the access router) which amounts to 70 ms for an average inter-domain delay of 20 ms. The lookup time for the hybrid approach is analog to the lookup time in the distributed approach except that it takes the access router an additional hop to reach the nodes that are responsible for the cache. As shown in Table 6.2 the lookup delay for the hybrid approach might be higher than in the distributed approach even with a smaller number of nodes.[5] However, the performance benefits in the hybrid approach are expected to come from extensive caching which is not shown in the table. A substantial improvement of the lookup times can be achieved using a one-hop DHT algorithm (e.g. [52, 127, 71]; see [135] for a larger selection) which performs essentially like a server based cache.

Simulation Results

Figure 6.4 shows the lookup times during simulation runs with different numbers of mobile nodes. The data shows the mean lookup delay which over time converges towards zero. This is because with each handover that cannot be performed with data from the local session cache, an access router learns the neighboring access router that a mobile node handed in from via the global session cache. Over time the neighborhood lists become more and more complete and data of mobile nodes can be shared prior to handovers. Eventually the neighborhoods are completely learned and the configuration data of any active mobile node can be disseminated as soon as a mobile node associates with any neighboring access router. The process of learning neighborhoods is getting faster the more mobile nodes are active and triggering handovers. As shown in the figure it takes considerably longer in the scenario with just one mobile node for the

[5] Note that the numbers for the hybrid approaches are based on 100 node DHTs while the numbers for the distributed approaches are based on 1,000 node DHTs. The reason is that in hybrid approaches the DHT is only formed between dedicated caching peers which can be expected to be deployed in a substantially smaller number than access routers.

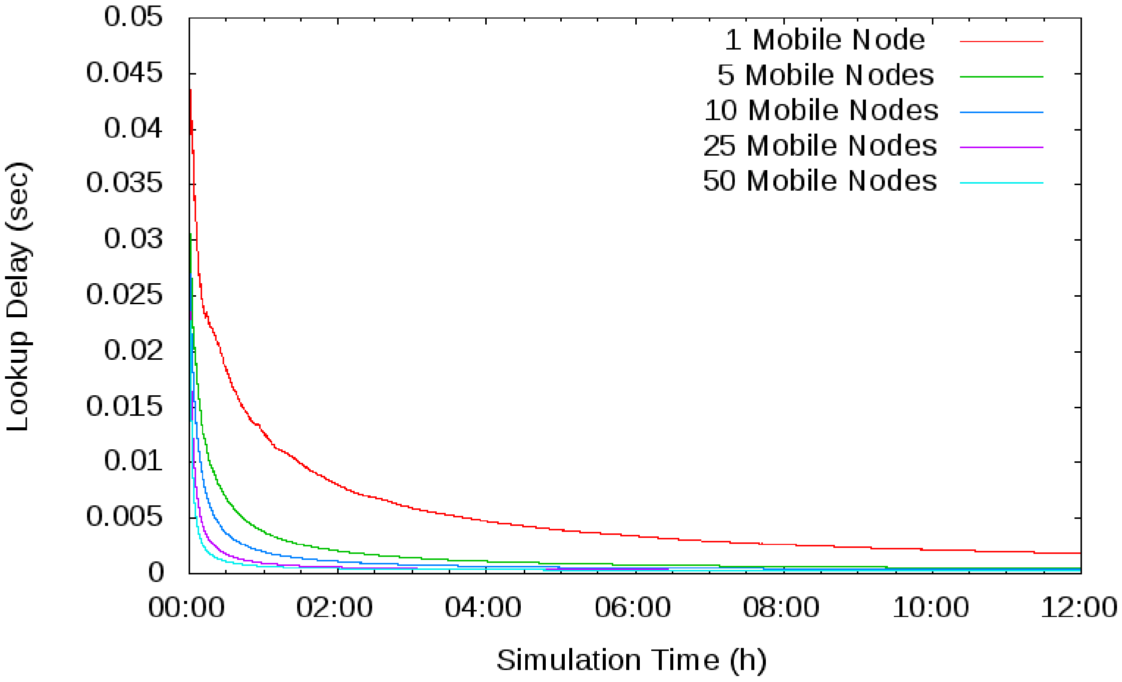


Figure 6.4.: Configuration lookup delays for different numbers of mobile nodes.

configuration look time to converge to low values. In this scenario the only available mobile node has to perform handovers between any possible (and feasible) combination of access routers once before the neighborhood learning process is complete. Every time the mobile node hands over to an access router from an unknown neighbor, the corresponding access router must look up the mobile nodes data in the global session cache which is reflected by long configuration lookup delays. New pairs of access routers are learned in the single node scenario even after multiple hours. In the scenarios with more active mobile nodes the likelihood that another mobile node has already triggered the neighborhood learning process increases with the number of active mobile nodes. Therefore the configuration lookup times converge much faster towards zero.

6.2.2. Forwarding Setup

ARA employs mobility anchors, as do many existing host based as well as network based mobility support solutions. These mobility anchors provide a fixed relay that forwards communication data from a correspondent host to the current location of a mobile node.

In ARA the session access router as the routing destination of a mobile node's prefix needs to forward any data to the mobile node's current access router which in turn can relay the data to the mobile node. When a mobile node changes access routers its new access router needs to look up the session access router and signal it to update the forwarding. The time it takes to complete this process induces a (one-time) delay in the packet forwarding.

Theoretical Analysis

In principle, setting up the forwarding is a straight forward process that requires an access router to send a Session Update Request message to the session access router of the particular mobile node. If an access router already has the information about a mobile node's session access router, for example via a neighborhood update, it can send the request directly. However, if the access router for a mobile node is unknown, the Session Update Request needs to be send indirectly via the global session cache. The overall delay can be modeled as $D_{lookup} + INL$. D_{lookup} is the lookup delay for the session access router information in the session and depends on the ARA deployment type. The global session cache will directly forward the Session Update Request to the session access router after the lookup is complete. Therefore, no additional delay is introduced by sending a response message back to the access router that is requesting the forwarding. If a session cache entry exists in an access router's local cache the lookup delay is insignificantly small as no network operation is required. INL is the average inter-node latency and models the time it takes to deliver the Session Update Request to the corresponding session access router. The session access router will send a Session Update Response back to the access router that requested the forwarding which will also be subject to an inter-node latency. However, the actual forwarding will be setup as soon as the Session Update Request is processed. Therefore, the delay that the response message is subject to is not relevant for the overall forwarding delay and is not represented in the formula.

Similar to the theoretical evaluation of the configuration lookup times done in Section 6.2.1, Table 6.3 shows theoretical values for the forwarding setup delay for different cache deployment models. If there is no hit in the local session cache the forwarding delay values are identical to the configuration lookup delay values. This is because in both operations the global session cache will perform a lookup of the session cache entry and subsequently send a message to an ARA node. The difference between a configuration lookup and a forwarding setup is simply that in the former case a Session Update Response will be send to the querying access router while in the latter case a Session Update Request will be forwarded to the session access router. However, contrary to a

Type of cache	**Avg. Setup Delay**	**Result**
Server-assisted/Distributed (One-Hop)	$2 * INL$	40 ms
Distributed (Chord)	$(log_{10}(N_{ar}) + 1) * INL$	80 ms
Distributed (Tapestry/Pastry)	$(log_B(N_{ar}) + 1) * INL$	$\sim$ 70 ms
Distributed (One-Hop)	$2 * INL$	40 ms
Hybrid (Chord)	$(log_{10}(N_{cp}) + 2) * INL$	80 ms
Hybrid (Tapestry/Pastry)	$(log_B(N_{cp}) + 2) * INL$	$\sim$ 73 ms
Hybrid (One-Hop)	$3 * INL$	60 ms
Local Session Cache Hit	$1 * INL$	20 ms

Table 6.3.: Forwarding setup delays for different deployment models. Results based on 20 ms inter-node latency (INL), 128 bit hexadecimal identifiers (B=16) for DHT algorithms, 1.000 access routers (N_{ar}) for the distributed scenarios, and 100 session cache peers (N_{cp}) for the distributed scenarios.

configuration lookup, a local session cache hit will not reduce the forwarding setup delay to zero. With local information the delay can be merely reduced to a single inter-node latency as a Session Update Request has to be sent to the session access router.

Simulation Results

The simulation results shown in Figure 6.5 corroborate the theoretical analysis. The figure shows the mean forwarding setup delay during handovers for scenarios with different numbers of mobile nodes. In the beginning of the simulation when neighborhoods are still formed data forward requests have to be forwarded predominantly via the global session cache which leads to overall higher forwarding setup times. However with an ongoing build-up of neighborhoods the forwarding setup times continuously decrease and converge towards the average inter-node latency of 16.5 ms. Note that this value is different from the inter-node latency that was assumed in the theoretical analysis as it includes intra-domain nodes that have an average inter-node latency of 2 ms (see Section 6.1.2). The theoretical analysis assumes every node to be an inter-domain node with an average latency of 20 ms as an upper bound.

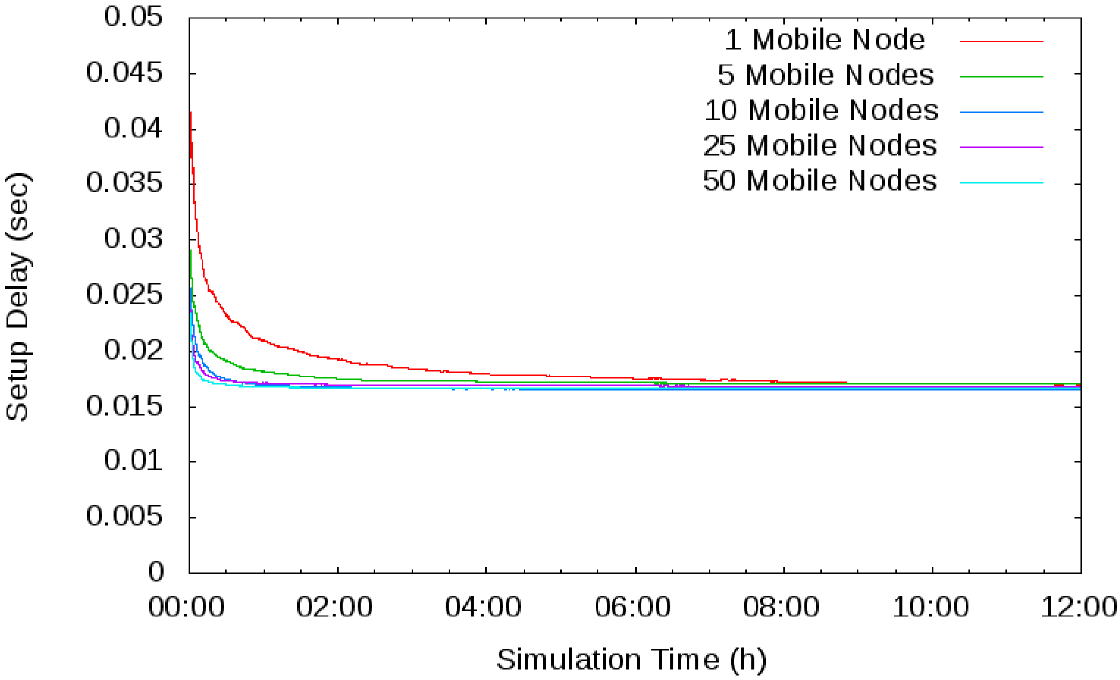


Figure 6.5.: Forwarding setup delays for different numbers of mobile nodes.

6.2.3. Cumulative Handover Delay

The overall delay that is induced by the ARA framework on the network layer during a handover is the combined delay of configuration lookup and forwarding setup. Although a mobile node can already send data after the configuration phase is complete, the forwarding setup needs to be completed in order for a mobile node be able to receive data. Therefore, a bi-directional communication is not established before configuration lookup as well as forwarding setup are performed. Once a bi-directional communication is re-established for the mobile node the handover is considered complete.

Theoretical Analysis

When a handover is performed, the configuration lookup and forwarding setup processes run in parallel. Therefore the overall handover delay can be modeled as

$$D_h = \begin{cases} D_{lookup} + max(d(GSC, CAR), d(GSC, SAR)) & \text{if local cache miss} \\ d(CAR, SAR) & \text{if local cache hit} \end{cases}$$

In case the local session cache does not contain an entry for the mobile node's configuration data and its corresponding session access router, a lookup query has to be sent to the global session cache which will experience a delay of D_{lookup} (see Section 6.2.1 for D_{lookup} values for different cache deployment options). The global session cache sends the (non-authoritative) configuration lookup result back to the current access router and in parallel forwards the session update request which will trigger the forwarding setup to the session access router. Therefore the overall delay is increased by the larger amount of delay between the global session cache and the current access router $d(GSC, CAR)$ and the global session cache and the session access router $d(GSC, SAR)$. If a local cache entry is available the overall handover latency is simply the delay between the current access router and the session access router $d(CAR, SAR)$ as a lookup in the global session cache is not necessary and the session update request to setup the forwarding can be sent directly from the current access router to the session access router. In the scenario with 20 ms as average inter-node latency and a server-assisted session cache the overall handover delay according to the model would be 40 ms in case a local cache entry is not available and 20 ms if a local cache entry is available.

Simulation Results

Figure 6.6 shows the simulation results for the completion times of handovers in the respective access routers. The completion times reflect the overall latency that is induced by the ARA framework during a handover and is measured from the time a mobile node requests a configuration (node sends a Router Solicitation message) until the mobile node can be provided a valid configuration (access router sends a Router Advertisement message) and the data forwarding has been setup (session access router configures new current access router as tunnel destination).

As the theoretical analysis suggests, the handover delay is very close to the forwarding setup delay which has been discussed in the previous section. After an initial stabilization time for the neighborhood forming process the mean handover time converges towards the same average inter-node latency of 16.5 ms. This value coincides with the average forwarding setup delay as a local session cache entry exists for most active mobile nodes in a neighborhood after the neighborhood forming process has taken place. The same considerations as in the previous section apply with regards to the difference between the inter-node latency assumed in the theoretical analysis and the inter-node latency shown by the simulation.

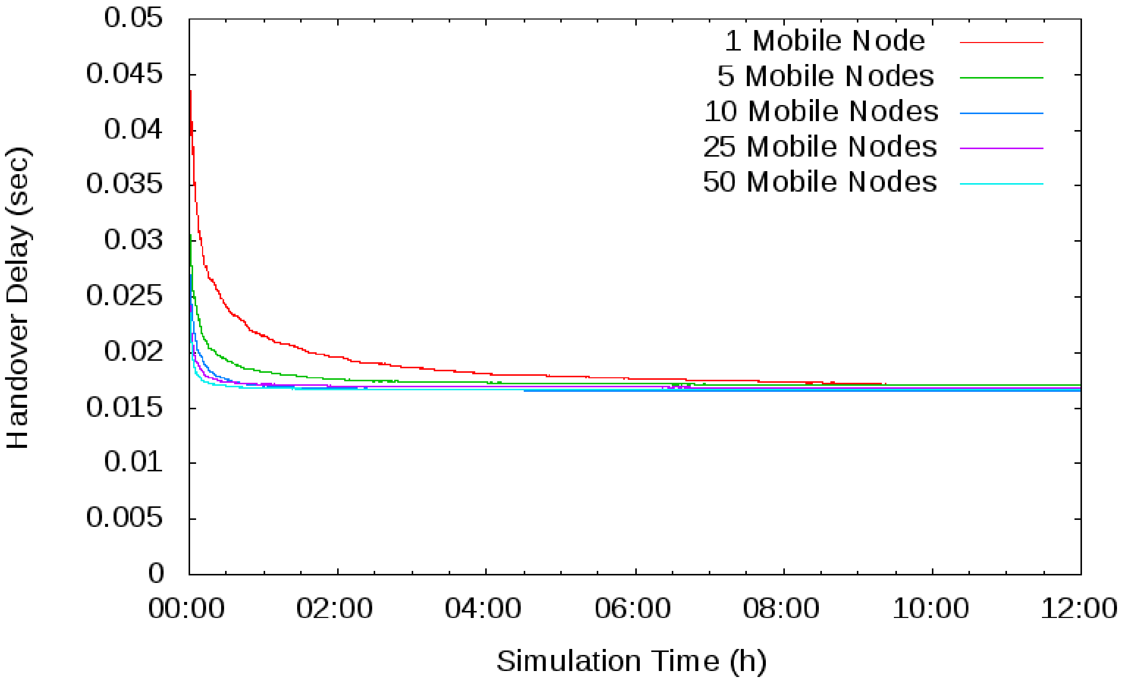


Figure 6.6.: Handover Delay for different numbers of mobile nodes.

Approach	Node Configuration	Forwarding Setup
ARA		
(Local Cache Miss)	$D_{lookup} + d(CAR, GSC)$	$D_{lookup} + d(CAR, SAR)$
(Local Cache Hit)	0	$d(CAR, SAR)$
Mobile IP	n/a	$d(MN, HA)$
Proxy Mobile IP	$2 * d(MAG, LMA)$	$d(MAG, LMA)$

Table 6.4.: Qualitative comparison of handover delays for different approaches

6.2.4. Comparison with Other Approaches

Table 6.4 shows a qualitative comparison of ARA handover delays with other approaches. With regards to node configuration delay, ARA without a hit in the local session cache performs comparable to Proxy Mobile IP in a server-assisted deployment scenario. In this case it can be assumed that

$$(D_{lookup} = d(CAR, GSC)) \leq d(MAG, LMA)$$

since the ARA session cache server and the local mobility anchor will be both deployed in comparable topological positions. The ARA global session cache server might even be deployed in a better position as it does not provide data forwarding for mobile nodes at the same time. Its deployment position, therefore, might be more flexible and be better optimized. The current access router (CAR) and the mobile access gateway (MAG) respectively is in both scenarios the particular access routers that a mobile node is connected to. If a mobile node's configuration is already present in the local session cache, the node configuration time in ARA is near instant. In Mobile IP a node configuration time does not apply since the mobile node will obtain a local configuration every time it moves. Comparing the forwarding setup delay in a server-assisted ARA deployment it can be assumed that

$$(D_{lookup} = d(CAR, GSC)) \equiv d(MN, HA) \equiv d(MAG, LMA)$$

for the same reason that nodes with similar roles will be placed in similar topological positions. In that case ARA performs comparable to Mobile IP and Proxy Mobile IP when a mobile node's configuration is present in the local cache. Otherwise the forwarding setup delay is about twice as much as the other approaches can signal the responsible mobility anchor directly to setup the forwarding while in ARA the signaling message is redirected via the session cache. Overall, the handover procedures among the different approaches induce comparable delay under the assumption of similar node placement.

6.2.5. Conclusion

The handover delay in the ARA framework is composed of two components: Configuration lookup delay occurs because an access router needs to lookup the proper IP configuration data for a mobile node and forwarding setup delay occurs because the forwarding of a mobile node's data to its new current access router has to be configured at the session access router. During the initial phase in which neighborhoods are still being formed the handover time is strongly affected by the global session cache lookup time as the global session cache still has to be queried regularly. In this phase the handover delay is upper bound by the global session cache lookup delay plus the maximum value of the delay between the global session cache and the current access router and the delay between the global session and the session access router. In a scenario with an average inter-domain latency of 20 ms and an average intra-domain latency of 1 ms the simulated handover delay converges towards a mean value of 16.5 ms after only a couple of hours. ARA can reduce configuration lookup times to zero (after proper neighborhoods are formed) and the forwarding setup time to one average inter-node latency between the current access router and the session access router. In comparison with Mobile IP and Proxy Mobile IP, ARA has comparable handover times under the assumption of similar node placement.

6.3. Routing Efficiency

Similar to other approaches, ARA uses a mobility anchor to redirect a mobile node's traffic to its current point of attachment to the network. While this is an effective solution to provide mobility support without requiring changes to existing protocols and nodes it also introduce routing inefficiencies. Unless the mobility anchor is positioned inside the direct communication path between the mobile node and the correspondent node the redirection via the mobility anchor is an indirection. In intra-domain mobility approaches, for example, Proxy Mobile IP, it might be possible to position the mobility anchor in such a way that it always is within a mobile node's direct routing path. A positioning as an edge router, for example, would suffice. However, in inter-domain scenarios, it is much more difficult to find a position that is a common intersection between all possible combinations of points of attachments for mobile nodes and correspondent nodes. As a design goal ARA is not supposed to impact nodes before they start roaming. With regards to the routing efficiency ARA realizes this by choosing the first access router that a mobile node attaches to as its mobility anchor.

6.3.1. Theoretical Analysis

The routing stretch is a metric to evaluate the routing overhead a packet experiences. It is defined as

$$\frac{D_{Act}}{D_{Opt}}$$

with D_{Act} being the distance (in ms or hops) that a packet actually travels and D_{Opt} being the distance of the optimal path. In mobility solutions a routing stretch occurs because the data between the corresponding node and the mobile node needs to be

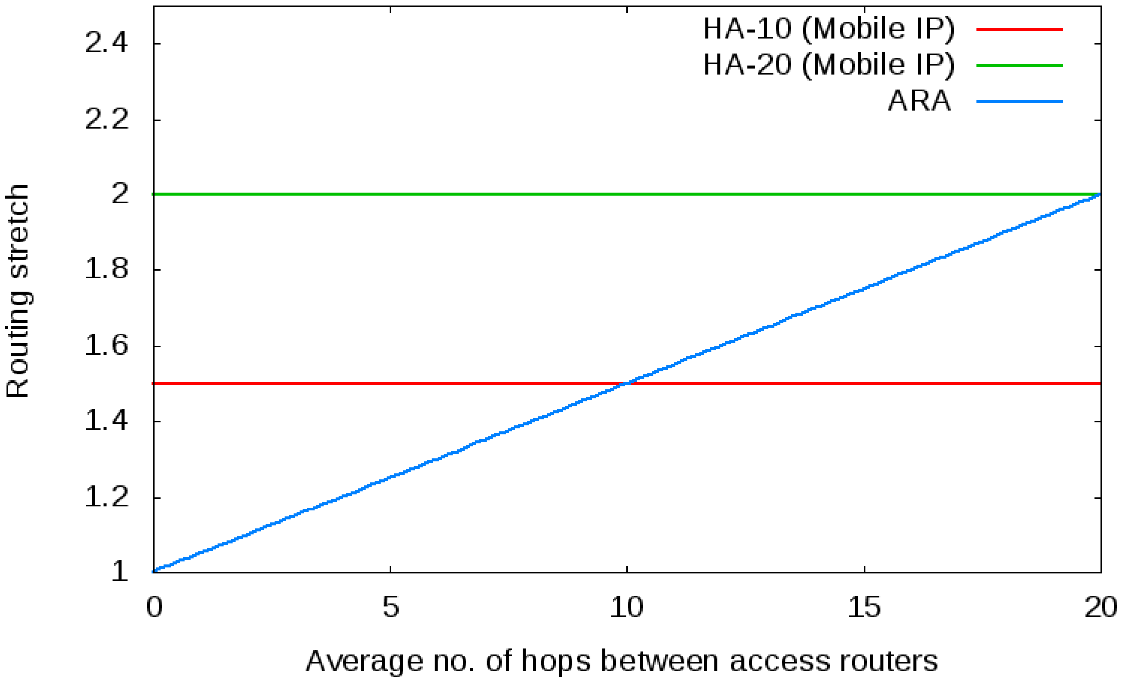


Figure 6.7.: Comparison of routing stretches

forwarded via the mobility anchor which might not be on the optimal path. Figure 6.7 shows an idealized comparison plot of routing stretches between ARA and Mobile IP. The “HA-10” line shows the routing stretch for a path via a Mobile IP home agent that is 10 hops away from the domain with the mobile node while the “HA-20” line shows a home agent 20 hops away. The optimal path is 20 hops and the x-axis shows the average number of hops between ARA access routers (i.e. the session access router and the current access router). As the Figure shows in Mobile IP the routing stretch can be significant if the home agent is far outside the optimal routing path. More importantly it is always there. Even if the mobile node has not moved away from its initial access router (average hop distance of zero) there is already routing stretch which introduces additional delay for any communication. While in ARA the mobile needs to move a significant number of hops away from its session router to experience a noticeable routing stretch.

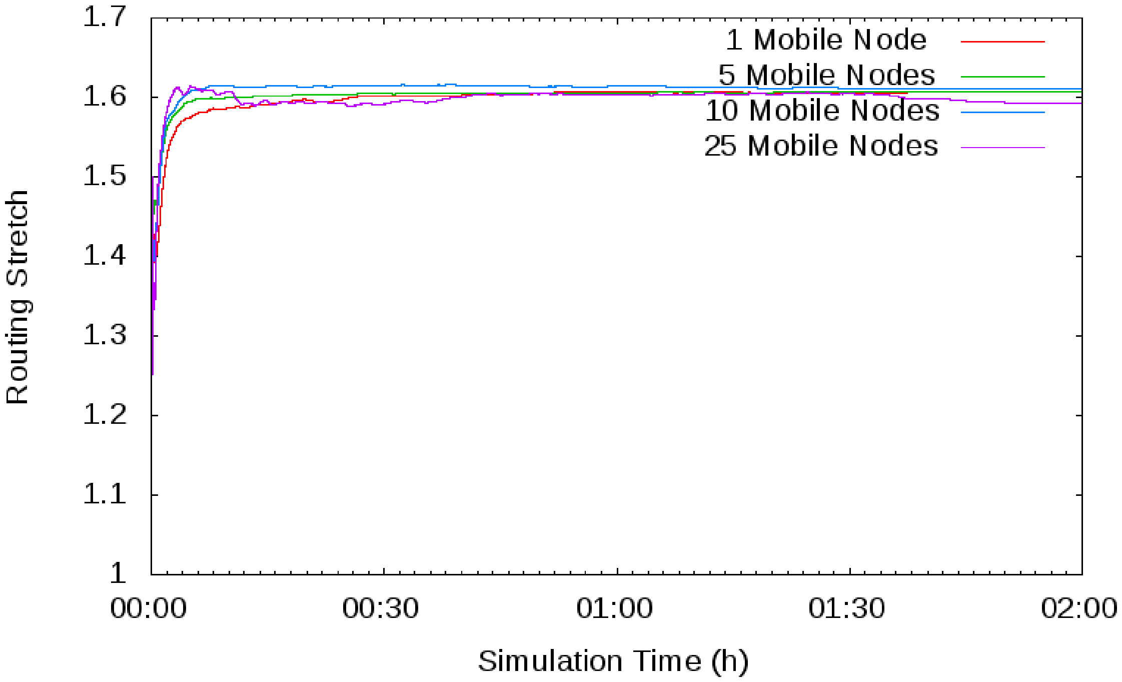


Figure 6.8.: Comparison of routing stretches

6.3.2. Simulation Results

Figure 6.8 shows the mean routing stretch for all mobile nodes in different ARA simulation runs. As shown there is no significant difference in routing stretch with a varying number of nodes. That is because the number of nodes does not have a direct or indirect impact on the routing of data traffic. Even though the number of nodes can influence the speed of the neighborhood learning process, neighborhood information is only used in the handover process and does not provide any means to optimize the routing of a mobile node's data. Another important observation is that the routing stretch in the beginning of the simulation when every mobile node is still attached to their respective session anchor is indeed 1. This means that the ARA framework satisfies its design goal not to impact non-roaming nodes with regards to routing efficiency (see Section 3.1.2 for the design goals of the ARA framework). The mean routing stretch converges towards 1.6. However, this value is determined solely by the simulation topology and does not allow to draw a general conclusion. In the simulation scenario there is an optimal path between any mobile node and correspondent node of only three hops. Intra-domain ac-

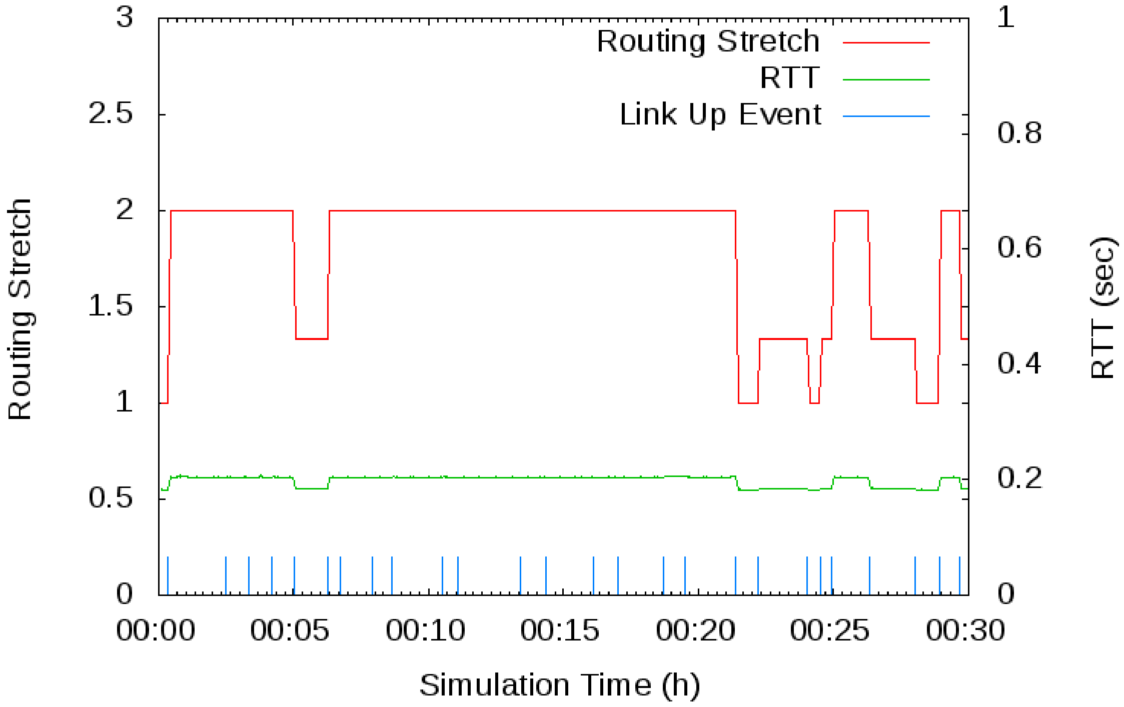


Figure 6.9.: Routing stretch for a single node

cess routers are one hop away and inter-domain access routers are only two hops away from any other access router. This is a very compact topology and even an increase by a small number of hops has a sizable impact. In real-world deployments it can be assumed that access router are geographically and topologically closer to each other than to the correspondent node which would lead to a smaller routing stretch.

Figure 6.9 shows a more detailed course of the routing stretch for a single node while it roams between access routers. Every "Link Up" event marks a handover to another access router. The routing stretch is plotted against the y-axis to the left while the round-trip-time between the mobile node and the correspondent node is plotted against the y-axis to the right. The figure shows three different magnitudes for the routing stretch. In the beginning of the simulation and every time the mobile node returns to its session access router the routing stretch is 1 ($D_{Act} = D_{Opt} = 3$). When the mobile node roams to an access router that is within the same domain as its session access router the routing stretch increases to 1.34 ($D_{Act} = 4$ and $D_{Opt} = 3$). Lastly, when the mobile node roams to an access router that is not within the same domain as its session access router the routing stretch increases to 2 ($D_{Act} = 6$ and $D_{Opt} = 3$). However, when the round-trip-time is considered the magnitude of the routing stretch is put into perspective. Although the routing stretch increases up to a factor of two, the round-trip-time only increases marginally from 180 ms to 204 ms which is a factor of about 1.13. The reason behind this is that the routing stretch based on simple hop counts does not consider the round-trip-times between the hops. In the simulation scenario the ARA domains that the mobile node moves in are connected via inter-domain links and have a much smaller delay between each other than to the correspondent node that is connected via the core network. Therefore, even with a high routing stretch, ARA can achieve a good routing efficiency assuming that the involved ARA domains are localized.

6.4. Signaling Overhead

There are two sources for signaling overhead in the ARA framework: Session update messages and neighborhood update messages. Session update messages are only exchanged between the session access router of a mobile, the current access router of a mobile node, and the global session cache and are essential for the basic operation of the ARA framework. Neighborhood update messages are exchanged proactively between neighboring access routers to exchange configuration data of mobile nodes that may be handing over in the future to speed up the handover process. This section will discuss the overhead introduced into a network by these two signaling mechanisms.

6.4.1. Message Sizes

The size of a basic session cache entry is about 60 bytes (see Section 4.4.3). Accounting for some additional data such as information about the current or previous access router, sequence numbers, or security tokens the payload of a session or neighborhood update message can be expected to be less than 240 bytes. An IPv4 header would add 20 bytes [120], an IPv6 header 40 bytes [34], a UDP header 8 bytes [119], and a TCP header 20 bytes [121] (assuming no additional header options are set for IP or TCP). In the worst case scenario (TCP over IPv6) this adds up to a total of 300 bytes for a single message. As 240 bytes of payload already seem a high estimate and are meant to provide an upper bound, session update messages and neighborhood update messages will both be assumed with a message size of 300 bytes for the remainder of this evaluation.

6.4.2. Session Update Messages

Every time a mobile node roams to another access router Session Update Request and Session Update Response messages must be exchanged between the session access router, the current access router and the global session cache to coordinate the handover and to keep configuration information consistent. Furthermore, depending on the state model, session update messages have to be regularly repeated to keep the states for the particular mobile node fresh.

Theoretical Analysis

Session update messages are triggered every time a mobile hands over to a new access router and if necessary periodically to keep the states for session cache entries from expiring. The overhead of session update messages can be modeled as

$$(\alpha + \beta) * N_{MN} * 2 * (s(SUReq) + s(SUResp)))$$

with α being the handover frequency, β being the refresh frequency, N_{MN} the number of active mobile nodes, and $s(SUReq)/s(SUResp)$ the size of a session update request and response message respectively. The factor of 2 is because every update triggers two Session Update Request messages: one is send to the session access router and one is send to the global session cache. If the session access router is known to the current access router, the current access router will send a Session Update Request directly to the session access router which in turn will send a second Session Update Request to refresh the global session cache. If the session access router is unknown to the current access router, the current access router will send a Session Update Request to the global session cache which will send a second Session Update Request to the session access router. If the handover frequency and the refresh frequency stay that same the signaling overhead can be expected to grow linearly with the number of active mobile nodes.

Simulation Results

The signaling overhead is expected to grow linearly with the number of mobile nodes which is confirmed by the simulation. Figure 6.10 shows the simulation results for 1 to 600 nodes and a one hour (simulated time) simulation run. For a complete list of simulation parameters please refer to Section 6.1.2. The overall signaling overhead is on average between about 0.2 (single node scenario) and 110 messages (600 nodes) that are being sent per minute per individual access router. Assuming an average message size of 300 bytes this translates to an average signaling traffic between 1 byte per second and

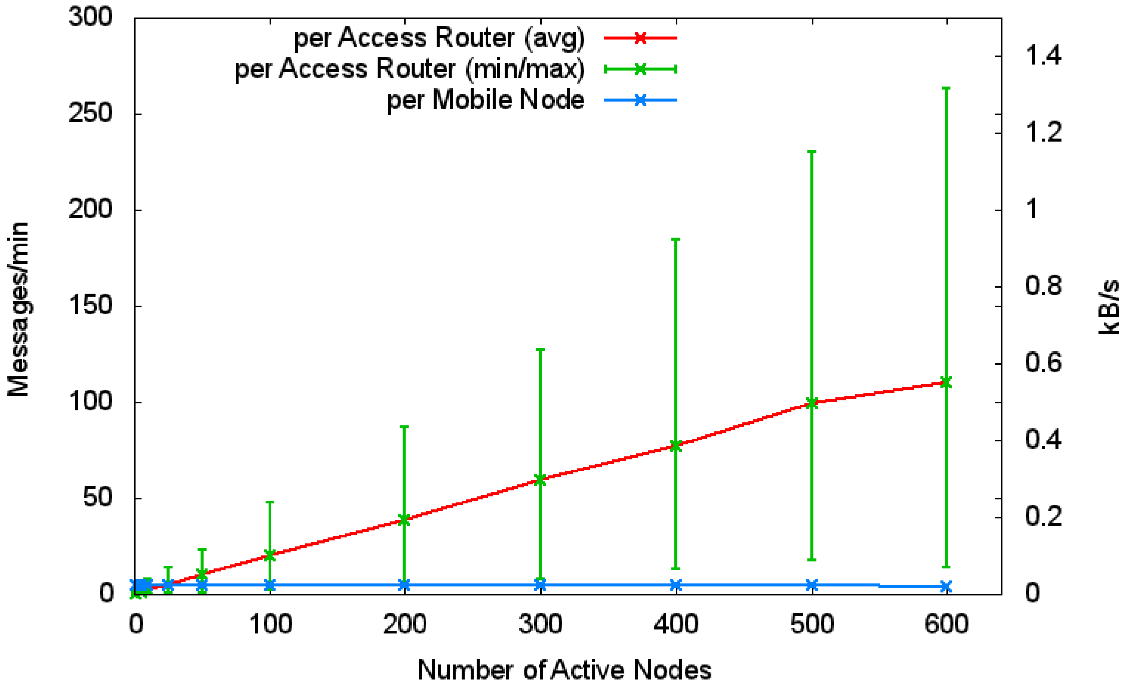


Figure 6.10.: Results of signaling overhead evaluation

550 bytes per second respectively per access router. As the figure shows the signaling load is not evenly distributed between the access routers. The reason is that the Random Waypoint Model used in the simulation implicitly favors central nodes. Therefore access routers that are in the center of the simulation topology serve more mobile nodes and have a higher signaling overhead. However, even the maximum signaling load of 260 messages per second or 1.3 kilobyte per second seems justifiable considering that the access router serves an average of over 50 nodes. The figure also shows that the signaling load per mobile node is constant and with about 4 messages per minute very small.

6.4.3. Neighborhood Update Messages

To reduce handover times access routers will synchronize data about active mobile nodes with neighboring access routers via Neighborhood Update messages. This mechanism can effectively reduce configuration lookup delays (see Section 6.2.1) but also induces signaling overhead. A concern that needs to be addresses with regards to neighborhoods is the scalability of the concept. Maintaining a neighborhood and exchanging information

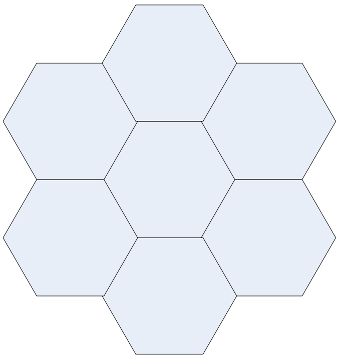


Figure 6.11.: Typical hexagon-shaped cellular radio access point deployment.

about any node that is active within this neighborhood is a process that creates a notable amount of overhead. There are a number of contributing factors that affect the amount of overhead a neighborhood creates: the size of the neighborhood, the number of active nodes, the amount of data exchanged per node, and the interval in which data is exchanged.

Theoretical Analysis

Neighborhood Update messages are sent periodically to every access router that is on the neighborhood list. They are unidirectional messages and do not generate a response. The overhead of neighborhood update messages can be modeled as

$$(\alpha + \beta) * \gamma * N_{MN} * s(NU)$$

with α being the handover frequency, β being the refresh frequency, γ being the average number of neighbors, N_{MN} the number of active mobile nodes, and $s(NU)$ the size of a Neighborhood Update message. Similar to the overhead of session update messages the overhead can be expected to scale linearly. As the size of a Session Update Request, a Session Update Response, and a Neighborhood Update message is within the same magnitude (see Section 4.2 for details about these protocol messages), thus $s(SUReq) \approx s(SUResp) \approx s(NU)$, the overhead of Neighborhood Update messages will be larger than the overhead of Session Update messages if the average number of neighbors is larger than 4 (cf. the theoretical analysis done in Section 6.4.2).

Estimating the Neighborhood Size As an access router synchronizes its local session cache with ever router in its neighborhood, the size of the neighborhood (i.e. the number of access routers in the neighborhood) has a direct, linear impact on the scalability of the synchronization overhead. In cellular networks such as GSM or UMTS radio access points are commonly distributed in hexagon-shaped cells as shown in Figure 6.11. In such a deployment scenario the number of neighbors for each access router would be six. However, such a deployment scenario is idealized and in reality the combination of different cell sizes, physical and structural constraints, opportunistic deployment, or heterogeneous deployments will lead to larger neighborhood sizes. Therefore, a neighborhood size of six can only be assumed as a lower bound.

Estimating the upper bound is more difficult as there are many factors that can influence the actual number of deployed radio access points in an area. However, it is important to keep in mind that a neighborhood only includes access routers that allow a mobile node to directly handover into the service are of the particular access point. Typically a more dense deployment also requires smaller radio cell sizes which naturally limits the neighborhood size. Otherwise the performance of the wireless medium in dense deployments will be impacted by interference or erratic handover behavior [42, 43, 123]. While in heterogeneous networks such as 3G cellular networks, a substantial amount of network planning is involved before radio towers are erected (cf. e.g. [153]) in other deployments such as shared Wi-Fi networks (e.g. FON[6]) access points are deployed without co-ordinated planning or control in a opportunistic manner. In such scenarios considerations about suitable placements of access routers to optimize overall performance and minimize interference are not necessarily taken into account. Instead access routers are deployed where the opportunity arises (e.g. in a person's home). In such uncoordinated deployments numerous access points can be clustered together in a small area which increases the size of each access router's individual neighborhood. Numerous approaches exist to limit the interference in such uncoordinated scenarios (e.g. [99, 7, 84, 28]). However, no approach can overcome the physical limitations of the wireless medium. Moreover, it can be assumed that even in uncoordinated deployments each individual access router is deployed with some feasibility in mind. For example, a user would probably not deploy tens of access routers in a small apartment or business. In conclusion, an upper bound for the neighborhood size is highly specific to a particular deployment and cannot be accurately estimated. For the purpose of this thesis it assumed that an average neighborhood size can up to 100 access routers.

Estimating the Number of Active Nodes The number of active nodes within a neighborhood is another important factor when trying to estimating the overhead of cache synchronization. The most limiting factor for the number of active users is the available

[6] http://www.fon.com/

wireless bandwidth within a certain access technology. Since the wireless medium is a shared access medium it does not scale with the number of radio access points in an area unless they are using different frequencies. In addition, the available frequency spectrum for a single radio access technology is commonly tightly regulated and limited. A direct limitation of active nodes is not given by current access technologies as bandwidth is usually dynamically allocated by the radio access point or preempted by the wireless clients. However, Wi-Fi commonly supports tens of users up to a hundred while wide range technologies such as WiMAX or 3G cellular networks commonly support hundreds of users up to a thousand.

Worst-Case Estimation Using the estimated worst-case numbers an access router would support 1,000 active mobile nodes and have 100 neighbors. Assuming a handover frequency of 0.2 per minute (i.e. a mobile node hands over on average every 5 minutes), an update frequency of 2 per minute (i.e. the neighborhood updates are sent every 30 seconds), and a message size of 300 bytes this amounts to 220,000 messages per minute or 1.1 megabyte traffic per second. While this amount of signaling traffic seem substantial it reflects a worse-case with very high estimates of the neighborhood size and the number of active mobile nodes. However, the estimation shows that even in such a worst-case scenario the signaling overhead stays within a manageable region. Moreover, it can be expected that in such a high capacity environment some of the signaling overhead optimization that were discussed in Section 4.4.3 are considered.

Simulation Results

The simulation results as shown in Figure 6.12 confirm a linear increase of the signaling overhead of Neighborhood Update messages. The overall signaling load per mobile node stays is stable. While the evaluation shows a lower signaling overhead for the scenarios with one and five mobile nodes this is due to the fact that the neighborhood forming process is overall slower in these scenarios. Once the neighborhoods are stable the signaling overhead is on the same level as in the scenarios with more mobile nodes. The overall neighborhood signaling overhead is on average between 0.14 (single node scenario) and 170 messages per minute per individual access router. Assuming a message size of 300 bytes this translates to an average signaling traffic between 0.7 bytes per second and 850 bytes per second. These values seem in line with the theoretical analysis as the average number of neighbors in the simulation scenario is 4.45 which is only slightly higher than theoretical value of 4 at which the neighborhood overhead would be in line with the session update overhead. Similar to the evaluation in the previous section the signaling load is not evenly distributed. The maximum signaling load is about 1,100

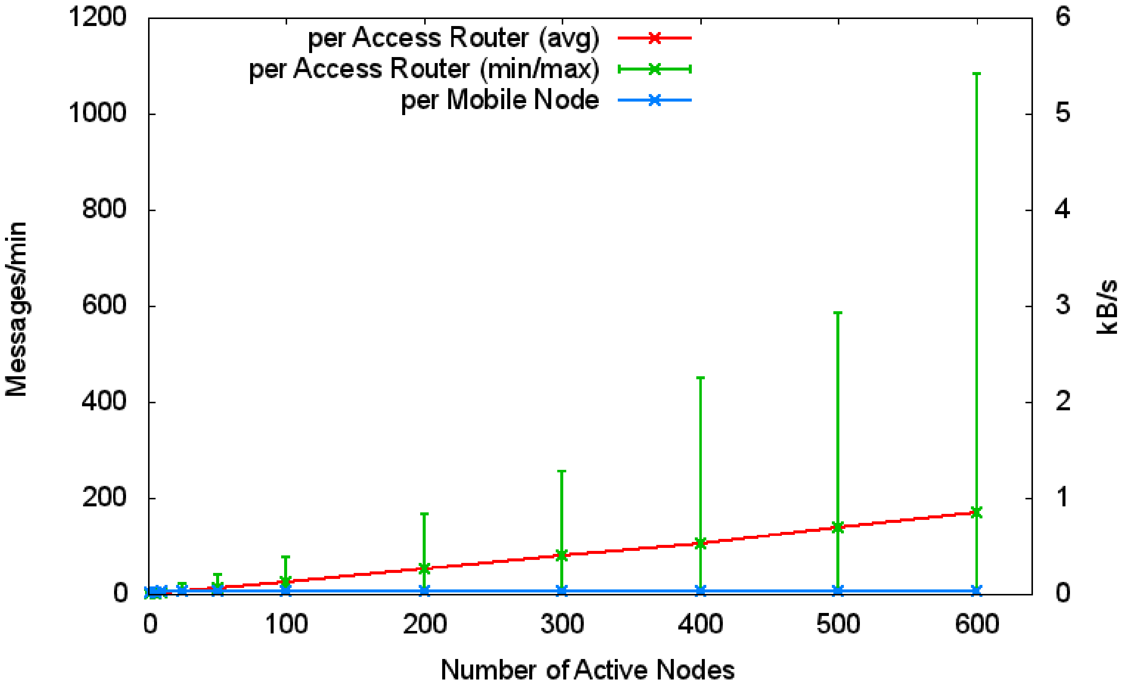


Figure 6.12.: Signaling overhead evaluation of neighborhood messages

messages per minute or 5.5 kilobyte per second. Assuming that such an access router would serve more than 50 mobile nodes this traffic still seems negligible considering the bandwidth that would need to be available for the mobile node's data traffic. The average signaling overhead per mobile node is about 6 messages per minute.

6.5. System Performance

The system performance of the ARA prototype implementation has been evaluated on embedded devices to show the feasibility of deploying ARA on such platforms.

6.5.1. Processing Demands

As processing power and memory is a constraint in embedded environments the maintenance of the session cache has been evaluated on two embedded devices. The Linksys WRT160N was able to handle up to 50,000 entries with its available memory and the Linksys WRT160NL was able to handle up to 100,000 entries. Figures 6.13 and 6.14

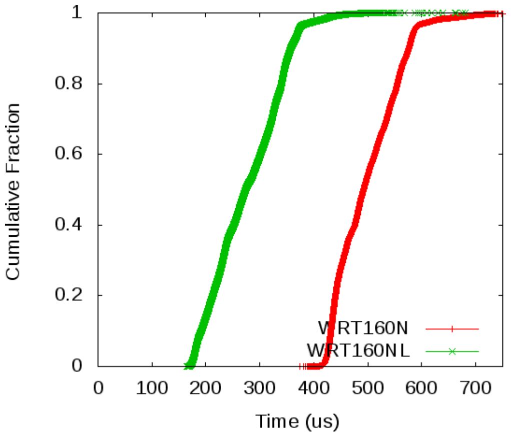


Figure 6.13.: Creation times for session cache entries

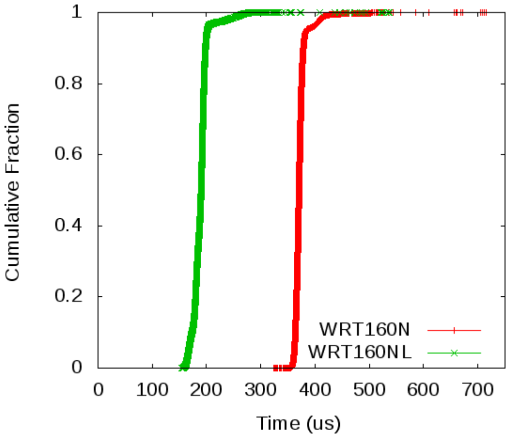


Figure 6.14.: Lookup times for session cache entries

Data	**Memory Footprint**
Mobile Node Configuration Data	78 Bytes
Management Data	32 Bytes
Management Overhead	18 Bytes
Sum	128 Bytes

Table 6.5.: Memory footprint details for a single session cache entry.

show the cumulative distribution function for the creation times and lookup times respectively to fill the local session cache with 50,000 and 100,000 entries respectively and perform the same number of read operations. While the creation times get marginally slower with an increasing size of the session cache both operations complete in a few 100 micro seconds. Theoretically this is enough to handle thousands of cache entries per second. Considering that commonly such an embedded access point only serves tens or maybe a few hundred nodes altogether it seems feasible to maintain a session cache in this environment.

6.5.2. Memory Consumption

Table 6.5 shows the memory footprint details for a single session cache entry based on the creation of 50,000 and 100,000 entries respectively (see previous section). The memory footprint amounts to about 128 Bytes of data per entry. The Linksys WRT160N has an internal flash RAM of 16 MB of which roughly 10 MB are occupied with the operating system. This leaves about 6Mb which allows for the in-memory storage of roughly 50,000 session cache entries. This seems to be a greatly sufficient number considering that one such device probably cannot support more than about 100 associated nodes. Even when neighborhood entries are taken into account from 100 neighboring access routers that support 100 associate clients themselves this only amounts to 10,100 session cache entries in total or about 1.3 MB of memory. In small deployments such an embedded device could even be used to host the global session cache. The Linksys WRT160NL has 32 MB of flash memory and therefore can maintain at least twice the amount of session cache entries.

7. Conclusion

This thesis has proposed ARA, a mobility management framework that provides network based mobility management in a distributed environment. ARA completely integrates mobility support into the access routers that serve the mobile nodes. No further support from the network, dedicated mobility anchors, or the mobile node itself is necessary. Because of this characteristics ARA has a very low deployment barrier. Furthermore because of its decentralized approach it is well suited not only for small deployments but also for large, multi-homed deployments, multi-provider deployments, and fragmented networks. The ARA framework, its design, implementation, and deployment have been thoroughly discussed and evaluated by means of theoretical analysis, simulation and a prototype implementation. The results show that ARA can provide mobility support with minimal handover times of 0.5 RTT between two access routers while introducing an overhead that scales linearly with the number of nodes and access routers in the network.

This chapter concludes the thesis. It will highlight the contributions made and discuss future work.

7.1. Contributions

This thesis presents a framework for decentralized network based mobility management. Its contributions are summarized below.

7.1.1. A Case for Decentralized Mobility Management

Currently deployable approaches to mobility management that do not require changes to the mobile node rely on centralized components. In localized networks this approach might be sufficient. However, this thesis makes a case for an approach to mobility management that does not rely on any centralized components. In decentralized or fragmented environments such as multi-homed networks, multi-provider scenarios, or extensive networks, centralized components not only present bottlenecks and single points of failure, they also lead to routing inefficiencies and increase handover delays. A decentralized approach can provide routing without any inefficiencies for non-roaming mobile nodes and handover delays that are only dependent on the distance between access routers irrespectively of the overall network topology.

7.1.2. Architectural Framework

An architectural framework has been developed in this thesis according to the following design goals:

- **Minimize Bottlenecks and Single Points of Failure:** The mobility support in the framework is provided only by the access router that a mobile node initially attached to and the access router that a mobile is currently attached to. This makes the initial access router that a mobile attached to the only node that is additional involved in a mobile node's network access. Any other node is involved to the same degree as it would be without mobility support.

- **No Changes to End Hosts:** ARA provides network based mobility support. This means that any mobility related operations are solely provided by the network. No changes are required to the mobile node or its correspondent nodes.

- **Deployable within the Current Internet:** ARA is implemented on the network layer and on top of IP. There are no deployment barriers with regards to the current Internet environment.

- **No Impact on Non-Roaming Nodes:** When a mobile node initially attaches to an access router it will be provisioned with an IP configuration from a local pool. There is no functional difference to a mobile node that attaches to any access router without mobility support. A small lookup delay does occur at this point in ARA. However, considering that the mobile node initially attaches to the network and no connections are established yet this small lookup delay seems negligible.

- **Low Deployment Barriers and Inherent Scalability:** ARA's functionality can be completely distributed among access routers. In the simplest case an operator just needs to deploy ARA-enabled access routers or update existing access routers with an ARA-enabled firmware. As more and more access routers are deployed the systems scales as mobile nodes can be distributed among them.

- **Minimize Configuration Delay:** The configuration delay during a handover determines how long it takes to provide the mobile node with a valid IP configuration so that it can send data. ARA proactively distributes a mobile node's configuration data in a neighborhood among any access router that is a potential handover candidate for a mobile node at any given time. This reduces configuration lookups to a local operation without any network induced latencies.

7.1.3. System Design

Based on the architectural framework this thesis presented a system design that implements the ARA framework. A simple IP based protocol has been introduced that is used for signaling operations during handovers and to manage neighborhoods. Synchronization heuristics have been discussed to provide consistent session cache states among multiple nodes, to maintain neighborhoods, and to efficiently distribute information in neighborhoods under various conditions.

7.1.4. Deployment Considerations

A system's deployability depends on numerous considerations that are not easily quantifiable or measurable. Therefore, particular considerations about the deployability in mixed environments, the security, and the accountability of the ARA framework have been made.

7.1.5. Evaluation

Using theoretical analysis, simulations, and prototype implementation the ARA framework has been thoroughly evaluated. Some noticeable results are as follows:

- The configuration lookup delay in ARA converges towards 0 after neighborhoods are formed.
- Neighborhoods are being formed in a matter of hours in a stable topology
- The forwarding setup delay converges towards the average inter-node latency
- The overall handover delay is defined by the forwarding setup delay and therefore also converges towards the average inter-node latency
- Compared to other approaches ARA performs better in terms of handover latency under the assumption that the average inter-node delay to a geographically localized access routers is smaller than to a centralized node
- ARA introduces a noticeable routing stretch, comparable to Mobile IP only when mobile nodes start roaming

- In terms of latency even a noticeable routing stretch can be negligible if the stretch is produced locally
- The signaling overhead of ARA increases linearly with the number of mobile nodes for session update messages and linearly with the number of mobile nodes and the average neighborhood size for neighborhood update messages
- The performance requirements of an ARA implementation can easily be provided by embedded systems

7.2. Future Work

A number of issues exist that have been considered out of scope of this thesis. The issues are not of fundamental nature but might be worth investigating in future work.

7.2.1. Dynamic Environments

The evaluations of the ARA framework have been made under the assumption of a stable topology which holds for the envisioned deployment scenarios. However, in highly dynamic scenarios where access routers move, are only sporadically available, or change their radio transmission range the efficiency of neighborhood updates might be impacted. Possible results of such dynamic environments are that neighborhood lists grow in an uncontrolled manner or become inaccurate as neighboring access change frequently. While the basic mechanism is still warranted in such scenarios the neighborhood maintenance algorithms might need to be adjusted.

7.2.2. Network Mobility

The ARA framework is designed with the goal of providing host mobility, that is mobility to single mobile nodes. Another form of mobility support provides network mobility, that is mobility to a whole subnetwork. Scenarios for network mobility are, for example buses, airplanes, or ships that carry a large number of mobile nodes. In such scenarios it often makes more sense to provide mobility to a whole subnetwork instead of every single host. Host based mobility and network based mobility share a number of basic concepts and assumptions. For example, the Network Mobility Basic Support Protocol [36] is implemented as an extension to Mobile IP. It is not clear if the ARA approach can be transferred to provide network based mobility. However, conceptually every subnetwork is also anchored at an upstream router. Therefore, the ARA framework might be adapted for such a scenario.

Bibliography

[1] J. Abley and K. Lindqvist. Operation of Anycast Services. RFC 4786 (Best Current Practice), Dec. 2006.

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

[3] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz. Extensible Authentication Protocol (EAP). RFC 3748 (Proposed Standard), June 2004. Updated by RFC 5247.

[4] U. G. Acer, S. Kalyanaraman, and A. A. Abouzeid. Weak state routing for large scale dynamic networks. In *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, MobiCom '07, pages 290–301, New York, NY, USA, 2007. ACM.

[5] A. Adams, J. Nicholas, and W. Siadak. Protocol Independent Multicast - Dense Mode (PIM-DM): Protocol Specification (Revised). RFC 3973 (Experimental), Jan. 2005.

[6] A. Adya, P. Bahl, R. Chandra, and L. Qiu. Architecture and techniques for diagnosing faults in ieee 802.11 infrastructure networks. In *Proceedings of the 10th annual international conference on Mobile computing and networking*, MobiCom '04, pages 30–44, New York, NY, USA, 2004. ACM.

[7] A. Akella, G. Judd, S. Seshan, and P. Steenkiste. Self-management in chaotic wireless deployments. In *Proceedings of the 11th annual international conference on Mobile computing and networking*, MobiCom '05, pages 185–199, New York, NY, USA, 2005. ACM.

[8] J. Arkko, C. Vogt, and W. Haddad. Enhanced Route Optimization for Mobile IPv6. RFC 4866 (Proposed Standard), May 2007.

[9] R. Atkinson, S. Bhatti, and S. Hailes. Mobility as an integrated service through the use of naming. In *Proceedings of 2nd ACM/IEEE international workshop on*

Mobility in the evolving internet architecture, MobiArch '07, pages 1:1–1:6, New York, NY, USA, 2007. ACM.

[10] R. Atkinson, S. Bhatti, and S. Hailes. A proposal for unifying mobility with multi-homing, nat, & security. In *Proceedings of the 5th ACM international workshop on Mobility management and wireless access*, MobiWac '07, pages 74–83, New York, NY, USA, 2007. ACM.

[11] R. Atkinson, S. Bhatti, and S. Hailes. Evolving the internet architecture through naming. *Selected Areas in Communications, IEEE Journal on*, 28(8):1319 –1325, october 2010.

[12] P. Bahl, R. Chandra, J. Padhye, L. Ravindranath, M. Singh, A. Wolman, and B. Zill. Enhancing the security of corporate wi-fi networks using dair. In *Proceedings of the 4th international conference on Mobile systems, applications and services*, MobiSys '06, pages 1–14, New York, NY, USA, 2006. ACM.

[13] A. Bakre and B. Badrinath. I-tcp: indirect tcp for mobile hosts. In *Distributed Computing Systems, 1995., Proceedings of the 15th International Conference on*, pages 136 –143, may-2 jun 1995.

[14] H. Balakrishnan, K. Lakshminarayanan, S. Ratnasamy, S. Shenker, I. Stoica, and M. Walfish. A layered naming architecture for the internet. In *Proceedings of the 2004 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '04, pages 343–352, New York, NY, USA, 2004. ACM.

[15] H. Ballani and P. Francis. Towards a global ip anycast service. In *Proceedings of the 2005 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '05, pages 301–312, New York, NY, USA, 2005. ACM.

[16] M. Bargh, B. Hulsebosch, H. Eertink, G. Heijenk, J. Idserda, J. Laganier, A. Prasad, and A. Zugenmaier. Reducing handover latency in future ip-based wireless networks: proxy mobile ipv6 with simultaneous bindings. In *World of Wireless, Mobile and Multimedia Networks, 2008. WoWMoM 2008. 2008 International Symposium on a*, pages 1 –10, 2008.

[17] M. Bauer, P. Bosch, N. Khrais, L. G. Samuel, and P. Schefczik. The umts base station router. *Bell Labs Technical Journal*, 11(4):93–111, Mar 2007.

[18] Y. Bejerano and I. Cidon. An anchor chain scheme for ip mobility management. *Wireless Networks*, 9:409–420, September 2003.

[19] V. Brik, A. Mishra, and S. Banerjee. Eliminating handoff latencies in 802.11 wlans using multiple radios: applications, experience, and evaluation. In *Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement*, IMC '05, pages 27–27, Berkeley, CA, USA, 2005. USENIX Association.

[20] M. Burrows and D. J. Wheeler. A block-sorting lossless data compression algorithm. Technical Report 124, Digital Equipment Corporation, 1994.

[21] M. Caesar, M. Castro, E. B. Nightingale, G. O'Shea, and A. Rowstron. Virtual ring routing: network routing inspired by dhts. In *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '06, pages 351–362, New York, NY, USA, 2006. ACM.

[22] M. Caesar, T. Condie, J. Kannan, K. Lakshminarayanan, and I. Stoica. Rofl: routing on flat labels. In *Proceedings of the 2006 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '06, pages 363–374, New York, NY, USA, 2006. ACM.

[23] P. Calhoun, J. Loughney, E. Guttman, G. Zorn, and J. Arkko. Diameter Base Protocol. RFC 3588 (Proposed Standard), Sept. 2003. Updated by RFCs 5729, 5719.

[24] A. Campbell, J. Gomez, S. Kim, A. Valko, C.-Y. Wan, and Z. Turanyi. Design, implementation, and evaluation of cellular ip. *Personal Communications, IEEE*, 7(4):42 –49, aug 2000.

[25] J. Cao, L. Zhang, H. Chan, and S. Das. Design and performance evaluation of an improved mobile ip protocol. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 1, pages 4 vol. (xxxv+2866), march 2004.

[26] C. Castelluccia. Hmipv6: A hierarchical mobile ipv6 proposal. *SIGMOBILE Mob. Comput. Commun. Rev.*, 4:48–59, January 2000.

[27] R. Chandra and P. Bahl. Multinet: connecting to multiple ieee 802.11 networks using a single wireless card. In *INFOCOM 2004. Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies*, volume 2, pages 882 – 893 vol.2, march 2004.

[28] R. Chandra, R. Mahajan, T. Moscibroda, R. Raghavendra, and P. Bahl. A case for adapting channel width in wireless networks. In *Proceedings of the ACM SIGCOMM 2008 conference on Data communication*, SIGCOMM '08, pages 135–146, New York, NY, USA, 2008. ACM.

[29] X. Chen and D. Qiao. Hand: Fast handoff with null dwell time for ieee 802.11 networks. In *INFOCOM, 2010 Proceedings IEEE*, pages 1 –9, march 2010.

[30] S. Cheshire, Z. Zhu, R. Wakikawa, and L. Zhang. Understanding apple's back to my mac (btmm) service. draft-zhu-mobileme-doc-05, March 2011.

[31] D. Clark. The design philosophy of the darpa internet protocols. In *Symposium proceedings on Communications architectures and protocols*, SIGCOMM '88, pages 106–114, New York, NY, USA, 1988. ACM.

[32] J. Cleary and I. Witten. Data compression using adaptive coding and partial string matching. *Communications, IEEE Transactions on*, 32(4):396 – 402, Apr. 1984.

[33] A. Conta and S. Deering. Generic Packet Tunneling in IPv6 Specification. RFC 2473 (Proposed Standard), Dec. 1998.

[34] S. Deering and R. Hinden. Internet Protocol, Version 6 (IPv6) Specification. RFC 2460 (Draft Standard), Dec. 1998. Updated by RFCs 5095, 5722, 5871.

[35] P. Deshpande, A. Kashyap, C. Sung, and S. R. Das. Predictive methods for improved vehicular wifi access. In *Proceedings of the 7th international conference on Mobile systems, applications, and services*, MobiSys '09, pages 263–276, New York, NY, USA, 2009. ACM.

[36] V. Devarapalli, R. Wakikawa, A. Petrescu, and P. Thubert. Network Mobility (NEMO) Basic Support Protocol. RFC 3963 (Proposed Standard), Jan. 2005.

[37] A. Dhraief and N. Montavont. Toward mobility and multihoming unification-the shim6 protocol: A case study. In *Wireless Communications and Networking Conference, 2008. WCNC 2008. IEEE*, pages 2840 –2845, 31 2008-april 3 2008.

[38] T. Dierks and E. Rescorla. The Transport Layer Security (TLS) Protocol Version 1.2. RFC 5246 (Proposed Standard), Aug. 2008. Updated by RFCs 5746, 5878, 6176.

[39] B. Donnet, B. Gueye, and M. Kaafar. A survey on network coordinates systems, design, and security. *Communications Surveys Tutorials, IEEE*, 12(4):488 –503, quarter 2010.

[40] R. Droms. Dynamic Host Configuration Protocol. RFC 2131 (Draft Standard), Mar. 1997. Updated by RFCs 3396, 4361, 5494.

[41] A. L. Dul. Global ip network mobility using border gateway protocol (bgp). http://www.quark.net/docs/Global_IP_Network_Mobility_using_BGP.pdf, March 2006.

[42] M. A. Ergin, K. Ramachandran, and M. Gruteser. Understanding the effect of access point density on wireless lan performance. In *Proceedings of the 13th annual ACM international conference on Mobile computing and networking*, MobiCom '07, pages 350–353, New York, NY, USA, 2007. ACM.

[43] M. A. Ergin, K. Ramachandran, and M. Gruteser. An experimental study of inter-cell interference effects on system performance in unplanned wireless lan deployments. *Comput. Netw.*, 52:2728–2744, October 2008.

[44] European Telecommunications Standards Institute. European digital cellular telecommunications system (phase 2); international mobile station equipment identities (imei) (gsm 02.16). *European Telecommunication Standard*, pages 1–9, 1994.

[45] D. Farinacci, T. Li, S. Hanks, D. Meyer, and P. Traina. Generic Routing Encapsulation (GRE). RFC 2784 (Proposed Standard), Mar. 2000. Updated by RFC 2890.

[46] B. Fenner, M. Handley, H. Holbrook, and I. Kouvelas. Protocol Independent Multicast - Sparse Mode (PIM-SM): Protocol Specification (Revised). RFC 4601 (Proposed Standard), Aug. 2006. Updated by RFCs 5059, 5796, 6226.

[47] D. Funato, K. Yasuda, and H. Tokuda. Tcp-r: Tcp mobility support for continuous operation. In *Network Protocols, 1997. Proceedings., 1997 International Conference on*, pages 229 –236, oct 1997.

[48] P. Georgopoulos, B. McCarthy, and C. Edwards. Towards a secure and seamless host mobility for the real world. In *Wireless On-Demand Network Systems and Services (WONS), 2011 Eighth International Conference on*, pages 134 –141, jan. 2011.

[49] A. Grilo, P. Estrela, and M. Nunes. Terminal independent mobility for ip (timip). *Communications Magazine, IEEE*, 39(12):34 –41, dec 2001.

[50] B. Gueye, A. Ziviani, M. Crovella, and S. Fdida. Constraint-based geolocation of internet hosts. *Networking, IEEE/ACM Transactions on*, 14(6):1219 –1232, 2006.

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

[52] A. Gupta, B. Liskov, and R. Rodrigues. Efficient routing for peer-to-peer overlays. In *NSDI'04: Proceedings of the 1st conference on Symposium on Networked Systems Design and Implementation*, pages 9–9, Berkeley, CA, USA, 2004. USENIX Association.

[53] G. K. Gupta and S. V. Raghavan. A framework for evolutionary networking. In *Proceedings of 2nd ACM/IEEE international workshop on Mobility in the evolving internet architecture*, MobiArch '07, pages 3:1–3:8, New York, NY, USA, 2007. ACM.

[54] A. Gurtov, M. Komu, and R. Moskowitz. Host identity protocol: Identifier/locator split for host mobility and multihoming. *The Internet Protocol Journal*, 12(1):27–32, Mar 2009.

[55] A. Gurtov, D. Korzun, A. Lukyanenko, and P. Nikander. Hi3: An efficient and secure networking architecture for mobile hosts. *Computer Communications*, 31:2457–2467, June 2008.

[56] H. Han, B. Sheng, C. Tan, Q. Li, and S. Lu. A measurement based rogue ap detection scheme. In *INFOCOM 2009, IEEE*, pages 1593 –1601, 2009.

[57] G. J. Heijenk, M. S. Bargh, J. Laganier, and A. R. Prasad. Reducing handover latency in future ip-based wireless networks: Fast proxy mobile ipv6. In *Second ERCIM workshop on eMobility, Tampere, Finland*, pages 79–92, Tampere, Finland, May 2008. Tampere University of Technology.

[58] R. Hsieh, Z. Zhou, and A. Seneviratne. S-mip: a seamless handoff architecture for mobile ip. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 3, pages 1774 – 1784 vol.3, march-3 april 2003.

[59] X. Hu, L. Li, Z. Mao, and Y. Yang. Wide-area ip network mobility. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 951 –959, april 2008.

[60] International Telecommunication Union. E.212. *Series E: Overall Network Operation, Telephone Service, Service Operation and Human Factors*, pages 1–28, 2008.

[61] J. Ioannidis, D. Duchamp, and G. Q. Maguire, Jr. Ip-based protocols for mobile internetworking. In *Proceedings of the conference on Communications architecture & protocols*, SIGCOMM '91, pages 235–245, New York, NY, USA, 1991. ACM.

[62] V. Jacobson, D. K. Smetters, J. D. Thornton, M. F. Plass, N. H. Briggs, and R. L. Braynard. Networking named content. In *Proceedings of the 5th international conference on Emerging networking experiments and technologies*, CoNEXT '09, pages 1–12, New York, NY, USA, 2009. ACM.

[63] D. Jen, M. Meisel, H. Yan, D. Massey, L. Wang, B. Zhang, and L. Zhang. Towards A New Internet Routing Architecture: Arguments for Separating Edges from Transit Core. In *Proceedings of the Seventh ACM Workshop on Hot Topics in Networks (HotNets-VII)*, October 2008.

[64] D. Johnson, C. Perkins, and J. Arkko. Mobility Support in IPv6. RFC 3775 (Proposed Standard), June 2004.

[65] S. Kandula, K. C.-J. Lin, T. Badirkhanli, and D. Katabi. Fatvap: aggregating ap backhaul capacity to maximize throughput. In *Proceedings of the 5th USENIX Symposium on Networked Systems Design and Implementation*, NSDI'08, pages 89–104, Berkeley, CA, USA, 2008. USENIX Association.

[66] D. Katabi and J. Wroclawski. A framework for scalable global ip-anycast (gia). In *Proceedings of the conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, SIGCOMM '00, pages 3–15, New York, NY, USA, 2000. ACM.

[67] E. Katz-Bassett, J. P. John, A. Krishnamurthy, D. Wetherall, T. Anderson, and Y. Chawathe. Towards ip geolocation using delay and topology measurements. In *Proceedings of the 6th ACM SIGCOMM conference on Internet measurement*, IMC '06, pages 71–84, New York, NY, USA, 2006. ACM.

[68] J. Kempf. Problem Statement for Network-Based Localized Mobility Management (NETLMM). RFC 4830 (Informational), Apr. 2007.

[69] S. Kent and R. Atkinson. IP Encapsulating Security Payload (ESP). RFC 2406 (Proposed Standard), Nov. 1998. Obsoleted by RFCs 4303, 4305.

[70] S. Kent and K. Seo. Security Architecture for the Internet Protocol. RFC 4301 (Proposed Standard), Dec. 2005. Updated by RFC 6040.

[71] C. Kim, M. Caesar, and J. Rexford. Floodless in seattle: a scalable ethernet architecture for large enterprises. In *Proceedings of the ACM SIGCOMM 2008 conference on Data communication*, SIGCOMM '08, pages 3–14, New York, NY, USA, 2008. ACM.

[72] R. Koodli. Fast Handovers for Mobile IPv6. RFC 4068 (Experimental), July 2005. Obsoleted by RFC 5268.

[73] R. Koodli. IP Address Location Privacy and Mobile IPv6: Problem Statement. RFC 4882 (Informational), May 2007.

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

[75] R. Koodli and C. E. Perkins. Fast handovers and context transfers in mobile networks. *SIGCOMM Comput. Commun. Rev.*, 31:37–47, October 2001.

[76] T. Koponen, M. Chawla, B.-G. Chun, A. Ermolinskiy, K. H. Kim, S. Shenker, and I. Stoica. A data-oriented (and beyond) network architecture. In *Proceedings of the 2007 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '07, pages 181–192, New York, NY, USA, 2007. ACM.

[77] S. Krishnamurthy, S. El-Ansary, E. Aurell, and S. Haridi. An analytical study of a structured overlay in the presence of dynamic membership. *Networking, IEEE/ACM Transactions on*, 16(4):814 –825, aug. 2008.

[78] D. Le, X. Fu, and D. Hogrefe. A review of mobility support paradigms for the internet. *Communications Surveys Tutorials, IEEE*, 8(1):38 –51, quarter 2006.

[79] D. Le, X. Fu, and D. Hogrefe. A cross-layer approach for improving tcp performance in mobile environments. *Wirel. Pers. Commun.*, 52:669–692, February 2010.

[80] D. Le, J. Lei, and X. Fu. A new decentralized mobility management service architecture for ipv6-based networks. In *Proceedings of the 3rd ACM workshop on Wireless multimedia networking and performance modeling*, WMuNeP '07, pages 54–61, New York, NY, USA, 2007. ACM.

[81] J. Lei and X. Fu. Evaluating the benefits of introducing pmipv6 for localized mobility management. In *Wireless Communications and Mobile Computing Conference, 2008. IWCMC '08. International*, pages 74 –80, aug. 2008.

[82] J. Li, J. Stribling, T. M. Gil, R. Morris, and M. F. Kaashoek. Comparing the performance of distributed hash tables under churn. In *IPTPS'04*, pages 87–99, 2004.

[83] J. Li, J. Stribling, R. Morris, M. Kaashoek, and T. Gil. A performance vs. cost framework for evaluating dht design tradeoffs under churn. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 1, pages 225 – 236 vol. 1, march 2005.

[84] L. E. Li, K. Tan, H. Viswanathan, Y. Xu, and Y. R. Yang. Retransmission &#8800; repeat: simple retransmission permutation can resolve overlapping channel collisions. In *Proceedings of the sixteenth annual international conference on Mobile computing and networking*, MobiCom '10, pages 281–292, New York, NY, USA, 2010. ACM.

[85] G.-H. Lu, S. Jain, S. Chen, and Z.-L. Zhang. Virtual id routing: a scalable routing framework with support for mobility and routing efficiency. In *Proceedings of the 3rd international workshop on Mobility in the evolving internet architecture*, MobiArch '08, pages 79–84, New York, NY, USA, 2008. ACM.

[86] E. K. Lua, J. Crowcroft, M. Pias, R. Sharma, and S. Lim. A survey and comparison of peer-to-peer overlay network schemes. *Communications Surveys Tutorials, IEEE*, 7(2):72 – 93, quarter 2005.

[87] L. Ma, A. Teymorian, and X. Cheng. A hybrid rogue access point protection framework for commodity wi-fi networks. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 1220 –1228, 2008.

[88] L. Ma, A. Y. Teymorian, X. Cheng, and M. Song. Rap: protecting commodity wi-fi networks from rogue access points. In *The Fourth International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness &#38; Workshops*, QSHINE '07, pages 21:1–21:7, New York, NY, USA, 2007. ACM.

[89] D. Malkhi, M. Naor, and D. Ratajczak. Viceroy: a scalable and dynamic emulation of the butterfly. In *Proceedings of the twenty-first annual symposium on Principles of distributed computing*, PODC '02, pages 183–192, New York, NY, USA, 2002. ACM.

[90] D. Maltz and P. Bhagwat. Msocks: an architecture for transport layer mobility. In *INFOCOM '98. Seventeenth Annual Joint Conference of the IEEE Computer and*

Communications Societies. Proceedings. IEEE, volume 3, pages 1037 –1045 vol.3, mar-2 apr 1998.

[91] V. Marques, X. Costa, R. Aguiar, M. Liebsch, and A. Duarte. Evaluation of a mobile ipv6-based architecture supporting user mobility qos and aaac in heterogeneous networks. volume 23, pages 2138 – 2151, Nov 2005.

[92] D. Massey, L. Wang, B. Zhang, and L. Zhang. A scalable routing system design for future internet. In *ACM SIGCOMM Workshop on IPv6 and the Future of the Internet*, aug 2007.

[93] P. Maymounkov and D. Mazières. Kademlia: A peer-to-peer information system based on the xor metric. In *Revised Papers from the First International Workshop on Peer-to-Peer Systems*, IPTPS '01, pages 53–65, London, UK, 2002. Springer-Verlag.

[94] M. Meisel, V. Pappas, and L. Zhang. Ad hoc networking via named data. In *Proceedings of the fifth ACM international workshop on Mobility in the evolving internet architecture*, MobiArch '10, pages 3–8, New York, NY, USA, 2010. ACM.

[95] U. Meyer and S. Wetzel. A man-in-the-middle attack on umts. In *Proceedings of the 3rd ACM workshop on Wireless security*, WiSe '04, pages 90–97, New York, NY, USA, 2004. ACM.

[96] U. Meyer and S. Wetzel. On the impact of gsm encryption and man-in-the-middle attacks on the security of interoperating gsm/umts networks. In *Personal, Indoor and Mobile Radio Communications, 2004. PIMRC 2004. 15th IEEE International Symposium on*, volume 4, pages 2876 – 2883 Vol.4, 2004.

[97] V. Mhatre and K. Papagiannaki. Using smart triggers for improved user performance in 802.11 wireless networks. In *Proceedings of the 4th international conference on Mobile systems, applications and services*, MobiSys '06, pages 246–259, New York, NY, USA, 2006. ACM.

[98] A. Mishra, M. Shin, and W. Arbaugh. An empirical analysis of the ieee 802.11 mac layer handoff process. *SIGCOMM Comput. Commun. Rev.*, 33:93–102, April 2003.

[99] A. Mishra, V. Shrivastava, D. Agrawal, S. Banerjee, and S. Ganguly. Distributed channel management in uncoordinated wireless environments. In *Proceedings of the 12th annual international conference on Mobile computing and networking*, MobiCom '06, pages 170–181, New York, NY, USA, 2006. ACM.

[100] I. Mitola, J. and J. Maguire, G.Q. Cognitive radio: making software radios more personal. *Personal Communications, IEEE*, 6(4):13 –18, aug 1999.

[101] Morgan Stanley Research. The Mobile Internet Report. http://www.morganstanley.com/techresearch/, Dec. 2009.

[102] R. Moskowitz and P. Nikander. Host Identity Protocol (HIP) Architecture. RFC 4423 (Informational), May 2006.

[103] R. Moskowitz, P. Nikander, P. Jokela, and T. Henderson. Host Identity Protocol. RFC 5201 (Experimental), Apr. 2008.

[104] S. Mtika and F. Takawira. Mobile ipv6 regional mobility management. In *Proceedings of the 4th international symposium on Information and communication technologies*, WISICT '05, pages 93–98. Trinity College Dublin, 2005.

[105] N. Neumann and X. Fu. Diameter webauth: An aaa-based identity management framework for web applications. In *Global Telecommunications Conference, 2008. IEEE GLOBECOM 2008. IEEE*, pages 1 –6, 30 2008-dec. 4 2008.

[106] N. Neumann, X. Fu, and G. Zhang. Ara: A routing and forwarding scheme for coordinated wide area mobility. In *Communications (ICC), 2010 IEEE International Conference on*, pages 1 –6, may 2010.

[107] N. Neumann, J. Lei, X. Fu, and G. Zhang. I-pmip: an inter-domain mobility extension for proxy-mobile ip. In *Proceedings of the 2009 International Conference on Wireless Communications and Mobile Computing: Connecting the World Wirelessly*, IWCMC '09, pages 994–999, New York, NY, USA, 2009. ACM.

[108] A. Nicholson, S. Wolchok, and B. Noble. Juggler: Virtual networks for fun and profit. *Mobile Computing, IEEE Transactions on*, 9(1):31 –43, jan. 2010.

[109] P. Nikander, A. Gurtov, and T. Henderson. Host identity protocol (hip): Connectivity, mobility, multi-homing, security, and privacy over ipv4 and ipv6 networks. *Communications Surveys Tutorials, IEEE*, 12(2):186 –204, quarter 2010.

[110] P. Nikander, T. Henderson, C. Vogt, and J. Arkko. End-Host Mobility and Multihoming with the Host Identity Protocol. RFC 5206 (Experimental), Apr. 2008.

[111] P. Nikander, J. Ylitalo, and J. Wall. Integrating security, mobility, and multihoming in a hip way. In *NDSS'03: Proceedings of the Network and Distributed Systems Security Symposium*, pages 87–99, Feb. 2003.

[112] E. Nordmark and M. Bagnulo. Shim6: Level 3 Multihoming Shim Protocol for IPv6. RFC 5533 (Proposed Standard), June 2009.

[113] V. N. Padmanabhan and L. Subramanian. An investigation of geographic mapping techniques for internet hosts. In *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '01, pages 173–185, New York, NY, USA, 2001. ACM.

[114] G. Pau, R. Wakikawa, and L. Zhang. Sail: A scalable approach for wide-area ip mobility. In *MobiWorld*, 2011.

[115] C. Perkins. IP Encapsulation within IP. RFC 2003 (Proposed Standard), Oct. 1996. Updated by RFC 3168.

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

[117] H. Petander, E. Perera, K.-C. Lan, and A. Seneviratne. Measuring and improving the performance of network mobility management in ipv6 networks. *Selected Areas in Communications, IEEE Journal on*, 24(9):1671 –1681, sept. 2006.

[118] H. Petander, E. Perera, A. Seneviratne, and Y. Ismailov. An experimental evaluation of mobile node based versus infrastructure based handoff schemes. In *World of Wireless, Mobile and Multimedia Networks, 2007. WoWMoM 2007. IEEE International Symposium on a*, pages 1 –4, june 2007.

[119] J. Postel. User Datagram Protocol. RFC 768 (Standard), Aug. 1980.

[120] J. Postel. Internet Protocol. RFC 791 (Standard), Sept. 1981. Updated by RFC 1349.

[121] J. Postel. Transmission Control Protocol. RFC 793 (Standard), Sept. 1981. Updated by RFCs 1122, 3168, 6093.

[122] B. Quinn and K. Almeroth. IP Multicast Applications: Challenges and Solutions. RFC 3170 (Informational), Sept. 2001.

[123] R. Raghavendra, E. M. Belding, K. Papagiannaki, and K. C. Almeroth. Understanding handoffs in large ieee 802.11 wireless networks. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07, pages 333–338, New York, NY, USA, 2007. ACM.

[124] M. Rahman and M. Atiquzzaman. Semo6 - a multihoming-based seamless mobility management framework. In *Military Communications Conference, 2008. MILCOM 2008. IEEE*, pages 1 –7, nov. 2008.

[125] K. Ramachandran, S. Rangarajan, and J. Lin. Make-before-break mac layer handoff in 802.11 wireless networks. In *Communications, 2006. ICC '06. IEEE International Conference on*, volume 10, pages 4818 –4823, june 2006.

[126] I. Ramani and S. Savage. Syncscan: practical fast handoff for 802.11 infrastructure networks. In *INFOCOM 2005. 24th Annual Joint Conference of the IEEE Computer and Communications Societies. Proceedings IEEE*, volume 1, pages 675 – 684 vol. 1, march 2005.

[127] V. Ramasubramanian and E. G. Sirer. Beehive: O(1) lookup performance for power-law query distributions in peer-to-peer overlays. In *1st Symposium on Networked Systems Design and Implementation*, NSDI 2004, pages 99–112, San Francisco, California, USA, Mar 2004.

[128] R. Ramjee, K. Varadhan, L. Salgarelli, S. Thuel, S.-Y. Wang, and T. La Porta. Hawaii: a domain-based approach for supporting mobility in wide-area wireless networks. *Networking, IEEE/ACM Transactions on*, 10(3):396 –410, jun 2002.

[129] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker. A scalable content-addressable network. In *Proceedings of the 2001 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '01, pages 161–172, New York, NY, USA, 2001. ACM.

[130] D. Rehunathan, R. Atkinson, and S. Bhatti. Enabling mobile networks through secure naming. In *Proceedings of the 28th IEEE conference on Military communications*, MILCOM'09, pages 2153–2160, Piscataway, NJ, USA, 2009. IEEE Press.

[131] Y. Rekhter, T. Li, and S. Hares. A Border Gateway Protocol 4 (BGP-4). RFC 4271 (Draft Standard), Jan. 2006.

[132] M. Riegel and M. Tüxen. Mobile sctp transport layer mobility management for the internet. *Transport*, pages 305–309, 2002.

[133] C. Rigney. RADIUS Accounting. RFC 2866 (Informational), June 2000. Updated by RFCs 2867, 5080, 5997.

[134] C. Rigney, S. Willens, A. Rubens, and W. Simpson. Remote Authentication Dial In User Service (RADIUS). RFC 2865 (Draft Standard), June 2000. Updated by RFCs 2868, 3575, 5080.

[135] J. Risson, A. Harwood, and T. Moors. Topology dissemination for reliable one-hop distributed hash tables. *Parallel and Distributed Systems, IEEE Transactions on*, 20(5):680 –694, may 2009.

[136] J. Ronan, S. Balasubramaniam, A. K. Kiani, and W. Yao. On the use of shim6 for mobility support in ims networks. In *Proceedings of the 4th International Conference on Testbeds and research infrastructures for the development of networks & communities*, TridentCom '08, pages 40:1–40:6, ICST, Brussels, Belgium, Belgium, 2008. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[137] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. SIP: Session Initiation Protocol. RFC 3261 (Proposed Standard), June 2002. Updated by RFCs 3265, 3853, 4320, 4916, 5393, 5621, 5626, 5630, 5922, 5954, 6026, 6141.

[138] V. Roth, W. Polak, E. Rieffel, and T. Turner. Simple and effective defense against evil twin access points. In *Proceedings of the first ACM conference on Wireless network security*, WiSec '08, pages 220–235, New York, NY, USA, 2008. ACM.

[139] A. I. T. Rowstron and P. Druschel. Pastry: Scalable, decentralized object location, and routing for large-scale peer-to-peer systems. In *Proceedings of the IFIP/ACM International Conference on Distributed Systems Platforms Heidelberg*, Middleware '01, pages 329–350, London, UK, 2001. Springer-Verlag.

[140] M. Sabeur, G. Alsukkar, B. Jouaber, D. Zeghlache, and H. Afifi. A new routing & mobility management solution for wireless mesh network. In *Mobile and Ubiquitous Systems: Networking Services, 2007. MobiQuitous 2007. Fourth Annual International Conference on*, pages 1 –3, aug. 2007.

[141] M. Särelä, J. Ott, and J. Ylitalo. Fast inter-domain mobility with in-packet bloom filters. In *Proceedings of the fifth ACM international workshop on Mobility in the evolving internet architecture*, MobiArch '10, pages 9–14, New York, NY, USA, 2010. ACM.

[142] Y. Sheng, K. Tan, G. Chen, D. Kotz, and A. Campbell. Detecting 802.11 mac layer spoofing using received signal strength. In *INFOCOM 2008. The 27th Conference on Computer Communications. IEEE*, pages 1768 –1776, 2008.

[143] M. Shin, A. Mishra, and W. A. Arbaugh. Improving the latency of 802.11 hand-offs using neighbor graphs. In *Proceedings of the 2nd international conference on Mobile systems, applications, and services*, MobiSys '04, pages 70–83, New York, NY, USA, 2004. ACM.

[144] W. Simpson. IP in IP Tunneling. RFC 1853 (Informational), Oct. 1995.

[145] A. C. Snoeren and H. Balakrishnan. An end-to-end approach to host mobility. In *Proceedings of the 6th annual international conference on Mobile computing and networking*, MobiCom '00, pages 155–166, New York, NY, USA, 2000. ACM.

[146] H. Soliman, C. Castelluccia, K. E. Malki, and L. Bellier. Hierarchical Mobile IPv6 Mobility Management (HMIPv6). RFC 4140 (Experimental), Aug. 2005. Obsoleted by RFC 5380.

[147] P. Srisuresh and M. Holdrege. IP Network Address Translator (NAT) Terminology and Considerations. RFC 2663 (Informational), Aug. 1999.

[148] R. Stewart. Stream Control Transmission Protocol. RFC 4960 (Proposed Standard), Sept. 2007. Updated by RFC 6096.

[149] R. Stewart, Q. Xie, M. Tuexen, S. Maruyama, and M. Kozuka. Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration. RFC 5061 (Proposed Standard), Sept. 2007.

[150] I. Stoica, D. Adkins, S. Zhuang, S. Shenker, and S. Surana. Internet indirection infrastructure. In *Proceedings of the 2002 conference on Applications, technologies, architectures, and protocols for computer communications*, SIGCOMM '02, pages 73–86, New York, NY, USA, 2002. ACM.

[151] I. Stoica, R. Morris, D. Liben-Nowell, D. R. Karger, M. F. Kaashoek, F. Dabek, and H. Balakrishnan. Chord: a scalable peer-to-peer lookup protocol for internet applications. *IEEE/ACM Trans. Netw.*, 11(1):17–32, 2003.

[152] T. Taleb, Y. Hadjadj-Aoul, and S. Schmid. Geographical location and load based gateway selection for optimal traffic offload in mobile networks. In *Networking*, 2011.

[153] W. L. Tan, F. Lam, and W. C. Lau. An empirical study on the capacity and performance of 3g networks. *Mobile Computing, IEEE Transactions on*, 7(6):737 –750, june 2008.

[154] The Institute of Electrical and Electronics Engineers, Inc. IEEE Standard for Local and Metropolitan Area Networks: Overview and Architecture. Amendment 2: Registration of Object Identifiers. *IEEE Std 802-2001 (Revision of IEEE Std 802-1990)*, page 0_1, 2002.

[155] The Institute of Electrical and Electronics Engineers, Inc. Ieee standard for information technology- telecommunications and information exchange between systems- local and metropolitan area networks- specific requirements part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Medium access control (mac) security enhancements. *IEEE Std 802.11i-2004*, pages 0_1 –175, 2004.

[156] The Institute of Electrical and Electronics Engineers, Inc. Ieee standard for local and metropolitan area networks- part 21: Media independent handover. *IEEE Std 802.21-2008*, pages c1 –301, 21 2009.

[157] The Institute of Electrical and Electronics Engineers, Inc. Ieee standard for local and metropolitan area networks - port-based network access control. *IEEE Std 802.1X-2010 (Revision of IEEE Std 802.1X-2004)*, pages C1 –205, 5 2010.

[158] S. Thomson and T. Narten. IPv6 Stateless Address Autoconfiguration. RFC 2462 (Draft Standard), Dec. 1998. Obsoleted by RFC 4862.

[159] G. Urdaneta, G. Pierre, and M. V. Steen. A survey of dht security techniques. *ACM Comput. Surv.*, 43:8:1–8:49, February 2011.

[160] A. G. Valkó. Cellular ip: a new approach to internet host mobility. *SIGCOMM Comput. Commun. Rev.*, 29:50–65, January 1999.

[161] A. Varga and R. Hornig. An overview of the omnet++ simulation environment. In *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops*, Simutools '08, pages 60:1–60:10, ICST, Brussels, Belgium, Belgium, 2008. ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering).

[162] R. Wakikawa and S. Gundavelli. IPv4 Support for Proxy Mobile IPv6. RFC 5844 (Proposed Standard), May 2010.

[163] R. Wakikawa, G. Valadon, and J. Murai. Migrating home agents towards internet-scale mobility deployments. In *CoNEXT '06: Proceedings of the 2006 ACM CoNEXT conference*, pages 1–10, New York, NY, USA, 2006. ACM.

[164] E. Wedlund and H. Schulzrinne. Mobility support using sip. In *Proceedings of the 2nd ACM international workshop on Wireless mobile multimedia*, WOWMOM '99, pages 76–82, New York, NY, USA, 1999. ACM.

[165] W. Wei, K. Suh, B. Wang, Y. Gu, J. Kurose, and D. Towsley. Passive online rogue access point detection using sequential hypothesis testing with tcp ack-pairs. In *Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, IMC '07, pages 365–378, New York, NY, USA, 2007. ACM.

[166] Wireless Application Protocol Forum Ltd. Wap 2.0 technical white paper. http://www.wapforum.org/what/WAPWhite_Paper1.pdf, Jan 2002. retrieved April 2011.

[167] B. Wong, I. Stoyanov, and E. G. Sirer. Geolocalization on the internet through constraint satisfaction. In *Proceedings of the 3rd conference on USENIX Workshop on Real, Large Distributed Systems - Volume 3*, pages 1–1, Berkeley, CA, USA, 2006. USENIX Association.

[168] W. Xing, H. Karl, and A. Wolisz. M-sctp: Design and prototypical implementation of an end-to-end mobility concept. In *5th Intl. Workshop The Internet Challenge: Technology and Applications*, Berlin, Germany, Oct 2002.

[169] F. Xu, C. Tan, Q. Li, G. Yan, and J. Wu. Designing a practical access point association protocol. In *INFOCOM, 2010 Proceedings IEEE*, pages 1 –9, march 2010.

[170] L. Zhang, R. Wakikawa, and Z. Zhu. Support mobility in the global internet. In *Proceedings of the 1st ACM workshop on Mobile internet through cellular networks*, MICNET '09, pages 1–6, New York, NY, USA, 2009. ACM.

[171] B. Zhao, L. Huang, J. Stribling, S. Rhea, A. Joseph, and J. Kubiatowicz. Tapestry: a resilient global-scale overlay for service deployment. *Selected Areas in Communications, IEEE Journal on*, 22(1):41–53, Jan. 2004.

[172] Z. Zhou, A. Seneviratne, R. Chan, and P. Chumchu. A software based indoor relative location management system. In *Wireless and Optical Communications*, WOC 2002, 2002.

[173] S. Zhuang, K. Lai, I. Stoica, R. Katz, and S. Shenker. Host mobility using an internet indirection infrastructure. In *Proceedings of the 1st international conference on Mobile systems, applications and services*, MobiSys '03, pages 129–144, New York, NY, USA, 2003. ACM.

[174] J. Ziv and A. Lempel. A universal algorithm for sequential data compression. *Information Theory, IEEE Transactions on*, 23(3):337 – 343, May 1977.

A. Curriculum Vitae

Personal Data

Name:	Niklas Neumann
Date of Birth:	11 October 1977
Address:	Vor der Laakenbreite 4 37075 Göttingen Germany

Research Interests

- Mobility
- Network Security
- Identity Management

Education

since November 2007	PhD student in Applied Computer Science Computer Networks Group University of Göttingen
November 2007	Master of Applied Computer Science University of Göttingen, Germany
March 2006	Bachelor of Applied Computer Science University of Göttingen, Germany
July 2001	Intermediate Diploma in Information Management University of Göttingen, Germany

Work Experience

since November 2007	Research Assistant Computer Networks Group University of Göttingen

Publications

- **ARA: A Routing and Forwarding Scheme for Coordinated Wide Area Mobility**, Niklas Neumann, Xiaoming Fu, Gong Zhang, IEEE International Conference on Communications (ICC) 2010, Cape Town, South Africa, IEEE, June 2010

- **I-PMIP: An Inter-Domain Mobility Extension for Proxy Mobile IP**, Niklas Neumann, Jun Lei, Xiaoming Fu, Gong Zhang, in the Proceedings of 5th International Wireless Communications and Mobile Computing Conference (IWCMC 2009), Leipzig, Germany, ACM Digital Library, June 2009

- **Diameter WebAuth: An AAA-based Identity Management Framework for Web Applications**, Niklas Neumann, and Xiaoming Fu, The 51th Annual IEEE Global Telecommunications Conference (GLOBECOM 2008), Computer and Communications Network Security Symposium, New Orleans, LA, USA, IEEE, December 2008

- **Decoupling Congestion Control Using Traffic Aggregates and Middleboxes**, Niklas Neumann, Ralf Lübben, Mayutan Arumaithurai, and Xiaoming Fu, IEEE International Conference on Network Protocols (ICNP 2008), poster session, Orlando, FL, USA, October 2008