HELSINKI UNIVERSITY OF TECHNOLOGY
Department of Electrical and Communications Engineering

Marko Repo

Supporting Group Mobility in Mission-Critical Wireless Networks for SIP-based Applications

Diploma thesis submitted for evaluation in fulfillment of the requirements for the degree of Master of Science in Technology

Espoo, Finland, Monday 29th September, 2008

Thesis supervisor: Prof. Jörg Ott
Thesis instructor: Lic.Sc. (Tech.) Markus Peuhkuri

Author: Marko Repo
Thesis title: Supporting Group Mobility in Mission-Critical Wireless Networks for SIP-based Applications
Date: 29/09/2008 **No. Pages:** 10 + 89

Department: Department of Electrical and Communications Engineering
Professorship: Networking Technology **Code:** S-38

Supervisor: Prof. Jörg Ott
Instructor: Lic.Sc.(Tech) Markus Peuhkuri

Abstract:

This thesis studies the provision of group mobility during inter-domain hand-offs for delay-sensitive SIP applications over wireless IPv4/IPv6 network environment, based on the IEEE 802.11x platform. In contemporary disaster relief operations, the role of real-time communications has been strongly escalating over the recent years. The communication systems used for these ends have been conventionally very expensive. The rapid evolution of wireless technologies has brought the focus of interest to the affordable Common-Off-the-Shelf civilian applications.

Long latencies during hand-offs for real-time traffic are a very important problem. As the studies have pointed out, the VoIP-based voice traffic can withstand maximum approximate disruption times of 100 ms, without too high degradation in the quality of service. Along with the link-layer hand-off, the duplicate address detection procedure during DHCP address acquisition and the SIP connection re-establishment both have a major impact on the hand-off latency.

The group mobility has gained high attention in the research of ad-hoc networks. The work studies the benefits that this scheme could possibly bring over the conventional hand-offs in hierarchical infrastructured SIP networks.

Different approaches to application-level mobility and the signaling efficiency are examined from the viewpoint of bandwidth usage and network security. In the experimental part, group hand-offs are modeled in a simple, simulated environment. In addition, a numerical analysis is used to assess the hand-off performance to support the made conclusions.

Keywords: Mobility management, Group mobility, Group hand-off, Predictive Address Reservation, SIP

TEKNILLINEN KORKEAKOULU TIIVISTELMÄ

Tekijä: Marko Repo

Otsikko: SIP-sovellusten ryhmäliike tehtäväkriittisissä langattomissa verkoissa

Päiväys: 29-09-2008 **Sivujen lkm:** 10 + 89

Osasto: Sähkö- ja tietoliikennetekniikan osasto

Professuuri: Tietoverkkotekniikka **Koodi:** S-38

Valvoja: Prof. Jörg Ott

Ohjaaja: TkL Markus Peuhkuri

Tiivistelmä:

Diplomityössä tarkastellaan viiveherkkien SIP-sovellusten verkkoalueiden välistä ryhmäliikkuvuutta langattomissa, IEEE 802.11x -pohjaisissa IPv4/IPv6 verkkoympäristöissä. Nykyaikaisissa kriisinhallintatehtävissä reaaliaikaisen viestinnän merkitys on viime vuosina vahvasti korostunut. Tähän tarkoitukseen käytetyt viestintäjärjestelmät ovat olleet tavallisesti erittäin kalliita. Langattomien teknologioiden nopea kehitys on kuitenkin suunnannut mielenkiinnon edullisiin, kaupallisiin siviilipuolen valmisratkaisuihin.

Pitkät yhteydensiirtoviiveet ovat tärkeä ongelma reaaliaikaliikenteen yhteydensiirron kannalta. VoIP-pohjaisen puheliikenteen on todettu kestävän enimmillään suuruusluokkaa 100 ms olevia viiveaikoja palvelunlaadun ratkaisevasti kärsimättä. Linkkitason yhteydensiirron ohella duplikaattiosoitteiden tarkistuksella DHCP-osoitteenhaun aikana ja SIP-yhteyden uudelleenmuodostuksella on saumattoman yhteydensiirron kannalta olennainen merkitys.

Ryhmäliikkuvuus on saanut osakseen paljon huomiota ad hoc -verkkojen tutkimuksessa. Työssä tutkitaan mahdollisesti saavutettavia hyötyjä, joita ryhmäliikkuvuusmalli pystyisi perinteiseen yhteydensiirtotapaan nähden tuomaan hierarkkisissa infrastruktuurisissa SIP-verkoissa.

Sovellustason liikkuvuutta ja signaloinnin tehokkuutta tarkastellaan kaistankäytön ja tietoturvallisuuden näkökulmasta. Kokeellisessa osiossa pyritään mallintamaan ryhmäyhteydensiirtoja yksinkertaisessa, simuloidussa ympäristössä. Päätelmien tueksi yhteydensiirtojen suorituskykyä arvioidaan lisäksi numeerisella analyysilla.

Avainsanat: liikkuvuudenhallinta, ryhmäliikkuvuus, ryhmäyhteydensiirto, ennakoiva osoitteenvaraus, SIP

Preface

This thesis was carried out as continuation for the project LaTe funded by the Finnish Defence Forces, being mostly written at the premises of the Networking Laboratory of Helsinki University of Technology during the years 2006–2008. The text was completely created using LaTeX typesetting system.

Today, when looking out of my workroom window, I think over the past years, about this long march. The years spent here have taught me a lot. About aspiration, hard work, joy of achievement, importance of friends, and finally — humility. Nothing good comes easy.

I want to express my greatest gratitude towards all the personnel of the laboratory – Lic.Sc. Markus Peuhkuri in particular for constantly providing his valuable assistance and information when needed, Professor Jörg Ott for his worthwhile ideas and advice, my room colleague Antti Mäkeläinen for so many moments of good conversations, and everybody else for all those bad jokes during the coffee hours.

And finally, I want to thank my little Tanja for giving me her greatest caring and support over all these dark, long months of persistent effort.

Leppävaara, Espoo
Monday 29th September, 2008

Marko Repo

Contents

Abbreviations

3GPP	3rd Generation Partnership Project
AAA	Authentication, Authorization & Accounting
ACK	Acknowledgment
AD	Administrative Domain
ALG	Application Layer Gateway
AoR	Address-of-Record
AP	Access Point
AR	Access Router
ARP	Address Resolution Protocol
AUC	Address Usage Collector
B2BUA	Back-to-Back User Agent
BG	Border Gateway
BOOTP	Bootstrap Protocol
BS	Base Station
BSS	Basic Service Set
CBR	Constant Bit Rate
CIP	Cellular IP
CN	Correspondent Node
COTS	Commercial Off-the-Shelf
DA	Global Domain Address
DAD	Duplicate Address Detection
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Server
DoS	Denial of Service
FA	Foreign Agent
FTP	File Transfer Protocol
GHO	Group Hand-off
GM	Group Mobility
HA	Home Agent
HID	Hierarchical ID
HMSIP	Hierarchical Mobile SIP
HOAS	Hand-off Assistive Server
HSR	Hierarchical State Routing
HTTP	Hypertext Transfer Protocol
IEEE	Institute of Electrical and Electronics Engineer
IETF	Internet Engineering Task Force
IMS	IP Multimedia Subsystem
IP	Internet Protocol
ITU	International Telecommunication Union

L2HO	Link-layer Hand-Off
LA	Local Address
LDAP	Lightweight Directory Access Protocol
MAC	Media Access Control
MIME	Multi-purpose Internet Mail Extension
MA	(SIP) Mobility Agent
MIP	Mobile IP
MM	Mobility Manager
MSS	Maximum Segment Size
MN	Mobile Node
MIME	Multipurpose Internet Mail Extensions
NAPTR	Naming Authority Pointer
NAT	Network Address Translation
NGN	Next Generation Network
PAR	Predictive Address Reservation
PCM	Pulse Code Modulation
QoS	Quality of Service
RFC	Request for Comments
RP	Reference Point
RPGM	Reference Point Group Mobility
RS	Router Solicitation
RTCP	Real-time Control Protocol
RTP	Real-time Transport Protocol
RTSP	Real-time Streaming Protocol
RTT	Round-trip Delay Time
SDP	Session Description Protocol
SBC	Session Border Controller
SBS	Serving Base Station
SCTP	Stream Control Transmission Protocol
SigComp	Signaling Compression
SIMPLE	SIP for Instant Messaging and Presence Leveraging Extensions
SIP	Session Initiation Protocol
SLP	Service Location Protocol
SMS	Short Message Service
SNR	Signal-to-Noise Ratio
SRV	Service Record
SS7	Signaling System #7
STUN	Simple Traversal of UDP Through NATs
TBS	Target Base Station
TCP	Transmission Control Protocol
TETRA	Terrestrial Trunked Radio

TLS	Transport Layer Security
TTL	Time To Live
TU	Transaction User
UA	User Agent
UAC	User Agent Client
UAS	User Agent Server
UDP	User Datagram Protocol
UDVM	Universal Decompressor Virtual Machine
UMTS	Universal Mobile Telecommunications System
URI	Uniform Resource Identifier
VIRVE	Viranomaisradioverkko ("Authority Radio Network")
VoIP	Voice over IP
WEP	Wireless Equivalent Privacy
WiMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless Local Area Network
WPA	Wi-Fi Protected Access

1 Introduction

1.1 Background

In contemporary disaster relief and crisis management, regardless of whether the setting requires the involvement of military, medical, fire & rescue, border guard or law enforcement authorities, minimizing human casualties becomes the first priority. The capability of acquiring, processing and producing updated and time-synchronized information is a mandatory pre-requisite for appropriate operation planning, effective decisions and rapid reaction to unexpected events in a dynamically changing, perilous environment. The heavy-duty communications equipment customarily used for these purposes in crisis zones have been without exceptions expensive and their evolution taken completely separate paths from those applications widely utilized in the civilian world.

A good example is the authority radio network VIRVE, a project launched in 1997 by the Ministry of the Interior of Finland and deployed by Nokia, providing coordinated collaboration properties for governmental agencies in Finland. The architecture is fundamentally based on the European TETRA standard, enabling multiple different communication modes between an arbitrary number of participants.

Mobile wireless networks are expected to play increasingly important role in future civilian and military settings. The proliferation of IEEE 802.11 technologies has been pushing the prices down, and pervasive wireless networks enable access to real-time communication services everywhere. This development has caused the advancement in military communication systems to undergo a partial paradigm shift from legacy technologies to embrace the usage of Commercially Off-the-Shelf (COTS) applications. Military doctrines are now endorsing the usage of ubiquitous, affordable and easily disposable civilian networking technologies [38].

However, guaranteeing flexible mobility with seamless session continuity and node reachability at the same time forms an enormous challenge. High hand-off latencies typically cause the session to terminate or suffer a considerable degradation. This is not acceptable in battlefield conditions, when human lives are on stake. The term *handover* can be interchangeably used in place of *hand-off*. We prefer the latter for the sake of generality: the previously mentioned term, adopted in use by 3GPP, is primarily used in the context of cellular telecommunication technologies such as GSM and UMTS.

The Internet consists of a packet-switched Internet Protocol (IP) core interconnecting multiple heterogeneous access networks. In other words, the network core serves as a fabric of invisible, packet forwarding bit pipes, connecting together millions intelligent terminals locating at the network edge. Since the data priorization is not properly supported, the Internet Protocol has some critical shortcomings from the viewpoint of delay sensitive applications such as the real-time audio and video. A fundamental problem with the conventional IP is that it does not natively address the need for terminal mobility. Which more important, the IP protocol guarantees neither hard security nor QoS, which can be seen as absolute preconditions for military usage.

A vast number of publications pertaining to the network mobility have been issued and innumerable engineering hours done on different protocol stack layers in the pursuit for a seamlessly working mobility mechanism. Attempts to solve the mobility issue on the network layer are numerous, the oldest of which probably being the Mobile IP (MIP) standard defined in RFC3344 [29] by IETF. MIP is an IP enhancement aiming at realizing user mobility through dynamically allocated care-of-addresses associated with the visited network, allowing nodes to change their point-of-attachment without affecting their actual home IP address. However, this technique suffers from several sub-optimalities such as increased hand-off delays and *triangular routing*, a well-known phenomenon meaning that the packets transmitted by *mobile nodes* (MN) are sent directly to the *correspondent node* (CN) when the traffic optimization is in use, while the traffic in opposite direction is encapsulated and tunneled through a *home agent* (HA). Though applicable when used for data applications with loose delay requirements, the tunneling process incurs latency that cannot be normally accepted for real-time traffic.

It has been suggested that mobility could be effectively managed on the application level. The Session Initiation Protocol (SIP) is considered to be a suitable protocol in terms of performance and flexibility. Over the years, SIP has undergone much development, introducing many new features and applications. The protocol supports user mobility natively; users moving around and accessing the network in different locations can be easily reached using their ordinary contact addresses, regardless of their actual physical location in the network. However, while the basic SIP protocol can be extended with terminal mobility management capacity, an adequate QoS cannot be stipulated for applications with stringent delay requirements. The hand-off latencies have typically not been an issue in non-realtime data applications such as FTP or HTTP, but for real-time streaming applications

ensuring strict upper bounds for data delay and packet loss becomes of extreme importance. Minimizing the packet loss becomes a critical design objective in order to achieve *seamless hand-offs*, i.e. hand-offs having ability to preserve all open connections for each application currently running on the terminal without any major interruptions in the user experienced end-to-end service. Substantial efforts have been made to push the hand-off latencies down to the millisecond range and make SIP a suitable protocol for mobile real-time application signaling.

The channel utilization is another factor having a major impact on hand-off delays in the real-world narrowband networks. The protocol efficiency can be improved by using certain compression schemes. Often in task-oriented scenarios, however, it is also possible to exploit the knowledge of mobility patterns of the involved network participants to optimize the signaling and avoid unnecessary traffic. The observation that the network participants can be considered as logically clustered groups rather than individual nodes gives rise to the concepts of *group motion* and *group mobility*, originally emerging by the advent of ad-hoc networks. Group mobility models allow predicting the short-term usage of network resources, potentially leading to increased signaling performance in infrastructured networks.

1.2 Objectives and methodology

The goal of this exploratory work is to evaluate the performance of existing mobility schemes implemented in SIP signaling networks, particularly as it comes to their ability in minimizing the signaling load and latency that inevitably occur during inter-domain hand-offs. We assess the benefits that could be gained by supporting group mobility for applications with high intolerance for delays, such as Voice over IP (VoIP). Finally, the theoretical observations are completed by the means of queuing theory analysis and computer-based simulation. For limited space, the core focus is kept on mobility and signaling, while some other important aspects such as the network security and AAA (Authentication, Authorization and Accounting) are merely outlined with brevity.

The traffic flows are assumed to happen in the IPv4/IPv6 environment built on 802.11x (WLAN) infrastructure networks. The mobility management is considered to be handled by the network, while end terminals are assumed to have little or no intelligence at all. This is an important assumption, since centralized resource management generally facilitates implementing the QoS, security and *lawful interception* functions in a network.

1.3 Thesis structure

The thesis is divided into six chapters. The remainder of the document is structured as follows:

Chapter 2 explains the history and basic functionality of Session Initiation Protocol, giving a big picture of the protocol structure and performance. After introducing the reader to the IETF RFC 3261 standard and some of its common extensions, we will evaluate briefly its shortcomings as a mobile signaling protocol.

Chapter 3 focuses on practical requirements and methods for managing mobility in SIP protocol. The chapter outlines some proposals to improve hand-off performance in terms of latency at different protocol layers.

Chapter 4 discusses the concepts of group mobility and group hand-offs, explaining the benefits and challenges that introducing group mobility into infrastructured wireless networks could bring. At the end of the chapter, we present a concrete example how to bring the group mobility mechanism as a part of the SIP.

Chapter 5 concentrates on finding the analytical means needed for modeling the hand-off performance quantitatively. The group hand-offs are studied mathematically within a queueing theoretical context.

Chapter 6 describes the proceedings and results of the simulation written in Java language specifically for evaluating hand-off performances in simple mobility scenarios. The simulation software was featured as a part of the thesis.

Chapter 7 concludes the thesis, providing the reader with a summary of the most crucial observations, and elaborating on the required future work.

Each chapter is preceded by a brief introduction into its contents and followed by a chapter summary recapitulating the major findings.

2 Session Initiation Protocol

This chapter delineates the history and functioning of the Session Initiation Protocol by having an overview on the RFC 3261 standard. The reader gets familiar with the rudimentary concepts of the protocol and its structure. The emphasis is put on the overall performance and mobility characteristics for VoIP and data applications.

2.1 Overview on SIP and RFC 3261

The Session Initiation Protocol (SIP) was developed as a general purpose protocol to provide an application level signaling mechanism for creating, modifying and terminating user sessions between two or more participants, where the session may consist of any form of data exchange between the parties. The roots of the protocol are in the Internet community rather than the telecommunications industry. The protocol functioning is detailedly specified in the IETF RFC 3261 [33], which superseded the now obsolete original standard (RFC 2543 [41]) from the year 1999 by improving it in many details, such as making the support for the Transmission Control Protocol (TCP) mandatory for all SIP elements. In addition, a substantial number of separate RFCs have been written to define the correct behavior of SIP utilizing applications. The Appendix A contains an extensive but not exhaustive list of relevant SIP related documents.

IETF developed SIP essentially as a text-based protocol for connecting multimedia calls. Using UTF-8 encoding in its syntax, the protocol utilizes request-response transaction model not much unlike that of HTTP 1.1. This has important consequences considering the data security and performance. As a flexible protocol, SIP has recently made significant inroads into the VoIP technology, gradually displacing the previously dominant ITU-T H.323 standard, that is considered now by some to be too complicated to evolve in practice [17]. Additionally, SIP has spawned multiple new projects and workgroups to introduce new features such as extensive support for presence indication, instant messaging and SMS-styled short messages while keeping the protocol interoperable backwards at the same time. One of such open protocol suites is SIMPLE [6]. The recent years' development, however, has caused the protocol evolution partially to miss the original idea of an extremely light and easy-to-read protocol: it has in many ways already reached H.323 in profoundness.

The Session Initiation Protocol currently supports the most of features commonly seen in standard telephone signaling systems. It is currently

being used widely in many highly popular instant messaging applications such as Microsoft Messenger based on SIMPLE. The protocol will be playing considerably more important role in the near future, since it was approved as a part of 4G NGN/IMS by 3GPP. The SIP Community performs regular interoperability testing events to ensure the enduring development and the future credibility of the protocol.

The SIP protocol is typically characterized by very low call setup times and the post-dial delay, reportedly being as low as 1.5×RTT (Round-trip Delay Time) when using User Datagram Protocol (UDP). As transport layer independent, it can be used virtually on any currently existing transport protocol: TCP, UDP (User Datagram Protocol) and SCTP (Stream Control Transmission Protocol), a few to mention. It gives a support for both unicast and multicast communications. In the most typical scenario SIP is used to establish and tear down real-time voice or video sessions. The protocol provides none of actual services itself, but merely the connection establishing primitives necessary for implementing these services. In particular, all real-time communications must be handled through separate protocol. When high bandwidth, fast hand-offs and low latency do matter, RTP on UDP is commonly being used. [45, 40]

Since SIP functions at the application layer, there is no need for changes in the IP stack residing underneath (see Figure 2.1). This property makes the protocol portable: SIP applications can be found nowadays in several mobile devices with sufficient memory and processing capabilities. SIP can be made to work in most packet switched networks where the intelligence is located at the network edge. This particularly contrasts SIP from traditional telephone systems such as SS7 where all the features are located in the network. SIP is also intercompatible with other major existing signaling systems: ITU-T specifications H.323, Q.931 (ISDN) and ISUP (SS7).

2.2 SIP network entities

A typical SIP scenario, a *session*, can be modeled as a set of participants and data streams (request/response based transactions) flowing from senders to receivers. A *dialog* represents ephemeral peer-to-peer relationship between two SIP user agents existing for a certain time.

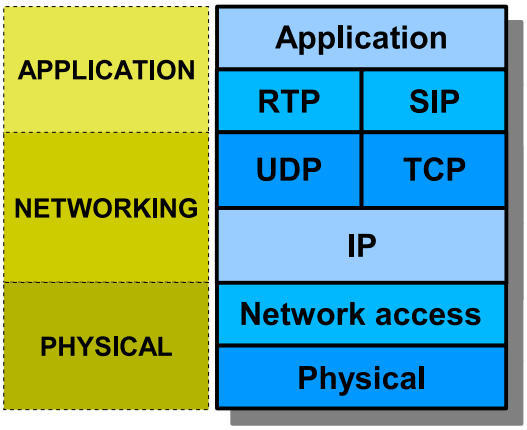


Figure 2.1: SIP architecture illustrated

SIP defines several logical entities. However, there does not need to be any physical difference between the server entities whatsoever; any SIP server can be logically configured to take role of any other entity. The functionality of entities such as proxy, registrar and location service could physically reside in a single device.

User Agents

User Agents (UA) originate and terminate requests. User agent clients (UAS) generate SIP requests and user agent servers (UAS) are capable of receiving, rejecting or redirecting them. UAs are the only SIP entities where signaling requests and media streams converge.

Proxy Servers

Proxies are entities which forward SIP messages towards a callee. If a proxy knows the exact callee address, it establishes a direct connection to deliver the message. In other case, it forwards the connection request to another SIP entity closer to the destination, functioning much like a router. Proxy servers are classified into two classes: *stateful* and *stateless*. The principal differences between these types are discussed later. Proxies may also act as implementation points for a variety or network policies and routing rules. Additionally, it is possible to provide user experience improving features by implementing them into proxy servers.

Redirect Servers

Redirect servers are particular UASs, which are capable of returning 3xx responses to the requests it receives from UACs, directing the caller to try an alternative set of addresses to contact the callee. This is the case, for instance, when the callee has moved temporarily outside of the home network. Redirect servers keep themselves out of the actual messaging loop by returning the routing information to the originating client instead of pushing the request forward. This generally allows for an improved network scalability.

Registrars

Registrars function as the interfaces to the location services for a domain, being able to receive REGISTER requests from user agents, then mapping SIP URIs and other relevant information obtained from these requests to the location service. Registrars also manage the user address translation, mapping global SIP URIs to actual user locations. This way, registrars provide a basic means for mobility management. User agents can be registered in several locations simultaneously.

Location Servers

Location servers are abstract entities which maintain the user location databases containing address-of-record (AoR) entries of user contact addresses. The entries, so called *bindings*, are updated by registrars any time when users join or leave the network. Location servers are not actually specified as a part of SIP architecture, and the protocol between SIP servers and location servers is not specified in the SIP standard. The location server may be for instance a local SQL database, directory services based on LDAP or X.500 protocols, or merely data obtained using *finger*/*whois* commands.

Back-to-Back User Agents

Back-to-Back User Agents (B2BUA) are much characterized by the properties of both UACs and UASs, capable of receiving requests like UASs and determining answers to them acting like UACs. They have practical value in service creation for call management, networks interconnection and hiding the internal network functionality from users.

2.3 SIP functional layers

The SIP protocol consists of a independent, loosely coupled processing stages. They are usually depicted as layers solely for presentation purposes, having little significance on how the actual protocol should be implemented.

Starting from the bottom of the SIP protocol stack (see Figure 2.2), the lowest layer is called *syntax and encoding* which contains the protocol syntax and parsing rules, which generally follow Backus-Naur Form grammar. [33]

The second level is called *transport layer* which defines the behaviour of UAs and how they communicate over the network by requests and responses. This involves determination of the used connection for each request or response. Every SIP entity must contain this layer.

The third layer above transport layer is called *transaction layer*, which handles application-layer re-transmissions, matches arriving responses to corresponding requests, and deals with application-layer timeouts. All UAs and stateful proxies always contain a transaction layer, whereas stateless proxies do not. Transaction layers can be seen as finite state machines handling various types of requests.

The fourth and the topmost layer is the *transaction user layer* (TU), which contains the UAC, UAS and proxy core functions, plus capability of creating new requests (client transaction instances) and sending them forward along with an IP address, a port number and the used transport protocol. TUs are furthermore able to CANCEL requests generated by them. All SIP entities with exception of stateless proxies have TU layer.

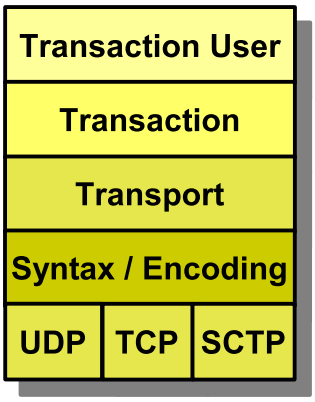


Figure 2.2: SIP functional layers[14]

2.4 SIP message structure

SIP messages share similarity with certain other text-based protocols, e.g. HTTP. Requests and responses consist of a start line defining the message name, header lines describing the parameters relevant to the working of the SIP protocol and a MIME body which describes the session more precisely, following immediately underneath the header lines, separated from them by an empty line. The following example describes a valid SIP header part:

```
INVITE sip:haggis@mil.fi SIP/2.0
Via: SIP/2.0/UDP esikunta.mil.fi;branch=z9hG4bKnashds8
Max-Forwards: 32
To: Gustav <sip:haggis@mil.fi>
From: Juhani <sip:kaski@mil.fi>;tag=1928301774
Call-ID: a84b4c76e66710@esikunta.mil.fi
CSeq: 314159 INVITE
Contact: <sip:juhani@esikunta.mil.fi>
Content-Type: application/sdp
Content-Length: 128

... MESSAGE BODY ...
```

2.4.1 Header field semantics

The RFC 3261 defines 44 different header fields, out of which we shall present briefly the most relevant ones.

To: specifies the logical recipient of the request in question, which may but does not have to be the ultimate destination of the request. The field may additionally include a display name to be shown in a human user interface, but this is optional. It may also contain a *tag*-value, which is used together with Call-ID field to identify a particular dialog.

From: specifies the initiator of the request. It follows the same format as *To:* field, containing a sender URI and an optional display name. Each request must use a new tag-value, that is always created on the client side.

Call-ID: helps to identify a session, request or registration uniquely. It facilitates matching the requests and responses belonging to a particular dialog, and detecting duplicate requests. It must be globally unique in regard of time and space for every message sent within an ongoing dialog. The field usually consists of a unique ID-number built by a UA, concatenated with a

host part suffix separated with a '@'-sign.

Via: defines where the response should be sent and indicates the used transport protocol; *branch* parameter is mandatory, it identifies a transaction generated by the request and used e.g. by the loop detection mechanism. All requests generated by UAs must have unique values in regard of time and space, with exception of CANCEL and ACK requests for other than 2xx responses. Via fields in all requests conforming to RFC3261 must start with letters '*z9hG4bK*', a so called magic cookie to ensure global unambiguity and facilitate servers recognizing the used standard from older, interim RFC2543 standard that does not require spatial and chronological uniqueness.

Max-Forwards: is an integer restricting the number of hops via proxies or gateways to stay within given bounds. On each hop the integer will be reduced. If the value ever reaches zero before reaching the recipient, the message is rejected and response 483 Too Many Hops generated. RFC 3261 recommends this field to be value 70 initially when a request is generated. The functioning of the field is in analogy to the TTL field used in Internet Protocol packets.

Contact: specifies the UA instance to which requests should be sent in future. Contact must be present in any INVITE messages, and has to contain explicitly one SIP or SIPS URI. It usually consists of a user name and domain.

CSeq: addresses and orders the transactions unambiguously. It consists of a sequence number and a request name. The sequence number is incremented every time when a new request is generated within the dialog.

Content-Type: defines the message body information type.

Content-Length: contains the length of payload data, i.e. the length of message body in octets.

Route: forces the request to visit a given set of locations (usually proxies) on its way towards the final destination.

Record-Route: is a field inserted by some proxies to force future requests associated with the same dialog to be routed through the proxy.

Require: is used by some UACs to inform UASs about specific options that must be supported by the UAS in order to process the request.

2.5 SIP message bodies

For session initialization purposes, SIP basically works as a carrier for the Session Description Protocol (SDP) [11, 26], which describes the session itself by carrying the media content information and required initialization parameters. This allows the session participants to agree about the details concerning used protocols, codecs, network addresses and ports. As noted earlier, SIP does not convey the actual media content, but it always works in conjunction with some real-time transport protocol such as RTP.

Both SIP requests and responses may contain message bodies, but this is not a general requirement. There are requests which typically require it, e.g. INVITE, whereas some messages such as BYE must not contain a message body. For responses, the original request and the following response status code determine how the body should be interpreted. The following is a valid example of a complete INVITE request containing an appropriate SDP body:

```
INVITE sip:kaski@mil.fi SIP/2.0
Via: SIP/2.0/UDP examplerouter.mil.fi;branch=z9hG4bKdaxn8
Require: 100rel
To: Juhani <sip:kaski@mil.fi>
From: Gustav <sip:haggis@mil.fi>;tag=3EA4EC56
Call-ID: b42c1d22f23104@esikunta.mil.fi
CSeq: 11 INVITE
Contact: <sip:haggis@esikunta.mil.fi>
Content-Type: application/sdp
Content-Length: 273

v=0
o=gustav 1765868262 8206083726 IN IP4 194.100.112.72
s=A High-Priority Call
i=This call is extremely urgent and confidential
c=IN IP4 194.100.110.71
t=0 0
m=audio 1700 RTP/AVP 0 2
a=rtpmap:0 PCMU/8000
a=rtpmap:2 GSM/8000
m=video 1702 RTP/AVP 16
a=rtpmap:16 H261/90000
```

Table 2.1: SIP requests (RFC 3261/RFC 3265)

INVITE	Establishes a dialog with other participant
ACK	Acknowledging for successful message exchange
BYE	Releases an existing connection
CANCEL	Cancels pending INVITE transactions
OPTIONS	Requests information on service capabilities
REGISTER	Register user's current location
NOTIFY	Indicates a change in the network state (RFC 3265)
SUBSCRIBE	Subscribe as the receiver of NOTIFY-messages (RFC 3265)

The demonstrated SDP fields are commonly present in real-time applications. The reader should be aware that there is an empty line separating the message body from the header part and another one for marking the end of the request. The field semantics is as follows: v is the protocol version, s is a session name, i provides optional session information, c is the contact information of session owner, t is the session expiration time, m describes the type of the media content and a represents the media related attributes.

2.6 Protocol operation

2.6.1 Requests

The Session Initiation Protocol typically uses port 5060 for signaling between SIP servers and endpoints. RFC 3261 describes the most important requests, which play the key role in SIP signaling. RFC 3265 [32] extends the message base by introducing an event-awareness mechanism and two relevant new requests: NOTIFY and SUBSCRIBE. Subscribing happens with the SUBSCRIBE request; the message is used to express intent in receiving certain NOTIFY messages. The most prominent SIP requests are listed in Table 2.1. Afterwards, it was seen as necessary to amend the list with a few more requests (see Table 2.2).

Table 2.2: Additional SIP requests

MESSAGE	Carries SIP instant messages
REFER	Ask recipient to issue a SIP request (call forwarding)
UPDATE	Modifies SIP session without altering current dialogue

2.6.2 Responses

The SIP responses are categorized into two main types: *final* and *provisional*, based on their status code. The final responses typically indicate success, redirection or error condition in transaction. Since they always answer already generated messages, an impinging request is required in order for them to become into existence. Response messages are never generated spontaneously. Final responses provide a means for conveying information about the results of request processing and ultimately terminating the associated transactions. Responses indicating success (2xx) are sometimes called *positive final responses*. Correspondingly, responses signaling an error condition (4xx-6xx) are called *negative final responses*. Redirection responses (3xx) indicate the target has moved outside of its usual location, suggesting the calling UA to try a different set of destinations instead.

Provisional responses are intermediary responses, generated before the definitive final responses, and used for indicating progress of transactions such as request processing at the callee side (100 Trying). For this reason, they are also sometimes dubbed *informational responses*. When the request originator receives an intermediary response, it stops sending the original request immediately and waits for a final response. Provisional responses are sometimes considered "weaker" than final responses, in the sense that they do not require acknowledging (ACK) from the receiving end. There are usually neither strict rules for their processing nor a guarantee in general that they are delivered reliably. For reliable delivery, provisional messages have their own acknowledgement type, *PRACK*, as defined in RFC 3262 [36].

As response codes are intended to be processed by a machine and not very informative from the user view, some response messages may also contain a human-readable *reason phrase*, providing information about the request processing – especially, the reason why a particular transaction failed.

The complete list of SIP responses is too long to be described here. We have left the most important SIP responses to be listed in Appendix B.

2.6.3 Registration

Conventionally, the DNS server is used to locate a local registrar during the registration procedure. This is done via SRV [10] and Naming Authority Pointer (NAPTR) [22] records. Alternatively, a mobile node can multicast REGISTER to enroll itself to the local registrar server or use the Service Location Protocol (SLP) instead, but this is not typical and we are not considering the case where they are used.

When the registrar receives REGISTER request, it extracts the required information from the message and saves it into a location information database. This is called *creating a binding*. Although the registrar can be physically the same entity as the proxy, it is important to make a strict logical distinction between these two types. Figure 2.3 shows an example of a successful user registration with a AAA functionality in use. This enables the user identity to be confirmed before creating a binding.

When registering, a UA may associate itself with several Contact-addresses. These addresses can be prioritized using so called *q-value* in the Contact header field, which allows indicating the relative preference for one contact address compared to other possible addresses.

The registering UA may want to negotiate special conditions such as specific *expiration interval*, that represents a period after which the registration is not valid anymore. It may also ask for revoking registration immediately by issuing a REGISTER request with expiration time as value 0. If no appropriate expiry time has been provided, the value 3600 s will be will be used as default.

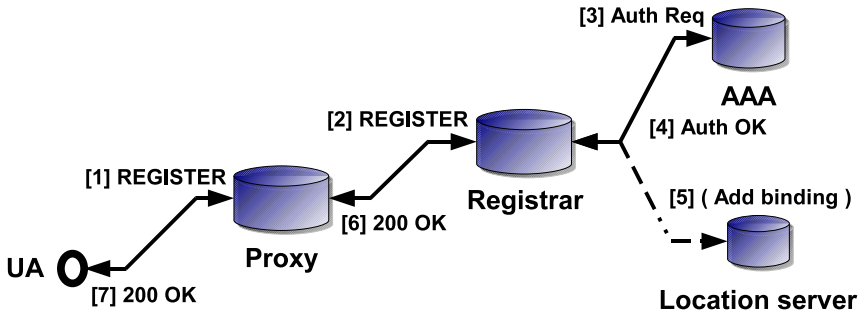


Figure 2.3: Successful registration with AAA

2.6.4 Proxy servers

The Session Initiation Protocol uses proxy servers extensively to locate users and route requests to correct addresses. The proxy servers may be either stateful or stateless. The fundamental difference between these two types is that the stateless proxies are not aware of the state of a connection, they just forward every arriving request and response. Moreover, stateless proxies have considerably higher traffic handling capacity due to the fact that all consecutive transactions can be processed on different computers.

Stateless proxies

Stateless proxies forward incoming messages towards exactly one destination from the target set. As mentioned earlier, this proxy type entirely lacks both transaction and TU layers, communicating directly with the transport layer instead. Since they do not thus have any way to maintain the transaction contexts, stateless proxies cannot distinguish the original messages from retransmitted ones. Neither do they have capability of generating new requests or responses. In a sense, stateless proxies work as message relays in analogous way as switches that operate at the data link layer.

Stateful proxies

Stateful proxies maintain connection state machines, that is, *call contexts*. Maintaining call contexts is required by many enhanced services, and for instance collecting charging information would be completely impossible without the ability of retaining information on session states. Stateful proxies are also capable of *forking* incoming INVITE transaction requests by multi-casting them to several destinations (see Fig. 2.4). This is especially useful feature when the actual location of a callee node is not known beforehand. Finally, only stateful proxies have the ability to multicast and handle TCP connections. Processing of SIP transactions is computationally more expensive for stateful servers than it is for stateless proxies.

Using 'q-values' mentioned in 2.6.3 as parameters enables *sequential forking*, where contacting a callee happens in such manner that the most preferred contact address will be tried first. If no answer occurs in a given time, the secondary address will be contacted. The contacts are processed from the highest q-value to lowest. If there exists several Contact-fields with a same q-value, both may be contacted in normal, parallel way.

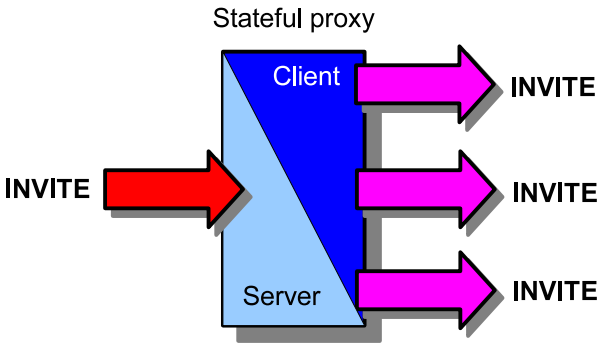


Figure 2.4: Stateful proxy multicasting INVITE requests

2.6.5 User identification

The SIP addressing follows generic URI syntax defined in RFC 2396 [3]. The protocol also provides support for E.164 numbers and H.323 addressing. Typically, however, each user is identified unambiguously through addressing *sip:user@domain* which notably resembles e-mail addresses. SIP parameters may additionally carry varying amount of extra information about URI components. Angle brackets (<>) are needed if question marks, semicolons or commas have been used on a header line.

One may alternatively use the *sips:* scheme, which allows secure session establishment securely using the Transaction Layer Security (TLS) between a calling UA and the called resource, albeit it cannot be guaranteed that TLS is in use end-to-end. This scheme enables merely secure signaling, as the actual media stream may still be completely insecure.

2.6.6 Session establishment

In the most trivial case, the INVITE sent by the caller is answered with a provisional 100 Trying response back to inform the caller that it may stop broadcasting the invitation. When the call is accepted with 200 OK, this must be yet acknowledged with ACK by the caller. The final response is continued being re-transmitted until the ACK is received. High signaling reliability is achieved this way. See Figure 2.5.

If the called UAS is unable to take a new call, it will generate a response 486 (Busy Here). If the system is in a busy state everywhere, which is rarely the case, it may generate a response 600 (Busy Everywhere) instead. If the incoming call is rejected for policy reasons, 488 (Not Acceptable Here) will

be returned provided with a warning message which indicates the reason for dropping the call.

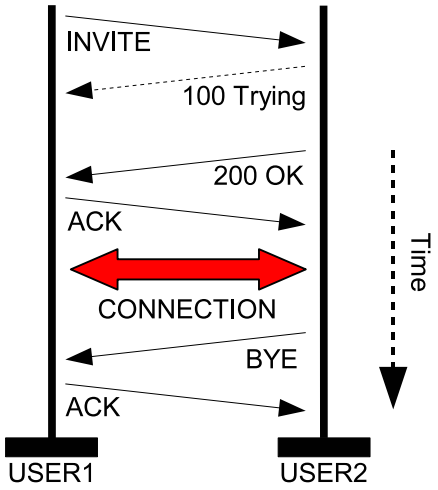


Figure 2.5: Session establishment between two peers

Figure 2.6 presents another simple case where a user INVITE request is sent using two stateful proxy servers, when the callee has moved outside of his usual location and the request is redirected towards the callee, using 302 Moved Temporarily response. After receiving a 200 OK message, a peer-to-peer connection is established between the participants. For conciseness, the ACK messages which would immediately follow are omitted from the illustration. It is also easy to recognize the commonly seen shape of *trapezoid routing* characterizing SIP signaling networks, implying that the actual media stream usually takes a different path than signaling.

2.7 NAT traversal and firewalls

Protecting the network becomes extremely important if it is connected to the public Internet with an appointed public border router or a gateway. If the network is completely segregated from the public network, concerning about network traversal is not necessary.

Firewalls and Network Address Translators (NAT) are nowadays commonly seen functions in network border routers. The principal reason for using the latter is the limited IPv4 address space. NATs provide a reasonable method of saving and re-using the valuable public addresses, but properly written

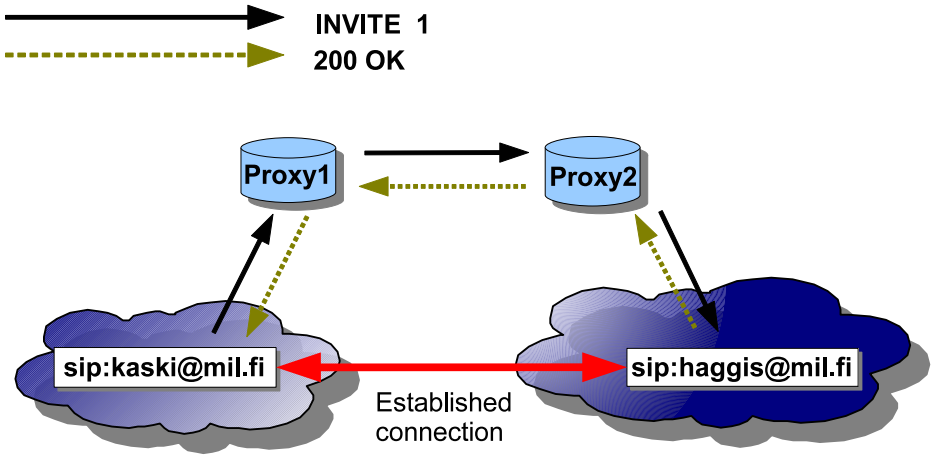


Figure 2.6: Connection establishment using stateful proxies and redirection

implementations also provide indirect security by rendering the private subnets behind them invisible to outsiders. These factors explain why they are extensively favored particularly in larger organizations. The security comes with a price, however: while NATs protect users behind them from attacks performed by malicious outsiders, they inevitably prevent accesses initiated from outside. SIP does also suffer from restrictions caused by firewalls and NATs, but there are various methods for traversal available. Currently there are three generally suggested ways of traversing.

Application Layer Gateway

Application Layer Gateways (ALGs) are components that augment the functionality of NATs or firewalls, supervising the traffic flow on the application level. ALGs are able to scan the traffic flow between public and private networks, allowing legitimate data streams to pass. The data which would be otherwise rejected by NAT is able to pass by using ephemeral TCP/UDP ports. ALGs furthermore convert network level addresses between formats which are addressable on either side of NAT/firewall, enabling also certain application level commands and providing data stream synchronization.

ALG permits SIP traffic through *deep-packet inspection*, which requires that the border gateway component has support for SIP traffic. It modifies the incoming packet appropriately before letting it pass and keeps the current address bindings valid until the current SIP session terminates.

Session Border Controllers

Session Border Controllers (SBC) are network entities residing on call paths. They are commonly seen elements in VoIP networks. They behave like user entities, exerting control over signaling and payload streams. SBCs are capable of forcing the signaling and/or data traffic through them, and able to modify the data crossing them at the same time. SBCs are also able to perform the tasks usually belonging to ALGs.

STUN

Simple Traversal using User Datagram Protocol Through Network Address Translators (STUN) [37] technology enables entities behind NATs to solicite their public network addresses.
STUN is able to traverse most types of address translation with exception of symmetrical NAT, which actually is the type most commonly found in large corporate networks. Additionally, STUN fails to address the need for TCP traffic. Given these observations, STUN cannot be generally considered as a complete method of traversal.

TURN

Traversal Using Relay NAT (TURN) [35] was developed from the need for address the limitations of STUN. TURN allows for both TCP and UDP based traffic, also addressing the problem of symmetrical NATs. TURN has practical value as a last resort solution only. The mechanism currently holds an Internet Draft status.

ICE

Interactive Connection Establishment (ICE) [34] provides a dynamic mechanism for discovering the optimal means of connectivity for media between network endpoints. It relies heavily on STUN/TURN in its working. ICE is suitable even in challenging network conditions, and it is known to work through almost all firewall/NAT types. As of the time of writing, ICE holds the status of an Internet Draft, but is likely to become an Internet Standard.

NAT traversal for SIP protocol and its problematics has been clarified in the Appendix C.

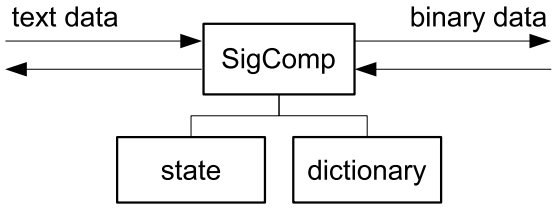


Figure 2.7: SigComp functioning principle

2.8 SIP bandwidth usage

The text-based format makes SIP somewhat inefficient with regard to the bandwidth, delay and processing power. The length of a typical SIP message can range anywhere from a few hundreds of bytes up to several kilobytes. This was not initially seen as a concern, since SIP was never considered to be a protocol of choice in narrow-band wireless environments. In the desktop world, bandwidth has never been an issue.

2.8.1 Signaling Compression

Signaling compression (SigComp) [31, 9, 5] is an attempt to optimize text-based traffic in terms of size by introducing an extra layer between the local application and the transport layer. Although the solution has been targeted specifically for compressing SIP and RTSP traffic, SigComp can be used with any application-level protocol. Thus, the SigComp specifications do not go too deeply into application dependent details. Open implementation libraries for SigComp provided with source codes are publicly available.

The functioning has been shown in the Figure 2.7. SigComp consists of a compressor, a state handler, and so called Universal Decompression Virtual Machine (UDVM) to run decompression algorithms for incoming messages. The state handler is a storage for the data needed for processing of future messages. The compressor encodes incoming text and maintains the state keeping it updated for later messages. The dictionary is a SIP/SDP specific SigComp part for achieving higher compression efficiency.

SigComp can improve the protocol performance especially in narrow bandwidth links, but the approach has some serious drawbacks. Introducing an extra layer always increases complexity and sets new requirements for the hardware in use. Hence, we shall not delve into the signaling compression more technically during the span of the thesis.

2.9 Chapter summary

This chapter explained the functioning of the Session Initiation Protocol, using the terms and definitions as they are presented in IETF RFC 3261 and RFC 3265.

The SIP protocol has proven out to be a flexible signaling mechanism, already supporting the most major features of typical signaling systems such as SS7. Though originally intended as a subscriber signaling protocol, SIP has also appeared for being a viable technology for various network core signaling purposes. The protocol stability and scalability make it suitable for dynamic network environments, where the intelligence is located at the network edges.

The protocol is not, however, particularly bandwidth conservative due to its uncompressed nature. Mechanisms such as SigComp have been issued to compress SIP based messages with varying success. In commercial use, perhaps the biggest issues are related to Network Address Translation (NAT) and traversing firewalls.

3 SIP mobility

This chapter discusses some of the essential factors which influence the hand-off performance, and the required amendments needed for bringing mobility into military grade mission-critical SIP networks. The key issue is how mobility can be effectively and efficiently managed on the application level.

3.1 Mobility definitions

The most generally adopted definition for *macro mobility* is the capability for movement between different networks, which may or not be located in different Administrative Domains (AD). In most civilian settings inter-domain mobility would often require Service Level Agreements (SLA) and roaming agreements between network operators, but in this work we may relatively safely assume that these subnets are under a single administration. *Micro mobility*, which deals with inter-subnet movement confined within a particular domain is generally out of the scope of this thesis. A short overview on different mobility types is given next.

Terminal mobility describes device mobility widely in the "conventional" telecommunications sense. The network terminal can access services while on move, and the network is able to recognize and locate the terminal. During the process, the user identity may change. If it is also required that after the connection hand-off the session stays in established state, that is, without interrupting the current session, the term *seamless mobility* is often used. In a strict sense, seamless mobility means that the hand-off causes no perceivable degradation in the experienced QoS [12].

Personal mobility is the type of mobility, where the user identity remains the same regardless of the point-of-attachment or device used to connect the network. User location changes are completely transparent to other communication parties. The user may alternate between terminal devices, while still keeping himself or herself reachable for other network users. This is often referred to as *user mobility*.

Service mobility means, that the service user does not have to change the operator or accessing device after moving into another subnet, he is able to use exactly the same services in similar way as before the transition. He may, however, do this upon his will: the service profile is not bound to one operator or device. The accessing device may be for instance a phone,

a PDA or a laptop. The customer preferences and service customizations remain unaltered.

Session mobility refers to the user's ability to maintain an active session while switching from a terminal to another. SIP implicitly supports this mobility type, through re-sending INVITE requests during the ongoing session. Also session parameters can be altered this way.

The aforementioned mobility types together are sometimes referred to as the *universal mobility* [30]. SIP can provide universal mobility when needed, making the limits between these mobility types vague: SIP mid-call mobility is something between the session and terminal mobility. SIP guarantees user mobility by allowing the user to be reached anywhere via the same logical address (SIP URI), using any terminal capable of running an IP stack. The media independent signaling strictly segregated from the interpretation of actual data streams enables the service mobility, turning the complexity needed for modifying a service to support different terminal device platforms into a data presentation problem at the terminal end. The term *"Mobile SIP"* usually refers to the needed protocol extensions to make SIP support terminal mobility. Terminal mobility is at the core of the focus, when seamless hand-offs become a question.

3.2 Supporting terminal mobility for SIP

The application-level terminal hand-off procedure functions in two phases. First, the mobile node registers in the new location. Second, the on-going session is redirected to the new location. The application running on the mobile device must be capable of noticing changes in network level addresses. When the mobile terminal discovers that it is soon to cross an edge of the current cell and the signal received from the target base station indicates that it is moving to another network domain, a new IP address will be acquired and the terminal mobility procedure initiated.

In order to support terminal mobility for SIP, a few additional extensions are required. The terminal mobility mechanism impacts at three distinct stages: pre-call mobility, mid-call mobility and network partition. [45, 46]

3.2.1 Pre-call mobility

The situation where the point of attachment changes prior to establishing a connection is called *pre-call mobility*. The Figure 2.6 illustrated a situation where the mobile node has temporarily transited outside of its home network. In this case, it must fetch a new IP address from the visited network in order to be able to establish any further connections, and additionally refresh its current location information at its home registrars database by re-registering. Keeping homeward location server updated allows incoming connections to be hereafter redirected to the correct new location.

3.2.2 SIP terminal hand-off (mid-call mobility)

When the mobile node transits into another IP domain while connection is established and data is flowing, the CN must be notified. This is made through a re-invitation procedure. New session parameters are negotiated, including a new IP address. Unfortunately, this signaling exchange produces a long delay both due to location updating and signaling overhead. In this process, DHCP seems to be a remarkable source of delay, potentially raising the IP address re-allocation time up to several seconds greatly depending on the used architecture.

3.2.3 Network partitions

In some occasions, a network may divide into multiple networks that are working seemingly correctly, but unable to communicate with each other. Should a network partition occur, SIP has an in-built recovery mechanism to restore the network into the functional state. If the partition lasts less than 30 s, restoring into the original state does not require any special measures: in this case SIP re-transmits the request as there is no answer. Longer lasting partitions are solved through re-INVITE procedure in such way that each side refreshes the session with the home proxy of other communicating side addressing its canonical address. UAs may implement an automatical session timer functionality, which periodically refresh the session at user configurable intervals. We must note that the problematic sub-case, what happens if the network split occurs and the user agents remain in a network part which does not contain location servers and thus address translation whatsoever, is important and interesting but generally out of topic.

3.3 SIP terminal hand-off performance

The terminal hand-off delay is coarsely composed of three different factors: network discovery, acquiring a new IP address, and restoring the connection with the CN by an INVITE request. This does not yet involve the security and AAA, which most probably introduce some extra delay depending on the used architecture. A closer look on the delay components that take place during network level hand-offs is given next.

3.3.1 Hand-off delay components

There are several delay sources contributing to the total hand-off delay. The major part of this delay originates from the link and network layers. The Figure 3.1 describes signaling taking place during the SIP mobile hand-off process. The presentation is simplified; in particular, signaling with the home registrar has been dropped for clarity. The hand-off sub-procedures are explained below. [18]

Link-layer delay (D_0) The L2 delay consists of scanning, authentication and reassociation of the terminal with the target Access Point (AP). The scanning phase comprises around 90% of this delay, and it is highly dependent on the used architecture [12]. Compared to the signaling delay, the link-layer establishment latency is usually considered to be negligible.

Movement detection delay (D_1) After concluding the link-layer hand-off, a MN needs to discover it has moved into another network domain. This can be done using Router Solicitation or examining Router Advertisement messages periodically sent by routers, or querying the network prefix using some external protocol when the access point MAC address is known.

Address acquisition delay (D_2) When using a DHCP server for address allocation, the whole process may typically take more than a second. This is majorly due to the Duplicate Address Detection (DAD) phase, which uses ARP to solicite potentially colliding addresses in a subnet. In IPv6 networks, this component consists of DHCPv6 delay, if stateless address autoconfiguration of hosts is not used.

Re-configuration delay (D_3) The delay incurring from re-configuring the MN network interfaces and setting network parameters to re-establish the connectivity. This time varies considerably from device platform to another.

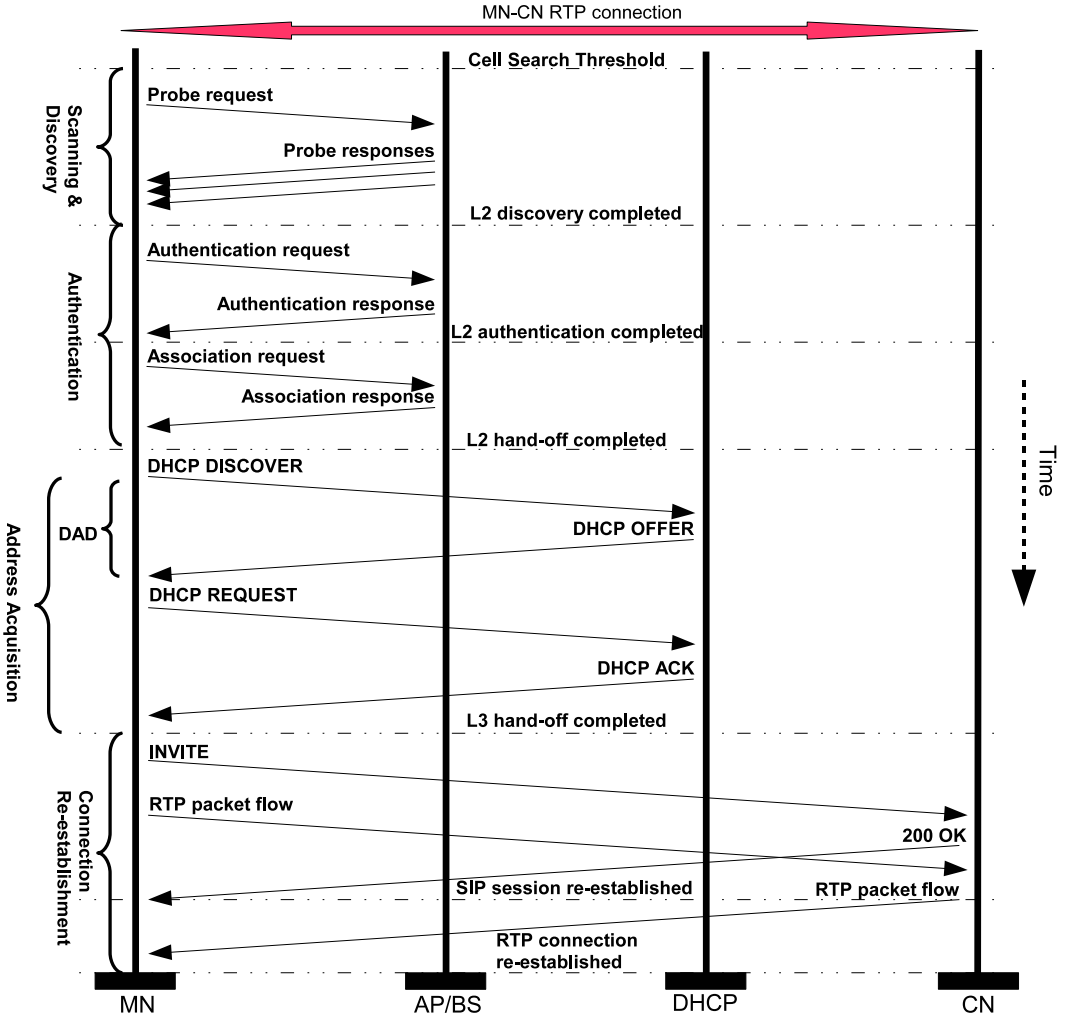


Figure 3.1: SIP hand-off signaling sequence diagram

SIP re-establishment delay (D_4) This source of delay comprises the RTT required for the re-INVITE transaction between participants, plus the message processing time at both UAs and any proxies between. This time can appear for up to 100 ms depending much on the logical distance between the communicating nodes.

RTP packet transmission delay (D_5) The time required for the first media packet to be successfully exchanged between MN and CN over the restored connection.

QoS and AAA (optionally) The provision of QoS and AAA introduces a delay of its own when implemented. The incurred latency varies highly depending on the used mechanisms.

The total hand-off delay D_{HO} can be written in a rigor mathematical format using these sub-procedures as present in the Formula 3.1:

$$D_{HO} = \sum_{N} D_N = D_{L2} + D_{MD} + D_{DHCP} + D_{RECONF} + D_{SIP} + D_{RTP} \quad (3.1)$$

The DHCP address acquisition (D2) and the SIP re-establishment (D4) have been recognized as the major delay sources. The D2 component seems to exist mostly due to the Duplicate Address Detection (DAD) procedure. Removing the duplicate detection mechanism could cause the D_{HO} delay to drop to approximately 100 ms [15].

3.4 Mobility mechanisms compared

The basic SIP protocol performs poorly when frequent or seamless hand-offs are required. SIP suffers from hand-off latencies severely hampering the performance during cell transitions, and causing it to be unsuitable for delay sensitive mobile applications. The latency caused by the conventional SIP mid-call mobility mechanism, cited to be as high as 1.5 s [18], is too high for the purpose of real-time media hand-offs. Most real-time voice applications typically tolerate delays of 50..200 ms without too high degradation in the experienced QoS. For seamless VoIP applications, latencies $<$100 ms are generally considered as sufficient [39]. Another major drawback in SIP is the absence of mobility management for long-term (persistent) connection TCP, leading to a communication breakdown when the network address changes. [8]

Several mechanisms have been drafted in attempt to enable hand-offs with sub-second latencies. Some important ones are summarized in the following.

3.4.1 Mobile IP

Mobile IP (MIP) is the earliest existing network-layer mobility standard, enabling transparent mobility in IPv4/v6 networks. There is a plenty of material published on MIP, so merely the protocol disadvantages are precised here instead of repetition. As explained in Chapter 1, the primary reason for its inefficiency is the resource consuming tunneling mechanism used for delivering incoming packets to a MN residing outside of the home network. Other problems include the requirement for a permanent home address and

the triangular routing leading to asymmetric traffic flows.

Route optimization can be used to partially address the triangular routing problem: the basic idea is to inform the CN about changes in the current IP address. There are several drawbacks here involved. One is the need to modify the IP stack at the CN to make it able to handle IP encapsulation and store care-of addresses relevant to the MN. This problem concerns IPv4 stacks merely, since IPv6 supports route optimization natively. Finally, MIP cannot support seamless mobile multimedia hand-offs due to the incurred high latencies for hand-off signaling.

Mobile IP is a sufficiently good choice for persistent TCP-based data traffic, when the transfer reliability outweighs the high hand-off latencies involved. For real-time traffic, however, the below par characteristics imply a need for a better faring mechanism.

3.4.2 Hierarchical Mobile SIP

Hierarchical Mobile SIP (HMSIP) [8] is a proposal to reduce the signaling overhead sustained during hand-offs while providing a proper micro-mobility (intra-domain) support for SIP. HMSIP enhances the conventional SIP by introducing a new entity, SIP Mobility Agent (SIP MA), to manage mobility. The SIP MA is essentially a domain border access point enhanced with SIP proxy and SIP registrar functionalities.

Each MN inside the domain gets two IP addresses, a local address (LA) and a global domain address (DA). The DA, identifying each MN unambiguously within the domain, is provided by the SIP MA. The LA reflects the MNs current point of attachment, and is provided by the serving access point. The SIP MA maintains a database of mappings between SIP URI, LA and DA for each terminal roaming inside the domain.

When arriving to the network, the MN first gets a new LA from the serving access point and a new DA from the SIP MA. The MN registers its location by sending a REGISTER destined for SIP MA, which associates its LA with the corresponding SIP URI. Subsequently, the MN DA is registered to the MN SIP home registrar, to make the mobile node reachable from outside.

When moving across subnets (intra-domain), the node needs to obtain just a new LA during each hand-off, while DA remains the same. This is called *regional registration*. This leads to optimized signaling efficiency, as

home registration (DA altering) is not required. If the MN moves from the current domain to another, updating both addresses, LA and DA, is needed. Incoming traffic will be first addressed to the MN using its DA address maintained by SIP MA. The SIP MA forwards the data to the appropriate receiver using its LA.

The primary value of this mechanism is in eliminating the time spent on home registration procedure. From the viewpoint of seamless services, however, several difficulties arise. Though the hand-off latency is reduced, it still remains too high for many real-time applications. On the other hand, HMSIP utilizes the IP address space inefficiently due to the need for two addresses per mobile node. Furthermore, HMSIP is designed to handle merely the intra-domain mobility, requiring external supporting protocols for macro-mobility. In IPv6 networks this would not be a problem, but in IPv4 address space the amount of free addresses is already an issue and problems might occur. Deployment may be also problematic due to the need for an extra component, SIP MA.

An improved mechanism, Fast HMSIP [8], based on the simple integration of the HMSIP mechanism and proactive address reservation has been proposed to address these shortcomings and to provide lower latencies, but the same fundamental weaknesses of the mechanism still remain: the inefficiency in terms of address space utilization and needed extra components. Given these facts, it can be concluded that the HMSIP mobility solution is not capable of addressing the need for seamless SIP macro-mobility properly.

3.4.3 Hybrid and integrated MIP-SIP schemes

Along the years, several "hybrid" or "integrated" mechanisms attempting to leverage the advantages and best practices of both MIP and SIP protocols have become into existence (e.g. see [20, 44]). The presented schemes vary in details and have had tendency to be extremely complicated, requiring some modifications on the used hardware and protocols.

One proposal, the hybrid MIP-SIP architecture [20], provides macro mobility by delegating the mobility management to application and network layers. The general idea is to use MIP for intra-domain signaling, i.e. between the visited network and the mobile node, while SIP is used in communication between the visited network, the home network and the corresponding node. The added value is in avoiding the arduous encapsulation process typical to the ordinary Mobile IPv4/v6. The mechanism introduces a new network

entity called Mobility Manager (MM), which translates signaling between MIP and SIP. It functions as a home agent and a UAC for a visiting mobile node, and so called Mobility Anchor Point (MAP) facilitating intra-domain hand-offs. The proposed mechanism has several difficulties. Firstly, new network components and software modifications are required. Secondly, the architecture introduces extra delay incurring during MIP-SIP translation at the mobility manager. The MM might thus appear for being a bottleneck in the system.

Because of their comparatively recent emergence, not enough experience on their performance exists to be found. Nevertheless, these techniques are very interesting and deserve to get properly analyzed in the future. Comparing to SIP, MIP has better bandwidth utilization, implying much lower signaling loads. It is, therefore, generally a good idea to adopt SIP for real-time audio and video, and to use MIP to support traffic with looser time requirements such as TCP data transmissions.

3.4.4 Predictive Address Reservation with SIP

Predictive Address Reservation (PAR) [18] is a promising mechanism for mitigating harmful high-latency effects incurring during hand-off process by performing the address re-allocation and the session updating proactively with aid of the link layer information. The underlying idea is, that the mobile node anticipating an imminent hand-off commences the necessary DHCP address reservation and re-INVITE operations before the actual link-layer hand-off procedure initiates.

The MN starts scanning for a new BS after the Signal-To-Noise Ratio (SNR) of the current base station, SBS, has fallen below the Cell Search Threshold. When the MN expects that the time for hand-off is getting near, it selects a suitable target BS from its internal database and sends a reservation request to its SBS. The SBS then consults its neighbor BS information table to see whether the MAC address of this predictive BS (TBS) belongs into the same network domain. If the predictive BS is confirmed to belong to the same domain, the SBS sends it a link layer hand-off (L2HO) request. Otherwise, a network level hand-off is needed; a new IP address will be obtained from the TBS via DHCP, and then forwarded to the MN using a reservation reply. This response contains procedure acknowledgements and the new reserved address which the MN can use when the hand-off is finished.

For instance, see Table 3.1: BS_0 governs two access points, belonging into a

Table 3.1: Example neighbor BS information table

BS_n	Access Point MAC	Domain prefix
BS_0	00-E0-A8-E2-6F-D5 00-C5-64-A0-B4-EA	194.100.112.0/24
BS_1	00-BD-A2-B6-24-F2	130.233.0.0/16
BS_2	00-CA-5B-16-78-AA	

network domain of its own. The base stations BS_1 and BS_2 share one domain, however being two separate base stations with their own disjoint access points. Should a hand-off from the BS_0 either to BS_1/BS_2 (or conversely) occur, a network level hand-off is necessary.

Subsequently, the MN sends a re-INVITE message to its CN, using the fresh IP address it just got. The CN answers this message with a 200 OK response, and opens a new session in parallel to the old session. The packet exchange now happens through both sessions until the hand-off procedure is completed. This *bi-casting* is for minimizing the chance of packet loss during the hand-off procedure. After the hand-off is completed, the old session will be torn down.

When deriving the formula for the PAR-SIP hand-off latency, the earlier presented Formula 3.1 reduces to:

$$D_{PAR-HO} = D_{L2} + D_{MD} + D_{RECONF} + D_{RTP} \tag{3.2}$$

Hence, as it can be seen from the Formula 3.2, the PAR mechanism allows the most arduous phases of hand-off in terms of time, namely D_{DCHP} and D_{SIP}, to be eliminated completely. In early experiments, approximate delays of $D_{PAR-HO} \approx 60$ ms have been achieved on a testbed when using PAR-SIP mechanism [18]. This should be good enough for most real-time applications.

The Figure 3.2 illustrates a full PAR-SIP hand-off procedure for a single mobile node. When the SBS receives a reservation request sent by the MN, it sends an address allocation request (HO_L3ADDRESS_REQ) to the target base station TBS, which in turn transmits back an acknowledgment carrying the information about allocated addresses. This information will be then forwarded to the MN using the reservation reply.

Subsequently, the MN starts preparing for the actual hand-off proactively by registering itself to the target domain, while simultaneously sending a

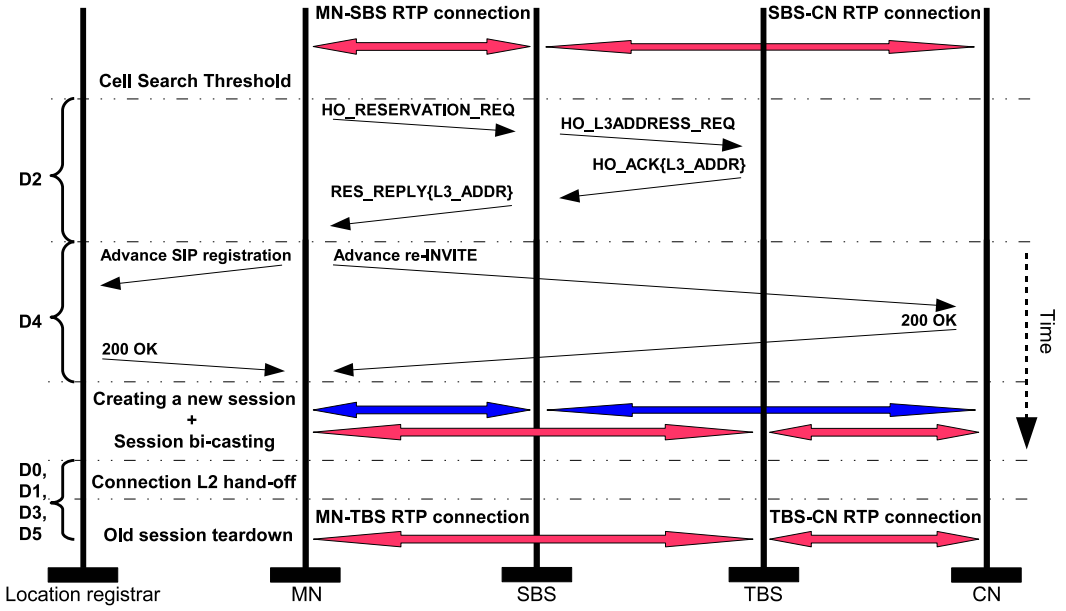


Figure 3.2: PAR-SIP hand-off procedure

re-INVITE request to the communicating partner CN. The CN reacts by opening a new session and informing the MN through 200 OK response; all future packets will now be bi-cast using both data pipes, until the MN indicates that the L2 hand-off procedure is finished, and the old session can be safely closed.

Deploying network-wide support for PAR-SIP entails substantial intervention from behalf of network operator. Another crucial issue remains, how bi-casting works if the corresponding node is also a mobile node. A viable solution is to perform the bi-casting at the base station. This approach would conserve precious bandwidth at the CN side. In any case, PAR-SIP requires significant modifications in SDP extensions to function.

3.4.5 Cross Layer Fast Hand-off for SIP

The Cross Layer Fast Hand-off for SIP (CF-SIP) [7] is a recent proposal for deploying seamless services in existing mobile SIP networks with minimal modifications. CF-SIP employs the same general ideas as PAR-SIP: creating a new session in advance using link layer information and then bi-casting. The essential difference between CF-SIP and PAR-SIP is using the Bootstrap Protocol (BOOTP) before obtaining a new IP address. The advantage of CF-SIP is that it does not require additional components to existing network

infrastructure as PAR-SIP does.

After receiving a base station beacon frame, the MN extracts the BS MAC address from the received frame. The MAC address is then used by the BOOTP protocol to solicite the corresponding network IP address prefix to see whether the frame originated from the same network domain or not. If the obtained prefix differs from the prefix of the current network, the MN sends a DHCP request to the DHCP server that sent the BOOTP reply. If the received prefix equals the current network prefix, no DHCP request is needed since network stays the same. After a new IP address has been obtained from the target network DHCP, the end of the procedure is similar to that of PAR-SIP. Before the actual link-layer hand-off is initiated, the MN registers its new address both its home registrar, meaning that the home registrar maintains two addresses for a single MN temporarily during the hand-off: the old address can be removed as unnecessary once the hand-off is over.

PAR-SIP and CF-SIP are fundamentally similar technologies, exploiting the link layer information, proactive address allocation and session restoration. These mechanisms do not differ much in terms of performance. Henceforth, we adopt PAR as a general term for referring to any of the aforementioned technologies. Predictive address reservation is potentially an enabling key technology for seamless SIP hand-offs.

3.5 Session bi-casting and SDP extensions

Bi-casting is an effective method for improving the packet loss ratio during hand-offs, but it carries the doubled bandwidth usage at the correspondent node side as a cost. The limited network resources may raise as a concern, and it is not typically desirable that a mobile node carries out the bi-casting. Thus, an external network element needs to be introduced.

A possible solution is to use a *Hand-off Assistive Server* (HOAS) network element, a separate bi-casting control point located between the user agents [13]. Upon a request, the HOAS element splits incoming data streams to several locations, acting effectively as a traffic multiplexing point. HOAS may also provide transcoding service functionality, allowing media streams to have different service qualities. This way, the assistive server could function as a traffic quality adaptor, providing the same data in different service classes to different MN point of attachments. In practice, the HOAS could be manifested by a stateful SIP proxy. For simplicity, it can be assumed

that the HOAS is located at the CNs outbound proxy.

For realizing the session bi-casting required during PAR-SIP and CF-SIP hand-offs, certain SDP extensions are needed. The HOAS must maintain state information for each active session during a hand-off. Therefore it needs to place itself in the signaling path, using *"Record-Route:"* header. A new session-level attribute, *"a=bicast"*, is used to indicate the need for bi-casting [13]. The HOAS recognizes the need for bi-casting by the presence of this header field. Modified UPDATE requests may be used to trigger the PAR hand-off.

Upon receiving a request containing a bicast attribute, the HOAS notifies the CN by an UPDATE request. This is needed to add the HOAS to the media path. Upon completion of the hand-off, the HOAS must be yet removed from the signaling path using another UPDATE, respectively.

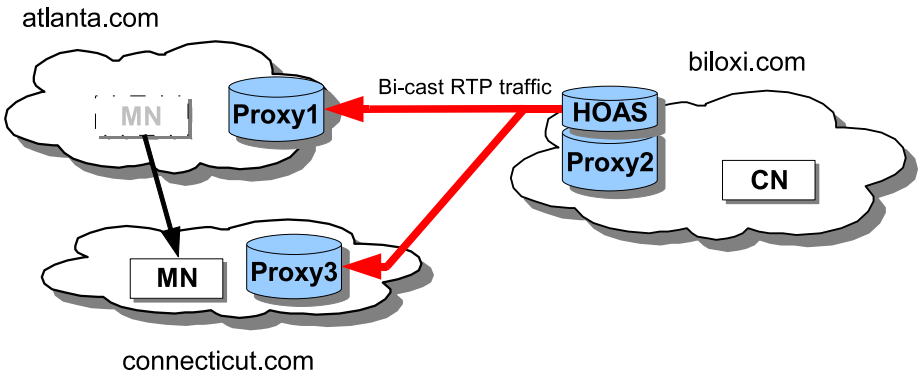


Figure 3.3: Bi-casting using Hand-off Assistive Server

3.6 Movement prediction errors

In some rare cases the movement prediction may prove out to be erroneous. For instance, when a MN resides in a region of three overlapping networks, the prediction mechanism might initially try to reserve resources from an incorrect domain (see Figure 3.4). A recovery method is needed to correct the situation. The erroneous base station or DHCP is first released and then a new hand-off initiated. Since the movement prediction and resource reservation are done well before the actual link-layer hand-off, the cost of such errors is typically increased signaling traffic. The details of the used recovery measures may vary.

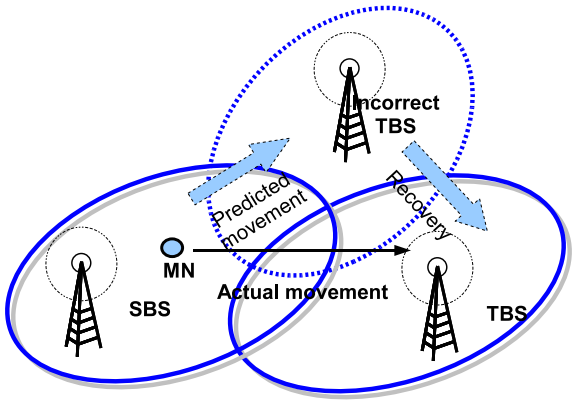


Figure 3.4: Prediction error during hand-off

3.7 Hand-off security and authentication

Besides various technical failures, networks are prone to attacks for theft, service disruption or other malicious purposes. Security considerations are of paramount importance particularly for military networks, where critical functions are in a constant risk of getting damaged or disturbed by a malevolent party. Network security is an extensive field of study, requiring several books to be dealt with comprehensively.

If compromised, sensitive information concerning group structures, identities, protocols and capabilities may be inferred by listening to signaling traffic. SIP message headers can reveal critical information about communicating parties and communication patterns. Message bodies often carry user data that should be kept secret. The interceptor may also be able to drive the communication into a malfunctional state by altering requests in transit or forge faked requests. The signaling traffic should be therefore soundly secured.

Though the diverse nature of SIP architecture makes sessions non-trivial to secure, strong data security capabilities can be provided using authentication and encryption, either basing on the challenge-response or public/private key cryptography. All security mechanisms available for HTTP can be utilized by SIP and are guaranteed to ensure data integrity and confidentiality [42]. The choice of security algorithms may have a significant impact on the hand-off performance.

Sometimes it becomes more important to corroborate the message sender's identity by message authentication. A simple way to ensure the data integrity

and authenticity for SIP message contents is by using message authentication codes (MAC).

3.7.1 HMAC

Hash Message Authentication Codes (HMAC) are based on cryptographic hash functions and secret keys.

Applying the following notations,

$k \triangleq$ a secret key
$M \triangleq$ a message
$f_H \triangleq$ a hash function (e.g. SHA-1, MD5)
$S_B \triangleq$ block length in bytes
$ipad \triangleq$ the byte 0x36 (hex) repeated S_B times
$opad \triangleq$ the byte 0x5C (hex) repeated S_B times
$len_k \triangleq$ the length of the secret key k
$k_+ \triangleq$ k || the byte 0x00 repeated $(S_B - len_k)$ times

The HMAC operation is defined as follows:

$$H_k(M) = f_H[(k_+ \oplus opad)||f_H((k_+ \oplus ipad)||M)] \tag{3.3}$$

In Formula 3.3, '||' denotes concatenation and '$\oplus$' the exclusive OR operator. Using HMAC-SHA-1, the output size produced by the formula is 20 bytes. The calculated output, a *tag*, may be attached to a request to indicate that the message is authentic and its contents have not been changed. The line containing an authenticity tag could look as follows:

```
Req-Auth: 83a79ec1fdab431b4fed4972fd11c68c52596b47
```

Upon receiving a SIP message containing this line in its header, the receiver removes the line and calculates a hash value for the remaining part of the message. Possible attempts to tamper with the request cause the tag value not to match and the message is rejected as potentially unsafe.

3.8 Chapter summary

This chapter covered the mobility for individual SIP mobile nodes, and the extensions required for SIP to support seamless hand-offs efficiently in infrastructured networks. We had a concise overview on what is needed to enable seamless terminal mobility for SIP. The protocol supports user mobility natively and can be extended to support both terminal and session mobility. The mid-call mobility appears for being the most problematic case due to high hand-off latencies involved.

While suffering from significant delays, the SIP hand-offs can be significantly improved to enable latencies below 100 ms, which is sufficient for most real-time applications. The primary causes for the excess delay during SIP hand-offs are the IP address allocation and session re-establishment procedures. After comparing some relevant mobility schemes, we found two mechanisms having potential to enable hand-offs with the required sub-100 ms delays. Predictive Address Reservation (PAR) is a promising mechanism for keeping the inter-domain hand-off signaling in access networks minimal, assuming that the domain size is kept large enough. PAR-SIP requires modifications to all base stations in the network as the group management and the address reservation is happening in them. Another mechanism, CF-SIP, tries to address this problem while retaining the beneficial features of PAR-SIP. Essentially these mechanisms are very similar, both aiming at mitigating the IP address allocation and SIP session re-establishment delays which are the major delay contributors.

When referring to PAR we mean any mechanism sharing the same functioning principle as the aforementioned technologies, that is, address allocation and session re-establishment before the actual link layer hand-off. In the next chapter, we extend the mobility concept to a wider context by investigating mobility for groups.

4 Group mobility

This chapter introduces the reader to group mobility, a concept often seen as a more realistic, though more challenging way to model group behavior in mobile networks than conventional mobility schemes, which put emphasis on the movement of individual network nodes instead. The benefits of group hand-offs are also explained, and a short overview given how the groups can be logically managed by means of hierarchical routing.

4.1 Group mobility models

Mobility models are extensively used to analyze the system and protocol performance in newly designed networks. In many occasions, it is more fruitful to observe the group as a whole instead of deeming the group members as individual, separate entities. Whereas conventional models (e.g. Random Walk, Gauss-Markov) concentrate on the movement of singular independent entities, group mobility models attempt to capture the motion and interaction for groups characterized by close collaboration and strong interdependence between the group participants.

Group mobility models were originally developed to capture group behavior in *ad hoc* environments, although these theories are applicable in fixed networks as well. Group mobility models can be used to predict the future need of resources, when group mobility patterns are known. Despite a minor semantic distinction between the definitions of "mobility pattern" and "mobility model" – mobility models are mathematical constructions derived from real world mobility patterns – these concepts are commonly used as interchangeably and tantamount.

A group and its forming members are treated as separate concepts. Group motion occurs relative to the group conceptual center, a *reference point*. All mobility characteristics of the group can be thoroughly parametrized through vectors associated with the group reference point (such as velocity and acceleration in some particular direction), defining the movement state of the group. A group is considered to be moving as the conceptual center moves. The conceptual center may be chosen to be the same as the group leader, but this assumption is not strict. Each group member may have an individual mobility pattern and a reference point of its own. There are several ways to model this movement.

Group mobility models tend to perform decently in task-oriented scenarios, where well-structured groups strive to achieve a well-defined goal, provided that each participating node has a fairly static group membership that does not change during the operation. Since mobility models are dependent on the application, there is no single comprehensive scheme.

4.1.1 Column Mobility

The Column Mobility model attempts to capture the group behavior during searching or scanning activity, where the searchers are proceeding forward in a row pattern or a queue form. A group of MNs are thought to be associated with a given line of reference. The motion of this line (and thus the group) is represented through the *advancement vector* $\overline{GM}$.

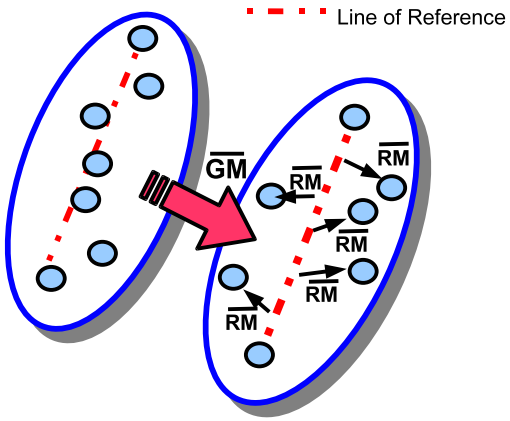


Figure 4.1: Column Group model

The group members have individual reference points upon the line, allowing them to wander randomly around the appointed reference points RP_i. The random movement is represented through a vector $\overline{RM_i}$. While the group advances, also the individual reference points move relative to the line of reference. See Figure 4.1.

For each time tick τ, the movement equation for each node RP_i can be simply written as:

$$\overline{RP_i}(\tau + 1) = \overline{RP_i}(\tau) + \overline{GM} \tag{4.1}$$

Correspondingly, for each group member node i:

$$\overline{MN_i}(\tau + 1) = \overline{RP_i}(\tau) + \overline{RM_i} \tag{4.2}$$

4.1.2 Pursue Mobility

The Pursue Mobility model represents a case where a number of MNs try to move towards a target node, as in typical tracking scenarios. The Figure 4.2 illustrates the basic idea of the model. If τ represents a time tick, $\overline{A_i}$ is a vector-valued acceleration function of the distance between the target node TN and MN_i, and $\overline{RM_i}$ denotes a random walk vector for each MN_i, the movement equation per each node can be written as follows:

$$\overline{MN_i}(\tau + 1) = \overline{MN_i}(\tau) + \overline{A_i}(\overline{TN} - \overline{MN_i}) + \overline{RM_i} \tag{4.3}$$

Note that the condition $|\overline{RM_i}| << |\overline{A_i}|$ must hold all the time in Formula 4.3, otherwise the group will scatter when $\tau \to \infty$.

4.1.3 Nomadic Community Mobility

While the Column Mobility model features individual reference points for group participants, Nomadic Community model introduces a common point of reference, RP. The model name follows from the behavior of wandering nomads camping out for the night; the camp fire now describes the reference point. The group mobility vector $\overline{GM}$ represents movement of this reference point (see Figure 4.3), and the distance how far MNs may wander from the appointed RP may be given as a parameter.

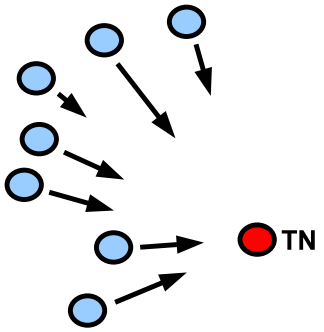


Figure 4.2: Pursue Mobility model

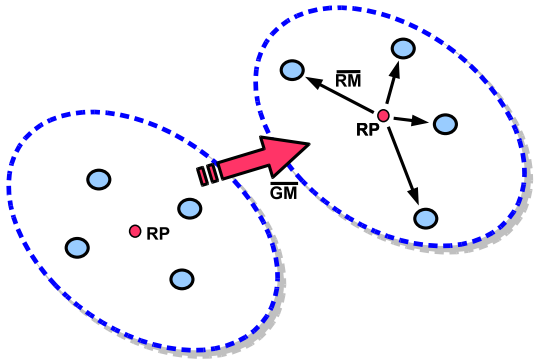


Figure 4.3: Nomadic Community model

Hence, the displacement vector for each node i can be written as:

$$\overline{MN_i}(\tau + 1) = \overline{RP_g}(\tau) + \overline{RM_i} \tag{4.4}$$

The Nomadic Community model can be considered as a less general form of Reference Point Group Mobility model described next.

4.1.4 Reference Point Group Mobility

The Reference Point Group Mobility (RPGM) model is one of the most well-known models as it comes to modeling military battlefield communication. In RPGM, each group node is given an individual reference point. Let τ be a time tick, then for the displacement vector of the reference point $\overline{RP}$ at every moment holds:

$$\overline{RP_g}(\tau + 1) = \overline{RP_g}(\tau) + \overline{GM} \tag{4.5}$$

Correspondingly, the position $\overline{RP_i}$ for each individual reference point can be given as:

$$\overline{RP_i}(\tau + 1) = \overline{RP_g}(\tau) + \overline{RP_i} \tag{4.6}$$

And finally for each node i it holds:

$$\overline{MN_i}(\tau + 1) = \overline{RP_i}(\tau) + \overline{RM_i} \tag{4.7}$$

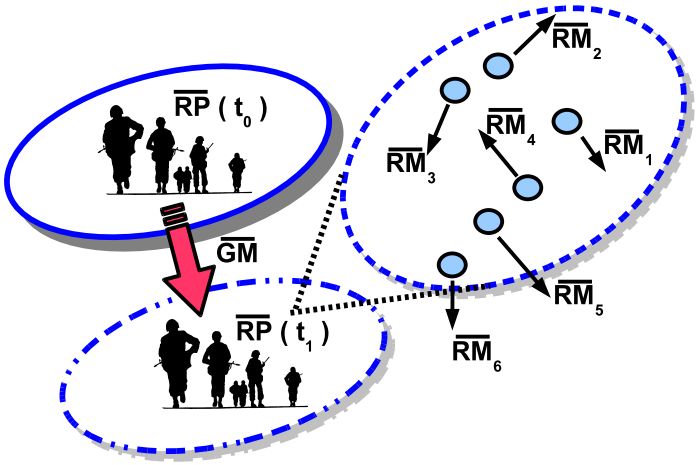


Figure 4.4: Reference Point Group Mobility model

Figure 4.4 illustrates the group movement of a patrol consisting of six soldiers. The conceptual center moves as the group proceeds, and at every tick some random motion is calculated for every participating node. The random motion vectors $\overline{RM_i} = (r, \theta) = (|\overline{RM_i}|, 0..2\pi)$ are calculated using uniform distribution.

As it can be seen, the RPGM model generalizes the three earlier presented models. A number of other spin-off models, such as Reference Region Group Mobility Model (RRGM) [25] and Reference Velocity Group Mobility Model (RVGM) [43], have been proposed fairly recently based on this model.

4.2 Group hand-offs defined

Group hand-offs can be defined as a way to carry out the hand-off process for multiple mobile nodes belonging into the same group in such manner, that the channel resource allocation happens simultaneously for all group participants, and so that the required signaling traffic is much less than the traffic that would incur if these nodes performed their hand-offs individually. Putting it another way, the group mobility is assumed as an optimized set of procedures needed to prepare the target network for an imminent surge of hand-offs for multiple channels, so that the required signaling is minimized. In the preferred case, the transmission does not degrade during hand-offs. The entire procedure should also be transparent to users.

If we assume that the group mobility is completely managed by the network, the mobile nodes are completely unaware of their membership in any logical groups: the group membership information must be maintained somewhere in a form or another – the information about which group each node belongs in and the total number of nodes belonging to each group should be included at minimum. Two parameters are specified for this purpose, the *group identifier* and the *number of nodes*, which are assumed to be present in every request for a group hand-off. Some information may be also included about the traffic and media capabilities of participant nodes.

For further examination of group mobile hand-offs, we need to introduce a virtual request for carrying the information required during such hand-off procedure. On this request with *groupid* and *nodes* given as parameters, all group participants are provided with the necessary connection parameters such as the new IP addresses to continue communicating seamlessly after the hand-off is finished. This process is illustrated as a signaling sequence diagram in the Figure 4.5. The serving base station, SBS, represents the current point of attachment for the group that is shortly to cede the control of the group traffic to the target base station, TBS.

When a mobile node indicates the need for hand-off, the SBS consults a database to see in which group the initiating node belongs. The identification can be done based on the SIP URI, MAC address or logical network address. The base station may also contain a group capability table – the information about individual node capabilities are signaled to the TBS during the address reservation request.

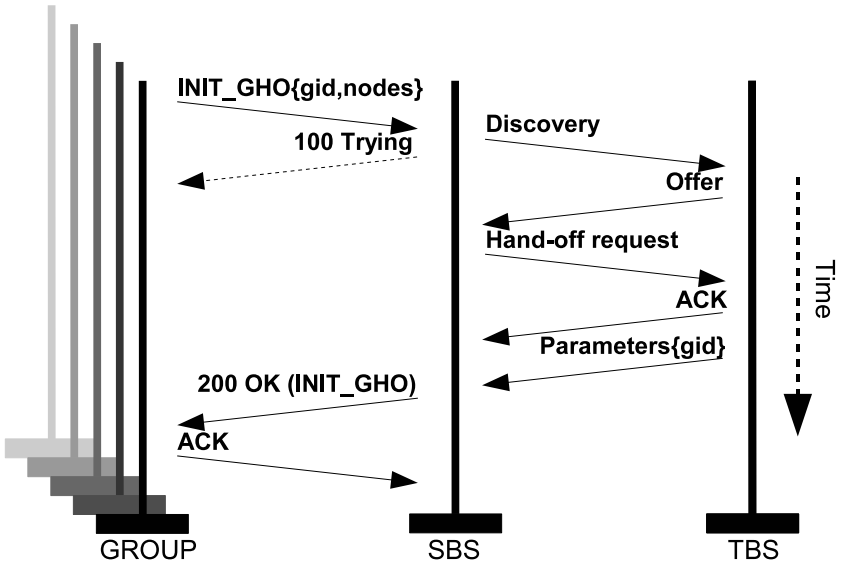


Figure 4.5: Signaling during a group hand-off

4.2.1 Statically configured and dynamically found groups

There are several ways to allocate group identifiers for groups. They may be pre-defined on per group basis during the operation preparation phase or generated dynamically during the operation.

Mobile groups may consist of 'dumb' terminals without configurable random access memory or sufficient computing power, or processing capabilities are otherwise minimal. In this case, the necessary group parameters must be statically pre-configured in databases located, for instance, at network base stations. The network terminals must work in "static" mode, where all group-related information and related decisions are managed by base stations.

Contrasting to the previous, 'dynamic' groups consist of nodes having enough processing capacity (e.g. laptops) to maintain and modify information about group structures and identifiers, and signal this information to the SBS during connection hand-off phase. Dynamic allocation may prove out as purposeful if group structures are likely to change during the operation. For instance, the dynamic group identifiers could be calculated as hash values of participating node addresses. This approach involves a difficulty with keeping databases synchronized if group identifiers need to be refreshed. The software must be also sophisticated enough to modify requests accordingly.

4.2.2 Group mobility influence on delay

Group mobility can take place at three different layers: application level, network level and link level. Deploying group mobility can have a greatly beneficial effect in improving hand-off efficiency in wireless networks. Much of these benefits source from the *anticipation of movement*, i.e. forecasting the future need for network resources. The forecasting can be done, for instance, based on the models discussed earlier in this chapter.

At the link layer, scanning, authentication and association delays are greatly hardware dependent. In 802.11-based networks, there is little to be done with the actual scanning delay (which actually comprises about 90% of the link-layer delay) [12]. However, we may influence on the required scanning frequency by the means of what we call *group caching*. After the L2 scanning operation, the possible AP/BS candidates are stored in a cache. The stored information can be used directly for reassociation during hand-offs without need for scanning. This data would be possible to store into a distributed group cache, where the whole group has an access. Only a single scan would be needed per group, and the scanning could be performed by the node first experiencing the field strength to drop under the Cell Search Threshold. Comparing to the scanning, cache operations happen very quickly.

A minor fraction of the L2 delay is used by authenticating mechanisms: Wireless Equivalent Privacy (WEP), Wi-Fi Protected Access (WPA/WPA2) and 802.11i. Group mobility does not have an effect on authentication and reassociation delays. We could explore group authentication for pure academic interest, but the achieved benefits are likely to remain marginal.

At the L3 layer, the need for network movement detection is completely eliminated by keeping an updated database of neighboring base stations and their network domains at the serving BS. The inefficient DHCP address allocation phase could be improved by performing the L3 address reservation for chunks of addresses instead of single network addresses. This means that the target BS reserves usable addresses at once for whole group of MNs.

At the application layer, we may envisage mechanisms allowing the whole group to be registered to a new domain using a single REGISTER message. This requires modifying SIP by introducing new requests or parameters.

4.3 Informing participating group nodes

After the group hand-off has been initiated, the information needs yet to be delivered to the other group participants. A viable solution would be using SUBSCRIBE/NOTIFY mechanism to alert group members about an imminent address change and deliver new network addresses. Subscription would be performed briefly after arriving to a new network and expired after the last group participating member leaves the current network. This mechanism requires a new event type, e.g. "address-change-listener", but does not impose the need for extensions to the current SIP message base or involve heavy modifications to the network facilities.

Another possible method for delivering parameters is the source-specific multicasting: all group members join as receivers to a multicast group managed by the domain administration upon arrival to the network. The new parameters are multicast to each of the joining members upon a hand-off occurrence. This approach would, however, bringing extra complexity by necessitating the support for multicast management protocols such as Internet Group Management Protocol (IGMP) or IPv6 Multicast Listener Discovery (MLD). If the goal is to keep the mobile equipment complexity minimal, this is not desirable. Additionally, since the IP multicast has never been widely used technology, there are doubts regarding the feasibility of such a solution.

Finally, it is possible that the other group members are kept uninformed about the group hand-off. All resource reservations are made beforehand by the AP/BS, and the incoming requests from MNs are mapped to match the new connection parameters at the AP/BS during the hand-off. Although this method enables transparent hand-offs from the MN perspective, the inevitable price is the increased complexity in the network side, as the AP/BS needs to hold updated state information for each MN.

4.4 Routing for groups

On the digital battlefield, communication systems are often characterized by distributed resources with limited bandwidth access and potentially rapidly changing network topology. In such environment, network scalability and efficient routing algorithms become extremely crucial. For efficient use of channel resources, also signaling must be kept conservative.

In systems utilizing *flat routing* schemes, a substantial amount of channel

capacity is spent on network controlling traffic such as periodic routing table updates as the network size grows. The good side is that the tables are available at any time, but the bandwidth is wasted considerably much. There are also on-demand routing schemes that can provide better scalability, but the drawback is increased routing latency that can render these protocols completely unsuitable for wide range of real-time applications.

In contrast, *hierarchical routing schemes* maintain a multilevel topology to separate networks and sub-networks recursively into logical clusters and larger super-clusters at different hierarchy levels. The signaling traffic is kept minimal by removing the need for flooding. The efficiency gain is achieved in expense of complexity in implementation and keeping databases updated as the network topology changes.

4.5 Hierarchical State Routing

The idea of Hierarchical State Routing model (HSR) [28] is based on the fact that group behavior is more likely to occur between nodes that possess some degree of spatial or temporal affinity. The nodes that exhibit particularly high level of reciprocal interaction can be clustered into groups formed by several participants. Taken as an example, a group may consist of a bunch of technical experts, a medical team on a battlefield or a group of people walking into the same direction.

By definition, HSR is a link-state routing protocol. It allows for multilevel clustering and logical subnet partitioning, providing a low-latency routing solution for applications which require group mobility support. The aim of clustering is to keep the radio channel utilization efficient, while reducing the size of routing tables. In a sense, clusters represent physical group affinity, whereas logical partition distinguishes different logical and functional levels where each node may reside. HSR is completely neutral on the question how the clusters should be formed.

4.5.1 Logical subnets

HSR features a notion of *logical subnet* to address group memberships. For instance, medical and sniper teams could be distinguished by their functions to belong into two separate groups, thus residing in separate logical subnets. Each mobile node may have one of three possible roles on different hierarchy layers.

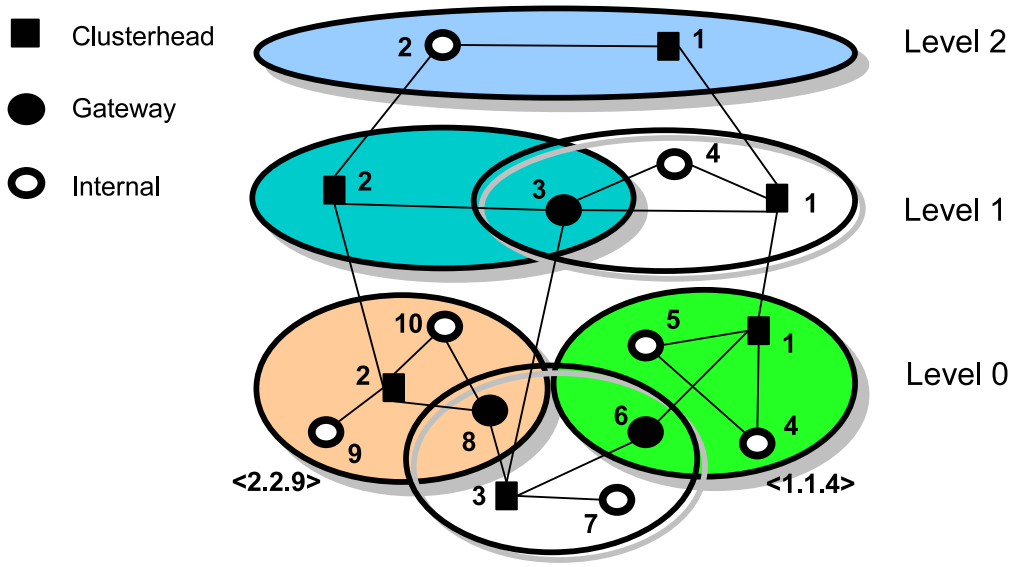


Figure 4.6: Hierarchical State Routing illustrated

Internal nodes are mobile nodes belonging explicitly to a single subnet, having no means to communicate with MNs belonging to other subnets without aid of gateway and clusterhead nodes.

Gateway nodes belong to multiple subnets, being responsible for forwarding (or bridging) data between different clusters. They are however unable to coordinate traffic.

Finally, each subnet must have one delegated *clusterhead node* coordinating inward and outward traffic within the subnet. Elected clusterheads on each level become representatives of their subnets on the next level.

A three-tier HSR network is exemplified in the Figure 4.6: the node <2.2.9> intends to deliver a datagram to the node <1.1.4> located in a disjoint cluster. Neither of the nodes have initially information about each other's actual location. The transmitting node first pushes the message to its group clusterhead <2.2.2>, which, after consulting its routing table, forwards the datagram logically upwards in the hierarchy until the receiving clusterhead, <1.1.1>, belongs to the same cluster, in this case at the level 2. This clusterhead knows a *virtual tunnel* between the nodes 1 and 2, namely 2-8-3-6-1-4. The message is returned back to the layer 0, and we see that the destination <1.1.4> is reached with a single logical hop, but the actual physical route for carrying the message becomes 9-2-8-3-6-1-4.

4.5.2 Node addressing

For addressing, each node has a unique *Node ID* coarsely corresponding to a usual MAC address. A *Hierarchical ID* (HID) is a sequence of Node IDs, consisting of node identifiers concatenated on the reference path starting from the uppermost level and ending at a mobile node. Any participating node can be unambiguously addressed through its HID.

Whereas HID addresses represent physical addresses (MAC), HSR utilizes its own logical network level addresses not much unlike IP addresses. This addressing mechanism associates each node with a subnet where it belongs, while each of these subnets corresponding to a particular user group. The *logical addresses* can be presented in format <*subnet, host*>. The logical network addresses are completely independent of NodeID (MAC) addresses, and logical subnets may span several physical clusters.

Each subnet must include at least one *Home Agent*, which manages logical group memberships in that subnet. Each node enrolls its HID in home agent database periodically, or when it performs a transition into another subnet. Correspondingly, Home Agents convey their own HIDs upward in the hierarchy. A time out mechanism takes care of erasing dead entries, should they not be be refreshed periodically. Every subnet joining node must know the HID address of the corresponding home agent.

Per default, the sender does not know in which cluster the message target resides. When it wants to send information to the destination, a *distributed location server* is first queried to find out the target's logical location. After the first transmission, the source and the destination have already learned each others' addresses, enabling the future transmissions to be carried out directly without any external help.

4.5.3 The HSR performance and issues

Assuming M to be the number of levels in the hierarchy, while N being the average number of nodes per each level, the hierarchical routing map requires merely $O(N \times M)$ entries comparing to $O(N^M)$ entries resulting from flat routing. The amount of control traffic remains low, since there is no need for search by flooding, even when the location of target node is not known. Because of the linearity, hierarchical routing offers considerably better performance in terms of routing overhead and scalability.

The downside of HSR, as with hierarchical routing protocols in general, is the constant need for updating routing databases, since cluster formations and hierarchical addresses of participant MNs may change in time. If a crucial network node (e.g. a clusterhead) is lost, the dynamic re-arrangement of the network topology may take a substantial time. The network performance might seriously suffer, or the network driven to a state of malfunction, should the recovery not happen quickly enough.

4.6 Example: group hand-offs for SIP

A HANDOFF request was once suggested for signaling SIP call hand-offs (Biggs & Dean, 2001) [4], but it was never seen as necessary to be adopted as a part of the protocol standard. Up to the present day, no proper mechanism for hand-off triggering has been implemented in SIP, let alone support for group mobility. Next we will summarize briefly the required changes.

We introduced an abstract request named GM_INVITE for triggering a group hand-off. This idealization, as well as the aforementioned draft [4], serves as an inspiration for our own group hand-off mechanism. For SIP implementation, we will introduce a new *"GHO-Reservation:"* header field. When the SBS receives an INVITE request with a *GHO-Reservation:* header-field included, it parses through the message contents, removing the whole line and storing the rest of the message, putting it on hold. Using an external protocol (how base stations actually communicate with each other is a detail), the SBS requests resources from the TBS for the given amount of MNs. When it receives an acknowledgment of the reserved resources, it sends the storaged INVITE request forward towards the CN.

The message may also contain explicit information about MNs' capabilities. Our example adheres to SDP, but any common session media description protocol is applicable.

```
INVITE sip:kaski@mil.fi SIP/2.0
Via: SIP/2.0/UDP examplerouter.mil.fi;branch=z9hG4bKdaxn8
Require: 100rel
To: Juhani <sip:kaski@mil.fi>
From: Gustav <sip:haggis@mil.fi>;tag=3EA4EC56
GHO-Reservation: gid=1234;nodes=6;
Call-ID: b42c1d22f23104@esikunta.mil.fi
CSeq: 11 INVITE
Contact: <sip:haggis@esikunta.mil.fi>
Content-Type: application/sdp
Content-Length: 273

v=0
o=gustav 1765868262 8206083726 IN IP4 194.100.112.72
s=A High-Priority Call
i=This call is extremely urgent and confidential
c=IN IP4 194.100.110.71
t=0 0
m=audio 1700 RTP/AVP 0 2
a=rtpmap:0 PCMU/8000
a=rtpmap:2 GSM/8000
m=video 1702 RTP/AVP 16
a=rtpmap:16 H261/90000
```

The behavior for the devised protocol extension can be formalized as follows:

1. The hand-off is initiated by the MN automatically when the field power decreases below the Cell Search Threshold. The MN sends an INVITE request with a *GHO-Reservation:* header field included, destined for the CN.

2. SBS receives the sent INVITE request, parsing the header line and extracting the relevant information from it. The message will be put temporarily on-hold, with the *GHO-Reservation:* header line removed.

3. The SBS consults its neighboring BS database, to see whether the MAC address of the TBS belongs into the same domain.

   (a) If yes, a normal link-layer hand-off (L2HO) will be performed. Go to Step 8.

   (b) If no, a network-level (L3) hand-off is needed. Go to Step 4.

4. The SBS requests the TBS to reserve L3-layer addresses and resources proactively using DHCP. A reservation reply containing the procedure acknowledgments and a list of allocated addresses is returned to the SBS.

5. The SBS modifies the INVITE request sent by the MN, updating the header fields using one of the addresses allocated at the Step 5. The INVITE request will now be sent to the CN.

6. The SBS updates the MN's location by sending a REGISTER request to a registrar server belonging to the new domain.

7. The CN opens a second session towards the MN and starts bicasting. The future packet exchange will happen through both tunnels.

8. The L2 hand-off for MN occurs. If L3 hand-off was involved: when the procedure is finished, the old session is closed. The new session is now used for all traffic.

9. Repeat steps 1–8 for each subsequent MN belonging to the group. Note that only one neighbor database consultation at the Step 3 and the L3 address allocation at the Step 4 is needed per group.

10. The group hand-off procedure is completed.

With the exception of group mobility, the outlined process has significant resemblance with the PAR-SIP mechanism described earlier at 3.4.4. In fact, the predictive address reservation can be implicitly assumed to be an integrated feature in the GM_INVITE mechanism.

The GM_INVITE formalization constitutes a coarse framework required in order to support seamless group hand-offs, addressing the elimination of the time-consuming DHCP and SIP re-establishment phases, while performing the address reservation for whole group at the same time. However, the framework neglects the details of intra-group traffic needed for signaling with other group participants, the HOAS operation and the signaling between base stations. Additionally, by not addressing recovery issues by making assumptions that all the necessary network resources are always available when the GM_INVITE process initiates and hand-offs are always successful, the model provides a theoretical template for mobile network design rather than a ready to be used implementation.

4.7 Chapter summary

This chapter discussed the group mobility concept and explained the benefits of group mobility models over classic individual mobility models, looking upon how the group mobility and seamless hand-offs could be efficiently integrated and deployed in infrastructure-based SIP protocol supporting networks. The chapter outlined the most common group mobility models, and at the end of the chapter, we proposed a simple SIP implementation for combining group mobility properties with PAR-SIP hand-off mechanism.

Mobility models are widely needed during the protocol design process and for system performance analysis. They are especially useful when trying to predict the future availability of wireless resources. When movement patterns and group constitutions are known, the future need of resources can be forecasted by using group mobility models.

Efficient routing schemes and group information databases become essential as the number of groups and group sizes grow. Flat routing models have a tendency not to scale well to network size, whereas traditional hierarchical models are unable to handle mobility efficiently enough. Hierarchical routing techniques such as HSR could be able to greatly enhance the routing performance and scalability in collaboration networks. Increased complexity and heavy bookkeeping requirements come as a cost as it comes to network implementation.

In the following chapters, the focus is put on defining the ways to measure group hand-off efficiency in SIP-signaled mobile networks and assessing group hand-offs quantitatively – first through mathematical observations, then by the means of computer simulation.

5 Performance metrics

In this chapter, we attempt to find appropriate metrics to evaluate hand-off performance quantitatively by first introducing a simplistic transmission path delay model applying theories provided by the field of queuing theory. Furthermore, we define and discuss the concept of hand-off efficiency in group mobility frame of reference. The discussed topics constitute the foundations for a simulation part described later, essentially by providing suitable parameters through which the simulation results are interpreted.

5.1 Analytical delay performance model

Several factors have an impact on the hand-off performance. Besides the actual network implementation, such factors as the user population, required DNS queries, used transport protocols and arrival process types all have an influence on the experienced latencies. In complex systems, modifying one parameter slightly might result in significant changes on other parameters. Therefore, estimating the impact of a differential change on a certain system variable requires us to do so having all other parameters unaltered.

As explained in the Chapter 3, the total hand-off delay can be expressed as a sum of its subcomponents. The subcomponents and the total accrued hand-off delay can be approximated using suitable mathematical methods. Some parameters, such as the network detection delay and re-configuration delay behave in a predictable fashion and have little contribution to the total latencies. Other components are highly dependent on the technological choices. For instance, the link layer delay is explicitly determined by the underlying network access technology. Taking as given that all mobile nodes are connected to the network via 802.11-based access, we may assume a simplified delay model for the link layer delay component.

Considering the link-layer delay (D_0) as given, only the DHCP (D_2) and SIP session re-establishment (D_4) delays have potential to affect the hand-off latencies significantly. For PAR-SIP hand-offs, these parameters get the value zero. The only remaining parameter of major analytical importance is the message transmission delay in the network.

5.1.1 Modeling transmission delay

A message in transit competes for the limited network capacity with other messages representing the same or some different traffic priority class. The

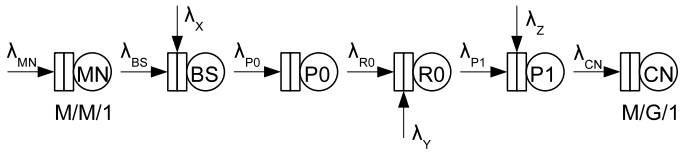


Figure 5.7: Transmission delay queue presentation

competing ambient traffic may originate from the other users connected to the same base station, the backbone network or the destination access network where the correspondent node resides. Functioning at the application layer, the SIP-based messages have systematically a lower priority than non-application layer messages.

Each node a message needs to pass in transit contributes to the transmission delay. Taking a queuing theoretical approach [2, 19, 27], each node can be presented as a separate queue. This bases on the idea of each node facing a message on the transmission path having a limited service rate and a queuing buffer with a certain amount of message places (see Fig 5.7). For simplicity, the model assumes no message is dropped or lost, in such manner that all messages reach the destination within a time ranging between $0..\infty$ milliseconds.

5.1.2 Priority queue-based delay model

Denoting the average message arrival rate into a queuing system by λ, the message processing rate at system nodes by μ, and the system load (or so called *utilization factor*) by $\rho = \lambda\mu^{-1}$, the average service delay $\overline{T}$ for a single customer in such a system can be approximated using priority queues.

Table 5.1: List of variables

$\lambda_{sip}, \lambda_{oth}$	Incoming traffic rates for SIP and other messages
ρ_{sip}, ρ_{oth}	Utilization factor for SIP messages and other messages
μ_k^{-1}	Mean service delay for traffic class k
S_k^2	Second moment for the traffic class k mean service delay

Using priority queues, it is possible to give an estimate for the service delay that a message spends at each intermediate node during transmission. Assuming that there are only two traffic classes, one for SIP traffic and another for all other IP traffic, we can approximate the sojourn delay at each node for each traffic type using formulas (variables as defined in Table 5.1):

$$\overline{T_{sip}} = \frac{(1-\rho_{sip}-\rho_{oth})\mu_{sip}^{-1} + (1/2)(\lambda_{oth}\overline{S_{oth}^2} + \lambda_{sip}\overline{S_{sip}^2})}{(1-\rho_{oth})(1-\rho_{sip}-\rho_{oth})} \tag{5.8}$$

$$\overline{T_{oth}} = \frac{(1-\rho_{oth})\mu_{oth}^{-1} + (1/2)(\lambda_{oth}\overline{S_{oth}^2})}{(1-\rho_{oth})} = \mu_{oth}^{-1} + \frac{\lambda_{oth}\overline{S_{oth}^2}}{2(1-\rho_{oth})} \tag{5.9}$$

The derivation for these formulas is detailed in Appendix D. As it can be seen, the system delay has a highly nonlinear dependency upon the average load parameters ρ_{oth} and ρ_{sip}.

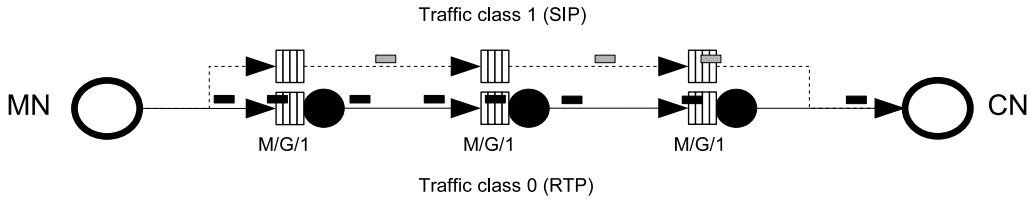


Figure 5.8: Separate priority queues for data and signaling traffic

The Figure 5.8 illustrates a transmission path with three intermediate nodes and two traffic classes. Although both classes share a common path of transmission physically, both classes can be imagined to have separate virtual queues at each node. The per-class sojourn times at each node can be estimated using Formulas 5.8 and 5.9. Assuming the arrivals and service times at each intermediating node similarly distributed, the total average times spent on the transmission path can be simplified as $3T_{sip}$ and $3T_{oth}$ respectively.

5.2 Analyzing group hand-off efficiency

Denoting the signaling traffic needed per mobile node during a conventional hand-off for a single MN by d_s, the involved group signaling traffic d_{gs} varies in the range: $d_s \leq d_{gs} \leq (n \times d_s)$, where n is the number of participating group members. The group hand-off efficiency can be conveniently estimated as a ratio $r_{gs} \mid (d_s \leq r_{gs} \leq \infty)$:

$$r_{gs} = \frac{d_{gs}}{n \times d_s} \tag{5.10}$$

The hand-off message size for a single node sets the information theoretical asymptotic lower bound for group hand-off message sizes. In the ideal case, the hand-off process can be carried out for the entire group by a single MN ($d_{gs} = d_s$), giving $r_{gs} = 1/n$ by the Formula 5.10. Values $r_{gs} \leq 1$ in general indicate improved efficiency and thus bandwidth saving. Hence, the signaling efficiency is improving proportionally to the amount of the hand-off signaling information we are able to pack in a single hand-off request and the number of nodes in the group. On the other hand, message sizes larger than the sum of message sizes required by ordinary hand-offs for individual nodes are wasting bandwidth (that is, $r_{gs} > 1$).

For the formulation presented in 4.6, the hand-off efficiency improves with the number of the allocated addresses the target TBS is capable of returning per single hand-off request, and decreases with the number of cycles needed to run the whole algorithm for an entire group.

5.3 Chapter summary

The chapter discussed the analytical methods to help us gain better understanding on measuring hand-off performances in SIP-based networks. Since the queuing and service delay times are the major contributors to delay, the queueing theoretical mathematical models become useful. Priority queues provide a convenient means for modeling delays on transmission paths. The network capacity and transmission times have a significant impact on hand-off times, therefore proper network dimensioning becomes crucial.

These parameters are later used as performance gauges to provide information about hand-offs in a computer simulated environment.

6 Computer simulation

The chapter summarizes the proceedings and outcome of the computer-based simulation conducted as a part of the thesis. We make conclusions on hand-off performance by observing simulated hand-off delays in the network as a function of system load and the signaling traffic aggregated in the network.

6.1 Overview on the simulation

The objective of this simulation part is to analyze the hand-off impact on data throughput and latency in a simulated network environment with varying utilization rates. Since no appropriate tools, neither software nor hardware-based, for modeling group hand-offs in infrastructure networks are available at the present time, our possibilities for carrying out large scale simulations with realistic test configuration are limited. Thus, we decide to adopt the computer simulation as the means of experimenting. Developing the simulation tool starting from scratch allows for high level of customization and control over the simulation process.

The simulations are carried out using a Java language-based discrete event software utility developed specifically for modeling the hand-off delays in SIP-based environments. The simulated scenarios are presented as a chronological sequence of events triggered by different entities in the simulated network. The used delay models are based on the mathematical assumptions presented in the previous chapter. A pseudorandom generator initialized using the current system time represents the only source of non-linearity in the system.

6.2 Simulation setup

We evaluate hand-off performances in a wireless network consisting of two domains, on a simulation area covering $1000 \times 1000m^2$. Both domains are regarded as individual 802.11 BSSs (Basic Service Sets) governed by a single wireless base station each. The network access points are labeled as AP0 (serving base station) and AP1 (target base station). The system comprises two out-bound SIP proxies, *"atlanta.com"* and *"biloxi.com"*. Figure 6.9 gives a schematic block model presentation of the system. The same scenario configuration is illustrated in Appendix E in detail. The involved network elements are as described in the Table 6.2.

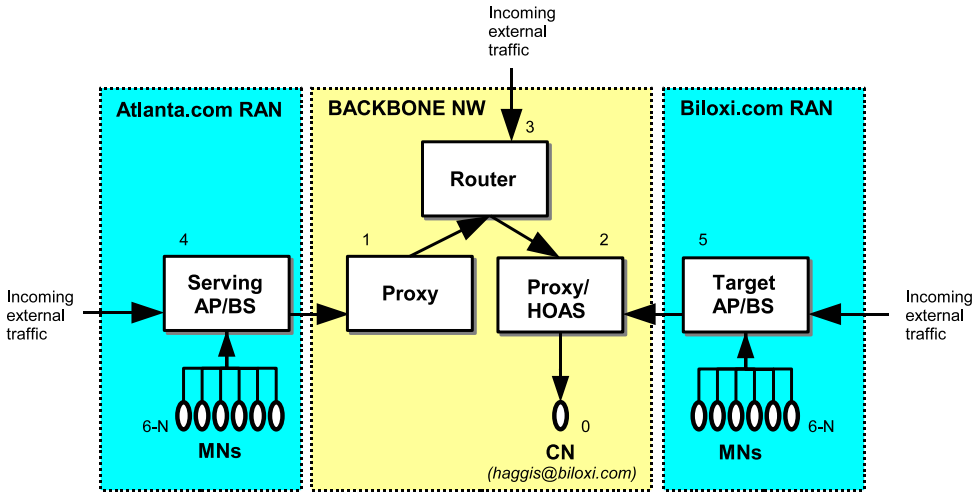


Figure 6.9: Block model presentation of the system

Table 6.2: Scenario nodes and their description

0	Correspondent node 'haggis@biloxi.com'
1	Out-bound proxy 'proxy.atlanta.com'
2	Out-bound proxy & HOAS 'proxy.biloxi.com'
3	Network router 'R1'
4	Base station 'AP0'
5	Base station 'AP1'
6-N	Mobile group members

The simulation scenario is taking place during a period of 10 seconds in the simulation time. It involves a group consisting of N mobile nodes (6 assumed in calculations) crossing the network border separating the domains, transmitting data unidirectionally to the corresponding node, *haggis@biloxi.com*, acting as a traffic sink. Besides the signaling traffic, the CN does not generate outgoing traffic flow.

The mobile group is considered to be moving in a column pattern. However, the simulation does not put strict assumptions on the group structure: the mobile node first arriving to the point of hand-off is considered to act as the group reference point during group hand-offs.

6.3 Simulation parameters

6.3.1 Traffic rates and service times

It is assumed that all mobile nodes produce G.711 PCM-based voice traffic with a bit rate of 64 kbit/s. Thus, the traffic volume generated by N mobile nodes varies in the range $0 \leq r_d \leq N \times 64$ kbit/s. During PAR bi-casting, the bit rate for a moving node temporarily doubles. At each time instant, the amount of aggregated data in the system varies in the range: $0 \leq r_d \leq 2N \times 64$ kbit/s.

All network links are assumed as having $BR = 1Mb/s$ capacity. All network nodes are assumed to pass IP packets at the line speed, $\mu_{oth}^{-1} = 1/BR$. The service times for SIP messages are considerably higher, $\mu_{sip}^{-1} = 100/BR$. This is intended to reflect lower traffic priority, and slower processing times in edge nodes.

6.3.2 SIP signaling messages

SIP messages get a higher weight factor for service delay calculations than other traffic types. The SIP message sizes are randomized using the Gaussian distribution, with the mean of 500 and deviation of 100 octets. That is, denoting the SIP message sizes in the system by s_M:

$$s_M \sim N(\mu, \sigma^2) = N(500, 100^2) \tag{6.11}$$

We make all signaling calculations following the worst-case assumption, that all (six) nodes are performing hand-offs simultaneously. In this case, the signaling consists $(6 \times 8 \times 0.5\ kB/s)/(6 \times 64\ kB/s + 6 \times 8 * 0.5\ kB/s) = 24/(768 + 24) \approx 3\%$ of the total traffic. We assume this ratio somewhat higher, assuming 5% instead. In practice, such signaling peaks would occur rarely. A group may also contain completely "silent" nodes, in which case the fractual part of signaling traffic from the total traffic can be remarkably higher.

6.3.3 Group mobility and bi-casting

The bicasting point, i.e. HOAS, is presumed to present somewhere in the simulation network, managing bi-casting during hand-offs. During a group hand-off, the data traffic is momentarily thought to contribute twice to the generated network traffic due to bi-casting. In real-world scenarios, the HOAS element could be envisioned to be a fixed part of a proxy server. Group hand-offs are assumed as ideal in such manner, that

the node first performing a hand-off triggers the hand-off for the whole group.

The traffic needed for carrying REGISTER and NOTIFY messages, as well as the signaling required by HOAS is not simulated. The presence of these messages is assumed implicitly by assuming a slightly higher signaling-to-payload ratio. It could be also assumed, that both group notification and re-registering procedures can be performed at some arbitrary time prior to (not during) hand-offs, so that the performance degrading impact on actual hand-off signaling is minimized. In particular, the re-registration could be performed immediately when the hand-off target network can be predicted with moderate accuracy. If the registration can be made for the whole group by a single node using a single message, the contribution of REGISTER messages on the total signaling traffic is negligibly small. The simulations assume that the signaling traffic consists a small fraction of the total traffic in transit, while payload data comprises the bulk. Otherwise, the proportional part of signaling may become significant, as it were the case if merely a few group nodes communicate while the others remain silent.

6.3.4 Simulating delay components

For parameters D_0, D_1, D_2 and D_3, we fix mean values as assumed in [18]: $D_0 = 50$ ms, $D_1 = 10$ ms, $D_2 = 1500$ ms, $D_3 = 10$ ms, with 20% mean deviation. The terms D4 and D5 are based on priority queue -based mean delay calculations with 10% deviation. The system delay characteristics can be studied using a black box -analysis by injecting traffic with well-known properties to the system and making conclusions from the resulted output.

The Figure 6.10 presents the impact of increasing system load on the SIP transmission delay in a single router as a function of system SIP traffic, assuming the ambient traffic λ_{oth} to be fixed at 50% of the maximum line capacity. The graph has been generated by sweeping the utilization in the range $0.5 \leq \rho_{sip} \leq 1.0$ with step size of 1/500, while injecting a single SIP message into the system on each run. After the node utilization reaches 75%, the expected delay grows abruptly, causing a rapid decline in the transmission performance and finally in hand-off latencies. The fluctuations visible in the graph in higher loads are due to the randomness in the assumed arrival process. This coarse computer-generated delay graph is based on a simplified M/G/1 priority queue delay model, demonstrating the drastic effects the improper capacity dimensioning in system components might have on transmission delays and hand-off latencies. For resilience, an appropriately dimensioned system provides enough redundant capacity even

for abrupt utilization peaks that might occur in the network.

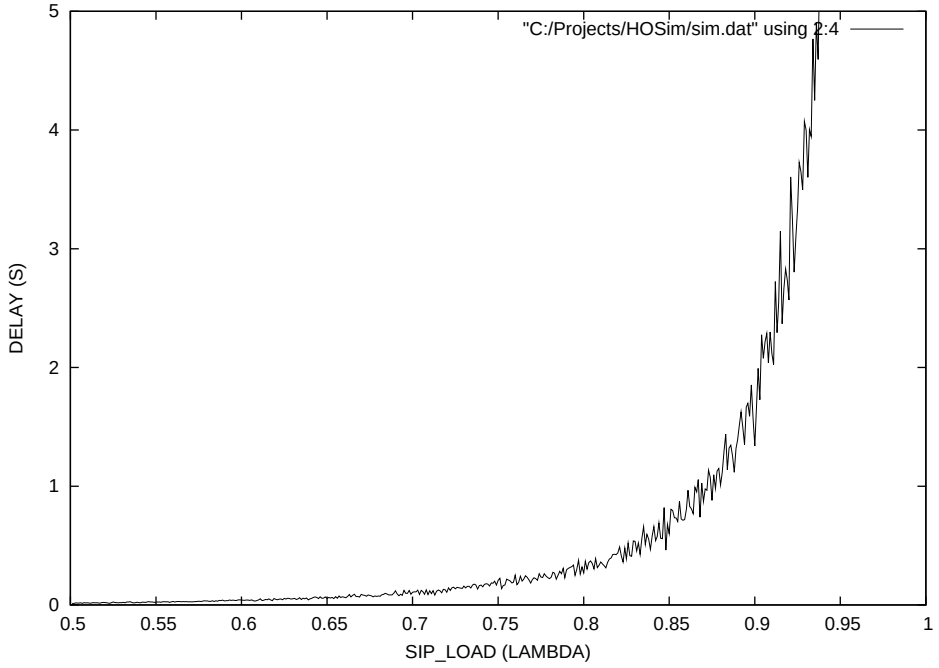


Figure 6.10: The signaling load impact on the signaling delay

6.4 Simulation results

6.4.1 Conventional and PAR hand-offs

Figure 6.11 shows the comparison between conventional and PAR-SIP hand-offs. PAR hand-offs (the lower graph) exhibit superior performance when comparing against conventional hand-offs. The DHCP impact on the total delay is clearly visible, it causes substantially higher expected delays and larger variance in hand-off times (the upper graph). At higher traffic loads, the impact of transmission related delay components D_4 and D_5 (for PAR-SIP merely D_5) become decisive, causing a rapid increase in the total hand-off delay.

Table 6.3 presents the total hand-off latencies for six mobile nodes, mean delays and standard deviations as a function of the system load ρ_{oth}. The results are calculated from six separate simulation rounds. The corresponding values for PAR-SIP hand-offs are presented in Table 6.4.

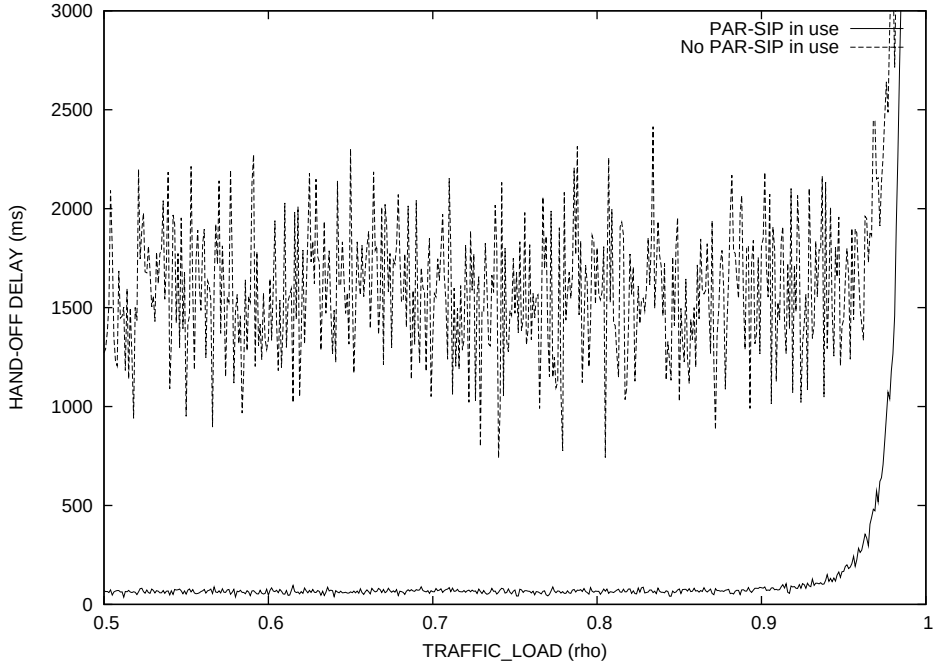


Figure 6.11: Conventional and PAR hand-off latencies compared

Table 6.3: Simulated conventional hand-off times (ms)

$\rho_{oth} \backslash MN_n$	MN_1	MN_2	MN_3	MN_4	MN_5	MN_6
0.50	1387.2	1571.7	1330.2	1911.4	1335.0	1706.8
0.75	1745.4	2072.9	2038.6	985.1	1858.7	1272.9
0.80	2001.0	1754.3	1602.4	1065.0	1975.1	1726.0
0.85	1844.5	1869.5	1996.6	1720.0	1976.2	2148.9
0.90	2722.4	2793.5	3540.8	3393.7	3420.4	3148.7
0.95	$\rightarrow \infty$	$\rightarrow \infty$	$\rightarrow \infty$	$\rightarrow \infty$	$\rightarrow \infty$	$\rightarrow \infty$
Stats	$\mu =$	1997.2 ms		$\sigma =$	677.5 ms	

As predicted, the PAR mechanism seems to bring enormous performance benefits when compared against conventional SIP hand-offs. Without the DHCP impact in the worst case accounting for more than 90% of the total latency during conventional hand-offs, delays seem to be low enough to make seamless mobility service provision possible.

Table 6.4: Simulated hand-off times (ms) using PAR

$\rho_{oth} \backslash MN_n$	MN_1	MN_2	MN_3	MN_4	MN_5	MN_6
0.50	71.2	49.1	51.2	59.3	71.5	58.3
0.75	60.8	61.9	63.8	78.0	75.3	63.9
0.80	70.8	62.9	66.8	68.9	67.1	65.0
0.85	61.6	71.5	66.3	75.0	79.8	81.9
0.90	72.9	74.9	72.5	79.6	81.2	91.7
0.95	130.3	121.8	129.4	108.3	120.5	102.2
Stats	$\mu =$	77.4 ms		$\sigma =$	21.0 ms	

Table 6.5: Individual vs. group PAR hand-off performances compared

$N_{nodes} \backslash MN_n$	MN_1	MN_2	MN_3	MN_4	MN_5	MN_6	r_{gs}
10	101.5	129.8	118.8	115.3	132.6	114.4	>1.0
20	140.2	121.4	126.8	127.2	124.3	103.8	1.0
50	119.3	152.8	125.8	132.0	126.3	125.0	0.4
100	129.3	146.7	127.0	127.9	116.0	140.2	0.2
250	174.6	153.5	150.7	165.6	170.8	146.9	0.08
1000	296.6	307.8	328.4	297.0	336.8	287.3	0.02
Stats	$\mu =$	162.2 ms		$\sigma =$	69.0 ms		—
GroupHO	132.5	141.5	119.8	86.4	129.5	122.9	—
Stats	$\mu =$	**122.1 ms**		$\sigma =$	**19.1 ms**		—

6.4.2 Group hand-offs

We assess the group hand-off impact for different group sizes of 10, 25, 50, 100, 250, 500 and 1000 nodes. The hand-off delays are calculated again for six nodes picked out randomly from the group, all thought as uniform in terms of characteristics. The nodes are assumed not to interact with each other in any other way than by generating traffic and thereby contributing to message delay times. The logic is as follows: considering a group of a given size N, we assume that at each time instant a group node performs a hand-off with a probability P. Hence, at each time instant, $N_{ho} = NP$ hand-offs are expected to occur. If also premised, that each hand-off request is sized S, the total signaling traffic generated is $\rho_{sip} = N_{ho}S$. Keeping ρ_{oth} fixed, group hand-offs can be studied by calculating hand-off latencies with $\rho_{sip} = 0$. Only a fraction of nodes is assumed to be active at each time instant, in such way that approximately 50% of utilization consists of payload traffic. The hand-off probabilities are assumed as time-invariant and uncorrelated in such manner that the hand-off probabilities stay the

same for each node all the time, regardless of whether the same or some other has performed a hand-off at the preceding time instant. The results are presented in the Table 6.5. First six lines are individual hand-off times for given group sizes, the last line presents experienced hand-off latencies during a group hand-off. We have presumed $S = 500$ (bytes) and P = 0.05. Also the group hand-off efficiency ratio r_{gs}, as defined by Formula 5.10, is shown. It should be noticed, that the group hand-off latency is dependent on the number of group nodes only indirectly by the generated loads ρ_{oth} and ρ_{sip}.

For small groups and low signaling traffic volumes, the group mobility impact appears negligible; the payload data flows are enormously voluminous compared to the signaling traffic, hiding the statistical features of signaling traffic behind fluctuations in the payload traffic and rendering the SIP traffic entirely indiscernible, when comparing to individual mobility scenario with PAR in use. The benefits become visible with large groups at very high channel utilization rates when the hand-off rate is high, in the area where the transmission delay component D_5 becomes dominant ($0.9 < \rho_{oth} < 1.0$). In very high-speed networks with low hand-off rate, group hand-offs benefits become marginally small.

6.5 Chapter summary

This chapter explained the proceedings and the numerical outcome of the computer simulation that was carried as a part of the thesis. Our intent was to gather information about the hand-off performance for conventional, PAR-SIP and group handoffs.

With PAR-SIP hand-offs enabled, the impact of address allocation and SIP re-establishment phases on the total hand-off delay are both practically eliminated, dropping the total hand-off delay times to a fraction from the original. Predictive Address Reservation seems to be the true enabler of seamless hand-offs. The obtained simulation results therefore support our theoretical hypotheses.

For small mobile groups, group hand-offs seem not to bring any substantial benefits over individual SIP hand-offs. This is due to the high payload-to-signaling traffic volume ratio. The performance gains emerge in measurable degree only in high-load systems with a considerable amount of signaling in progress all the time, and hundreds or thousands of mobile nodes. We do not rule out the possibility of other beneficial and potentially achievable side effects, e.g. security or hand-off reliability related, but this topic is out of the scope. The bi-casting impact on hand-off reliability has not been studied comprehensively, either.

7 Conclusions

This chapter concludes the thesis and provides a recapitulation of the observations made during the preceding chapters, presenting the reader a relevant summary of main points and issues. We also evaluate the realizability of the introduced technologies and furthermore assess the future prospects concisely.

7.1 Summary of the findings

This exploratory thesis focused on the provision of group mobility for delay sensitive SIP-based applications in narrowband authority networks, considering the capabilities and performance of such technologies. The superior price and the commoditization of WLAN-based technologies gave us a well-rounded reason to examine their viability for such purpose.

SIP has matured into a widely accepted signaling protocol. Being modular, extensible and undergoing rapid development, it has attained a high level of popularity in civilian settings, particularly in desktop VoIP applications. Therefore, it provides a suitable basis for our observations.

However, its unsuitability for narrowband environments have raised major performance concerns. The basic SIP protocol specified in RFC 3261 is not yet applicable in such environments due to the incurring signaling overhead and latency primarily originating from the address re-allocation process. The protocol is suitable for real-time applications (such as VoIP) signaling, but cannot guarantee appropriate service level for TCP mobility; for persistent data connections, MIP provides a better approach.

When group sizes grow bigger, the radio links may incur congestion due to the increased amount of signaling needed for hand-off executions. We presented several potential alternatives to reduce the latencies involved in hand-off process. The SIP mid-call mobility can be made more efficient by Predictive Address Reservation mechanism, which aims at decreasing the hand-off latency through the proactive address allocation.

The second important finding is how to make hand-offs more efficient and error resilient by dealing with logical groups instead of individual nodes, resulting in minimized amount of signaling with optimal bandwidth efficiency. By the reduced signaling, also a fraction of the delay occuring during hand-offs could be potentially eliminated.

Our theoretical contribution was a brief example how group hand-offs can be supported in SIP. We also modeled group mobility in a simulated environment to support our considerations. The final conclusion is that the group mobility enhanced PAR-SIP mechanism would allow flexible and efficient signaling required in order to enable seamless terminal mobility.

7.2 Limitations, final remarks & future work

Our goal was to explore the possibilities for group mobility in infrastructured networks, and to provide a theoretical framework for possible real-world applications, considering rather on defining the problem and the related requirements instead of analyzing the particular technologies involved. The adopted network model was simplified, neglecting the inter-BS signaling, access network operation, network collisions, bi-casting and the HOAS operation entirely. The possibility that a node can be connected to several base stations at a time (i.e. soft hand-off) was not considered. A detailed analysis per access technology platform would be needed for future implementations. The intra-group signaling was considered only briefly. Also, security issues have been left with little attention.

The decision to simulate the group mobility in 802.11x-based environment is not necessarily very realistic, since very promising technologies with better scalability properties are emerging. Brand new technologies such as IEEE 802.21, also known as Media Independent Handover (MIH), are arriving and expected to fare better in supporting seamless hand-offs. However, we justify the choice of using WLAN instead of, say, WiMAX by popularity and price; the previous clearly excels the latter in both criteria.

Although we focused on the most common transport protocols available, emphasizing TCP and UDP while paying little attention on more advanced protocols, it can be anticipated that the transport protocols will yet undergo considerable development regarding mobility and security properties. Thus, the performance for group hand-offs should be also evaluated in such systems.

The performance benefits that can be obtained by group mobility and PAR-SIP remain to be investigated in real-life settings. A logical continuation for this research is to build a working testbed using real network terminals. For early testing purposes, a simple 802.11x -based WLAN testbed should suffice.

References

[1] Audet, F.; Jennings, C., "Network Address Translation (NAT) Behavioral Requirements for Unicast UDP", IETF RFC 4787, Best Current Practice, January 2007

[2] Banerjee, N.; Basu, K.; Das, S., "Hand-off Delay Analysis in SIP-based Mobility Management in Wireless Networks", IPDPS'03, IEEE, 2003

[3] Berners-Lee, T. et al., "Uniform Resource Identifiers (URI): Generic Syntax", IETF RFC 2396, Standards Track, August 1998

[4] Biggs & Dean, "SIP Call Control: Call Handoff", IETF Internet Draft, draft-dean-handoff-00.txt, 19 January 2001

[5] Camarillo, G., "Compressing the Session Initiation Protocol (SIP)", IETF RFC 3486, Standards Track, February 2003

[6] Campbell, B. (Ed.); Rosenberg, J.; Schulzrinne, H.; Huitema, C.; Gurle, D., "Session Initiation Protocol (SIP) Extension for Instant Messaging", IETF RFC 3428, Standards Track, December 2002

[7] Cha, E.; Lee, K.; Kim,M., "Cross Layer Fast Handoff for SIP", AINA'07, IEEE, 2007

[8] Chahbour, F.; Nouali, N.; Zeraoulia, K.: "Fast Handoff for Hierarchical Mobile SIP Networks", April 2005

[9] Garcia-Martin, M.; Bormann, C.; Ott, J.; Price, R.; Roach, A.B, "The Session Initiation Protocol (SIP) and Session Description Protocol (SDP) Static Dictionary for Signaling Compression (SigComp)", IETF RFC 3485, Standards Track, February 2003

[10] Gulbrandsen, A.; Vixie, P.; Esibov, L., "A DNS RR for specifying the location of services (DNS SRV)", IETF RFC 2782, Standards Track, February 2000

[11] Handley, M.; Jacobson, V., "SDP: Session Description Protocol", IETF RFC 2327, Standards Track, April 1998

[12] Hautala, M., "Mobile IPv6 performance in 802.11 networks: handover optimizations on the link and network layer", HUT, Department of Electrical and Telecommunications Engineering, 13 March 2006

[13] Izumikawa, H.; Lillie, R., "SIP-based Bicasting for Seamless Handover between Heterogeneous Networks", IETF Internet Draft, draft-izumikawa-sipping-sipbicast-00, November 2007

[14] Ott, J., "SIP: Session Initiation Protocol", lecture handouts, p.26, 2006. http://www.netlab.tkk.fi/opetus/s383150/2006/sip-slides/5-sip-intro.pdf

[15] Jung, J-W.; Kahng, H-K.; Mudumbai, R.; Montgomery, D., "Performance Evaluation of Two Layered Mobility Management using Mobile IP and Session Initiation Protocol", 2003

[16] Kawata, T.; Shin, S.; Forte, A.G.; Schulzrinne, H., "Using Dynamic PCF to Improve the Capacity for VoIP Traffic in IEEE 802.11 Networks", 2005

[17] Kwon, T.T. et al., "Mobility Management for VoIP Service: Mobile IP vs. SIP", IEEE, October 2002

[18] Kim, W. et al., "Link Layer Assisted Mobility Support Using SIP for Real-time Multimedia Communications", MobiWac'04, 1 October 2004

[19] Kleinrock, L., "Queuing Systems Vol. I: Theory", John Wiley & Sons, New York, 1975

[20] Lopez, G. G.; Wang, Q.; Abu-Rgheff, M. A.; Akram, A., "A MIP-SIP Macro-mobility Management Scheme for VoIP Across Wired and Wireless Domains", IEE, 2004

[21] McAuley, A.; Subir, D.; Sunil, M.; Shinichi, B.; Yasuro, S., "Dynamic Registration and Control Protocol", draft-itsumo-drcp-01.txt, 14 July 2000

[22] Mealling, M., "Dynamic Delegation Discovery System (DDDS), Part Three: The Domain Name System (DNS) Database", IETF RFC 3403, Standards Track, October 2002

[23] Merger, M., "Liikkuvuudenhallinta Mobile IP versio 6 -protokollalla", HUT, Department of Electrical and Telecommunications Engineering, 7 December 2004

[24] Mäenpää, J., "Performance of Signalling Compression in the Third Generation Mobile Network", HUT, Department of Electrical and Telecommunications Engineering, 7 June 2005

[25] Ng, J.M.; Zhang, Y., "Impact of Group Mobility on Ad hoc Networks Routing Protocols", 2006

[26] Olson, S.; Camarillo, G.; Roach, A.B., "Support for IPv6 in Session Description Protocol (SDP)", IETF RFC 3266, Standards Track, June 2002

[27] Harrison, P.G.; Patel, N.M., "Performance Modelling of Communication Networks and Computer Architectures", Addison-Wesley, 1992

[28] Pei, G.; Gerla, M.; Hong, X.; Chiang, C-C., "A Wireless Hierarchical Routing Protocol with Group Mobility", IEEE, 1999

[29] Perkins, C., "IP Mobility Support for IPv4", IETF RFC 3344, Standards Track, August 2002

[30] Popescu, I., "Supporting Multimedia Session Mobility using SIP", CNSR 2003 Conference, May 2003

[31] Price, R. et al., "Signaling Compression (SigComp)", IETF RFC 3320, Proposed Standard, January 2003

[32] Roach, A.B., "Session Initiation Protocol (SIP)-Specific Event Notification", IETF RFC 3265, Standards Track, June 2002

[33] Rosenberg, J. et al., "SIP: Session Initiation Protocol", IETF RFC 3261, Standards Track (Obsoletes RFC 2543 [41]), June 2002

[34] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", IETF Internet Draft, draft-ietf-mmusic-ice-19, October 2007

[35] Rosenberg, J.; Mahy, R.; Huitema, C., "Traversal Using Relay NAT (TURN)", IETF Internet Draft, draft-rosenberg-midcom-turn-08, September 2005

[36] Rosenberg, J.; Schulzrinne, H., "Reliability of Provisional Responses in the Session Initiation Protocol (SIP)", IETF RFC 3262, Standards Track, June 2002

[37] Rosenberg, J.; Weinberger, J.; Huitema, C.; Mahy, R., "STUN - Simple Traversal of User Datagram Protocol (UDP) Through Network Address Translators (NATs)", IETF RFC 3489, Standards Track, March 2003

[38] Sass, P., "Communications networks for the Force XXI Digitized Battlefield", 5 October 1999

[39] Schulzrinne, H. et al., "VoIP in IEEE 802.11 Networks", MobiArch 2006 Keynote Speeches, March 2006. http://www3.ietf.org/proceedings/06mar/slides/raiarea-1/raiarea-1.ppt

[40] Schulzrinne, H. et al., "RTP: A Transport Protocol for Real-Time Applications", IETF RFC 3550, 2003

[41] Schulzrinne, H. et al., "SIP: Session Initiation Protocol", IETF RFC 2543, Standards Track (Obsolete), March 1999

[42] Steffen, A.; Kaufmann, D.; Stricker, A., "SIP Security", 18 September 2004, http://security.hsr.ch/docs/DFN_SIP.pdf

[43] Wang, K. H.; Li, B., "Group mobility and partition prediction in wireless ad hoc networks", Proc. IEEE Int'l Conf. on Communications (ICC), 2002

[44] Wanq, Q.; Abu-Rgheff, M. A., "Signalling analysis of cost-efficient mobility support by integrating mobile IP and SIP in all IP wireless networks", Intern. Journal of Comm. Systems, vol. 19, iss. 2, 2006

[45] Wedlund, E.; Schulzrinne, H., "Mobility Support using SIP", 1999

[46] Wedlund, E.; Schulzrinne, H., "Application-Layer Mobility Using SIP", Mobile Computing and Communications Review, Volume 4, Number 3, July 2000

A List of SIP & SDP specifications

RFC 2327: SDP: Session Description Protocol
RFC 2976: The SIP INFO Method
RFC 3261: SIP: Session Initiation Protocol
RFC 3262: Reliability of Provisional Responses in the Session Initiation Protocol
RFC 3263: Locating SIP Servers
RFC 3264: An Offer/Answer Model with the Session Description Protocol (SDP)
RFC 3265: Session Initiation Protocol (SIP)-Specific Event Notification
RFC 3311: The Session Initiation Protocol (SIP) UPDATE Method
RFC 3320: Signaling Compression (SigComp)
RFC 3321: Signaling Compression (SigComp) - Extended Operations
RFC 3323: A Privacy Mechanism for the Session Initiation Protocol (SIP)
RFC 3324: Short Term Requirements for Network Asserted Identity
RFC 3325: Private Extensions for Asserted Identity within Trusted Networks
RFC 3398: ISDN User Part (ISUP) to Session Initiation Protocol (SIP) Mapping
RFC 3407: Session Description Protocol (SDP) Simple Capability Declaration
RFC 3428: Session Initiation Protocol (SIP) Extension for Instant Messaging
RFC 3485: SIP and SDP Static Dictionary for Signaling Compression (SigComp)
RFC 3486: Compressing the Session Initiation Protocol (SIP)
RFC 3515: The Session Initiation Protocol (SIP) Refer Method
RFC 3665: Session Initiation Protocol Basic Call Flow Examples
RFC 3666: Session Initiation Protocol PSTN Call Flows
RFC 3702: Authentication, Authorization and Accounting Requirements for SIP
RFC 3725: Best Current Practices for Third Party Control (3pcc) in the SIP
RFC 3824: Using E.164 Numbers with the Session Initiation Protocol (SIP)
RFC 3841: The Caller Preferences for the Session Initiation Protocol
RFC 4077: Negative Acknowledgement Mechanism for Signaling Compression
RFC 4168: The SCTP as a Transport for the Session Initiation Protocol (SIP)
RFC 4474: Enhancements for Authenticated Identity Management in the SIP
RFC 4485: Guidelines for Authors of Extensions to the Session Initiation Protocol
RFC 4568: SDP Security Descriptions for Media Streams

B List of SIP responses

1xx	PROVISIONAL RESPONSE
	100 Trying
	180 Ringing
	181 Call Is Being Forwarded
	182 Queued
	183 Session Progress
2xx	SUCCESSFUL TRANSACTION
	200 OK
	202 Accepted (request received by a forwarding service, not the destination)
3xx	REDIRECTION MESSAGE
	300 Multiple Choices
	301 Moved Permanently
	302 Moved Temporarily
	305 Use Proxy
	380 Alternative Service
4xx	ERROR IN CLIENT
	400 Bad Request
	401 Unauthorized (for registrar use only)
	402 Payment Required (reserved for future use)
	403 Forbidden
	404 User not found
	405 Method Not Allowed
	406 Not Acceptable
	407 Proxy Authentication Required
	408 Request Timeout (time-out for user search exceeded)
	410 Gone (the user no more exists)
	413 Request Entity Too Large
	414 Request URI Too Long
	415 Unsupported Media Type
	416 Unsupported URI Scheme
	420 Bad Extension (protocol extension not understood by the server)
	421 Extension Required

	423 Interval Too Brief
	480 Temporarily Unavailable
	481 Call/Transaction Does Not Exist
	482 Loop Detected
	483 Too Many Hops
	484 Address Incomplete
	485 Ambiguous
	486 Busy Here
	487 Request Terminated
	488 Not Acceptable Here
	491 Request Pending
	493 Undecipherable (undecryptable S/MIME part)
	494 Security Agreement Required
5xx	ERROR IN SERVER
	500 Server Internal Error
	501 Not Implemented (the requested method not implemented here)
	502 Bad Gateway
	503 Service Unavailable
	504 Server Time-out
	505 Version Not Supported (no support for this SIP version)
	513 Message Too Large
6xx	GLOBAL FAILURE
	600 Busy Everywhere
	603 Decline
	604 Does Not Exist Anywhere
	606 Not Acceptable

C NAT traversal

Network Address Translation involves always mapping of internal network address/port-pairs to external address/port-pairs using a NAT functionality typically present in the border router. The primary difference between the NAT types is the way how this mapping is done, and which policy is used when communicating with external entities. Traditionally, four main types of NATs (defined in RFC 3489 [37]) exist, although the strict categorization has lost much of its significance by the modern NATs that combine the features of the different specified types. Despite the old classification is being superseded by RFC 4787 [1] featuring a revised terminology, it is still commonly used.

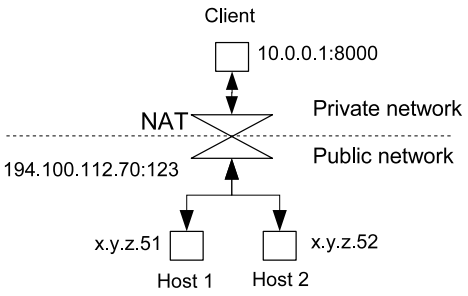


Figure C.1: Open Cone NAT

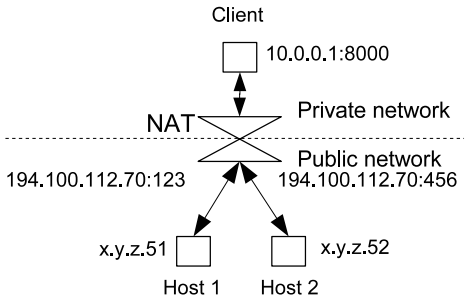


Figure C.2: Symmetric NAT

Full Cone All requests from the same internal IP addresses are mapped to the same external addresses and ports. External entities can communicate with the internal hosts via the mapped external addresses. See Figure C.1.

Restricted Cone Functions just as full cone NAT, with the exception that the external host cannot communicate with the internal host until the internal host has sent a packet to the external host's IP address first.

Port Restricted Cone The same as restricted cone, but including port numbers. The external hosts may communicate with the internal host using a specific port, if it is contacted first at this by the internal host.

Symmetric NAT All requests from an internal address/port pair are mapped to a unique address for each external destination. Each host in the public network sees the client behind a symmetric NAT at a different port (see Figure C.2). An external host cannot contact a host behind the NAT unless it

has been contacted by the host first. Symmetric NAT is the most modern and the most problematic address translation method from the viewpoint of SIP.

While having numerous favorable features, using NAT has a few drawbacks that need to be addressed when deploying SIP or peer-to-peer services. Many NAT types restrict external hosts from initiating for instance a VoIP-connection from outside, which is not desirable. Since the SIP itself usually relies on a well-known port 5060, it is not usually the problematic part in forming a connection through a NAT/firewall. The problem is with RTP streams, which are usually ephemeral and dynamically reserved. The SIP-signaled media commonly takes a different path and port numbers than the signaling. Without analyzing the contents of a packet, a firewall does not have any way to know if the stream should be associated with a particular ongoing session; generally, if such a stream should be let through or not. In practice, this means that when the caller is trying to connect the callee, the signaling goes through and the phone rings, but when the callee picks up the call and answers, no voice is going through and the line stays mute.

One solution for the problem is using SIP Application Layer Gateway (SIP ALG). Other solutions involve walkarounds such as Simple Traversal of UDP Through NATs (STUN).

D Queuing theory

Queuing theory constitutes a branch of applied probability theory, providing methods for the mathematical analysis of queues and processes. A concise introduction to the rudimentary concepts of queueing theory is given next.

Kendall notation

Queueing models are typically classified using so called *Kendall notation*. The extended Kendall notation consists of six parameters, but often a simple version with three parameters are used. In the simplest form, queueing processes are described as:

$$A/B/C,$$

where A denotes the arrival process, B the service time distribution, and C the number service units. There are several distribution types introduced in literature, but the most commonly seen and the simplest types are:

M	Exponential (or *Markovian*) distribution
D	Deterministic (or fixed) distribution
G	General distribution

Example queuing models

M/M/1 A single server process with Markovian distribution for inter-arrival and service processes. Calculations for such processes are usually simple.

M/D/1 A single server process with Markovian inter-arrival and fixed time service processes.

M/G/1 A single server process with Markovian inter-arrival and general distribution for service process.

G/G/∞ A process with infinite number of service processors and general distribution for inter-arrival and service processes. Typically analyzing such processes tends to be challenging.

Little's theorem

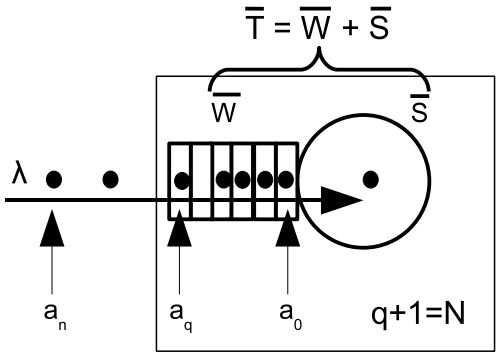


Figure D.3: Little's theorem illustration

The Figure D.3 illustrates a one-server system with q queueing places and incoming traffic rate λ. Thus, the number of customers in the system $\overline{N}$ at an arbitrary time instant consists of the number of customers queuing plus the one being currently served. The total time a customer is expected to spend in the system, the *sojourn time* $\overline{T}$, therefore consists of the time $\overline{W}$ spent in the queue plus the service time $\overline{S}$ required.

To establish a dependency between the number of system customers and the sojourn time, we may formalize:

$$\begin{aligned} \overline{N} &\triangleq \text{total number of customers} \\ \lambda &\triangleq \text{arrival traffic intensity,} \\ \overline{T} &\triangleq \text{average service time,} \end{aligned}$$

$$\overline{N} = \lambda \overline{T} \tag{D.12}$$

The Formula D.12 is commonly known as *Little's theorem*. It approximates the long-term number of customers in a system in equilibrium state, when the arrival rate and the service time are known. The theorem is a general result, being valid for all queueing models.

Service delay evaluation in M/G/1 models

The queueing delay a new incoming customer faces at arrival consists of the total service time required for $\overline{N_q}$ customers ahead of him to be served, plus the residual time $\overline{R}$ required to complete the service for the customer currently being served.

Denoting,

$$\overline{R} \triangleq \text{mean residual service time},$$
$$\overline{N_q} \triangleq \text{number of queueing customers},$$
$$\overline{\mu} \triangleq \text{system service rate},$$
$$\overline{S} = \mu^{-1} \triangleq \text{mean service delay},$$
$$\overline{\rho} = \lambda\mu^{-1} = \lambda\overline{S} \triangleq \text{system load (i.e. } \textit{utilization factor}),$$

Applying Little's theorem D.12 by substituting $\overline{N_q} = \lambda\overline{W}$, the total expected queuing time $\overline{W}$ for a randomly chosen tagged message can be written as:

$$\overline{W} = \overline{R} + \overline{N_q S} = \overline{R} + \lambda\overline{WS} = \overline{R} + \rho\overline{W} \Rightarrow \overline{W} = \frac{\overline{R}}{1-\rho} \tag{D.13}$$

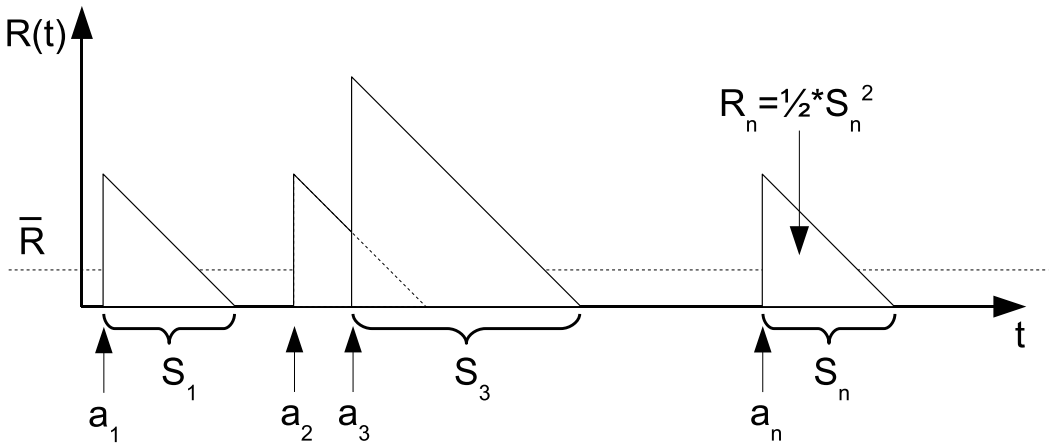


Figure D.4: Mean residual time $\overline{R}$ on a long time period

In the general case, finding the residual time $\overline{R}$ is an intractable operation. The Figure D.4 illustrates customer arrivals at time instants $a_1, a_2, ..., a_n$ and their service times $S_1, S_2, ..., S_n$ during a long period τ. The mean residual $\overline{R}$ then equals the average of residual times over the given period. That is, on a given time period τ the residual time $\overline{R}$ can be written as:

$$\overline{R} = \lim_{\tau\to\infty} \frac{1}{\tau}\int_0^\tau R(t)dt = \frac{1}{\tau}\sum_{k=0}^{n}\frac{1}{2}S_k^2 = \underbrace{\frac{n}{\tau}}_{\to\lambda} \times \underbrace{\frac{1}{n}\sum_{k=0}^{n}\frac{1}{2}S_k^2}_{\to\frac{1}{2}\overline{S^2}} = \frac{\lambda\overline{S^2}}{2} \tag{D.14}$$

In Formula D.14, $\overline{S^2}$ denotes the *second moment* of service delay. Applying this result to the Formula D.13, the queueing delay $\overline{W}$ can be rewritten as:

$$\overline{W} = \frac{\overline{R}}{1-\rho} = \frac{\lambda\overline{S^2}}{2(1-\rho)} \tag{D.15}$$

The Formula D.15 is known as *Pollaczek-Khinchin mean value formula* for the M/G/1 queue. The sojourn time $\overline{T}$ for a tagged customer can be now expressed as a sum of the service delay $\overline{S}$ required for serving the customer itself and the expected service time $\overline{W}$ for the other customers arrived before him:

$$\overline{T} = \overline{S} + \overline{W} = \mu^{-1} + \frac{\overline{R}}{1-\rho} = \mu^{-1} + \frac{\lambda\overline{S^2}}{2(1-\rho)} \tag{D.16}$$

This can be yet presented in another way by introducing a *squared variation coefficient* $C_v^2 = \sigma_S^2/\overline{S}^2$ as follows:

$$\overline{T} = \mu^{-1} + \frac{\lambda\overline{S^2}}{2(1-\rho)} = (1 + \frac{1+C_v^2}{2}\frac{\rho}{1-\rho})\mu^{-1}, \tag{D.17}$$

so that $\overline{S^2} = \sigma_S^2 + \overline{S}^2 = (1+C_v^2)\overline{S}^2$. The variation coefficient represents a normalized measure for the variability of a statistical distribution. When observing the coefficient C_v^2 at different values in range 0..1, it can be seen that all averages increase as the variance grows higher, implying the increased system stochasticity carries higher mean queuing times. In particular, with values $C_v^2 = 0$ and $C_v^2 = 1$ the Formula D.17 gives rise to deterministic M/D/1 and Poissonian M/M/1 processes, respectively. Values $C_v^2 >> 1$ are possible for especially bursty traffic processes.

Priority queues

Priority queues can be seen as a special type of an M/G/1 queue. The traffic is categorized into different classes $k = 1, 2..., K$, the class 1 considered as having the highest priority, and the class K the lowest. Higher the priority class, the better delay properties the corresponding traffic class is deemed to have.

The residual time for priority queues can be presented in analogous way to the Pollaczek-Khinchin formulas for M/G/1 queues. Denoting the second moment of the service delay for a traffic class k by $\overline{S_k^2}$, the expected residual time for processing messages in the queue arrived before the tagged message can be written as follows:

$$\overline{R_k} = \frac{1}{2}\sum_{k=1}^{K} \lambda_k \overline{S_k^2} = \frac{\lambda_1\overline{S_1^2} + ... + \lambda_K\overline{S_K^2}}{2} \tag{D.18}$$

Priority queues can be further categorized into *non-preemptive* and *preemptive*.

Non-preemptive priority-based queues

In non-preemptive priority queues, serving the current customer is finished despite of incoming higher priority traffic. The highest prioritized customer in the queue is served immediately after the current customer. Therefore, the traffic priority classes form logically several separate priority queues, of which the highest prioritized non-empty queue is served next.

For the first traffic class, the queueing delay can be written as:

$$\overline{W_1} = \frac{\overline{R}}{1-\rho_1} \tag{D.19}$$

The queuing delay for the second class can be written as:

$$\overline{W_2} = \overline{R} + \overline{S_1 N_{q1}} + \overline{S_2 N_{q2}} + \overline{S_1}\lambda_1 \overline{W_2} \Rightarrow W_2 = \frac{\overline{R} + \rho_1 \overline{W_1}}{1-\rho_1-\rho_2} \tag{D.20}$$

By substituting Formula D.19 to D.20, the latter can be re-written as:

$$\overline{W_2} = \frac{\overline{R}}{(1-\rho_1)(1-\rho_1-\rho_2)} \tag{D.21}$$

In the general case, the queueing delay for a class k can be presented:

$$\overline{W_k} = \frac{\overline{R}}{(1-\rho_1-...-\rho_{k-1})(1-\rho_1-...-\rho_k)} \tag{D.22}$$

$$= \frac{\frac{1}{2}\sum_{i=1}^{K}\lambda_i S_i^2}{(1-\rho_1-...-\rho_{k-1})(1-\rho_1-...-\rho_k)} \tag{D.23}$$

Eventually, the sojourn time T_k can be calculated as follows:

$$\overline{T_k} = \overline{S_k} + \overline{W_k} = \mu_k^{-1} + \frac{\frac{1}{2}\sum_{i=1}^{K}\lambda_i S_i^2}{(1-\rho_1-...-\rho_{k-1})(1-\rho_1-...-\rho_k)} \tag{D.24}$$

$$= \frac{(1-\rho_1-...-\rho_{k-1})(1-\rho_1-...-\rho_k)\mu_k^{-1} + \frac{1}{2}\sum_{i=1}^{K}\lambda_i S_i^2}{(1-\rho_1-...-\rho_{k-1})(1-\rho_1-...-\rho_k)} \tag{D.25}$$

Preemptive priority-based queues

Serving a low-priority message may be preempted by an incoming higher priority message, and resumed once the processing of the higher prioritized message has been completed. The lower priority traffic appears completely transparent to higher-priority classes. For a lower priority message k, the

expected service time for high-priority messages *1,...,k-1* arriving during the sojourn time $\overline{T_k}$ is also needed. It can be written as:

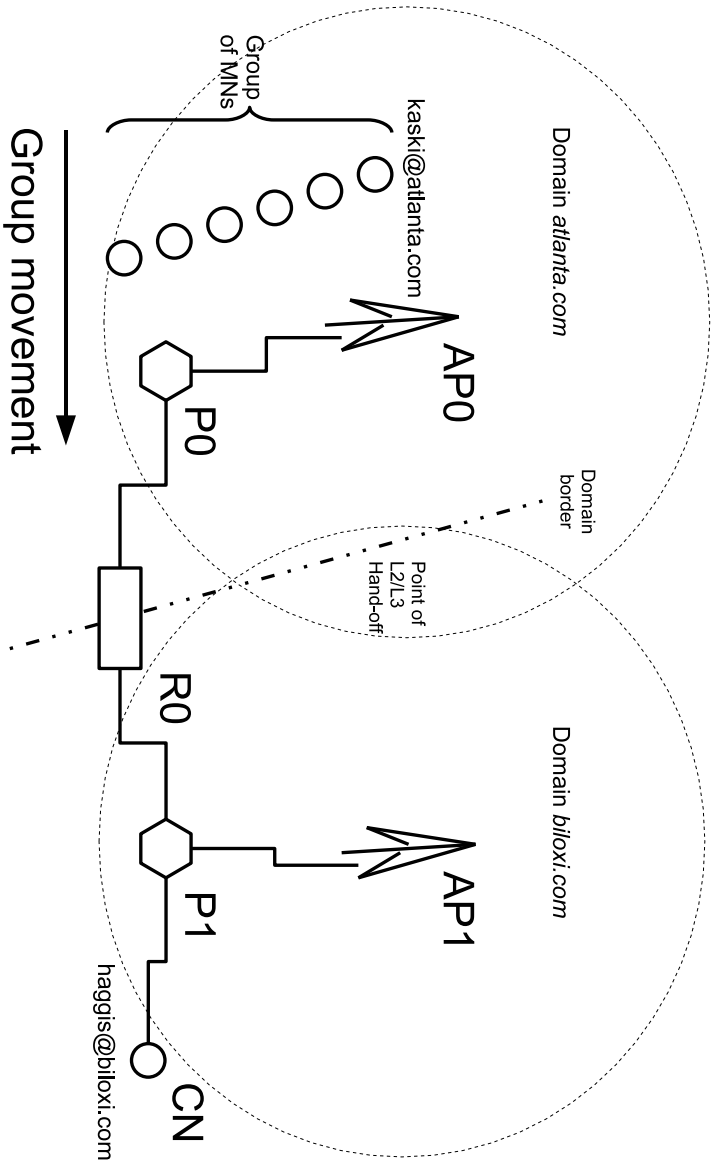
$$\sum_{i=1}^{k-1} \mu_i^{-1} \lambda_i \overline{T_k} = \sum_{i=1}^{k-1} \rho_i \overline{T_k} = \left(\sum_{i=1}^{k-1} \rho_i \right) \overline{T_k} \quad | \quad k > 1 \quad (0 \quad | \quad k = 1) \tag{D.26}$$

Generally, for each traffic class k, the mean sojourn time in the system is:

$$\begin{aligned}
\overline{T_k} &= \mu_k^{-1} + \frac{\overline{R_k}}{(1 - \rho_1 - ... - \rho_k)} + \left(\sum_{i=1}^{k-1} \rho_i \right) \overline{T_k} && \text{(D.27)} \\
\Rightarrow \quad & (1 - \sum_{i=1}^{k-1} \rho_i) \overline{T_k} = \mu_k^{-1} + \frac{\overline{R_k}}{(1 - \rho_1 - ... - \rho_k)} && \text{(D.28)} \\
\Rightarrow \quad & \overline{T_k} = \frac{(1 - \rho_1 - ... - \rho_k) \mu_k^{-1} + \overline{R_k}}{(1 - \rho_1 - ... - \rho_k)(1 - \sum_{i=1}^{k-1} \rho_i)} && \text{(D.29)} \\
= \quad & \frac{(1 - \rho_1 - ... - \rho_k) \mu_k^{-1} + \frac{1}{2} \sum_{i=1}^{k} \lambda_i S_i^2}{(1 - \rho_1 - ... - \rho_k)(1 - \rho_1 - ... - \rho_{k-1})}. && \text{(D.30)}
\end{aligned}$$

It must be noted that the condition $\sum_{k=1}^{K} \rho_k < 1$ must always hold for system to remain in equilibrium.

E Simulation topology



Authentication, Autherization & Accounting
Support for access control, policy enforcement and auditing.

Address-of-Record
Represents the long-term, device-independent identity of a service user.

Bi-casting
Using two sessions to send the same data during a hand-off to minimize the possibility of data loss.

Bindings
A set of *Address-of-Records* contained by a user location database.

Call context
A finite state machine maintaining information about the connection state.

Delay
A temporal break or a period of degraded service level that a process incurs.

Final response
A definitive response that finalizes and eliminates a particular transaction.

Flat routing
Every router has an entry towards every other router.

Forking
Multicasting a received INVITE request towards multiple destinations.

Group
A set of individual nodes exhibiting a remarkably degree of cohesion, inter-dependence and/or co-operation.

Group caching
The base station information obtained by scanning is stored into a database, that the whole group can avail when needed for L2 hand-offs.

Group hand-off
Performing hand-offs for a group in such way, that the amount of signaling is less than it would be if each node performed hand-offs separately.

Group mobility
Predicting the movement and forecasting the future need for channel resources for groups in place of individual nodes.

Hand-off
A process of transferring an ongoing data session from one AP/BS to another.

Hierarchical routing
Routing based on hierarchical topology, where clusters at lower layers have a representative at higher layers.

Informational response
See *Provisional response*.

Inter-domain
Happening over the borders of different network domains.

Intra-domain
Happening within a single network domain.

Jitter

Fluctuations in receiver-experienced packet arrival times.

Latency
See *Delay*.

Lawful interception
Policy-keeping or law-enforcing authorities have the possibility to transparently monitor and "eavesdrop" the specified network traffic.

Logical address
A network level identifier corresponding to a particular user group, applied in Hierarchical State Routing.

Logical subnet
A group consisting of network participants performing a particular task.

Macro mobility
Mobility occuring between network domains or Administrative Domains, so that network address renewal is needed.

Maximum Segment Size
The amount of data bytes that can be sent in a single, unfragmented packet. Data pieces larger than this need to be divided in multiple packets.

Media traffic
The payload user data; consisting of voice, video and data traffic.

Mid-call mobility
The hand-off is needed during an established session.

Micro mobility
Mobility occuring between subnets, within a single network domain.

Mission-critical
Refers to factors crucial for the success of the project, where an extreme level of reliability required.

Mobile SIP
The proposed extensions to make SIP support terminal mobility capabilities.

PAR
See *Predictive Address Reservation*.

Predictive Address Reservation
A process where a mobile node acquires a new network address proactively, before the actual link level hand-off occurs.

Pre-call mobility
Prior-to-call mobility, the mobile node updates its location to the home registrar while on move.

Provisional response
Intermediary responses used for indicating the progress of transactions.

QoS
Quality of Service. Guaranteeing a certain level of performance for a given data flow. Also, the overall user perceived service performance.

Reference point
A point of reference, related to which the group motion occurs.

Request
A message type initiated by a client to trigger some network functionality.

Response
A message initiated by a server as an answer or acknowledgment for an incoming *Request*.

Seamless hand-off
A hand-off which maintains the current session without degradation in the experienced *QoS*.

Seamless mobility
The ability to move between subnets or network domains without degradation in the experienced *QoS*.

Sequential forking
The most preferred contact address will be tried first, in decreasing order.

Service mobility
The user can access the same set of services with unaltered profiles, regardless of his/hers location in the network.

Serving BS
The AP/BS currently having the control of a session, releasing the connection during a hand-off.

Session mobility
The user may change network devices during a session, without the session being interrupted.

SigComp
Signaling compression aiming to improve signaling efficiency by removing the unnecessary redundancy.

Signaling
Using electronic signals to indicate or alter the state of communication, exchanging information relevant to the functions of the network.

Stateful proxy
A SIP network entity with an ability to maintain call contexts.

Stateless proxy
A SIP network entity without a functional transaction user layer; unable to maintain call contexts.

Target BS
A network which gains the control of a session as a result of a hand-off.

Terminal mobility
A network is able to maintain the established session, regardless of the node moving between neighboring networks.

Trapezoid routing
a typical phenomenon in SIP networks, the signaling and the payload traffic take different paths.

Triangular routing
The up-/downlink traffic is asymmetric; the incoming and outgoing traffic

take different paths.

Uniform Resource Identifier
A strings of characters, identifying uniquely a network user or a resource.

Universal mobility
Refers to the terminal, user, service and session mobility types together.

User mobility
The user is able to retain his/hers identity while moving around the network.

Virtual tunnel
A route between group participants known by a HSR clusterhead entity.