Multimedia Session Continuity in the IP Multimedia Subsystem: Investigation and Testbed Implementation

Prepared by: Keoikantse Orapeleng Armstrong Marungwana

Supervisor: Neco Ventura

This thesis is submitted in fulfilment of the academic requirements
for the degree of
Master of Science in Electrical Engineering
in the Faculty of Engineering and The Built Environment
University of Cape Town
February 2010

As the candidate’s supervisor, I have approved this dissertation for submission.

Name: Neco Ventura

Signed: ______________________ Date: ________________________

Declaration

I hereby declare that: (1) the above thesis is my own unaided work, both in conception and execution, and that apart from the normal guidance of my supervisor, I have received no assistance apart from that stated below; (2) except as stated below, neither the substance or any part of the thesis has been submitted in the past, or is being, or is to be submitted for a degree in the University or any other University.

I am now presenting the thesis for examination for the Degree of Master of Science (MSc) in Electrical Engineering.

Keoikantse O.A Marungwana February 2010

____________________ ________________________

Name Date

Abstract

The advent of Internet Protocol (IP) based rich multimedia services and applications has seen rapid growth and adoption in recent years, with an equally increasing user base. Voice over IP (VoIP) and IP Television (IPTV) are key examples of services that are blurring the lines between traditional stove-pipe approach network infrastructures. In these, each service required a different network technology to be provisioned, and could only be accessed through a specific end user equipment (UE) technology. The move towards an all-IP core network infrastructure and the proliferation of multi-capability multi-interface user devices has spurred a convergence trend characterized by access to services and applications through any network, any device and anywhere.

Innovative multimedia services comprising voice, video and data applications such as internet and web services can now be packaged as a single service subscription providing the user with increased convenience and an enriched user experience. They can be coupled with television or video streaming subscription and provisioned over IP-based networks and accessed through a multi-interface end user device. This converged environment, however, presents a number of challenges for terminal and session mobility management.

A key challenge faced by network and service providers, is to provide a context-aware and mobility-enabled service environment for ultimate user convenience. The ability to transfer ongoing communication sessions across access networks and devices, while maintaining session continuity and a consistent user experience, is an attractive feature for end users. To achieve this, the IP Multimedia Subsystem (IMS) plays a key role in this convergence of networks, services and devices. It is a standards-based service control architecture enabling the creation and rapid deployment of innovative IP-based and access-independent multimedia services.

A recent 3GPP standardization effort, IMS Service Continuity, specifies mechanisms to provide session continuity across packet switched and circuit switched networks, and between end user devices.

This thesis investigates the session continuity service and determines its effect on IMS user experience. To enable this, the work begins with a review of related standardization efforts, and highlights key challenges for session continuity. It presents a critical analysis of proposed solutions in literature. The analysis reveals that while substantial progress has been made on general session continuity, a clear misalignment exists between proposed approaches, and the standardized session continuity service. The thesis, therefore, plays a key role in carrying out a practical feasibility study of the standards-based session continuity service, using a real IMS testbed. The evaluation determines the additional signalling and delay overheads introduced by session continuity, and thus, its effect on IMS user experience. Particularly, the effect on session setup, session termination, user registration and session transfers are investigated. Different access technologies are used in the evaluations to determine the effect of heterogeneous access on session continuity.

The thesis, then, explores a possible evolution path of this service through the incorporation of rich presence information into session continuity. It exploits the role of service enablers in the IMS such as location based and presence aware services, which enable a variety of innovative and integrated services. The aim is to provide presence-aware session transfers for video on demand sessions. A scenario analysis of this enhancement is presented to establish the theoretical feasibility of a feature-rich session continuity service.

While session continuity is in essence not a new area of research, its practical feasibility and application in the recently 3GPP-specified IMS-based approach, is largely unexplored and untested. This thesis provides a novel contribution of a practical and proof of concept evaluation of this service, and forms an essential and timely point of departure for further research and development in this area.

Acknowledgements

I would like to acknowledge the following people for their assistance during the course of my research:

Neco Ventura, my supervisor. Thank you for the guidance and supervision during my research, and your inspirational dedication to the Centre of Excellence and the Communications Research Group

Richard Good, for your guidance and excellent proof reading assistance during the course of my research, and your contributions to CRG

Eugene Golovins, for the unique and inspirational discussions

Past and present CRG members for the discussions and feedback you provided during my presentations

Special thanks and dedications to my family for their support and encouragement during my studies

Table of Contents

List of Figures

List of Tables

Abbreviations

ACK	Acknowledgement
AS	Application Server
CAPEX	Capital Expenditure
GRUU	Globally Routable User Agent (UA) URI
HSS	Home Subscriber Server
iFC	Initial Filter Criteria
IETF	Internet Engineering Task Force
ICSCF	Interrogating Call Session Control Function
IP	Internet Protocol
IMS	IP Multimedia Subsystem
IPTV	Internet Protocol Television
IP CAN	Internet Protocol Connectivity Access Network
LAN	Local Area Network
MMSC	Multimedia Session Continuity
NGN	Next Generation Network
PCSCF	Proxy Call Session Control Function
RAN	Radio Access Network
RTSP	Real Time Streaming Protocol

SCTP	Stream Control Transfer Protocol
SCSCF	Serving Call Session Control Function
SIP	Session Initiation Protocol
SPT	Service Point Trigger
ST	Session Transfer
STI	Session Transfer Information
3G	Third Generation
3GPP	Third Generation Partnership Project
TCP	Transmission Control Protocol
TP	Trigger Point
UMTS	Universal Mobile Telecommunications System
UDP	User Datagram Protocol
UE	User Equipment
URI	Universal Resource Identifier
WLAN	Wireless Local Area Network
WiMAX	Worldwide Interoperability for Microwave Access
XML	Extensible Markup Language
XCAP	XML Configuration Access Protocol

Chapter 1 Introduction

Heterogeneous access networks, multi-interface and multimode user devices, and a plethora of multimedia rich and interactive services characterize the current era of communications. It is a trend of convergence between the telecommunications, broadcasting and internet worlds [2]. Technologically, it is the convergence of networks, the convergence of services and converged devices. Wired and wireless access networks are converging to an All-IP (Internet Protocol) based core network architecture supporting different access technologies, such as digital subscriber line (xDSL), the Worldwide Interoperability for Microwave Access (WiMAX), Wireless Local Area Network (WLAN) access, and 3GPP (3rd Generation Partnership Project) systems (GPRS/UMTS/HSPA, etc.). Operators can offer services to a wider user base with access to any of these technologies. Users, in turn, can access quad-play services comprising data, voice and video, coupled with mobility, for a rich multimedia communications experience. These include calls enriched with file sharing, enhanced phonebook with device capability information, presence information, and conversational messaging.

There is significant effort towards achieving rapid adoption and deployment of these services, while maintaining interoperability across different devices, and consistent quality of service across different networks [3], [4]. In addition, users are able to access these services using a single device with many capabilities such as voice and video calling, internet browsing, digital television (TV) and radio (FM – frequency modulation) tuning, file sharing, instant messaging (IM), etc. This converged environment presents a number of significant challenges for service developers, service providers and network providers. It requires interoperability of different devices, and technology-independent access to these services; users no longer need to own and/or operate different devices for different services on different networks.

This is supported by significant global efforts in the form of standards and specifications from bodies such as the 3rd Generation Partnership Project (3GPP), the Internet Engineering Task Force (IETF) and the Open Mobile Alliance (OMA). A particular reference standard from 3GPP is the Internet Protocol Multimedia Subsystem (IMS), which is defined as a

"global, access-independent and standard-based IP connectivity and service control architecture that enables various types of multimedia services to end-users using common

Internet-based protocols." [2]

It allows operators and service providers to offer subscribers secure access to IP-based, innovative multimedia services and applications while ensuring end-to-end (end-to-end) Quality of Service (QoS). It enables policy enforcement and control for efficient use of bearer resources, and enables the use of different charging models for these services. The access-independence allows users to access services through different access technologies while the circuit switched (CS) interworking capability allow users from both packet switched (PS) and circuit switched (CS) domains to access IMS-based services [5].

This rich environment of ubiquitous access and integrated services creates major challenges for mobility management, particularly the management and maintenance of ongoing communication sessions during terminal and session mobility events. Consider the scenario shown in Figure 1, where user A with a mobile user equipment UE-1, is involved in a conversational video session with user B on UE-2. The session has instant messaging chat features, file sharing, and presence information. User A then decides to invite a third user C on UE-3, into the conversation, thus establishing a video conference.

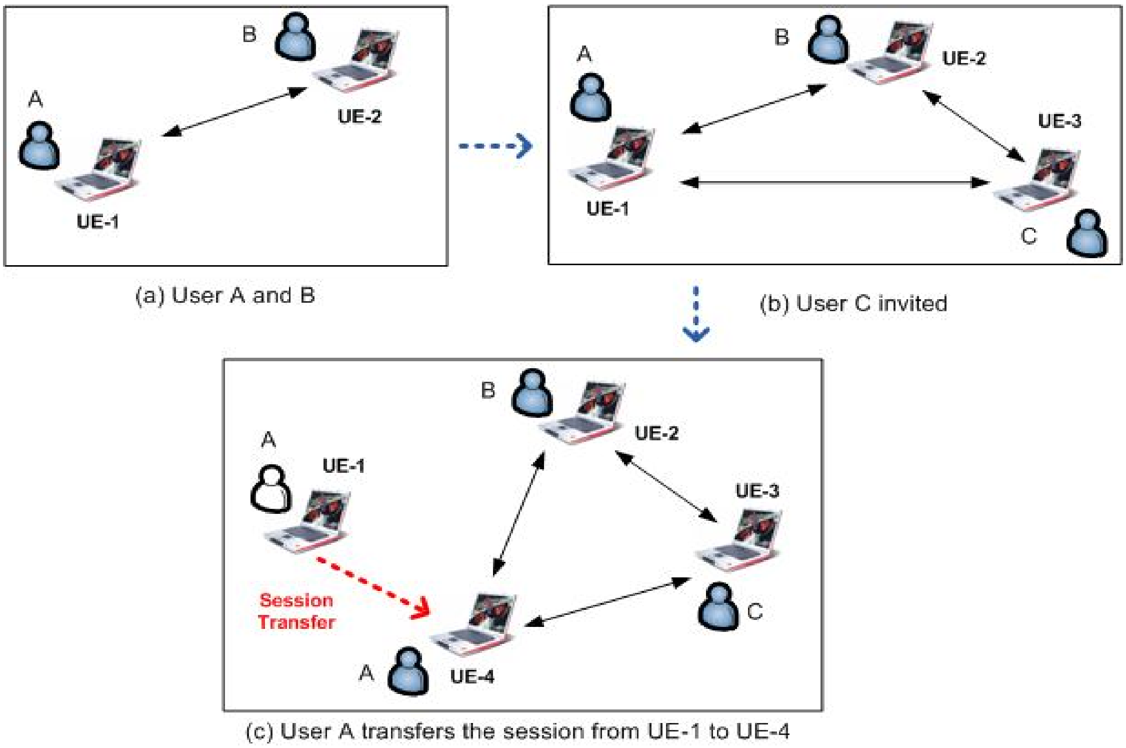


Figure 1: Sample scenario

All users are subscribed to different services from different providers and the UEs are attached to different IP Connectivity Access Networks (IP-CANs), e.g. WiMAX, WLAN, xDSL, WCDMA, etc. During the session, user A transfers the session from UE-1 to his other UE-4, offering higher bandwidth.

There are different ways to implement this. A simplistic approach is to terminate the session on UE-1, and re-establish it with all its media flows using UE-4. A preferable approach, however, is to perform a user-triggered, but automated and seamless transfer of the session from UE-1 to UE-4. The initial step is to capture session state information on UE-1; this is information, which describes the current state of the communication session, consisting of the media description and connection parameters so that it can be transferred to UE-4. UE-4 uses this information to initialize the relevant software application, essentially recreating the session state on the device, to resume the media exchange. UE-4 updates the remote ends of the conference session, UE-2 and UE-3, with new connectivity information so that media can be sent to the correct device.

The session-state signalling, device configuration, application initialization and media adaptation are some of the steps required to move a multimedia session across devices. The time taken to perform all these operations has a direct impact on how seamless the session transfer is, which affects the user experience. The main objective is to minimize the delay incurred or time taken to complete the session transfer and resume the session on the new device, and therefore minimize adverse effects on the user experience. This problem of session mobility continues to be a very active area of research.

1.1 Session Mobility

The scenario presented above highlights the complexities of moving a session across devices. Due to the different QoS requirements of different multimedia services and applications, the application layer in the protocol stack has largely been accepted as the most appropriate for hosting session transfer functionality for session mobility [6][7]. The motivation for this is the many documented drawbacks of lower layer Mobile IP approaches [8][9], which leaves the Session Initiation Protocol (SIP) [10] as the most suitable protocol for session mobility.

SIP is a simple, text-based application layer protocol for creating, modifying, and terminating multimedia sessions, and is transport layer independent. Although non SIP-based session mobility proposals exist [11], most have largely adopted the SIP protocol for mobility management, since it can handle the different mobility types of session, personal, terminal and service mobility [6].

The protocol uses a minimal number of specific commands, also referred to as SIP methods, (e.g. REGISTER, INVITE, PUBLISH, etc.), highlighting its simplicity. Furthermore, it is easily extensible and the commands can be overloaded by carrying different types of protocol data specific to different applications; e.g. Session Description Protocol (SDP) data to describe multimedia sessions. For example, the session state information mentioned in the previous section, which is necessary for session mobility, can be carried in the body of a SIP request sent to a receiving device. Hence, unlike other session mobility approaches [11], the same protocol used to establish a session is used to transfer session state information for mobility purposes. This indicates a very rich protocol, which can be used to establish any kind of multimedia session.

1.2 Session Continuity in the IMS

3GPP Release 8 specifications introduce the concept of Multimedia Session Continuity (MMSC) [12], [13] in the IMS. It is a mobility management service addressing three mobility scenarios. These are:

a) Continuity of multimedia sessions across packet-switched systems (PS-PS)
b) Session continuity between packet-switched and circuit-switched systems (PS-CS)
c) Inter-device session transfers, i.e. session continuity between devices of the same user – also referred to as session mobility.

In other words, multimedia sessions can be transferred across access networks when a UE changes Radio Access Network (RAN) connectivity, e.g. between WiMAX and E-UTRAN, and can be transferred across devices of the same user. The service is hosted and controlled by an application server (AS) residing in the services and applications layer of the IMS.

In this thesis, the terms IMS Session Continuity, IMS Service Continuity, Session

Mobility and Session Transfer are used as follows:

IMS Session Continuity – maintaining continuous media flow of an IMS session (voice, video, or data) during terminal or session mobility events.

IMS Service Continuity – the IMS service subscribed to by IMS users so they can maintain session continuity during mobility events.

Session mobility – the mobility of a communication session from one device to another without terminating the session.

Session transfer – the action of transferring a session's media and/or signalling flows from one device to another to effect or realize session mobility.

An important aspect of the IMS-based solution is the service-oriented approach, whereby the capability of moving sessions across networks and/or devices is a service to IMS users who explicitly subscribe to the service. In this way, only subscribed users have access to the service, and non-subscribers are unaffected. This service-oriented approach to mobility provisioning creates opportunities for innovative composite services creation whereby users can subscribe to various services such as location service, presence service, etc. which inter-work with the session continuity service to provide an enriched user experience.

In such a case, an important parameter becomes the handover delay between the source and target networks or the source and target devices. This includes mobility scenarios whereby single interface devices change their network connection between access points (terminal mobility), multi-interface devices switching between interfaces connected to different networks, and session handovers/transfers across devices. The challenge is to minimize the delay incurred during this transition period of horizontal or vertical handovers (between similar or different access technologies), and session handovers (between devices that may or may not use the same access technology).

The main selling point of the service-oriented approach to mobility provisioning is that it allows service providers to offer subscriptions to the session continuity service with defined QoS levels for session transfers. The service provider can guarantee different maximum session transfer delay thresholds for different subscription packages. For example, premium package subscribers can be offered the best levels of QoS whereby the transfer delay is guaranteed to be within the lowest threshold, while intermediate subscribers get a higher delay threshold, and the rest of the subscribers get a best effort mobility service.

1.3 Problem Statement

The service-oriented approach to session mobility, introduced by the IMS session continuity service inserts an extra entity (the session continuity server) between the end-to-end signalling paths of communicating parties. This allows the session to be anchored at the session continuity server, enabling it to control session transfers across devices. As the IMS session control layer involves extensive signalling [14], additional nodes in the end-to-end signalling path add more signalling interactions and thus increase the delays and overheads incurred during, for example, session establishment and session transfers.

Practical testbed implementations have investigated the performance of end-to-end IMS signalling for different scenarios. Some looked at signalling performance between the IMS core entities [15], as well as the suitability of different wireless technologies for IMS transport [16]. In [17] different access technologies are investigated for IMS sessions, with SIP-based terminal mobility across these networks. These works, however, only dealt with IMS signalling scenarios, which do not involve application servers in the signalling plane. Since session continuity uses an application server, the effect of including an additional service node in the signalling also has to be determined in quantifiable terms.

Work on IMS session continuity standardization is recent; there are thus limited reports of experimentation evaluating the performance and practical feasibility of this new service. Thus, to perform such an investigation, this thesis has to start with a clean slate implementation of a prototype IMS session continuity server. It is used in a testbed environment to determine, both qualitatively and quantitatively, the practical feasibility of the IMS session continuity service, and its effect on the end user experience.

IMS subscribers will access IMS services through different IP-CANs. IMS session continuity will thus be required to handle session transfer requests for session mobility across devices on different access networks. For this reason, the delay characteristics have to be investigated in a heterogeneous access networks environment.

1.4 Thesis Objectives

This thesis investigates the IMS-based session continuity service undergoing standardization in 3GPP, particularly, the effect this service has on end user experience. It discusses literature related to the topic, particularly the session mobility problem, whereby an ongoing communication session is transferred across devices of the same user. This is one of the envisaged scenarios of IMS session continuity [12].

The thesis aims to review IMS-based service continuity and efforts leading to its standardisation. It analysis related literature and how this has affected or influenced the developing specifications.

Primarily, the thesis aims to perform an investigation and practical evaluation of the session continuity service. It investigates its impact on user experience, specifically, the additional delay and end-to-end signalling overheads due to introduction of the session continuity server in the IMS. This is done through a standards-compliant implementation of the service on a real IMS testbed. The thesis aims to provide an indication of the practical feasibility of the service, in a heterogeneous access networks environment.

The IMS enables creation and deployment of innovative and integrated services. The thesis exploits this aspect by further proposing enhancements for the IMS session continuity service to provide users a feature-rich communications experience.

1.5 Scope of Research

There are many technical issues to consider during a session transfer, especially during an ongoing multimedia flow. These include the dropping of packets, buffering techniques during IP address changes, session adaptation for different network characteristics, and devices with different specifications and capabilities, e.g. screen resolution, memory capacity, and power consumption. These factors are discussed in the thesis, although not in equal detail, to present a broader view of the session continuity challenge. Solutions to these, however, are outside the scope of the thesis.

The thesis focuses on session continuity based on the recent 3GPP Release 8 specifications on IMS service continuity. It focuses on one of the envisioned session continuity scenarios whereby an ongoing multimedia communication session is transferred across devices of the same user. This scenario is also known as session mobility. The investigation and implementation performed focus on signalling aspects of the service, i.e. service layer delay metrics for registration, session setup, session transfer signalling, service triggering and anchoring. It is not a full implementation of the IMS standard on multimedia session continuity. Further, the implementation is proof of concept and thus only limited to achieving the specific objectives of the thesis.

A practical testbed implementation was chosen for its ability to produce near real-life evaluation results than a simulation environment. However, unlike a simulation, it has limitations on scalability testing due to the limited testing equipment. This metric thus lies outside the scope of the investigation.

1.6 Contributions

Significant effort has led to the finalization of the IMS core architecture. The remaining challenges include the development of services and applications that will take advantage of the many benefits of IMS. Central to this, are testbed implementations of innovative services and applications which are aligned and compliant with IMS standards. This ensures that different service providers and developers can rapidly develop and deploy services which are interoperable with other services, deployable on different platforms and equipment, and hence maintaining a consistent Quality of user Experience (QoE). Further, practical evaluations of recent standardization developments are critical to determine early results to form a point of departure for further investigations and establish quality benchmarks for potential service deployments.

The proof-of-concept testbed implementation in this thesis makes use of open source software tools such as the UCT IMS client [18], the FOKUS Open IMS Core [19], and the Sailfin open source application server [20]. The Open IMS Core and Sailfin application server are Java based implementations, while the UCT IMS client is based on the eXoSIP software package, which is a C programming language package. The aim is to achieve and prove interoperability and programming language independence of the implemented service.

The contributions of this thesis are documented in the following selected peer reviewed publications:

1. **K. Marungwana** and N. Ventura, “Performance Evaluation of IMS Session Continuity Signalling with Heterogeneous Access”, *IEEE Wireless Communication and Networks Conference (WCNC 2010),* Sydney, Australia, 18 – 21 April, 2010 (Accepted for publication)
2. **K. Marungwana**, L. Dikgole, and N. Ventura, “An Efficient Community-centric IPTV Deployment Model for Developing Regions”, *IEEE International Conference on Ultra Modern Telecommunications (ICUMT 2009),* St. Petersburg, Russia,12 – 14 October, 2009
3. **K. Marungwana**, and N. Ventura, “Presence and Real Time Streaming Protocol (RTSP) for Feature-rich Session Continuity in the IP Multimedia Subsystem (IMS)”, *Southern African Telecommunication Networks and Applications Conference (SATNAC 2009),* Royal Swazi Spa, Swaziland, 30 August – 02 September, 2009 **(Received Best Paper Award)**
4. **K. Marungwana**, and N. Ventura, “Multimedia Session Continuity in the IMS: Investigation and Testbed Implementation”, *Southern African Telecommunication Networks and Applications Conference (SATNAC 2008),* Wild Coast Sun, KZN, South Africa, 7 – 10 September, 2008
5. R. Good, T. Mvere, P. Wilson, **K. Marungwana** and N. Ventura, "An Evaluation of SIP Based Mobility in the IP Multimedia Subsystem," *Southern Africa Telecommunication Networks and Applications Conference (SATNAC 2008),* Wild Coast Sun, KZN, South Africa, 7 – 10 September 2008

1.7 Thesis Outline

The rest of the thesis is structured as follows:

Chapter 2 presents an overview of standardization work related to IMS session continuity. It discusses key developments leading to the finalization of the specification and outlines the architecture of the service within the IMS. Key challenges are discussed with reference to proposed solutions in literature. A critical analysis of literature with an IMS-based approach to session continuity is then presented, with a focus on key contributions and limitations of the proposed solutions.

Chapter 3 presents key service requirements for the service implementation of session continuity, and a detailed design and architecture of the service. A description of the service integration into an IMS testbed environment is also presented.

Chapter 4 proposes service enhancements to session continuity by incorporating the use of presence mechanisms and the Real Time Streaming Protocol (RTSP) for performing session transfers.

Chapter 5 presents the architecture and implementation of the evaluation platform, with a description of its objectives and limitations. It also describes the tools and equipment used for the evaluation platform.

Chapter 6 outlines the experiments performed and a discussion of the evaluation results obtained. Particularly, the chapter presents qualitative and quantitative indicators of the effect of session continuity signalling on IMS user experience, and determines the effect of heterogeneous access on the signalling interactions. It then presents a scenario-based impact analysis of the proposed presence-based service enhancement.

Chapter 7 presents conclusions drawn from the thesis and highlights key contributions. The chapter ends with recommendations for further study.

Chapter 2 Literature Review

This chapter presents a brief overview of standardization work on the IMS session continuity service. It discusses key developments leading to IMS service continuity, and outlines the main functional components of the service architecture. While the chapter does not attempt to present an extensive and exhaustive description of the standardization work around IMS service continuity, it gives the necessary background and context of the reviewed literature, and highlights the gaps in literature addressed in this thesis.

A discussion of related literature is presented focusing on different challenges associated with session continuity. Several works and approaches focusing on session continuity and IMS-based mobility are then critically analyzed, highlighting key contributions and limitations of the existing literature on this topic.

2.1 Efforts towards Standardisation

2.1.1 3GPP to WLAN Systems Interworking

The continuity of a communication session across different access domains or network technologies was made significant by the proliferation of IEEE 802.11 wireless networks (WLAN), which are now widely deployed and typically within cellular network coverage. Interworking architectures between WLAN and 3GPP systems thus gained great research interest [21], [22] due to the potential benefits cellular operators envisaged from WLAN users. Key issues in this environment include integrated billing, common authorization and authentication mechanisms, mobility and seamless roaming.

Song *et al.* present a comprehensive analysis of these challenges and overviews and contrasts some of the earlier specified architectures [23]. The interworking architectures take two approaches, namely tight-coupling and loose-coupling. The former uses the WLAN access network as a cellular RAN to reuse the cellular-based protocols and core network elements. The authors highlight the following disadvantages of this approach:

1. Exposing a 3G core network interface to WLAN which may be developed and deployed by independent operators
2. Large amounts of WLAN traffic going through the 3G core network, possibly causing a bottleneck
3. The requirement on each interworking WLAN network to have a 3G compatible protocol stack.

The loose-coupling approach connects the WLAN indirectly to the 3G network through an external IP network, such as the internet. This approach has less impact on both networks as they don't need direct interfaces between each other. However, there is relatively high signalling latency due to the longer path between the cellular core network and the user device, especially for MIP-based mobility management [23].

This approach is technically and operationally simpler to implement. However, the signalling latency severely impacts its ability to provide seamless service continuity across the networks, which is one of the 3GPP-envisaged interworking scenarios between 3GPP systems and non-3GPP based systems [24]. The scenarios are, in order of increased interworking:

a) Common billing and customer care
b) 3GPP system based access control and charging
c) Access to 3GPP system packet switched services by WLAN
d) Service continuity
e) Seamless service continuity
f) Access to 3GPP circuit switched services using WLAN.

Pinto *et al* present an interworking architecture that aims to handle the third, fourth and fifth interworking scenarios listed above [25]. It uses a 'core-level' loose-coupling approach and introduces two new components for controlling the interworked WLAN hotspots and managing PDP contexts for the hotspots. These components are architectural counterparts of the SGSN and GGSN in the 3G core network. They avoid extensive modifications to the 3G core network, and handle all WLAN traffic to avoid overloading the 3G network.

Service continuity is enabled by allowing the multi-interface terminal to simultaneously connect to both the interworked WLAN and the 3G network. The core-level interworking allows packets from services connected to the user's device to be sent to either of the device interfaces, which provides redundancy and enables quick handover.

2.1.2 Voice Call Continuity

The sixth interworking scenario of 3GPP and WLAN systems provides 3G circuit switched services using WLAN. A related standardization effort, Voice Call Continuity, was introduced in 3GPP Release 7 specifications [26]. It provides call continuity between circuit switched and packet switched systems and forms the architectural basis of IMS service continuity (in 3GPP Rel-8). It is overviewed here briefly to depict the foundation of IMS service continuity.

2.1.2.1 Service Description

Voice Call Continuity (VCC) is a subscription-based IMS service which provides capabilities to transfer ongoing voice sessions across 3GPP circuit switched (CS) domain and the IMS. A user device with an active voice call on the circuit switched domain can request a domain transfer of the call to IMS VoIP carried on WLAN without discontinuing the call. Figure 2 shows a basic VCC architectural diagram, adapted from [27].

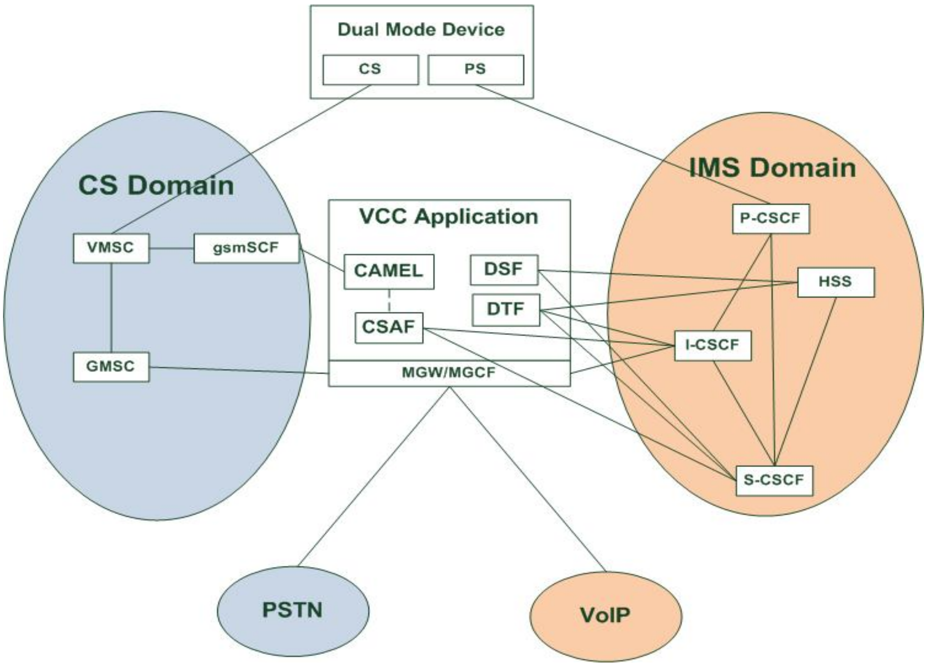


Figure 2: VCC Basic Architecture

A VCC application is introduced in the subscriber's home domain to anchor all calls from and destined to the VCC subscriber. Call anchoring is controlled by an operator policy, which by default, anchors all calls to and from subscribed users. This enables the calls to be transferred across the IMS and CS domains.

2.1.2.2 Functional Aspects

The VCC application has three main functions: the domain transfer function (DTF), domain selection function (DSF), and the circuit switched adaptation function (CSAF).

The DTF executes domain transfer procedures through third-party call control (3pcc) of established sessions, initiated by the user's equipment (UE). It also maintains the domain transfer policy, and provides charging data for the service.

The DSF, as the name suggest, provides policy-based domain selection for the subscriber's incoming and outgoing calls. It can use the operator policy, or the user preferences to make the selection, with the operator policy rules taking precedence over user preferences.

The CSAF acts as a proxy for the VCC subscriber's UE for circuit switched originated calls. It acts as a SIP User Agent (UA) to IMS for the user's equipment during CS calls, and provides call data between IMS and CS domain. It can interwork with the CAMEL service which deals with enforcing the CS redirection policy and routing CS calls to IMS.

2.1.2.3 Evaluation of Voice Call Continuity

Schmidt *et al* present a practical evaluation of VCC with a dual mode terminal connected to a 3G UMTS network and a WLAN (PS) network [27]. The authors investigate domain transfer performance between the packet switched and circuit switched domains during a voice call. The results show noticeable discontinuities of the voice stream during the domain transfer. The transfer delay, however, was found to be shorter for CS to PS transfer than for PS to CS.

The authors attribute the discontinuities to limitations in the prototype developed. However, no specific details of the implemented VCC prototype are provided, particularly, the domain transfer and selection functions. They are fundamental to a VCC application; hence, we argue that the associated signalling interactions during domain transfers would incur further delays and result in longer discontinuities than was experienced by the authors.

Voice Call Continuity, by definition, is designed for continuity of voice calls across circuit switched and IMS (packet switched) domains. This, however, limits its scope in the multimedia communications environment, which typically involves video media. The following section discusses the subsequent 3GPP release effort that succeeds voice call continuity, IMS service continuity, which provides multimedia call continuity.

2.2 3GPP IMS Service Continuity

The context of this thesis is within the IMS Service Continuity architecture specified by 3GPP. The service was introduced in the opening chapter. In this section, we present a more detailed, yet high level overview of IMS Service Continuity, giving a brief discussion of the service architecture, and depicting its development and relation to other standardization efforts.

2.2.1 Service Overview

IMS Service Continuity is an IMS based service which supports session transfers during terminal and session mobility events. Continuity of sessions is maintained between packet switched networks, and between packet switched and circuit switched networks. Figure 3 depicts the standardization efforts towards achieving IMS-based service continuity, and highlights relationships with other developing specification efforts.

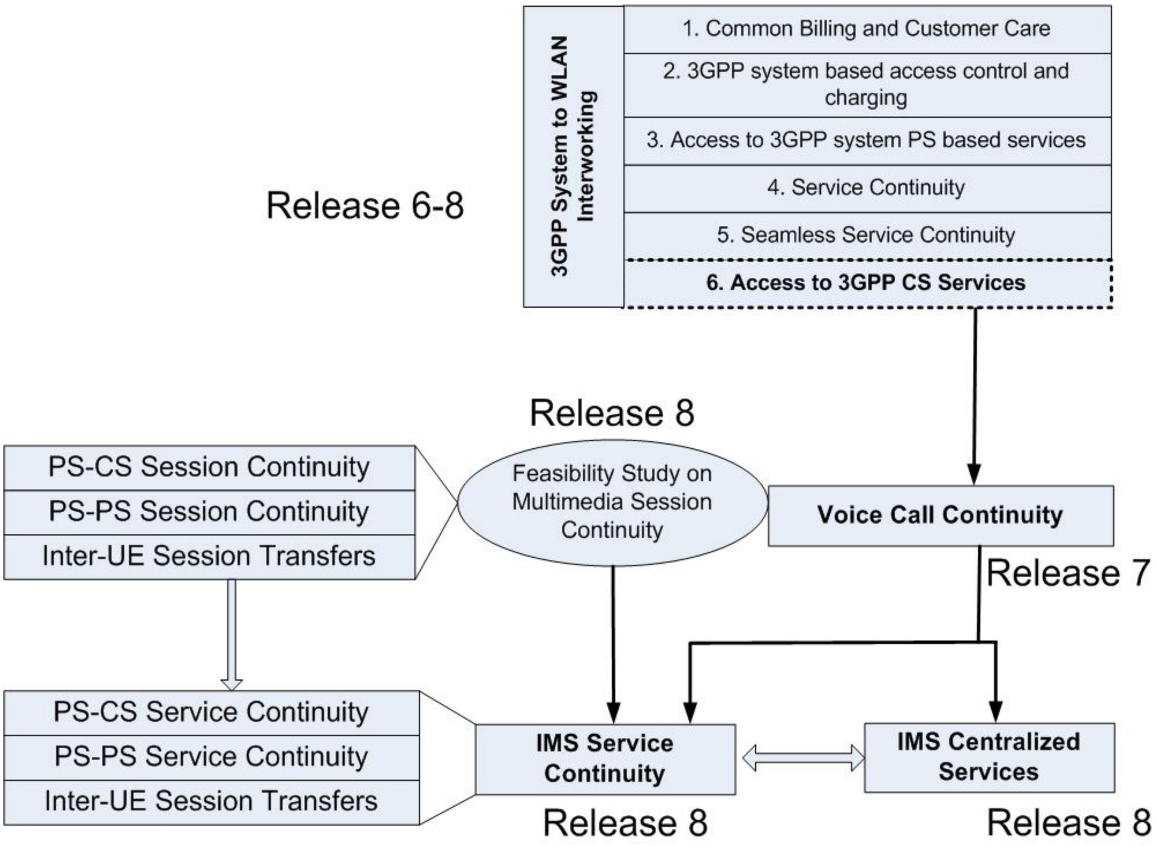


Figure 3: Towards IMS Service Continuity

The sixth scenario of 3GPP system to WLAN interworking aimed to provide access to circuit switched services using WLAN access. Since viable use case scenarios of this were not identified, it was removed from further specification work by 3GPP. However, interworking of CS calls with IMS-based (packet switched) VoIP was viable, and was thus the focus of VCC. The scope expanded to multimedia sessions with IMS Service Continuity, which interworks with IMS Centralized Services [28] to provide continuity of IMS services for subscribers using circuit switched bearers.

The three service continuity scenarios are discussed in the following subsection. From the feasibility study, a further increase in scope is highlighted by the general term of “Service Continuity” in the place of “Session Continuity”.

2.2.1.1 PS-CS Service Continuity

This functionality allows the continuity of IMS based services across the IMS (packet switched) domain and circuit switched bearers. The service combines mechanisms of IMS Service Continuity and IMS Centralized Services in a single IMS application server, referred to as the Service Centralization and Continuity (SCC) server.

The aim is to centralize control and execution of services in the IMS in order to provide a consistent user experience when using either packet switched or circuit switched bearers. Domain transfer between the two domains is based on the transfer procedures introduced in voice call continuity, and enhanced to include multimedia IMS sessions such as video communication.

2.2.1.2 PS-PS Service Continuity

This provides continuity of multimedia services across packet switched networks, such as WLAN and E-UTRAN, during mobility events. An ongoing multimedia session is transferred from one IP-CAN to the other, with either a full session transfer, or a partial session transfer. With full session transfer, the user’s UE transfers all media flows and signalling to the second IP-CAN. In a partial session transfer, the UE transfers only part of the multimedia session to the second IP-CAN. The remaining media continues on the original IP-CAN. This functionality requires the UE to have capabilities of simultaneous transmission and reception on both IP-CANs.

2.2.1.3 Inter-UE Session Transfers

Inter-device session transfers were discussed in the feasibility study of IMS session continuity. During IMS Service Continuity specification (Rel-8), however, the functionality was migrated to Release-9 specifications. The functionality refers to the transfer of multimedia session signalling paths, and/or associated media paths across devices of the same user, also known as session mobility. The UEs are under control of the same user, i.e. share the same IMS subscription.

2.2.2 Service and Architectural Requirements

While IMS service continuity is relatively a new standardization effort, which aims to perform session transfers across access networks and user devices, mechanisms operating at lower layers of the protocol stack have, for some time, already been in place. One of the key requirements of IMS service continuity, therefore, is to be able to function whether these mechanisms are deployed or not. The main protocol used for service continuity is SIP, whereas the dominant network layer mobility protocol is Mobile IP, and/or adaptations thereof. There are efforts that aim to interwork SIP and MIP by proposing hybrid SIP/MIP mobility solutions; these are, however, outside the scope of this thesis.

Service continuity is provided between 3GPP access systems and/or non-3GPP access systems, with real-time and non real-time services supported on either PS or CS domains. Another key requirement is to avoid impact on radio and transport layers of the PS core network, and on UEs which do not support this functionality. This allows ease of integration with already existing systems. For UEs which support service continuity, no new functionality should be required on the Remote UE – this is handled by the SCC (Service Centralization and Continuity) server using third party call control (3pcc).

2.3 Service Continuity Challenges

The introductory chapter discussed a brief list of key stages or steps involved when managing handover during mobility events. To recap, these include capturing the session state, also referred to as context information, transferring this context to the target device or target network, and re-establishing the same context in the target device or target network. Other aspects are media buffering and session adaptation to suit the target network and/or device characteristics. Proposals in literature can be grouped and analyzed according to these focus areas.

2.3.1 Context-Awareness for Session Continuity

The use of context information [29] is highlighted as the main aspect of the work in [30] for the handover decision. Information about the user's device such as capabilities, software, interfaces, the access network, and the user profile are regarded as static parameters of the context information, while the location information and the network's QoS parameters and traffic are dynamic information. Kaloxylos *et al* further extend the dynamic context information with CPU usage, bandwidth, loss rate, delay and jitter information, and the static context with user preferences and operator policy [30].

In a session mobility scenario, the context information from the source device has to be transferred to and reestablished in the target device. Cui *et al* [11] implement their own context transfer mechanism for transferring session state or context information. The authors implement a separate context exchange protocol, which is independent of the type of application running. Context transfer [31] aims to minimize handoff latency and packet loss by avoiding re-initiation of session setup signalling on the target device during handoff. Other proposals [32], [33] use middleboxes (non-IP-routing intermediary nodes between a source host and a destination host) to handle context transfers. In [34], the SIP Specific Event Notification framework is used to communicate the context information for handoff. This approach exploits the extensibility of SIP by defining its own event package, *contextAwareness*, which is then communicated through SUBSCRIBE and NOTIFY SIP requests.

2.3.2 Session Adaptation for Session Continuity

An important consideration for moving sessions across different networks or different devices is the difference in the network conditions or different device capabilities. For example, the target network for handover may have insufficient resources to support the QoS requirements of the incoming session. Alternatively, the device receiving the session may not have the capabilities to handle the incoming session, e.g. a high definition video session on an IPTV screen with a high bandwidth wired network connection, being transferred to a portable mobile device connected to a cellular network.

Bai *et al* demonstrate the effect of mobility on a streaming video session on a WLAN connection [35]. Although this does not include handover to another network, the illustration of the changing network conditions due to mobility events is sufficient to highlight the need for session adaptation. The authors show the degradation of media streaming performance indicated by queue overflows at the access points and packet losses due to excessive retransmissions during client mobility. What is required in this situation is a dynamic mobility-aware solution to dynamically adapt the media to suit the conditions of the wireless channel and/or the device capabilities.

A number of approaches exist for adaptation of video streaming to suit different channel conditions and device capabilities. These include precoding [36], whereby a streaming server has multiple copies of the content with differing characteristics such as format and bit rate. This is useful for session mobility whereby a high definition (HD) streaming session can be transferred to a lower capability device, albeit, at a lower quality and resolution.

Schwartz *et al.* present a comprehensive overview of another technique developed as an extension of the H.264/AVC coding standard, the Scalable Video Codec (SVC) [37]. The video bit stream is encoded into multiple layers. The video file consists of a base layer with the lowest resolution, frame rate and quality. The subsequent layers add spatial scalability (resolution), temporal scalability (frame rate), and quality scalability. To perform the adaptation, these ‘enhancement’ layers can be removed as required to match the capability of the receiving device or transmission channel. This allows low capability devices (small screens, low processing and/or battery power), and those with low bandwidth connections to still successfully be able to receive the stream, and enables high-end devices on high bandwidth links to view the full quality stream.

2.3.3 Policy-based Mobility Management for Session Continuity

This section discusses the role of policy-based mobility management and its relation and use in session continuity. Murray *et al.* define a policy based management system as one which operates by a set of rules and instructions triggered by different events. Policy rules are evaluated when a particular event from a specified set occurs [38]. The types of trigger events, and thus the evaluated rules, depend on the policy architecture being used. The policy could be a network based policy for managing network access control i.e. call admission control and resource allocation, or a device-based policy for selecting access networks to use for different services and applications.

These events include loss of network coverage, selection of a new preferred network by the user, or a trigger event to transfer a session between devices. For example, a number of factors can influence how a handover is performed. The simplest include whether the user has a subscription with the access network provider, and then whether the current device-in-use supports the access technology (e.g. loss of WiFi coverage prompting a move to 3G). Others include:

- Whether there is an ongoing session during the handover (mid-call session transfer)
- The type of session being transferred and its QoS requirements (voice, video, i.e. real-time or non real-time)
- Network load, i.e. the ability of the network to support these QoS requirements (jitter, delay, bandwidth)
- User preferences, e.g. to allow specific services on specific networks with certain device profiles etc.)

Some of these factors could be dynamic, e.g. changing based on time of day, or static and not changing frequently. The policy based approach has been proposed and applied in different areas in wireless networks. Murray *et al.* propose a network access control policy that performs network load balancing. It accepts service requests into either an EDGE or a UMTS network based on their residual capacity for the requested service types, e.g. voice, video, data [38]. While this is a network-based solution, the authors further propose a device-based handover policy which monitors current QoS during an active session, and makes a handover decision if it falls below a certain threshold.

2.4 Analysis of IMS Mobility Approaches

The previous sections presented standardization work behind the session continuity service, and an overview of its architecture. Topics involved in session continuity and related literature were discussed. This section presents a critical analysis of different approaches in literature to the IMS session continuity problem.

2.4.1 PMIPv6, MIPv6 and SIP mobility for IMS

Chiba *et al.* present an analysis of three mobility protocols for IMS networks: Proxy mobile IP v6 (PMIPv6), MIPv6, and SIP [39]. The authors evaluate the protocols using a WLAN and an emulated CDMA2000 network. The SIP solution uses the traditional approach of SIP re-INVITEs to re-establish the session after IP-CAN change, thus, it is found to incur the longest handoff delay. PMIPv6 and MIPv6 have virtually similar performance on both networks. PMIPv6, however, is preferred for localizing mobility functions by implementing a ‘mobile access gateways’ co-located with edge routers.

In summary, the authors contrast the network layer approach with the application layer approach to mobility management for IMS networks. While the results presented may offer some guidelines for consideration of the protocols, the SIP-based approach (albeit with improvements) has been adopted for session continuity in the IMS. The following sections discuss the different SIP-based approaches to IMS mobility.

2.4.2 Server-based Solutions

In this approach, an application server in the IMS is responsible for session continuity (or session mobility) between two IMS clients. This is architecturally similar to the 3GPP specified approach in [12].

Munasinghe *et al* present a WLAN to 3G networks interworking architecture that uses the IMS as an ‘arbitrator’ for interworking [40]. The aim is to provide WLAN-attached devices ‘the highest possible level of access to the UMTS services’. In this architecture, Figure 4, the WLAN network emulates a Serving GPRS Support Node (SGSN), and interacts with the 3G core network as a normal UMTS node. This enables mobility to be managed using normal UMTS mechanisms, and limits impact or changes, if any, on the UMTS network. The authors further claim reduced impact on the UMTS network since the Base Station Subsystem (BSS) functionalities are by-passed by using the emulator.

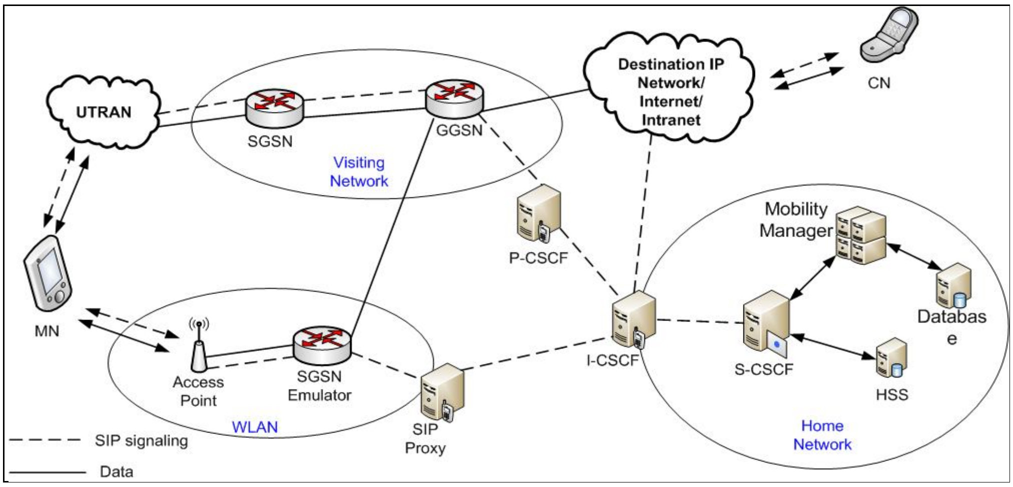


Figure 4: IMS-based Interworking (adapted from [40])

These benefits, however, while positive for limiting CAPEX implications for the UMTS network operator, place a technical burden on the WLAN operators who would interwork with the UMTS network. All the WLAN networks would require a UMTS protocol stack extension for the emulation. In addition, the seeming technical simplicity of emulation for the UMTS operator has to be weighed against the potential difficulty of establishing service level agreements (SLA) with each of the typically many different WLAN operators, whose networks will potentially interwork with the UMTS network.

The architecture includes a 'mobility manager' node attached to the IMS Serving Call Session Control Function (S-CSCF). This is architecturally equivalent to the session continuity server. While this approach is compliant with general service integration into IMS, more specifics on the operation of the mobility manager or simulations and practical evaluations of the architecture are not provided. These would further support the authors' assessment of the proposed architecture against the interworking scenarios specified in [24].

Mani and Crespi [41] present a similar approach, using a 'Mobility Server' in the IMS for session mobility. It is an application server and connects to the IMS using the normal ISC interface. The benefit of using the application server based approach, as mentioned by the authors, is that it can interwork with other services such as service discovery, and location-based services e.g. GPS (Global Positioning Services), to provide users with advanced combined services.

The authors also discuss session splitting and merging as an advanced feature of the mobility server. It splits the voice and video media of a session between two devices. Architecturally, the server uses the SIP REFER method and third party call control for managing the session transfers across the devices, which is the method followed in the specifications [12].

As with the previous article, the authors do not evaluate the proposed solution with either simulation, testbed implementation or other comparative analysis to test the claimed benefits of using the mobility server. Particularly, they do not evaluate or discuss the signalling and delay overheads introduced by the application server.

Bellavista *et al* present a more detailed service continuity solution, which includes handoff prediction and multimedia session adaptation [42]. The solution focuses on terminal mobility, i.e. a single device performing a vertical handover. It uses three components for handoff prediction, SIP signalling, and media adaptation. The handoff prediction occurs in the dual-interface client device to monitor wireless link conditions. A proactive activation of handoff to the second interface (thus network) is triggered before losing signal on the current active interface. An IMS application server performs the service continuity SIP signalling and interacts with a media gateway to adapt media according to the new network conditions. The testbed implementation results show an audio media adaption from a Wi-Fi connection to a Bluetooth connection.

The authors focused on handoff between WiFi and Bluetooth networks using audio media. Further investigation is thus still necessary for the more typical multimedia (voice, video and data) environment comprising WiFi, 3G and WiMAX networks. In addition, the target scenario is terminal mobility, using a combined vertical handoff and media adaption solution. In applying this approach to session mobility, however, the handoff prediction would be replaced by the more suitable device discovery phase, since the session handoff is not only across interfaces, but also across devices.

The concepts presented by the authors are encouraging and worth investigating further for the session mobility case. The results obtained, however, have limited relevance due to the scenario chosen, i.e. handoff from WiFi to Bluetooth. A more typical use case scenario involves multimedia communication sessions over larger networks such as WiFi, 3GPP-based (UMTS, HSPA, etc.), and WiMAX.

2.4.3 Proxy-based Solutions

In this approach, a signalling proxy performs an intermediary role between the client device and a server hosting a service, e.g. a multimedia streaming server. The primary aim is to avoid any extensive modifications on either the client side or network side for the functionality being implemented.

Lee *et al* propose a seamless service mobility solution using a signalling proxy between a client device and a streaming server [43]. The solution focuses on session continuity signalling in client-server streaming scenarios whereby an ongoing session is transferred between client devices based on the user’s mobility. The home gateway node in the user’s home network performs the proxy functionality. All signalling messages between the remote streaming server and the devices in the user’s home network pass through the proxy.

The mobility solution uses the Real Time Streaming Protocol (RTSP) for setting up and tearing down streaming sessions. In a typical scenario, the user’s client device (say UE-1) sets up a streaming session through the proxy, to the streaming server. The proxy records or caches all messages and the corresponding responses, and performs a mapping of its media ports connected to the user’s device and to the streaming server. When the user moves towards another device, UE-2, it sends a session description to the proxy. Since the proxy already has this information, it replies immediately without contacting the streaming server. UE-2 then requests the media setup, to which the proxy responds by simply updating the ports mapping to forward streaming media to UE-2.

The proxy responses eliminate the proxy-to-streaming server delay and thus reduce the signalling latency to minimise session discontinuity during mobility. The evaluation is not performed in an IMS environment; the concepts presented, however, can easily be adapted for such an environment.

Thanh *et al* propose another proxy-based solution for multimedia session continuity [44]. The authors focus on IMS-based networks and use the mobile Stream Control Transmission Protocol (mSCTP). An mSCTP-based proxy using multi-homing sits between a mobile node (MN) and an application server hosting a service (e.g. a multimedia-streaming server). Multi-homing enables a multi-interface terminal to connect to different access networks. mSCTP thus allows the MN to perform vertical handovers over the different interfaces. Since most terminals and application servers typically use TCP/UDP connections, the proxy splits the client-server connection into two TCP/UDP sessions, and creates an mSCTP tunnel between them, allowing it to exploit the multi-homing feature. The authors also propose content-adaptation capability for the proxy, such that media can be adapted according to the prevailing network conditions after handover.

The results obtained indicate a reduction in the number of signalling messages exchanged for handover. The authors, however, limit the evaluation to only this single metric. The proposed capabilities of the proxy such as QoS adaptation and mSCTP tunnelling are not thoroughly evaluated to determine their contribution to signalling latency. In addition, the mSCTP agent installed in the client device adds further processing to de-capsulate the TCP/UDP data within the mSCTP tunnel.

2.4.4 Full vs. Partial SIP-based Handover

In our previous work [17], we investigated two approaches to terminal mobility in the IMS. The first approach is the traditional SIP session re-negotiation in which, for vertical handover, ongoing sessions are terminated due to de-registration of the old interface, and subsequent registration of the new interface. The MN then sends a SIP re-INVITE to re-establish the session. We compared the handover delays across different access technologies, i.e. 3G HSDPA, 802.11g, Fast Ethernet LAN, and EDGE. The handovers were successful across all the networks, but not seamless since the handover is break-before-make.

The second approach uses a partial handover in which only the session media is transferred to the second interface and signalling remains on the initial interface. This avoids the typically long de-registration of the old interface and registration of the new interface. A SIP re-INVITE is sent from the initial interface, with media path parameters indicating the second interface. The handover delay is thus only comprised of re-INVITE signalling latency and media setup, without registration and deregistration latencies.

This approach, while not seamless, results in a better user experience compared to the first approach. The results highlight the benefits of decoupling IMS signalling paths from the media path. An example use case scenario is when a user with a dual interface device prefers to perform signalling on a more reliable, but otherwise costly network such as 3G, but prefers to carry media on a higher bandwidth and cheaper, albeit less reliable network such as WLAN.

The work focused on performance metrics of terminal mobility in IMS environment. The concepts evaluated, however, are strictly relevant to session continuity across devices, which is the focus of this thesis. For example, the session mobility counterpart scenario to the one above, would be to keep signalling on, say UE-1 (on UMTS), and transfer the media path to UE-2 (on WLAN); this is referred to as ‘Keep Control’ mode, the opposite of which is ‘Release Control’ mode [12].

2.5 Performance Evaluations of IMS Core

The previous section reviewed different approaches to session mobility in an IMS-based environment. The evaluations performed in the above works were based on an IMS testbed platform, the performance of which will inherently have an effect on the obtained results. This section, therefore, reviews performance evaluations of the IMS core platform using the basic IMS signalling scenarios.

Vingarzan and Weik [16] present an evaluation of IMS signalling using different access technologies. The reference IMS implementation is the Open IMS Core platform; it is tested on Fast-Ethernet LAN, Wireless LAN (802.11g), HSDPA, W-CDMA and GPRS networks. The authors aim to establish which of the networks are most suitable for carrying IMS services.

Typical IMS signalling use cases are evaluated on each of the access networks. These are IMS registration/deregistration, instant messaging, SIP session setup, and IMS session setup (SIP session setup with QoS reservation). The testbed results obtained provide a reference measure of pure IMS signalling performance in a heterogeneous networks environment.

It is, however, noteworthy that the authors implemented all the IMS core components on the same physical host. This limits the scope of the measurements to a single domain IMS environment, and localizes all signalling interactions between the IMS core components, i.e. the call session control functions (P/I/S-CSCF) and the HSS to a single machine. This eliminates relevant signalling delays and network traffic overheads that would occur between the IMS core elements. The IMS clients belong to the same IMS domain. This, in practical scenarios, will not always be the case.

Further investigation is, therefore, necessary for the more realistic and complex deployment scenario in which the IMS core components are in separate host machines, and the IMS clients are registered in different IMS domains. We argue that this scenario is more representative of potential use cases whereby IMS sessions are between users who are in different geographic (and administrative) locations. This scenario is investigated in the thesis.

Tang *et al.* also evaluate the performance of the IMS core testbed and determine its conformance with IMS specifications [15]. Basic IMS signalling scenarios are evaluated: registration, session setup responses (100 Trying and 180 Ringing) and session termination (disconnect). The authors intently deploy the IMS core components on separate machines and perform the measurements using two IMS subscribers. The results obtained provide insight into the raw performance of the IMS core testbed in a wired network environment, i.e. without background traffic.

While the above works evaluate the IMS core platform under ideal and simple scenarios, they provide useful data for the specific scenarios investigated, and sufficient reference for further testbed evaluations using more complex scenarios.

2.6 Chapter Discussion

This chapter has presented a comprehensive overview of standardization efforts around the IMS session continuity service, its relation to 3GPP-WLAN interworking, its evolution from voice call continuity, and relation to IMS centralized services. The chapter has outlined the main aims of the service, with a discussion of the main aspects and challenges of providing a seamless service continuity user experience. A critical analysis of proposals in literature focusing on these challenges was provided, highlighting key contributions and outlining shortcomings and misalignments in these proposals with the standardized service continuity architecture.

An important finding from the literature analysis is the lack of thorough investigative evaluations using practical testbeds to assess the performance of IMS service continuity. While some proposals present valid contributions to related aspects of IMS service continuity, the scope is often outside the IMS environment, and thus, omits critical issues which are necessary for IMS deployment. This, therefore, limits the relevance of obtained results, and further warrants the investigations performed in these thesis.

In addition, while the terminology used by many authors may often use ‘session continuity’ and/or ‘service continuity’, there is seldom direct relation or reference to the 3GPP standardized architecture of IMS Service Continuity. When similarities do occur, it is only by the use of an application server in the IMS, without much compliance with the service continuity specifications.

The following chapter presents the architecture and implementation of a standards-compliant session continuity service, and outlines key architectural service requirements. The architecture is based on the service continuity specifications, to enable a targeted investigation and evaluation of the standardized service.

Chapter 3 Design and Architecture of Session Continuity

The previous chapter outlined the development of standardization efforts towards session continuity, and highlighted key challenges and proposed solutions in literature. Particularly, it identified misalignments between proposed solutions and the 3GPP-specified approach to session continuity. This chapter, therefore, presents the implementation of a standards-compliant prototype session continuity service. It outlines service requirements of the architecture and details its implementation.

3.1 Architectural Service Requirements

This section presents key requirements for the service implementation presented in this chapter. These ensure that the service implementation meets the objectives set out in the opening chapter, and addresses the mismatches between previous solutions and implementations discussed in the literature review, and the standardized service architecture.

3.1.1 Compliance with the Specifications

An important consideration for the service implementation is to conform to the specified base architecture and the associated signalling behaviour. This close conformity with the specifications allows the evaluations performed to produce relevant and conclusive results on the service capabilities.

3.1.2 Scenario-based Session Continuity Subscription

Session continuity comprises a number of different scenarios depending on the access domain the user is in, which may be either packet-switched or circuit-switched domains. Further, whether the user’s communication session comprises real-time and/or non real-time media, determines the session continuity scenario. The type of session continuity required by the user will thus depend on their subscription with the service provider, and the network access provider.

It should therefore be possible for a user to subscribe to only the specific session continuity capabilities that they require. For example, a user who only has one UE may only require session continuity between access networks, without inter-UE session continuity capability.

3.1.3 Service Extensibility

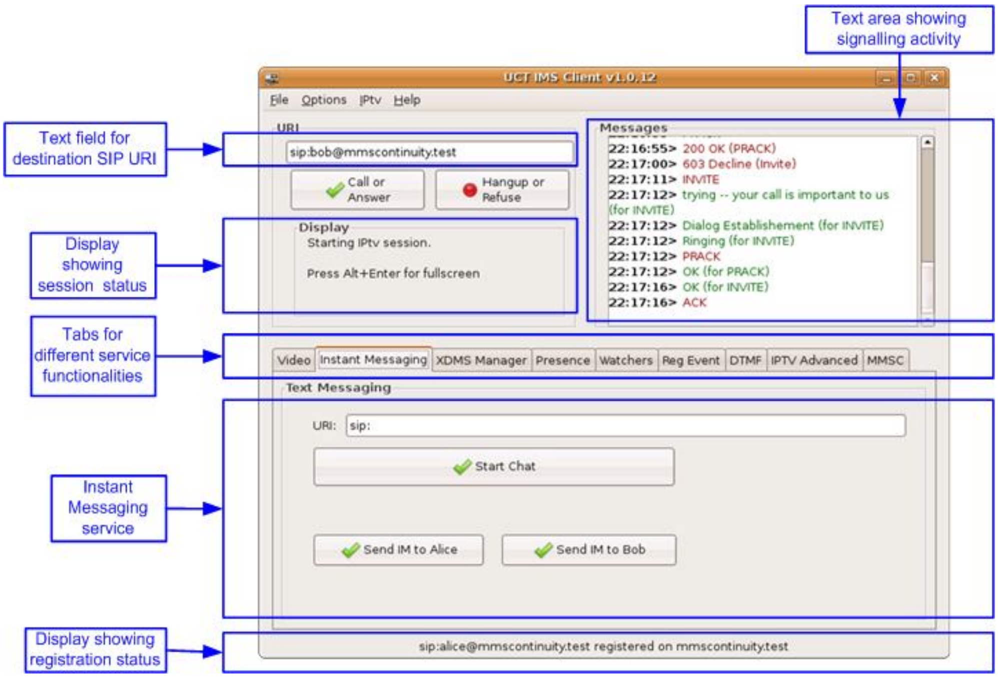
In relation to the previous requirement, a service provider who currently offers service continuity for only one of the scenarios may decide to extend the service offering. It should thus be possible to extend the service capability to cater for new service continuity scenarios based on user demand.

The following sections of the chapter detail the extensions performed on the UCT IMS Client for session continuity, including GUI capability and back-end signalling functionality. It outlines the functional requirements, specifications and implementation of the session continuity application server, with a description of its extensible modular architecture and the signalling functionality of the different use case scenarios. The chapter concludes with integration of the service with the IMS Core.

3.2 UCT IMS Client Functionality and Architecture

This section presents a brief overview of the UCT IMS Client architecture and functionality. A more detailed description of the client architecture is presented in [45].

The UCT IMS Client is an open source software client developed at the Communications Research Group, at the University of Cape Town. The client is part of ongoing IMS research around an IMS testbed environment based on the FOKUS Open IMS Core [19]. It is at the centre of a number of key research projects [46] in the research group focusing on IPTV, security, mobility management, and policy and QoS control. It has led to the development of a policy control framework [47], advanced IPTV and streaming servers, and a back-to-back user agent (b2bua). It is under continual development and improvement to ensure stability, feature update and bug fixes. Figure 5 below shows the UCT IMS Client with the Instant Messaging feature tab clicked.



URI: sip:bob@mmscontinuity.test

Display
Starting IPtv session.

Press Alt+Enter for fullscreen

Messages

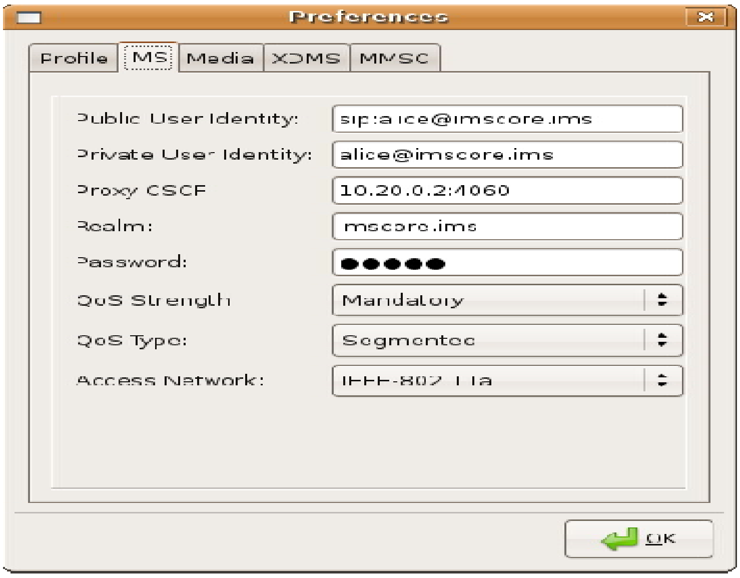
```
22:16:55> 200 OK (PRACK)
22:17:00> 603 Decline (Invite)
22:17:11> INVITE
22:17:12> trying -- your call is important to us
(for INVITE)
22:17:12> Dialog Establishement (for INVITE)
22:17:12> Ringing (for INVITE)
22:17:12> PRACK
22:17:12> OK (for PRACK)
22:17:16> OK (for INVITE)
22:17:16> ACK
```

Text Messaging
URI: sip:

sip:alice@mmscontinuity.test registered on mmscontinuity.test

Figure 5: UCT IMS Client GUI

The client supports full IMS SIP signalling which includes registrations (REGISTER), session initiation (INVITE), messaging (MESSAGE), subscriptions to events (SUBSCRIBE), notifications (NOTIFY), publishing of information (PUBLISH), and acknowledgements(ACK). The client's IMS preference configuration screen is shown in Figure 6.



Public User Identity: sip:alice@imscore.ims
Private User Identity: alice@imscore.ims
Proxy CSCF 10.20.0.2:4060
Realm: mscore.ims
Password: ●●●●●
QoS Strength Mandatory
QoS Type: Segmented
Access Network: IEEE-802.11a

Figure 6: IMS preferences configuration

The client supports MD5, AKAv1 and AKAv2 MD5 authentication schemes and supports voice and video sessions with different codecs. The client also supports both pager-mode and session-based instant messaging. Improvements through different theses studies resulted in functionality upgrades to incorporate presence functionality, IPTV and XML Configuration Access Protocol (XCAP) capability.

SIP signalling on the client is based on the eXoSIP software package [48], which extends the oSIP package [49]. This powerful API hides much of the complexities of creating and transmitting SIP messages, and can be easily extended for new SIP messages. The oRTP [50] software package is used for handling media, while the Graphical User Interface (GUI) design uses the GTK [51] software library. The client runs on the Linux operating system environment, and is released open source under the GPLv2 [52] licence.

It is simple to integrate new services into the client due to its modular design. It consists of four main classes and other supporting classes dealing with security, IPTV, presence, and other new service extensions. Figure 7 shows the interaction of the main classes, which are ***imsua.c***, ***ims_interface_event_handler.c***, ***ims_exosip_event_handler.c*** and ***callbacks.c***.

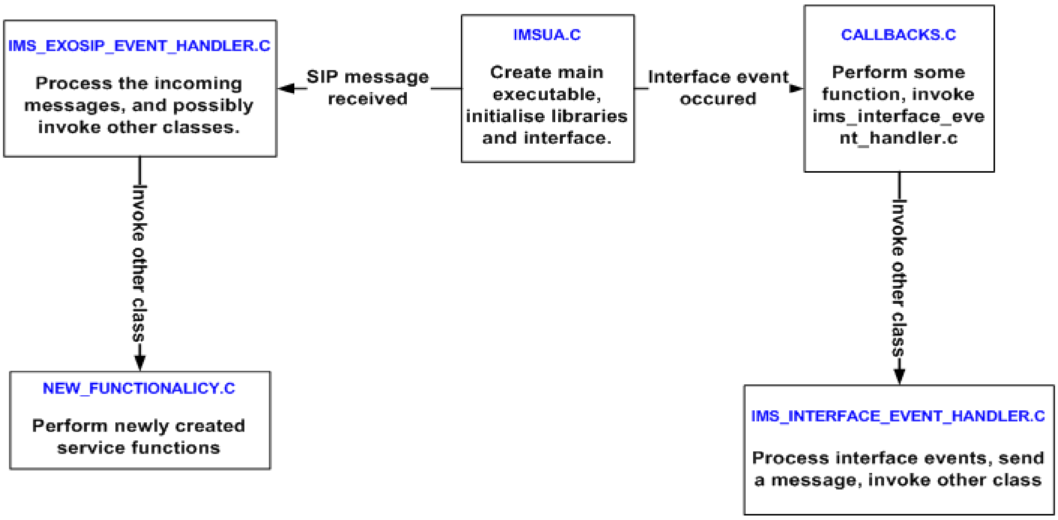


Figure 7: Interaction of core classes of the UCT IMS Client

The steps for creating a new service on the client are as follows [45]:

- Make a detailed design and specification of the new service. Decide on the required client functionality for the GUI interface, buttons, displays, text fields etc. Outline the signalling messages between the server hosting the service, the client, and the IMS core. Define appropriate SIP extensions if necessary.

- Write code for catching events in ***callbacks.c*** and invoke the relevant methods in ***ims_interface_event_handler.c*** to process the events and perform the required actions, e.g. sending out a SIP message.
- Add code in ***ims_interface_event_handler.c*** to process events, and invoke newly defined classes to perform new service functions.
- Add code in ***imsua.c*** to catch incoming messages and invoke relevant methods in ***ims_exosip_event_handler.c*** to process the messages.
- Add code in ***ims_exosip_event_handler.c*** to process the incoming messages, perform appropriate actions, possibly invoke other classes, and reply to the messages.

3.3 Session Continuity for UCT IMS Client

The UCT IMS Client was designed to be compliant with 3GPP IMS standards. In light of this, the following requirement is essential for new service capabilities for the IMS client:

- New functionality on the client must use standardized mechanisms to ensure interworking with other IMS nodes and services, by observing relevant IETF RFCs and 3GPP specification documentation regarding server and client signalling behaviour.

However, the above requirement does not imply standardization of functionality on the UCT IMS client. It merely ensures that the communication mechanisms between the client and other IMS entities and services conform to specifications. The following sub-sections outline the implemented high-level functionality and feature upgrades on the UCT IMS Client for session continuity.

3.3.1 UCT IMS Client upgrade – Graphical User Interface

The Graphical User Interface provides a front end for user interaction. It also provides information about call state, registration state, and shows the backend signalling – which makes it easier to determine if correct signalling is performed.

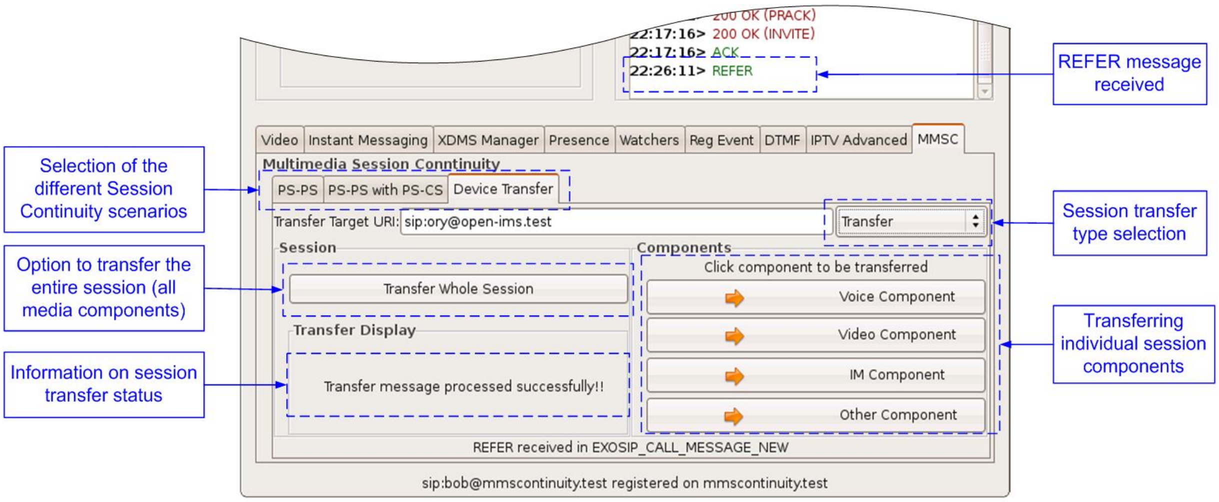
For session continuity, the modifications include:

- Session transfer notification – when a transfer request is sent and received
- Session transfer state – when a transfer is in progress, completed successfully, or failed.
- Alerting when incorrect session transfer attempts are made, e.g. trying to perform a session transfer when no session is in progress.

Figure 8 summarises the implemented GUI functionality for session continuity interaction, i.e. selecting scenarios, choosing media components to transfer, etc. The GUI provides:

- Ability to select the type of transfer scenario, i.e. selecting from the three envisioned session continuity scenarios (PS-PS, etc.)
- Ability to select the transfer type, i.e. media transfer, retrieval, addition, or removal.
- Selecting the specific media component to transfer/retrieve/add/or remove, i.e. voice, video, IM, or transferring the whole session.

These capabilities address one of the service requirements discussed earlier in the chapter. A user subscribed to only one of the session continuity scenarios can perform session transfers for that particular scenario, and be restricted to this functionality by the session transfer operator policy based on their subscription.



200 OK (PRACK)
22:17:16> 200 OK (INVITE)
22:17:16> ACK
22:26:11> REFER

Transfer Target URI: sip:ory@open-ims.test
Transfer

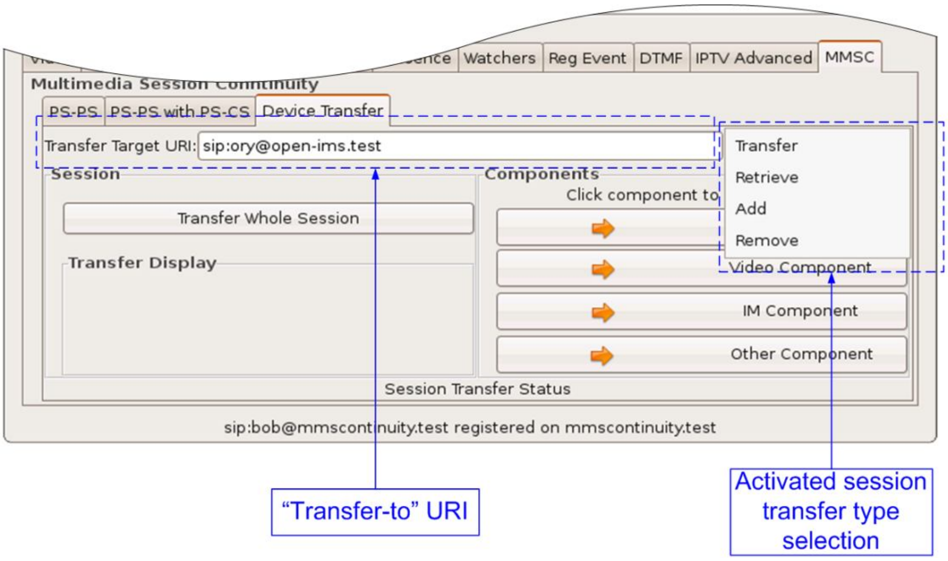
Transfer message processed successfully!!

REFER received in EXOSIP_CALL_MESSAGE_NEW

sip:bob@mmscontinuity.test registered on mmscontinuity.test

Figure 8: Session Continuity for UCT IMS Client GUI (a)

Figure 9 shows the activated options for transfer type selection, and the transfer target URI. The “Transfer-to” is a SIP header field that identifies the party to which the called party, (“Transferee”), is being referred [53].



Transfer Target URI: sip:ory@open-ims.test

sip:bob@mmscontinuity.test registered on mmscontinuity.test

Figure 9: Session Continuity for UCT IMS Client GUI (b)

To initiate a session transfer from one UE to another, there has to be a session in progress. The added functionality on the client checks this, and displays an appropriate message.

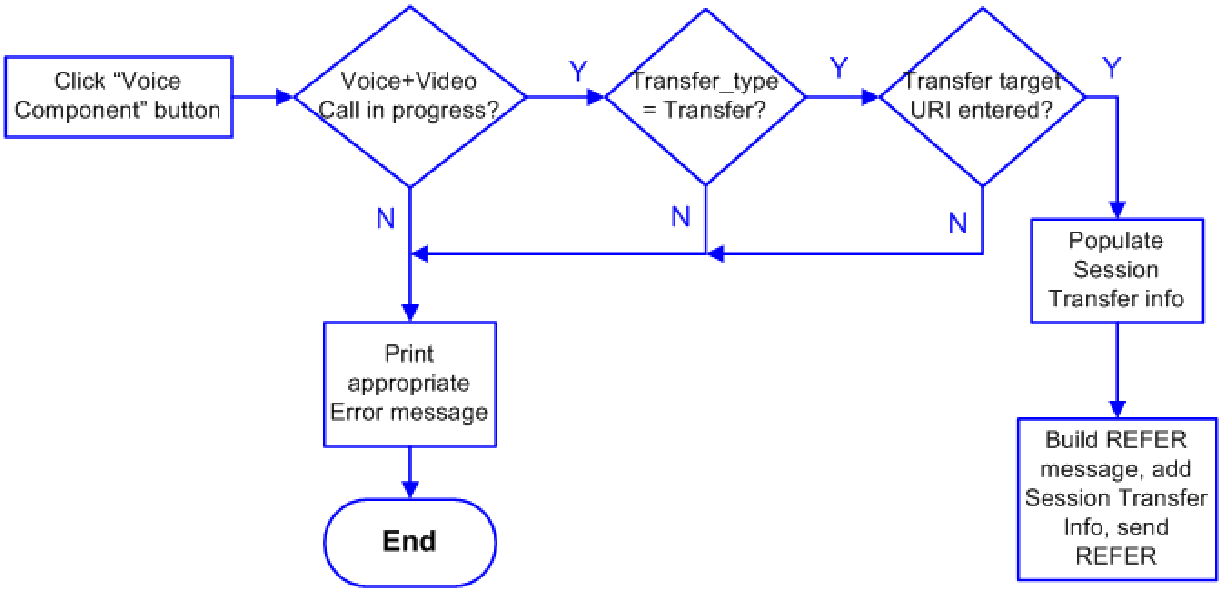


Figure 10: Session Transfer initiation sequence

The sequence of actions for performing a transfer is summarised in Figure 10. The steps for performing a 'retrieve/remove/add' follow a similar pattern.

3.3.2 UCT IMS Client – Session Continuity Signalling

The implementation of session continuity signalling uses the standardized SIP extensions approach. The SIP REFER method [53] is used to initiate and request a session transfer. There are two possible approaches of initiating a session transfer from a current device-in-use, to a second preferred device. The REFER may be sent directly to the second device, carrying the session transfer information, commanding it to request a transfer from the MMSC AS. Alternatively, the REFER may be sent to the MMSC AS, requesting a transfer of the multimedia session to the second preferred device. This is depicted in Figure 11.

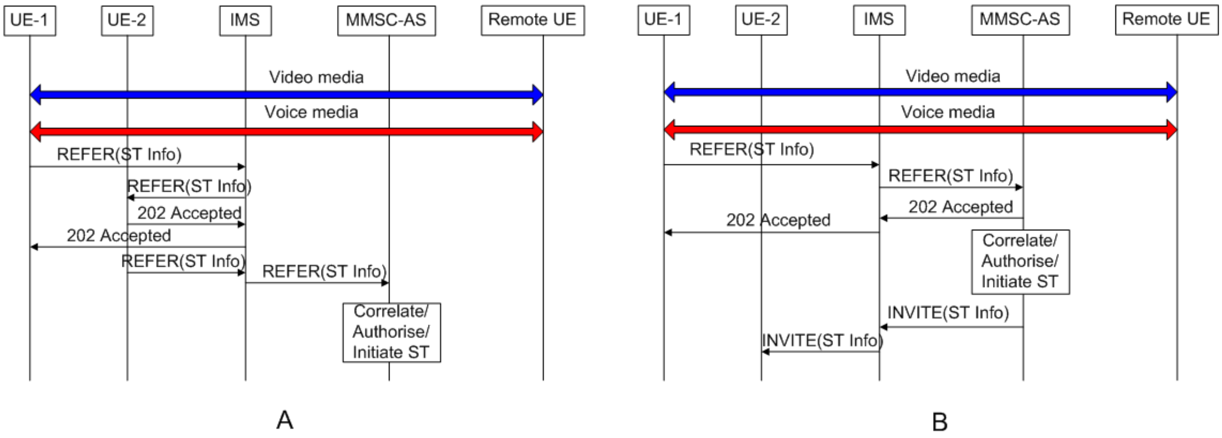


Figure 11: UE to UE vs. UE to MMSC AS Session Transfer Initiation

3.3.2.1 The SIP REFER Method

The method carries relevant session transfer information which includes session parameters such as session transfer identifier, session transfer type, media components to be transferred, connection information such as port numbers, and IP address or GRUU depending on client capabilities.

The method contains a special header `Refer-to` indicating the target of the session transfer. The REFER method is specified to implicitly create a subscription to the event `refer`. This forces the recipient of the REFER to send notifications NOTIFY regarding the status, i.e. outcome of the reference. For implementation, this requires further management of NOTIFY messages by the application server, specifying `Subscription-state` header information in the NOTIFY, determining and complying with the `refer` event subscription duration.

The implicit subscription, however, is not always required and may thus cause unnecessary additional signalling. A method to avoid implicit subscriptions is presented in [54]. In this thesis, this functionality is implemented as part of session continuity in the UCT IMS client. It suppresses the implicit registration to the `refer` event. It involves the inclusion of an extra SIP header, `Refer-Sub,` in the REFER message, which informs the recipient that the sender of the REFER message does not want to subscribe to the `refer` event, and to thus not receive NOTIFY messages. The value of the header is set to `false` to avoid the subscription and to `true` to enable the subscription.

Another extension specified in [54] is the `norefersub` option tag, used in the `Supported` header field of SIP requests. A UA includes this option tag in a dialog creating request or response as an indication that it supports suppression of the REFER implicit registration.

3.3.2.2 GRUU for Session Continuity

A user can register the same IMPU with the IMS from different UEs. For session continuity purposes, especially for session transfer between devices of the same user, it is required that the user can transfer a session from their current device-in-use, to a specific preferred registered device. The IMS client thus supports the Globally Routable User Agent (UA) URI (GRUU) mechanism [55]. This mechanism avoids the forking of a SIP message to all the user devices registered with the same IMPU. It identifies a specific IMPU-UE pair. For example, if a user has three devices registered with the same IMPU, a personal computer (PC), laptop, and an IPTV display, and they wish to transfer a VoD session from the laptop to the IPTV display, the session transfer cannot otherwise be sent specifically to the display without the message also being forked to the PC. The GRUU provides a mechanism to refer specifically to a particular registered device, in this case, the IPTV display. The IMPU-UE relationship is shown in Figure 12.

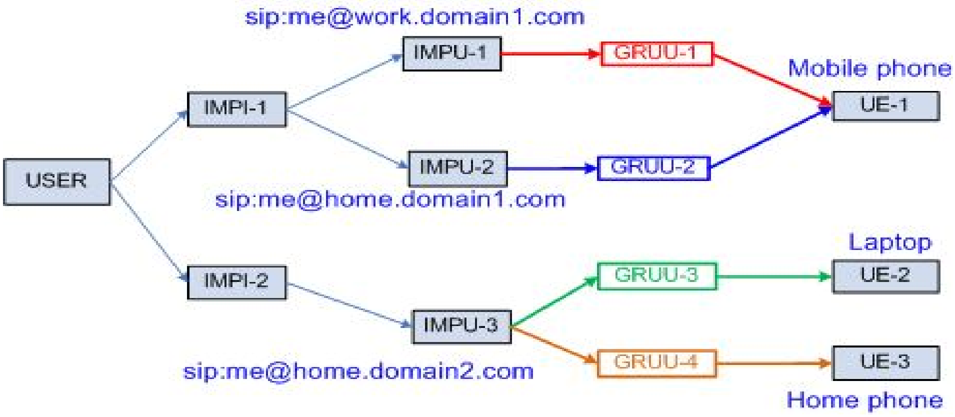


Figure 12: GRUU Relationship between IMPUs and UEs

3.3.2.3 The Session Transfer Information

The session transfer information (STI) identifies the ongoing session subject to session transfer. The specific format of the session transfer information is not specified in the standards. However, it is required to use standards-compliant mechanisms such as embedding in SIP headers, SDP parameters, and/or XML format.

In this work, the session transfer information is carried in the SIP message body in XML format. This format is also used in the IMS during user registration to communicate the user profile information of the user. Also, it is used by the presence service for communicating presence information. The contents of the STI depend on the session continuity scenario. For example, if the scenario is PS-PS session continuity, it would include the target packet-switched network to which the multimedia session is to be transferred. An example is shown below.

```
<scenario type=”ps-ps”>
     <current-network>wimax</current-network>
     <target-network>
          <type>wlan</type>
          <network-parameters>
               ...
          </network-parameters>
     </target-network>
</scenario>
```

In this implementation, the scenario of focus is session transfer between the UEs, the session transfer information (STI), thus, indicates the type of transfer operation being requested, e.g. transfer, retrieval, addition or removal of media. It is indicated by the `<transfertype>` tag; while the `<component>` tag indicates which media component (voice or video) is due for transfer. An example is shown in Figure 13.

```xml
<?xml version="1.0" encoding="ISO-8859-15"?>
<!DOCTYPE mstinfo SYSTEM "mst_info.dtd">
<!--DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
 "http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
  This is the Multimedia Session Transfer information contained in the REFER method -->
<mstinfo>
       <sessiontransferid>
              <cid>15631413519</cid>
              <fromtag>60062982</fromtag>
              <totag>701139848</totag>
       </sessiontransferid>
       <transfertype>"transfer"</transfertype>
       <component>video</component>
       <currenthost>
              <gruu>sip:ory@mmscontinuity.test;gr=urn:uuid:48e1f41e-5ec0-11de-9f6c-53a134cae679</gruu>
              <port>240015</port>
       </currenthost>
       <transfertarget>
              <gruu>sip sip:ory@mmscontinuity.test;gr=urn:uuid:51d1g92e-5ec0-11de-9f6c-53a134cae679</gruu>
              <port>244442</port>
              <regstatus>registered</regstatus>
       </transfertarget>
</mstinfo>
```

Figure 13: Session Transfer information carried in REFER message

The figure shows the body contents of the REFER message in XML format. Note the use of the GRUU extension to indicate a specific UE. A typical REFER message, excluding the content, is shown below.

```
REFER sip:alice@uct-ims.test SIP/2.0
Via: SIP/2.0/UDP ue1.uct-ims.test:5060;branch=z9hg4bk1105147607
From: <bob@uct-ims.test>;tag=2088335140
To: <alice@uct-ims.test>;tag=37516745
Call-ID: 1407251864@ue1.uct-ims.test
CSeq: 20 REFER
Contact: <sip:alice@uct-ims.test:5060>
Refer-to: <sip:ory@uct-ims.test;gr=urn:uuid:481ef41e-5ec0-11...>
Refer-Sub: false
Supported: norefersub
Content-Type: application/mstinfo+xml
Content-Length: 290
```

3.4 Session Continuity Application Server

This section presents the architecture and development of the session continuity application server. It highlights some of the main requirements discussed in [12], and shows how they guide the implementation. The tools used for the development are also discussed.

The IMS session continuity application server has an extensive set of functionality and capabilities. Some of these include splitting and re-merging of multimedia sessions which have components in both the circuit-switched domain and the packet-switched domains. The implementation performed in this work, however, is not an exhaustive and complete implementation of the server; it is only limited to the functionality required to perform the proof-of-concept evaluations of partial session continuity signalling.

3.4.1 Functional Requirements

The MMSC AS is the main functional element for IMS service continuity. It is responsible for accepting session transfer requests from UEs and executing the session transfer procedures based on the session transfer operator policy. The following are implementation-specific functional requirements for the application server:

- Host the session transfer operator policy of IMS session continuity subscribers
- Anchor multimedia sessions initiated by all session continuity subscribers for session continuity purposes
- Receive session transfer requests from IMS service continuity subscribers
- Authorize session transfers based on input factors such as the transfer policy and the session continuity scenario

3.4.2 Server Architecture

The IMS has a layered architecture which comprises the services and applications plane, the control plane, and user plane. Further, the services and applications plane can be regarded as a docking platform where new services and applications can be easily integrated into existing networks by simply adding a new application server, either logically or physically. This modularization of services and applications is applied in the design and implementation of the session continuity server in this work.

3.4.2.1 Architectural Layout

The application server has a modular architecture, where each function is implemented as a SIP servlet handling specific SIP requests. The modular design enables easy extension and upgrading of the server.

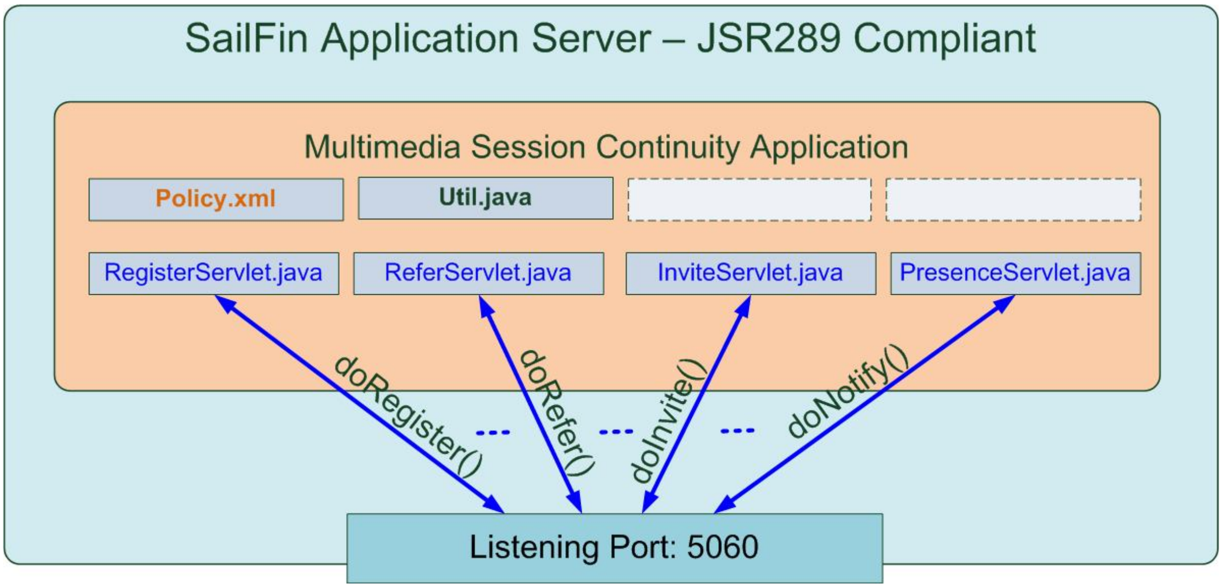


Figure 14: Server Architecture

As shown in the figure, the server implementation is a collection of SIP servlets, which handle specific SIP messages. The next section discusses the different servlets in more detail. SailFin is configured to listen on the native SIP port 5060. These and more configuration settings can be changed through an administrative console.

3.4.2.2 Server Modules

RegisterServlet.java – This servlet handles the third party registration message (REGISTER) received from the S-CSCF in the IMS Core. Third party registration is enabled by configuring the Initial Filter Criteria in the IMS to inform the session continuity server about user registrations.

InviteServlet.java – This servlet handles all invite messages received by the server. All the sessions of session continuity subscribers are anchored at the application server. The server receives the INVITE messages and extracts session information, to enable it to handle potential session transfer requests. The information is compiled into a session transfer information digest and saved for later retrieval. Specifically, the server captures information about who is initiating the call, who is the intended recipient, what are the session parameters, i.e. media types, contact addresses, etc. Further detail of the information captured is shown in Table I.

Table I: Session Information Extracted from INVITE Message

Information	Example
caller	sip:alice@uct-ims.test
callee	sip:bob@uct-ims.test
caller_state	registered
caller_subscription	active
media_audio	yes
media_video	yes
souce_address	10.5.0.2
source_address_gruu	*sip:alice@uct-ims.test;gr=urn:uuid:38511f70-70af-11de-9f6c-53a134cae679*
souce_audio_port	30200
source_video_port	34992
destination_address	10.200.0.8
destination_address_gruu	*sip:bob@uct-ims.test;gr=urn:uuid:51d1g92e-5ec0-11de-9f6c-53a134cae679*
destination_audio_port	16802
destination_video_port	16620

The information in blue text is known before receipt of the INVITE message. The captured information is also used to verify that the user has a session continuity subscription with the server. The server then checks if a session transfer policy exists for this user, in preparation for potential session transfer requests.

It should be noted that the information digest is built incrementally from the initial INVITE message, and subsequent requests and responses exchanged before the final 200 OK message is sent to the caller. For example, the initial INVITE from the caller provides the information about the caller's session parameters, and the subsequent responses provide information about the target's session parameters. The caller's session parameters may be updated as required by the SIP UPDATE message. This ensures that the server always has the latest session information.

ReferServlet.java – The ReferServlet handles session transfer requests delivered through SIP REFER messages. As discussed in previous sections, the message carries information regarding the type of transfer required, which is then evaluated by the server against the existing transfer policy.

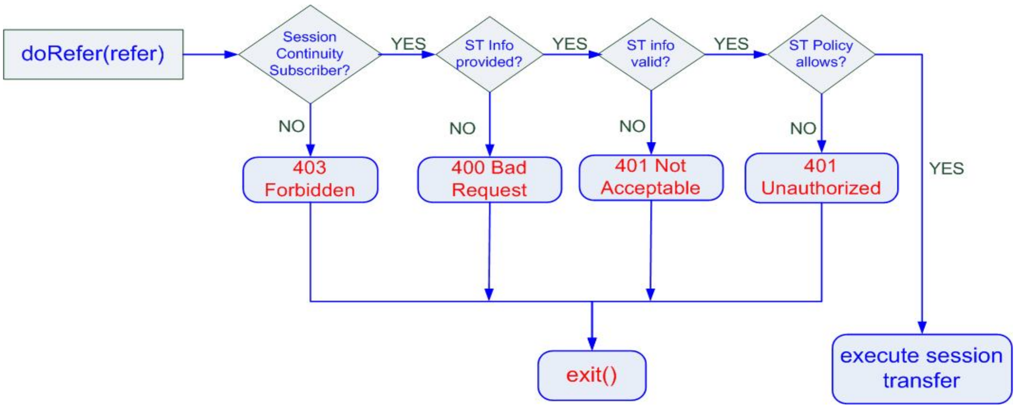


Figure 15: Processing the REFER message for Session Transfer

On receipt of a session transfer request, the server performs a user subscription check; one of the service requirements of session continuity is that users specifically subscribe to the service. Although the service subscription is configured in the IMS by iFC, a third party service provider offering IMS service continuity may still prefer to verify user subscriptions locally for added security. This functionality is therefore implemented by hosting a local user subscription database on the server.

The server then determines the required transfer type and session continuity scenario from the session transfer information contained in the message. If this information is not provided, the request is rejected with a SIP 400 status code. The server then exits and returns to idle. The next step is to check if the session transfer information provided correlates with the ongoing session belonging to this user. The ongoing multimedia session must be transferable according to the requested scenario and session transfer type. For example, the server will reject a transfer request for the video component of a session, if the ongoing session comprises only voice media.

A valid session transfer request is then evaluated against the session transfer operator policy. The policy provides a way for the operator to manage network resources, perform load balancing across network nodes, while providing acceptable QoS to users. Possible scenarios for session transfer policy decisions are listed below:

- A user wishes to transfer a session from a 3G network to a WiMAX network. Due to congestion in the WiMAX network, the operator rejects the user's request to transfer. The operator later updates the transfer policy when the WiMAX network load decreases. This information may then be communicated to (or explicitly requested by) the user's device, and then triggering a session transfer request.

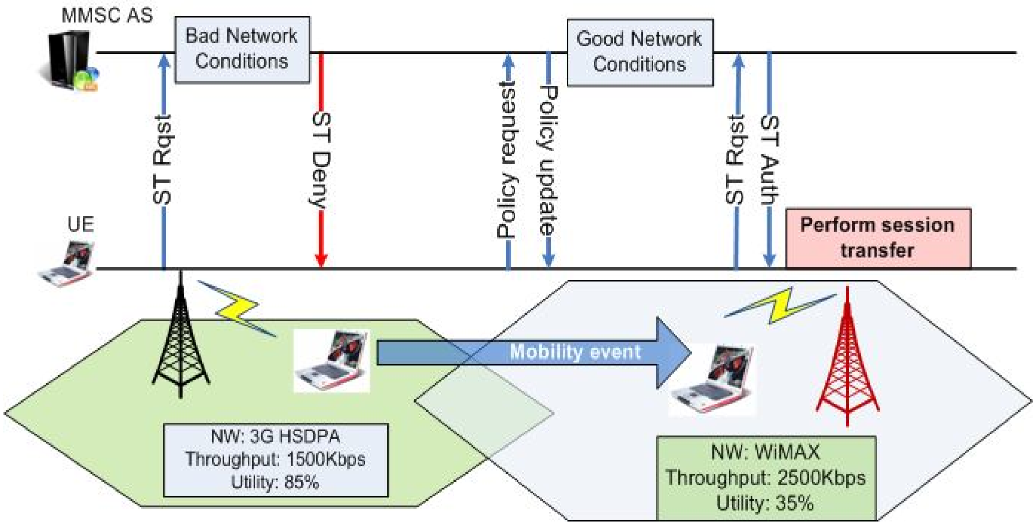


Figure 16: Transfer Policy Decision affected by Network Conditions

- Billing considerations may also affect the transfer policy decision. If a user's service continuity credit is depleted, they can no longer perform session transfers.
- Roaming agreements will also inform the session transfer policy, whereby users may be allowed to perform session transfers only within the service provider's administrative domain.

PresenceServlet.java – The presence module is included as a service enabler for enhanced session continuity. It allows the service to subscribe to a user's presence status, and perform presence-aware session initiation, termination and transfer. The purpose of this will be clearer in the following chapter.

3.4.3 Session Continuity Signalling

The application server interacts with the IMS client and core for session initiation, termination, third party registrations and can also perform subscriptions. The server is invoked based on user subscriptions to the session continuity service. The following sections detail the server's participation in the various session signalling scenarios.

3.4.3.1 Session Initiation

A typical IMS session initiation procedure involves the creation of a SIP INVITE message at the IMS client, sending it to the IMS core, evaluation of filter criteria, invocation of one or more application servers, and forwarding the INVITE towards the recipient. Figure 17 shows more detail on this, followed by further discussion of the steps in the procedure. The discussions focus on the activities of the session continuity server. Note that in the figure, the IMS core entities are grouped as one item, the IMS CORE.

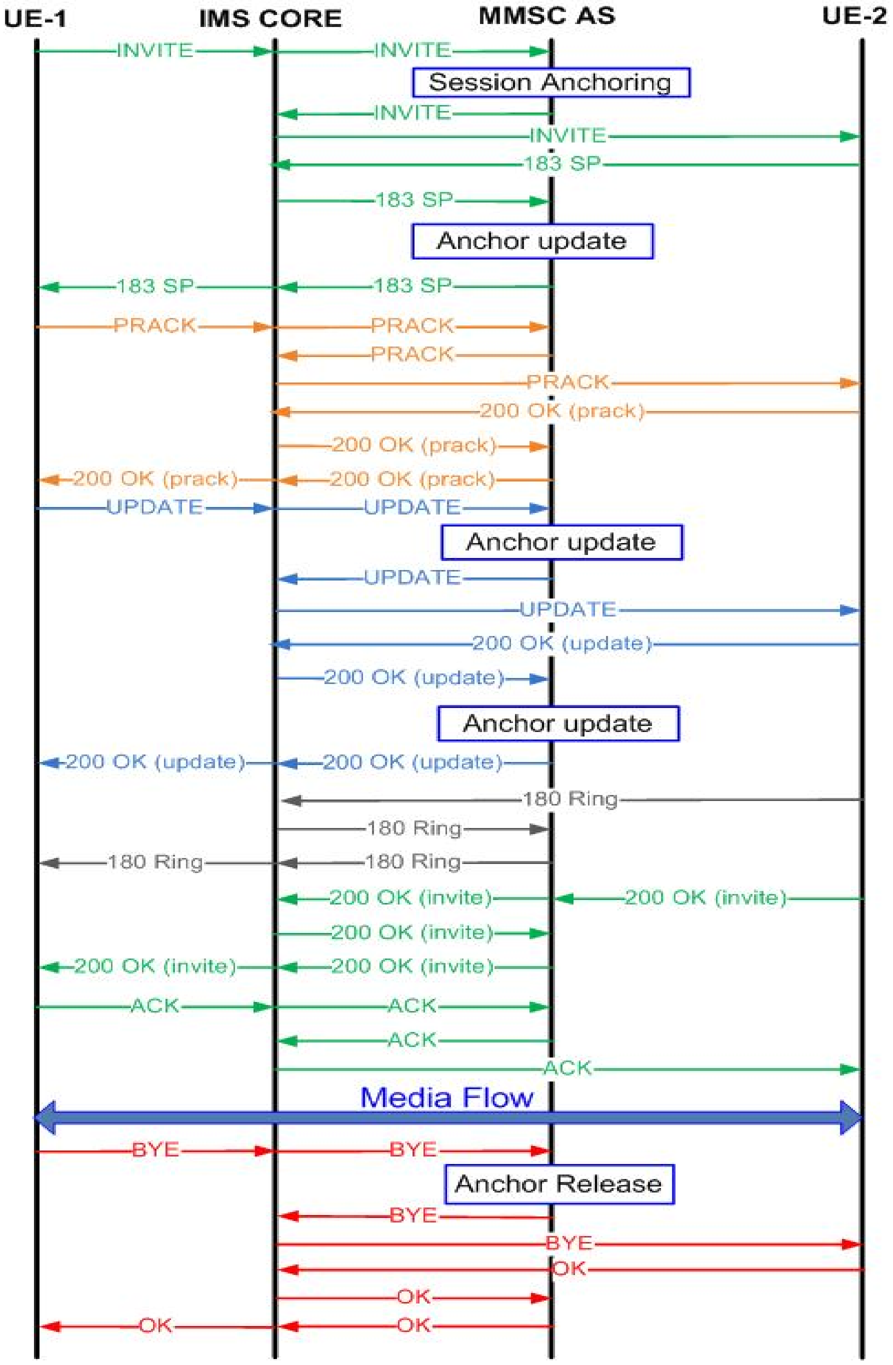


Figure 17: Server Invocation for Session Initiation

In the figure, UE-1 performs session initiation towards UE-2 by sending an INVITE message destined to UE-2. On receipt of the INVITE message, the S-CSCF forwards the INVITE message to the application server, and includes itself in the routing information, causing the server to return the INVITE message after processing. Initial filter criteria indicated the server. The server performs session anchoring, as shown in the figure, in which it extracts session information for the forward direction of this session. It then sends the INVITE message back to the S-CSCF so that normal request routing can proceed.

The server updates anchoring information during the interactions between the originating party and the receiving party. With full IMS session initiation signalling, the anchoring procedure is performed and updated each time session description information is exchanged, i.e. during the initial INVITE message, the subsequent 183 Session Progress, the UPDATE message, and its response if it contains SDP information. This ensures that the application server has up to date anchoring information when session setup is complete.

3.4.3.2 Session Termination

The application server is also involved in the signalling during session termination. When a BYE message is received from either the originating party, or the terminating party, the server clears all state information related to the ongoing session.

There are two approaches of handling a session termination request during an ongoing session. In the first approach, when a BYE message is received at the application server, the server sends a BYE message to the second leg, receives a final 200 OK response, and only then responds to the original BYE request from the 1st leg. This is shown in Figure 18.

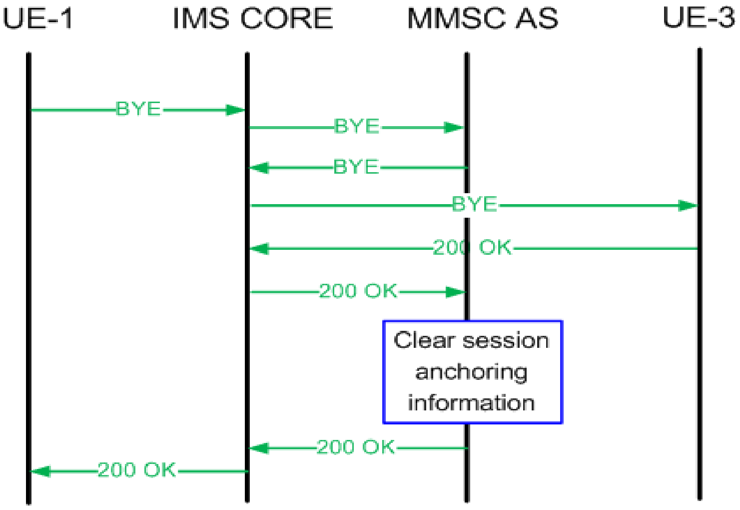


Figure 18: Session Termination

In this approach, the server awaits confirmation from the second leg of the communication session, before clearing session information.

In the second approach Figure 19, when the server receives a BYE messages from either end of the communication session, it immediately responds with a final 200 OK, and afterwards sends a BYE message to the second leg of the communication session. We call this approach, expedited session termination.

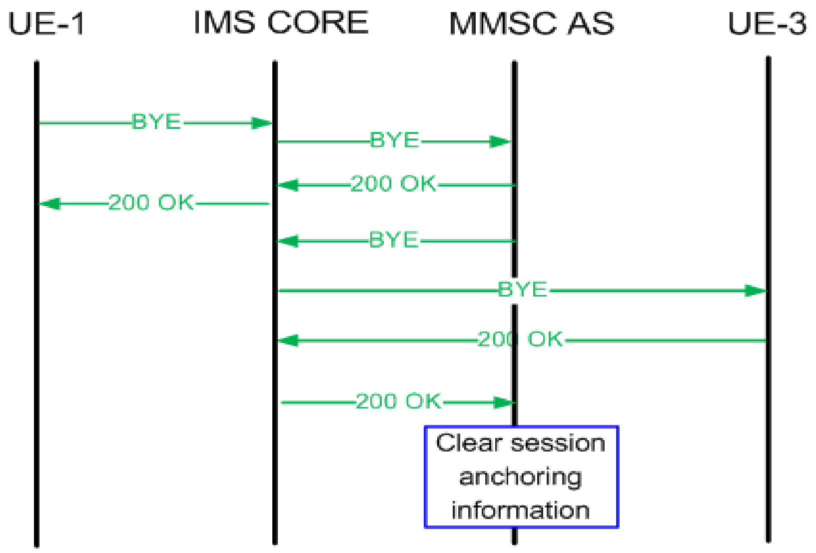


Figure 19: Expedited Session Termination

The server starts clearing session information after receiving a final session termination response from the second call leg.

3.4.3.3 Session Transfer

Session transfer is the main functionality of service continuity. Section 3.3.2 described the two approaches a UE can take to initiate a session transfer request. The options were to send the transfer request directly to the user's second device, commanding it to request the transfer from the session continuity server. The second approach sends the transfer request directly to the server, requesting it to initiate a session transfer to the second device using third party call control. This approach caters for devices which do not have sufficient capabilities to initiate session transfers directly. The former approach, depicted in more detail is shown in Figure 20.

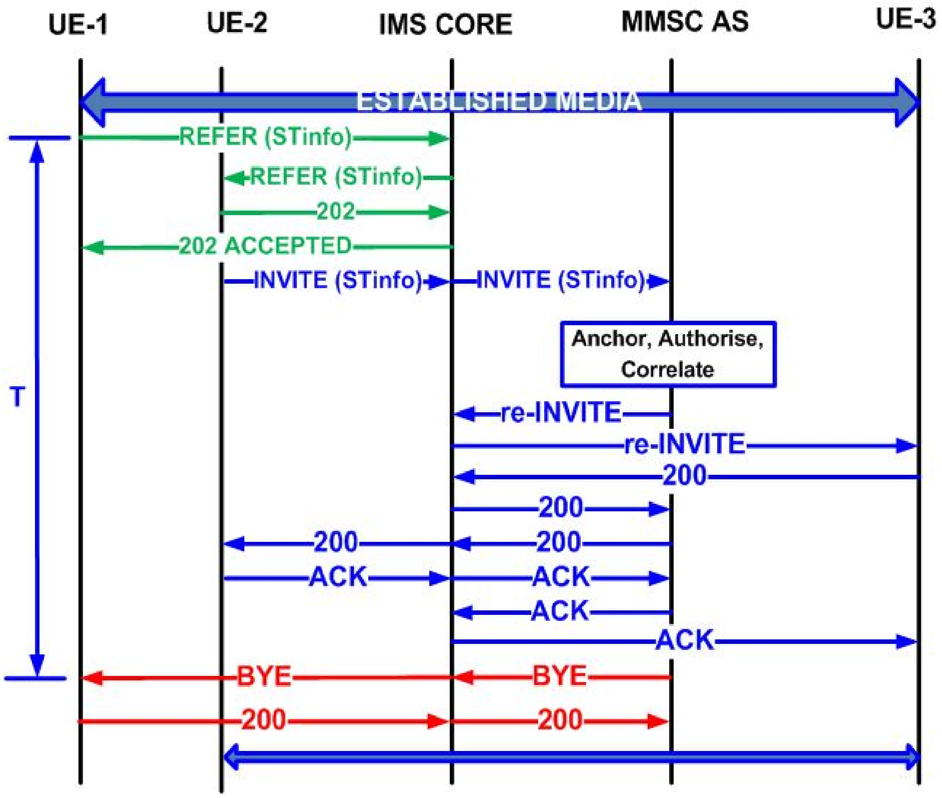


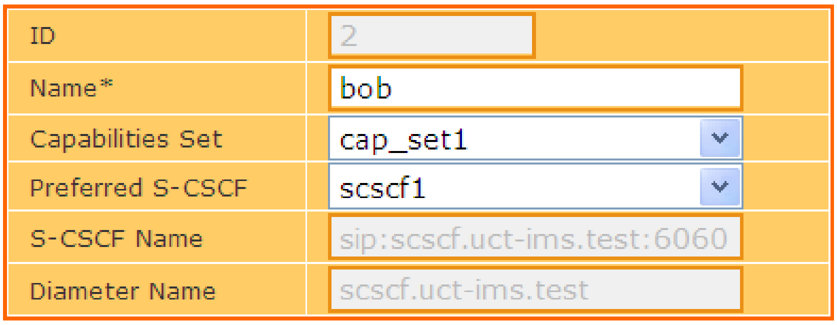
Figure 20: Session Transfer Signalling

3.5 Service Integration into IMS Core

This section outlines how the implemented session continuity service is integrated with the Open IMS Core. This includes the various steps of provisioning user profiles on the FHoSS, configuring filter criteria, service point triggers, and the details of the application server.

3.5.1 User Configuration

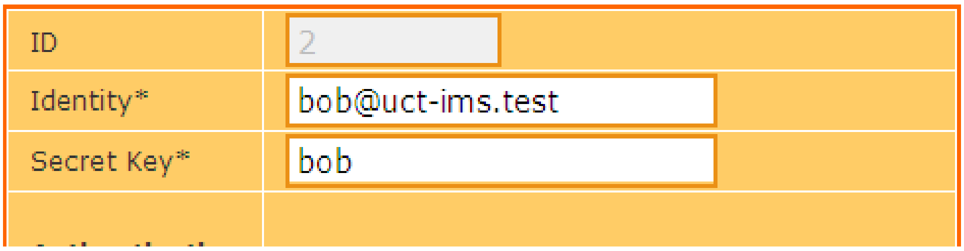
Each IMS session continuity subscriber is provided a user profile. The service and user profiles are configured on the FHoSS through a web user interface. This includes the user identities, services and IMS network configurations. The user identity information comprises the user subscription (IMSU) information, the public user identity (IMPU), and private identity (IMPI) details.



ID	2
Name*	bob
Capabilities Set	cap_set1
Preferred S-CSCF	scscf1
S-CSCF Name	sip:scscf.uct-ims.test:6060
Diameter Name	scscf.uct-ims.test

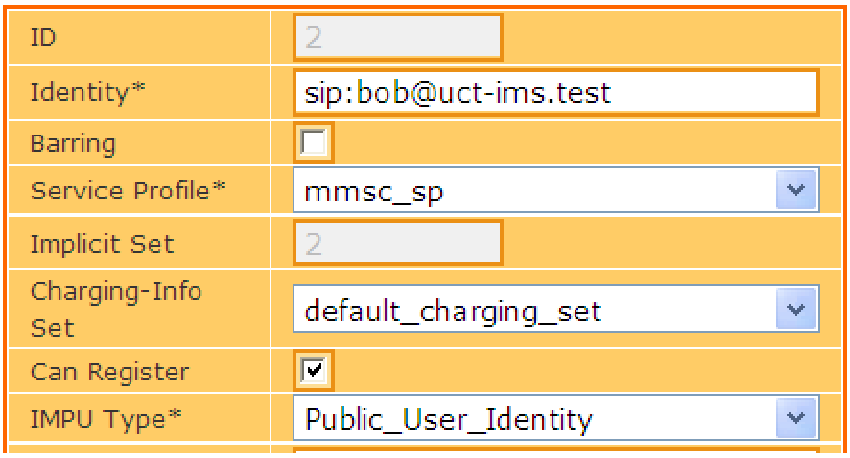
Figure 21: Subscription information (IMSU)

Figure 21 shows how a user's subscription is captured with their name, and linked to a capability set and the details of the serving CSCF of the home IMS domain. The user's private user identity configuration is shown in Figure 22.



ID	2
Identity*	bob@uct-ims.test
Secret Key*	bob

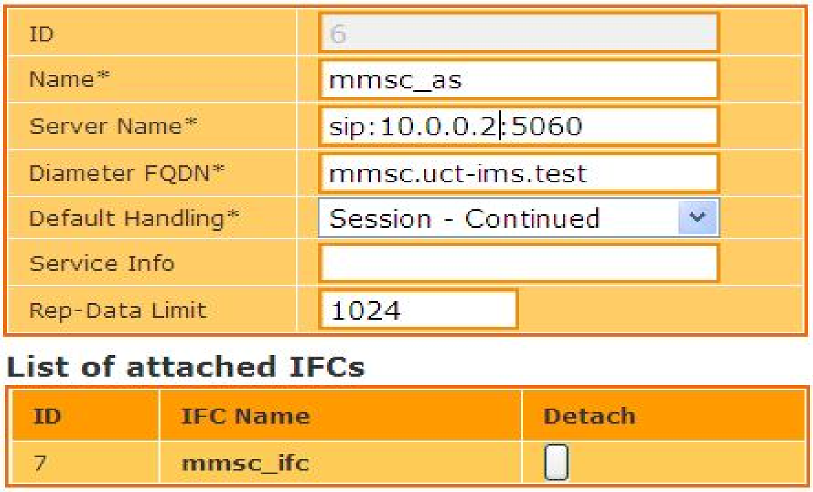
Figure 22: Private User Identity (IMPI)



ID	2
Identity*	sip:bob@uct-ims.test
Barring	☐
Service Profile*	mmsc_sp
Implicit Set	2
Charging-Info Set	default_charging_set
Can Register	☑
IMPU Type*	Public_User_Identity

Figure 23: Public User Identity (IMPU)

Figure 23 shows how the public user identity configuration links to the user's service profile and a charging set. The server's network connection details and the associated filter criteria are shown in Figure 24.



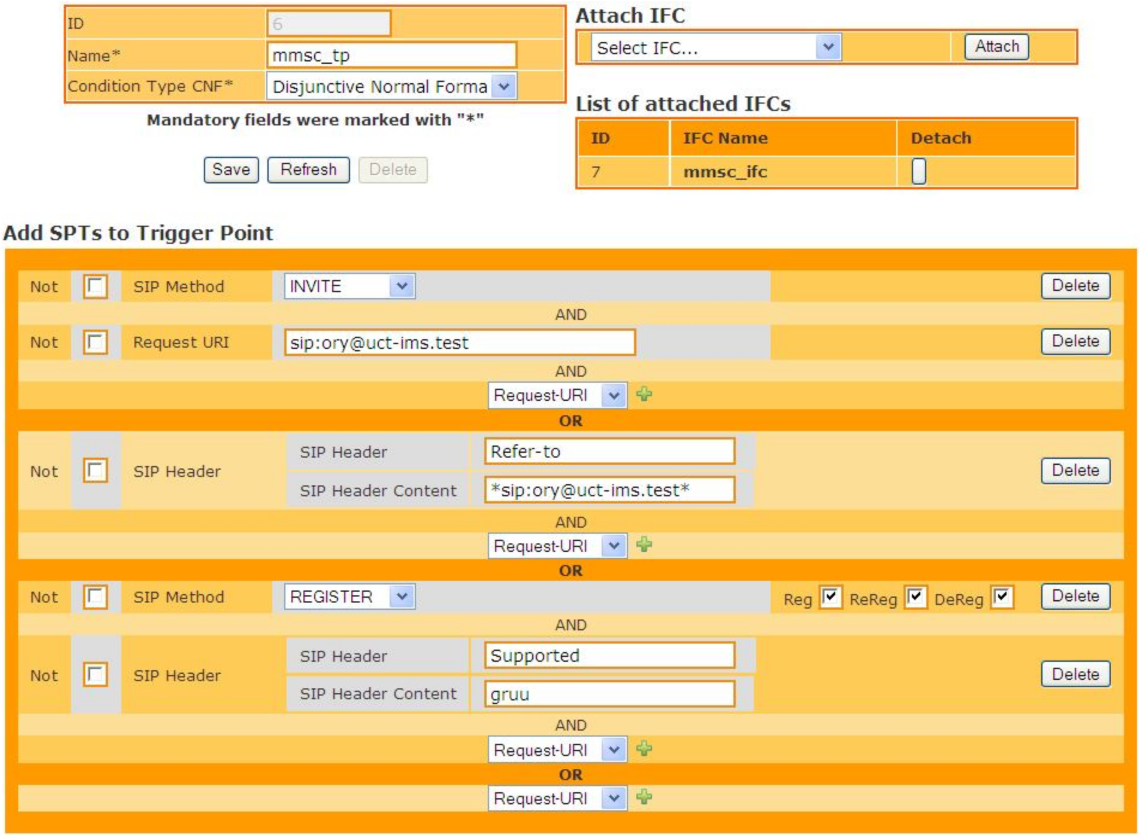
ID	6
Name*	mmsc_as
Server Name*	sip:10.0.0.2:5060
Diameter FQDN*	mmsc.uct-ims.test
Default Handling*	Session - Continued
Service Info	
Rep-Data Limit	1024

List of attached IFCs

ID	IFC Name	Detach
7	mmsc_ifc	☐

Figure 24: Application Server with attached Filter Criteria

The `Default Handling` parameter ensures that the basic IMS session setup procedure can continue if the session continuity server is not responding to requests. Figure 25 shows the trigger point with the associated filter criteria and service point triggers.



ID	6
Name*	mmsc_tp
Condition Type CNF*	Disjunctive Normal Forma

Mandatory fields were marked with "*"

Attach IFC

Select IFC...

List of attached IFCs

ID	IFC Name	Detach
7	mmsc_ifc	☐

Add SPTs to Trigger Point

Not ☐ SIP Method: INVITE

AND

Not ☐ Request URI: sip:ory@uct-ims.test

AND

Request-URI

OR

Not ☐ SIP Header — SIP Header: Refer-to; SIP Header Content: *sip:ory@uct-ims.test*

AND

Request-URI

OR

Not ☐ SIP Method: REGISTER — Reg ☑ ReReg ☑ DeReg ☑

AND

Not ☐ SIP Header — SIP Header: Supported; SIP Header Content: gruu

AND

Request-URI

OR

Request-URI

Figure 25: Trigger Point with Service Point Triggers

The service point triggers are for different phases of the session continuity signalling procedures. The `INVITE` trigger is used for session anchoring during the initial session setup stage. The `Refer-to` header is found in the `REFER` message, which issues a session transfer request to the server. The `Supported` header with the GRUU value is in the `REGISTER` message and triggers third party registration towards the session continuity server.

3.5.2 Relationship of User Configuration Entities

The relationship between the different service configuration parameters is depicted in Figure 26. The profile part indicator indicates the requirement for the user's registration state, i.e. whether the user should be registered or not for filter criteria to apply.

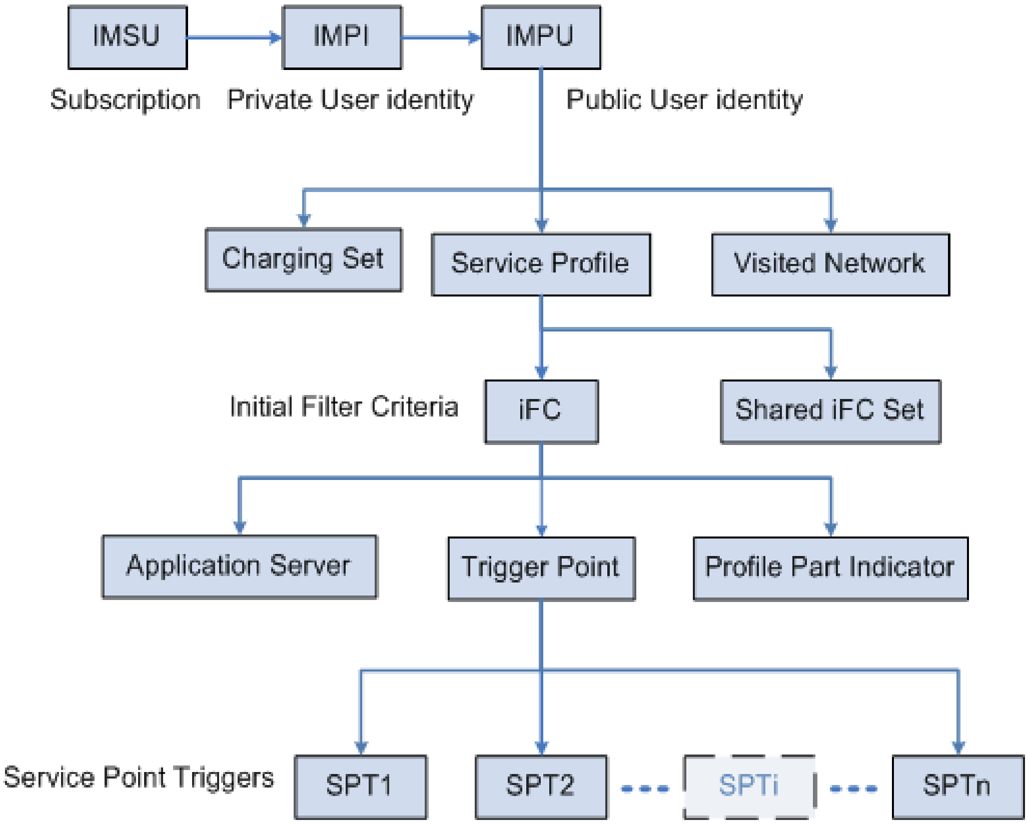


Figure 26: Service Configuration Parameters

3.6 Chapter Discussion

This chapter has presented key service requirements for the implementation of service continuity, and presented a detailed design and architecture of the service. It presented a detailed description of the various functions and signalling interactions involved in session continuity. The previous chapter revealed significant misalignments between proposals in literature and standardization efforts on IMS Service Continuity. This chapter, therefore, also highlights the strict compliance of the implemented signalling interactions with IMS specifications.

Chapter 4 Feature-rich Multimedia Session Continuity

The literature review outlined the progress made around general session continuity and the IMS-based service. The traditional functionality based approaches were found to be inadequate for the services-based IMS environment. While the previous chapter detailed our design and implementation of the session continuity service, this chapter proposes feature-rich enhancements to the standardized session continuity service [60] through a service interworking model of session continuity mechanism, presence service mechanisms, and the RTSP protocol. Figure 27 illustrates the focus of this chapter in relation to the thesis.

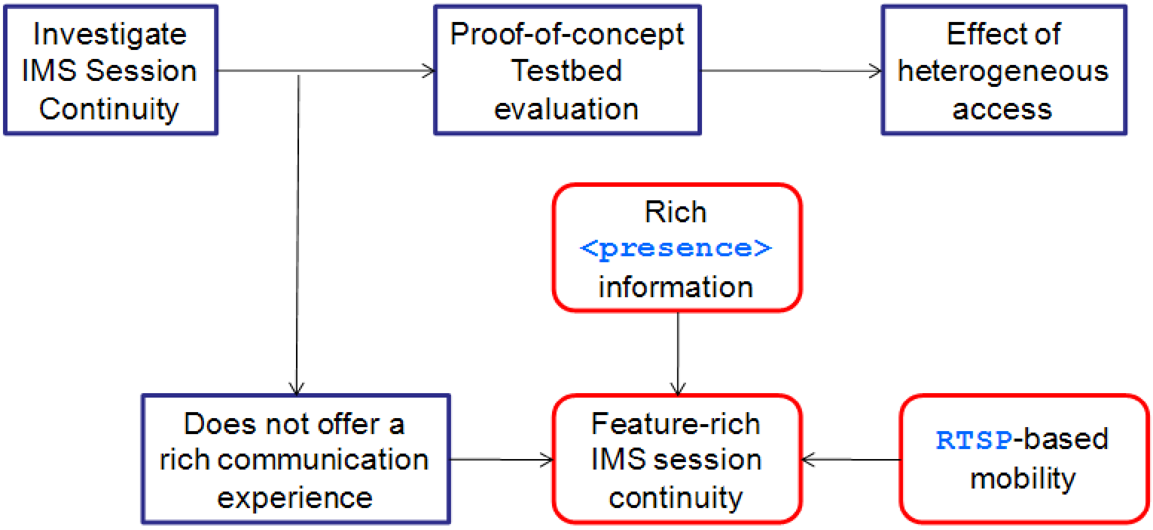


Figure 27: Chapter Research Focus

The analysis of the literature indicated a narrow focus in the state of the art, whereby proposed session continuity solutions only focused on mobility functionality aspects. This chapter extends the service aspect of session continuity, by proposing an integrated services model which provides an enriched user experience using presence-aware session transfers. The proposed enhancements, however, are not implemented as part of the work in the previous chapter. The evaluation of this enhanced session continuity service, therefore, uses a scenario-based theoretical impact analysis instead of practical testbed evaluations.

4.1 Design Considerations

A key enabling aspect of IMS service provisioning is standardization of common interfaces through which different services can communicate. This enables innovative service composition by packaging different service enablers to provide an enriched communication experience to users. A key requirement, therefore, for any new functionality or service introduced, is that it complies with these mechanisms. For a presence-enhanced session continuity service, the design requirements are as follows:

4.1.1 Impact on existing Session Continuity mechanisms

The session continuity service contributes additional signalling to normal session setup, termination and registration interactions. Thus, the additional signalling due to presence enhancement should be minimized.

4.1.2 Seamless Interworking with the Session transfer Operator Policy

The role of policy based management in managing mobility was reviewed in earlier chapters. While the policy enables the operator or service provider to control and manage the requested session transfers, it is unidirectional and thus has no information from the subscriber's side, such as user availability and preferences. The presence enhancement, therefore, provides a mechanism to incorporate subscriber-side information in managing session transfers. Importantly, it should not conflict with the transfer policy such that incoming sessions, or session transfers are not executed properly, thereby resulting in blocked calls.

4.1.3 No impact on UEs which do not support presence

A key architectural requirement of session continuity is that it should not impact UEs which do not support the functionality. Similarly, UEs which do not support the presence enhancement should not be adversely impacted. Thus, when a user (or UE) subscribes to the session continuity service, it should be possible to indicate whether the presence enhancement is supported or not.

4.2 Enhanced Service Architecture

4.2.1 Incorporation of Presence Mechanisms

The high level architectural solution is illustrated in Figure 28. A typical UE-1 has a subscription to the enhanced session continuity service, while UE-2 only subscribes to the presence service. The session continuity server is extended with presence functionality to subscribe to UE-1 user's presence information. For presence subscription, the server acts in user agent mode and performs normal presence subscription signalling using SUBSCRIBE messages. It is then notified (NOTIFY) of the changes in the user's presence status.

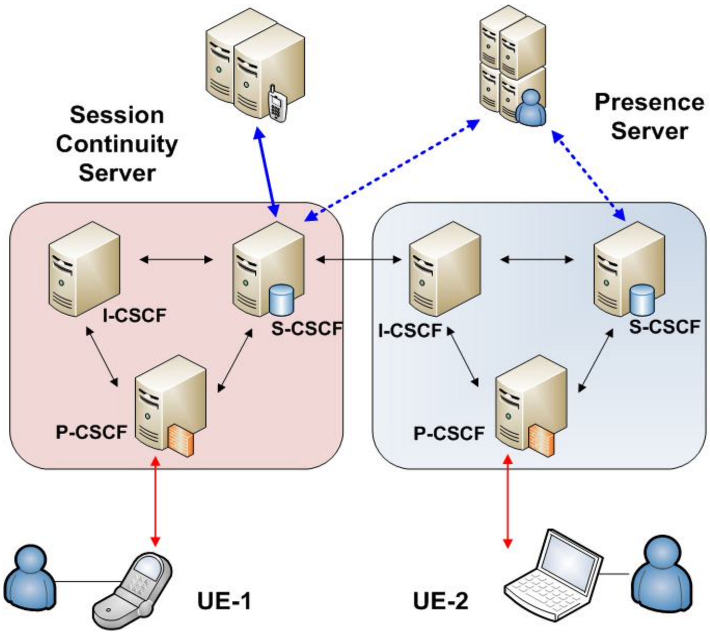


Figure 28: Enhanced Session Continuity Service Interaction

The presence extension adds another dimension to the server's decision making process when responding to session transfer requests, or call terminations. Originally, the server only used the session transfer operator policy to authorize and control transfers, and the session transfer information provided by the requesting UE. Figure 29 shows a sample message body of the transfer information indicating the user's UEs, their capabilities, and their priority of usage.

```
<?xml version="1.0" encoding="UTF-8"?>
<presence xmlns="urn:ietf:params:xml:ns:pidf"
                  xmlns:mmsc="urn:ietf:params:xml:ns:pidf:mmsc"
                 entity="sip:bob@mmscontinuity.test">
        <tuple id="Cellphone">
                <status>
                        <basic>open</basic>
                </status>
                <contact priority="0.5">
        <teluri>tel:+27-12-345-6789</teluri>
      </contact>
                <note>Emergencies only!</note>
        </tuple>
        <tuple id="UCTIMSClient">
                <status>
                        <basic>open</basic>
                        <contact priority="0.85">
                                <mmsc:p-gruu>sip:bob@mmscontinuity.test;gr=mylatestcellphone</mmsc:p-gruu>
                        </contact>
                        <mmsc:capabilities voice="true" video="true"/>
                </status>
        </tuple>
        <tuple id="IPTV">
                <status>
                        <basic>closed</basic>
                        <contact>
                                <mmsc:p-gruu>sip:bob@mmscontinuity.test;gr=myiptvscreen</mmsc:p-gruu>
                        </contact>
                        <mmsc:capabilities>
                                <mmsc:audio surroundsound="true"/>
                                <mmsc:video highdefinition="true"/>
                        </mmsc:capabilities>
                </status>
        </tuple>

</presence>
```

Figure 29: Example Session Transfer Information

The information shows three UEs, the user's cellphone with an "open" presence status indicating willingness to communicate, and a specific contact number. The user has indicated that the number only be used for emergencies. Two more UEs are listed, with corresponding status information, and device-specific contact information indicated by the use of Globally Routable User Agent (UA) URI (GRUU). Device capability is also provided for the IPTV device. The user has identified their preferred communication devices by assigning call-routing-priority values to the devices.

4.2.2 RTSP for Session Continuity

The feature-rich service enhancement employs the Real Time Streaming Protocol to handle multimedia streaming control. The protocol provides trick-play functionality allowing the traditional PAUSE, PLAY, REWIND, etc. commands such as normally found on Video Cassette Recorder (VCR) equipment. This provides users a ‘normal’ viewing and media interaction experience for IPTV/VoD sessions. This rich functionality is exploited, automated and interworked with presence mechanisms, to provide users an enriched session continuity experience. Multimedia streaming sessions can thus be played, paused and resumed in response to session transfer scenarios.

The protocol provides detailed media description mechanisms which expose important aspects of a streaming multimedia session. For example, for a session transfer, the exact time range indicating where a media stream was paused, with reference to its beginning, can be specified by the protocol. This allows the stream to be resumed from the exact paused location after a transfer to a different device. The complex interworking of this rich functionality, and the expressive and dynamic presence service, contribute the essence of the service enhancement proposed in this chapter.

4.3 Enhanced Session Continuity Scenario

This section illustrates how the integration of presence functionality can enrich the user experience for session continuity. The service interaction is described through a hypothetical use case scenario illustrated in Figure 30. In this scenario, an enhanced session continuity server provides an enriched session continuity experience whereby session transfer operations are affected by the user’s device preferences and presence information.

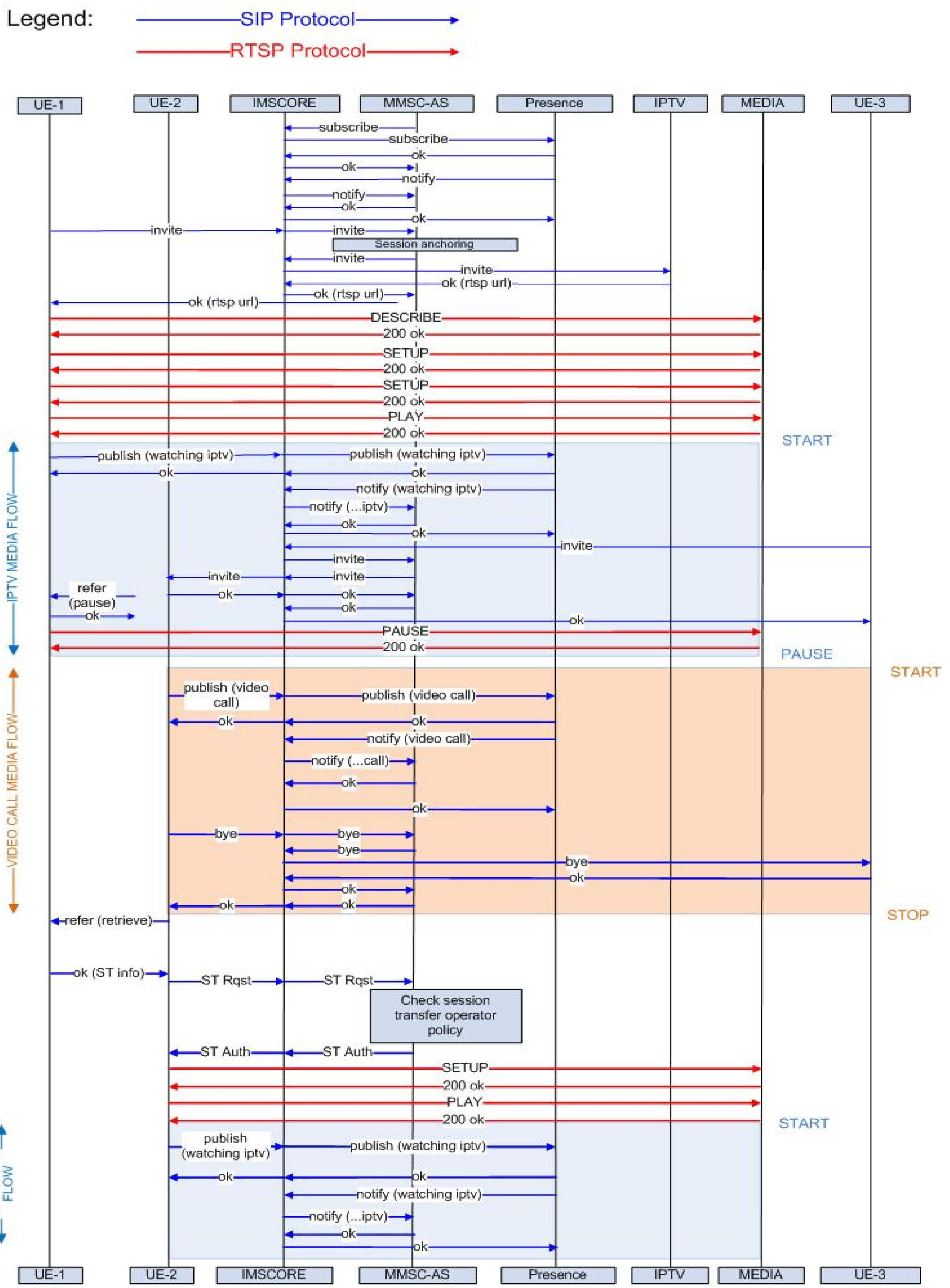


Figure 30: Enhanced Session Continuity Scenario

In the scenario, the server is subscribed to the user's presence information, and receives notifications of the status from the presence server. When the user establishes a VoD/IPTV streaming session using the `<preferred_device>` for IPTV (UE-1), the session continuity server is invoked due to initial filter criteria. The server anchors this session for session transfer purposes, and the INVITE is continued towards the IPTV server. The UE receives media location through RTSP signalling and requests the media.

As part of the enhancement, once the UE starts receiving the media stream, it publishes its `<current_activity>`**`streaming`**`</current_activity>` through normal presence mechanisms, informing its watchers (e.g. session continuity server), of its current activity. When the user receives a voice call from UE-3, the session continuity server routes the request to the user's preferred device for voice calls, UE-2. On call establishment, UE-2 sends an indication (REFER message) to UE-1 to PAUSE the ongoing streaming session, allowing the user to take the call. UE-2 also sends a PUBLISH message to the presence server indicating its `<current_activity>`**`videoCall`**`</current_activity>` status.

On session termination, the user opts to retrieve the paused streaming session from UE-1 and transfer it to UE-2. The session transfer request is sent to the session continuity server, which applies the transfer policy and authorizes the transfer. The streaming session is then continued on UE-2, which publishes its `<current_activity>`**`streaming`**`</current_activity>` status.

4.4 Chapter Discussion

In order to provide attractive services to users, IMS-based service provisioning has to compete with rapidly evolving interactive services in the Web environment. Innovative services, which combine different features and functionalities, are packaged as a single service to users. This chapter presents a novel service enhancement to the IMS-based session continuity service. It exploits the rich RTSP protocol and the versatile presence service, to provide a rich session continuity experience for users, with the added benefit of assured quality of service level enabled by the IMS.

Chapter 5 Evaluation Platform

This chapter presents the evaluation of the implemented session continuity service. It establishes the functionality and performance of the service, focusing on the implementations performed on the UCT IMS Client, the session continuity server, and the signalling performance of the service in a heterogeneous access networks environment.

5.1 Objectives of the Testbed Evaluation

The functionality evaluation establishes whether the implemented prototype session continuity service meets the requirements and specifications for correct message formatting, signalling behaviour and sequences. The evaluation is qualitative and the obtained results ascertain the suitability of the implemented nodes for performance evaluations.

The performance evaluations provide quantitative data establishing the conformance of the implemented service to acceptable parameters for IMS signalling during the registration, instant messaging, session setup, and session transfer phases.

IMS subscribers will access services using a variety of end user devices with different access technologies. A key aspect of the evaluation, therefore, is to determine the effect of heterogeneous access on session continuity user experience.

5.2 Development Tools

The session continuity application server is implemented as a SIP Servlet [56] application and deployed on the JSR289 [57] compliant Sailfin [20] application server.

5.2.1.1 Sailfin Application Server

The Sailfin application server is a sub-project under the GlassFish project. It is an open source implementation of the Sun GlassFish Communication Server [59]. GlassFish is Java EE based application server. Sailfin extends GlassFish by adding a JSR289 compliant SIP Servlet Container. A summary diagram of the testbed setup is shown in Figure 31.

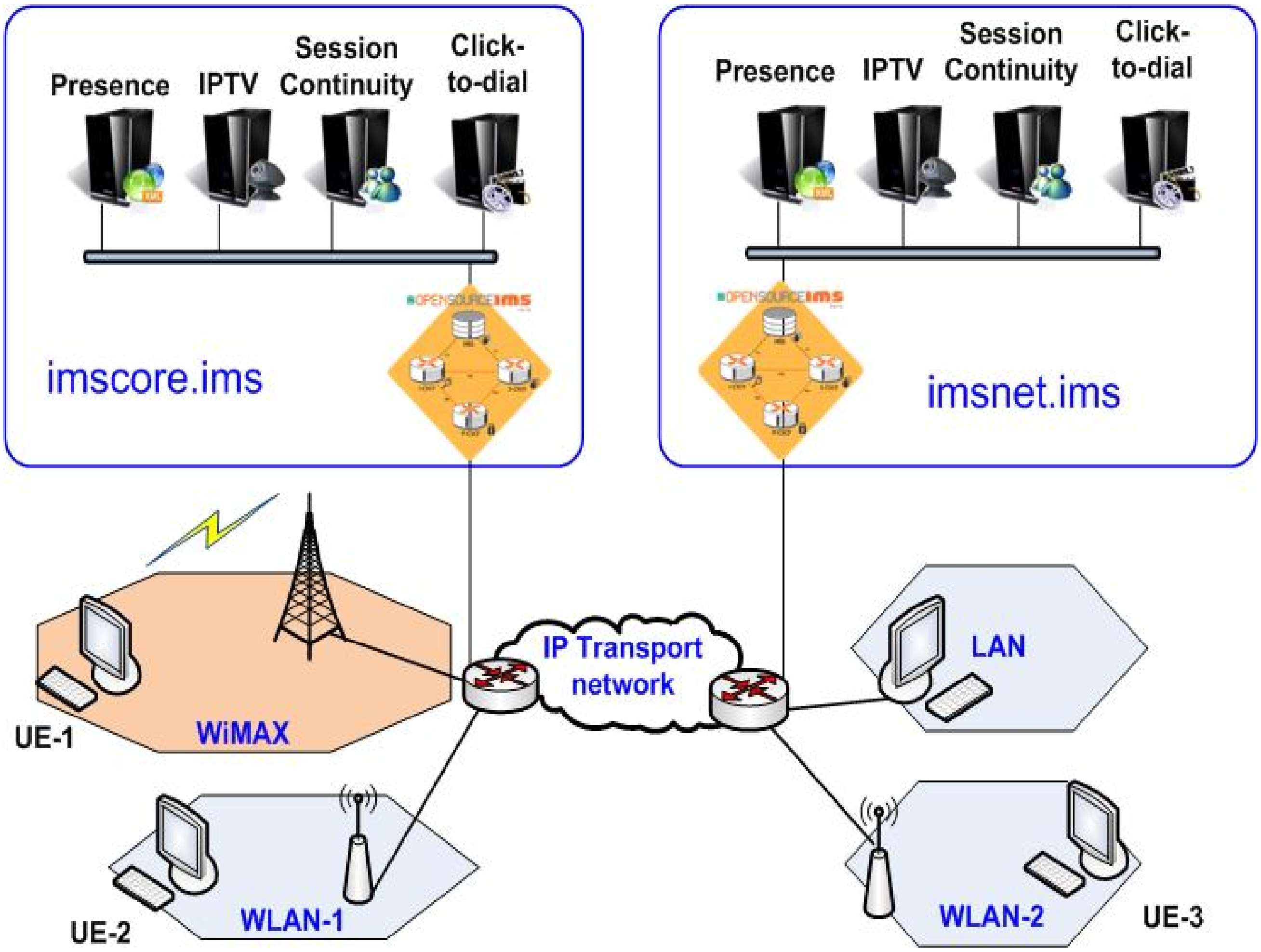


Figure 31: High Level Experiment Setup

5.2.1.2 SIP Servlet Technology

SIP servlets are Java based server-side applications which perform SIP signalling, based on the SIP specification. They are similar to HTTP servlets in that they interact with clients by responding to requests. However, unlike HTTP servlets, SIP servlets can also initiate the communication with clients by sending SIP *requests* to clients. A SIP request, by specification, can have multiple responses – such as provisional and final responses. As such, a SIP servlet can send multiple responses to a client's single request, unlike HTTP servlets. The SIP Servlet specification is based on the generic Java Servlet Specification.

A SIP servlet can interact with other SIP servlets, HTTP servlets and other Java EE components to provide a converged application. These are applications which integrate SIP and HTTP functionality, and components like Java enterprise beans and JavaServer Faces (JSF) applications.

To create an instance of `SipServlet`, the SIP servlet specification defines a `SipFactory` interface. A `SipFactory` instance has methods to create SIP requests, addresses, application sessions, etc. Some of the `SipFactory` methods are listed below (the method parameters are not shown for brevity):

- createRequest() – Returns a `SipServletRequest` object
- createApplicationSession() – Returns a `SipApplicationSession` object
- createSipURI() – Returns a `SipURI` object

The specification also provides a `SipSession` interface which allows the SIP servlet application to keep state, and thus enable a SIP servlet to associate SIP messages with this state information. Particularly, since SIP servlets are stateless, a SIP servlet is now, through a `SipSession` instance, able to process multiple related SIP requests within a dialog. The SipSession can also then be accessed through a `SipApplicationSession` instance, which enables a converged application to access data from different protocol sessions, such as SIP sessions and HTTP sessions.

5.2.1.3 Java Specification Request – JSR289 API

The JSR289 Application Programming Interface (API) is the Java implementation of the SIP servlet specification. It is an enhancement of the SIP Servlet 1.0 specification, earlier implemented in the JSR116 API [58]. To create a SIP application, a SIP servlet class extends the `javax.servlet.sip.SipServlet` class and overrides the methods that it aims to use. The inherited methods correspond to the SIP requests and responses defined in the SIP RFC [10]. An abridged list of the methods is shown in Table II.

Table II: SIP Servlet Methods

SIP Servlet Method	Method Function
doInvite	Process an incoming INVITE request
doRegister	Process an incoming REGISTER request
doMessage	Process an incoming MESSAGE request
doRefer	Process an incoming REFER request
doPublish	Process an incoming PUBLISH request
doSuccessResponse	Process an incoming 2xx response
doProvisionalResponse	Process an incoming 1xx provisional response

5.3 Evaluation Environment

The experiments are performed on an IMS testbed environment comprising the FOKUS Open IMS Core, the UCT IMS Client, and the session continuity server. The testbed consists of a combination of Fast Ethernet, Wireless LAN and WiMAX access technologies. There are two IMS domains, `imscore.net` and `imsnet.ims` located at opposite ends of the testbed. This represents the more typical and more complex communication scenario whereby the communicating parties belong to and are registered on different IMS domains. The scenario test cases use different combinations of access networks for the caller and called parties. A more detailed representation of the testbed is shown in Figure 32.

While 3G access was available for use in the testbed environment, the connection goes through the public cellular network. This contributes additional delays not experienced with the other access technologies, which are directly connected to the testbed.

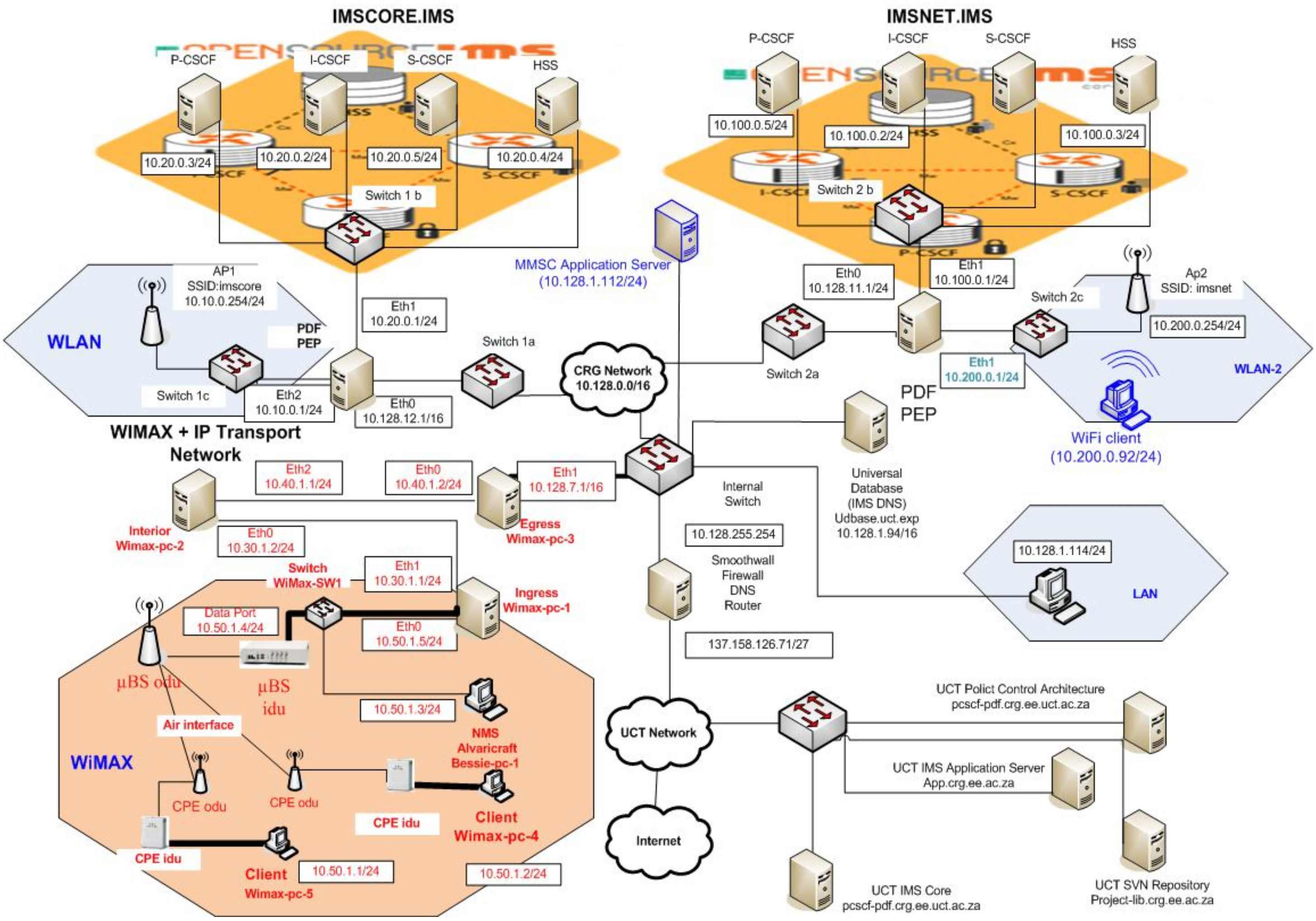


Figure 32: Detailed Testbed Diagram

The detailed diagram shows the connection setup of each of the network technologies used in the evaluation. The two IMS client hosts connecting through WiMAX are connected via Fast Ethernet to the WiMAX customer premises equipment (CPE) indoor units (IDU). The IDUs in turn connect through Fast Ethernet to the outdoor unit (ODU). Since the testbed environment is a private research network, the WiMAX air interface cannot be used due to regulatory spectrum restrictions. The CPE ODU, therefore, connects to the WiMAX base station's ODU through a radio frequency (RF) cable. The RF connections have fixed and variable attenuators for output power control. This configuration is shown in Figure 33.

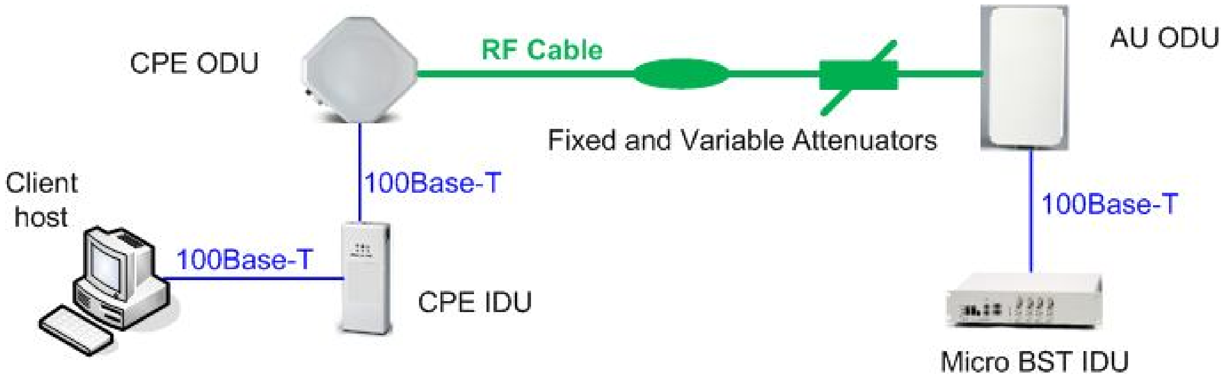


Figure 33: WiMAX Connection setup

The IMS client hosts all run the UCT IMS Client software, and the IMS core hosts run the FOKUS Open IMS Core software. The rest of the hardware configurations of the main nodes of the testbed are shown in Table III.

Table III: Hardware Specifications of Evaluation Platform

Host	Details	Connectivity
8 x IMS Core hosts	Intel® Pentium® Dual-Core CPU 1.80GHz 1GB RAM	All on Fast Ethernet (100Base-T)
5 x IMS Client hosts	Intel® Celeron® CPU 2.53GHz 1GB RAM	2 x 802.11g (WLAN)
		2 x 802.16d (WiMAX)
		1 x Fast Ethernet (100Base-T)
1 x Application Server	Intel® Celeron® CPU 2.53GHz 1GB RAM	Fast Ethernet (100Base-T)
2 x Access Points	D-Link DWL-7100AP Wireless Access Point	802.11a/b/g
2 x WiMAX CPE	BreezeMAX PRO-S CPE	802.16d (WiMAX)

Chapter 6 Evaluations and Results

This chapter presents rigorous evaluations of the envisioned IMS session continuity service using strictly standards-compliant IMS signalling interactions. It determines the impact of this service on IMS user experience by investigating basic IMS session signalling interactions, and the effect of introducing an extra entity, the session continuity server, in the signalling path. Particularly, the overheads introduced by this service are determined.

The chapter then evaluates the effect of the proposed service enhancement discussed in Chapter 4, using impact analysis of a typical use case scenario of the enhanced service.

6.1 Session Continuity Evaluations

The IMS uses extensive and elaborate signalling for session control using the session initial protocol (SIP). Introduction of new services in the network introduces yet more signalling. It adds additional traffic and delay overheads during multimedia session initiation, session termination, and during registrations. The complexity of a service will affect the amount of overhead. The service may require third party registrations, database queries, communication with other third-party services etc. to respond to a user's service request. The associated signalling latencies may have a negative effect on the user experience.

6.1.1 Session Setup Delay Overhead

Session setup is the most crucial procedure in IMS session control. It is the main action performed by a user who decides to initiate a communication session through the network. It is therefore important to ensure that the user's initial contact experience with the network is not adversely affected when new services and capabilities are introduced on the network.

6.1.1.1 Setup Delay without Session Continuity Signalling

Initially, we establish benchmarks for session setup during which no services are invoked for both the originating and the terminating party. The corresponding signalling is shown in

Figure 34. The delay incurred in this scenario is then used to determine the extent of additional delay introduced by the session continuity service.

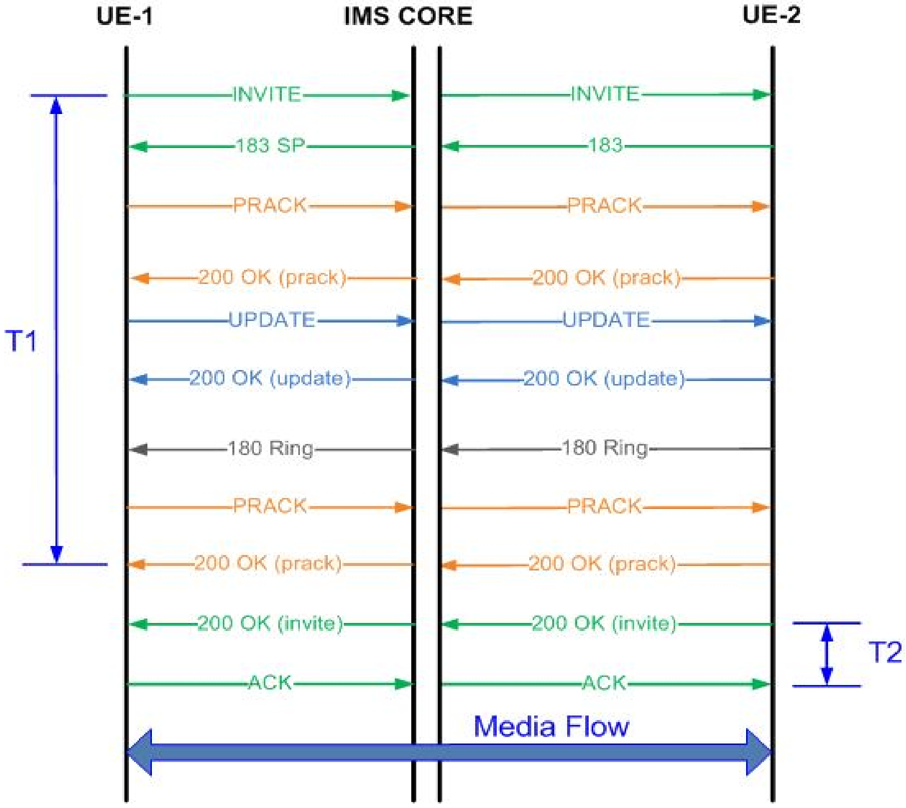


Figure 34: Session Setup without other services

This scenario was performed with different combinations of the access networks. This allows comparative analysis of how the access technology affects the session setup delay. A brief summary diagram of the network arrangements is show in Figure 35 for ease of reference.

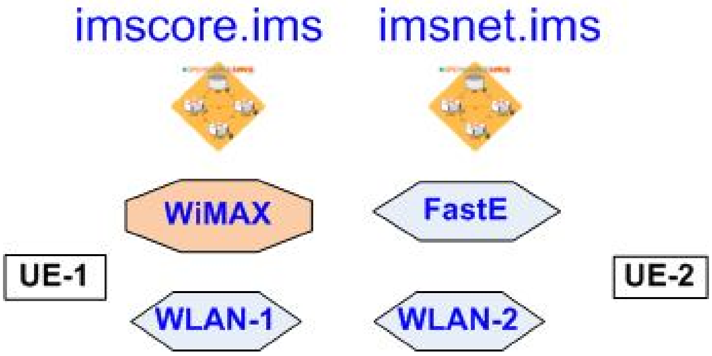


Figure 35: Access Network Layout

In each combination of access technologies, the session setup was performed in both directions (for example, WiMAX to WLAN, and then WLAN to WiMAX) to establish if there were any significant variations in the delays incurred. The measured values are shown in Table IV.

Table IV: Session Setup Delay without Session Continuity Server

UE-1		UE-2		Minimum (seconds)	Average (seconds)	Maximum (seconds)
Connectivity	Domain	Connectivity	Domain			
WiMAX	imscore.ims	Fast Ethernet	imsnet.ims	0.713	0.815	1.209
Fast Ethernet	imsnet.ims	WiMAX	imscore.ims	0.672	0.751	1.105
WLAN-2	imsnet.ims	WiMAX	imscore.ims	1.056	1.207	1.416
WiMAX	imscore.ims	WLAN-2	imsnet.ims	0.965	1.159	1.229
Fast Ethernet	imsnet.ims	Fast Ethernet	imsnet.ims	0.482	0.611	0.806
Fast Ethernet	imsnet.ims	WLAN-1	imscore.ims	0.521	0.701	0.981

In the first instance, UE-1 is connected via WiMAX and is registered on the `imscore.ims` domain. UE-2 is connected via Fast Ethernet and is registered on the `imsnet.ims` domain. A session is then established from UE-1 to UE-2, and the delay measurements recorded. The second instance repeats the operation in the opposite direction and the measurements are recorded. The third and fourth instances follow a similar pattern. The fifth and sixth instances, however, are not reverse versions of each other.

From the table, we note that the least setup delay, average of 0.611 seconds, is obtained in the case of a Fast Ethernet to Fast Ethernet (FE-FE) session setup. This is not surprising since in this case, the UEs are connected to the same access network, and further, are registered with the same IMS domain `imsnet.ims`.

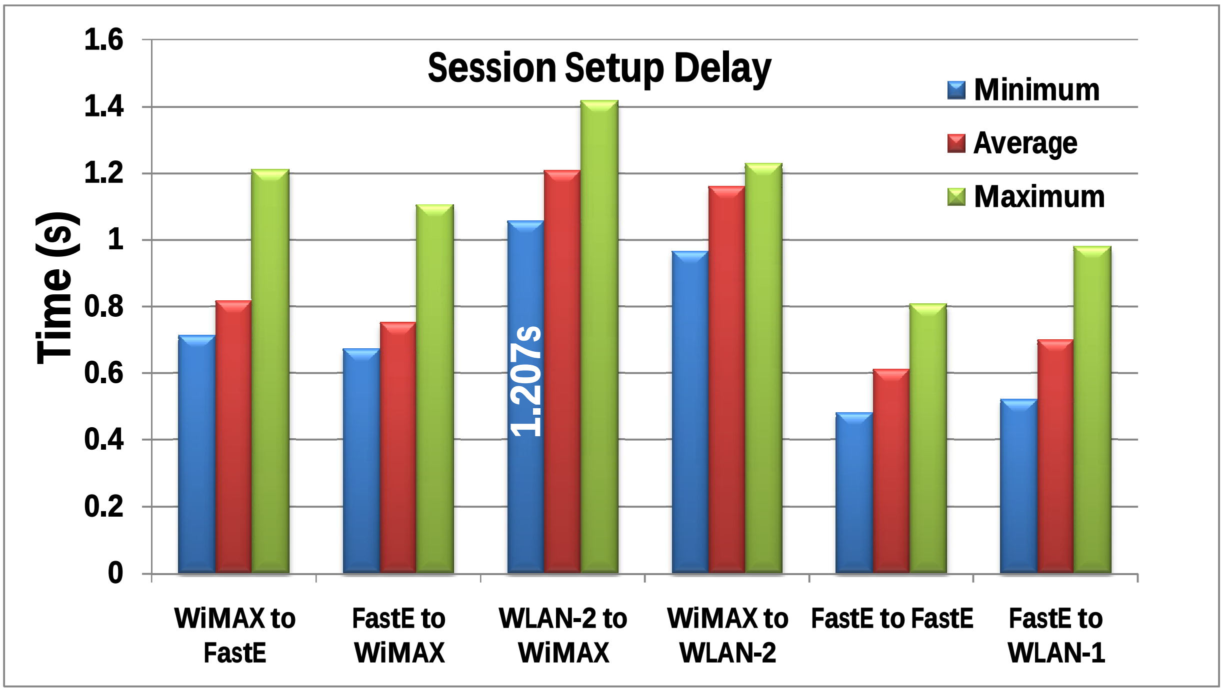


Figure 36: Session Setup Delay with Heterogeneous Access

The largest average session setup delay (1.207 seconds), shown in Figure 36, was incurred in the case of WLAN-2 to WiMAX access networks, with the UEs registered on different IMS domains. The reverse direction presented the second highest delay, with an average of 1.159 seconds, a difference of 3.98%. The worst-case (maximum) delay occurs with the WLAN-2 to WiMAX combinations, with a delay of 1.416 seconds which can be attributed to both heterogeneous access inter-domain IMS registration. When considering the user experience during session initiation, an average waiting time of 1.2 seconds for session setup signalling is considered to be acceptable.

Another perspective of the measured results is presented in Figure 37. The figure extracts the measurements involving Fast Ethernet connectivity when paired with the other access technologies. In the three configurations shown in Figure 37, i.e. FastE to FastE, FastE to WiMAX and FastE to WLAN, the wired configuration consistently results in the least session setup delay.

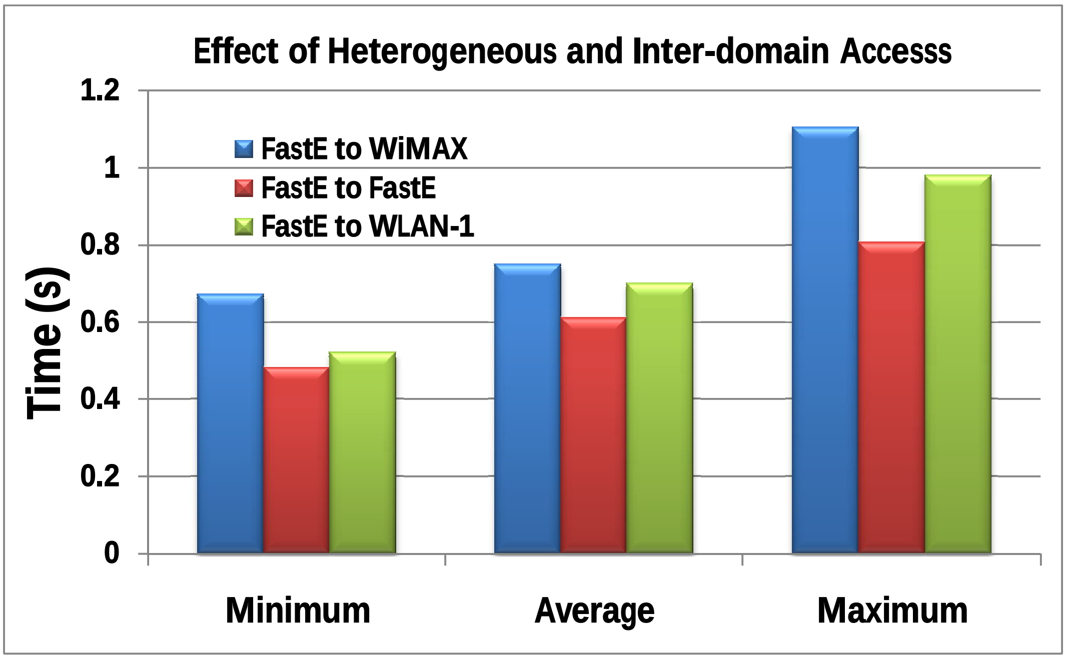


Figure 37: Effect of Heterogeneous and Inter-domain Access

Comparing the FastE-WiMAX and FastE-WLAN scenarios, the case involving WLAN access has a relatively lower delay than the WiMAX case. Particularly, in these two scenarios, the FastE device is registered with the `imsnet.ims` domain, and the wireless devices both register with the `imscore.ims` domain. The difference in the delays, therefore, can be attributed specifically to the access technologies, since the devices are registered with the same IMS domain.

6.1.1.2 Setup Delay with Session Continuity Signalling

The session continuity server is now invoked at session setup to perform session anchoring, in preparation for potential session transfer requests for the session being setup. The anchoring stages during session setup are shown in Figure 38. Full IMS session setup signalling is performed with reliable transmission of provisional responses.

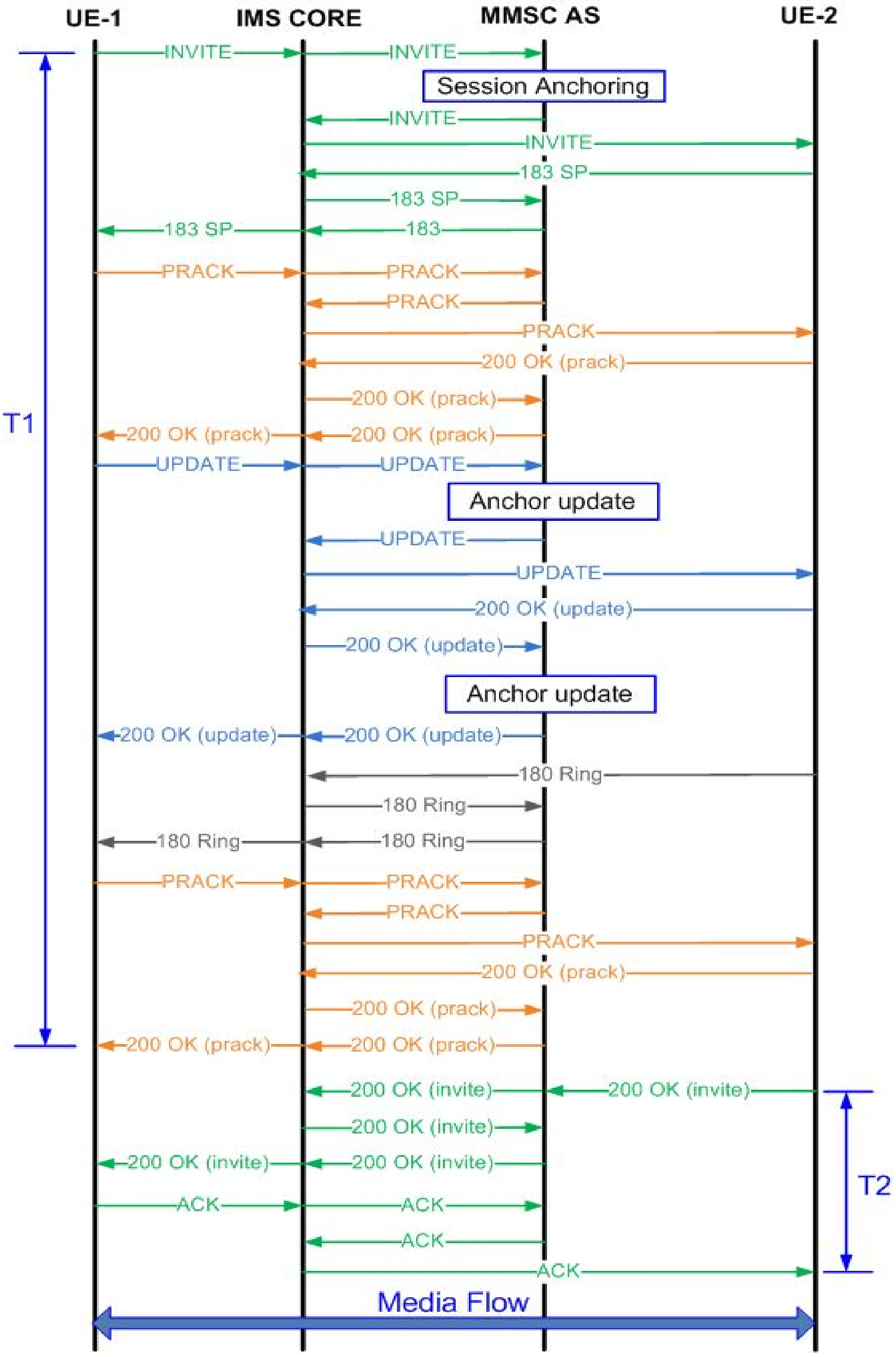


Figure 38: Effect of Session Anchoring on Session Setup

The anchoring affects the signalling of both the requests and responses. For the requests, it extracts session parameters specified by the originating party. This information, which includes media parameters such as the media address and ports, is used to build the session information used later for handling session transfers requests. When a response message is received, session anchoring information is updated with media parameters from the terminating party. The UPDATE message is also sent to the server for anchoring purposes. Session setup thus incurs a number of additional delays due to session continuity signalling.

The delay is measured from the INVITE message from the originating party to the ACK message received by the terminating party.

$$T_{setup} = T1 + T2$$

Table V shows a comparative analysis of the delay results for each of the originating networks in this scenario. UE-2 is permanently connected to the Fast Ethernet network, and UE-1 connects to the WiMAX, then to WLA-1, and then also the Fast Ethernet LAN. The delay is then recorded in each case.

Table V: Session Setup Delay with Session Continuity Server

UE-1 connected to	Minimum (sec)	Average (sec)	Maximum (sec)
WiMAX	**2.513**	**2.685**	**2.847**
WLAN-1	**2.304**	**2.531**	**2.774**
Fast Ethernet (FE)	**1.912**	**2.440**	**2.578**

Similar behaviour is experienced in the case of a FE to FE session setup delay to the case when no server was in the signalling path, i.e. the average delay experienced, 2.440 seconds, is the lowest of all the different network combinations.

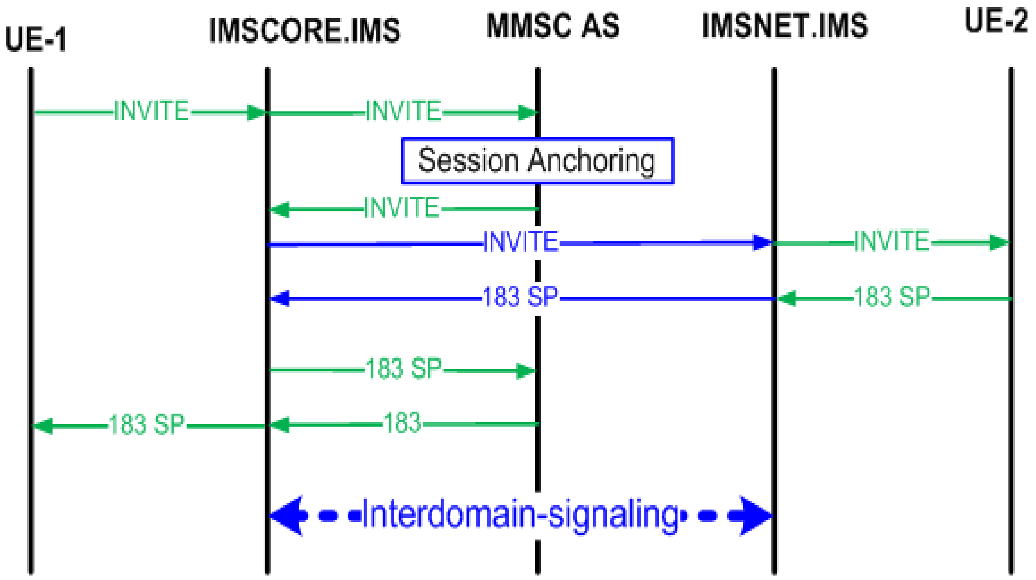


Figure 39: Additional Inter-domain Delay

The largest average delay, 2.685 seconds, is incurred with UE-1 connected to the WiMAX network, which is 9.12% larger than the FE to FE session setup delay, and 5.74% larger than the WLAN-1 to Fast Ethernet setup delay. In each instance of the measurements, UE-1 registers on a different IMS domain `imscore.ims` while UE-2 is on `imsnet.ims.` This adds extra inter-domain signalling delay, see Figure 39. It is noteworthy that although the Fast Ethernet network has substantially larger bandwidth than the wireless networks, its average delay is only 9.12% smaller than the largest incurred delay. This highlights the effect of the additional inter-domain signalling between the `imscore.ims and imsnet.ims` domains.

Comparing the results of this scenario to the results obtained in the previous section indicates the extent of additional delay overhead introduced by the session continuity service. The first measurements of both scenarios, see Figure 40, capture the delay for session setup signalling between the WiMAX and Fast Ethernet networks.

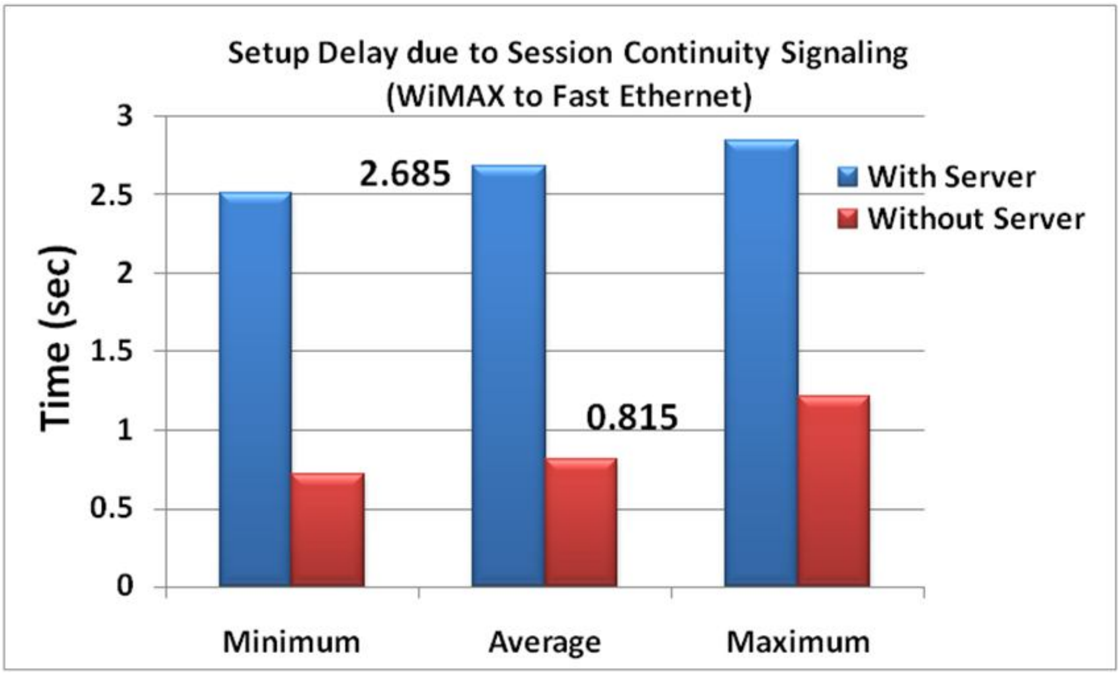


Figure 40: Additional Delay due to Session Anchoring

Focusing on only the average values, the session continuity service introduces 230% additional average delay to the session setup process. Stated differently, the session setup delay is 3.29 times as long due to session anchoring.

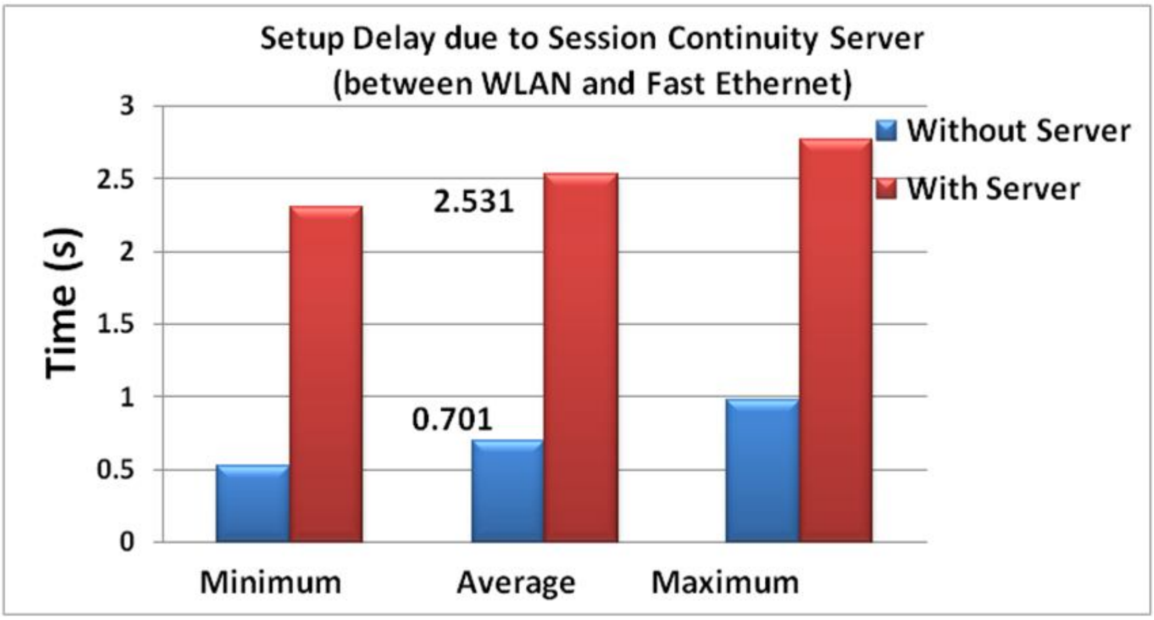


Figure 41: Setup Delay with WLAN to Fast Ethernet Access

The last measurement of Table IV (average delay of 0.701 seconds) and the second measurement of Table V (average delay of 2.531 seconds) are the average session setup delays between the WLAN-1 and Fast Ethernet networks. Session anchoring and inter-domain signalling contribute to the 261% additional session setup delay shown in Figure 41.

6.1.2 Session Termination Delay Overhead

This evaluation determines the delay incurred during session tear-off of a multimedia session with different originating and terminating network configurations. The session continuity server receives session tear-off signalling so that it clears the associated session anchoring information. This is shown in Figure 42.

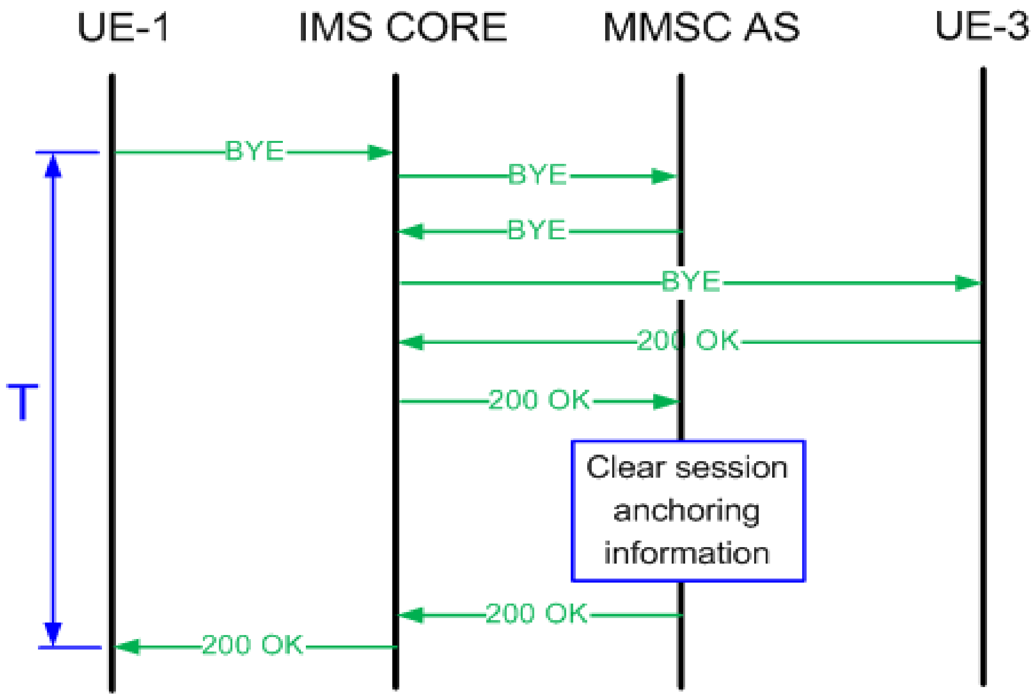


Figure 42: Session Termination with Session Continuity Server

The session anchoring information can be cleared before the final 200ok response during session tear-off signalling, or at the end of the signalling after the 200ok response. The latter is referred to as **expedited session** termination (Figure 43), whereby the session continuity server first completes the session termination signalling before clearing session anchoring information.

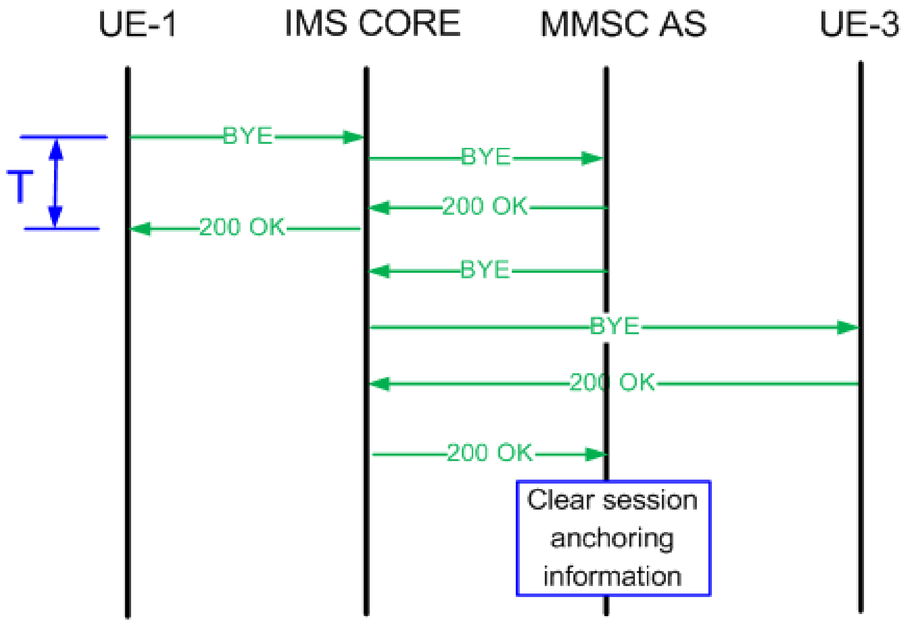


Figure 43: Expedited Session Termination

With expedited termination, Figure 43, the sender of the BYE message is expeditiously released from a potentially long waiting period after sending the BYE message. The measured results for the two approaches of session termination, compared to the case without the server, are summarized in Table VI.

Table VI: Session Termination and Expedited Termination (seconds)

	Without Session Continuity Server	With Session Continuity Server		Server with Expedited Session Termination	
			Factor Increase		**% Increase**
WiMAX to FastEthernet	0.074	0.813	**10.98**	0.198	**167.6%**
WiMAX to WLAN	0.085	0.971	**11.42**	0.222	**161.2%**
FastEthernet to FastEthernet	0.045	0.635	**14.11**	0.131	**191.1%**

In expedited session tear-off, the session continuity server logically assumes the role of a terminating party by issuing a final response to the received BYE message. In this way, it is acting as a normal user agent client. It is, however, interesting to note that even though the session continuity server acts in user agent mode, this scenario yields larger average delays {0.198s; 0.222s; 0.131s} when compared to the ‘normal’ scenario {0.074s; 0.085s; 0.045s}when no session continuity server is involved.

The delay incurred with WiMAX to FastEthernet signalling is 167.57% larger, while the WiMAX to WLAN signalling incurred 161.18% additional delay. FastEthernet to FastEthernet expedited session tear-off incurred a 191.11% additional signalling delay. For simplicity, the additional delays due to the session continuity are expressed as a multiplicative factor since the percentage increases are large than 1000% percent.

6.1.3 Session Transfer Delay Overheads

Session transfer measurements were taken with the method shown in Figure 44. The measuring starts when UE-1 commands UE-2 to initiate a transfer request to the server. The server evaluates the request against the session transfer policy and correlated it with the session established between UE-1 and UE-3. For a full transfer, the session leg towards UE-1 is terminated and fully migrated to UE-2. The delay is measured up to receipt of the BYE message at UE-1.

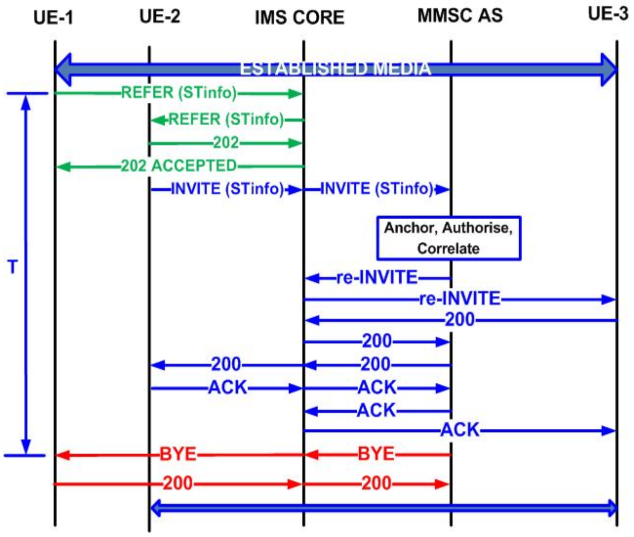


Figure 44: Session Transfer Delay Measurement

Table VII summarizes the delays incurred for session transfers. UE-2 is connected to the WiMAX network, and UE-1 is tested for both the WiMAX and WLAN-1 networks. This determines the difference between homogeneous access and heterogeneous access session transfers.

Table VII: Session Transfer Delay (UE-2 on WiMAX)

UE-2 (Transfer target) connected via WiMAX			
UE-1 connection	**Minimum (sec)**	**Average (sec)**	**Maximum (sec)**
WiMAX	1.501	1.521	1.557
WLAN-1	1.620	1.632	1.678

The signalling delay is found to be larger when the devices are on different access networks. While this may indicate differences in network capabilities, the contribution of locality between UE-1 and UE-2 when they are on the same network must be acknowledged. Further, the difference in average delays between the two networks is minimal.

In Table VIII, UE-2 is connected via WLAN-1 access network. A similar effect of network locality is observed. The delay is shorter when both UE-1 and UE-2 are on the same network, in this case, the WLAN.

Table VIII: Session Transfer Delay (UE-2 on WLAN-1)

UE-2 (Transfer target) connected via WLAN-1			
UE-1 connection	**Minimum (sec)**	**Average (sec)**	**Maximum (sec)**
WiMAX	1.630	1.659	1.695
WLAN-1	1.312	1.376	1.462

At closer analysis, it becomes clear that a WiMAX to WiMAX session transfer (1.521 seconds from Table VII) incurs a larger delay than a WLAN to WLAN session transfer (1.376 seconds from Table VIII). This can be attributed to the higher bandwidth of the WLAN access.

6.1.4 Session Setup and Termination Traffic Overhead

In this measurement, full IMS session setup signalling is enabled with the reliable transmission of provisional responses. The session continuity application server is invoked in both directions of the signalling, each time updating any changes for session anchoring. The traffic overhead is measured in terms of the total number of SIP messages exchanged.

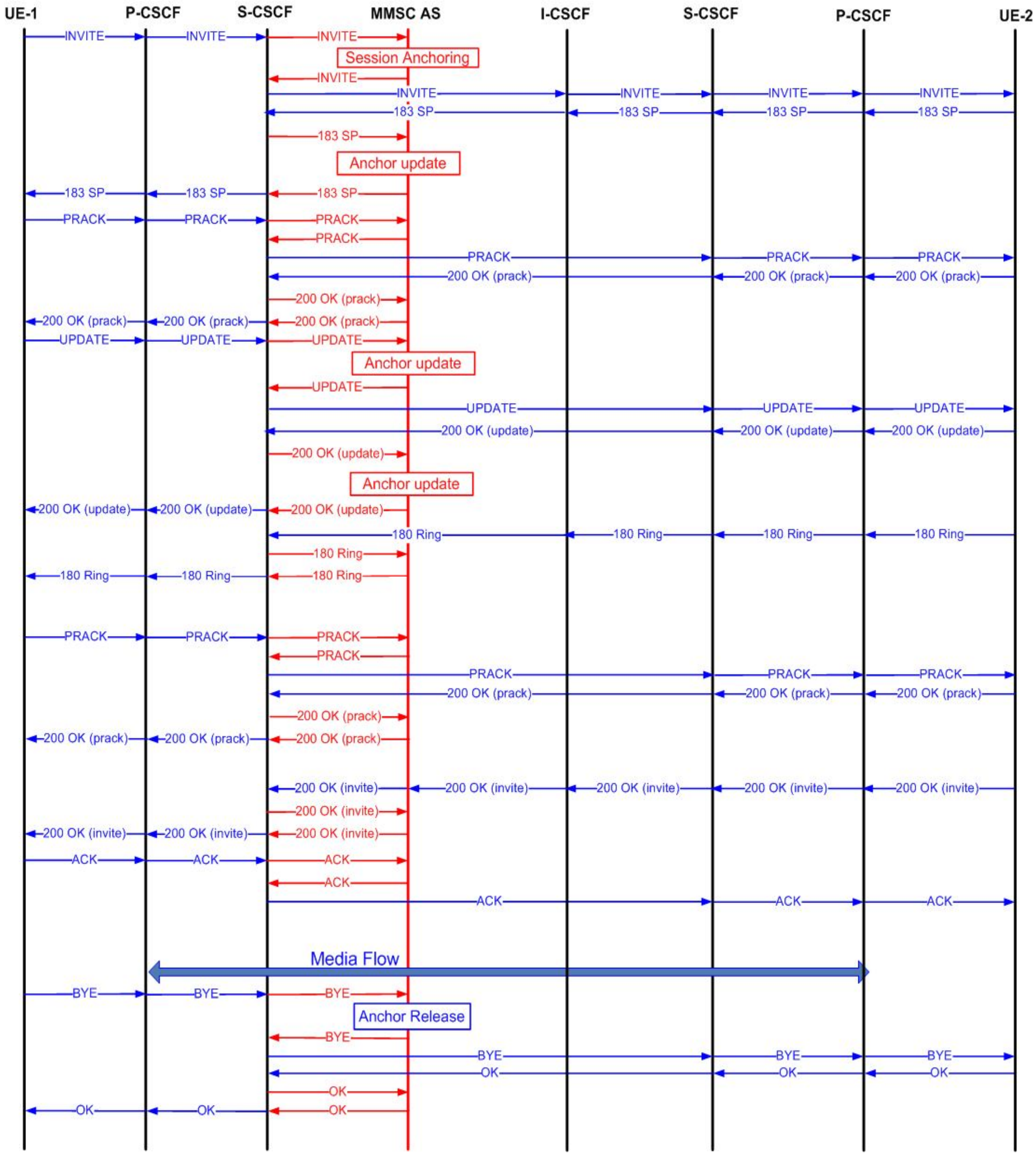


Figure 45: Signalling Overhead due to Session Continuity Server

In the above scenario, only the originating user is subscribed to the session continuity service, whereas the terminating user is not. The application server is thus only invoked at the originating domain. The scenario depicts the worst case scenario whereby the server is invoked in most of the exchanged messages. Table IX summarises the signalling overheads in Figure 45.

Table IX: Signalling Overhead during Session Setup, Termination and Transfer

	Without Session Continuity Server	With Session Continuity Server	Percentage Increase
	(number of messages)	(number of messages)	
Session Setup	60	82	37%
Session Termination	10	14	40%

The session continuity server introduces a total signalling overhead of 37% for session setup. The total overhead recorded for session termination is 40%. Notably, while the increase in session termination is larger in terms of percentages, the actual number of additional messages for session setup is larger, recording 22 additional messages compared to only four extra messages for termination.

6.2 Evaluation of Presence-enhanced Session Continuity

The previous section evaluates the prototype implementation of the session continuity service as specified by 3GPP. This section evaluates the enhanced session continuity service proposed in Chapter 4. The evaluation is performed using a theoretical analysis of the scenario described in Chapter 4. The illustration is shown again in Figure 46 for ease of reference.

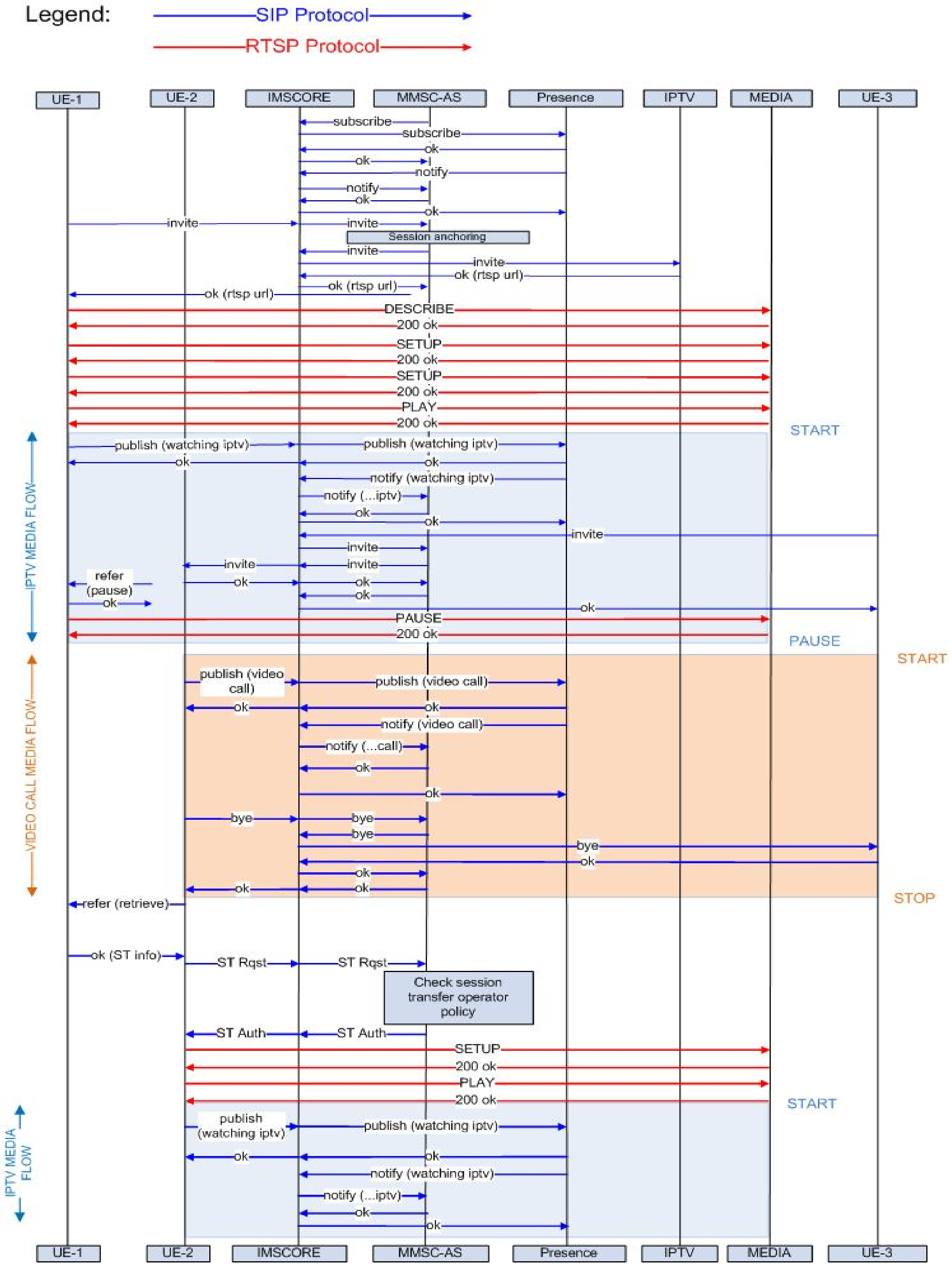


Figure 46 Enhanced Session Continuity Scenario (Recap of Fig. 29)

The analysis focuses on key impact areas of session signalling to determine if there are adverse effects introduced by the enhancement.

6.2.1 Impact on Service Deployment

The integration of presence functionality only requires modifications to the application server hosting the service, and presence capabilities on the user's UEs subscribed to the enhanced service. Users not subscribed to the service enhancement are not impacted. Operators can therefore seamlessly integrate the service enhancement into already deployed session continuity servers.

6.2.2 Impact on Service Invocation

There are two possible ways of invoking the enhanced service functionality. A possible method is to deploy a specific service profile for the enhanced session continuity service, and a separate profile for the basic service. Users would then subscribe to either service explicitly.

An alternative approach is to deploy the presence enhancement on all session continuity servers. The UEs would then indicate support for the enhanced session continuity service, "`ensecon`", using normal SIP supported extensions mechanism, i.e. by including the service identifier, "`ensecon`", in the "`Supported`" header field of a SIP request as indicated below:

```
Supported: ensecon
```

6.2.3 Impact on Service Interworking

The enhanced session continuity server interworks with the presence server using normal presence mechanisms. The presence server requires no modifications, and is therefore not adversely impacted by interworking with the session continuity service.

6.2.4 Impact on Signalling Overhead

The earlier sections of the chapter investigated the signalling overhead due to session continuity. Additional impact due to the presence enhancement was evaluated using a qualitative analysis. The user’s presence information watched by the enhanced session continuity server is very dynamic and changes in a non–deterministic fashion. Quantitative measures of additional signalling overheads will thus only give insight into the specific scenario under investigation. This thesis, therefore, only performs a qualitative evaluation of the service enhancement.

The user experience is enriched with new functionality that allows session transfers to be affected by the user’s dynamic presence information. The presence functionality is already a generally supported feature of most end user devices. In addition, the additional presence signalling for session continuity purposes is required on an occasional basis when transfers are performed. This implies minimal adverse effect on the user experience due to the service enhancement.

6.3 Results Analysis and Chapter Discussion

This chapter has presented a detailed yet comprehensive analysis of the IMS session continuity service, based on the implemented prototype service on a practical IMS testbed. It outlined the impact of this service on various IMS signalling interactions, and the associated user experience. For basic session initiation, i.e. without session continuity signalling, the evaluations using heterogeneous networks revealed wireless access to incur the largest signalling delay. In comparison, wired access (fast Ethernet to fast Ethernet) had the least session setup delays compared to the WLAN and WiMAX combinations. Similar results were obtained for session termination, with the wired access incurring the least delays.

Introduction of the session continuity server in the signalling path contributes additional delays to both session setup and session termination. For session transfers, the effect of heterogeneous access on the signalling delay was observed from the larger delays incurred when the devices were connected using WiMAX in comparison to WLAN access.

The chapter further evaluated the additional service enhancement introduced in Chapter 4 using a typical use case scenario. This additional functionality upgrade was found to enrich the user experience without significant adverse effects from both the user perspective, and the network operator in terms of extensive technical requirements or modifications of existing infrastructure.

Chapter 7 Conclusions and Recommendations

This thesis has performed an in-depth investigation of the envisaged IMS session continuity service. Specifically, it has explored the practical feasibility of the service and its impact on IMS user experience. A future outlook for service enhancements was also investigated. This chapter presents the conclusions drawn from the thesis and a summary of the contributions. The chapter ends with recommendations for further study.

7.1 Conclusions

7.1.1 The Role of Standardization on Session Continuity

Session continuity, or explicitly, ensuring the seamless continuity of a multimedia session during mobility events, is a significant challenge, particularly, when it is required across heterogeneous access networks or devices with different capabilities. While a lot of research has explored different aspects of this challenge, the majority of proposals have different focus areas, and typically have very limited scope individually. Standardization efforts from 3GPP on IMS service continuity have identified a harmonious approach to handling the continuity of multimedia sessions during mobility events. The practical implementation and testbed evaluations performed in this thesis, have provided qualitative and quantitative data supporting this service architecture.

7.1.2 IMS-based Service Deployment

The rapid development and deployment of different services and applications in the Web (and now Web 2.0) environment, has placed significant pressure on the adoption of IMS. Particularly, the value proposition and service differentiation of IMS-based services versus the large array of innovative and rapidly adopted web services requires significant attention. Rigorous evaluations of IMS-based services, as performed in this thesis, provide clear indications of achievable service quality levels. This allows operators to guarantee specific quality of service levels, differentiating their services from the best-effort model used in the web environment.

7.1.3 Enriched User Experience through Service Integration

In order to attract users to IMS-based services, enriched and integrated services comprising innovative features and functionalities and assured quality of service levels, will allow service providers to offer a compelling value proposition to even the most sceptical Web 2.0 users.

The thesis has explored a novel service enhancement to the 3GPP-specified IMS-based session continuity service. It illustrated an interworking model incorporating the use of presence mechanisms, the session transfer operator policy, and user preferences to control session transfers. The evaluations revealed that the basic session continuity service and its presence-enhanced version have minimal impact on the various IMS signalling scenarios and service deployment. The model ensured a seamless integration of the enhancement, providing subscribed users with an enriched experience without adversely affecting existing infrastructure, unsupported devices, or unsubscribed users.

7.2 Recommendations and Future Work

This thesis has established an essential and timely point of departure for further research on IMS service continuity. The practical evaluations performed support the theoretical feasibility of this service. However, while the thesis has also explored a possible evolutionary path for the further development and enhancement of the service, there is still significant scope for further research in this area.

7.2.1 Evaluations of PS-CS Session Continuity Scenarios

The 3GPP system is undergoing evolution to fully IP-based network architecture, i.e. the Long Term Evolution (LTE) of both the Core network and Radio access network. The thesis focused on inter-device session continuity in a heterogeneous access networks environment using packet switched networks. IMS service continuity, however, extends to session continuity between packet switched and circuit switched systems using IMS Centralized Services mechanisms. Further research should investigate scenarios whereby multimedia sessions are transferred between the circuit switched and packet switched domains.

7.2.2 Service Evaluations for the Media Plane

The service evaluations mainly focused on the impact of service continuity on the signalling plane. Further study should focus on the impact of session continuity on media flow during a communication session. Media plane issues such as media adaptation and device capability exchange during inter-device session transfers should be investigated further.

7.2.3 Charging and Session Continuity

Session continuity in the IMS is provided as a service explicitly subscribed to by end users. As a service, charging models for this service are yet to be developed. Further research should investigate ways of how, or if, users should be billed when subscribing to the service. The research should particularly focus on the effect these charging models have on the service uptake by users. In addition, charging implications for session transfers across different administrative domains should be investigated.

7.2.4 Quantitative Evaluation of the Presence Enhancement

The thesis proposed an evolutionary path for the session continuity service, through interworking or integration of presence mechanisms. While qualitative evaluations where performed on this enhancement, further research should investigate the value, if any, of a quantitative impact analysis of this service enhancement. The dynamic and non-deterministic changes of presence status information make it considerably difficult to determine a consistent signalling overhead due to the presence enhancement. Further research should investigate the use of sample historical data of presence status updates, and determine conclusively and quantitatively, the signalling overhead due to the presence enhancement.

References

[1] Wingley and Company, “Convergence of Telecommunications, Broadcasting and the Internet: A Regulatory Perspective”, *8th Annual Telecommunications & ICT Summit*, 26 June 2008

[2] Miika Poikselka, Georg Mayer, “The IMS: IP Multimedia Concepts and Services”, 3rd Edition, 2009

[3] GSM Association, “Rich Communication Suite: Functional Description, Release 1, Version 1.0, 15 December 2008.

[4] GSM Association, “Rich Communication Suite: Technical Realization, Release 1, Version 1.0, 16 December 2008.

[5] 3GPP TS 23.228: “IP Multimedia Subsystem; Stage 2”

[6] H. Schulzrinne and E. Wedlund, "Application-Layer Mobility using SIP", *ACM Mobile Computing and Communications Review, Vol. 4, No. 3,* July 2000, pp. 47-57.

[7] A. Dutta, F. Vakil, J. Chen, M. Tauil, S. Baba, N. Nakajima and H. Schulzrinne, “Application Layer Mobility Management Scheme for the Wireless Internet”, *IEEE International Conference on Third Generation Wireless and Beyond (3Gwireless'01)*

[8] C. Perkins, “IP Mobility Support for IPv4”, RFC 3344, January 2002

[9] D. Johnson, C. Perkins and J. Arkko, “Mobility Support in IPv6”, RFC 3775, June 2004

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

[11] Y. Cui, K. Nahrstedt and D. Xu, "Seamless User-Level Handoff in Ubiquitous Multimedia Service Delivery", *Multimedia Tools and Applications*, Vol. 22, Issue 2, pg 137-170, February 2004

[12] 3GPP TR 23.893, “Feasibility Study on Multimedia Session Continuity; Stage 2”

[13] ETSI TS 123 237 v8.5.0, IP Multimedia Subsystem (IMS) Service Continuity, Stage 2”

[14] ETSI TS 124.229, “Internet Protocol (IP) multimedia call control protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3”

[15] Jin Tang, Carol Davids, Yu Cheng, “A Study of An Open Source IP Multimedia Subsystem Testbed”, *ICST Conference on Heterogeneous Networking for Quality, Reliability, Security an Robustness*, Article 45, Hong Kong, 2008.

[16] D. Vingarzan, P. Weik, “IMS Signalling over Current Wireless Networks: Experiments Using the Open IMS Core”, *Vehicular Technology Magazine, IEEE, Vol. 2(1),* pp28-34, August 2008.

[17] R. Good, P. Wilson, T. Mvere, K. Marungwana, and N. Ventura, “An Evaluation of SIP Based Mobility in the IP Multimedia Subsystem”, *Southern Africa Telecommunications Networks and Applications Conference (SATNAC*), Wild Coast Sun, KZN, 7-10 Sep. 2008

[18] David Waiting, Richard Good, Richard Spiers and Neco Ventura, “The UCT IMS client”, *Proceedings of the 5th International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TridentCom), pg 1-6,* Washington, DC, USA, 6 – 9 April, 2009

[19] D. Vingarzan, P. Weik, and T. Magedanz, “Development of an Open Source IMS Core for emerging IMS Testbeds for Academia and Beyond”, *Special Issue on IMS, Journal on Mobile Multimedia (JMM), Vol. 2(3), Rinton Press,* Princeton, USA, 2007

[20] The SailFin Project, https://sailfin.dev.java.net

[21] A. K. Salkintzis, “Interworking Between WLANs and Third-Generation Cellular Data

Networks", *57th IEEE Seminannual Vehicular Technology Conference (VTC), Vol. 3, pg 1802-1806,* Athens, Greece, 22 – 25 April, 2003

[22] A. K. Salkintzis, "WLAN/3G Interworking Architectures for Next Generation Hybrid Data Networks", *IEEE International Conference on Communications (ICC), Vol. 7, pg 3984-3988,* Athens, Greece, 20 – 24 June, 2004

[23] W. Song and W. Zhuang, "Interworking of 3G cellular networks and wireless LANs", *International Journal of Wireless and Mobile Computing, Vol. 2(4),* 2007

[24] ETSI TR 122 934 v8.0.0, "Feasibility study on 3GPP system to Wireless Local Area Network (WLAN) interworking", January 2009

[25] Paulo Pinto, Luis Bernardo, and Pedro Sobral, "Seamless Continuity of PS-services in WLAN/3G interworking", *ELSEVIER Computer Communications,* Vol. 29(8), pg 1055-1064, 2006

[26] ETSI TS 123 206 v7.5.0, "Voice Call Continuity (VCC) between Circuit Switched (CS) and IP Multimedia Subsystem (IMS); Stage 2", January 2008

[27] Mischa Schmidt, Bernd Lamparter, and Stefan Schmid, "Voice Call Continuity – A Critical Step towards All-IP based Next Generation Networks", *IEEE Global Telecommunications Conference (GLOBECOM),* 30 November – 4 December 2008, New Orleans, LA, USA

[28] ETSI TS 123 292 v9.4.0, IP Multimedia Subsystem (IMS) Centralized Services, Stage 2

[29] Wei, Q., Farkas, K., Prehofer, C., Mendes, P., and Plattner, "Context-aware handover using active network technology", *The International Journal of Computer and Telecommunication Networking, Vol. 50(15),* pg 2855 – 2872, October, 2006

[30] Alexandros Kaloxylos, George Lampropoulos, Nikos Passas, and Lazaros Merakos, "A flexible handover mechanism for seamless service continuity in heterogeneous environments", *Journal on Computer Communications, Vol. 29(6),* pg 717-729, 31 March 2006

[31] J. Loughney, Ed., M. Nakhjiri, C. Perkins, R. Koodli, "Context Transfer Protocol (CXTP)", RFC 4067

[32] Michael Georgiades, Tasos Dagiuklas, Rahim Tafazolli, "Middlebox Context Transfer for Multimedia Session Support in all-IP Networks", *International Wireless Communications & Mobile Computing Conference (IWCMC 2006),* 03 – 06 July, 2006, Vancouver, British Columbia, Canada

[33] B. Carpenter, S. Brim, "Middleboxes: Taxonomy and Issues", RFC 3234

[34] Paolo Bellavista, Antonio Corradi, Luca Foschini, "SIP-Based Proactive Handoff Management for Session Continuity in the Wireless Internet", *Proceedings of the 26th IEEE International Conference on Distributed Computing Systems Workshops (ICDSCW'06),* 04 – 06 July, 2006, Lisboa, Portugal

[35] Guangwei Bai and Carey Williamson, "The Effect of Mobility on Wireless Media Streaming Performance", *Proceedings of Wireless Networks and Emerging Technologies (WNET), pg 596 – 601,* July 2004

[36] B. Shen, W.-T. Tan, and F. Huve, "Dynamic Video Transcoding in Mobile Environments," IEEE MultiMedia, vol. 15, p. 42 51, January 2008

[37] H. Schwartz, D. Marpe, and T. Wiegand, "Overview of the Scalable Video Coding Extension of the H.264-AVC Standard", *IEEE Transactions on Circuits and Systems for Video Technology, Vol. 17(9),* September, 2007

[38] Ken Murray, Rajiv Mathur, and Dirk Pesch, "Adaptive Policy Based Access management

in Heterogeneous Wireless Networks", *Proceedings of the 6th Wireless Personal Multimedia Conference (WPMC),* Yokosuka, Japan, October, 2003

[39] Tsunehiko Chiba, Hidetoshi Yokota, Ashutosh Dutta, Dana Chee, and Henning Schulzrinne, "Performance Analysis of Next Generation Mobility Protocols for IMS/MMD Networks", *International Communications and Mobile Computing Conference (IWCMC),* Crete Island, Greece, 6 – 8, August, 2008

[40] K. S. Munasinghe, A. Jamalipour, and B. Vucetic, "Interworking between WLAN and 3G Cellular Networks: An IMS Based Architecture", *1st IEEE Australian Conference on Wireless Broadband and Ultra Wideband Communications (AusWireless 2006),* Sydney, Australia, 13 – 16, March, 2006

[41] Mehdi Mani, Noel Crespi, "Session Mobility between Heterogeneous Accesses with the Existence of IMS as the Service Control Overlay", *IEEE International Conference on Communications Systems (ICCS),* Singapore, October, 2006

[42] Bellavista *et al*, "An IMS Vertical Handoff Solution to Dynamically Adapt Mobile Multimedia Services", *IEEE Symposium on Computers and Communications (ISCC),* Marrakech, Morocco, 6 – 9 July, 2008

[43] J-M. Lee, M-J. Yu, S-G. Choi and B-S. Seo, "Proxy-based Multimedia Signalling Scheme Using RTSP for Seamless Service Mobility in Home Network", *IEEE Transactions on Consumer Electronics,* Vol. 54(2), May 2008

[44] N.H. Thanh, N.T. Hung, Tran Ngoc and T. Magedanz, "mSCTP-based Proxy in Support of Multimedia Session Continuity and QoS for IMS-based Networks", *2nd International Conference on Communications and Electronics*, Jun. 2008

[45] D. Waiting and R. Good, "Part II UCT IMS Client Overview", University of Cape Town IP Multimedia Subsystem Workshop notes, August 2007

[46] David Waiting, Richard Good, Richard Spiers and Neco Ventura, "Open Source Development Tools for IMS Research", *Proceedings of the 4th International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities,* Innsbruck, Austria, 18 – 20 March, 2008.

[47] Richard Good, Fabricio Carvalho de Gouveia, Shengyao Chen, Neco Ventura, and Thomas Magedanz, "Critical Issues for QoS Management and Provisioning in the IP Multimedia Subsystem," *Journal of Networks and System Management,Vol. 16(2),* pg129-144, June 2008

[48] http://www.antisip.com/doc/exosip2/eXosip_8h.html

[49] http://www.gnu.org/software/osip/

[50] http://www.antisip.com/doc/ortp/ortp_8h.html

[51] http://library.gnome.org/devel/gtk/

[52] http://www.gnu.org/licences/gpl-2.0.html

[53] R. Sparks, "The Session Initiation Protocol (SIP) Refer Method", RFC 3515, April 2003.

[54] O. Levin, "Suppression of Session Initiation Protocol (SIP) REFER Method Implicit Subscription", RFC 4488, May 2006

[55] J. Rosenberg, "Obtaining and Using Globally Routable User Agent (UA) URIs (GRUU) in the Session Initiation Protocol (SIP)", draft-ietf-sip-gruu-15, October 2007

[56] Sun Microsystems, "The SIP Servlet Tutorial, Part No: 820-307-10", January 2009

[57] http://jcp.org/aboutJava/communityprocess/final/jsr289/index.html

[58] http://jcp.org/aboutJava/communityprocess/final/jsr116/index.html

[59] Sun Microsystems, "Sun GlassFish Communications Server 1.5 Installation Guide, Part

No:820-4277-10”, January 2009

[60] Keoikantse Marungwana, Neco Ventura, “Presence and Real Time Streaming Protocol (RTSP) for Feature-rich Session Continuity in the IP Multimedia Subsystem (IMS)”, *Southern Africa Telecommunication Networks and Applications Conference (SATNAC), September 2009*

Appendix A: Accompanying CD-ROM

The thesis submission includes a CD-ROM containing the following information:

- Reference material – A collection of literature included in the reference list
- Published articles – A collection of published papers during the course of the research
- Thesis documents – A copy of this thesis document, and a separate copy of the abstract in Portable Document Format (PDF)
- Software – A collection of software development tools used in this thesis, with detailed instructions for reconstructing the test environment and evaluations