COIN: A Customisable, Incentive Driven Video on Demand Framework for Low-Cost IPTV Services

Submitted in fulfilment
of the requirements of the degree of

MASTER OF SCIENCE (MSC)

of Rhodes University

Ray Musvibe

Grahamstown, South Africa
March 2012

Declaration

I declare that the above thesis is my own unaided work, both in concept and execution, and that apart from the normal guidance from my supervisor, I have received no assistance except as stated in the text of this document.

This work is being submitted for the Master of Science Degree in Computer Science at Rhodes University. Neither the substance nor any part of the above thesis has been submitted in the past, or is being, or is to be submitted for a degree at this university or at any other university.

Ray Musvibe

....

2011

Acknowledgments

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

Alfredo Terzoli, my supervisor. Thank you for your guidance and dedicated commitment to the Centre of Excellence and its students. This work would not have been possible without your expert guidance and financial backing.

Mosiuoa Tsietsi, colleague. Thanks a lot Mos for all the technical assistance rendered through the years. Your expertise has proven invaluable not only my work, but all the researchers in the Convergence Research Group.

Zelalem Sentayehu, colleague. Thanks Zelalem for your input in the final stages of this write-up and technical input throughout my research. Much appreciated.

Researchers at the University of Cape Town for collaboration and releasing the UCT MCF-MDF to us for research purposes at the opportune time.

Moses Nkhumeleni, colleague. Thanks a lot Moses for your input regarding the Ericsson emulation server and the library you developed for billing Mobicents based applications. You work allowed me to add a significant component to my final architecture

Much thanks to my family and friends for the love and assistance throughout my M.Sc. journey in the past two years.

Last but not least, I thank my Lord and maker, Jesus Christ for the gifts of provision, knowledge and life, without which this work would not have been possible.

Synopsis

There has been a significant rise in the provision of television and video services over IP (IPTV) in recent years. Increasing network capacity and falling bandwidth costs have made it both technically and economically feasible for service providers to deliver IPTV services. Several telecommunications (telco) operators worldwide are rolling out IPTV solutions and view IPTV as a major service differentiator and alternative revenue source. The main challenge that IPTV providers currently face, however, is the increasingly congested television service provider market, which also includes Internet Television. IPTV solutions therefore need strong service differentiators to succeed.

IPTV solutions can doubtlessly sell much faster if they are more affordable or low-cost. Advertising has already been used in many service sectors to help lower service costs, including traditional broadcast television. This thesis therefore explores the role that advertising can play in helping to lower the cost of IPTV services and to incentivise IPTV billing. Another approach that IPTV providers can use to help sell their product is by addressing the growing need for control by today's multimedia users. This thesis will therefore explore the varied approaches that can be used to achieve viewer focused IPTV implementations.

To further lower the cost of IPTV services, telcos can also turn to low-cost, open source platforms for service delivery. The adoption of low-cost infrastructure by telcos can lead to reduced Capital Expenditure (CAPEX), which in turn can lead to lower service fees, and ultimately to higher subscriptions and revenue. Therefore, in this thesis, the author proposes a CustOmisable, INcentive (COIN) driven Video on Demand (VoD) framework to be developed and deployed using the Mobicents Communication Platform, an open source service creation and execution platform. The COIN framework aims to provide a viewer focused, economically competitive service that combines the potential cost savings of using free and open source software (FOSS), with an innovative, incentive-driven billing approach. This project will also aim to evaluate whether the Mobicents Platform is a

suitable service creation and execution platform for the proposed framework. Additionally, the proposed implementation aims to be interoperable with other IPTV implementations, hence shall follow current IPTV standardisation architectures and trends. The service testbed and its implementation are described in detail and only free and open source software is used; this is to enable its easy duplication and extension for future research.

Contents

List of Figures

List of Tables

Glossary

3GPP	3rd Generation Partnership Project.
ADSL	Asymmetric Digital Subscriber Line.
ANI	Application-Network Interface.
API	Application Programming Interfaces.
ARPU	Average Revenue Per User.
ATIS	Alliance for Telecommunications Industry Solutions.
AU	Application Usage.
AVP	Attribute Value Pairs.
CoD	Content on Demand.
CSCF	Call Session Control Function.
DVB	Digital Video Broadcasting.
ETSI	European Telecommunications Standards Institute.
FOSS	Free and Open Source Software.
GUI	Graphical User Interface.
HSS	Home Subscriber Server.
IETF	Internet Engineering Task Force.
IP	Internet Protocol.
IPTV	Internet Protocol Television.

MCF	Media Control Function.
MPS	Mobicents Presence Service.
NNI	Network-Network Interface.
OIPF	Open IPTV Forum.
OMA	Open Mobile Alliance.
OSA	Open Service Access.
OTT	Over-The-Top.
PSTN	Public Switched Telephone Network.
PVR	Personal Video Recorder.
ROI	Return On Investment.
RTP	Real-time Transport Protocol.
RTSP	Real Time Streaming Protocol.
SCF	Service Control Function.
SCTP	Stream Control Transmission Protocol.
SDF	Service Discovery Function.
SDO	Standards Development Organisation.
SDP	Service Delivery Platform.
SIP	Session Initiation Protocol.
SSF	Service Selection Function.
STB	Set Top Box.
TCP	Transmission Control Protocol.
UDP	User Datagram Protocol.
UE	User Equipment.
UNI	User-Network Interface.

URL	Uniform Resource Locator.
VoD	Video on Demand.
VoIP	Voice over IP.
WWW	World Wide Web.

Chapter 1

Introduction

The Internet has undergone tremendous growth since the first data packet was successfully sent across a computer network nearly half a century ago. The Internet has since evolved into a platform for a myriad of services including social networking, e-commerce, communication and entertainment; spurred by the advent and growth of the World Wide Web (WWW). Internet based services are often low-cost and can be accessed from a variety of devices, and through various access technologies such as broadband.

The rising popularity and use of low-cost Internet based services has however impacted negatively on several businesses, including the telco industry. Telcos traditionally relied on revenue from voice services, but the advent of free, Internet based services such as Voice over IP (VoIP) and instant messaging has greatly eroded these revenues. This has led to lower Average Revenue Per User (ARPU) across the telco industry [51][77][92][109].

Ironically, access to these Internet-based services is often provided by the telco operators themselves through access technologies such as ADSL (Asymmetric Digital Subscriber Line). It is feared that in the future, the role of telco networks will be of pure Internet Protocol (IP) transport bearers, while the delivery of services recognisable by consumers will be through over-the-top (OTT) service providers such as Google®, Skype® and Facebook® (Over-the-top is a general term for services that users can utilise over a network without any business or technology affiliations with the network operator to access such services). These companies are able to provide customers with sophisticated, low-cost, Internet-based services that compete with telco services. This creates a difficult environment for telcos as they must provide the same level of competitive innovation provided by OTT players while providing access to telecommunications and IP services [14].

In addition to these challenges, several key changes have been taking place within telco service architectures that require significant capital investments.

1.1 Service Architecture Evolution within the Telecommunications Industry

The traditional approach to service deployment in the telco Industry was through the "stovepipe" architecture. This essentially meant that applications were developed in an isolated environment to meet specific needs. The stovepipe architecture allowed for vertically integrated, application-specific service provider networks. Figure 1.1 illustrates the stovepipe or 'silo' architecture.

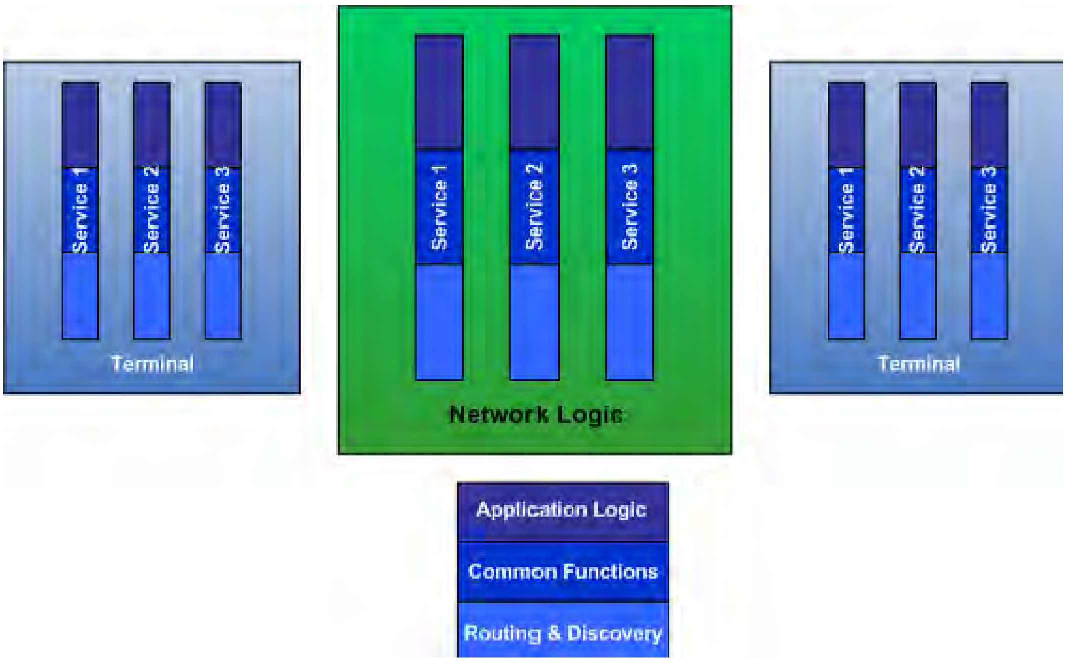

Figure 1.1: Telco stovepipe or 'silo' architecture [91].

Because each network had its own vendor-proprietary applications platform, there was limited interoperability and almost no shared applications. Service development over this 'stovepipe' architecture was also expensive because it required the use of specialised telco engineers and was often characterised by lengthy service development cycles. Many telco operators are still reliant on such architectures because they represent significant capital investments [91].

In the early 1980s, the need for competitive, shorter service development cycles led to the concept of Service Delivery Platforms (SDPs). SDPs generally refer to a set of components that are optimised to provide an environment for service creation, orchestration and execution [25]. SDPs also aimed to provide standardised components and interfaces that would grant network operators independence from vendor specific service solutions and proprietary network technologies.

From the mid-1990s, telco Application Programming Interfaces (APIs) were introduced to further accelerate service development cycles. Telco APIs aid application development by abstracting the signaling protocol details of the underlying networks. Several telco API standards have since been introduced, these include JAIN (Java APIs for Intelligent Networks), Parlay and the 3GPP Open Service Access (OSA) APIs. These provide access to network capabilities such as call control, conferencing and messaging [91].

One of the major shifts in telco service architectures came with the introduction of the NGN by the International Telecommunications Union (ITU). The first specification—NGN Release 1, was completed at the beginning of 2006 [74]. Among the key aims of the NGN was to provide a more flexible way of developing telecommunications services, as well as to provide reliable infrastructure to deliver broadband applications. The ITU-T defines the NGN as "a packet-based network able to provide telecommunication services and able to make use of multiple broadband, QoS-enabled transport technologies and in which service-related functions are independent from underlying transport related technologies. It enables unfettered access for users to networks and to competing service providers and/or services of their choice. It supports generalised mobility which will allow consistent and ubiquitous provision of services to users"[71][72].

In terms of its architecture, the NGN has two distinct strata: a transport stratum, and a service stratum. This architecture accelerates service deployment by allowing service providers to enable their service directly on the service layer without considering the transport layer. The NGN architecture also defines a Network-Network Interface (NNI), a User-Network Interface (UNI) and an Application-Network Interface (ANI) [136][155] as shown in Figure 1.2.

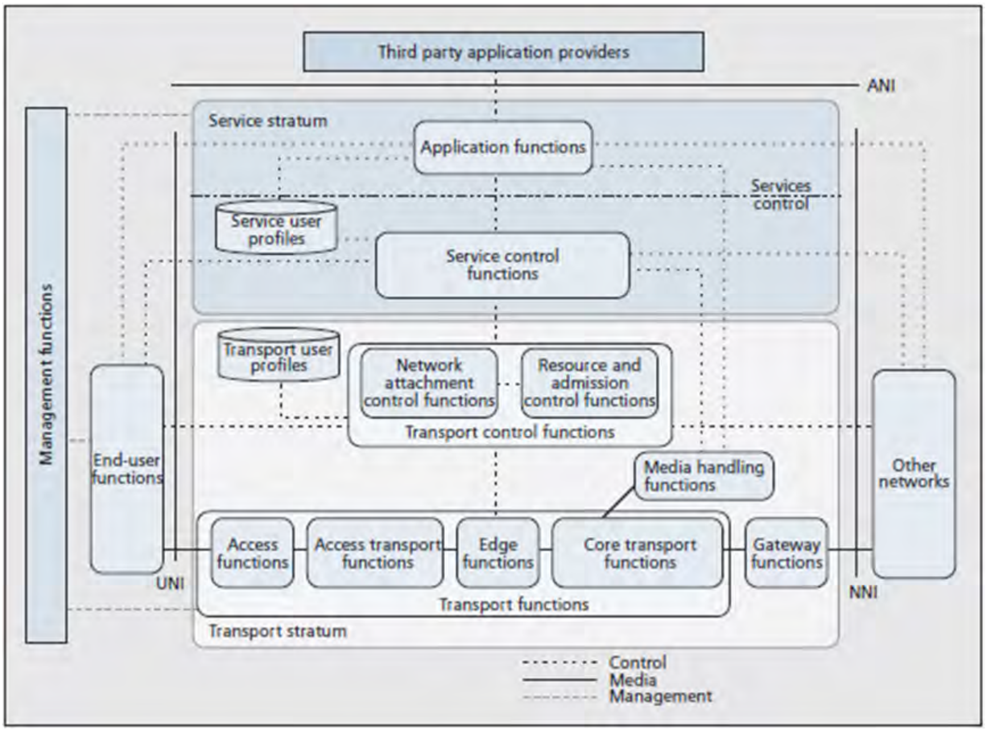

Figure 1.2: ITU's NGN Functional Architecture [155] .

The NGN represents the convergence of operator networks that include the Public Switched Telephone Network (PSTN), mobile carrier networks and the data network (the Internet) into an "all-IP" network. This creates possibilities for different companies to develop and deliver services across technology platforms, and for users to get access to new kinds of communication and media services. This has subsequently led to greater competition among service providers and reduced costs for access to telecommunications services. Figure 1.3 illustrates access network convergence within the NGN and associated services.

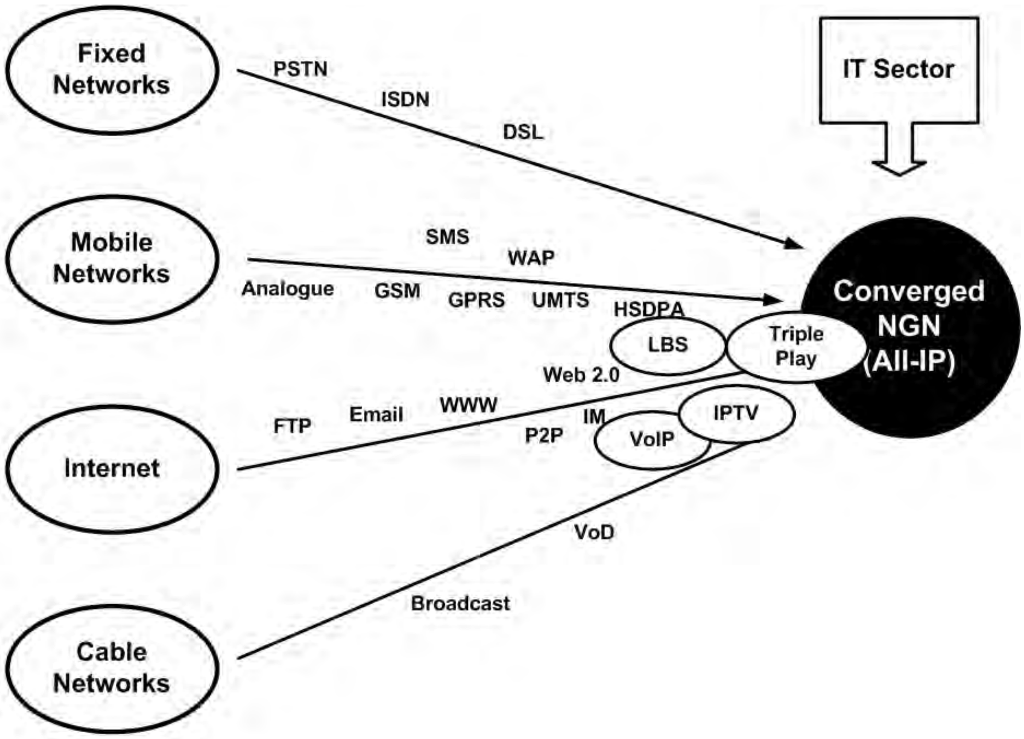

Figure 1.3: Access network convergence and emerging NGN services (Adapted from [87]).

With the convergence of these various access technologies through the NGN, there was a need for a suitable control layer in the converged IP core of the NGN that would facilitate fixed-mobile network convergence. In response, the IP Multimedia Subsystem (IMS) was introduced by the 3rd Generation Partnership Project (3GPP). IMS was initially specified under 3GPP R5 specifications in 2002 [27][157]. IMS is the unified telecommunication industry approach toward an "all-IP" network architecture that merges the Internet with the cellular and fixed telecommunication worlds. IMS facilitates convergence by providing seamless handovers between these dissimilar networks. IMS standardisation is regarded as a key component to NGN architectural design [2][88].

Among the major challenges to NGN/IMS deployment has been high costs [52][138]. As such, new revenue streams must be identified that can help finance these architectural changes and offset the problem of declining ARPU within the telco industry [151]. In this regard, several new and upcoming multimedia services such as IPTV and multimedia group communications have been identified as alternative revenue sources that can help shoulder these costs and boost ARPU.

1.2 IPTV Overview

The ITUs IPTV Focus Group broadly defines IPTV as “multimedia services, such as television, video, audio, text, graphics, and data, delivered over IP-based networks managed to provide the required level of quality of service (QoS), quality of end-user experience (QoE), security, interactivity and reliability “. VoD systems are a subset of IPTV services that allow users to select and view videos on demand.

IPTV subscribers generally access IPTV services through a set top box (STB). A STB is a device that connects the television to an external source or signal, turning the signal into content which is then displayed on the television screen or other display device. Typical IPTV infrastructure additionally consists of a head-end system, middleware, media servers and a Conditional Access/Digital Rights Management (CA/DRM) system. The head-end system is the facility at a local television providers’ office that originates and communicates television services to subscribers. Middleware is a general term for any platform that serves to "glue together" or mediate between two separate layers. A media server is software that makes audio and video files on a computer available on the network. CA/DRM are content protection systems that limit access of media to authorised users only. IPTV programming is viewed primarily through a television rather than a computer [116]. Figure 1.4 shows a typical IPTV network.

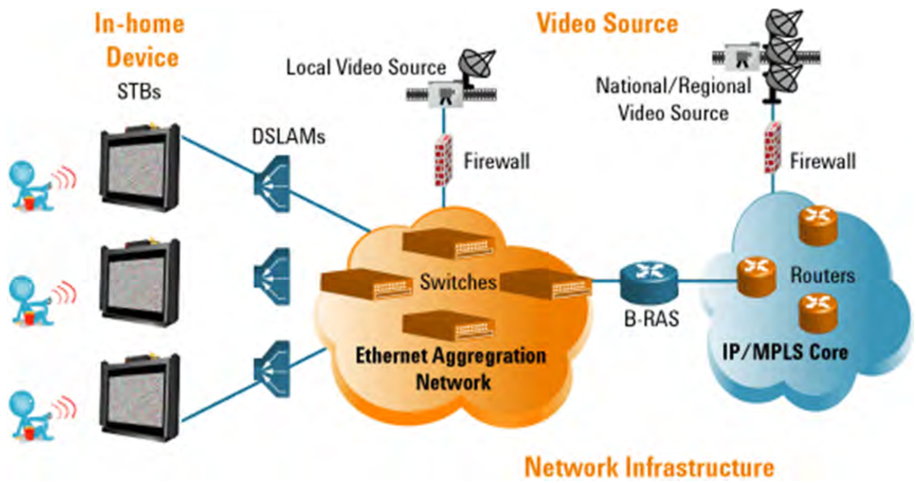

Figure 1.4: Typical IPTV network architecture [133]

Due to the growing interest in IPTV, there are currently numerous efforts toward the integration of these (IPTV) services into emerging NGN architectures. Standards based IPTV services will ensure interoperability between the different implementations [32][45].

Several Standards Development Organisations (SDOs) are currently involved in the formulation of IPTV standards. These include ETSI TISPAN, the ITU-T's IPTV Focus Group and the Open IPTV Forum [17][124]. (ETSI (European Telecommunications Standards Institute) set-up TISPAN (Telecommunications and Internet converged Services and Protocols for Advanced Networking) as the key standardisation body for creating NGN specifications).

Closely related to IPTV is Internet Television. Although IPTV and Internet Television both use IP, Internet Television is classified as a separate entity from IPTV. Internet Television refers to television services distributed via the Internet, not over dedicated private networks as IPTV. Viewing is generally through a computer, not an STB, and there are no QoS and QoE guarantees. Internet Television has however become very popular with services such as Hulu® in the United States and the BBC's iPlayer® in the United Kingdom [91]. The success and widespread use of Internet Television has been attributed to its ubiquitous availability and low-cost [132].

IPTV services are divided into three main groups: Broadcast (BC) services, Content on Demand (CoD) services and Personal Video Recorder (PVR) services. BC services consist of live TV and radio channels. Content on Demand (CoD) services are unicast services provided on demand, for example movies (Video on Demand) and music. PVR services are services with recording, pause or time-shift capabilities for live content [32][135].

One of the main drivers behind consumer interest in IPTV services is the potential for Content on Demand (CoD), or Video on Demand (VoD) offerings [91]. This is because unlike traditional broadcast television, VoD grants users access to video content whenever they desire it. Moreover, VoD grants users VCR-like controls, or trick modes such as PLAY, PAUSE, FAST-FORWARD and so on. It is therefore expected that an increasing amount of video content will be provided to the subscriber on-demand, rather than as a broadcast service [112]. This means that VoD will likely be a key service to telco IPTV service providers in the near future.

Another advantage that IPTV has over traditional broadcast television is that it allows for two way communication between the service provider and the IPTV user. Customers have the ability to send data upstream and receive data downstream, much like using an Internet connection. This allows highly interactive and sophisticated services such as personalised interactive program guides, interactive game shows, television shopping and content rating systems to be developed [91][151].

Realising this business opportunity, telcos worldwide are rolling out IPTV solutions. In

Europe, most of the major telcos are already offering Triple Play services to subscribers [100] (Triple Play is when data, voice and video are offered to a subscriber on a single telephone or cable connection). There were an estimated 40 million IPTV subscribers worldwide at the end of 2009 with the number of subscribers expected to grow considerably over the next few years [145]. Figure 1.5 shows the projected rise in IPTV subscriptions for the period from 2010 to 2014, at a projected compound annual growth rate of 25.3% [54].

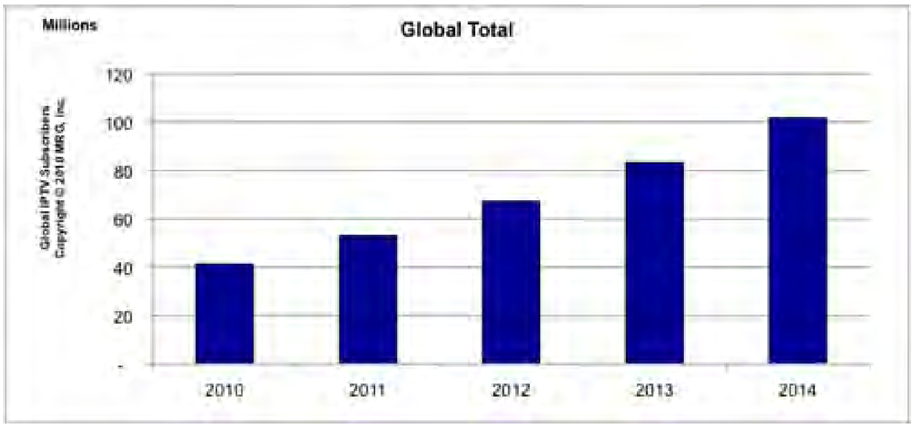

Figure 1.5: Global IPTV subscription forecasts [54].

IPTV therefore promises to provide a viable alternative revenue stream for telcos. The main challenge however, is that the IPTV market is becoming increasingly congested as more and more telcos offer IPTV services. In addition, telco IPTV solutions currently face stiff competition from traditional broadcast television, as well as low-cost Internet Television. To compound this situation, several top tier IT firms such as Apple®, Microsoft® and Google® are also introducing IPTV solutions [83]. In such a globally competitive environment, telcos can turn to service differentiators such as value added services, innovative billing approaches and implementations supporting strong viewer focus to help sell their product or service [47][66][94][141]. Other telco IPTV providers are exploring the use of advertisements to help pay for content and augment user subscriptions [28][85].

Deployment of services such as IPTV by telcos can be achieved through a service creation and execution platform such as the open source Mobicents Communications Platform.

1.3 Mobicents Overview

The Mobicents Communications Platform is an open source service creation and execution platform, it is the first and only open source implementation of JAIN SLEE 1.1 and SIP (Session Initiation Protocol) Servlet 1.1 specifications. The Mobicents project consists of the Mobicents JAIN SLEE Application Server, the Mobicents Media Server, the Mobicents Presence Service and Mobicents SIP Servlets [101]. Figure 1.6 shows a high level overview of the Mobicents Communications Platform.

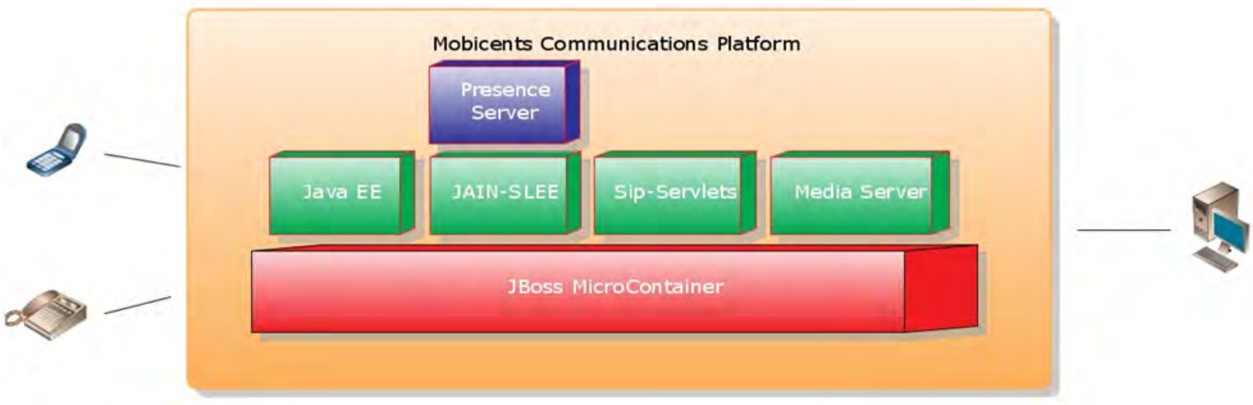

Figure 1.6: High level overview of the Mobicents Communications Platform [30].

Mobicents is built on top of the JBOSS Application Server using the JAIN API. Mobicents JAIN SLEE is an event driven application server that sits at the 'heart' of the Mobicents Communications Platform. SLEE (Service Logic Execution Environment) is an already existing telco standard for event driven applications [80].

Mobicents is designed to enable the creation, deployment and management of services and applications that integrate voice, video and data. As such, the Mobicents Communications Platform drives convergence by providing the infrastructure required to build NGN applications and services.

There is already growing interest in the Mobicents project from various parties. The project's forums are the 3rd most active on Java.net and several vendors and telco operators have already pledged their support to the Mobicents project [64]. These include Motorola®, Siemens®, Telecom Italia®, T-Mobile® and Open Cloud® to name a few.

The use of free and open source software (FOSS) in the development of commercially oriented services has now become commonplace. The main advantage of using FOSS in any enterprise is the low-cost of acquisition. Telcos have traditionally relied on proprietary solutions, but the adoption of FOSS by telcos can undoubtedly lead to lower CAPEX. This financial benefit is expected to filter down to clients in the form of lower subscription fees. Such cost savings could prove crucial in a market in which telco products and services compete with free or low-cost, Internet based services such as VoIP and Internet Television.

1.4 Problem Definition

The ongoing architectural changes taking place within telco networks such as NGN and IMS rollout need associated deployment applications that can justify these investments. IMS rollout has already been stalled partly because of the absence of an associated 'killer app' i.e. a service that will see widespread use and drive IMS adoption [86][138]. Many telco operators have turned to IPTV to fill this void [151][156]. This is mainly because IPTV revenues continue to rise globally. Global IPTV revenues were valued at US$17.8 billion in 2010 and forecast to grow to US$47.9 billion in 2014. This represents a compound annual growth rate of 28% [54]. IPTV thus presents a viable alternative revenue source, yet the increasingly saturated IPTV market and strong competition from low-cost Internet Television indicates that strong service differentiators will be needed in order to sell telco IPTV solutions.

Service differentiation can be achieved through implementations with strong viewer focus. Today's users of multimedia services and applications increasingly want more control [31][65][151]. Several successful communications applications today make use of personalisation and customisation to address this challenge. Throughout this thesis, the term *personalisation* shall refer to the process in which a service provider makes intelligent "guesses" on the type of content the client may want to see as defined in [38]. These "guesses" are usually based on previously collected user data and viewing habits, although this raises privacy issues. The term *customisation* shall refer to the process in which a user proactively decides their user experience based on preferences or taste as defined in [107]. Recent studies show that customisation is more user centered than personalisation [38][107]. This is highlighted in Figure 1.7. Substantial research is currently being done on IPTV service personalisation [124][151][152], but not much has been done concerning

IPTV customisation. IPTV customisation could prove to be a key value added IPTV service and hence substantial research needs to be done on its implementation.

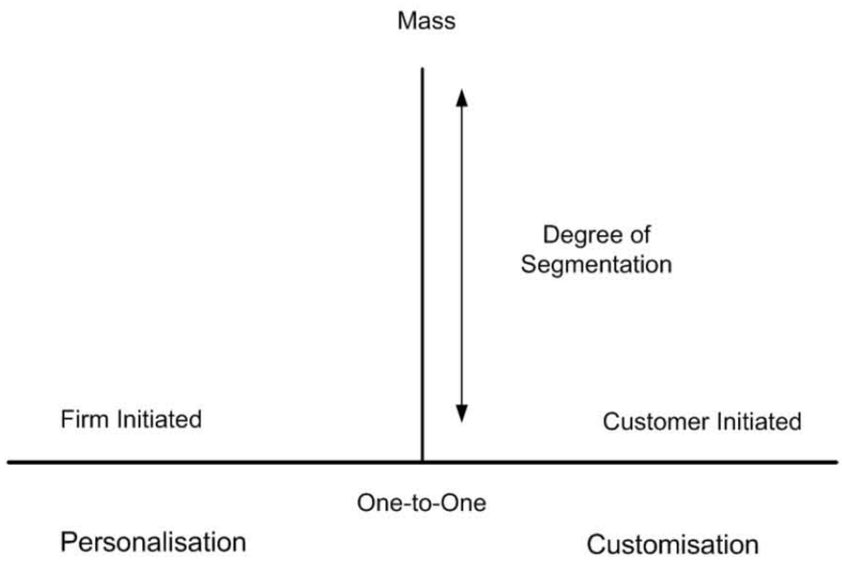

Figure 1.7: Personalisation vs. Customisation [107].

Another way of making IPTV services more attractive to consumers is by making them more affordable, or low-cost. Traditional broadcast television achieved this by using advertisements. Advertising has already been used by telcos in the past to help pay for a service or content. For example, the German mobile carrier E-Plus® offers a cheaper mobile service to subscribers in exchange for receiving advertisements. Customers who sign up also get free minutes and texts as a reward for receiving advertisements [59][151]. Similar approaches have already been proven to achieve better than average returns for telcos [58][147]. Telco IPTV providers can therefore adopt similar advertiser driven incentive models to lower service fees and as a result, boost service subscriptions and revenue.

IPTV also affords advertisers the opportunity to present targeted advertisements. Targeted advertising refers to the presentation of advertisements to a target audience who are more likely to buy the product being sold. This is currently being achieved by collecting data from IP transactions between the user and the IPTV provider, and using this to build (personalised) user profiles. These profiles are then used in targeted advertising and other value added services [151], although privacy issues about this has been raised. Targeted advertisements are considered user focused as irrelevant advertisements are not shown to users. Targeted advertisements have also been associated with higher returns for advertisers [28][151]. Hence, the use of targeted advertisements would be of benefit to all stakeholders involved, i.e., higher revenues for telcos, a better user experience for customers and quicker return on investment (ROI) for advertisers. Hence the advertiser driven incentive VoD framework being proposed should endeavor to incorporate targeted advertisements into the implementation framework.

Low-cost telco services and products can also be realised through the use of FOSS. The use of FOSS can lead to significant cost savings, particularly in terms of CAPEX due to the low cost of acquisition. It is however difficult to conclude for certain that the adoption of FOSS will result in overall cost savings when we consider the generally high life cycle costs of commercial software implementations. The life cycle cost for using FOSS is sometimes higher because there is often the need to bring in developers with specialised skills to handle any software related challenges that might arise. These kind of challenges generally do not exist or are more contained when using proprietary software since the software vendor responsible for the product generally provides software support and updates to thier clients.

In addition to potential cost savings, open architectures allow for easier and faster service development [34]. As such, telcos should explore the available FOSS applicable to their enterprises, this could prove crucial in offsetting low-cost service offerings from OTT providers. In this regard, Mobicents JAIN SLEE is a free and open source platform optimised for low latency and high throughput event processing. It is the first open implementation of SLEE, an already existing telco standard for event driven applications. It can be used as a high performance core engine for IMS. The proposed IPTV platform will hence be developed using Mobicents JAIN SLEE to evaluate its potential as a viable telco SDP.

It would also be ideal for the proposed Mobicents-based VoD service to be in line with IPTV standards as this would ensure interoperability with other implementations. Though this research work does not use an IMS control layer, the proposed implementation will be built according to ETSI TISPAN IMS-based IPTV specifications. This is possible because full fledged SDPs and IMS have several overlaps in functionality, hence SDPs can be deployed without IMS support. In most cases, operators will likely deploy an SDP before migrating to a full IMS [18]. This approach will allow a telco operator with or without an IMS control layer to deploy a standards based solution. The author felt this was an intuitive approach for a market in which many telco operators are still reluctant to adopt IMS, although IMS rollout is ongoing. The proposed framework is therefore designed to allow for easy integration with an IMS layer, although no such layer is used in the proposed framework.

1.5 Thesis Objectives

This thesis aims to overcome the above mentioned problems and incentivise telco IPTV solutions through targeted advertising. Furthermore, this thesis aims to investigate and analyse the suitability of Mobicents as a VoD SDP. More specifically, the objectives of this thesis are to:

- Analyse the feasibility of developing an interoperable, standards based VoD service using Mobicents JAIN SLEE as an SDP, i.e., investigate current IPTV standardisation efforts and trends, and hence evaluate whether a Mobicents based VoD service could provide the required service functionality that falls in line with current IPTV standardisation efforts.
- Analyse the feasibility of integrating an advertising platform into an IPTV solution from a telco perspective, i.e., to investigate related work on generating revenue through provisioning advertising of third party products and services.
- Propose and develop the CustOmisable, INcentive (COIN) driven Video on Demand (COIN) framework for a Mobicents based VoD service that grants users more control over the service and presents targeted advertisements in a manner that is non-invasive and does not hinder the viewer experience.
- Analyse any impact the proposed framework and its implementation might have on the user's QoE by conducting tests on an evaluation testbed.
- Since the target platform aims to provide a low cost IPTV service, this thesis will make an estimate of the cost of deploying such an IPTV solution over the Mobicents platform.

1.6 Scope and Limitations

The scope of this thesis is limited specifically to telco IPTV development and deployment. It looks at the various options that IPTV providers have in providing competitive IPTV solutions, including the use of low cost infrastructure for service development and generating revenue through advertising third party products and services. This thesis will also specify the requirements that are necessary for the telco to provide the proposed service.

Although this thesis presents an overview of IPTV in general, it specifically focuses on the VoD subset of IPTV. Additionally, this thesis will not focus on Content Access/Digital Rights Management controls, but will assume that the IPTV provider already has the content and issues of digital rights have been handled.

In terms of architecture, the proposed framework is built according to the ETSI TISPAN IMS-based IPTV architecture, but will not be using an IMS control layer. This thesis will use the standardised IMS-based IPTV components in the proposed framework, but the functions that IMS performs in IMS-based IPTV such as registering clients and session control will be handled by Mobicents.

The Mobicents Platform used for the development and deployment of the proposed services consists of the JAIN SLEE server, SIP Servlets container, the Media server and the Presence server, but most of the implementation of the proposed framework will be built over the JAIN SLEE server. This is to take advantage of JAIN SLEE's event driven architecture. The proposed framework will however make use of the the other servers if the need should arise.

Finally, to quantify and measure users' QoE of the IPTV service, service latency will be the key parameter. This thesis will also analyse call flow efficiency in setting up IPTV sessions and evaluate any aspect of the proposed framework that may affect the QoE. Underlying factors that affect QoE from the transport layer, such as available network bandwidth, buffering techniques, media coding and QoS classes, are not considered in this thesis. It is assumed that the network has sufficient bandwidth to cater for the VoD service.

1.7 Thesis Outline

The remainder of this thesis is outlined below:

Chapter 2 discusses relevant literature concerning IPTV, and the role that value added services and advertisers are expected to play. The chapter also looks at ongoing IPTV standardisation efforts and trends, and concludes with a detailed examination of the Mobicents Communications Platform.

Chapter 3 presents design considerations and architectural requirements for the proposed COIN framework. This chapter discusses the impact that the proposed framework might have on the three main stakeholders; namely the telco, the user and the advertiser. The chapter concludes by providing insights into the required IPTV functional entities as specified in IMS-based IPTV specifications as well as other components that were used in efforts to provide an implementation of the proposed framework.

Chapter 4 presents the proposed COIN framework. This chapter will also provide detailed system design and analysis of three evolving implementation architectures set-up using the Mobicents Platform in developing the COIN framework.

Chapter 5 provides an evaluation of the final architecture. Results of experiments conducted on the testbed are presented and analysed.

Chapter 6 presents conclusions that are derived from this research work. The conclusions answer the research questions which were introduced earlier, based on the work investigated in the literature review and on the evaluation of the proposed framework presented in chapter 5. Recommendations for future work are also presented.

Chapter 2

Background and Literature Review

The previous chapter discussed challenges currently facing the telco industry and the role that IPTV is expected to play. The author also introduced targeted advertising and the Mobicents Communications Platform. The author then highlighted the need for telcos to adopt low-cost infrastructure and discussed the need for value added IPTV services in a highly competitive television service provider market. This chapter looks further in detail at IPTV value added services and examines the role that advertisers can play in IPTV services. This chapter will also discuss in detail current IPTV standardisation efforts and trends, particularly looking into the ETSI TISPAN IMS-based IPTV specifications, on which the proposed implementation framework will be based. The chapter will conclude with an in depth look into the architecture of the Mobicents Communications Platform.

2.1 Value Added IPTV Services and Advertising

IPTV deployments began a few years ago with great expectations and encouraging projections. IPTV was being hailed as the saviour of fixed-line network operators and Internet Service Providers (ISP). Apart from promoting broadband usage in general, IPTV would give existing telephone subscribers an incentive to upgrade their service to higher-quality broadband connections. IPTV also promised new and attractive features such as VoD and enhanced interactivity. In spite of all this, the number of IPTV deployments in some countries has not reached even a fraction of projected numbers [89]. This has mainly been attributed to strong competition in the television service provider market. In many of the markets in which IPTV is being introduced, there are other, well established television technologies already in place, these include Cable TV and Satellite TV. These technologies generally command large viewership bases. Unless IPTV establishes a unique selling proposition (USP), it is unlikely to become a popular choice for television viewing (A USP is the factor or consideration presented by a seller as the reason that one product or service is different from and better than that of the competition). The promise that IPTV offers will therefore most likely be realised when network operators and service providers come up with service differentiators or value added services that can capture the attention of consumers.

According to [70], value added services refer to advanced or additional services a content provider offers to possibly increase their revenues, or make their offering more competitive. An example of this is the movie recommendation engine for Netflix®, an American corporation that offers on-demand video streaming over the Internet. The recommendation engine automatically recommends movies based on customer's viewing habits. According to Netflix®, 60 % of subscribers who use the recommendation engine add these suggested movies to their viewing queues [153]. Other value added IPTV services that have been proposed include;

- Internet browsing over TV.
- IPTV-based converged services, such as being able to handle a call on your TV and accessing E-mail on TV.
- Real time interactivity e.g. with game shows and advertisements.
- Video conferencing on TV [6].

IPTV implementations with strong viewer focus can also be used to provide service differentiation. Customisation and personalisation have been key features of many new and upcoming communication services. Many successful multimedia applications today allow customers to either proactively decide their own user experience based on their preferences, or the service itself can make intelligent guesses on what the user might be interested in, all this in an attempt to make the user experience or QoE more pleasant [151][152]. An example of personalisation is the Netflix® movie recommendation engine mentioned earlier in this section.

There is currently a lot of research activity being done concerning IPTV personalisation. Researchers in [140] proposed a web-enhanced personalised IPTV service while researchers in [48] proposed a personalised IMS-based IPTV service. Various SDOs are also looking into personalised IPTV services. These include ETSI TISPAN, which will release personalised IPTV specifications under TISPAN Release 3 [124]. According to [6], IPTV personalisation falls into stage three of the potential stages of ongoing IPTV services evolution. This is shown in Figure 2.1.

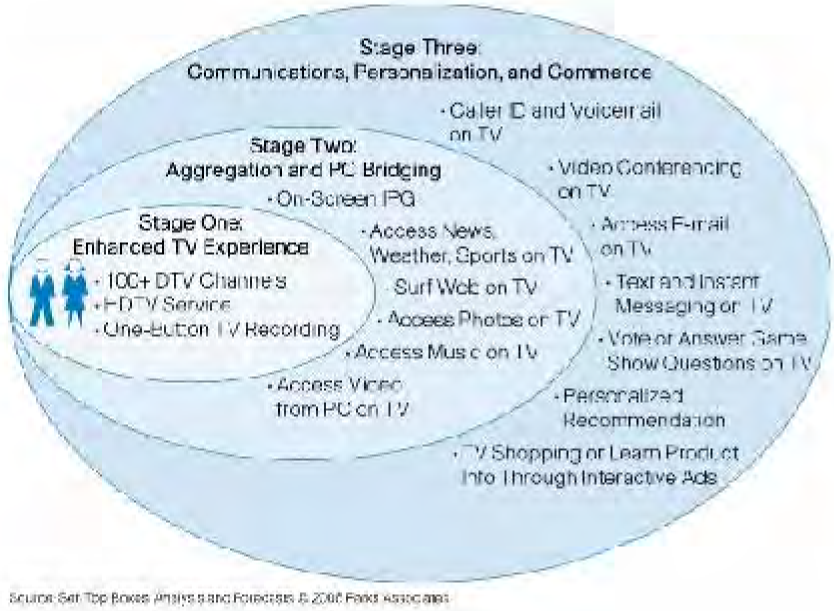

Figure 2.1: Potential stages of IPTV services evolution [6]

2.1.1 IPTV and Advertising

In 2006 alone, over $350 billion was spent on advertising [9]. This included print, television and digital marketing. Advertising has already proven it can boost revenue in a wide spectrum of business ventures; IPTV hence stands to benefit from a union with advertisers. The media industry has historically been partly funded by advertisers, therefore relying on subscriptions or pay-per-view models will most likely be unsustainable. Many telco IPTV providers are therefore experimenting with opt-in advertising plans to help fund content [28].

The cost of the IPTV service is one of the key issues for IPTV at the moment. Naturally consumers want the highest quality of service at the lowest possible price, while IPTV service providers look to break even over the shortest possible time frame. It is however challenging for IPTV service providers to offer low cost IPTV services because of associated costs of deploying an IPTV service. Service provider costs include transcoding and distribution equipment along with the cost of STBs to be deployed in large numbers at the consumer end. In addition, service providers have to pay royalties to middleware vendors and for Conditional Access/Digital Rights Management (CA/DRM) [89]. IPTV providers can bridge this gap by allowing advertisers to help pay for the content.

Traditional television broadcasters used this model and now even Internet video streaming providers such as Hulu® and YouTube® have teamed up with advertisers to boost revenues. Hulu® provides advertiser supported streaming video of TV shows and movies from NBC®, Fox®, ABC®and other networks. The advertisements can't be avoided but the service provides viewers a chance to skip commercials after viewing a portion of the advertisement. Most of Hulu®'s revenues come from advertisers and in 2009, the company posted revenues of about 120 million (US) dollars [8]. YouTube introduced video advertisements in 2007. The video advertisements are designed to be as unobtrusive as possible since disruptive advertisements tend to be less effective [85]. Text boxes appear over a video 15 seconds into playback. The advertisement occupies about 20% of the video window and disappears after 10 seconds if it is not clicked on. The text advertisement is related to the video being watched and contains a link which is connected to another video advertising a product [151].

The major drawback with traditional broadcast television advertising was that it involved advertisers deliberately interrupting the content a user really wanted to see with the advertisement. Studies have shown that a lot of consumers are found to "tune in but turn off" when it comes to traditional media advertising when advertisements are intrusive, irrelevant and un-entertaining [123]. Re-evaluation of advertising models has shown that it is not just about delivering the advertisements efficiently that counts, but rather, the emphasis should be on delivering the right advertisements to the right consumers, and at the right time [151]. In this regard, IPTV offers several alternative forms of advertising:

2.1.1.1 Targeted Advertising

As discussed in the previous chapter, targeted advertising refers to the presentation of advertisements to a target audience who are more likely to buy the product being sold. Two approaches can be used to achieve this. The first and most widely used approach is through personalised advertising. Personalised advertising is when the service provider makes intelligent "guesses" on the type of advertisements the client may want to see. These "guesses" are based on previously collected customer data and viewing habits [38].

This obviously raises privacy concerns. Google's AdSense is an example of personalised advertising [152].

The second approach is through customised advertising, which is when the consumer proactively decides the nature of advertising material that will be available for viewing based on taste. This is non-invasive and gives more control of the service to the user [107].

2.1.1.2 On-Demand Advertising

On-Demand advertising addresses the new advertising opportunities emerging from VoD, digital video recorder (DVR) and other forms of non-linear television. On-Demand advertising is a user oriented form of advertising in which users dictate when and where they want to view advertisements [3]. On-Demand advertising is regarded as non invasive and expected to result in a better viewer experience [152].

In the next section, the author discusses IPTV standardisation efforts and the associated IPTV service architectures.

2.2 IPTV Standardisation Efforts

The growing interest in IPTV services and its market potential has led to numerous IPTV standardisation efforts from various SDOs. Standardisation is necessary to ensure interoperability between different implementations [12]. Currently, most IPTV implementations are built on vendor-specific platforms without integration with NGN subsystems [100].

There are three main bodies working towards the standardisation of IPTV: ITU-T IPTV Focus Group (ITU-T IPTV FG), ETSI TISPAN and the Open IPTV Forum (OIPF). Other notable IPTV standardisation efforts are being conducted by the Digital Video Broadcasting (DVB) group, The Alliance for Telecommunications Industry Solutions (ATIS) and the Open Mobile Alliance (OMA).

2.2.1 ITU-T IPTV Focus Group (ITU-T IPTV FG)

The mission of the ITU-T IPTV FG is to coordinate and promote the development of global IPTV standards, taking into account the existing work of the various ITU study groups, as well as other SDOs, Fora and Consortia [76]. The ITU-T IPTV FG has also begun work on the integration of IPTV into NGN architectures. Two approaches for this are currently being studied by the ITU-T IPTV FG, one approach is a dedicated IPTV subsystem, and the other is an IMS-based IPTV architecture. ITU-T IPTV FG has also proposed a non-NGN IPTV structure [146].

2.2.2 ETSI TISPAN

TISPAN was set up by ETSI in 2003 as the key standardisation body in creating NGN specifications, including NGN based IPTV [90]. NGN based IPTV was introduced under TISPAN Release 2 in 2008. Potential IPTV enhancements such as P2P IPTV, personalisation and user generated content will be released under TISPAN Release 3 [124]. Similar to the ITU-T IPTV FG, TISPAN–NGN has proposed a dedicated IPTV subsystem and an IMS-based IPTV architecture[134][146]. Figure 2.2 shows the respective IPTV specifications for ETSI TISPAN dedicated IPTV subsystems and IMS-based IPTV.

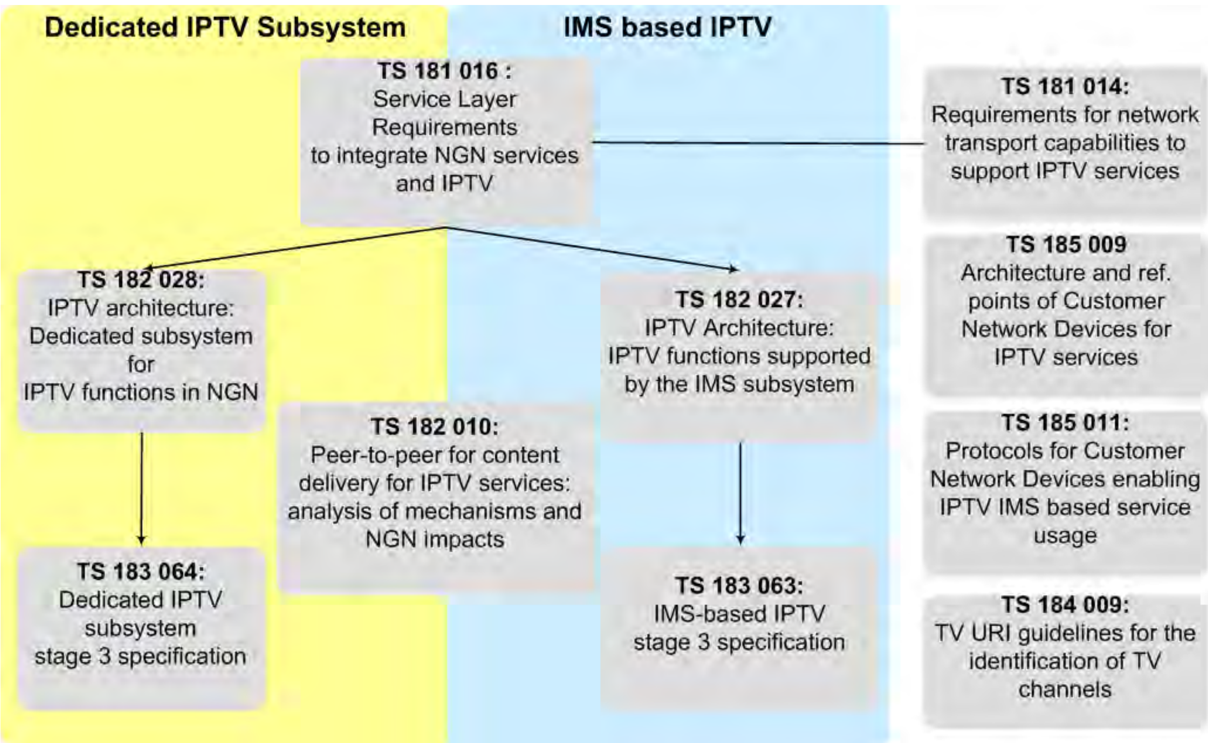

Figure 2.2: ETSI TISPAN IPTV Specifications [142].

2.2.3 Open IPTV Forum (OIPF)

The OIPF is a consortium of companies (about 60 members to date) that has formulated and published an open IPTV standard. This group consists of major players in IT (Information Technology) services such as Intel®, Panasonic® and Philips® and has participation from major operators such as British Telecom®, Telecom Italia® and T-Mobile® in Germany [17][110]. The Open IPTV Forum aims to provide interoperable E2E (end-to-end) specifications for IPTV and to increase the QoE by blending TV services with value added services [111].

2.2.4 Digital Video Broadcasting (DVB)

Digital Video Broadcasting (DVB) is a consortium of around 250 broadcasters, manufacturers, network operators, software developers, regulatory bodies and others in over 35 countries. It aims to develop open technical standards for the global delivery of digital television and data services [35]. The standard DVB over IP (DVB-IPI) was developed by DVB to provide a set of technical specifications that cover the delivery of DVB MPEG-2-based services over bidirectional IP networks, including specifications of the transport encapsulation of MPEG-2 services over IP and the protocols to access such services [139].

2.2.5 Alliance for Telecommunications Industry Solutions (ATIS)

ATIS is an SDO that develops technical and operational standards for the telecommunication industry. ATIS set up the IPTV Interoperability Forum (IIF) to develop standards that enable the interoperability, interconnection, and implementation of IPTV systems and services, including VoD and interactive TV services [100].

2.2.6 Open Mobile Alliance (OMA)

OMA is an SDO that develops open standards for the mobile phone industry. OMA is working towards IP-based mobile broadcasting networks. OMA introduced the concept of mobile broadcast services enablers to address functional issues that are generic enough to be common to many broadcast services. These service enablers can be defined and implemented in a bearer-independent way [16][100].

2.3 NGN IPTV Architectures

As mentioned in the previous section, there are two main IPTV architectures currently under investigation by the relevant SDOs for NGN deployment, these are the dedicated IPTV architecture and an IMS-based IPTV architecture.

2.3.1 Dedicated IPTV Architecture

Dedicated IPTV/VoD architectures make use of dedicated subsystems to deliver IPTV functionality to users. Dedicated IPTV/VoD subsystems are implemented in fully-managed networks similar to broadband cable networks. QoS is guaranteed. Users presently need to install specific vendor-defined STBs to access the IPTV/VoD services [152]. Current standardisation efforts focus on integrating these existing market solutions into an NGN environment and standardising IPTV functional entities and reference points. Research has shown that a dedicated IPTV subsystem can offer performance improvements when compared to an IMS-based IPTV subsystem because its architecture does not involve the extra control messages that are required to communicate with the IMS subsystem [95].

2.3.2 IMS-based IPTV Architecture

IMS was introduced by the 3GPP as the architectural subsystem dedicated to control and provide multimedia services over packet based core networks within third generation mobile networks [2][16]. The IMS constitutes the control plane in the converged IP core of NGN networks. Since NGN networks are heterogeneous, the IMS layer can be accessed from various access networks [151].

The key elements of the IMS core are the Call Session Control Functions (CSCFs) and the Home Subscriber Server (HSS). A CSCF can be generalised as a SIP server and the HSS can be generalised as a user profile database. There are three CSCFs that form the IMS, these are the Proxy-CSCF (P-CSCF), Interrogating-CSCF (I-CSCF), and the Serving-CSCF (S-CSCF). IMS uses SIP for signaling, session, and mobility management [122]. The P-CSCF is the first to receive all SIP requests, it forwards these requests to the S-CSCF via the I-CSCF. The task of the I-CSCF is selecting the appropriate S-CSCF by checking with the HSS. The S-CSCF behaves as a SIP registrar server and performs user registration and handles session management in the IMS [27][72][103].

An IMS-based IPTV platform is a service platform that is able to provide IPTV services managed by the IMS core subsystem. An IMS-based IPTV platform still contains all the components found in typical IPTV networks such as media servers, a CA/DRM system and STBs. IMS-based IPTV makes use of embedded IMS functionality such as Authentication, Authorisation and Accounting. Other advantages of deploying IPTV over IMS include roaming support, QoS control and unified billing [99][100].

The IMS-based IPTV architecture presented in this section is based on ETSI TISPAN IMS-based IPTV architecture since it is the most widely adopted IMS-based IPTV framework in evolving standards and research. It has been developed in conjunction with other standardisation bodies such as the ITU-T IPTV Focus Group, ATIS IPTV Interoperability Forum and DVB to ensure interoperability between vendor specific IPTV systems [152].

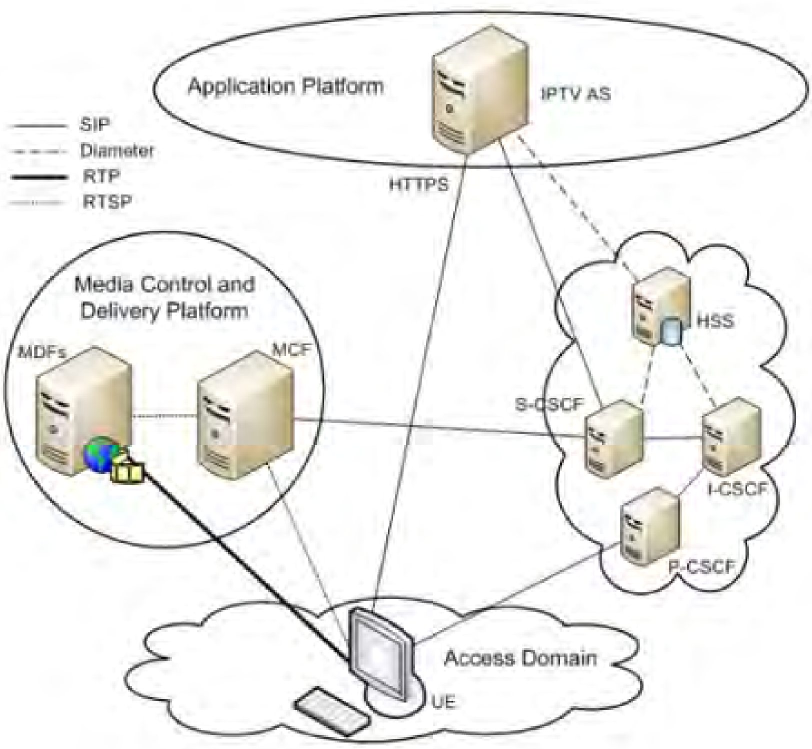

Figure 2.3: ETSI TISPAN IMS-based IPTV architecture [151].

Figure 2.3 illustrates the architecture of the ETSI TISPAN IMS-based IPTV system, including the relevant entities, reference points and protocols used in IPTV. Detailed descriptions of the relevant entities, reference points and protocols used in the ETSI TISPAN IMS-based IPTV system are provided in Appendix A. The User Equipment

(UE) is a functional entity that provides the user with access to IPTV services. In this architecture, the UE interacts with four entities: the core-IMS, the Media Control Function (MCF), the Media Delivery Function(MDF) and the IPTV Application Server.

The MCF, in conjunction with the MDF, are in charge of media control and delivery to the UE. The MCF and MDF can be co-located and they are collectively referred to as the Media Function (MF). The MCF and the MDF have a master-slave relationship in which the MCF is responsible for controlling the protocol signaling that establishes media connections. As such, the MCF can control media flow and monitor the MDF's status while the MDF handles the delivery of media to the user and reports back to the MCF. The IPTV application server is presented in Figure 2.3 as a single entity for simplicity, but in fact consists of three separate servers or functions, namely a Service Control Function (SCF), a Service Discovery Function (SDF) and a Service Selection Function (SSF). The functions of these three logical entities are as follows;

a. Service Control Function (SCF): The SCF is responsible for performing session authorisation for the purpose of granting or denying a user request for the initialisation of a new session or modification of an existing one. The SCF also performs credit control and is responsible for the selection of the relevant media functions [42].

b. Service Discovery Function (SDF): The SDF is a SIP application server that is responsible for generating the service attachment information, as well as to provide a means of discovering personalised IPTV services in the IMS network. The service attachment information consists of Service Selection Function (SSF) addresses in the form of URIs and/or IP-addresses which the UE can use to contact an appropriate SSF.

c. Service Selection Function (SSF): The SSF allows the UE to retrieve the service selection information, which is essentially a list of available services that the UE can then browse and select. This information is usually presented to the user in the form of an Electronic Program Guide (EPG), which is a graphical on-screen display of the information. The SSF can also support personalised service selection [42][91].

According to ETSI TISPAN IPTV specifications, Service Discovery and Selection (SD&S) can be achieved in (multicast) Push or (unicast) Pull mode using the SIP protocol. In Push mode, service discovery information is sent to the UE from the SDF in a SIP message once the UE registers with the IPTV service provider. In Pull mode, the UE sends a SIP SUBSCRIBE request to the SDF. The SDF responds with a SIP NOTIFY message containing the service attachment information as an XML body of the response message. Figure 2.4 shows SD&S using Pull mode and Figure 2.5 shows SD&S using Push mode in an IMS-based IPTV environment [42][91].

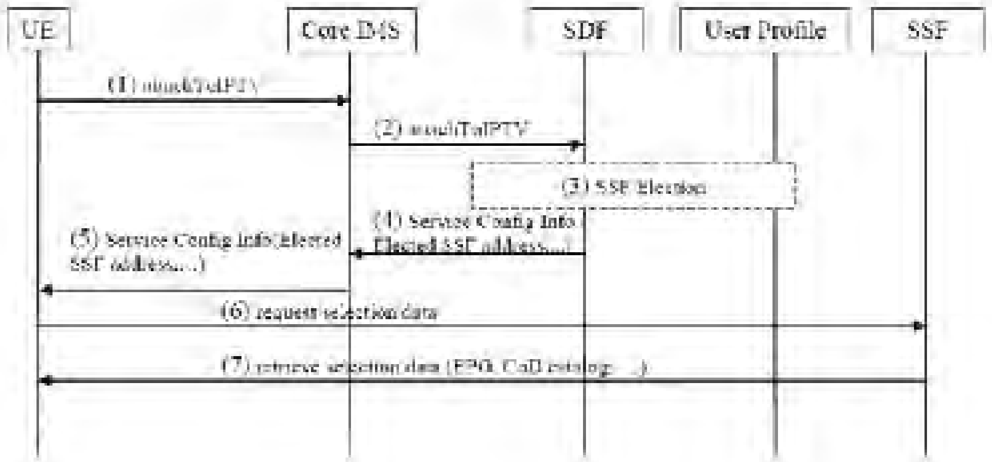

Figure 2.4: ETSI TISPAN service discovery and selection in Pull mode [144].

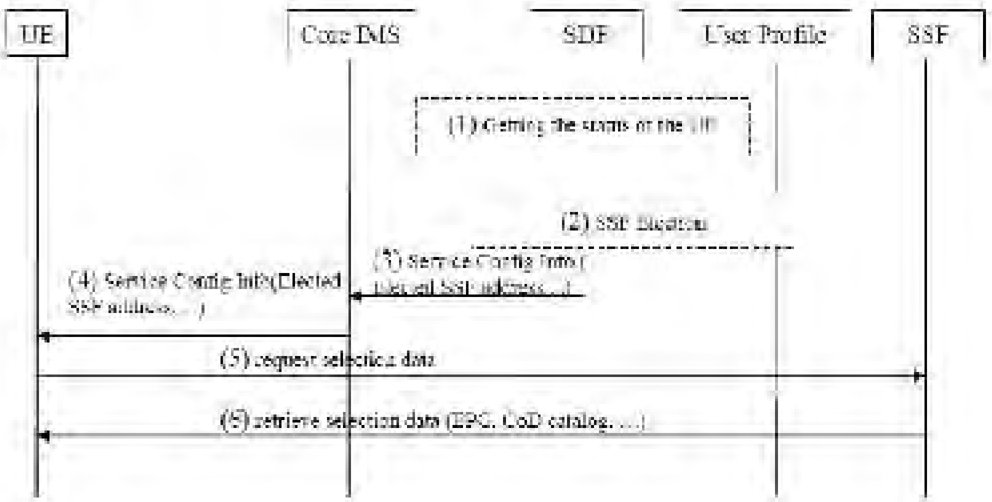

Figure 2.5: ETSI TISPAN service discovery and selection in Push mode [144].

The Pull mode of SD&S is also possible through HTTP as specified under ETSI TS 102034 [36][40].

2.4 Related Protocols

Several key protocols are specified by the various SDOs for use in standards based IPTV architectures. This section discusses the major protocols and their roles in ongoing IPTV standardisation efforts.

2.4.1 Session Initiation Protocol (SIP)

SIP is an application layer signaling protocol for creating, modifying, and terminating media sessions over IP. These sessions include Internet telephone calls, video conferencing, streaming multimedia distribution and instant messaging. The latest version of the specification is RFC 3261 from the IETF SIP Working Group [122]. SIP is currently the predominant signaling protocol in the NGN [49][50][148]. It is a text-based protocol designed to be independent of the underlying transport layer protocol; and can run on Transmission Control Protocol (TCP), User Datagram Protocol (UDP) and the Stream Control Transmission Protocol (SCTP). SIP messages are syntactically composed of a start line, one or more headers, and an optional body.

There are two different types of SIP messages, namely requests and responses. RFC 3261 [122] defines the following SIP request methods;

a. REGISTER: This method is used by a User Agent (UA) to notify the SIP provider of its current IP address and the URLs for which it would like to receive calls.

b. INVITE: Used to establish a media session between UAs.

c. ACK: Confirms reliable message exchanges.

d. CANCEL: Terminates a pending request.

e. BYE: Terminates a session between two users in a conference.

f. OPTIONS: Requests information about the capabilities of a caller, without setting up a call.

According to RFC 3261, SIP responses fall into one of the following categories:

a. Provisional (1xx): Request received and being processed.

b. Success (2xx): The action was successfully received, understood, and accepted.

c. Redirection (3xx): Further action needs to be taken by the sender to complete the request.

d. Client Error (4xx): The request contains bad syntax and cannot be fulfilled at the server.

e. Server Error (5xx): The server failed to fulfill a valid request.

f. Global Failure (6xx): The request cannot be fulfilled at any server [122].

SIP has also been extended to various application domains. These extensions include the Session Initiation Protocol for Instant Messaging and Presence Leveraging Extensions (SIMPLE), which is used for instant messaging and to carry presence information [78]. SIP extensions have also been defined for IMS signaling by the 3GPP [43].

SIP is currently the predominant signaling protocol in most standards based IPTV implementations. In these implementations, SIP is used for a variety of procedures, such as service attachment and session initiation [42]. Various IETF Internet drafts have also been presented that evaluate the use of SIP as a streaming control protocol [98][150].

2.4.2 Real Time Streaming Protocol (RTSP)

RTSP is a network control protocol used to establish and control media sessions between end points. RTSP was developed by the Multiparty Multimedia Session Control Working Group (MMUSIC WG) of the IETF and published as RFC 2326 in 1998. It is a client-server multimedia presentation protocol that provides VCR like controls (like PAUSE, FAST-FORWARD and PLAY) for audio and video streams. The transmission of the actual streaming data is not a task of the RTSP protocol. Most RTSP servers use the Real-time Transport Protocol (RTP) for media stream delivery. RTSP is similar to HTTP in that it uses URLs, but unlike HTTP, RTSP is a stateful protocol [127]. The basic RTSP requests are listed below;

a. OPTIONS: The client or the server tells the other party the options it can accept.

b. DESCRIBE: The client retrieves the description of a presentation or media object identified by the request URL from the server.

c. SETUP: This specifies how a single media stream must be transported. This must be done before a PLAY request is sent. The request contains the stream URL and a local port for receiving RTP data (audio or video), and another for Real Time Control protocol (RTCP) data, which is essentially meta information.

d. PLAY: The client asks the server to start sending data on a stream allocated via SETUP.

e. PAUSE: The client temporarily halts the stream delivery without freeing server resources.

f. TEARDOWN: The client asks the server to stop delivery of the specified stream and free the resources associated with it.

g. REDIRECT: The server informs the clients that it must connect to another server location. A mandatory location header contains the URL the client should connect to.

h. RECORD: This request can be used to send a stream to the server for storage [127].

According to RFC 2326, RTSP responses fall into one of the following categories:

a. Informational (1xx): Request received, continuing process.

b. Success (2xx): The action was successfully received, understood, and accepted.

c. Redirection (3xx): Further action must be taken in order to complete the request .

d. Client Error (4xx): The request contains bad syntax or cannot be fulfilled.

e. Server Error (5xx): The server failed to fulfill an apparently valid request [127].

In an IPTV setting, ETSI TISPAN standards make use of RTSP for Trick-Play (e.g PLAY, PAUSE e.t.c.) functionality in their IMS-based IPTV architecture. Trick play functions are used with on-demand content such as VoD. But since RTSP was developed as a media session setup and control protocol, some of its functionality duplicates much of what SIP already does. Therefore ETSI TISPAN standards specify two methods for setup and control of IPTV sessions in the IMS, using both SIP and RTSP. The first method (Method 1) only uses a subset of RTSP commands, omitting the session setup commands, while the second (Method 2) uses all RTSP commands specified in the RTSP RFC [137][127]. These two methods are shown in Figure 2.6 and Figure 2.7, as specified in ETSI TISPAN TS 183 063 [42]. The UE can use either of these methods to set up media sessions.

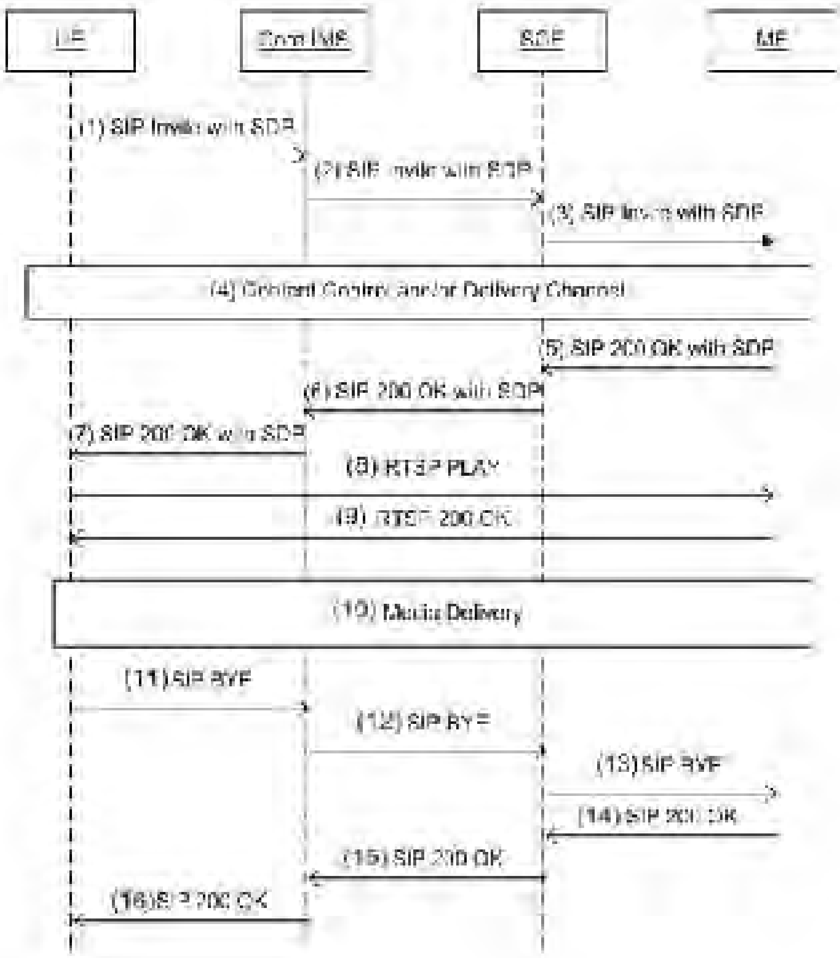

Figure 2.6: ETSI TISPAN IPTV Trick-Play Session (Method 1) [137].

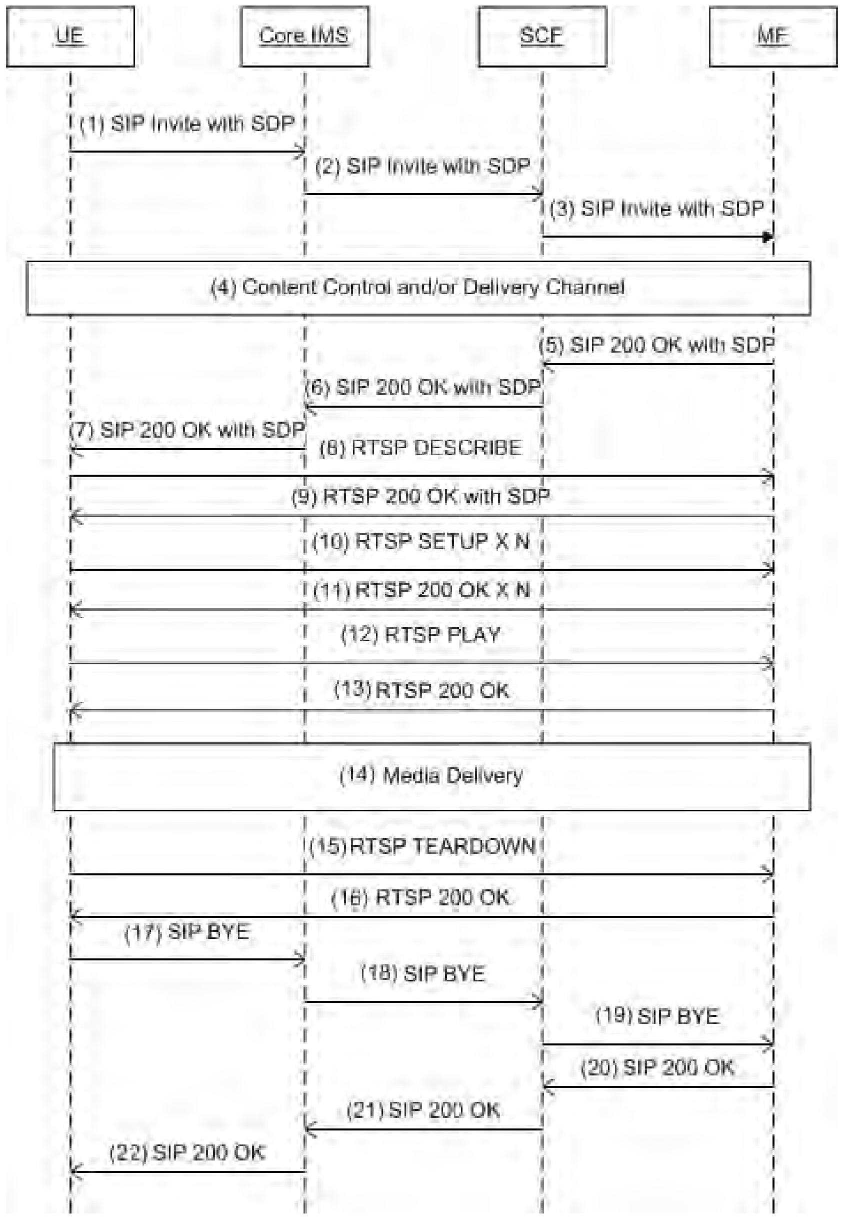

Figure 2.7: ETSI TISPAN IPTV Trick-Play Session (Method 2) [137].

Other researchers have proposed the integration of SIP and RTSP to create a comprehensive media control protocol [131].

2.4.3 Real-time Transport Protocol (RTP)

RTP is a standardised packet format and protocol for delivering audio and video over IP. RTP was developed by the Audio-Video Transport Working Group of the IETF and first published as RFC 1889, which was superseded by RFC 3550 [126]. RTP is used mainly for communication and entertainment systems that involve streaming media, such as IP telephony and IPTV. RTP is used in conjunction with RTCP (RTP Control Protocol), which monitors transmission statistics and QoS information. RTP is typically run on top of UDP to make use of its multiplexing and checksum functions. Compression standards such as H.264, MPEG and H.263 are used to encode the payload or data [119]. Figure 2.8 shows an RTP packet encapsulated in a UDP/IP packet.

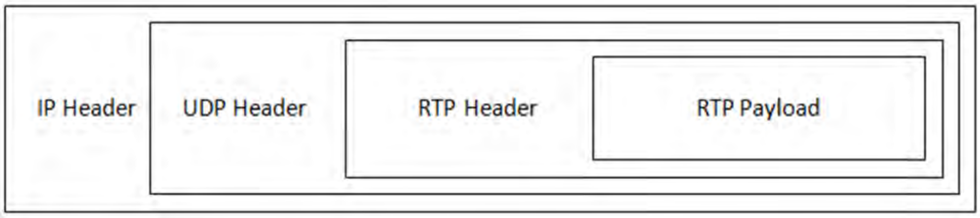

Figure 2.8: RTP packet encapsulated in a UDP/IP packet [82]

2.4.4 Hypertext Transfer Protocol (HTTP)

HTTP is a stateless application layer protocol for distributed, collaborative, hypermedia information systems [44]. HTTP is the main protocol behind the World Wide Web (WWW), though its use goes beyond hypertext. HTTP can also be used with name servers and in distributed object management systems. The first version of HTTP was defined in 1991 and is referred to as HTTP/0.9. This was replaced by HTTP/1.0, which is defined in RFC 1945 [10]. HTTP/1.0 was itself replaced by HTTP/1.1, which was specified under RFC 2216 in 1999 [44].

The World Wide Web Consortium (W3C) and the IETF have jointly coordinated the standardisation of HTTP. In most cases, HTTP is implemented under TCP/IP, but it can be implemented on any other protocol on the Internet, or on other networks [44].

In the IPTV environment, HTTP has been used for service selection, which is essentially HTTP-based data retrieval from the SSF [36][42].

2.4.5 Session Description Protocol (SDP)

SDP is designed for describing multimedia sessions for the purposes of session announcement, session invitation, and other forms of multimedia session initiation. SDP was initially published by the IETF as a Proposed Standard in 1998 under RFC 2327 [56]. The latest specification is published under RFC 4566 [57]. Unlike RTP, SDP does not deliver the media itself, but is used during the negotiation between end points for media session parameters. SDP is designed to be extensible to support new media types and formats. SDP uses attribute/value pairs to describe a media session.

SDP messages usually contain information such as session name, media type, transport protocol and information about the bandwidth to be used by the session. SDP is typically used in conjunction with signaling protocols such as SIP and RTSP [7].

2.4.6 Diameter Protocol

Diameter is a networking protocol developed by the IETF to provide Authentication, Authorisation and Accounting (AAA) in computer networks. It was introduced to cater for growing demands on AAA caused by factors such as the growth of the Internet and the introduction of new access technologies, including wireless, DSL, Mobile IP and Ethernet. It replaced the RADIUS protocol that was used for AAA prior to that. The Diameter base protocol is defined by RFC 3588 [15].

Diameter sessions consist of an exchange of commands and attribute value pairs (AVPs) between authorised Diameter servers and clients. Some of the command values are used by the Diameter protocol itself, while others deliver data associated with the particular applications that employ Diameter.

In an NGN setting, Diameter has various applications in the 3GPP IMS subsystem and in Mobile IP networks. Diameter is also intended to work in both local AAA and in roaming situations [15]. Under IPTV, Diameter has been used to handle user billing/charging transactions in an IMS-based IPTV environment as stipulated under 3GPP TS 32.299 for charging management in IMS-based applications [39][151].

2.4.7 XML Configuration Access Protocol (XCAP)

The XML Configuration Access Protocol (XCAP) is an application layer protocol that allows a client to read, write and modify application configuration data stored in XML format on a server. XCAP defines two logical roles: an XCAP server and an XCAP client. The XCAP server acts as a repository for XML documents while the XCAP client is the entity that reads, writes or modifies XML data. XCAP maps XML document sub-trees and element attributes to HTTP URIs, so that these components can be directly accessed via HTTP. XCAP uses the HTTP methods PUT, GET and DELETE to retrieve or modify XML documents stored on the server. XCAP is specified under RFC 4825, RFC 4826, RFC 4827 and RFC 5025 [121].

In an IPTV setting, the XCAP server is used to store IPTV related data. XCAP commands can then be used by clients to retrieve or update this data [42].

2.5 Mobicents Communications Platform

Figure 2.9: Mobicents Communications Platform [118]

This section discusses the Mobicents Communications Platform. The Mobicents projects comprises of the Mobicents JAIN SLEE server, Mobicents SIP Servlets, Mobicents Media Server and Mobicents Presence Service sub-projects. Figure 2.9 shows a simplified version of the Mobicents Communications Platform.

2.5.1 Mobicents JAIN SLEE

Mobicents JAIN SLEE is an open source SDP for the Java environment. It is the first and only open source SLEE certified application server. Mobicents JAIN SLEE is designed and optimised for event driven applications, also known as asynchronous applications. The latest version of Mobicents JAIN SLEE server is 2.x [101].

Mobicents JAIN SLEE is built using the SLEE specification, which allows portable, carrier grade telecommunication services to be built, managed and executed within a secure environment [79]. Telco applications are typically run over fault tolerant and highly available platforms, hence JAIN SLEE constraints vary considerably from say Java Enterprise Edition (JEE/J2EE) principles. Table 2.1 contrasts J2EE and JAIN SLEE principles.

Criteria	J2EE	JSLEE
Average Transaction Span	< 2 seconds	< 100ms
Transaction Volume	1'000s/sec	10'000s/sec
Persistence	Wide range	Read mostly
Uptime	99.9% (9 hours/year)	**99.999%** (5min/year)

Table 2.1: J2EE vs. JSLEE principles [63]

Other advantages of developing telco applications over JAIN SLEE are;

a. Integrated JEE and JAIN SLEE Environment: enables standard based converged voice, web and data services.

b. Portable Services: JAIN SLEE supports the write once, run anywhere feature of the Java programming language. This means a Java application can be developed on any device, compiled into a standard byte code that is expected to run on any device equipped with a Java virtual machine. Application components can be developed and then deployed on JAIN SLEE compliant platforms from different vendors without recompilation or source code modification.

c. Network Independence: JAIN SLEE is independent of any particular network protocol, API or network topology. This means JAIN SLEE can be used to address business problems that involve multiple networks.

d. Supports Integration with Existing Management Interfaces: JAIN SLEE defines a management API that allows a SLEE to be controlled by external management systems.

e. Reliable: JAIN SLEE enforces the use of a transactional programming model. Applications written according to the transactional programming model have well defined policies under failure conditions [81].

Apart from the telco industry, the Mobicents Platform is suitable for a variety of problem domains demanding an event-driven architecture for high-volume, low-latency signaling. These include financial trading, online gaming, RFID sensor network integration, and distributed control [101].

2.5.1.1 Mobicents JAIN SLEE Architecture

Figure 2.10 shows a high level overview of Mobicents JAIN SLEE.

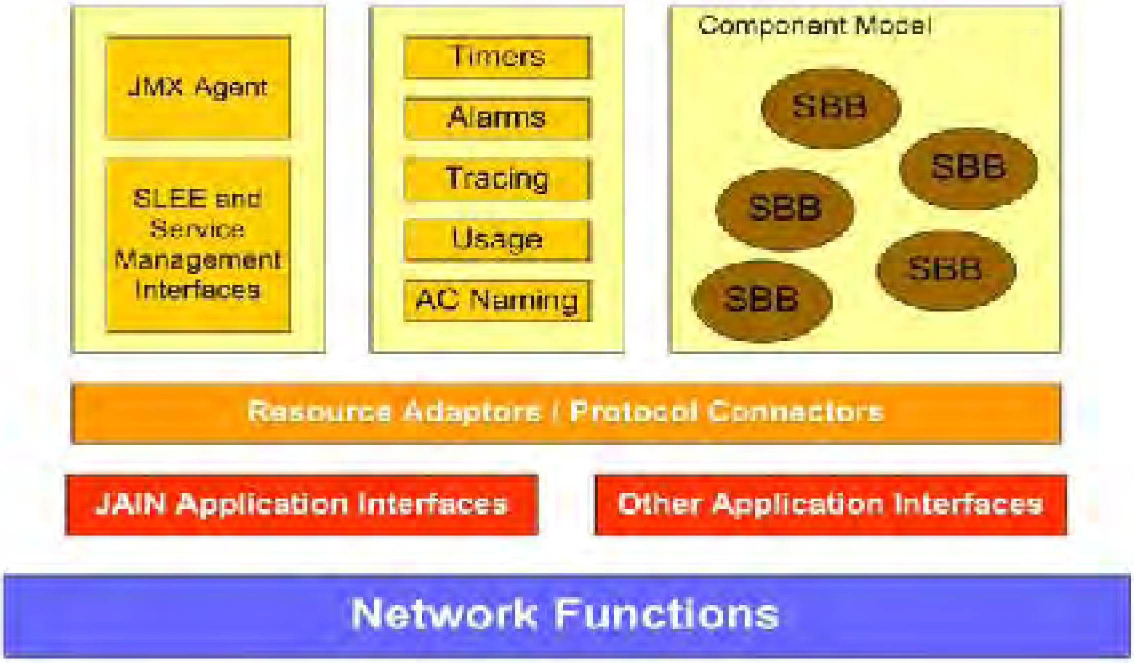

Figure 2.10: Mobicents JAIN SLEE component architecture [101]

The key components of Mobicents JAIN SLEE are as follows;

Service Building Blocks (SBBs)

SBBs are programmer defined blocks of code that contain the application logic for services such as call control and user billing. Each SBB component identifies the event types that it accepts and has event handler methods that contain application logic that processes events of this type [79]. A SLEE event represents an occurrence that requires application processing. SLEE also defines an Activity Context which acts as an event channel and SBBs will only receive events from attached Activity Contexts. An entity called the Event Router is responsible for routing events to the respective SBBs interested in that particular event. Figure 2.11 shows the call flow for event handling within Mobicents JAIN SLEE.

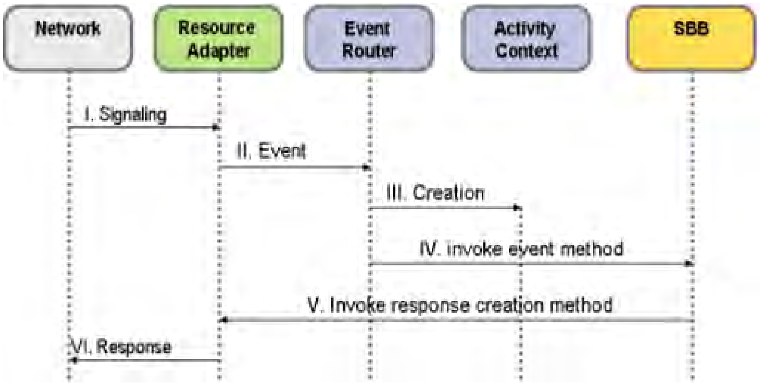

Figure 2.11: Mobicents event model [62].

Resource Adaptors

Mobicents also allows well known protocols such as SIP and HTTP to be plugged into the SLEE as Resource Adaptors (RAs). JAIN SLEE RAs provide an interface between the SLEE and underlying protocols. RAs are responsible for communicating with a particular resource that is external to JAIN SLEE and for communicating with the event routing logic within the SLEE implementation [37][81].

Management Interfaces

Out-of-the-box monitoring and management of Mobicents components is achieved through the use of Java Management Extension (JMX) consoles [37].

Facilities

Mobicents JAIN SLEE defines several facilities that SBBs, RAs and other SLEE components can make use of. These facilities are the Timer, Alarm, Trace, Usage and the Activity Context (AC) naming facilities [37][84].

2.5.1.2 Benchmarking Mobicents JAIN SLEE

In terms of performance, benchmarking tests are regularly performed by the Mobicents development team. The peak performance for Mobicents Jain-SLEE (version 2.1.0.GA) is reported at between 1000 - 1200 transactions per second on a Dual Core CPU @ 2.8 GHz. This was on an OSX operating system [102].

2.5.2 Mobicents SIP Servlets

The Mobicents SIP Servlets container provides an open platform on which to develop and deploy portable and distributable SIP and converged JEE services. Mobicents SIP Servlets provides the first open source certified implementation of the SIP Servlet 1.1 specification [26][101].

2.5.3 Mobicents Media Server

Mobicents Media Server (MMS) is an open source media gateway that is part of the Mobicents project. MMS is available as a standalone media server or embedded in a JBOSS Application Server [93]. MMS presently supports both the Media Gateway Control Protocol (MGCP), Media Server Markup Language (MSML) and JSR 309 for media server control. Video support for MMS is at present still limited to the H.261 standard [101].

2.5.4 Mobicents Presence Service

The Mobicents Presence Service, provides presence capabilities to SIP-based networks, and is built on Mobicents JAIN SLEE. It is comprised of three separate but interrelated servers, namely the Mobicents Presence Server, XML Document Management Server (XDMS) and the Resource List Server (RLS). The Mobicents Presence Server is the entity that accepts, stores and distributes SIP based presence information. The XDMS is responsible for managing XML documents stored on the network, such as presence authorisation rules, contact and group lists (also known as resource lists), as well as static presence information. The RLS is the element that handles subscriptions to resources lists [93].

2.6 Chapter Summary

This chapter provided some background on IPTV value added services and the role that advertising can play. This chapter also assessed ongoing IPTV standardisation efforts and trends, since one of the key objectives of this thesis is to develop an interoperable IPTV solution. This chapter particularly looked in detail at the ETSI TISPAN IMS-based specifications, since the implementation of the proposed framework will be based on this.

The author concluded the chapter with an assessment of the Mobicents Communications Platform.

Chapter 3

Design Considerations and Implementation Requirements for the COIN Framework

This chapter presents design considerations and implementation requirements for the COIN framework to address the research problems identified in chapter 1. This chapter will also provide details on the implementation of the components and presents a framework that will allow users to proactively define the nature of viewing material and advertisements available to them. The framework does not impose advertisements on clients, but allows users to view advertisements on-demand, while granting them a billing incentive or discount for viewing advertisements during the pay-per-view video they may be watching at the time. This advertiser-driven billing incentive is designed so that users cannot abuse the system by excessively requesting advertisements. It is put forward as an effective solution to providing a user-focused, and competitively low-cost IPTV solution that hinges on low-cost infrastructure and advertiser support. It also addresses the challenge of users demanding more control over multimedia applications and aims to deliver targeted advertisements in a non-invasive manner.

The proposed framework uses online charging since it is best suited for VoD. Online charging is a billing system that allows service providers to charge their customers in real time, based on service usage. This method of charging performs credit checks at session setup to ensure the user has sufficient credit for the requested service before granting access.

3.1 Design Considerations

The author identified the key stakeholders in the proposed service as the telco, the user, and the advertiser. To ensure the proposed framework is suitable for all stakeholders involved, the author evaluated their needs and interests in an IPTV setting. The author discusses these below.

3.1.1 The Telco

Telcos have turned to IPTV in recent years to help boost revenue, but IPTV deployments presently face stiff competition from Internet Television and traditional forms of television such as Cable and Satellite television. Selling IPTV solutions to entrenched traditional television viewers will only be possible if telcos can provide a unique selling proposition. One such differentiator for IPTV is VoD, on which the proposed implementation is based. VoD is currently one of the main drivers in consumer interest towards IPTV [91].

IPTV solutions can sell much faster if they are low cost and if they can address the growing consumer need for more control over multimedia applications. With IPTV revenues continuing to rise globally, IPTV solutions deployed using low-cost infrastructure with advertiser support could potentially re-energise telco revenues. The use of low-cost infrastructure or FOSS will lower CAPEX. This is expected to lead to low-cost telco services, which in turn is expected to boost subscriptions and ultimately revenues. The ability to customise the service will make the service more appealing to users by placing more control in their hands. Advertiser driven incentives will make the service even more appealing to users by helping to further lower the cost of the IPTV service, while granting IPTV operators additional revenue from advertisers.

3.1.2 The User

As identified in chapter 1, today's multimedia users increasingly want more control over multimedia applications. Personalisation has been used in the IPTV market to help customers select viewing material. This has met with some success but the major argument against this approach is that it invades viewer privacy since it involves the service provider collecting data on viewer habits. A more user-centered approach would involve users stipulating (or customising) the kind of material they want to watch before-hand. Therefore the ability to customise the IPTV service will grant users more control.

Advertisements are often quite informative and in some cases provide a refreshing and welcome break from the content a viewer may have been watching, particularly if the topic is of interest to the viewer. Re-evaluation of advertising models indicates that it is important to present the right kind of advertisements to the right audience and at the right time [151]. Advertisers are attempting to achieve this through targeted advertising and on-demand advertising. As discussed in chapter 2, targeted advertisements can be achieved via customisation or personalisation. The more user oriented approach of achieving this is customisation. Therefore the proposed framework will use customisation to achieve targeted advertisements. This means that the user must be allowed to explicitly state (via a suitable interface) what kind of advertising material they are interested in, rather than have the operator 'spy' on them and use that data/profile to generate targeted advertisements.

In terms of how the advertisements are presented, the traditional approach was to interrupt the program the user was watching with the advertisement. A key strength of IPTV is that it allows advertisements to be viewed on-demand. This is a more pleasant setup and will lead to a better user experience. This also places more control in the hands of users in that users now have the power to request advertisements at whatever time best suits them, that is if they want to view advertisements at all. The added incentive of billing discounts for watching advertisements on-demand will mean viewers pay less for the VoD content.

3.1.3 The Advertiser

Advertisers are embracing newer forms of advertising such as targeted advertising and pay-per-click (PPC) advertising with encouraging results (PPC is an Internet advertising model used on websites where advertisers pay the host website only when the advertisement is clicked) [20]. Targeted advertisements in particular ensures that the advertising message reaches a select group of customers more likely to purchase the product. This results in a boost on ROI. Targeted advertising is therefore a more effective advertising approach as compared to traditional broadcast advertising.

3.2 Implementation Requirements

This section details the components that are required to incorporate the above requirements into a framework that closely follow ETSI TISPAN IMS-based IPTV specifications, which are presently the most comprehensive set of specifications on IMS-based IPTV [151]. This section will also provide details of the implementation of these components. It is crucial that the proposed framework make use of standard NGN functional components, interfaces and protocols wherever possible as defined by the ITU and ETSI TISPAN. This will ensure interoperability with other standards based implementations and NGN entities. As highlighted in chapter 1, the proposed framework strives to use only free and open source software. This will enable the easy duplication of this work for future research. The required architectural components are described below.

3.2.1 User Equipment (UE)

The UE/IPTV/VoD client is the entity that is used by the customer or user to access the IPTV service hosted on the Mobicents Platform. According to IPTV standardisation efforts, the UE is required to support a number of protocol interfaces. The key signaling and session set-up protocols are SIP and RTSP. SIP is typically used to request IPTV services and to contact the relevant IPTV application servers, such as the SDF for service discovery purposes. RTSP is used for trick play functions with the MDF. The UE can also have an HTTP interface, which can be used to retrieve service selection information from the SSF in Pull mode. The UE is additionally required to possess video playback capabilities.

For the implementation, a client was sought that could be used across several platforms. Such a client would provide the flexibility to perform investigations on a variety of platforms if need arose. In this regard, it was decided that due to its "write once, run anywhere" feature, a Java based client would most likely fit this criteria. Hence a Java client was preferred, but not in the strictest sense.

The investigation began with a search for free and open source IPTV/VoD clients that were tailored to the requirements. According to the findings, the only free and open source IPTV/VoD client was the UCT IMS client. This client had the advantage of IMS compliance, which would be ideal if tests needed to be done in an IMS environment. However, the major drawback to this client was that it was restricted to the Linux environment, and even on this platform, it had major interoperability issues owing to the significant differences between the various Linux distributions, therefore the UCT IMS client was disqualified. This meant that a custom IPTV/VoD client would need to be developed if cross-platform support was to be realised.

As discussed in chapter 2, the predominant IPTV signaling protocol is SIP. Therefore any open source SIP client that also contains a suitable media library capable of video playback and RTSP signaling could be used as a VoD client. At this stage, the author reasoned that the easiest approach to obtaining a cross platform IPTV client was to find a free and open source Java based SIP client, and add a suitable media library to it. The author therefore proceeded to investigate the available free and open source Java based SIP clients. Initially, the SIP Communicator and JAIN SIP Applet Phone were identified and noted as possible candidates for this purpose. Table 3.1 contrasts relevant features of the SIP Communicator, the JAIN SIP Applet Phone and the UCT IMS client [21][22][114].

Property/SIP Client	SIP Communicator	JAIN SIP Applet Phone	UCT IMS client
IPTV/VoD Support	No	No	Yes
Cross-platform Support	Yes	Yes	No
Developer Support	Good	Moderate	Moderate-Good
Auto-configuration Support	No	No	No
IMS Compliance	No	No	Yes

Table 3.1: Contrasting properties of the SIP Communicator, JAIN SIP Applet Phone and UCT IMS client

The SIP Communicator appeared the best option mainly because it had a larger, and more active developer community than the JAIN SIP Applet Phone. This would be crucial if development problems were encountered. In terms of its architecture, the SIP Communicator is based on the OSGI framework, which is a module system and service platform for Java applications. This means in theory that adding functionality to the SIP communicator is about adding extra modules into the framework [22]. Attempts to add a media library to the SIP Communicator however proved futile. A separate VoD module was developed and added to the OSGI framework of the SIP Communicator, but unfortunately the video object required for video playback failed to work from within the OSGI framework, although the same module worked fine outside this framework. The author sought assistance on the matter from the SIP Communicator forums but no solution could be found. This client would have been ideal because of the said reasons as well as the vast features this client possesses, but unfortunately the required functionality couldn't be added. At this stage, attention was shifted towards adding a media library to the other candidate, the JAIN SIP Applet Phone. A media library was successfully added to the JAIN SIP Applet Phone, and it became the UE/VoD client for the testbed. Figure 3.1 shows an architectural overview to the JAIN SIP Applet Phone.

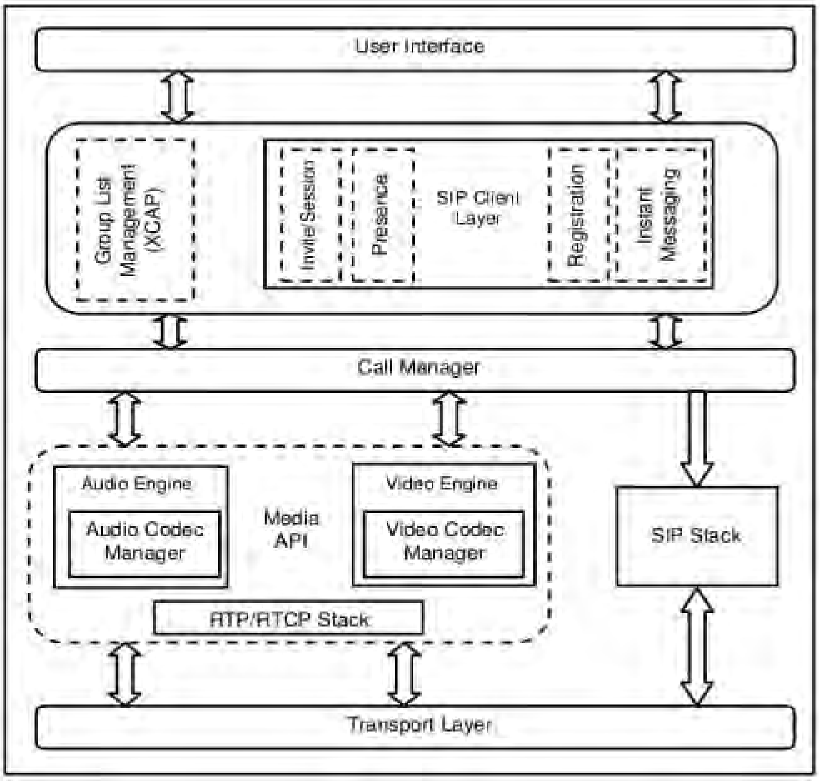

Figure 3.1: JAIN SIP Applet Phone architecture [104]

3.2.1.1 Media Library

Most SIP clients have in-built media libraries for multimedia processing, such as the audio codecs required for voice calls. The problem is that these media libraries are typically very limited in the number of video codecs they support and seldom have RTSP support, which is needed for trick play sessions with on-demand viewing. Therefore a separate media library would be needed to facilitate reliable video playback in the UE and for RTSP signaling.

Several free and open source media libraries were identified as possible candidates for use in the proposed UE/VoD client. The most stable and active projects were found to be the *gstreamer* project and the *libvlc* project, and both these had Java versions, namely jVLC and gstreamer-Java [67][69]. Experiments using jVLC revealed several problems, which included very little developer support and that the library interferes with the proper functionality of Java GUI components [29]. Gstreamer–Java was initially integrated into the SIP Communicator without success, and finally into the JAIN SIP Applet Phone with no difficulty. Figure 3.2 shows a Screen-shot of a modified JAIN SIP Applet Phone and the IPTV/VoD interface.

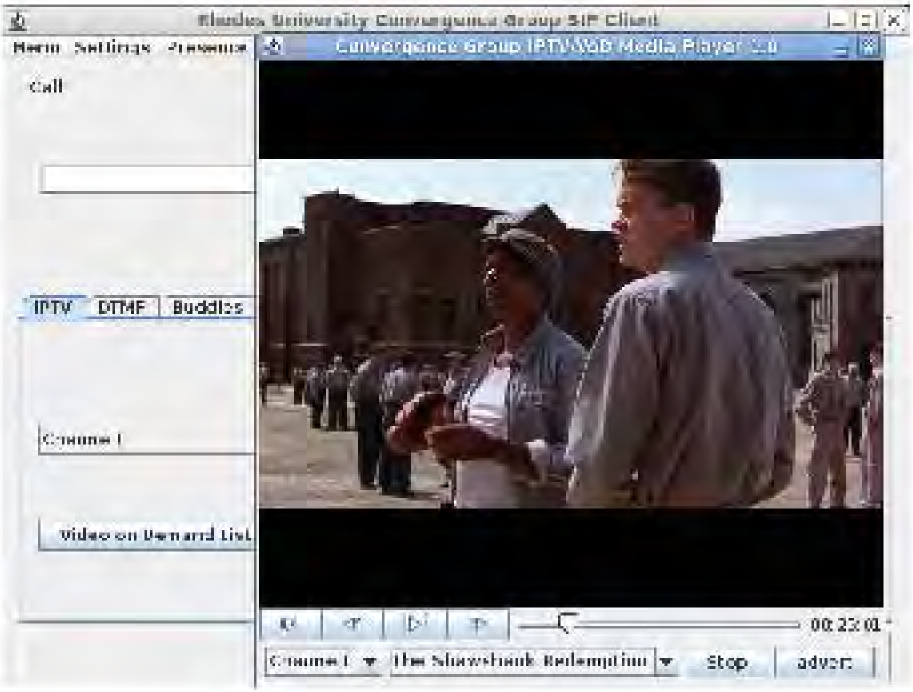

Figure 3.2: Screen-shot of a modified JAIN SIP Applet Phone main window and the IPTV/VoD media player in the foreground

3.2.2 IPTV Application Servers

Application servers are essentially SIP servers that are responsible for hosting and executing services. A SIP server may operate in one of four modes: SIP proxy mode, SIP User Agent (UA) mode, SIP redirect mode, or SIP Back-to-Back User Agent (B2BUA) mode. SIP proxy mode is when the SIP server receives and forwards client request to another SIP server, while SIP UA mode is when the SIP server behaves as a SIP client itself and may make request to other SIP servers. SIP redirect mode allows a SIP server to push routing information back to the client in a response message, this allows the server to take itself out of the loop for any further messaging for that transaction. Redirect mode can be useful in reducing processing loads on proxy servers that are responsible for routing requests. SIP B2BUA mode is when the SIP server is used to connect two SIP UAs [1].

The key functions or SIP servers with regard to ETSI TISPAN IMS-based IPTV are the SCF, MCF, SDF and the SSF. As a re-cap, the SCF is responsible for performing session authorisation and credit control, while the MCF controls and monitors MDFs for media delivery. The SDF is responsible for generating the service attachment information and the SSF allows the UE to retrieve service selection information, which is essentially a list of available services that the UE can then browse and select.

To meet ETSI TISPAN IMS-based IPTV specifications, the proposed framework will need implementations of these servers within the Mobicents environment. As mentioned in chapter 2, the Mobicents JAIN SLEE architecture allows service logic to be developed and composed into SBBs. Therefore the IPTV functions can be implemented in Mobicents JAIN SLEE as SBBs, and since these IPTV functions have SIP interfaces, implementation within Mobicents as SBBs would simply require a SIP RA. These IPTV functions will be able to take full advantage of the SLEE environment that Mobicents JAIN SLEE offers for low latency, event driven applications.

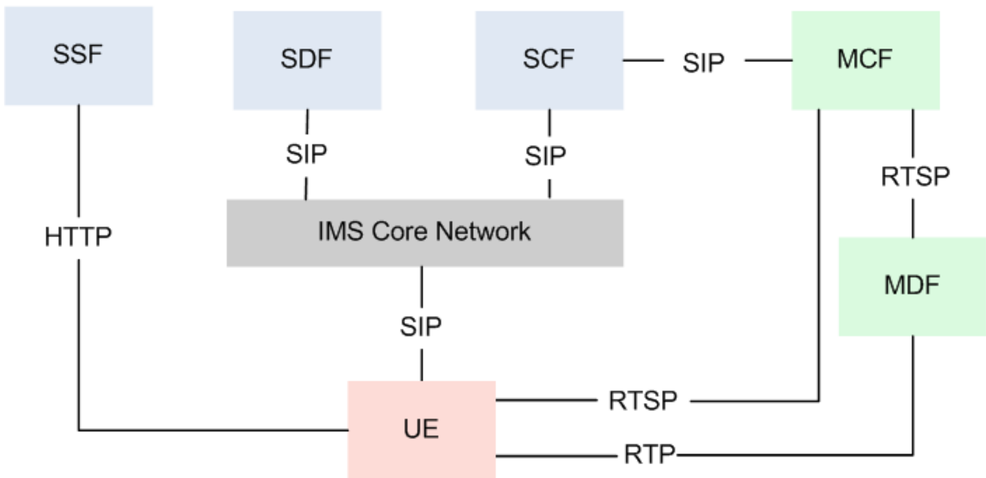

Figure 3.3: Message routing and relationships between functional components in ETSI TISPAN IMS-based IPTV (adapted from [129])

Finally, the decision to use a stand-alone SDP without IMS support will require that IMS functionality such as registering clients and session control be performed by SBBs within Mobicents. The IMS layer is also responsible for message routing in an IMS-based IPTV system, this will also be handled within Mobicents. Figure 3.3 shows an overview of message routing and the relationships between the various components in ETSI TISPAN IMS-based IPTV. Figure 3.4 shows an overview of message routing and relationships between the functional components in a hypothetical Mobicents based IPTV system. As discussed in chapter 1 and as illustrated in Figure 3.4, the proposed Mobicents based IPTV service is designed to allow for easy integration with an IMS control layer. It is however important to note that in Figure 3.4, only the SDF and SCF are required to be located on the same system or server because these two components will need to share a common event router, which is restricted to the server on which it is running. The other servers (SSF, MCF and MDF) can be located on different systems for load balancing purposes.

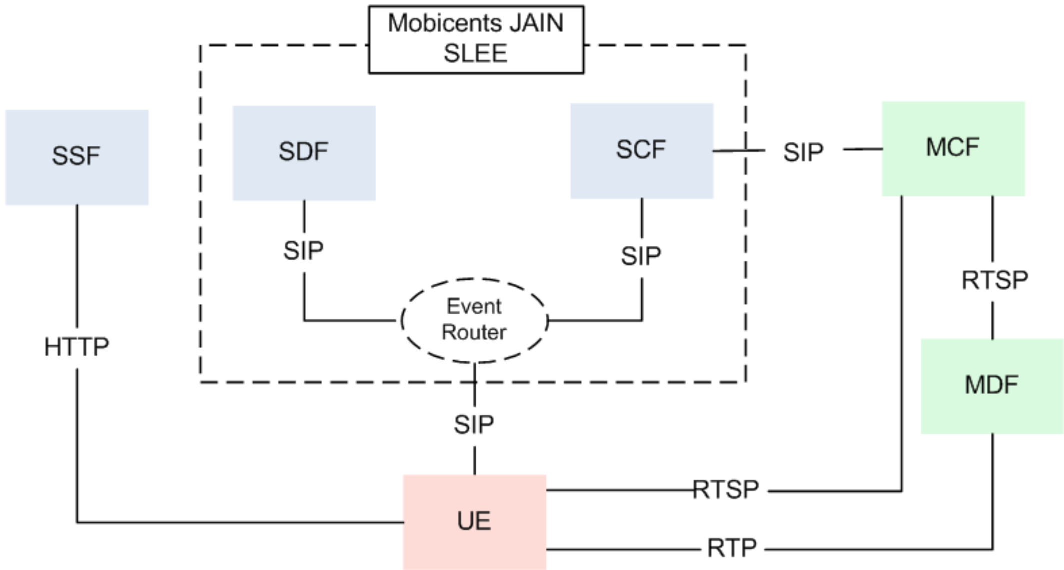

Figure 3.4: Message routing in a hypothetical Mobicents based IPTV system and associated components

Transforming the proposed Mobicents based IPTV system into an ETSI TISPAN compliant IMS-based IPTV system would require the addition of an IMS core network to handle client registration, routing and session control. Additionally, any logic contained within Mobicents for client registration and session control becomes unnecessary. Everything else remains unchanged.

3.2.2.1 SBB Deployment Within JAIN SLEE

Once the application logic for an SBB has been written, the SBB can be deployed within the SLEE using build tools such as Apache Ant or Apache Maven. Build tools are used to automate the software build process. To use the Apache Ant build tool, an XML build-file has to be created, while Maven requires an XML Project Object Module (POM) file.

These files describe the build process and specify any dependencies [5][96]. The “target” or output of a successful build process is typically an executable Java Archive, or Jar file. Once this has been created, “deployment” involves copying the Jar file into the “deploy” directory of the JAIN SLEE server. Once deployed, SBBs can be monitored through JMX. Figure C.1 of Appendix C shows the POM file used for the deploying the SCF.

3.2.3 IPTV Media Functions

IPTV specifications make use of an MF to provide media streaming and control capabilities. The MF internally consists of two entities: the MDF and the MCF. These may be co-located as discussed in chapter 2. The MDF is essentially a streaming server with an RTSP interface. The MCF is responsible for controlling the MDF and possesses both a SIP and an RTSP interface. Setting up a media session with an MF initially involves sending a SIP INVITE message to the MCF to initialise a content control and delivery channel with the UE as depicted in Figure 2.6. Once this is done, the UE can then request the media stream from the MF by using RTSP commands. These RTSP commands are initially received by the MCF, and then forwarded to the MDF as depicted in Figure 3.5 for Method 1 and as specified in [42]. The call flow is analogous for Method 2.

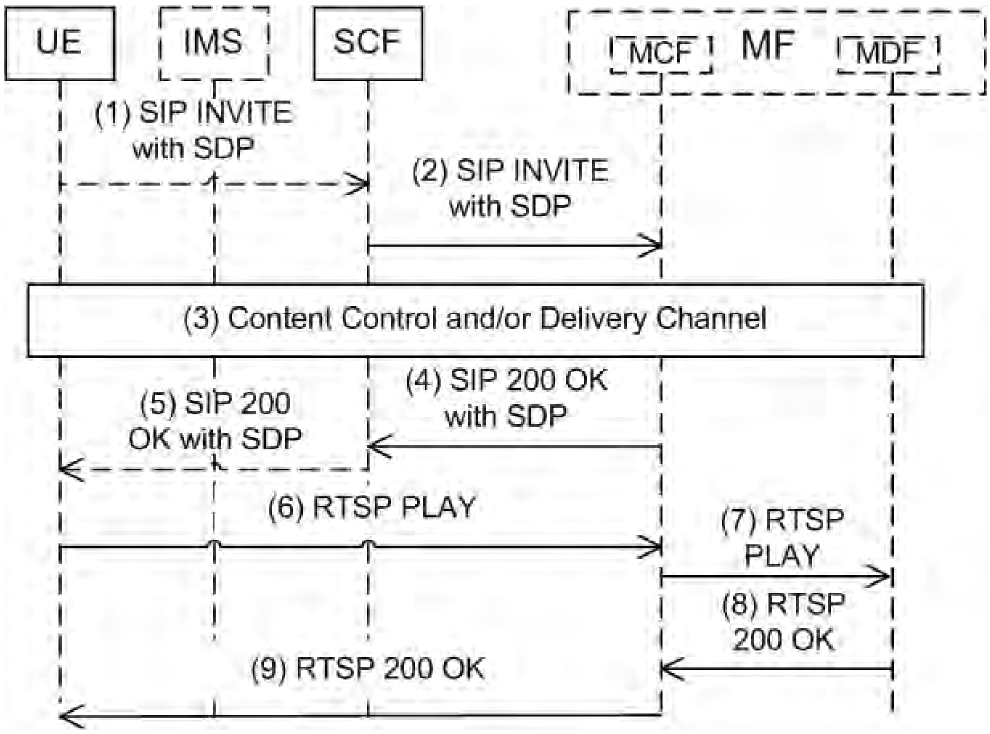

Figure 3.5: Call flow signaling highlighting the relationship between MCF and MDF for Method 1.

Full fledged media servers can be used to provide MF capabilities, i.e. co-located MCF and MDF functions. This is because media servers possess both SIP and RTSP interfaces. But due to the lack of free and open source media servers, the research community is forced to turn to readily available, free and open source streaming servers for purely MDF capabilities [130]. Using a streaming server to provide MDF functionality means that a separate MCF will be required to handle the SIP messaging as discussed above and as highlighted in chapter 2. To 'control' the MDF, the MCF is expected to act as an RTSP proxy and forward RTSP messages between the UE and the streaming server/MDF. The major challenge to such an implementation is that most streaming servers do not allow a separate entity to initiate a media session on behalf of another [130]. A simple, but insecure approach to getting around this problem is to use the MCF as a redirect server for the MDF [137]. In this approach, when the MCF receives a SIP INVITE request for an IPTV media session, the MCF returns the URL or RTSP address of the requested video to the UE in a SIP 200 OK message, rather than setting up a content control and delivery channel with the UE as required by the IPTV specifications. The URL or RTSP address of the requested video is essentially the location of the video on the streaming server. Once the UE has this address, it can directly request the video stream from the streaming server or MDF using RTSP commands. This method is specified under RFC 4483 for content indirection under the SIP protocol [13]. The drawback to this approach is that the MCF does not in fact control the MDF and clients can request media sessions directly from the MDF if they know the RTSP addresses of videos stored on the MDF. Figure 3.6 shows the MCF acting in redirect mode for the MDF.

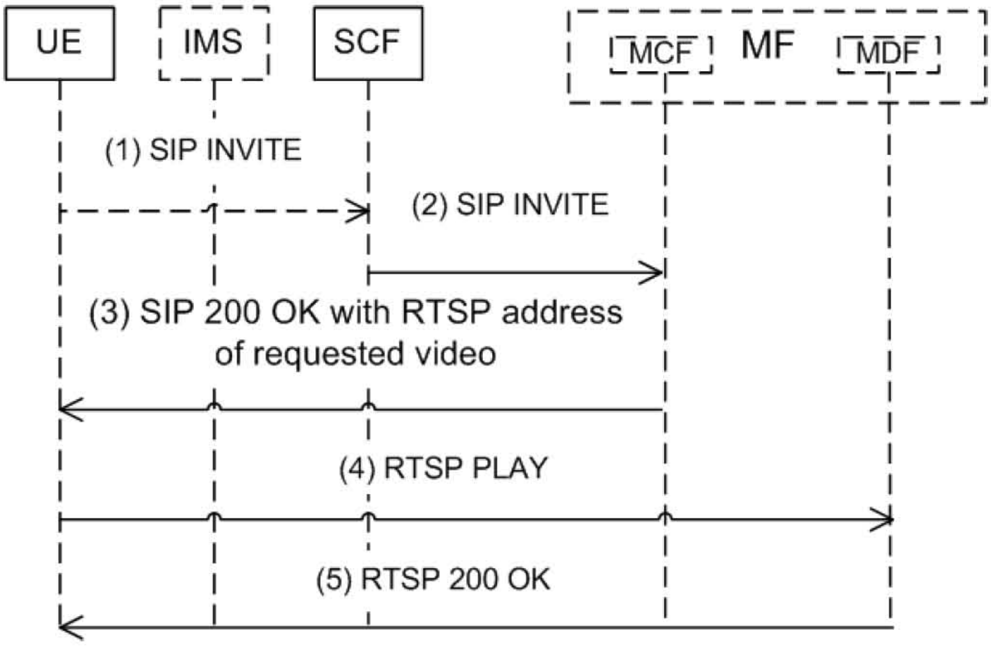

Figure 3.6: Call flow showing the MCF in redirect mode

In choosing a streaming server, a requirement was that the streaming server support a large selection of video codecs. This would grant the flexibility to stream almost any video file in the testbed. In addition, the streaming server is required to support RTSP, which is necessary for VoD trick modes. Investigations into to the available open source streaming servers revealed three options, these were Red5, Darwin and VLC [128][117][60]. Red5 is an open source Flash streaming server, meaning that its capabilities are mainly limited to Flash videos, therefore it was disqualified. Darwin streaming server is an open source version of Apple's QuickTime streaming server. It has streaming support for hinted QuickTime, MPEG-4 and 3GPP files using RTSP and RTP [128]. On the other hand, VLC is an open source streaming server that uses codecs provided by the *libvlc* media library. It has support for a wider range of codecs, these include MPEG-4, H.263, VP5, VP6, and Theora. It supports streaming over RTSP, RTP and HTTP [60]. In addition to this, VLC streaming server is supported on more platforms than Darwin streaming server, which was primarily released for the Mac OS environment [128]. For these reasons, VLC was chosen to provide streaming capabilities in the testbed.

3.2.4 Databases

According to ETSI-TISPAN IMS-based IPTV specifications, the HSS in the IMS core acts as a central repository for storing user-related information that is used to handle IPTV sessions. It typically contains location information, security information (authentication and authorisation information) and client profile information. User profile information includes private and public user identities, subscription information and filtering criteria [42][91]. In the absence of the HSS from the IMS core, the relevant IPTV information can be stored in databases.

In addition, an IP Geo-location database is required to map client IP addresses to their geographical origin. The ability to determine the origin of a client request can be useful information to an IPTV provider, particularly in terms of how to deliver the media to the client. The ideal set-up is to have the client receive the streaming video from the closest MF (Geographically). This approach generally reduces service latency and can improve the QoE. In most cases, an IPTV deployment will have more than one MF for load balancing purposes, it is the responsibility of the SCF to select the most appropriate MF to service a client request. This decision is affected by a number of factors, including the location of the UE and the load of the available MFs [42]. Therefore to determine the Geo-location of client IP addresses, the free and open source version of the Maxmind Geo-location database was used. The database also comes with an API that is used to access the database from any application [97]. Figure 3.7 shows a code snippet for determining the Geo-location of an IP address using the Maxmind database and its API in the SCF SBB.

```
String db_location = "/usr/local/share/GeoIP/GeoLiteCity.dat";
cl = new LookupService(db_location,LookupService.GEOIP_MEMORY_CACHE );
l2 = cl.getLocation(IP);
logger.info("The incoming request is from the city of : " + l2.city);
```

Figure 3.7: Code for using the Maxmind Geo-location database and its API .

3.2.5 Web Interface

In order for system users to customise their profiles, a suitable interface is necessary. Most Internet users today are familiar with the WWW and therefore are comfortable with web interfaces. It therefore seemed appropriate that a web interface be used by system users to customise the VoD listings or advertisement preferences.

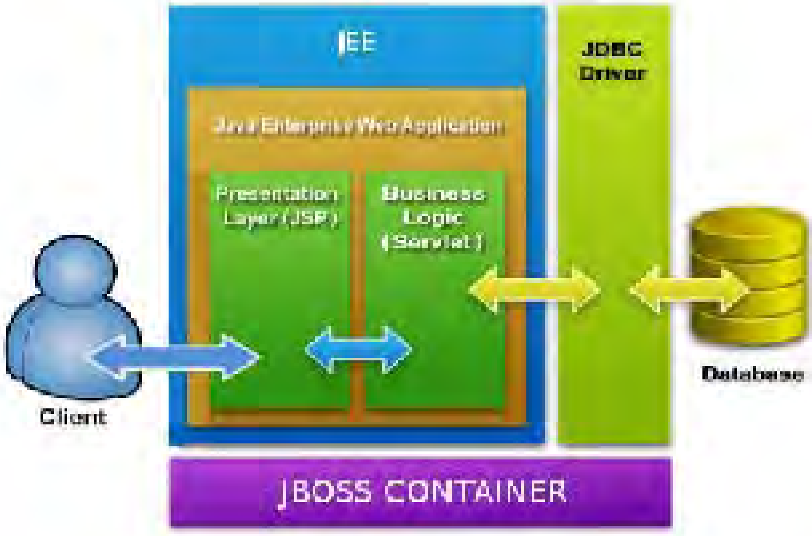

Figure 3.8: Design of web application used for video and advertisement customisation in the Mobicents Platform.

Mobicents is an integrated SLEE/JEE application server, meaning it has support for JEE components such as Java Server Pages (JSPs) and Servlets used in enterprise web development. Therefore a simple web application was set-up using JSPs and Servlets and deployed in the Mobicents container. The web application allowed clients to log onto the site and customise their video and advertisement preferences. The open source MySQL database was used to store these preferences in the profile database [105]. The logic for accessing and manipulating this database was contained in a Servlet. Figure 3.8 shows the design of this web application. Figure B.1 of Appendix B shows a screen-shot of the web interface used for service customisation.

3.2.6 Video and Advertisement Classifications

In order for system users to be able to customise their viewing preferences through the web interface, the author created video/movie genre classifications and separate advertisement categories. For video classification, the following classes were used: Drama, Action, Horror, Sci-Fi, Comedy, Thriller, Romance and Musical. Because the relationship between a video and its classification can typically be one to many, additional video classes like Romantic-Comedy and Action-Comedy were created to cater for those videos/movies falling into more than one class. On service request, the SSF will create and send the custom video and advertisement lists to the UE.

Advertisement categories used in the framework were as follows: Food, Clothing, Electronics, Vehicles, Alcohol and Beverages, Finance, Sports and Department Stores. User profile data contained the advertisement classes (if any) that the user desires. Upon service request, advertisements falling into any class not stipulated by the user get filtered out, and only relevant advertisements get sent to the client for possible viewing on-demand, but the user does not know which advertisement they are requesting as the UE hides this information.

3.2.7 Charging Functions

Dedicated charging functions will be required to handle user billing and maintain client billing transactions data. The SCF will generate charging information on a Diameter interface for multimedia sessions as specified in ETSI TISPAN IMS-based IPTV specifications [39]. As specified by 3GPP, IMS charging involves the following main entities: a Charging Trigger Function (CTF), a Charging Data Function (CDF), and Online Charging Function (OCF). The CTF is deployed as an integral part of every application server or entity where charging information should be collected, in this case the SCF. The CDF is used for offline charging processes, and involves exchange of Diameter accounting messages with the CTF. The OCF is required for online charging and exchanges Diameter credit control messages with the CTF [113]. Since the proposed framework performs online charging, only an OCF is required. The SCF will perform CTF functions.

3.2.8 Auto-configuration Servers

Manually configuring network parameters for a device or application can be a tedious task, and in many cases, these parameters are unknown. Hence, discovering network services with little or no configuration is an important feature of networked applications. Additionally, one of the key aims of the NGN is to "provide unfettered access for users to networks and to competing service providers and/or services of their choice" [71][72]. In the IPTV setting, this implies that there can be more than one IPTV provider available for user selection. Therefore being able to establish who the available IPTV providers are automatically becomes an important property of the UE. Once the user has the list of available IPTV service providers, he/she can make a decision on which IPTV provider to attach to for IPTV services. Auto-configuration or service discovery in an NGN setting can be achieved using the Network Attachment Subsystem (NASS) [36]. At the local network level, auto-configuration can also be performed by Dynamic Host Configuration Protocol (DHCP) and Domain Name Service (DNS) servers [108].

3.2.8.1 DHCP Auto-configuration

DHCP servers are generally used in most networks to assign IP addresses to network devices, but DHCP can also be used by network devices to discover the details of available network services. These configuration parameters are carried in the 'Options' field of a DHCP message. RFC 2132 specifies the various DHCP Options and assigns a code for each network resource type, e.g. SIP Server Options are assigned the code 120 [125] and Simple Mail Server Options are assigned the code 69 [4]. The DHCP server itself does not automatically discover what these available services are, but the local network administrator has to manually configure them. Once this is done, network applications can query the DHCP server for these services using a DHCP DISCOVER request at boot time. This request message is broadcast in a UDP packet to the address 255.255.255.255, or to the specific subnet broadcast address. The client specifies the desired network services in the Parameter Request List of the Options field in the DHCP DISCOVER message [108].

On receipt of the DHCP DISCOVER request, the DHCP server responds with an offer of configuration parameters in a DHCP OFFER response message. Because a client can receive DHCP OFFER messages from multiple servers, the client must sent a DHCP REQUEST message specifying whose offer the client has accepted, and whose offer has been rejected. On receiving the DHCP REQUEST message from the client, the selected DHCP server sends a DHCP ACK response message which includes the lease duration and the configuration information requested by the client. The call flow for the entire process is shown in Figure 3.9 [53].

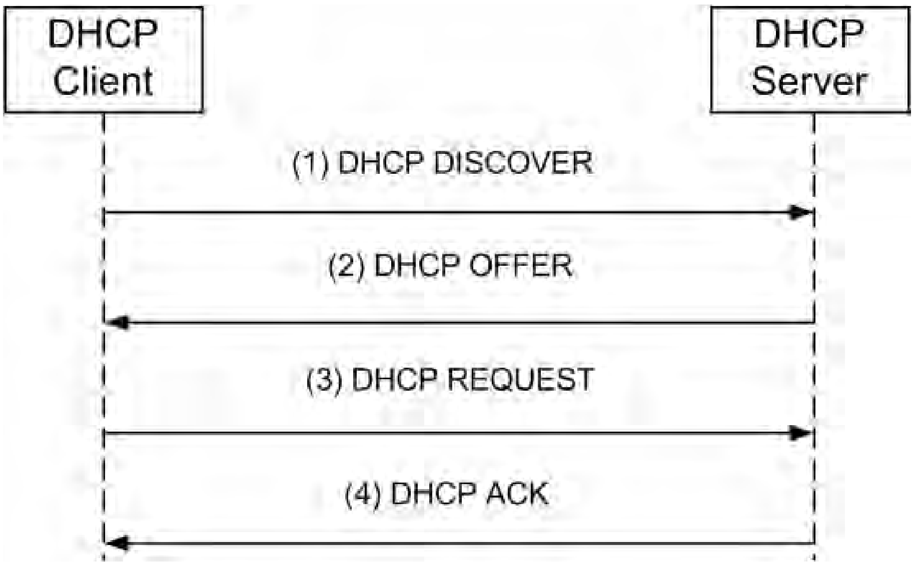

Figure 3.9: Call flow for DHCP auto-configuration.

3.2.8.2 DNS Auto-configuration

DNS can also be used by network devices to discover relevant information about available network resources. RFC 2782 describes a convention for naming and structuring DNS resource records (SRV records) [55] and Internet-Draft [33] outlines DNS based auto-configuration. Given a type of service that a client is looking for (e.g. a printing service), and a specific domain in which to search, this convention allows clients to discover a list of named instances of the desired service using standard DNS queries. Once these instances have been obtained, the client can then select a service provider instance from the DNS query results. These SRV records typically contain the port as well as the priority information for a requested resource. DNS based service discovery assumes that the UE already has the domain wherein the desired service resides. This is perhaps a drawback for DNS based auto-configuration in that the domain name has to be pre-configured or discovered via other means (like DHCP).

DNS SRV records can also be obtained by network administration tools such as the *domain information groper* (dig) utility. The following command can be used from the command line to query the DNS server for SIP SRV records in the *myexample* domain:

dig SRV _sip._tcp.myexample.com

The server will either respond with an error message if an entry of the requested resource is not available at the server, or will respond with the list of named instances of the desired service in the ANSWER section of the response message [73]. This process only requires the client dig request and a single server response, unlike the DHCP auto-configuration process which requires four messages.

Figure C.2 and Figure C.3 of Appendix C show the configuration files used to configure the DHCP and DNS servers for auto-configuration support.

3.3 Chapter Summary

This has chapter presented design considerations regarding the proposed framework and has evaluated the suitability of the framework for the major stakeholders in the proposed service. These are the telco, the user and the advertiser. This chapter found that the proposed framework provided a win-win-win scenario for these parties as they all stood to benefit in one way or the other from this implementation. The author then described the architectural requirements of the proposed system and provided implementation details.

The next chapter provides details of implementation architectures for the proposed COIN framework.

Chapter 4

System Design Evolution and Implementation Details for the COIN Framework

This chapter presents three systems developed in the effort to provide a standards compliant, CustOmisable, and INcentive (COIN) driven VoD service using the Mobicents Platform. The first or initial system presents a minimalist, proof of concept VoD system with on-demand advertising capabilities using the Mobicents Platform. The second or intermediate system builds on top of the first system to add VoD customisation support and advertiser-driven billing incentives, which are central to this thesis. The final system builds on its predecessors and presents a more modular, secure, and standards compliant architecture. This chapter also provides implementation details for some of the components used in these systems.

4.1 Initial System Architecture

During the system design phase of this research, it seemed prudent to first develop a minimalist VoD service that could later be used as a platform to develop more complex features, i.e. customisation and incentive based billing. This minimalist system would also serve as a platform for testing the SIP/RTSP signaling capabilities of the newly developed UE as described in chapter 3, as well as testing the UE's multimedia processing capabilities (video playback). This system would additionally serve to verify the choice of

VLC media server for streaming purposes and provide the initial opportunity for learning service development and deployment over Mobicents JAIN SLEE.

The basic IPTV functionality required by a client from an IPTV service provider is SD&S capability, which essentially allows the UE to attach to the IPTV service and obtain a list of videos or an Electronic Program Guide (EPG) available for viewing in Pull or Push mode. The initial system architecture is shown in Figure 4.1. This system did not perform billing transactions and had an MDF to perform streaming functions. Additional ETSI TISPAN IMS-based IPTV functional entities such as SCFs and an OCF were deemed unnecessary at this time, although they will be added in subsequent iterations. This system also did not include an MCF for simplicity, implying that control over the MDF was fully in the hands of the user. The address of the SDF-SSF was manually configured in the client to simplify implementation, hence no auto-configuration servers were required. In addition, since customisation support was also not a requirement for us at the time, the web interface used for video and advertisement classifications is also missing.

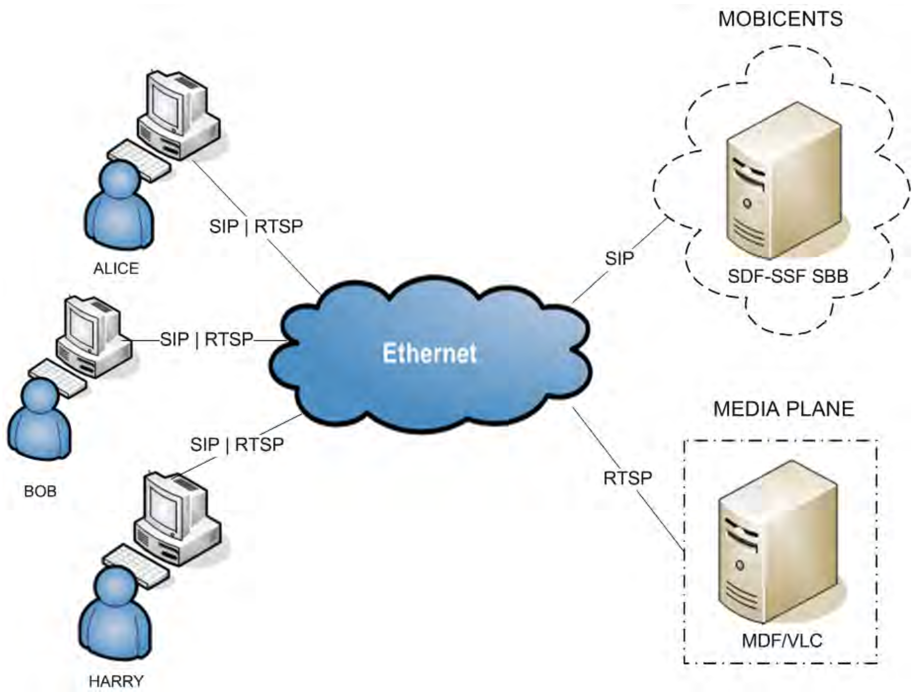

Figure 4.1: Initial System Design

To further simplify the implementation, the SDF and SSF were implemented as a single SBB (SDF-SSF) to simultaneously perform SD&S in Push Mode. In this implementation, the UE would contact the SDF-SSF using a SIP REGISTER message. This seemed intuitive since SD&S in Push mode is triggered once the client registers in ETSI TISPAN

IPTV specifications. On receiving the REGISTER message from the UE, the SDF-SSF would retrieve lists of both available videos and advertisements from a (MySQL) content database and return these lists to the UE in a SIP 200 OK response message. Mobicents JAIN SLEE comes with a basic Registrar SBB to handle SIP register requests, so this was modified to support this added functionality and become the SDF-SSF.

The base Mobicents Registrar SBB registers all clients without actually authenticating them. Under IMS conditions, the UE authenticates the username against a password using HTTP digest authentication over SIP with either Message Digest 5 (MD5) or Authentication and Key Agreement (AKA) digest algorithms. SIP/HTTP digest authentication using MD5 is specified under RFC 2831 and AKA digest authentication over SIP/HTTP is specified under RFC 3310 [75][106]. Therefore to align with IMS-based IPTV specifications, the SDF-SSF was fitted with support for MD5 digest authentication. An implementation of the MD5 algorithm can be found in the Java security package which comes with the standard Java installation, but no Java implementation of the AKA digest algorithm could be found at the time of the investigation, hence the implementation was limited to just MD5 digest authentication. The code used to calculate the MD5 hash is shown in Figure 4.2.

```
public static String MD5Hash(){
    try{
        messageDigest = MessageDigest.getInstance(DEFAULT_ALGORITHM);
        }catch(Exception e){
            e.printStackTrace();
            }
    String A1 = getName() + ":" + getRealm() + ":" + getPword();
    String A2 = getmethodInUpperCase() + ":" + getUri();
    byte mdbytes[] = messageDigest.digest(A1.getBytes());
    String HA1 = toHexString(mdbytes);
    mdbytes = messageDigest.digest(A2.getBytes());
    String HA2 = toHexString(mdbytes);
    String KD = HA1 + ":" + NONCE + ":\"" + nc;//ie nc = 1
    if (getCnonce() != null) {
        KD += "\":" + cnonce;
    }
    KD += ":" + getQop()+":"+ HA2;
    System.out.println(KD);
    mdbytes = messageDigest.digest(KD.getBytes());
    String mdString = toHexString(mdbytes);
    System.out.println("MD5 hash "+mdString);

    return mdString;
}
```

Figure 4.2: Code used to calculate the MD5 hash.

When the UE sends a SIP REGISTER request to a registration server that implements SIP/HTTP digest authentication, it responds with a SIP 401 Unauthorised error message that contains a WWW-Authenticate Header. This header includes a realm value as a string, and a randomly generated value called a nonce. Other parameters in this header include the method name, a URI, and a Qop value [19]. When the UE receives these parameters, it creates an MD5 (or AKA) hash using the client's username and password, plus the parameters retrieved from the WWW-Authenticate Header mentioned above and as depicted in the code snippet in Figure 4.2. This ash is then send back as a response to the SIP registrar server in an Authorisation Header of a SIP REGISTER message. The server will then compute its own hash value using the same parameters used by the client and a password it has obtained from a user database (we assume here that the client has already created a SIP account and selected a password, this is usually done via a web interface). If the Registrar's computed hash value matches that from the UE, that indicates that the client has entered the correct password, and the server responds with a SIP 200 OK message [19][115]. If they don't match, a registration error message gets sent to the UE.

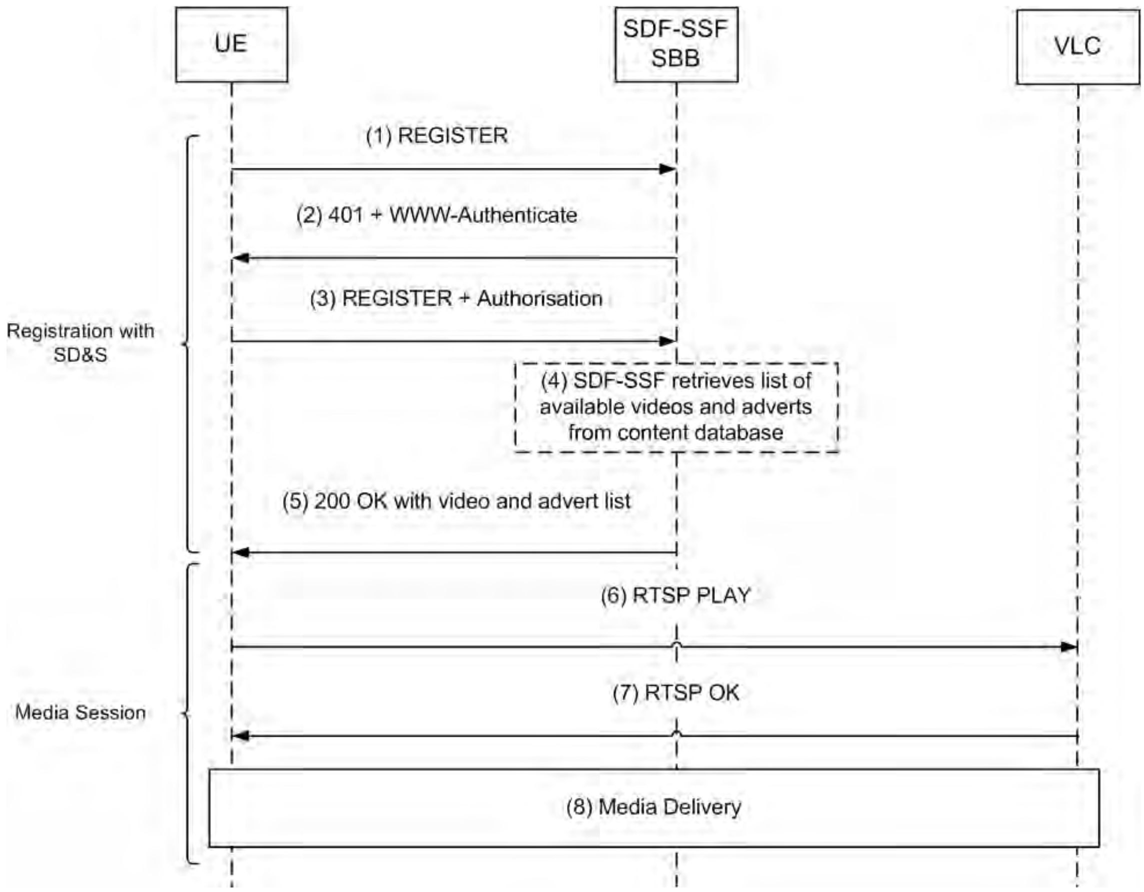

Figure 4.3: The initial system's call flow for SD&S and subsequent video session

The call flow for SD&S and subsequent video session set-up in the initial system is shown in Figure 4.3. In this system, the UE is aware of the address of the media server (VLC), but not the names of the available videos or advertisements. So once the UE has registered and has the list of videos and advertisements, it can directly contact the media server and watch the videos or advertisements on-demand. The code for the IPTV/VoD player that facilitates this implementation is shown in Figure 4.4.

```
public launchIPTVClient(){
    super("Convergence Group Video On Demand Media Player 1.0");
    initialiseIPTVClient();
    stopButton.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e){
           player.getMediaPlayer().stop();
           }
    });
    playButton.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e){
           player.getMediaPlayer().stop();
           String request=(String)combo.getItemAt(combo.getSelectedIndex());
           String url = iptvserver+"/"+request;
           player = new VideoPlayer(url);
           panel.add(player, BorderLayout.CENTER);
           player.getMediaPlayer().play();
        }
    });
    advertButton.addActionListener(new ActionListener(){
        public void actionPerformed(ActionEvent e){
            player.getMediaPlayer().stop();
            String url = iptvserver+"/"+randomAdvert();
            player = new VideoPlayer(url);
            panel.add(player, BorderLayout.CENTER);
            player.getMediaPlayer().play();
            }
    });
    /*Initialise gstreamer library*/
    Gst.init();
    container = getContentPane();
    container.setLayout(new BorderLayout());
    container.add(panel, BorderLayout.CENTER);
    container.add(controls, BorderLayout.SOUTH);
    this.pack();
    this.setLocationRelativeTo(null);
    /*Show blank screen when UE starts up*/
    String url = getCurrentDir()+"blank1sec.mpg";
    player = new VideoPlayer(url);
    player.setControlsVisible(false);
    panel.add(player, BorderLayout.CENTER);
    setVisible(true);
    player.getMediaPlayer().play();
}
```

Figure 4.4: The code for launching the UE's IPTV/VoD GUI.

4.1.1 Evaluation of the Initial System Architecture

This system allowed the viewing of videos and advertisements on demand, and provided a standards compliant approach for registering the UE using SIP/HTTP digest authentication. Through it, the author was able to verify the SIP/RTSP signaling ability of the UE as well as the VLC media server's streaming capability. In terms of the proposed framework, this system lacks the customisation support and any implementation of the advertiser driven billing incentives, which are the focal points of this thesis. Several key ETSI TISPAN IPTV functional entities are also missing in this system. For these reasons, this system is not evaluated further .

In spite of its shortcomings, this system is included for completeness since subsequent systems were build upon it and it was felt that the picture would be incomplete without mentioning it and the evaluations that were performed through it.

4.2 Intermediate System Architecture

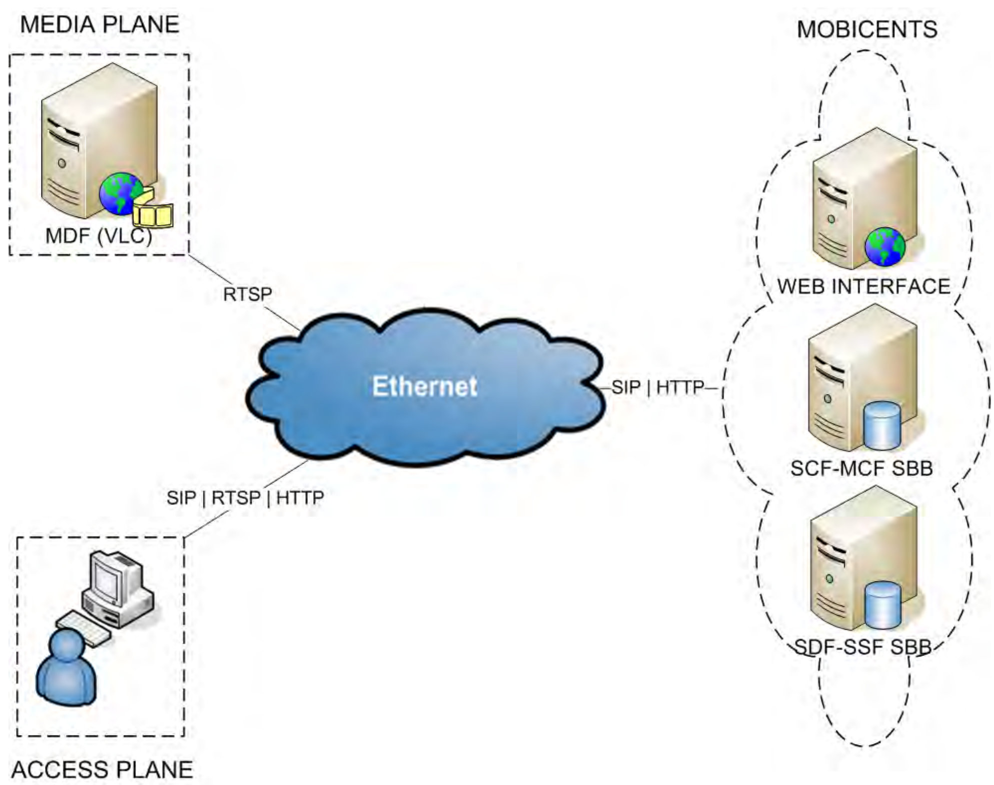

Figure 4.5: Intermediate architecture for the COIN framework

The intermediate system presented in this section is aimed at introducing customisation and advertiser-driven billing incentives on top of work already accomplished through the initial system. This system also attempts to narrow the gap between the initial system and ETSI TISPAN IMS-based IPTV specifications by introducing several standardised ETSI TISPAN IPTV functional entities omitted in the first system, but several compromises are also made to simplify implementation and reduce service latency. Figure 4.5 shows the intermediate system architecture for the proposed COIN framework and illustrates how the various components interact with the Mobicents Platform.

The intermediate system did not include an OCF charging function for simplicity, but rather used a database to store client transactional data. Other notable differences with ETSI TISPAN IMS-based IPTV specifications is that the SDF and SSF were implemented as a single SBB or server (SDF-SSF SBB) to simplify implementation. The SCF and MCF (SCF-MCF SBB) were also implemented within a single SBB for this reason. Apart from simplifying implementation, bundling these servers (or functions) in pairs greatly reduces SIP messaging load, which is large and can increase service latency [61]. To further reduce service latency, user data from the profile database and metadata on videos and advertisements from the content database were loaded into hash maps when the SDF-SSF is initially deployed, as opposed to reading the database when a request is made by the UE. This is possible in JAIN SLEE through an onServiceStarted() event handler method [80]. The onServiceStarted() event handler does not handle any incoming messages from the network, but is only triggered when the SDF-SSF is initially deployed into the JAIN SLEE container. This approach would greatly lower service latency since hash maps are stored in memory and their access times are much shorter than database access times [11].

To further simplify implementation, the UE/VoD client was not fitted with support for auto-configuration. The UE was statically configured with the addresses of the SDF-SSF as in the first system.

System users are able to customise their profiles using a web interface. This data was stored in a user profile database as mentioned in chapter 3. If the user does not customise their profile, the assumption is that the client is comfortable with all the videos and advertisements available, and hence the service will show no discretion regarding the videos or advertisements available to the user. Figure B.1 of Appendix B shows a screenshot of the web page used to customise the movie/video preferences in this architecture.

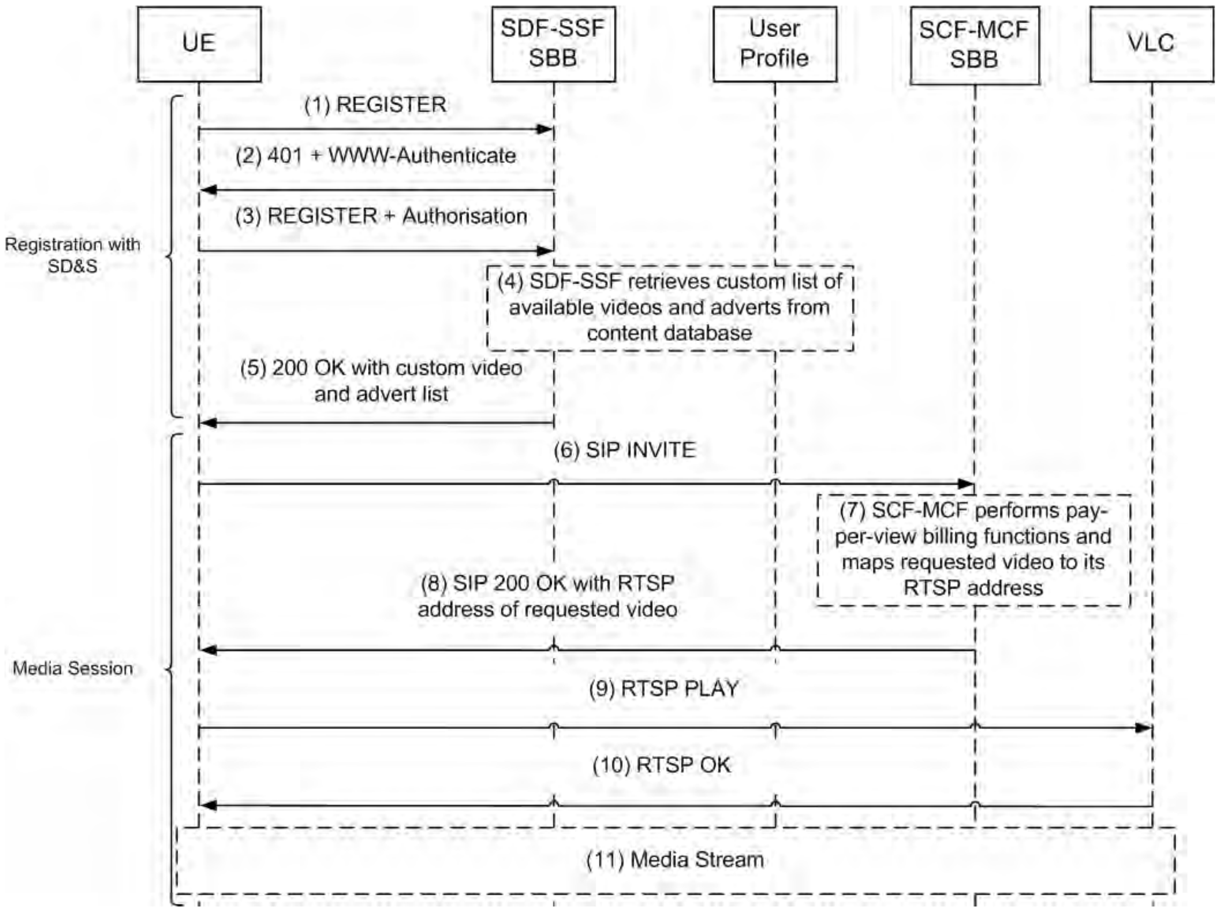

Figure 4.6: Call flow for SD&S, with subsequent media session set-up via the SCF-MCF and MDF (VLC) in the intermediate architecture

Figure 4.6 details the call flow for SD&S, followed by subsequent video request. The schematic assumes that the client has already customised their profile via the web interface, if not, the default profile is used. The entry point to the service is via the SDF-SSF, which was implemented in Mobicents as an SBB to handle SIP REGISTER messages. The SDF-SSF will primarily register the UE using their username and password, and then use the matching user profile data from the profile database to provide the UE with 'customised' or filtered VoD listings and advertisements that are in line with the user's custom preferences as shown in Figure 4.6. The code for generating the customised VoD listings used in the SDF-SSF SBB is shown in Figure 4.7, and the code for generating targeted advertisements is shown in Figure 4.8.

```
Collection d = mov.keySet();
logger.info("The movie keyset is "+d.toString());
Iterator ita = d.iterator();
while(ita.hasNext()){
    String moviekey = (String)(ita.next());
    String genre = (String)mov.get(moviekey);
    for(int k=0;k<movieTaste.length;k++){
        if(genre.equalsIgnoreCase(movieTaste[k])){
            if(vids.equalsIgnoreCase(""))
                vids+=moviekey;
            else
                vids+=","+moviekey;
            break;
        }
    }
```

Figure 4.7: Code for generating the customised VoD listings used in the SDF-SSF SBB

```
Collection c = adv.keySet();
Iterator itr = c.iterator();
while(itr.hasNext()){
    String key = (String)(itr.next());
    String value = (String)adv.get(key);
    for(int k=0;k<advertTaste.length;k++){
        if(value.equalsIgnoreCase(advertTaste[k])){
            if(adds.equalsIgnoreCase(""))
                adds+=key;
            else
                adds+=","+key;
            break;
        }
    }

}
```

Figure 4.8: The code for generating targeted advertisements used in the SDF-SSF SBB.

The SCF-MCF is implemented as an SBB with an event handler method for SIP INVITE messages. The SCF-MCF is designed to simultaneously perform the SCF and MCF functions within a single SBB, and in one step. Clients send the requested video string (or name) to this SBB in a SIP INVITE message after registering, the SCF-MCF then performs the necessary credit checks to determine if the client has sufficient credit for the transaction and verify if the UE is registered. The credit information in this case is stored in a back-end database. In the event that the user has sufficient credit, the SCF-MCF then acts as a redirect server for the MDF and maps the requested video to its RTSP address on the MDF as discussed earlier. The RTSP address is therefore sent to the UE in a SIP 200 OK message. Table 4.1 shows an example mapping between a video and its URL. Upon receipt of the RTSP address within the SIP 200 OK message, the UE would immediately request the stream from the streaming server or MDF, in this case VLC.

This is shown in Figure 4.6 as well. If the user has insufficient credit, a SIP error message is sent back to the client. An error message is also generated if the UE is unregistered.

Video Name	URL/RTSP address on streaming server
Forrest Gump	rtsp://media_server.testbed.ru/forrest_gump

Table 4.1: Example mapping of a video to its URL on the streaming server.

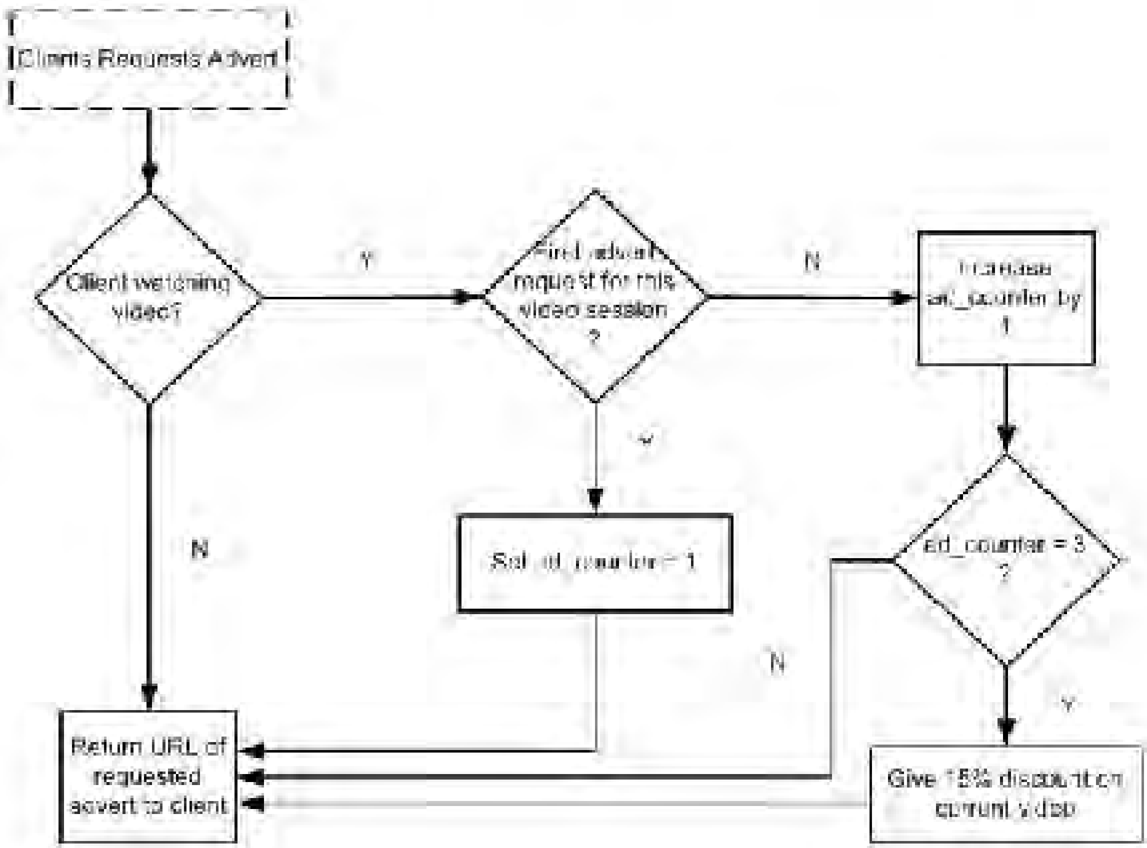

Figure 4.9: Decision algorithm for granting advertiser driven incentives.

The UE can make an on-demand request for advertisements at any time by clicking on the 'advertisement button' in the UE. Clicking this button triggers a 'custom' or targeted advertisement requests to the SCF-MCF. These advertisement requests are sent to the SCF-MCF in a SIP INVITE message, just like a normal video request as described above, but instead of checking for user credit, the SCF-MCF is set up to run the decision algorithm shown in Figure 4.9 for advertisement requests. The SCF-MCF can differentiate between advertisement requests from movie/video requests because of the naming or identifiers used for video and advertisement files. This algorithm will determine whether or not to grant a billing discount or incentive to the user for requesting the advertisement. Once this is one done, the SCF-MCF returns the URL of a targeted advertisement to the client, which would pause any existing VoD session, and play the advertisement. After playing the advertisement, the client can resume the previous media session (if any existed). The call flow for this process is shown in Figure 4.10.

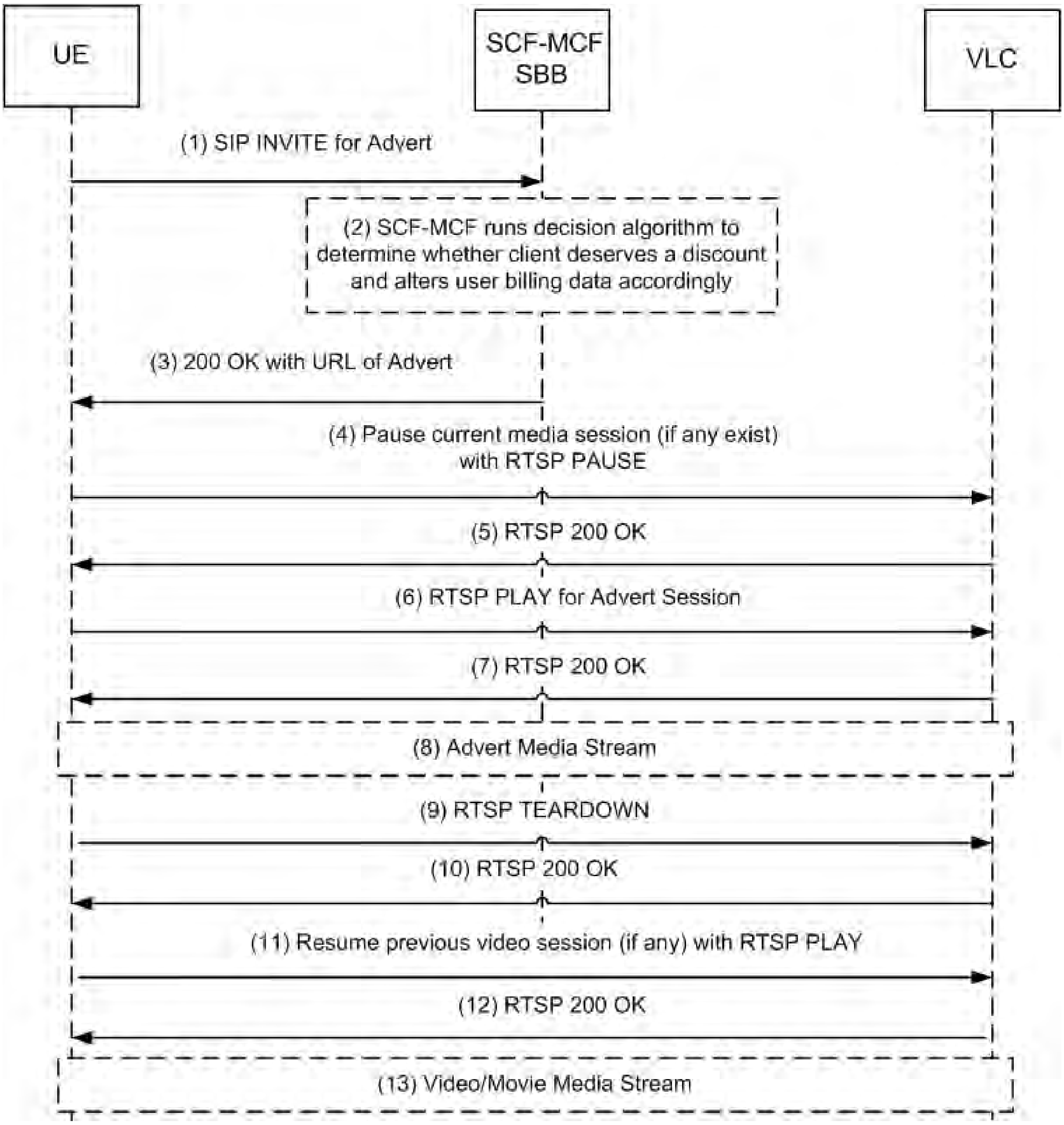

Figure 4.10: Call flow for advertisement requests in the intermediate architecture

4.2.1 UE Changes

To support the architectural changes detailed above, the IPTV/VoD interface needed to trigger SIP INVITE messages for videos or advertisements to the SCF-MCF, rather than make RTSP requests directly to the MDF as in the previous system.

A separate thread was also created to listen for the SIP 200 OK response message from the SCF-MCF containing the RTSP address of the requested video or advertisement, and pass this to the IPTV/VoD interface for playback from the streaming server.

4.2.2 Evaluation of the Intermediate System Architecture

The intermediate system provides a proof of concept implementation of the COIN framework. The key strength of this system is that it minimises 'SIP messaging tax' by bundling some of the servers together, but consequently means that the system is not in line with ETSI TISPAN IMS-based IPTV specifications. The ETSI TISPAN IMS-based IPTV is comparatively more modular in terms of the implementation of the SDF, SSF, MCF and SCF. The intermediate system also used a non-standard interface in the web pages used to customise the VoD listings. An HTTP interface is typically only used for service selection purposes in Pull mode [42].

A key security flaw exists in the decision to use a database to hold client transaction data. ETSI TISPAN IMS-based IPTV specifications use Diameter for this purpose. As mentioned in chapter 2, Diameter was introduced to cater for growing AAA requirements in computer networks, and hence presents a more secure approach of handling and storing client billing transactions data, as compared to database queries and database storage.

Another security flaw exists in the manner in which media sessions are set up with the MDF (VLC), i.e. using the MCF as a redirect server. Because the MDF is not controlled by the MCF, any user may request a video stream from the MDF if they can correctly guess the RTSP address of whatever video they want to watch, and request it directly from the MDF using RTSP commands. This can result in Denial of Service (DoS) attacks [130].This approach for setting up media sessions with the MDF also means that a user can bypass the SDF-SSF and SCF-MCF procedures, and watch the video directly from the MDF, without actually paying for it.

The intermediate system also used a Push method for service selection. This approach demands more bandwidth in that the entire customised VoD listing is sent to the client when it registers. A bandwidth efficient approach would be using a Pull approach over an HTTP/XCAP interface. An XCAP-DIFF command over HTTP can be used to retrieve only the changes in the VoD listings, rather than downloading the entire list, and hence save bandwidth. An XCAP-DIFF is defined by the XCAP standard and is used to track changes in an XML document.

4.3 Final System Architecture

In response to the evaluations of the intermediate system, we decided to make changes and present a modular, standards based, and secure architecture for the proposed framework. To achieve this, several architectural changes needed to be made to the intermediate design. These design changes also necessitated a few changes in the UE/VoD client. Figure 4.11 shows the final system architecture.

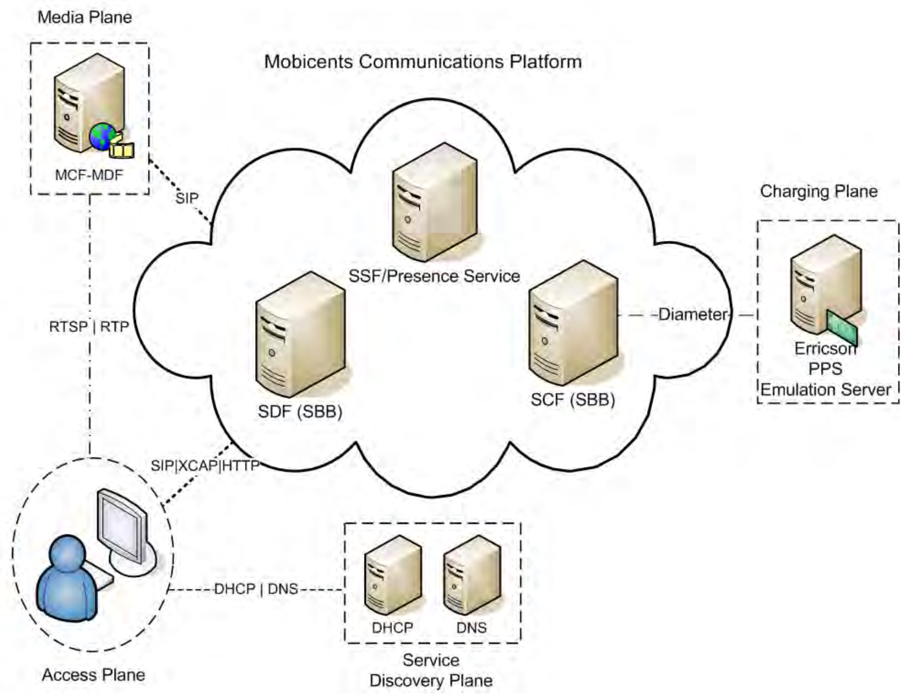

Figure 4.11: Final architecture for the COIN framework.

First of all, to make the system modular, the SDF and SCF were implemented as individual SBBs within Mobicents. These entities would be dependent on the SIP RA since they both require SIP interfaces. These entities will perform their standardised IPTV functions as discussed in chapter 2 and as specified in ETSI TISPAN IMS-based IPTV specifications.

In addition to modularising the implementation of the SDF and SCF, the SSF was implemented as a separate entity within the Mobicents Presence Service (MPS). This decision was taken to make use of the XDM server contained within the MPS. This would allow the UE to access customised service selection information in the form of XML documents on the XDM server via an HTTP/XCAP interface in Pull mode. For this, an Application Usage (AU) was defined in the MPS. An AU essentially defines what a server application needs to do to be used with XCAP. Setting an AU involves several steps that include defining a unique AU identifier, defining an XML schema for the data, and specifying authorisation rules [120][121]. An XML schema is used to define the constraints in an XML document, in this case the XML documents contain information about the available videos. Figure C.4 of Appendix C shows the XML schema used to describe videos in the MPS. To load the names and attributes of the videos and advertisements into the XDMS, the author used an XCAP client API that comes bundled with the Mobicents Platform. The main advantage of this approach is that it allows the client to set customisation parameters within the UE itself, rather than using a non-standard web interface as in the previous system. Pull mode is also bandwidth efficient because it allows the use of XCAP-DIFF commands which only retrieves changes in the available service offerings, rather that download the entire offering each time the client boots up. Service Selection using HTTP is specified in [40]. The decision algorithm for granting discounts described in the previous system remains essentially the same and is implemented in the SCF.

The author also managed to mitigate the security flaw introduced by the use of a redirect server for the MDF in the initial and intermediate systems. Researchers at the University of Cape Town (UCT) in South Africa had encountered the same problem and in response had developed a media function that would deliver media according to the ETSI TISPAN IPTV specifications as shown in Figure 2.6 and Figure 2.7. Through collaboration with them, their implementation of a media function, the UCT MCF-MDF, was made available to us. The UCT MCF-MDF was integrated into the final system architecture as shown in Figure 4.11. Figure 4.12 and Figure 4.13 shows call flow for media session set-up using Method 1 and 2 for the testbed, using the UCT MCF-MDF as the MF (note, both figures abstract against any signaling between the MCF and MDF of the MF, but the MCF within the MF is able to control the MDF as required by the IPTV specifications). In both figures, the SCF acts in B2BUA mode. Pay-Per-View billing logic is triggered in the SCF when a video request is made using a SIP INVITE as in the previous system.

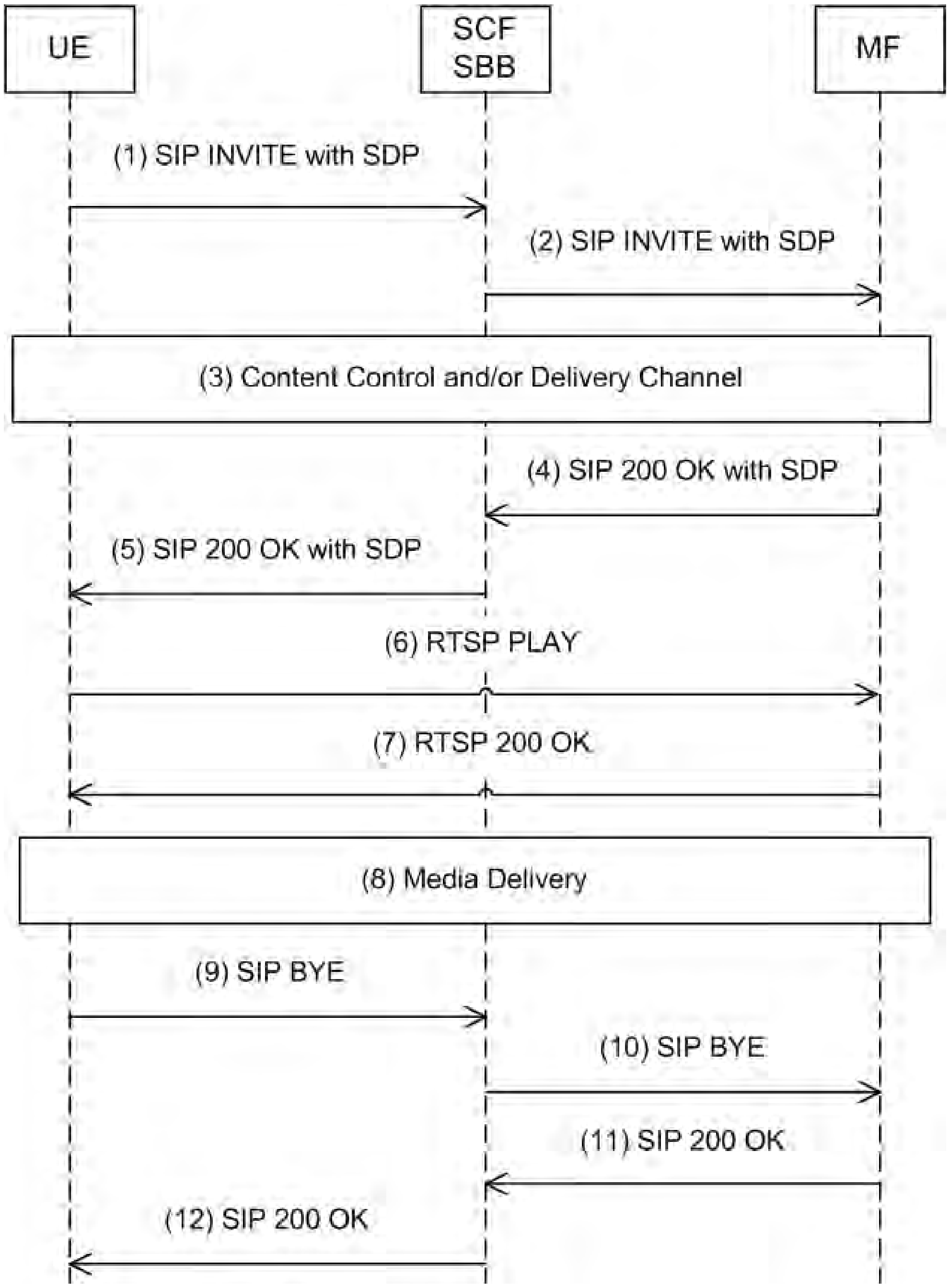

Figure 4.12: Trick-Play Session in the final system using Method 1 and the UCT MCF-MDF as the MF

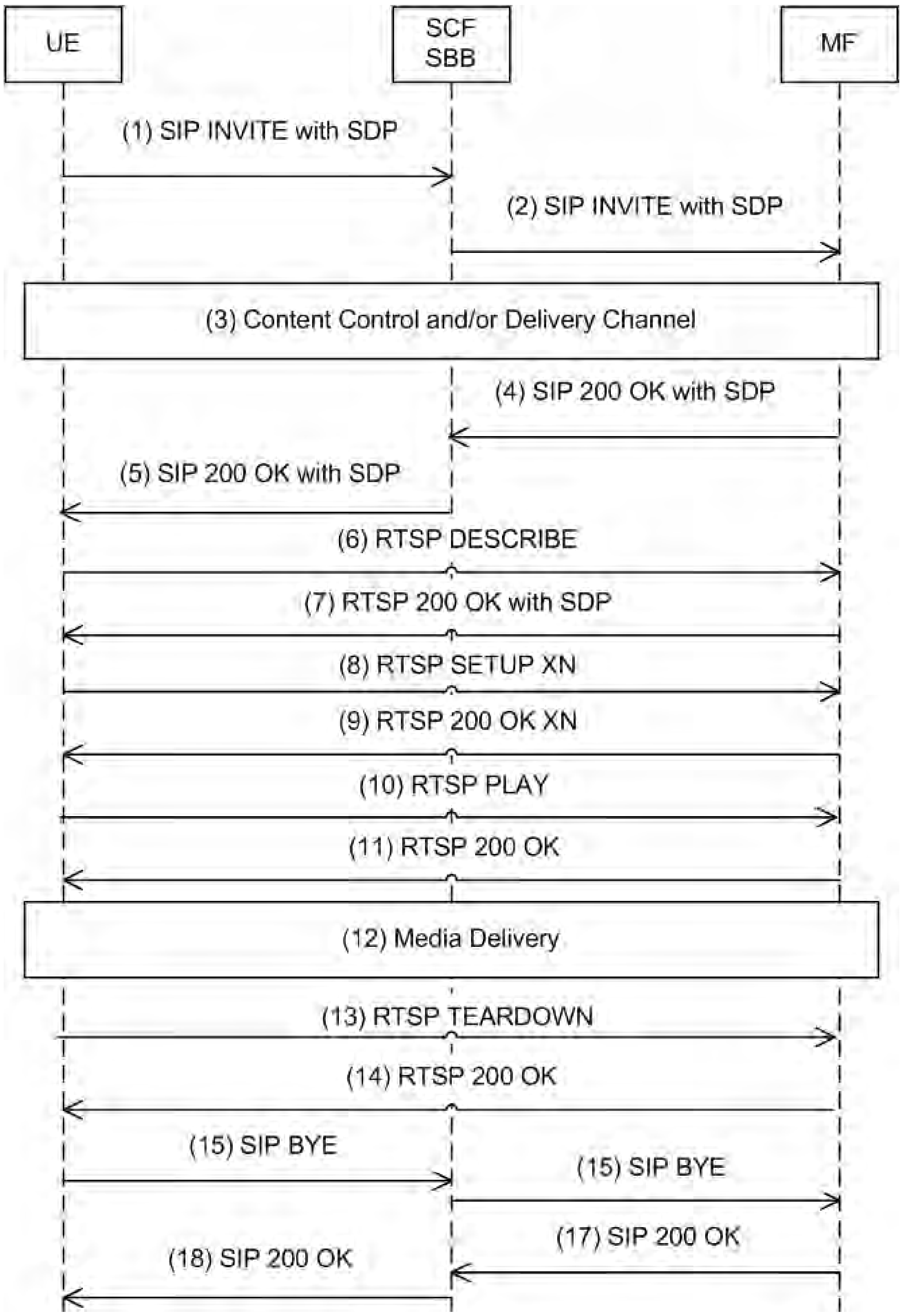

Figure 4.13: Trick-Play Session in the final system using Method 2 and the UCT MCF-MDF as the MF

Another key change made to the intermediate system was the introduction of a Diameter based OCF charging function, which falls in line with ETSI TISPAN IPTV specifications and 3GPP IMS standards. For the OCF, the Ericsson Pre-paid System (PPS) was used.

The Ericsson PPS is a Diameter emulation server released by Ericsson under the Diameter Charging Software Development Kit (SDK). The Ericsson PPS server provides a real time, pre-paid charging system and is written in Java [143]. A researcher in the same research group as the author, Moses Nkhumeleni, had developed a software library for billing Mobicents based services using the Ericsson SDK. This library was integrated into the SCF SBB and set-up to point to an Ericsson PPS Diameter emulation server as shown in Figure 4.11. Figure B.2 of Appendix B shows a screen-shot of the Ericsson PPS server, along with the UE and the Mobicents console.

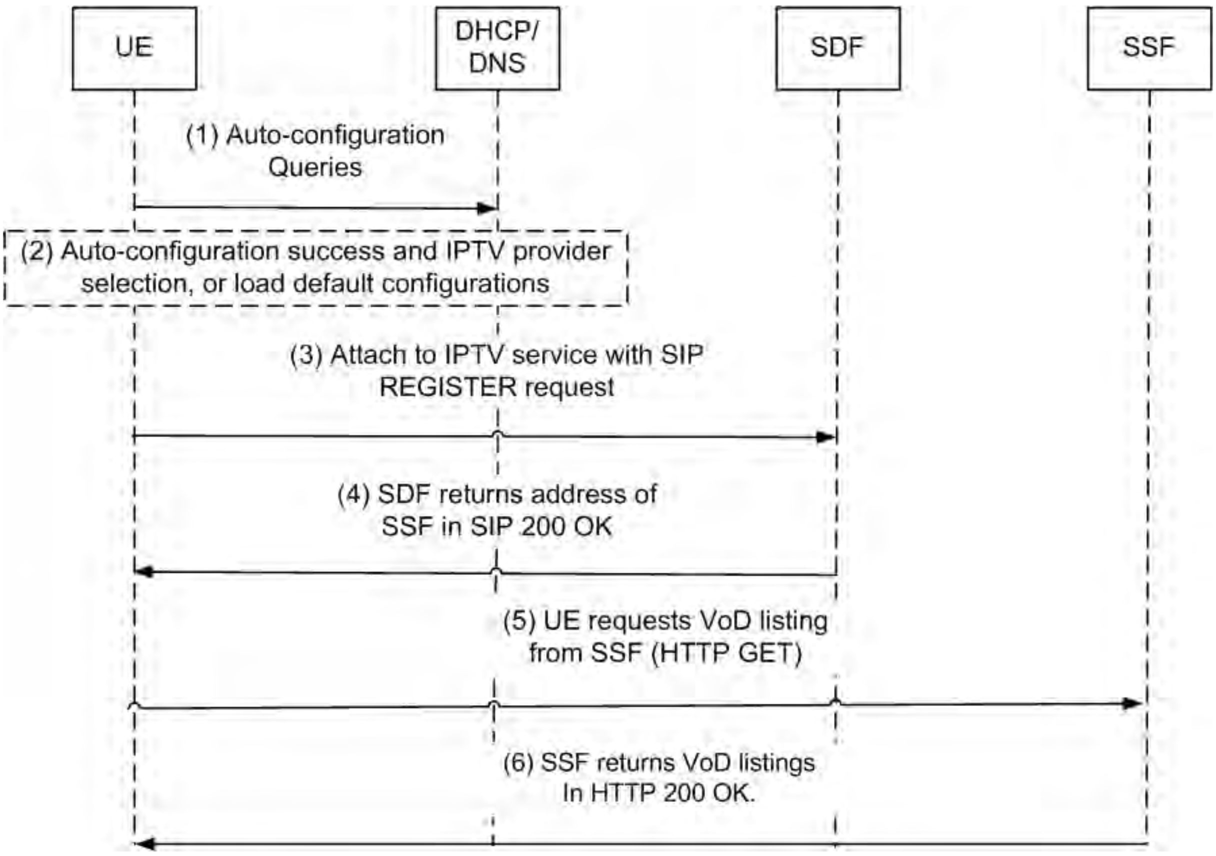

Figure 4.14: Auto-configuration process with subsequent SD&S in the final system.

The final system also added support for auto-configuration through DHCP and DNS servers. With this approach, there would be no need to manually configure the address of the SDF into the client as this would be discovered automatically over the network. In the implementation, the DHCP server was the primary respondent for auto-configuration queries, with the DNS server acting as an auto-configuration fail-over server as the author described in [144]. In the testbed, the open source DHCP server, *dhcp-3* was used for DHCP auto-configuration and the open source *BIND* server for DNS auto-configuration

[23][24]. If both DHCP and DNS auto-configuration failed, the UE would resort to a statically configured default configuration. The configuration files used to specify auto-configuration parameters in the DHCP and DNS servers to are shown in Appendix C. The call flow for the auto-configuration process and subsequent SD&S is shown in Figure 4.14.

The final architecture is evaluated in chapter 5.

4.3.1 UE Changes

Two key changes to the intermediate system necessitated changes to the UE. These were the implementation of the SSF within the MPS, and the introduction of DHCP and DNS auto-configuration servers. The implementation of the SSF within the MPS meant that the UE would need to support an HTTP/XCAP interface in order to customise viewer preferences in the form of XML documents stored on the XDMS, and to retrieve custom VoD listings and advertisements from the XDMS. Due to time constraints however, we were unable to develop an implementation that could retrieve customised VoD listings or advertisements directly from the MPS using XCAP commands, nor could we implement the XCAP-DIFF commands that could save on bandwidth. So rather we went for the simpler approach of retrieving the entire VoD listing and the advertisements from the MPS using HTTP, and performing the customisation or filtering process within the client.

To facilitate the retrieval of the VoD listings and advertisement lists from the MPS, the Apache HTTP library was added to the UE [46]. The code used to query the MPS from within the UE using the Apache HTTP library is shown in Figure 4.15. In addition, querying the MPS from an HTTP client requires that the client first authenticate using HTTP digest authentication and MD5. Since we already had a function to calculate this for user registration as described in the initial system, the function was reused.

```
HttpClient client = new HttpClient();
setUri(url);
GetMethod getMethod = new GetMethod(url);
boolean done=false;
getMethod.getParams().setParameter(HttpMethodParams.RETRY_HANDLER,
    new DefaultHttpMethodRetryHandler(3, false));
do{
    try{
        int statusCode = client.executeMethod(getMethod);
        if (statusCode != HttpStatus.SC_OK)
            System.err.println("Method failed: " + getMethod.getStatusLine());
        switch (statusCode) {
        case HttpStatus.SC_OK: /*Success!*/
            System.out.println(getMethod.getResponseBodyAsStream().toString());
            /*Read the response body*/
            byte[] responseBody = getMethod.getResponseBody();
            extractOffering(new String(responseBody));
            done=false;
            break;
        case HttpStatus.SC_UNAUTHORIZED:
            extractParams(getMethod.getResponseHeader("WWW-Authenticate").
                getValue());
            String resp = a.generateAuthResponseString();
            done=true;
            getMethod.addRequestHeader("Authorization", resp);
            break;
        case HttpStatus.SC_TEMPORARY_REDIRECT:
        case HttpStatus.SC_MOVED_TEMPORARILY:
        case HttpStatus.SC_MOVED_PERMANENTLY:
            url = getMethod.getResponseHeader("Location").getValue();
            System.out.println("Redirect: " + url);
        default:
            done = false;
        throw new IOException("Response status not OK:" + statusCode);
        }
    }catch(Exception e){
        e.printStackTrace();
    }
}while(done);
```

Figure 4.15: UE HTTP client code for retrieving VoD listings from the MPS.

Finally, to support DHCP and DNS auto-configuration, Java based DHCP and DNS libraries had to be added to the UE. For DHCP, the open source jDHCP-1.1.1 library provided DHCP messaging capability, while the open source dnsjava-2.0 library provided DNS messaging support [68][149]. For DHCP auto-configuration, the client was configured to request for the address of the SDF/SIP server. This meant that the DHCP DISCOVER message broadcast from the client would have a SIP server entry in the Parameter Request List of the Options field. Since DHCP auto-configuration settings are finally sent to the client in a DHCP ACK message, the requested parameters must be extracted on receipt of a DHCP ACK.

For DNS auto-configuration to work, the UE has to be know beforehand the domain in which to search. In order to simplify implementation, the domain was statically configured into the UE. As mentioned in chapter 3, the dig utility can be used to query name servers (DNS). The dnsjava-2.0 library allowed the UE to issue dig requests to the local DNS server for the desired SDF/SIP SRV details.

4.4 Chapter Summary

This chapter presented three iterations of the proposed COIN framework. An initial system was set up to mainly provide a base VoD framework for the Mobicents Platform with on-demand advertising capabilities. With it, it was possible to test the UE for SIP/RTSP signaling capabilities and video playback functionality. The MDF/VLC was also tested against the UE to confirm its streaming capabilities. The second system was built on top of the initial system to add customisation support and incentive-based billing. The final system was also built on its predecessors to present a more secure and standards compliant system. Table 4.2 summarises the major differences between these systems.

Property/ System	Initial System	Intermediate System	Final System
IPTV Functional Entities	SDF-SSF and MDF	SDF-SSF, SCF-MCF and MDF	SDF, SSF, SCF and MCF-MDF
Customisation Support	No	Yes	Yes
Advertiser Incentives	No	Yes	Yes
Service Selection Mode	Push Mode (SIP)	Push Mode (SIP)	Pull Mode (HTTP)
Auto-configuration Support	No	No	Yes
Mobicents Servers Used	JAIN SLEE	JAIN SLEE, JEE	JAIN SLEE, MPS

Table 4.2: Summary of system implementation architectures

The next chapter will evaluate and analyse the final system architecture.

Chapter 5

Evaluation and Analysis

This chapter focuses mainly on evaluating how the design of the final architecture could impact on the user's QoE, rather than focusing on its architecture, standards compliance, or security. ETSI TS 181 016 defines QoE as "a purely subjective measure from the user's perspective of the overall value of the service provided". ETSI TS 181 016 goes on to state that apart from being user dependent, QoE can also be influenced by the user's terminal device, his environment, his expectations, and the nature of the content and its importance [41]. This chapter will present evaluations and analysis on three key areas that the author felt could influence the user's QoE.

5.1 Evaluation Methodology

This chapter will use both experiments and relevant literature to evaluate the final architecture. Experiments would be used as the primary tool for evaluations and analysis, but if relevant literature exists on similar experiments, then relevant literature would take precedence over conducting experiments. This is because the author felt that repeating experiments already done elsewhere, perhaps under near identical circumstances, would be akin to re-inventing the wheel and would not add to the body of knowledge. The physical experiments that are performed will be described in detail so that results obtained herein can be duplicated and verified.

The three evaluations that will be conducted will focus on service latency, the impact of the advertising framework on QoE, and the impact of DHCP/DNS auto-configuration on UE start-up delay.

5.1.1 Service Latency

Service latency refers to the delay from the time the client requests content, to the time playback starts. This can also be called session set-up delay. According to [154], the top three reasons why IPTV providers lose customers are slow channel switching, poor image quality and slow boot-up time in STBs. Therefore in order to ensure an enhanced QoE, the target platform must be aimed at reduced service latency. In the final architecture, the UCT MCF-MDF was used for media delivery, but experiments on service latency using the UCT MCF-MDF have already been done [91][151]. Therefore the focus will instead be on call flow signaling efficiency. The assumption here is that a minimum number of messages exchanged during session set-up will generally result in lower service latency, which in turn will lead to enhanced QoE as discussed above.

Figure 5.1 and Figure 5.2 show side by side comparison of both Method's 1 and 2 for ETSI TISPAN IMS-based IPTV and the Mobicents based IPTV solution.

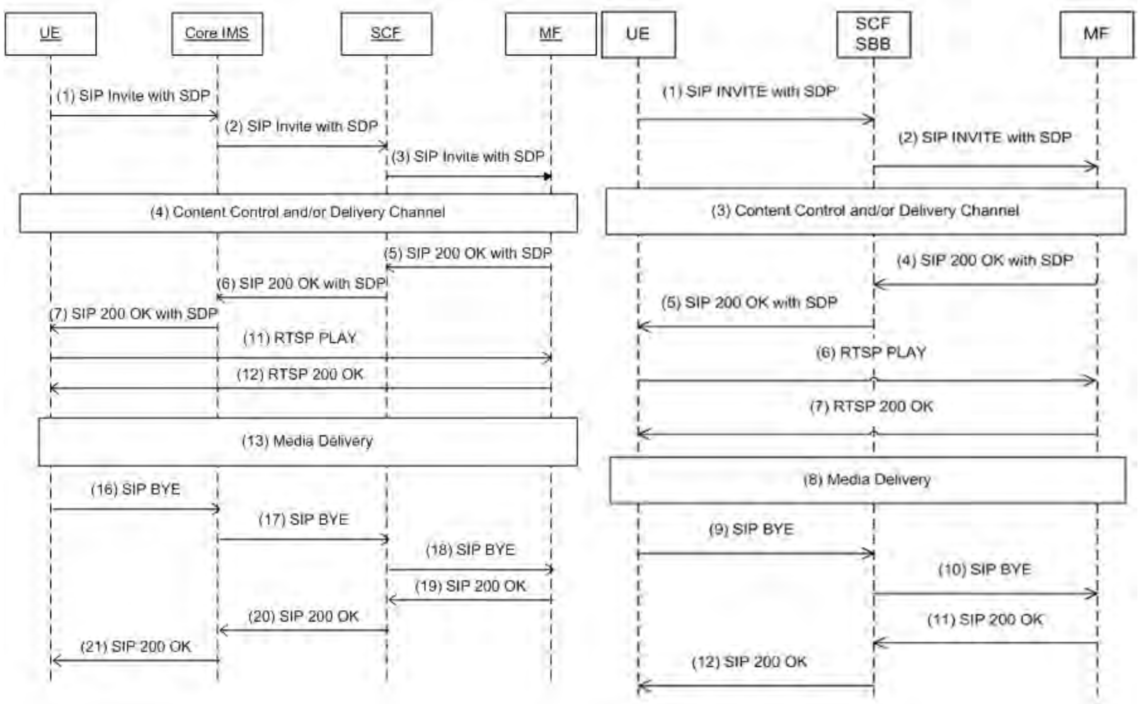

Figure 5.1: Comparing Method 1 implementation in IMS-based IPTV vs Method 1 in Mobicents based IPTV.

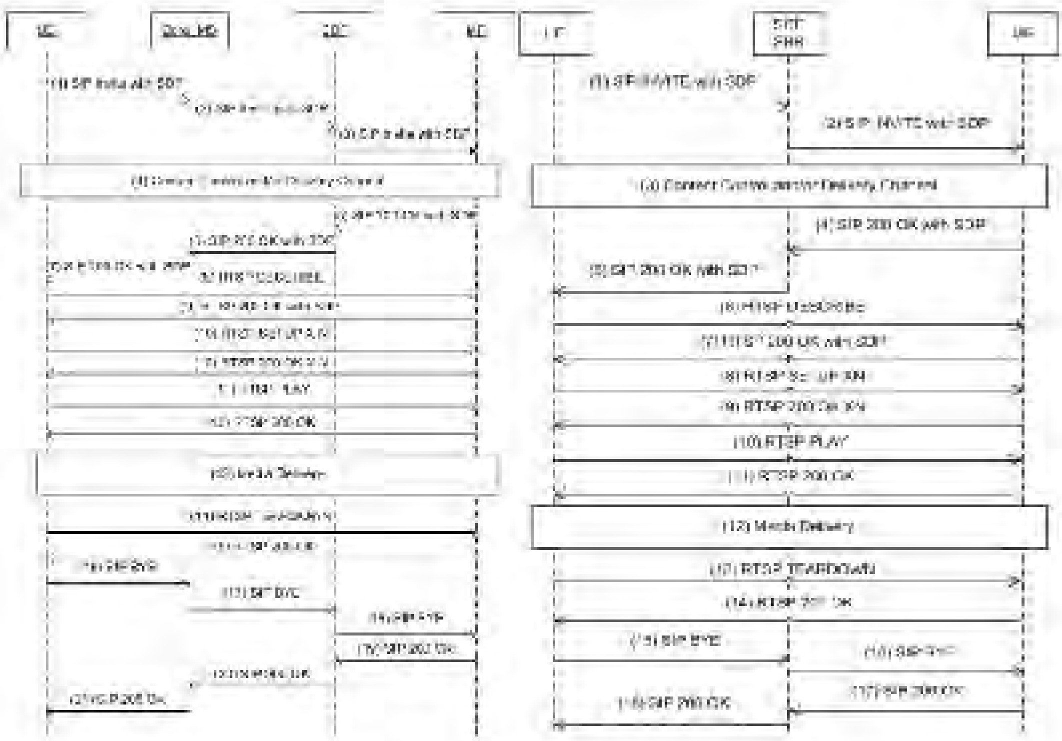

Figure 5.2: Comparing Method 2 implementation in IMS-based IPTV vs Method 2 in Mobicents based IPTV.

As can be observed, IMS-based IPTV involves more signaling than the Mobicents based IPTV solution. This is because extra signaling is needed to communicate with the IMS control layer, whereas in the Mobicents based IPTV set-up, only the SCF is involved. The total count for messages exchanged for IMS-based IPTV using Method 1 is 16, whereas the count for Mobicents based IPTV is only 12. As for Method 2, IMS-based IPTV needs 22 messages to complete session signaling while the Mobicents based implementation uses only 18. Hence the Mobicents based implementation is more efficient in terms of signaling efficiency as compared to IMS-based IPTV.

5.1.2 Impact of Advertising on QoE

The proposed framework uses an on-demand approach to advertising, meaning that advertisements will only be streamed to the client on request. If a user feels that advertisements are intrusive, then they are not compelled to request them, though the billing incentive is meant to encourage users to view them. In terms of service latency for on-demand advertisements, the advertisements are located on the same server as the video content and the call flow signaling for advertisements is the same for video requests, therefore the same QoE will likely be encountered as with video requests. Additionally, the decision algorithm for granting billing discounts is only performed when the client ends the advertisement session, unlike in the first system when the algorithm was triggered upon service request with a SIP INVITE message. In this case, the decision algorithm is triggered with a SIP BYE request for ending the advertisement session from the UE. Figure 5.3 shows Method 1 call flow for setting up and termination the advertisement session as well as highlighting when the decision algorithm for granting billing discounts is triggered. The call flow is analogous for Method 2. This approach will mean the decision algorithm will not add any latency to the setting up of the advertisement session itself and will therefore not impact negatively on the QoE.

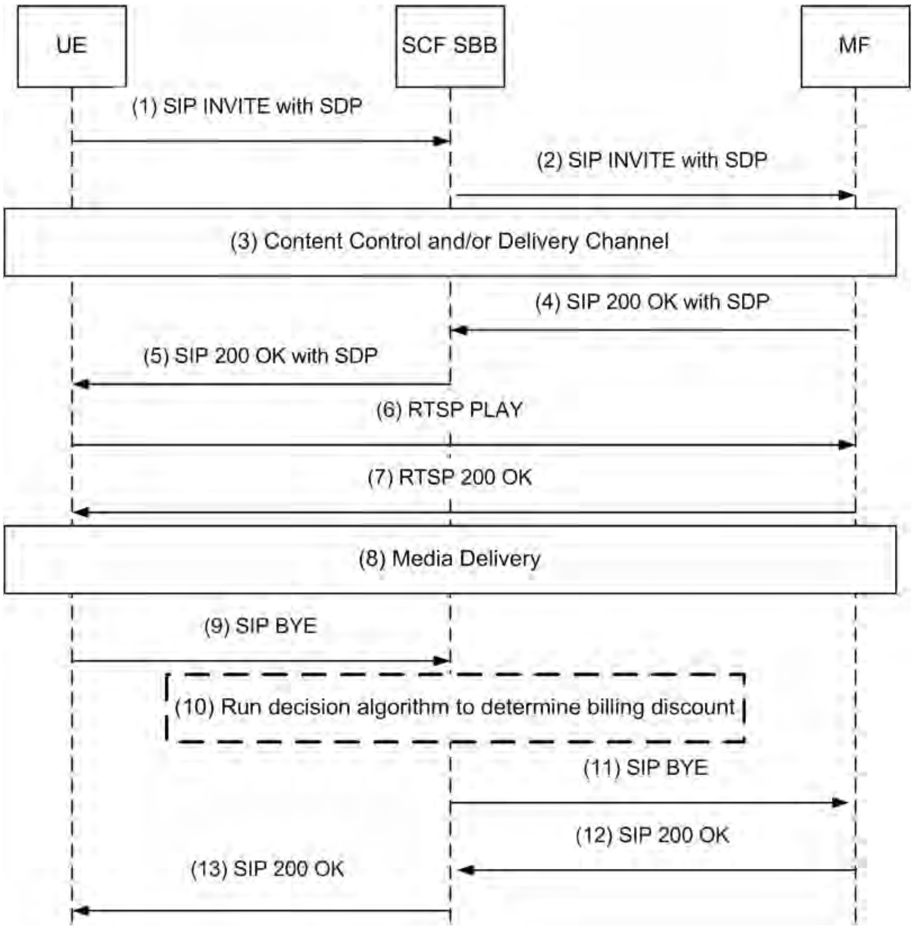

Figure 5.3: Call flow for setting up and termination the advertisement session with the MF

5.1.3 Impact of Auto-configuration on UE start-up time

The benefits of auto-configuration were discussed in chapter 3. Auto-configuration however slows down UE start-up time. This is because several messages have to be exchanged between the client and the auto-configuration server before the configuration parameters can be accepted by the client. This section will evaluate the impact of DHCP and DNS auto-configuration procedures on UE start-up time. This is important for the QoE since users generally don't enjoy lengthy waiting periods as applications boot-up [154].

In the final system, DHCP is used as the primary auto-configuration server because it allows the UE to discover virtually all network parameters without the assistance of any other auto-configuration server. The call flow for successful DHCP auto-configuration and subsequent SD&S is shown in Figure 5.4.

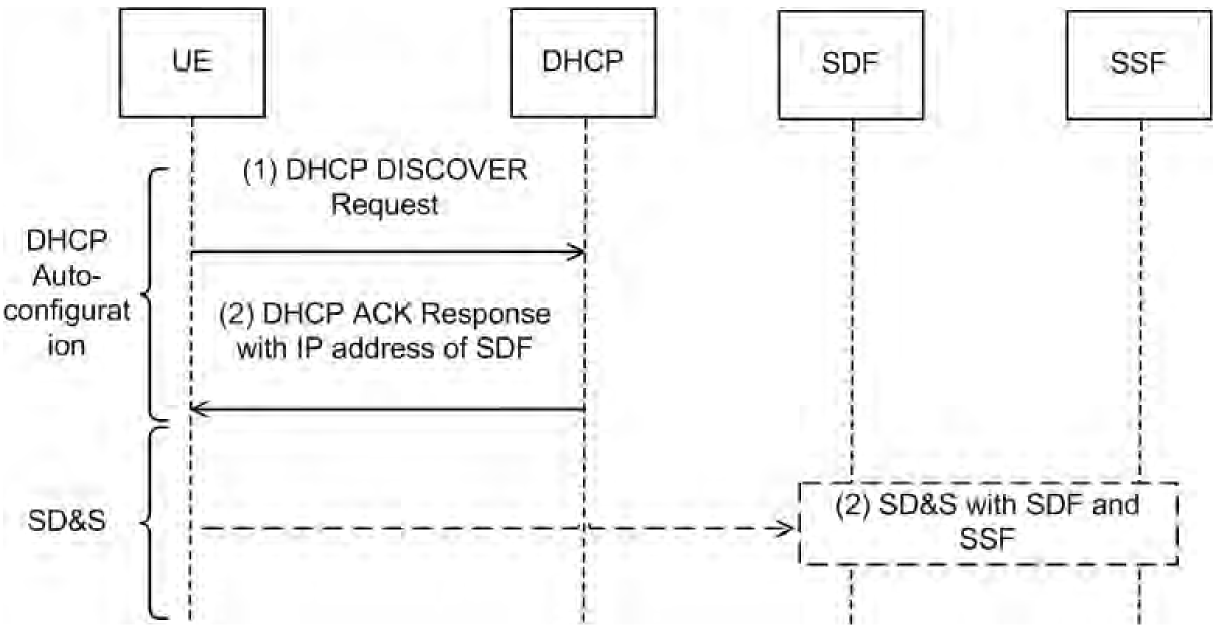

Figure 5.4: Call flow for successful DHCP auto-configuration request, with subsequent SD&S.

DNS on the other hand needs to know which domain to search before requesting the auto-configuration parameters. The domain name can of course be found using DHCP or can be manually configured into the UE beforehand. For these reasons, DHCP auto-configuration was chosen ahead of DNS auto-configuration. Hence DNS provides a fallback mechanism in case of DHCP auto-configuration failure. Figure 5.5 shows the call flow for DNS fallback auto-configuration.

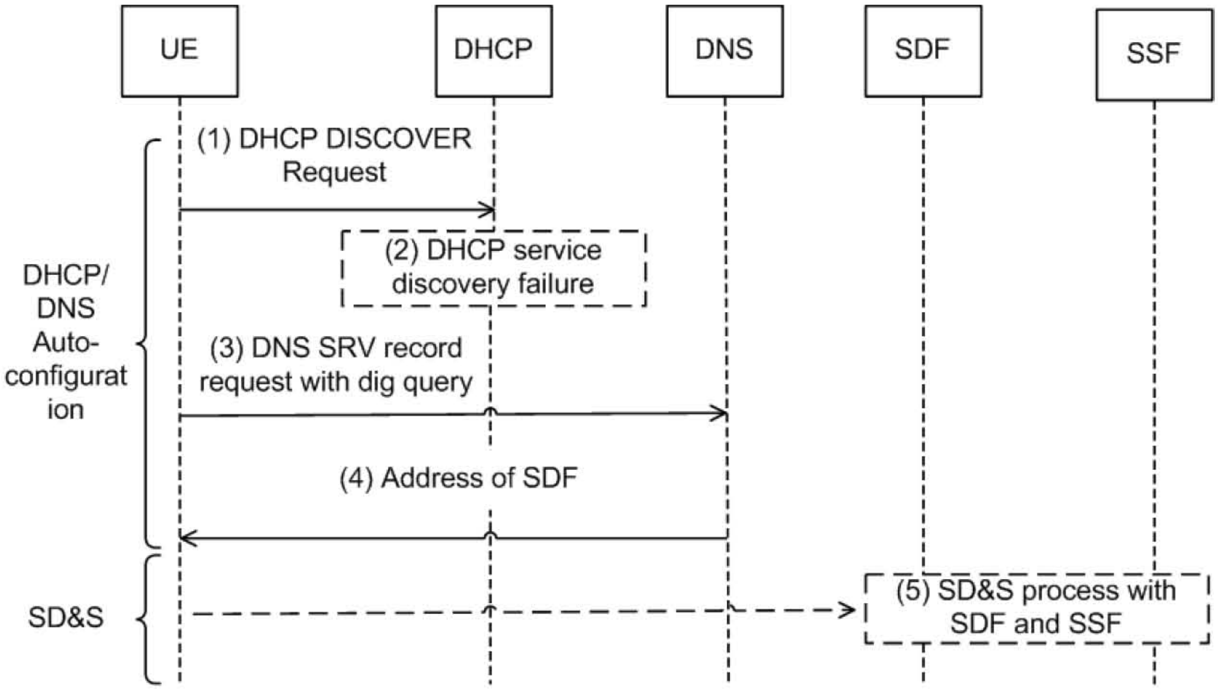

Figure 5.5: Call flow for successful DNS fallback mechanism after DHCP failure, with subsequent SD&S.

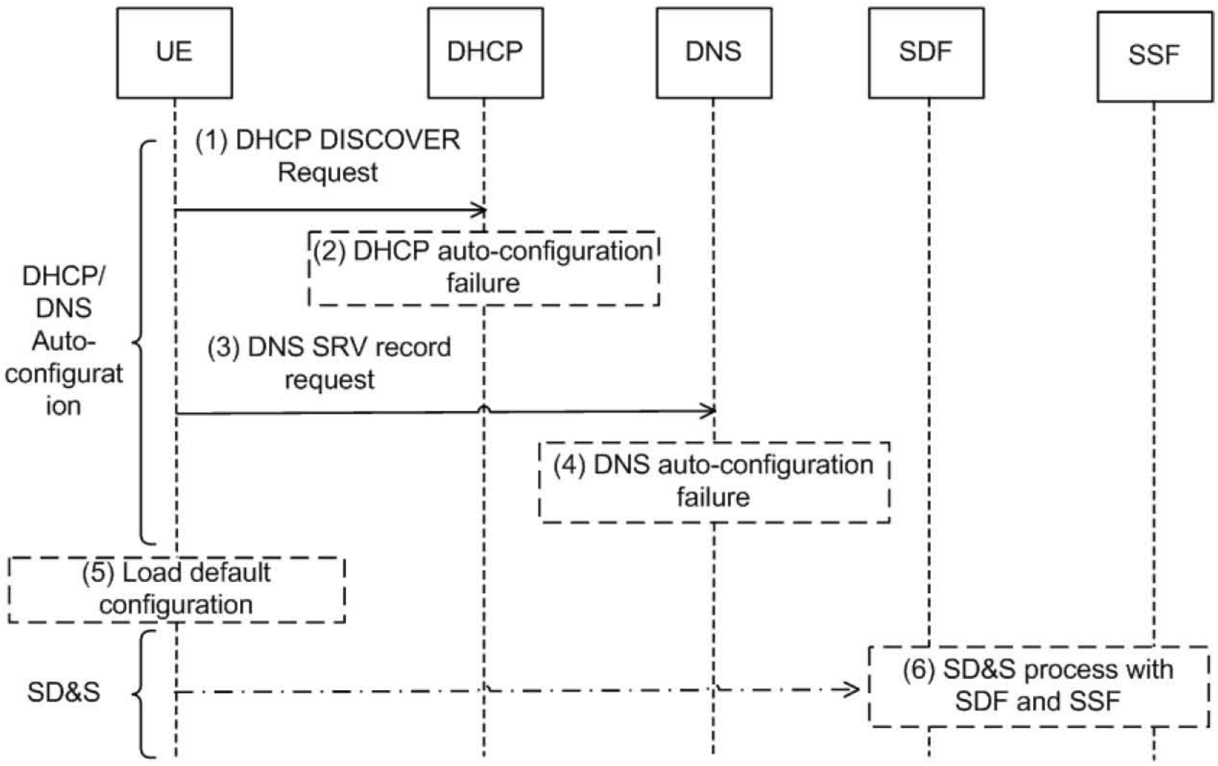

Figure 5.6: Call flow for DHCP and DNS auto-configuration failure, followed by loading the default configuration and subsequent SD&S.

In the event of DNS fallback failure, the UE would use statically/manually configured parameters. Figure 5.6 shows the call flow scenario leading to static configuration after DHCP and DNS fallback failure.

Experiments were conducted to determine the impact of DHCP/DNS auto-configuration on UE start-up, Table 5.1 shows average times for both DHCP and DNS fallback auto-configuration.

DHCP auto-configuration (ms)	DHCP failure + DNS fallback auto-configuration (ms)
560	843

Table 5.1: Average DHCP auto-configuration delay versus average DNS fallback auto-configuration delay in milliseconds (ms).

The DNS fallback auto-configuration used here assumes that the client is already aware of which domain to search, although this is not always the case. Because DHCP auto-configuration took on average just under 600 ms, the DHCP auto-configuration function (runDHCPClient()) used in the UE was programmed to try DHCP auto-configuration for a maximum of 800ms, after which it would exit and the DNS auto-configuration function (runDNSClient()) would attempt to gather network parameters. This approach would accommodate worst case DHCP auto-configuration time delay before attempting DNS fallback auto-configuration. The code for timing the DHCP and DNS fallback auto-configuration delay is shown in Figure 5.7.

```
long start = System.currentTimeMillis();
long elapsed;
runDHCPClient();
if(sdf_is_found){
    elapsed = System.currentTimeMillis()-start;
    System.out.println("DHCP: Time elapsed is: "+String.valueOf(elapsed));
}
else{
    runDNSClient();
    if(sdf_is_found){
        elapsed = System.currentTimeMillis()-start;
        System.out.println("DNS: Time elapsed is: "+String.valueOf(elapsed));
    }
    else{
        System.out.println("DHCP and DNS failure... \nLoading defaults..");
        loadDefaults();
    }
}
```

Figure 5.7: Code for timing DHCP and DNS fallback auto-configuration delays.

The DNS auto-configuration function itself was configured to attempt DNS auto-configuration for a maximum of 100ms because standalone DNS auto-configuration queries were taking approximately 50ms to complete. This would also accommodate worst case DNS auto-configuration time delay before resorting to static configurations. Therefore, the total delay on UE start-up before loading default configurations would be approximately 900ms if both DHCP and DNS auto-configuration failed. This is the worst case scenario but the author felt it is a justifiable trade-off considering the benefits of auto-configuration.

5.1.3.1 Software

For the DHCP and DNS auto-configuration experiments conducted, the DHCP function was provided by the Linux *dhcp-3* (version 3.1.1) server on an Ubuntu Karmic (9.04) virtual machine. The DNS function was provided by the Linux *BIND* (version 9.5.0) server on an Ubuntu Jaunty (8.10) virtual machine. The UE (JAIN SIP Applet Phone) was deployed on an Ubuntu Lucid (10.04) virtual machine. The host operating system was Ubuntu Lucid (10.04).

5.1.3.2 Hardware

The virtual machines set-up in the DHCP/DNS auto-configuration experiments described above were running on a physical machine running on an Intel(R) Core(TM) i7 CPU 870 @2.93GHz processor, with 3.9 GB RAM.

5.2 Chapter Summary

This chapter has presented evaluations and an analysis of the final architecture. Due to latency tests having already been conducted with the UCT MCF-MDF, we felt that experiments on session set-up delays would be un-necessary and would duplicate work already done by other researchers, hence the focus was rather on call flow evaluation and analysis. This chapter also presented findings on the impact of DHCP and DNS fallback auto-configuration on UE start-up. The justification for this being that lengthy UE start-up can impact negatively on the QoE.

The next and final chapter presents a set of conclusions concerning the work reported in this thesis. This chapter will also present recommendations for future work.

Chapter 6

Conclusions and Recommendations for Future Work

This thesis has presented and evaluated the COIN framework for the Mobicents Platform. This chapter will present a set of conclusions for the work presented and will also present recommendations for future work.

6.1 Conclusions

The COIN framework that has been presented considers the adoption of low-cost infrastructure by telcos and the use of advertising to provide competitively low-cost IPTV solutions. With telcos only recently entering the television services market, the proposed framework offers an IPTV service at low-cost in order to help spur IPTV adoption in the face of strong television service provider competition. Apart from offering users low-cost IPTV, the COIN framework addresses the growing demand from users for more control over multimedia applications by providing interfaces that allow them to customise their video and advertisements preferences. Customisation is regarded as a more viewer centered approach with regards to viewer focused implementations and is considered less invasive as compared to personalisation.

The following conclusions are drawn from this thesis.

6.1.1 Impact of proposed framework on key stakeholders

This thesis provided motivation for the proposed framework from the telco, user and advertiser perspectives. The proposed framework was found to provide a win-win-win scenario for these three stakeholders. The telco could potentially derive more revenue from the proposed framework, while users would have access to a viewer centered and affordable IPTV service. Advertisers stood to benefit by having a platform on which they can deploy targeted advertisements using viewer preference customisation.

6.1.2 Standards compliance and performance issues

In terms of implementation, this thesis did not focus on scalability issues concerning application deployment within the Mobicents Platform, rather the main focus was on delivering the proposed framework over standardised IPTV interfaces and components. Developing standardised IPTV functionality required careful examination of ongoing IPTV standardisation efforts and building the proposed framework following such specifications, in this case, we focused on the ETSI TISPAN IMS-based IPTV specifications. ETSI TISPAN IMS-based IPTV specifications are currently the most comprehensive set of specifications on IMS-based IPTV. The decision had already been made to build the proposed framework using IMS-based specifications to accommodate telcos with or without IMS support. However, scalability and performance remain critical issues for telco application servers that typically must operate under stringent telco requirements and be able to handle large numbers of requests per unit time. As was mentioned in the introductory chapter, SLEE is an already existing telco industry standard for event driven applications, and Mobicents is currently the only open SLEE 1.1 compliant application server. Application servers deployed within a SLEE are able to take advantage of the high throughput, low-latency environment that SLEE provides. Therefore the IPTV functions deployed within Mobicents JAIN SLEE will meet telco performance expectations.

The SDF and SCF were successfully developed and deployed within Mobicents JAIN SLEE to help provide a customisable, incentive driven IPTV framework (COIN). The SSF was deployed within the MPS to make use of the XDMS contained therein, rather than using the high performance JAIN SLEE server. The decision to implement the SSF within the MPS would allow users to customise their profiles using a standardised HTTP/XCAP interface and offer service selection functionality at a lower bandwidth cost. Although not fully implemented, this approach provided an alternative to the non-standard web interface used in the intermediate architecture. The author felt this was a reasonable trade-off since one of the key aims was to deliver a standards based IPTV solution. In addition, the author made use of the standards compliant UCT MCF-MDF to provide media capabilities.

6.1.3 The feasibility of integrating an advertising platform into an IPTV solution

In terms of the investigation into the use of advertising of third party products and services by telcos to generate additional revenue, it was found that telcos have generally been able to make above average returns when using advertisers to incentivise their service offerings. Investigations also found that the traditional broadcast television industry relies on advertisers to help pay for content, hence it will most likely prove unsustainable for telco IPTV providers to rely strictly on viewer subscriptions. Because of this, several telcos are investigating opt-in advertising plans to help fund IPTV content.

6.1.4 Impact of proposed framework on QoE

This thesis also conducted investigations into how the proposed framework will impact on the user's QoE. It was found that the Mobicents based IPTV implementation requires less signaling than the IMS-based IPTV implementation. Lower signaling generally translates to lower service latency, a key performance parameter for IPTV provision. The impact that the advertising approach could have on the QoE was also investigated. It was found that the advertising approach will not likely have any negative effects on the user's QoE. This is mainly because the advertisements are at the viewers discretion and the advertisements themselves are stored and accessed in the same way as the video content. The decision algorithm to grant billing discounts could potentially introduce some latency in the service delivery, but the algorithm is only triggered when the advertisement session ends, so there should be no noticeable difference between advertisement and video request in terms of service latency. Evaluations were concluded by looking at the impact that DHCP and DNS auto-configuration might have on UE start-up times. According to experimental findings, the worst case delay that could emanate from the auto-configuration mechanisms employed is approximately 900ms. It was the conclusion that this is an acceptable trade-off considering the benefits of auto-configuration.

6.1.5 Estimated cost of proposed IPTV solution

As mentioned earlier in the thesis, the deployment of an IPTV service requires a significant capital investment to cater for several IPTV service provider costs such as the acquisition of transcoding and distribution equipment and the payment of royalties for Conditional Access/Digital Rights Management (CA/DRM). Although some of these input costs can be easily ascertained, this is not so for some of the cost parameters. Among some of the larger input costs that are difficult to estimate is the cost of the actual IPTV content. The actual content that will be available to viewers will need to be acquired from content providers or typically from large television networks that own such material. These costs can only be determined through high level consultations with the owners of such material. It is therefore difficult at this stage to attach a figure to the overall cost of such a solution because we were in no realistic position to negotiate for content with any of the owners..

If we were however to make the reasonable assumption that we faced similar operational costs as other IPTV providers, then the cost of the proposed COIN framework would have the cost advantage of the advertiser driven billing incentive for lower cost.

6.2 Recommendations for Future Work

There are several extensions that can be done with this work. To begin with, there are several value added services that can be added to the IPTV solution to make it more attractive. Some value added IPTV services were mentioned in chapter 2 such as web browsing on television and video conferencing over the IPTV service. One value added service that was planned for the final solution was the integration of presence information into the IPTV service. However, time constraints did not permit this. Integrating presence data within the IPTV service could allow users to share what they are watching with approved contacts/buddies in real time. Users can also rate a movie/video they are viewing or one they have just seen and publish the rating as a status message, again in real time.

The customisation approach that was used was based on movie/video genre, the VoD customisation can be taken further to incorporate additional information such as the critical rating of the video. This could be a welcome feature for users who are particular about what they watch. Many people today first read how well critics rate a movie/video and will only watch a movie or show if it is well rated. Movies are typically rated on a scale of between 0-100% by critics. It is felt that a good extension of the customisation presented in this thesis would be to allow users to state that they want to watch shows rated at say at least 55 % by critics, or whatever they feel is good enough for them.

The customisation implemented here could also be extended for use with parental controls. This would require the use of content ratings (SNLV) on each video. It would also require setting up two types of profiles, guardian/parent profiles and child profiles. Guardians can set up their children's profiles and state what kind of material they want their children to watch. The children could then log onto the IPTV service and watch movies and shows, but have limited profiles in that they wouldn't be able to change settings on what the guardian has stipulated will be available to them.

Finally, it is felt that a commercially oriented framework like the one presented in this thesis needs to be tested with real users in a home setting, and have users give feedback on what they feel and think about the service. Only then can there be a more holistic evaluation of the proposed framework.

Appendix A

IMS IPTV Functional Entities, Reference Points and Protocols.

Figure A.1 shows the functional entities and reference points of the ETSI TISPAN IMS-based IPTV system. The full specification can be found at [42].

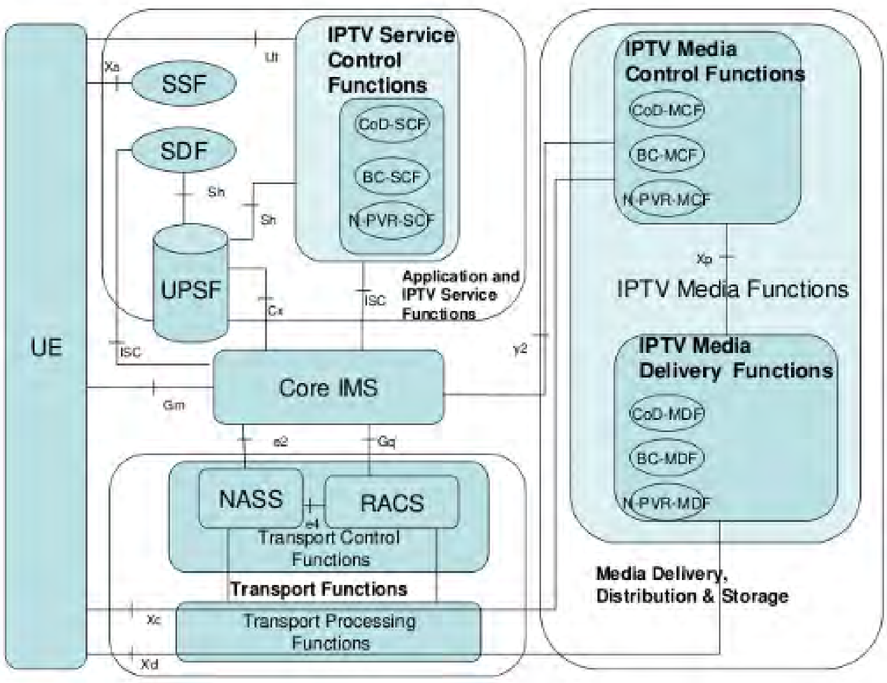

Figure A.1: ETSI TISPAN IMS-based IPTV system [42].

Table A.1 shows the names of the interfaces identified in figure A.1 and the signaling protocols associated with those interfaces. Ten interfaces are relevant to this thesis: Ut, Xa, Gm, Xc, Xd, ISC, Cx, Sh, y2 and Xp. For the full specification, refer to [42].

FE/ Reference point (protocol)	UE	IMS core	UPSF	SDF	SSF	SCF	MCF	MDF	ECF/ EFF
UE	—	Gm (SIP/SDP)	—	via Core IMS (SIP/SDP)	Xa (HTTP, DVBSTP, FLUTE)	Ut (HTTP), via Core IMS (SIP/SDP)	Xc (RTSP) (Note 1)	Xd (UDP/RT) (Note 1)	Dj, Di IGMP/ MLD
IMS core	Gm (SIP/SDP)	—	Cx (Diameter)	—	—	ISC (SIP/SDP)	y2 (SIP/SDP)	—	—
UPSF	—	Cx (Diameter)	—	Sh (Diameter)	—	Sh (Diameter)	—	—	—
SDF	via Core IMS (SIP/SDP)	—	Sh (Diameter)	—	—	—	—	—	—
SSF	Xa (HTTP, DVBSTP, FLUTE)	—	—	—	—	—	—	—	—
SCF	Ut (HTTP), via Core IMS (SIP/SDP)	ISC (SIP/SDP)	Sh (Diameter)	—	—	—	via Core IMS and y2 (SIP/SDP)	—	—
MCF	Xc (RTSP) (Note 1)	y2 (SIP/SDP)	—	—	—	via Core IMS and y2 (SIP/SDP)	—	Xp (not defined)	—
MDF	Xd (UDP/RTP) (Note 1)	—	—	—	—	—	Xp (not defined)	—	—
ECF/ EFF	—	—	—	—	—	—	—	—	—

NOTE 1: As described in TS 182 027 [2], clauses 6.4 and 6.5, Xc and Xd are logical reference points that can be decomposed into Dj and possibly Di, Ds or Iz reference points depending on the location of the MCF or MDF.
NOTE 2: Annex H lists compliance requirements for the protocols listed in table 4.1.

Table A.1: Interfaces and protocols for communication between IMS IPTV entities [42].

Ut (UE - IPTV Service Control Functions) is used by the UE for configuration of the user's IPTV profile using the HTTP protocol.

Xa (UE - Service Selection Function) is used by the UE for the selection of IPTV services using the HTTP protocol.

Gm (UE - Core IMS) is used by the UE to access IPTV services using the SIP protocol.

Xc (UE - IPTV Media Control Functions) is used by the UE for the exchange of media control messages to control IPTV media flows using the RTSP protocol.

Xd (UE - IPTV Media Delivery Functions) is used by the UE for the delivery of media content using the RTP protocol.

ISC (Core IMS - Service Discovery Function) is used to provide service attachment in- formation and for discovering personalised IPTV services using the SIP protocol.

Cx (Core IMS - UPSF) is used to obtain user profile information during session setup and modification using the Diameter protocol.

Sh (UPSF - IPTV Service Control Functions) is used for authorisation during session setup and modification using the Diameter protocol.

y2 (Core IMS - IPTV Media Control Functions) is used by the IPTV AS to indirectly control the MCF through the S-CSCF using the SIP protocol.

Xp (IPTV Media Control Functions - IPTV Media Delivery Functions) is used by the MCF to control the MDF using the RTSP protocol.

Appendix B

Screen-shots

Figure B.1 shows a screen-shot of the web page used to customise movie/video preferences.

Figure B.2 shows the Ericsson PPS emulation, the UE and the Mobicents console in the background.

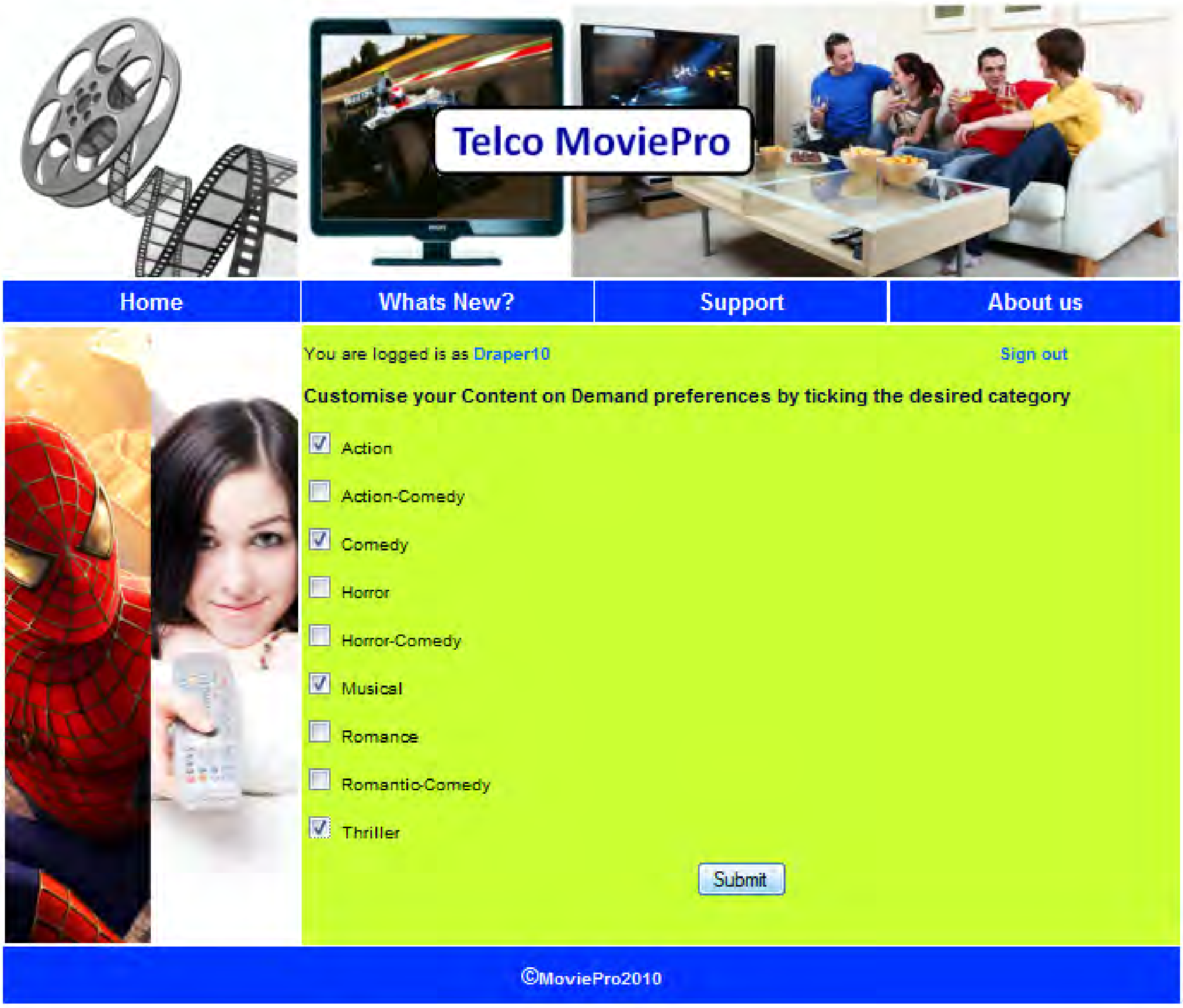

Figure B.1: Web interface for video customisation

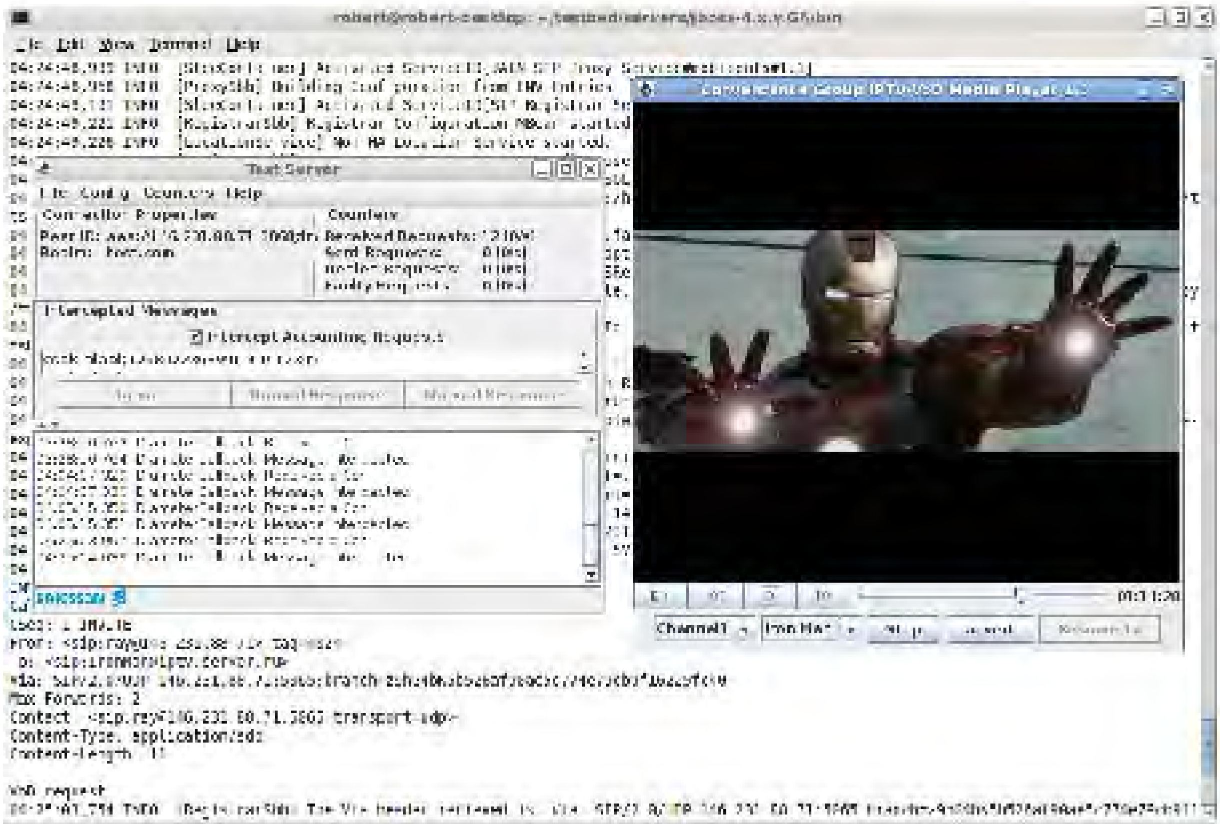

Figure B.2: Screen-shot of the Ericsson PPS server, the UE and the Mobicents console in the background.

Appendix C

Configuration files

Figure C.1 shows the POM file used to deploy the SCF in the final architecture.

```
<project xmlns="http://maven.apache.org/POM/4.0.0"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 http://maven.apache.org/maven-
v4_0_0.xsd">

        <modelVersion>4.0.0</modelVersion>

        <parent>
                <artifactId>sip-services-parent</artifactId>
                <groupId>org.mobicents.examples</groupId>
                <version>1.2.6.GA</version>
        </parent>

        <artifactId>sip-services-scf-sbb</artifactId>

        <dependencies>
                <dependency>
                        <groupId>org.mobicents.examples</groupId>
                        <artifactId>sip-services-location-sbb</artifactId>
                        <version>${pom.version}</version>
                </dependency>
                <dependency>
                        <groupId>comp.BillingComponent</groupId>
                        <artifactId>BillComp</artifactId>
                        <version>10.2.0</version>
                        <scope>compile</scope>
                        </dependency>
        </dependencies>

</project>
```

Figure C.1: POM file used to deploy the SCF.

Figure C.2 shows the configuration settings used to facilitate service discovery using the DHCP server. These were entered into the dhcpd.conf file, which is the configuration file for the dhcp daemon.

```
# option definitions common to all supported networks.
option domain-name "convergence.ru";
option domain-name-servers 10.0.2.1;
option subnet-mask 255.255.255.0;
option broadcast-address 10.0.2.255;
option sip-server-address code 120 = array of ip-addre
option service-discovery-function code 222 = array of i

subnet 10.0.2.0 netmask 255.255.255.0 {
range 10.0.2.1 10.0.2.10;
option sip-server-address 10.0.2.3;
option service-discovery-function 10.0.2.1;
}

default-lease-time 600;
max-lease-time 7200;
```

Figure C.2: DHCP auto-configuration settings.

Figure C.3 shows the zone file used to configure the DNS server for service discovery.

```
convergence.ru. IN SOA ns1.convergence.ru. admin.convergence.ru. (
2007031001
28800
3600
604800
38400
)
convergence.ru. IN NS ns1.convergence.ru.
convergence.ru. IN MX 10 mail.convergence.ru.

_sip._udp IN SRV 0 1 5060 provider.convergence.ru.

ns1 A 10.0.2.1
iptv A 10.0.2.1
provider A 10.0.2.4
```

Figure C.3: DNS SRV record configuration for IPTV entry point.

Figure C.4 shows the XML schema used to describe videos in the Mobicents Presence Server.

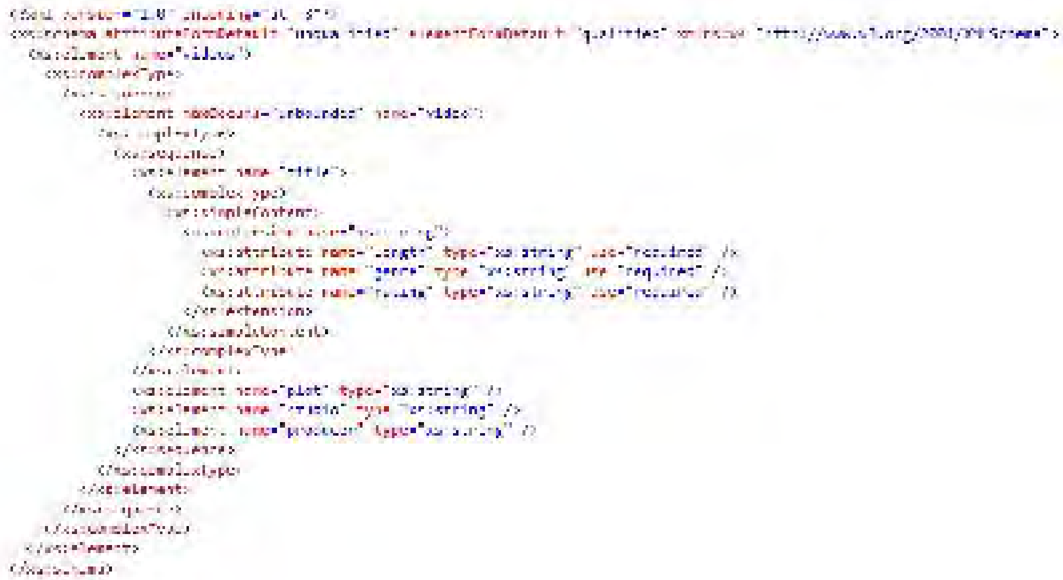

Figure C.4: XML schema used to describe videos in the MPS

Appendix D

Publications resulting from this research effort

1. R. Musvibe and A. Terzoli. Developing a Video on Demand Service using Mobicents JSLEE. In *Southern Africa Telecommunication Networks and Applications Conference (SATNAC)*, 2009.

2. R. Musvibe and A. Terzoli. Towards a Customised Video on Demand Service for the Mobicents Platform. In *Southern Africa Telecommunication Networks and Applications Conference (SATNAC)*, 2010.

3. M. Tsietsi, R. Musvibe, A. Terzoli, and G. Wells. Towards IPTV Service Discovery and Selection in an IMS environment. In *Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2010 International Congress on*, pages 195201. IEEE.

References

[1] 3GPP. TS 23.218, IP Multimedia (IM) session handling; IM call model; Stage 2. 2006.

[2] 3GPP. TS 23.228 V7.7.0, IP Multimedia Subsystem (IMS); Stage 2. 2007.

[3] On-Demand Television Advertising. [online] available: http://www.koeppelinteractive.com/on_demand_advertising.html [december 2010].

[4] S. Alexander and R. Droms. RFC 2132: DHCP Options and BOOTP Vendor Extensions. 1997.

[5] Apache Ant. [online] available: http://ant.apache.org [december 2010].

[6] The Evolving IPTV Service Architecture. [online] available: http://www.cisco.com/en/us/solutions/collateral/ns341/ns525/ns537/ns549/ns746/net_implementation_white_paper0900aecd806530a4.html [decemder 2010].

[7] J. Arkko, F. Lindholm, M. Naslund, K. Norrman, Erricsson, and E. Carrara. RFC 4567: Key management extensions for Session Description Protocol (SDP) and Real Time Streaming Protocol (RTSP). 2006.

[8] J.P. Artero. Online video business models: Youtube vs. hulu. *Palabra Clave*, 13(1):111–123, 2010.

[9] S. J. Berman, L. B. Battino, Shipnuck, and A. Neus. The end of advertising as we know it. *IBM Global Business Services, 2007.*

[10] T. Berners-Lee, R. Fielding, and H. Frystyk. RFC 1945: Hypertext Transfer Protocol HTTP/1.0, May 1996.

[11] P. Boncz, S. Manegold, and M. Kersten. Database architecture optimized for the new bottleneck: Memory access. In *Proceedings of the International Conference on Very Large Databases*, pages 54–65. Citeseer, 1999.

[12] D. Boswarthick. ETSI IPTV Standards Visible Benefits for your Business. *IPTV World Forum Asia, 4-5 December*, 2010.

[13] E. Burger. RFC 4483: A Mechanism for Content Indirection in Session Initiation Protocol (SIP) Messages. 2006.

[14] Global Telecoms Business. Beating over-the-top players at their own game [online] available: http://www.globaltelecomsbusiness.com/article/2746158/sectors/25198/beating-over-the-top-players-at-their-own-game.html [december 2010].

[15] P. Calhoun, J. Loughney, E. Guttman, and G. Zorn. RFC 3588: Diameter Base Protocol. 2003.

[16] G. Camarillo and M.A. Garcia-Martin. *The 3G IP multimedia subsystem (IMS): Merging the Internet and the cellular worlds.* Wiley, 2006.

[17] M. Cedervall, U. Horn, Y. Hu, I.M. Ivars, and T. Nasstrom. Open IPTV Forum–Toward an open IPTV standard. *Ericsson Review*, 3, 2007.

[18] C.V. Chakravarthy. IP Multimedia Subsystems: A Tutorial, 2006. [on- line] available http://images.tmcnet.com/expo/sd06/presentations/c-chakravarthy- ims-tutorial.pdf [december 2010].

[19] C.C. Chang, Y.F. Lu, A.C. Pang, and T.W. Kuo. Design and implementation of SIP security. In *Information Networking*, pages 669–678. Springer, 2005.

[20] P. Christian. IPTV advertising measurements - making more money from advertising [online] available: http://www.iptvmagazine.com/2007_01/iptv magazine_2007_01_iptv_advertising. html. ip television magazine, [december 2010].

[21] UCT IMS Client. [online] available: http://uctimsclient.berlios.de/ [december 2010].

[22] Sip Communicator. [online] available: http://sip-communicator.org/ [december 2010].

[23] Internet Systems Consotium. Bind [online] available: http://www.isc.org/software/bind [december 2010].

[24] Internet Systems Consotium. DHCP [online] available: http://www.isc.org/software/dhcp [december 2010].

[25] Nokia Corporation. Servce Delivery Platforms. page 12, 2005. [online] available http://www.nokia.com/NOKIA_COM_1/About_Nokia/Press/White_Papers/pdf_files/whitepaper_service_delivery.pdf [december 2010].

[26] Y. Cosmadopoulos and M. Kulkarni. JSR 289: SIP Servlet v1. 1, 2008. [online] available: http://download.oracle.com/otndocs/jcp/sip_servlet-1.1-fr-eval-oth-JSpec/ [december 2010].

[27] A. Cuevas, J.I. Moreno, P. Vidales, and H. Einsiedler. The IMS service platform: a solution for next-generation network operators to be more than bit pipes. *Communications Magazine, IEEE*, 44(8):75–81, 2006.

[28] R. Van Den Dam. IPTV Advertising: a gold mine for telcos? [online] available http://broadcastengineering.com/rf/broadcasting_iptv_advertising_gold/ [december 2010].

[29] JVLC Dead? [online] available: http://forum.videolan.org/viewtopic.php?f=2&t=50688 [december 2010].

[30] J. Deruelle. Mobicents Communications Platform. Java EE, JAIN-SLEE, SIP Servlets, Media Server. *JavaOne*, 2008.

[31] S. Diaz. Facebook's privacy settings: users get more control as wall gets torn down [online] available: http://www.zdnet.com/blog/btl/facebooks-privacy-settings-users-get-more-control-as-wall-gets-torn-down/20611 [december 2010].

[32] V. Dikgole and N. Ventura. Video on Demand Service for Next Generation Networks. *SATNAC Proceedings*, 2008.

[33] DNS-Based Service Discovery. [online]. available : files.dns-sd.org/draft-cheshire-dnsext-dns-sd.tx [december 2010].

[34] D. Dumur, P. Boucher, and J. Roder. Advantages of an open architecture structure for the design of predictive controllers for motor drives. *CIRP Annals-Manufacturing Technology*, 47(1):291–294, 1998.

[35] DVB. What is DVB? - DVB homepage [online] available: http://www.dvb.org/ [december 2010].

[36] DVB-IP. DVB-IP Phase 1.3 in the context of ETSI TISPAN NGN. DVB Document A128, September 2008. [online] available: http://www.dvb.org/technology/standards/a128.ipi2480r9.DVB-IP1.3_in_ETSI_TISPAN_NGN.pdf [december 2010].

[37] A. Mendonca et al. Mobicents JAIN SLEE User Guide [online] available: http://hudson.jboss.org/hudson/view/mobicents/job/mobicentsbooks/lastsuccessfulbuild/artifact/jain-slee/core/index.html[hash]management_consoles [december 2010].

[38] A. Neeraj et al. Putting one–to-one marketing to work: Personalization, customization, and choice. *Marketing letters journal 2008, 19, 305-32. Springer Netherlands.*

[39] ES ETSI. 282 010: Telecommunications and Internet Converged Services and Protocols for Advanced Networking (TISPAN).

[40] TS ETSI. 102 034: Digital Video Broadcasting (DVB); Transport of MPEG-2 TS Based DVB Services over IP Based Networks .

[41] TS ETSI. 181 016:Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN); Service Layer Requirements to integrate NGN services and IPTV.

[42] TS ETSI. 183 063:Telecommunications and Internet converged Services and Protocols for Advanced Networking (TISPAN);IMS based IPTV Stage 3.

[43] S. M. Faccin, P. Lalwaney, and B. Patil. IP multimedia services: analysis of mobile IP and SIP interactions in 3G networks. *Communications Magazine, IEEE*, 42(1):113–120, 2005.

[44] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and T. Berners-Lee. RFC2616: Hypertext Transfer Protocol–HTTP/1.1. *RFC Editor United States*, 1999.

[45] J.F. Fleury. IPTV: The need for standards. *Communications Technology, Vol. 22, No. 11, November 2005, pp., 30-37*, 2005.

[46] Apache Software Foundation. HttpClient overview [online] available: http://hc.apache.org/httpcomponents-client-ga/ [december 2010].

[47] J. Frey. The Semiotic Web: Dont forget the user amongst all the semantics. 2008. [online] avaliable: http://journal.webscience.org/38/ [december 2010].

[48] O. Friedrich and S. Seeliger, R.and Arbanowski. Interactive and Personalized Services for an Open IMS-Based IPTV Infrastructure. In *Proceedings of the Seventh International Conference on Networking*, pages 302–307, Washington, DC, USA, 2008. IEEE Computer Society.

[49] D. Geneiatakis and C. Lambrinoudakis. A Cost-Effective Mechanism for Protecting SIP Based Internet Telephony Services Against Signaling Attacks. In *Proceedings of the IMS and Mobile Multimedia Workshop*, 2008.

[50] D. Geneiatakis, C. Lambrinoudakis, G. Kambourakis, A. Kafkalas, and S. Ehlert. A first order logic security verification model for SIP. *Communications, 2009. ICC'09. IEEE International Conference on*, pages 1–6, 2009.

[51] T.J. Gerpott and N.M. Jakopin. The degree of internationalization and the financial performance of European mobile network operators. *Telecommunications Policy*, 29(8):635–661, 2005.

[52] R. Good and N. Ventura. An evaluation of transport layer policy control in the IP multimedia subsystem. In *Personal, Indoor and Mobile Radio Communications, 2008. PIMRC 2008. IEEE 19th International Symposium on*, pages 1–5. IEEE, 2008.

[53] K. Grochla, W. Buga, J. DzierSega, P. Pacyna, and A. Seman. Autoconfiguration procedures for multiradio wireless mesh networks based on DHCP protocol. In *World of Wireless, Mobile and Multimedia Networks & Workshops, 2009. WoWMoM 2009. IEEE International Symposium on a*, pages 1–6. IEEE, 2009.

[54] Media Reserach Group. IPTV Global Forecast 2010 to 2014 [online] available: http://www.mrgco.com/iptv/gf0610.html [december 2010].

[55] A. Gulbrandsen, P. Vixie, and L. Esibov. RFC 2782: A DNS RR for specifying the location of services (DNS SRV).

[56] M. Handley and V. Jacobson. RFC 2327: SDP: Session Description Protocol. *Internet Engineering Task Force April*, 1998.

[57] M. Handley, V. Jacobson, and C. Perkins. RFC 4566: SDP: Session Description Protocol. 2006.

[58] L. Harte. *IPTV Business Opportunities, How to Make Money in the Emerging IP Television Industry.* Althos Books, 2008.

[59] S. Higginbotham. Germans get cheaper cell phone service in exchange for ads [online]. available: http://gigaom.com/2009/06/29/germans-get-cheaper-cell-phone-service-in-exchange-for-ads/. [december 2010].

[60] VLC home page. [online] available: http://www.videolan.org/vlc/ [december 2010].

[61] IPTV and IMS in Next-generation Networks. [online] available: www.tmcnet.com/redir/?u=1004216 [december 2010].

[62] I. Ivanov. Mobicents: JSLEE for the People, by the People [online] available: http://today.java.net/pub/a/today/2006/03/09/mobicents-jslee.html.

[63] I. Ivanov. Mobicents Open Source VoIP Platform [online] available: http://mobicents.googlecode.com/svn/tags/b1.2.02-libraries_remove/core/docs/presentations/javapolis2005/javapolis-mobicents.ppt [december 2010].

[64] I. Ivanov. Talking Mobicents with Ivelin Ivanov [online] available: http://it.toolbox.com/blogs/opensource-unleashed/talking-mobicents-with- ivelin-ivanov-11687 [december 2010].

[65] JapanToday. Google gives Gmail users more control over inboxes [online] available: http://www.japantoday.com/category/technology/view/google-gives-gmail-users-more-control-over-inboxes [december 2010].

[66] K. Jarvi, L.M. Sainio, P. Ritala, and A. Pellinen. Partnership Business Model : Case IP Multimedia Communication Services. In *EUROMOT Conference, September 17th–19th*, 2008.

[67] Gstreamer Java. [online] available: http://code.google.com/p/gstreamer-java/ [december 2010].

[68] jDHCP. [online] available: http://www.dhcp.org/javadhcp/news.html [december 2010].

[69] Class JVLC. [online] available: http://javaplayer.sourceforge.net/doc/vlc/jvlc.html [december 2010].

[70] R. Kalakota and M. Robinson. *e-Business*. Addison-Wesley Reading, MA, 1999.

[71] K. Knightson. What is NGN: Basic Architecture. *ITU-T workshop on next generation networks and grids, Geneva*, 2006.

[72] K. Knightson, N. Morita, and T. Towle. NGN architecture: generic principles, functional architecture, and implementation. *Communications Magazine, IEEE*, 43(10):49–56, 2005.

[73] OnSIP Knowledgebase. What is an SRV Record? [online] available: http://www.junctionnetworks.com/knowledgebase/onsip/sip-and-xmpp-hosting/what-is-srv-record [december 2010].

[74] A. Kurokawa and I. Higashi. Standardization Trends of the Next Generation Network in ETSI TISPAN. *NTT Technical Review*, 4(6), 2006.

[75] P. Leach, C. Newman, and A. Melnikov. RFC 2831: Using Digest Authentication as a SASL Mechanism. 2000.

[76] C.S. Lee. IPTV over next generation networks in ITU-T. *InternationalWorkshop on Broadband Convergence Networks, BcN'07. 2nd IEEE/IFIP*, pages 1–18, 2007.

[77] Z. Lei and X. Zhanhong. Mathematical and Empirical Analysis of Mobile ARPU Decline. *China Communications*, page 29, 2006.

[78] W. Li. A service oriented SIP Infrastructure for Adaptive and Context-Aware wireless Services. In *Proceedings of the 2nd International Conference on Mobile and Ubiquitous Multimedia, Norrkoping, Sweden*. Citeseer, 2003.

[79] S. B. Lim and D. Ferry. JAIN SLEE 1.0 Specification, final release. *Sun Microsystems, Inc. and Open Cloud Limited March*, 2004. [online] available: http://phoenix.labri.fr/documentation/sip/Documentation/Papers/ Programming_SIP/ Paper_Publication_and_Draft/Java_SIP/jain_slee-1_0-fr-spec.pdf [december 2010].

[80] S.B. Lim and D. Ferry. Jain SLEE 1.0 Specification. *Sun Microsystems Inc. & Open Cloud Ltd*, 2002. [online] available: http://download.oracle.com/otndocs/jcp/7781-jain_slee-1.0-pfd-spec-oth-JSpec/ [december 2010].

[81] Open Cloud Limited. A SLEE for all Seasons [online] available:http://www.opencloud.com/documents/whitepaperll

[82] C. Liu. Multimedia Over IP: RSVP, RTP, RTCP, RTSP. *Handbook of Communication Technologies: The Florida*, 2000.

[83] K. Liu, A. Gil, and LA Berendt. IPTV: The Entertainment Game Has Changed [Guest Editorial]. *Communications Magazine, IEEE*, 46(8):88–89, 2008.

[84] D.T. Long and D.C. Page. Supporting portable applications with JAIN SLEE. In *Telecommunication Networks and Applications Conference, 2007. ATNAC 2007. Australasian*, pages 454–459. IEEE, 2008.

[85] C. Longbottom. IPTV - the new age of web ads [online] available: http://www.it-director.com/content.php?cid=8474 [december 2010].

[86] S.H. Maes. Pragmatic Approaches to True Convergence with or without IMS. In *Network Operations and Management Symposium Workshops, 2008. NOMS Workshops 2008. IEEE*, pages 44–51. IEEE, 2008.

[87] T. Magedanz. Evolution of telco service platforms in the Face of Network Convergence and SOA Principles-A Deja Vu. *Fraunhofer Fokus*, 2007.

[88] T. Magedanz, P. Weik, D. Vingarzan, F.C. de Gouveia, S. Wahle, et al. Experiences on the Establishment and Provisioning of NGN/IMS Testbeds-The FOKUS Open IMS Playground and the Related Open Source IMS Core. In *11th International Conference on Intelligence in service delivery Networks (ICIN), Emerging Web and Telecom Services: Collision or Coopetition*, pages 8–11.

[89] N. Mahendra. IPTV Deployment Issues, Opportunities, and Challenges in India. *BROADCAST & CABLESAT*, 2008.

[90] J. Maisonneuve, M. Deschanel, J. Heiles, W. Li, H. Liu, R. Sharpe, and Y. Wu. An overview of IPTV standards development. *Broadcasting, IEEE Transactions on*, 55(2):315–328, 2009.

[91] R. J. Marston. Multimedia Content Adaptation for Internet Protocol Television Services in the IP Multimedia Subsystem. Master's thesis, Department of Electrical Engineering, UNIVERSITY OF CAPE TOWN, 2008.

[92] O. Martikainen. Internet Revolution in Telecom. In *Modern Computing, 2006. JVA'06. IEEE John Vincent Atanasoff 2006 International Symposium on*, pages 58–62. IEEE, 2006.

[93] E. Martins, D. Silas, and J. Morgan. Mobicents SIP Presence Service User Guide [online] available: http://hudson.jboss.org/hudson/job/mobicentsbooks/lastsuccessfulbuild/artifact/sip-presence/index.html [december 2010].

[94] E. Martinsson. IPTV the future of television? [online] available http://www.cs.chalmers.se/ tsigas/courses/dcdseminar/files/iptvrapport.pdf [december 2010].

[95] I. Mas, V. Berggren, R. Jana, J. Murray, and CW Rice. IMS-TV: an IMS-based architecture for interactive, personalized IPTV. *Communications Magazine, IEEE*, 46(11):156–163, 2008.

[96] Apache Maven. [online] available: http://maven.apache.org/ [december 2010].

[97] Maxmind GeoIP. [online] available: http://www.maxmind.com [december 2010].

[98] T. Melanchuk. An Architectural Framework for Media Server Control. *draft-ietf-mediactrl-architecture-00 (work in progress)*, 2007. [online] available: http://tools.ietf.org/html/draft-ietf-mediactrl-architecture-00 [december 2010].

[99] E. Mikoczy. Next generation of multimedia services-NGN based IPTV architecture. *15th International Conference on Systems, Signals and Image Processing*, pages 523–526, 2008.

[100] E. Mikoczy, D. Sivchenko, B. Xu, and V. Rakocevic. IMS based IPTV services: architecture and implementation. *Proceedings of the 3rd international conference on Mobile multimedia communications*, pages 1–7, 2007.

[101] Mobicents. Mobicents - homepage [online] available: http://www.mobicents.org/products.html [december 2010].

[102] Mobicents. Mobicents FAQ's [online] available: www.mobicents.org-a.googlepages.com/faq.html [december 2010].

[103] K.S. Munasinghe and A. Jamalipour. Interworked WiMAX-3G cellular data networks: an architecture for mobility management and performance evaluation. *Wireless Communications, IEEE Transactions on*, 8(4):1847–1853, 2009.

[104] W.T. Muswera. Development of an IMS Compliant, Cross platform Client Using the JAIN SIP Applet Phone. In *SATNAC 2010*, 2010.

[105] MySQL database - home page. [online] available: http://www.mysql.com/ [december 2010].

[106] A. Niemi, J. Arkko, and V. Torvinen. RFC 3310: Hypertext transfer protocol (HTTP) digest authentication using authentication and key agreement (AKA) . 2002.

[107] P.F. Nunes and A. Kambil. Personalization? No Thanks. *Harvard Business Review*, 79(4):32–34, 2001.

[108] G. Obiltschnig. Automatic Configuration and Service Discovery for Networked Smart Devices. In *Electronica Embedded Conference Munich*, 2006.

[109] S. Oestmann. Mobile Operators: their contribution to universal service and public access. *Intelecon Research. Retrieved July*, 2005.

[110] OIPF. Open IPTV Forum - homepage [online] available: www.openiptvforum.org [december 2010].

[111] OIPF. Open IPTV Forum Overview Presentation. *Open IPTV Forum*, page 5, 2010. [online] available: http://www.openiptvforum.org/docs/OIPF_Overview_Presentation_Sep_2010.pdf [december 2010].

[112] Advanced Optical and Digital Architectures for Video-On-Demand. 2003. [online] available: http://www.harmonicinc.com/view_collateral.cfm?ID=40 [december 2010].

[113] V.G. Ozianyi and N. Ventura. Charging in IP Multimedia Networks. *SATNAC, 2010.*

[114] JAIN SIP Applet Phone. [online] available: https://jain-sip-applet-phone.dev.java.net/ [december 2010].

[115] M. Poikselka and G. Mayer. *The IMS: IP Multimedia Concepts and Services*. Wiley, 2009.

[116] D. Ramirez. *IPTV security: protecting high-value digital contents*. Wiley, 2008.

[117] Red5 Streaming Server. [online] available: http://www.red5.org [december 2010].

[118] Redhat. JBOSS COMMUNICATIONS PLATFORM (JBCP) [online] available: http://www.redhat.com/solutions/telco/communications_platform/ [december 2010].

[119] I.E.G. Richardson. *H. 264 and MPEG-4 video compression: video coding for next-generation multimedia*. John Wiley & Sons Inc, 2003.

[120] J. Rosenberg. XCAP Tutorial [online] available: jdrosen.net/papers/xcap-tutorial.ppt.

[121] J. Rosenberg et al. RFC 4825: The Extensible Markup Language (XML) Configuration Access Protocol (XCAP), 2007.

[122] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler. RFC 3261: SIP: Session Initiation Protocol. *Internet Engineering Task Force*, 2002.

[123] S. Sahai. Interactive television advertising [online]. available: http://www.iptv magazine.com/2009_02/iptv_magazine_2009[und erscore]02_interactive_tv_advertising.html. ip television magazine, [december 2010].

[124] M. Schmidt. IMS based IPTV: ETSI TISPAN standards applied. *Proceedings of the 5th International ICST Mobile Multimedia Communications Conference*, pages 1–4, 2009.

[125] H. Schulzrinne. RFC 3361: Dynamic Host Configuration Protocol (DHCP-for-IPv4) Option for Session Initiation Protocol (SIP) Servers . 2002.

[126] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson. RFC 3550: RTP, A transport protocol for real-time applications. *Internet Engineering Task Force, Network Working Group*, 2003.

[127] H. Schulzrinne, A. Rao, and R. Lanphier. RFC 2326: Real Time Streaming Protocol (RTSP). 1998.

[128] Darwin Streaming Server. [online] available: dds.macosforge.org/post/40/ [december 2010].

[129] IPTV service delivery architecture over IMS and possible applications. [online] available: http://whitelassiblog.wordpress.com [december 2010].

[130] Z. Shibeshi, A. Terzoli, and K. Bradshaw. Using an RTSP proxy to implement the IPTV Media Function via a streaming server. In *Ultra Modern Telecommunications and Control Systems and Workshops (ICUMT), 2010 International Congress on*, pages 253–259. IEEE.

[131] R.G. Shiroor. IPTV and VoD services in the context of IMS. In *IP Multimedia Subsystem Architecture and Applications, 2007 International Conference on*, pages 1–5. IEEE, 2008.

[132] W. Simpson. *Video over IP: a practical guide to technology and applications.* Focal Pr, 2006.

[133] D. Siradovic. Telekom predstavio Open IPTV [online] available: http://ns-linux.org/vesti/iptv-i-kod-nas/ [december 2010].

[134] S. Sivasothy, G.M. Lee, and N. Crespi. A unified session control protocol for IPTV services. *11th International Conference on Advanced Communication Technology*, 2:961–965, 2009.

[135] D. Sivchenko, B. Xu, and D. Telekom. IPTV Services over IMS: Architecture and Standardization. *IEEE Communications Magazine*, page 129, 2008.

[136] J. Song, M.Y. Chang, S.S. Lee, and J. Joung. Overview of ITU-T NGN/IMS QoS control. *Communications Magazine, IEEE*, 45(9):116–123, 2007.

[137] R. Spiers, R. Marston, R. Good, and N. Ventura. The UCT IMS IPTV Initiative. *Third International Conference on Next Generation Mobile Applications, Services and Technologies*, pages 503–508, 2009.

[138] R. Spiers and N. Ventura. A Converged IMS Client for the IP Multimedia Subsystem. In *Southern Africa Telecommunication Networks and Applications Conference (SATNAC)*, 2010.

[139] A.J. Stienstra. Technologies for DVB services on the Internet. *Proceedings of the IEEE*, 94(1):228–236, 2005.

[140] S. Szekely, T. Szasz, Z. Szappanyos, and Z. Tofalvi. Challenges in a Web-enhanced Personalised IPTV Service. *Future*, 4:5.

[141] Dont Forget the User. [online] available: http://www.pse.be/files/cms1 /epicor_knowledgementor-br.pdf [december 2010].

[142] ETSI TISPAN. Overview of ETSI TISPAN IPTV [online] available: http://docbox.etsi.org/tispan/open/information/ngn_presentations/tispanork

[143] S. Tomac, M. Sikirica, L. Skorin-Kapov, and M. Matijašević. Implementation of the diameter-based cx interface in the ip multimedia subsystem. *In Proceedings of 29th International Convention MIPRO 2006, volume 2, pages 22-26*, 2:22–26.

[144] M. Tsietsi, R. Musvibe, A. Terzoli, and G. Wells. Towards IPTV service discovery and selection in an IMS environment. In *Ultra Modern Telecommunications and*

Control Systems and Workshops (ICUMT), 2010 International Congress on, pages 195–201. IEEE.

[145] TVOVER.NET. Telco/IPTV Subscribers to Total Almost 40 Million Households Worldwide in 2009 [online] available: http://www.tvover.net/ [december 2010].

[146] M. Volk, J. Guna, A. Kos, and J. Bester. IPTV systems, standards and architectures: Part II-quality-assured provisioning of IPTV services within the NGN environment. *Communications Magazine, IEEE*, 46(5):118–126, 2008.

[147] T. Walsh. Next generation advertising: a quantified shift to the network edge. *Openet Transactional Intelligence*, 2008.

[148] F. Weber, W. Fuhrmann, U. Trick, U. Bleimann, and B. Ghita. QoS in SIP-based NGN–state of the art and new requirements. In *SEIN 2007: Proceedings of the Third Collaborative Research Symposium on Security, E-Learning, Internet and Networking*, page 201. Lulu. com.

[149] B. Wellington. dnsjava home page [online] available: http://www.dnsjava.org/ [december 2010].

[150] S. Whitehead, MJ Montpetit, and X. Marjou. An Evaluation of Session Initiation Protocol (SIP) for use in Streaming Media Applications. *Work in progress, Internet Engineering Task Force, February*, 2006.

[151] P.R. Wilson. Delivering Personalised Advertising in IMS-Based IPTV through Direct Marketing. Master's thesis, Department of Electrical Engineering University of Cape Town, 2010.

[152] P.R. Wilson and N. Ventura. A Direct Marketing Platform for IMS-Based IPTV. *Southern African Networks and Applications Conference (SATNAC 2009)*, 2009.

[153] T.V. Wilson. Netflix Recommendations [online] available: http://electronics.howstuffworks.com/netflix2.htm [december 2010].

[154] WITBE. Know your Set-top box, know your customer satisfaction [online] available: www.csimagazine.com/csi/whitepapers/witbe-set-top.pdf [december 2010]. White Paper.

[155] Robert Wojcik. Next Generation Networks Architecture by ITU-T. *AGH University of Science and Technology, Department of Telecommunications*, January 2009.

[156] Y. Xiao, X. Du, J. Zhang, F. Hu, and S. Guizani. Internet Protocol Television (IPTV): the killer application for the next-generation Internet. *Communications Magazine, IEEE*, 45(11):126–134, 2007.

[157] F. Xu, L. Zhang, and Z. Zhou. Interworking of Wimax and 3GPP networks based on IMS [IP Multimedia Systems (IMS) Infrastructure and Services]. *Communications Magazine, IEEE*, 45(3):144–150, 2007.