Linking Session Based Services with Transport Plane Resources in IP Multimedia Subsystems

Richard Good

Supervisor:

Neco Ventura

Thesis Presented for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Electrical Engineering

UNIVERSITY OF CAPE TOWN

May 2009

Synopsis

The massive success and proliferation of Internet technologies has forced network operators to recognise the benefits of an IP-based communications framework. The IP Multimedia Subsystem (IMS) has been proposed as a candidate technology to provide a non-disruptive strategy in the move to all-IP and to facilitate the true convergence of data and real-time multimedia services.

Despite the obvious advantages of creating a controlled environment for deploying IP services, and hence increasing the value of the telco bundle, there are several challenges that face IMS deployment. The most critical is that posed by the widespread proliferation of Web 2.0 services. This environment is not seen as robust enough to be used by network operators for revenue generating services. However IMS operators will need to justify charging for services that are typically available free of charge in the Internet space. Reliability and guaranteed transport of multimedia services by the efficient management of resources will be critical to differentiate IMS services.

This thesis investigates resource management within the IMS framework. The standardisation of NGN/IMS resource management frameworks has been fragmented, resulting in weak functional and interface specifications. To facilitate more coherent, focused research and address interoperability concerns that could hamper deployment, a Common Policy and Charging Control (PCC) architecture is presented that defines a set of generic terms and functional elements.

A review of related literature and standardisation reveals severe shortcomings regarding vertical and horizontal coordination of resources in the IMS framework. The deployment of new services should not require QoS standardisation or network upgrade, though in the current architecture advanced multimedia services are not catered for. It has been found that end-to-end QoS mechanisms in the Common PCC framework are elementary.

To address these challenges and assist network operators when formulating their

NGN strategies, this thesis proposes an application driven policy control architecture that incorporates end-user and service requirements into the QoS negotiation procedure. This architecture facilitates full interaction between service control and resource control planes, and between application developers and the policies that govern resource control. Furthermore, a novel, session based end-to-end policy control architecture is proposed to support inter-domain coordination across IMS domains. This architecture uses SIP inherent routing information to discover the routes traversed by the signalling and the associated routes traversed by the media. This mechanism effectively allows applications to issue resource requests from their home domain and enable end-to-end QoS connectivity across all traversed transport segments. Standard interfaces are used and transport plane overhaul is not necessary for this functionality.

The Common PCC, application driven and session based end-to-end architectures are implemented in a standards compliant and entirely open source practical testbed. This demonstrates proof of concept and provides a platform for performance evaluations. It has been found that while there is a cost in delay and traffic overhead when implementing the complete architecture, this cost falls within established criteria and will have an acceptable effect on end-user experience.

The open nature of the practical testbed ensures that all evaluations are fully reproducible and provides a convenient point of departure for future work. While it is important to leave room for flexibility and vendor innovation, it is critical that the harmonisation of NGN/IMS resource management frameworks takes place and that the architectures proposed in this thesis be further developed and integrated into the single set of specifications. The alternative is general interoperability issues that could render end-to-end QoS provisioning for advanced multimedia services almost impossible.

Contents

Linking Session Based Services with Transport Plane Resources in IP Multimedia Subsystems

Richard Good

Supervisor:

Neco Ventura

Thesis Presented for the Degree of

DOCTOR OF PHILOSOPHY

in the Department of Electrical Engineering

UNIVERSITY OF CAPE TOWN

May 2009

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 Doctor of Philosophy Degree in Electrical Engineering at the University of Cape Town. 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.

Richard Good

.....

May 2009

Acknowledgments

I would like to express my thanks to the following individuals for their assistance and guidance during the course of this project.

Neco Ventura, for the guidance and numerous opportunities afforded to me throughout the project.

David Waiting, the criticism was often severe but always constructive.

Fabricio Carvalho de Gouveia, the discussions and exchange of ideas were invaluable throughout the project.

The Open IMS Core architects from Fraunhofer FOKUS and all contributors to open source initiatives used throughout the project. Without the dedication and support of these individuals the open nature of this work would not have been possible.

Vitalis Ozianyi, Eugene Golovins and all past and present members of the Communications Research Group.

My family and friends, for their support and encouragement.

Synopsis

The massive success and proliferation of Internet technologies has forced network operators to recognise the benefits of an IP-based communications framework. The IP Multimedia Subsystem (IMS) has been proposed as a candidate technology to provide a non-disruptive strategy in the move to all-IP and to facilitate the true convergence of data and real-time multimedia services.

Despite the obvious advantages of creating a controlled environment for deploying IP services, and hence increasing the value of the telco bundle, there are several challenges that face IMS deployment. The most critical is that posed by the widespread proliferation of Web 2.0 services. This environment is not seen as robust enough to be used by network operators for revenue generating services. However IMS operators will need to justify charging for services that are typically available free of charge in the Internet space. Reliability and guaranteed transport of multimedia services by the efficient management of resources will be critical to differentiate IMS services.

This thesis investigates resource management within the IMS framework. The standardisation of NGN/IMS resource management frameworks has been fragmented, resulting in weak functional and interface specifications. To facilitate more coherent, focused research and address interoperability concerns that could hamper deployment, a Common Policy and Charging Control (PCC) architecture is presented that defines a set of generic terms and functional elements.

A review of related literature and standardisation reveals severe shortcomings regarding vertical and horizontal coordination of resources in the IMS framework. The deployment of new services should not require QoS standardisation or network upgrade, though in the current architecture advanced multimedia services are not catered for. It has been found that end-to-end QoS mechanisms in the Common PCC framework are elementary.

To address these challenges and assist network operators when formulating their

NGN strategies, this thesis proposes an application driven policy control architecture that incorporates end-user and service requirements into the QoS negotiation procedure. This architecture facilitates full interaction between service control and resource control planes, and between application developers and the policies that govern resource control. Furthermore, a novel, session based end-to-end policy control architecture is proposed to support inter-domain coordination across IMS domains. This architecture uses SIP inherent routing information to discover the routes traversed by the signalling and the associated routes traversed by the media. This mechanism effectively allows applications to issue resource requests from their home domain and enable end-to-end QoS connectivity across all traversed transport segments. Standard interfaces are used and transport plane overhaul is not necessary for this functionality.

The Common PCC, application driven and session based end-to-end architectures are implemented in a standards compliant and entirely open source practical testbed. This demonstrates proof of concept and provides a platform for performance evaluations. It has been found that while there is a cost in delay and traffic overhead when implementing the complete architecture, this cost falls within established criteria and will have an acceptable effect on end-user experience.

The open nature of the practical testbed ensures that all evaluations are fully reproducible and provides a convenient point of departure for future work. While it is important to leave room for flexibility and vendor innovation, it is critical that the harmonisation of NGN/IMS resource management frameworks takes place and that the architectures proposed in this thesis be further developed and integrated into the single set of specifications. The alternative is general interoperability issues that could render end-to-end QoS provisioning for advanced multimedia services almost impossible.

Contents

List of Figures

List of Tables

List of Acronyms

3GPP	Third Generation Partnership Project
AAA	Authorisation, Authentication and Accounting
AF	Application Function *or* Assured Forwarding
API	Application Programming Interface
APN	Access Point Name
A-RACF	Access - RACF
ARP	Allocation and Retention Priority
AS	Application Server
ATIS	Alliance for Telecommunications Industry Solutions
ATM	Asynchronous Transfer Mode
AUID	Application Unique Identifier
AVP	Attribute Value Pair
B2BUA	Back to Back User Agent
BBERF	Bearer-Binding and Event-Reporting Function
BE	Best Effort
BGF	Border Gateway Function
BGP	Border Gateway Protocol
BTF	Basic Transport Function
CAMEL	Customised Applications for Mobile Network Enhanced Logic
CDMA	Code Division Multiple Access
CERN	European Laboratory for Particle Physics Research
CMI	CP Management Interface
COPS	Common Open Policy Service
CoS	Class of Service
CP	Control Plane
CPE	Customer Premises Equipment

CPS	Calls Per Second
C-RACF	Core - RACF
CSCF	Call Session Control Function
DARPA	Defence Advanced Research Projects Agency
DHCP	Dynamic Host Configuration Protocol
Diffserv	Differentiated Services
DMTF	Distributed Management Task Force
DNS	Domain Name Server / System
DoS	Denial of Service
DSCP	DiffServ Code Point
DSL	Digital Subscriber Line
EDGE	Enhanced Data Rates for GSM Evolution
EF	Expedited Forwarding
EPC	Evolved Packet Core
EPG	Electronic Program Guide
ERF	Event Reporting Function
ETSI	European Telecommunications Standards Institute
FHoSS	FOKUS Home Subscriber Server
FOSS	Free and Open Source Software
GIST	General Internet Signalling Transport
GMPLS	Generalised Multi-Protocol Label Switching
GSM	Global System for Mobile Communication
GPL	GNU General Public Licence
GPRS	General Packet Radio Services
GNU	GNU's Not Unix
GUI	Graphical User Interface
HSDPA	High Speed Downlink Packet Access
HSS	Home Subscriber Server
HTTP	Hypertext Transfer Protocol
IARI	IMS Application Reference Identifier
I-CSCF	Interrogating - CSCF
ICSI	IMS Communication Service Identifier
ICMP	Internet Control Message Protocol
IDU	Indoor Unit

IETF	Internet Engineering Task Force
IF	Intermediate Frequency
iFC	initial Filter Criteria
IMS	IP Multimedia Subsystem
IMPI	IP Multimedia Private Identity
IMPU	IP Multimedia Public Identity
IMSSF	IP Multimedia Service Switching Functions
IMT	International Mobile Telecommunications
IN	Intelligent Network
INAP	IN Application Protocol
IntServ	Integrated Services
IP	Internet Protocol
IP-CAN	IP - Connectivity Access Network
IPSec	IP Security
IPTV	Internet Protocol Television
ISC	IMS Service Control
ISM	Industrial, Scientific and Medical
ITU-T	International Telecommunications Union - Telecommunications Standardisation Sector
JAIN	Java APIs for Integrated Networks
JSP	JavaServer Pages
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
LGPL	GNU Lesser GPL
LSP	Label Switched Path
LTE	Long Term Evolution
MCF	Media Control Function
MDF	Media Distribution Function
MMTel	Multimedia Telephony
MPDF	Media Policy Dataset Format
MPLS	Multi-Protocol Label Switching
MSRP	Message Session Relay Protocol
NACF	Network Attachment Control Function
NASS	Network Attachment Subsystem

NAT	Network Address Translation
NGN	Next Generation Network
NSIS	Next Steps In Signalling
NSLP	NSIS Signalling Layer Protocol
NTLP	NSIS Transport Layer Protocol
NTRD	Network Topology and Resource Database
O&M	Operations and Management
ODU	Outdoor Unit
OMA	Open Mobile Alliance
OSA	Open Services Architecture
OSIMS	Open Source IMS Core
P-CSCF	Proxy - CSCF
PBMAN	Policy-Based Management of Ambient Networks
PBNM	Policy Based Network Management
PCC	Policy and Charging Control
PCEF	Policy and Charging Enforcement Function
PCIM	Policy Core Information Model
PCIMe	PCIM extensions
PCRF	Policy and Charging Rules Function
PD-FE	Policy Decision - Functional Entity
PDF	Policy Decision Function *or* Portable Document Format
PDN	Packet Data Network
PDP	Policy Decision Point *or* Packet Data Protocol
PE-FE	Policy Enforcement - Functional Entity
PHB	Per Hop Behaviour
PoE	Power over Ethernet
PSTN	Public Switched Telephone Network
PQIM	Policy QoS Information Model
QCI	QoS Class Identifier
QMO	QoS Parameter Matching and Optimisation
QNE	QoS NSLP Entity
QNI	QoS NSLP Initiator
QNR	QoS NSLP Responder
QoS	Quality of Service

QOSM	QoS Model
QSPEC	QoS Specification
RA	Resource Allocator
RACF	Resource and Admission Control Function
RACS	Resource and Admission Control Subsystem
RAN	Radio Access Network
RCEF	Resource Control Enforcement Function
RCIP	Resource Connection Initiation Protocol
RF	Radio Frequency
RFC	Request For Comments
RM	Resource Manager
RMF	Resource Management Function
RSVP	Resource reSerVation Protocol
RTCP	Real Time Control Protocol
RTSP	Real Time Streaming Protocol
RTP	Real Time Protocol
RTT	Round Trip Time
SCF	Service Control Function
S-CSCF	Serving - CSCF
S-PDF	Service-Based Policy Decision Function
SAE	Services Architecture Evolution
SBLP	Service-Based Local Policy
SDP	Session Description Protocol *or* Service Delivery Platform
SDPng	Session Description Protocol next generation
SER	SIP Express Router
SIP	Session Initiation Protocol
SLA	Service Level Agreement
S/MIME	Secure / Multipurpose Internet Mail Extensions
SNMP	Simple Network Management Protocol
SOA	Service-Oriented Architecture
SOAP	Simple Object Access Protocol
SPR	Subscription Profile Repository
SQL	Structured Query Language
SS7	Signalling System 7

SU	Subscriber Unit
TCP	Transmission Control Protocol
THIG	Topology Hiding Inter-network Gateway
TISPAN	Telecoms and Internet converged Services and Protocols for Advanced Networks
TLS	Transport Layer Security
TRC-FE	Transport Resource Control - Functional Entity
TRCG-FE	Transport Resource Control GMPLS - Functional Entity
TRE-FE	Transport Resource Enforcement - Functional Entity
TRIS	Topology and Resource Information Specification
UE	User Equipment
UDP	User Datagram Protocol
UCT	University of Cape Town
UMTS	Universal Mobile Telecommunications Standard
UNI	User to Network Interface
URI	Uniform Resource Identifier
UTRAN	UMTS Terrestrial RAN
VoD	Video on Demand
VoIP	Voice over IP
WLAN	Wireless LAN
XACML	eXtensible Access Control Markup Language
XCAP	XML Configuration Access Protocol
XDMS	XML Document Management Server
XML	eXtensible Markup Language
x-RACF	Generic RACF
µBST	Micro Base Station

Chapter 1

Introduction

The research project initiated by the US Defence Advanced Research Projects Agency (DARPA) in 1973 to investigate techniques and technologies for interlinking packet based networks, developed what has become known as the TCP/IP Suite. TCP/IP is named after the two primary developed protocols, the Transmission Control Protocol (TCP) and the Internet Protocol (IP). The system of networks that emerged from this research became known as the Internet.

Since inception, the Internet has revolutionised the way we connect, interact, do business and essentially co-exist with one another. The invention of the World Wide Web in 1990 by Tim Berners-Lee of the European Laboratory for Particle Physics Research (CERN) paved the way for the Internet as the application platform we know today. With the number of end-users connected to the Internet exceeding 1400 million, or 20% of the world population, as of June 2008 [1], IP has become the dominant network layer protocol.

Telecommunications advances have introduced exceedingly portable devices and developments in wireless networking standards have allowed these devices to achieve true mobility. Furthermore, the underlying technology over which the Internet is run has seen a bandwidth increase to the point where a new generation of users have been created that are "always on" the Internet. These users value connectivity as a vital service that provides them with access to the broad set of Internet applications. As well as technological advances, there has been a paradigm shift in the way the Internet is used. The landscape has evolved from a predictable, operator managed, single service environment, to one with a multitude of unpredictable, often user-created services. The typical behaviour of a connected node has become a moving target in the Internet arena.

These increasingly empowered and contributing users have resulted in an explosion of new multimedia applications that require improved network performance or Quality of Service (QoS) guarantees; where QoS is defined as the aggregate effect of service performance that determines the degree of satisfaction of a user of the service. It has been suggested that advances in wireless and wireline access networks, resulting in higher data rates, may downplay the need for QoS management [2]. However, the Global IP Traffic Forecast and Methodology 2006 - 2011, carried out by Cisco Systems [3], indicates that global IP traffic will grow exponentially over this time period, driven by high definition video applications and high-speed broadband penetration. Essentially some form of managed network performance will be mandatory for the satisfactory delivery of these services over the future Internet.

The voice telephony world has experienced mixed advancement; mobility support has increased dramatically, and the development of the Intelligent Network (IN) platform has allowed operators to differentiate themselves by providing value-added services including televoting, call screening, number portability, toll-free calls and pre-paid accounting. However, network operators have invested hugely in legacy Public Switched Telephone Network (PSTN), Global System for Mobile Communication (GSM) and other circuit-switched technologies and have been reluctant to evolve from this status quo by introducing modern packet-switched communications infrastructure. With the massive success and proliferation of Internet technologies it is no surprise that operators are now recognising the benefits of an IP-based communications framework.

1.1 All-IP - Next Generation Network

The Next Generation Network (NGN) as defined by the International Telecommunications Union - Telecommunication Standardisation Sector (ITU-T) is a packet-based network capable of providing telecommunications services. Such a network should make use of broadband QoS-enabled transport technologies that separate the service related functions from the underlying transport functions [4].

This envisaged high-speed, secure, ubiquitous network, capable of catering for diverse application domains is often seen as the convergence of fixed and mobile technologies, interworking most existing access networks with a packet-switched core. Due to its intrinsic technology heterogeneity and wide scale deployment, it can be assumed that IP will form the basis of this NGN.

The realisation of an All-IP based network architecture has benefits for both subscribers and network operators. The maintenance of a single core network will greatly reduce operational expenditure and network complexity, and the deployment of new services will be faster and more efficient because of the nature of IP. In particular a single service could be deployed across a range of access technologies with minimum modifications. In the current architecture, services are integrated vertically across access technologies. These services need to be duplicated for deployment across each access, greatly delaying time to market, and capital expenditure. With an All-IP architecture the subscribers in turn will benefit from a wider range of services, simplified billing and reduced costs.

1.1.1 Access and Bandwidth Proliferation

Legacy cellular and wired networks were designed for voice transmission only and utilised circuit-switching technology, which is characterised by low bit rates and low delay variation. Cellular technologies first introduced packet-switched provisioning through General Packet Radio Services (GPRS), commonly known as 2.5G systems. This technology evolved to the Enhanced Data Rates for GSM Evolution (EDGE) and the Universal Mobile Telecommunications Standard (UMTS). The next step in this evolution is the Long Term Evolution (LTE) [1]. This architecture supports higher data rates, increased spectral efficiency and lower infrastructure costs. Central to this framework is the Evolved Packet Core (EPC), a flat, simplified, All-IP based core. This technology meets key NGN requirements as specified in the ITU-T International Mobile Telecommunications-Advanced (IMT-Advanced) specification for NGN mobile systems [5] - in particular LTE integrates existing and evolved access network systems via the EPC and provides an advanced, high-speed air interface capable of supporting advanced services.

Similarly, great steps have been taken with the development of Digital Subscriber Line (DSL) and Fibre access. These technologies offer exceptionally high data rates, and while deployment is not suitable for some developing environments, these are the predominant technologies pushing broadband proliferation in developed countries. Fixed wireless communications systems have also advanced; IEEE 802.11x, 802.15x and 802.16x standards provide portable connectivity. Furthermore 802.16e promises full mobility

[1] GPRS, EDGE, UMTS and LTE are Third Generation Partnership Project (3GPP) developed mobile standards; sister standardisation body 3GPP2 specifies similar standards for Code Division Multiple Access-based networks.

and is another candidate to be included as part of the ITU-T IMT-Advanced Systems specification. While these packet-based technologies were initially designed for data only, they now incorporate real-time multimedia services. This proliferation of access technologies and the convergence of data and real-time multimedia services has helped accelerate broadband penetration, driving the migration to an All-IP network.

1.1.2 IP QoS Provisioning

The realisation of an All-IP core requires careful planning due to the complexity of signalling, QoS, security and mobility issues handled by the previous circuit-switched protocols; customers will expect service quality to be at least as good on the All-IP network. IP was conceived as a "best-effort" protocol, providing robust but unreliable service delivery. While this model suited early Internet applications, such as e-mail and file transfer, current and future applications may have strict service requirements and might not adapt well to changes in network load. Despite the fact that the NGN architecture is based on large capacity core fibre networks, bandwidth intensive services with strict real-time requirements present QoS challenges for network operators.

Various QoS models exist to optimise IP performance, these include Differentiated Services (DiffServ) and Multi-Protocol Label Switching (MPLS). These techniques differentiate between service classes using packet marking techniques, and provide statistical guarantees of packet delivery assuming all traversed domains support the QoS model. Integrated Services (IntServ) paired with the Resource reSerVation Protocol (RSVP) attempt to provide end-to-end delivery guarantees, but this model suffers from scalability constraints. The Internet Engineering Task Force (IETF) is working on an end-to-end signalling protocol suite, the Next Steps In Signalling (NSIS), with QoS as its first use case. The QoS NSIS Signalling Layer Protocol (NSLP) extends RSVP, addressing many shortcomings including scalability. The QoS NSLP is independent of a specific QoS model and all information particular to the QoS model is encapsulated in a separate object known as the QoS Specification (QSPEC); QSPEC templates have been defined for various QoS models. QoS NSLP signalling requires modification of all routing devices in the transport plane, and while this protocol may be important for end-to-end service delivery in the future Internet, practical implementation challenges limit the applicability of this approach. In particular network operators heavily invested in legacy networks will be hesitant to commit the necessary capital expenditure for such an overhaul. In-

teroperability between different QoS models and networks, is also a critical factor; the lack of standardised reference points in network equipment makes it complex to reuse built-in mechanisms to provide end-to-end QoS across administrative domains.

One of the aforementioned advantages of IP based networks is the rapid provisioning of new services. However without a comprehensive QoS framework that guarantees the amount of bandwidth assigned to a user, or the delay packets will experience, the quality of real-time multimedia services will vary greatly, thus leading to an unsatisfactory end-user experience. Highly integrated services are another important end-user requirement; end-users want to be able to use services developed by large vendors and operators, as well as services developed by smaller third parties. Furthermore, they want to be able to combine these services to create entirely new services. Without standardised interfaces, service integration between multiple vendors becomes complex because of interoperability issues. In addition to QoS provisioning, charging mechanisms are critical for real-time multimedia sessions. The current "one size fits all" data charging scheme will not work with the envisaged myriad of available services. Instead custom business models for each service will be necessary to ensure that customers are charged according to the service they receive. The IP Multimedia Subsystem (IMS) has been proposed as an architecture to address these issues and facilitate the true convergence of data and real-time multimedia services.

1.2 The IP Multimedia Subsystem

The IMS was initially defined as an overlay architecture for the evolution of GSM systems for multimedia service provisioning. It aimed to provide a non-disruptive strategy that allowed network operators to create new revenue streams by changing their business focus from access provisioning to service provisioning.

It has since evolved and is now a central IP service element within the NGN architecture. This communications framework allows for the rapid development and deployment of highly integrated, multimedia rich services across numerous access platforms. IMS technology promises to merge two paradigms that have experienced massive success and proliferation: the Internet and Cellular worlds, effectively making Internet services available to any user, anywhere and at any time.

GPRS and the evolution to LTE have introduced packet-switched capabilities to mobile technologies. However IMS offers true convergence as it incorporates three important

concepts: QoS, charging and highly integrated services [6]. IMS provides a framework for the synchronisation of session establishment and transport plane resources ensuring a predictable end-user experience even when carrying bandwidth hungry, multimedia rich services. Additionally IMS provides information about services being invoked by users that can be used to define appropriate business models. No particular business model is specified, instead relevant information is collected and collated, allowing an operator to create suitable charging scenarios, be they traditional time and distance-based charging, subscription-based charging, QoS-based charging, or any new charging concept. Highly integrated services are another important justification for IMS technology. The modern, connected user wants access to services from a multitude of providers, as well as the ability to combine and integrate these services to create new services. To facilitate this and maintain all existing services, IMS defines standardised reference points and utilises existing Internet protocols. Essentially IMS provides ubiquitous access through Cellular technologies and allows for innovative and rapid service creation through Internet technologies. The process of choosing protocols, and defining necessary interactions and functionality, is a complex and contentious task, consequently the standardisation of IMS technology has been fragmented between numerous standardisation bodies and has been a lengthy procedure.

1.2.1 IMS Standardisation

The ITU-T defined International Mobile Telecommunications-2000 (IMT-2000) specification is the global standard for 3G mobile networks. IMT-2000 is the result of collaboration between different standardisation bodies. ITU-T has since released the IMT-Advanced standard stipulating the requirements for 4G mobile systems. The Third Generation Partnership Project (3GPP) and its sister organisation 3GPP2 are two bodies involved with the development of IMT-2000 compliant systems. 3GPP systems have evolved from GSM, while 3GPP2 systems have their origins in Code Division Multiple Access (CDMA) 2000 technology. The 3GPP defined LTE specification is a candidate for the IMT-Advanced systems.

3GPP was the first to recognise the need to evolve its network to a packet-based core and introduced the IMS as part of their Release 5 specification in 2002. This architecture has since evolved and at the time of writing 3GPP Release 8 has recently been finalised (Functional freeze date December 2008 [7]). The European Telecommunica-

tions Standards Institute (ETSI) initiated the Telecoms and Internet converged Services and Protocols for Advanced Networks (TISPAN) technical committee to standardise the NGN framework for fixed access; the ITU-T also defines a general NGN Framework. IMS is a central IP service element in both of these architectures.

There is an increasing amount of collaboration between standardisation bodies to maintain a single set of IMS specifications, dubbed Common IMS. In 2007 TISPAN shifted their IMS specifications and requirements to the 3GPP working groups ensuring that standardisation of the architecture only takes place within the 3GPP organisation. In North America, the Alliance for Telecommunications Industry Solutions (ATIS) studies the applicability of IMS and NGN to North American fixed access networks; they introduce new requirements to the 3GPP. PacketCable is an initiative from the Cable industry established by CableLabs. The PacketCable 2.0 specifications define IMS applicability for provisioning multimedia services over cable networks. PacketCable is an active contributor to the single IMS specification maintained by 3GPP. Other interested standardisation bodies like the WiMax Forum and the Broadband Forum specify requirements for interaction and integration with their own architectures, these requirements eventually find their way into the 3GPP single set of specifications.

Despite the attempt to maintain a single specification, IMS exists in various forms, one common factor is that they are all based on Internet protocols. The Internet Engineering Task Force (IETF) is a loosely, self-organised collection of interested individuals that is responsible for the creation and maintenance of Internet protocols. Having recognised the power and flexibility of these protocols and not wanting to re-invent the wheel, IMS specifications wherever possible, reuse existing protocols, one of which is the Session Initiation Protocol (SIP). SIP is an application layer signalling protocol used for session management and due to its flexibility and simplicity it has become the dominant protocol for IP multimedia services. Of particular importance is the separation of the control and transport planes; while SIP signalling must traverse a determined set of proxies, the media will typically make use of IP routing principles and follow the shortest path to the destination. While this separation simplifies operation and minimises media delay it does introduce a need for synchronisation between the planes, in particular a mechanism is needed to provide flexible but stringent control of network resources based on session negotiation signalling. SIP, as a flexible application layer protocol, can run over User Datagram Protocol (UDP), TCP or Asynchronous Transfer Mode (ATM) transport planes. To fulfil certain IMS requirements 3GPP has worked with the IETF to define

SIP extensions; these include support for mobility, provisional acknowledgements and event subscriptions and notifications, amongst others.

Diameter is another IETF defined protocol critical to the functionality of IMS. Diameter is an Authorisation, Authentication and Accounting (AAA) protocol that extends the Radius protocol. The power of Diameter lies in Diameter applications; the Diameter specification defines a base protocol on which extensions can be performed creating new commands and/or attributes, allowing the protocol to cover a wide range of functions, from resource admission control to charging information management. 3GPP and the IETF work together on the definition of these IMS specific Diameter applications. Diameter runs over IP Security (IPSec) or Transport Layer Security (TLS) ensuring the integrity of the connection. IMS utilises other well regarded IETF protocols such as the Real Time Protocol (RTP), the Real Time Control Protocol (RTCP) and the Session Description Protocol (SDP).

It is clear that there are a myriad of bodies involved in the process of IMS standardisation, this has resulted in a quite lengthy procedure. There has been much confusion and interoperability concern among vendors and operators, and it is hoped that 3GPP Release 8 will resolve these concerns and lead to wide-spread implementation and deployment. An important part of the Release 8 specification is the introduction of an evolved QoS concept that maximises operator control over QoS functions distributed across different network nodes.

1.2.2 Service Delivery Platforms / Service Oriented Architectures

With the introduction of the Intelligent Network (IN) platform in the 1980's, the concept of service independent platforms was introduced. Before this the development of any new service required the creation of a corresponding architecture. This "stove pipe" approach led to the implementation of many service specific installations within operator's networks. IN defined an overlay service architecture and the IN Application Protocol (INAP) on top of Signalling System 7 (SS7) to facilitate real-time interactions between functional components. Object orientation and distributed middleware took off in the 1990's and Application Programming Interfaces (API) were introduced to allow for flexible service creation - telecommunication API standards include Parlay, 3GPP Open Services Architecture (OSA) and Java APIs for Integrated Networks (JAIN). These APIs

aimed to make service implementation simpler by abstracting the underlying signalling protocols and telecommunications architecture. While this concept exposed network capabilities to third parties and was a promising technology, market uptake was poor; the main reason being that the APIs were too complex for developers not familiar with the underlying telecoms architecture [8].

The Service-Oriented Architecture (SOA) principle extends these concepts; it does not specify any API or overlay architecture but rather refers to the use of services as individual building blocks to create an enriched end-user experience. In the IMS architecture these building blocks or *service enablers* are hosted on Application Servers (AS) and the IMS acts as a mere docking station for these services. This means that any existing or future service platform can be reused and combined, as long as the relevant IMS adaptor is implemented. Service enablers provide reusable service capabilities that can be combined to create combinational services. Common service enablers provided by IMS include capability negotiation, authentication, service invocation, addressing, routing, group management, presence, resource provisioning, session establishment and charging [9]. While the exact definition of the Service Delivery Platform (SDP) is open to interpretation, the term usually refers to a set of components that enable the creation, orchestration, control and execution of any number of services. A set of SOA based service enablers interacting and combining to create advanced multimedia rich applications, constitutes an SDP. IMS acts as a docking station and launching point for the invocation of these applications. Essentially IMS provides a signalling infrastructure with well defined interfaces, and the SDP uses these interfaces to create and orchestrate services.

3GPP do not focus on the standardised implementation of services apart from basic voice, video and conferencing applications. The primary body responsible for this is the Open Mobile Alliance (OMA). This organisation focuses on usability and interoperability of service enablers; while there is some overlap in the work done by the OMA and 3GPP, the general agreement is that OMA specifies service requirements on the IMS, and 3GPP extends the IMS specifications to meet these requirements. Though this relationship is evolving, shown by the creation of the Multimedia Telephony (MMTel) standard jointly developed by 3GPP and TISPAN. This converged service offers real-time, multimedia communication using a range of media capabilities.

The driving motivation for IMS adoption is the rapid and efficient creation of highly integrated, scalable and chargeable services - critical to this adoption will be rapid ser-

vice uptake and time to market. Customer uptake is heavily dependent on end-user experience, thus scarce network resources need to be managed end-to-end to ensure sufficient quality of experience. Additionally, services need to be rapidly deployed and standardisation of QoS support for individual services should not be necessary.

1.2.3 Web 2.0 Revolution

During the lengthy IMS standardisation process, the Internet has changed dramatically. The huge base of application developers working on the open Internet platform, using mainstream Internet programming technologies has resulted in the development of a wide range of innovative Internet services. While the exact definition of Web 2.0 is open to interpretation, it generally refers to the perceived transition of the World Wide Web from a collection of websites to a complete computing platform serving Internet applications, or as the new generation of Internet services with the defining theme being the harnessing of collective intelligence. This new age of community web services, with millions of empowered and contributing users, exploits the wisdom of the masses and the fact that a large number of people sharing a common experience is preferred to static expert judgement. The phenomenon is constantly evolving as new services shape the Internet landscape, hence the fluid definition. The extremely popular social networking applications, and Web Mashups, where Mashup refers to the sourcing of data from multiple applications to create a new, integrated and distinct service, are part of this revolution.

In recognition of the massive success and proliferation of Internet services, Parlay X was created in 2000, and later endorsed by 3GPP [8]. Parlay X provides a specific set of Web service interfaces similar to those provided by the object-oriented and middleware based APIs, but provides improved mechanisms for integration with existing Web based applications. SIP servlets have also been defined; these extend Java servlet technology typically used for Web application development, to implement value added services over SIP based architectures.

The business model for such online services is still under investigation and there seems to be no "one size fits all" approach. The comparatively low entry costs have led to a large number of small start up companies. At present most of these services are available free of charge and revenues are largely based on the potential for personalised advertising. The huge popularity of these services is shown by the fact that Internet

advertising revenue grew 20.9% world wide from $41.35 billion in 2007 to $49.99 billion in 2008. Furthermore the Internet is the only advertising medium where increased revenue is expected for 2009 [10].

This new model of service delivery poses a threat to IMS deployment but also an opportunity; it puts pressure on operators to provide innovative and interactive services, often created by third party developers who have only recently entered the market.

1.2.4 IMS Deployment Challenges

The objective to create a controlled environment for deploying IP services, and hence increase the value of the telco bundle, is desirable for all network operators. However there are several critical issues that raises the question whether or not IMS is the technology to achieve this objective.

The first issue is presenting a business model for deploying IMS services, network operators will be reluctant to commit the necessary capital expenditure to roll out IMS based networks without a sound business case for doing so. IMS provides the means for deploying IP multimedia services, but does not actually specify these services. Without a "Killer Application", IMS deployment remains risky, particularly when considering that most deployments are done on an application by application basis, and the point at which the IMS becomes more cost-effective than service specific approaches is difficult to determine [11]. Proponents argue that as well as acting as a common platform for multiple applications and services, IMS allows operators to deploy converged voice services without waiting for network convergence to happen, and this in itself is a "Killer Application" [12].

The 3GPP Release 8 specifications promise at least a base IMS specification to drive early implementations, however these specifications define a reference architecture and not a physical implementation. This is the case with most standardisation bodies, room is left for innovation and implementation interpretation so vendors may differentiate themselves. This has led to concerns over interoperability and increasing integration and R&D costs when deploying IMS technology. Despite these concerns, and the general Fear Uncertainty and Doubt surrounding the IMS, vendor surveys indicate that IMS will form an integral part of the telecommunications environment [13]. Global sales of IMS equipment increased by 94% in 2008, from 2007. Over 100 network operators have begun investigating IMS and have chosen vendors, though less than 50% of these have deployed

the technology and are carrying live traffic [14]. Adoption will be fueled in part by the expected deployment of the Long Term Evolution (LTE) and Evolved Packet Core (EPC) technologies, of which IMS is a central IP service element, in the first quarter of 2010 [15].

One of the most critical challenges is that posed by the widespread proliferation of Web 2.0 services. This dynamic environment is not seen as sufficiently robust to be used by mainstream network operators for revenue generating, real-time services. However operators deploying IMS technology face the difficult task of justifying charging for services that are typically freely available on the Internet. IMS proponents maintain that this justification can be achieved through service differentiation. Particularly IMS services will be more secure than Web services, through service enablers and combinational services IMS will provide a more integrated and rich end-user experience, and finally IMS will provide reliability and guaranteed transport of multimedia services through efficient management of network resources.

1.3 Policy Based Resource and Admission Control

The IMS framework places important emphasis on resource management to ensure that real-time and multimedia applications perform as intended through the allocation of limited resources. Despite the speed increases promised by the latest wireless technologies, bandwidth requirements for multimedia applications are growing exponentially. Preferential treatment based on the type of service provided will be critical in differentiating IMS services from typical web services.

With the separation of the control and transport planes, a resource management framework is needed to manage resources on both planes in a tight but flexible manner, decoupling the core network components and procedures from the subtleties of the access and transport networks. Typically when IMS sessions are established, the service requirements for the session are described in the SIP signalling using SDP. This manner of service classification needs to be translated to actual resources allocated in the transport plane. A critical component in this resource management framework is a logical entity that provides a northbound interface to the control elements and a south bound interface to the transport elements to provide synchronisation and linkage between these two planes.

The high level requirements of the IMS resource management framework include minimal effect on session setup delay, backward compatibility, convergence towards ag-

nostic access, and rapid time to market of new services [16]. Additionally the framework should guarantee resources and perform admission control along all traversed transport segments, including across administrative domains, ensuring end-to-end QoS connectivity. Differentiation between users and services and the ability to flexibly manage resources should be under the control of the network operator. Policy Based Network Management (PBNM) is a network management model that can simplify the resource management function while facilitating this operator controlled environment.

1.3.1 Policy Based Network Management

PBNM as a network management model facilitates the automatic and distributed management of networks. This paradigm simplifies the configuration and administration of complex networks through the specification of abstract policies. This model allows a network to provide an automatic response to changing network conditions according to operator policies.

Policies are central to the functioning of the PBNM system as they define the set of rules to administer, manage and control access to network resources. Each policy is made up of rules that may have static or dynamic characteristics. Policy rules are defined by conditions and actions; a condition is evaluated when a policy decision is triggered; if a condition is evaluated to true, the associated actions are taken. There are a number of different policy levels within the PBNM system, these range from platform independent business policies to device specific configuration policies each representing a different level of policy abstraction. The process of translating policy information between the different levels is known as policy refinement.

Policy concepts are stored in Policy Information Models; these models are typically platform and technology independent. Policy Data Models represent the lower levels of policy abstraction and typically contain technology specific configuration information. There are numerous proposals for Policy Information and Policy Data Model representation. The IETF in conjunction with the Distributed Management Task Force (DMTF) have defined object oriented models for policy representation, namely the Policy Core Information Model (PCIM), its extensions (PCIMe) and the Policy QoS Information Model (PQIM) specifically for representing QoS policies. These generic models are vendor and device independent but can be extended to create business level and network level policies. The main challenge facing deployment of PBNM systems is the variety

of policy representation formats that are involved. However there is a move to synchronise policy formats using the eXtensible Markup Language (XML) as a basis for creating self-describing, human-readable and portable policies at all levels of the policy life cycle [17].

The IETF have defined a PBNM architecture to apply across all areas of network management. This architecture defines critical functions for policy storage according to policy information models, policy retrieval, policy decision triggering and policy enforcement. Essentially a Policy Management Station uploads policies to a Policy Repository; based on asynchronous events or explicit requests a Policy Decision Point downloads relevant policies and makes policy decisions; these decisions translate into configuration policies that are enforced on target devices or Policy Enforcement Points distributed across the network. This architecture can be applied to network security, privacy control, resource management, or any scenario where access to a resource needs to be restricted, and distributed and automated management of this access is desirable. The IETF PBNM has been adopted by IMS and NGN standardisation bodies, including 3GPP, TISPAN and ITU-T, to form the basis of their resource and admission control frameworks.

1.3.2 Policy Control Frameworks

The resource management framework for NGN architectures needs to efficiently manage a series of network resources to guarantee delivery of a wide range of QoS sensitive services over multiple transport technologies through QoS resource control. This involves policy control and admission control, where policy control refers to the process by which a new dynamic service flow is created in the transport plane upon a resource request, and admission control refers to the process of allowing a new service flow access to resources. QoS resource control consists of service based admission control or *Authorisation*, resource-based admission control or *Reservation,* and enforcement of reserved resources or *Commitment* [18].

The 3GPP was the first to open up network resource control to applications; they introduced the Service-Based Local Policy (SBLP) architecture based on PBNM in 2002. This architecture has evolved and as of Release 7 incorporated the Flow Based Charging architecture to form the Policy and Charging Control (PCC) Architecture. This architecture included in-depth interface and protocol definitions, but was limited to the IMS as a service element and was designed specifically for mobile networks. The Release 8

PCC architecture is extended to process resource requests from any number of IP service elements. This evolved architecture also defines new interactions for inter-domain communications and access agnostic enforcement.

Having adopted the IMS as the basis for their NGN, TISPAN began work on their Resource and Admission Control Subsystem (RACS) in 2003. This architecture defines a resource control plane to manage access in the transport plane. The RACS is largely based on the PCC architecture but attempts to facilitate truly access agnostic resource reservation, by processing resource requests from any IP service element and enforcing decisions in any access network.

In 2004 ITU-T introduced the Resource and Admission Control Function (RACF) based on the early work of the 3GPP and TISPAN. The RACF is a high level reference framework that covers the broad aspect, encompassing fixed and mobile networks and defining comprehensive control scenarios. CableLabs, the WiMax Forum and the Broadband Forum also define policy-based resource management frameworks for their particular access technologies. While there are no significant conflicts between these architectures, there are subtle differences [19], essentially the same standardisation harmonisation that took place with IMS technology needs to take place with the resource management functions. In Chapter 2 the PCC, RACS and RACF architectures are examined in detail; architectural alignment is performed and a generic QoS management framework model is presented that defines terms and functional elements to be used throughout the remainder of this thesis.

1.4 Policy Control Challenges

The NGN resource management framework faces similar deployment challenges to IMS technology in general, but the technology is in even earlier development stages. This is evident in the fact that most preliminary IMS deployments support policy controlled resource management in a scaled down and limited manner. Architectural alignment and harmonisation between the various standardised frameworks will be critical to avoid interoperability concerns that could cripple deployment, and the fact that specifications only define a reference architecture and not a physical implementation could exacerbate these concerns.

In this section critical open areas surrounding the resource management function are discussed; the support of NGN QoS is a broad topic but in this work the focus is on two

identified areas: vertical coordination and horizontal coordination of resources. Vertical coordination refers to the interaction between the applications requesting resources and the transport plane resources that will carry the application traffic; while horizontal coordination refers to the ability to provide seamless end-to-end QoS connectivity across administrative domains [20]. This covers the major deployment challenges faced when facilitating inter- and intra-domain policy controlled resource management across heterogeneous transport technologies.

1.4.1 Vertical Resource Coordination

Rapid development and deployment of innovative new services is the major draw card for early IMS adoption. This shapes a key requirement of NGN resource management frameworks, specifically that they should in no way hinder the innovative creation of services. This means that no new QoS standardisation should be necessary when deploying a new service, nor should a network upgrade be necessary when deploying a service with new requirements. The translation of complex, highly interactive, multimedia rich services into efficient, aggregated QoS resource requests is an ongoing research area [21][22]. Furthermore application developers have very little control over the way their services are treated in the transport plane, besides the highly granular description of the service requirements using SDP. There is very little interaction between the application developers and the policies that govern resource allocation.

There are numerous proposals for policy information representation, but the IMS and resource management specifications do not specify any particular model. While it is clear that technology independent policies should fully characterise a network path, and technology specific policies should include transport specific classifiers and/or link layer QoS information, there is no standardised method to perform this policy refinement and effectively map QoS descriptors across different layers of the policy life cycle.

To achieve full vertical coordination and reconcile the semantics of the control and transport planes, a multi-layered approach is necessary; the IMS model is further refined into the service control plane, the resource control plane and the transport plane. The service control plane comprises the IMS control elements and carries control signalling, the transport plane encompass the logical networking devices over which the media traffic is carried, and the resource control plane carries out the mediation between these two planes. Policy enforcement should take place at each of these planes, and automatic

policy refinement between different planes should be performed.

1.4.2 Horizontal Resource Coordination

End-to-end QoS support is a broad term, in this thesis it is used to describe inter-domain coordination across IMS administrative domains. This coordination can be facilitated at any of the planes within the IMS model. NGN architectures supporting IMS service control will likely implement some form of resource control in most domains and network segments, therefore it makes sense to exploit these already implemented mechanisms at the resource control plane. However proprietary interfaces in network equipment, resulting in highly vendor specific solutions, may hamper deployment. When combined with the lack of a general interface specification between service control functions and resource management functions, this could lead to general interoperability issues.

Transport plane signalling like RSVP and its successor QoS NSLP dynamically perform explicit QoS resource reservations and provide mechanisms for end-to-end coordination across administrative domains and QoS models. However as already mentioned these QoS signalling approaches require a transport plane overhaul, and operators who are already invested in expensive packet-switched networks will be reluctant to evolve from these technologies without exacting full return on investment.

Resource management frameworks in coordination with IMS can facilitate full QoS control in the originating and terminating domains along the session control signalling path using standardised mechanisms. However there is no mechanism to reserve resources in transit domains nor can the transport plane carrying the media be reconciled with, and bound to, the service control signalling path to provide end-to-end QoS connectivity for signalling and media.

Despite the agreement reached between standardisation bodies on the need for a framework to control admissions and resource allocation, the harmonisation of the functions is far from complete. Increased interaction between applications and the policies that govern resource control, and automatic refinement between these planes will help encourage innovative creation of IMS services, while still providing sufficient resource management to ensure correct treatment of real-time and multimedia applications. Furthermore performing inter-domain coordination at the service control plane and linking this resource reservation with the transport plane handling the media will allow for end-to-end mediation of resources in a network agnostic fashion, without significant

modifications to the transport plane.

1.5 Open Testbed Platforms

The term testbed refers to a development framework separate from the live environment and associated hazards. In the telecommunications context, such a platform can contain hardware and software components and facilitates rigorous, transparent and replicable testing of new technologies. While simulation models allow large scale representations of a particular problem, the models are typically subject to a number of assumptions that, though theoretically accurate, could simplify the problem and not take into account all variables of a practical network. In a testbed environment deployed technologies can be proved almost beyond a doubt.

The massive success and proliferation of Internet technology has shown that a large number of application developers working on an open infrastructure is needed for the development of successful and market driven services. The definition of an open testbed is inexact, but it is agreed that such a testbed architecture should be based on openly available standards. A deeper level of openness involves the distribution of the platform source code as part of the testbed package, this increases the set of developers by giving access to the core functions of the testbed. The final level involves releasing the source code freely to any interested individuals; the concept of Free and Open Source Software (FOSS) has been exploited by various initiatives and can facilitate rapid creation of new features by opening up the development to all and any users of the framework.

Service Delivery Platforms and IMS technologies in particular are topics of huge complexity. Open testbed initiatives like the Fraunhofer FOKUS Open Source IMS Core project [23], and the University of Cape Town (UCT) IMS Client project [24], have helped expose the IMS platform as a docking station for service deployment to an open set of developers, bringing academia and industry together. These projects encourage innovation in the field by ensuring reproducibility, and provide a convenient point of departure for future research.

The standardisation process of IMS/NGN resource management frameworks still has much ground to cover, including the harmonisation of the various specified architectures. This is highlighted by the limited number of policy based resource management framework deployments within the IMS context. Just like IMS technology in general, NGN/IMS resource management frameworks need to be exposed to a wide set of applica-

tion developers and researchers, to accelerate technology maturity and encourage future innovation in the field. Open testbeds are an ideal vehicle to achieve this exposure.

1.6 Thesis Objectives

The success of IMS will depend heavily on how appealing services are to end-users; reliability through efficient management of resources will be a critical factor in differentiating IMS services from typical web services. A comprehensive IMS resource management framework will be necessary to facilitate multi-network/multi-domain policy provisioning without hindering the innovative creation of IMS services. Such a deployment should allow end-to-end QoS connectivity across administrative domains, and provide interaction between application developers and policies that govern resource control.

This thesis has several objectives. First, it is critical to carry out a comprehensive review of existing standardisation work on IMS resource management frameworks. While such reviews exist in the literature [20][19], the standards have advanced rapidly and a snapshot of the current state of the art is necessary to perform architectural alignment and define a generic resource management framework that encompasses all the standardised architectures. This generic framework will define terms and functional elements used through the remainder of this thesis.

Second, it is important to address a key requirement of IMS resource management frameworks: providing linkage between session based services and transport plane resources without hindering the innovative creation of IMS services. Application developers should have greater freedom when describing their services and new service deployment should require no new QoS standardisation or network upgrade. This will involve addressing the interaction between planes within the IMS model and various areas left open by the standardisation bodies to allow for innovation and vendor differentiation, including policy representation, policy prioritisation, policy provisioning and application-policy interaction.

Third, it is imperative to identify IMS mechanisms to perform inter-domain coordination to facilitate end-to-end QoS connectivity. Backward compatibility with existing infrastructure will be vital to IMS deployment, hence this coordination should be performed at the service control plane where resources can be requested and reserved along all traversed transport segments without the need for transport plane overhaul. The question remains how to link the discovered service control inter-domain routes with the

routes followed by the media in the transport plane.

Last, the primary objective of this thesis is to design and implement a standards-compliant resource management framework that addresses the previous objectives, and to deploy this framework within a real IMS testbed. This will allow proof of concept of the various objectives, and facilitate realistic evaluations of the proposed extensions. In particular the overheads introduced by the proposed framework need to be evaluated in a practical IMS setting. The evaluations need to contrast the effects on network performance when incorporating a resource management framework and the proposed end-to-end QoS extensions to support advanced multimedia services, into the IMS architecture. The framework should be released as Free and Open Source Software to encourage collaboration and innovation, and provide a convenient point of departure for further research.

1.7 Thesis Scope

While IMS is central to the NGN architecture, it is seen as just one of many IP service elements. Hence an NGN resource management framework should be capable of receiving and processing requests from any number of service platforms. This work is limited to the IMS as a service platform, hence the resource management framework need only cater for requests of a certain nature, and information inherent in IMS signalling can be used for resource reservation and end-to-end route discovery. However it may be possible to apply this work to other IP service elements.

When referring to IMS, this work typically refers to the 3GPP maintained, single set of IMS specifications. Though these specifications are largely finalised, the policy controlled resource management framework still faces several deployment challenges and is an ongoing work within the standardisation bodies. All technical specifications and recommendations are the latest as of December 2008. This leads to another limitation, that is, the scarcity of IMS implementations; a Turkish operator has deployed an IMS based architecture and there have been IMS rollout announcements [12], but deployment results are preliminary at best. While comparisons between related future Internet architectures are possible the concepts examined in this thesis are largely based on a comprehensive literature review and in the context of IMS are still largely hypothetical.

The focus of this work is the interaction between the service control and resource control planes, and mechanisms to provide end-to-end QoS connectivity across admin-

istrative domains. While IP QoS models may be implemented as part of the proof of concept testbed, they are not the concentration of this work. Hence transport plane metrics like jitter and packet loss are not under study; instead metrics that might be affected by policy control mechanisms and in turn have an effect on end-user experience or network utilisation are examined, e.g. session setup delay and incurred signalling overhead.

An additional focus of this work is to facilitate innovative creation of IMS services by giving application developers greater freedom when describing the QoS requirements of their services, and more control over the way these services are treated in the transport plane. This work does not try to predict future service trends or IMS "Killer Applications".

PBNM can be applied to a wide range of applications, and IMS uses it to facilitate inter-application interaction, otherwise known as service brokering. This allows services and service components to expose their capabilities and interfaces based on different technologies to other services and service components via a dedicated policy mechanism. Policies are also used to facilitate flow based charging for both online and offline billing. This work does not consider these PBNM application scenarios.

1.8 Contributions

The major contributions of this thesis include:

- The critical review of the most prominent resource management frameworks for IMS/NGN, and the definition of a generic QoS management framework that encompasses work done by all standardisation bodies. The definition of terms and functional elements allows for more coherent and focused future research.

- Co-author of the UCT IMS Client, the first Free and Open Source IMS Client implementation released under GNU General Public Licence version 3 (GPLv3). This IMS emulation tool implements full IMS signalling, several multimedia rich services and a mechanism with which to test other IMS network components.

- Author of the UCT Policy Control Framework, a standards compliant, Free and Open Source IMS policy controlled resource management implementation released under GPLv3. This architecture extension incorporates policy control interactions

into the FOKUS Open Source IMS Core project, and allows applications to authorise, reserve and commit resources for IMS sessions.

- Design, implementation and evaluation of an application driven multi-layered policy control framework to facilitate full interaction between service control and resource control planes within the IMS and between application developers and the policies that govern resource control.

- Design, implementation and evaluation of a novel end-to-end policy control extension that uses SIP inherent routing information to discover the routes traversed by the signalling and the associated routes traversed by the media. This mechanism effectively allows applications to issue resource requests from their home domain and enable end-to-end QoS connectivity across transit domains.

These contributions are documented in the following selected peer reviewed publications.

Journal and Book chapter publications:

1. **R. Good**, F. Gouveia, N. Ventura and T. Magedanz, "Session Based End To End Policy Control in 3GPP Evolved Packet System", accepted for publication in the *Wiley International Journal of Communications Systems* special issue on Next Generation Networks.

2. **R. Good**, D. Waiting and N. Ventura, "Quality of Service Provisioning in the IP Multimedia Subsystem", book chapter accepted for publication in *"Quality of Service Architectures for Wireless Networks: Performance Metrics and Management"*, Edited by S. Adibi and R. Jain, published by IGI Global, 2009.

3. **R. Good**, F. Gouveia, S.Chen, N. Ventura and T. Magedanz, "Critical Issues for QoS Management and Provisioning in the IP Multimedia Subsystem", *Springer Journal of Network and Systems Management (JNSM)*, vol. 16(2), pp. 129-144, June 2008.

Conference publications:

1. **R. Good**, F. Gouveia, N. Ventura and T. Magedanz, "Policy-based Middleware for QoS Management and Signalling in the Evolved Packet System", *Proceedings of 2009 2nd International Conference on Mobile Wireless Middleware, Operating Systems, and Applications (MOBILWARE'09),* April 2009.

2. **R. Good** and N. Ventura, "Application Driven Policy Based Resource Management for IP Multimedia Subsystems", *Proceedings of 2009 5th International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TRIDENTCOM'09)*, April 2009.

3. D. Waiting**, R. Good**, R. Spiers and N. Ventura, "The UCT IMS Client", *Proceedings of 2009 1st Open NGN and IMS Testbeds Workshop (ONIT'09) in conjunction with TRIDENTCOM'09*, April 2009.

4. **R. Good** and N. Ventura, "End to End Session Based Bearer Control for IP Multimedia Subsystems", *Proceedings of IEEE/IFIP 2009 International Symposium on Integrated Network Management (IM'09)*, June 2009.

5. V. Ozianyi, **R. Good**, N. Carrilho and N. Ventura, "XML-Driven Framework for Policy-Based QoS Management of IMS Networks", *Proceedings of 2008 IEEE Global Communications Conference (GLOBECOM'08)*, November 2008.

6. **R. Good** and N. Ventura, "An Evaluation of Transport Layer Policy Control in the IP Multimedia Subsystem", *Proceedings of 2008 19th International IEEE Symposium of Personal, Indoor and Mobile Radio Communications (PIMRC'08)*, September 2008.

7. **R. Good** and N. Ventura, "Linking Session Based Services and Transport Layer Resources in the IP Multimedia Subsystem", *Proceedings of 2008 11th Southern African Telecommunication Network and Applications Conference (SATNAC'08)*, September 2008.

8. D. Waiting, **R. Good**, R. Spiers and N. Ventura, "Open Source Development Tools for IMS Research", *Proceedings of 2008 4th International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TRIDENTCOM'08)*, March 2008.

9. **R. Good**, F. Gouveia, S. Chen, N. Ventura and T. Magedanz, "Extending IMS QoS Provisioning to the Access Network", *Proceedings of 2007 10th Southern African Telecommunication Network and Applications Conference (SATNAC'07)*, September 2007.

1.9 Thesis Outline

In Chapter 2 an extensive review of the ongoing standardisation work regarding IMS resource management frameworks is presented. In particular the 3GPP PCC, the TISPAN RACS, and the ITU-T RACF are presented. This chapter provides a snapshot of the state of the art and performs architecture alignment presenting a generic QoS management framework that encompasses all of the standardised architectures. This chapter defines terms and functional elements used throughout the remainder of the thesis.

A comprehensive literature review is presented in Chapter 3, examining both vertical and horizontal coordination of resources in QoS management frameworks. Works concerning deployment of innovative IMS services over QoS aware transport planes are examined, and focus is given to Future Internet projects that define adaptive, end-to-end QoS mechanisms for heterogeneous networks.

Chapter 4 proposes extensions to the existing policy controlled resource management framework that allows multiple levels of policy control. This application driven concept facilitates interaction between services and the policies that govern resource control.

In Chapter 5 a novel mechanism is proposed that uses SIP inherent routing information to discover signalling routes, these routes are used to discover and bind associated media routes. This mechanism allows an application to effectively issue resource requests across all traversed transport segments ensuring end-to-end QoS connectivity.

Chapter 6 presents the testbed implementation of a standards compliant IMS resource management control function, and the integration of the multi-layered extensions and end-to-end route discovery mechanisms. The testbed is implemented and described in such a manner as to ensure easy reproducibility, and to provide a convenient point of departure for future work.

In Chapter 7 the described testbed and proposed enhancements are subjected to validation and performance tests and results are presented. Due to the nature of the practical testbed, it is possible to subject each incorporated component to realistic use case scenarios. The evaluations demonstrate proof of concept and also the effectiveness of the proposed enhancements.

Chapter 8 presents the conclusions drawn from the thesis and summarises the contribution. Recommendations are made for areas of further study.

Chapter 2

IMS/NGN Resource Management Framework Overview

Resource and admission control for the NGN architecture is a research area that has enjoyed much attention. There are a number of standardisation bodies working on various forms of resource management frameworks, however this complex area is relatively immature in the practical environment. Analyses of policy based resource and admission control functions exist in the literature [20][19], however because of recent advancements in the field, a thorough review of the standardised frameworks is necessary. The 3GPP PCC, the TISPAN RACS, and the ITU-T RACF are the major frameworks under investigation; standardisation bodies such as the WiMax Forum, Broadband Forum and 3GPP2 define their own resource management functions. These approaches are largely based on the aforementioned and hence are omitted from this study. This chapter aims to provide a comprehensive snapshot of the state of the art regarding mediation between QoS control elements and transport plane resources in the IMS/NGN framework, and to address the issue of harmonisation between architecture specifications.

In particular, duplicate standardised functions are identified and harmonised to address interoperability concerns that could hamper deployment. To facilitate more coherent and focused research, a set of generic terms and functional elements are defined encompassing the work done by all standardisation bodies.

The chapter begins with the 3GPP-defined PCC framework from which other NGN resource management frameworks have evolved; functional elements, control scenarios and interface definitions are presented for this framework. Similar analysis is carried out for the TISPAN defined RACS and the ITU-T defined RACF. The chapter concludes

with an examination of architectural alignment and a generic QoS management model that incorporates work from each of the aforementioned specifications.

2.1 3GPP Policy and Charging Control Framework

Along with the introduction of IMS technology as part of their Release 5 specification, 3GPP exposed resource management functions to applications through the Service Based Local Policy (SBLP) architecture. This architecture was further developed in Release 6; logical elements were separated and new reference points were defined. In Release 7 the SBLP architecture was combined with the Flow Based Charging architecture; this co-location made logical sense as the scope and functionality of the resource management and charging functions were closely aligned. The created architecture was known as the Policy and Charging Control (PCC) architecture. The PCC framework extended the interaction between the service control, resource control and transport planes, facilitated additional control scenarios, but was still tailored specifically for UMTS access, though attempts were made to integrate PacketCable and Wireless Local Area Network (WLAN) access [25].

Release 8 extends the scope of the framework. In particular IMS is now seen as one of many IP service elements, and authorisation requests do not only originate from the IMS service control plane. Additional extensions include support for a greater number of access technologies and QoS models, and support for inter-domain communication. In this analysis, the PCC architecture as defined in Release 7 is described, and the scope and architecture characteristics of the Release 8 specifications are presented.

2.1.1 Functional Elements

The Release 7 PCC architecture has three critical components: an Application Function (AF), a Policy and Charging Rules Function (PCRF) and a Policy and Charging Enforcement Function (PCEF) [26]. The AF logically resides in the service control plane and is essentially any element that might request resources; a Proxy - Call Session Control Function (P-CSCF) or an Application Server are typical AFs. These elements lie on the signalling path, and in the case of IMS session based services, they extract service information from the SIP signalling. This information is used to create authorisation requests that are passed to the PCRF in the resource control plane.

The PCRF instantiates the Policy Decision Point (PDP) specified by the IETF PBNM model and plays the role of the mediation element with a northbound interface to the service control plane and a southbound interface to the transport plane. The PCRF performs policy control which consists of authorisation, binding, establishment of media paths and QoS control.

The PCRF receives authorisation requests and upon extracting the service information performs authorisation. This authorisation is based on policies stored in a Policy Repository; the format, content, provisioning, storage and retrieval of these policies is regarded as network operator specific and therefore not standardised, though the definition of a Subscription Profile Repository (SPR) implies that subscription profile related policies should be present. Upon authorisation the PCRF defines a PCC Rule that contains service data flow filters to identify packet flows that constitute a service data flow. The PCC Rule also contains parameters that essentially describe how the service data flow should be treated in the transport plane, these rules can be dynamically created or pre-defined by the operator. The creation of the PCC rule is not specified and is deliberately left for operator configuration.

For the correct coordination of transport plane resources and QoS control, all elements involved in the process must be able to identify one another and in this way form a binding for each session; in particular the service information must be associated with the transport plane path that is to carry the service data flow. This session binding is performed using user-identity based identification; this scheme uses the UE IP address, or any kind of UE identity to identify the home domain and hence the QoS elements involved.

The manner in which the PCC Rule is enforced depends on the QoS reservation procedure in use; there are two models for requesting QoS enabled paths: end-point initiated establishment or *pull mode* and network-initiated establishment or *push mode*. In pull mode, intelligent User Equipments (UE) make resource requests from the transport plane themselves, these requests can be end-to-end or limited to a segment of the end-to-end path. Typically this model requires QoS negotiation support at the transport plane, like NSLP, or link layer QoS signalling capabilities, where protocol aware entities along the path determine if resources can be allocated to the requesting device [27]. In push mode, the entities involved with session negotiation make the request for resources; the PCRF installs or pushes the PCC rules to the PCEFs that logically reside on the transport plane devices. In both models the devices are configured to establish the transport

plane paths. The installed PCC rules identify service data flows based on the service data flow filters and the associated flows are treated accordingly, this is referred to as QoS control.

The Evolved Packet Core (EPC) is central to the Services Architecture Evolution (SAE) work item currently under standardisation by the 3GPP, where the SAE forms the All-IP based core network for the LTE architecture. The EPC supports mobility between heterogeneous access networks and incorporates an evolved QoS concept that is aligned with the PCC framework.

In this evolved architecture the IMS is seen as one of several IP service elements, hence the AF is no longer limited to IMS specific elements. The PCRF provides connectivity between the EPC and the IP service elements. This element performs the same role as before but has its functionality split into home domain and visited domain functions. The definition of home and visited PCRFs allows the EPC to offer breakout for data in the home and visited domains. This new system introduces service level QoS parameters that are conveyed in the PCC rules, in particular a QoS Class Identifier (QCI), an Allocation and Retention Priority (ARP) and authorised Guaranteed and Maximum Bit Rate values for uplink and downlink [28]. The QCI is a scalar that represents the QoS characteristics that the EPC is expected to provide for each service data flow. This value is used by transport plane devices to access and configure device specific parameters that control packet forwarding treatment.

The interaction between the PCRF and the transport plane has been extended; the PCRF interacts and enforces PCC rules across a greater number of access technologies and QoS models. The PCEF, as the element residing in the transport plane, is separated into the Packet Data Network (PDN) Gateway, the Serving Gateway and the Access Gateway. The Serving Gateway is a router that resides on the local network to which the end-user is attached, it performs connectivity provisioning including access control and resource provisioning for end-users attaching via 3GPP access systems (e.g. GSM/EDGE Radio Access Network (RAN), UMTS Terrestrial RAN (UTRAN) or LTE-RAN). The PDN Gateway has similar functionality but is located in the home network of the end-user. The Access Gateway authenticates end-users connecting via non-3GPP access networks and monitors traffic. PCC rules are received by these logical elements and used to configure the transport plane devices accordingly. An additional logical element co-located with the transport functions, the Event Reporting Function (ERF), is defined to provide a feedback mechanism that monitors flow conditions through pre-defined

transport plane events. This QoS reporting can be used to generate charging parameters, calculate usage statistics and detect erroneous behaviour in the transport plane.

Fig. 2.1 shows the evolved PCC architecture and how it interacts with the Services Architecture Evolution and in particular the EPC. The EPC supports multiple accesses and mobility protocols. To cater for the mobile IP family of protocols a Bearer-Binding and Event-Reporting Function (BBERF) is introduced. The BBERF location is specific to the particular access technology and its function is similar to the PCEF, it is omitted for simplicity.

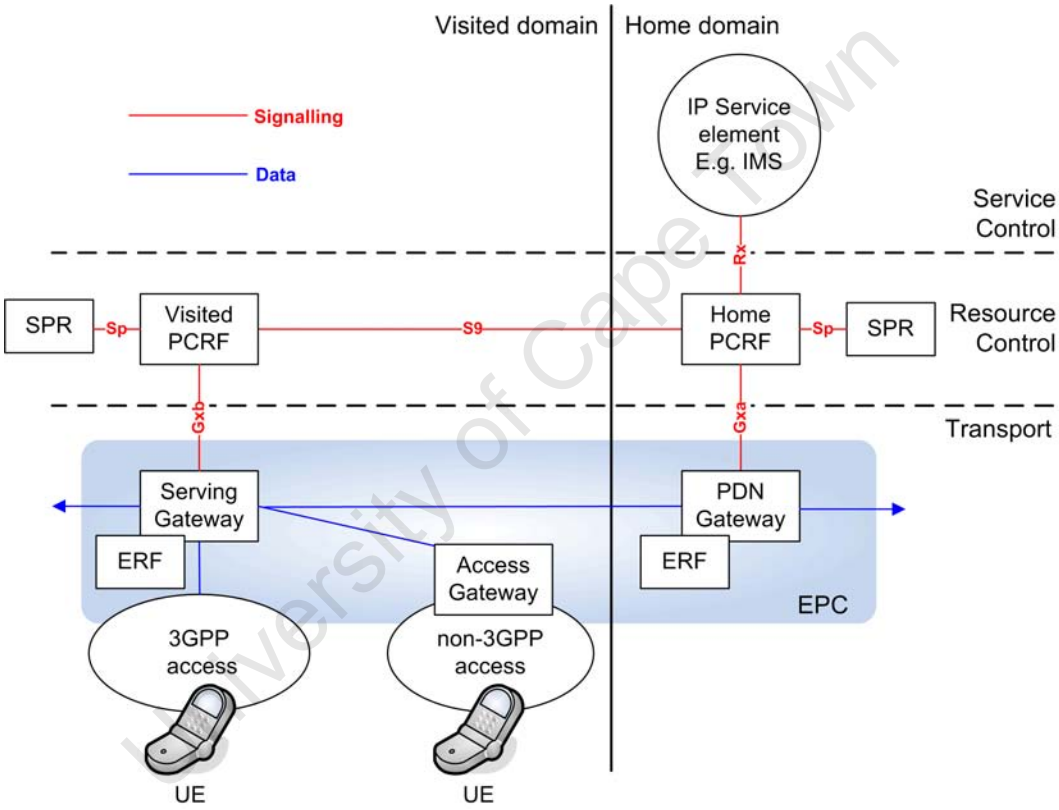

Figure 2.1: The 3GPP Release 8 PCC has extended scope and interacts with the EPC.

2.1.2 Reference Point Definitions

3GPP defines reference points that describe functional requirements and in depth protocol specifications for the associated interface, to provide interaction between the aforementioned logical elements. A reference point is a conceptual point at the conjunction of two functional groups, while an interface is a common boundary between associated

groups; essentially a reference point separates logical entities, while an interface separates physical entities. In most cases the terms are used interchangeably.

Release 7 defines the Rx reference point as method of interaction between the service control and resource control planes. This interface extends the Diameter base protocol and defines new commands and Attribute Value Pairs (AVP) to facilitate interaction between the AF and the PCRF [29]. This interface carries authorisation requests for new or existing sessions and feedback reports triggered by transport plane events. Release 8 extends this interface by allowing the PCRF to interact with a number of IP Service elements.

To provision PCC rules and install them on the logical elements in the transport plane, 3GPP defines the Gx reference point, also an extension of the Diameter base protocol [30]. Additional Diameter commands and AVPs allow a PCRF to provision and install PCC rules on PCEFs through Diameter resource requests. Preliminary support for PacketCable and integrated WLAN was included in the protocol definition, Release 8 extends this reference point to be access agnostic [28].

With the PCRF split into home and visited functionality as of Release 8, the S9 interface has been introduced to support inter-domain communication between PCRFs in neighbouring domains [31]. This reference point is Diameter based and in the early stages of development, it allows a PCRF to request resources in a neighbouring domain and supports two basic roaming scenarios: home routed access and visited access [32]. In home routed access roaming, the transport functions are controlled by the home operator. For certain scenarios, home routed access is impractical, for example if the visited and home networks are geographically distant. In these cases visited access roaming or local breakout takes place, where a connection is established directly through a PDN Gateway in the visited network. However, these scenarios do not support end-to-end resource reservation across multiple domains, nor do they link the service control inter-domain routes with the routes followed by the media in the transport plane.

The Sp reference point lies between the SPR and the PCRF and it allows a PCRF to request subscription information related to an authorisation request [26]. While the functional requirements of this interface are defined, the interface specification is deemed as operator specific.

2.2 TISPAN Resource and Admission Control Subsystem

TISPAN Release 1 specifications were finalised in December 2005, and provided a robust and open set of standards necessary for the development, implementation and testing of the first NGN architectures. The Resource and Admission Control Subsystem (RACS) was included as part of these specifications; largely based on the early 3GPP PCC framework, the RACS specifications provided only high level functional requirements. Release 2 was finalised in early 2008 and included in depth control scenarios and protocol specifications; in particular the scope of the RACS was extended to access and core networks, as well as to points of interconnection between networks in order to support end-to-end QoS provisioning. In the current specification end-to-end QoS handling is limited to wholesale and basic roaming between two domains only.

Like the PCC architecture, TISPAN divides the NGN into service control, resource control and transport planes and, through the RACS, offers a set of generic policy based transport control services to applications. The RACS supports transport plane resource reservation for both session based and non-session based applications. In this analysis the RACS as specified in TISPAN Release 2 is presented.

2.2.1 Functional Elements

The Release 2 specified RACS consists of two critical components: the Service-Based Policy Decision Function (SPDF) and the generic Resource and Admission Control Function (x-RACF) [33]. These elements interact with an AF in the service control plane, and the transport processing functions in the transport plane. As with the PCC architecture, the AF communicates with the RACS and requests resource authorisation by providing dynamic QoS-related service information. Unlike the PCC architecture, this element can request resource authorisation for session and non-session based applications.

The role of the IETF PBNM defined PDP is split in the RACS architecture; the SPDF acts as a final policy decision point for the administrative domain, while the x-RACF acts as a local policy decision point regarding subscriber access admission control and resource handling control. The SPDF offers a single point of contact for AFs and neighbouring SPDFs for requesting resource authorisation and thus hides the underlying network from requesting entities. Upon receiving authorisation requests, the SPDF applies operator

specific policies to perform admission control. If the request is authorised, the SPDF checks resource availability by querying the x-RACF. As of Release 2, the x-RACF has two functional specialisations, the Access-RACF (A-RACF) and the Core-RACF (C-RACF). The A-RACF retrieves the authenticating users' QoS profile from the Network Attachment Subsystem (NASS) and authorises the request. This element is deployed in the access network domain where network resources may be provisioned on a per-subscriber basis. The C-RACF is deployed in the core network and allocates network resources, but not on a per-subscriber basis.

The transport processing functions are divided into the Resource Control Enforcement Function (RCEF), the Border Gateway Function (BGF) and the Basic Transport Function (BTF). The BTF consists of elementary forwarding functions and elementary control functions; essentially this element represents the functional attributes of the physical network elements. The BGF is a gateway between different IP transport domains and is under the control of the SPDF; it provides capabilities such as Network Address Translation (NAT), gate control, marking of outgoing flows, policing of incoming traffic, topology hiding, IPv6/IPv4 interconnection, usage metering and resource allocation. The RCEF exists in the access network domain, or IP edge nodes and is under the control of the x-RACF. The RACS framework supports both push and pull mode resource reservation mechanisms; in pull mode operation the RCEF requests and installs policy decisions from the x-RACF based on requests from BTFs, while in push mode operation, traffic policies are pushed directly from the x-RACF for installation. RACS provides mechanisms for guaranteed and relative QoS. Relative QoS is achieved through packet marking and traffic class differentiation at IP edge and core nodes; while guaranteed QoS defines absolute bounds on QoS parameters and is achieved via tight traffic control and policing.

An important functional requirement for the RACS is an architecture for resource monitoring and QoS reporting. QoS reporting is the ability of a network element to gather QoS metrics related to a single service instance, while resource monitoring is the ability to monitor topologies and transport segments under RACS control [34]. A Topology and Resource Information Specification (TRIS) should be maintained to hold information that includes, but is not limited to, physical topology, logical topology, routing information, resource information and selection information [35]. The TRIS should be populated and managed by the SPDF and x-RACF while authorising resources. QoS reporting should be supported by all transport processing functions, information should

be collected for each service instance and interfaces between QoS reporting collectors and QoS reporting users (e.g. RACS) should be implemented. Detailed logical architectures for these functional requirements are not specified. Fig. 2.2 shows the functional elements and logical relationships in the TISPAN Release 2 RACS.

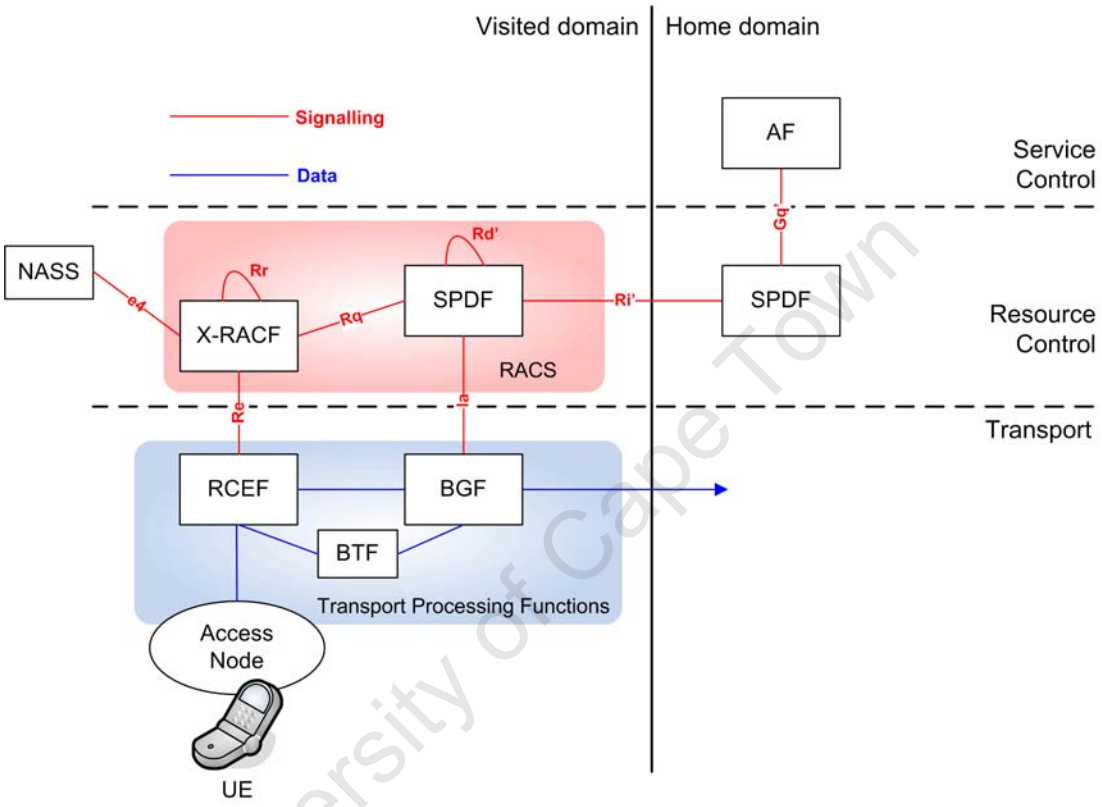

Figure 2.2: The TISPAN Release 2 RACS includes resource management and policy control entities that govern transport processing functions.

2.2.2 Reference Point Definitions

Release 2 defines the Gq' reference point between the AF and SPDF, this interface exchanges session based policy information between the service control and resource control planes. The reference point to support non-session based resource authorisation is not defined. The Gq' interface is based on the 3GPP Release 6 Gq Diameter application; a specific Diameter application is defined that instantiates new commands and AVPs [36].

The SPDF queries the x-RACF via the Diameter based Rq reference point. Via this interface the SPDF issues resource requests in the core and access networks to the

x-RACF, indicating IP QoS characteristics; the x-RACF uses the IP QoS information to perform admission control and indicates its admission control decisions to the SPDF [37]. Depending on possible business roles, the Rq interface can be an intra- or inter-domain interface.

The e4 reference point is instantiated to allow the x-RACF to query the Network Attachment Subsystem (NASS) for user QoS profiles; this Diameter based interface, together with the NASS, plays a similar role to the 3GPP defined SPR and Sp interface. The SPDF installs policy decisions and configures the BGF in the transport plane via the Ia reference point; this interface defines a profile of the Gateway Control Protocol (H.248) to control the various capabilities of the BGF [38]. The Re reference point lies between the x-RACF and the RCEF; this Diameter based interface allows policy rules to be requested by, or pushed to, the RCEF. The interactions between the BTF, the RCEF and BGF are not specified; these interfaces are considered to be internal relationships within the same physical node.

The RACS end-to-end QoS support allows for basic roaming scenarios and the Ri' reference point is implemented for inter-domain communication between SPDFs. Additionally an intra-domain reference point, the Rd' interface, allows SPDFs in the same administrative domain to communicate with, and issue resource requests to, one another. TISPAN has released a draft version of the Resource Connection Initiation Protocol (RCIP) to facilitate inter-domain communication via the Ri' interface, but this is an ongoing standardisation work [39]. The Rr reference point is defined between x-RACFs of the same administrative domain and allows one x-RACF to delegate resource admission control responsibility to another x-RACF instance; the protocol for this interface is not specified.

2.3 ITU-T Resource and Admission Control Functions

The ITU-T, as an inter-government, public-private partnership, promotes global convergence of, and consensus on, technologies and services. In 2004 it began developing the Resource and Admission Control Function (RACF) based largely on the early work of 3GPP and TISPAN. The ITU-T QoS control architecture defines a high level reference framework and covers the broad aspect of extending the region of control to core and

access networks, and defines additional control scenarios. Subsequent PCC and RACS developments, described earlier, incorporate the extended scope and additional control scenarios.

The ITU-T define a service stratum and transport stratum; the RACF is logically situated in the transport stratum and provides interaction between the service control functions and the transport functions It determines the availability of resources and appropriately controls network elements.

2.3.1 Functional Elements

The Service Control Function (SCF) is responsible for the application signalling for the service setup and logically resides in the service stratum. The SCF derives the QoS needs of the requested service and sends them to the RACF in the transport stratum for authorisation. Service information is extracted from application signalling for session and non-session based services, and is used to create authorisation requests. The RACF has two functional entities: The Policy Decision-Functional Entity (PD-FE) and the Transport Resource Control-Functional Entity (TRC-FE) [18]. The PD-FE is the single contact point between any SCF and the transport stratum, this element performs authorisation, reservation and commitment of network resources. The TRC-FE monitors the network topology and the resource state of the network; it performs technology-dependent admission control on behalf of the PD-FE. Essentially the PD-FE is responsible for the transport technology independent aspect, while the TRC-FE is responsible for the transport technology dependent aspect.

Authorisation decisions taken at the PD-FE are subject to operator specific policies and are based on service information, subscription information and transport network information. Service information is received from the SCF in authorisation requests; transport subscription information is retrieved via the subscription profile from the Network Attachment Control Function (NACF); and transport network information is collected from the TRC-FE.

The transport functions are divided into the Transport Resource Enforcement-Functional Entity (TRE-FE) and the Policy Enforcement-Functional Entity (PE-FE). The TRE-FE is dynamically controlled by the TRC-FE to perform polling of network usage, bandwidth reservation and allocation and traffic shaping. In this way the TRC-FE carries out QoS reporting and resource monitoring and can provide transport network infor-

mation to the PD-FE. TRC-FE implementations are described for different transport technologies, including IP, MPLS, Ethernet and broadband wireless. In each case the TRC-FE maintains a Network Topology and Resource Database (NTRD) based on information provided by the TRE-FE. The TRE-FE also enforces policy rules received from the TRC-FE at the technology-dependent aggregate level.

Once the request is authorised, resources need to be reserved in the transport stratum. The PD-FE pushes service definitions in the form of policy rules to the PE-FE, located at the edge or border of an administrative domain in the core or access network. These elements provide capabilities including NAT traversal, bandwidth allocation, gate control, QoS marking and rate limiting. The RACF architecture supports both push and pull mode operation, and thus caters for a range of Customer Premises Equipment (CPE) capabilities. Once all resource control decisions are enforced on the transport functions, the gate is opened and resources are committed. The authorisation, reservation and commitment phases constitutes QoS resource control, and in the RACF architecture these phases can be performed in separate or combined steps to cater for the variety of applications and performance requirements.

The RACF specifies two end-to-end QoS control scenarios. In the first scenario QoS requirements for a given service can be passed over the end-to-end path through the application signalling or via an inter-domain reference point. In the second scenario QoS requirements are passed over the end-to-end path through path-coupled QoS signalling, such as NSLP. Detailed logical architectures for these high level functional requirements are not specified. The functional architecture of the RACF is presented in Fig. 2.3.

2.3.2 Reference Point Definitions

The Rs reference point is instantiated to allow QoS resource request information, needed for QoS control, to be exchanged between the SCF and the PD-FE in the same or different domains. The Diameter protocol is specified for this interface and it allows the SCF to make requests for resource authorisation and reservation, for QoS handling, for gate control of transport functions, for resource usage information, and for notification of transport stratum events.

The Ru reference point allows the PD-FE to query the NACF for subscription information. This interface allows the PD-FE to retrieve access network specific profile information and user subscription information, and incorporate it into the authorisation

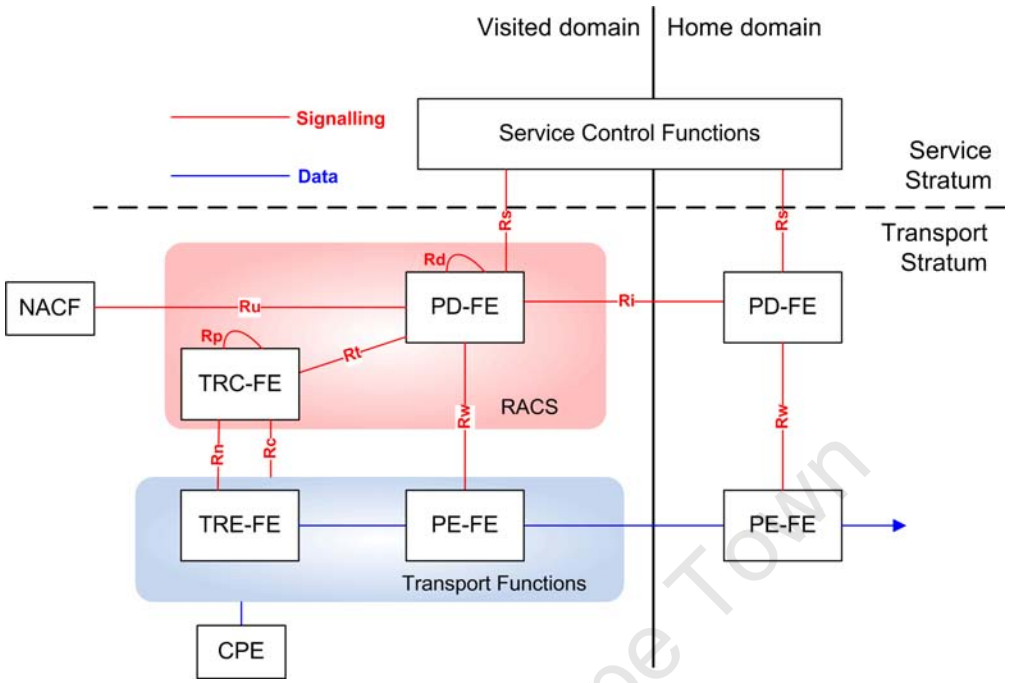

Figure 2.3: The RACF arbitrates between the service stratum and the transport functions.

decision. This interface is based on the Diameter protocol.

The PD-FE interacts with the TRC-FE via the Rt reference point. This interface allows the PD-FE to determine via the TRC-FE whether or not the requested resources are available for a given media flow, and to request relevant TRC-FEs to detect and monitor the usage of a particular media flow. The interface is based on Diameter and the definition is very similar to that of the Rs interface between the SCF and the PD-FE.

To collect the network topology and resource status information to populate the NTRD, the Rc reference point is instantiated between the TRC-FE and the transport functions. The Rn reference point between the TRC-FE and TRE-FE, carries policy decisions that are enforced at the TRE-FE at the technology-dependent aggregate level. The details of these reference points are for further study. The PE-FE is the key injection node to enforce dynamic QoS rules in the transport stratum, the Rw reference point allows the PD-FE to install the final decisions, either by push or pull mode, to the PE-FE. This interface should support both fixed and mobile access networks and lies between elements in the same administrative domain. The protocol definition for this interface is unspecified [20].

In an operators core network there may be multiple TRC-FE instances to control

multiple sub-domains. The Rp reference point, defined between TRC-FE instances in the same domain, enables the PD-FE to contact a single TRC-FE instance, but detect and determine the requested resource in all sub-domains within an administrative domain. The protocol definition of this interface is Diameter based and similar to that of the Rt interface between the PD-FE and TRC-FE.

In the case of multiple PD-FEs in a single domain, the Rd reference point provides intra-domain communication between PD-FEs; this ensures that the Rs reference point provides a single point of contact for an SCF to request resource authorisation, and allows for network scalability. The Ri reference point facilitates inter-domain communication between PD-FEs, and is used when an SCF is not capable of interacting with the PD-FEs in each domain traversed by the media flow. This interface could be used to request resource authorisation and admission control over a third party transit domain. The details of these reference points, and the format and content of the information conveyed, are for further study.

2.4 Generic Resource Management Framework

It is clear that common attributes exist between the developed frameworks; the replication of functional elements and reference points poses a challenge to the creation of a harmonised resource management framework. It is important that the same harmonisation that resulted in the single set of Common IMS specifications takes place in the resource management sphere.

The harmonisation of the RACS Gq' reference point and the PCC Rx reference point is an ongoing joint initiative between TISPAN and 3GPP [40], and is a proposed work item under 3GPP Release 9 specifications. This work item examines the impact such an aligned IMS-facing reference point would have on architecture and protocol aspects. The overall harmonisation of resource management frameworks is not investigated.

In this section we examine common functional elements and reference points in the aforementioned resource management frameworks, and propose a generic architecture, dubbed Common PCC. The Common PCC framework adopts harmonised functional elements and reference points. 3GPP specifications provide the most comprehensive reference point definitions, consequently they form the basis for most reference points in the Common PCC architecture. The broader functional requirements specified by TISPAN and the ITU-T are incorporated, and element and reference names are adopted

from the standardisation body that defines the most in depth specifications for the element or reference point. Because the Common PCC framework is generic and not restricted to fixed or mobile technologies, QoS mechanisms designed for this framework can be mapped on to the 3GPP/TISPAN/ITU-T architectures.

2.4.1 Architectural Alignment

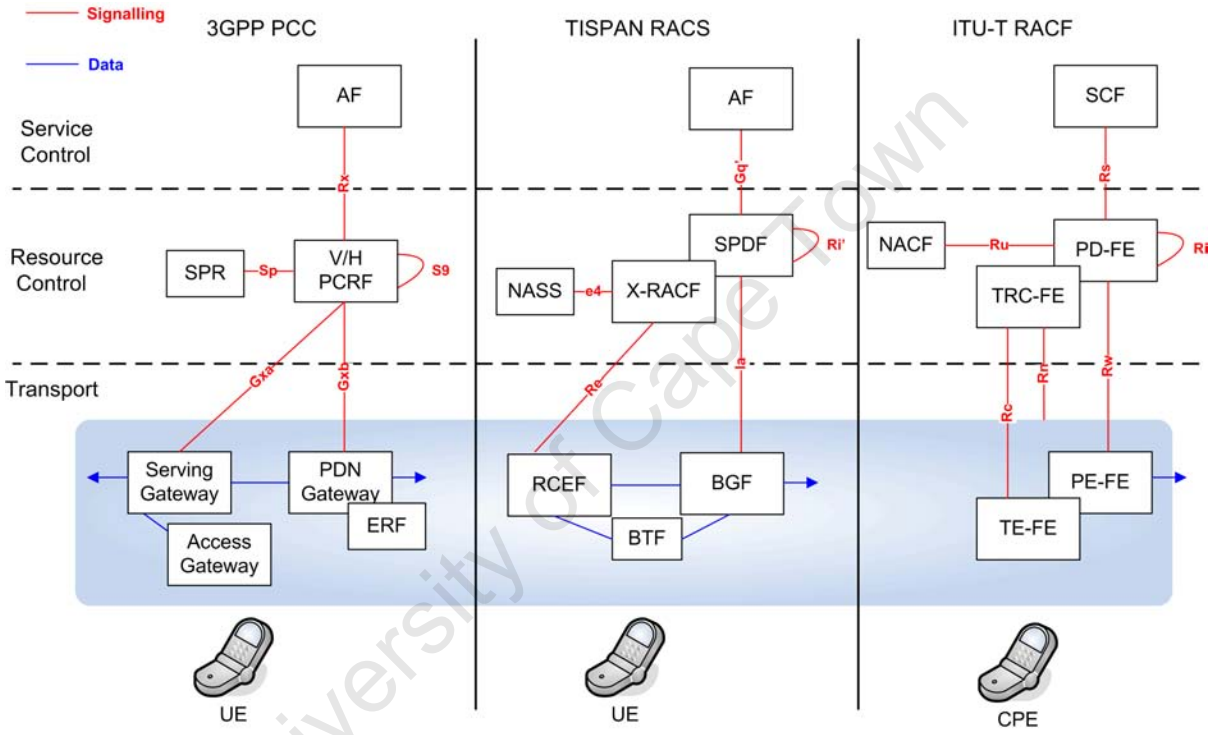

Figure 2.4: Common attributes exist between the PCC, RACS and RACF.

Referring to Fig. 2.4, the separation of application, resource and transport control planes is consistent in all NGN resource management frameworks, similarly an element in the service control plane should be able to intercept signalling and request resource authorisation from the resource control plane. In the scope of this thesis such an element is limited to requesting resources on behalf of session based IMS services, but the Common PCC framework should support session based and non-session based applications. The interaction between the service control and resource control planes for session based services, by the Rx, Gq' and Rs reference points respectively, is based on the Diameter protocol for all examined architectures. The harmonisation of these Diameter applica-

tions is an important work item under study in 3GPP Release 9, and it is likely that a harmonised interface will be similar to the 3GPP Rx interface, as this is the most extensively defined.

When considering multiple transport technology deployments and the interconnection of administrative domains, the division of the policy decision element into two functions, as done by the RACS and RACF, is justified. This supports network scalability while also providing a further layer of abstraction; a primary element acts as the single point of entry for resource requests and as the final policy decision point for the administrative domain, while a secondary element performs technology dependent admission control, and monitors and detects the usage of individual media flows.

The reference to operator specific policies is common in all architectures, and in the broad sense the 3GPP SPR, TISPAN NASS and ITU-T NACF have similar functions. It is clear that a Policy Repository is necessary, that includes, among other operator specific policies, subscription and QoS specific profile information. The RACS e4 reference point between the NASS and x-RACF specifies Diameter for the exchange of subscription information, but this interface could utilise the Lightweight Directory Access Protocol (LDAP) or any operator specific protocol. Policies will control other aspects of the network, and it is expected that more than one protocol will be supported for this access to cater for the wide range of application scenarios.

The QoS resource control performed at the PDP element, though described using different nomenclature, is essentially identical for each of the architectures; it consists of service based admission control or authorisation, resource based admission control or reservation, and enforcement of reserved resources or commitment. Additionally the information available to carry out the authorisation is similar for all three architectures and includes service specific information, subscription specific information, and transport network specific information.

The transport plane elements are diverse in the architectures; the 3GPP PCC caters for various different accesses including 3GPP IP access, trusted non-3GPP IP access and untrusted non-3GPP IP access, while the RACS and RACF are representative of a more generic transport plane encompassing both fixed and mobile transports. A border gateway element that provides connectivity capabilities is essential in all architectures, and by housing a gateway in the home and visited domain data breakthrough can be supported in both domains. A specialised transport plane element in the access or IP edge nodes is also desirable; these elements enforce policy decisions at the technology

dependent aggregate level. The interaction between the resource control and transport plane elements, by the Gx, Re/Ia and Rw/Rc reference points, encompasses more than one protocol. The RACS utilises H.248 for controlling the BGF, and Diameter for controlling the RCEF. 3GPP specify an extensive Diameter application for the Gx reference point; much work has gone into making this reference point access agnostic. Although it is likely that some reference points will be expected to support more than one protocol, the Gx Diameter application is ideal for this interaction because of its extensive definition. There are several candidate protocols to collect network topology and resource status information, including the Common Open Policy Service (COPS) protocol, the Simple Network Management Protocol (SNMP) and Simple Object Access Protocol (SOAP). None of these are specified by any of the standardisation bodies, and hence this interaction is unspecified in the Common PCC framework, though there is support for transport plane event detection via ERFs in transport plane elements.

End-to-end QoS support is elementary in all of the architectures. In the RACF architecture the inter-domain reference point is for further study, the RACS is currently standardising RCIP for this interface, while the 3GPP S9 reference point defines a Diameter application for inter-domain resource authorisation. High level requirements for the intra-domain reference points between resource control plane elements are specified by the RACS and RACF architectures, but the details of these interfaces are left for further study. While these reference points are included in the Common PCC framework, the protocols for this interaction are unspecified.

2.4.2 Common PCC Framework

The Common PCC framework as defined in this thesis includes the following functional elements:

- An AF in the service control plane that extracts QoS information from application signalling to create authorisation requests.
- A Policy Decision Function (PDF) that acts as a single point of entry for authorisation requests from AFs and neighbouring PDFs. The PDF is split into home and visited functions.
- An x-RACF to perform technology dependent admission control and maintain a

Technology and Resource Information Specification (TRIS) by monitoring transport functions.

- A Policy Repository that contains, among other operator specific policies, subscription and QoS profile specific policies.
- A Visited-Border Gateway Function (BGF) in the visited domain and home-BGF in the home domain, to provide connectivity capabilities.
- A RCEF in the access or IP edge node to enforce policies at the technology dependent aggregate level.
- Visited and Home BGFs, and RCEFs that support detection of transport plane events via ERFs housed in these transport plane elements.

The Common PCC framework includes the following reference points:

- The 3GPP defined Diameter based Rx interface for interaction between the AF and the PDF, incorporating AVPs and messages defined by the Gq' and Rs interface Diameter applications.
- The TISPAN defined Diameter based Rq interface for interaction between the PDF and x-RACF.
- The 3GPP defined Diameter based Gxa interface for interaction between the PDF and BGF, and Gxb for interaction between the x-RACF and RCEF.
- The unspecified Rc interface between the x-RACF and transport plane functions, to collect network topology and resource information to populate the TRIS.
- The unspecified, operator specific Sp interface for interaction between the PDF and the Policy Repository.
- The 3GPP defined Diameter based S9 interface for inter-domain interaction between neighbouring PDFs.
- The unspecified intra-domain interfaces Rd and Rr, for interaction between PDFs and x-RACFs respectively.

The Common PCC framework supports:

- Push and pull mode operation.
- Session binding based on user-identity based identification.
- QoS resource control at the PDF, including authorisation, reservation and commitment of resources.
- Service level QoS parameters conveyed in PCC rules, including a QCI, an ARP and authorised Guaranteed and Maximum Bit Rate values for uplink and downlink.

Fig. 2.5 shows the Common PCC logical architecture, the defined terms and functional elements are used throughout the remainder of this thesis. These definitions will allow for more coherent and focused future research.

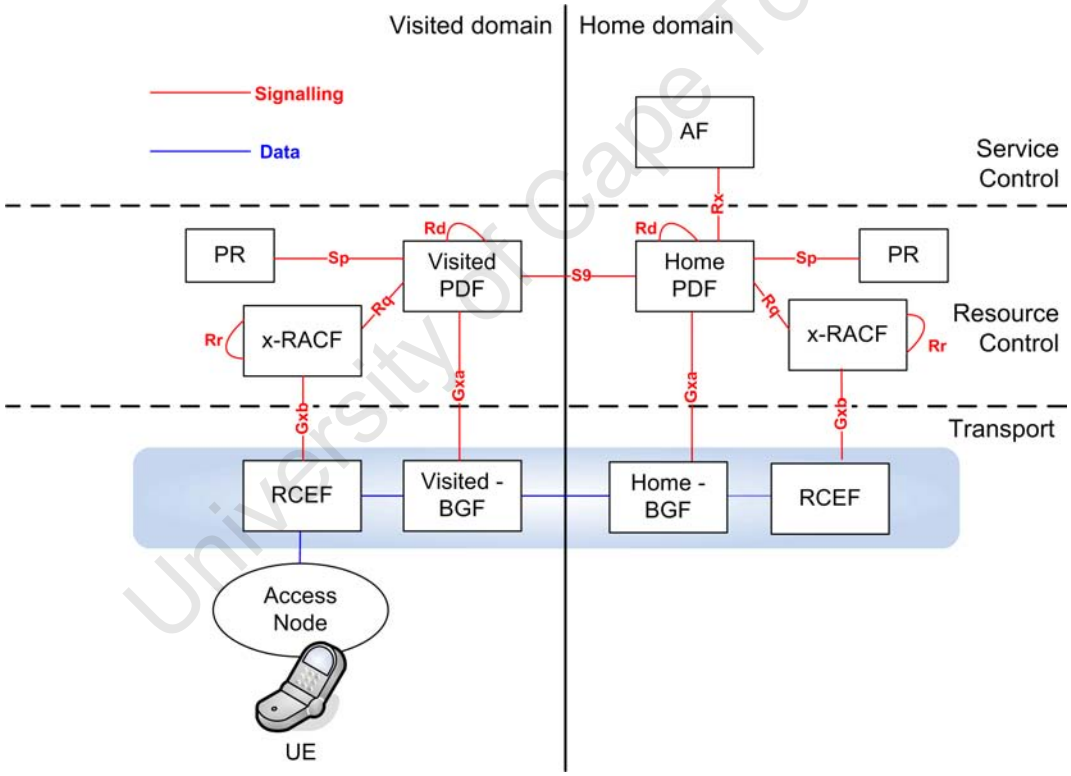

Figure 2.5: The Common PCC Framework encompasses work done by all standardisation bodies and defines a generic set of terms and functional elements.

2.5 Discussion

The chapter has outlined the predominant architectures for policy based resource management currently under standardisation. A Common PCC framework is defined that encompasses the functional requirements specified by the standardised frameworks, but addresses these requirements using detailed functional definitions. This common architecture combines duplicate elements and reference points to create a set of functional terms and elements to be used throughout the remainder of this thesis. Shortcomings in the standardisation work have also been highlighted. In particular the authorisation of resource requests and enforcement based on operator policies are not described in detail, and the translation of complex, highly interactive, multimedia rich services into efficient, aggregated QoS resource requests is still under investigation by the relevant standardisation bodies. The end-to-end QoS mechanisms specified are elementary; QoS connectivity across all traversed transport segments is not supported in any of the architectures and inter-domain reference point definitions are for further study or preliminary at best. This thesis places emphasis on these shortcomings when reviewing related literature and developing an architecture to achieve end-to-end, application driven policy control for IMS services.

Chapter 3

Literature Review

The previous chapter examined the resource management frameworks currently under standardisation and presented the Common PCC framework. This chapter reviews the state of the art in application driven policy control for IMS/NGN architectures, and the provisioning of end-to-end QoS enabled paths. Rapid time to market to deploy new IMS services has long been identified as a critical requirement for the Common PCC architecture [16], but the state of research regarding tailoring the policy control framework to ensure rapid and innovative service creation, without the need for prior standardisation of QoS support, is relatively immature. However there have been advances in the field and proposed solutions do exist.

The provisioning of end-to-end QoS routes in IP networks has been an important area of research for some time, largely due to the Voice over IP (VoIP) revolution. Additionally a number of projects focusing on the future Internet propose a range of solutions. Many of these do not specifically focus on the NGN or IMS architectures. However because IMS standards incorporate a number of IETF protocols designed for the Internet, these solutions can be adapted and applied. When extending the Common PCC framework to facilitate innovative and rapid development of IMS services, and inter-domain policy control, the work of those who have examined these problems in the Internet is pertinent.

Much of this work has emanated from the IETF in the form of Internet Drafts and IETF Request for Comments (RFC). Internet Drafts are temporary documents that expire 6 months after they are issued; once these drafts have sufficiently matured they are published as IETF RFCs. While under development, Internet Drafts can be revoked or changed at any time and do not take into account backward compatibility. For this

reason it is recommended that the latest version of any cited Internet Draft be examined, as the contents may have changed since December 2008.

The chapter begins by reviewing literature related specifically to the vertical coordination of resources in the IMS and the Common PCC framework. This includes work that addresses the shortcomings of the Common PCC architecture regarding innovative and rapid application deployment, as well as the complexities of policy refinement and the translation of highly interactive, multimedia-rich services into efficient, aggregated QoS resource requests. The chapter concludes with an examination of horizontal coordination of resources or inter-domain resource provisioning in the IMS. Attention is paid to future Internet projects that address QoS routing and the problem of finding a feasible path between a source and destination node satisfying one or more QoS constraints.

3.1 Vertical Coordination of Resources

Hilt, Camarillo and Rosenberg were the first to identify the shortcomings of policy-based resource management when applied to SIP sessions in October 2006 [41]. The authors point out the fact that a major design principle of SIP was to enable the creation of end-to-end services without requiring the upgrading of network elements between endpoints. The work suggests that the adopted IMS policy framework, whereby sessions are described using the Session Description Protocol (SDP), prohibits the innovative creation of new services. If any service requires a new SDP extension to describe itself, or the use of a separate description format, it would be necessary to upgrade all SIP proxy and policy control elements in the network, thus breaking a major SIP design principle. This document has led to the publication of several other IETF Internet Drafts, all of which define a model, an overall architecture and new protocol mechanisms to implement a concept known as Session Policies.

3.1.1 A Framework for Session Policies

In the Internet Draft *A Framework for Session Initiation Protocol (SIP) Session Policies* [42], Hilt *et al.* present standardised mechanisms by which SIP proxy servers can define or influence policies on sessions. The main problem identified with SIP policy control is that end-users are not always informed as to why session requests are rejected, nor are they told how to establish new sessions that will be accepted. Spurious behaviour

can also occur where an end-user may receive a successful session establishment indication, immediately followed by a session termination indication. Additionally the policy mechanisms assume that SIP proxy servers have access to the SDP bodies of the SIP messages. This means that end-to-end encryption mechanisms like Secure / Multipurpose Internet Mail Extensions (S/MIME) are not supported, and as previously mentioned, end-users must use SDP as their session description format. SIP proxy servers also modify SDP bodies to convey QoS information to end-users; this practice can lead to unexpected interactions if SDP extensions are not supported. This makes it difficult for developers to debug their implementations because of unpredictable behaviour, and inhibits end-to-end integrity protection mechanisms.

This Internet Draft describes extensions that allow proxy servers to communicate different policies to the end-users without accessing end-to-end bodies such as session descriptions. A policy server is defined that delivers session policies to the end-user; the policy server can query an external entity for policy retrieval or it can directly incorporate policy decision point functionality and locally generate policies. Two kinds of policies are defined: session-independent policies and session-specific policies. Session-independent policies are created independently of a session and generally apply to all initiated sessions; typically these policies are stable and sustained over long periods though they can change over time. Session-specific policies are created for a particular initiated session, and are typically different for different sessions; these policies exist for the duration of the session and can change over time.

Because session-independent policies are applicable to all initiated sessions, they are downloaded from the policy server whenever an end-user connects or attaches to a new domain, or when the policies on the policy server change. On the other hand, session-specific policies need to be downloaded each time an end-user initiates a session. By having access to these policies before initiating a session, an end-user will be more informed as to why their session was rejected, it also removes spurious behaviour inherent in SIP policy control when rejecting sessions.

Protocol mechanisms are defined that allow end-users to request and retrieve policies from the policy servers. In a typical scenario an end-user will request session-independent policies from the policy server in the local network and home network domains; the end-users request session-independent policies using SUBSCRIBE requests, and the policy server selects the policies that apply to that end-user and returns them in the body of a NOTIFY request. Because session-independent policies are a form of configuration

information, the event package for user agent profiles defined by D. Petrie *et al.* [43] is used to transfer these policies.

End-users also use SUBSCRIBE requests to obtain session-specific policies. When an end-user initiates a session the SIP proxy server responds to the initial INVITE request with a 488 (NOT ACCEPTABLE HERE) response that encapsulates the Uniform Resource Identifier (URI) of the proxy server; a new SIP header field, the *Policy-Contact* header, is defined to carry the URI. Using this URI, the end-user sends a SUBSCRIBE request to the policy server. This request uses the event package for session-specific policies defined by Hilt *et al.* [44], and includes details of the session to be established, this information is in the form of an XML document in the body of the SUBSCRIBE request.

The policy server processes the SUBSCRIBE request, and based on the session description, returns its policies in a NOTIFY request. These policies can accept the session, reject the session or propose changes to the session parameters that would deem the request acceptable. Once the session is accepted, the end-user re-initiates the session with an INVITE request. A new SIP header field is defined, the *Policy-Id* header; this header is included in the new INVITE request to indicate that a policy server has been contacted, and the policies received have been accepted.

The Media Policy Dataset Format (MPDF) defined in the Internet Draft *A User Agent Profile Data Set for Media Policy* [45] proposes a document format to describe the media properties of SIP sessions. The document format is based on XML and extends the Schema for SIP User Agent Profile Data Sets. This format can be used in two ways, either to describe the properties of a given SIP session in the form of *session-info* documents, or to define policies for SIP sessions in *session-policy* documents. Session info documents are used in conjunction with session-specific policies, this format defines the XML document in the body of the SUBSCRIBE request sent from the end-user to the policy server. The Internet Draft specifies mapping between SDP session description information and MPDF session info documents. *Session policy* documents define the format for session-independent policies.

The authors argue that this XML-based representation is more flexible and extensible than the SDP format, and also removes the need for proxy servers on the signalling path to be able to interpret the SDP bodies of the SIP messages. In this way the session policies framework allows for innovative application development, as when a new service requires extensions to the session description format, an upgrade of all SIP proxy servers in the network will not be necessary. Fig. 3.1 shows a typical *session policy* document

that allows the use of audio and video but prohibits other media, all codecs are allowed except G.723 and G.729. Fig. 3.2 shows a typical session-info document describing an audio and video SIP session using standard codecs.

```
<property-set>
  <session-policy>
    <context>
      <domain>example.com</domain>
      <contact>sip:policy_manager@example.com</contact>
      <info>Access network policies</info>
    </context>
    <media-types excluded-policy="disallow">
      <media-type policy="allow">audio</media-type>
      <media-type policy="allow">video</media-type>
    </media-types>
    <codecs excluded-policy="allow">
      <codec policy="disallow">
        <mime-type>audio/G729</mime-type>
      </codec>
      <codec policy="disallow">
        <mime-type>audio/G723</mime-type>
      </codec>
    </codecs>
  </session-policy>
</property-set>
```

Figure 3.1: An MPDF *session-policy* document allows video and audio, and rejects codecs G.723 and G.729.

It is important to note that once the URI of the policy server is retrieved, the end-user and policy server communicate directly over a dedicated policy channel. This essentially decouples the signalling between endpoints and the policy exchange between an endpoint and a policy server. This decoupling means that separate encryption mechanisms can be used on the signalling path and the policy channel, and policies can be submitted directly from the policy server to the endpoint without traversing the signalling path. The disadvantage is the increase in signalling necessary for the exchange of policies. Fig. 3.3 demonstrates typical signalling in the session policies framework for session initiation with session-specific policies. The originating end-user retrieves the policy server URI from its proxy server (1-3), and contacts the policy server over the dedicated policy channel (4, 5). The session is described in the XML document included in the body of the SUBSCRIBE request, this document is based on the MPDF session info document. Once the session is accepted, a new INVITE request is created, this time with the *Policy-Id* header field instantiated (6). The INVITE request is forwarded to the terminating proxy server, where the *Policy-Id* header is replaced with a *Policy-Contact* header, indicating the URI of the terminating policy server (7). The terminating end-user confers with the policy server in the terminating domain (9,10), and once authorised sends a 200 OK

```
<property-set>
 <session-info>
  <context>
   <contact>sip:alice@somewhere.example</contact>
   <info>session information</info>
  </context>
  <streams>
   <stream>
    <media-type>audio</media-type>
    <codec><mime-type>audio/PCMU</mime-type></codec>
    <codec><mime-type>audio/1016</mime-type></codec>
    <codec><mime-type>audio/GSM</mime-type></codec>
    <local-host-port>host.somewhere.example:49562</local-host-port>
   </stream>
   <stream>
    <media-type>video</media-type>
    <codec><mime-type>video/H261</mime-type></codec>
    <codec><mime-type>video/H263</mime-type></codec>
    <local-host-port>host.somewhere.example:51234</local-host-port>
   </stream>
  </streams>
 </session-info>
</property-set>
```

Figure 3.2: An MPDF session info document describes an audio and video session using standard codecs.

response (11-13). The originating end-user again confers with the originating policy server (14, 15), this time including the session description from the terminating end-user, and eventually the session is initiated. The 200 OK messages sent in response to the SUBSCRIBE and NOTIFY requests are omitted from the diagram for simplicity.

3.1.2 Session Policies in IMS

Incorporating session policies into the Common PCC framework requires an additional logical function in the architecture: the policy server. The policy server has a Diameter-based interface to the PDF and a SIP-based interface to the User Equipment (UE). The interface between the PDF and policy server is not defined, but would be similar to the Rx interface between the Application Function (AF) and PDF [9]. IMS roaming users would need to consult policy servers in their home and visited networks before initiating a session, though these interactions could be performed in parallel to avoid additional delays.

Currently the IMS architecture does not provide a means for the network to inform end-users about policy changes that apply to ongoing sessions. With session policies, if the policies applicable to an end-user are modified, this event is immediately conveyed to the end-user in a NOTIFY request. For example in a mobile, multi-access environment,

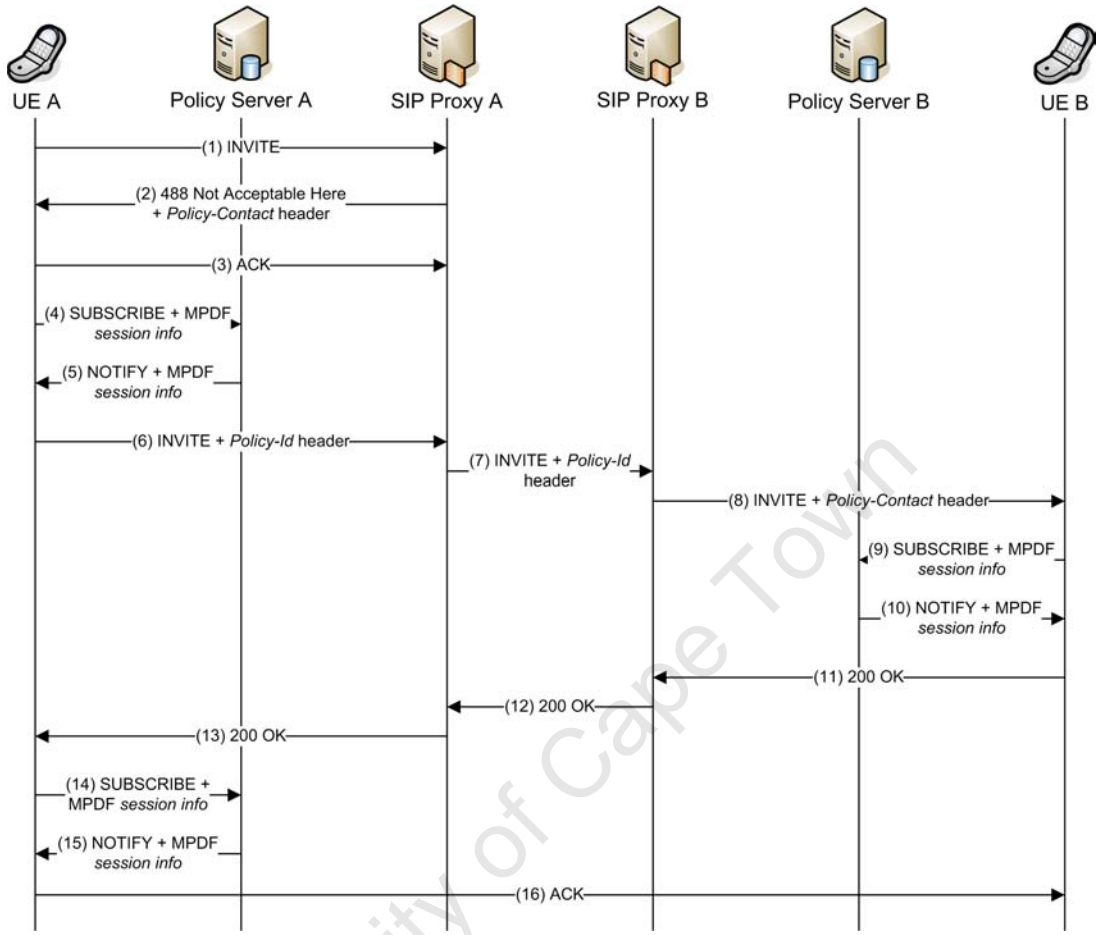

Figure 3.3: Session initiation with session-specific policies.

an end-user may attach to a mobile network that does not allow video streams. With session policies, when the end-user attaches to this new network, it is informed of the updated policies and can re-initiate an applicable session.

The main disadvantage of the session policies framework is the additional signalling required to initiate sessions. Limited effect on session setup delay has been identified as a critical requirement for the Common PCC framework [16], and incorporating session policies introduces additional round-trip times between the end-user and policy server for each initiated session. The authors argue that this is the price to pay for having policy control and end-to-end negotiations, and maintain that unaccepted sessions that would be terminated by the network, can now be established because of additional policy information available to the end-user. The framework also introduces complexity at the end-user, where the mapping between SDP session information and MPDF *session-info*

documents is performed. End-users need also process and store MPDF *session-policy* documents for a number of domains. While the framework ensures that SIP proxy servers need not be upgraded to deploy new services, a new service that requires extensions to the MPDF *session-info* document would still require policy server upgrades. The authors point out that these extensions are in the early stages of development and will not reach maturity for some time, and hence are not expected to be adopted by the IMS in the immediate future [6].

3.1.3 QoS for Advanced Multimedia Applications

Future IMS services are expected to be exceedingly multimedia rich and customised to meet end-user preferences and capabilities. The regular deployment of new services will result in a constantly changing network dynamic and the Common PCC framework will have to cater for complex and unpredictable QoS requirements.

Skorin-Kapov *et al.* describe enhancements to the standardised IMS QoS negotiation procedure necessary to address these dynamically changing QoS requirements [21]. The authors point out the need to incorporate end-user preferences, network constraints and service requirements into the QoS negotiation procedure. A networked virtual reality service is used to illustrate these requirements; end-user preferences would allow a user to set the relevance of different events such as timing constraints for displaying and interacting with the virtual service, audio and text chat. Network constraints require that the service adapt to dynamic changes in the network occurring during service execution. In terms of the virtual reality service if the available bandwidth is unexpectedly reduced (e.g. due to a wireless link) the desired action might be to drop audio chat or switch to a text-based chat to maintain the maximum quality of experience. Service requirements might require that the application be available in several customised versions, the virtual reality service could be offered as a low-cost version suitable for dial-up access, and a default version with attractive graphics. The authors argue that current standards lack techniques to address these issues in a comprehensive manner.

In the article *End-to-End Signalling for Future Multimedia Services in the NGN* [46], Skorin-Kapov *et al.* present a model for dynamic negotiation and adaptation of QoS requirements, which uses generic client and service profiles as a basis. A *client profile* specifies end-user terminal and access network constraints, and application related preferences; such preferences are set by the end-user, though a generic client profile is defined.

A *service profile* may specify different supported configurations of a service to address diverse end-user capabilities; such preferences are set by the application developer, though a generic service profile is defined.

To incorporate these concepts into the IMS architecture, the authors propose the addition of a QoS parameter matching and optimisation (QMO) Application Server (AS). The QMO AS examines the service and client profiles to determine feasible service parameters and suitable service versions, and to perform optimisation. As with every other AS, the decision whether or not to involve the QMO AS for a particular service is taken by the Serving - Call Session Control Function (S-CSCF). When an end-user initiates a session the client profile specific to the requested service is encapsulated in the INVITE request; this request is forwarded to the terminating AS via the QMO AS. The terminating AS defines the service profile and encapsulates the information in the 183 SESSION PROGRESS response that is conveyed to the end-user via the QMO AS. The service and client profiles are represented using XML-based SDP-next-generation (SDPng). Typical IMS session negotiation takes place, but additional interactions between the QMO AS and the terminating AS facilitate optimisation and the selection of feasible service versions, based on the provided client and service profiles.

This configuration has the advantage of customised service delivery, optimised for the end-user preferences and terminal capabilities. The definition of generic client and service profiles allows new services to be rapidly deployed while inheriting advanced QoS negotiation and optimisation support. The matching and optimisation procedures carried out during session initiation and renegotiation increases signalling traffic, the effect that this has on the respective delays is still to be investigated. The model also suffers from poor scalability; for a large number of users, running QMO procedures separately for each session will be time-consuming and costly. The authors suggest that service configurations could be calculated in advance for particular services to offset these effects. The model separates the network resource authorisation and reservation procedures performed by the Common PCC framework from the customised delivery of advanced services performed by the QMO AS. A compelling alternative would be to incorporate the concepts of client and service profiles into the policy control mechanism of the Common PCC framework.

3.1.4 Policy Refinement and Enforcement

Policy refinement refers to the translation of policies at different levels of the management hierarchy, and, regarding general policy based management, has been under study for some time. However in the IMS and Common PCC context there is no standardised method to perform policy refinement and effectively map QoS descriptors across different layers of the policy life cycle.

As discussed in Chapter 2, the PDF receives authorisation requests from the service control plane, and upon extracting information creates PCC rules to be enforced in the transport plane. Albaladejo *et al.* discuss the creation of the PCC rule [47], and point out that this process is not specified in the 3GPP standards and is left for operator configuration. In particular the authors argue that there is no universal interpretation of how to map the content received in the Media-Component-Description Attribute Value Pair (AVP) in the authorisation request, into PCC rules. They propose two solutions for creating PCC rules for a session. The first creates a PCC rule for each Media-Component-Description AVP, while the second approach creates a PCC rule for each associated Media-Sub-Component AVP. The first approach complicates the structure and its operation; as there are essentially an unlimited number of Media-Sub-Component AVPs, the number of Flow-Description AVPs within each service data flow is also unlimited. The second approach limits the number of Flow-Description AVPs in each PCC rule to two, but splits media components into separate PCC rules. Through experimental analysis and the implementation of a practical testbed, the authors found that despite complicating the operation, the first approach, where the PCC rule is based on the Media-Component-Description AVP, was better suited to the IMS environment, as the second approach suffered serious scalability problems when more than one Media-Component-Description AVP was included in the authorisation request.

Kamienski *et al.* argue that the IETF defined Policy Based Network Management (PBNM) framework was not designed to deal with the heterogeneous and dynamic networks that characterise the NGN [48]. In the paper *XACML-based Composition Policies for Ambient Networks* [22], the authors present a framework for Policy-Based Management of Ambient Networks (PBMAN) that addresses these shortcomings. Ambient networks refers to a network integration solution that uses network composition to provide instant service access to users; it was chosen as an example of the heterogeneity and dynamics typical of an NGN technology, though the PBMAN framework is designed

for the management of any large-scale distributed wireless environment.

PBMAN adopts an integrated view of network and system management and the use of policies goes beyond the traditional tasks of access control, trust management and resource authorisation. The framework allows policies to be used for tasks such as system and network bootstrapping, setup of the distributed environment, negotiation and realisation of Service Level Agreements, and integrated real-time monitoring of user-level and system-level services. Policy composition refers to the process of managing a network of interconnected policy nodes; the framework defines composition policies that ensure that upon receiving a request, all the processes and subsequent actions are automatic and policy oriented. Policies are selected and processed automatically and in sequence, performing authorisation, authentication and eventually granting the end-user access to the service.

The PBMAN framework utilises the eXtensible Access Control Markup Language (XACML) policy language. XACML defines a declarative access control policy language in XML and an access decision language to pass requests and decisions between elements. Essentially these policies are made up of rules, each rule defines conditions and actions, if a condition is met an action is taken. Originally designed for access control, the XACML action parameters are limited to *permit* or *deny*; the authors extended this to include a range of different action clauses. This framework is generalised and has not been mapped to the IMS or Common PCC framework. Many standardised mechanisms and protocols have been replicated or not used, however the automatic refinement and processing of policies could be integrated into the Common PCC framework.

In the paper *Supporting Control Plane-enabled Transport Networks within ITU-T Next Generation Network (NGN) architecture* [49], Baroncelli *et al.* propose enhancements to the Common PCC framework to include Generalised Multi-Protocol Label Switching (GMPLS) Control Plane (CP)-enabled transport networks among the supported NGN transport technologies. GMPLS provides a common CP that enables network clients to request reserved circuits, called Label Switched Paths (LSP), via a User to Network Interface (UNI). The authors examine various methods of interaction between the Common PCC framework and the GMPLS CP and propose the addition of a new element, the Transport Resource Control GMPLS Functional Entity (TRCG-FE), to communicate directly with the CP entity of a GMPLS network via the CP Management Interface (CMI). The TRCG-FE accepts connectivity requests from a PDF and translates them into a set of GMPLS-enabled directives that are communicated to the CP entity

and enforced on the edge routers. This work demonstrates simplistic policy refinement, but defines complex translation models to enforce policies on technology specific (GMPLS) and vendor specific (Juniper) network equipment. These models use proprietary management APIs to configure the devices, making wide scale deployment in a multiple technology, multiple vendor environment, complex.

3.2 Horizontal Coordination of Resources

In the context of this thesis end-to-end QoS support refers to inter-domain coordination between administrative IMS domains. The IETF began work on a generalised end-to-end signalling protocol for the Internet in 2005. This led to the publication of RFC 4080, *Next Steps In Signalling (NSIS): Framework* [50], that describes a suite of protocols for signalling information about a data flow along its path in the network. The signalling protocol stack is divided into a generic (lower) layer and an upper layer specific to each application; this abstraction ensures that the NSIS protocol stack can be adapted to a wide range of applications. The generic layer is defined by a common NSIS Transport Layer Protocol (NTLP) known as the General Internet Signalling Transport (GIST) [51]. A NSIS Signalling Layer Protocol (NSLP) is defined for each individual application. The first use case of the NSIS protocol suite has been QoS signalling, and several IETF Internet drafts have been published that define the NSLP for QoS reservations and its applicability in the Internet, known as the QoS NSLP.

3.2.1 IETF NGN QoS Signalling

The Internet Draft *NSLP For Quality of Service Signalling* [52], defines a protocol that establishes and maintains state at nodes along the path of a data flow for the purpose of providing some forwarding resources for that flow. The design of the QoS NSLP is conceptually similar to RSVP, and the primary state management mechanism is peer-to-peer, i.e. state installation/refresh is performed between adjacent NSLP nodes, and not in an end-to-end fashion along the complete signalling path. The QoS NSLP extends the set of reservation mechanisms to meet the requirements stipulated in RFC 3726, *Requirements For Signalling Protocol*s [53]. In particular, support for sender and receiver-initiated reservation is incorporated. The Internet Draft defines three nodes: a QoS NSLP Entity (QNE), any element that supports QoS NSLP; a QoS NSLP Initia-

tor (QNI), the first node in a sequence of QNEs that issues a reservation request for a session; and a QoS NSLP Responder (QNR), the last node in a sequence of QNEs that receives a reservation request for a session. A QoS NSLP signalling session consists of a single QNI, any number of QNEs and a single QNR and can span the end-to-end data path, or a segmented portion of the path. QoS NSLP defines four messages: RESERVE, QUERY, RESPONSE and NOTIFY; these are used to manipulate QoS reservation state, request information about the data path, provide information in response to a previous QoS NSLP message and convey information asynchronously to a QNE, respectively. Because messages are sent peer-to-peer, a QNE considers its adjacent upstream or downstream peer to be the source of each message.

It is important to note that a distinction is made between the operation of the signalling protocol, and resource allocation and management techniques. A Resource Management Function (RMF) is defined that is responsible for all resource provisioning, monitoring and assurance functions in the network and is particular to a specific QoS Model (QOSM). This means that the QoS NSLP is independent of a specific QOSM, and all information related to RMF functions is carried in a QoS Specification (QSPEC) object that is encapsulated in the NSLP messages. The node initiating the QoS NSLP signalling adds an initiator QSPEC to the RESERVE message, which indicates the QSPEC parameters that must be interpreted by the downstream nodes to ensure that the intentions of the QoS NSLP initiator are preserved and resources are provisioned along the path. The QSPEC object is conveyed in QoS NSLP messages but is opaque to the NSLP signalling as it is only interpreted by the RMF, where the information is used to provision resources.

Ash *et al.* define a generic template for the QSPEC object, including a number of QSPEC parameters [54]. This template provides a common language to be reused in several QOSMs and aims to ensure the extensibility and interoperability of the QoS NSLP. The generic QSPEC template has been extended to included a number of QOSMs including DiffServ [55] and a model based on ITU-T Recommendation Y.1541 Network QoS Classes [56].

QoS NSLP is a candidate for the resource reservation protocol used by the Common PCC Framework for the pull mode approach to request QoS-enabled paths [27]. During typical IMS session initiation QoS resources are authorised for each domain, the second phase of operation involves resource allocation. To allocate resources the initiating UE creates an NSLP RESERVE message incorporating the relevant QSPEC object and a to-

ken data structure created during resource authorisation that allows the PDF to identify the particular session. At each QNE (now also acting as a PEP) the token data structure is encapsulated in a policy decision request and sent to the relevant PDF to identify the session and the decision taken during the earlier QoS authorisation phase. The decision is delivered to the QNE, where depending on the QOSM and QSPEC object information the transport plane device is configured. It is clear that the token data structure provides the necessary coordination between bearer resources and QoS control elements.

Unfortunately practical issues limit the applicability of this approach in real world scenarios in the short to medium term. The token based session binding mechanism relies on the ability of the UE to process and pass the token data structure into the serving network. UMTS uses PDP context activation operations to perform this but there is no standardised mechanism for all access networks [57]. Additionally the introduction of the QoS NSLP requires modification to all routing devices in the transport plane. Network operators heavily invested in legacy networks will be hesitant to commit the necessary capital expenditure for such an overhaul; link layer QoS signalling, like PDP context activation in UMTS, goes against the principle of separating core procedures from the subtleties of the access network. An alternative to this method of passing QoS requirements over the end-to-end path through path-coupled signalling, is to pass the requirements for a given service over the end-to-end path through application signalling via the inter-domain reference points.

3.2.2 Future Internet

There are a number of research initiatives examining the evolution of the Internet with regard to its structure and management in the future, the Future Internet. These initiatives range from evolutionary approaches where the majority of Internet structures are maintained, to revolutionary, clean slate approaches where the core protocols are redesigned from scratch. With the clean slate approach the applied technologies shall not be limited by existing standards or paradigms, this approach is based on the experience that supplementary additions to an established design are limited in their acceptance and introduction. The provisioning of advanced QoS connectivity services has been identified as a key driver for the operators business role in the Future Internet [39].

The End-to-End Quality of Service Support over Heterogeneous Networks (EuQoS) project is an ongoing European research initiative aimed at building an entire QoS frame-

work, addressing all relevant network layers, protocols and technologies [58]. The framework has been prototyped and tested in a multi-domain environment distributed across Europe. The project has a broad scope but deals with a research topic pertinent to this thesis, that of QoS routing or finding a feasible path between a source and destination node satisfying one or more QoS constraints.

The EuQoS framework defines a virtual network layer to decouple network decisions from network technologies. This layer is split into technology independent and technology specific layers. The technology independent layer houses Resource Managers (RM) that manage QoS reservation and authorisation for each domain, while the technology specific layer houses Resource Allocators (RA) that enforce specific decisions on transport plane devices. Essentially the RM acts as a PDF and RAs act as distributed PEPs. In order to check the availability of resources, EuQoS uses path-coupled signalling based on NSIS extensions defined by Cordeiro *et al.* in the Internet Draft *GIST Extensions for Hybrid On-path Off-path Signalling (HyPath)* [59]. The authors argue that the path-coupled signalling must reach all RMs along the path to ensure end-to-end resource availability, even though these RMs may not lie on the data path. The extension, known as EQ-NSIS, allows some routers to re-direct the end-to-end signalling to RMs that are not necessarily on the data path. To enable this a transparent middle layer between the NSIS Transport Layer Protocol (NTLP) and NSIS Signalling Layer Protocol (NSLP) is defined, known as the Hybrid Path. Essentially all border routers and RMs need to be Hybrid Path aware; at each border router the EQ-NSIS signalling is intercepted and re-directed to the local RM where resource availability is determined. This approach is similar to the pull mode operation used by the Common PCC framework in coordination with NSIS, however it also provides a means for RMs or PDFs to discover ingress and egress points through which the data-path will pass in its domain, and supports non-NSIS domains.

The telecoms operators participating in the EuQoS project considered the issue of QoS routing as the most important research item. Five different classes of service (CoS) are defined based on ITU recommendations. These CoSs are known and visible to applications and end-users. The routing tables, routing decision processes and traffic control mechanisms are CoS-specific. In the paper *EQ-BGP: An Efficient Inter-domain QoS Routing Protocol* [60], Beben *et al.* define extensions to the Border Gateway Protocol-4 (BGP-4) for the EuQoS framework; the extension is known as EQ-BGP. The extended protocol takes into account intra- and inter-domain QoS information to create a road map of available QoS paths between source and destination networks; these end-to-end

QoS paths are advertised to neighbouring domains using EQ-BGP. The protocol includes an optional path attribute that conveys information about the QoS capabilities of a path, and a QoS assembling function for computing aggregated values of QoS parameters for end-to-end routing paths. EQ-BGP also handles multiple routing tables in order to store the available paths for each end-to-end CoS.

As with any path-coupled approach to end-to-end QoS routing, significant modification to the legacy transport plane is necessary. Additionally the EuQoS framework requires the sharing of potentially sensitive QoS and topology information with neighbouring domains. Many of the standardised mechanisms and protocols of the Common PCC framework have been replicated or not used in this framework.

3.3 Discussion

The chapter has presented the state of the art regarding tailoring the Common PCC framework to ensure rapid and innovative service creation, and the provisioning of end-to-end QoS enabled paths in IP networks. The IETF has been strongly involved in both these areas, and the fact that it is the policy of the 3GPP to adopt IETF standards wherever possible into the IMS standards [6], deems it necessary to place importance on the solutions and mechanisms proposed by the IETF. In particular, this thesis places emphasis on the concept of session policies and the need to incorporate end-user preferences, network constraints and service requirements into the Common PCC framework. Innovative applications need to be rapidly deployed, and they should be able to inherit advanced QoS support without any standardisation. The effect that this enhanced QoS control mechanism has on end-user experience should be negligible. Additional signalling should be limited to the core network wherever possible to avoid expensive round trip delays.

The proposed inter-domain routing algorithms are path-coupled or variations thereof; in this thesis we examine the alternative approach of performing inter-domain coordination at the service control layer by passing QoS requirements over the end-to-end path through application signalling via inter-domain reference points. This has the added advantage of backward compatibility with legacy equipment because of limited transport plane overhaul. The questions of how to facilitate end-to-end resource reservation across multiple domains using currently standardised elements and interfaces, and how to link service control inter-domain routes with the routes followed by the media in the

transport plane, need to be answered.

Chapter 4

Application Driven Policy Control Framework

The previous chapters, in reviewing the standardised NGN resource management frameworks and related literature, highlighted the necessary vertical coordination between applications requesting resources and the transport plane devices that will carry the service data flows. Open issues within the standardised Common PCC framework became apparent, including policy representation, policy processing, policy prioritisation and application-policy interaction. The deployment of new IMS services without the need for QoS standardisation or network upgrade was an important focal point in related literature. This chapter proposes extensions to the Common PCC framework for innovative service creation. These extensions give application developers greater control over the way their services are handled in the transport plane, and allow new services to inherit advanced QoS provisioning support, facilitating rapid deployment [61].

The chapter begins by outlining design considerations pertinent to the extended Common PCC framework; these considerations were formulated based on the review of the standardised frameworks and related literature. The proposed architecture is presented, with emphasis on the multi-policy, multi-layered and modular aspects. The architecture design is described in detail, regarding functional elements and extended interactions. It is important to note that this chapter presents a logical architecture that is not implementation specific.

4.1 Design Considerations

Any extensions to the IMS resource management framework should take into account certain fundamental considerations. These considerations place particular emphasis on vertical coordination and service deployment, and are described in order of precedence.

4.1.1 Service differentiation through policy controlled QoS

The primary concern of the Common PCC framework is to distinguish IMS services from typical Web 2.0 services. The widespread proliferation of dynamic web services poses a challenge to IMS service deployment; operators will justify charging for services that are typically available freely on the Internet through service differentiation. The main driver for this differentiation is increased reliability and guaranteed transport of multimedia traffic. Real time applications will require increased bandwidth, lower packet loss and lower latency. This can be achieved by efficiently managing scarce network resources through mediation with the Common PCC framework.

While the scope of the Common PCC framework is broad and this chapter discusses extensions to address a limited set of shortcomings, the proposed extensions should not compromise this primary goal or the critical process of QoS resource control that includes the authorisation, reservation and commitment of network resources.

4.1.2 QoS standardisation not required for new services

Rapid service creation is a defining characteristic of the IMS framework. Application developers can create complex services that inherit functionality from reusable service enablers. These services can be deployed across different platforms with minimum effort because of standardised interfaces. A critical requirement of the Common PCC framework is to have no impact on efficient service deployment [16], and consequently no QoS standardisation or network upgrade should be required to deploy a new service.

Extensions to the Common PCC framework, giving application developers greater control over the way their services are handled in the transport plane, should not compromise this requirement. Increased interaction between applications and the policies that govern resource allocation should not add complexity to the service deployment procedure. Furthermore, it should be possible for new services to inherit advanced QoS negotiation support without prior configuration.

4.1.3 End-user preferences and service requirements as a QoS resource control aspect

To cater for increasingly complex interactive multimedia applications and the dynamic environment brought about by regular service deployment, it is desirable to incorporate end-user preferences and service requirements into the QoS resource control procedure in the Common PCC framework [21]. End-users should be able to specify terminal and access network constraints that might affect the requested service that they are requesting and the way the service data flow is handled in the transport plane. Furthermore, applications should have access to these specified preferences to customise the service delivery.

One issue not addressed by the standardised frameworks and related literature is increased interaction between applications and policies that govern QoS resource control. Apart from the highly granular description of service requirements contained in the Session Description Protocol (SDP) body of the application signalling, applications have very little control over the way their services are treated in the transport plane. Applications should be able to actively participate in the creation of the PCC rule, as it is this rule that exhaustively describes how the service data flows should be treated in the transport plane. Essentially detailed service requirements specific to each application, should be incorporated into the policies that govern QoS resource control and in particular the creation of the PCC rule.

4.1.4 Push versus Pull mode operation

Push mode or network initiated QoS control uses the network to initiate the signalling to request resources and push PCC rules to the transport plane devices. The main motivation for push mode operation is that services are typically provided by the access network operator, potentially through peering agreements with 3rd party service providers. Therefore it is natural that the access network and service owner assigns the QoS level, per packet flow associated with a particular service. Push mode operation has the following additional advantages over pull mode [62]:

- It supports access agnostic client applications.
- It supports the split-terminal scenario where the client application resides in a node that is physically separate from the terminal.

- QoS control policies are centralised in the network as opposed to being distributed across numerous terminals from multiple vendors, ensuring policy consistency.

The proposed framework considers push mode operation the most useful in cases where the operator controls the service. While the extensions may support pull mode operation, the possibility is not elaborated upon in this thesis.

4.1.5 Standards conformance

There have been a number of bodies involved in the standardisation of the IMS and the NGN resource management framework. The first priority is to conform to the Common IMS specification maintained by 3GPP [63], and the Common PCC framework defined in this thesis that encompasses the principal standardised NGN resource management frameworks.

The primary concern of the IMS core network is to handle authentication, authorisation and call routing. Service execution occurs in the service control plane Application Servers (AS) and QoS resource control is managed at the resource control plane by the Common PCC framework. IMS specifications do not define how advanced services should be provisioned, but do define an interface, the IMS Service Control (ISC) interface, that connects the Serving - Call Session Control Function (S-CSCF) to ASs. The Diameter based Sh interface facilitates interaction between ASs and the Home Subscriber Server (HSS). IMS specifications do not prescribe how policies should perform authorisation, reservation and commitment of resources. However standardised interfaces are defined for interaction between the different planes of the IMS model.

One approach to advanced QoS negotiation is to house the extended functionality in QoS specific ASs and include these on the relevant signalling path based on initial Filter Criteria (iFC) defined in the HSS. This approach could be used to incorporate end-user preferences and service requirements into the QoS negotiation procedure, but is limited in its ability to actively contribute to the creation of the PCC rule. This approach also separates the logical functions of QoS resource control, performed by the Common PCC framework, and the customised delivery of services. A second approach that conforms to the standardised architecture is to incorporate this functionality into the policy control mechanism of the Common PCC framework. While supporting all standardised interfaces this approach would use policies to incorporate end-user preferences and service requirements into the authorisation procedures. Furthermore application-specific poli-

cies would allow direct interaction between applications and the PCC rules that define how service data flows are treated in the transport plane. The ASs should have access to the relevant set of policies, to facilitate customised service delivery, optimised for end-user preferences and access network constraints.

As it is the policy of the 3GPP to adopt solutions and mechanisms proposed by the IETF wherever possible, any extensions to the Common PCC framework should emphasise IETF efforts, in particular the framework for SIP session policies, proposed by Hilt *et al.* [42]. Once the session policy framework reaches maturity and the relevant Internet Drafts are compiled as IETF Request for Comments (RFC), it is possible that the concept could become part of the 3GPP Common IMS standard. The session policies framework need not necessarily form part of the proposed application driven framework, however extensions should not inhibit its implementation.

4.1.6 Negligible effect on end-user experience

A critical requirement of the Common PCC framework is that the QoS resource control mechanism has a negligible effect on session setup delay [16], hence additional round-trip times between the UE and core network should be avoided. An equally important metric is the signalling overhead; a resource management framework that guarantees resource reservation but introduces excessive load on network elements is unacceptable. These metrics are of utmost importance when verifying and evaluating extensions to the Common PCC framework. Another requirement is that the complexity of the UE be minimised. The creation, storage and processing of policy documents at the UE are impractical, these procedures should be limited to the core network.

4.1.7 Scalability and extensibility

Regarding network security, scalability is of utmost importance to prevent Denial of Service (DoS) attacks. Simple mechanisms to prevent this involve the distribution of critical elements across a number of servers with incorporated fail-over support. In the Common PCC framework the PDF is a critical element, which acts as the single point of contact for resource requests. The separation of the policy decision element into the PDF and x-RACF exhibits scalability and introduces a layer of abstraction.

Another method of distribution involves the creation of sub-domains within a single domain, each with their own PDF element. The specified Rd intra-domain interface

means that while there is still a single point of contact for resource requests, distributed PDFs in sub-domains within the same administrative domain can detect and determine requested resources. Any extensions should support this distribution, and load sharing and fail-over mechanisms.

The IMS architecture will see regular deployment of new services, resulting in a constantly changing network dynamic. Hence the resource management function will need to be flexible and adaptable. Specifically network operators should be able to rapidly create and deploy new policies within the Common PCC framework. Additionally, the Common PCC framework will need to adapt as the service requirements for advanced applications adapt. Any extensions should be modular, and should allow new solutions to be incorporated and old mechanisms to be replaced. Operator configuration should be simple, policies should be easily processed and human-readable. The use of XML to represent the policy control structure would improve interoperability and simplicity, while also taking into consideration factors such as usability and flexibility.

4.2 Application Driven Policy Control

The primary aim of the application driven policy control framework is to allow greater interaction between applications and QoS control policies. A framework for multiple policies is defined that uses policies to enable end-user preferences and service requirements for customised service delivery, as well as the traditional tasks of access control and resource authorisation. The framework conforms to IMS and Common PCC standards by incorporating these extensions into the policy control mechanisms. Policy refinement occurs at both the service control and resource control planes. ASs have access to the policies that affect service customisation, these policies help to create resource authorisation requests. The architecture is essentially modular, new policies and policy processing mechanisms can be rapidly defined and deployed; new concepts including the IETF defined framework for SIP session policies can be incorporated.

4.2.1 Multi-policy

In Chapter 3, it was pointed out that the creation of the PCC rule is left to operator discretion, and that there is no universal interpretation of how to map service requirements into PCC rules [47]. There is also no standardised mechanism to incorporate policies

into the authorisation, reservation and commitment procedures, or the creation of the PCC rule.

The proposed architecture instantiates the policy repository element defined in the Common PCC framework and extends XML schema defined by the IETF for the policy control structure. Policy storage and retrieval mechanisms implemented by the Open Mobile Alliance (OMA) initiative are incorporated. The PDF and x-RACF are extended to combine service information from authorisation requests with information from policy documents to create PCC rules that describe the transport plane treatment of a particular service data flow.

QoS resource control can incorporate a wide range of policy information, from operator specific constraints, to access network limitations, to end-user preferences. The proposed architecture includes a processing mechanism that allows any number of policies to be sourced for QoS resource control. In particular application policies are prescribed that are specific to each deployed IMS service; these policies consist of service requirements as defined by the application developer, and end-user preferences as defined by the individual end-users. Generic application policies allow new services to inherit advanced QoS processing mechanisms without the need for standardisation. However application developers have the ability to tailor the creation of the PCC rule and hence have greater control over the way the service data flows are handled in the transport plane.

When multiple policies are involved in QoS resource control a deadlock situation is possible where different policies specify conflicting constraints. To cater for this eventuality each end-user has an associated policy profile that defines a priority for each control policy. The higher the priority of the policy, the sooner that policy is invoked in the QoS resource control procedure. Essentially lower priority policies must conform to the constraints specified by higher priority ones. The policy profile also specifies filter criteria that describe which authorisation requests should invoke which control policies. Policies will be specific to certain sessions, e.g. an application policy specific to a Video on Demand (VoD) service need not be invoked for a basic voice session. Upon receipt of an authorisation request the policy profile specific to the requesting end-user is retrieved, and filter criteria result in policies being selectively incorporated into the QoS resource control mechanism.

The application driven model creates some compelling business cases. With the incorporation of application specific policies, 3rd party application developers could pay for the privilege of specifying minimum constraints for the delivery of the service in the

transport plane. Additionally a further premium could allow applications access to more resources by assigning a higher priority to the relevant application specific policy. A premium service with real-time interactive components would likely be willing to pay for preferential treatment through increased policy priorities.

By incorporating multiple policies into the QoS resource control procedure, the processes involved can be controlled by a wide range of sourced information making the system flexible and extensible while still maintaining the core functions of the architecture. Application specific policies incorporate service requirements and end-user preferences into the creation of the PCC rule. The extensions are incorporated into the policy control mechanism in the core network and do not introduce additional round trip times between UE and core network, or require additional terminal processing. This minimises the impact on end-user experience and necessary terminal capabilities.

4.2.2 Multi-layered

The IMS spans the service control, resource control and transport planes, and the effective delivery of services with QoS guarantees can not be performed in just a single layer [20]. The proposed application policies provide the critical link between end-users, applications and the Common PCC framework. These policies consist of two components, an end-user defined component and an application defined component. In the proposed framework ASs in the service control plane have access to the application policies, in particular the end-user defined preferences. This information is combined with the highly granular service information contained in the SDP body of the signalling and incorporated into the authorisation requests. The information is also used by the AS to customise service delivery to end-user preferences and terminal capabilities.

Upon receipt of authorisation requests, the Common PCC framework has access to a number of different policies including the relevant application specific policy. Of particular importance is the service requirement component defined by the application developers. These service requirements are incorporated into the QoS resource control mechanism and can be used to further refine the PCC rule by including transport specific classifiers and/or link layer QoS information. It is clear that both the AS and Common PCC framework require access to the policy repository, and to ensure system scalability these elements should maintain as little state as possible. Consequently policies are not stored at the elements and need to be retrieved each time an authorisation request is

created or received.

This approach to inter-plane interaction conforms entirely to the Common IMS specifications and the Common PCC framework - by deploying application specific policies and opening access to the Common PCC framework and relevant ASs, QoS resource control is performed across the IMS planes and customised service delivery is incorporated into the mechanism.

4.2.3 Modular Policy Processing

The importance of rapid service deployment has been stressed already, of equal importance is the ability of the Common PCC framework to adapt to the dynamic requirements of advanced multimedia services. The application driven framework defines four generic policies: domain specific, access specific, subscription specific and application specific policies. These are just base policies and new policies can be dynamically incorporated into the framework.

Incorporating a new policy into the framework involves defining the new policy adhering to the XML schema, defining a method of interaction between the policy repository and any elements that might require the policy information, and processing the policy information at those elements. The policy structure based on extended IETF XML schema allows for the creation of flexible policies. The interface between the policy repository and elements consuming policy information is undefined in the Common PCC framework and flexibility is of utmost importance. The deployment of new policies should not require any standardisation or the implementation of new protocol applications - the chosen protocol ensures security but is transparent in its transport of policy information. Each element that consumes policy information defines a policy processor block that interprets that particular policy, hence each deployed policy needs an associated policy processor block. These blocks are modular and invoked in the QoS resource control procedure based on the priority of the relevant policy. This ensures the flexibility of the system and allows operators to adapt to changing network conditions by rapidly adding or removing policies.

The extensions conform to the Common IMS and Common PCC frameworks, no additional requirements or external modifications are necessary to implement the architecture. This means that other solutions that conform to the aforementioned standards can be easily incorporated into the application driven framework, allowing new concepts

to be added as the requirements adapt.

4.3 Solution Architecture

The application driven policy control framework is depicted in Fig. 4.1. The functionality is implemented in four reference elements, namely the policy repository, the AF, the PDF and the x-RACF. The policy repository is instantiated and designed to facilitate secure, transparent and flexible deployment of policies. The AF is extended to interact with the policy repository and to incorporate the end-user preferences component of the application specific policies into authorisation requests. The PDF and x-RACF are extended to interact with the policy repository to include multiple policies into QoS resource control and particularly the procedure of creating the PCC rule.

These extended elements and interactions incorporate end-user and service preferences into the QoS resource control mechanism. This is a reference architecture, which defines logical elements. The actual physical architecture is implementation specific and elements can be co-located or distributed across several hosts.

The reference architecture is shown in conjunction with IMS core elements and a UE to illustrate the necessary interactions. The architecture is limited to a single IMS domain; Chapter 5 describes inter-domain policy control mechanisms that can be used to extend the application driven policy control framework to operate across domains.

4.3.1 Policy Repository

The policy repository is instantiated by an XML Document Management Server (XDMS) which has been standardised by the Open Mobile Alliance (OMA) [64]. An XDMS is a widely used IMS AS used for the secure storage and retrieval of XML documents from a centralised server. XDMS is a powerful service enabler because it is capable of storing any information encoded in XML format, and is commonly used for storing service related information for network address books, conferencing applications, instant messaging and push-to-talk.

In the proposed architecture the XDMS is used to store and manipulate policies concerned with QoS resource control. The information is stored in a Structured Query Language (SQL) database, and the XDMS authorises users that attempt to upload, download or modify the stored documents. Documents are also verified to ensure that

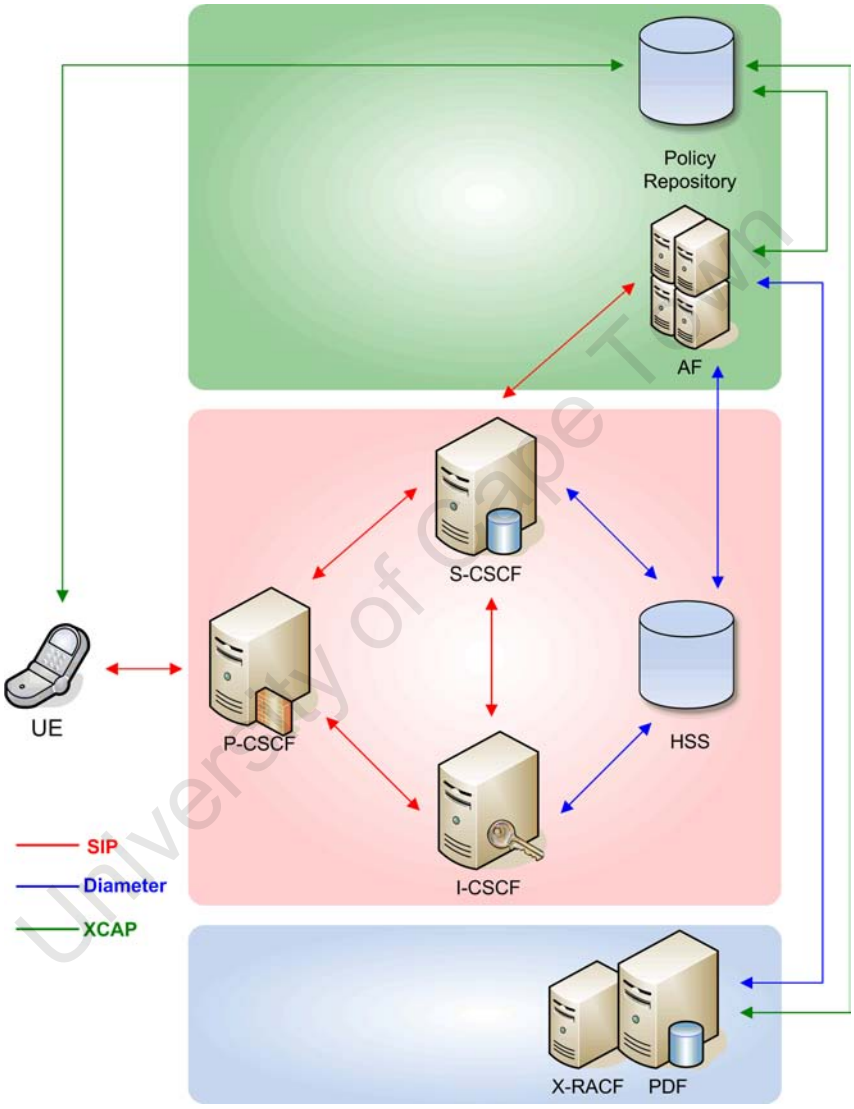

Figure 4.1: The solution architecture for application driven policy control.

they are well formed and valid. Uploaded documents are only accepted if they conform to the schema described in the next section, which is based on the Media Policy Dataset Format (MPDF) proposed by Hilt *et al.* [45].

The interface between the policy repository and related elements is not defined in the Common PCC framework. The XDMS specification uses the XML Configuration Access Protocol (XCAP) to interact with external elements [65], which, in the case of the proposed architecture, are the UE, the AF, the PDF and the x-RACF. This protocol defines Hypertext Transfer Protocol (HTTP) PUT, GET and DELETE methods to store, retrieve and remove documents respectively. The information is carried over a secure HTTP connection out of band of IMS signalling and all elements that need access to the policy repository are equipped with XCAP interfaces. Fig. 4.2 shows the logical functions of the policy repository.

The XDMS is chosen as the means of policy storage for several reasons. XDMS is a widely used IMS service enabler and has been incorporated into the 3GPP Common IMS specification. The connection to the XDMS is defined as the Ut interface and the protocol as HTTP, as shown in Fig. 4.2. Hence using this enabler as the mechanism to store and manipulate policies is entirely conformant with IMS standards. The use of XML as a basis for creating self-describing, human-readable and portable policies at all levels of the policy life cycle, is gaining popularity [17]. The XDMS can process any XML documents, and verify them against defined schema; this provides great flexibility and extensibility in terms of rapid definition and deployment of new policies. Using XCAP as the protocol of interaction, as opposed to an authentication protocol like Diameter, ensures that the system is flexible and can adapt to changing network conditions. In the case of Diameter, each deployed policy would require a separately defined Diameter application. XCAP, while secure, is transparent in its transport of policy information and no extensions are required to deploy new policies.

Policy Format

Policy Based Network Management (PBNM) differentiates between two policy types, these policies are given various names, in the context of this thesis they are called control policies and enforcement policies. Control policies protect resources through admission control and, combined with service information and requirements, generate enforcement policies. Enforcement policies define the treatment of a session in the transport plane by specifying the rules used to configure the physical devices. In the proposed architecture,

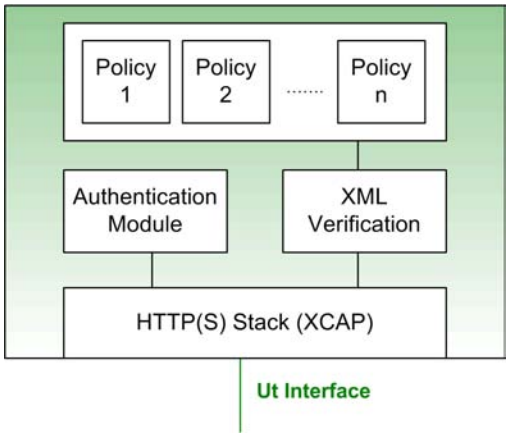

Figure 4.2: Logical architecture of the policy repository.

control policies assist with the creation of authorisation requests in the service control plane, and combine with service information in the resource control plane to create enforcement policies, while enforcement policies represent PCC rules.

This differentiation is similar to the concept of session-independent and session-specific policies defined in the framework for SIP session policies [42]. Session-independent policies, like control policies, are created independently of each session, are stable and are typically sustained over long periods of time. Session-specific policies, like enforcement policies, are created for a particular initiated session and only exist for the duration of the session. The Media Policy Dataset Format (MPDF) proposes an XML document format to represent *session-policy* and *session-info* documents that extends the schema for SIP User Agent Profile Datasets [66]. In the application driven policy control framework, *session-policy* schema define the control policy structure and *session-info* schema define the enforcement policy structure.

Control policies can be broad regarding their participation in the QoS resource control procedure and are stored in the policy repository. The MPDF allows extensions to the document format, this means that new elements can be defined in the *session-policy* document format to incorporate the broad range of control policies. The application driven policy control framework defines four base control policies: domain, subscription, application and access control policies. Domain policies will be set by the network operator and essentially define network constraints, including available bandwidth, authorised codecs, etc. Subscription policies define the subscription state of the end-user, the registered services and service priorities. These policies play the basic role of the standardised

SPR/NASS/NACF elements. This information is stored elsewhere in the IMS core, including in the Home Subscriber Server (HSS). The harmonisation and distribution of the various policy repositories are beyond the scope of this thesis. Application policies are specific to each deployed IMS service, allow applications to participate in the creation of the PCC rules and facilitate service delivery customised to end-user preferences. Access specific control policies define extensions to the basic domain policies to include access network specific constraints.

Fig. 4.3 shows a typical domain policy represented as a *session-policy* document. These will be network operator specific and this is only an example policy. New elements *<domain-policy>, <domain>* and *<qos-class>* are defined that specify authorised domains and QoS classes. Also included are *bw-downlink* and *bw-uplink* attributes, these demonstrate actions that policies can take, e.g. if available bandwidth is insufficient, disallow the session. This information could be incorporated into QoS resource control from the Technology and Resource Information Specification (TRIS) maintained by the Common PCC framework, but is included in this domain policy to demonstrate how the *session-policy* document format can be extended for full QoS resource control.

Once the control policies have performed the authorisation phase of QoS resource control, reservation needs to take place. For this, enforcement policies are defined that describe exactly how the session should be treated in the transport plane. These policies are created based on service information included in the authorisation requests and input from relevant control policies. Enforcement policies are defined for each authorised IMS session and are stored in the policy repository for the duration of that session. This is important as it assists with the maintenance of the TRIS, which is used for QoS reporting and resource monitoring. The *session-info* document that represents the enforcement policies can also be extended, this caters for advanced multimedia applications with new service requirements.

The use of XML schema as defined in the framework for SIP session policies conforms to the Common IMS specification and paves the way for any future IETF session policy extensions. The flexibility and extensibility of XML has already been discussed, by adopting *session-info* and *session-policy* documents to represent control and enforcement policies, the proposed architecture uses XML throughout the policy life cycle to represent the various levels of policy refinement. However in the context of the Common PCC framework the XML-based enforcement policies need to be translated into PCC rules to ensure that configuration information can be interpreted by, and enforced in, the

```
<property-set>
  <domain-policy>
    <domains excluded-policy="disallow">
      <domain policy="allow">
        <address bw-downlink='20480000' bw-uplink='20480000'>open-ims.test</address>
        <address bw-downlink='20480000' bw-uplink='20480000'>qos-ims.test</address>
        <address bw-downlink='20480000' bw-uplink='20480000'>visited-ims.test</address>
      </domain>
    </domains>
    <qos-classes excluded-policy="disallow">
      <qos-class policy="allow">
        <class-id bw-downlink='20480000' bw-uplink='20480000'>1</class-id>
        <class-id bw-downlink='20480000' bw-uplink='20480000'>2</class-id>
        <class-id bw-downlink='20480000' bw-uplink='20480000'>3</class-id>
        <class-id bw-downlink='20480000' bw-uplink='20480000'>4</class-id>
        <class-id bw-downlink='20480000' bw-uplink='20480000'>5</class-id>
        <class-id bw-downlink='20480000' bw-uplink='20480000'>6</class-id>
      </qos-class>
    </qos-classes>
    <media-types excluded-policy="disallow">
      <media-type policy="allow">audio</media-type>
      <media-type policy="allow">video</media-type>
    </media-types>
    <codecs excluded-policy="disallow">
      <codec policy="allow">
        <mime-type>audio/PCMU</mime-type>
      </codec>
      <codec policy="allow">
        <mime-type>audio/GSM</mime-type>
      </codec>
      <codec policy="allow">
        <mime-type>video/H263</mime-type>
      </codec>
    </codecs>
  </domain-policy>
</property-set>
```

Figure 4.3: The domain policy extends the MPDF *session-policy* document to include authorised domains and QoS classes.

transport plane devices. This translation process takes place at the PDF and x-RACF.

Application Policy Interaction

In the Common PCC framework an AF composes service requirements based on information contained in the SDP bodies of signalling messages, to request authorised resources. Apart from this highly granular service information, applications have no input into the PCC rule creation process. Additionally applications do not consider end-user preferences when creating the authorisation requests, or delivering the service to the end-user.

The concept of session-specific and session-independent policies has been discussed. In addition to this, application specific policies are defined. To cater for diverse end-user

requirements and terminal capabilities, it is likely that multiple application configurations will exist; for each application configuration an application specific policy is defined. Advanced applications will likely have a set of configurable preferences that define end-user and terminal constraints; for each subscriber to the service an end-user specific application policy is defined. Hence a single application policy may consist of a number of application specific components and a number of end-user specific components. These policies are loosely based on the client and service profiles proposed by Skorin-Kapov *et al.* [46]. Application developers create these policies when deploying their service, and end-users and applications can modify the policies via the standardised Ut interface.

To ensure that the concept of application specific policies does not interfere with the rapid deployment of new services, generic application policies are instantiated that define basic end-user preferences and service requirements and have predefined sane default values. This ensures that new services can be rapidly deployed and, by modifying basic service requirements in the generic application policy, they can benefit from advanced QoS negotiation when customising service delivery to end-user requirements. The generic application policy includes the following end-user specific attributes: supported media types, acceptable service formats and, audio/video/text priority and desired quality. Generic service requirements include minimum constraints on bandwidth, delay, loss, jitter and bit error rate in the uplink and downlink. As with all control policies based on the *session-policy* schema, application policies can be extended by adding new policy elements and attributes, to be customised for a particular service.

The concept of application policies incorporates end-user preferences and service requirements into QoS resource control and the definition of generic application policies means services can inherit advanced QoS negotiation support without any QoS standardisation. Incorporating these concepts into the policy control procedure of the Common PCC framework is entirely conformant with relevant standards. The configuration of application policies is a once off event and occurs out of IMS band signalling. This has a negligible effect on session initiation delay as no additional round trip times are incurred. The storage, interpretation and processing of policy documents occur within the core network and no extensions to the UE are required.

Access Network Policy Refinement

Access specific control policies are defined as an extension to domain policies; these policies are used by the A-RACF to further refine the enforcement policy in the access do-

main by incorporating access specific network constraints. A typical access control policy would specify network constraints, including maximum uplink and downlink bandwidth per session and path latency. Access specific policies could also incorporate traffic classifiers (e.g Y. 1541 QoS classes) and link layer QoS information (e.g. 802.1p priority values) into the enforcement policies. These policies facilitate access specific policy refinement and extend IMS policy control to the access network. The actual inter-working of different access technologies with the IMS core architecture is beyond the scope of this thesis and only simple generic access policies are defined.

Policy Profiles

With multiple policies incorporated into QoS resource control, a deadlock situation is possible where different control policies specify conflicting constraints. The proposed architecture defines a policy profile for each end-user. The policy profile defines priorities for each control policy and also filter criteria that describe which kind of authorisation requests should invoke which policies. The policy profile is essentially similar to the end-users service profile downloaded from the HSS during registration. In fact this information could be incorporated into the service profile, though to maintain standards conformity and the use of standardised reference points, the policy profiles are retrieved from the policy repository upon receipt of an authorisation request.

Certain control policies will be specific to certain sessions. For example an application control policy for a Video on Demand (VoD) service will not need to be invoked for a simple audio session. Applications that have numerous available versions, might have a number of different application policies specific to each configuration; only the policy specific to the requested service need be invoked. Filter criteria defined in each policy profile result in policies being selectively invoked and incorporated into QoS resource control based on service parameters included in the authorisation request. These parameters include media type, flow descriptions, maximum requested uplink and downlink bandwidth, codec data and any service parameters that might be incorporated into the Diameter Rx application in the future. Essentially policies are triggered as a result of pattern matching on any Diameter AVPs.

3GPP Release 7 introduced the notion of the IMS Communication Service Identifier (ICSI), which is a mechanism that allows an end-user and network to identify which service is intended to be invoked with the SIP signalling. Furthermore an IMS Application Reference Identifier (IARI) is defined that links a service to the application invoking

it. With this information available, the decision of which policies to invoke becomes simple, however the proposal is still under development and suffers from several drawbacks when implemented incorrectly, including fraud, systemic interoperability failures, and a complete stifling of the innovation that SIP was meant to achieve [67]. Domain, subscription and access control policies need always be invoked, but for other policy types, the creation of complex and flexible filter criteria will be necessary.

Additionally the policy profile defines a priority for each incorporated control policy - this is a positive integer value where 1 specifies the highest priority. The higher the priority the sooner that control policy is invoked in the QoS resource control procedure. A higher priority control policy would perform authorisation and create a base enforcement policy, which lower priority policies must build upon. In this way lower priority control policies must conform to constraints specified by higher priority ones. Fig. 4.4 shows an example policy profile for the IMPU *sip:bob@qos-ims.test,* the subsequent order in which control policies are invoked is domain policy, subscription policy and VoD policy. Control policy addresses are identified by their XCAP URIs. It is assumed that the XCAP username is the same as the IMPU, the harmonisation of XDMS and IMS authentication is beyond the scope of this thesis.

4.3.2 IMS Service Control

When initiating a session involving advanced multimedia services the signalling typically traverses an AS based on the initial Filter Criteria (iFC) stored in the end-users service profile. This AS acts as an AF and creates resource authorisation requests by mapping the SDP information directly into Attribute Value Pairs (AVP) encapsulated in Diameter Authorisation Request messages; this mapping is specified in 3GPP TS 29.214 [29].

The proposed architecture incorporates end-user preferences, as specified in the application policies, into this procedure. When creating an authorisation request, the AF retrieves the application control policy, specific to the requested service and to the requesting UE, from the policy repository via the Ut interface. This information specifies end-user constraints and is used to deliver customised services. For example if multiple service configurations are available for a particular service, this information is used to deliver the most applicable version. This information is also used to modify the service information in the authorisation request that is sent to the resource control plane via the Rx interface, tailoring it to the specific service configuration that will be streamed

```
<PolicyProfile>
  <PublicIdentity>sip:bob@qos-ims.test</PublicIdentity>
  <FilterCriteria>
    <Priority>1</Priority>
    <TriggerPoint>
      <SPT>
        <ConditionNegated>0</ConditionNegated>
        <Command-Code>265</Command-Code>
      </SPT>
    </TriggerPoint>
    <ControlPolicy>
      <PolicyName>http://xcap/domain-policy/global/domain-policies.xml</PolicyName>
    </ControlPolicy>
  </FilterCriteria>
  <FilterCriteria>
    <Priority>2</Priority>
    <TriggerPoint>
      <SPT>
        <ConditionNegated>0</ConditionNegated>
        <Command-Code>265</Command-Code>
      </SPT>
    </TriggerPoint>
    <ControlPolicy>
      <PolicyName>http://xcap/sub-policy/users/sip:bob@qos-ims.test/sub-policies.xml</PolicyName>
    </ControlPolicy>
  </FilterCriteria>
  <FilterCriteria>
    <Priority>3</Priority>
    <TriggerPoint>
      <SPT>
        <ConditionNegated>0</ConditionNegated>
        <Command-Code>265</Command-Code>
      </SPT>
      <SPT>
        <ConditionNegated>0</ConditionNegated>
        <AVP>Media-Type</AVP>
        <Value>Video</Value>
      </SPT>
      <SPT>
        <ConditionNegated>0</ConditionNegated>
        <AVP>Max-Requested-Bandwidth-DL</AVP>
        <Value>131072</Value>
      </SPT>
    </TriggerPoint>
    <ControlPolicy>
      <PolicyName>http://xcap/vod-policy/users/sip:bob@qos-ims.test/vod-policies.xml</PolicyName>
    </ControlPolicy>
  </FilterCriteria>
</PolicyProfile>
```


Figure 4.4: Policy profile containing filter criteria for domain, subscription and Video on Demand control policies.

to the UE.

Fig. 4.5 shows the logical architecture of the extended AF in the proposed framework. These extensions allow service delivery to be customised to end-user preferences. The design is modular and future modules can be incorporated to assist in creating authorisation requests. The additional signalling introduced during session initiation is

within the core network, no additional round trip times are introduced and the effect on end-user experience will be negligible.

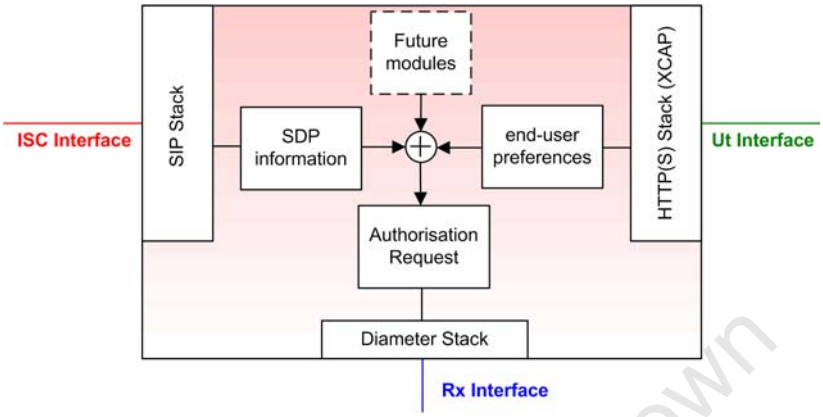

Figure 4.5: The extended application function combines service information with end-user preferences to create resource requests.

4.3.3 Extended PDF and x-RACF

The PDF and x-RACF are similarly extended to incorporate multiple policies into the QoS resource control procedure, for simplicity the PDF and x-RACF are co-located in this description. When the PDF/x-RACF receives a Diameter authorisation request, the relevant service information is extracted. Policy profiles specific to the requesting UE are downloaded from the policy repository and, based on the filter criteria, relevant control policies are retrieved. Authorisation requests can contain service information for a complete session, or can be separated based on media components.

Control policies can be involved in the process of authorisation, of refining already created enforcement policies or both. For each control policy a policy process block is created that defines how the policy is interpreted. Generic processor blocks are defined and these can be tailored to allow control policies to perform complex authorisation and policy refinement. This modular design means that creating and deploying a new policy involves defining the policy structure based on the *session-policy* schema and creating the relevant policy processor block that interprets that policy. Generic control policies and policy processor blocks allow network operators to adapt to ever changing network conditions by rapidly adding and removing control policies.

The PDF/x-RACF creates a technology independent enforcement policy based on

the *session-info* schema, sourcing information from the service information in the authorisation request, from the control policies, from the maintained Topology and Resource Information Specification (TRIS) and from any other additional modules. The mapping of the XML-based enforcement policy into the PCC rule, to be conveyed to the transport plane via the Gx interface, is based on the investigation by Albaladejo *et al.* [47]. Though an enforcement policy is defined for the entire authorisation request, it is composed into PCC rules, with one PCC rule for each Media-Component-Description AVP in the authorisation request as this ensures system stability and scalability, even with complex authorisation requests. Fig. 4.6 shows the logical functions of the extended PDF/x-RACF.

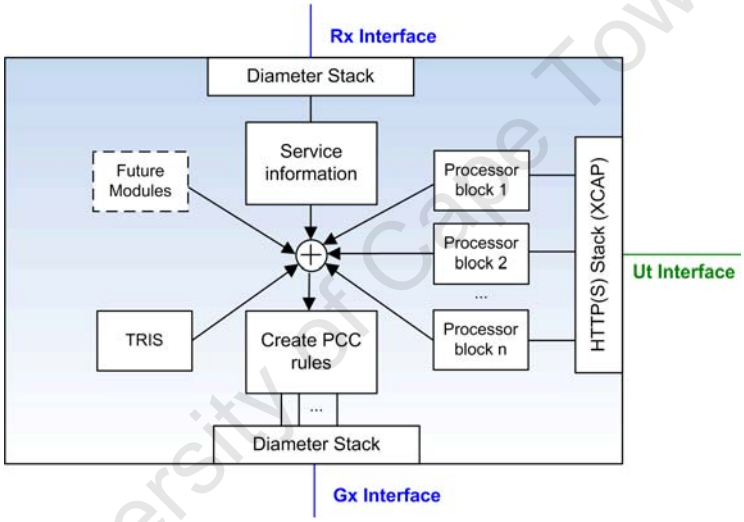

Figure 4.6: Logical architecture of extended PDF / x-RACF.

4.3.4 Element Interaction

To illustrate the proposed concepts and the extended interactions between elements, a basic IMS multimedia service scenario is demonstrated. In the scenario a single IMS domain comprising relevant CSCFs and HSSs interacts with a Common PCC framework. A Video on Demand (VoD) AS is deployed within the network that offers various different service configurations. A high-end service configuration streams requested content to the UE in High Definition format, and populates and distributes a multimedia-rich interactive Electronic Program Guide (EPG) as part of the same session. The standard configuration delivers content in Standard Definition format and includes a scaled down

text-based EPG with limited interactive functionality. Finally the low-end configuration streams content at very low resolutions, sufficient for dial-up access or mobile terminals with display constraints, and does not include an EPG component. IMS end-user UE A, connects via HSPA, is registered with the domain and is a valid subscriber to the VoD service. This is an example scenario used for illustrative purposes and the details of service implementation and execution are not considered important.

Referring to Fig. 4.7, UE A accesses and manipulates the VoD application control policy via the HTTP Ut interface (1, 2). In addition to the generic application policy attributes, the UE specifies supported resolutions and maximum uplink and downlink bandwidth. This configuration occurs out of band of IMS session setup signalling and is a once off event. UE A initiates a session by composing service requirements and encapsulating them in the SDP body of an INVITE request that is forwarded to the VoD AS via the IMS core elements (3, 4). The description of service requirements is highly granular and includes basic codec and bandwidth requirements. The VoD AS extracts the service information and retrieves the end-user component of the application policy specific to the requesting UE (5, 6). The VoD AS uses the service information and end-user set preferences to select the low-end service configuration as most applicable and creates a Diameter authorisation request using the service information, the end-user preferences and the known parameters for the low-end service configuration. This request is forwarded to the Common PCC framework (7).

To simplify the scenario the logical functions of the PDF and x-RACF are co-located. Upon receiving an authorisation request the PDF/x-RACF extracts the service information and retrieves the policy profile specific to the requesting UE from the policy repository (8, 9). The domain, subscription and access control policies are invoked for all requests. The policy profile defines filter criteria that, based on the service information, invokes the component specific to the low-end VoD service configuration. These selected control policies are downloaded from the policy repository (10, 11). Based on the priority specified in the policy profile the relevant policy processor blocks are called, first the domain, then subscription, then application, then access. The domain policy processor block uses general network constraints and information stored in the TRIS to perform admission control, and creates a technology independent enforcement policy based on the service information. The subscription policy processor block ensures that the UE is a valid and registered subscriber to the requested service and otherwise makes no changes to the enforcement policy. The application control policy specific to the low-

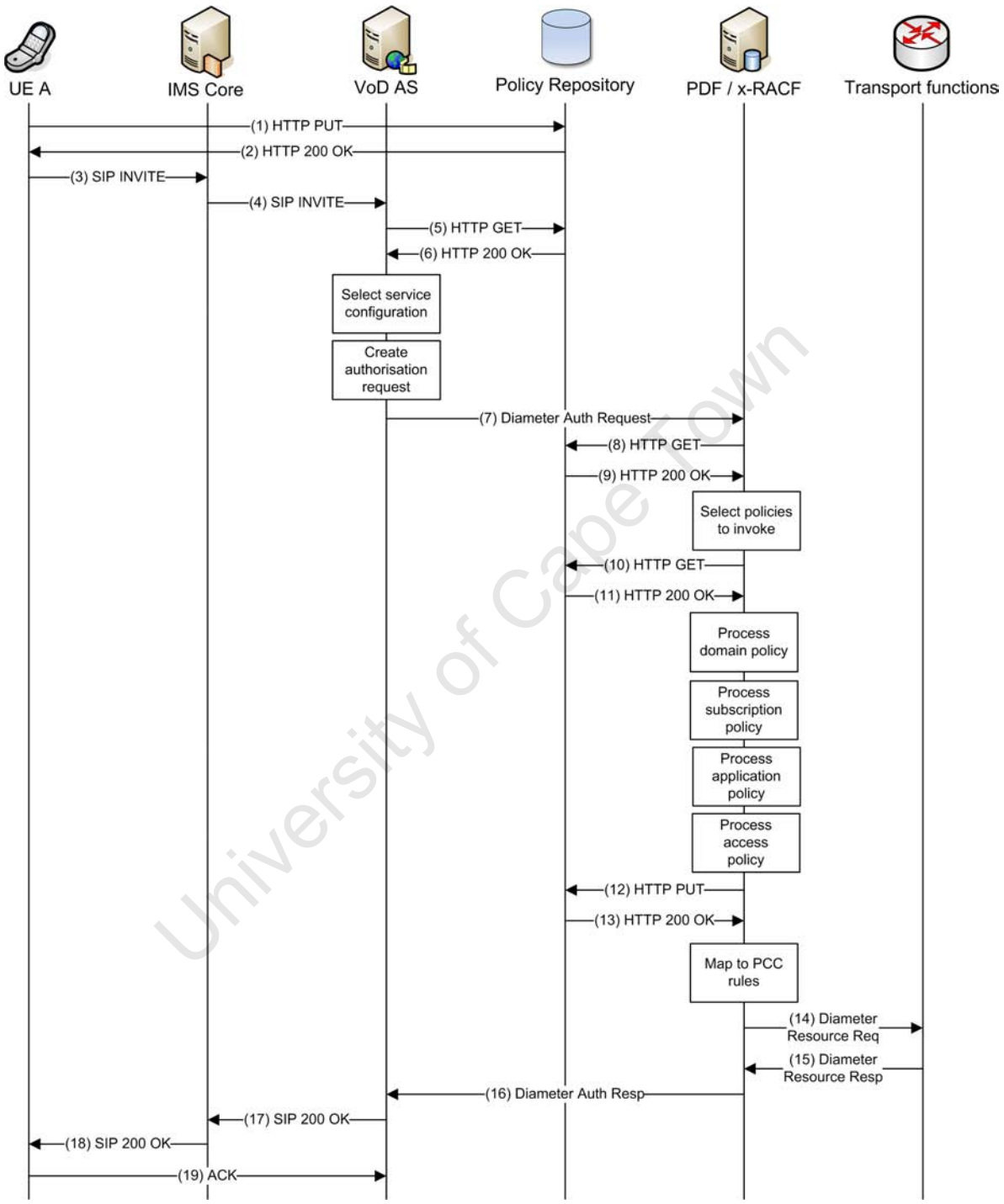

Figure 4.7: The signalling flow for a QoS enabled VoD service illustrates the proposed extensions and interactions.

end service configuration instantiates the generic application control policy and defines service requirements including constraints on bandwidth, jitter, delay and bit error rate in the uplink and downlink. This control policy further refines the enforcement policy by incorporating these constraints. The complete technology independent enforcement policy is further refined by the invocation of the access control policy specific to HSPA access. Based on this control policy a traffic classifier is incorporated into the enforcement policy and additional parameters are modified to further suit the access, e.g. path jitter, path latency, etc.

The full enforcement policy is stored in the policy repository for the duration of the session to assist with the maintenance of the TRIS, used for QoS reporting and resource monitoring (12, 13). The enforcement policy is mapped to PCC rules to be enforced on the transport functions via standardised interfaces. In the example scenario there are two Media-Component-Description AVPs included in the authorisation request and hence two PCC rules are created and populated. These rules are forwarded to the transport functions via the Gx reference points (14, 15). The result of the resource authorisation is conveyed back to the VoD AS (16), and a 200 Ok response is returned to UE A (17, 18). Note that the signalling for this scenario does not include the optional precondition extensions that are typically applied to basic IMS session initiation [68]. Its use is not necessary for this example and would add extra complexity. The resulting session (19) is tailored to the end-user preferences and terminal requirements, and takes place over a QoS aware transport plane ensuring acceptable end-user experience.

4.4 Discussion

This chapter has presented an application driven policy framework that incorporates end-user preferences and service requirements into QoS resource control through multi-layered and multi-policy interactions. The architecture defines generic application control policies, allowing new services to inherit advanced QoS negotiation support without any need for QoS standardisation. The application driven model introduces compelling business cases where 3rd party service developers could pay for the privilege of specifying minimum constraints in application policies and for access to more resources through higher priority policies.

The architecture is entirely conformant to Common IMS, Common PCC and relevant IETF specifications ensuring interoperability and extensibility. The framework design

is entirely modular, allowing new modules to be incorporated and older ones removed if necessary. The defined extensions are largely limited to the core network, minimising UE complexity and introducing no additional round trip times between the UE and core network ensuring minimal effect on end-user experience.

The description of this framework has thus far been limited to a single IMS domain. The next chapter details extensions to the end-to-end policy control mechanisms, which allow applications to effectively request resources from their home domain and enable end-to-end QoS connectivity across all traversed transport segments.

Chapter 5

Session Based End-to-end Policy Control Framework

The findings of the IMS/NGN resource management framework and literature review chapters show the end-to-end QoS provisioning mechanisms to be sorely lacking. The support for inter-domain reservations is rudimentary in all reviewed frameworks, with reference point definitions either undefined or in the early stages of development. While there is preliminary support for path-coupled reservation mechanisms used in conjunction with pull mode operation, no protocols have been standardised. The inter-domain approaches discussed in the literature are path-coupled algorithms or variations thereof, which suffer from poor compatibility with legacy transport plane devices. This chapter proposes extensions to the end-to-end inter-domain mechanisms that discover the signalling routes in the service control plane, and use this information to determine the paths traversed by the media at the resource control plane [69]. By operating at these planes, the proposed extensions are compatible with existing transport networks and exploit already existing mechanisms at the resource control planes.

The chapter presents design considerations pertinent to extending the existing inter-domain mechanisms to enable QoS connectivity in all traversed transport segments. The session based end-to-end policy control framework is presented and necessary requirements are discussed. The architectural extensions are described in detail and the route discovery and binding mechanisms are demonstrated. As with the proposed architecture in Chapter 4, the description of the end-to-end policy control mechanisms defines the logical elements of a reference architecture.

5.1 Design Considerations

Several considerations mentioned in the previous chapter apply equally to horizontal coordination of resources and providing seamless end-to-end QoS connectivity across administrative domains. These include standards conformance, particularly to Common IMS and Common PCC specifications, to ensure interoperability and extensibility. The extensions should not hinder the possible deployment of path-coupled QoS signalling solutions, like the IETF defined QoS NSIS Signalling Layer Protocol. The effect on end-user experience, most notably on session setup delay and signalling overhead, will be an important metric when verifying the proposed extensions. Scalability and the ability to rapidly deploy services without the need for QoS standardisation or network upgrade will also be critical. Further considerations, specific to the end-to-end policy mechanisms, are described in order of precedence.

5.1.1 End-to-end QoS connectivity across all traversed transport segments

To establish end-to-end connectivity the routes traversed need to be determined. Having full knowledge of the intra-domain topology is an efficient but practically infeasible approach. A first level of inter-domain routing, where domains are considered as black box nodes and only inter-domain routes are taken into account, is necessary [70].

The two specified end-to-end QoS scenarios in the Common PCC framework involve passing QoS information over the end-to-end path via QoS signalling, or through the application signalling via inter-domain reference points. The second approach performs resource reservation in the domains of the originating and terminating UEs but has no way of detecting transit domains and enforcing resource reservations in these domains, even if they incorporate IMS and Common PCC mechanisms.

The primary goal of the proposed extensions is to facilitate policy controlled resource management in all transport segments that support IMS and Common PCC functionality, thereby providing end-to-end QoS connectivity. In particular, the extensions aim to support typical roaming scenarios where a UE attaches to a visited network and signalling traffic is still routed via the home network, but media will typically make use of IP routing principles and follow the shortest path to the destination. The proposed architecture should reserve resources only in the domains traversed by the media.

5.1.2 Backward compatibility and reuse of existing resource management mechanisms

The gradual but inevitable move to an All-IP infrastructure has seen massive deployments of comprehensive networking equipment to serve the expanding requirements of end-users wishing to make use of Internet applications. These deployments have been costly in terms of capital and operating expenditure, especially for operators serving large geographical areas. In fact in some circumstances competing operators have cooperated when building long-distance networks to contain the increasing costs of deployment and maintenance [71].

While clean slate approaches to end-to-end QoS provisioning have the benefit of hindsight, they are limited in terms of practical applicability. Network operators heavily invested in transport plane technology will want to extract full return on investment before carrying out network wide upgrades to support path-coupled QoS signalling, especially when there will likely already be some form of traffic management mechanisms in place. Compatibility with existing infrastructure will be critical in the deployment of end-to-end QoS extensions, hence these extensions should be implemented at the IMS service control plane.

It is likely that NGN architectures supporting IMS service control will implement some form of resource control in most network segments [20]. It makes sense to exploit these already implemented mechanisms by implementing inter-domain policy extensions in the resource control plane. This would allow the reuse of autonomous and independent reservation mechanisms in each domain.

5.1.3 Home routed access

In the IMS framework advanced multimedia services are typically hosted and executed on Application Servers (AS) that can be located in the end-users home network or in an external third-party network, with which the home operator maintains a service agreement. In the application driven policy control framework described in the Chapter 4, ASs act as Application Functions (AF) and use application policies to incorporate end-user constraints and service requirements into authorisation requests.

In the Common PCC framework, elementary end-to-end QoS support facilitates resource reservation in originating and terminating domains; AFs create authorisation requests based on SDP information encapsulated in the signalling. This process uses

only the highly granular SDP information to create authorisation requests and does not inherit any of the benefits of advanced QoS negotiation described in the Chapter 4.

An important consideration for end-to-end policy extensions is to ensure that authorisation requests originate from the domain hosting the requesting AF where end-user preferences, known application configuration parameters and SDP information can be used to create the requests. In this home routed access, the transport functions are controlled by the home operator. A mechanism is needed to distribute the authorisation requests to all Common PCC frameworks managing transport segments traversed by the session.

5.2 Session Based End-to-end Policy Control

The end-to-end policy control extensions defined in this thesis are based on the architectural platform comprising the Common PCC framework and application driven policy control extensions, and the aforementioned design considerations. The proposed architecture aims to extend advanced QoS negotiation support to operate across administrative domains. The extensions are conformant to IMS and Common PCC specifications, though minor attributes are incorporated into standardised reference points. This means that any extensions that conform to these specifications can be incorporated.

Backward compatibility with legacy transport networks is ensured because end-to-end policy control occurs at the service and resource control planes. This approach allows the reuse of existing resource control mechanisms, maintaining the independence of each domain because operators can use reservation mechanisms of their choice.

5.2.1 Service Control Plane

Because the focus of this work is on advanced multimedia services, AFs are typically instantiated by ASs. The extensions are not limited to advanced services hosted and executed on ASs, and other elements, like the P-CSCF, could request resource authorisation. Depending on the service platform in use, ASs can host SIP native applications, act as Open Service Access (OSA)-Service Capability Servers with interfaces to the OSA application programmer interface (API), or as IP Multimedia Service Switching Functions (IMSSF) to allow the reuse of legacy Customised Applications for Mobile network Enhanced Logic (CAMEL) services. Regardless of the implementation, ASs behave as

SIP ASs towards the IMS network by implementing an IMS adaptor that interfaces with the S-CSCF using SIP [6]. This is important because the proposed extensions use routing information inherent in SIP signalling, to discover the end-to-end signalling path.

Because information in SIP signalling is used to determine traversed domains, it is important that this information be visible to all traversed proxies. For security, competition and privacy reasons an operator might want to hide configuration, capacity and topology information from outside the network. For this a Topology Hiding Inter-network Gateway (THIG) is used. The THIG is placed on the routing path when receiving requests or responses from other IMS networks and performs encryption and decryption of all SIP headers that reveal topology information [72]. The Interrogating - Call Session Control Function (I-CSCF), as the proxy located at the edge of an administrative domain, typically hosts the THIG. However even with this function in operation, the address of the gateway is always present and added to the routing information. This means that each gateway leaves a footprint in the signalling, which is sufficient to determine the traversed domains.

In the proposed framework only the AF in the home network of the originating UE creates authorisation requests that are conveyed to its Common PCC framework. These requests are processed and distributed to Common PCC frameworks in relevant domains at the resource control plane. This simplifies the architecture and allows it to inherit the benefits of the application driven concepts introduced in Chapter 4, including services tailored to end-user preferences and application refined PCC rules.

The main scenario that these extensions aim to support, is the typical roaming scenario where UEs attach to visited networks but still have home located service control. In this scenario the media and signalling originate and terminate in the same domains, but follow distinctly different paths between source and destination. In more complex scenarios involving multiple transit domains, it is necessary that the signalling be constrained through IMS CSCFs in each domain to provide the necessary routing information for processing at the resource control plane. This can be achieved, for example, by defining static routes at the S-CSCF. While this introduces domain routing at the service control plane, it ensures that legacy transport networks can be reused and that each operator can implement their own resource reservation mechanisms. For the typical roaming scenario this signalling routing is not necessary.

The proposed mechanisms involve minimal processing at the AF, besides extraction and interpretation of SIP header information, which is already supported at these ele-

ments. Hence the extensions do not compromise the performance of the high performance AFs and are highly scalable.

5.2.2 Resource Control Plane

The resource control plane is extended to receive signalling path information encapsulated in authorisation requests, and to use this information to discover the path traversed by the media. The information is processed at the PDF and unnecessary transit domains traversed by the signalling but not by the media, governed by IP routing principles, are excluded from the resource management procedure.

Operators that agree to carry other operators traffic will typically negotiate and exchange Service Level Agreements (SLA) that establish committed levels of network service performance and responsiveness. As part of this agreement the Diameter Uniform Resource Identifier (URI) of the top level PDFs should be exchanged. In this way all PDFs know the contact URI of PDFs in neighbouring domains with negotiated SLAs. This is important as the mapping between the signalling and media paths relies on this knowledge. Alternatively these addresses could be manually configured or discovered automatically using topology discovery mechanisms [49].

When more than one inter-domain path exists between source and destination, the best path must be selected based on common parameters such as resource cost along the path, availability of security associations and SLAs between domains. This is an ongoing research area and is beyond the scope of this thesis. For simplicity only scenarios with single inter-domain paths are discussed.

Because authorisation requests are received from AFs in the home domain of the originating UE only, application policies can be incorporated at the home domain Common PCC framework when creating inter-domain resource requests. These inter-domain resource requests can be further refined at each local Common PCC framework where policies specific to each domain can be applied to the final PCC rule to be installed. This inherits the benefits of application policies while still allowing local policies to be incorporated into QoS resource control, and the use of autonomous resource management in each domain. It also negates the need for distributed policies across domains, which can be a complex and contentious task.

5.3 Solution Architecture

The end-to-end session based policy control architecture is illustrated in Fig. 5.1. A typical roaming scenario involving two UEs is demonstrated. Both UEs attach via visited networks but still route signalling via their home networks. The AF in the home domain of the originating UE, UE A, is extended to calculate the signalling path from routing information inherent in the SIP signalling. The interface between the service and resource control planes is extended to include the signalling path information, and the PDF in the home Common PCC framework determines which domains require resource reservation.

The extensions allow applications to effectively issue resource requests from their home domain, enabling QoS connectivity across the end-to-end path in all domains that incorporate IMS and Common PCC mechanisms. Autonomous resource management is maintained in each domain, legacy transport networks are supported and sharing of potentially sensitive topology information is not necessary.

5.3.1 Signalling Path Discovery

In RFC 3261, *SIP: Session Initiation Protocol* [73], Rosenberg *et al.* define a transaction as a request, zero or more provisional responses and a final response, while a dialog is defined as a SIP relationship that persists for some time and may include a number of transactions. All SIP responses must traverse the same elements as their related requests, this is achieved using the *Via* header. The address of each element the SIP request traverses (including the originating UE) is added to the *Via* header in the order in which they are reached. The response in turn traverses all elements listed in the *Via* header and as each element is reached its address is stripped from the header.

Subsequent transactions within a dialog need not follow the same path, however some elements, for example CSCFs, might want to ensure that they are included on the path of all messages in the same dialog. CSCFs must be traversed by all messages in the same dialog for charging, subscription, service invocation and policy control purposes. This is achieved using the *Record-Route* header. Any element that wants to be included on the path for all requests in that dialog, adds itself to the *Record-Route* header of the initial request. When subsequent requests are created, the elements listed in the *Record-Route* header are included in the *Route* header in the order in which they will be traversed. The *Route* header essentially defines the route a request message must take. The first request of a dialog discovers the necessary routes and the element addresses are stripped

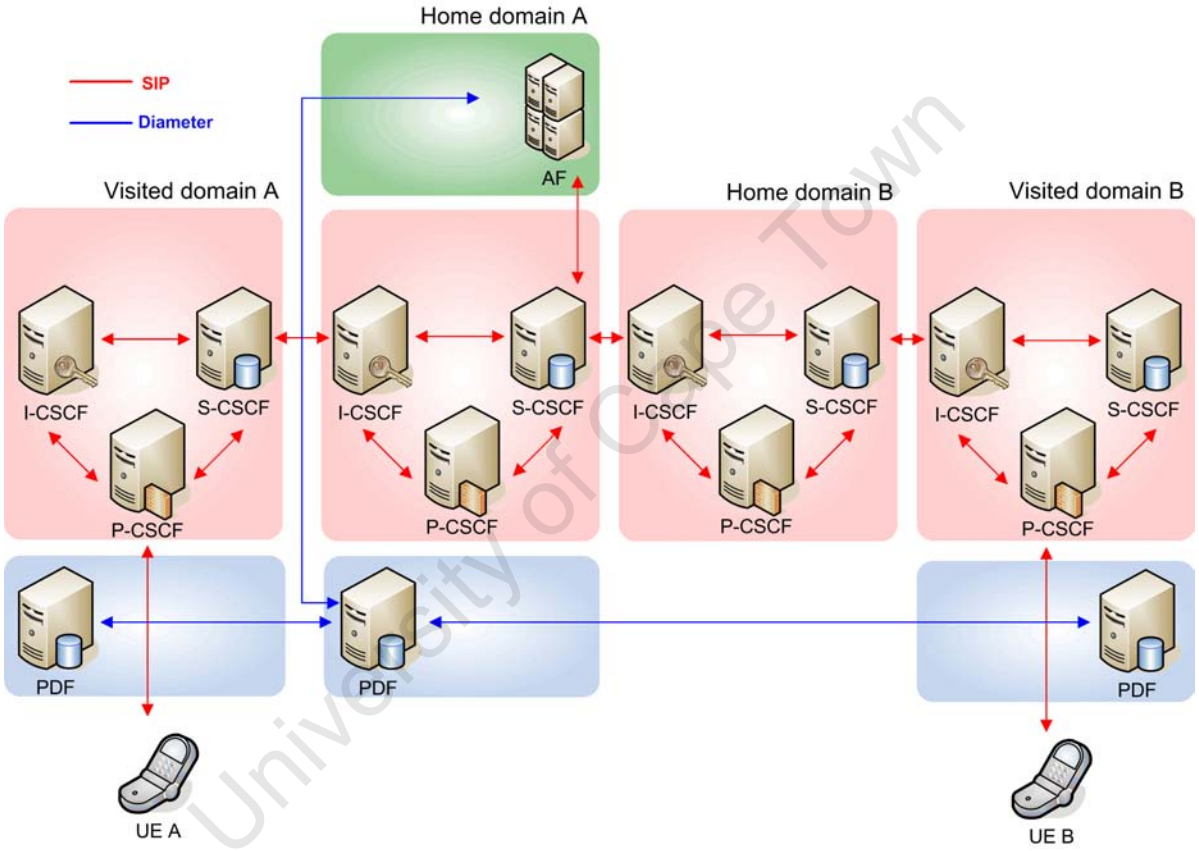

Figure 5.1: The solution architecture for end-to-end policy control in a typical roaming scenario.

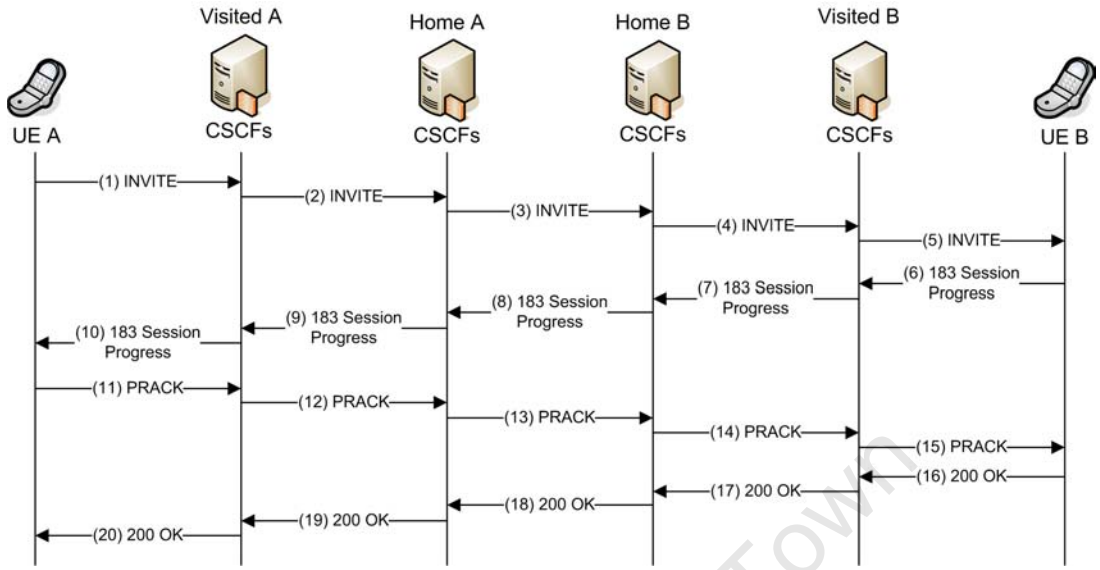

Figure 5.2: The end-to-end signalling path can be determined from the *Route* and *Via* headers of the subsequent PRACK request.

from the *Route* header as they are traversed. As mentioned earlier, when each element is traversed its address is added to the *Via* header of the request message to ensure that its response follows the correct path. Hence for any subsequent request, the entire signalling path is mapped out by the *Via* and *Route* headers.

The described principles are illustrated using an IMS session initiation example. In this case the optional precondition extensions are pertinent to the explanation. Precondition extensions introduce constraints that must be met before a session can be established. Essentially additional request/response exchanges are included within the session initiation signalling that allow service parameters to be negotiated and ensures that resources are reserved in originating and terminating domains. Fig. 5.2 shows an extract of IMS session initiation signalling, for simplicity only the first two request/response exchanges are depicted.

UE A in visited domain A, creates an INVITE request and sends it to UE B in visited domain B. The signalling follows the path: visited domain A, home domain A, home domain B, visited domain B. The CSCFs in all domains add themselves to the *Record-Route* and *Via* headers as they are traversed (1-5). UE B replies with a 183 SESSION PROGRESS response that traverses all elements listed in the *Via* header (6-10). The subsequent PRACK request has its *Route* header populated by all elements that included themselves in the *Record-Route* header of the previous request. As the PRACK

request traverses each element in the order they are listed in the *Route* header (11-15), the element addresses are stripped from the *Route* header and appended to the *Via* header. Hence at any element along the path the entire route from originating UE to terminating UE can be determined by examining the *Route* and *Via* headers of the PRACK request.

The session based end-to-end policy control framework extends the AF to extract originating, transit and terminating domains from the SIP signalling information and to pass this information to the PDF in the resource control plane. The *Route* and *Via* headers contain element addresses, however it is possible to extract the domains of these elements by taking only the network part of the URI. When an AF receives a SIP response it examines the corresponding SIP request; if the request was the first in the dialog the last entry in the *Via* header is extracted and listed as originating domain and the terminating domain is taken from the Request URI. Because it is an initial request and routes are still being discovered, full path information is not known, hence only the originating and terminating domain can be determined.

When the AF receives a response to a subsequent request it examines the request and can determine the end-to-end signalling path. All information is extracted from the *Route* and *Via* headers. The last entry in the *Via* header is stored as originating domain and the last entry in the *Route* header is stored as terminating domain. The remaining domains are included as transit domains in the order in which they are traversed by the signalling, though duplicate domains are discarded. This information is encapsulated in authorisation requests that are sent to the PDF in the resource control plane. To convey this information three new Attribute Value Pairs (AVP) are defined for the Diameter Rx Application of the Common PCC framework with attribute names *Originating-Domain*, *Transit-Domain* and *Terminating-Domain* and associated AVP codes 5xx. The logical functions of the extended AF are illustrated in Fig. 5.3.

5.3.2 Media Path Discovery

Once an authorisation request is received by the PDF in the resource control plane the media path needs to be determined. The signalling path information encapsulated in the request includes all IMS domains traversed, while media is decoupled and will typically traverse an optimised path between originating and terminating domains. The PDF has been extended to implement inter-domain QoS resource control. There are three cases

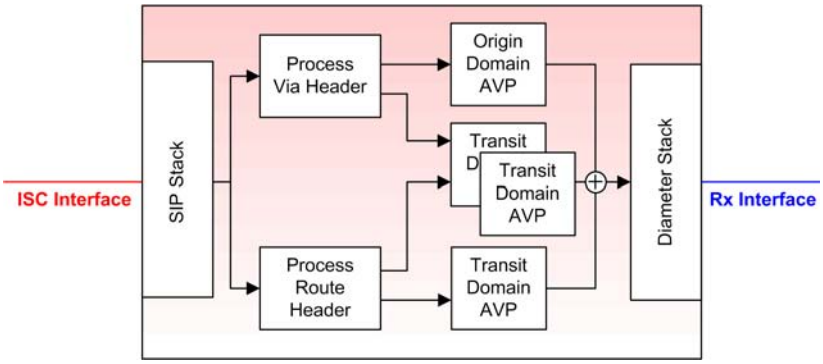

Figure 5.3: The extended AF maps out the signalling path and passes this information to the resource control plane.

to cater for:

Originating and terminating domains are the same

In this case the service data flow is confined to a single domain, and all transit domains may be excluded from the resource management procedure as they do not carry any media. The PDF in the home network of the originating UE performs QoS resource control incorporating application control policies, but instead of creating individual PCC rules to be enforced in the transport plane, an inter-domain resource request is created based on the Diameter S9 application [31]. This request is forwarded to the domain in question, where local policy control is performed and PCC rules are created and enforced in the transport plane, using reservation mechanisms specific to that network operator.

Originating and terminating domains are different and there are no transit domains

In this scenario the service data flow takes place across two domains and the signalling path does not cross any transit domains. The PDF creates inter-domain requests, incorporating application control policies, and sends them to both domains where local QoS resource control is performed. These requests are sent and processed simultaneously and thus have a limited effect on session setup delay.

Originating and terminating domains are different and transit domains exist

In this case it needs to be determined whether the included transit domains are traversed by the media, or if they can be discarded as unnecessary transit domains along the signalling path. The PDF in the home domain of the originating UE detects this case and forwards the request, as is, via the Diameter S9 interface to the PDF in the originating domain. The address of this PDF would be known as it would have been exchanged in the roaming agreement that allows the UE to attach to this network.

From the originating domain the PDF in the resource control plane discovers the path traversed by the service data flow. The PDF examines the signalling path information in reverse order, from terminating domain to originating domain. When a neighbouring domain, whose PDF address is known, is discovered, all subsequent transit domains are removed from the authorisation request. This new request is forwarded to that neighbouring PDF where the same process is repeated. This operation repeats until the terminating domain is reached. In this way domains traversed by the media are discovered and unnecessary transit domains are discarded.

5.3.3 Element Interaction

To illustrate the session based end-to-end policy control concept and extended interactions between elements, the typical roaming example depicted in Fig. 5.1 is extended and demonstrated. In this scenario there are two UEs, UE A connects via visited domain A and is registered with home domain A, and UE B connects via visited domain B and is registered with home domain B. Home domain A has an agreement with visited domain B that it will carry its traffic. During typical session initiation, signalling will traverse visited domain A, home domain A, home domain B, visited domain B. Because of the agreement between visited domain B and home domain A, the service data flow will be carried over visited network A, home network A and visited network B.

The path discovery mechanisms are illustrated in Fig. 5.4 where UE A initiates a typical IMS session with UE B; for simplicity abbreviated signalling is shown. To initiate an IMS session UE A creates a SIP session request that traverses visited domain A, home domain A, home domain B, visited domain B and is eventually delivered to UE B (1-5). UE B returns a SIP session response with preferred service parameters (6-8). The extended AF in home domain A, the home domain of the originating UE, extracts this service information to create an authorisation request. Included in this request are the

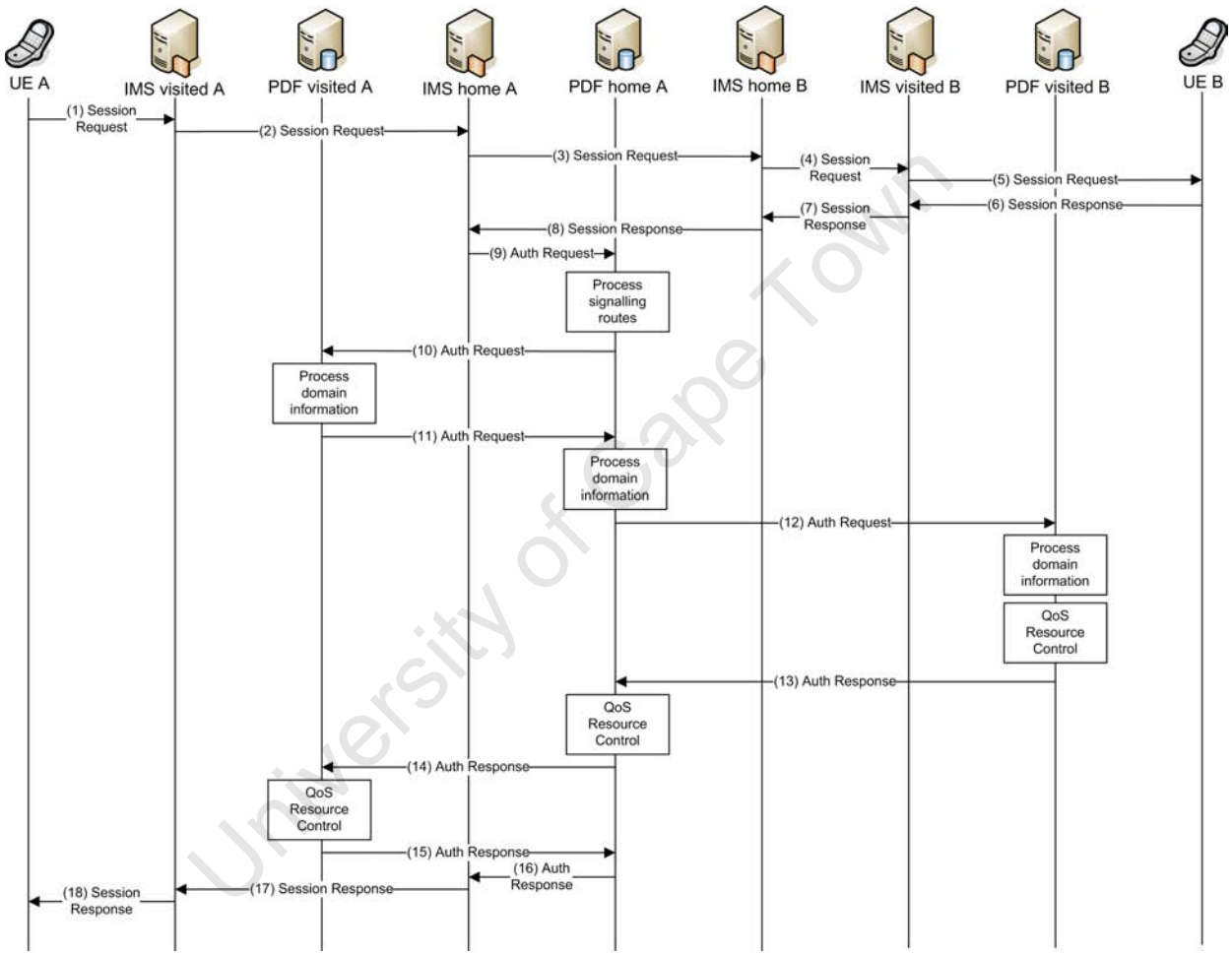

Figure 5.4: A typical roaming scenario demonstrates how the signalling and media paths are discovered.

additional *Originating-Domain*, *Transit-Domain* and *Terminating-Domain* AVPs. Table 5.1 shows the signalling path calculated from SIP routing information. The authorisation request is sent to the PDF in home domain A (9). Along with the normal QoS resource control, the route information is processed and it is discovered that originating and terminating domains are different, and there are transit domains. The request is forwarded to the PDF in the originating domain, visited domain A (10), based on the already described algorithm.

The PDF in visited domain A examines the routing information in reverse order. The address of the PDF for the terminating domain, visited domain B, is unknown, as is the address of the PDF for the first transit domain, home domain B. However the PDF in visited domain A does know the address of the PDF in home domain A, as it would be exchanged in the roaming agreement that allows UE A to attach to visited network A. Home domain A is removed from the list of transit domains and the new authorisation request is forwarded to the PDF in home domain A (11). The same process takes place again, though this time the address of the PDF in visited domain B is known because of the agreement between these two domains. The subsequent transit domain, home domain B, is removed from the list and the authorisation request is forwarded to the PDF in visited domain B (12). Visited domain B is the terminating domain, which means that the process of media path discovery is complete. Based on the request and local operator policies, QoS resource control is performed and PCC rules are created and installed in the transport plane (omitted for simplicity). The response to the authorisation request is conveyed to the PDF in home domain A (13) and the PDF in visited network A (14), where similar QoS resource control takes place. Eventually a positive authorisation response indicating successful end-to-end resource reservation is conveyed to the PDF in home domain A (15), where it is forwarded to the service control plane (16). The SIP session response is delivered to the originating UE, UE A (17, 18), and the resulting session has resources reserved along all traversed transport segments, enabling end-to-end QoS connectivity. Table 5.1 shows the subsequent calculated media paths.

In the worst case scenario with different originating and terminating domains and transit domains, the discovery of the media path comes at the cost of increased session setup delay as the PCC rules are not enforced in each domain simultaneously. However for all other scenarios resource reservation is performed simultaneously in each domain.

Table 5.1: The signalling and media paths determined from the SIP Request.

```
UPDATE sip:alice@10.128.10.1:5060 SIP/2.0
Via:  SIP/2.0/UDP pcscf1.visitedA.net;branch=z9hG4bt6g
Via:  SIP/2.0/UDP 10.128.10.1:5061;branch=z9hG4bK75
Route:  <sip:  scscf1.homeA.net;lr>
Route:  <sip:  scscf2.homeB.net;lr>
Route:  <sip:  pcscf2.visitedB.net;lr>
```

Administrative domains traversed by signalling

Originating-Domain	Transit-Domain	Terminating-Domain
visitedA.net	homeA.net, homeB.net	visitedB.net

Administrative domains traversed by media

Originating-Domain	Transit-Domain	Terminating-Domain
visitedA.net	homeA.net	visitedB.net

5.4 Discussion

The chapter has detailed extensions to the Common PCC framework that allow applications to effectively request resources from their home domain, enabling QoS connectivity across multiple domains. The proposed architecture discovers signalling routes at the service control plane and uses this information at the resource control plane to determine which domains are traversed by the media and hence need to have resources reserved.

The proposed framework allows resource requests to originate from the domain hosting the requesting AF, and to be distributed across domains. Thus the application driven concepts described in Chapter 4 can be incorporated into inter-domain QoS resource control. The extensions ensure backward compatibility with legacy transport networks and allow autonomous and independent resource reservation in each domain. The sharing of potentially sensitive topology information is not necessary to determine the end-to-end path traversed by the service data flow.

The proposed extensions to the Common PCC framework, to facilitate application driven policy control and end-to-end resource reservation, are conformant to relevant

standards and have been described as reference architectures only. The next chapter details a practical implementation of the proposed architectures to show proof-of-concept and provide a platform for evaluations.

Chapter 6

Implementation of an Evaluation Framework

Previous chapters have described the resource management framework specified for mediating between the service control and transport planes in the IMS model. Extensions to the Common PCC framework to incorporate vertical and horizontal coordination of resources, allowing rapid deployment of services with advanced QoS negotiation support and inter-domain resource reservation mechanisms, have been detailed. However these extensions only define a reference architecture with logical functions and not a physical implementation. To verify that the proposals are feasible in typical deployment scenarios, they need to be implemented in a practical testbed where evaluations can be carried out to show the reliability and performance of the framework in a realistic environment. A simulation of the proposed architecture will typically be subject to a number of assumptions that, though theoretically accurate, could simplify the problem and not take into account all the variables of a practical network. In a testbed environment deployed technologies can be proved almost beyond doubt.

The University of Cape Town (UCT) has several open source projects that form part of the IMS research initiative [74]. The testbed deployed at UCT covers various realms of IMS research, from application and client development, to charging and policy control architectures. This chapter details the deployment of a standards compliant IMS testbed and the software tools created as part of this thesis. Limited resources and ease of extensibility to incorporate new modules were important considerations during implementation. Emphasis is placed on the concept of open testbed platforms because of the early deployment and testing stages of IMS technology, and resource management

frameworks in particular. The described testbed comprises entirely Free and Open Source software (FOSS), licensed under the GNU General Public Licence (GPL), or variations thereof, opening up the system development to any users of the framework. This means that the testbed components can be easily and legally extended to incorporate future modules, which is critical for young technologies.

This open testbed exposes the complex concepts and components associated with IMS technology and resource management frameworks to a wide set of developers. From the research point of view, the testbed facilitates rapid prototyping and experimentation with new ideas, and provides a safe and controlled area where technologies, protocols and applications can be analysed, separate from the live environment and associated hazards. It also encourages innovation in the field by ensuring reproducibility, and provides a convenient point of departure for future research.

6.1 Testbed Components

A number of core components are necessary for the proposed extensions to be implemented and verified in a practical environment. The IMS defines a large number of functional elements including gateways to other domains, media originating functions and breakout elements for legacy networks. Only those pertinent to the scope of this thesis form part of the testbed implementation. Essentially an IMS core network, comprising core Call Session Control Functions (CSCF) and a Home Subscriber Server (HSS), is necessary. Such a core network should support full IMS registration, session control and the provisioning of Application Servers (AS). It should be possible to create initial Filter Criteria (iFC) to invoke services by selectively forwarding SIP requests to the ASs.

For the purpose of this thesis, the application layer is only necessary to demonstrate the proposed mechanisms and how QoS requirements are created and enforced in the transport plane. Hence a generic Application Function (AF) is required to intercept SIP signalling, incorporate the application driven and end-to-end concepts, and mediate with the resource control plane.

It is also of interest to investigate the impact that the proposed architecture has on the performance of particular applications. A practical IMS Video on Demand (VoD) AS is deployed to permit this examination. The VoD service is chosen for several reasons. This service is driving consumer interest in Internet Protocol Television (IPTV) technology [75]. VoD is typical of the multimedia and content-rich services envisaged in an

IMS network, requiring stringent control of resources. It is a fully standardised service with a comprehensive IMS-based IPTV architecture defined by TISPAN [76]. A VoD AS that implements content on demand is necessary to determine, in conjunction with the generic AF, the management overheads introduced by the Common PCC extensions.

A Common PCC framework is required, including a Policy Decision Function (PDF) and relevant transport functions. The focus of the thesis is the interaction between the service and resource control planes, and mechanisms to provide end-to-end QoS across administrative domains. Hence transport functions can be elementary in implementation, though full signalling, including the provisioning of PCC rules, should be supported. A policy repository that interacts with the service control and resource control planes is required to store and manage the XML-based policies.

To demonstrate the principles in practical scenarios, UEs will be needed to register with the core network, initiate sessions, terminate sessions, and handle media and user interaction. The UE should support full IMS signalling, including precondition extensions necessary for the proposed session based end-to-end policy control architecture. It should also be able to interact with the policy repository, supporting the storage, retrieval and deletion of XML documents. Finally IP-Connectivity Access Networks (IP-CAN) are necessary to interface between the IMS terminals and the core network and services. IP-CANs typical of a practical IMS network should be employed. Standards conformance is of critical importance to ensure that evaluations accurately mirror the expected performance of the proposed extensions in a real world IMS deployment. Fig. 6.1 shows a high level abstraction of the testbed components; these components are complex and may be made up of a number of physical elements.

6.2 IMS Core Network and Application Layer

The primary elements that make up an IMS core network, as specified in the 3GPP Common IMS standard, are the Proxy-CSCF (P-CSCF), the Interrogating-CSCF (I-CSCF), the Serving-CSCF (S-CSCF) and the HSS [63]. These are implemented in the testbed using the Fraunhofer FOKUS Open Source IMS Core (OSIMS) [77]. This project was first released in November 2006 as free and open source under GPL version 2 (GPLv2). The project is based on 3GPP specifications, widely used and under constant active development by a large base of developers. It has incorporated extensions defined by 3GPP2, ETSI TISPAN and the PacketCable initiative. These software tools were chosen

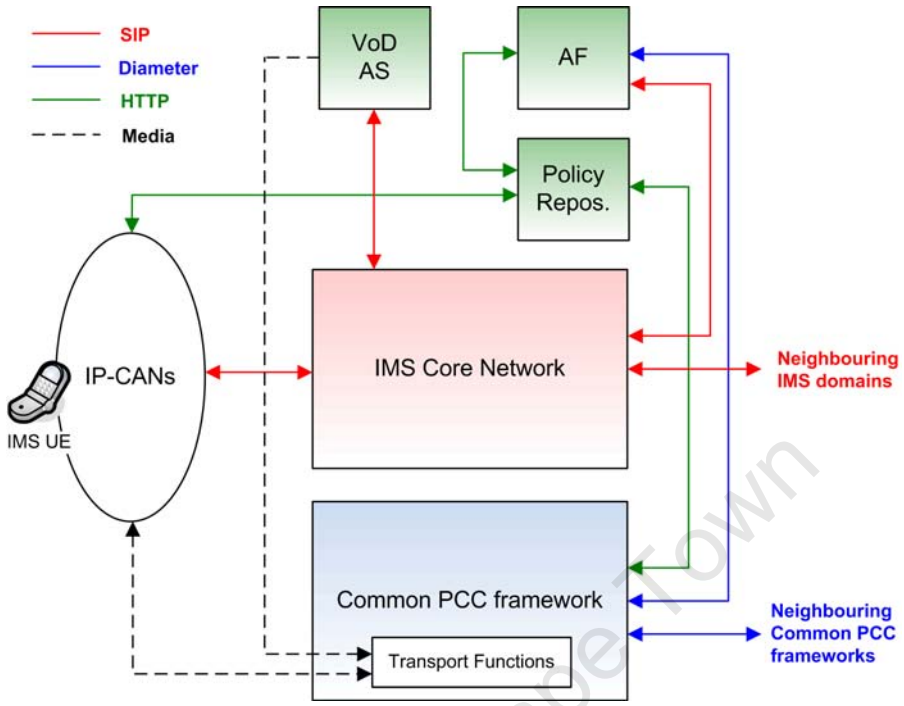

Figure 6.1: Evaluation framework high level components.

for the testbed as they are open source, reliable, standards conformant and highly efficient regarding signalling volumes. The OSIMS runs on the Linux platform; the Ubuntu Linux operating system is used throughout the testbed implementation [78].

6.2.1 Call Session Control Functions

The OSIMS CSCFs are based on the SIP Express Router (SER) [79], which has recently joined the SIP Router Project, a common collaboration framework for all projects related to SIP proxies [80]. The power of SER lies in its performance, flexibility and interoperability. SER runs extremely well under heavy load due to large subscriber populations or abnormal operational conditions, its modular interface allows a high degree of customisation, and its strict conformance to RFC 3261, *SIP: Session Initiation Protocol* [73], makes it an ideal base for the IMS CSCF extensions. The CSCFs extend the SER code base to provide necessary specialised functionality; SER and the CSCF extensions are written in the C programming language, known for high performance in delay sensitive applications. The CSCFs are highly scalable and can run simultaneously on a standard desktop machine or can be distributed across a number of high performance servers.

This means that if a Domain Name Server (DNS) is configured accordingly, a number of IMS domains can be interconnected with minimal resource costs.

6.2.2 Home Subscriber Server

The FOKUS Home Subscriber Server (FHoSS) is a lightweight HSS implementation that forms part of the OSIMS project. The FHoSS is implemented in the Java programming language and uses SQL databases for data storage. Registration and the creation of iFCs are two important procedures facilitated by the FHoSS that are critical to this thesis.

IMS registration is performed to bind the UEs IP address to the UEs Public User Identity (IMPU). When somebody wants to contact a registered UE they only need the IMPU (e.g. *sip:bob@qos-ims.test*) and require no knowledge of the UE terminal or IP address. Registration also verifies the UE and the network, and downloads the end-user service profiles from the HSS to the S-CSCF. For authentication purposes each end-user has at least one Private User Identity (IMPI). An end-user can have several IMPIs and IMPUs associated with their IMS subscription but for the purpose of this testbed each end-user is allocated a single IMPI and IMPU per subscription. During registration the S-CSCF downloads an authentication vector from the HSS that is used to challenge the registering UE. Based on the IMPI, the UE creates a response, which results in registration, once verified. The creation of the challenge and corresponding response depends on the authentication protocol in use; the testbed uses the AKAv2-MD5 protocol exclusively [81], as specified by the 3GPP Common IMS specification, though the FHoSS supports a wide range of authentication protocols.

Filter criteria are included as part of the end-user service profile stored on the FHoSS. These contain the collection of triggers that determine whether a request has to traverse one or more ASs that provide services to the end-user. The FHoSS incorporates a web interface that allows an operator to easily add, remove or modify end-users and ASs. It also allows filter criteria to be added to a particular session, which is necessary in the testbed to forward relevant requests to the AF where end-to-end resource authorisation is performed. The triggers are enforced as a result of pattern matching on any SIP header or body. Fig. 6.2 illustrates a trigger point that forwards all INVITE requests that originate from a UE in a specific domain to the generic AF for resource reservation; in the end-to-end policy control architecture authorisation requests are only created in the home domain of the originating UE. Together the FHoSS and the CSCFs provide a

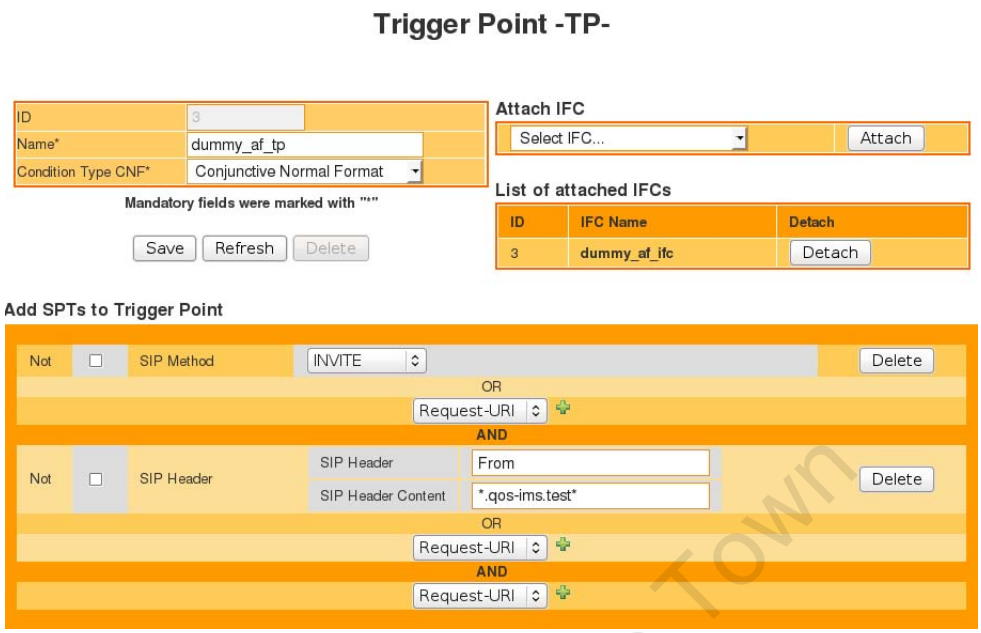

Figure 6.2: The FHoSS web interface allows easy configuration of iFCs and trigger points.

reliable and realistic IMS testing environment.

6.2.3 Application Function

A generic AF in the application layer is necessary to incorporate end-user preferences into authorisation requests and to customise service delivery. This element is selectively added to the signalling path based on iFCs. It needs to extract service and routing information from SIP messages, and to create authorisation requests for the Common PCC framework. Additionally it needs to implement the extended interactions with the XML Document Management Server (XDMS) policy repository and to calculate end-to-end signalling paths. A native SIP AS is used to instantiate this element. This approach, as opposed to the use of complex service creation APIs, allows lightweight projects to be rapidly deployed with little more than a SIP stack. The generic AF can operate in a number of different modes depending on the evaluation scenario. It can act as a terminating UE, an originating UE, a SIP proxy, or a Back to Back User Agent (B2BUA). Fig. 6.3 demonstrates the different modes of operation.

The SIP functionality of the generic AF is based on the oSIP [82] and extended oSIP (eXosip) libraries [83], both released under the GNU Lesser GPL (LGPL). These libraries, and hence the generic AF, are written in the C programming language. To-

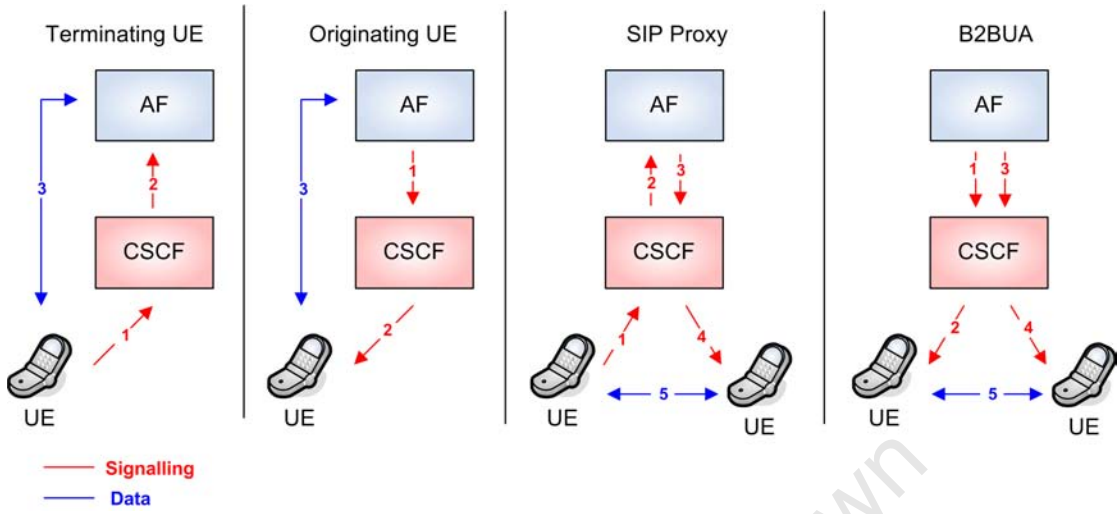

Figure 6.3: The generic AF can act as an originating UE, terminating UE, SIP Proxy or B2BUA.

gether these libraries provide a mix of high level abstraction and low level manipulation that allows the AF to implement different modes of AS operation and to access and modify the SIP header and SDP body information. This information is used to create authorisation requests and to calculate the end-to-end signalling path.

CDiameterPeer is a lightweight implementation of RFC 3588, *Diameter Base Protocol* [84], and is a component of the OSIMS project. This module was originally written in the C programming language as a module for SER, but can be used as a stand-alone protocol implementation. This module implements the base Diameter protocol and is extended to instantiate the Rx interface between the generic AF and the Common PCC framework. New AVPs, including those defined in Chapter 5, are incorporated and used to pass authorisation requests from the service control plane to the resource control plane.

The generic AF uses the libcURL library, released under an MIT/X derivative licence that allows free modification and distribution of source code, to interact with the XDMS server [85]. This library instantiates the Ut interface, and allows the generic AF to store, retrieve and remove application control policies on the XDMS server using HTTP PUT, GET and DELETE methods. The XML content of the HTTP messages is created and parsed by the libXML2 library, also released under the MIT licence [86].

For testing purposes it is required that sessions traverse the generic AF in the home domain of the originating UE. This is achieved using iFCs at the S-CSCF, described

in Section 6.2.2. Depending upon the mode of operation the AF will process the SIP signalling differently. Acting as a SIP proxy is the most typical configuration for the testing scenarios in this thesis. In this case when a session matches the trigger points described in the iFC, the S-CSCF adds the address of the generic AF to the *Route* header. This ensures that the message is forwarded to the AF based on standard SIP loose routing rules. The S-CSCF also adds its own address as a second *Route* header to ensure that the request is routed back to itself; state information in this second *Route* header ensures that a routing loop is avoided. The generic AF adds itself to the *Via* and *Record-Route* headers ensuring that all responses and subsequent requests in the same dialog, traverse the proxy. It also removes itself from the *Route* header to prevent the request from being routed back to itself.

6.2.4 VoD AS

The primary elements of the TISPAN defined IPTV architecture are the Service Control Function (SCF), the Media Control Function (MCF) and the Media Distribution Function (MDF) [76]. The UCT Advanced IPTV project is a bundled implementation of these functions that is fully compliant with the TISPAN IMS-based IPTV stage 3 architecture [87]. The architecture supports both broadcast and content on demand services, though in the framework only content on demand is demonstrated. The SCF allows a UE to initiate and negotiate a VoD session and selects the relevant media functions. The MCF provides functions that allow a UE to control media flows using the Real Time Streaming Protocol (RTSP), while the MDF is tasked with the actual distribution of the media. This project is part of the UCT IMS initiative and is released free and open source under GPL version 3 (GPLv3). The VoD AS can be selectively added to the signalling path based on iFCs. It can be used, in conjunction with the generic AF, to evaluate a practical IMS service with incorporated application driven policy control and end-to-end QoS mechanisms.

6.3 Common PCC Framework and Policy Repository

The primary elements of the Common PCC framework are the PDF, the x-RACF, the policy repository and related transport functions. When the OSIMS project was first

released in November 2006, there were no open source resource management implementations available, nor were there any core extensions to support resource authorisation. The UCT Policy Control Framework was developed by the author of this thesis to fill this gap and to provide IMS enthusiasts an opportunity to experiment with resource management and associated signalling [88]. The implementation was released free and open source under GPLv3 in December 2007, and has since grown in terms of robustness and scale.

This framework implements a co-located PDF/x-RACF element, a policy repository and elementary transport functions. It is written in the Java programming language and provides an associated web management interface. Additionally a scaled down version was implemented in the C programming language for high performance load testing. The framework was designed to be used in conjunction with the OSIMS. However it is able to interact with any subsystem that implements the standardised reference points. As with the OSIMS, the architecture is highly scalable and can run on a single desktop machine. This allows multiple domain Common PCC frameworks to be deployed with limited resources.

6.3.1 PDF / x-RACF

The PDF and x-RACF, as elements within the proof of concept testbed, are physically co-located. The behaviour of this element depends on its role in the QoS resource control procedure. As the PDF in the home domain of the originating UE, this element receives and processes authorisation requests from the generic AF. It interacts with the XDMS and retrieves policy profiles and relevant policies to create PCC rules. Additionally this element creates inter-domain authorisation requests to ensure end-to-end connectivity, as detailed in Chapter 5. Once the end-to-end resources are reserved, the created PCC rules are conveyed in Diameter resource requests to the transport plane functions.

When acting as the PDF in the transit domains, this element receives inter-domain authorisation requests and processes them based on basic domain policies. The PDF forwards the request to the PDF in the next domain on the media path, as detailed in Chapter 5. Once resources have been reserved on the subsequent end-to-end path, the created PCC rule is passed to the transport plane functions. The PDF in the terminating domain plays a similar role, but also incorporates relevant subscription control policies into the QoS resource control procedure. This ensures that the terminating UE is a valid

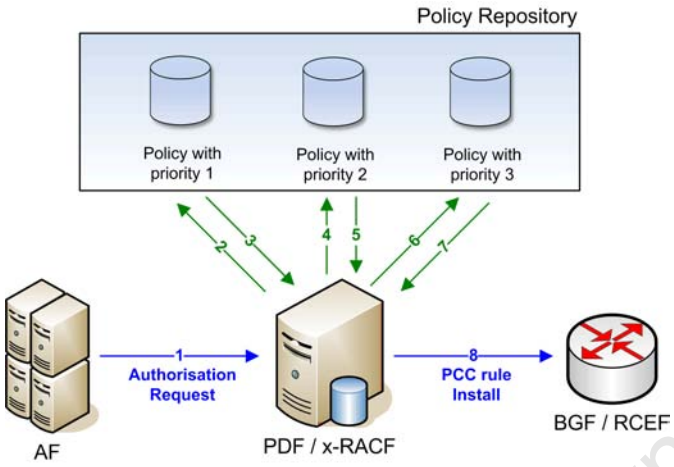

Figure 6.4: Control policies are invoked in a serial fashion based on the priority specified in the end-user's policy profile.

and registered subscriber to the requested service.

The PDF is implemented in the Java programming language. This allows for rapid prototyping of new concepts and elements, incorporates object oriented design and allows easy extensibility. JavaDiameterPeer is a lightweight implementation of the Diameter protocol and a component of the OSIMS project. This library forms the core of the PDF element. Additional AVPs and Diameter messages were created to instantiate the Rx, S9 and Gx interfaces. A state machine defines the behaviour of the element based on the information received in the requests. The Jakarta Common HTTPClient is used to create the necessary HTTP messages to interact with the XDMS [89]. This provides an efficient, up-to-date and feature-rich package implementing the client side of most recent HTTP standards and recommendations. The package is released under the Apache Licence, Version 2.0, which provides maximum flexibility for source code reuse, modification and distribution.

The policy profile defines filter criteria that allow control policies to be selectively invoked as a result of pattern matching on any Diameter AVP present in a request. The policy profile provides the XCAP URI of the applicable control policy to be retrieved from the XDMS. It also defines a priority that specifies the order in which control policies are invoked. Fig 6.4 shows the serial fashion in which control policies are incorporated into QoS resource control.

For each incorporated control policy a corresponding policy processor block is created. These are Java classes that are instantiated and executed in the order specified in the

end-user's policy profile. The policy processor block is specific to the control policy. It retrieves the relevant control policy from the XDMS, parses the information and either performs access control by allowing or disallowing the session or creates/refines enforcement policies, or both. A generic policy processor block is defined that reads basic policy information from control policies adhering to the SIP *session-policy* schema described in Section 4.3.1.

Typically a Topology and Resource Information Specification (TRIS) would be populated by information received from the transport plane via dedicated interfaces. These interfaces and procedures are still under standardisation and beyond the scope of this work. However the created enforcement policies, adhering to the SIP *session-info* schema described in Section 4.3.1, are stored on the XDMS for the duration of the session. This provides one aspect of the information necessary to maintain a TRIS and can be used to perform elementary QoS reporting. This information is incorporated into QoS resource control and can be used, for example, to limit the number of concurrent sessions of a certain kind.

The addresses of neighbouring PDFs are stored along with this information. As discussed in Section 5.2.2, these addresses would be exchanged as part of Service Level Agreements (SLA) between operators, or alternatively discovered automatically using topology discovery mechanisms [49], though in the testbed these are manually configured. The PDF uses this information to incorporate the logic defined in Section 5.3.2 to discover the end-to-end media path and to ensure that resources are reserved in all traversed transport segments.

6.3.2 Transport Functions

The focus of this work is the interaction between the service and resource control planes to ensure end-to-end QoS negotiation for advanced multimedia services. IP QoS models are not under study and are implemented in an elementary manner to facilitate proof of concept evaluations. The transport functions are represented by a single Border Gateway Function (BGF) in the UCT Policy Control framework. This element is implemented in the Java programming language and uses the JavaDiameterPeer library to instantiate the Diameter Gx interface. The BGF processes PCC rules and creates service data flow filters that identify a particular flow based on a 5-tuple. A software router is implemented in Linux using the *iptables* utility. Iptables is a user space Linux application that allows

Table 6.1: The BGF translates QCI characteristics in the PCC rule into DiffServ PHBs.

Priority	Example Service	QCI	DiffServ PHB
1	IMS signalling	5	EF
2	Conversational voice	1	
3	Real time gaming	3	AF4
4	Conversational video	2	
5	Non-conversational video	4	AF3
6	Video (buffered)	6	
7	Voice, video, interactive	7	
8	Video (buffered)	8	AF2
9	Sharing	9	AF1

an administrator to define tables containing chains of rules, which specify the treatment of packet flows. DiffServ classes are created and packet flows are matched to rules, and marked and queued according to information in the PCC rule. The router implements DiffServ Per Hop Behaviours (PHB): Expedited Forwarding (EF), Assured Forwarding (AF) (four classes with different drop precedence levels) and Best Effort (BE). There is no strict one-to-one mapping between Qos Class Identifier (QCI) characteristics and DiffServ Code Points (DSCP), and the operation should be performed based on operator policies, configured into the node through an Operations & Management (O&M) system. There are 9 traffic categories defined by the QCI; 3GPP TS 23.203, *Policy and Charging Control Architecture* provides detailed information on the standardised characteristics of these categories [26]. Table 6.1 shows the basic mapping implemented at the BGF in the testbed.

The BGF includes elementary event reporting functionality as part of the proof of concept implementation. By monitoring the existing flows using the iptables utility, event triggering can be supported, including the events Loss of Bearer, Recovery of Bearer and QoS Change Exceeds Authorisation. These events are based on changing conditions in the transport plane and can result in session modification or termination in the service control plane. Fig. 6.5 shows the message exchange for a Loss of Bearer trigger event. The BGF detects the Loss of Bearer event and informs the PDF (1), which in turn informs the AF (2). The AF makes a decision based on operator policy and informs the PDF of this decision (3), which in turn reconfigures the BGF (6, 7) and updates relevant policies (8, 9). In this scenario the result is session termination (11, 12), though other actions could be taken depending on operator policy.

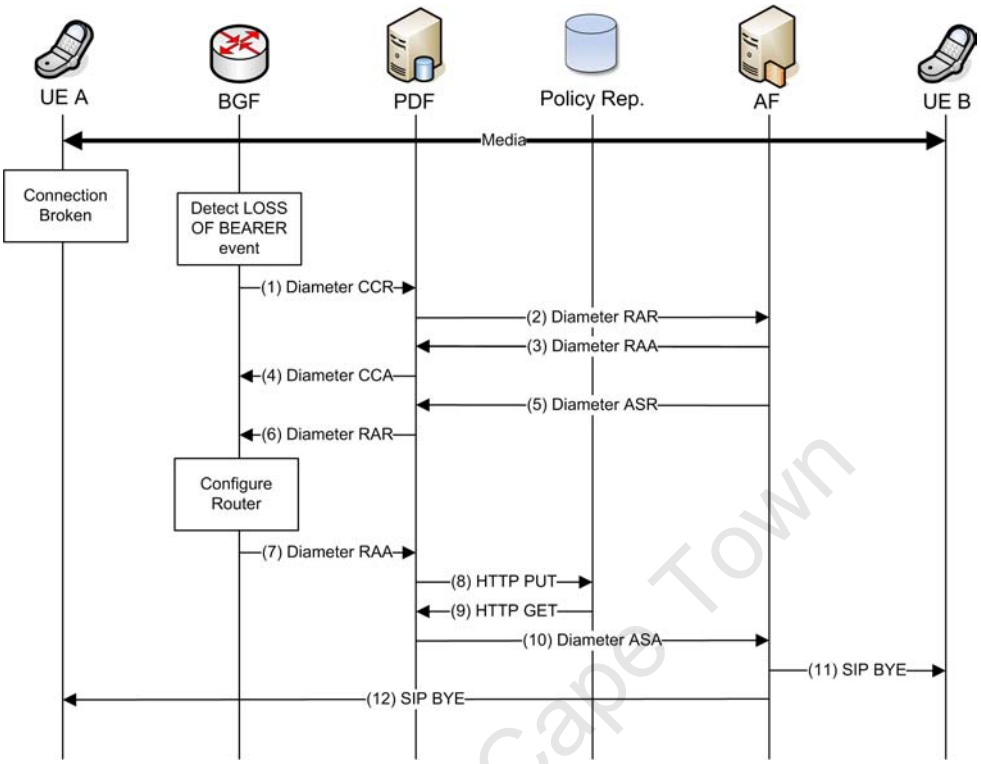

Figure 6.5: The BGF detects transport plane events and these are processed at the resource and service control planes.

For example, the session could be maintained but the reserved resources freed until such time as a Recovery of Bearer event is triggered. The BGF in the testbed also writes all flow information to a log file that can be used for offline analysis or incorporated into other modules.

6.3.3 Policy Repository

The XDMS was chosen to represent the policy repository for several reasons. It is a widely used IMS AS, and XML, as the standard for policy representation, is gaining popularity. It also ensures that the system is flexible and can adapt to changing network conditions by rapidly deploying new policies. The XDMS uses XCAP to transport the XML documents. XCAP defines HTTP Put, Get and Delete methods to store, retrieve and remove XML documents respectively [65]. The information is carried over a secure HTTP connection, which is out of band of IMS signalling. OpenXCAP is an implementation of an XCAP server that is written in the Python language. It uses

SQL to store XML documents, and is released under the BSD licence [90]. To ensure the integrity of the connection to the XDMS, OpenXCAP supports Transport Layer Security (TLS) and basic HTTP authentication. The HTTP authentication is similar to SIP authentication and involves a challenge/response mechanism, but does not traverse the IMS core elements.

The UE, AF and PDF each act as XCAP clients. The policy repository interacts with the UE over the specified Ut interface. This allows UEs to access and modify end-user specific application control policies, as described in Section 4.3.1. While default application control policies are defined to ensure that no QoS standardisation is required to deploy new services, the actual creation of these documents is beyond the scope of this work. It is expected that this will be performed by a dedicated piece of software, possibly through a web interface. The AF interacts with the policy repository to retrieve end-user specific application policies and uses this information to tailor the authorisation request and to customise the service delivery according to the end-user requirements. The PDF also interacts with the policy repository to retrieve policy profiles and to incorporate control policies into QoS resource control, and to store enforcement policies for current sessions.

The XDMS stores control and enforcement policies that are based on *session-info* and *session-policy* schema, defined by Hilt *et al.* [45]. These schema are extended to support the wide range of control policies; OpenXCAP supports XML document validation using stored XML schema, to verify documents uploaded via the HTTP PUT command. The policy-profile schema is based on the subscription-profile schema defined in 3GPP TS 29.228, *IP Multimedia (IM) Subsystem Cx and Dx interface; Signalling flows and message contents* [91].

An XCAP URI comprises an XCAP root and a document selector. An XCAP root is a valid HTTP URI that the XCAP client can resolve, for example:

http://www.xcap.com:8000

The document selector includes the Application Unique ID (AUID) and the user's subtree. *Domain-policy, sub-policy, access-policy* and *vod-policy* AUIDs are defined based on the *session-policy* schema. Additional AUIDs can be defined as control policies are deployed. The user's subtree informs the XDMS which user the requested XML document is specific to. Optionally the file name of the stored XML document is appended to the URI. A full XCAP URI for an end-user's subscription control policy is shown below.

http://www.xcap.com:8000/sub-policy/users/sip:bob@qos-ims.test/sub-policies.xml

The use of the AUID and user's subtree to identify XML documents stored on the XDMS means that the documents are specific to a particular application and a particular end-user. However global documents specific to all users of a particular application are supported. In this case the user's subtree does not form part of the document selector. This would be used, for example, for domain control policies that are specific to the network operator rather than an individual user. An example XCAP URI for domain policies that apply to all end-users is shown below.

http://www.xcap.com:8000/domain-policy/global/domain-policies.xml

Fig. 6.6 shows typical interactions with the XDMS server. A UE uploads end-user specific VoD control policies (1-4). When the AF receives a SIP request it retrieves the end-user specific VoD control policy to customise service delivery and to help create authorisation requests (5-8). When the PDF receives an authorisation request, it retrieves domain control policies to incorporate into QoS resource control (9-12).

6.3.4 Management Interface

A web management interface was implemented to interact with the XDMS. This allows an operator to view and modify the policies stored on the XDMS and to perform basic operations and maintenance on the Common PCC elements. This tool is extremely useful as it allows real time monitoring of the control and enforcement policies. As proof of concept the base control policies can be modified through the web interface, including the TRIS. Design of an interface for non-technical application developers to create new control policies is beyond the scope of this work. The web management interface uses the Apache Tomcat Servlet container [92]. Apache Tomcat implements the Java Servlet and JavaServer Pages (JSP) specifications, developed under Java Community Press, and provides a Java Based HTTP web server. It is released free and open source under the Apache Licence, Version 2.0. Many of the modules developed in the PDF element could be directly reused and incorporated into the web management interface because of the shared implementation language and modular design. Fig. 6.7 shows a typical screenshot where all active IMS sessions are listed.

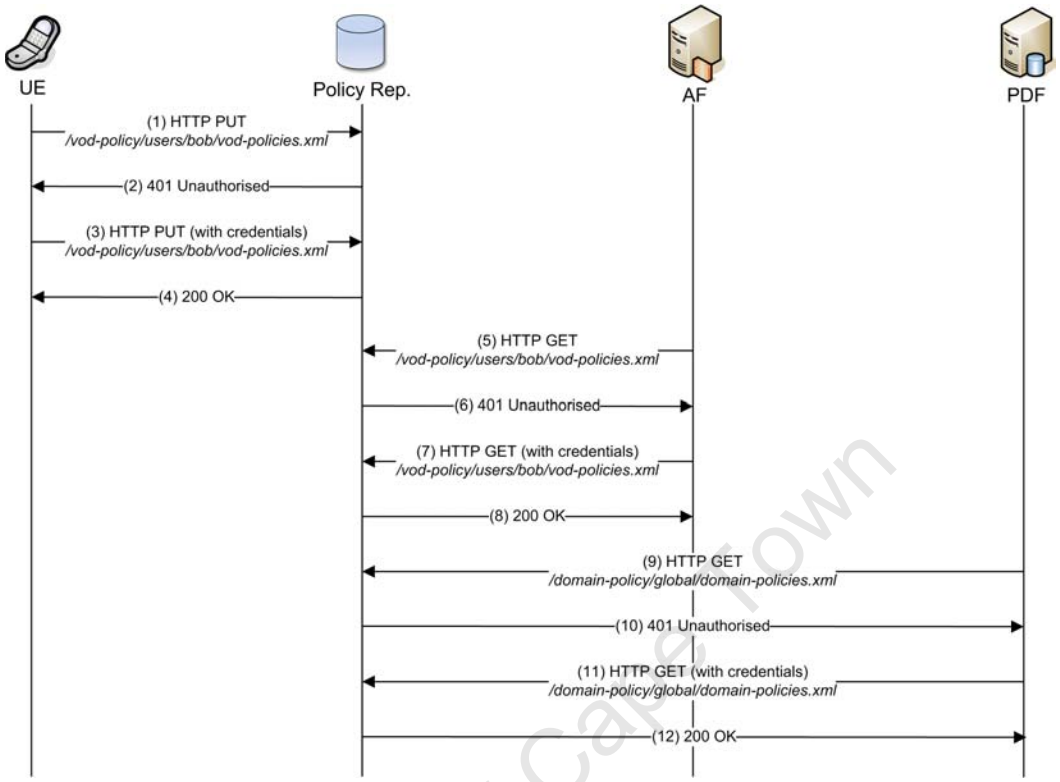

Figure 6.6: The policy repository interacts with the UE, the PDF and AF.

6.3.5 High Performance Framework

Java, as a programming language, permits the rapid prototyping of advanced concepts and is extremely useful for new technology implementations like the UCT Policy Control Framework. The framework should be subjected to realistic use case scenarios involving numerous subscribers and simultaneous sessions. This is to ensure that the evaluations carried out on the testbed provide an accurate representation of the expected performance of the proposed extensions in a practical IMS deployment. The Java programming language introduces overheads, and can result in bulky and inefficient implementations.

For this reason a scaled-down version of the UCT Policy Control framework was implemented in the C programming language. This framework defines the PDF and BGF elements, Diameter Rx, Gx and S9 interfaces and the HTTP Ut interface. CDiameterPeer was extended for the Diameter reference points, and the libcURL library along with libXML2 was used to implement the Ut interface. This implementation does not include the policy processor blocks, policy profiling, or web management mechanisms. Rather it implements only the signalling required for end-to-end QoS negotiation for

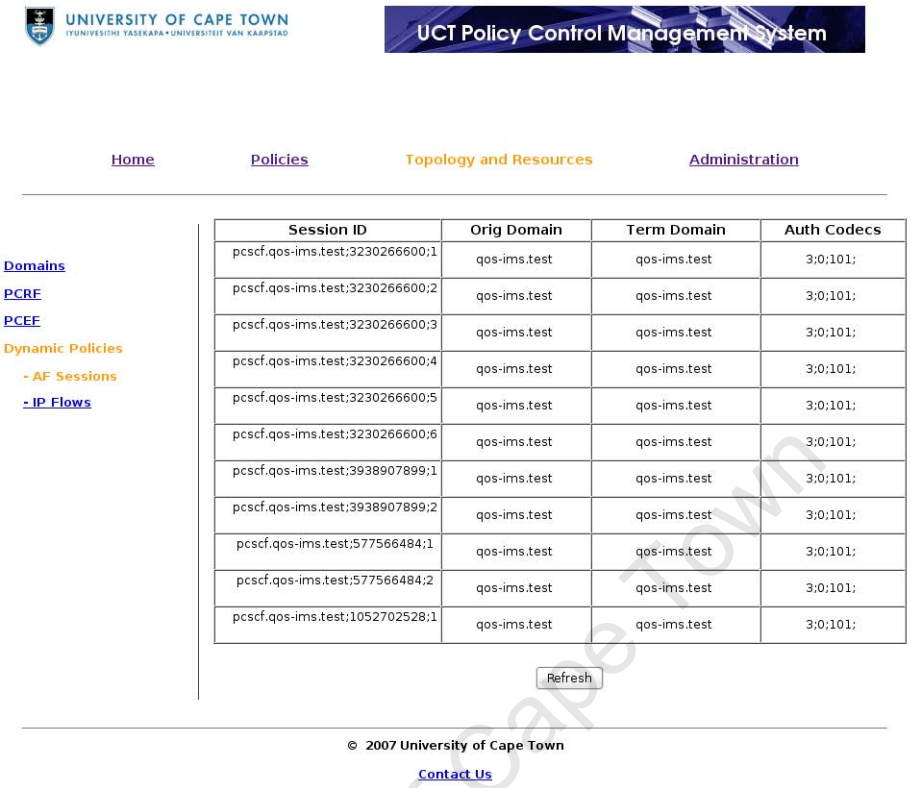

Figure 6.7: The web management interface allows monitoring of control and enforcement policies in real time.

advanced multimedia services. The signalling overhead will be the predominant metric when processing numerous subscribers and simultaneous requests. This high performance framework is used to verify the proposed extensions under stress, while the Java UCT Policy Control framework demonstrates and validates the concepts under normal load.

6.4 User Equipment

A UE implementation is necessary to perform IMS registration with the core network, initiate and terminate sessions, provision advanced multimedia services on the client side, and handle media and user interaction. The UE would need to implement the three-way offer/answer model, where session parameters are described and negotiated in the SDP bodies of SIP messages [93]. The UE also needs to integrate resource management with SIP signalling, including support for extended IMS signalling and preconditions [68].

This is necessary for the discovery of the end-to-end signalling path, described in Section 5.3.1, and to ensure that the results are an accurate representation of a typical IMS deployment.

6.4.1 UCT IMS Client

When the OSIMS was first released, a key component, the IMS Client, was not available. Instead a SIP-to-IMS gateway was provided to allow vanilla SIP clients to interact with the core network. This was limiting in terms of functionality and the ability to realise accurate IMS deployment scenarios. The UCT IMS Client project was started as an open source initiative and released freely under the GPLv3 in December 2006, in order to take advantage of the large pool of developers and testers around the world [94]. The client was co-developed by a fellow researcher and the author of this thesis and has grown in robustness and scale through 13 releases. When made available to the public, it was the first free and open source IMS client implementation. At the time of writing, supported features include audio and video calling, page-mode and session-based instant messaging, MD5/AKAv1/AKAv2 authentication, SIP presence, and Real Time Streaming Protocol (RTSP) for Video on Demand (VoD) applications. The code-base has been used extensively in a number of different IMS projects [74][95][87], and since its release there have been in excess of 11000 downloads.

The goal of the project is to provide software that is free, and can be easily installed, operated and modified by IMS researchers. Many protocols are required to perform the relevant functions, including SIP, XCAP, Message Session Relay Protocol (MSRP) and RTSP. The Linux operating system was chosen because of the freely available libraries needed to implement these interfaces. While this means that the client does not operate in the ubiquitous Windows environment, it does not detract from its primary function as a research and testing tool.

Full IMS signalling, including support for the offer/answer model and preconditions, is implemented using the oSIP and eXosip libraries [82][83]. The HTTP Ut interface is implemented using libcURL [85], and libXML2 populates and parses the XML components of the messages [86]. The Gstreamer framework provides support for the media plane [96]. It provides modules to capture audio and video, encode the media in a variety of codecs, and packetise the resultant bit stream into RTP packets. The framework allows a developer to flexibly link modules together to form a pipeline. RTSP functions

allow for the description, setup and control of on-demand video streams, and support trick-play functionality like PLAY, PAUSE, FAST-FORWARD, etc. The libVLC library, released under GPLv2, incorporates this functionality into the client [97]. Session-based instant messaging allows the client to transmit a series of related instant messages in the context of a session. LibMSRP is a high level API for session-based instant messaging. It is released free and open source under GPLv2, and is used to implement simple methods to initiate, process and terminate session-based instant messaging sessions in the UCT IMS Client [98]. All libraries except for libMSRP are dynamically linked at run-time and must be installed on the host machine. LibMSRP is not currently available in standard Ubuntu software repositories and is therefore included as part of the client source code.

The UCT IMS Client fully satisfies the requirements to demonstrate and validate the proposed Common PCC extensions in the practical testbed. Registration with the OSIMS using AKAv2-MD5 authentication is supported. It implements full IMS signalling, needed to calculate the end-to-end signalling path. The Ut interface is realised and allows a UE to interact with the policy repository, and upload and manipulate end-user specific application policies. Multimedia rich IMS sessions incorporating audio, video and session-based instant messaging are supported, and advanced IMS services are provisioned, including a VoD client. A typical screenshot of the UCT IMS Client, including configurable IMS preferences, is shown in Fig. 6.8.

6.4.2 SIPp

A large number of UEs and concurrent sessions will be necessary to subject the testbed to realistic use case scenarios, and to portray an accurate live IMS deployment. It is impractical to physically deploy hosts, as necessary, in the testbed. For load and stress testing purposes, SIPp is used to represent UEs.

SIPp, licenced under GPLv2, is an open source testing tool and traffic generator for the SIP protocol [99]. It can emulate call flows ranging from simple to very complex. As a testing tool, it provides only a basic text-based interface and has minimal support for media. However it does feature a dynamic display of statistics, including call rate, round trip delays and message timeouts, which are updated in real time. It also supports AKAv2-MD5 authentication, allowing registration with the OSIMS.

Custom XML scenario files define the appropriate SIP requests and responses that must be generated, and in what order. Registration, session initiation and session termi-

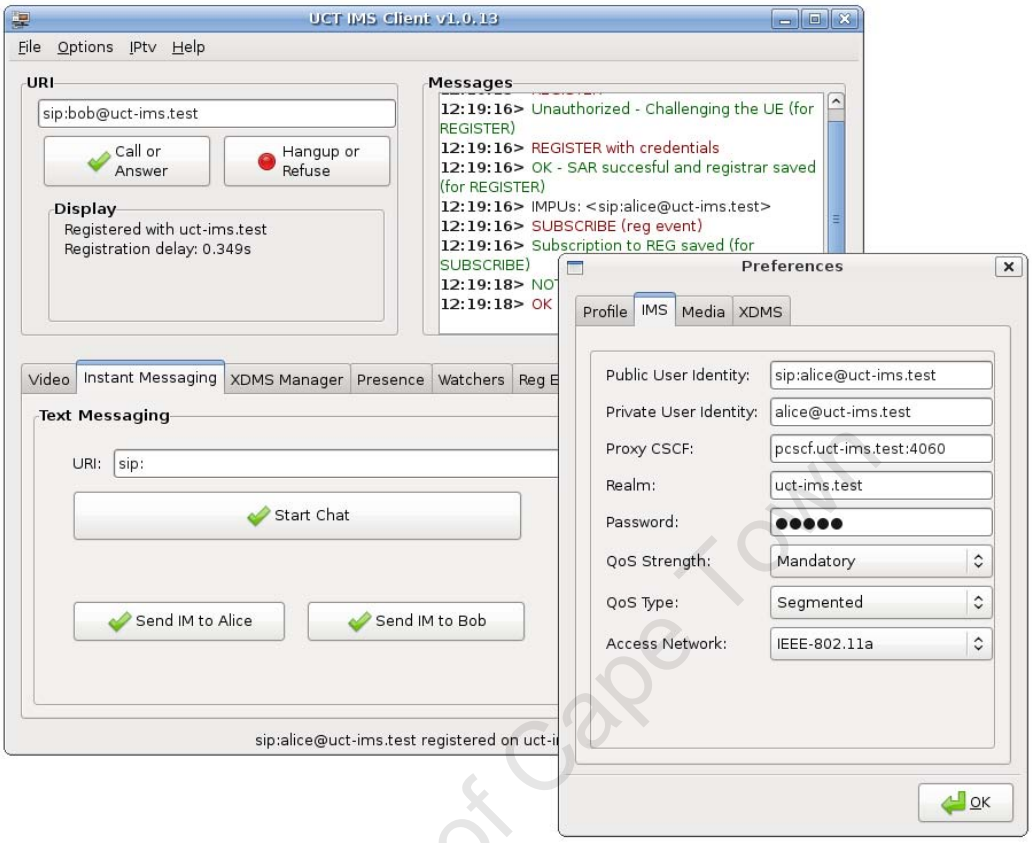

Figure 6.8: The UCT IMS Client and configurable IMS preferences.

nation scenarios can generate and process full IMS signalling for these procedures, and because SIPp allows a variable call rate, a large number of subscribers and simultaneous sessions can be emulated on a single host. The UCT IMS Client demonstrates and validates the proposed extensions in normal conditions, while SIPp is used to subject the system to heavy load.

6.5 IP-Connectivity Access Networks

Before a UE starts an IMS related operation, there are a number of prerequisites that have to be met. First the IMS service provider needs to authorise the UE, typically through some kind of subscription or contract, agreed upon between the IMS network operator and the subscriber. Second, the UE needs to connect to an IP-Connectivity Access Network (IP-CAN) that provides access to the IMS home network or visited

network and provisioned services. Third, the UE needs to discover the IP address of the outbound/inbound SIP proxy server, and register, at the SIP application level, with the IMS network.

As part of the process of connecting to an IP-CAN, the UE needs to acquire an IP address. It was assumed that by the time of the first IMS deployment, IP version 6 (IPv6) would be the most common IP version on the Internet. Network Address Translation (NAT) mechanisms are necessary to allow private IP addresses under the IP version 4 (IPv4) scheme, because of the limited number of available addresses. SIP and its associated protocols (SDP, RTP, etc.) have known issues when traversing NATs. For these reasons, when IMS was under development, IPv6 was chosen as the only version of IP for IMS connectivity. In June 2004 3GPP re-examined this decision [6]. Market indications revealed that IPv6 had not yet gone mainstream and work on NAT traversal for SIP had developed substantially. Based on these indications, 3GPP decided to allow early deployments of IMS over IPv4. In the testbed deployment all hosts are allocated IPv4 addresses, and only publicly routable addresses are assigned to avoid the necessary NAT optimisations.

The UE needs to discover the address of the P-CSCF, acting as the inbound/outbound SIP proxy, once it has been assigned an IP address. Once the P-CSCF discovery process is complete, the UE can send and receive SIP signalling to and from the P-CSCF. Automatic P-CSCF discovery can form a part of the process to obtain IP-connectivity, for example via the Dynamic Host Configuration Protocol (DHCP). In the testbed none of the IP-CANs support this automatic discovery and the address and port number of the P-CSCF are manually configured at the hosts.

The testbed uses real IP-CANs and mechanisms to connect the UE to the IMS core network. The network elements associated with these IP-CANs have proprietary interfaces and do not support the standardised Gx interface to facilitate PCC rule enforcement. Enforcement of policy rules in the transport plane (including in the access networks) is emulated at the BGF described in Section 6.3.2. This is sufficient to depict the signalling between the service and resource control planes in a realistic IMS deployment.

6.5.1 Local Area Network

The Local Area Network (LAN) IP-CAN connects the UE directly to the P-CSCF element of the IMS core via a 100 Mbps Fast Ethernet link. The UE, IMS core and Common PCC elements are all hosted in the laboratory environment. The average Round Trip Time (RTT) for an Internet Control Message Protocol (ICMP) echo request is 0.172 ms. The hosts obtain IP addresses from the UCT DHCP server, and the UCT DNS is used to resolve domain names.

6.5.2 IEEE 802.11

The testbed UEs are equipped with IEEE 802.11 access technology, to connect to the P-CSCF element of the IMS core. This IP-CAN operates exactly as the LAN, but the UE connects to an 802.11g enabled access point that is connected directly to the IMS core and Common PCC elements hosted in the laboratory. 802.11g offer a theoretical throughput of up to 54 Mbps and the average RTT in the testbed is 1.381 ms. It operates in the Industrial, Scientific and Medical (ISM) 2.4GHz radio band and suffers from interference, which limits its range to a few hundred metres. The UCT DHCP server and DNS are used for IP address allocation and domain name resolution.

6.5.3 EDGE

The UEs use Huawei E620 Mobile Connect PCMCIA data cards to access wireless cellular networks. The hardware supports Enhanced Data Rates for GSM Evolution (EDGE) technology that allows a theoretical downlink throughput of up to 236 kbps. This allows the UEs to access the IMS Core and PCC elements via a wireless cellular network, in this case the *Vodacom* network operating in South Africa.

The EDGE IP-CAN performs standard GPRS attach, and Packet Data Protocol (PDP) context activation procedures, to set up a connection and acquire an IP address from the Vodacom DHCP server. An unrestricted Access Point Name (APN) is utilised to ensure that the IP address is publicly routable. The Vodacom DNS servers are used for domain name resolution. The Vodafone Mobile Connect Card driver for Linux is run on the IMS terminals to connect to the IMS core network, via the EDGE network [100]. This project, initiated by the Vodafone R&D Labs, is written in the Python language and is released under GPLv2. It provides a graphical management interface for GPRS/UMTS/HSPA

devices, including the Huawei E620, under the Linux operating system.

6.5.4 HSDPA

The Huawei E620 also features High Speed Downlink Packet Access (HSDPA), if it is supported by the mobile operator. Vodacom has extensive HSDPA coverage throughout South Africa. The link between the UE and IMS core is the same as with the EDGE CAN. The same GPRS attach and PDP context activation procedures are used to acquire a public IP address, and the Vodacom DNS resolves domain names. This particular HSDPA deployment supports theoretical downlink throughput of up to 1.8 Mbps.

6.5.5 IEEE 802.16

An experimental 802.16d IP-CAN allows UEs to connect to the IMS core via fixed WiMax technology. The IP-CAN is made up of BreezeMax equipment and includes a BreezeMax Micro Base Station (µBST) and a BreezeMax PRO Subscriber Unit (SU). Because of limited resources and licensing issues the testbed used Frequency Division Duplexing in the 3.5GHz frequency band, and a radio frequency (RF) cable provides the link between the µBST and SU, with fixed 60 decibel (dB) attenuators and a variable 0-70 dB step attenuator. The µBST offers a wide range of features including the creation of specific services, service profiles and QoS profiles. For simplicity, all services utilise the Continuous Grant service profile, which defines forwarding rules, priority classifiers and MAC scheduling mechanisms. Continuous Grant is tailored for real-time services characterised by fixed packets transmitted on a periodic basic, like VoIP, and was selected as most applicable for general multimedia services. This is sufficient as the IP-CAN evaluations will be exclusively related to signalling overhead. The 802.16d IP-CAN connects the UE to the IMS core elements hosted in a private LAN behind the µBST. This private network hosts a DHCP and DNS server for IP address allocation and domain name resolution. The average RTT between UE and P-CSCF over this access is 31.122 ms. To ensure reproducibility the full details of the 802.16d IP-CAN architecture and physical layer specifications are detailed in Appendix B.

6.6 Summary

This chapter has detailed the components necessary to create a fully functional IMS testbed and demonstrate end-to-end QoS negotiation for advanced multimedia services. The testbed comprises elements in the service control and resource control planes, and connects UEs via real IP-CANs to the IMS core network.

All developed elements exhibit strict conformance to relevant 3GPP and IETF standards. This is critical to ensure that when the testbed is evaluated, the results are an accurate representation of a realistic IMS deployment. The framework is a true open testbed and comprises entirely Free and Open Source software. This exposes the complex concepts to a wide range of developers and helps accelerate technology acceptance. It also provides a convenient point of departure for further research in the field, as all evaluations are fully reproducible in the practical environment and the components can be easily and legally extended to incorporate future experimental technologies.

The testbed implementation demonstrates proof of concept, the next chapter details extensive evaluations of the proposed concepts. The framework provides a practical environment where the performance and reliability of the experimental mechanisms can be studied.

Chapter 7

Performance Evaluation

The design and practical implementation of extensions to the Common PCC framework were detailed in the previous chapters. While the evaluation framework alone demonstrates proof of concept, the proposed extensions need to be subjected to realistic evaluations to determine the suitability and viability of the solution. A network simulation environment could be used to demonstrate the concepts and this would allow large scale testing. However, in this thesis, the evaluations are carried out exclusively on the already described testbed platform. Despite the fact that practical implementations suffer a degree of randomness making exact replication of the testing environment impossible, they have a number of important advantages. An evaluation framework allows the developer to address issues on all levels of the design process. Furthermore when implementing a framework many problems that go undetected in a simulation, become readily apparent, because testbeds are typically not subject to as many assumptions.

It is possible, due to the nature of the practical testbed, to subject each individual component to realistic use case scenarios. In this chapter the proposed architectures for application driven policy control and end-to-end inter-domain coordination are incorporated incrementally into the testbed and evaluated. Additionally an IMS Video on Demand (VoD) deployment scenario is analysed to provide insight into the impact on application performance when the proposed architecture is adopted.

Metrics under investigation are those that might be affected by the extended policy control mechanisms and in turn have an effect on end-user experience or network utilisation. Transport plane metrics like jitter and packet loss are not presented. Analysis of session setup is a recurring theme throughout the evaluations as this is a procedure that occurs frequently in a telecommunications network and will be most affected by the

additional signalling.

The results of the performed evaluations aim to demonstrate the effectiveness of the proposed solution, particularly regarding the introduced overheads. Moreover they show the limitations of the deployment and provide insights for future work. Every effort has been made to ensure full reproducibility of all testing procedures through the use of open source software and open testbeds.

7.1 Application Driven Policy Control Framework

The first set of tests are dedicated to the application driven policy control framework described in Chapter 4. Focus is placed on the extended functionality of, and interactions between, elements:

- The extended Application Function (AF) is placed on the signalling path and extracts Session Description Protocol (SDP) information from the SIP messages. This information is used to create Diameter authorisation requests.
- The AF retrieves application control policies from the XML Document Management Server (XDMS) via the Ut interface. These control policies are specific to the requested service and the requesting UE and they define end-user constraints for the customised delivery of services. They are also used to create Diameter authorisation requests.
- The extended Policy Decision Function (PDF) creates a PCC rule for each Media-Component-Description AVP in the authorisation request.
- The PDF uses policy profiles to selectively invoke control policies, which are used to create and tailor the PCC rules. The policy profiles and control policies are retrieved from the XDMS.
- The Border Gateway Function (BGF) translates individual PCC rules into technology dependent configuration information - in this implementation, *iptables* configuration rules.

A negligible effect on session setup delay has been cited as a critical requirement of the Common PCC framework [16]. The design of the application driven framework limits

additional signalling to the core network to meet this requirement. An equally important metric is the effect on signalling overhead. The proposed framework introduces a number of additional messages during session establishment; this increase could overload network elements and decrease user utility in the network. The processing delays incurred for specific tasks at each element give an indication of the overhead introduced at different stages of execution. The framework needs to be assessed regarding scalability and ability to handle load. This can be tested by subjecting it to multiple simultaneous requests. The evaluations aim to determine the severity of the overheads introduced by the application driven policy control framework in standard operating conditions and when subjected to extreme load.

7.1.1 Scenarios

The performance of individual components and procedures can be better analysed if the proposed changes are incorporated into the testbed in stages and evaluated at each increment; hence three scenarios are introduced for the test procedures. The first scenario is the reference case that implements only basic IMS signalling with no resource authorisation mechanisms. This includes two UEs and core IMS elements, with an AF included on the signalling path. The second scenario incorporates Common PCC functionality as described in Chapter 2. This includes resource authorisation at the PDF and enforcement at the BGF. The third scenario applies when the network operator implements the proposed application driven policy control architecture. Application control policies are incorporated at the AF, and four levels of control policies (domain, subscription, application and access) are incorporated at the PDF. In this way the impact on end-user experience when adopting the proposed system can be benchmarked. The scenarios are shown in Fig. 7.1 and are limited to a single IMS domain. To ensure reproducibility the full hardware set for each scenario is detailed in Appendix A.

7.1.2 Session Setup Delay

In order to measure session setup delay a typical IMS session with audio and video media components is initiated between two UEs and evaluated in each of the three described scenarios. IMS session setup is verbose without resource management support, and requires a minimum of 59 SIP message exchanges to initiate a session [74]. The application driven policy control framework introduces additional exchanges at several

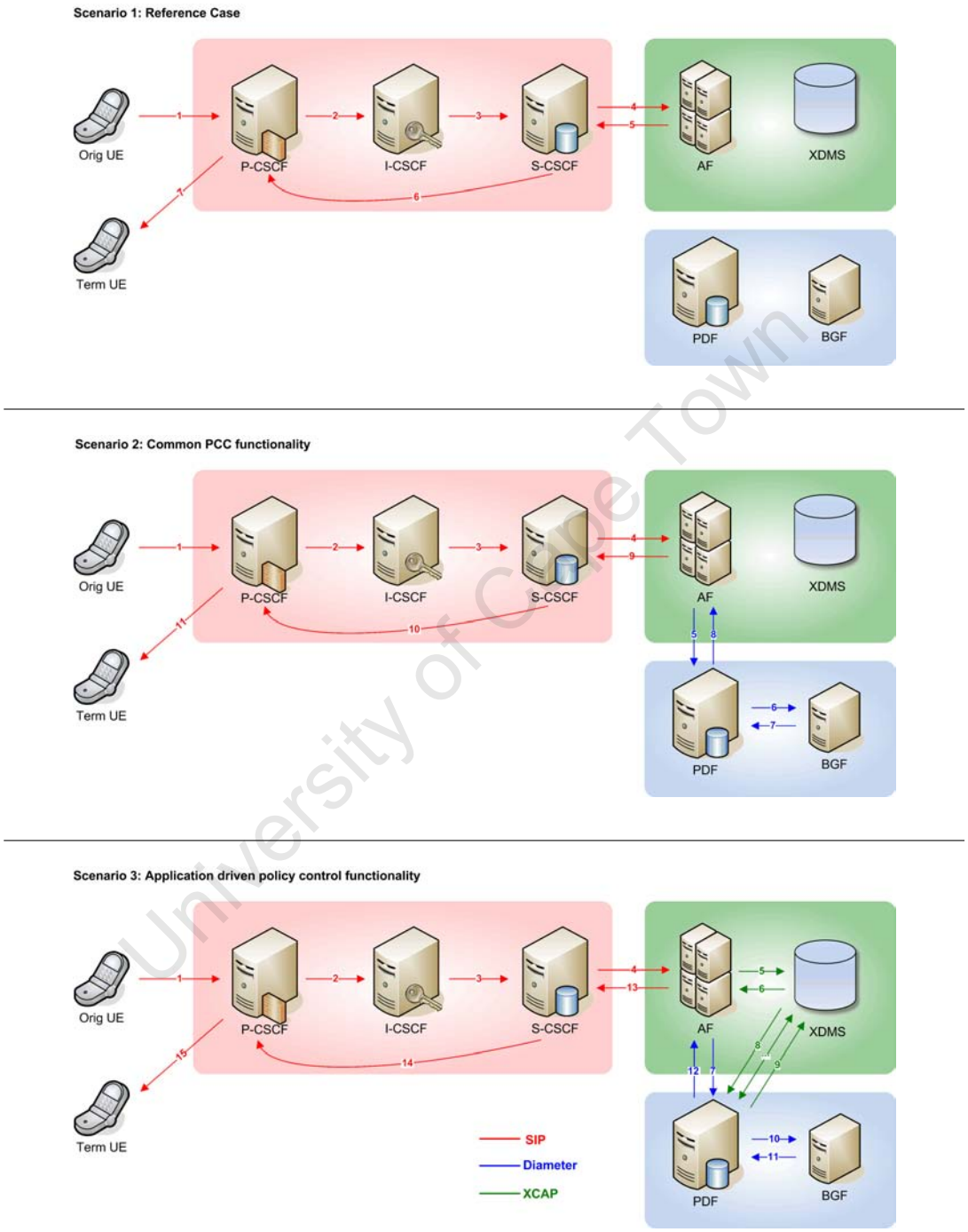

Figure 7.1: There are three validation scenarios that are subjected to evaluations.

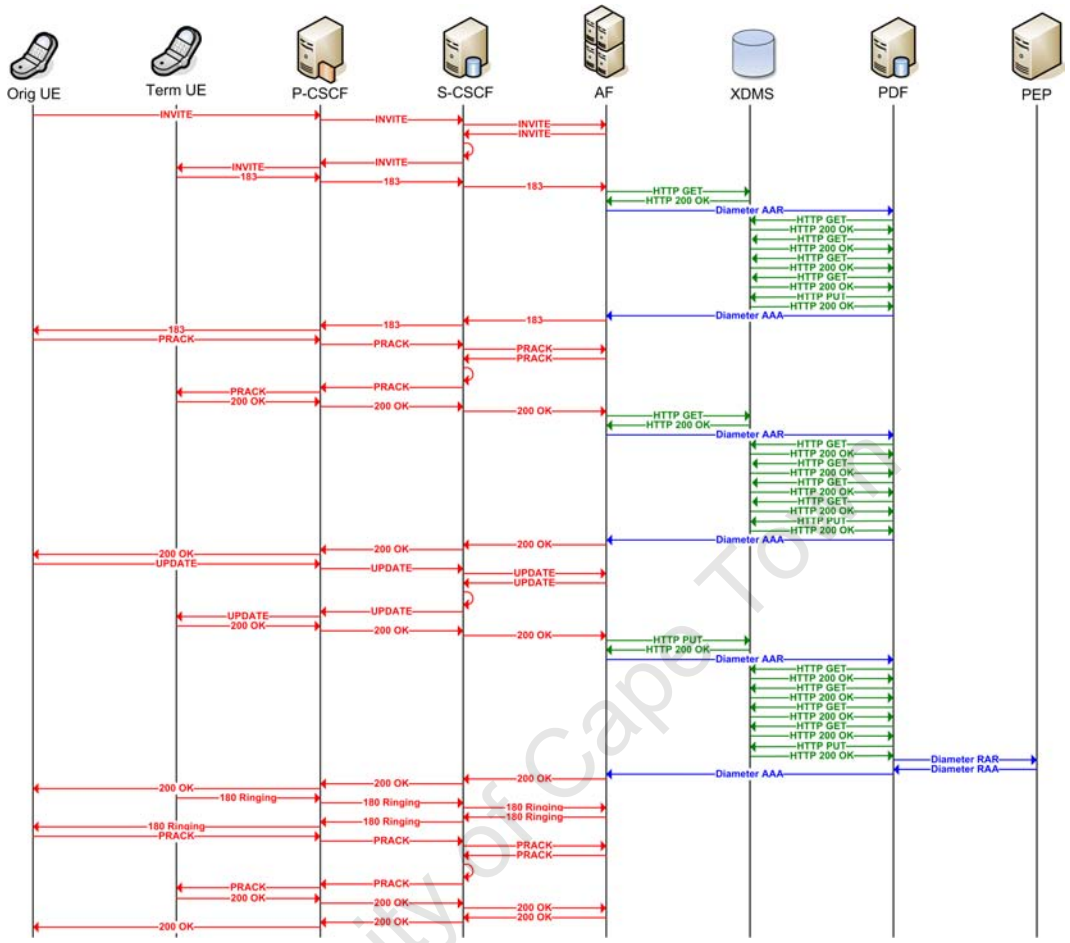

Figure 7.2: Session setup signalling for scenario 3.

elements.

The AF performs an XCAP look up to retrieve a generic application policy from the XDMS, and creates a Diameter authorisation request for the session for each exchange of Session Description Protocol (SDP) information. The PDF performs XCAP look ups for the policy profile and the control policies specified in the profile. In these tests four control policies are retrieved by the PDF: domain, subscription, access and a generic application control policy. The PDF creates a Diameter resource request containing PCC rules for each media component once the flow is enabled; the flow is only enabled once both SDP answers have been received at the AF. The PDF creates a Diameter authorisation response for the AF. The BGF extracts the PCC rule information and translates it to *iptables* configuration information and enforces the rules. It also creates a Diameter response for the PDF. The full signalling for the third scenario is shown in

Fig. 7.2; interactions with the I-CSCF are omitted for simplicity. In this particular case 62 additional messages are introduced, although this will vary depending on the network configuration, the properties of the initiated session and the number of invoked control policies. These additional messages are limited to the core network.

The session setup delay is measured from the first DNS look up for the initial INVITE request until the receipt of the 200 OK response to the Provisional Acknowledgement (PRACK) for the 180 RINGING response. The time taken for the terminating UE to accept the session is an uncontrollable variable and hence does not form part of the measured delay. The *gettimeofday* function is used to determine the time elapsed between events; this function accurately obtains the current time with resolution in the order of microseconds. The tests are carried out over five IP-CANs: 802.3 100BASE-T Ethernet LAN, 802.11g, HSDPA, EDGE and 802.16d. The originating UE connects via the different access technologies, while the terminating UE is always connected via the LAN IP-CAN. This gives an indication of how the access technology affects the session setup time and the proportion of overhead that is added by the application driven policy control architecture. The results for each IP-CAN are shown in Table 7.1, Table 7.2, Table 7.3 and Table 7.4, respectively. Due to the random nature of practical implementations and the varying response time of the elements involved, the results are obtained over 50 test runs to ensure an accurate representation. Fig. 7.3 shows the individual session setup delays measured over the LAN IP-CAN for each scenario. This demonstrates the randomness introduced by a practical implementation and graphically shows the effect on session setup delay for each scenario.

The performance of the EDGE access technology is unacceptable in all scenarios. This increased delay stems from SIP retransmission timeouts configured for the Internet environment and therefore frequently triggered in the unpredictable wireless medium. One possible solution is to adjust the SIP retransmission timeouts according to the access network in use [101], though this is not investigated in this thesis. The results obtained for the EDGE IP-CAN are erratic and call failure is frequent, hence they are omitted from the investigation.

Table 7.1: Session setup delay results for LAN IP-CAN access.

	Scenario 1	Scenario 2	Scenario 3
Minimum (s)	0.809	0.813	0.906
Mean (s)	1.150	1.179	1.248
95th percentile (s)	1.228	1.420	1.632
Delay overhead (%)	-	2.521	8.522

Table 7.2: Session setup delay results for 802.11g IP-CAN access.

	Scenario 1	Scenario 2	Scenario 3
Minimum (s)	1.009	0.933	0.885
Mean (s)	1.190	1.211	1.276
95th percentile (s)	1.228	1.430	1.632
Delay overhead (%)	-	1.765	7.227

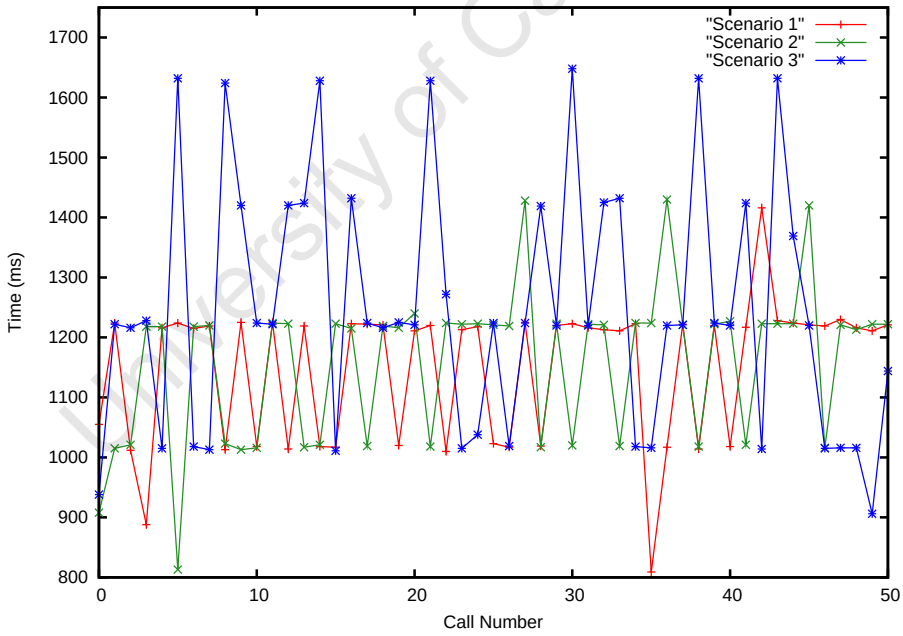

Figure 7.3: Individual session setup delay measurements for LAN IP-CAN access.

Table 7.3: Session setup delay results for HSDPA IP-CAN access.

	Scenario 1	Scenario 2	Scenario 3
Minimum (s)	2.028	1.860	1.671
Mean (s)	2.312	2.339	2.386
95th percentile (s)	2.979	2.708	2.984
Delay overhead (%)	-	1.168	3.201

Table 7.4: Session setup delay results for 802.16d IP-CAN access.

	Scenario 1	Scenario 2	Scenario 3
Minimum (s)	1.020	0.874	1.016
Mean (s)	1.372	1.433	1.443
95th percentile (s)	1.429	1.632	1.632
Delay overhead (%)	-	4.446	5.175

Discussion

The results show that for the LAN IP-CAN, incorporating the Common PCC framework (scenario 2) adds 2.5% delay overhead, while adopting the application driven policy control framework (scenario 3) increases the delay by 8.5%. Access via 802.11g shows similar results: the Common PCC implementation adds 1.8% delay overhead, while the full solution adds 7.2%. With the HSDPA and 802.16d IP-CANs the session setup delays are considerably larger, due to the nature of the wireless medium. In these cases the added overhead, as a percentage, is lower. For 802.16d access the Common PCC framework adds 4.4% delay overhead, while adopting the proposed solution adds 5.2%. The HSDPA IP-CAN exhibits the lowest delay overhead, and the respective increases are 1.3% and 3.2%.

When utilising LAN, 802.11g, 802.16d and HSDPA access, the results are acceptable with delay increases in the order of 100ms for all access technologies. The reason for the limited effect is largely due to the fact that additional signalling is limited to the core network, unlike the framework for session policies described in Chapter 3. This framework introduces a number of round trip delays, and would have a considerably greater effect on this metric in a practical implementation.

Guenkova-Luy *et al.* propose an evaluation criterion that session establishment for mobile multimedia applications should not last longer than 2 - 5 seconds when incorporating QoS coordination [102]. The results meet this criteria for all but the EDGE

access.

7.1.3 Traffic Overhead

Traffic overhead between core elements is critical during session setup. Core elements are typically high capacity nodes that need to process a large number of requests. The introduction of a small number of additional messages could overload the network elements.

The traffic generated between core elements during session setup is measured by monitoring all interfaces of the machine that hosts the core elements. The signalling is similar to that considered for the session setup delay tests, i.e from the receipt of the first INVITE request until the transmission of the 200 OK of the PRACK for the 180 RINGING response. The Wireshark protocol analyser is used to capture and measure the incurred signalling overhead during session setup for each of the scenarios [103]. The measurements need only be carried out once over the LAN IP-CAN, because with no retransmissions the message flow is constant and only signalling between core network elements is considered. The results are shown in Table 7.5.

Table 7.5: The traffic overhead incurred for each evaluation scenario.

	Scenario 1	Scenario 2	Scenario 3
Total core traffic (bytes)	83290	92687	148592
Traffic overhead (%)	-	11.282	78.403

Discussion

The results show that the traffic overhead increases by 11.3% when incorporating Common PCC functionality, and by 78.4% when incorporating the application policy control extensions. This is a considerable amount of traffic and results from the interactions between the policy repository and policy control elements for each exchange of SDP information. For a typical scenario with four control policies the architecture introduces 18 XCAP look ups during session setup.

All the core network elements are hosted on a single machine in these evaluations. In realistic operating conditions the network elements would be distributed across several high performance machines with high speed interconnections and would not be overloaded by what amounts to under 60kB of additional core network traffic per session.

The traffic loads have limited effect on the session setup delay as shown in the evaluations reported in Section 7.1.2.

7.1.4 Comparative Processing Delay

The AF and PDF perform an extended set of functions when the application driven policy control architecture is adopted. It is important to study the impact of these additional execution stages on the performance of each node, as this gives an indication of which operations have the greatest effect on session setup delay and points to areas of future work regarding optimisation.

This section focuses solely on the third scenario and uses the *gettimeofday* function to accurately measure the duration of different stages of execution at each element during session setup. The number of media components included in the session requests range from 1 to 3 (audio, video, instant messaging). The complexity of the session request has an effect on the operations carried out at each element as each media component is processed, authorised and enforced separately. This analysis concentrates on processing delays at core elements and hence only the LAN IP-CAN is utilised. For the purposes of the experiment 50 sessions are initiated and results are averaged.

The following execution stages are defined for each element:

Application Function

- Stage 1: Extract service information from SDP body of SIP messages, including SDP offer and answer.
- Stage 2: Perform XCAP read to retrieve application specific control policy from XDMS.
- Stage 3: Create and send Diameter authorisation request to PDF.
- Stage 4: Wait for response to authorisation request from PDF.

Policy Decision Function

- Stage 1: Process Diameter authorisation request and perform XCAP read to retrieve policy profile and set of control policies from XDMS.
- Stage 2: Perform XCAP write to update control policies to XDMS.

- Stage 3: Create and send Diameter resource request, including PCC rules, to BGF.
- Stage 4: Wait for response to resource request from BGF.

Fig. 7.4 and Fig. 7.5 show the processing delays incurred for each stage of execution at the AF and PDF respectively.

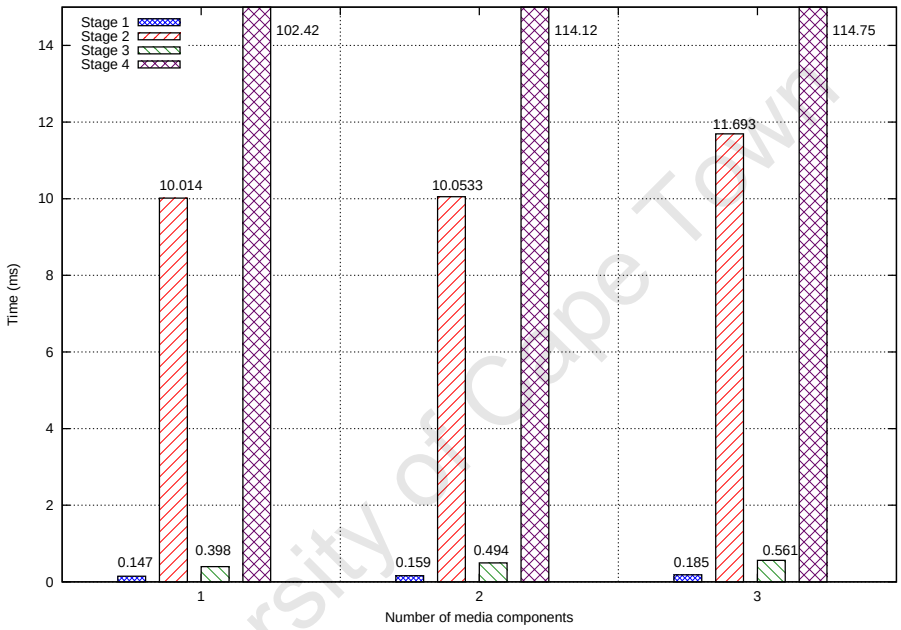

Figure 7.4: Comparative delay for different execution stages at the AF.

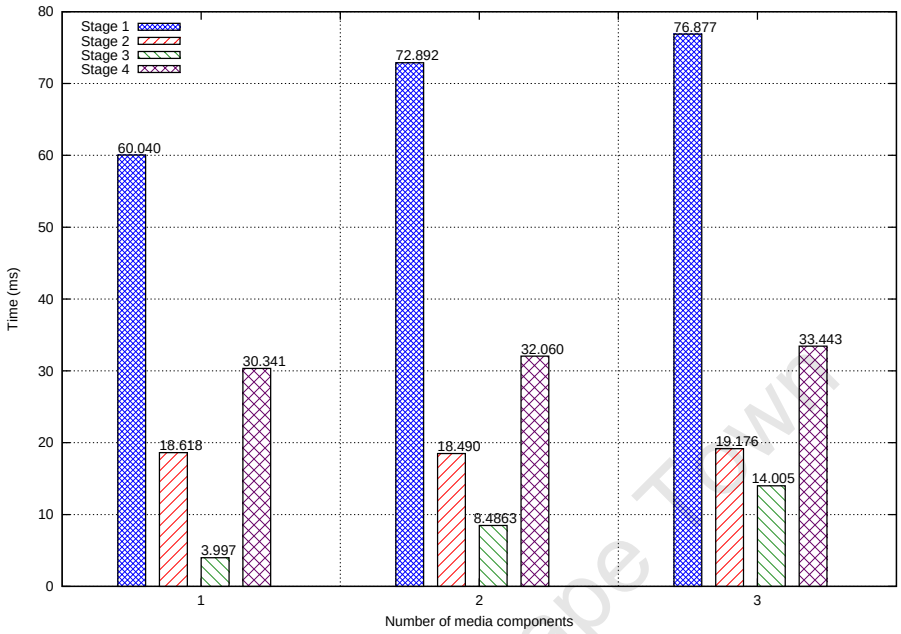

Figure 7.5: Comparative delay for different execution stages at the PDF.

Discussion

The results show that the processing delay incurred at the AF has a minor effect on the overall delay to process an authorisation request. Stage 4, waiting for the response from the PDF, is an order of magnitude larger than any of the other examined stages.

At the PDF, Stage 1, performing the XCAP reads for the policy profile and necessary control policies, takes longer than the sum of all the other stages. This stage involves 5 XCAP look ups at the policy repository and contributes the major portion of the overall processing delay for each request. Though on average it only takes 69.9 milliseconds.

The number of media components has a negligible effect on the overall processing delay. While this does have an impact on the initial processing of the authorisation request, the number and nature of the XCAP look ups for control policies are the same regardless of session request complexity.

No execution stage shows unacceptable performance when processing authorisation requests. If optimisation should be necessary for future extensions, the process of reading control policies from the policy repository could be examined and policies could be

combined to reduce the number of XCAP operations.

7.1.5 Load Testing

The testbed architecture is proof of concept and by no means a commercial grade implementation. However it is noted that core elements, including the PDF and AF, are high capacity nodes that will need to process multiple requests per second to serve a large subscriber base. In this section the effects on processing delay are analysed when the core network is subjected to multiple simultaneous session requests.

These evaluations focus solely on the third scenario. The time taken to process a SIP message, create an authorisation request and process the reply to that request at the generic AF, is measured. This operation encompasses the entire overhead of the proposed extensions and it is important to determine how different loads might effect the execution.

The SIPp open source test tool is used to simulate SIP traffic [99]. In the test context, a SIPp instance defines scenarios that perform registration with the IMS network and, as an originating UE, initiate sessions at a given rate. Another SIPp instance defines the terminating UE, which registers with the network and receives the session requests from the originating UE. The session requests are typical IMS sessions with 2 media components, and a single control policy is retrieved from the XDMS.

Sessions are initiated at a rate of 10, 20, 30 and 40 Calls Per Second (CPS) with constant inter-arrival times. The tests are run over the LAN IP-CAN for each of the aforementioned scenarios long enough for at least 400 measurements to be recorded. Only the last 200 measurements are used to obtain results to prevent erroneous delays from a cold start. This gives an accurate representation of how the adopted architecture might perform in a real world deployment serving a large subscriber base. The measured delay for different session initiation rates is shown in Table. 7.6.

Table 7.6: Processing delay at the generic AF for different session initiation rates.

	10 CPS	20 CPS	30 CPS	40 CPS
Minimum (s)	10.824	10.688	10.675	11.749
Mean (s)	19.476	19.735	26.796	38.329
95th percentile (s)	45.959	39.884	63.491	81.192

Discussion

In these tests the application driven policy control architecture is subject to signalling loads ranging from 10 CPS to 40 CPS. The results show that when 10 calls are initiated per second, the mean time to process a single authorisation request at the generic AF is 19.5ms. For 20 CPS this increases by 1.3% to 19.7ms, for 30 CPS by 37.6% to 26.8ms and for 40 CPS by 96.8% to 38.3ms.

The sharp increase is largely due to SIP retransmissions as a result of timeouts. From the previous evaluations it is clear that the XCAP operations incur the most significant delay. It can be concluded that the interactions with the XDMS are the cause of the increasing processing delay under load.

However it is important to note that in the evaluations all sessions are successfully established. Despite the processing delay overhead of almost 100% for the higher load scenario, the delay is still under 40ms, which is an acceptable percentage of the overall session setup delay.

7.2 Session Based End-to-end Policy Control Framework

The session-based end-to-end policy control architecture, described in Chapter 5, proposes end-to-end inter-domain mechanisms that discover the signalling routes in the service control plane and use this information to determine the paths traversed by the media at the resource control plane. This functionality, in coordination with the application driven policy control framework, enables QoS connectivity in all traversed transport segments. The next set of tests evaluates the extended functionality and interactions necessary for the architecture to be adopted:

- The extended AF determines the signalling path during session setup, including all domains traversed from originating to terminating UE. This information is encapsulated in Diameter authorisation requests.
- The extended PDF processes the signalling path information contained in the Diameter authorisation requests and uses this information to determine paths traversed by the media.

- The PDF in the home domain creates inter-domain Diameter resource authorisation requests. Transit PDFs perform local policy control, process these inter-domain requests, and forward them to the next traversed domain. The PDF in the terminating domain processes and terminates the inter-domain request, and creates and enforces PCC rules on the BGF.
- For typical roaming scenarios PCC rules are created and enforced simultaneously in each domain. For the worst case scenario, with different originating and terminating domains, and transit domains, PCC rules are enforced successively in each domain.

As with the application driven policy control framework, metrics under investigation include session setup delay, additional signalling overhead, processing delays at different stages of execution and performance under load. The aim of the evaluations is to determine the impact on end-user experience when enabling end-to-end QoS connectivity in a range of use case scenarios, ranging from typical to worst case.

7.2.1 Scenarios

Incremental evaluations provide insight into the performance of individual elements. There are five recurring scenarios that are examined throughout the tests; these are illustrated in Fig. 7.6 and Fig. 7.7. The first scenario is the reference case and involves two UEs attached to separate visited networks, *qos1-visited* and *qos2-visited*, and registered with different home domains, *qos1-home* and *qos2-home*. When sessions are established between the UEs only typical IMS signalling is implemented with no resource authorisation. The second scenario incorporates Common PCC policy interactions in both visited domains. This scenario represents the current support for end-to-end QoS provisioning in the Common PCC framework where resource reservation is limited to the originating and terminating domains, and transit domains are ignored. The third, fourth and fifth scenarios implement the session based end-to-end interactions but exhibit different levels of media path complexity. In the third scenario both visited domains are neighbouring and hence resource authorisation is performed only in these domains; this represents a typical roaming scenario. In the fourth scenario the media traverses three domains: both visited domains and the home domain of the originating UE. In the fifth scenario the media traverses all four domains, and resources are authorised successively in each domain. This wide range of use case scenarios allows the evaluations to contrast the

adopted architecture with a vanilla IMS implementation with no resource management framework, and to provide insights into the effects of media path complexity on the overall architecture viability. Scenarios 3, 4 and 5 also incorporate the application control policy interactions in each domain, and represent operators that have adopted the full architecture proposed in this thesis. The hardware set for each scenario is detailed in Appendix A.

7.2.2 Session Setup Delay

In the most typical session scenario where originating and terminating UEs have different originating and terminating domains but no transit domains, the end-to-end QoS solution architecture authorises and enforces resources in each domain simultaneously and thus has a negligible effect on session setup delay. However in the case of transit domains, the path needs to be discovered and resources are authorised in each domain consecutively. Depending on the number of domains included on the media path, this could have a drastic effect on session setup delay, even though additional interactions are limited to the core network.

In this section the effect on session setup delay, when enabling end-to-end QoS connectivity, is examined in all five scenarios. This allows for comparisons between the complete architecture proposed in this thesis, a vanilla IMS deployment, and an IMS deployment with standardised resource management capabilities. Furthermore the results provide insights into the performance of the architecture in scenarios ranging from typical to worst case.

The test procedure is identical to that described in Section 7.1.2. Fig. 7.8 shows the full signalling for the fifth scenario. IMS core elements are combined for simplicity; the XDMS and BGF interactions are omitted as these are the same as depicted in Fig. 7.2.

The results for the LAN, 802.11g, HSDPA and 802.16d IP-CANs are shown in Table 7.7, Table 7.8, Table 7.9, and Table 7.10 respectively.

Discussion

The results show that for the LAN IP-CAN the delay overhead increases by 6.1% when incorporating the basic Common PCC framework at originating and terminating domains (scenario 2). When utilising the session based end-to-end policy framework for a simple media path (scenario 3) there is a considerable jump in the delay overhead to

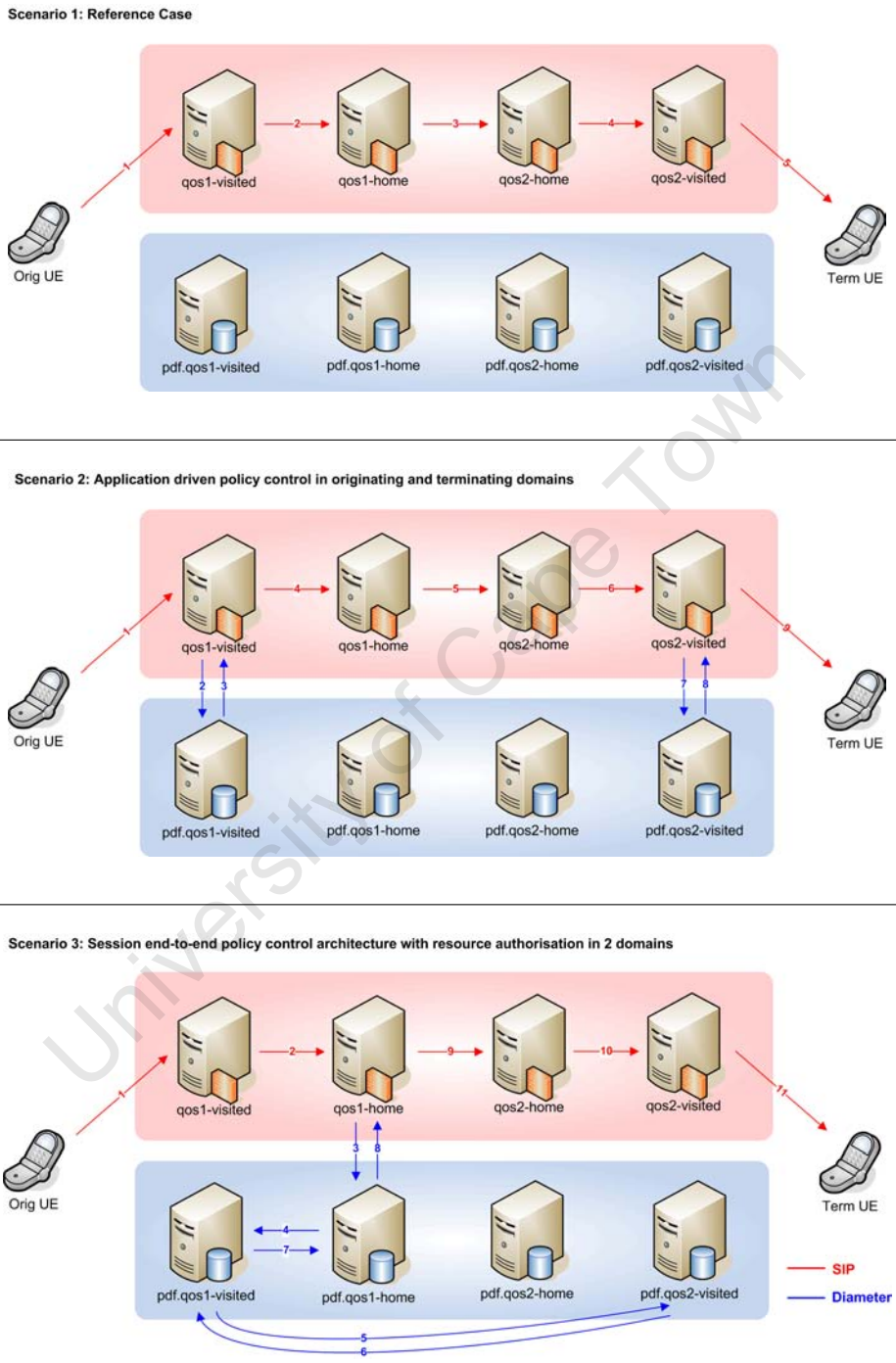

Figure 7.6: Evaluation cases 1 - 3.

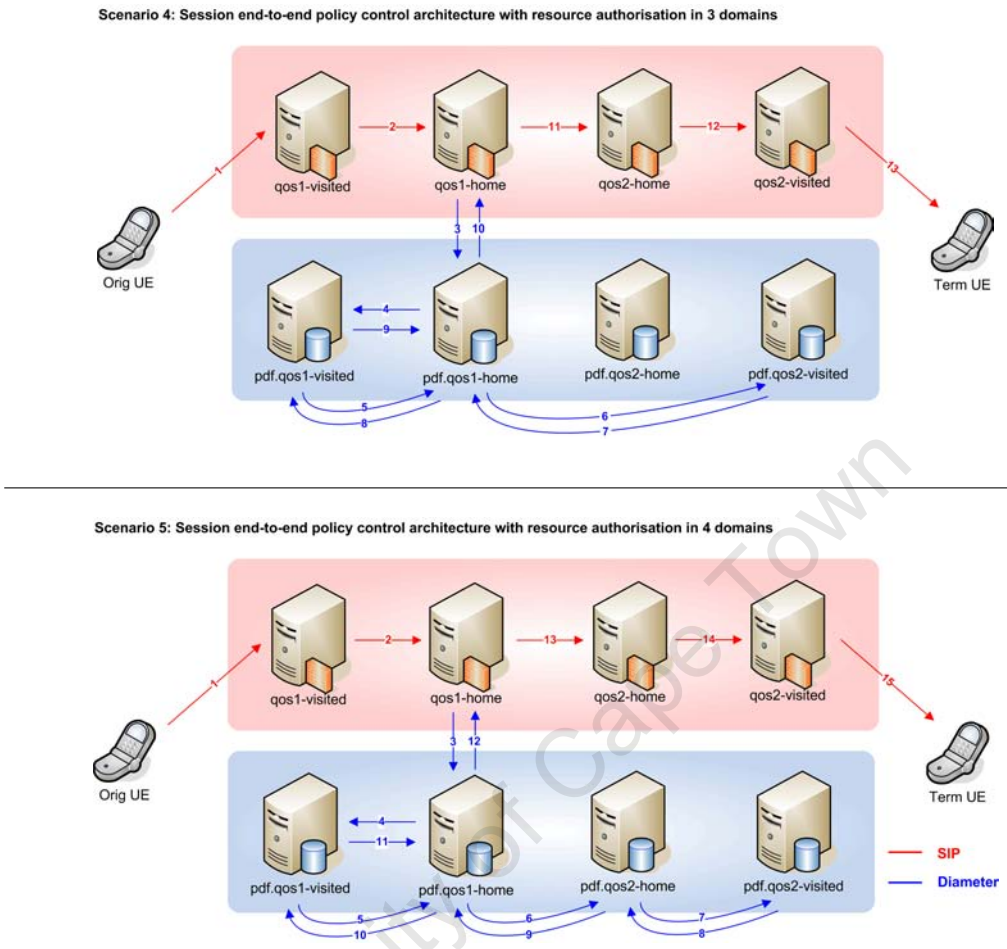

Figure 7.7: Evaluation cases 4 and 5.

14.9%. As the media path becomes more complex in scenario 4 and 5, the delay overhead increases to 16.8% and 17.1% respectively. The 802.11g and 802.16d IP-CANs exhibited similar behaviour. When accessing the network via HSDPA, scenario 3 shows a moderate increase in delay overhead as a percentage, to 8.0%. When adding domains to the media path in scenario 4 and 5, the delay increased by 11.7% and 11.8% respectively.

The reason for the sharp rise in session setup delay for scenario 3 is the fact that this is a worst case scenario where resources are reserved consecutively in each domain on the discovered media path. This and subsequent scenarios include the full application control policy interactions. It is interesting to note that the more complex media paths (scenario 4 and 5) have a limited impact on the delay overhead for all IP-CANs. This can be attributed to the fact that additional signalling is limited to the core network.

Utilising LAN, 802.11g, HSDPA and 802.16d IP-CANs all yield results that meet the

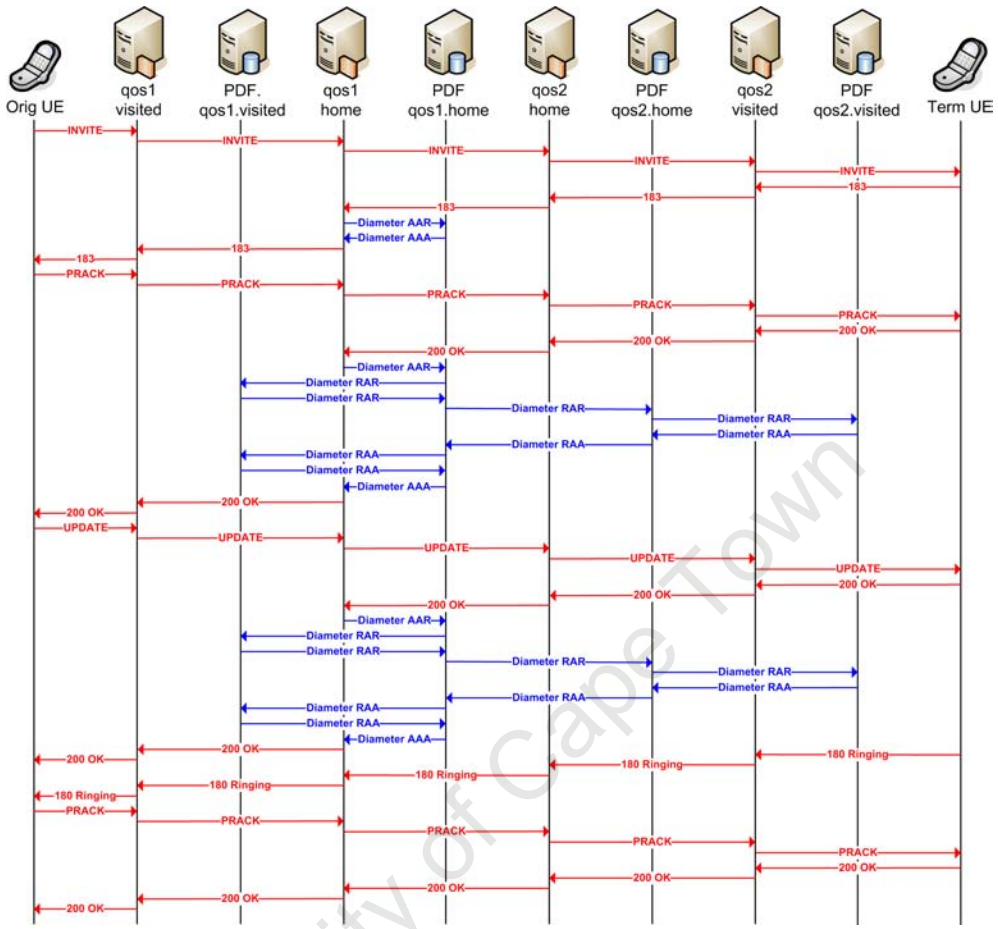

Figure 7.8: Session setup signalling for scenario 5.

criteria of 2 - 5 seconds to establish a multimedia session, stipulated by Guenkova-Luy *et al.* [102]. Considering that these evaluations incorporate the full application control policy interactions and enable QoS connectivity across up to four domains, the results are acceptable. While there is a cost in delay overhead to implement the end-to-end QoS solution, this cost does not rise linearly with the number of domains included on the media path.

7.2.3 Traffic Overhead

The PDF in the home domain of the originating UE plays a critical role when incorporating the end-to-end policy control extensions. This element receives all authorisation requests from the service control plane for a particular session and must process them to calculate the next step on the media path. In certain cases, including scenario 4 and 5,

Table 7.7: Session setup delay results for LAN IP-CAN access.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Minimum (s)	1.118	1.014	1.011	1.021	1.014
Mean (s)	1.288	1.366	1.480	1.505	1.508
95th percentile (s)	1.429	1.632	1.633	1.835	1.838
Delay overhead (%)	-	6.056	14.907	16.848	17.081

Table 7.8: Session setup delay results for 802.11g IP-CAN access.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Minimum (s)	1.212	1.009	1.015	1.221	1.016
Mean (s)	1.300	1.408	1.511	1.535	1.573
95th percentile (s)	1.614	1.632	1.634	1.837	2.414
Delay overhead (%)	-	8.338	16.231	18.077	21.000

this domain may also act as a transit domain; the PDF must also process inter-domain resource requests and enforce PCC rules in the transport plane.

Hence this element is considered as a critical and high capacity node regarding traffic overheads during session setup as it needs to process all messages entering the resource control plane for a particular session. This section presents the incurred traffic overhead during session setup, at the machine hosting the IMS core and resource control elements for the home domain of the originating UE. The results are obtained using the same method as in Section 7.1.3. Table 7.11 shows the results for each of the evaluation scenarios. The results for scenario 4 and 5 are identical as the PDF in the home domain plays an identical role. The traffic is measured between core elements in all domains and summed for each scenario, to give the total core traffic overhead. These results are shown in Table 7.12.

Discussion

The results show a considerable increase in traffic overhead at the home domain of the originating UE. When incorporating end-to-end session policy control in scenario 3, the overhead increases by 84.0% to 105115 bytes. When the media path complexity increases in scenario 4 and 5 the overhead becomes 95.9%, though this is constant no matter how many domains are included on the media path. This increase in traffic by almost 100% is largely due to the XCAP look ups that form part of the application policy control

Table 7.9: Session setup delay results for HSDPA IP-CAN access.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Minimum (s)	2.028	2.036	2.092	1.870	2.036
Mean (s)	2.406	2.460	2.598	2.688	2.690
95th percentile (s)	3.036	3.048	3.061	3.057	3.226
Delay overhead (%)	-	2.244	7.980	11.721	11.804

Table 7.10: Session setup delay results for 802.16d IP-CAN access.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Minimum (s)	1.210	1.220	1.424	1.223	1.216
Mean (s)	1.400	1.522	1.765	1.786	1.799
95th percentile (s)	1.6442	1.634	2.040	2.4	2.438
Delay overhead (%)	-	8.714	26.071	27.571	28.5

interactions. As mentioned before, in a practical deployment these elements will be distributed across several high performance machines, with high speed interconnections. However one solution to decrease this overhead is to implement message compression when sending and receiving XCAP messages [17], though this is not further investigated in this thesis.

The total core traffic tests need to be put into context by the fact that this traffic is spread across four separate administrative domains. The results show that though the overhead increases by 87.2% in scenario 3, the subsequent increases in scenario 4 to 93.7% and in scenario 5 to 112.3% are not significant. Essentially under 20kB of overhead is added to the total core traffic for each additional domain included on the media path, per session.

Table 7.11: The traffic overhead incurred at the home domain of the originating UE for each scenario.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4/5
Total core traffic (bytes)	57134	57134	105115	111900
Traffic overhead (%)	-	0	83.978	95.855

Table 7.12: The traffic overhead incurred in all domains for each scenario.

	Scenario 1	Scenario 2	Scenario 3	Scenario 4	Scenario 5
Total core traffic (bytes)	104187	163609	195027	201812	221209
Traffic overhead (%)	-	57.034	87.189	93.702	112.319

7.2.4 Comparative Processing Delay

This section focuses entirely on the third scenario, representing a network that has adopted both the application driven policy control architecture and end-to-end QoS extensions. Execution stages are defined for the AF, the PDF in the originating UEs home domain, and the PDF in a transit domain. Measuring the processing delays at each execution stage during session setup sheds light on the impact that the extensions have on the performance of each node. The results also demonstrate some of the limitations of the solution architecture that could be investigated in future work. The test procedure is the same as described in Section 7.1.4. The number of media components present in each session request ranges from 1 to 3 to ensure consistency, even though it was shown to have a limited effect on the processing delay in Section 7.1.4. 50 sessions are initiated over the LAN IP-CAN during the evaluation to obtain an accurate representation of the processing delay at each stage.

The following stages of execution are defined for each element:

Application Function

- Stage 1: Determine the signalling path from the information in the SIP session request.
- Stage 2: Create and send Diameter authorisation request to PDF.
- Stage 3: Wait for response to authorisation request from PDF.

PDF in home domain of originating UE / PDF in transit domain

- Stage 1: Extract authorisation information from Diameter authorisation request and calculate next PDF along the media path.
- Stage 2: Create and send Diameter resource request to PDF.
- Stage 3: Wait for response to resource request from PDF.

The comparative delays for each execution stage at the AF, the PDF in the home domain of the originating UE, and the PDF in a transit domain are illustrated in Fig. 7.9, Fig. 7.10 and Fig. 7.11 respectively.

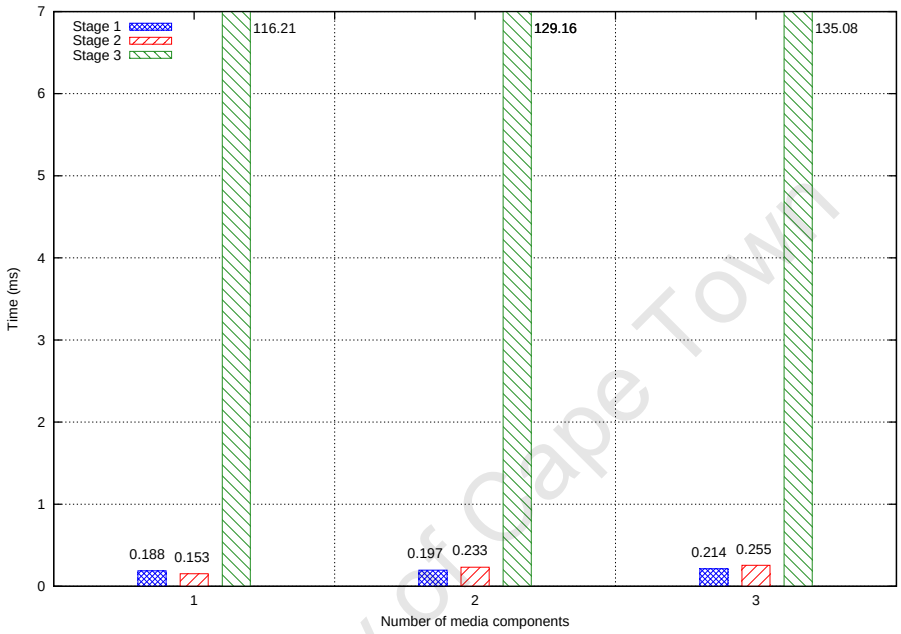

Figure 7.9: Comparative delay for different execution stages at the AF.

Discussion

The results show that the processing at the AF has a minimal contribution compared to the delay when waiting for the response from the resource control plane. This is a due to the XCAP operations that take place but are not shown in these results. The PDF in the home domain of the originating UE performs a minimal amount of processing and most of the delay is incurred while waiting for the inter-domain response. This will depend on the number of subsequent domains on the media path that need to be discovered and traversed. In this case there are two subsequent domains and the delay while waiting for the response is 50.6ms for a typical IMS session with 2 media components. The PDF in the transit domain exhibits similar behaviour to that in the home domain of the originating UE. However in this case there is only one subsequent domain on the media

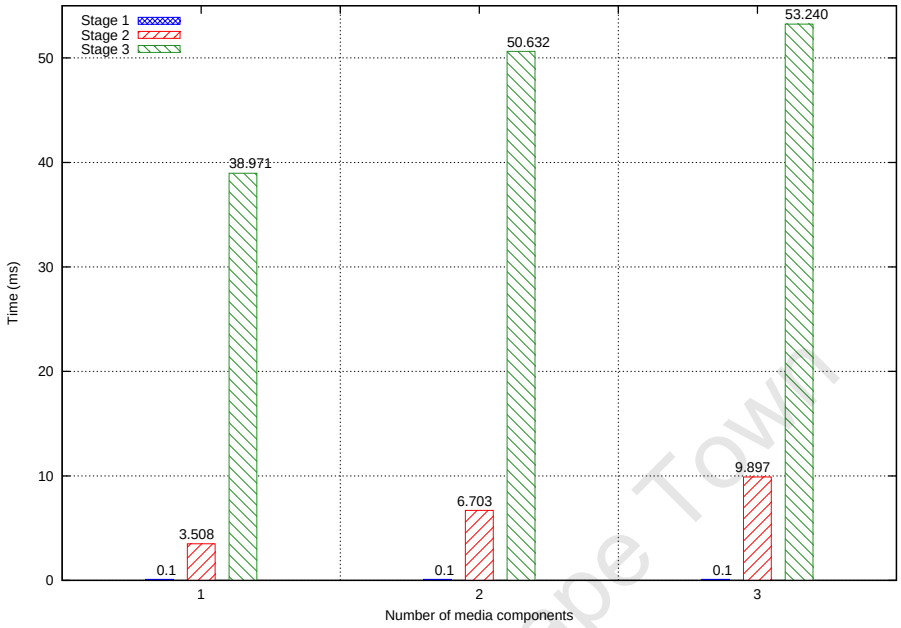

Figure 7.10: Comparative delay for different execution stages at the PDF in the home domain of the originating UE.

path, and the delay while waiting for the inter-domain response is 23.2ms for a typical IMS session.

It is clear from these results and those reported in Section 7.1.4 that the XCAP operations contribute far more to the overall delay than other processing stages, even in a multiple domain scenario. If the XCAP results are ignored the number of media components has a more pronounced effect on the processing delay. This is because a PCC rule is created for each media component and for each inter-domain authorisation request; the PCC rules are processed separately in each domain. The delay while waiting for an inter-domain response depends on the complexity of the media path. For an additional domain included on the media path, a delay increase in the order of 25ms is experienced. This is an insignificant percentage of the overall session setup delay and shows that in the proposed scheme the delay overhead does not increase linearly with the number of domains included on the media path.

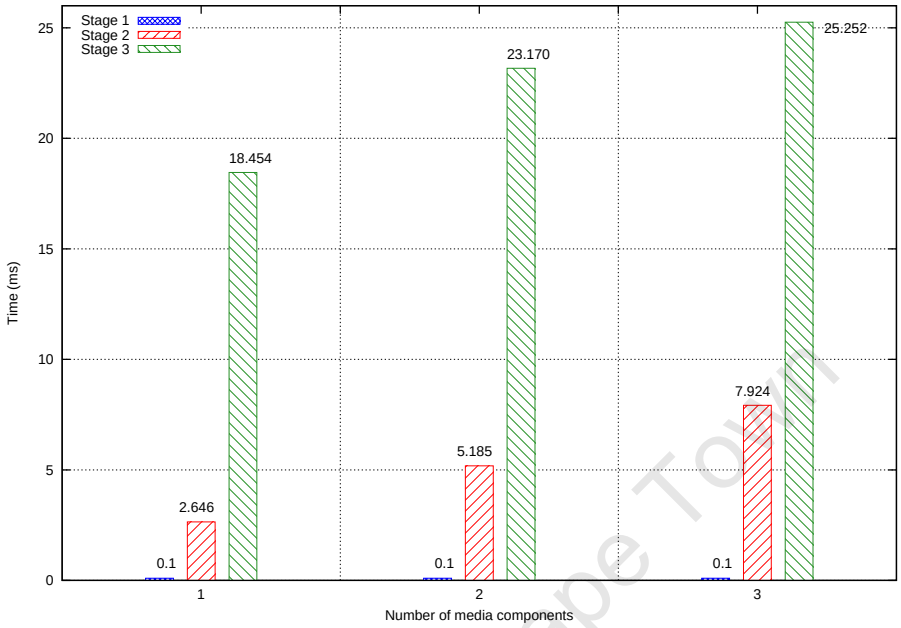

Figure 7.11: Comparative delay for different execution stages at a transit domain PDF.

7.2.5 Load Testing

The architecture is subjected to varying load using the SIPp traffic generator, to assess the effects that the end-to-end extensions might have on scalability and ability to handle load. The extended interactions, particularly the discovery of the end-to-end media path, which is performed successively for each domain, may put strain on the core network when subjected to multiple simultaneous session requests. These tests quantify that effect, and show how the adopted architecture performs with a large subscriber base.

The load tests performed are the same as those described in Section 7.1.5. Two SIPp instances define originating and terminating UEs and sessions are initiated at a rate of 10, 20, 30 and 40 CPS over the LAN IP-CAN. The session requests include 2 media components and a single control policy is retrieved from the XDMS. Only the third scenario is considered in these evaluations. The time to process a SIP message, create an authorisation request and process the reply to the request, is measured at the generic AF. At least 200 measurements are taken subject to each session initiation rate; Table 7.13 shows the processing delay for different session initiation rates.

Table 7.13: Processing delay at the generic AF for different session initiation rates.

	10 CPS	20 CPS	30 CPS	40 CPS
Minimum (s)	19.867	22.586	32.297	29.189
Mean (s)	72.131	79.590	90.003	90.100
95th percentile (s)	135.513	153.711	145.67	159.89

Discussion

The results show that for a session initiation rate of 10 CPS, the mean time to process a single authorisation request at the generic AF is 72.1ms. With a rate of 20 CPS this measurement increases by 10.3% to 79.6ms, with 30 CPS by 24.8% to 90.0ms and with 40 CPS by 24.9% to 90.1ms.

When subject to 40 CPS, 4 sessions out of 200 fail to initiate successfully, while in all other scenarios no failed sessions are reported. As with the load tests in a single domain scenario reported in Section 7.1.5, the increase in processing delay is largely attributed to the XCAP operations. In this scenario the media path includes two administrative domains that perform policy control consecutively. With this in mind, a processing delay of under 100ms for each authorisation request is acceptable performance under load.

7.3 Video on Demand Application Invocation

The previous tests have demonstrated the effectiveness and feasibility of the proposed architecture while using a generic application or service. The signalling has been examined but the impact that the extensions might have on particular applications is also of interest. To avoid the debate of predicting IMS service trends, this thesis examines a well known and fully standardised VoD IMS application that allows a UE to negotiate the VoD session with a Serving Control Function (SCF), to control the media playback using RTSP via a Media Control Function (MCF) and to receive streaming media from a Media Distribution Function (MDF). These elements are hosted on a single AS.

The AS and generic AF can be added to the signalling path consecutively through configuration of the initial Filter Criteria (iFC) and by assigning respective priorities in the HSS. The authorisation requests and PCC rules can be tailored specifically for the requested media stream using VoD specific application policies. Furthermore the end-to-end interactions allow resource authorisation in all traversed domains.

The response time for session establishment is of critical importance for VoD services as users will expect a near immediate reaction. The response time, from an end-user's point of view, is from the first request for service until the requested media is displayed. The traffic overhead, processing delay and load testing analysis have already been examined for IMS signalling scenarios in Section 7.1 and Section 7.2 and hence are omitted from these scenarios.

7.3.1 Scenarios

A typical roaming example is implemented to demonstrate the full architecture, including application policy control and end-to-end QoS capabilities. Two scenarios are defined for these tests. The first is the reference case and does not include any of the extended interactions. A roaming UE attaches via visited domain, *qos-visited*, and is registered with home domain, *qos-home.* The bundled VoD AS is hosted in the UEs home domain and typical session negotiation takes place between the UE and the SCF component. The MDF streams the requested media to the UE once the session is established.

The second scenario incorporates the full resource management framework. The generic AF is hosted in the home domain of the UE and is added to the signalling path before the VoD AS. The generic AF performs application policy control and discovers signalling routes, while the PDF discovers the end-to-end media path and authorises resources in both the visited and home domains.

The HSDPA IP-CAN is utilised for these evaluations. This is the most suitable access technology for mobile delivered multimedia content and allows for a realistic evaluation of a carrier grade IMS service deployment. The scenarios are illustrated in Fig. 7.12.

7.3.2 VoD session setup

VoD session setup, as dictated in the TISPAN IMS-based IPTV Stage 3 specification [76], does not include compulsory SIP preconditions to negotiate the session. For certain services these extensions are not required and therefore not deemed compulsory for all IMS sessions [6]. In this scenario the AF is located in the same domain as the VoD AS, which is the terminating domain, hence full IMS signalling is not necessary to determine the end-to-end signalling path.

The originating UE may request information about the VoD service using the OPTIONS request, and use the information encapsulated in the 200 OK response to create

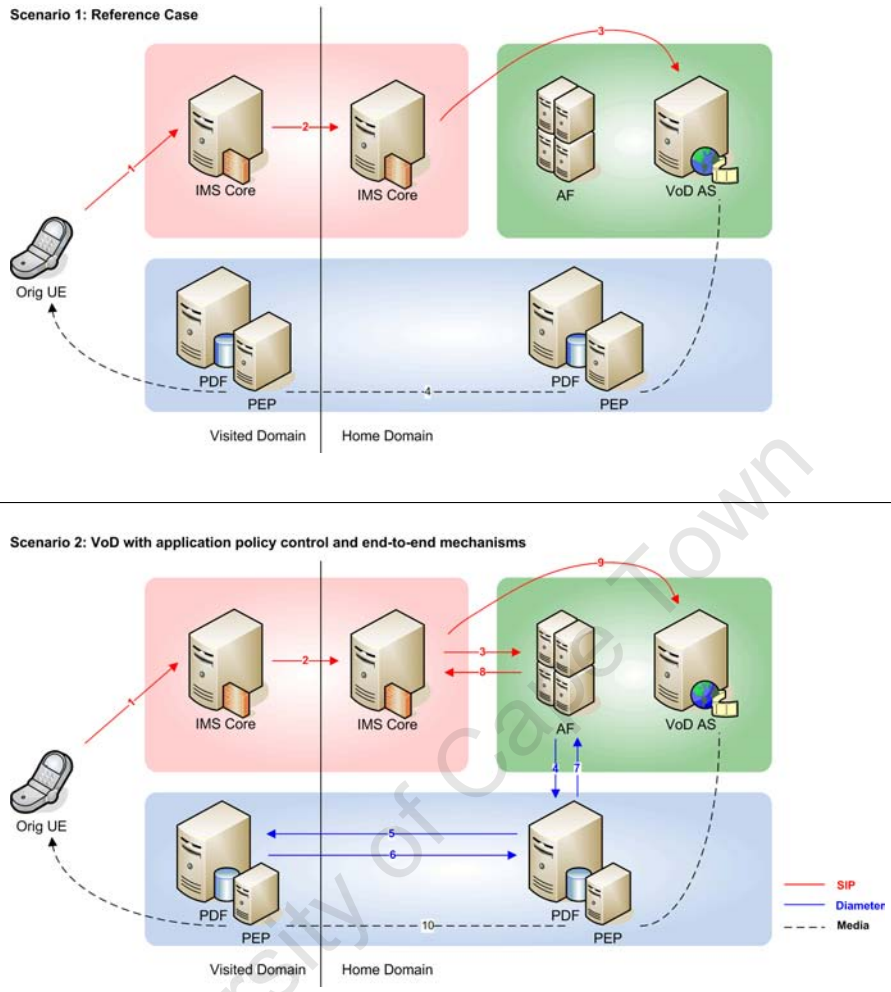

Figure 7.12: The evaluation scenarios for VoD application invocation.

the initial INVITE request for the session. Following this normal IMS session initiation, bar precondition extensions, takes place. The VoD AS sends a 200 OK response with relevant RTSP connection information included in the SDP body. The UE responds with a final ACK and the AS initiates the RTSP session. The session setup delay for both scenarios is measured, to discover the effects that the proposed architecture has on a particular application invocation. The results are obtained using the *gettimeofday* function; session setup includes signalling from the first DNS look up for the OPTIONS request until the receipt of the 200 OK in response to the INVITE request. Fig. 7.13 shows the session setup signalling for the VoD service for the second scenario. For simplicity the IMS core elements are grouped and the XDMS and transport plane interactions for application driven policy are omitted.

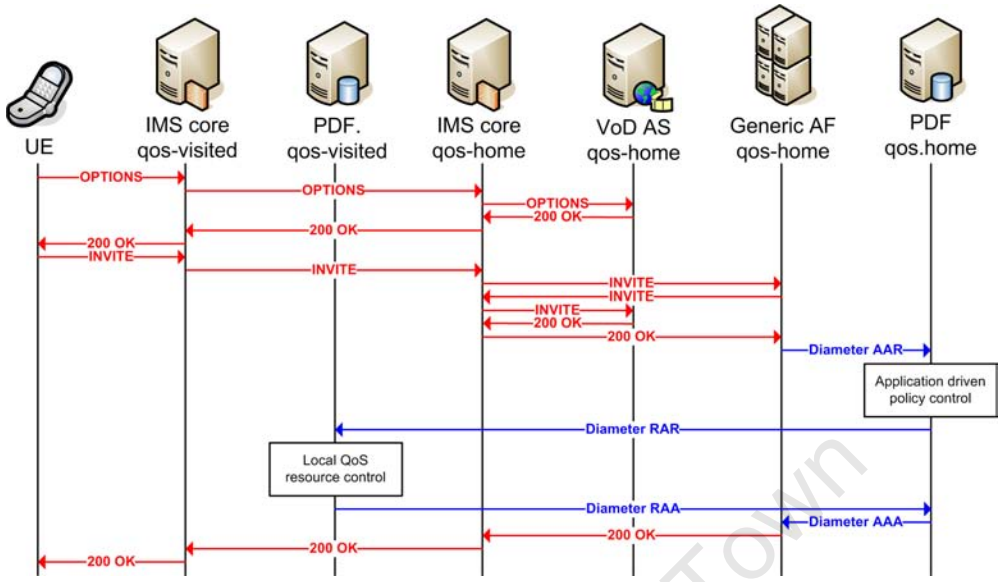

Figure 7.13: Session setup signalling for scenario 2.

The tests are run over 50 iterations to ensure accurate representation. The averaged results for session setup delay are shown in Table 7.14 and the full set of session delay measurements is illustrated in Fig. 7.14.

Table 7.14: Session setup delay results for VoD service invocation over HSDPA.

	Scenario 1	Scenario 2
Minimum (s)	1.795	1.962
Mean (s)	2.143	2.399
95th percentile (s)	2.762	3.089
Delay overhead (%)	-	11.946

7.3.3 Discussion

This evaluation examined the proposed architecture when used in coordination with a VoD AS, to determine the effects that the extended signalling might have on service provisioning. In scenario 2 the VoD AS has access to end-user preferences included in the VoD control policy that allow the service delivery to be customised; i.e. a lower quality service configuration is used to cater for the bandwidth constrained HSDPA access. The creation of the PCC rule includes domain, subscription, access and VoD specific control policies, refining the rule to be specific to the requested service. This

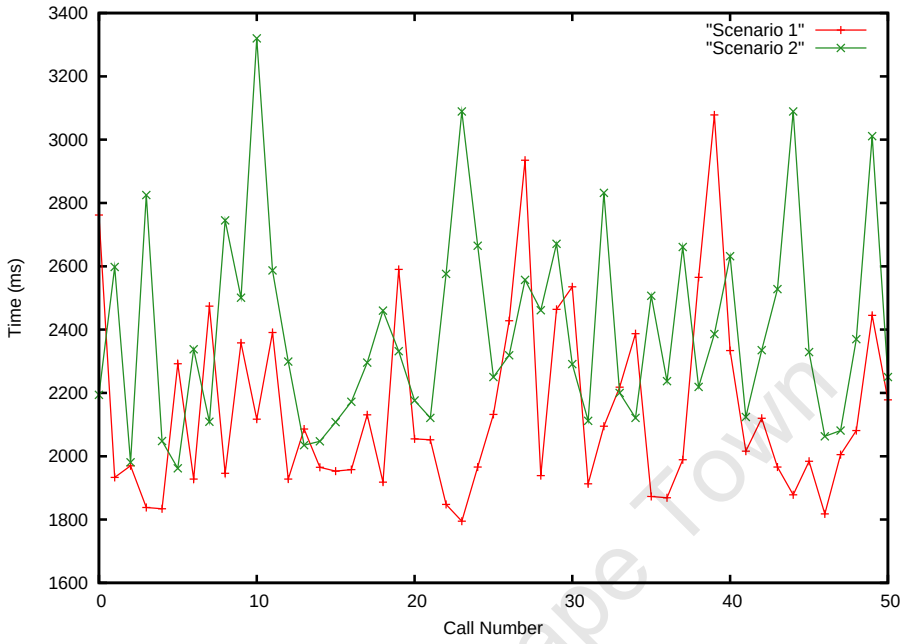

Figure 7.14: Full set of session setup delay measurements.

PCC rule is distributed across all traversed domains where local QoS resource control took place.

The results show that incorporating the application driven and end-to-end policy control framework adds 12.0% delay overhead, or 256ms. The session setup delays are slightly lower than those recorded in Section 7.2.2 because precondition extensions are not implemented.

The delay overheads represent an insignificant percentage of the total session setup delay. The resource authorisation functionality could be implemented within the AS should optimisation be necessary. However a separate generic AF element ensures modularity and allows new ASs to easily inherit advanced QoS support.

7.4 Summary

This chapter has described a range of testbed evaluations, to demonstrate proof of concept and identify critical performance attributes. The modular design and imple-

mentation allowed each concept to be evaluated separately. Comprehensive scenarios incrementally incorporated the proposed architecture into the testbed to highlight the performance of individual components.

The results show that the testbed has been implemented successfully and few design shortcomings were experienced. The results demonstrate the viability of the proposed concepts. Most importantly it has been shown that the overheads introduced as a result of the extended interactions do not significantly affect end-user experience and these overheads fall within acceptable criteria. Th next chapter concludes the thesis and recommendations for future work are proposed.

Chapter 8

Conclusions and Recommendations

The previous chapters have presented an in depth analysis of the challenges facing resource management frameworks and how these can be overcome within the NGN architecture, with specific application to the emerging IMS platform. This chapter presents conclusions drawn from the thesis and summarises contribution. Recommended areas for further study are identified.

8.1 Conclusions

8.1.1 IMS Deployment

The first chapter of this thesis discusses growing access and bandwidth proliferation, the convergence of fixed and mobile technologies and the steady migration to an All-IP architecture. The IP Multimedia Subsystem (IMS) is a candidate technology to provide a non-disruptive strategy in the move to All-IP and to facilitate fixed and mobile convergence. The IMS hype phase is over and the core architecture is largely finalised as of 3GPP Release 8. The centralisation of IMS standardisation has helped alleviate interoperability concerns and it can be concluded that widescale commercial deployments will accelerate in the near future. This will be fueled in part by the deployment of the Long Term Evolution (LTE) and Evolved Packet Core (EPC) technologies, of which IMS is a central IP service element.

This thesis found that the IMS as an emerging technology still faces challenges to adoption, despite the centralised Release 8 specification. In particular the business case for deploying IMS technology is a concern; typically deployments are done on an

application-by-application basis and the point at which IMS becomes more cost-effective than service specific approaches is difficult to determine. However the most critical challenge is that posed by the widespread proliferation of Web 2.0 services. This environment is not robust enough to be used by network operators for revenue generating services. IMS operators will need to justify charging for services that are typically free of charge in the Internet space. It was found that reliability and guaranteed transport of multimedia services by the efficient management of resources will be critical to differentiate IMS services.

8.1.2 Common PCC Framework

The review of IMS/NGN resource management frameworks reveals a fragmented approach involving numerous standardisation bodies and overlapping scopes. Three primary frameworks, the 3GPP PCC, TISPAN RACS and ITU-T RACF are analysed in detail regarding functional architecture and reference points in this thesis. Architectural alignment is performed and a centralised framework, the Common PCC framework, is presented. This is a first attempt at harmonising the duplicate functions across the architectures. A generic set of terms and elements are defined to allow for more coherent and focused research in the future.

8.1.3 Policy Based Resource Management Challenges

The literature and standardisation reviews identify shortcomings in the Common PCC framework. This thesis concentrates on two identified areas: vertical coordination and horizontal coordination of resources. Vertical coordination refers to the interaction between applications requesting resources and the transport plane devices that carry the application traffic. The Common PCC framework does not specify mechanisms for policy representation, policy prioritisation, policy provisioning or application-policy interaction. The deployment of new services should not require QoS standardisation or network upgrade, though in the current architecture advanced multimedia services are not catered for. Essentially application developers have very little control over the way their services are handled in the transport plane.

Horizontal coordination of resources refers to inter-domain interactions to facilitate end-to-end QoS connectivity. The thesis found that end-to-end QoS mechanisms in the Common PCC framework are elementary. QoS connectivity across all traversed

transport segments is not supported by any of the reviewed architectures and inter-domain reference points are for further study or preliminary at best. The inter-domain routing algorithms proposed in the literature, are path-coupled or variations thereof, which suffer from poor compatibility with legacy networks and existing QoS mechanisms. These challenges need to be considered and addressed by operators when formulating their NGN strategies.

8.1.4 Application Driven Policy Control

To address the challenge of vertical coordination of resources, this thesis proposes an application driven policy control architecture that incorporates end-user and service requirements into the QoS negotiation procedure. Through the definition of application specific control policies, this framework gives application developers control over the policies that govern QoS resource control. Generic control policies allow services to inherit advanced QoS support without the need for further standardisation. The architecture defines extensions to the AF, PDF and policy repository elements within the Common PCC framework. These extensions are entirely conformant with 3GPP and IETF standards and new concepts can be incorporated into the solution architecture. This architecture introduces no additional round trip times and shows that such enhanced functionality can be realised without increasing UE complexity or having a drastic effect on end-user experience.

8.1.5 Session Based End-to-end Policy Control

The thesis proposes a session based end-to-end policy control architecture to support inter-domain coordination across IMS administrative domains. This architecture defines novel mechanisms at the service control and resource control planes to discover signalling routes and bind them to the media path. This effectively allows applications to request end-to-end QoS-enabled paths, where resources are reserved in all traversed transport segments. The architecture uses inter-domain reference points to pass QoS requirements over the end-to-end path as part of the application signalling. This approach has the benefit of backward compatibility with existing transport plane technology, allows the reuse of autonomous and independent reservation mechanisms in each domain and does not require the exchange of sensitive topology information. This architecture demonstrates end-to-end QoS connectivity using standard interfaces within the Common PCC

framework, and shows that transport plane overhaul is not necessary for such functionality.

8.1.6 Open Testbed Implementation

The proposed architectures are implemented in a practical testbed to facilitate proof of concept and provide a platform for evaluations. IMS and resource management frameworks are complex technologies and need to be exposed to a wide set of application developers and researchers to accelerate technology maturity and encourage future innovation in the field. For this reason a primary goal of the testbed implementation is to ensure that it can be entirely reproduced and legally extended. This is achieved through the use of Free and Open Source Software (FOSS) for all testbed components. This enables a rapid implementation of the proposed architectures in a practical setting and ensures that all subsequent evaluations can be reproduced and verified, providing a convenient point of departure for future work and innovation. Hence the findings show that a comprehensive IMS network emulation can be implemented and verified using open source tools like the UCT IMS Client, OpenXCAP and the Open Source IMS Core.

8.1.7 Session Setup Delay

Session setup is an important procedure essential to the multimedia-oriented telecommunications network and is a key metric for evaluating resource management optimisations. The performance evaluations consider the effects on session setup delay when incorporating the application driven policy control architecture in a single domain, and when extended to inter-domain scenarios. The number of domains included in the scenarios are varied to demonstrate different levels of media path complexity.

The evaluations in a single domain for LAN, 802.11g, 802.16d and HSDPA IP-CANs show delay increases in the order of 100ms or less, when incorporating the application driven policy control interactions. The delay overhead when utilising the LAN IP-CAN is 8.5%, this drops to 3.2% for the HSDPA IP-CAN.

When incorporating the end-to-end mechanisms, the delay overhead increases sharply. For LAN access the delay overhead is 14.9% when extending the architecture to incorporate multiple domains. This figure dropps to 8.0% for the HSDPA access. The more complex media paths with multiple transit domains have a limited effect on delay overhead because no new round trip times are introduced.

Session setup delay is also examined when using the proposed framework in coordination with a VoD AS to determine the effects that the signalling extensions might have on application invocation and service provisioning. These evaluations are carried out across multiple domains and the delay overhead is 12.0% when the proposed architectural extensions are incorporated.

It can be concluded that while there is a cost in delay overhead when implementing the complete architecture, this cost falls within performance criteria for mobile multimedia applications defined by Guenkova-Luy *et al.* [102]. Considering that the evaluations incorporate full application policy control interactions and enable QoS connectivity across up to four domains, this cost in delay overhead is acceptable and does not rise linearly with the number of domains included on the media path. The effect on application invocation is similarly acceptable. The VoD service is customised to end-user requirements, and the PCC rule is tailored specifically to the requested service and distributed across multiple administrative domains.

8.1.8 Traffic Overhead

Evaluations are carried out to determine the effect the proposed architectural extensions might have on traffic overhead within the core network. Typically core elements are high capacity nodes and an increase of only a few messages could overload them and decrease user utility in the network. When incorporating the application policy control interactions in a single domain, the signalling traffic in the core network increases by 78.4%. When extending the architecture to multiple domains and incorporating the end-to-end policy control extensions, the traffic overhead between core elements in all domains increases to 87.2%. It was found that additional domains included on the media path do not substantially increase the traffic in the core network.

The results are put into context by the fact that in a practical environment these elements would be distributed across high performance machines with high speed interconnections. Furthermore the core traffic overhead for the end-to-end policy control evaluations is spread across four administrative domains. Essentially the extended application policy control interactions add less than 60kB of additional core traffic per session, and the end-to-end policy control interactions add less than 20kB of core traffic per additional domain included on the media path. It can be concluded that this will not overload network elements and have an effect on end-user experience. This is

shown by the fact that the increase in traffic overhead does not cause a proportional increase in session setup delay, primarily because additional signalling was limited to the core network. However it is recommended that high bandwidth pipes provide the inter-connection between these elements. This would ensure that even low specification equipment would not be adversely affected by the increase in traffic overhead.

8.1.9 Comparative Processing Delay

The proposed architectures extend the AF and PDF elements and it is important to examine the effect that these extensions have on individual performance. Different execution stages are defined for each element and evaluations are carried out to determine the processing delays for each stage. These tests are carried out in single domain and multiple domain scenarios with full application driven and session based end-to-end policy control interactions.

The results show that the XCAP operations, to retrieve and store control policy documents from the XDMS, contribute most significantly to the overall processing delay for all scenarios. The number of media components has a minor effect on the processing delay, as this does not effect the number of XCAP operations. However the difference is more pronounced in multiple domain scenarios because each media component results in individual PCC rules, which are created and processed separately in each domain. In the multiple domain scenario the processing delay increase is in the order of 25ms for each additional domain included on the media path.

It can be concluded that none of the execution stages incur unacceptable processing delays, as the session setup delay results fall within acceptable criteria. However there is room for optimisation regarding the XCAP operations. The processing delay incurred for additional domains included on the media path is a small percentage of the total session setup delay. This further illustrates the fact that the delay overhead does not rise linearly with the number of domains included on the media path.

8.1.10 Load Testing

The proposed architecture is subject to signalling loads ranging from 10 CPS to 40 CPS and the time to process a single authorisation request is measured. The testbed is by no means a commercial grade implementation where hundreds of sessions will need to be processed per second, standard PC hardware is used throughout. This evaluation

assesses the viability of the proposed concepts and extensions when processing multiple simultaneous requests.

In the single domain scenarios all sessions are successfully established and the processing delay is under 40ms for all session initiation rates. When extended to inter-domain scenarios the processing delay increases but is still less than 100ms for all session initiation rates. However when subject to 40 CPS, 4 sessions out of 200 fail to initiate successfully.

The proposed concepts behave predictably when subject to simultaneous multiple session requests and the processing delays are acceptable for all session initiation rates. However the XCAP operations should be examined and optimised to limit session failures as a result of SIP retransmission timeouts.

8.2 Future Work

8.2.1 Application Driven Policy Control Signalling

The evaluation results show that the signalling overhead and subsequent effect on session setup delay is acceptable for real time multimedia services with end-to-end QoS connectivity. However future modules may add further overhead and create the need for signalling optimisation.

The retrieval of the XML control policy documents contributes the most significant part of the processing delay and results in failed sessions when subject to high signalling load. Further study could examine the optimisation of this process. Possible solutions include XML message grouping to limit the number of XCAP read operations, locally stored policy documents and associated synchronisation, and XML message compression.

8.2.2 Session Based Route Discovery

The route discovery mechanisms proposed in this thesis perform inter-domain coordination at the service control plane and pass QoS resource requests between domains through application signalling. This approach examines inter-domain routing from a high level, where domains are considered as black box nodes and only inter-domain routes are taken into account. Despite its shortcomings, the alternative approach of passing QoS requirements over the end-to-end path through path-coupled QoS signalling, will feature in future networks. This is partly due to the large scale efforts of the IETF with

the Next Steps In Signalling (NSIS) framework. Future work should examine hybrid approaches and migration strategies to incorporate both of these methods for requesting QoS enabled paths for advanced multimedia services.

8.2.3 Resource Management, Mobility and Security

The EPC framework handles the heterogeneity of 3GPP IP-CANs and supports various mobility mechanisms. Mobility is closely linked to resource management, in that when a mobile node attaches to a new network, resources need to be allocated and policy rules created and enforced. It is recommended that future research investigate the effects that the proposed application driven policy extensions and end-to-end QoS mechanisms have on mobility issues.

The IMS framework supports enhanced security mechanisms that prevent address spoofing and can ensure the integrity of connection between UE and P-CSCF using IP Security (IPSec). Further security measures include the Topology Hiding and Interworking Gateway (THIG) used to encrypt all signalling information related to a specific domain and in this way hide sensitive topology information from neighbouring domains. The extensions proposed in this thesis do not compromise any of these security mechanisms, however the end-to-end policy control interactions require administrative domains to base routing decisions on signalling information provided by other domains. Future research should investigate the trust and security implications of this architecture.

8.2.4 Emerging Access Technologies

The LTE and 802.16e access technologies are both candidates for the IMT-Advanced specification for NGN Mobile Systems. Extending QoS resource management to the access network is critical. In particular exploiting existing QoS mechanisms in the access network under control of the Common PCC framework, and mapping the QoS and service characteristics between domains will be necessary to enable end-to-end QoS connectivity. Future work should concentrate on integrating various access technologies with the Common PCC framework to ensure resource management in the link from the end-user to the first hop router. The testbed implementation has an elementary transport plane, this should be extended to incorporate resource management on the devices in the access network.

8.2.5 Resource Management as a Service

The IMS service capability and interoperability manager or service broker is an element responsible for exposing associated service capabilities between ASs and controlling the way in which services interact with one another. The processes defined involve policy based access control to service enabler APIs, similar to the policy based access control to resources through the Common PCC framework. A compelling and interesting area for future work is the identification of common functionalities between these two architectures, with the goal of implementing resource management as a service enabler or as a building block within the services cloud. This approach would simplify resource reservation from the application developers perspective and allow for a completely generic approach. However open questions exist regarding policy refinement and enforcement on technology specific devices.

8.2.6 Common Standardisation

The realisation of the Common IMS specification and centralised management of associated standardisation has come a long way in helping resolve confusion and interoperability concerns. The standardisation of the IMS/NGN resource management framework has been even more fragmented, resulting in weak functional and interface specifications. Harmonisation work has begun on the reference point between the Common PCC framework and the service control plane, the Rx/Gq' interface. This is a joint initiative between 3GPP and TISPAN but is only a proposed work item under 3GPP Release 9 with few deliverables thus far. Furthermore the overall harmonisation of the frameworks is not investigated.

With some forethought a comprehensive end-to-end framework based on dynamic policies can flexibly control resources and provide a sound business case for deploying IMS services. However it is imperative that the widescale harmonisation of NGN/IMS resource management frameworks takes place and that the issues raised in this thesis are further developed and addressed within the Common PCC specifications. The alternative is general interoperability issues that could render end-to-end QoS provisioning for advanced multimedia services almost impossible.

Bibliography

[1] Internet World Stats, "Internet Usage Statistics - World Internet users and Population Stats," *http://www.internetworldstats.com/stats.htm*, June 2008. Internet usage information comes from data published by Neilsen/Netratings and the International Telecommunications Union.

[2] J. Waclawsky, "IMS: A Critique of the Grand Plan," *Business Communications Review*, pp. 54–58, October 2005.

[3] C. Systems, "Global IP Traffic Forecast and Methodology, 2006-2011," January 2008.

[4] ITU-T, "Rec. Y. 2001 General Overview of NGN," 2004.

[5] ITU-R, "Rec. M. 1645 Framework and Overall Objectives of the Future Internet of IMT-2000 and System beyond IMT-2000," 2003.

[6] G. Camarillo and M. Garcia-Martini, "The 3G IP Multimedia Subsystem (IMS): Merging the Internet and Cellular Worlds 3rd Edition," *John Wiley and Sons Ltd.*, September 2008.

[7] 3GPP, "3GPP Specifications - Release (and phases and stages)," *http://www.3gpp.org/Releases*, Accessed November 2008.

[8] T. Magedanz, N. Blum, and S. Dutkowski, "Evolution of SOA Concepts in Telecommunications," *Computer*, vol. 40, pp. 46–50, November 2007.

[9] G. Camarillo, T. Kauppinen, M. Kuparinen, and I. M. Ivars, "Towards an Innovation Oriented IP Multimedia Subsystem," *IEEE Communications Magazine*, pp. 130–135, March 2007.

[10] Zenith Optimedia, "Adspend Forecasts April 2009," *http://www.zenithoptimedia.com/gff/pdf/Adspend forecasts April 2009.pdf*, April 2009.

[11] R. McConville, "Report: IMS Goes from Hero to Zero," *Light Reading*, July 2007.

[12] NGN Forum, IMS Forum in cooperation with the University of New Hampshire InterOperability Lab, "IMS NGN Report Card (Plugfest 3, 4 and 5)," *http://www.imsforum.org/ims-report-card*, November 2008.

[13] J. Crawshaw, "Carriers on IMS: Fear, Uncertainty and No Doubt," *Light Reading*, March 2006.

[14] Infonetics Research, "IMS Equipment and Subscribers report," *http://www.infonetics.com/pr/2009/4q08-ims-market-highlights.asp*, March 2009.

[15] J. Robson, "The LTE/SAE Trial Initiative: Taking LTE/SAE from Specification to Rollout," *IEEE Communications Magazine*, pp. 82–88, April 2009.

[16] R. Ludwig, H. Ekstrom, P. Willars, and N. Lundin, "An Evolved 3GPP QoS Concept," *Proceedings of 2006 IEEE Vehicular Technology Conference (VTC'06)*, pp. 388–392, September 2006.

[17] V. Ozianyi, R. Good, N. Carrilho, and N. Ventura, "XML-Driven Framework for Policy-Based QoS Management of IMS Networks," *Proceedings of 2008 IEEE Global Communications Conference (GLOBECOM'08)*, December 2008.

[18] ITU-T, "Rec. Y. 2111 Resource and Admission Control Functions in Next Generation Networks," 2006.

[19] J. Song, M. Chang, and S. Lee, "Overview of ITU-T NGN QoS Control," *IEEE Communications Magazine*, pp. 116–123, September 2007.

[20] C. Rothenberg and A. Roos, "A Review of Policy-Based Resource and Admission Control Functions in Evolving Access and Next Generation Networks," *Journal of Network and Systems Management - Special Issue on Management of IP Multimedia Subsystem*, vol. 16(1), pp. 14–45, March 2008.

[21] L. Skorin-Kapov, M. Mosmondor, O. Dobrijevic, and M. Matijasevic, "Application-Level QoS Negotiation and Signaling for Advanced Multimedia Services in the IMS," *IEEE Communications Magazine*, pp. 108–116, July 2007.

[22] C. Kamienski, J. Fidalgo, R. Dantas, D. Sadok, and B. Ohlman, "XACML-Based Composition Policies for Ambient Networks," *Proceedings of 2007 8th IEEE Workshop on Policies for Distributed Systems and Networks (POLICY'07)*, pp. 77–86, June 2007.

[23] Fraunhofer Institute FOKUS, "Open Source IMS Core," *http://www.openimscore.org/*.

[24] D. Waiting and R. Good, "UCT IMS Client," *http://uctimsclient.berlios.de/*.

[25] 3GPP, "TS 23.234 3GPP system to Wireless Local Area Network interworking V8.0.0," December 2008.

[26] 3GPP, "TS 23.203 Policy and Charging Control Architecture V8.4.0," December 2008.

[27] F. Alfano, P. McCann, and T. Towle, "IMS Service-Based Bearer Control," *Bell Labs Technical Journal*, vol. 10(4), pp. 151–166, 2006.

[28] 3GPP, "TS 23.401 General Packet Radio Service enhancements for Evolved Universal Terrestrial Radio Access Network V8.4.1," December 2008.

[29] 3GPP, "TS 29.214 Policy and Charging Control over Rx reference point V.8.3.0," December 2008.

[30] 3GPP, "TS 29.212 Policy and Charging Control over Gx reference point V8.2.0," December 2008.

[31] 3GPP, "TS 29.215 Policy and Charging Control over S9 reference point V8.0.2," December 2008.

[32] J. Balbas, S. Rommer, and J. Stenfelt, "Policy And Charging Control in the Evolved Packet System," *IEEE Communications Magazine*, pp. 68–74, February 2009.

[33] ETSI TISPAN, "ES 282 003 Resource and Admission Control Subsystem (RACS) Functional Architecture," July 2008.

[34] ETSI TISPAN, "TS 181 018 Requirements for QoS in a NGN," August 2007.

[35] ETSI TISPAN, "TR 182 022 Architectures for QoS handling," December 2007.

[36] ETSI TISPAN, "TS 183 017 Diameter protocol for session based policy set-up information exchange between the AF and the SPDF," September 2008.

[37] ETSI TISPAN, "ES 283 026 Protocol for QoS reservation information exchange between the SPDF and the A-RACF," November 2008.

[38] ETSI TISPAN, "ES 28 018 H.248 Profile for controlling Border Gatewat Functions (BGF) in the Resource and Admission Control Subsystem (RACS)," June 2008.

[39] M. Callejo-Rodriguez and J. Enriquez-Gabeiras, "Bridging the Standardization Gap to Provide QoS in Current NGN Architectures," *IEEE Communications Magazine*, pp. 132–137, October 2008.

[40] 3GPP, "TR 23.822 Framework for Gq'/Rx Harmonization V0.2.0," February 2008.

[41] V. Hilt, G. Camarillo, and J. Rosenberg, "A Framework for Session Initiation Protocol (SIP) Session Policies - draft-ietf-sip-session-policy-framework-00," *IETF Internet Draft (work in progress)*, October 2006.

[42] V. Hilt, G. Camarillo, and J. Rosenberg, "A Framework for Session Initiation Protocol (SIP) Session Policies - draft-ietf-sip-session-policy-framework-05," *IETF Internet Draft (work in progress)*, November 2008.

[43] D. Petrie and S. Channabasappa, "A Framework for Session Initiation Protocol User Agent Profile Delivery - draft-ietf-sipping-config-framework-15," *IETF Internet Draft (work in progress)*, February 2008.

[44] V. Hilt and G. Camarillo, "A Session Initiation Protocol (SIP) Event Package for Session-Specific Session Policies - draft-ietf-sipping-policy-package-05," *IETF Internet Draft (work in progress)*, July 2008.

[45] V. Hilt, G. Camarillo, and J. Rosenberg, "A User Agent Profile Data Set for Media Policy - draft-ietf-sipping-media-policy-dataset-06," *IETF Internet Draft (work in progress)*, July 2008.

[46] L. Skorin-Kapov and M. Matijasevic, "End-to-end QoS Signaling for Future Multimedia Services in the NGN," *Proceedings of 2006 6th International Conference on Next Generation Teletraffic and Wired/Wireless Advanced Networking (NEW2AN)*, pp. 408–419, June 2006.

[47] A. Albaladejo, F. Gouveia, M. Corcici, and T. Magedanz, "The PCC Rule in the 3GPP IMS Policy and Charging Control Architecture," *Proceedings of 2008 IEEE Global Communications Conference (GLOBECOM'08)*, November 2008.

[48] C. Kamienski, J. Fidalgo, R. Dantas, D. Sadok, and B. Ohlman, "Design and Implementation of a Policy-based Management Framework for Ambient Networks: Choices and Lessons Learned," *Proceedings of 2008 20th IEEE/IFIP Network and Operations Management Symposium (NOMS'08)*, pp. 775–778, April 2008.

[49] F. Baroncelli, B. Martini, V. Martini, and P. Castoldi, "Supporting Control Plane-enabled Transport Networks within ITU-T Next Generation Networks (NGN) architecture," *Proceedings of 2008 20th IEEE/IFIP Network and Operations Management Symposium (NOMS'08)*, pp. 271–278, April 2008.

[50] R. Hancock, G. Karagiannis, J. Loughney, and S. V. den Bosch, "RFC 4080 - Next Steps In Signaling (NSIS) : Framework," June 2005.

[51] H. Schulzrinne and R. Hancock, "GIST: General Internet Signalling Transport - draft-ietf-nsis-ntlp-17," *IETF Internet Draft (work in progress)*, October 2008.

[52] J. Manner, G. Karagiannis, and A. McDonal, "NSLP for Quality-of-Service Signaling - draft-ietf-nsis-qos-nslp-16," *IETF Internet Draft (work in progress)*, February 2008.

[53] M. Brunner, "RFC 3726 - Requirements for Signaling Protocols," April 2004.

[54] G.Ash, A. Bader, C. Kappler, and D. Oran, "QoS NSLP QSPEC Template - draft-ietf-nsis-qspec-21," *IETF Internet Draft (Work in Progress)*, November 2008.

[55] A. Bader, L. Westberg, G. Karagiannis, C. Kappler, and T. Phelan, "RMD-QOSM - The Resource Management in Diffserv QOS Model - draft-ietf-nsis-rmd-13," *IETF Internet Draft (work in progress)*, July 2008.

[56] G. Ash, A. Morton, M. Dolly, P. Tarapore, C. Dvorak, and Y. Mghazli, "Y.1541-QOSM – Y.1541 QoS Model for Networks Using Y.1541 QoS Classes - draft-ietf-nsis-y1541-qosm-07," *IETF Internet Draft (work in progress)*, October 2008.

[57] R. Good, F. Gouveia, S. Chen, N. Ventura, and T. Magedanz, "Critical Issues for QoS Management and Provisioning in the IP Multimedia Subsystem," *Journal of Network and Systems Management*, vol. 16(2), pp. 129–144, April 2008.

[58] X. Masip-Bruin, M. Yannuzzi, R. Serral-Gracia, J. Domingo-Pascual, J. Enriquez-Gabeiras, M. Callejo, M. Diaz, F. Racaru, G. Stea, E. Mingozzi, A. Beben, W. Burakowski, E. Monteiro, and L. Cordeiro, "The EuQoS System: A Solution for QoS Routing in Heterogeneous Networks," *IEEE Communications Magazine*, pp. 96–103, February 2007.

[59] L. Cordeiro, M. Curado, E. Monteiro, V. Bernado, D. Palma, F. Racaru, M. Diaz, and C. Chassot, "GIST Extension for Hybrid On-path Off-path Signaling (HyPath) - draft-cordeiro-nsis-hypath-05," *IETF Internet Draft (work in progress)*, February 2008.

[60] A. Beben, "EQ-BGP: An Efficient Inter-domain QoS Routing Protocol," *Proceedings of 2006 20th International IEEE Conference on Advanced Information Networking and Applications (AINA'06)*, pp. 5–11, April 2006.

[61] R. Good and N. Ventura, "Application Driven Policy Based Resource Management for IP Multimedia Subsystems," *Proceedings of 2009 5th International Conference on Testbeds and Research Infrastructures for the Development of Networks and Communities (TRIDENTCOM'09)*, April 2009.

[62] H. Ekstrom, "QoS Control in the 3GPP Evolved Packet System," *IEEE Communications Magazine*, pp. 76–83, February 2009.

[63] 3GPP, "TS 23.228 IP Multimedia Subsystem(IMS) Stage 2 V8.7.0," December 2008.

[64] Open Mobile Alliance, "XML Document Management (XDM) Specification - candidate version 2.0," September 2008.

[65] J. Rosenberg, "RFC 4825 - The Extensible Markup Language (XML) Configuration Access Protocol (XCAP)," May 2007.

[66] M. Dolly, S. Channabasappa, S. Ganesan, V. Hilt, and D. Petrie, "A Schema and Guidelines for Defining Session Initiation Protocol User Agent Profile Datasets - draft-ietf-sipping-profile-datasets-02.txt," *IETF Internet Draft (work in progress)*, October 2008.

[67] J. Rosenberg, "Identification of Communications Services in the Session Initiation Protocol (SIP) - draft-ietf-sipping-service-identification-03," *IETF Internet Draft (work in progress)*, August 2008.

[68] G. Camarillo, W. Marshall, and J. Rosenberg, "RFC 3312 - Integration of Resource Management and Session Initiation Protocol (SIP)," October 2002.

[69] R. Good and N. Ventura, "End to End Session Based Bearer Control for IP Multimedia Subsystems," *Proceedings of IEEE/IFIP 2009 International Symposium on Integrated Network Management (IM'09)*, to be published in June 2009.

[70] C. Bouras and K. Stamos, "An efficient architecture for Bandwidth Brokers in DiffServ networks," *International Journal of Network Management*, vol. 18(1), pp. 27–46, February 2008.

[71] Business Times South Africa, "MTN, Neotel to build fibre optic network," *http://www.thetimes.co.za/Business/BusinessTimes/Article1.aspx?id=919049*, January 2009.

[72] M. Poikselka, G. Mayer, H. Khartabil, and A. Niemi, "The IMS - IP Multimedia Concepts and Services 2nd Edition," *John Wiley and Sons Ltd.*, August 2006.

[73] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, "RFC 3261 - SIP: Session Initiation Protocol," June 2002.

[74] D. Waiting, R. Good, R. Spiers, and N. Ventura, "Open Source Development Tools for IMS Research," *Proceedings of 2008 4th International Conference on Testbeds*

and Research Infrastructures for the Development of Networks and Communities (TRIDENTCOM'08), March 2008.

[75] G. Koleyni, "Internet Protocol Television - A brief introduction and report on standardisation activities," *ITU-T News*, pp. 28–30, October 2008.

[76] ETSI TISPAN, "TS 183 063 IMS-based IPTV stage 3 specification," November 2008.

[77] P. Weik, D. Vingarzan, and T. Magedanz, "The FOKUS Open IMS Core - A Global Reference Implementation," *Book chapter in IMS Handbook: Concepts, Technologies and Services, Mohamed Ilyas and Syed Ahson*, pp. 113–132, November 2008.

[78] "Ubuntu Linux," *http://www.ubuntu.com/*.

[79] Iptel.org, "The SIP Express Router project," *http://www.iptel.org/ser/*.

[80] "The SIP Router Project," *http://sip-router.org/*.

[81] V. Torvinen, J. Arkko, and M. Naslund, "RFC 4169 - Hypertext Transfer Protocol (HTTP) Digest Authentication Using Authentication and Key Agreement (AKA) Version-2," November 2005.

[82] A. Moizard, "The GNU oSIP library," *http://savannah.gnu.org/projects/osip/*.

[83] A. Moizard, "The GNU eXoSIP library," *http://savannah.gnu.org/projects/exosip/*.

[84] P. Calhoun, J. Loughney, E. Guttman, G. Zorn, and J. Arkko, "RFC 3588 - Diameter Base Protocol," September 2003.

[85] "The libcURL library," *http://curl.haxx.se/*.

[86] "The LibXML2 library," *http://www.xmlsoft.org/*.

[87] R. Marston, "Multimedia Content Adaptation for Internet Protocol Television Services in the IP Multimedia Subsystem," M. Sc. Thesis, University of Cape Town 2009.

[88] R. Good and N. Ventura, "An Evaluation of Transport Layer Policy Control in the IP Multimedia Subsystem," *Proceedings of 2008 19th International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'08)*, September 2008.

[89] "Jakarta Commons HTTPClient package," *http://hc.apache.org/httpclient-3.x/*.

[90] M. Amarascu, "OpenXCAP," *http://www.openxcap.org/*.

[91] 3GPP, "TS 29.228 IP Multimedia (IM) Subsystem Cx and Dx interfaces; Signalling flows and message contents V6.14.0," December 2008.

[92] "Apache Tomcat," *http://tomcat.apache.org/*.

[93] J. Rosenberg and H. Schulzrinne, "RFC 3264 - An Offer/Answer Model with the Session Description Protocol (SDP)," June 2002.

[94] D. Waiting, R. Good, R. Spiers, and N. Ventura, "The UCT IMS Client," *Proceedings of 2009 1st Open NGN IMS Testbeds Workshop (ONIT'09) in conjunction with TRIDENTCOM'09*, April 2009.

[95] D. Waiting, "On Privacy and the Prevention of Unsolicited Sessions in the IP Multimedia Subsystem," Ph.D. Thesis, University of Cape Town, 2008.

[96] "Gstreamer," *http://gstreamer.freedesktop.org/*.

[97] "The libVLC library," *http://wiki.videolan.org/Libvlc*.

[98] L. Mineoro, "The libMSRP library," *http://sourceforge.net/projects/libmsrp/*.

[99] "SIPp," *http://sipp.sourceforge.net/*.

[100] Vodafone Group Research and Development Lab, "Vodafone Mobile Connect Driver for Linux," *https://forge.betavine.net/projects/vodafonemobilec*.

[101] D. Vingarzen and P. Weik, "IMS Signalling over Current Wireless Networks: Experiments using the Open IMS Core," *IEEE Vehicular Technology Magazine*, pp. 28–34, March 2007.

[102] T. Guenkova-Luy, A. Kassler, and D. Mandato, "End-to-End Quality-of-Service Coordination for Mobile Multimedia Applications," *IEEE Journal on Selected Areas in Communications (JSAC)*, vol. 22(5), pp. 889–903, June 2004.

[103] "Wireshark Network Protocol Analyser," *http://www.wireshark.org/*.

Appendix A

Evaluation Framework Hardware Specifications

The evaluation framework described in this thesis comprises entirely Free and Open Source Software (FOSS), effectively opening up the system development to any user of the framework. This means that all elements can be reproduced. This facilitates rapid prototyping and experimentation, where new technologies, protocols and applications can be analysed, separate from the live environment.

The hardware set for each evaluation scenario is described in full, to ensure that the evaluations performed on the testbed can be fully reproduced, and provide a convenient point of departure for any future research in the field. In all scenarios the User Equipment (UE) is hosted on mobile laptop equipment, while the core elements are hosted on personal computer (PC) machines.

A.1 Application Driven Policy Control Framework

The framework for application driven policy control is shown in Fig. A.1. Laptop 1 hosts the UEs, while PC 1 hosts the IMS core and policy control elements. Table A.1 shows the hardware specifications for these machines.

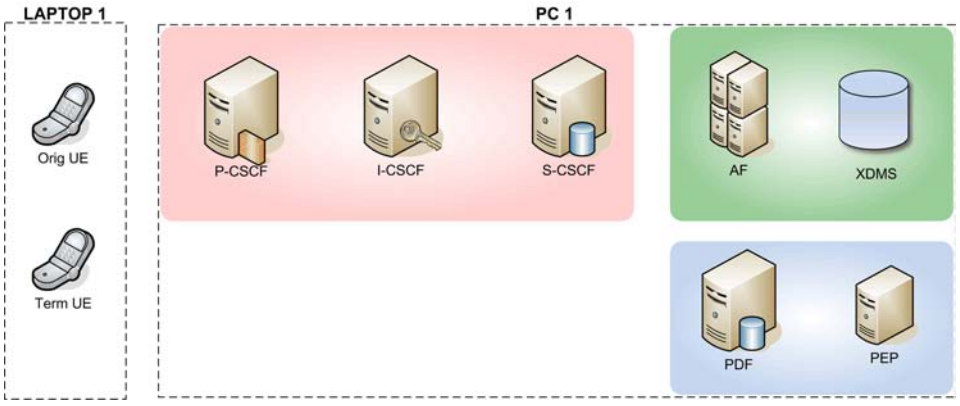

Figure A.1: Application driven policy control scenario.

Table A.1: Hardware specification for application driven policy control framework scenarios.

	Laptop 1	PC 1
Processor	Intel (R) Celeron (R) M Processor	Intel (R) Pentium (R) 4 CPU
CPU (MHz)	1496.685	3191.950
Cache size (kB)	1024	1024
RAM (kB)	5050520	1016500
Swap memory (kB)	1477940	2980016
Operating System (OS)	Ubuntu 8.10 Intrepid	Ubuntu 8.10 Intrepid
OS Kernel	2.6.27-11	2.6.27-11

A.2 Session Based End-to-end Policy Control Framework

The framework for session based end-to-end policy control is shown in Fig. A.2. Laptop 1 hosts the UEs while each IMS domain, including core and policy control elements, is hosted on a separate PC machine. The hardware specifications for each machine are listed in Table A.2.

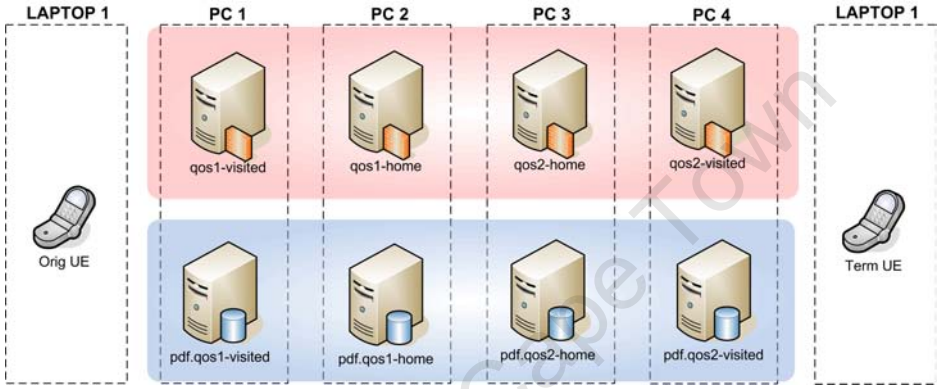

Figure A.2: Session based end-to-end policy control scenario.

Table A.2: Hardware specification for session based end-to-end policy control framework scenarios.

	Laptop 1	PC 1	PC 2	PC 3	PC 4
Processor	Intel (R) Celeron (R) M CPU	Intel (R) Pentium (R) Dual CPU	Intel (R) Pentium (R) 4 CPU	Intel (R) Pentium (R) Dual CPU	Intel (R) Pentium (R) 4 CPU
CPU (MHz)	1496.685	1203.000	3191.950	1203.000	2992.687
Cache size (kB)	1024	1024	1024	1024	512
RAM (kB)	5050520	1025064	1016500	1025064	1016500
Swap mem (kB)	1477940	3004112	2980016	3004112	1646620
Operating System (OS)	Ubuntu 8.10 Intrepid	Ubuntu 8.10 Intrepid	Ubuntu 8.10 Intrepid	Ubuntu 8.10 Intrepid	Ubuntu 8.10 Intrepid
OS Kernel	2.6.27-11	2.6.27-11	2.6.27-11	2.6.27-11	2.6.27-11

A.3 Video on Demand (VoD) Application Invocation

The Video on Demand application invocation scenario is shown in Fig. A.3. Laptop 1 hosts the the UE and each IMS domain is hosted on a separate PC machine. Table A.3 shows the hardware specifications for the machines.

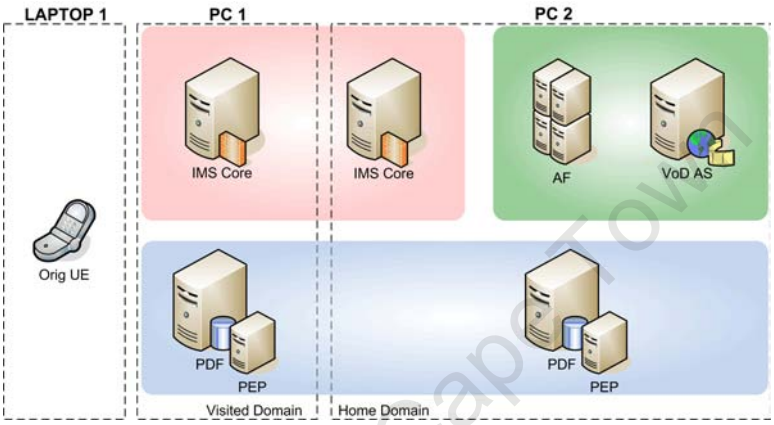

Figure A.3: Video on Demand application invocation scenario.

Table A.3: Hardware specification for VoD application invocation scenarios.

	Laptop 1	PC 1	PC 2
Processor	Intel (R) Celeron (R) M Processor	Intel (R) Pentium (R) 4 CPU	Intel (R) Pentium (R) 4 CPU
CPU (MHz)	1496.685	2992.687	3191.950
Cache size (kB)	1024	512	1024
RAM (kB)	5050520	1016500	1016500
Swap memory (kB)	1477940	1646620	2980016
Operating System (OS)	Ubuntu 8.10 Intrepid	Ubuntu 8.10 Intrepid	Ubuntu 8.10 Intrepid
OS Kernel	2.6.27-11	2.6.27-11	2.6.27-11

Appendix B

802.16d IP-CAN Hardware Specifications

An experimental 802.16d IP-Connectivity Access Network (IP-CAN) is included as part of the evaluation platform. This allows the tests to be carried out over a real IP-CAN and provides insight into how the access technology affects the session setup time and the proportion of overhead that is added by the proposed architecture.

The WiMax network equipment has proprietary management interfaces and does not support the standardised Gx interface. Optimisation of the WiMax hardware for IMS services and the configuration of the equipment from the service or resource control layers are not part of the thesis objectives. The IP-CAN acts merely as a bearer for the signalling from the UE to the fixed network hosting the IMS core and policy control elements.

The hardware specifications, configuration settings and operational conditions are described to ensure that all performed tests are fully reproducible.

The experimental access network is implemented using Alvarion BreezeMax equipment. This hardware is used in numerous commercial deployments and extensively in practical research implementations. The testbed platform at the University of Cape Town (UCT) is illustrated in Fig. B.1 and consists of the following networking elements:

- BreezeMax Micro Base Station (µBST) Indoor Unit (IDU) (Product number: BMAX-MBST-IDU-2CH-AC-3.5).
- BreezeMax Base Station Outdoor Unit (ODU) with connector for separate antennae (Product number: BMAX-BST-AU-ODU-2CH-3.5a1).

- BreezeMax Data Bridge IDU (Product number: BMAX-CPE-IDU-1D).
- BreezeMax CPE PRO ODU with connector for separate antennae (Product number: BMAX-CPE-ODU-PRO-SE-3.5).
- 3 X Agilent 30 decibels (dB) fixed attenuators (Product number: 8495A-001).
- Agilent manual step attenuator 0-70dB (Product number: 8491A-030).

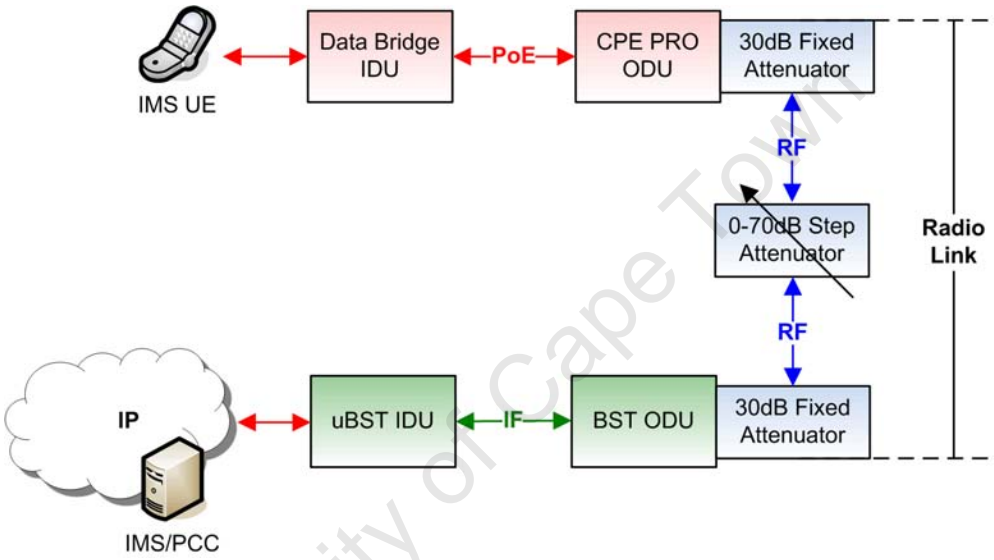

Figure B.1: Experimental 802.16d access network.

The IMS UE connects via LAN to the Data Bridge IDU, which connects through Power over Ethernet (PoE) to the CPE PRO ODU. The equipment operates in the 3.5GHz frequency band. To avoid licensing issues the air interface, between the CPE PRO ODU and the Base Station ODU, is facilitated by Radio Frequency (RF) cables. The 30dB fixed attenuators are attached to the transmitters of the CPE PRO ODU and the Base Station ODU, and the step attenuator is placed between the two units, to ensure optimal impedance. This allows an operator to vary the physical layer conditions of the radio link, on the fly. The Base Station ODU connects to the µBST through a specialised Intermediate Frequency (IF) cable. The µBST provides access to the private IP network over an Ethernet connection, where IMS core and policy control elements are hosted.

Table B.1: Physical layer parameters for the experimental WiMax equipment.

Bandwidth (MHz)	3.5
Uplink Tx Frequency (MHz)	3451.750
Tx power (dBm)	20
Uplink Current Rate	QAM-16 3/4
Uplink SNR (dB)	17.4
Uplink RSSI (dBm)	-85.70
Downlink Current Rate	QAM-16 3/4
Downlink SNR (dB)	18
Downlink RSSI (dBm)	-85

This architecture allows a research institute to implement a real 802.16d IP-CAN with limited resources and without needing access to the frequency spectrum. The resulting platform can be used for a multitude of testing and research purposes.

The parameters that characterise the radio link between the CPE PRO ODU and the Base Station ODU are configured in such a way that conditions are more or less between optimal with zero packet loss, and the worst case where the link can no longer be maintained. These operating conditions are considered typical and most suitable for the necessary evaluations. The relevant physical layer parameters are shown in Table B.1.

Appendix C

Accompanying CD-ROM

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

- Software - Developed source code and software tools used in the development of the evaluation platform including:
  - FOKUS Open Source IMS Core
  - UCT Advanced IPTV (TISPAN IMS-based IPTV stage 3)
  - UCT Policy Control Framework
  - OpenXCAP
  - UCT IMS Client
  - SIPp
- Evaluation results - Raw data collected during performance evaluations.
- Reference material - Collection of publications included in the bibliography of this thesis.
- Published articles - Collection of published papers resulting from this work.
- Thesis documents - Source files for all thesis components and Portable Document Format (PDF) copies of the main thesis document and abstract.