

FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO
DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES



Universidade do Porto
Faculdade de Engenharia
FEUP

**A criação de redes de próxima geração IMS usando
produtos open source**

LUIS FILIPE MAGALHÃES DA COSTA REIS

DISSERTAÇÃO SUBMETIDA PARA SATISFAÇÃO PARCIAL DOS
REQUISITOS DO GRAU DE MESTRE EM
ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES

DISSERTAÇÃO REALIZADA SOB A SUPERVISÃO DO
PROFESSOR DOUTOR EURICO CARRAPATOSO,
DO DEPARTAMENTO DE ENGENHARIA ELECTROTÉCNICA E DE COMPUTADORES
DA FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO PORTO

PORTO, MARÇO DE 2008

Aos meus pais, família e amigos.

Resumo

O IP Multimédia Subsystem (IMS) é uma arquitectura de redes de telecomunicações de próxima geração, baseada em IP, que permite a convergência de terminais fixos e móveis, tipos de rede e aplicações multimédia. Esta normalização procura introduzir uma solução para a integração entre as redes tradicionais de telecomunicações e as tecnologias emergentes como o VoIP e os serviços de internet. Para isso, usa extensivamente o protocolo Session Initiation Protocol (SIP) da IETF e define uma camada de controlo em cima de redes fixas e móveis baseadas em IP, permitindo o desenvolvimento de serviços multimédia independentes do meio de acesso. Esses serviços são fornecidos com recurso a voz, texto, imagens e vídeo procurando ser adaptados ao utilizador e estando disponíveis através de várias tecnologias de acesso como DSL, cabo, WiFi, etc. Em contraste com os serviços comuns de internet, as soluções IMS pretendem aumentar a sua competitividade propondo incrementar a segurança, qualidade de serviço e a flexibilidade da taxação.

O IMS é, actualmente, a arquitectura procurada pelos operadores telecomunicações para fornecimento de serviços e aparenta ser um tema fundamental na estratégia futura dos mesmos assim como aconteceu com as redes inteligentes (IN) na década passada. Por outro lado, o *software open source* prospera dadas as suas vantagens a nível económicos e de fácil “customização”.

Por esta razão, pretendeu-se com este trabalho mostrar a viabilidade de construção de uma rede IMS recorrendo apenas a soluções *open source* existentes no mercado. Desta forma é possível minimizar o impacto dos custos de implementação de uma nova rede de telecomunicações tornando mais atraente a adopção da tecnologia. Neste documento é apresentada uma solução competitiva a nível de custo dado que o *software open source* está isento custos de utilização e há projectos que implementam directamente elementos da rede IMS eliminando custos de desenvolvimento. Infelizmente nem todos os elementos da rede IMS são objecto de projectos *open source*. Mas mesmo nestes casos é possível utilizar o *software open source* como base de trabalho a partir de *stacks* SIP ou Diameter, Um exemplo de um componente IMS que não dispõe de projectos *open source* que o implementem é o Media Resource Function (MRF). Contudo, projectos que disponibilizam *media servers* IP ou IP PBX podem ser alterados para criar um produto IMS *compliant*. Todas as alterações necessárias são documentadas nesta dissertação, contudo, dada a complexidade das redes IMS, este trabalho concentra-se num estudo sobre os principais componentes *core* do IMS apresentando uma solução para uma rede básica mas funcional e respectiva prova de conceito.

Na dissertação em primeiro lugar são descritos conceitos importantes e arquitectura das redes IMS. Em seguida, são analisados pormenorizadamente cada componente da rede e respectivas interligações. Também é apresentado o resultado da pesquisa do “estado da arte” dos componentes IMS *open source*. No final são seleccionados os componentes mais coerentes sendo apresentados na solução com respectiva prova de conceito e são retiradas conclusões sobre a viabilidade da mesma.

Palavras-chave: IMS, IP Multimedia Subsystem, *open source*, CSCF, MRF, optimização de custos, *Media server*, Asterisk, telecomunicações, operadores, *gateways*, SIP, PSTN.

Abstract

IP Multimedia Subsystem (IMS) is an IP based telecommunication network architecture that allows fixed-mobile convergence with multimedia services. This standard introduces a solution for the integration of traditional telecommunication networks and emerging technologies like VoIP and internet services. In order to achieve this it extensively uses IETF's Session Initiation Protocol (SIP) protocol and defines a control layer on top of both IP based fixed and mobile networks allowing the development of multimedia converged services independent of access networks. These services may have voice, text, image and video contents and are available using access technologies such as DSL, cable, Wifi, etc. In contrast with common internet services, IMS solutions allow operators to increase their competitiveness by augmenting security, quality of services and the flexibility of the billing infrastructure.

Nowadays, IMS is the most desired architecture by telecommunication operators for services delivery and looks like it will be essential as a future strategy as has happened with the Intelligent Network (IN) on the last decade.

On the other hand, open source software becomes even more popular given its economical advantages and easy customization. This dissertation will explore this theme and try to show the viability of building an IMS Next-Generation Network (NGN) using widely available open source software, allowing operators to minimize the impact of implementation costs on a new network, making it more attractive.

This document tries to find a cost wise competitive IMS solution using open source components and to achieve that goal several products are compared in order to pick the most effective and reliable solution for each IMS component. The overall solution uses IMS compliant open source components like the OpenIMS core and some other components that are not IMS compliant but with a few modifications may become so. For example, the open source IP PBX called Asterisk PBX that is not IMS compliant may take the place of a Media Resource Function with a few modifications.

This document will also show a few examples of systems based on open source that are commercially available.

Given the IMS complexity, this dissertation focusses on the main IMS components trying to come up with a solution for a basic but functional IMS network and proving it with a proof of concept.

This dissertation is organized in six chapters. Firstly, it describes valuable concepts and the IMS architecture. Then, each IMS component and its interconnections are described. The result for IMS open source state of the art research is also presented. Finally, a solution to the problem is proposed by selecting the most coherent open source components for the IMS network with a proof-of-concept, followed by an advantages and disadvantages analysis.

Keywords: IMS, IP Multimedia Subsystem, open source, CSCF, MRF, cost effectiveness, optimization, Media server, Asterisk, telecommunications, operators, gateways, SIP, PSTN, telco.

Agradecimentos

Gostaria de apresentar os meus agradecimentos ao Prof. Dr. Eurico Carrapatoso por todo o apoio e disponibilidade tanto como orientador desta dissertação como docente no meu percurso académico na Faculdade de Engenharia da Universidade do Porto.

Gostaria de agradecer a todas as pessoas que contribuíram e me apoiaram na elaboração deste trabalho nomeadamente na FEUP e na PT Inovação.

Agradeço, também, à PT Inovação e aos meus colegas pelo excepcional ambiente de trabalho no qual muito tenho aprendido. Foram esses ensinamentos e experiências que potenciaram a elaboração desta dissertação. Um especial agradecimento ao Eng. Joaquim Azevedo, Eng. Nuno Ferreira, Eng. Fernando Delfim, Eng. Sérgio Ramalho, Eng. Luis Almeida, Eng. Nuno Beires e restantes colegas. Um agradecimento ao César Pinto que também está a trabalhar na sua dissertação pelas inúmeras trocas de ideias.

Fica uma palavra de apreço à Faculdade de Engenharia da Universidade do Porto por ser uma instituição que, sem sombra de dúvida, mudou a minha vida tanto pelo que aprendi como pelos amigos que fiz.

Quero agradecer à Prof. Dra. Madalena Dias e ao Prof. Dr. José Carlos Dias pela sua inestimável ajuda e sugestões na elaboração desta dissertação.

Finalmente, gostaria de agradecer aos meus pais, em especial à minha mãe, por todo o apoio, paciência, carinho e mordomias que me tem dado ao longo dos anos. Mas também por ser a força motivadora deste trabalho.

Índice

RESUMO	3
ABSTRACT	4
AGRADECIMENTOS	5
ÍNDICE	6
ÍNDICE FIGURAS.....	8
ÍNDICE TABELAS.....	9
ABREVIATURAS E ACRÓNIMOS	10
1 INTRODUÇÃO.....	16
1.1 MOTIVAÇÃO.....	16
1.2 OBJECTIVO.....	17
1.3 ESTRUTURA DO DOCUMENTO	18
2 A ARQUITECTURA IMS	19
2.1 ARQUITECTURA	19
2.2 OS PRINCIPAIS COMPONENTES.....	23
2.3 GATEWAYS	26
2.4 CHARGING	28
2.5 INTERFACES	29
3 REDES IMS E APLICAÇÕES.....	31
3.1 BENEFÍCIOS.....	31
3.2 SERVIÇOS.....	32
3.3 DESAFIOS	35
4 COMPONENTES OPEN SOURCE.....	36
4.1 CALL SESSION CONTROL FUNCTION	37
4.2 HOME SUBSCRIBER SERVER.....	38
4.3 MEDIA RESOURCE FUNCTION	38
4.4 APPLICATION SERVERS	42
4.5 USER EQUIPMENT	43
4.6 GATEWAYS	46
4.7 FERRAMENTAS DE TESTE.....	47
4.8 STACKS DIAMETER	47
4.9 STACKS SIP.....	48
5 SOLUÇÃO PROPOSTA	50

5.1	ARQUITECTURA DA SOLUÇÃO	50
5.2	CALL SESSION CONTROL FUNCTION	51
5.3	HOME SUBSCRIBER SERVER.....	52
5.4	MEDIA RESOURCE FUNCTION	52
5.5	SIP APPLICATION SERVER	61
5.6	USER EQUIPMENT	62
5.7	GATEWAYS	63
5.8	PROVA DE CONCEITO.....	67
6	CONCLUSÕES.....	70
6.1	ANÁLISE.....	70
6.2	AVALIAÇÃO	71
6.3	CONCLUSÕES	72
6.4	LIÇÕES APRENDIDAS	74
6.5	TRABALHO FUTURO	75
7	REFERÊNCIAS E BIBLIOGRAFIA	76
7.1	REFERÊNCIAS BIBLIOGRÁFICAS.....	76
7.2	REFERÊNCIAS WEB	78
7.3	NORMAS 3GPP	81
7.4	NORMAS IETF	84

Índice Figuras

Figura 1 – Arquitectura da rede IN.....	19
Figura 2 – A arquitectura lógica IMS.....	20
Figura 3 – Arquitectura IMS (Release 6)	21
Figura 4 – Arquitectura do MRF.....	25
Figura 5 – Arquitectura funcional para os serviços IMS.....	26
Figura 6 – Configuração de uma signalling gateway	28
Figura 7 – Arquitectura de Charging IMS.....	28
Figura 8 – Serviços IMS.....	33
Figura 9 – Exemplo do OpenIC numa rede IMS.....	44
Figura 10 – Solução proposta.....	50
Figura 11 – Solução proposta (Release 6).....	51
Figura 12 – Arquitectura OpenIMS Core.....	51
Figura 13 – OpenIMS core numa rede IMS	52
Figura 14 – Arquitectura MRF da solução.....	53
Figura 15 – Arquitectura MRF Asterisk.....	55
Figura 16 – Arquitectura Mobicents.....	61
Figura 17 – Interface gráfica X-Lite.....	62
Figura 18 – Interface gráfica do IMS Communicator	62
Figura 19 – Gateways IMS.....	63
Figura 20 – Interface Web Asterisk GUI.....	66
Figura 21 – Arquitectura da prova de conceito	67
Figura 22 – Configuração Web do HSS.....	68
Figura 23 – Exemplos de configurações do mosaico do MCU	68
Figura 24 – Screenshot da página de configuração mcuWeb.....	69
Figura 25 – X-Lite User Equipment.....	69

Índice Tabelas

Tabela 1 – Interfaces IMS	30
Tabela 2 – Estatísticas de utilização Asterisk PBX.....	39

Abreviaturas e Acrónimos

3G	<i>Third generation wireless technology</i>
3G-324M	<i>Protocol for conversational video telephony</i>
3GPP	<i>Third Generation Partnership Project</i>
AAA	<i>Authentication, Authorization e Accounting</i>
ABG	<i>Access Border Gateway</i>
AIN	<i>Advanced Intelligent Network</i>
AKA	<i>Authentication and Key Agreement</i>
ALG	<i>Application Level Gateway</i>
AMR	<i>Adaptive Multi-Rate</i>
AN	<i>Access Node</i>
API	<i>Application Programming Interface</i>
AS	<i>Application Server</i>
ATCA	<i>Advanced Telecom Computing Architecture</i>
AuC	<i>Authentication Center</i>
B2BUA	<i>Back-to-Back User Agent</i>
BB	<i>Broadband Access</i>
BD	<i>Billing Domain</i>
BG	<i>Border Gateway</i>
BGCF	<i>Breakout Gateway Control Function</i>
BS	<i>Billing System</i>
CAMEL	<i>Customized Applications for Mobile networks Enhanced Logic</i>
CAP	<i>CAMEL Application Part</i>
CAPI	<i>Common ISDN Application Programming Interface</i>
CCF	<i>Charging Collector Function</i>
CCXML	<i>Call Control eXtensible Markup Language</i>
CDR	<i>Call Detail Records</i>
COTS	<i>Commercial Off-The-Shelf</i>
CPU	<i>Central Processing Unit</i>
CRM	<i>Customer Relationship Management</i>
CS	<i>Circuit-Switched</i>
CSCF	<i>Call Session Control Function</i>

CTF	<i>Charging Trigger Function</i>
DB	<i>Database</i>
DSL	<i>Digital Subscriber Line</i>
DSLAM	<i>Digital Subscriber Line Access Multiplexer</i>
DTMF	<i>Dual-Tone Multi-Frequency</i>
DVR	<i>Digital Video Recorder</i>
EAP	<i>Extensible Authentication Protocol</i>
ECF	<i>Event Charging Function</i>
ECUR	<i>Event Charging with Unit Reservation</i>
FastAGI	<i>Fast Asterisk Gateway Interface</i>
FMC	<i>Fixed-Mobile Convergence</i>
GGSN	<i>Gateway GPRS Support Node</i>
GNU	<i>GNU's Not Unix</i>
GPL	<i>General Public License</i>
GPRS	<i>General Packet Radio System</i>
GSM	<i>Global System for Mobile communication</i>
H.323	<i>ITU standard for videoconferencing over local area networks and packet-switched networks</i>
HA	<i>High Availability</i>
HLR	<i>Home Location Register</i>
HSDPA	<i>High Speed Downlink Packet Access</i>
HSS	<i>Home Subscriber Server</i>
HSUPA	<i>High Speed Uplink Packet Access</i>
HTTP	<i>HyperText Transfer Protocol</i>
IAX	<i>Inter Asterisk Exchange</i>
IBCF	<i>Interconnection Border Control Function</i>
I-CSCF	<i>Interrogating CSCF</i>
IEC	<i>Immediate Event Charging</i>
IETF	<i>Internet Engineering Task Force</i>
IM	<i>Instant Messaging</i>
IMPI	<i>Multimedia Private Identity</i>
IMPU	<i>IP Multimedia Public Identity</i>
IMS	<i>IP Multimedia Subsystem</i>
IMS IPGW	<i>IMS IP Gateway</i>

IMSI	<i>International Mobile Subscriber Identity</i>
IM-SSF	<i>IP Multimedia Services Switching Functions</i>
IN	<i>Intelligent Network</i>
IP	<i>Internet Protocol</i>
IP CAN	<i>IP Connectivity Access Network</i>
IPSec	<i>IP Security</i>
IPTV	<i>Television over Internet Protocol</i>
IPv4	<i>Internet Protocol version 4</i>
IPv6	<i>Internet Protocol version 6</i>
ISC	<i>IMS Service Control</i>
ISDN	<i>Integrated Services Digital Network</i>
ITU	<i>International Telecommunication Union</i>
ITU-T	<i>ITU Telecommunication Standardization Sector</i>
IVR	<i>Interactive voice response</i>
IVVR	<i>Interactive Voice and Video Response (IV²R)</i>
J2EE	<i>Java 2 Platform, Enterprise Edition</i>
JAIN	<i>Java Advanced Intelligent Network</i>
JMF	<i>Java Media Framework</i>
JMX	<i>Java Management eXtensions</i>
JSLEE	<i>Java Service Logic Execution Environment</i>
LAN	<i>Local Area Network</i>
LBS	<i>Location Based Services</i>
LGPL	<i>Lesser General Public License</i>
MAP	<i>Mobile Application Part</i>
MCU	<i>Multipoint Conference Unit</i>
MD5	<i>Message Digest 5</i>
MEGACO	<i>Media Gateway Control Protocol (H.248)</i>
MGW	<i>Media Gateway</i>
MMS	<i>Multimedia Message Service</i>
MOML	<i>Media Object Markup Language</i>
MPEG	<i>Moving Picture Experts Group</i>
MRF	<i>Media Resource Function</i>
MRFC	<i>Multimedia Resource Function Controller</i>
MRFP	<i>Media Resource Function Processor</i>

MS	<i>Media Server</i>
MSCML	<i>Media Server Control Markup Language</i>
MSISDN	<i>Mobile Subscriber ISDN Number</i>
MSML	<i>Media Sessions Markup Language</i>
MTP	<i>Message Transfer Part</i>
NAT	<i>Network Address Translation</i>
NEP	<i>Network Equipment Provider</i>
NGN	<i>Next Generation Network</i>
OAM	<i>Operations, Administration and Maintenance</i>
OCAF	<i>Open Communications Architecture Forum</i>
OCF	<i>Online Charging Function</i>
OEM	<i>Original Equipment Manufacturer</i>
OSA	<i>Open Service Architecture</i>
OSA SCS	<i>OSA Service Capability Server</i>
OSA-GW	<i>Open Service Access Gateways</i>
PBX	<i>Public Branch Exchange</i>
PC	<i>Personal Computer</i>
PCM	<i>Pulse-Code Modulation</i>
P-CSCF	<i>Proxy CSCF</i>
PDA	<i>Personal Digital Assistant</i>
PDF	<i>Policy Decision Function</i>
PDG	<i>Packet Data Gateway</i>
PDN	<i>Packet Data Network</i>
PEF	<i>Policy Enforcement Function</i>
PLMN	<i>Public Land Mobile Network</i>
PS	<i>Packet-Switched</i>
PSTN	<i>Public Switched Telephone Network</i>
PTT	<i>Push-to-Talk</i>
QoS	<i>Quality of service</i>
RA	<i>Resource Adapter</i>
RADIUS	<i>Remote Authentication Dial In User Service</i>
RAID	<i>Redundant Arrays of Independent Disks</i>
RAN	<i>Radio Access Network</i>
RDIS	<i>Rede Digital Integrada de Serviços (ISDN)</i>

RTP	<i>Real-time Transport Protocol</i>
SAF	<i>Service Availability Forum</i>
SBB	<i>Service Building Blocks</i>
SBC	<i>Session Border Controller</i>
SCCP	<i>Signaling Connection Control Part</i>
SCF	<i>Session Charging Function</i>
SCIM	<i>Service Capability Interaction Management</i>
SCTP	<i>Stream Control Transmission Protocol</i>
SE	<i>Standard Edition</i>
SER	<i>SIP Express Router</i>
SGSN	<i>Service GPRS Support Node</i>
SGW	<i>Signaling Gateway</i>
SIP	<i>Session Initiation Protocol</i>
SIP UA	<i>Session Initiation Protocol User Agent</i>
SLEE	<i>Service Logic Execution Environment</i>
SLF	<i>Subscriber Location Function</i>
SMS	<i>Short Message Service</i>
SNMP	<i>Simple Network Management Protocol</i>
SOA	<i>Service-Oriented Architecture</i>
SOAP	<i>Simple Object Access Protocol</i>
SS7	<i>Signaling System 7</i>
S-SCSF	<i>Serving CSCF</i>
TAS	<i>Telephony Application Servers</i>
TCO	<i>Total Cost of Ownership</i>
TDM	<i>Time Division Multiplexing</i>
THIG	<i>Topology Hiding Inter-network Gateway</i>
TISPAN	<i>Telecoms & Internet converged Services & Protocols for Advanced Networks</i>
TLS	<i>Transport Layer Security</i>
TPF	<i>Traffic Plain Function</i>
TS	<i>Time-Switched</i>
UA	<i>User Agent</i>
UAC	<i>User Agent Client</i>
UAS	<i>User Agent Server</i>

UE	<i>User Equipment</i>
UM	<i>Unified Messaging</i>
UMTS	<i>Universal Mobile Telecommunications System</i>
VoiceXML	<i>Voice eXtensible Markup Language</i>
VoIP	<i>Voice over IP</i>
WiFi	<i>Wireless Fidelity</i>
WiMax	<i>Worldwide Interoperability for Microwave Access</i>
WWW	<i>World Wide Web</i>
XCAP	<i>XML Configuration Access Protocol</i>
XML	<i>eXtensible Markup Language</i>
XMPP	<i>eXtensible Messaging and Presence Protocol</i>

1 Introdução

O IP Multimedia Subsystem (IMS) é uma arquitectura para redes de telecomunicações da próxima geração. Esta arquitectura é baseada em IP e tem o intuito de permitir a convergência de terminais fixos e móveis, vários tipos de rede e de aplicações multimédia. As redes IMS procuram introduzir uma solução para a integração entre as redes de telecomunicações inteligentes (rede IN) tradicionais e as tecnologias emergentes como o VoIP e os serviços de Internet. Para isso, o IMS procura usar o melhor dos dois mundos aproveitando a flexibilidade do protocolo Session Initiation Protocol (SIP) da Internet Engineering Task Force (IETF) [IETF 07] e define uma camada de controlo em cima de redes fixas e móveis baseadas em IP, permitindo o desenvolvimento de serviços multimédia independentes do meio de acesso. Os serviços IMS fornecidos têm acesso a recursos multimodais com voz, texto, imagens e vídeo, procurando ser adaptados ao utilizador e estando disponíveis através de várias tecnologias de acesso como o Digital Subscriber Line (DSL), cabo, WiFi, etc. Em contraste com os serviços comuns de internet, as soluções IMS pretendem aumentar a sua competitividade propondo aumentar a segurança, a qualidade de serviço e a flexibilidade da infra-estrutura de taxaço.

O conceito de rede IMS ou IP Multimedia Subsystem foi criado pelo Third Generation Partnership Project (3GPP) [3GPP 07] para evoluir e sobretudo substituir as actuais redes inteligentes (IN) por redes convergentes que unissem o mundo IP onde assenta a internet com as comunicações fixas e móveis actuais. O conceito nasceu inicialmente com a 3GPP *Release 5*. A *Release 6* introduziu o IMS sobre WLAN e melhorias na interligação com redes não IMS. Contudo, a *Release 7* que estava previsto ser fechada em 2007 é considerada por muitos como a primeira especificação estável e completa. Apesar da R7 não estar ainda fechada, já foram lançados requisitos e alguns documentos relativos à *Release 8*.

Paralelamente à especificação IMS, após o lançamento da *Release 6*, o grupo Telecoms & Internet converged Services & Protocols for Advanced Networks (TISPAN) [TISPAN 07] lançou uma arquitectura de telecomunicações baseada no IMS e denominada Next-Generation Network (NGN) R1 que tenta expandir o IMS para o uso em redes fixas da próxima geração. Como essa iniciativa iria criar duas normalizações distintas e eventualmente, foi decidido fundir as normas criadas pela 3GPP e TISPAN na *Release 7* IMS.

1.1 Motivação

O *software open source* tem sido usado por muitas empresas como uma base sólida para desenvolvimento de produtos tanto na área das telecomunicações como noutras. A Portugal Telecom Inovação [PT Inovação 07] é um exemplo de uma empresa multinacional que procura relacionar-se com o universo *open source* oferecendo à comunidade *software* ou contribuições que são seu desenvolvimento. Por outro lado, usa *software open source* na base de alguns produtos comerciais. O IMS Communicator [IMS Com 07] é um exemplo de *software* que nasceu na PT Inovação [PT Inovação 07] e foi oferecido à comunidade *open source*. Por outro lado, sistemas como o DiNO, uma plataforma de gestão e distribuição de conteúdos para terminais móveis, são suportados a 100% por *software open source*. Esta plataforma é um caso de sucesso, tendo actualmente mais de 50 milhões de utilizadores

distribuídos por empresas do grupo PT com a Vivo, TMN, Uzo, CVT Móvel, entre outros [Grupo PT 07].

Por outro lado, a indústria do *Software* mostra que, por vezes, para se ter um domínio sobre determinada tecnologia, tem que se tornar aberta e sem custos. Veja-se o exemplo da SUN Microsystems que ofereceu a sua linguagem de programação Java ao mundo sem custos. Em pouco tempo, o Java tornou-se numa das linguagens mais utilizadas pelos programadores. Outro exemplo muito significativo é o da Digium® [Digium 07]. Esta empresa de *hardware* decidiu lançar um *software open source* que potenciase a utilização das suas placas telefónicas denominado Asterisk® PBX [Asterisk 07]. Actualmente, é mais fácil associar a Digium® ao Asterisk® do que a placas telefónicas.

Ao observar esta conjuntura que propicia uma redução de custos e uma abertura por parte dos operadores ao *software open source* surge uma possível resolução para um problema antigo dos operadores de telecomunicações e que se tornou evidente recentemente com o lançamento da terceira geração de telemóveis que é como reduzir o custo de substituição ou actualização de uma rede de telecomunicações por uma arquitectura mais actual e avançada.

1.2 Objectivo

Neste documento pretende-se apresentar uma solução para um problema que tem vindo a limitar a implantação das redes de próxima geração: os custos. Um exemplo recente deste problema foi a introdução das redes Universal Mobile Telecommunications System (UMTS) para a terceira geração de telemóveis. Os valores elevadíssimos das licenças combinados com os investimentos nas estruturas e equipamentos de rede criaram sérias dificuldades aos operadores móveis. Esta situação foi agravada pela conjuntura económica criando situações de fragilidade económica dos operadores.

Para solucionar esse problema, propõe-se uma metodologia que tem vindo a ganhar um ímpeto e uma popularidade assinaláveis no mercado: o *open source*. Assim sendo, pretende-se demonstrar que com *software* livre e sem custos de *download* ou utilização, se pode construir uma rede de próxima geração IMS e combater a resistência associada a qualquer investimento avultado por parte dos operadores de telecomunicações. Contudo, a solução proposta será mais abrangente que os operadores, ganhando relevância para os fabricantes de equipamentos para testes de interoperabilidade ou instituições de ensino para formação dos seus alunos, dado os seus custos reduzidos e facilidade de implementação.

A solução a propor deverá ser constituída fundamentalmente por componentes na área do *software open source* com licenças GNU General Public License (GPL) ou Lesser General Public License (LGPL) [GNU Lics 07] e, dada a complexidade da arquitectura, abranger todos os elementos principais do *core* de uma rede IMS. Os equipamentos de acesso para Wireless LAN (WLAN) ou Digital Subscriber Line Access Multiplexer (DSLAM) para redes de banda larga saem do âmbito deste trabalho. Contudo, as Gateways com a PSTN deverão ser referenciadas já que actualmente há muitos IP PBX *open source* que implementam esses sistemas.

1.3 Estrutura do documento

O documento está organizado em seis capítulos. No âmbito geral, pretende-se fazer uma introdução aos conceitos, arquitecturas, estado da arte da tecnologia e apresentar uma solução para o problema com a respectiva prova de conceito. Assim, o primeiro capítulo (1) faz uma breve introdução ao conceito, motivação e objectivos deste trabalho.

No segundo capítulo (2) faz-se uma introdução as redes IMS num contexto histórico e descreve-se a arquitectura da rede IMS nas suas diferentes camadas. Em seguida, são descritos pormenorizadamente os componentes do *core* IMS indicando as suas funções e interligações com outros elementos. Também são apresentadas as interfaces existentes em cada componente e é introduzido o conceito de *charging*.

O capítulo 3 apresenta os benefícios que a adopção da arquitectura IMS pode trazer para os fornecedores de serviços, operadores e clientes finais. Também enumera alguns dos principais serviços IMS e termina com uma análise dos desafios que as redes IMS enfrentam.

No capítulo 4 é apresentado o estudo do estado da arte dos projectos *open source* que apresentam soluções que poderão ser seleccionadas para integrar a rede IMS. No caso de não existir nenhuma solução que respeite na totalidade as normas IMS, são estudados componentes que possam ser modificados para integrar a rede e, eventualmente, se não for possível encontrar uma solução viável são sugeridos sistemas comerciais que possam colmatar a falha.

O capítulo 5 apresenta a solução proposta. São apresentados os componentes seleccionados e as alterações necessárias para compatibilizar determinados projectos com as normas IMS. Em seguida, é apresentada uma prova de conceito recorrendo a um serviço de videoconferência para demonstrar a viabilidade da solução proposta.

No capítulo 6 é analisada a solução e feito um balanço das vantagens e desvantagens da abordagem, sendo retiradas conclusões sobre a evolução das redes, viabilidade da solução proposta e trabalho futuro.

2 A arquitectura IMS

Este capítulo faz uma introdução à arquitectura de rede IMS e aos seus principais elementos. Mas antes de apresentar a arquitectura de rede IMS é necessário entender as actuais redes de telecomunicações, nomeadamente a rede IN, para compreender como poderão as redes IMS eventualmente vir a substituí-las.

A rede inteligente (Intelligent Network), também conhecida simplesmente pelo acrónimo IN, é uma arquitectura de rede usada actualmente tanto nas redes fixas como nas redes móveis baseada no protocolo de sinalização nº7 (SS7).

O conceito, arquitectura e protocolos da IN foram desenvolvidos inicialmente pela International Telecommunication Union Standardization Sector (ITU-T) e o seu objectivo principal era o de melhorar as capacidades dos serviços oferecidos pela rede dado que anteriormente à IN, todos os serviços eram implementados directamente sobre os comutadores. Com o aparecimento de serviços como números verde, azul ou portabilidade houve a necessidade de criar uma arquitectura mais simples para operação e de disseminação dos mesmos. Assim, esses serviços foram movidos para nós específicos criando uma rede mais modular e aumentando a facilidade de implementação dos serviços. Desde então, serviços complexos como o pré-pago, usam a IN para serem implementados.

A Figura 1 mostra a arquitectura de uma rede IN [IEC 07].

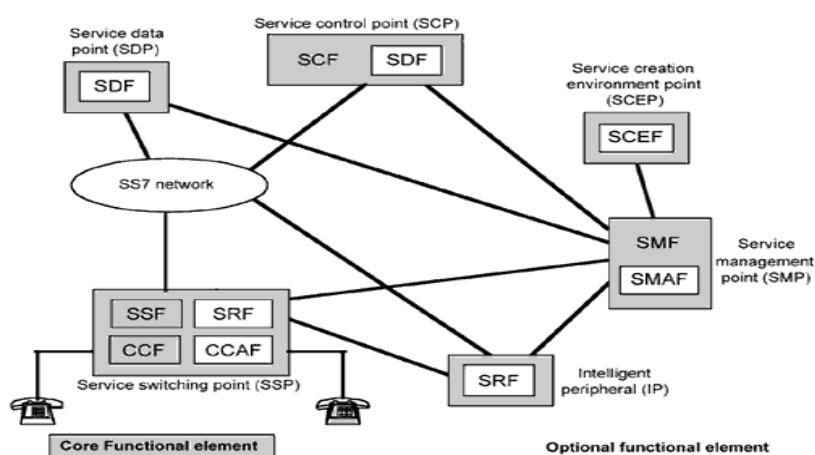


Figura 1 – Arquitectura da rede IN

2.1 Arquitectura

Nos anos recentes e excluindo os sistemas DSL, a evolução tecnológica tem ocorrido sobretudo nas redes móveis. O aumento de capacidades dos terminais móveis tem permitido aos fornecedores de serviços disponibilizar novos serviços de áudio, vídeo e dados. Contudo, a infra-estrutura não consegue acompanhar as necessidades dos novos serviços. Assim, foi criado o projecto Third Generation Partnership Project (3GPP) [3GPP 07] e 3GPP2 para garantir que todos os operadores iriam seguir a mesma abordagem, os mesmos protocolos e interagir entre eles da mesma forma para garantir a melhor desempenho móvel.

Estes projectos começaram a debruçar-se sobre problemas da rede móvel e como poderiam ser oferecidos serviços fixos e serviços internet consolidados numa oferta única. Para atingir este objectivo, dois grupos distintos juntaram forças: o supracitado 3GPP, representando os operadores móveis, e a Internet Engineering Task Force (IETF) [IETF 07], representando os operadores fixos. E assim criaram um comité para encontrar uma solução donde nasceu o IMS!

O IMS é a primeira arquitectura de telecomunicações especificada pela própria indústria. Oferece suporte para os maiores fornecedores de equipamentos de rede Network Equipment Providers (NEPs) e fornecedores de serviços. O IMS unifica aplicações usando o Session Initiation Protocol (SIP) para ligar aos serviços telefónicos tradicionais e não telefónicos como instant messaging (IM), *push-to-talk*, *video streaming*, *multimedia messaging* e Multimedia Messaging Service (MMS).

A arquitectura apresenta-se como a solução para a convergência entre redes fixas e redes móveis — terminais, redes e serviços. Foi desenhada para permitir a migração gradual da actual infra-estrutura de telecomunicações para uma nova infra-estrutura baseada em IP que permitirá o lançamento de novos serviços reduzindo a complexidade e os custos. Estas reduções trarão benefícios tanto para os fornecedores de serviços como para os subscritores dos mesmos. Desta forma, os fornecedores de serviços têm a oportunidade de apresentar novos e melhores serviços sobre redes *wireless*, *wireline* e *broadband*.

Actualmente, a versão das normas IMS oficialmente fechada é a *Release 6*; contudo, dadas as suas limitações, os fornecedores de equipamentos IMS estão a concentrar os seus desenvolvimentos em torno da *Release 7* que apesar de ainda estar em discussão apresenta uma solução mais satisfatória e completa. Assim sendo, a elaboração deste trabalho concentrar-se-á na arquitectura definida na *Release 7* das normas IMS. Essa arquitectura está dividida em 3 camadas distintas: Transport and Endpoint Layer, Session and Control Layer e Application and Services Layer.

A Figura 2 [NMS 07] mostra a arquitectura lógica de uma rede IMS segundo a *Release 7*. Nela é possível analisar a disposição das camadas supracitadas e dos elementos IMS que as constituem. (Os acrónimos existentes na figura podem ser consultados na secção Abreviaturas e Acrónimos).

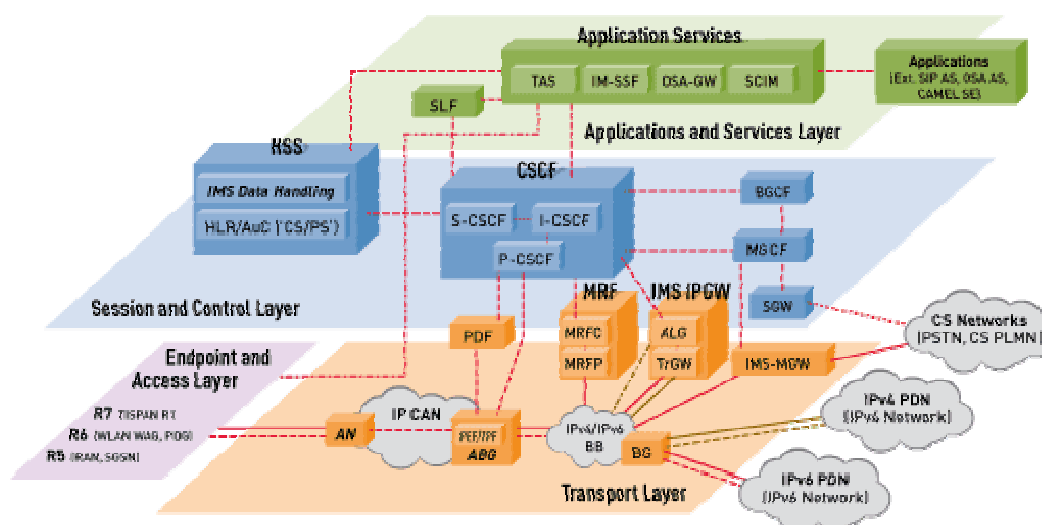


Figura 2 – A arquitectura lógica IMS

A forma horizontal como as camadas estão dispostas permite criar uma abstracção entre elas e as tecnologias usadas. Desta forma, por exemplo, as tecnologias de controlo de sessão são agnósticas ao tipo de rede de acesso utilizado. Também são disponibilizadas funções comuns que podem ser reutilizadas por múltiplas aplicações facilitando a utilização da rede.

Apenas a título de exemplo, a Figura 3 [UPF 07] mostra a arquitectura IMS definida na versão *Release 6* das normas IMS. Como é possível observar, os principais elementos do *core* da rede IMS mantêm-se inalterados mas as camadas têm nomes ligeiramente diferentes e o HSS foi transferido da camada de aplicação para a camada de sessão e controlo (o HSS é uma entidade semelhante ao HLR existente nas redes GSM). A nível de implementação, esta transição do HSS não trouxe diferenças nas interfaces ou formas de comunicação, sendo apenas uma alteração formal.

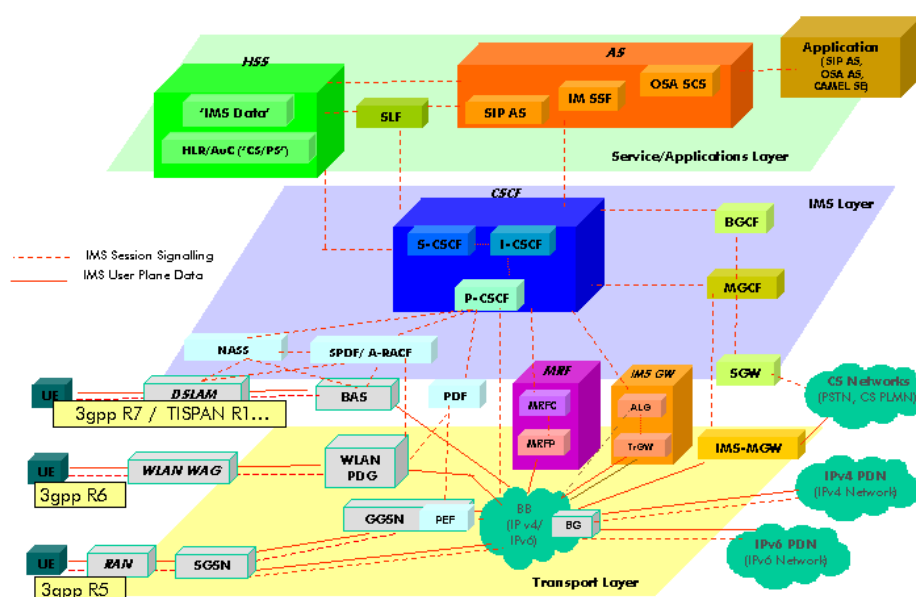


Figura 3 – Arquitectura IMS (Release 6)

Relativamente à Figura 2, a primeira camada (Transport and Endpoint Layer) traduz os canais de *media* e sinalização das redes tradicionais em *streams* de comutação de pacotes. Transforma toda a sinalização e *media* proveniente das redes analógicas, digitais e banda larga em protocolos SIP e Real-time Transport Protocol (RTP). Isto é conseguido através de *media* e *signalling gateways*. Também inclui *media servers* (os Media Resource Functions) com recursos avançados de multimédia para tocar anúncios, sinalização *in-band* e conferência.

Estes *media servers* são partilhados por todas as aplicações (*voicemail*, Interactive Voice and Video Response (IVVR) e *push-to-talk*) maximizando a taxa de utilização dos equipamentos e flexibilizando a utilização dos recursos.

A segunda camada (Session and Control Layer) fornece as funções elementares de *media* para as aplicações de alto nível e controla as ligações lógicas entre vários elementos de rede. Fornece o registo de *endpoints*, faz o encaminhamento de mensagens SIP e coordena os recursos multimédia e de sinalização. Esta camada contém dois dos elementos de rede mais

significativos - o Call Session Control Function (CSCF), também conhecido como *softswitch*, e o Home Subscriber Server (HSS). O HSS detem o perfil de serviço de cada utilizador incluído informação de registo, preferências, *roaming*, opções de *voicemail* e lista de contactos. A centralização desta informação neste componente permite o fácil aprovisionamento de serviços, acesso a aplicações e partilha do perfil entre múltiplas redes de acesso. Por exemplo, o HSS permite que as mesmas configurações de *voicemail* sejam aplicadas tanto à rede móvel como à rede fixa e totalmente integrado com soluções de Unified Messaging (UM). O CSCF garante qualidade de serviço (QoS), implementa segurança e suporta taxação (*billing*) por sessão.

O IMS usa serviços de alto nível e APIs para permitir a terceiros o acesso a sessões e preferências dos subscritores.

A Terceira Camada (Application Services Layer) contém múltiplos servidores de aplicação Application Servers (AS) como o Telephony Application Servers (TAS), IP Multimedia Services Switching Functions (IM-SSF), Open Service Access Gateways (OSA-GW), etc. Cada um destes servidores é responsável por executar operações sobre sessões subscritas, mantendo o controlo da chamada. Também cria uma ligação entre o legado de serviços Advanced Intelligent Network (AIN) e o mundo IMS. Por exemplo, o IM-SSF faz a ponte entre SIP e aplicações móveis Customized Applications for Mobile Network Enhanced Logic (CAMEL) para permitir a portabilidade de número ou serviços 800. Os diferentes tipos de *application servers* serão descritos mais pormenorizadamente no capítulo 4.4.

Ao usar *application servers*, os operadores podem dar acesso ao seu ambiente de serviços a fornecedores externos desenvolvendo parcerias para criação de novas aplicações fora do seu domínio de conhecimento. O que permite manter o controlo sobre os seus próprios serviços e sobre a integridade da rede mas diversificar o portfolio de serviços disponíveis, partilhando a sua base de clientes e infra-estrutura com terceiros. Esta facilidade de criação por entidades externas combinada com a existência de normas específicas para desenvolvimento de serviços é uma das vantagens das redes IMS. Além disso, os fornecedores de serviços, na criação novas aplicações de sucesso, têm à sua disposição a:

- A inteligência dos equipamentos SIP IMS para auto negociar recursos e capacidades com outros sistemas;
- A estrutura horizontal das plataformas IMS para abstrair as redes de acesso e os serviços;
- Uma rede de serviços IMS onde residem as lógicas de serviços e as funcionalidades das aplicações;

As vantagens do IMS serão realmente sentidas quando os pontos supracitados, incluindo os equipamentos (p. ex: telemóveis, PCs, PDAs, etc.), estiverem harmonizados e a trabalhar em conjunto

Serviços como o *push-to-talk*, conferência multimédia com colaboração e outras combinações de serviços necessitam negociação directa entre serviços IMS e equipamentos IMS. Estas associações ponto a ponto são possíveis através de *brokers* existentes na camada Application Services Layer denominados Service Capability Interaction Manager (SCIM). O SCIM explora o poder do SIP para ligar os extremos – um serviço IMS com um determinado equipamento IMS.

2.2 Os principais componentes

Esta secção descreve os principais elementos *core* de uma rede IMS e as interligações entre eles [Leig 06][Magedanz 06].

As interligações usam sobretudo os protocolos SIP, HTTP e Diameter. O SIP é um protocolo de sinalização, o HTTP é um protocolo de transferência de informação e o Diameter é um protocolo de rede para autenticação, autorização e *accounting* (AAA). O Diameter é conhecido como o sucessor do protocolo RADIUS descritos no RFC 2138, 2865 e 2866.

Call Session Control Function

O Call Session Control Function (CSCF) consiste num conjunto específico de sub-elementos que gerem a sinalização associada com o estabelecimento, desligamento e troca de mensagens SIP das chamadas.

O CSCF é constituído por 3 sub-elementos:

- I-CSCF – Interrogating Call Session Control Function
- P-CSCF – Proxy Call Session Control Function
- S-CSCF – Serving Call Session Control Function

Estes elementos actuam como servidores ou *proxies* SIP e são o núcleo de processamento dos pacotes de sinalização SIP numa rede IMS.

O Interrogating-CSCF (I-CSCF) tem um endereço IP público e é através dele que servidores remotos contactam um determinado domínio. O I-CSCF interroga o HSS usando a interface Diameter Cx para conhecer a localização de um utilizador e encaminha os pedidos SIP para o S-CSCF que serve esse cliente. Até a *Release 6* da normalização IMS, podia ser usado como o ponto de entrada escondendo a rede do domínio público tornando-se num Topology Hiding Inter-network Gateway (THIG). A partir da *Release 7*, este papel é desempenhado pelo Interconnection Border Control Function (IBCF).

O Proxy-CSCF (P-CSCF), como o nome indica, é um proxy SIP que funciona como o primeiro e exclusivo ponto de contacto de um terminal IMS (User Equipment). Por ele passam todas as mensagens SIP enviadas ou recebidas pelo terminal. Algumas redes usam um SBC – Session Border Control para desempenhar esta função. Este elemento é indicado ao terminal IMS quando este se regista e não muda durante o período do registo. É, também, o responsável por gerar os registos de utilização para taxação.

O P-CSCF pode ser responsável pela implementação de QoS, gestão de largura de banda e controlo de acessos se tiver incluído o elemento Policy Decision Function (PDF). Apesar de estar logicamente separado do P-CSCF, é usual encontrar o PDF embutido no CSCF.

O P-CSCF desempenha, também, um papel na segurança da rede sendo o responsável pela autenticação e estabelecimento de um protocolo IPSEC com o terminal. Previne ataques através da alteração dos pacotes SIP e protege a privacidade do utilizador. Os outros componentes da rede estabelecem uma relação de confiança com o P-CSCF e não autenticam o utilizador novamente.

Em suma, o P-CSCF executa as seguintes tarefas:

- É ponto de contacto com o UE; faz as primeiras verificações de segurança do cliente através dos dados existentes no HSS;
- Estabelece o protocolo IPSEC para criar um domínio seguro (*trusted*);
- Gere dados de *billing*;
- Implementa QoS;

O Serving-CSCF (S-CSCF) é o centro de encaminhamento do sistema, sendo o responsável pelos registos SIP que fazem a interligação entre o terminal e o respectivo endereço SIP. Por ele passam todas as mensagens de sinalização SIP e é o responsável por determinar para que Application Server (AS) deve encaminhar as mensagens SIP de modo a fornecer o serviço desejado.

O S-CSCF usa as interfaces Diameter Cx e Dx para contactar o HSS e é este último que define o S-CSCF que o utilizador deve utilizar quando é contactado por uma rede externa via I-CSCF. O S-CSCF está localizado na *home network*.

Em suma, o Serving-CSCF executa as seguintes tarefas:

- É o responsável pelo controlo da sessão;
- Evita a utilização não autorizada dos serviços;
- É o responsável pelo mapeamento entre a localização do utilizador e o seu endereço público (por exemplo, o mapeamento entre o número de telefone e o IP do terminal);

O CSCF está situado no centro da rede IMS, na camada Session and Control Layer.

Home Subscriber Server

O Home Subscriber Server (HSS) é a base de dados central de utilizadores de uma rede IMS. O seu papel é semelhante ao Home Location Register (HLR) e ao Authentication Center (AuC) encontrados nas redes GSM. Contem o perfil de utilizador, informação de autenticação, autorização e localização física do mesmo.

De acordo com a especificação 3GPP, o HSS é a entidade que armazena a informação relativa aos subscritores para permitir que os elementos de rede possam fazer o tratamento das sessões. Proporciona suporte à gestão de mobilidade, estabelecimento de chamadas e/ou sessões, geração de informação de segurança dos utilizadores, identificação e autorização de acesso dos mesmos. O HSS guarda chaves secretas para os clientes móveis e gera informação de segurança dinâmica para cada um deles. Também fornece funções de aprovisionamento e autorização dos serviços.

O HSS também pode mapear o número de telefone do utilizador na rede PLMN e guardar informação como qual o tipo de terminal e a localização de cada cliente. Nele podem constar as identidades IP Multimedia Public Identity (IMPU), Multimedia Private Identity (IMPI IP), International Mobile Subscriber Identity (IMSI), Mobile Subscriber ISDN Number (MSISDN), entre outras para cada utilizador.

O Subscriber Location Function (SLF) é um elemento da rede IMS que está associado ao HSS. Este elemento tem a função mapear os endereços do utilizador com o respectivo HSS em redes IMS com múltiplas instâncias HSS. O HSS e o SLF comunicam por Diameter.

Media Resource Function

O MRF (Media Resource Function) é um *media server* IP que controla os *streams* multimédia como áudio, vídeo e texto sendo responsável por fornecer à rede funções de toque de tons, anúncios multimédia, áudio e vídeo-conferência, *video broadcast*, recolha de dígitos, etc. O MRF gere os *coders* para os vários *encoders/decoders* de *media* tendo, também, capacidade de *transcoding*, cancelamento de eco e como já referido detecção/geração de DTMF.

Cada MRF é dividido em duas partes: o Media Resource Function Controller (MRFC) e o Media Resource Function Processor (MRFP). O MRFC é o responsável pelo controlo da *media*. Funciona como um SIP User Agent (SIP UA) para o S-CSCF, um *broker* para o AS e controla o MRFP (Media Resource Function Processor) através da interface Mp usando o protocolo H.248 também conhecido por MEGACO. O MRFP implementa as funções de *media* supracitados, nomeadamente, toque de anúncios multimédia.

A Figura 4 mostra a arquitectura definida na norma TS 23.228.

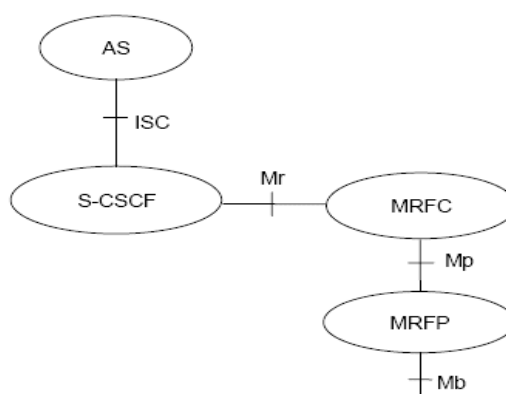


Figura 4 – Arquitectura do MRF

Application Server

Como já foi referido, uma das principais vantagens do IMS está no valor acrescentado dos seus serviços multimédia. O Application server (AS) como o seu nome indica é o servidor responsável pelo armazenamento e execução desses serviços multimédia. Mas dependendo do tipo de serviço, o AS pode funcionar como proxy, SIP User Agent ou B2BUA.

O SIP AS comunica com o S-CSCF que lhe encaminha determinadas sessões baseado em informações obtidas do HSS. De acordo com regras criadas para a sessão, o SIP AS decide qual das suas aplicações deverá tratar a sessão. Durante a execução da lógica do serviço, o SIP AS pode questionar o HSS para obter mais informação sobre o cliente. Para melhorar a experiência do serviço, a rede IMS antecipa as interações necessárias do serviço e faz uso de facilitadores como serviços de localização, presença e gestão de grupos.

Segundo a *Release 6* do IMS, o *application server* pode integrar três tipos de elementos:

- SIP-AS – Servidor de aplicações IMS nativo;
- IM-SSF – O interface IP Multimedia - Service Switching Function para aplicações CAMEL usando CAP;

- OSA SCS – O interface OSA Service Capability Server do *framework* OSA para a rede IMS. Usa OSA/Parlay para aceder à rede;

Contudo, na *Release 7* surge um novo conceito, o TAS – Telephony Application Server. É um SIP AS que proporciona suporte de rede a serviços telefónicos multimédia.

A Figura 5 mostra a arquitectura funcional para o aprovisionamento de serviços nas redes IMS.

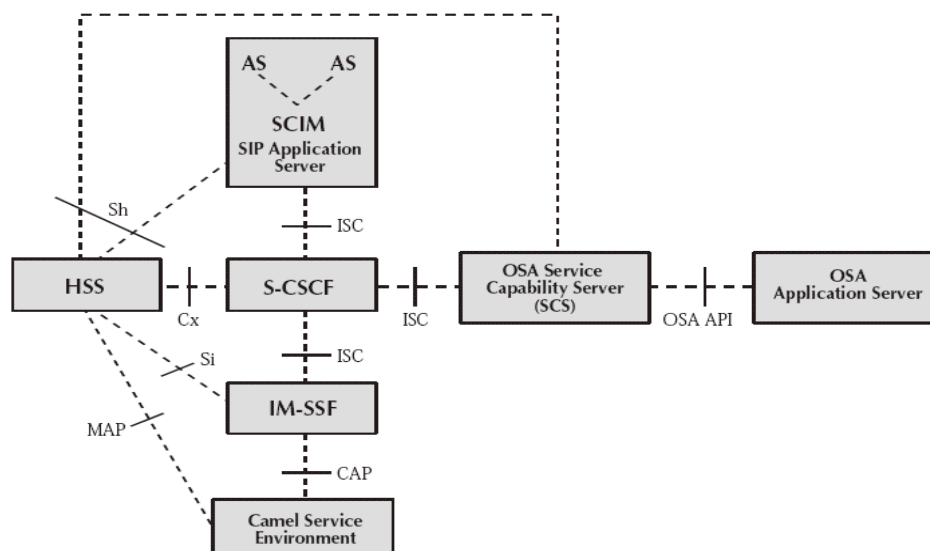


Figura 5 – Arquitectura funcional para os serviços IMS

Os *application servers* podem permitir o acesso dos serviços a outros elementos da rede IMS através de *servlets* SIP. Como a rede IMS é modular, se um novo serviço necessitar de algum requisito não disponível, é apenas necessário actualizar ou substituir o AS onde o serviço vai ficar alojado. Esta implementação contrasta com os modelos antigos das redes de telecomunicações em que cada serviço tinha os requisitos específicos de equipamento proprietário.

Para aumentar a flexibilidade na criação de serviços, um AS pode estar localizado na *home network* ou numa rede de terceiros (de um fornecedor de serviços). Se estiver localizada na *home network* pode comunicar directamente com o HSS através da interface Diameter Sh no caso de ser um SIP-AS ou via MAP se for um IM-SSF. Como os *applications servers* são servidores OSA podem aceder seguramente à rede IMS através de redes externas e interagir com servidores CAMEL GSM.

User Equipment

De todos os elementos da rede IMS, o terminal IMS é a entidade mais crítica para o sucesso global da rede e para garantir o acesso dos utilizadores às suas mais-valias. O UE actua como ambiente multi-aplicações potenciando o uso dos serviços disponíveis na rede IMS. O terminal faz a interface entre o utilizador e a rede IMS.

2.3 Gateways

Breakout Gateway Control Function

O Breakout Gateway Control Function (BGCF) é um servidor SIP que inclui funções de encaminhamento (*routing*) baseado em números de telefone. Este elemento da rede IMS está encarregado de seleccionar a rede em que o “*breakout*” PSTN deve ocorrer e na rede em que o *breakout* ocorre selecciona o Media Gateway Control Function (MGCF) responsável pelo controlo da IMS Media Gateway (IMS MGW). Nesse caso, o MGCF recebe a sinalização SIP pelo BGCF com o intuito de interligar às redes PSTN.

O BGCF apenas é usado quando é efectuada uma chamada da rede IMS para a rede de comutação de circuitos Public Switched Telephone Network (PSTN) ou Public Land Mobile Network (PLMN).

Media Gateway Control Function

O Media Gateway Control Function (MGCF) é o elemento responsável por controlar e gerir os recursos do MGW incluindo o *transcoding*, processamento de *media* usando H.248. Também controla a conversão de protocolos de *call control* feita pela Signalling Gateway (SGW) usando Stream Control Transmission Protocol (SCTP).

IMS Media Gateway

A IMS Media Gateway (IMS MGW) converte *streams* multimédia entre redes de pacotes e outras redes. No caso do IMS, é uma interface entre a rede IMS e outras redes de comutação de circuitos ou de comutação de pacotes. O seu papel principal é fazer *transcoding* de *media* de um formato para outro permitindo ao terminal IMS fazer e receber chamadas do mundo PSTN. Um exemplo é a conversão de Real-time Transport Protocol (RTP) com um *codec* de áudio Adaptive Multi-Rate (AMR) para Pulse-Code Modulation (PCM) com G.711 no sentido SIP – PSTN.

O MGW é controlado pelo MGCF usando o protocolo H.248 e opera na camada de transporte da rede IMS. Por vezes a Media Gateway e a Signalling Gateway estão juntas no mesmo elemento.

Signalling Gateway

A Signalling Gateway (SGW) implementa a interface de sinalização entre as redes de comutação de circuitos e as redes IP. O papel principal deste elemento é encapsular e transportar a sinalização do MGCF. Por exemplo, transforma o Stream Control Transmission Protocol (SCTP), um protocolo IP, em Message Transfer Part (MTP) parte da sinalização número 7 (SS7) para passar o ISDN User Part (ISUP) da MGCF para a rede de comutação de circuitos.

A Figura 6 demonstra a configuração de uma *signalling gateway* conforme a norma TS 23.228.

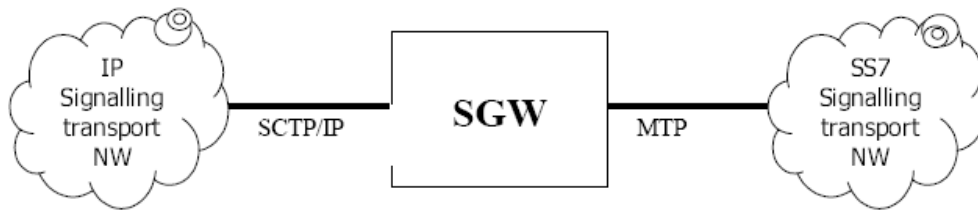


Figura 6 – Configuração de uma signalling gateway

IMS IPGW – IMS IP Gateway

A IMS IPGW permite que clientes IP não IMS *compliant* se possam autenticar na rede através da ligação Authentication and Key Agreement (AKA) existente entre ela e o P-CSCF. O cliente pode assim usar métodos de autenticação comuns como o MD5-digest entre ele e a IMS IPGW.

2.4 Charging

O *charging* ou taxação não é um elemento da rede IMS, é um conceito. Contudo, alguns elementos anteriormente apresentados como o P-CSCF, I-CSCF, S-CSCF, BGCF, MRFC, MGCF e AS usam interfaces de taxação que torna importante a apresentação deste conceito. No IMS há dois tipos de *charging*: *online* e *offline*.

A Figura 7 [Dev2dev 07] mostra uma conjugação da arquitectura de *charging online* e *offline* especificada na norma TS 32.260.

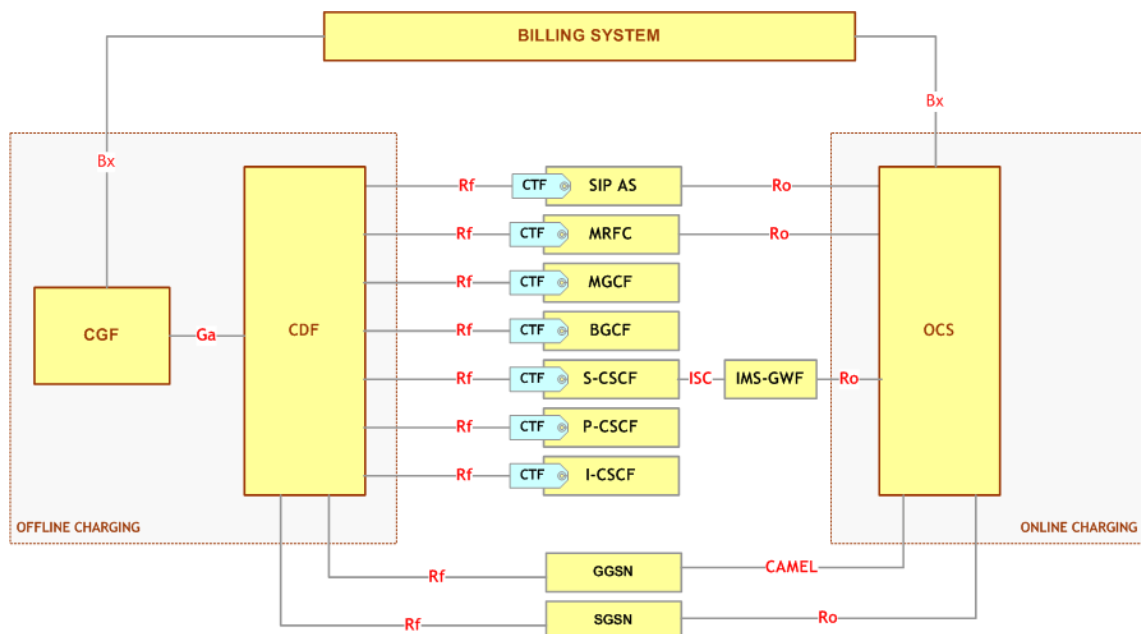


Figura 7 – Arquitectura de Charging IMS

No *online charging*, como o nome indica as funções de tarifação são usadas imediatamente. Por exemplo, se for usado o Immediate Event Charging (IEC), o número de unidades de crédito são instantaneamente deduzidas na conta do utilizador pelo Event Charging Function (ECF) e o MRFC ou AS são então autorizados a disponibilizar ou não o serviço.

O S-CSCF fala com um Session Charging Function (SCF) que age como um SIP AS comum. Esse SCF pode comunicar ao S-CSCF que termine a chamada quando terminarem os créditos.

A interface Diameter Ro permite que os Charging Trigger Function (CTF) enviem eventos de *charging* para o Online Charging Function (OCF). Estes eventos podem ser baseados em eventos ou em sessões.

Quando é usado o Event Charging with Unit Reservation (ECUR), o ECF primeiro reserva o número de créditos da conta do utilizador e só depois autoriza o MRFC ou o AS. No final do serviço, o número de créditos gasto é deduzido da conta.

No *offline charging*, como o nome indica a tarifação é feita posteriormente. Um exemplo de *offline charging* é a criação de Call Detail Records (CDR). O *offline charging* é aplicado aos utilizadores que usam os serviços periodicamente (p. ex: uma assinatura). O *online charging* por sua vez é usado para serviços pré-pagos ou para controlo de crédito *realtime* de pós-pagos. Todos os elementos SIP da rede envolvidos numa sessão (P-CSCF, I-CSCF, S-CSCF, BGCF, MRFC, MGCF, AS) usam a interface Diameter Rf para enviar informação de *accounting* para o Charging Collector Function (CCF). O CCF agrega esta informação e constrói um CDR que é enviado para o Billing System (BS).

Tanto o *offline* como o *online charging* podem ser usados numa sessão, até simultaneamente.

2.5 Interfaces

As interfaces IMS permitem que cada elemento da rede comunique com outros através de uma protocolo definido para essa interface.

A Tabela 1 faz uma descrição de todas as interfaces existentes numa rede IMS. Ao analisar esta tabela, fica claro que protocolos como o SIP e o Diameter são usados extensivamente nas redes IMS. Em relação ao Diameter, foram criados interfaces IMS específicos para tornar um protocolo normalizado nestas redes. Por exemplo, o Diameter é usado para implementar Autenticação, Autorização e Accounting (AAA).

Tabela 1 – Interfaces IMS

Nome	Entidades IMS	Descrição	Protocolos
Cr	MRFC, AS	Usado pelo MRFC retirar documentos (<i>scripts</i> ou outros) do AS	HTTP sobre canais TCP/SCTP
Cx	I-CSCF, S-CSCF, HSS	Comunicação entre o I-CSCF/S-CSCF e o HSS	Diameter
Dh	SIP AS, OSA, SCF, IM-SSF, HSS	Usado pelo AS para encontrar o HSS correcto num cenário de múltiplos HSS	Diameter
Dx	I-CSCF, S-CSCF, SLF	Usado pelos I-CSCF/S-CSCF para encontrar o HSS correcto num cenário de múltiplos HSS	Diameter
Gm	UE, P-CSCF	Usado para troca de mensagens entre o UE e os CSCFs	SIP
Go	PDF, GGSN	Permite o controlo de QoS e correlação de informação de entre a rede IMS e GPRS	COPS (Rel5), Diameter (Rel6+)
Gq	P-CSCF, PDF	Usado para troca de mensagens relativas a políticas P-CSCF e PDF	Diameter
ISC	S-CSCF, I-CSCF, AS	Usado para troca de mensagens entre o CSCF e o AS	SIP
Ma	I-CSCF -> AS	Usado para encaminhar os pedidos SIP que são destinados a um Public Service Identity existente num AS	SIP
Mg	MGCF -> I-CSCF	MGCF converte a sinalização ISUP em SIP e encaminha-a para o I-CSCF	SIP
Mi	S-CSCF -> BGCF	Usado para troca de mensagens entre o S-CSCF e BGCF	SIP
Mj	BGCF -> MGCF	Usado para troca de mensagens entre o BGCF e MGCF na mesma rede IMS	SIP
Mk	BGCF -> BGCF	Usado para troca de mensagens entre BGCFs de diferentes redes IMS	SIP
Mm	I-CSCF, S-CSCF, external IP network	Used for exchanging messages between IMS and external IP networks	Não especificado
Mn	MGCF, IM-MGW	Permite o controlo dos recursos <i>user-plane</i>	H.248 MEGACO
Mp	MRFC, MRFP	Usado para troca de mensagens entre o MRFC e MRFP	H.248
Mr	S-CSCF, MRFC	Usado para troca de mensagens entre o S-CSCF e MRFC	SIP
Mw	P-CSCF, I-CSCF, S-CSCF	Usado para troca de mensagens entre CSCFs	SIP
Rf	P,I,S-CSCF, BGCF, MRFC, MGCF, AS	Usado para troca de informação de <i>charging offline</i> com o CCF	Diameter
Ro	AS, MRFC	Usado para troca de informação de <i>charging online</i> com o ECF	Diameter
Sh	SIP AS, OSA SCS, HSS	Usado para troca de mensagens entre o SIP AS/OSA SCS e o HSS	Diameter
Si	IM-SSF, HSS	Usado para troca de mensagens entre o IM-SSF e o HSS	MAP
Sr	MRFC, AS	Usado pelo MRFC retirar documentos (<i>scripts</i> e outros) do AS	HTTP
Ut	UE, AS (SIP AS, OSA SCS, IM-SSF)	Permite ao UE gerir informação relacionada com os serviços	HTTP(s)

3 Redes IMS e aplicações

3.1 Benefícios

As redes IMS são anunciadas como redes da próxima geração que terão como benefício principal uma verdadeira convergência de serviços, aplicações, redes e terminais. Essa convergência é potenciada através da arquitetura de rede em camadas [IP Unity 05].

A convergência de terminais permite que qualquer terminal móvel ou fixo, como telefones, PDAs, computadores pessoais ou televisores, possam aceder à rede.

A convergência de redes significa que a rede móvel, a rede fixa e as redes de banda larga são vistas como uma entidade para a camada de abstracção IMS. Desta forma, serviços de rede são oferecidos independentemente do tipo de acesso. A convergência de rede envolve todas as funcionalidades necessárias aos serviços, como acesso aos perfis dos utilizadores, autenticação, facturação, serviços de localização e controlo de recursos multimédia através de APIs abertas e normalizadas [Tompros 07].

Finalmente, as aplicações convergentes IMS podem residir no fornecedor de serviços ou em qualquer plataforma do operador, podem aproveitar e usar serviços comuns de rede e serem acedidas por subscritores a partir de qualquer rede ou terminal. Podem ser inclusivamente aplicações da rede inteligente (IN) portadas para a rede IMS usando o IM-SSF.

O IMS trará inúmeros benefícios para os fornecedores de serviços como a facilidade de criar e instalar novas aplicações e serviços já que as interfaces são normalizadas. A complexidade desses serviços será mais reduzida. As aplicações serão mais fáceis de desenvolver usando APIs abertas e serviços de rede partilhados [Handa 05]. Como as interfaces são normalizadas e não dependem de protocolos proprietários, os fornecedores de serviços poderão mais facilmente fornecer pacotes de serviços oferecendo descontos de quantidade.

Outro benefício será a reutilização. O recurso a pequenos componentes permite que sejam reutilizados na construção de novos serviços minimizando o tempo de desenvolvimento e melhorando o tempo de entrada em exploração comercial dos mesmos. Em alguns casos, não será necessário o desenvolvimento de novos módulos, apenas recombinação dos mesmos.

Como a rede permite a partilha dos recursos de rede com terceiros, os fornecedores de serviços externos podem aceder à rede para oferecer as suas aplicações e serviços podendo até os operadores optar por cenários de partilha dos lucros para reduzir os riscos dos investimentos em serviços.

Mas a vantagem mais apelativa para os fornecedores de serviços será a possibilidade de criar novos serviços multimédia. O acesso a recursos multimédia (voz, vídeo e dados) estará disponível na mesma chamada.

Os serviços IMS permitirão captar novos clientes e fidelizar os existentes recorrendo a uma melhor qualidade de voz e vídeo para aplicações comerciais, como conferência, usando *wideband coders*. Por outro lado, a convergência permite a oferta de aplicações móveis (como SMS, etc.) a subscritores fixos ou de banda larga.

Os clientes empresariais também poderão retirar mais valias das redes IMS como a personalização dos serviços conforme o seu modelo de negócio. Mesmo havendo dois clientes

numa determinada área de negócio, os seus modelos podem ser diferentes. Com o IMS, a implementação de serviços personalizados é mais facilitada. Por outro lado, a rede permite uma melhoria das capacidades de comunicação e marketing, dado que os clientes podem expandir os seus modelos de comunicação para incluir qualquer aplicação ou serviço internet. Contudo, a vantagem que poderá fazer a grande diferença para o cliente empresarial será a redução do tempo de entrega do serviço ao cliente. No mercado global em que as empresas competem, pode ser determinante o tempo que leva desde a concepção da ideia até à colocação da solução no mercado.

Do ponto de vista dos operadores, as redes IMS permitem aumentar o grau de satisfação clientes e reduzir os custos com criação de serviços (podendo mais facilmente optar por modelos de *outsourcing*) podendo usufruir de uma redução nos custos de operação e Total Cost of Ownership (TCO). A própria implementação dos serviços será mais eficiente dado que um único serviço pode ser fornecido a vários tipos de acesso (fixo, móvel, banda larga). Para além destas vantagens, as redes IMS podem trazer aos operadores um aumento da simplicidade de operação e manutenção das redes dado que os sistemas de gestão, aprovisionamento e facturação são comuns a todas as redes. Novamente se pode diminuir custos através da simplificação de processos. Mas as vantagens económicas estendem-se, também, aos custos com a rede de transporte que sofrerão uma redução significativa com a migração de canais de comutação de circuitos para comutação de pacotes (infra-estrutura IP). Finalmente, os operadores terão a capacidade de oferecer novos serviços e produtos baseados em recursos da Internet.

Portanto, a implementação de redes IMS resultará numa redução de despesas para disponibilizar conteúdos aos subscritores em qualquer formato, dispositivo ou rede.

O IMS será importante para os operadores móveis porque as redes “pré-IMS” são centradas na voz. Mesmo as redes de vídeotelefonia 3G (3G-H.324M) operam sobre canais de comutação de circuitos. Mas as tecnologias de rádio móvel (WiFi, WiMax, etc.) estão a aproximar-se das velocidades banda larga. Se os operadores não suportarem serviços baseados em IP, outros fornecedores o farão usando o acesso à internet. Também será importante para os operadores fixos que têm perdido relevância face aos operadores móveis e VoIP. No mínimo, precisam de oferecer convergência fixo-móvel mas idealmente precisam de fornecer novos serviços. O IMS tem potencial para fornecer esses novos serviços com uma qualidade que a voz sobre IP comum não pode. Outra promessa do IMS é a rapidez de criação e entrega de aplicações novas ou recombinadas usando o Service Capability Interaction Manager (SCIM).

3.2 Serviços

Alguns dos serviços básicos previstos para uma rede IMS incluem [Croslin 05]:

- Vídeo e *messaging* móvel bidireccional
- Colaboração
- *Call centers* baseados em IP e *video call centers*
- *Video-on-demand* e *download* de conteúdos

- Serviços baseados em localização
- Evolução dos actuais serviços *push-to-talk* e conferência
- Convergência fixo-móvel.

A Figura 8 mostra exemplos dos serviços básicos IMS.



Figura 8 – Serviços IMS

Serviços Quad-Play

Como já foi mencionado, o IMS utiliza o IP para disponibilizar serviços de vídeo, voz e dados sobre qualquer tipo de acesso fixo ou móvel. Desta forma, o IMS tem a capacidade de criar uma oferta *quad-play* sem precedentes. Ao unir a Television over IP (IPTV) com o IMS, as *set-top boxes* tornam-se terminais IMS, tal como qualquer outro tipo de terminal (PC, telemóvel, etc.).

Desta forma, os utilizadores podem usufruir, consistentemente, de uma vasta gama de serviços entre redes de acesso. Os serviços de voz e dados podem ser expandidos à IPTV da mesma forma que a um terminal fixo ou móvel.

Os operadores podem oferecer novos serviços que ajudam os utilizadores a gerir e aceder a bibliotecas pessoais de conteúdos comerciais ou privados fazendo-o a partir de qualquer terminal. Os serviços de vídeo, como Digital Video Recorder (DVR) em rede, podem ser expandidos da TV para o terminal móvel permitindo aos utilizadores levar os seus conteúdos consigo [Nortel 01].

Este tipo de serviços avançados não só compete no preço, como representa uma oferta integrada que proporciona uma experiência mais rica. Ao aumentar o valor dos serviços fornecidos, o operador pode aumentar os ganhos por utilizador e aumentar a fidelização dos mesmos.

Exemplos de serviços de telefonia na TV:

- Visualização do número do chamador
- Click-to-Call
- Gestão de chamadas:
 - o Ignorar ou rejeitar chamadas
 - o Encaminhamento para *voicemail*
- Indicador de *voicemail*
- Chamada de vídeo
- Vídeo-conferência

Exemplos de serviços de dados na TV:

- *Chat – Instant Messaging*
- Partilha de informação
 - o Partilha de fotos
 - o *Upload* para *blogs* ou *webpage*
- Lista de amigos e presença
- Lista de endereços como o Outlook/PDA.

A antevisão estes serviços que resultam da união da IPTV com as redes IMS traz perspectivas muito animadoras. Estas novas capacidades vão enriquecer a experiência do utilizador através da utilização de uma panóplia de terminais para ver TV ou sessões interactivas mas também podem acrescentar capacidades multimédia à TV ou a terminais móveis. Esta convergência entre terminais e televisor permitirão partilhar conteúdos pessoais com outros subscritores e gerir ou aceder a conteúdos de vídeo em qualquer lugar. Esta combinação do vídeo com serviços de comunicação aumentará a produtividade e/ou entretenimento trazendo mais valias em todas as soluções.

3.3 Desafios

Apesar de na actualidade o IMS parecer a evolução lógica das redes de telecomunicações, ainda enfrenta alguns obstáculos [NMS 06 B]:

O primeiro é a complexidade. Há muitas especificações IMS com múltiplas versões que continuam a evoluir e os testes de interoperabilidade não têm acompanhado o desenvolvimento e a introdução do produto.

O segundo desafio é o dos serviços. O IMS é uma plataforma que pode suportar novos serviços, mas o operador não ganha dinheiro enquanto estes são lançados e se tornam rentáveis. Nos dias de hoje, há um número relativamente pequeno de novos serviços disponíveis com o IMS e, quando comparado com a internet, tem uma pequena comunidade de desenvolvedores.

O terceiro é a normalização. A especificação R7 tem estado parada em termos de novas funcionalidades mas ainda não foi lançada oficialmente o que faz com que funcionalidades como o suporte para telefonia sobre cabo tenham sido relegadas para a *Release 8*.

A especificação R5 é considerada incompleta e várias alterações significativas foram introduzidas na R6 e na R7 (por exemplo na infra-estrutura) o que significa que os desenvolvimentos para a R6 terão que ser remodelados para o R7.

O quarto é o lançamento de equipamentos compatíveis. Os terminais (*handsets*) IMS são críticos para determinadas aplicações como IM e presença. No entanto, há aplicações que não necessitam ou podem circundar a necessidade, de terminais específicos.

O quinto desafio está relacionado com o lançamento de tecnologias recentes como o High Speed Downlink Packet Access (HSDPA) e High Speed Uplink Packet Access (HSUPA) (sistemas rádio que fornecem banda larga a redes UMTS) [Turner 07 A] que aumentam a probabilidade de um ou mais operadores oferecer uma tarifa plana para o acesso à internet. Esse tipo de tarifas, associados aos elevados débitos, aumentam a probabilidade de terceiros desenvolverem serviços (não IMS *compliant*) sobre essas redes.

Estes factores aliados aos serviços implementados directamente sobre IP ou sobre a internet trazem dificuldades à implementação do IMS. Contudo, estes serviços não usufruem da qualidade de serviço e segurança do IMS.

Como o IMS para as redes fixas só começou com a especificação R7, a maioria das implementações IMS são uma mistura de elementos SIP (IMS-*compliant* ou não) e redes tradicionais. Sendo as redes fixas IMS tipicamente constituídas por circuitos de voz controlados por um *softswitch* com alguma infra-estrutura SIP para novas aplicações como *instant messaging* ou convergência fixo-movel. Contudo, há redes totalmente IMS a serem instaladas mas são usadas apenas para novas aplicações como IM, Fixed-Mobile Convergence (FMC) e *video sharing*; não como redes de telefonia básica.

Em suma, as aplicações ou serviços atraentes ao público (aplicações rentáveis) são uma questão importante. A normalização e interoperabilidade aumenta as opções disponíveis para o operador quando pretende adquirir um equipamento mas no final, só novas aplicações farão aumentar os lucros. No entanto, estas novas aplicações poderão não se reflectir em novos serviços a baixo custo para os clientes; dependerá da abertura dos operadores ao desenvolvimento por terceiros.

4 Componentes Open Source

Este capítulo descreve o “estado da arte” de soluções *open source* que implementam (ou poderão vir a implementar após determinadas alterações) cada componente IMS e posteriormente serão avaliadas para integrar a solução final. Contudo, dado que o IMS é uma normalização vocacionada para os operadores de telecomunicações e não para o mundo de IT, há componentes que não existem na comunidade *open source* por não serem atraentes ao público em geral. Esses componentes terão que ser desenvolvidos de raiz. Assim sendo, também serão referenciadas *stacks* SIP, Diameter e outras soluções que poderão ser usadas como base para esse desenvolvimento. Há, também, a eventualidade desses componentes serem adquiridos comercialmente. Por outro lado, há componentes apenas são disponibilizados por um projecto *open source*. Esses serão automaticamente seleccionados para a solução final.

O *software open source* é definido como *software* cujo código fonte é disponibilizado ao público permitindo a qualquer pessoa copiar, modificar e redistribuir o código sem pagar royalties. O *software open source* é desenvolvido por comunidades de programadores individuais e grandes companhias.

Contudo, no caso das grandes empresas pode colocar-se a questão: se o *software* é gratuito, o que ganham as empresas em pertencer à comunidade? A resposta é simples. As empresas podem não ganhar ao ver recompensado o custo de desenvolvimento com a venda do *software* ou por receber royalties de utilização mas vão procurar ganhar dinheiro noutra forma. Veja-se o exemplo da Digium que é um fabricante de placas telefónicas que lançou um sistema de PBX IP *open source* denominado Asterisk PBX [Asterisk 07] especialmente desenvolvido para utilizar as suas placas. Segundo a estratégia da empresa, era pretendido que os utilizadores do Asterisk ao ter necessidade de fazer a ligação entre o mundo IP e telefónico adquirissem o *hardware*. Outro exemplo menos nobre é o de empresas que lançam um produto novo e o oferecem à comunidade como o OpenVXI [OpenVXI 07], um interpretador de VoiceXML, e aguardam que os contribuidores madurem o produto e corrijam os *bugs*. Ao fim de algumas interacções, lança uma evolução comercial do mesmo abandonando a versão *open source*.

Deve ser, desde já, mencionada a valorosa contribuição de uma iniciativa *open source* denominada Open IMS Playground lançada pelo Fraunhofer Institute FOKUS em Julho de 2004 que pretende desenvolver e integrar os componentes do core IMS numa plataforma. Segundo a FOKUS, o “*playground*” é um ambiente de testes NGN/IMS aberto e independente que pode ser usado como tanto academicamente como pela indústria para testar componentes, protocolos e aplicações. Alguns destes componentes serão inevitavelmente na solução final dada a escassez de propostas ou ausência de soluções concorrentes.

Neste capítulo também serão analisados alguns sistemas não *open source* mas gratuitos dado que num contexto não comercial e meramente académico poderão ter alguma relevância.

4.1 Call Session Control Function

OpenSER

O OpenSER é projecto *open source* (GPL) que pretende criar um servidor SIP robusto e escalável. É um *spinoff* do FhG FOKUS SIP Express Router (SER) [OpenSer 07 A].

Vantagens deste pacote:

- Servidor SIP gratuito com alto desempenho e elevada flexibilidade;
- Implementa *authentication, authorization and accounting* (AAA) usando Base de Dados, RADIUS ou DIAMETER;
- Pode-se facilmente construir novos módulos para serem adicionadas novas funcionalidades;
- Muito popular na comunidade *open source*;
- Adoptado por diversos vendedores e universidades;

Desvantagem:

- Não é IMS *compliant*, necessitando de desenvolvimento;

SER

O projecto SER – SIP Express Router é muito popular na comunidade *open source* [SER 07].

Vantagens deste pacote:

- Servidor SIP gratuito com alto desempenho e elevada flexibilidade;
- Pode actuar como servidor SIP Proxy, Redirect e Registrar;
- Podem-se facilmente construir novos módulos para serem adicionadas novas funcionalidades;
- O mais popular na comunidade *open source*;
- Adoptado por diversas empresas e universidades;

Desvantagem:

- Não é IMS *compliant*, necessitando de desenvolvimento;

OpenIMS Core

O projecto *open source* OpenIMS Core está a ser desenvolvido pelo instituto Fraunhofer Institute FOKUS e pretende colmatar o vazio existente no mundo do *open source* em relação ao IMS. O objectivo do projecto é criar uma comunidade de *developers* de componentes para redes de próxima geração IMS. A PT Inovação [PT Inovação 07] é um dos parceiros que contribuem para este projecto.

A ideia na génese deste *software* era a de permitir aos criadores de serviços IMS um ambiente de teste com elementos do *core* altamente flexíveis. O ambiente pode ainda ser aplicado para provas de conceito ou testes de integração [OSIMS 07].

Vantagens deste pacote de *software*:

- Implementa o CSCF e HSS;
- O CSCF é baseado no SIP Express Router (SER) mencionado anteriormente, do qual retira as suas vantagens;
- É IMS *compliant*;
- A sua interoperabilidade tem sido testada com produtos IMS comerciais;
- Disponibiliza uma interface IMS Service Control (ISC) para permitir à camada de aplicação a criação de serviços IMS para várias plataformas (SIP AS, OSA/Parlay ou IN através do IM-SSF);

4.2 Home Subscriber Server

OpenIMS Core

Como mesmo as funções mais básicas de encaminhamento de sinalização no IMS necessitam de uma consulta ao HSS, este foi incluído no OpenIMS Core para permitir uma utilização normal do mesmo. Assim sendo, o Open IMS core disponibiliza uma implementação simples HSS, o FOKUS Home Subscriber Server (FHoSS) [OSIMS 07].

Vantagens deste pacote de *software*:

- É IMS *compliant* evitando a necessidade de integração;
- Foi o único HSS *open source* encontrado;

4.3 Media Resource Function

Asterisk

O Asterisk PBX é provavelmente o IP PBX *open source* mais popular. O projecto é gerido pela Digium e é utilizado por outros projectos open source como o TrixBox para criar uma solução para *call centers*, por exemplo. É definido pelos seus criadores como um motor de telefonia e um *toolkit*. Suporta vários protocolos, tais como SIP, H.323, IAX, MGCP, Skinny e GoogleTalk. [Asterisk 07].

Vantagens deste pacote:

- Servidor PBX IP gratuito com alto desempenho e elevada flexibilidade;
- Pode actuar como servidor SIP Registrar, *Location Server* e pode actuar como UAC;
- O mais popular na comunidade *open source*;
- Adoptado por diversos vendedores e universidades;
- É integrado em projectos como o TrixBox que pode ser uma óptima solução para um pequeno *call center* IP;
- Pode funcionar como Gateway SIP-ISDN usando placas Digium ou Sangoma ou Gateway SIP-SS7 incluindo a biblioteca libSS7;

- Pode funcionar como Gateway SIP-3G usando a biblioteca *open source* h324m [FontVenta 07];
- Tem capacidade de áudio e vídeoconferência com o MeetMe e app_mixer ligado a um Multipoint Conference Unit (MCU), respectivamente;

Desvantagem:

- Não é IMS *compliant*, necessitando de desenvolvimento;

A Tabela 2 [Pika 07] mostra as estatísticas de um estudo de mercado executado pela PIKA Technologies sobre um número de 322 participantes.

Tabela 2 – Estatísticas de utilização Asterisk PBX

Type	Using Asterisk for own internal application – 30%
	Using Asterisk to develop system for re-sale – 70%
Location	N/S America - 67%
	Europe - 33%
Currently using Asterisk®	1.0 - 5%
	1.2- 49%
	1.4 - 46%
Currently using Linux distribution	AsteriskNOW - 2%
	CentOS - Total: 34% v4.3 - <1%; v4.4 - 25%; v4.5 - 1%; v5.0 - 7%
	Debian 4 - 20%
	Fedora - Total: 16% Core 3 - <1%; Core 4 - 5%; Core 5 - <1%; Core 6 - 8%; Core 7 - 1%
	Ubuntu – 6%
	Slackware – 6%
	Other (all less that 1%): CoLinux, CRUX, Edgy EFT, Free BSD, Gentoo, Enterprise 3, Enterprise 4, Mandriva, Open SUSE, PLD, Red Hat
Applications being built on Asterisk®	Standard business telephone system for in-house use: 36%
	Standard business telephone system for re-sale: 29%
	Call Center telephone system for re-sale: 6%
	Hosted business PBX: 4%
	VOIP long distance service: 4%
	System used in own home: 3%
	Special purpose business telephone system: 3%
	Predictive dialing: 3%
	Gateway: 2%
	Educational purposes: 3%
	Misc other: all less than 1% Credit Card, IVR, Government, Distance Learning, Fax, Messaging, Video, Conferencing, Testing, Logging

SEMS – SIP Express Media Server

O SEMS é baseado no *software open source* SER e foi desenvolvido pela iptel.org. Corre em Linux e é distribuído com uma variedade de exemplos de aplicações como *voicemail*, IVR, audioconferência, etc [SEMS 07].

Vantagem deste pacote:

- Baseado no SER que é conhecido pelo seu alto desempenho e elevada flexibilidade;

Desvantagem:

- Não é IMS *compliant*, necessitando de desenvolvimento;

CallWeaver

O CallWeaver, anteriormente conhecido como Open PBX é um *spin-off* do Asterisk PBX mas licenciado com a licença GNU GPL. Assim sendo, ambos os projectos partilham um código base idêntico mas o CallWeaver tem uma comunidade de desenvolvimento menor o que o torna menos atraente [CallWeaver 07].

Vantagem deste pacote:

- Partilha código com o Asterisk;

Desvantagens:

- Não é IMS *compliant*, necessitando de desenvolvimento;
- Comunidade de desenvolvimento menor do que a do Asterisk;

SIPfoundry SipX

A SIPfoundry é uma comunidade *open source* que fornece vários *development kits* SIP e VoIP. A sua *stack* SIP é usada em diversos produtos e a solução sipX PBX é usada por muitas empresas. Este projecto está ligado aos produtos Vision, permitindo interoperabilidade com RDIS e SS7.

A solução sipX consiste num PBX SIP empresarial *open source*. Projectado como uma solução puramente SIP, é distribuído com múltiplas capacidades como *voicemail* e *auto-attendant*. O sipX é indicado para pequenas ou grandes implementações, suportando *call control* redundante com alta disponibilidade (HA) assim como um interface Web baseado em Simple Object Access Protocol (SOAP). Desta forma, o sipX oferece uma gestão web *plug & play* que inclui gestão e configuração dos dispositivos ligados (telefones e gateways).

O sipX é um servidor modular que corre em Linux e não necessita de *hardware* extra para interoperar com telefones, gateways ou aplicações SIP [SipX 07].

Vantagem deste pacote:

- Solução robusta;

Desvantagens:

- Não é IMS *compliant*, necessitando de desenvolvimento;

- Não tem suporte para H.323 ou outros protocolos para além do SIP;

YATE

O Yet Another Telephony Engine é um PBX e cliente *open source* com recursos SIP, H.323 e IAX. Funciona como um tradutor SIP-H.323 sendo suportado comercialmente pela Null Team **Error! Reference source not found.**

Vantagens deste pacote:

- Solução interessante para *gateway* SIP-H.323;
- Pode funcionar como Gatekeeper H.323;

Desvantagem:

- Não é IMS *compliant*, necessitando de desenvolvimento;

FreeSWITCH

A FreeSWITCH é uma plataforma de telefonia *open source* desenhada para facilitar a criação de produtos orientados à voz e ao *chat*. Pode ser usado apenas como motor de comutação, *media gateway* ou um *media server* com aplicações de IVR criadas usando scripts ou XML [FreeSwitch 07].

Suporta diversos protocolos, como SIP, H.323, IAX2 e GoogleTalk tornando muito simples a interface com outros PBXs *open source*, como o sipX, OpenPBX, Bayonne, YATE ou Asterisk.

Vantagem deste pacote:

- Suporta diversos protocolos (SIP, H.323, IAX2 e GoogleTalk);

Desvantagem:

- Não é IMS *compliant*, necessitando de desenvolvimento;

Bayonne

O GNU Bayonne é um servidor de telefonia *open source* que oferece um ambiente de desenvolvimento para sistemas escaláveis e independentes do tipo de *media* usada [Bayonne 07].

Vantagem deste pacote:

- Desenvolvido pela GNU;

Desvantagens:

- Não é IMS *compliant*, necessitando de desenvolvimento.
- O projecto parece estar parado;

Vovida Vocal

O projecto parece estar parado e há dificuldades em aceder ao site [Vovida 07].

4.4 Application Servers

SIPSEE

O *application server* SIP Servlet Execution Environment (SIPSEE) é a solução da FOKUS para desempenhar o papel de SIP Application Server (SIP AS) com base em tecnologia de *servlets* SIP. O SIPSEE integra a camada de aplicação e dá aos serviços IMS a capacidade de controlo das sessões multimédia sobre SIP/HTTP/Diameter.

O SIPSEE actua como um SIP proxy, SIP Redirect ou um Back to Back User Agent (B2BUA). A Application Programming Interface (API) usada pelo SIPSEE é compatível com o JSR116 - SIP Servlet API 1.0, que tem vindo a ser adoptada como API HTTP *servlet*.

O SIPSEE foi desenhado e implementado com suporte para SIP e HTTP de modo a permitir o acesso a serviços IMS por parte de clientes SIP ou através de *browser* HTTP potenciando serviços como o *Click2Dial*, *See What I See*, etc. Desta forma, une os protocolos mais usados da internet e do VoIP. Implementa, também, recursos de IMS Session Control (ISC), acesso ao perfil do utilizador, taxação IMS e Parlay X (versão 2.1) web services [SIP SEE 07].

Vantagem deste pacote:

- IMS *Compliant*;

Desvantagem:

- Não é open source

SailFin

O GlassFish é um Application Server *open source* para a plataforma Java EE 5 desenvolvido pela Sun Microsystems. Desde 2005 que a Sun se uniu à Ericsson para ajudar a comunidade a desenvolver um Application Server IMS *compliant*. O SailFin assenta no GlassFish para fornecer capacidades SIP *servlet* ao mesmo [SailFin 07].

Vantagens deste pacote:

- IMS *Compliant*;
- A Ericsson faz parte da comunidade de desenvolvimento;

Mobicents

O Mobicents é um SIP *application server open source* para a plataforma Java. É ambiente de desenvolvimento Jain Service Logic Execution Environment (Jain SLEE) que permite a migração dos actuais serviços IN para as redes IMS definindo Resource Adapters (RA) para vários protocolos de comunicação. O projecto já inclui RAs para SIP, Asterisk, eXtensible Messaging and Presence Protocol (XMPP), Diameter, e Parlay/Parlay-X.

O Mobicents é certificado para JSLEE 1.0 e complementa o J2EE para permitir aplicações de voz, vídeo e dados convergentes e de próxima geração. Permite a criação de Service Building Blocks (SBB) para *call control*, *billing*, aprovisionamento de utilizadores, administração e presença. A arquitectura extensível permite criar pontos de integração com aplicações empresariais como Web, Customer Relationship Management (CRM) e Service-Oriented Architecture (SOA). Integra, também, componentes de gestão e monitorização via SLEE baseado em Java Management Extensions (JMX) e Simple Network Management Protocol (SNMP). Isto faz com que o servidor possa ser integrado com sistemas Operations Support Systems (OSS) e Network Management Systems (NMS) [Mobicents 07].

Vantagens deste pacote:

- IMS *Compliant*;
- Muito popular;
- Arquitectura *event driven* para elevados volumes de sinalização e baixa latência;

WeSIP

WeSIP é um SIP e HTTP *application server* construído sobre o OpenSER. Não é *open source* sendo gratuito para uso não comercial.

WeSIP adiciona uma camada Java 2 Enterprise Edition (J2EE) ao OpenSER de forma a aproveitar alguns dos módulos e recursos já existentes como *Registrar*, *NAT Transversal*, autenticação Diameter ou *accounting* Radius. Desta forma, permite a criação de aplicações convergentes SIP e HTML num ambiente J2EE [WeSIP 07].

Desvantagem:

- Não é *open source* mas é gratuito para uso não comercial;

4.5 User Equipment

OpenIC

O OpenIMS Client está a ser desenvolvido pelo instituto Fraunhofer Institute FOKUS e implementa um User Equipment (UE) IMS permitindo o acesso aos serviços multimédia de uma rede IMS. Este terminal não é *open source*. O nome open pretende mostrar que é expansível mas não deve ser confundido com *open source*.

O OpenIC continua em desenvolvimento para melhorar a sua robustez. Consiste num *framework* que oferece um interface de programação que permite desenvolver aplicações IMS compatíveis com as especificações IETF, 3GPP, TISPAN e pretende ser um cliente IMS multi-serviços para Fixed Mobile Convergence (FMC) [OpenIC 07].

O OpenIC suporta sistemas operativos Microsoft Windows Mobile ou Windows XP [MS Windows 07] e Linux.

Vantagem deste pacote:

- IMS *Compliant*;

Desvantagem:

- Não é *open source*;

A Figura 9 [OpenIC 07] mostra uma parte da arquitectura de rede IMS em que o OpenIC se insere.

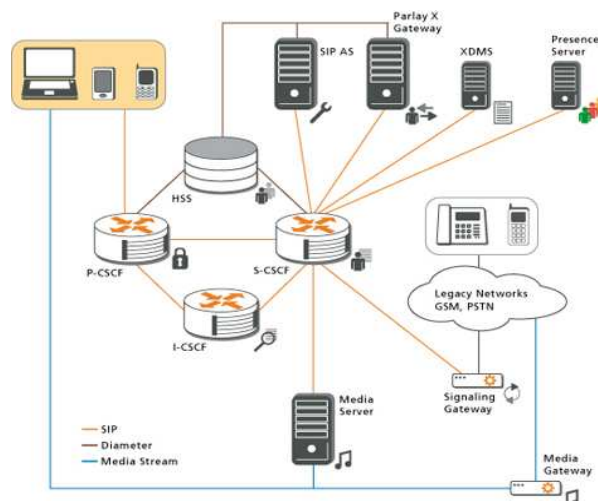


Figura 9 – Exemplo do OpenIC numa rede IMS

IMS Communicator

O IMS-Communicator é baseado na antiga versão do *softphone* SIP-Communicator em Java [IMS Com 07]. Foi construído em cima do JAIN-SIP RI, tendo sido feitas contribuições para suportar extensões SIP IMS definidas pelo 3GPP e IETF. O IMS-Communicator usa a *stack* de *media* fornecida pela API Java Media Framework (JMF).

O IMS Communicator está a ser desenvolvido pela PT Inovação, SA. Implementa um User Equipment (UE) IMS com capacidades vídeo [IMS Dev 07 A].

Vantagens deste pacote:

- IMS *compliant*;
- Suporta vídeo;
- É *open source*;

UCT IMS Client

O UCT IMS Client foi desenvolvido pelo Communications Research Group at the University da cidade do cabo na África do sul. Actualmente o projecto ainda está activo mas tem muitos bugs conhecidos.

O cliente suporta autenticação AKA, e tenta emular a sinalização IMS. A versão actual suporta chamadas de voz e vídeo, *pager-mode instant messaging*, (XML Configuration Access Protocol) XCAP, funcionalidades de IPTV e presença [UTC IMS 07].

Vantagens deste pacote:

- IMS *compliant*;

- É *open source*;

Desvantagens:

- Pode revelar problemas de inter-operação já que claramente afirma que foi construído para ser usado com o OpenIMS Core;
- Tem alguns *bugs* conhecidos;

MiniSIP

O Minisip é um User Agent SIP. Tem capacidade de efectuar chamadas de áudio/vídeo e *instant messaging*.

Vantagens deste pacote:

- Suporta vídeo;
- Suporta muitos sistemas operativos (Linux PC, Linux familiar IPAQ PDA, Windows XP e Windows Mobile 2003 SE);
- É *open source*;

Desvantagem:

- Não é *IMS compliant*, necessitando de desenvolvimento;

X-Lite

Desenvolvido pela CounterPath, não é *open source* mas é gratuito. É um *softphone* *IMS compliant*, muito popular, muito intuitivo e com capacidades vídeo.

Vantagem deste pacote:

- É *IMS compliant*;

Desvantagem:

- Não é *open source*;

SJphone

O SJPhone é um *softphone* não *IMS compliant* muito popular, não *open source* mas gratuito.

Desvantagens deste pacote *open source*:

- Não é *IMS compliant*; necessitando de desenvolvimento;
- Não é *open source*;

Kapanga

Não é *open source* mas é gratuito. É um *softphone* não *IMS-compliant* mas com muitas opções de configuração. Existe uma versão para Windows Mobile. Dispõe de *codecs* vídeo como o H.264 e áudio como o AMR (na versão paga).

Desvantagens deste pacote *open source*:

- Não é IMS *compliant*;
- Não é *open source*;

4.6 Gateways

Durante o trabalho de pesquisa que antecedeu esta secção não foi encontrado nenhum projecto que implementasse os componentes BFCF e MGCF de uma rede IMS.

Contudo, há projectos *open source* não IMS *compliant* que aliados a *hardware* comercial podem implementar *media* e *signalling gateways*.

Em relação à componente IMS-GW também não há projectos *open source* que sejam IMS *compliant* mas qualquer SIP Registrar pode ser modificado para efectuar esta função.

Asterisk

O Asterisk PBX dispõe de suporte para placas telefónicas com interface CAPI ou ZAPTEL. Com essas placas pode implementar facilmente uma gateway SIP-ISDN. Através da libSS7, é possível implementar uma gateway SIP-SS7 e usando a H324m lib [FontVenta 07] é possível implementar uma gateway SIP-3G tornando-se esta solução uma forte candidata para implementar as componentes de gateway IMS [Asterisk 07].

Vantagens deste pacote:

- Pode actuar como servidor SIP Registrar, *location server* e pode actuar como UAC;
- Pode funcionar como Gateway SIP-ISDN usando placas Digium ou Sangoma e como Gateway SIP-SS7 usando as mesmas placas e a biblioteca libSS7;
- Pode funcionar como Gateway SIP-3G usando a biblioteca h324m da sip.fontventa.com;

Desvantagem:

- Não é IMS *compliant*, necessitando de desenvolvimento;

SIPfoundry SipX

Este projecto está ligado aos produtos Vision, permitindo interoperabilidade com RDIS e SS7 [SipX 07].

Vantagens deste pacote:

- Pode funcionar como Gateway SIP-ISDN usando produtos Vision;
- Pode actuar como Gateway SIP-SS7 usando produtos Vision;

Desvantagens:

- Não é IMS *compliant*, necessitando de desenvolvimento;
- Não tem suporte para Gateway SIP-3G;

YATE

Pode implementar uma gateway para clientes IP H.323 que não suportem SIP **Error! Reference source not found.**

Desvantagem:

- Não é IMS *compliant*, necessitando de desenvolvimento;

4.7 Ferramentas de teste

Esta secção apresenta dois projectos que implementam ferramentas que podem ser usados na solução final para testes de carga.

Seagull

O Seagull é um gerador de tráfego multi-protocolo da HP orientado ao IMS. É *open source* (GPL v2) e orientado aos protocolos IMS (3GPP, TISpan, CableLabs) sendo um complemento ao SIPp. É um gerador de tráfego para testes funcionais, de carga, de durabilidade, stress e desempenho [Seagull 07].

Sipp

O SIPp é uma ferramenta de teste/gerador de tráfego open source para o protocolo SIP. Inclui cenários como User Agent (UAC e UAS) e estabelece múltiplas chamadas com os métodos SIP INVITE e BYE. Também pode implementar fluxos de chamada simples programados em XML e mostra estatísticas dos testes efectuados. Inclui suporte para IPv6, TLS, autenticação SIP, retransmissão UDP, injeção de campos no protocolo, etc. O SIPp pode enviar tráfego RTP com áudio e/ou vídeo [SIPp 07].

4.8 Stacks Diameter

Esta secção apresenta uma lista com os projectos *open source* que implementam *stacks* Diameter.

OpenBloX

O OpenBloX é um *framework* Java Diameter *open source* para implementação de interfaces IMS Diameter conforme a especificação 3GPP e 3GPP2. É mantido pela Traffix Systems.

A *stack* implementa a base do protocolo Diameter descrita pelo IETF RFC 3588 e admite qualquer extensão necessária para suportar as camadas superiores, como especificado nas normas 3GPP como as interfaces Rx, Gx, Ro, Cx, Sh entre outros.

O utilizador pode introduzir melhorias à implementação usando uma API de alto nível orientada a objectos e dispõe de um conjunto de utilitários para o auxiliar [OpenBloX 07].

OpenDiameter

Open Diameter é uma biblioteca C++ que potencia a implementação do protocolo Diameter e suporta Linux, BSD e Microsoft Windows. Usa o *software* ACE para implementar um nível de abstracção relativo aos sistemas operativos suportados [OpenDiameter 07].

JDiameter

JDiameter é uma API e implementação do protocolo Diameter em Java baseada no RFC 3588. A API pode ser usada para construir sistemas comerciais como *stacks*, NASREQ, Extensible Authentication Protocol (EAP) based Authentication, Credit-Control ou mesmo aplicações 3GPP (Sh, Ro, Cx, etc) [JDiameter 07].

4.9 Stacks SIP

A seguinte enumeração de *stacks* SIP apresenta apenas alguns projectos mais conhecidos no caso ser necessário desenvolver um elemento IMS com interfaces SIP de raiz. Como no âmbito desta dissertação, tal não será necessário, fica apenas com referência.

Open Sip Stack

A OpenSIPStack é uma *stack* independente da plataforma que implementa o protocolo SIP segundo a norma RFC 3261.

Jain-sip

A Jain-SIP é uma API em Java para implementação do protocolo de sinalização SIP.

PJSIP

O website PJSIP.ORG oferece uma biblioteca *open source* escrita em C para construção de aplicações VoIP.

OPAL

A OPAL fornece uma *stack* dupla para SIP e H323. O projecto OPAL é o sucessor do OpenH323.

VOCAL

O VOCAL é um projecto *open source* para VoIP em colaboração com a Vovida.org, que é uma unidade da Cisco dedicada a gerir um fórum para *software open source* usado em ambientes de telecomunicações e comunicações de dados.

SipX

SipX é uma família de projectos relacionados com o SIP geridos pela SIPFoundry. O SipX (completamente separado do reSIPProcate) foi uma contribuição feita pela Pingtel. Alias, foi a Pingtel que formou a organização SIPfoundry para contribuir num esforço de criação de *software open source* para telefonia IP, *messaging*, presença e colaboração. A sipXtacklib é uma *stack* SIP orientada a objectos escrita em C++ (LGPL).

reSIPProcate

A reSIPprocate, também gerida pela SIPfoundry e implementa uma biblioteca SIP escrita em C++. É um *spin-off* do projecto VOCAL.

5 Solução proposta

Este capítulo apresenta uma solução para o problema da construção de uma rede IMS baseado em *software open source*. Actualmente existem projectos *open source*, nomeadamente o OpenIMS Core [OSIMS 07], que implementam elementos da rede IMS. No entanto, isso não acontece com todos os elementos. Nesses casos, serão utilizados outros projectos *open source* não IMS *compliant* como base de desenvolvimento e serão apresentadas recomendações, com exemplos de código, para os tornarem normalizados.

5.1 Arquitectura da solução

A solução implementa os elementos core da rede IMS nomeadamente o CSCF, HSS, SIP AS, MRF e as diversas gateways. Esta solução é válida para criação de uma rede IMS segundo as normas da 3GPP *Release 6* e da mais actual 3GPP *Release 7*.

A Figura 10 apresenta a arquitectura da solução proposta de acordo com a 3GPP *Release 7*. Nesta figura estão assinalados os elementos da rede propostos na solução, unidos por código de cores que determina o projecto *open source* usado para o implementar.

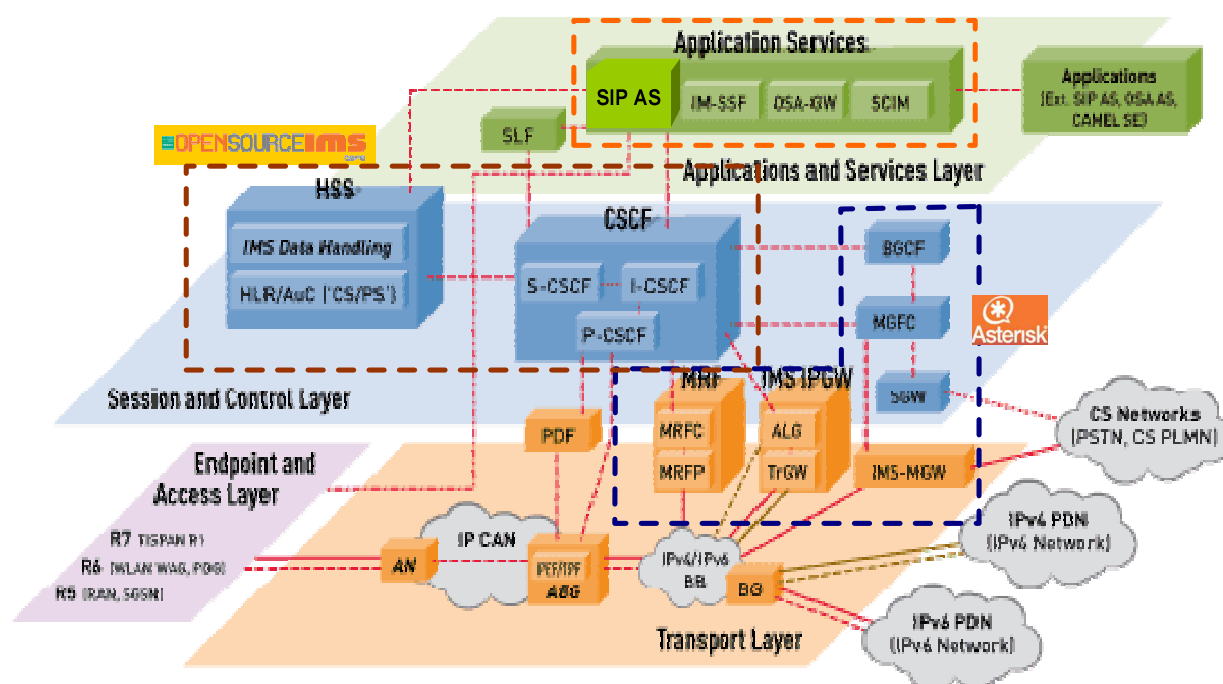


Figura 10 – Solução proposta

A Figura 11 apresenta a solução no âmbito da *Release 6* das normas IMS. As alterações mais significativas são referentes à posição do HSS e constituição dos Application Servers. Nesta figura, também, estão assinalados os elementos de rede propostos pela solução segundo um código de cores.

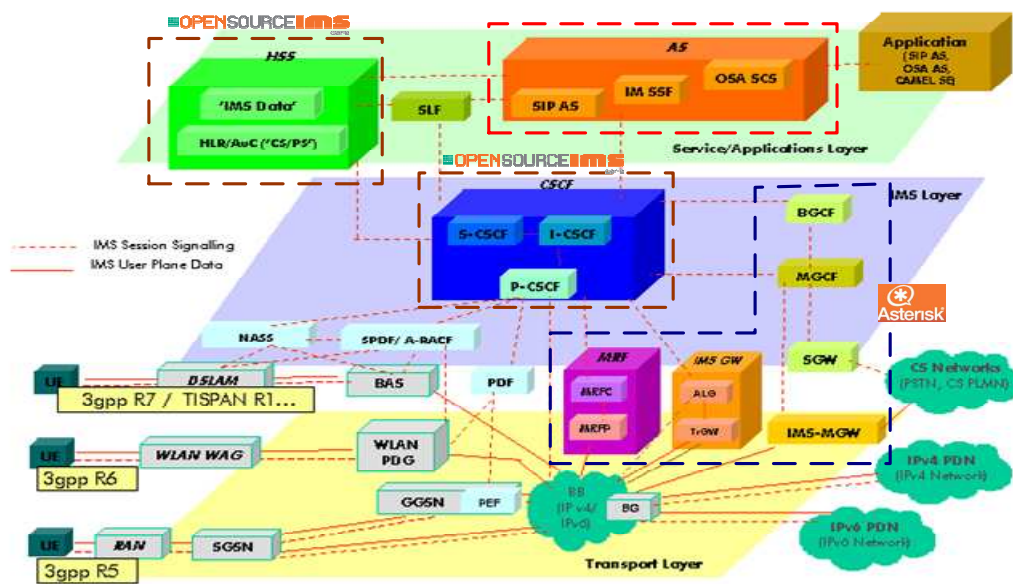


Figura 11 – Solução proposta (Release 6)

Os elementos da camada de transporte, como DSLAMs ou Gateway GPRS Support Node (GGSNs), não são incluídos na proposta porque introduziriam uma complexidade e requisitos de *hardware* que inviabilizava este trabalho. Elementos como Billing Domain e sistemas de O&M também não são apresentados porque são elementos que não estão consagrados directamente nas normas IMS.

5.2 Call Session Control Function

A escolha do CSCF, como a de todos os outros componentes desta solução, está limitada à disponibilidade de projectos *open source* que implementam esse elemento. Neste caso, apenas o projecto OpenIMS Core [OSIMS 07] implementa um CSCF e portanto a escolha está altamente condicionada.

A Figura 1 mostra a arquitectura do OpenIMS Core e as interligações respectivas.

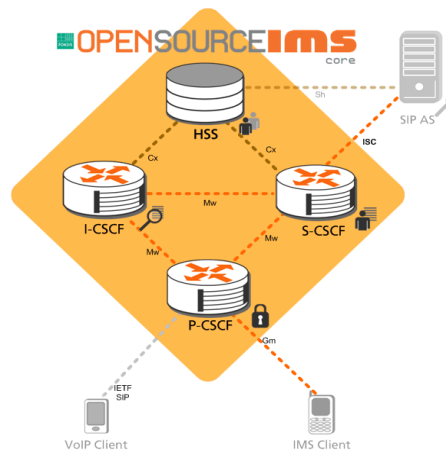


Figura 12 – Arquitectura OpenIMS Core

5.3 Home Subscriber Server

O mesmo acontece com o HSS. Como apenas o openIMS Core [OSIMS 07] implementa um Home Subscriber Server, a decisão da solução fica condicionada a esse projecto.

A Figura 12 mostra um exemplo de integração do conjunto de componentes disponibilizados pelo OpenIMS Core numa rede IMS.

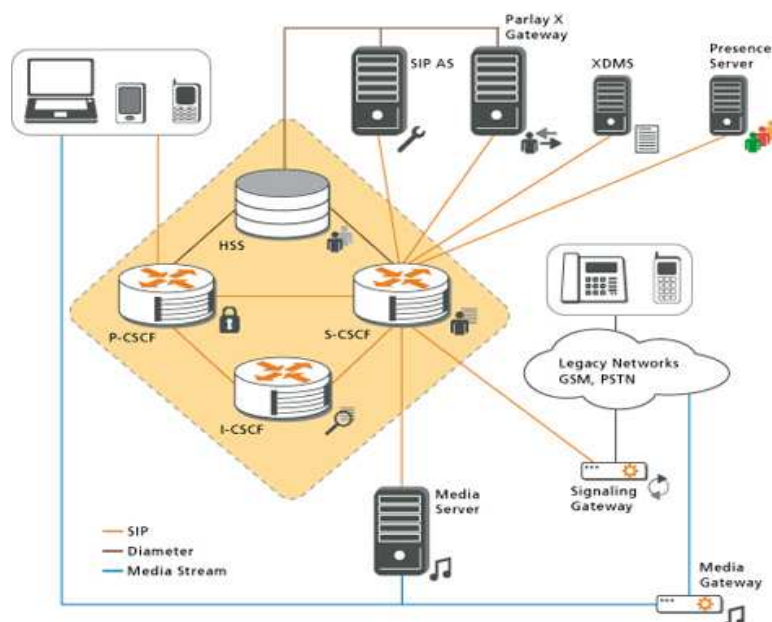


Figura 13 – OpenIMS core numa rede IMS

Na arquitectura da rede IMS, o Subscriber Location Function (SLF) é necessário para mapear endereços do utilizador quando são usados múltiplos HSSs. Como a solução proposta pretende criar uma rede IMS funcional mas com apenas com os elementos básicos e dado que para isso não será necessário implementar sistemas de redundância ou ter múltiplos HSSs, não foi implementado um SLF.

Esta decisão de não implementar o SLF facilita a implementação mas introduz limitações ao nível do número máximo de utilizadores que a solução pode servir dado que apenas existirá um HSS.

5.4 Media Resource Function

Como é possível concluir do estudo apresentado no capítulo 4.3, não há nenhum projecto *open source* que implemente um MRF. Assim, foi necessário usar um *media server* IP referido nesse capítulo (4.3) e efectuar alterações ao código para o tornar IMS *compliant*. Para a solução final foi seleccionado o Asterisk. Esta escolha prende-se com a popularidade deste *software* e pela quantidade e qualidade de informação que existe sobre configurações deste *media server*. Há uma panóplia de projectos que gravitam em torno do Asterisk, desde aplicações de *call centers*, a serviços de *fax-to-email* até *hardware* como telefones IAX e placas telefónicas. O Asterisk é conhecido pela sua flexibilidade, modularidade do código fonte e pelas suas capacidades como *media server* IP. Inclusivamente, a FEUP utiliza o Asterisk para implementar a sua rede VoIP.

A Figura 14 mostra um esquema do papel do Asterisk como MRF.

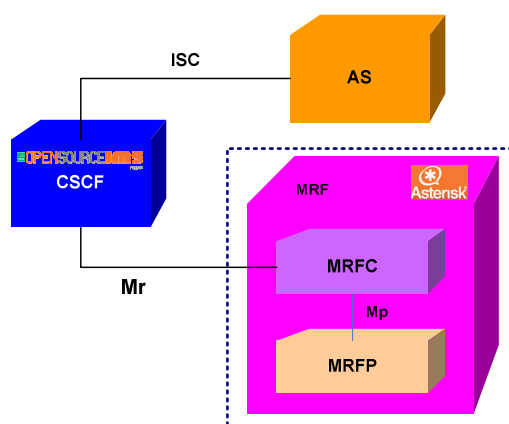


Figura 14 – Arquitectura MRF da solução

Como já foi referido, o Asterisk não é IMS *compliant*. Esse atributo pode ser atingido com algumas configurações e/ou modificações no código fonte.

Alterações a implementar no Asterisk

O primeiro problema que surge na implementação de um MRF é o facto da arquitectura lógica da rede IMS o especificar como um conjunto de MRFP e MRFC (ver Figura 14). Se o intuito do criador da rede IMS for usar estes elementos separadamente, então terá de ser criada uma interface Mp implementando o protocolo H.248. Dado que nenhum dos *media servers* IP investigados implementa ou suporta H.248, a sugestão seria modificar o canal CHAN_MGCP do Asterisk (já que o H.248 é baseado em MGCP) criando um CHAN_MEGACO que suporte o protocolo com o mesmo nome (H.248). Esse suporte pode ser conseguido usando a *stack open source* Erlang/MEGACO [OS MEGACO 07].

Contudo, se os dois elementos lógicos (MRFC e MRFP) forem unirmos fisicamente num MRF, será mais fácil apresentar uma solução MRF com poucas alterações. Teoricamente, se o MRF for apresentado como um todo, a interface Mp não tem visibilidade do exterior e pode ser ignorada. A Cantata Technology lançou um *paper* [Burger 05] em que avalia o requisito da interface Mp e propõe soluções para a fusão do MRFC com o MRFP.

O segundo problema está na especificação da interface SIP Mr que ainda não está concluída (interliga o CSCF com o MRF). O *paper* da Cantata Technology supracitado [Burger 05] também propõe soluções para o interface Mr, nomeadamente implementar o suporte para NETANN no MRFC. O NETANN está normalizado no RFC 4240 e esta extensão ao protocolo SIP permite que outros elementos da rede IMS controlem o MRF através de parâmetros passados no Request-URI do INVITE.

O seguinte exemplo mostra as especificidades de um INVITE para tocar um anúncio de vídeo com o URI `file://fs.example.net//clips/my-intro.dvi`:

```
sip:annc@ms.example.net; \  
  play=file://fs.example.net//clips/my-intro.dvi; \  
  content-type=video/mpeg%3bencode%3d3314M-25/625-50 .
```

No exemplo que se segue, é demonstrado como invocar scripts VXML que definem um fluxo de chamada através do INVITE:

```
sip:dialog@mediaserver.example.net; \  
    voicexml=http://vxmlserver.example.net/cgi-bin/script.vxml .
```

Finalmente, o último exemplo demonstra com criar e aceder a conferências *ad-hoc*:

```
sip:conf=<uniqueIdentifier>@mediaserver.example.net
```

Outras soluções podem ser usadas para implementar a interface Mr nomeadamente o Media Server Control Markup Language (MSCML) definido na norma RFC 4722 ou o Media Sessions Markup Language / Media Object Markup Language (MSML/MOML) que consistem em mensagens SIP INFO com conteúdo XML específico definindo um determinado *call flow* (por exemplo o toque de um anúncio seguido de recolha de dígitos).

Actualmente, os fornecedores de MRFs têm optado sobretudo pelo RFC4240 e por MSCML. Por exemplo, a BEA Dev2Dev recomenda o RFC 4240 para controlo do MRFC [Burke 06] numa rede IMS. Contudo, o MRF da Ericsson suporta o MSCML. Esta decisão é tomada sobretudo para controlar o fluxo de chamada. Se o fornecedor usar o RFP4240 terá que acoplar um interpretador de VXML; no entanto, o MSCML já implementa o controlo do fluxo através de mecanismos existentes na norma RFC 4722.

Na solução proposta será exemplificada a implementação do RFC 4240 no Asterisk na vertente de toque de anúncios multimédia e conferência.

Arquitectura Asterisk

O Asterisk foi construído de forma muito flexível de modo a que qualquer utilizador pudesse acrescentar suporte para protocolos específicos ou criar aplicações que satisfizesse as suas necessidades. Para isso, há duas componentes muito importantes: os canais e as aplicações.

Os canais implementam os protocolos. Actualmente, existem canais SIP, MGCP, IAX, H.323, Bluetooth, CAPI, SCCP, entre outros. São *loadable modules* que podem ser acrescentados sem necessidade de recompilação do Asterisk.

As aplicações conjugadas com o DialPlan (do `extensions.conf`) implementam fluxos de chamada. O exemplo seguinte demonstra como criar uma extensão 2000 que usa a aplicação Answer para atender a chamada (tipicamente envia um 200 OK) e em seguida usa a aplicação Playback para tocar o anúncio “welcome”. A chave `exten` é usada para definir a extensão, a ordem de utilização da função no fluxo, a função a utilizar e os parâmetros, respectivamente.

```
-----  
extensions.conf
```

```
[default]  
exten => 2000,1,Answer  
exten => 2000,2, Playback (welcome)  
-----
```

A Figura 15 mostra a arquitetura do Asterisk como MRF. Como foi mencionado anteriormente, o Asterisk usa dois conceitos importantes: as aplicações e os canais. No enquadramento das redes IMS, as aplicações e o *Dial Plan* serão responsáveis por implementar o suporte ao RFC 4240 ou MSCML assumindo o papel de Media Resource Function Controller (MRFC). Os canais associados aos codecs implementados no Asterisk serão responsáveis por implementar as funções de *media* como tocar anúncios ou fazer *transcoding*.

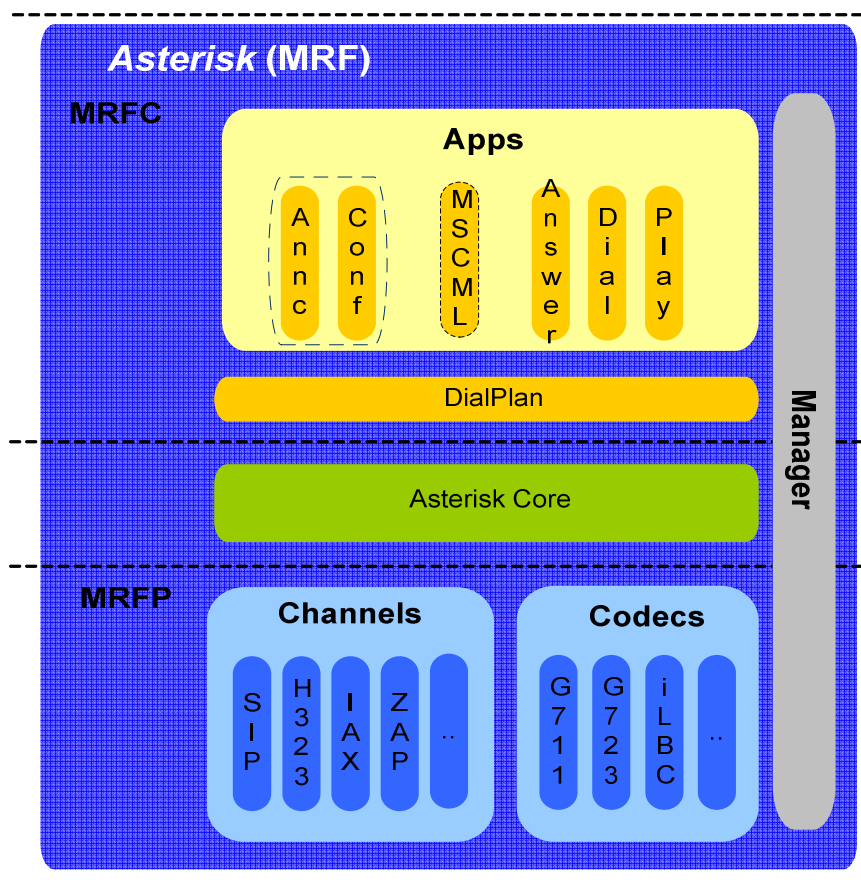


Figura 15 – Arquitetura MRF Asterisk

Portanto, as aplicações podem ser considerados pequenos *plugins* para executar tarefas. Contudo, uma só aplicação pode executar todo o fluxo de chamada. Essa será a abordagem seguida para implementar o RFC 4240 em que será necessário criar uma aplicação que implemente o tratamento do INVITE. Para isso será aproveitado o código da aplicação `app_playback` para criar uma `app_netannnc`.

Implementação do suporte RFC4240

A versão do Asterisk usada na solução proposta é a 1.4.17. Na altura de elaboração desta Dissertação, a versão 1.6 está em fase RC1. As alterações estão a **bold**.

Para o toque de anúncios (`annnc`) é necessário alterar o `CHAN_SIP` para que envie os parâmetros extra do Request-URI para as aplicações. Assim, será adicionada à estrutura partilhada com as aplicações um campo com o URI completo denominado **`complete_ruri`**:

```

static struct sip_pvt {
    //NEW PART
    char complete_ruri[AST_MAX_PARAMS];
    //END -----
    [...]
    static int handle_request(struct sip_pvt *p, struct sip_request *req, struct sockaddr_in *sin, int
        *recount, int *nounlock)
    {
        [...]
        cmd = req->rIPart1;

        //NEW PART – This section fills complete_ruri field with the complete request-URI
        if (!strcasecmp(cmd, "INVITE"))
        {
            strcpy(p-> complete_ruri, req->rIPart2);
        }
        //END -----

        e = req->rIPart2;
    }
}

```

O seguinte código mostra as alterações a efectuar sobre a app_playback. As linhas a **bold** indicam as secções que devem ser substituídas:

```

static int playback_exec(struct ast_channel *chan, void *data)
{
    [...]
    ast_stopstream(chan);
    while (!res && (front = strsep(&back, "&"))) {
        if (option_say)
            [...]
        }
    }
}

```

Por:

```

static int netannc_exec(struct ast_channel *chan, void *data)
{
    [...]

    char *strpos, *play = NULL, *a, request-uri[AST_MAX_PARAMS];
    int times = 0, repeat = 0, delay = 0, duration = -1;

    //NEW PART – This section fills will decode the request-URI and get the play, repeat, delay and
    // duration tags
    strcpy(request-uri, chan->complete_ruri);
    strpos = request-uri;
    while(*strpos != '\0')
    {
        if ((a = strchr(strpos, ",")) != NULL) *a = '\0';
        if (strncasecmp(strpos, "play=", 5) == 0)
        {
            play = strpos + 5;
        }
    }
}

```



```

}
if (strncasecmp(strpos, "repeat=forever", 14) == 0)
{
    repeat = 100; //Limit to avoid Denial Of Service
}
else if (strncasecmp(strpos, "repeat=", 7) == 0)
{
    repeat = atoi(strpos + 7);
}
if (strncasecmp(strpos, "delay=", 6) == 0)
{
    delay = atoi(strpos + 7);
}
if (strncasecmp(strpos, "duration=", 9) == 0)
{
    duration = atoi(strpos + 7);
}
}
//NEW PART – This section plays the announcement according to the parameters specified
while(times <= repeat)
{
    res = ast_streamfile(chan, front, chan->language);
    if (!res) {
        res = ast_waitstream(chan, "");
        ast_stopstream(chan);
    } else {
        ast_log(LOG_WARNING, "ast_streamfile failed on %s for %s\n", chan-
>name, (char *)data);
        if (priority_jump || ast_opt_priority_jumping)
            ast_goto_if_exists(chan, chan->context, chan->exten, chan-
>priority + 101);
        res = 0;
        mres = 1;
    }
    ++ times;
    if (times <= repeat && delay) usleep(delay * 1000);
}
}

```

Para suportar o RFC 4240 na vertente de conferência, não há necessidade de alterações no código fonte, apenas a configuração do *dial plan* no ficheiro *extensions.conf*. Essa configuração limita-se a retirar o prefixo “conf=” do INVITE usando a função `=${CUT(EXTEN|=|1)}` e a enviar a identificação da conferência para o MeetMe:

extensions.conf

```

[default]
exten => _conf.,1,Set(confID=${CUT(EXTEN|=|1)}) ;retira o conf ID
exten => _conf.,2, Meetme($confID)
exten => _conf.,3,Hangup

```

Implementação do suporte RFC 3325 e 3455

Para tornar o Asterisk IMS *compliant*, há duas formas de suportar as normas RFC 3325 e 3455.

A primeira forma é através alterações no código fonte do canal SIP Asterisk (CHAN_SIP):

```
static struct sip_pvt {
    //NEW PART
    char complete_ruri[AST_MAX_PARAMS];
    //END -----

    // RFC 3325
    char p_asserted_id[AST_MAX_EXTENSION]; // P-Asserted-Identity
    char p_preferred_id[AST_MAX_EXTENSION]; // P-Preferred-Identity
    char p_privacy_id[AST_MAX_EXTENSION]; // Privacy
    // RFC 3455
    char p_called_party_id[AST_MAX_EXTENSION]; // P-Called-Party-ID
    char p_associated_uri[AST_MAX_EXTENSION]; // P-Associated-URI
    char p_visited_network_id[AST_MAX_EXTENSION]; // P-Visited-Network-ID
    char p_access_network_info[AST_MAX_EXTENSION]; // P-Access-Network-Info
    char p_charging_vector[AST_MAX_EXTENSION]; // P-Charging-Vector
    char p_charging_function_address[AST_MAX_EXTENSION]; // P-Charging-Function-
    Addresses
    [...]
}
```

A função reqprep faz a criação dos vários pedidos SIP no chan_sip onde vão serão introduzidos os SIP *headers* normalizados nos RFC 3325 e RFC 3455. Desta forma, não só os INVITES mas todas as mensagens SIP e nos campos associados.

```
/*! \brief Initialize a SIP request message (not the initial one in a dialog) */
static int reqprep(struct sip_request *req, struct sip_pvt *p, int sipmethod, int seqno, int
    newbranch)
{
    [...]
    // RFC 3325
    if (!last_strlen_zero(p->p_asserted_id)) add_header(req, "P-Asserted-Identity", p->
        p_asserted_id);
    if (!last_strlen_zero(p->p_preferred_id)) add_header(req, "P-Preferred-Identity", p->
        p_preferred_id);
    if (!last_strlen_zero(p->p_privacy_id)) add_header(req, "Privacy", p->p_privacy_id);
    // RFC 3455
    if (!last_strlen_zero(p->p_called_party_id)) add_header(req, "P-Called-Party-ID", p->
        p_called_party_id);
    if (!last_strlen_zero(p->p_associated_uri)) add_header(req, "P-Associated-URI", p->
        p_associated_uri);
    if (!last_strlen_zero(p->p_visited_network_id)) add_header(req, "P-Visited-Network-ID", p->
        p_visited_network_id);
    if (!last_strlen_zero(p->p_access_network_info)) add_header(req, "P-Access-Network-
        Info", p->p_access_network_info);
    if (!last_strlen_zero(p->p_charging_vector)) add_header(req, "P-Charging-Vector", p->
        p_charging_vector);
    if (!last_strlen_zero(p->p_charging_function_address)) add_header(req, "P-Charging-
        Function-Addresses", p->p_charging_function_address);

    // End
}
```

Em seguida, é necessário obter os campos dos RFC 3325 e 3455 dos INVITES que são enviados para o Asterisk. Para isso, é necessário adicionar a chamada a uma função `get_ims_headers()` na `handle_request_invite()`:

```
static int handle_request_invite(struct sip_pvt *p, struct sip_request *req, int debug, int seqno,
    struct sockaddr_in *sin, int *recount, char *e, int *nounlock)
{
    [...]
    gotdest = get_destination(p, NULL); /* Get destination right away */
    get_rdnis(p, NULL); /* Get redirect information */

    // NEW PART -----
    get_ims_headers(p, NULL);
    // END -----
    extract_uri(p, req); /* Get the Contact URI */
    build_contact(p); /* Build our contact header */
    [...]
}
```

Finalmente, é necessário definir a função `get_ims_headers()`:

```
/*--- get_ims_headers: get RFC 3325 and RFC 3455 headers ---*/
static int get_ims_headers(struct sip_pvt *p, struct sip_request *oreq)
{
    char tmp[256], *c, *a;
    struct sip_request *req;

    req = oreq;
    if (!req)
        req = &p->initreq;

    // Get all the header
    ast_copy_string(tmp, get_header(req, "P-Asserted-Identity"), sizeof(tmp));
    if (!last_strlen_zero(tmp))
        ast_copy_string(p->p_asserted_id, tmp, sizeof(p->p_asserted_id));

    ast_copy_string(tmp, get_header(req, "P-Preferred-Identity"), sizeof(tmp));
    if (!last_strlen_zero(tmp))
        ast_copy_string(p->p_preferred_id, tmp, sizeof(p->p_preferred_id));

    ast_copy_string(tmp, get_header(req, "Privacy"), sizeof(tmp));
    if (!last_strlen_zero(tmp))
        ast_copy_string(p->p_privacy_id, tmp, sizeof(p->p_privacy_id));

    ast_copy_string(tmp, get_header(req, "P-Called-Party-ID"), sizeof(tmp));
    if (!last_strlen_zero(tmp))
        ast_copy_string(p->p_asserted_id, tmp, sizeof(p->p_asserted_id));

    ast_copy_string(tmp, get_header(req, "P-Associated-URI"), sizeof(tmp));
    if (!last_strlen_zero(tmp))
        ast_copy_string(p->p_associated_uri, tmp, sizeof(p->p_associated_uri));

    return 0;

    [...]
}
```

A outra forma é apenas um *workaround* que recorre a alterações no “dial plan” do `extensions.conf`. Este *workaround* consiste em adicionar os headers SIP especificados na norma e é baseado num exemplo existente no *wiki* voip-info.org para integração do Asterisk com o Acme SBC [Voip-Info 07]:

Envio dos headers RFC 3325 nas chamadas SIP

`extensions.conf`

[default]

```
exten => _9.,1,SIPAddHeader(P-Preferred-Identity: <sip:+1${CALLERID(num)}\;user=phone>)
exten => _9.,1,SIPAddHeader(P-Asserted-Identity: <sip:+1${CALLERID(num)}\;user=phone>)
exten => _9.,n,SIPAddHeader(Privacy: id)
exten => _9.,n,Set(CALLERID(num)=)
exten => _9.,n,Set(CALLERID(name)=Anonymous)
exten => _9.,n,Dial(SIP/>${EXTEN:1}@sipcarrier)
exten => _9.,n,Hangup()
```

Envio dos headers RFC 3455 nas chamadas SIP

`extensions.conf`

[default]

```
exten => _9.,1,SIPAddHeader(P-Called-Party-ID: <sip:+1${CALLERID(num)}\;user=phone>)
exten => _9.,n,Set(CALLERID(num)=)
exten => _9.,n,Set(CALLERID(name)=Anonymous)
exten => _9.,n,Dial(SIP/>${EXTEN:1}@sipcarrier)
exten => _9.,n,Hangup()
```

Envio do header Sip Diversion

`extensions.conf`

[macro-diversion-header]

```
exten => s,1,SIPAddHeader(Diversion: <tel:>${ARG1}>\;reason=user=busy\;screen=no\;privacy=off)
```

Esta configuração não é tão intrusiva como alterar código fonte mas para “DialPlans” muito extensos aumenta-lhes significativamente a complexidade.

5.5 SIP Application Server

O SIP AS dependerá das preferências de cada programador. A solução pode envolver uma união de *servlets* SIP usando o SIPSEE e Jain-Slee usando o Mobicents. Não há uma escolha mais acertada em relação ao SIP AS, e segundo as normas IMS, podem existir diversos *application servers* numa rede. Como o Asterisk já implementa audioconferência com funções de *floor control* seria interessante considerar a aplicação de conferência MeetMe como um SIP AS para este tipo de serviços. Já para soluções de vídeoconferência sugere-se a utilização do MCU MediaMixer da sip.fontventa.com [FontVenta 07] que pode ser integrado com o GlassFish e Sailfin SIP AS. O MCU foi desenvolvido com uma interface Web que pode ser integrada neste SIP AS dotando-o de mecanismos de configuração e *floor control*.

Segundo as normas, o MRF é controlado por uma interface SIP denominada Mr. Contudo, como já foi referido previamente, esta interface ainda não está definida. Os fornecedores de equipamentos o RFC 4240 ou MSCML para controlar o MRF, contudo, se eventualmente se desejasse uma solução rápida e que se desvia ligeiramente do mercado o AS da Mobicents já integra um *resource adapter* (RA) para Asterisk.

O Asterisk RA controla o Asterisk através da interface Manager, no entanto, também está planeado fazê-lo através de FastAGI. O RA faz a tradução de eventos ou respostas do Manager Asterisk e envia-as para o Service Logic Execution Environment (SLEE). A interface SBB do RA consiste numa função (*sendAction*) que é usada para enviar os pedidos para o Asterisk. A adopção deste método elimina a necessidade de efectuar as alterações sugeridas na secção 5.4 [AsteriskRA 07].

A Figura 16 mostra a arquitectura Mobicents com os diversos *resource adapters*, incluindo Asterisk RA.

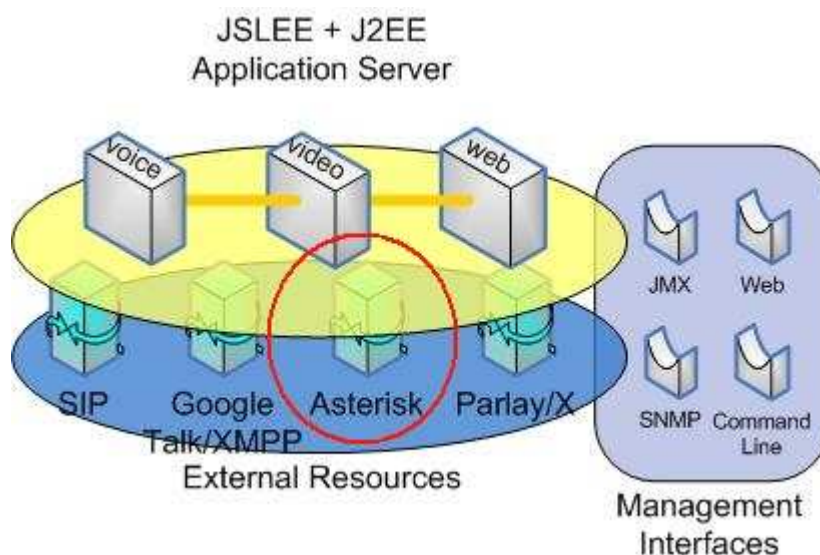


Figura 16 – Arquitectura Mobicents

Para a prova de conceito será usada a combinação GlassFish com Sailfin já que será demonstrado um serviço de vídeoconferência.

5.6 User Equipment

Não há nenhuma escolha final para o User Equipment, isto é, qualquer um pode ser usado. Alguns dos UEs listados têm especificidades que podem determinar a sua utilização em determinado serviço. Por exemplo, o CounterPath X-Lite é muito interessante para aplicações de vídeo e conta com uma interface de utilizador apelativa.

A Figura 17 mostra a interface do CounterPath X-Lite.



Figura 17 – Interface gráfica X-Lite

Contudo, o CounterPath X-Lite não é *open source* apesar de ser gratuito. Se se desejar aprofundar o tema do *open source* com aplicações de vídeo, seria mais recomendável usar o IMS Communicator.

A Figura 18 mostra a interface gráfica do IMS Communicator.



Figura 18 – Interface gráfica do IMS Communicator

Numa rede com IMS-IPGW, até os UEs não IMS *compliant* podem ser usados dependendo da preferência de cada utilizador.

5.7 Gateways

Como pode ser visto no trabalho de pesquisa do estado da arte no *software open source*, não há projectos que disponibilizem *software* que realize as funções de BGCF e MGCF. Mesmo para MGW e SGW será necessário usar gateways não *compliant* com o IMS e proceder a modificações. Assim, se unirmos todos os componentes de gateways lógicos num só componente físico como é feito com o MRF, podemos usar o Asterisk com placas telefónicas como uma solução abrangente e capaz de desempenhar os papéis das diversas gateways.

A Figura 19 mostra os elementos que constituem as gateways IMS que são propostos nesta secção.

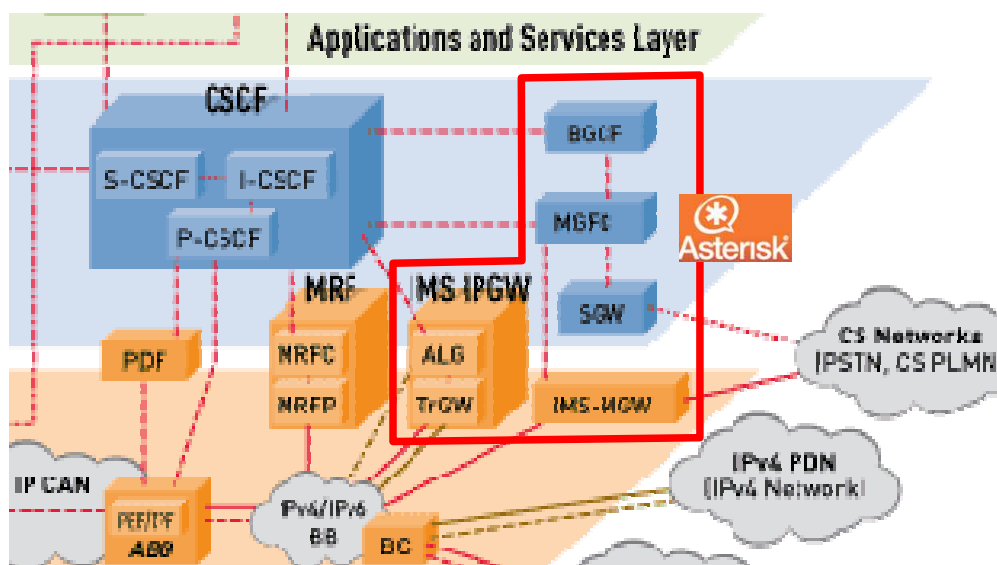


Figura 19 – Gateways IMS

Breakout Gateway Control Function

O BGCF determina se uma chamada deve ser encaminhada para a PSTN e, nesse caso, que MGCF se deve usar para o fazer. O *dial plan* do Asterisk pode ser alterado para determinar que se um URI for telefónico, o deve enviar para a PSTN directamente ou para outra rede IMS via SIP (onde ocorrerá o *breakout*). O seguinte exemplo demonstra um encaminhamento de URIs de nove dígitos começados por 2 ou 9 (segundo o plano de numeração português) para uma placa telefónica com suporte CAPI:

 extensions.conf

[default]

```
; chamadas para qualquer número da pstn ( com prefixo 02 )
exten => _2XXXXXXXX,1,Dial(CAPI/g1/${EXTEN},30,r)
; destino Rede móvel : 9x xxx xx xx
exten => _9XXXXXXXX,1,Dial(CAPI/g1/${EXTEN},30,r)
```

O suporte para URIs telefônicas (tel:) segundo o RFC 2806 não é suportado pelo Asterisk. Contudo, para implementar a gateway, será necessário ser compatível com esta norma e alterar o Asterisk de modo a que todos os pedidos com URI tel: sejam encaminhados para as placas telefônicas. O *patch* para alterar esse comportamento está disponível no fórum da Digium [Digium Bugs 07].

O Asterisk possui um mecanismo de inclusão do parâmetro “;user=phone” no URI sempre que um número telefônico é detectado no mesmo. Todos os URIs que contenham a extensão “;user=phone” também devem ser encaminhados para a PSTN.

Media Gateway Control Function

A função MGCF também será desempenhada pelo Asterisk. Neste caso será seguido o exemplo do MRF em que não há separação física entre MRFC e MRFP; portanto, a ligação entre MGCF e MGW não usará o H.248. Esta abordagem não permite interligação entre o MGCF proposto e uma MGW de outro fornecedor, limitando a compatibilidade da solução. Se a solução proposta satisfizesse esse requisito seria necessário implementar essa interface.

IMS Media Gateway

A Media Gateway (MGW) actua como uma unidade de tradução multimédia entre redes como PSTN, NGN, 2G, 2.5G e 3G. Como a MGW se liga a diferentes redes, uma das suas principais funções é fazer a conversão entre diferentes tipos de transmissão e codificação. Por exemplo, numa *gateway* SIP-3G é comum ser feito *transcoding* entre o *codec* de áudio e de vídeo da chamada 3G e o *codec* de áudio e vídeo da chamada SIP, já no caso de uma *gateway* SIP-PSTN, a *gateway* é responsável por encapsular o áudio G.711 da PSTN em tramas RTP e vice-versa. Funções de multimédia como cancelamento de eco, DTMF, e geração de tons estão também localizados na MGW.

A forma mais simples de criar uma MGW é usar o Asterisk com placas telefônicas compatíveis. O Asterisk implementa um canal CAPI que lhe permite fazer a interface com inúmeros fornecedores de placas telefônicas. Por outro lado, os drivers Zaptel permitem ligar o Asterisk a placas Digium ou Sangoma. O tipo de placas a usar depende do número de chamadas simultâneas que deve suportar e do tipo de tecnologia a usar podendo variar entre uma placa com um acesso analógico até placas com quatro acessos primários.

Uma MGW Asterisk vai, essencialmente, permitir que chamadas SIP/RTP da rede IMS possam ser encaminhadas para a PSTN ou PLMN.

Em baixo, é sugerida uma configuração do canal SIP (sip.conf) e do DialPlan (extensions.conf) que permitem implementar a MGW para a PSTN [IMS Dev 07 B].

```
-----  
sip.conf
```

```
[general]  
context=ims-outgoing ; Default context for incoming calls  
allowguest=yes ; Allow guest calls  
-----
```

extensions.conf

```
[ims-outgoing]
;Use zaptel to dial the phone number contained in the SIP INVITE URI
exten => _X.,1,Dial(Zap/1/${EXTEN})
exten => _X.,2,Hangup
```

```
[from-pstn]
; Answer PSTN calls with a default recording
exten => _X.,1,Wait(2)
exten => _X.,2,Answer
exten => _X.,3,Playback(tt-monkeys)
exten => _X.,4,Hangup
```

Signalling Gateway

Na solução proposta a SGW estará fisicamente incluída na MGW e SGW. Para implementar uma SGW IMS-DSS1, basta usar o Asterisk com placas telefónicas compatíveis. Essas placas podem conter acessos básicos ou primários.

Para implementar uma SGW para redes 3G, é sugerida a utilização de módulos criados por Sérgio Murillo da sip.fontventa.com [FontVenta 07]. Esses módulos para o Asterisk implementam um canal H.324m e o *codec* AMR. Para implementar uma SGW IMS-SS7, é sugerida a utilização da libSS7 desenvolvida para o Asterisk. Para além disso, é necessário usar *hardware* compatível com o Asterisk para efectuar a interface telefónica.

IMS IP Gateway

O IMS IP Gateway era suportado directamente pelo CSCF do OpenIMS Core mas, infelizmente, foi descontinuado. Assim, dado que não há nenhum projecto *open source* que possa mapear directamente este elemento podemos sugerir qualquer projecto que implemente um SIP REGISTRAR. O OpenSER ou SER são exemplos possíveis.

Para simplificar a arquitectura, dado que o Asterisk foi modificado para se tornar IMS *compliant* na solução final e pode desempenhar esta função, será o seleccionado para desempenhar estas funções. Este *software* possui uma interface gráfica onde é possível aprovisionar os clientes que se registarão lá. Esse aprovisionamento pode também ser feito numa base de dados mySQL.

A Figura 20 mostra a interface Web do Asterisk onde podem ser registrados os clientes.

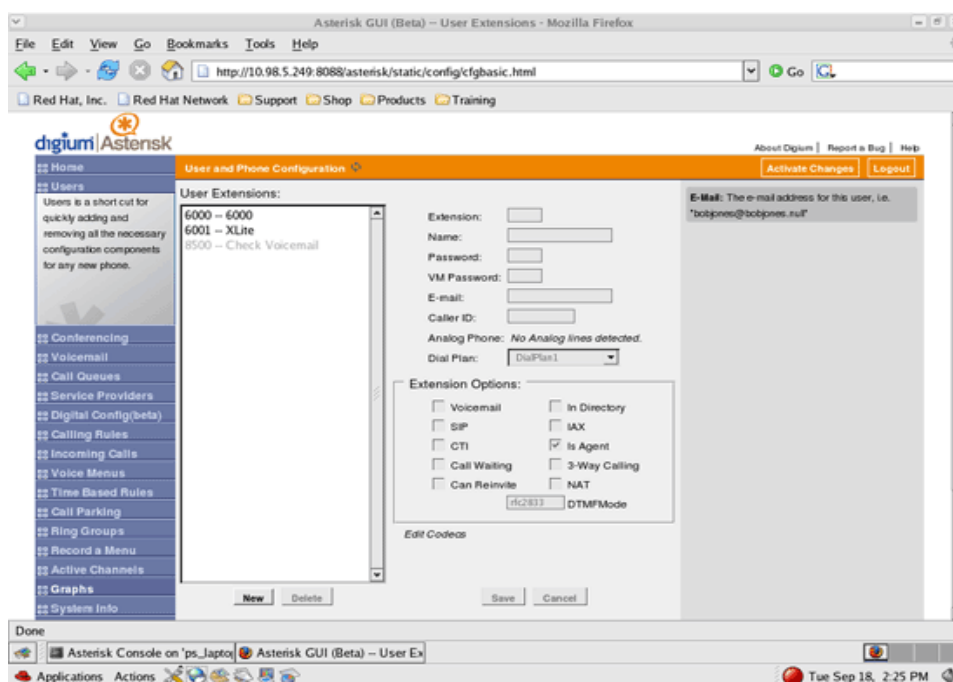


Figura 20 – Interface Web Asterisk GUI

Para clientes H.323 seria indicado usar o YATE como IMS IPGW já que implementa um H.323 Gatekeeper.

5.8 Prova de conceito

Com esta prova de conceito pretendeu-se demonstrar a viabilidade da solução proposta para o problema analisado nesta dissertação. Para isso, foi construída uma rede IMS com os principais elementos CSCF, HSS, SIP AS, MRF e UE em que foi implementado um serviço de videoconferência. Na prova de conceito não foram usadas gateways telefónicas dada a necessidade de aquisição de *hardware*.

Arquitectura

A arquitectura da rede usada na prova de conceito pretende ser uma versão simplificada da solução apresentada no capítulo 5.

A Figura 21 demonstra a arquitectura da prova de conceito.

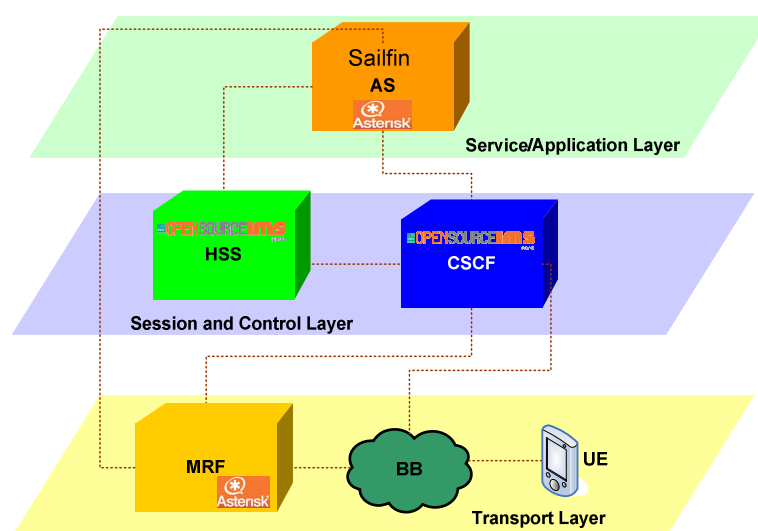


Figura 21 – Arquitectura da prova de conceito

Instalação dos componentes

Para esta prova de conceito foram instalados num PC com o sistema operativo RedHat AS4 update4, o CSCF e HSS do OpenIMS Core, o SIP AS GlassFish com Sailfin [SailFin 07] e o MRF Asterisk [Asterisk 07].

A instalação do CSCF e HSS é bastante intuitiva, recorrendo-se a um utilitário Java denominado ANT para compilar todo o código. Os passos de instalação estão detalhados no *Installation Guide* na página do projecto [OSIMS 07].

As páginas dos projectos GlassFish e Sailfin disponibilizam um *Quick Start Guide* muito detalhado para a instalação dos mesmos. Em relação ao Asterisk, a instalação pode ser feita a partir de RPM o que simplifica bastante o processo.

Para implementar o *mixer* dos *streams* de vídeo na conferência será necessário recorrer a um MCU (Multipoint Control Unit) *open source* denominado Media Mixer da sip.fontventa.com [FontVenta 07].

Configuração dos componentes

A primeira configuração necessária após a instalação dos diversos elementos é o provisionamento dos clientes no HSS criando uma subscrição, uma identidade pública e uma privada. Para isso, é usada a interface Web do FHoSS da Fokus mostrado na Figura 22 – Configuração Web do HSS.

The screenshot shows the FHoSS web interface for HSS configuration. The page has a yellow header with the FOKUS logo and navigation tabs: HOME, USER PROFILES, SERVICE PROFILES, NETWORK INFRASTRUCTURE, INFO. The main content area is divided into several sections:

- Private Identity:** Fields for Private ID* (testuser@open-ims.org), IMSI, S-CSCF Name (sip:scscf.open-ims.org:6060).
- Security:** Authentication Scheme (Digest-AKAv1-MD5), Algorithm (AKAv1-MD5), Show char values (radio buttons for yes/no), Secret Key (74657374757365720000000000000000), AMF (0000), Operator ID (00000000000000000000000000000000), Sequence Number (1000000000) with a Reset button.
- Charging Info Set:** Charging Info Set (Charging_Set_1) with [GUSS] [Refresh] buttons.
- Public Identity:** A table with columns Public ID and Action. It lists two entries: sip:bestuserzwei@open-ims.org and sip:testuser@open-ims.org.
- Roaming Networks:** A section for Roaming Network Identifiers with entries for open-ims.org and umts-at-fokus.de.

Figura 22 – Configuração Web do HSS

Em seguida, através da interface Web do MCU (mcuWeb), pode ser configurada e gerida a videoconferência. É possível intervir sobre a conferência silenciando (Mute/UnMute) e removendo utilizadores (*floor control*). Também é possível configurar o mosaico da conferência e determinar a posição dos utilizadores na mesma. A Figura 23 mostra as diferentes configurações de mosaico possíveis do MCU Media Mixer.

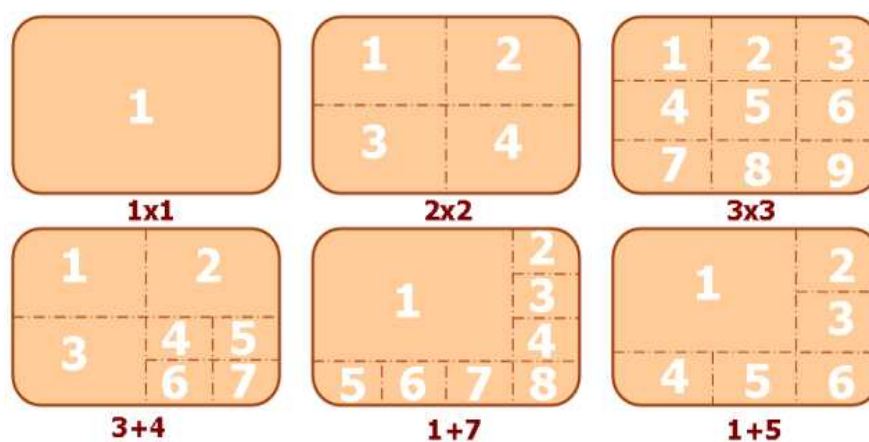


Figura 23 – Exemplos de configurações do mosaico do MCU

A Figura 24 mostra um *screenshot* da interface Web de gestão de conferências mcuWeb.

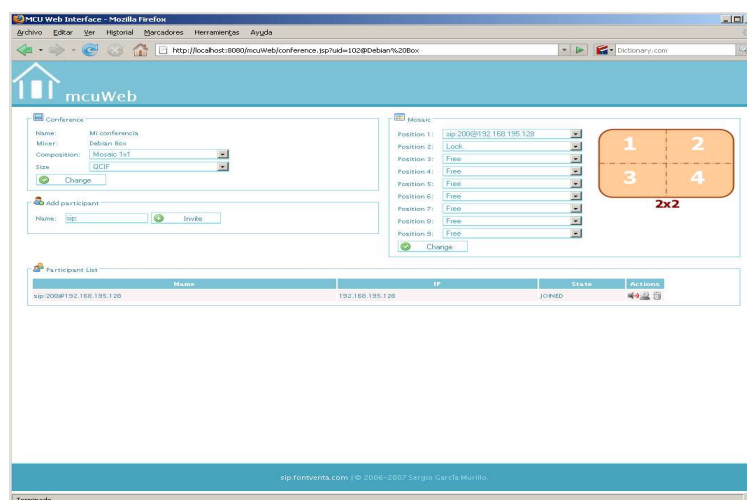


Figura 24 – Screenshot da página de configuração mcuWeb

Demonstração

A Figura 25 mostra um *screenshot* do User Equipment CounterPath X-Lite, durante uma demonstração do serviço videoconferência IMS implementado plataforma da prova de conceito, com quatro participantes activos usando a rede IMS da prova de conceito.



Figura 25 – X-Lite User Equipment

Esta demonstração prova que é possível implementar um serviço complexo como o de videoconferência com soluções exclusivamente *open source* e que satisfazem as normas IMS. Mesmo fora do âmbito IMS, seria fácil implementar este serviço recorrendo ao Asterisk como *media server IP*.

Relativamente ao serviço em demonstração, a página do Sailfin AS disponibiliza *servlets* SIP que podem facilmente ser integradas para implementar outros serviços como o Click-to-Dial.

6 Conclusões

6.1 Análise

No mundo das telecomunicações há três factores que se tem revelado muito relevantes para a escolha do tema abordado nesta dissertação e que procura propor uma solução *open source* para a implementação de uma rede IMS com o intuito de reduzir custos. Em primeiro lugar, os operadores de telecomunicações têm revelado um crescente interesse por redes IMS e no fornecimento de equipamentos que apresentem conformidade com essas especificações.

Em segundo lugar, apesar do crescente interesse, as operadoras de telecomunicações ainda não recuperaram da experiência negativa com redes móveis de terceira geração. Este estigma certamente introduzirá relutância em qualquer investimento avultado de implantação de uma rede de próxima geração. Alias, dada a actual conjuntura económica em Portugal, dificilmente algum operador móvel ou fixo fará um investimento avultado sem não estiver convencido de que a solução terá grandes hipóteses de criar valor e trazer mais valias.

Em terceiro lugar, o nível de aceitação de utilização de *software open source* por parte dos operadores tem vindo a aumentar. Por exemplo, sistemas operativos como o Linux são muito respeitados pela sua estabilidade. Também sistemas de monitorização como o Nagios [Nagios 07] ou o Multi Router Traffic Grapher (MRTG) [MRTG 07] são amplamente usados pelos operadores.

Tendo em conta estas premissas, a utilização de *software open source* no auxílio ao desenvolvimento das redes IMS trará certamente algumas vantagens. Contudo, no início deste estudo era um pouco difícil saber se, de facto, seria possível construir uma rede IMS a partir de *software open source* mas, na globalidade, o resultado é bastante animador.

Uma questão relevante que surge nesta análise é a de o OpenIMS Core ser o único *player open source* a desenvolver soluções a nível do core IMS. Este projecto do Fraunhofer Institute tem vindo a trazer notoriedade ao mesmo tornando-o num parceiro obrigatório de quem pretenda criar uma rede IMS *open source*. Por outro lado, o facto deste projecto existir veio permitir que esta dissertação pudesse apresentar uma solução exequível para o problema da construção de uma rede IMS *open source*. No entanto, é seguro afirmar que a solução final não será interessante como um todo a nível comercial no imediato dado que tem lacunas a nível do *charging* que envolveriam desenvolvimentos avultados. O próprio projecto FOKUS IMS afirma que o seu produto tem um cariz académico.

Outra questão interessante é a das tendências do mundo *open source* e da disponibilidade de *software* específico criado neste âmbito. Ainda há um vazio na área das redes IMS *open source* como se pode verificar no estudo do estado da arte em que se demonstrou, por exemplo, que não há um único projecto a trabalhar sobre um MRF. No entanto, a lista de projectos que desenvolvem *media servers* IP é a mais extensa. Isto porque no que concerne o *software open source*, este depende de uma comunidade de desenvolvimento. Se houver um interesse particular por um tipo de *software*, este será mais facilmente encontrado nos projectos *open source*. Vejamos o exemplo dos *media servers* IP. Recentemente tem havido uma grande evolução das redes de banda larga IP que têm potenciado as comunicações de baixo custo VoIP. Esta conjuntura tem levado empresas e alguns particulares a instalar PBX

IP *open source* em PCs antigos para as comunicações internas eliminando custos na aquisição de PBXs comerciais. Eventualmente, esse PBX IP poderá fazer a interligação com a rede PSTN ou PLMN tornando-se o único sistema de telecomunicações da empresa. Os fornecedores de placas telefônicas anteviram essas necessidades e criaram os PBX IP *open source* que estas empresas usam para que interaja com o seu *hardware*. Actualmente os PBX IP como o Asterisk são um fenómeno de popularidade. A Faculdade de Engenharia da Universidade do Porto é um paradigma de instituições que usam este tipo de *software*.

Por outro lado, institutos como o Fraunhofer Institute anteviram a necessidade de componentes *open source* para as redes IMS e quiseram-se tornar uma referência das redes IMS criando um projecto que por um lado os torna conhecidos mundialmente e por outro lhes dá experiência na área revertendo na melhoria da formação dos seus alunos.

Finalmente, há uma questão que diz respeito à interface Mp do MRF e por ser amplamente discutida pelo mercado merece referência: há uma oposição dos fornecedores de equipamentos à utilização do H.248. Num *draft* da IETF [Burger 06] é referido que o facto de ser um protocolo de baixo nível e a relutância que o mercado tem em o adoptar têm dificultado a implementação das normas IMS relativamente à separação lógica do MRFC e MRFP ou MGCP e MGW.

6.2 Avaliação

Quando os operadores de telecomunicações encontram uma nova necessidade têm de enfrentar a decisão entre desenvolver ou comprar. Como as aplicações são muito específicas e não têm qualquer relevância para um utilizador individual, os projectos *open source* relacionados com estas áreas são marginais. Veja-se o exemplo do *charging* no IMS. Como é uma área relevante apenas para os operadores e com interfaces específicas, não há muitos projectos que se dediquem à sua implementação. Por outro lado, o *software* dedicado aos operadores é criado por empresas específicas que dominam essa área e dificilmente fornecerão o seu código à comunidade excepto se se enquadrar numa qualquer estratégia comercial.

Se um operador optar pelos elementos *open source* listados no capítulo 4 não só terá de implementar interfaces como as de *charging* como terá de seguir continuamente as actualizações das normas. E, independentemente, do que se possa dizer, como a especificação do IMS é feita pelos próprios fornecedores e operadores de telecomunicações, as normas só vão evoluído conforme as necessidades dos próprios assim que testam os seus produtos e identificam falhas ou inconsistências nas mesmas.

Para um fornecedor de equipamentos IMS, o *open source* pode ser uma boa base para implementar os seus produtos partindo de uma *stack SIP open source*, ou de um *media server* IP para criação de um MRF ou mesmo de uma *stack Diameter* para criar todos os interfaces necessários. Um dos benefícios apontados pelos criadores do IMS é o do operador poder adquirir produtos a diferentes fornecedores para criação de uma rede já eles são normalizados e interoperáveis. Mas para isso é essencial que os fornecedores de equipamentos sigam as normas e estabeleçam parcerias para testar os seus produtos com os de outras empresas.

6.3 Conclusões

Há duas abordagens relativamente distintas ao problema de otimização de custos em redes IMS com produtos *open source*: a implementação de uma rede IMS *open source* completa e a construção de um componente IMS baseado em *open source*.

Em relação à primeira, parece ser difícil que alguém crie uma solução comercial completa para uma rede IMS *open source* dada a sua dimensão e complexidade. Toda a parte de *charging* parece esquecida nos projectos *open source* o que invalida uma instalação directa sem desenvolvimentos. Contudo, a segunda abordagem é muito interessante e já tem sido usada por alguns fornecedores que usam, por exemplo, *stacks SIP open source* para construir os seus produtos IMS *compliant*. Eventualmente, estes fornecedores podem até construir uma rede IMS completa nas suas instalações a partir de componentes *open source* para testes de interoperabilidade com as suas soluções comerciais.

Em suma, as principais desvantagens da abordagem *open source* sobre uma versão puramente comercial são as limitações no desempenho, nas interfaces de taxaço e falta de suporte ao *software*.

Em primeiro lugar, se olharmos para esta solução do ponto de vista de um operador, há muito desenvolvimento a fazer, sobretudo na parte mais importante para o operador, o *charging*. Apesar do OpenIMS Core e os *application servers* implementam as respectivas interfaces de *charging*, é necessário desenvolvê-las para o MRF, a BGCF, MGCF assim como desenvolver os elementos CDF, CGF, OCS e BD. Claro que estes últimos, sobretudo os elementos existentes no Billing Domain também conhecidos por Billing System, podem ser adquiridos comercialmente criando um misto de *software open source* e *software* comercial.

Da perspectiva dos operadores de telecomunicações, um outro requisito muito importante é o *uptime* dos sistemas e o seu desempenho. Conceitos como o AdvancedTCA, os servidores NEPS *compliant*, a redundância e alta disponibilidade são muito valiosos. Aqui entra um dos principais entraves ao *open source* que é o suporte ao *software*. Se um operador detectar um bug no código, terá que aguardar que a comunidade o solucione ou terá que ele próprio o resolver o que é raro. O mesmo acontece com o desempenho do *software*. A comunidade *open source* não costuma ter requisitos de desempenho (transacções por segundo) semelhantes aos de um operador nem tão pouco terem taxas de *uptime* tão exigentes pelo que estes requisitos são um pouco negligenciados, tornando o *open source* ainda menos apelativo para os operadores. Contudo, muitos dos sistemas *open source* actuais têm uma vertente comercial precisamente para estas questões de suporte ao *software*. Um exemplo de projectos que apresentam diferenças notórias em termos de desempenho entre a versão *open source* e a versão com suporte comercial é a do Bladeware.

Ainda na perspectiva dos operadores, o mercado das telecomunicações, sobretudo com a internet e a voz sobre IP, tem-se tornado num mercado global e altamente competitivo. Portanto, a oportunidade de reduzir custos de implementação com *software open source* apesar de todas as possíveis limitações de desempenho ou suporte, torna-se cada vez mais apelativa. Essa tendência de redução de custos pode ser observada na introdução de novas plataformas em que, por vezes, os operadores preferem usar servidores convencionais do que plataformas AdvancedTCA que apesar de serem mais dispendiosas para pequenas escalas satisfazem todos os requisitos de redundância e alta disponibilidade.

Uma limitação que esta solução apresenta é da escalabilidade. Por exemplo, ao limitar o HSS a um elemento removendo a necessidade de um SLF, é limitado também o número de utilizadores que a solução pode enquadrar. Também não são incluídos sistemas de balanceamento de carga e tolerância a falhas dada a já grande complexidade da rede IMS e por sair do âmbito deste trabalho. Contudo, projectos como o OpenSer implementam balanceamento de carga e tolerância a falhas para elementos como o MRF usando Asterisk. Exemplos dessas configurações estão disponíveis no seu WIKI e podem ser facilmente implementados se for necessário [OpenSer 07 B]. Por outro lado, o SLF é um elemento relativamente simples de implementar com o recurso a uma *stack Diameter open source*.

Da perspectiva dos fornecedores de componentes IMS, a utilização de *open source* para desenvolvimento de um produto é meio caminho andado para a solução final. Claro que tem que resolver potenciais problemas de desempenho e terão, possivelmente, que introduzir interfaces que não existam originalmente como o H.248 ou interfaces Ro e RF no MRF.

Apesar das limitações mencionadas, há vantagens inequívocas sobre as soluções comerciais e que contrapõe estes obstáculos:

A primeira vantagem é, sem dúvida, o preço. Alguém que deseje implementar uma rede IMS básica com gateways para a PSTN necessita apenas de um PC e uma placa telefónica. Para uma implementação não comercial, é difícil apresentar desvantagens que combatam este argumento.

A construção de uma rede IMS *open source* pode ser excelente para um laboratório de testes para uma empresa que esteja no mundo IMS comercial ou até mesmo para uma instituição de ensino como uma faculdade ou instituto.

Imagine-se uma empresa fornecedora de equipamentos IMS que deseja testar a interoperabilidade do seu produto. Há três soluções possíveis, adquirir uma rede IMS completa (o que não faz qualquer sentido), efectuar parcerias com outros fornecedores para realizarem testes do seu componente com componentes de terceiros ou, finalmente, construir uma rede baseada na solução proposta enquadrada no IMS. Num cenário de testes de interoperabilidade, esta última solução é muito apelativa. Já no caso de uma universidade, pode ser criado um projecto numa cadeira de telecomunicações em que cada grupo deva implementar o seu elemento IMS a partir de *open source* e no final interligar todos os elementos numa rede. Este exercício trará certamente um conhecimento mais alargado sobre a rede em estudo.

Uma outra vantagem é a possibilidade de customização dos elementos da rede. Ao ter acesso ao código fonte de cada elemento, há a possibilidade de introduzir novos interfaces ou melhorias que num sistema comercial, possibilidade esta que o cliente não poderá fazer. Em sistemas comerciais, o fornecedor pode não estar disposto a efectuar as alterações solicitadas pelo cliente sem contrapartidas económicas.

Por outro lado, uma solução pequena como a que foi proposta pode ser muito interessante. Recentemente, uma empresa alemã denominada 4S newcom GmbH apresentou um IP PBX com capacidade para 250 utilizadores e 30 chamadas simultâneas que cabe no mais pequeno dos iPod Shuffle®. Esse PBX não corre no iPod mas sim num Apple mini mas é carregado a partir do iPod. Este é um exemplo do que se poderia fazer com a solução proposta. Imagine-se que se adquire um router Wi-Fi com sistema operativo Linux e, em seguida, se instala toda a

solução proposta. Seria possível ter uma rede IMS completa numa caixa de 3,5 polegadas semelhante aos discos rígidos externos. Só a portabilidade e efeito visual têm um valor publicitário inequívoco em demonstrações a clientes.

No cômputo geral, esta dissertação faz uma compilação dos projectos *open source* existentes no mercado analisando as suas vantagens e desvantagens. A partir desse estudo, são propostos os projectos mais funcionais e coerentes para propor uma solução final em que estão consagrados elementos oriundos de projectos *open source*, nomeadamente o CSCF, HSS e SIP AS, e elementos não implementados por projectos *open source*, como o MRF, IMS IPGW, IMS MGW, SGW, BGCF e MGCF. Para facilitar esse desenvolvimento destes últimos, foi aproveitado código fonte de *software open source* não conforme com as normas IMS. Na descrição da solução são incluídas porções de código que permitem dar uma noção das alterações efectuadas mas não refletem na totalidade os desenvolvimentos efectuados para concretizar a rede IMS proposta.

A solução apresentada é totalmente funcional se como pode concluir com a prova de conceito e responde na totalidade ao problema de optimização de custos na implementação de uma rede de próxima geração IMS. Se o principal objectivo de uma entidade for reduzir custos, esta solução sem dúvida terá relevância no seu projecto.

6.4 Lições aprendidas

Uma das dificuldades maiores na investigação sobre as redes IMS tem a ver com a constante evolução das normas. Apesar do core se manter o mesmo, cada *Release* introduz novas tecnologias e algumas alterações na configuração da rede. Se isto aconteceu durante a elaboração deste documento, imagine-se os inconvenientes que trará para os fornecedores de equipamentos IMS que competem no mercado. A *Release 7* parece ser a versão mais coerente e com mais futuro mas ainda não está fechada o que levanta sérias dificuldades a quem compete nestas áreas de desenvolvimento. A sua data de encerramento estava prevista para o ano de 2007 mas já foi ultrapassada.

Os operadores normalmente pedem aos seus fornecedores equipamentos NEPs *compliant* ou AdvancedTCA (ATCA). Funcionam num mundo a -48v mas, no entanto, sempre que lhes é proposta uma plataforma querem apenas um demonstrador. Nestes casos, o *open source* pode fornecer uma plataforma de demonstração muito eficiente a nível de custo para futuramente colocar uma solução puramente comercial e com todos os requisitos de disponibilidade.

Da investigação feita do mercado, os operadores ainda não têm redes 100% IMS. Têm apenas fragmentos dessas redes. Fragmentos esses que incluem os *softswitches* da *Release 4* e MRFs da *Release 6*. Aparentemente, o que acontece é que essas redes vão sendo implementadas conforme vai surgindo a necessidade de novos serviços. Os operadores simplesmente não substituem a rede para implementar serviços que já estejam a disponibilizar com a sua arquitectura actual já que não teriam forma de justificar um novo investimento numa solução já implementada.

A utilização de MRFs específicos por aplicação parece fazer sentido. Por exemplo, ao olhar para a prova de conceito apresentada no capítulo 5.8 com um serviço de vídeoconferência usando o Asterisk, facilmente se poderá concluir que a capacidade desse MRF é finita e será esgotada com poucos utilizadores deixando de ter recursos para outros serviços. Como um

operador deve sempre possuir uma reserva de recursos, um MRF partilhado pode arruinar a experiência dos serviços. Este ponto contraria um dos benefícios apresentados no IMS que é a possibilidade de partilha do MRF por vários serviços para reaproveitamento dos recursos mas em serviços exigentes como a videoconferência faz sentido ter um MRF específico.

6.5 Trabalho futuro

Após este trabalho, torna-se evidente que há lacunas no universo *open source* dedicado ao IMS que seria interessante colmatar. Um exemplo disso é o facto de não haver nenhum projecto *open source* que desenvolva um MRF, uma BGCF ou um MGCF.

Para fazer evoluir esta solução seria necessário desenvolver os elementos supracitados com todas as interfaces normalizadas, sobretudo, as H.248 e de *charging*.

Poderia ser interessante para um estabelecimento de ensino, como a Faculdade de Engenharia da Universidade do Porto, criar um projecto que suprisse esta necessidade. Poderia iniciar um projecto para implementar um MRF *fully compliant* com as normas IMS aproveitando um *media server IP open source*. Esse projecto certamente traria mais visibilidade á instituição e mais experiencia neste domínio.

Na área do ensino seria interessante criar um projecto prático em que cada grupo implementasse o seu elemento IMS *open source* e o integrasse com os restantes.

7 Referências e bibliografia

7.1 Referências bibliográficas

- [Americas 04] **3G Americas**, (2005), *IP Multimedia Subsystem Overview and applications*, White Paper, 3G Americas
- [Bhaskaran -] **Bhaskaran, Sridhar, Motiwala, Yusuf**, (-), *Rapid IMS Development Using Highly Redundant Telecom Middleware*, IntelliNet Technologies, USA
- [BMC 06] **BMC Software**, (2006), *A New Paradigm for IMS Database Reorganization*, BMC Software
- [Burger 05] **Burger, Eric, Redmill, Guy** (2005), *Media Services in the IMS: Evolution for Innovation*, Cantata technology.
- [Burger 06] **Burger, E.**, (2006), “*Media Server Control Language and Protocol Thoughts*”, draft-burger-mscl-thoughts-01, capítulo 4.1
- [Burke 06] **Burke, David, O'Flanagan, Darragh**, (2006), “*An IMS Application Example Based on SIP Servlets and VoiceXML*”, Dev2Dev, <http://dev2dev.bea.com/lpt/a/513>
- [Chiruvolu 05] **Chiruvolu, Girish et al.**, (2005), *Evolution of Media Servers for IMS/NGN Services*, Technology White Paper, Alcatel
- [Croslin 05] **Croslin, David**, (2005), *Delivering Unique Services That Meet Customer Needs*, The IP Multimedia Subsystem: IMS Architecture, White Paper, MCI
- [Daniel 05] **Daniel, Chris**, (2005), *IMS Reconciliation White Paper*, Leapstone, MultiService Forum
- [Denis 05] **Attal, Denis**, (2005), *IMS: Internet Age Telephony*, Technology White Paper, Alcatel
- [Dialogic 07 A] **Dialogic**, (2007), *A Developing Higher Density Solutions with Dialogic® Host Media Processing Software - A Strategy for Load Balancing and Fault Handling*, Dialogic Corporation, Canada

- [Dialogic 07 B] **Dialogic**, (2007), *Enhanced Service Delivery — IP Multimedia Subsystems and AdvancedTCA*, White Paper, Dialogic Corporation, Canada
- [Handa 05] **Handa, Arun**, (2005), *The 3GPP IP Multimedia Subsystem - Building Value with Next Generation Services and Applications*, White Paper, Outubro 2005, InteliNet Technologies
- [Handa 06 A] **Handa, Arun**, (2005), *Fixed Mobile Convergence – FMC Today!*, White Paper, Outubro 2005, InteliNet Technologies
- [Handa 06 B] **Handa, Arun**, (2006), *Beyond Sip - Diameter*, White Paper, Maio 2006, InteliNet Technologies
- [Handa 06 C] **Handa, Arun**, (2006), *Importance of True Interoperability if IMS is to be Deployed in 2007*, Fierce Wireless Friday Feature Published May 12, 2006
- [Intel 06] **Intel**, (2006), *Accelerating Deployment of IP Multimedia Subsystems (IMS) - IMS Telco Server Proof of Concept*, Application Note, Intel, USA
- [IP Unity 05] **IP Unity**, (2005), *IP Multimedia Subsystem – IMS, Technical White*, IP Unity
- [Leig 06] **Leih, Dan, Halliday, Dave**, (06), *Introduction to IMS - Standards, protocols, architecture and functions of the IP Multimedia Subsystem*, Computer Speech and Language, Motorola Inc
- [Magedanz 06] **Magedanz, Prof. Dr. Thomas**, (2006), *IP Multimedia System (IMS) - Principles, Architecture and Applications*, Technical University of Berlin / Fraunhofer FOKUS
- [Matos 93] **Matos, Manuel A.**, (1993), *Normas para apresentação de dissertações – Bases Essenciais*, Porto
- [NMS 06 A] **NMS Communications**, (2006), *The Value of Mobile Client Applications in SIP-Based IMS Deployments*, White Paper, NMS Communications, USA
- [NMS 06 B] **NMS Communications**, (2006), *IMS and the Challenges of New Platforms*, White Paper, NMS Communications

- [NMS 06 C] **NMS Communications**, (2006), *NMS and Open Source Projects for Telecom Applications*, NMS Telecom Innovators News - October 2006, NMS Communications
- [NMS 07] **NMS Communications**, (2007), “*SS7 in an IMS World*”, Telecom Innovators News - Jan 2007, NMS Communications, USA
- [Nokia 04] **Nokia**, (2004), *IP multimedia – a new era in communications*, White Paper, Nokia Corporation
- [Nortel 01] **Nortel**, (2001), *Delivering converged quad-play services with IPTV and IMS*, White Paper, Nortel
- [Turner 07 A] **Turner, Brough**, (2007), *IMS-FAQs*, NMS Communications
- [Turner 07 B] **Turner, Brough**, (2007), *Lessons Learned Implementing IMS*, NMS Communications
- [Verrington 06] **Verrington, Pat**, (2006), *IMS-Lite: Small Can Be Beautiful Too*, IntelliNet Technologies, USA
- [Wonac 07] **Wonak, Dan**, (Abril 2007), *Why not open source? – A Whitepaper*, IntelliNet Technologies.

7.2 Referências Web

- [3GPP 07] **Third Generation Partnership Project**, (2007), www.3gpp.org
- [3GPP Specs 07] **3GPP Specifications**, (2007), www.3gpp.org/specs/specs.htm
- [Asterisk 07] **Asterisk**, (2007), <http://www.asterisk.org/>
- [AsteriskRA 07] **Asterisk RA for Mobicents** - <http://wiki.java.net/bin/view/Communications/MobicentsAsteriskRA>
- [Bayonne 07] **Bayonne**, (2007), <http://www.gnu.org/software/bayonne/>
- [CallWeaver 07] **CallWeaver**, (2007), <http://www.openpbx.org>
- [Confidence 07] **CONFIDENCE**, (2007), “*CONFerencing IMS-enabled Architecture for Next-generation Communication Experience*”, <http://confiance.sourceforge.net/>

- [CounterPath 07] **CounterPath X-Lite**, (2007), <http://www.counterpath.com/x-lite.html>
- [Dev2dev 07] **Dev2Dev**, (2007), “*Understanding the IMS Charging Architecture*”, <http://dev2dev.bea.com/pub/a/2007/07/IMC-charging-architecture.html>
- [Digium Bugs 07] **Digium**, (2007), “*Patch Asterisk tel: URI - bugs.digium.com*”, http://bugs.digium.com/bug_view_page.php?bug_id=0002244
- [Digium 07] **Digium**, (2007), <http://www.digium.com>
- [FontVenta 07] **FontVenta**, (2007), <http://sip.fontventa.com>
- [FreeSwitch 07] **FreeSWITCH**, (2007), <http://www.freeswitch.org>
- [GNU Lics 07] **GNU Licenses**, (2007), www.gnu.org/licenses/licenses.html
- [Grupo PT 07] **Grupo Portugal Telecom**, (2007), <http://www.telecom.pt>
- [IEC 07] **International Engineering Consortium**, (2007), “*International Intelligent Network*” http://www.iec.org/online/tutorials/intern_in/topic02.html
- [IETF 07] **Internet Engineering Task Force**, (2007), www.ietf.org
- [IETF RFC 07] **IETF RFC Page**, (2007), www.ietf.org/rfc.html
- [IMS Com 07] **IMS Communicator**, (2007), <http://imscommunicator.berlios.de>
- [IMS Dev 07 A] **IMS Developer**, (2007), *IMS Clients*, <http://www.ims-developer.org/content/view/28/41/>
- [IMS Dev 07 B] **IMS Developer**, (2007), *Media GW*, <http://www.ims-developer.org/content/view/44/56/>
- [JDiameter 07] **JDiameter**, (2007), <https://jdiameter.dev.java.net/>
- [Mobicents 07] **Mobicents**, (2007), <http://www.mobicents.org>
- [MRTG 07] **The Multi Router Traffic Grapher**, (2007), <http://oss.oetiker.ch/mrtg/>

- [MS Windows 07] **Microsoft Windows Products**, (2007), <http://www.microsoft.com/windows/>
- [Nagios 07] **Nagios**, (2007), <http://www.nagios.org>
- [OpenBloX 07] **OpenBloX**, (2007), <http://www.traffixsystems.com/>
- [OpenDiameter 07] **Open Diameter**, (2007), <http://www.opendiameter.org/>
- [OpenIC 07] **Fokus Open IMS Playground**, (2007), *Open IC*, http://www.fokus.fraunhofer.de/ims/components/open_ic.php
- [OpenSer 07 A] **OpenSER**, (2007), <http://www.openser.org/>
- [OpenSer 07 B] **OpenSer**, (2007), “*Asterisk Load balancing and High Availability with OpenSER*”, <http://www.openser.org/dokuwiki/doku.php/asterisk:load-balancing-and-ha>
- [OpenVXI 07] **OpenVXI**, (2007), <http://www.openvxi.org/>
- [OS MEGACO 07] **Open Source MEGACO**, (2007), <http://erlang.stacken.kth.se/project/megaco/>
- [OSIMS 07] **OpenIMS Core**, (2007), <http://www.openimscore.org/>
- [Pika 07] **PIKA Technologies**, (2007), *PIKA Asterisk Market Study Data*, <http://www.pikatechnologies.com/>, Julho-Agosto 2007, PIKA Technologies Inc
- [PT Inovação 07] **Portugal Telecom Inovação, SA**, (2007), www.ptinovacao.pt
- [Radvision 07] **RADVISION**, (2007), *RADVISION IMS Developer Suite - Advanced building blocks for developing IMS-compliant products*, RADVISION, Ltd
- [SailFin 07] **SailFin**, (2007), <https://sailfin.dev.java.net/>
- [Seagull 07] **Seagull**, (2007), <http://gull.sourceforge.net>
- [SEMS 07] **SIP Express Media Server**, (2007), <http://www.iptel.org/sems>
- [SER 07] **SIP Express Router**, (2007), <http://www.iptel.org/ser/>

- [SIP SEE 07] **Fokus Open IMS Playground**, (2007), *SIPSEE*, http://www.fokus.fraunhofer.de/bereichsseiten/testbeds/ims_playground/components/sipsee.php?lang=en
- [SIPp 07] **SIPp**, (2007), <http://sipp.sourceforge.net/>
- [SipX 07] **SIPfoundry SipX**, (2007), <http://www.sipfoundry.org/sipx>
- [Tiegre 07] **Tiegre, Greger V.**, (2007), *Thoughts from Greger on iptel.org, SIP Express Router (SER), SEMS, SERWeb, and SIP in general*, <http://sipstuff.blogspot.com/2007/02/competing-eco-systems-ims-ser-asterisk.html>
- [TISPAN 07] **Telecoms & Internet converged Services & Protocols for Advanced Networks**, (2007), www.etsi.org/tispan
- [Tompros 07] **Tompros, L, Spyros (Keletron Ltd)**, (2007), *NGN networks; A new enabling technology or just a network integration solution?*, VITAL project, <http://www.ist-vital.eu>
- [UPF 07] **Universitat Pompeu Fabra**, (2007), http://iua-share.upf.es/wikis/seminaris/index.php/Conversational_Services_Now_and_in_the_Future_%E2%80%93_NGN,_IMS_and_Beyond
- [UTC IMS 07] **UTC IMS Client**, (2007), <http://uctimsclient.berlios.de/>
- [Voip-Info 07] **Voip-Info.org**, (2007), “*Asterisk config extensions.conf*”, <http://www.voip-info.org/tiki-index.php?page=Asterisk%20config%20extensions.conf>
- [Vovida 07] **Vovida**, (2007), <http://www.vovida.org>
- [WeSIP 07] **WeSIP**, (2007), <http://www.wesip.com/>
- [YATE 07] **Yet Another Telephony Engine**, (2007), <http://yate.null.ro/>

7.3 Normas 3GPP

Lista de normas 3GPP que mencionam ou estão relacionadas com o IMS [3GPP Specs 07]:

- [TR 21.905] Vocabulary for 3GPP Specifications
- [TS 22.066] Support of Mobile Number Portability (MNP); Stage 1

-
- [TS 22.101] Service Aspects; Service Principles
 - [TS 22.141] Presence Service; Stage 1
 - [TS 22.228] Service requirements for the IP multimedia core network subsystem; Stage 1
 - [TS 22.250] IMS Group Management; Stage 1
 - [TS 22.340] IMS Messaging; Stage 1
 - [TR 22.800] IMS Subscription and access scenarios
 - [TS 23.002] Network Architecture
 - [TS 23.003] Numbering, Addressing and Identification
 - [TS 23.008] Organization of Subscriber Data
 - [TS 23.107] Quality of Service (QoS) principles
 - [TS 23.125] Overall high level functionality and architecture impacts of flow based charging; Stage 2
 - [TS 23.141] Presence Service; Architecture and functional description; Stage 2
 - [TS 23.167] IMS emergency sessions
 - [TS 23.207] End-to-end QoS concept and architecture
 - [TS 23.218] IMS session handling; IM call model; Stage 2
 - [TS 23.221] Architectural Requirements
 - [TS 23.228] IP Multimedia Subsystem (IMS); stage 2
 - [TS 23.234] WLAN interworking
 - [TS 23.271] Location Services (LCS); Functional description; Stage 2
 - [TS 23.278] Customized Applications for Mobile network Enhanced Logic (CAMEL) - IMS interworking; Stage 2
 - [TR 23.864] Commonality and interoperability between IMS core networks
 - [TR 23.867] IMS emergency sessions
 - [TR 23.917] Dynamic policy control enhancements for end-to-end QoS, Feasibility study
 - [TR 23.979] 3GPP enablers for Push-to-Talk over Cellular (PoC) services; Stage 2
 - [TR 23.981] Interworking aspects and migration scenarios for IPv4-based IMS implementations (early IMS)
 - [TS 24.141] Presence Service using the IMS Core Network subsystem; Stage 3

- [TS 24.147] Conferencing using the IMS Core Network subsystem
- [TS 24.228] Signalling flows for the IMS call control based on SIP and SDP; Stage 3
- [TS 24.229] IMS call control protocol based on SIP and SDP; Stage 3
- [TS 24.247] Messaging using the IMS Core Network subsystem; Stage 3
- [TS 26.235] Packet switched conversational multimedia applications; Default codecs
- [TS 29.207] Policy control over Go interface
- [TS 29.208] End-to-end QoS signalling flows
- [TS 29.209] Policy control over Gq interface
- [TS 29.228] IMS Cx and Dx interfaces: signalling flows and message contents
- [TS 29.229] IMS Cx and Dx interfaces based on the Diameter protocol; Protocol details
- [TS 29.278] CAMEL Application Part (CAP) specification for IMS
- [TS 29.328] IMS Sh interface : signalling flows and message content
- [TS 29.329] IMS Sh interface based on the Diameter protocol; Protocol details
- [TR 29.962] Signalling interworking between the 3GPP SIP profile and non-3GPP SIP usage
- [TS 31.103] Characteristics of the IMS Identity Module (ISIM) application
- [TS 32.240] Telecommunication management; Charging management; Charging architecture and Principles
- [TS 32.260] Telecommunication management; Charging management; IMS charging
- [TS 32.299] Telecommunication management; Charging management; Diameter charging applications
- [TS 32.421] Telecommunication management; Subscriber and equipment trace: Trace concepts and requirements
- [TS 33.102] 3G security; Security architecture
- [TS 33.108] 3G security; Handover interface for Lawful Interception (LI)
- [TS 33.141] Presence service; security
- [TS 33.203] 3G security; Access security for IP-based services
- [TS 33.210] 3G security; Network Domain Security (NDS); IP network layer security
- [TR 33.978] Security aspects of early IP Multimedia Subsystem (IMS)

7.4 Normas IETF

Lista de normas IETF que mencionam ou estão relacionadas com o IMS [IETF RFC 07]:

- [RFC 2327] Session Description Protocol (SDP)
- [RFC 2748] Common Open Policy Server protocol (COPS)
- [RFC 2782] A DNS RR for specifying the location of services (SRV)
- [RFC 2806] URLs for telephone calls (TEL)
- [RFC 2915] The naming authority pointer DNS resource record (NAPTR)
- [RFC 2916] E.164 number and DNS
- [RFC 3087] Control of Service Context using SIP Request-URI
- [RFC 3261] Session Initiation Protocol (SIP)
- [RFC 3262] Reliability of provisional responses (PRACK)
- [RFC 3263] Locating SIP servers
- [RFC 3264] An offer/answer model with the Session Description Protocol
- [RFC 3265] SIP-Specific Event Notification
- [RFC 3310] HTTP Digest Authentication using Authentication and Key Agreement (AKA)
- [RFC 3311] Update method
- [RFC 3312] Integration of resource management and SIP
- [RFC 3319] DHCPv6 options for SIP servers
- [RFC 3320] Signalling compression (SigComp)
- [RFC 3323] A privacy mechanism for SIP
- [RFC 3324] Short term requirements for network asserted identity
- [RFC 3325] Private extensions to SIP for asserted identity within trusted networks
- [RFC 3326] The reason header field
- [RFC 3327] Extension header field for registering non-adjacent contacts (path header)
- [RFC 3329] Security mechanism agreement
- [RFC 3420] Internet Media Type message/sipfrag
- [RFC 3428] SIP Extension for Instant Messaging
- [RFC 3455] Private header extensions to SIP for 3GPP

-
- [RFC 3485] SIP and SDP static dictionary for signaling compression
 - [RFC 3515] The SIP REFER method
 - [RFC 3550] Real-time Transport Protocol (RTP)
 - [RFC 3574] Transition Scenarios for 3GPP Networks
 - [RFC 3588] Diameter base protocol
 - [RFC 3589] Diameter command codes for 3GPP release 5 (informational)
 - [RFC 3608] Extension header field for service route discovery during registration
 - [RFC 3665] SIP Basic Call Flow Examples
 - [RFC 3680] SIP event package for registrations
 - [RFC 3725] Best current practices for Third Party Call Control (3pcc) in SIP
 - [RFC 3824] Using E164 numbers with SIP
 - [RFC 3840] Indicating user Agent Capabilities in SIP
 - [RFC 3841] Caller preferences for SIP
 - [RFC 3842] SIP event package for message waiting indication and summary
 - [RFC 3856] SIP event package for presence
 - [RFC 3857] SIP event template-package for watcher info
 - [RFC 3858] XML based format for watcher information
 - [RFC 3891] The SIP Replaces Header
 - [RFC 3903] SIP Extension for Event State Publication
 - [RFC 3911] The SIP Join Header
 - [RFC 4028] Session timers in SIP
 - [RFC 4235] An INVITE-Initiated dialog event package for SIP
 - [RFC 4475] Session Initiation Protocol (SIP) Torture Test Messages