Universidade de Aveiro
2009
Departamento de Electrónica, Telecomunicações e Informática

Luís Filipe Carvalho da Silva

Plataformas de Serviços em Redes de Próxima Geração (IMS)

Universidade de Aveiro
2009
Departamento de Electrónica, Telecomunicações e Informática

Luís Filipe Carvalho da Silva

Plataformas de Serviços em Redes de Próxima Geração (IMS)

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia e Electrónica e Telecomunicações, realizada sob a orientação científica do Dr. Rui Luís Andrade Aguiar, Professor Auxiliar e do Dr. Paulo Jorge Salvador Serra Ferreira, Professor Auxiliar Convidado do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro.

Aos meus colegas pelo espírito de equipa e companheirismo

o júri

presidente

Professor Doutor José Carlos da Silva Neves
Professor Catedrático do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

arguente

Professor Doutor Manuel Alberto Pereira Ricardo
Professor Associado do Departamento de Engenharia Electrotécnica e de Computadores da Faculdade de Engenharia da Universidade do Porto

orientador

Professor Doutor Rui Luis Andrade Aguiar
Professor Auxiliar do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

agradecimentos

Ao Prof. Dr. Rui Aguiar pela sua orientação e apoio ao longo do percurso da minha tese.

À minha família, especialmente à minha mãe, por toda a compreensão e ajuda.

Aos meus colegas Herlander, André, Sancho, Nuno e em especial José Carlos, Marco Monteiro e Rui Gomes pelo forte apoio prestado e por partilharem o seu conhecimento e sabedoria.

Palavras-chave

Redes de Próxima Geração, Plataformas de Serviços, IMS, OMA, *Charging, Service Enablers*.

Resumo

Numa tentativa de atrair novos clientes e manter os actuais, os operadores de telecomunicações procuram novas plataformas e tecnologias que lhes permitam o desenvolvimento rápido e eficiente de novos serviços.

A arquitectura *IP Multimedia Subsystem* (IMS) normalizada pelo *Third Generation Partnership Project* (3GPP), inclui a capacidade de adicionar, modificar e remover sessões durante uma sessão multimédia, abrindo um novo leque de serviços combinando simultaneamente componentes de voz e dados entre outros componentes de *Media*.

O IMS define uma arquitectura independente da rede de acesso, separando claramente o nível de transporte, o nível de controlo e o nível de serviços. No nível de serviços residem as Plataformas de Serviço (*Service Delivery Platforms* – (SDPs)), que controlam toda a lógica de execução de serviços.

SDPs são uma nova aproximação arquitectural que tem como finalidade garantir o rápido desenvolvimento e execução de novos serviços multimédia, de um modo económico e simples. Tipicamente, as SDPs deverão fornecer um ambiente de criação, execução e gestão de serviços permitindo uma abstracção do ambiente de acesso ao serviço. A indústria das telecomunicações reconheceu a necessidade das SDPs, principalmente se estiverem de acordo com especificações industriais e construídas no topo de arquitecturas normalizadas (como é o caso do IMS).

Um dos problemas inerentes às SDPs consiste em perceber como tirar proveito de serviços distribuídos pelas várias plataformas (tradicionais/legadas, SIP/IMS, etc.), de uma forma dinâmica, de modo a construir um serviço composto (*Service Bundling*) de maior valor para o utilizador final.

O trabalho apresentado nesta dissertação de mestrado, foi o resultado de um estudo de toda a envolvente em torno das redes de próxima geração, analisando ainda detalhadamente a camada superior de serviços (plataforma de serviços) que irá, num futuro próximo, ter um papel fundamental na obtenção de receitas por parte do operador, dando ainda destaque aos elementos de taxação da arquitectura IMS.

Para demonstrador, foi desenvolvido um *Enabler* de *Charging* e um conector *Diameter* experimentais para tarifar um serviço de Vídeo Portal numa rede IMS e um serviço de chamadas de utilizadores IMS-PSTN com tarifação em tempo real, isto é, elaboração de um serviço pré-pago.

keywords **Next Generation Networks, Service Delivery Platforms, IMS, OMA, Charging, Service Enablers.**

Abstract

In an attempt to attract new customers and keeping the current ones, telecommunication operators are now searching new platforms and technologies for enabling fast and efficient new services development.
The IP Multimedia Subsystem (IMS) architecture, standardize by Third Generation Partnership Project (3GPP), includes capabilities to add, modify and remove sessions during an ongoing multimedia session, opening the opportunity for creating new services that allow combining voice, data and other media types simultaneously.
IMS defines an access independent architecture, composed by three separated layers: the transport layer, the control layer and the service layer. In the service layer we can find the Service Delivery Platforms (SDPs), which contains all the service execution logic.
SDPs are a new architectural approach intended to enable the rapid development and deployment of new converged multimedia services. SDPs should typically provide a service creation, execution and management environment, and a network abstraction layer. The telecommunications industry has recognized the need for an SDP, particularly, one that conforms to industry standards and are built on top of a standard architecture (IMS case).
The problem lies on how to profit with the different services distributed by different platforms (legacy, SIP/IMS, etc.) in a dynamic way, allowing the creation of value added compose service for the end user.
The work here presented on was the result of a detailed study about new next generation networks, in particular the application layer, which in a near future, will produce high revenues for the operators at the 3GPP environment. This will be based on the well defined 3GPP Charging Framework for the IMS architecture.
A demonstrator was developed using an experimental Charging Enabler and a Diameter Resource Adaptor to allow charging an IMS Video Portal service and IMS user calls to PSTN – PC2Phone service, allowing online (real time) charging, more concretely pre-paid services.

Índice

Índice de Figuras

Índice de Tabelas

Acrónimos

3GPP	Third Generation Partnership Project
3PCC	3rd Party Call Control
AAA	Authentication, Authorization and Accounting
ABMF	Account Balance Management Function
ALG	Application Layer Gateway
API	Application Programming Interface
AS	Application Server
AVP	Atribute Value Pair
B2BUA	Back 2 Back User Agent
BD	Billing Domain
BGCF	Breakout Gateway Control Function
BICC	Bearer Independent Call Control
CAMEL	Customized Applications for Mobile network Enhanced Logic
CAP	CAMEL Application Part
CDF	Charging Data Function
CDR	Charging Data Records
CGF	Charging Gateway Function
CRF	Charging Rule Function
CTF	Charging Trigger Function
CS	Circuit Switching
CSAF	Circuit Switched Adaptation Function
CSCF	Call/Session Control Function
CSI	Combinational Services
CSRN	CS Domain Routing Number
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DSF	Domain Selection Function
DTF	Domain Transfer Function
EBCF	Event-Based Charging Function
ECUR	Event Charging with Unit Reservation
EJB	Enterprise JavaBeans
ENUM	TElephone NUmber Mapping
ETSI	European Telecommunication Standards Institute
FBC	Flow Based Control
FQDN	Fully Qualified Domain Name

GGSN	Gateway GPRS Support Node
GPRS	General Packet Radio Service
GRX	GPRS Roaming Exchange
GSM	Global System for Mobile communications (originally, Groupe Spécial Mobile)
gsmSCF	GSM Service Control Function
GTP	GPRS Tunneling Protocol
HLR	Home Location Register
HSS	Home Subscriber Server
IBCF	Interconnection Border Control Function
I-CSCF	Interrogating-CSCF
IEC	Immediate Event Charging
IESG	Internet Engineering Steering Group
IETF	Internet Engineering Task Force
IMRN	IP Multimedia Routing Number
IMS	IP Multimedia Subsystem
IMSI	International Mobile Subscriber Identifier
IMS-MGW	IMS- Media Gateway
IM-SSF	IP Multimedia Service Switching Function
IN	Intelligent Network
ISC	IMS Service Control
ISDN	Integrated Services Digital Network
ISUP	ISDN User Part
ITU	International Telecommunication Union
J2EE	Java 2 Platform, Enterprise Edition
JCP	Java Community Process
JMX	Java Management Extensions
JSLEE	Java Service Logic Execution Environment
MGCF	Media Gateway Control Function
MMD	Multimedia Domain
MRFC	Media Resource Function Controller
MRFP	Media Resource Function Processors
NAI	Network Access Identifies
NAT	Network Address Translator
NGN	Next Generation Network
O&M	Operation and Management
OCF	Online Changing Function
OCS	Online Charging System

OMA	Open Mobile Alliance
OSA-SCS	Open Service Access-Service Capability Server
PCC	Policy and Charging Control
PCM	Pulse Code Modulation
PCRF	Policy and Charging Rule Function
P-CSCF	Proxy-CSCF
PDF	Policy Decision Function
PEP	Policy Enforcement Point
PLMN	Public Land Mobile Network
PSTN	Public Switched Telephone Network
PTT	Push to Talk
QoS	Quality of Service
RF	Rating Function
RFC	Request for Comments
RTP	Real-time Transport Protocol
SBB	Service Building Block
SBCF	Session-Based Changing Function
SBLP	Session-Based Local Policy
SCF	Session Charging Function
SCTP	Stream Control Transmission Protocol
SCUR	Session Charging with Unit Reservation
S-CSCF	Serving-CSCF
SDP	Service Delivery Platforms
SDP	Session Description Protocol
SGW	Signaling Gateway
SIP	Session Initiation Protocol
SLF	Subscription Locator Function
SPR	Subscription Profile Repository
SPT	Service Point Triggers
TCP	Transmission Control Protocol
THIG	Topology Hiding Inter-network Gateway
TISPAN	Telecoms and Internet converged Services and Protocols for Advance Network
TPF	Traffic Plane Function
TrGW	Transition Gateway
UDP	User Datagram Protocol
UE	User Equipment
UMTS	Universal Mobile Telecommunication System

VCC	Voice Call Continuity
VOD	Video On Demand
VoIP	Voice over IP

Capitulo 1 – Introdução

As redes de comunicações sofreram grandes desenvolvimentos nos últimos anos. Alguns exemplos são fáceis de constatar; simples conversas telefónicas através de um acesso fixo progrediram para comunicações sem fios, o *broadcast* da TV analógica evoluiu para digital, as ligações da Internet evoluíram para banda larga. Existem mais formas de comunicar do que nunca e a necessidade de convergir tanto ao nível das redes como dos serviços de comunicações tornou-se obrigatória. Assim, as redes de comunicações tenderão a ser, progressivamente e cada vez mais, suportadas sobre IP. Ao nível dos serviços tende-se para uma situação em que independentemente do tipo de rede de acesso, terminal e localização, o utilizador possa ter acesso aos mesmos serviços.

1.1 **Motivação**

Com a crescente generalização e difusão do *IP Multimedia Subsystem* (IMS), começam a ser resolvidos os problemas de conectividade entre domínios de rede, abrindo caminho para novos desafios, nomeadamente ao nível dos serviços e aplicações. Serviços economicamente viáveis requerem plataformas de servidores aplicacionais que garantam interoperabilidade entre as redes de próxima geração e redes tradicionais/legadas, disponibilidade universal, flexibilidade e alta performance. É necessário cada vez mais aumentar o leque de serviços, assegurando os recursos de rede necessários, de forma a ser possível aos operadores recuperarem as perdas para as tecnologias tipo *Voice over IP* (VoIP), onde o operador já não tem o domínio tanto da rede, como de todos os serviços. Mais que tudo, é necessário perceber como tirar receitas desses novos serviços.

Torna-se necessário o uso *Service Delivery Platforms* (SDP), plataformas que proporcionem ambientes de execução de aplicações de rede robustos e flexíveis, que disponibilizem as funcionalidades da rede mascarando a sua complexidade de forma a permitir que os serviços que façam uso destas funcionalidades sejam desenvolvidos, testados e implementados rapidamente.

Como resposta da PT Inovação (PTIN), empresa fornecedora de sistemas e plataformas, às novas necessidades resultantes da evolução para a convergência de redes (redes de próxima geração), surgiu a iniciativa *Service Handling on IP NETworks* (SHipNET®). Esta iniciativa engloba um conjunto de produtos no nível de serviços e aplicações, nível de controlo e no nível de transporte e acesso. A arquitectura SH*ip*NET® pretende fornecer, aos clientes de soluções PTIN, um cenário de evolução no sentido da convergência, que

satisfaça os seus requisitos e necessidades, garantindo simultaneamente os investimentos já feitos em tecnologias PTIN. Esta evolução pretende ainda oferecer uma solução que garanta a disponibilidade a um leque mais extenso de serviços e funcionalidades, que poderão resultar em novos modelos de negócio. A longo prazo, leva ainda à amortização dos custos no investimento na evolução das redes e obviamente a um exponencial aumento de receitas com o acesso a novos serviços mais rapidamente e que primem pela diferenciação.

Nesta iniciativa enquadra-se o desenvolvimento de dois serviços apresentados nesta dissertação, o serviço Vídeo Portal e o serviço PC2Phone. O serviço Vídeo Portal é basicamente um serviço de *Video On Demand* (VOD) na rede IMS, através do *streaming* do vídeo. O serviço PC2Phone é um serviço de tarifação de chamadas de utilizadores IMS com a PSTN.

1.2 **Objectivos**

Pretende-se com este projecto estudar tecnologias para implementação de plataformas de serviços, como ainda implementar componentes ao nível de conectores de rede e serviços que sejam compatíveis com os requisitos do IMS. Deste modo, é necessário um estudo prévio das Redes de Próxima Geração e do novo conceito a nível da camada aplicacional, as plataformas de serviços SDPs. Como demonstrador, pretende-se criar um *Enabler* de *Charging* experimental e um conector *Diameter*, que permitam tarifar serviços IMS. Em concreto é necessário que este demonstrador permita tarifar um serviço de Vídeo Portal na rede IMS, bem como realizar a tarifação do serviço PC2Phone. Assim, é ainda necessário compreender a *framework* de *Charging* do *3rd Generation Mobile System* (3GPP), dando especial destaque a esta arquitectura para o subsistema IMS.

Os objectivos do trabalho resumem-se assim ao:

- Estudo da arquitectura e serviços IMS (nomeadamente na camada aplicacional, VCC – *Voice Call Continuity*, CSI – *Combinational Services* e *Multimedia Telephony*);
- Estudo dos protocolos *Session Initiation Protocol* (SIP) e *Diameter*;
- Estudo de tecnologias para desenvolvimento de SIP *Application Servers* (AS);
- Estudo dos *Enablers Open Mobile Alliance* (OMA);
- Estudo da *framework* de *Charging* do 3GPP;
- Desenvolvimento de um *Enabler* de *Charging* e de um conector *Diameter* para um SIP AS da arquitectura IMS;

1.3 **Estrutura do Documento**

Esta tese encontra-se organizada da seguinte forma:

O Capitulo 2 descreve o estudo efectuado ao nível da Rede de Próxima Geração (RPG em inglês, *Next Generation Networks* - NGNs) definida pelo 3GPP, a arquitectura IMS. Este capítulo pretende demonstrar o que está a levar os operadores a empenharem-se na mudança das suas plataformas, apresentando os princípios básicos do IMS bem como uma descrição dos elementos principais que compõem esta nova arquitectura. São apresentados os protocolos SIP e *Diameter*, importantes na interacção entre muitos dos elementos da rede IMS e utilizados no demonstrador desta dissertação. Para finalizar, são apresentados alguns dos serviços que fazem parte da arquitectura IMS e que permitem a interligação com as redes legadas (os chamados *Transition Services*).

O Capitulo 3 ilustra uma das novas tendências do mercado, as SDPs. É possível encontrar algumas das características destas plataformas (que não são normalizadas, mas que de um modo geral seguem uma arquitectura semelhante) que permitem fornecer serviços de uma forma fácil e rápida. É ainda explorada a coexistência entre as SDPs e o IMS, isto é, o modo como as SDPs encaixam na camada aplicacional do IMS. Para tal, é importante apresentar o papel da OMA nesta área, e o seu esforço para interligar os serviços definidos com os componentes presentes na arquitectura IMS. Para terminar, é apresentada a iniciativa *JAIN Service Logic Execution Environment* (JSLEE), que pretende ser uma extensão à plataforma Java ao mundo das telecomunicações, fornecendo um ambiente de criação, execução e gestão de serviços para estas plataformas.

No Capitulo 4 é feita uma abordagem à *framework* definida pelo 3GPP para a taxação, sendo explicado em detalhe o caso específico de tarifação em IMS.

O Capitulo 5 apresenta o trabalho prático elaborado, que juntou todo o conhecimento adquirido com os elementos de informação dos capítulos anteriores, permitindo criar um *Enabler* de *Charging* e um *Resource Adaptor* (RA) *Diameter* utilizando a plataforma *Open Source Mobicents* (plataforma *compliant* com a *standard* JSLEE 1.0) e a stack proprietária *Diameter, Intelligent Diameter Stack* (IDS).

O último capítulo apresenta o sumário e as conclusões desta tese.

Em anexo, temos ainda as organizações responsáveis pela normalização e desenvolvimento das RPGs, descrevendo a área de trabalho onde cada um dos organismos se insere e ainda as colaborações existentes entre elas de forma a criar sistemas inter-operacionais (Anexo A). No Anexo B encontra-se definido o modo de interacção entre a camada de controlo e a

camada de serviços. Para finalizar, presente no Anexo C, é possível observar a forma como os utilizadores são identificados no domínio IMS.

Capitulo 2 – 3GPP IP Multimedia Subsystem

Durante os últimos anos, um dos temas mais emergentes e promissores no seio das telecomunicações foi sem dúvida, a convergência da Internet com as redes móveis. Hoje essa discussão tornou-se realidade e o IMS é a tecnologia-chave para atingir esse objectivo. A arquitectura IMS foi definida pelo 3GPP para as redes móveis all-IP e adoptada de forma semelhante pelo 3GPP2, servindo de base às evoluções para redes convergentes sob normalização no âmbito do *European Telecommunication Standards Institute* (ETSI) *Telecoms and Internet converged Services and Protocols for Advance Network* (TISPAN). O IMS é especificado como uma arquitectura que permite a integração de aplicações multimédia baseadas em IP e o estabelecimento de sessões multimédia entre utilizadores de redes baseadas em pacotes (WLAN, xDSL, WiMAX, etc.) e em circuitos (PSTN, ISDN, etc.), para além das redes móveis (Figura 1).

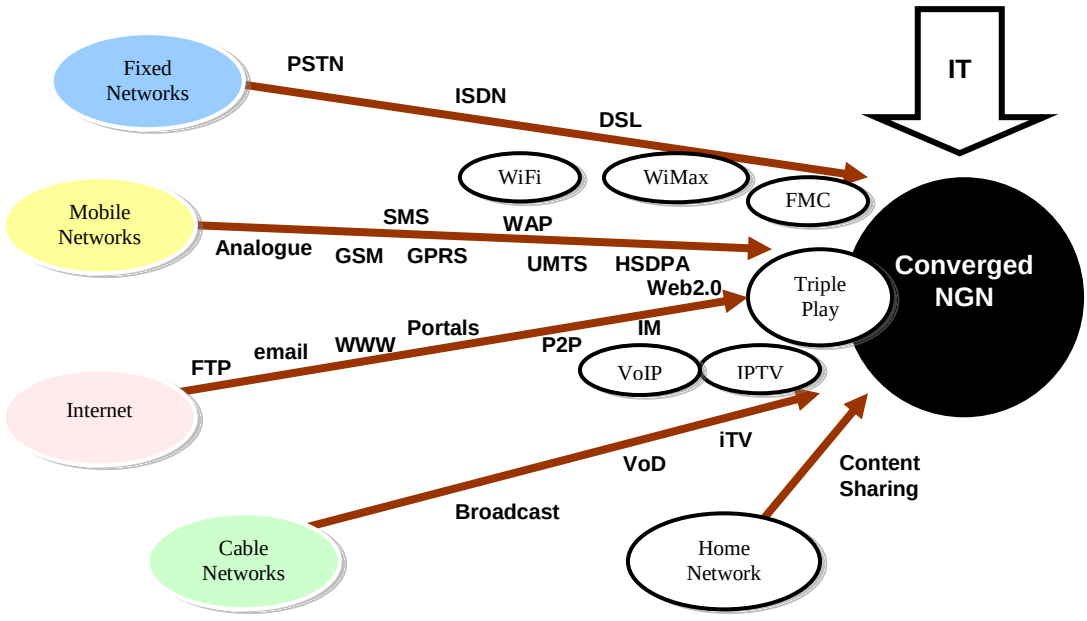

Figura 1 – Convergência nas redes de próxima geração [79]

O 3GPP específica o *core* IMS como um sistema independente da tecnologia de acesso, permitindo, por outras organizações, a normalização de uma arquitectura de convergência de várias redes de acesso utilizando o *core* IMS como elemento principal. O IMS especifica um conjunto nuclear de entidades funcionais de rede que proporcionam o acesso aos serviços de comunicações baseados em *Session Initiation Protocol* (SIP), providenciados pelos *Communications Service Providers* (CSPs).

Actualmente, e devido às constantes mudanças no mercado (muito por culpa de tecnologias como o VoIP), os operadores têm a necessidade de reajustarem os seus componentes de forma a suportarem todos os serviços e funcionalidades que irão aparecer com esta nova arquitectura.

Do ponto de vista do mundo das telecomunicações, o IMS permitirá uma grande evolução no sentido de aproximar as telecomunicações fixas das telecomunicações móveis, nomeadamente ao nível dos serviços.

Na perspectiva dos operadores, o IMS leva o conceito da arquitectura por camadas mais longe, na medida em que define uma arquitectura horizontal (ver posteriormente) onde as funções comuns e *service enablers* possam ser reutilizadas em diversas aplicações. Esta arquitectura também especifica a interoperabilidade com outras redes, *bearer control*, tarifação e segurança. Sendo uma arquitectura que proporciona integração com as redes existentes, é um ponto-chave na convergência fixo-móvel, assim como na migração para uma arquitectura "all-IP" (Figura 2).

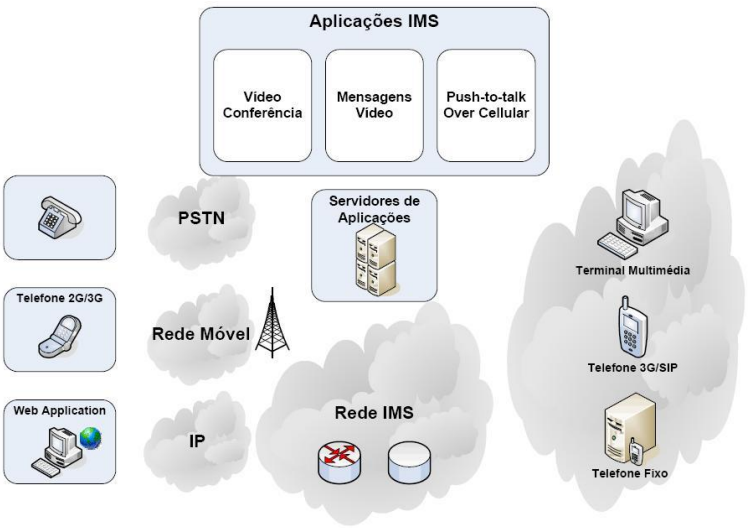

Figura 2 – Convergência das diferentes redes

Através de uma arquitectura estratifica, com interfaces abertas, é possível disponibilizar aplicações e serviços independentes das redes de acesso e dos terminais (multi-serviço e multi-acesso).

Este capítulo apresenta os princípios básicos nos quais o IMS se rege bem como os elementos e alguns protocolos da plataforma IMS. São apresentados os protocolos SIP e *Diameter*, importantes na interacção entre alguns dos elementos da rede IMS e parte dos utilizados no demonstrador desta dissertação. O protocolo SIP pode ser considerado como a tecnologia chave do IMS já que foi o protocolo de sinalização escolhido pelo 3GPP para a grande maioria das interfaces entre os elementos da rede IMS. Passando dos protocolos

para a arquitectura, é realizada uma descrição dos elementos pertencentes ao IMS. Para finalizar, são apresentados alguns dos serviços que fazem parte da arquitectura IMS e que permitem a interligação com as redes legadas (*Transition Services*).

2.1 Características Básicas do IMS

Antes de surgir o conceito IMS, a situação na qual os operadores se encontravam não era muito encorajadora. Se por um lado, o mercado baseado em voz sobre comutação de circuitos acomodou-se, tornando complicado para os operadores obterem apenas receitas fornecendo e tarifando chamadas de voz, por outro, serviços sobre comutação de pacotes ainda não tinham grande penetração no mercado, não obtendo os operadores ainda grandes receitas com a sua utilização.

Os operadores precisavam de uma forma de fornecer serviços sobre pacotes de uma forma mais atractiva, atraindo assim os utilizadores para o domínio de comutação sobre pacotes [17].

Deste modo, o IMS foi criado, tendo como objectivos:

- Combinar as últimas tendências da tecnologia;
- Permitir a interligação e mobilidade na Internet;
- Criar uma plataforma comum para desenvolver serviços multimédia diversificados;
- Criar um mecanismo que permitisse aumentar a performance da utilização da rede devido ao uso excessivo da rede de comutação de pacotes;

Para atingir estes objectivos, o 3GPP definiu o IMS como uma *framework* arquitectural criada com o objectivo de oferecer serviços multimédia sobre IP para os utilizadores, definido na norma [18]. Inicialmente desenhada para fornecer serviços através da rede *General Packet Radio Service* (GPRS), a partir da *release* 6[1], foi adicionado um novo requisito de forma a suportar redes de acesso diferentes, dado que, como foi dito, pretende-se que o IMS seja independente da rede de acesso [17].

2.1.1 *Sessões Multimédia IP*

Apesar do IMS possibilitar a transmissão de um leque vasto de serviços, tem obrigatoriamente de suportar um requisito de especial importância: comunicações de áudio e vídeo. Este requisito ilustra a necessidade de suportar o principal serviço a ser entregue pela arquitectura IMS: sessões multimédia sobre redes de comutação de pacotes.

[1] Ver Anexo A para uma compreensão da estrutura das especificações do 3GPP

Multimédia refere-se à simultânea existência de diferentes tipos de *Media*, neste caso áudio e vídeo.

Este requisito já se encontrava presente em normas 3GPP anteriores a esta especificação, mas essas comunicações eram efectuadas sobre redes de comutação de circuitos, e não sobre a rede de comutação de pacotes.

2.1.2 *Qualidade de Serviço*

Um dos principais requisitos da arquitectura IMS, e que a distingue de uma simples ligação VoIP, é a utilização de mecanismos de *Quality of Service* (QoS) que garantam serviços multimédia de qualidade entre os utilizadores. Um dos maiores problemas dos tradicionais serviços multimédia baseados em IP é a falta de QoS e o serviço ser feito em modo *best effort*. Os serviços em tempo-real são os mais afectados por este problema. São estes os mais sensíveis a certas condições da rede como atrasos, variações de atraso, perdas de pacotes pois necessitam de uma largura de banda mínima para que o serviço disponibilizado seja uma boa experiência para o utilizador.

O IMS suporta diferentes mecanismos de QoS para sessões multimédia entre o utilizador e o operador, no momento em que são estabelecidas e mesmo durante as sessões. O mesmo se aplica para a negociação de QoS de determinados componentes *Media* individuais. Este requisito deve então permitir que para uma sessão de voz entre utilizadores do domínio IP se faça com igual qualidade em relação ao domínio de comutação de circuitos. Isto leva a que a qualidade de uma sessão multimédia seja igual e independente da tecnologia da rede onde se encontra o utilizador.

A QoS negociada para uma determinada sessão depende do tipo de aplicação utilizada, bem como outros factores, como o máximo de largura de banda que pode ser reservada para esse utilizador, dependendo do tipo da sua subscrição, ou estado da rede. O IMS deve permitir que os operadores controlem os parâmetros de QoS dos seus utilizadores para que possam diferenciar certos grupos de clientes. Esta solução é especificada em [20] e [19].

A partir da *release* 7, o 3GPP definiu ainda uma nova plataforma de integração de QoS e tarifação *Policy and Charging Control* (PCC), que proporciona, por exemplo, diferentes níveis de QoS para subscritores que são tarifados de forma distinta (ver secção 4.2.4).

2.1.3 *Interligação de Redes*

É obvio que um dos requisitos básicos do IMS é a interligação com a Internet, devido a este domínio possuir um vasto número de potenciais destinatários para as sessões multimédia iniciadas pelos utilizadores IMS.

Para além das redes de comutação de pacotes (como a rede IP), é necessário que o IMS permita o acesso às suas aplicações pelos utilizadores das redes fixas de comutação de circuitos (como a PSTN e ISDN), das redes móveis normalizadas pelo 3GPP (como o *Global System for Mobile communications* (GSM) e *Universal Mobile Telecommunication System* (UMTS)) bem como WLAN, WIMAX, etc.

No caso do requisito "Múltiplos Acessos" a diferença é que não se trata de interligação, mas sim de ter os utilizadores a acederem ao IMS através desses novos acessos.

2.1.4 *Roaming*

O suporte de roaming tem sido um requisito geral desde a segunda geração de redes móveis.

Os utilizadores terão obrigatoriamente de ter a opção de comunicar com diferentes redes. O IMS herda obviamente esta capacidade, podendo os utilizadores comunicarem, por exemplo, entre países diferentes (sujeito às negociações de *roaming* existentes entre as redes que realização a comunicação).

2.1.5 *Controlo de Serviços*

Tipicamente, os operadores querem impor políticas de controlo sobre os serviços entregues ao utilizador, sendo possível dividir estas políticas em duas categorias:

- Políticas Gerais: Estas políticas são aplicáveis a todos os clientes de um operador IMS. Por exemplo, a restrição de utilização de *codecs* de vídeo e áudio que necessitem de uma grande largura de banda.
- Políticas individuais: Estas políticas são particulares para cada um dos utilizadores ou grupo de utilizadores de uma rede IMS. Nesta situação, são configuradas especificamente para cada utilizador e dependem da sua subscrição para com o seu operador IMS. Por exemplo, podem existir clientes com subscrições num operador IMS que restrinjam o acesso a vídeo. Nesta situação, caso o cliente tente estabelecer uma sessão vídeo, o operador IMS tem a possibilidade de não autorizar a sua utilização.

2.1.6 *Ambiente de rápida criação de Serviços*

Este requisito teve um grande impacto no desenho da arquitectura IMS, dado que declara que um serviço IMS não necessita de ser normalizado. Pelo contrário, é sim o sistema de suporte a estes serviços que deve ser normalizado de forma a permitir o desenvolvimento de novos serviços, para os operadores e utilizadores, por outras entidades independentes. Este modelo é o oposto ao seguido em anteriores redes móveis, onde os serviços eram normalizados para um determinado operador/rede de acesso. Nesta situação não havia garantias que um determinado serviço iria funcionar em caso de *roaming*, como é por exemplo o caso do acesso ao *voicemail*.

A arquitectura IMS deve permitir que entidades independentes do operador possam desenvolver e fornecer rapidamente aplicações e serviços multimédia que funcionem de igual forma em diferentes redes de operadores IMS.

2.1.7 *Múltiplos Acessos*

Este requisito tem como objectivo estender o acesso à arquitectura IMS por redes de acesso diferentes de GPRS, isto é, tornar o IMS independente da plataforma de acesso. O IMS eventualmente funcionará em qualquer tipo de rede (fixa, móvel ou *wireless*) com sistemas de comutação de pacotes, tais como com GPRS, UMTS, CDMA2000, WLAN, WiMAX, DSL, cabo, entre outros. Sistemas legados de comutação de circuitos (POTS, GSM) são suportados através de *gateways*. A coexistência de chamadas/sessões de diferentes redes de acesso é suportada recorrendo a interfaces abertos entre as camadas de controlo e serviço.

2.2 **Arquitectura IMS**

Antes de explorar a arquitectura geral no IMS é necessário ter em mente que o 3GPP não normaliza os elementos físicos da arquitectura IMS, mas sim as funcionalidades e interfaces entre estes mesmos elementos. Isto significa que a arquitectura IMS é uma colecção de funcionalidades interligadas por interfaces normalizadas. Os fabricantes têm a possibilidade de combinar diferentes funcionalidades num único elemento da arquitectura (isto é, num único elemento físico). Similarmente, os fabricantes poderão separar uma única funcionalidade em dois ou mais elementos.

É importante referir que a arquitectura IMS definiu inicialmente o uso exclusivo de IPv6 nas suas redes. Porém, durante os últimos anos, o progresso na passagem de v4 para v6 por parte dos operadores não foi muito significativo. Como acréscimo, o trabalho realizado relativo a problemas de *Network Address Translator* (NAT) no SIP progrediram

substancialmente. Desta forma, para permitir a integração das redes IPv4 tradicionais foi necessário definir dois novos elementos, o *Application Layer Gateway* (ALG) e o *Transition Gateway* (TrGW). O primeiro realiza a interoperabilidade entre IPv4 e IPv6 no plano da sinalização (mensagens SIP e *Session Description Protocol* (SDP)), enquanto o ultimo processa o tráfego do *Media* (por exemplo., *Real-time Transport Protocol* (RTP)).
A arquitectura IMS possui uma divisão lógica em três camadas funcionais de acordo com os elementos que a compõe e as suas funções para com a arquitectura. Estas camadas são: camada de transporte, camada de controlo e camada de serviços (Figura 3).

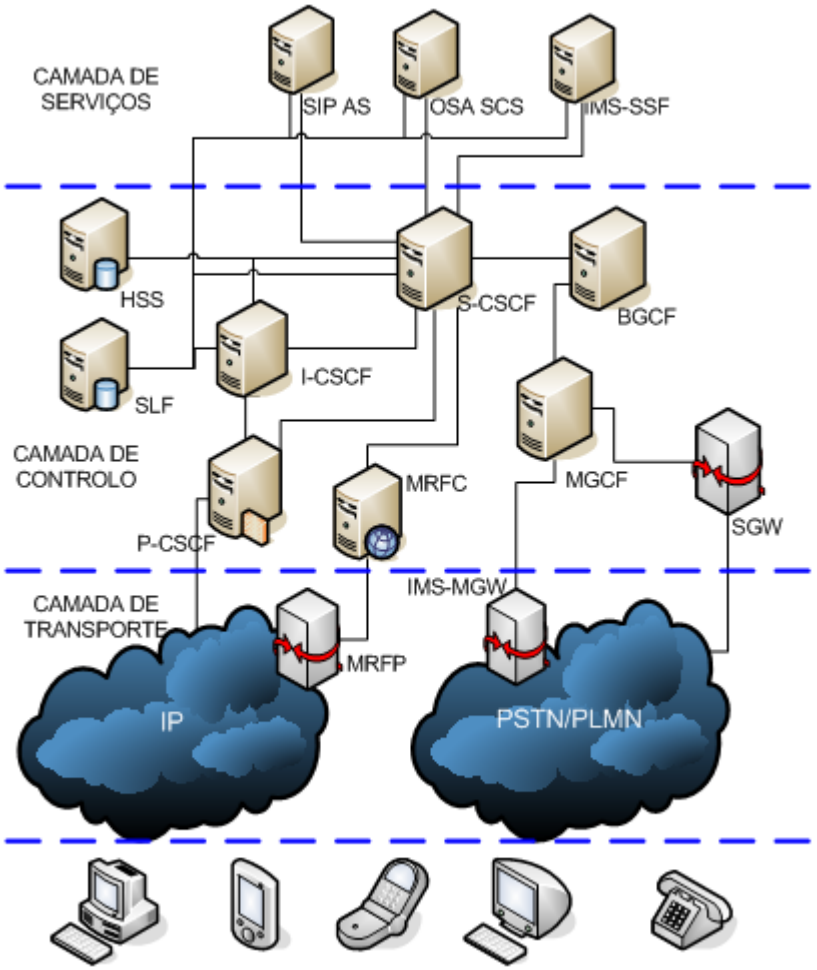

Figura 3 – Diferentes camadas do IMS

De seguida é feita uma descrição, de acordo com a norma [21], das funcionalidades de cada camada lógica do IMS e das funções desempenhadas por cada um dos elementos de cada camada.

2.2.1 *Camada de controlo*

Comporta servidores de controlo de rede que gerem o estabelecimento de sessões multimédia, a sua manutenção e libertação, além de manterem os dados relativos à

subscrição dos utilizadores. É ainda a camada responsável por garantir a independência entre as aplicações e o tipo de tecnologia utilizada nas redes de acesso. Sendo o mais importante destes servidores o *Call/Session Control Function* (CSCF), esta camada contém ainda um conjunto completo de funções de suporte, tais como aprovisionamento (sendo o HSS o principal elemento), tarifação e Operação & Gestão (O&M – *Operation & Management*). A interoperabilidade de redes não IMS é conseguida através de *gateways*.

2.2.1.1 Home Subscriber Server (HSS) e Subscriber Location Function (SLF)

O HSS é uma entidade comum tanto ao domínio PS como ao CS (engloba as funcionalidades presentes no *Home Location Register* (HLR)), estando ainda ligado à rede IMS. Funciona como um repositório de informação para um determinado utilizador, contendo a informação relativa à sua subscrição, permitindo dar suporte às entidades da rede responsáveis pelas chamadas/sessões. Uma rede poderá conter um ou mais HSS, dependendo do número de utilizadores, na capacidade do equipamento e na organização da rede. De forma a obter o endereço dos diferentes HSSs utiliza-se um *Subscription Locator Function* (SLF), que não é mais do que uma simples base de dados que contem o mapeamento entre o utilizador e o endereço do HSS. Os nós de rede questionam o SLF, com o endereço do utilizador como parâmetro, e obtêm o HSS que contém a informação relativa ao utilizador em questão.

Como exemplo, podemos considerar o HSS como uma entidade que permite dar suporte aos servidores de controlo de chamadas de forma a completar os procedimentos de encaminhamento/*roaming* permitindo realizar, indirectamente, autenticação, autorização, localização entre outras coisas.

O HSS é responsável por armazenar a informação relativa ao utilizador:

- Identificação, número e endereço;
- Informação de segurança do utilizador;
- Informação de localização do utilizador;
- Perfil do utilizador;
- Endereços de taxação;
- Etc.

Ambos implementam o protocolo *Diameter* [31] com aplicações *Diameter* específicas para o IMS.

É possível encontrar no Anexo C uma descrição das diferentes identidades de um utilizador no IMS e o modo em como estas se relacionam.

2.2.1.2 Call/Session Control Function (CSCF)

O CSCF, que é um servidor SIP, é um dos elementos essenciais no IMS. Este elemento tem como tarefa processar a sinalização SIP no IMS. Existem três diferentes tipos de CSCF, dependendo das funcionalidades que apresentam:

- *Serving*-CSCF (S-CSCF)
- *Interrogating*-CSCF (I-CSCF)
- *Proxy*-CSCF (P-CSCF)

2.2.1.3 Serving-Call/Session Control Function (S-CSCF)

O S-CSCF é considerado o cérebro do IMS, isto é, o elemento central do *core* IMS no processamento de sinalização SIP e controlo das sessões multimédia dos utilizadores IMS. Localizado na rede natural[2] do subscritor, proporciona o serviço de registo e controlo de sessão aos *User Equipments* (UE)s, além de interagir com as plataformas de serviço através da interface IMS *Service Control* (ISC) (ver no anexo A como é feita esta interacção).

Tal como acontece com os restantes CSCFs, também pode existir mais do que um S-CSCF por rede, sendo possível deste modo, diferentes funcionalidades e finalidades para cada um deles.

As funções que o S-CSCF pode desempenhar são as seguintes [22]:

- Funciona como um SIP *Registrar* [25]. Isto significa que estabelece uma associação entre o endereço IP do terminal do utilizador e do P-CSCF com o endereço público SIP desse utilizador que se pretende registar. Para isso, este elemento aceita as mensagens de registo (SIP *REGISTERs*) dos utilizadores e verifica a sua validade através dos vectores de autenticação obtidos do HSS;
- Autentica os utilizadores, através dos vectores de autenticação que obtém do HSS;
- Acede ao HSS para obter informações sobre a subscrição de um determinado cliente IMS. Estas informações incluem os serviços ao qual o utilizador está autorizado a aceder e os *Application Servers* (ASs) que devem ser incluídos;
- *Proxy Server*: aceita pedidos e serve-os internamente ou encaminha-os, podendo efectuar traduções se necessário (por ex., traduzir números de formato E.164 para

[2] Entenda-se "rede natural" a rede onde se encontra a subscrição do utilizador, isto é, a sua operadora.

SIP URIs, usando um servidor *TElephone NUmber Mapping* (ENUM)); além disso mantém os *timers* das sessões, o que lhe permite saber o seu estado e terminar sessões "penduradas";

- *User Agent*: pode terminar e gerar transacções SIP de forma independente (controlo de sessão);
- Decide quando um *request*/*response* deve ser encaminhado para um AS específico para ser processado;
- Supervisiona os *timers* de registo com a capacidade de eliminar registos se necessário.
- Encaminha o tráfego terminado para o P-CSCF e o tráfego originado para o I-CSCF/IBCF, para o *Breakout Gateway Control Function* (BGCF) ou para o AS;
- Gera *Charging Data Records*[3] (CDRs) com informação relativa às sessões e registos dos utilizadores para efeitos de taxação.
- Disponibiliza a informação de um determinado cliente aos outros elementos da sua rede durante o período de tempo em que esse cliente se encontra registado neste elemento;
- Controla a sessão multimédia dos utilizadores registados neste elemento e pode impedir o estabelecimento de determinadas sessões para um utilizador com base em determinados critérios subjacentes ao tipo de subscrição ou que podem prejudicar o cliente.

2.2.1.4 Interrogating-Call/Session Control Function (I-CSCF)

O elemento I-CSCF é o ponto inicial do *core* IMS para as sessões multimédia destinadas aos clientes dessa rede, ou clientes de outros operadores que estão registados nessa rede IMS (*roaming*).

Durante o processo de registo tem o papel de comunicar com o HSS, realizando a autorização do utilizador, como ainda engloba mecanismos para a selecção de um S-CSCF que fique responsável pelo utilizador.

O endereço deste elemento está presente no servidor *Domain Name System* (DNS) do seu domínio IMS, para que elementos remotos de outras redes IMS possam aceder ao endereço deste elemento e utilizá-lo para encaminhar mensagens SIP para essa rede IMS.

O I-CSCF funciona também como um SIP *proxy* com as seguintes funcionalidades de acordo com a norma [18]:

- Reserva um S-CSCF para um determinado utilizador durante o seu processo de registo. Para isso, quando recebe um registo (SIP *REGISTER*) proveniente do P-CSCF do utilizador a registar, este elemento comunica com o HSS e com o SLF, se existir mais que um HSS, através do protocolo *Diameter* para saber se o utilizador é cliente dessa rede e para obter informações sobre a sua subscrição. Em caso afirmativo, encaminha o registo para o S-CSCF eleito com a finalidade de registar este utilizador nesse elemento e proceder à sua reserva, caso o utilizador ainda não esteja registado;
- Cifra certas partes da mensagem SIP contendo informação confidencial do operador IMS. Esta funcionalidade designada por *Topology Hiding Inter-network Gateway* (THIG) é opcional e está documentada na *release* 6 do 3GPP. A partir da *release* 7 esta função passa a ser desempenhada pelo elemento *Interconnection Border Control Function* (IBCF) do *core* IMS;
- Encaminha as mensagens SIP de sessões multimédia de outros operadores IMS para o S-CSCF do utilizador destinatário. O endereço do S-CSCF reservado para esse utilizador é obtido do HSS antes do encaminhamento da mensagem;
- Encaminha as mensagens SIP de sessões multimédia para o S-CSCF ou I-CSCF, no caso de THIG activo, de outro operador IMS do utilizador destinatário, permitindo assim esconder a configuração, a topologia e as capacidades da rede para o exterior.
- Gera CDRs com informação relativa às sessões e registos dos utilizadores para efeitos taxação;

2.2.1.5 Proxy–Call/Session Control Function (P-CSCF)

O P-CSCF é o primeiro ponto de contacto ao nível da sinalização entre o terminal e a rede IMS. É um SIP *stateful* proxy ou seja, valida os SIP *requests*, encaminha-os para o destino devido e processa e encaminha as respostas SIP, mantendo o estado das sessões.

Uma rede IMS pode conter um ou mais P-CSCFs para efeitos de redundância ou para o caso de ter de servir um elevado número de utilizadores.

O seu endereço é descoberto através de mecanismos de *Dynamic Host Configuration Protocol* (DHCP) (genérico) ou com sinalização específica GPRS, *PDP Context Activation* (3GPP) [33]. No caso da rede fixa, por exemplo, este endereço é atribuído pelo elemento *Network Attachment Subsystem* (NASS) (não especificado neste documento por tratar-se

[3] CDRs são informação sobre eventos de taxação (ex. duração de uma chamada, quantidade de dados transferidos, etc.)

de um elemento específico da arquitectura TISPAN), também por mecanismos de DHCP, durante o processo de autenticação do utilizador na rede de acesso.

Como exemplo de um cenário de *roaming* em GPRS, o utilizador tem a possibilidade de se registar num P-CSCF na rede visitada (no caso desta ser *full* IMS) ou na rede natural (no caso da rede visitada ainda não ser completamente IMS *compliant*), Figura 4 e Figura 5, respectivamente.

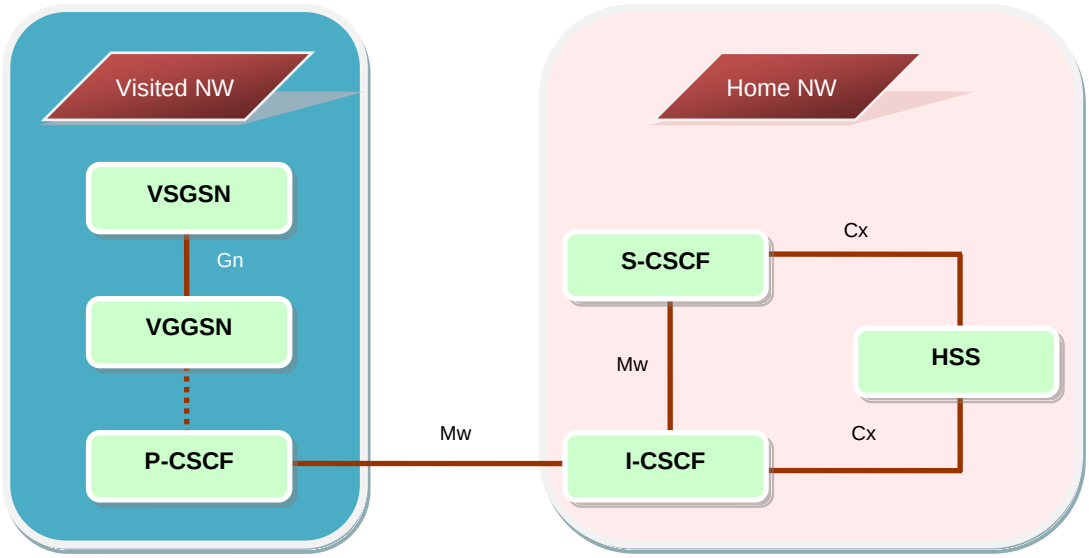

Figura 4 – Roaming em IMS: full IMS

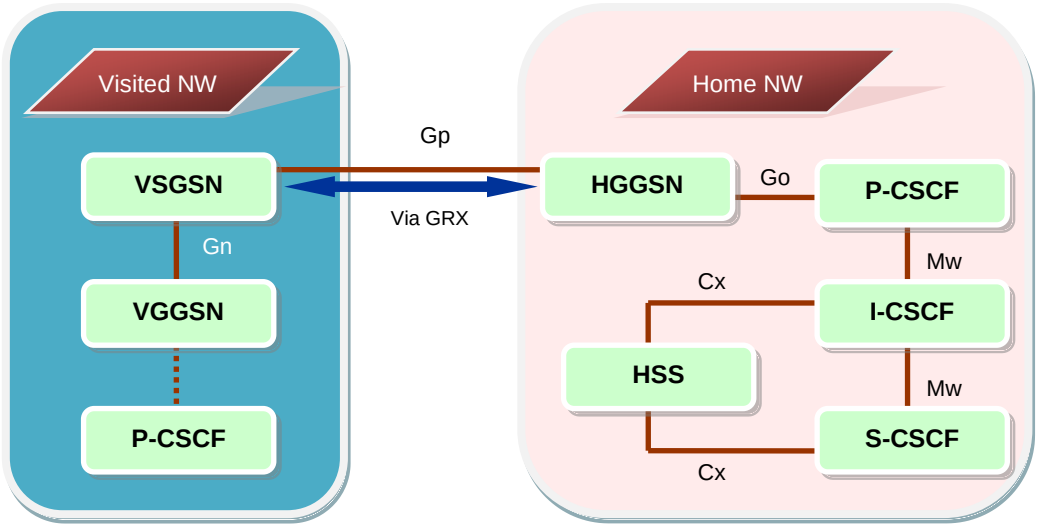

Figura 5 – *Roaming* ao nível da camada de acesso

No primeiro cenário, as normas IMS propõem um método mais eficiente de implementar roaming utilizando o *Gateway GPRS Support Node* (GGSN) e o P-CSCF na rede visitada como ponto de entrada para o IMS. Apesar de melhorar a utilização dos recursos, leva a problemas de compatibilidade nos mecanismos de descoberta do P-CSCF.

No segundo cenário, o modelo tradicional de *roaming* GPRS é utilizado. Os utilizadores utilizam os seus GGSN e P-CSCF (elementos localizados na rede operadora do utilizador) para acederem aos serviços IMS. Tanto a sinalização como os dados são reencaminhados via *GPRS Roaming Exchange* (GRX).

O primeiro cenário, em que o P-CSCF está na rede visitada, apresenta algumas vantagens mas é uma visão a longo prazo da implementação do *roaming* IMS. Este cenário requer que a rede visitada também suporte IMS e não é expectável que todos os operadores com quem são estabelecidos contratos de *roaming* suportem IMS. Assim, numa primeira fase, será o segundo cenário, em que o P-CSCF está na rede de origem (*home network*), o modelo preferido pelos operadores. Este cenário é similar ao *roaming* GPRS, quer do ponto vista técnico, quer do ponto de vista de negócio, na medida em que o tráfego IMS (sinalização e *Media*) é transparente à rede visitada. Iremos desta forma ter problemas, por exemplo, ao nível de QoS já que não vai ser possível controlar as sessões do extremo a extremo. Este processo de aquisição do endereço de um P-CSCF está descrito em [22].

A aquisição do endereço de um P-CSCF e a sua reserva para um determinado utilizador é realizada durante o processo de registo desse utilizador na rede IMS. Depois de reservado, toda a sinalização trocada entre esse utilizador e o *core* IMS passa por este elemento durante o período de tempo em que o utilizador se encontra registado.

O P-CSCF funciona como um SIP *proxy* com as seguintes funcionalidades:

- Autentica o utilizador e estabelece uma ligação segura IPsec com o seu terminal. Esta ligação permite assegurar a integridade e confidencialidade dos dados trocados entre o utilizador e o seu P-CSCF. Os outros elementos do core IMS não necessitam de autenticar o utilizador, pois esta tarefa é desempenhada pelo P-CSCF e todos os outros elementos confiam neste elemento. Para mais detalhes sobre os mecanismos e procedimentos de segurança entre o utilizador e o P-CSCF, consultar a norma [23];
- Encaminha o registo (SIP *REGISTER*) do utilizador para o respectivo I-CSCF determinado a partir domínio do operador no qual o utilizador está subscrito (através de uma configuração manual ou resolvendo o domínio através de um servidor DNS);
- Encaminha as mensagens SIP, provenientes do terminal do utilizador no processo de estabelecimento de sessões, directamente para o S-CSCF, ou para um I-CSCF que depois encaminhará para o S-CSCF. O endereço do S-CSCF é obtido pelo P-CSCF como resultado do processo de registo do utilizador;

- Comprime as mensagens SIP para reduzir o tempo de transmissão e de estabelecimento da sessão;
- O P-CSCF pode conter um elemento externo ou interno (dependendo se for *release* 6 ou 5, respectivamente) designado por *Policy Decision Function* (PDF). Este elemento é responsável pela gestão de recursos e da QoS do plano de dados (*Media*) para o utilizador. Para mais detalhes, consultar a norma [24]. No caso da *release* 7 este elemento deixa de existir, sendo substituído pela *framework* PCC.
- Gera CDRs com informação relativa às sessões e registos dos seus utilizadores para taxação;

2.2.1.6 Breakout Gateway Control Function (BGCF)

O BGCF é uma entidade importante no que toca à interoperabilidade com redes de circuitos comutados; é um servidor SIP que inclui funcionalidades de *routing* para números telefónicos e é a entidade responsável por estabelecer o ponto onde ocorre a transição entre domínio IMS e o domínio CS (*Circuit Switching*). Esta transição pode ocorrer tanto na rede onde se encontra o BGCF como noutra rede. No primeiro caso o BGCF selecciona um MGCF (*Media Gateway Control Function*) na própria rede para continuar com a sessão; no segundo, escolhe um BGCF de outra rede reencaminhando o pedido. Neste último caso, é necessário realçar que poderá existir um I-CSCF ou um IBCF em situações onde os operadores não queiram colocar na fronteira de rede os BGCFs.

O BGCF é utilizado exclusivamente para sessões iniciadas por utilizadores de um operador IMS destinadas a utilizadores das redes PSTN ou PLMN, ou entre utilizadores que se encontram em redes PSTN diferentes e que utilizam a rede IMS como intermediária (cenário de transito).

As principais funcionalidades e procedimentos do BGCF, de acordo com a norma [22] são:

- Tomar a decisão de escolher a rede IMS onde será realizada a interacção com a rede PSTN/PLMN. Se esta interacção for feita na mesma rede IMS, então o BGCF encaminha a mensagem SIP para o MGCF apropriado da rede, baseado no número telefónico do destinatário. Se for realizada noutra rede IMS, o BGCF encaminha a mensagem SIP para o BGCF da outra rede;
- Gera CDRs com informação relativa às sessões dos utilizadores para efeitos de taxação.

2.2.1.7 Media Gateway Control Function (MGCF)

O MGCF é outra entidade importante no domínio da interoperabilidade com as redes CS, como é o caso das PSTN, já que se trata da *gateway* que torna possível a comunicação entre utilizadores da rede CS e utilizadores IMS. Toda a sinalização proveniente das chamadas originadas em redes CS passa através deste *gateway*, que converte ISDN *User Part* (ISUP) ou *Bearer Independent Call Control* (BICC) em SIP e vice-versa.

O IMS definiu um conjunto de elementos que formam o chamado sistema *Media Gateway* de *Trunking*, que em conjunto e com funções específicas permitem esta interacção e a troca de sessões entre utilizadores destes domínios. Os elementos que compõem este sistema são: MGCF, IMS-MGW (pertence à camada de transporte, ver mais à frente) e SGW (Figura 6).

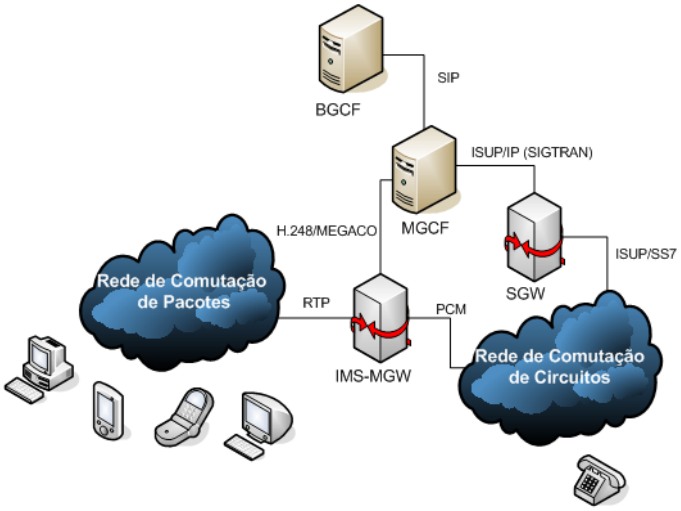

Figura 6 – *Media Gateway* de *Trunking*

Algumas das funcionalidades e procedimentos da MGCF são:

- Controlar os recursos de vários elementos IMS-MGW da camada lógica de transporte da arquitectura IMS, através do protocolo H.248 especificado em [26].
- Reservar canais de áudio na IMS-MGW para uma sessão de voz entre utilizadores IMS e PSTN;
- Escolher o *codec* de áudio apropriado na IMS-MGW para uma sessão de voz;
- Rejeitar a componente de vídeo de uma sessão e controlar os recursos da IMS-MGW somente para a componente de áudio;
- Gerar CDRs com informação relativa às sessões dos utilizadores para efeitos de taxação.

2.2.1.8 Signalling Gateway (SGW)

O elemento SGW, juntamente com o elemento MGCF descrito anteriormente, permite a troca de sinalização necessária ao estabelecimento de sessões de voz entre utilizadores da rede PSTN, onde é utilizada sinalização SS7, e utilizadores da rede IMS, onde é utilizada sinalização SIP. A principal função deste elemento passa pela conversão protocolar ao nível do transporte, para que a informação das camadas superiores (ex. ISUP) possa ser transferida entre domínios IP e PSTN sem sofrer qualquer alteração. Mais precisamente, este elemento converte o *Message Transfer Part* (MTP), especificado em [27], da sinalização SS7 no protocolo SCTP sobre IP com destino ao elemento MGCF. Ou seja o ISUP ou BICC sobre SCTP (IP) proveniente da MGCF é transformado na SGW em ISUP ou BICC sobre MTP (SS7) e vice-versa.

A SGW é transparente à sinalização das camadas superiores à de transporte (ex. ISUP, TCAP, BICC, etc.).

2.2.1.9 Media Resource Function Controller (MRFC)

O elemento MRFC no plano de sinalização, juntamente com o elemento MRFP no plano de dados, permite chegar aos clientes de um operador IMS anúncios provenientes de um servidor multimédia, estabelecer sessões de vídeo e/ou áudio conferencia entre vários clientes, reproduzir conteúdos multimédia em tempo-real, etc. É usado para tocar anúncios (áudio e vídeo), conferências multimédia, reconhecimento de fala e conversão *text-to-speech* (TTS), além de poder efectuar transcodificação em tempo real de dados multimédia.

Algumas das funcionalidades e procedimentos do elemento MRFC, de acordo com a norma [22], são:

- Controlar o recurso multimédia do elemento MRFP através do protocolo H.248;
- Processar as mensagens SIP provenientes do elemento S-CSCF e do AS de forma a controlar convenientemente o MRFP;
- Gerar mensagens SIP e enviá-las para o S-CSCF e para o AS, de acordo com os recursos do elemento MRFP.
- Gerar CDRs com informação relativa às sessões dos utilizadores para efeitos de taxação.

Este elemento está sempre localizado no *core* IMS do operador.

2.2.2 *Camada de Transporte*

Esta camada é composta por *router, switches* e elementos de acesso que se encontram na fronteira das redes do operador. A camada de transporte é aquela na qual os utilizadores se encontram ligados através do seu equipamento.

Ainda dentro desta camada, é possível encontrar algumas entidades IMS com diferentes funcionalidades. Entre estas funcionalidades, é possível destacar a possibilidade de alteração do *codec* em tempo-real do fluxo de áudio e/ou vídeo, caso os utilizadores não possuam um *codec* em comum na comunicação, permitindo a transferência de fluxo de dados de áudio e/ou vídeo entre domínios diferentes, através de *Media Gateways*, e controlam os fluxos de conteúdos multimédia fornecidos a determinados clientes IMS.

É ainda possível observar as funcionalidades de controlo de admissão, reserva de recursos, bem como controlo de QoS.

De seguida é feita uma descrição individual dos elementos desta camada lógica.

2.2.2.1 **Multimedia Resource Function Processor (MRFP)**

O elemento MRFP, controlado pelo elemento MRFC já descrito, possui as seguintes funcionalidades e procedimentos de acordo com a norma [22]:

- Juntar os diferentes fluxos de *Media* provenientes dos diferentes utilizadores de uma conferência;
- Fornecer anúncios multimédia e outros conteúdos semelhantes aos clientes IMS;
- Gerir os conteúdos multimédia.

O elemento MRFP pode ser utilizado pelo AS para enviar pedidos dos clientes subscritos a determinados conteúdos e funcionalidades multimédia provenientes deste elemento. Estes pedidos são enviados para o elemento de controlo MRFC através do elemento S-CSCF. Como exemplo, o MRFP permite efectuar a mistura de *media streams* de entrada (ex. conferência). Mistura de diferentes fontes de *Media* (ex. anúncios multimédia) e processamento geral de *media streams* (ex. transcodificação).

2.2.2.2 **IMS – Media Gateway (IMS-MGW)**

No capítulo sobre a camada de controlo de sessão da arquitectura IMS foram descritos os elementos MGCF e SGW como peças fundamentais na convergência da sinalização entre os domínios IP e PSTN/PLMN. O elemento IMS-MGW, por seu lado, situa-se na "fronteira" entre o domínio IP e o domínio PSTN/PLMN e possui as seguintes funcionalidades e procedimentos no plano de dados:

- Conversão entre dados RTP [28] do domínio IP e dados PCM do domínio PSTN/PLMN;
- Interacção com o elemento MGCF da camada de controlo de sessão para controlo de recursos através do protocolo H.248;
- Processamento do áudio (codificação/descodificação, cancelamento do eco, etc.);
- *Transcoding* em situações em que o terminal do utilizador do lado IP não suporta o *codec* utilizado no lado PSTN/PLMN;
- Passagem de eventos (ex. anúncios) para o lado PSTN/PLMN e vice-versa;
- Detecção e processamento de sinalização *Dual Tone MultiFrequential* (DTMF) de acordo com [29].

2.2.3 *Camada de Serviços ou aplicacional*

Camada onde incidiu o trabalho realizado no âmbito desta dissertação. Compreende as aplicações e servidores de conteúdos que executam serviços de valor acrescentado para os utilizadores. Especialmente orientada para servidores aplicacionais standard tais como o *Java 2 Platform, Enterprise Edition* (J2EE) ou o JSLEE, esta camada é onde está contida a lógica de negócio ou de serviço. O IMS especifica uma interface comum baseada em SIP, a *IMS Service Control* (ISC), através da qual as aplicações contidas em servidores de aplicações SIP, Parlay/OSA e *Customized Applications for Mobile networks Enhanced Logic* (CAMEL) interagem com a rede core IMS (camada de controlo). Estes últimos dois *Application Server* (ASs) não são considerados entidades IMS "puras" (apenas o SIP AS é nativo IMS); antes podem ser considerados como funções em cima do IMS, funcionando como adaptadores para este subsistema. Apesar desse facto, são descritos como entidades IMS por fornecerem serviços multimédia de valor acrescentado em IMS.

No anexo A é possível encontrar uma descrição do modo como os ASs e o S-CSCF interagem através da interface ISC [21].

A Figura 7 ilustra a relação entre os diferentes tipos de ASs na arquitectura IMS. É possível observar os três diferentes tipos de servidores aplicacionais, ligados ao S-CSCF através da interface ISC e ligados ao HSS através das interfaces Sh [32] e Si [36] e [37].

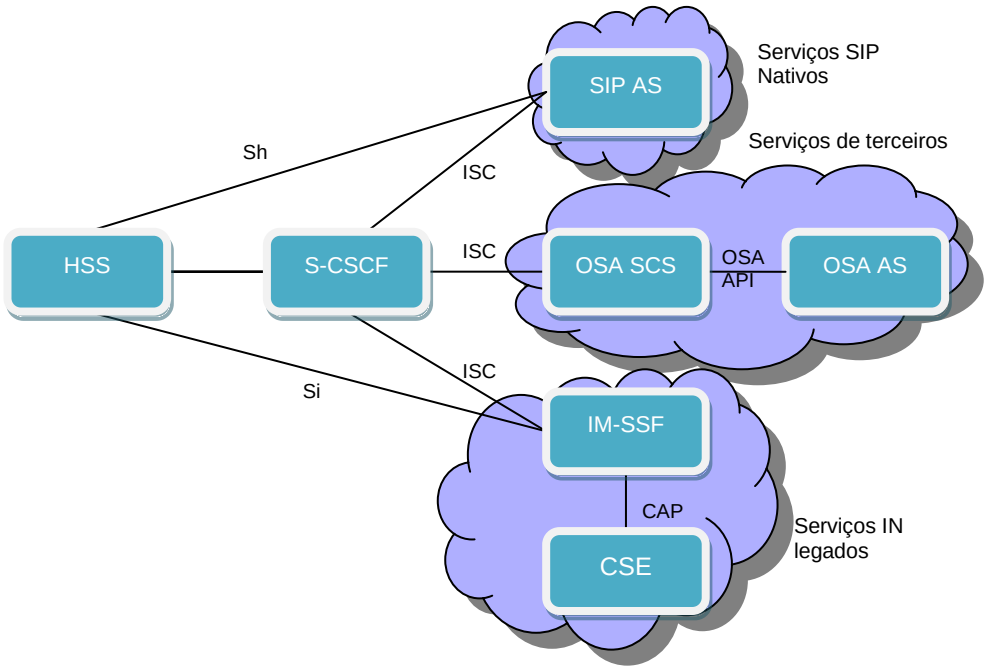

Figura 7 – Relação entre os diferentes tipos de ASs

Entre vários tipos de serviços e aplicações oferecidos por estes elementos destacam-se [30] (posteriormente neste relatório é feito um estudo de alguns destes serviços):

- *Caller ID*, *Call Waiting*, *Call Forwarding*, *Call Transfer*, *Call hold*, *Call pickup*;
- *Conference Call*;
- *Voice mail*;
- *Text-to-speech*, *Speech-to-text*;
- *Push-to-talk* (PTT);
- *Location based services*;
- *Presence*, *Instant Messaging*;
- *Voice Call Continuity* (VCC);
- *Combinational Services* (CSI);
- Etc.

2.2.3.1 SIP AS

Tal como já foi referido, o SIP AS é o único *Application Server* nativo que armazena e executa serviços IMS baseados em SIP. Todos os serviços futuros que estão a ser desenvolvidos para o IMS, serão implementados usando este tipo de AS. Para além da interface SIP com o elemento S-CSCF do plano de controlo, este elemento pode comunicar também com o HSS através do protocolo *Diameter* [31]. A interface utilizada é a Sh [32] e permite consultar ou actualizar a informação respeitante à subscrição de serviços de um determinado cliente (Figura 7).

2.2.3.2 Open Service Access-Service Capability Server (OSA-SCS)

O OSA-SCS [35] é o elemento da camada de aplicação que funciona como uma *interface* de segurança entre as aplicações do OSA AS e os elementos da camada de controlo do IMS. O uso deste elemento permite criar níveis de segurança aos operadores IMS, permitindo que ASs externos (OSA AS) pertencentes a entidades independentes possam utilizar a rede IMS de um determinado operador para fornecer os seus serviços aos clientes dessa rede, sem que a segurança interna desse operador seja comprometida. Este elemento permite esconder toda a sinalização SIP trocada entre os elementos da camada de controlo do IMS, dos servidores de aplicações externos. O OSA-SCS possui uma *interface* SIP [25] com o elemento S-CSCF e uma interface OSA API, especificada em [34], com o servidor de aplicações OSA AS. Este elemento também possui uma *interface* com o HSS através do protocolo *Diameter* [31], interface Sh [32], com uma finalidade idêntica à do SIP AS.

2.2.3.3 IP Multimedia Service Switching Function (IM-SSF)

O IM-SSF é um *Application Server* que fornece uma interface entre o domínio IMS e o ambiente de serviços legados CAMEL. CAMEL é uma iniciativa do 3GPP para estender os tradicionais serviços da *Intelligent Network* (IN), presentes na rede fixa, para a rede móvel. Na rede IN, a "inteligência" é fornecida pelos nós de rede próprios do operador, deferindo de soluções onde a lógica do serviço se encontra no próprio equipamento. É através desta rede que, actualmente, é possível, por exemplo, realizar taxação pré-pago.

Se por um lado, do ponto de vista da *interface* ISC, este elemento funciona como um AS ligado à camada de controlo de sessão S-CSCF, por outro, funciona como um *Session Charging Function* (SCF) através da interface com o elemento gsmSCF baseado no protocolo CAP definido em [36].

Este elemento também possui uma *interface* com o HSS, *interface Si*, através do protocolo baseado em MAP especificado pela norma [37]. Caso o HSS não implemente esta *interface*, é ainda possível ao IM-SSF comunicar com o HSS através da *interface Diameter* Sh [32].

2.3 O protocolo Session Initiation Protocol (SIP)

O protocolo SIP foi escolhido pelo 3GPP para ser o protocolo base da plataforma IMS, e encontra-se definido em [25]. Desenvolvido com o objectivo de serem estabelecidas sessões multimédia entre utilizadores da rede IP, pode ser considerado como a tecnologia

chave por detrás do IMS, já que trata-se do protocolo subjacente a um grande número de interfaces entre os elementos IMS.

Similar ao HTTP, o protocolo SIP é baseado num modelo de pedido/resposta e é um protocolo baseado em texto. Adicionalmente, o protocolo SIP utiliza SIP *Uniform Resource Identifiers* (SIP URI), semelhantes a um endereço de e-mail.

É um protocolo extensível, já que permite facilmente a criação de sessões multimédia com qualquer tipo de *Media*, seja voz, vídeo ou dados. O transporte da informação da descrição da sessão é feito através do SDP, enviado embebido no corpo das mensagens SIP. O SDP transporta a identificação da sessão, a definição do *Media* (áudio, vídeo, etc.) e ainda a informação ligada ao endereçamento para a troca do *Media* (ips, portos).

Este protocolo assegura que a chamada chega ao utilizador, qualquer que seja a sua localização, garantindo mobilidade pessoal.

Em SIP existe um conjunto de métodos (mensagens de sinalização) apresentados na Tabela 1, que permitem iniciar acções (convidar um utilizador para uma chamada, registar-se). As trocas de mensagens SIP utilizadas para registar um utilizador ou para iniciar, terminar ou modificar uma sessão são designadas por transacções. Uma sessão SIP corresponde à chamada tradicional entre dois utilizadores. Uma transacção SIP é constituída por um pedido seguido de uma ou mais respostas informativas, e por uma ou mais respostas finais. Para estabelecer uma sessão SIP é necessário enviar um método específico (INVITE) ao destinatário. O pedido pode ter que passar por um *Proxy Server* que encaminha o pedido para o destinatário caso o destinatário pertença ao seu domínio ou para outro SIP Server caso o destinatário seja de um domínio diferente. O *Proxy Server* responde imediatamente ao emissor com o envio de uma resposta informativa (TRYING). Só quando o utilizador atender o dispositivo é enviado um sinal a indicar o estabelecimento de chamada (OK), esta passa pelo trajecto inverso ao do método INVITE. Para finalizar a transacção de estabelecimento de chamada, o originário envia um ACK que será encaminhado pelo mesmo caminho que o do INVITE.

O transporte das mensagens SIP pode ser feito quer através do protocolo *Transmission Control Protocol* (TCP) quer *User Datagram Protocol* (UDP). Caso seja utilizado o protocolo UDP existe um comportamento adicional de forma a garantir a recepção das mensagens no destino, sendo necessário repetir o envio das mensagens até chegar uma resposta de recepção. As respostas informativas também podem ser confirmadas pelo receptor. É usada para isso a mensagem de PRACK. Esta mensagem é enviada quando a mensagem de resposta informativa transporta explicitamente o pedido de confirmação.

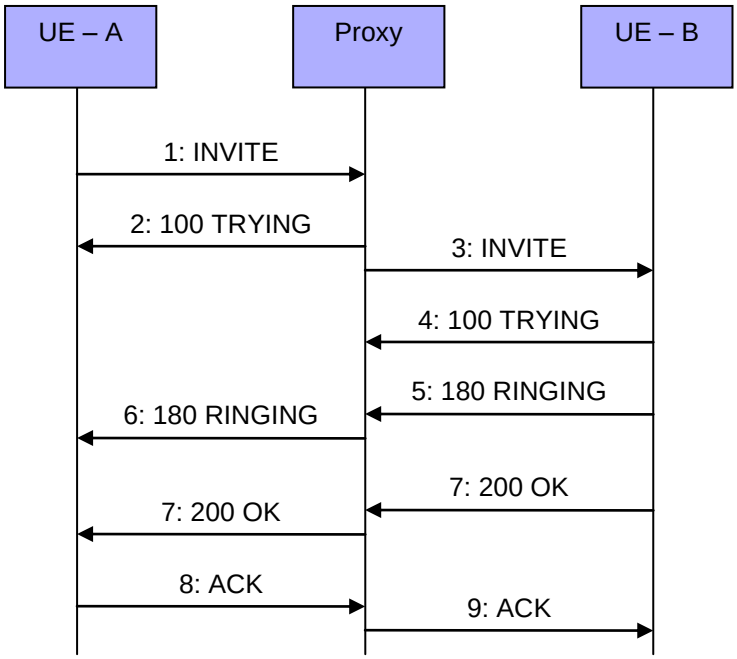

Figura 8 – Trocada de mensagens SIP

A troca de mensagens da Figura 8 é uma visão simplificada da inicialização de uma sessão, onde poderão existir mais componentes envolvidos consoante a estrutura da rede SIP. Esses componentes são, os servidores *Proxy*, *Redirect*, *Registrar* e *Location Server*.

2.3.1 *Componentes SIP*

***User Agent* (UA)** – Corresponde ao equipamento terminal SIP do utilizador e integra como componente SIP o *User Agent* (UA) que inicia ou aceita as chamadas. Recebe e envia pedidos de forma a estabelecer sessões, ou seja, comporta-se como um *User Agent Server* (UAS) ou como um *User Agent Client* (UAC), respectivamente.

Proxy Server – O Proxy Server recebe os pedidos dos UAC e encaminha-os de acordo com o Request URI (URI contido no cabeçalho do método SIP), e alguns outros cabeçalhos. O *Proxy Server* encaminha o pedido SIP para o domínio a que pertence o utilizador chamado.

Redirect Server – Um *Redirect Server* recebe pedidos mas não emite nenhum pedido. Sempre que recebe um pedido responde com uma mensagem 3xx (Tabela 2). Esta mensagem contem uma ou mais localizações do utilizador destinatário.

Location Server – O *Location Server* é usado pelo *Registrar* para localizar o utilizador chamado, bem como guardar informação dos utilizadores registados.

***Registration Server* (SIP REGISTRAR)** – O *Location Server* é actualizado através do *Registration Server*. Quando um utilizador fica activo num terminal envia para o servidor *Registration Server* uma mensagem de registo onde é transportada a informação da sua localização actual e do respectivo período de validade. Este servidor autentica o utilizador

e regista a sua localização e período de validade do registo no *Location Server*. São os procedimentos de registo que permitem a mobilidade dos utilizadores na rede SIP.

***Back 2 Back User Agent* (B2BUA)** – Um B2BUA é uma entidade lógica que recebe pedidos, processa-os como um UAS e, com o objectivo de determinar como um pedido deve ser respondido, actua como um UAC e gera pedidos. Um B2BUA tem de manter estado das chamadas e participar activamente no envio de pedidos e respostas a diálogos nos quais se encontra envolvido (um diálogo representa uma relação ponto-a-ponto entre dois *User Agents* que se mantém por algum tempo). Um B2BUA possui mais controlo sobre uma chamada que um Proxy, e pode, por exemplo, desligar uma sessão sem a intervenção dos utilizadores. Importante quando pretendemos tarifar um utilizador que tenha um serviço pré-pago e que seja terminada a sessão quando este fica sem saldo.

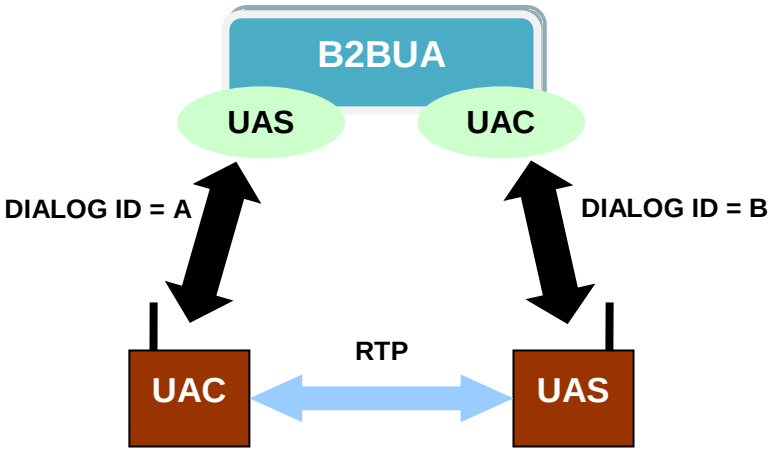

Figura 9 – Exemplo de um B2BUA

O B2BUA da Figura 9 comporta-se como um UAS com o utilizador que iniciou a sessão, mantendo o diálogo A, e como UAC com o utilizador destinatário da chamada, mantendo o diálogo B.

2.3.2 *Mensagens SIP*

Como já foi referido o funcionamento do protocolo SIP baseia-se em métodos (mensagens de sinalização), que iniciam acções originadoras de mensagens de resposta. De seguida apresenta-se na Tabela 1 os métodos base.

Tabela 1 – Conjunto de métodos SIP

INVITE	Inicia uma sessão, ou muda os parâmetros de uma sessão já existente (re-INVITE).
ACK	Enviado como confirmação a uma resposta final de um INVITE.
BYE	Terminação da sessão.

CANCEL	Cancelamento da sessão em estabelecimento.
REGISTER	Efectua o registo, ou desregisto do utilizador.
OPTIONS	Indicação das capacidades disponibilizadas nos UA.
INFO	Envia informação durante uma sessão que não modifica o estado da sessão (por exemplo, dígitos DTMF gerados durante uma sessão).
PRACK	Confirmação de respostas provisórias (Este pedido tem um papel idêntico ao ACK mas para respostas provisórias).
UPDATE	Permite o update de parâmetros de sessão (como por exemplo os *codecs* do *Media*) não tendo impacto no estado do diálogo. Semelhante ao re-INVITE mas ao contrário do re-INVITE pode ser enviado sem que o INVITE inicial receba uma resposta final.
REFER	Este método indica ao receptor (identificado pelo Request-URI) que tem de contactar uma terceira entidade usando a informação enviada neste pedido (usada na implementação de serviços suplementares conferência e transferência de chamadas).
SUBSCRIBE	Pedido de notificação de um evento (recepção de um *e-mail*, estado de presença, por ex.)
NOTIFY	Notificação de um evento.
PUBLISH	Usado para publicar um estado. Similar ao REGISTER pois permite que o utilizador crie, modifique e remova estados numa entidade que gere esses estados pelo utilizador.
MESSAGE	Usado para envio de conteúdo sobre a forma de texto no corpo da mensagem.

Para cada método recebido, o destinatário pode responder com respostas pertencentes a um conjunto de seis classes. A resposta é identificada por um identificador da mensagem de 3 dígitos, onde o dígito das centenas identifica a classe. As classes de 1 a 6 agrupam as respostas Tabela 2.

Tabela 2 – Conjunto de Respostas SIP

1xx	Provisória	Pedido recebido, continuando a processar o pedido. Ex: 180 Ringing
2xx	Sucesso	O pedido foi recebido, percebido e aceite com sucesso. Ex: 200 OK
3xx	Redireccionamento	É necessária a realização de outras acções para processar completamente o pedido. Ex: 302 Moved Temporarily
4xx	Erro do Cliente	O pedido contem sintaxe errada ou não pode ser completamente servido por este servidor. Ex: 404 Not Found.
5xx	Erro do Servidor	O servidor falhou ao servir um pedido aparentemente válido. Ex: 504 Server Time-out
6xx	Falha Global	O pedido não pode ser processado em nenhum servidor. Ex: 603 Decline

Tal como no HTTP, as mensagens SIP são formadas por cabeçalhos e pelo conteúdo da mensagem. Os cabeçalhos da Tabela 3 são cabeçalhos que podem estar presentes nos pedidos (*Requests*) e nas respostas (*Responses*).

Tabela 3 – Campos gerais de uma mensagem SIP

TO e FROM:	Destino e origem de uma mensagem respectivamente. Apenas são usados para consulta, não são alterados ao longo do caminho da mensagem e não são usados no encaminhamento.
Request-URI:	Indica o endereço destino a chamar.
CALL-ID:	Actua como um identificador único e tem de ser o mesmo para todos os pedidos e respostas enviados por qualquer UA num diálogo.
CONTACT:	Fornece o URI que indica a localização do utilizador que efectuou o pedido para que o destino o possa contactar de volta em futuros pedidos.
MAXFORWARDS:	Serve para limitar o número de saltos que um pedido poder efectuar até chegar ao destino. Quando o valor chega a zero a mensagem é descartada.
SUBJECT:	Informação auxiliar.
CONTENT-LENGTH:	Indica o tamanho do corpo da mensagem.
VIA:	Por cada servidor SIP que a mensagem passa, é adicionado o endereço desse servidor neste cabeçalho para que, as respostas percorram o mesmo trajecto no sentido inverso. Ao passar uma resposta por cada servidor SIP é removido o endereço que ele inseriu.
CSEQ:	Tem como função identificar e ordenar transacções. Consiste num número de sequência e num método.

Existem cabeçalhos que são usados apenas nos pedidos (*Requests*), estes são usados pelos UA para modificar ou dar informação adicional acerca de um pedido Tabela 4.

Tabela 4 – Cabeçalhos de Pedidos SIP

RECORD-ROUTE:	Este campo pode ser usado por um Proxy Server para pedir ao UA que envie os próximos pedidos através desse Proxy Server. O campo VIA é idêntico, mas para respostas. Este é válido por uma sessão.
ROUTE:	Cabeçalho usado para definir o caminho a seguir por uma mensagem.

Dois exemplos do papel do Route e do Record-Route estão presentes na Figura 10 e na Figura 11. Em ambos os casos, e dado que o utilizador A está a tentar ligar para o utilizador B, para as mensagens passarem pela *proxy* o utilizador A terá de adicionar no cabeçalho Route o endereço da *proxy*. Só assim a mensagem é reencaminhada pela *proxy*. O que difere nas duas imagens é a proxy fazer, ou não, Record-Route. No primeiro caso,

situação onde a *proxy* faz Record-Route é possível observar que as restantes transacções entre os utilizadores continuam a ser reencaminhadas pela *proxy*. No segundo caso, pelo contrário, apenas as respostas ao primeiro pedido passam pela *proxy* (já que isso depende do cabeçalho Via e não do Record-Route) e as restantes transacções já não são encaminhadas pela *proxy*.

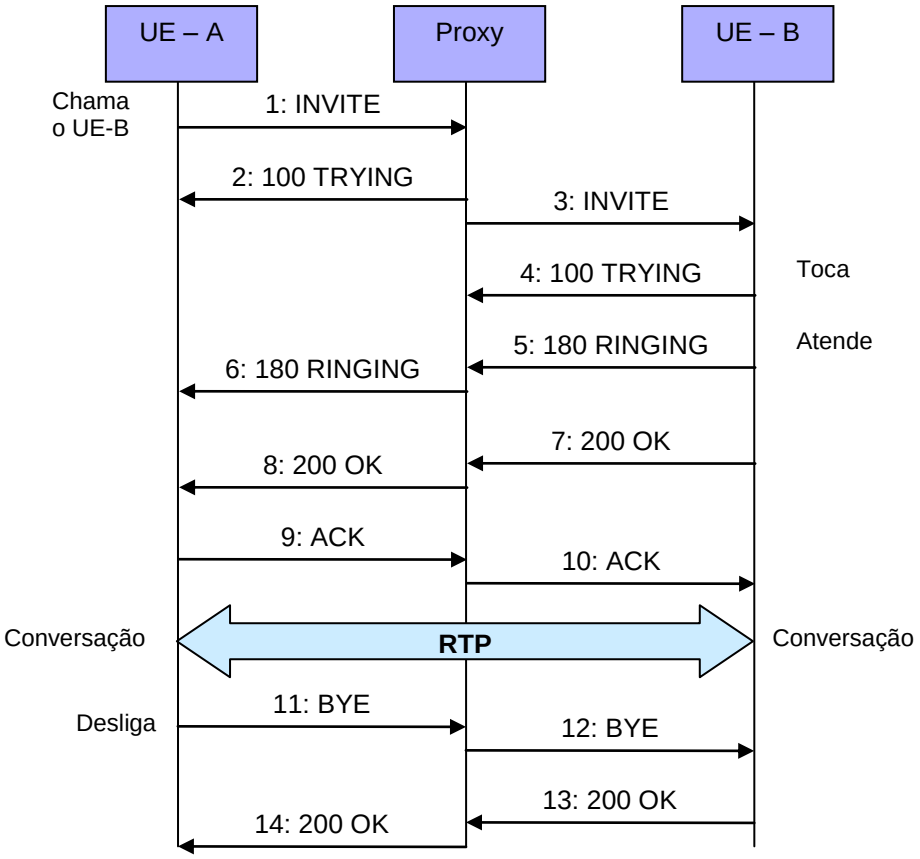

Figura 10 – Fluxo de chamada SIP com Proxy (faz *Record-Route*)

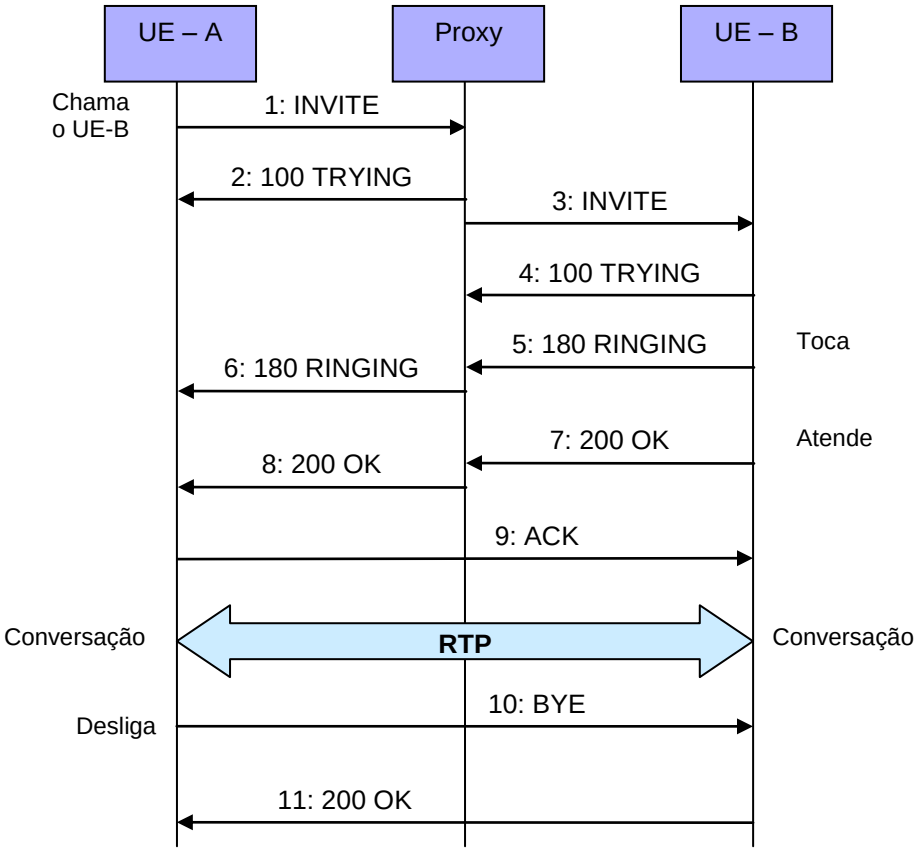

Figura 11 – Fluxo de chamada SIP com Proxy (não faz *Record-Route*)

2.3.3 *Endereçamento SIP*

Os *Uniform Resource Identifiers* (URI) [38] são utilizados na rede SIP para localizar/identificar os utilizadores. Para chamadas estabelecidas dentro da rede IP o protocolo SIP usa o *Domain Name System* (DNS) e o encaminhamento IP para fazer seguir os pedidos.

No caso de chamadas entre a rede IP e a PSTN há a necessidade de mapear os números da rede PSTN na rede SIP. Como tal, duas alternativas estão disponíveis. A primeira é utilizar um TEL URL [39] e a segunda é utilizar um SIP URI, contendo a parte *username* no *formato* E.164 e a *tag* "*user=phone*". Quando a *gateway* recebe uma chamada vinda da PSTN, é feita uma correspondência do endereço telefónico no endereço SIP. Este processo é realizado através de um servidor ENUM [77], que utilizando as capacidades DNS e um domínio próprio para números telefónicos, realiza esta conversão. Seguem-se alguns exemplos:

- SIP URI – sip:luisilva@ptinovacao.pt
- SIP URI – sip:+351234232453@ptinovacao.pt;user=phone
- TEL URL – tel:+351234232453

2.3.4 *Descoberta de endereços de Servidores SIP*

Para o equipamento do utilizador estabelecer uma ligação necessita de saber o endereço IP e a porta dos servidores SIP que o servem. O endereço deste pode ser configurado manualmente ou descoberto dinamicamente através de um servidor DHCP. O *Registration Server* (Registrar) é também descoberto com o mesmo tipo de mecanismos. Sendo conhecido o endereço do *Proxy Server*, pode ser estabelecida uma sessão com outro utilizador (UA) do domínio local. Mas caso o utilizador chamado seja de outro domínio, o *Proxy Server* local necessita de descobrir o endereço do *Proxy Server* remoto para onde a chamada deve ser encaminhada. Na descoberta dinâmica do *Proxy Server* remoto é usado o mecanismo do DNS [78]. Inicialmente é feita uma *query* do tipo *Naming Authority PoinTeR* (NAPTR) para o domínio em questão (por exemplo, "ptinovacao.pt"). Como resposta, é obtido o tipo de transporte associado a esse domínio (por exemplo, TCP, UDP). Supondo que o domínio suporta múltiplos transportes, é necessário encontrar o transporte mais prioritário. Para tal, é incluída na resposta um elemento que define essa prioridade. Além da prioridade, ainda é importante filtrar os transportes não suportados pelo cliente que realiza a descoberta dinâmica. Após a escolha, e tendo já o transporte, é necessário identificar o *host*. É então realizada uma segunda *query*, agora do tipo DNS SRV *(Location of Services*) [44], com o resultado obtido da primeira *query* (que irá ser algo semelhante a "_sip._tcp.ptinovacao.pt", onde neste exemplo, o transporte com mais prioridade e suportado pelo cliente é TCP sem TLS). A resposta à segunda *query* devolve o porto e o *canonical hostname* da máquina a descobrir. Assim, falta apenas traduzir o *hostname* num endereço IP. Este processo é realizado através de uma *query* do tipo *A*. É possível na *query* SRV o servidor DNS devolver mais do que um porto e endereço, sendo atribuído a cada par endereço/porto um determinado peso. Este mecanismo irá possibilitar a realização de balanceamento de carga/pedidos através de DNS.

2.3.5 *Trapézio SIP*

O trapézio SIP presente na Figura 12, é uma representação habitualmente utilizada para ilustrar o processo de inicialização de uma sessão SIP entre dois utilizadores pertencentes a redes (domínios) diferentes.

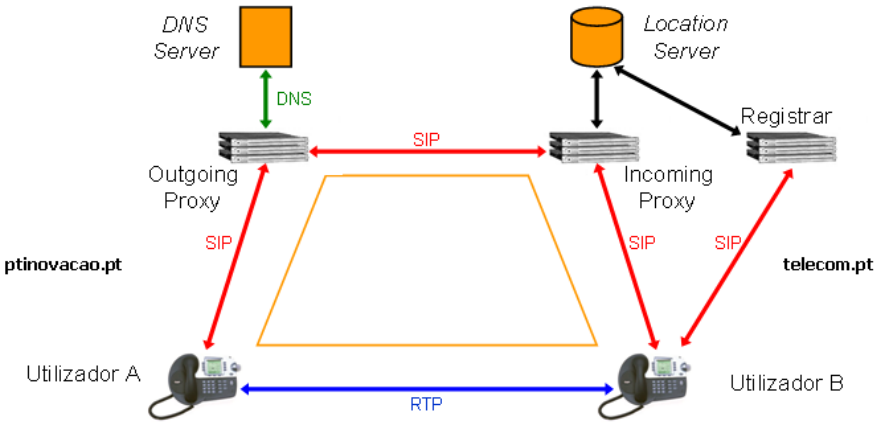

Figura 12 – Trapézio SIP/Utilização do DNS para descobrir um utilizador noutra rede

1. Utilizador A inicia uma sessão com utilizador B através do seu *Outgoing Proxy*;
2. Servidor *Proxy* verifica que o domínio destino não é o seu e através de uma consulta ao DNS procura o SIP *Proxy* responsável pelo processamento daquela sessão;
3. Ao receber o pedido, o SIP Proxy do utilizador B (aqui assume o papel de *incoming proxy*) consulta o *Location Server* para perceber a localização do utilizador B (localização previamente inserida pelo Registrar server);
4. O SIP Proxy do domínio "telecom.pt" comunica com o utilizador B;
5. RTP entre os utilizadores é enviado *peer-to-peer*;

2.4 O protocolo Diameter

O protocolo *Diameter* [31] é um protocolo de *Authentication*, *Authorization* e *Accounting* (AAA) especificado como um protocolo base (*Diameter Base Protocol*) e um conjunto de aplicações *Diameter* que complementam as funcionalidades do protocolo base. O *Diameter Base Protocol* contém as funcionalidades básicas e é implementado em todos os nós *Diameter* independente de *qualquer* aplicação. O *Diameter Base Protocol* é usado para entregar unidades de dados *Diameter*, para permitir capacidades de negociação e manipulação dos erros. Por outro lado, as aplicações *Diameter* são extensões às funcionalidades base e são talhadas para casos particulares do *Diameter* em ambientes particulares, definindo funções e dados específicos, como por exemplo as aplicações para *Mobile IP* [45], *Diameter Credit-Control Application* [42] e a *Diameter SIP Application* [41].

A Figura 13 pretende ilustrar a relação existente entre o *Diameter Base Protocol* e as aplicações que possam correr sobre a camada base. Estas aplicações podem ter quatro

diferentes modos de operação, estando divididas em aplicações cliente/servidor para Autenticação/Autorização e Accounting (Figura 39).

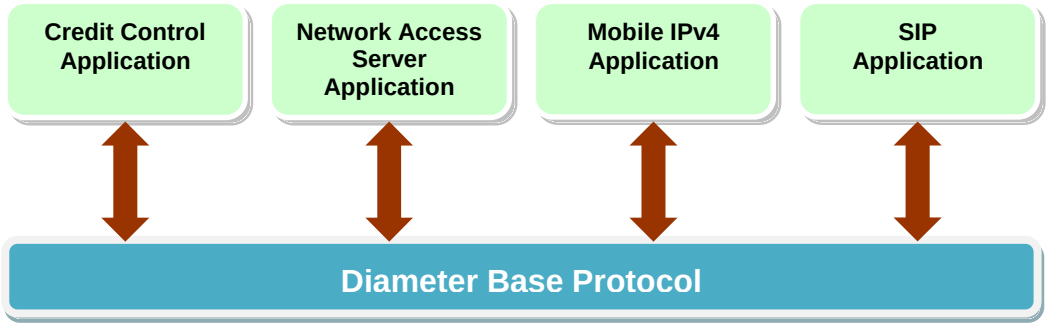

Figura 13 – *Diameter* Base Protocol e aplicações

O *Diameter Base Protocol* permite-nos o seguinte:

- Entrega de *Attribute Value Pairs* (AVPs);
- Troca de *Capabilities* (Capacidades de cada Servidor);
- Notificação de erros;
- Extensibilidade (possível adicionar novos *Command Code* e novos AVPs);
- Serviços Básicos como sessões de autorização e *accounting*.

Todos os dados entregues têm o formato de AVP, alguns destes AVPs também são usados por parte do protocolo *Diameter*, diga-se Base, enquanto que outros são apenas usados por aplicações *Diameter*.

Os AVPs que são usados pelo protocolo *Diameter* têm a seguinte finalidade:

- Transportar informação para autenticação;
- Transportar informação de autorização utilizada para permitir criar sessões;
- Troca de informação de recursos;
- Encaminhar mensagens *Diameter*.

O protocolo *Diameter* fornece os requisitos mínimos necessários para um protocolo AAA. O *Diameter Base Protocol* pode ser usado por si só para questões de *accounting* ou em conjunto com aplicações *Diameter*. Novas aplicações são facilmente criadas, através da combinação de novos ou existentes comandos (mensagens trocadas) ou AVPs.

2.4.1 *Componentes Diameter*

O protocolo *Diameter* define diferentes entidades funcionais com o objectivo de realizar as funções de AAA. Esses elementos são chamados nós *Diameter* e correspondem a um ponto onde é possível comunicar através do protocolo *Diameter*. Cada nó pode iniciar um pedido

Diameter, o que faz deste protocolo um protocolo ponto-a-ponto. Os nós podem ser do tipo Cliente *Diameter*, Agente *Diameter* ou Servidor *Diameter*.

Cliente Diameter: Entidade funcional, que tipicamente se encontra na extremidade da rede e que exerce o controlo de acesso como um *Network Access Server* (NAS). O cliente *Diameter* gera pedidos *Diameter* para requerer o uso de serviços (aplicações) de autenticação, autorização e *accounting*.

Servidor Diameter: Entidade que trata de realizar a autenticação e/ou autorização do utilizador, permitindo assim o acesso às aplicações por ele disponibilizadas. No entanto, pode também ser ele a iniciar pedidos *Diameter* (por exemplo, terminar uma sessão). Os nós podem actuar simultaneamente como agentes e servidores.

Agentes Diameter (Relay, Proxy, Redirect e Translation):

Agentes *Relay*: Entidade funcional que recebe pedidos *Diameter* e reencaminha-os para outros nós *Diameter* através da informação presente na mensagem (por exemplo, *Destination-Realm*). A decisão de para onde deve reencaminhar é feita utilizando uma lista de *realms* suportados, e de *peers* conhecidos. Esta lista é conhecida como *Routing Table* [31]. Um *relay* é tipicamente transparente, podendo modificar as mensagens através da inserção ou remoção da informação de *routing*, mas não poderá modificar os restantes dados da mensagem (Figura 14).

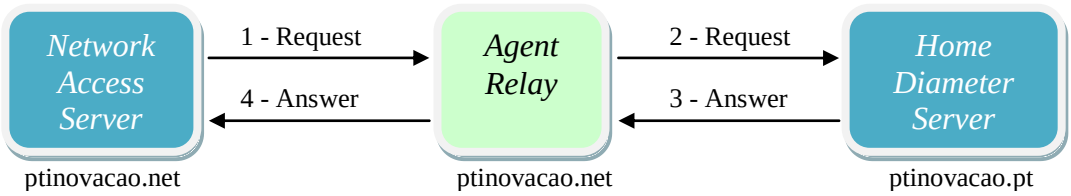

Figura 14 – Agente *Relay*

Agentes *Proxy*: Similares aos *relays*, os agentes *proxy* reencaminham as mensagens *Diameter* usando a *Routing Table*. Contudo, estes elementos diferem, pois podem modificar as mensagens de forma a implementar decisões de *policy* (por exemplo, o controlo da utilização dos recursos, controlo de acesso e aprovisionamento).

Agentes *Redirect*: Entidades funcionais que, ao contrário dos agentes *proxy* e *relay*, apenas respondem aos elementos *Diameter* com informação que os permita comunicarem directamente (Figura 15).

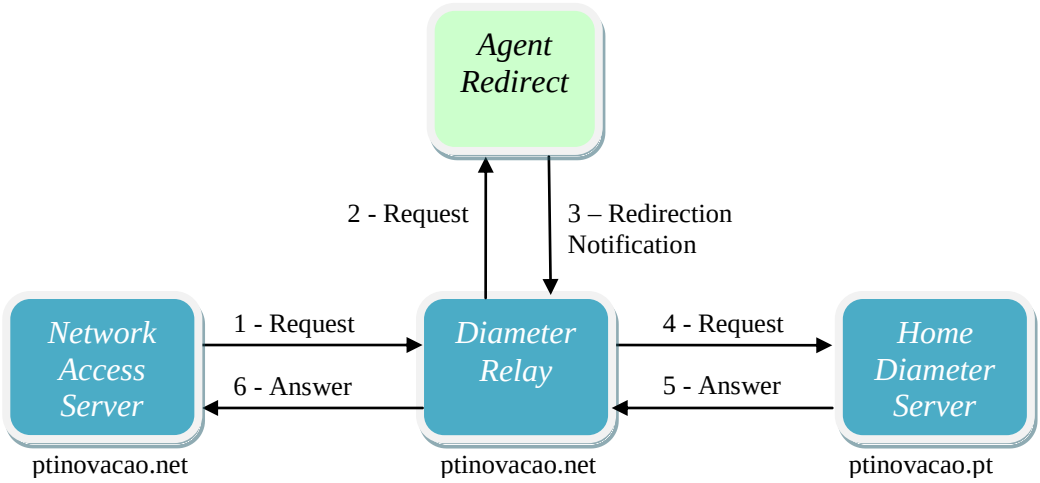

Figura 15 – Agente *Redirect*

Agentes *Translation*: Entidades funcionais que fornecem a tradução de protocolos entre o *Diameter* e outros protocolos AAA, como por exemplo *Radius*[4] (Figura 16).

Figura 16 – Agente *Translation*

O sistema de transporte pode ser TCP ou SCTP para os clientes, no entanto os agentes e servidores tem que suportar ambos protocolos. Entre os nós deve ser feita uma conexão através da porta 3868 (porto por defeito). Aqui existe a noção de sessão, que é um conceito lógico da camada aplicacional. Segue-se uma figura representativa deste conceito.

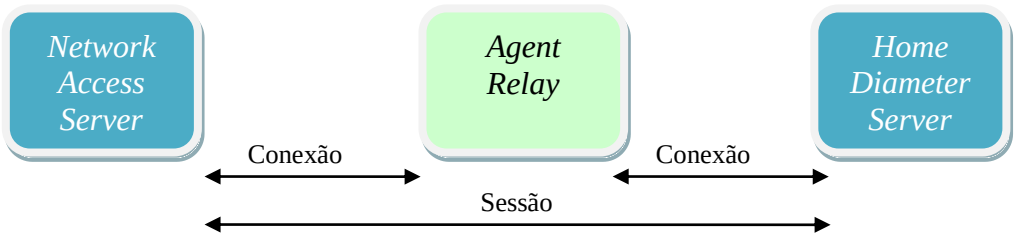

Figura 17 – Sessões e Conexões *Diameter*

2.4.2 *Mensagens Diameter*

A Figura 18 mostra o formato da mensagem *Diameter*. A mensagem *Diameter* consiste num cabeçalho de 20 octectos e um número de *Attribute Value Pairs* (AVPs). O tamanho

[4] *Radius* é considerado o antecessor do protocolo *Diameter*

do cabeçalho é sempre fixo, e terá de estar sempre presente na mensagem *Diameter*. Pelo contrário, o número de AVPs é variável e depende do tipo de mensagem *Diameter*. Um AVP é um "repositório" de dados (tipicamente informação de autenticação, autorização e *accounting*).

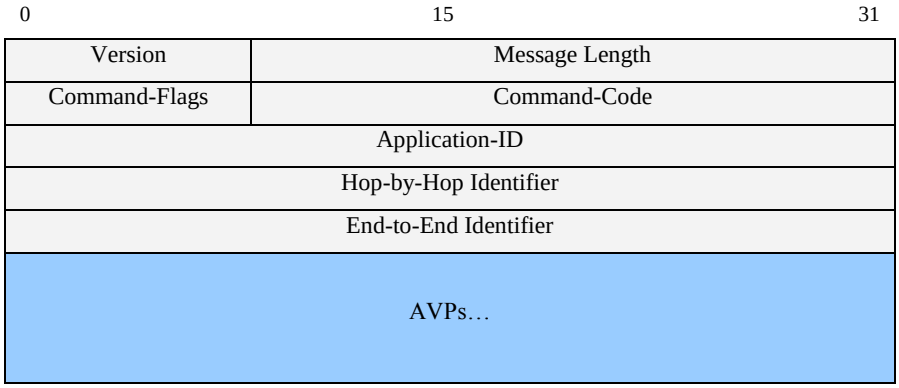

Figura 18 – Formato da mensagem *Diameter*

O cabeçalho *Diameter* contém os seguintes campos:

- *Version* – deve estar preenchido a 1 (versão do protocolo).
- *Message Length* – Tamanho da Mensagem.
- *Command Flag – Flag* a indicar o tipo de mensagem (*Request*, *Proxiable*, *Error*).
- *Command Code* – Código do comando (mensagem) de uma determinada aplicação.
- *Application-Id* – Identificador da aplicação a usar.
- *Hop-by-Hop Identifier* – utilizado para associar as respostas aos pedidos.
- *End-to-End Identifier* – utilizado para detectar mensagens duplicadas.
- *AVPs* – conjunto de Avps da mensagem.

2.4.3 *Attribute Value Pairs (AVPs)*

A mensagem *Diameter* transporta um conjunto de *Attribute Value Pairs* (AVPs), sendo este um "repositório" de dados. A Figura 19 ilustra a estrutura do AVP.

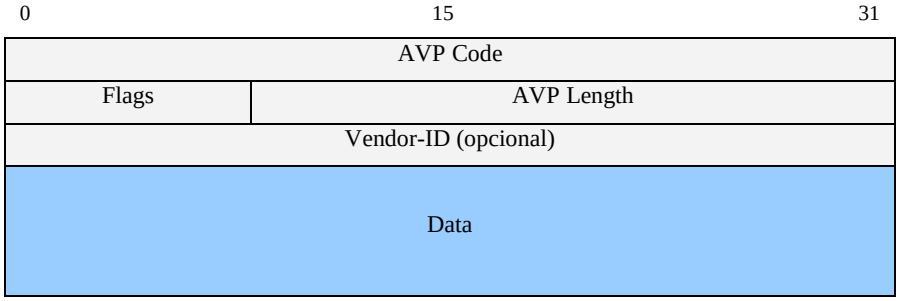

Figura 19 – Estrutura de um AVP *Diameter*

Os AVPs transportam informação específica às aplicações em causa. A composição de um AVP é a seguinte:

- *AVP Code* – corresponde ao identificador do AVP;
- *AVP Flags* – serve para informar o receptor da mensagem como o atributo de ser tratado;
- *AVP Length* – dimensão do AVP;
- *Vendor-Id* – identificador do *vendor*;
- *Data* – dados do AVP;

O *AVP Code*, juntamente com o campo *Vendor-ID*, identificam de forma única o AVP. A ausência do campo *Vendor-ID* ou no caso de este se encontrar com o valor zero indica que o AVP é normalizado e encontra-se especificado numa especificação do IETF.

2.4.4 *The AAA and AAAS URIs*

Os protocolos AAA utilizam um *aaa* ou *aaas* URI para identificar recursos AAA. O *aaas* URI indica que deverá ser utilizada segurança no transporte. A sintaxe destes URIs é a seguinte:

"aaa://" FQDN [port] [transport] [protocol]

"aaas://" FQDN [port] [transport] [protocol]

port = ":" 1*DIGIT

transport = ";transport=" transport-protocol

protocol = ";protocol=" aaa-protocol

transport-protocol = ("tcp" / "sctp" / "udp")

aaa-protocol = ("diameter" / "radius" /"tacacs+")

onde FQDN é um *Fully Qualified Domain Name*. Os URIs poderão ser ligados por um porto opcional, um protocolo de transporte opcional ou um protocolo opcional para aceder aos recursos AAA. Caso o porto não se encontre presente, o valor assume o valor do porto por defeito do *Diameter* (3868). Da mesma forma, caso o parâmetro de transporte não se encontre presente, então o protocolo SCTP é assumido por defeito. Por último, caso o parâmetro do protocolo não esteja presente, o *Diameter* é assumido.

Exemplos incluem:

aaa://server.homel.net

aaas://server.home1.net

aaa://server.homel.net:3868

aaa://server.homel.net;transport=tcp;protocol=diameter

2.5 **Serviços IMS**

Os serviços IMS são serviços disponíveis aos utilizadores que utilizam o IMS como uma plataforma de fornecimento e controlo de serviços. Numa visão global, o termo "serviço IMS" é utilizado com o mesmo sentido do termos "serviço IN" ou "serviço CAMEL".

Esta classe de serviços inclui os serviços *Multimedia Telephony*, serviços de *messaging*, serviços de transição, serviços de presença entre outros.

O serviço IMS *Multimedia telephony* engloba os serviços básicos de voz/multimédia como ainda os serviços de simulação suplementares. Mecanismos de inicialização de chamadas, reencaminhamentos até ao destinatário, reserva de recursos e a possível invocação de serviços nos AS estão incluídos.

Os serviços IMS ainda incluem serviços que combinam as funcionalidades IMS com os serviços de rede de comutação de circuitos e pacotes, isto é, permitir a entrega dos serviços de forma consistente independentemente do tipo de acesso de rede. Como tal, para garantir uma correcta transição para o IMS, o 3GPP definiu diferentes serviços que permitem que as funcionalidades já existentes nas redes actuais possam ser reutilizadas nesta nova arquitectura. O serviço de *Voice Call Continuity* (VCC) é uma aplicação IMS capaz de transferir chamadas de voz entre dois domínios: a rede de comutação circuitos (por exemplo, GSM ou CDMA) e a rede de comutação de pacotes (por exemplo, WIFI ou WIMAX). Através desta aplicação o usuário obtém uma experiência contínua e transparente enquanto a sua chamada é transferida entre domínios. O serviço *Combinational Services* (CSI) permite a combinação simultânea de serviços no domínio CS e no IMS entre dois utilizadores. Com o auxilio do CSI, os utilizadores poderão realizar uma chamada de voz através rede legada, podendo criar uma sessão de vídeo através do IMS para a mesma sessão, utilizando deste modo, os recursos da rede CS com os do domínio IMS.

2.5.1 *Multimedia Telephony*

O serviço de comunicação IMS *Multimedia Telephony* [46] permite a comunicação multimédia entre dois ou mais pontos terminais. Tipicamente os pontos terminais estão localizados nos equipamentos dos utilizadores, mas também podem encontrar-se em elementos da rede.

O serviço IMS *Multimedia Telephony* consiste em duas partes principais, uma relativa às comunicações mais básicas e outra, opcional, relativa aos serviços suplementares.

A comunicação básica de uma sessão de comunicação IMS *Multimedia Telephony* é realizada através de uma sessão SIP única. Utiliza os mecanismos de *Media* e a flexibilidade proporcionada pelo protocolo SIP e as especificações do 3GPP. Estes mecanismos, e de acordo com [46], incluem transferência de voz, vídeo e texto em tempo-real baseada em RTP, como ainda transferência de texto, ficheiros e partilha de ficheiros de *Media* com formatos predefinidos baseada em TCP/*Message Session Relay Protocol*[5] (MSRP). Para assegurar interoperabilidade, os *codecs* de *Media* e formatos, tanto para transferências baseadas em RTP como MSRP, estão especificados em [47]. O serviço é extremamente dinâmico em termos de utilização dos componentes de *Media*: os protocolos permitem que uma sessão de uma comunicação inicie com um ou mais componentes de *Media*, e os componentes poderão ainda ser adicionados e/ou removidos durante a sessão. Os protocolos permitem transferências entre os dois pontos tanto num sentido como nos dois. Os casos típicos são a comunicação bidireccional de voz e a voz combinada com outros componentes de *Media*, mas os protocolos não obrigam a utilizar a voz em todas as sessões.

A parte de serviços suplementares do serviço IMS *Multimedia Telephony* consiste num número de serviços suplementares especificados. Estes últimos encontram-se normalizados para garantir interoperabilidade entre múltiplos pontos terminais, e entre pontos terminais e os elementos de rede da camada de controlo, O comportamento dos serviços suplementares é semelhante aos serviços suplementares especificados para a comunicação de voz no domínio das redes de comutação de circuitos.

2.5.1.1 Serviços Suplementares

2.5.1.1.1 Originating Identification Presentation (OIP)

O serviço *Originating Identification Presentation* (OIP) [69] fornece ao utilizador destinatário da chamada, a possibilidade de receber a informação de identificação do utilizador que iniciou a chamada. Esta informação entregue ao destinatário irá permitir identificar o originador da chamada de forma segura, já que esta é fornecida pela própria rede.

2.5.1.1.2 Originating Identification Restriction (OIR)

O serviço *Originating Identification Restriction* (OIR) [69] permite ao originador da chamada restringir a apresentação da sua informação ao utilizador destinatário.

[5] Protocolo que permite, por exemplo, o envio de mensagens instantâneas mas no contexto de uma sessão. As

Quando o serviço é OIR é aplicável e activo, a rede do utilizador originário fornece à rede do utilizador destinatário uma indicação que a informação do utilizador não deverá ser apresentada ao destino da chamada.

2.5.1.1.3 Terminating Identification Presentation (TIP)

O serviço *Terminating Identification Presentation* (TIP) [70] fornece ao utilizador que iniciou a chamada a possibilidade de obter a informação da identidade do utilizador que irá receber a chamada. Numa situação normal, este serviço basicamente não tinha qualquer fundamento, mas no caso de, por exemplo, o utilizador destinatário reencaminhar a chamada, o utilizador que a originou não tem forma de obter a identidade do utilizador para onde a chamada foi reencaminhada a chamada, a não ser que tenha este serviço activo.

2.5.1.1.4 Terminating Identification Restriction (TIR)

O serviço *Terminating Identification Restriction* (TIR) [70] permite restringir a apresentação da informação dos destinatários ao originador da chamada. Este serviço só é válido em situações em que o destinatário da chamada não é o mesmo destino marcado pelo utilizador que iniciou a chamada, por exemplo, no caso de reencaminhamento.

2.5.1.1.5 Communication Diversion (CDIV)

O serviço *Communications Diversion* (CDIV) [73] permite reencaminhar chamadas destinadas a um utilizador para outro destinatário.

Este serviço encontra-se dividido em cinco casos particulares:

- *Communication Forwarding Unconditional* (CFU): O serviço CFU permite que a rede reencaminhe as chamadas destinadas a um utilizador, independentemente do estado do utilizador;
- *Communication Forwarding on Busy user* (CFB): O serviço CFB permite que a rede reencaminhe as chamadas destinadas a um utilizador quando este rejeita a chamada;
- *Communication Forwarding on no Reply* (CFNR): O serviço CFNR permite que a rede reencaminhe as chamadas destinadas a um utilizador que não atende nem rejeita a chamada, sendo reencaminhada após um período de tempo;

mensagens são tratadas como se fossem streams de *Media*.

- *Communication Deflection* (CD): O serviço CD permite que seja o próprio utilizador, para onde as chamadas são destinadas, indicar à rede (AS) para reencaminhar a chamada;
- *Communication Forwarding on Not Logged-in* (CFNL): O serviço CFNL permite que a rede reencaminhe as chamadas destinadas a um utilizador quando este não se encontra registado na rede;

2.5.1.1.6 Communication Hold (HOLD)

O serviço *Communication* HOLD [72] permite ao utilizador suspender uma *Media stream*(s) numa sessão multimédia já estabelecida como ainda permite resumir *Media streams*(s) de sessões suspensas.

2.5.1.1.7 Communication Barring (CB)

O serviço de *Communication Barring* (CB) [76] permite barrar chamadas estando dividido em três diferentes casos:

- O serviço *Incoming Communications Barring* (ICB) permite rejeitar chamadas que cheguem ao utilizador, através de condições pré-configuradas na rede do utilizador terminal;
- O *Anonymous Communication Rejection* (ACR) é um caso particular do serviço ICB, ao permitir barrar chamadas recebidas quando o utilizador que originou a chamada é "Anónimo".
- O serviço *Outgoing Communication Barring* (OCB) permite rejeitar chamadas em nome do utilizador que iniciou a chamada através de condições pré-configuradas na rede deste;

2.5.1.1.8 Message Waiting Indication (MWI)

O serviço *Message Waiting Indication* (MWI) [75] permite a um utilizador ser notificado quando recebe mensagens na sua caixa de correio. É possível configurar dois modos de funcionamento. No primeiro modo, o utilizador "subscreve-se" à sua caixa de correio através da mensagem SIP SUBSCRIBE, evento *message-summary* (caso suporte). Após esta subscrição, o utilizador é notificado sempre que uma nova mensagem é depositada na sua caixa de correio (esta notificação é realizada pelo serviço através do envio de uma mensagem SIP NOTIFY, contendo a informação da mensagem nova recebida). O outro processo existente que é obrigatório para os clientes que não suportem o mecanismo SUBSCRIBE/NOTIFY para o evento *message-summary*, é efectuado através de uma

mensagem SIP MESSAGE com a informação da mensagem recebida. O serviço terá de verificar o estado de registo do utilizador antes de enviar a mensagem (através, por exemplo, da interface Sh com o HSS ou de um *trigger* para esse serviço quando o utilizador se regista), pois o utilizador terá de se encontrar registado.

2.5.1.1.9 Conference (CONF)

O serviço de Conferência [74] permite ao utilizador participar e controlar comunicações simultâneas envolvendo um número de utilizadores.

Quando o serviço é invocado, os recursos são reservados para o utilizador a ser servido. Após a conferência ser activa, os utilizadores podem juntar-se ou sair da conferência, e utilizadores remotos podem ser adicionados ou removidos.

2.5.1.1.10 Explicit Communication Transfer (ECT)

O serviço *Explicit Call Transfer* (ECT) [71] permite transferir uma chamada que já se encontre estabelecida para outro contacto.

A transferência pode ser efectuada de duas formas. A primeira forma é o utilizador que transfere a chamada ter já um diálogo com o utilizador para onde a chamada irá ser transferida (a chamada transferência com consulta). O segundo caso, o utilizador que transfere a chamada não tem indicação que o utilizador para onde a chamada vai ser transferida está disponível (a chamada transferência às "cegas").

2.5.2 *Voice Call Continuity (VCC)*

O VCC [48] é uma aplicação que se encontra na rede IMS do utilizador que permite transferir chamadas de voz entre o domínio CS e o IMS. O VCC fornece funcionalidades tanto para chamadas de voz originadas pelo utilizador, como para chamadas em que o utilizador é o destinatário.

De acordo com as especificações do 3GPP, a arquitectura do VCC é dividida em quatro componentes (Figura 20):

- *Domain Transfer Function* (DTF);
- *Domain Selection Function* (DSF);
- *Circuit Switched Adaptation Function* (CSAF);
- *Camel Service;*

Estes componentes encontram-se hospedados no SIP AS da arquitectura IMS.

O DTF é o elemento responsável pelo controlo das chamadas do serviço de VCC. Através da interface ISC com o S-CSCF, o DTF executa a transferência entre domínios, obtendo a

informação de sinalização funcionando como um 3rd *Party Call Control* (3PCC) permitindo a transferência entre o domínio CS e IMS (e vice-versa). Este elemento ainda armazena a informação do domínio onde o utilizador VCC se encontra para subsequentes selecções de domínio. Ao nível SIP, o DTF comporta-se como um B2BUA (ver Figura 9), contendo uma *leg* com o elemento originador da chamada e uma segunda *leg* com o destinatário. Quando o utilizador VCC (correspondente ao VCC *User Equipment* (VCCUE) na figura), sendo ele o originador da chamada ou o destinatário, decide mudar de domínio, o utilizador envia um segundo pedido, sendo o DTF responsável por realizar o *handover* da chamada, isto é, mudar a *leg* inicial pelo segundo pedido. Este elemento permite ainda tarifar o utilizador de acordo com o domínio onde se encontra. Esta interacção pode ser feita com o domínio de taxação, utilizando a interface Ro definida pelo 3GPP (explicação em detalhe no Capitulo 4). O DTF interage com o DSF informando-o da existência de alguma chamada em curso e o domínio da mesma.

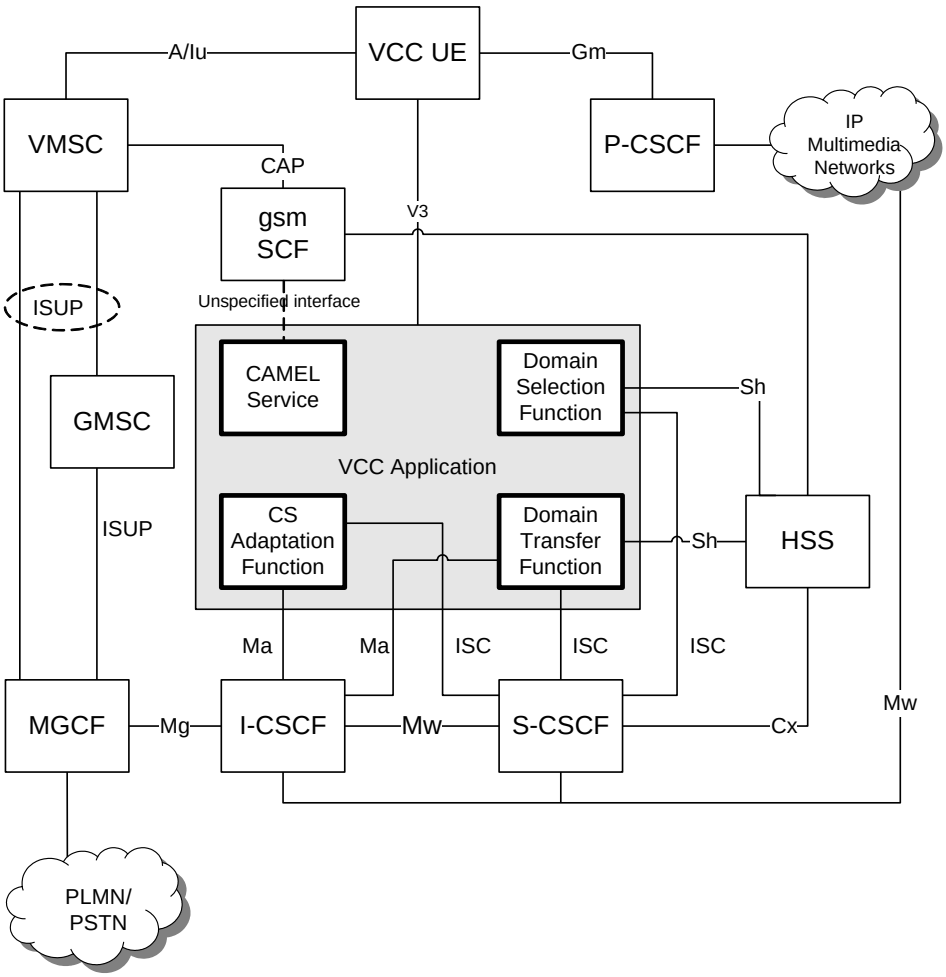

Figura 20 – Arquitectura VCC [48]

O DSF é o elemento responsável pela selecção do domínio de entrega das chamadas **destinadas**[6] aos assinantes do serviço de VCC. Neste processo de selecção, o DSF utiliza a informação do estado de registo do utilizador em cada um dos domínios, e ainda, como foi dito, comunica com o DTF para obter o domínio actual, isto é, no qual o utilizador se poderá se encontrar ligado. O estado de registo do utilizador no domínio IMS é obtido através da interface Sh *Diameter* com o HSS, enquanto no domínio CS essa informação é transferida utilizando a interface existente, não definida, com o gsmSCF. No caso de ser um HSS que englobe as funcionalidades do HLR (dado que isso nem sempre vai acontecer, pois os operadores não vão querer abdicar dos HLRs já presentes nas suas redes), a informação das redes legadas pode ser obtida também através da interface Sh.

A preferência de entrega da chamada é configurada pelo usuário através da interface V3 (ver Figura 20), e o DSF é responsável pelo armazenamento e recuperação desta informação. O DSF determina ainda os CS *Domain Routing Numbers*[7] (CSRNs) usados nas chamadas entregues à rede legada.

O CSAF, por seu lado, actua como uma *proxy* para o utilizador do VCC dentro do IMS para as chamadas originadas no domínio CS ou a leg estabelecida para transferência de domínio para o CS. Este elemento identifica o utilizador VCC na rede IMS, obtendo o *called party number* e o *calling party number* através do *IP Multimedia Routing Number*[8] (IMRN) presente na mensagem SIP que chega da rede legada. Através da interacção com o CAMEL Service, o CSAF consegue obter os dados anteriormente indicados, adaptando na mensagem que será reencaminhada para o DTF.

Por último, o CAMEL *service* interage com o domínio CS, podendo operar de forma independente ou em colaboração com o CSAF.

A interacção do core IMS com a PSTN é realizada, como referido na secção 2.2.1.7, através do elemento MGCF.

2.5.3 *Combinational Services (CSI)*

O serviço CSI [49] combina serviços CS e IMS de modo a serem usados em paralelo entre dois utilizadores no contexto ponto a ponto, utilizando uma sessão CSI. Uma sessão CSI é uma sessão multimédia que utiliza o domínio CS e o domínio IMS/PS para transportar em

[6] Apenas destinadas pois no caso do utilizador VCC ser o originador, este já escolheu o domínio que pretende utilizar.

[7] O CSRN é um número utilizado para reencaminhar a chamada do subsistema IMS para um utilizador do domínio CS

[8] Campo na mensagem de sinalização no domínio CS que indica que a mensagem é para ser reencaminhada para o domínio IMS

simultâneo diferentes componentes de *Media* para a mesma sessão. Tipicamente, transporta a voz no domínio CS e os restantes componentes de *Media* (vídeo, dados, …) da sessão no domínio IMS/PS. Uma sessão CSI pode ser criada tanto a partir do estabelecimento inicial de uma chamada CS e subsequentemente uma ou mais sessões IMS concorrentes, como também com o estabelecimento de uma sessão inicial IMS seguida de uma chamada concorrente CS.

De forma a tornar este processo transparente para o utilizador, o equipamento terá de apresentar a chamada CS e a sessão IMS num contexto único para o utilizador. Para facilitar este processo, o equipamento do utilizador e a rede envolvente deverão suportar as seguintes funcionalidades:

- Troca de informação do ambiente rádio no momento em que se pretende iniciar uma sessão CSI;
- Troca de informação das funcionalidades do terminal;
- Suportar a adição uma sessão IMS numa chamada CS a decorrer;
- Suportar a adição de uma chamada no domínio CS numa sessão IMS a decorrer;
- Suporte da rede para estabelecer sessões multimédia entre um utilizador originário IMS e um utilizador destinatário que suporta sessões CSI.

A troca de informação com o conjunto de serviços suportados pelos os terminais é extremamente vantajosa durante (ou logo após) o estabelecimento da comunicação. Esta informação pode ser utilizada para indicar ao utilizador os serviços disponíveis para uma sessão entre os utilizadores. Isto leva à utilização dos serviços disponíveis e ainda evita a invocação de serviços que possam não estar disponíveis. Assim, poupa-se na utilização dos recursos e no número de tentativas falhadas. Dois tipos de informação podem ser trocadas, a informação do ambiente rádio e as funcionalidades do terminal.

Durante o processo de troca da informação rádio, o UE verifica se o actual ambiente rádio suporta simultaneamente serviços CS e PS e ainda, através do campo *IM Status*, se é ou não possível iniciar uma sessão IMS. Caso seja possível, o UEs deverão tentar registar-se no domínio IMS (caso ainda não estejam).

A troca das funcionalidades do terminal é feita através da mensagem SIP OPTIONS. É possível encontrar nesta mensagem, entre outros elementos, os diferentes tipos de *Media* suportados pelo terminal, os formatos suportados para cada tipo de *Media*, o MSISDN e o SIP URI e um identificador do terminal para o caso do suporte de múltiplos terminais para o mesmo MSISDN ou o mesmo SIP URI.

A Figura 21 ilustra a arquitectura alto nível de dois elementos que suportam simultaneamente sessões IMS e chamada no domínio CS.

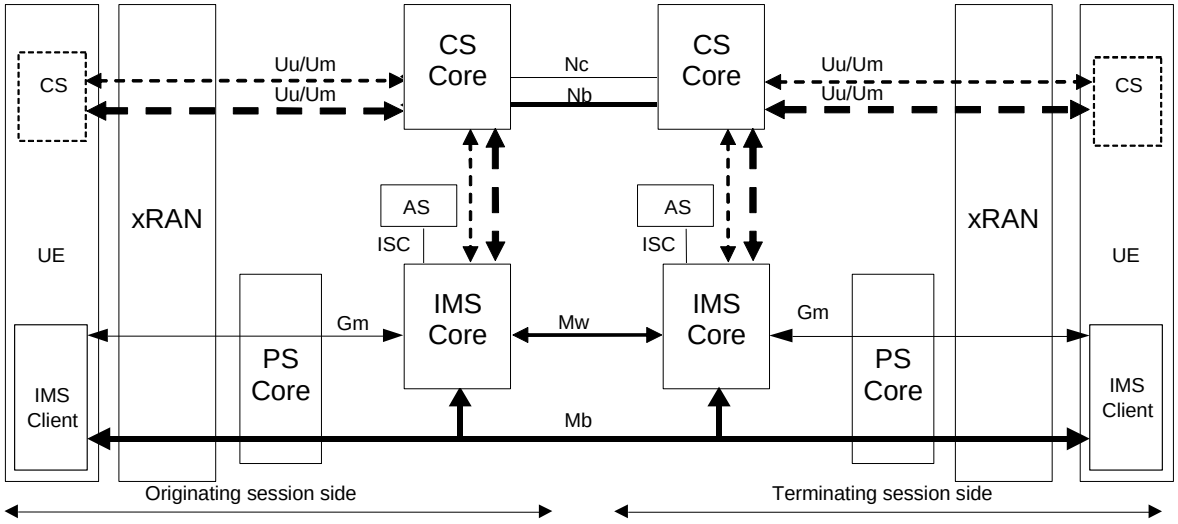

Figura 21 – Arquitectura Alto Nível CSI [49]

Como se observa, o equipamento do utilizador terá de suportar o acesso tanto ao domínio de comutação de circuitos como de pacotes.

O papel do AS nesta arquitectura é idêntico ao utilizado em IMS. É possível utilizar este elemento para controlo da sessão, por exemplo taxação. Apenas terá um papel relevante para o serviço CSI, caso seja necessário suportar a interligação entre um utilizador IMS, que inicia uma chamada para um utilizador CSI (suporta CS e IMS). Neste caso, o AS deverá ter o papel de um CSI AS. No caso contrário, isto é, CSI originador da chamada e cliente que apenas suporta o domínio IMS o destinatário, a norma [49] não contempla este caso de uso, contendo apenas em anexo a forma como poderá ser implementado.

No caso do core IMS, nenhum requisito adicional terá de ser suportando, tendo apenas de garantir a circulação da informação.

Capitulo 3 – *Service Delivery Platforms* (SDP)

Actualmente, as telecomunicações estão a sofrer importantes transformações, criando novos desafios e novas oportunidades para as redes dos operadores. É possível observar neste mercado actual, em grande alvoroço, os utilizadores seduzidos por mais e melhores serviços, aumentando cada vez mais a sua procura.

Esta tendência do mercado está a aumentar e é necessário um novo tipo de serviços de dados que sejam aceites numa escala global, serviços inter-operáveis e agnósticos às camadas inferiores da rede.

Desde modo, as redes dos operadores estão agora a deparar-se com este novo conceito arquitectural, uma plataforma para o fornecimento de serviços que é inteiramente desacoplada da infra-estrutura da rede, onde os serviços não são criados tendo como dependência a arquitectura de rede.

O mundo das telecomunicações apercebeu-se da necessidade de um novo e promissor modelo de criação e execução de serviços. Como sempre, as questões financeiras foram o elemento preponderante para esta transição, isto é, a promessa de receitas maiores. Os operadores compreenderam a necessidade das *Service Delivery Platforms* (SDP) para reduzir o custo e a complexidade de fornecer novos e melhores serviços.

3.1 O que é uma SDP

Tal como o nome indica, SDPs têm como função a entrega de serviços. Essencialmente, este conceito foi criado para descrever uma solução IT que pode ser utilizada pelos fornecedores de serviços para oferecer novos serviços de voz e dados (serviços de próxima geração) para empresas e utilizadores tradicionais.

O actual espectro de serviços fornecidos pelas SDPs não se encontra ainda claramente definido, podendo variar de vendedor para vendedor. Os serviços entregues por estas plataformas podem ser simples aplicações de telecomunicações, como são os casos do *click-to-dial* ou uma tradução de número, como podem englobar serviços avançados como serviços baseados na localização, tarifação do conteúdo, *call centers* virtuais, *messaging*, entre outros.

Numa tentativa de capturar o contexto completo do conceito da SDP, uma SDP ‘ideal’ pode ser definida da forma seguinte:

- Uma SDP fornece um completo ambiente de criação, execução e gestão de serviços;
- Uma SDP suporta a entrega de serviços de voz e dados de uma forma independente da rede e dos dispositivos;
- Uma SDP agrega diferentes funcionalidades da rede e dos serviços, isto é, permite aos criadores de aplicações aceder aos serviços disponíveis na plataforma e aos recursos da rede de uma forma agregada e uniforme;
- Opcionalmente, uma SDP poderá fornecer o acesso de forma aberta e segura às funcionalidades dos serviços para ser utilizadas por outras entidades externas e/ou empresas;

Uma SDP faz parte da infra-estrutura IT dos operadores de rede ou fornecedores de serviços, não sendo um elemento do *Core* da rede. Deste modo, uma SDP opera com os restantes elementos da rede, como por exemplo, os sistemas de autenticação, tarifação ou mesmo os elementos do core da rede.

No caso dos operadores móveis, é necessário uma integração com algumas das plataformas já existentes na rede, como é o caso dos SMSCs, MMSCs, servidores WAP, servidores de localização, alargando deste modo a variedade de opções que possam existir nos serviços.

3.2 **Arquitectura e elementos**

Até à data, ainda não existe um consenso na definição da arquitectura e elementos pertencentes a uma SDP. Mesmo assim, a partir da observação das diferentes SDPs que se encontram disponíveis no mercado, é possível encontrar quatro elementos que permitem caracterizar uma SDP (Figura 22) [50]:

- *Service Execution Platform* (SEP);
- *Service Exposure Layer* (SEL);
- *Content Delivery Platform* (CDP);
- *Network Abstraction Layer* (NAL);

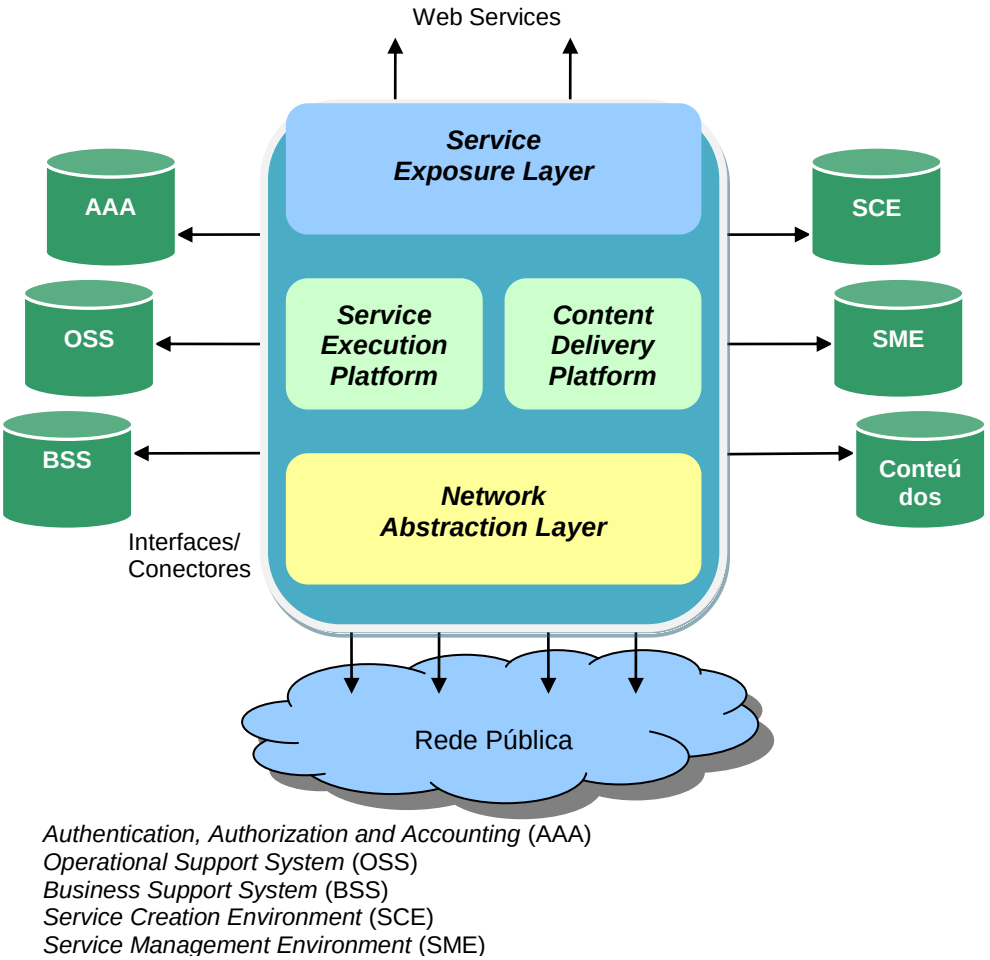

Figura 22 – Elementos de uma SDP [50]

3.2.1 *Service Execution Platform (SEP)*

O SEP é um elemento do core da SDP que fornece um ambiente para *deployment* e execução dos serviços. O SEP pode ser construído utilizando os *Application Server* J2EE ou .Net, como ainda através de ASs especializados para telecomunicações (por exemplo, baseada em JSLEE [51]).

Devido à variedade de funcionalidades numa SDP, a lógica de serviços é, em alguns casos, implementada utilizando mais do que uma aplicação, isto é, diferentes ASs interagem, tendo sido implementados utilizando diferentes tecnológicas, tais como, J2SE, J2EE, JSLEE, etc. Utilizar, por exemplo, J2EE para plataforma de gestão *web* de um serviço e JSLEE como ambiente de execução para serviços de telecomunicações.

3.2.2 *Network Abstraction Layer (NAL)*

O NAL fornece interfaces normalizadas entre os serviços e os elementos da rede. Além desta funcionalidade, o NAL deverá garantir uma abstracção da rede aos restantes

elementos que compõem a arquitectura SDP. Esta abstracção esconde as discrepâncias dos diferentes tipos de redes e das diferenças que possam existir entre diferentes fabricantes. Por exemplo, uma aplicação a correr numa SDP deverá ser capaz de obter, por exemplo, a informação de localização de um utilizador da mesma forma, apesar de este poder encontrar-se numa rede móvel, fixa ou VoIP. Isto permite às aplicações que correm numa SDP serem, desta forma, portáveis e independentes da rede.

De modo a fornecer um correcto nível de abstracção dos elementos da rede, o NAL usualmente utiliza as especificações emergentes com a OSA/Parlay, JAIN, Parlay X, Web Services, OMA (exemplo, LIF), IMS (SIP e *Diameter*), entre outros. Normalmente estas interfaces definidas por estas especificações são expostas aos criados das aplicações como componentes Java/J2EE ou outros tipos de conectores (no caso do JSLEE, utilizado para a realizar o demonstrador, os conectores são chamados de RAs, sendo a ligação feita através do envio de eventos Java entre os elementos).

3.2.3 *Service Exposure Layer (SEL)*

O SEL é um elemento opcional da SDP que expõe as funcionalidades dos serviços (usualmente utilizando *Web Services*) a terceiros e empresas. Através do SEL, um operador pode oferecer a terceiros o acesso indirecto às funcionalidades dos elementos de rede, de forma segura e rentável. Por exemplo, a SDP poderá disponibilizar uma interface simples e segura para o envio de MMS, independente do protocolo subjacente a este processo.

Este alto nível de abstracção e simplicidade permite a utilizadores inexperientes a utilização das funcionalidades oferecidas pela SDP.

3.2.4 *Content Delivery Platform (CDP)*

A CDP é um elemento opcional frequentemente presente em SDPs para redes móveis de forma a aprovisionar conteúdos multimédia em dispositivos móveis.

3.3 **SDP e IMS**

A SDP pode ser vista como a antecessora do IMS. Originalmente, a SDP provem da necessidade dos operadores móveis fornecerem serviços de uma forma mais flexível e económica, sobre infra-estruturas de rede proprietárias e complexas. O IMS, por seu lado, foi desenvolvido para responder às necessidades dos operadores GSM de encontrar e fornecer aplicações IP na nova rede 3G.

A tendência é que ambas as tecnologias coexistam, de forma a fornecer serviços de próxima geração a consumidores e empresas (Figura 23). Juntos, as SDPs e o IMS foram desenvolvidos com um objectivo comum em mente: a entrega de serviços ao nível aplicacional a partir dos mecanismos existentes na rede, e de um modo agnóstico à rede.

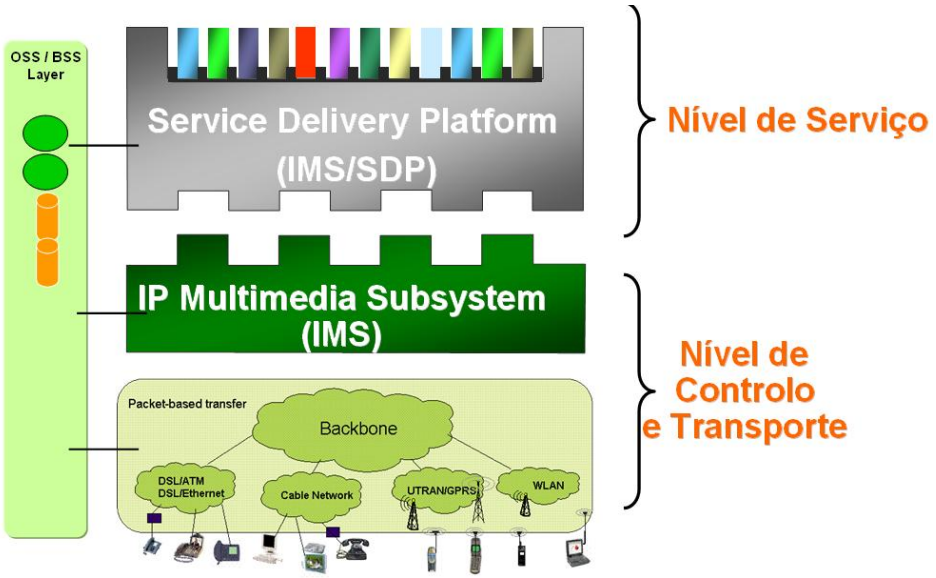

Figura 23 – Interoperabilidade entre as SDPs e o IMS

Muitos vendedores de SDP estão actualmente a rotular as suas soluções como IMS *compliant*. No caso dos primeiros fornecedores de SDPs, as suas soluções estão neste momento a serem reajustados para responderem a novos requisitos erguidos pelo IMS. Para uma coexistência positiva e saudável entre as SDPs e o IMS, não deverá existir nenhuma funcionalidade no domínio SDP que se sobreponha às do IMS. Uma das organizações responsáveis por esta coexistência é a OMA.

3.4 **A OMA e o seu papel nas SDPs e no IMS**

O objectivo da OMA é proporcionar um único ambiente comum para todas actividades de normalização na área dos *service enablers*, permitindo deste modo, a harmonização das diferentes organizações numa única actividade de normalização *"together we stand, divided we fall"*.

Os *Enablers* são blocos de código que fornecem interfaces *Northbound* simples para terceiros (ver camada SEL das SDPs), isolando o utilizador das complexidades das tecnologias da rede.

Enablers é uma tecnologia com o objectivo de facilitar o desenvolvimento, instalação e utilização de um serviço. Os *Enablers* são definidos numa especificação, ou num grupo de especificações e publicados pela OMA.

A adopção desta tecnologia, permite assegurar a interoperabilidade entre os diferentes serviços, como ainda a interoperabilidade e independência da rede de acesso, dos fornecedores de serviços, entre outros.

O resultado destas especificações leva a negócios inovadores e diferenciados, já que os criadores de serviços não necessitam de desperdiçar o seu tempo com questões de interoperabilidade. Por outras palavras, este tipo de tecnologia de *enablers* de serviços baseado em especificações globais e públicas, como é o caso do MMS, XHTML, irão facilitar a criação de novos serviços e consequentemente, influenciar o crescimento da indústria.

Questões como a interoperabilidade entre os serviços e a arquitectura IMS, são resolvidas por esta organização, permitindo que não exista sobreposição entre as SDPs que seguem a OMA e o core IMS.

É importante referir que a OMA e a indústria móvel têm vantagens em explorar as funcionalidades existentes no IMS [52]. Entre esses benefícios estão:

- O processo para a definição dos serviços dentro da OMA é facilitado e acelerado;
- Focar o trabalho nas especificações dos *service enablers* explorando a arquitectura IMS já definida apoiando o 3GPP/3GPP2 no processo de especificação;
- A duplicação do trabalho entre a OMA e o 3GPP/3GPP2 pode ser evitado e a mesma arquitectura IMS pode ser adoptada;
- Melhorar a troca de informação e comunicação entre a OMA e o 3GPP/3GPP2 quando os termos, definições e conceitos IMS são os mesmos (já que a primeira abordagem da OMA foi de uma arquitectura IMS com extensões para a OMA);
- Reutilização máxima das funcionalidades IMS e dos mecanismos de rede protegendo os investimentos e esforços da indústria celular e redução de custos adicionais em nivelar investimentos correntes para IMS;

Na Figura 24 é possível observar-se uma lista de serviços especificados pelas diferentes organizações e a sua ligação com o core IMS. Temos então o core IMS definido pelo 3GPP/3GPP2 e adoptado pelo TISPAN a dar suporte à camada aplicacional onde se encontram os diferentes serviços normalizados (CSI, VCC, *Presence*, *Group/List Manager Server* (GLMS), etc.).

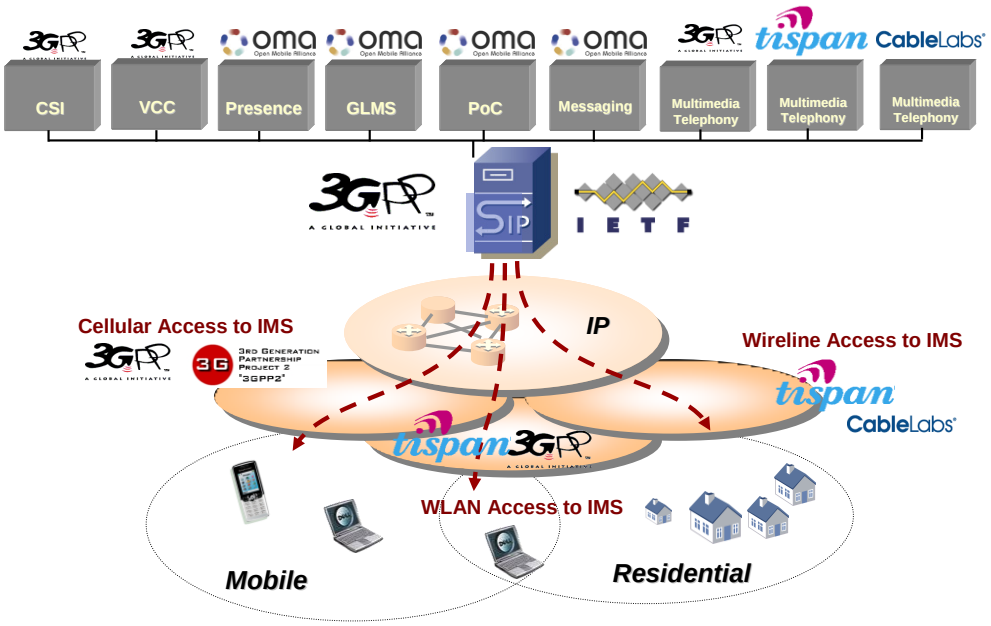

Figura 24 – Interoperabilidade entre organizações

Para a OMA garantir a interoperabilidade dos seus enablers com o core IMS normalizou, através de um *enabler* específico, o IMSinOMA *Enabler*, que define o modo como os *enablers* podem utilizar as capacidades do core IMS. Através desta especificação, os restantes enablers percebem como terão de utilizar o IMS e como deverão interagir com o IMS de modo consistente.

Por exemplo, no caso do serviço *Push-to-Talk*, já existem alguns recursos na arquitectura IMS que poderão ser reutilizadas, não sendo necessário a reutilização do esforço para a criação de uma arquitectura similar para este serviço [53].

É importante notar que isto não implica que os *enablers* da OMA usem o IMS; entretanto, para aqueles enablers que o façam, o IMSinOMA fornece orientação. A Figura 25 é um exemplo de uma das interfaces IMS que terá de ser implementada pelos *enablers* da OMA para estarem consistentes com o IMS. Nesta figura é apresentada a interface ISC entre a camada de serviços (os diferentes *enablers* da OMA) com o S-CSCF do core IMS.

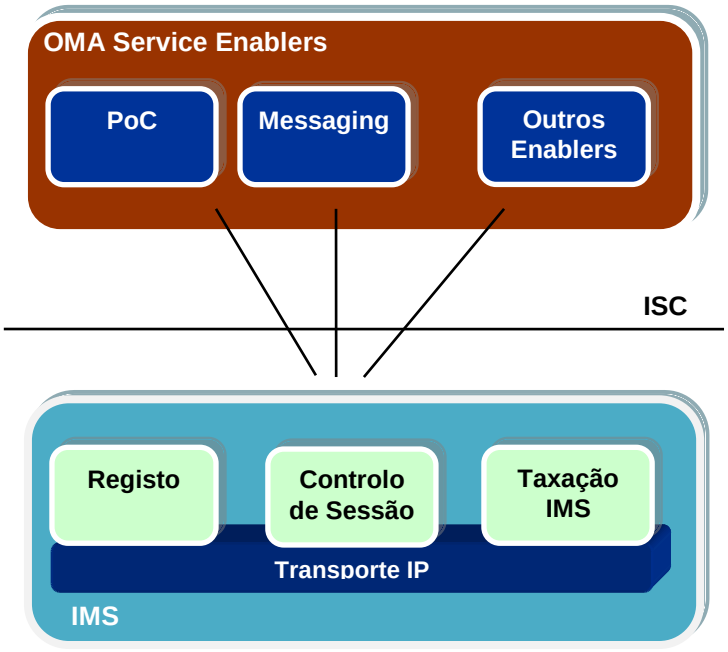

Figura 25 – Interface para gestão de sessões [80]

Como já explicado, o principal objectivo da especificação do IMS foca-se nos elementos core da rede IMS para sessões IMS, incluindo serviços em tempo real e *messaging*. Como tal na evolução das soluções *soft-switching* (solução pré-IMS). Outra preocupação prende-se com a normalização dos protocolos SIP e *Diameter*, que servem de suporte às interfaces com os AS (interfaces ISC, ro, rf, sh, etc.). As aplicações IMS em si não são normalizadas. Neste contexto, a OMA realiza então essa normalização dos serviços no topo da arquitectura IMS.

O IMS fornece aos serviços definidos pela OMA, as seguintes funcionalidades:

- Reencaminhamento da sinalização;
- Mecanismos de autenticação;
- Compressão da sinalização;
- Privacidade;
- Armazenamento do perfil do utilizador;
- Tarifação, *Accounting*;
- Interface para a rede de acesso;
- Etc.

3.5 **JAIN SLEE**

O JSLEE é um standard em Java definida pela SUN que funciona como um ambiente de criação, execução e gestão de serviços, tendo sido o ambiente escolhido para a criação do demonstrador.

3.5.1 *Iniciativas JAVA*

Lançada pela SUN em 1998, a iniciativa *Java APIs for Intelligent Networks* (JAIN) [55] estende a plataforma JAVA ao mundo das Comunicações. Esta iniciativa visa as necessidades das redes de nova geração, desenvolvendo um conjunto de *Application Programming Interfaces* (APIs) e Contentores de Aplicações para redes integradas [56].

O JAIN é um conjunto de APIs de JAVA que permitem desenvolver serviços de forma simples, económica e de forma rápida independentemente da tecnologia da rede utilizada *"Write once – Run anywhere"*. O JAIN ao se basear sobre a plataforma JAVA, permite introduzir portabilidade de serviços entre sistema e permite o acesso seguro a recursos de diferentes redes. A tecnologia JAIN vai alterar radicalmente o mercado das telecomunicações permitindo a passagem de sistemas fechados e proprietários a sistemas abertos que oferecem uma inter conexão total das diferentes redes existentes. Actualmente, mais de 80 empresas fazem parte e são activos na comunidade de desenvolvimento da tecnologia JAIN sob o controlo da *Java Community Process* (JCP), que garante assim a qualidade das novas APIs, a sua homogeneidade e a sua compatibilidade a longo prazo [54].

Como exemplo prático, é possível observar as facilidades que esta tecnologia leva aos criadores de serviços. Ao implementar um serviço, o criador pode desenvolve-lo seguindo a API sem se preocupar com a implementação da mesma. Depois só precisa de procurar no mercado a solução mais atractiva que implemente essa API, sem se preocupar se é interoperável com o seu serviço. Caso apareça no mercado uma solução mais atraente, por exemplo, a nível de preço ou mesmo performance, não irá ser necessário alterar o serviço em si, tornando-se o processo mais flexível. Como os fornecedores seguem a mesma API, irão ser interoperáveis entre si.

O JAIN é composto por vários grupos que agem a níveis diferentes, oferecendo cada um as suas vantagens:

- JAIN *Protocol* API
- JAIN *Call Control* API
- JAIN *Service Logic Execution Environment*
- JAIN *Service Creation Environment*

É sobre a Interface e Contentor de Aplicações JAIN SLEE que trata os próximos itens.

3.5.2 *Plataforma de execução de serviços*

A plataforma de execução de serviços (ou SLEE – *Service Logic Execution Environment*) disponibiliza um ambiente para o aprovisionamento e execução dos serviços.

O JSLEE é uma norma JAIN produzida no âmbito do JCP orientada para ambientes de execução de aplicações da área de Telecomunicações que oferece uma abstracção em relação ao que lhe cerca. Permite ao criador de serviços concentrar-se no serviço que deseja criar, sem estar a preocupar-se com as infra-estruturas e os protocolos de sinalizações utilizados, ou seja, os serviços são totalmente independentes das mudanças da rede e dos protocolos utilizados. Assim qualquer serviço desenvolvido para este ambiente funcionará num outro ambiente JSLEE.

A arquitectura do JSLEE (Figura 26) define ligações entre componentes, adaptadores e o contentor que os armazena durante a sua execução. Estas ligações garantem a todos os componentes e adaptadores a sua portabilidade para qualquer plataforma JSLEE com as mesmas funcionalidades de gestão, administração e com os níveis de desempenho, disponibilidade e escalabilidade necessárias para as aplicações em tempo real.

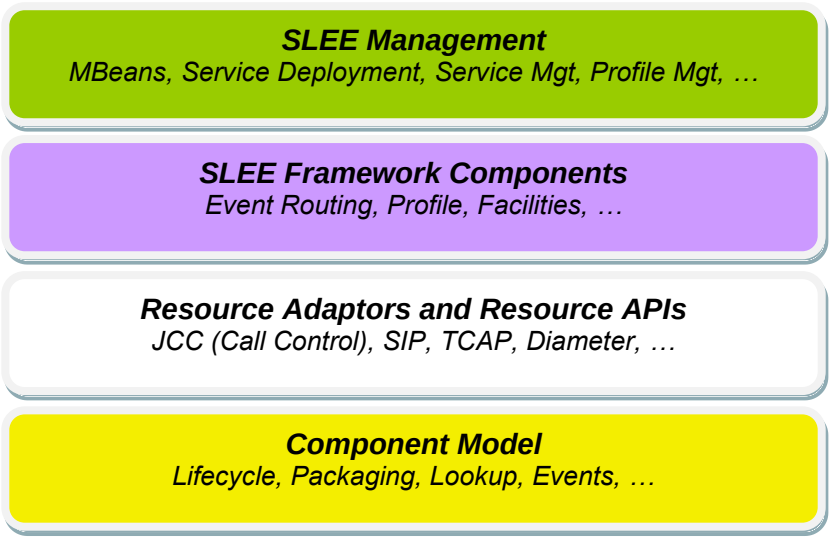

Figura 26 – Arquitectura do JSLEE [57]

O JSLEE possui um modelo constituído por componentes optimizados para eventos e é projectado para as exigências de processamento de eventos (quase) em tempo real. É este modelo de componentes que define o modo como a lógica de serviço é construída, empacotada, recebe eventos e é executada. Este modelo permite estruturar a lógica de aplicações em conjuntos reutilizáveis de componentes orientados para objectos e a composição desses componentes em serviços mais sofisticados e de mais alto nível. O modelo foi desenhado de modo a simplificar o trabalho do programador de aplicações, eliminando erros de programação habituais e garantindo o rápido desenvolvimento de

aplicações robustas. De seguida é apresentado um resumo dos benefícios da plataforma de execução de serviços de Telecomunicações:

- JAIN SLEE permite fácil inter conexão entre domínios diferentes;
- Não existe bloqueio de recursos devido a sua natureza assíncrona;
- O conceito de adaptadores do recurso (RA) oferece uma variedade infinita de adaptadores para alcançar qualquer recurso, incluindo redes IP e SS7;
- JSLEE oferece um modelo de programação baseado em componentes;
- Os componentes são desenvolvidos uma vez e funcionam em vários servidores de aplicações de JSLEE;
- As aplicações são independentes dos fornecedores de rede;
- Os componentes são reutilizáveis devido ao ambiente normalizado da execução;
- Execução dos serviços optimizada através do conceito de actividades;

3.5.3 *Elementos do JSLEE*

O JSLEE é constituído por quatro elementos básicos (Figura 27):

- *Resource Adaptors* (RA);
- Eventos;
- *Activity Contexts*;
- Ambiente de *runtime*, que hospeda os objectos *Service Building Block* (SBB);

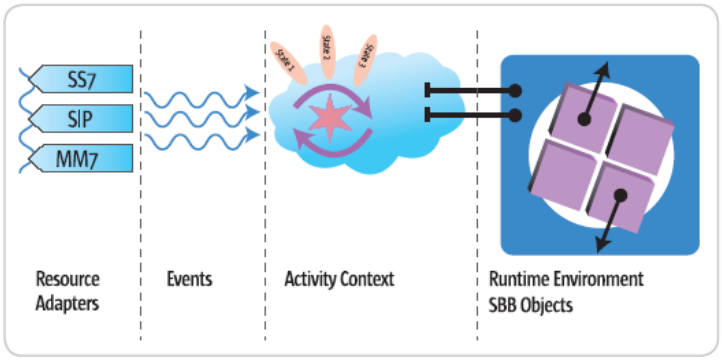

Figura 27 – Os 4 elementos básicos do JSLEE

3.5.3.1 Resource Adaptors (RA)

Recursos são entidades externas ao SLEE (SIP *Proxies*, Bases de Dados,...) que são potenciais geradores de eventos a consumir por entidades do SLEE. Um RA adapta as interfaces e requisitos de um recurso às interfaces e requisitos do JSLEE. Ao receber uma mensagem de uma identidade externa, o RA pode alterar o seu estado interno e submeter um evento que abstrai a mensagem ao SLEE. O SLEE irá "rotear" o evento pelos SBB

interessados em o receber. Por outro lado, um SBB pode invocar um método no RA (Figura 28).

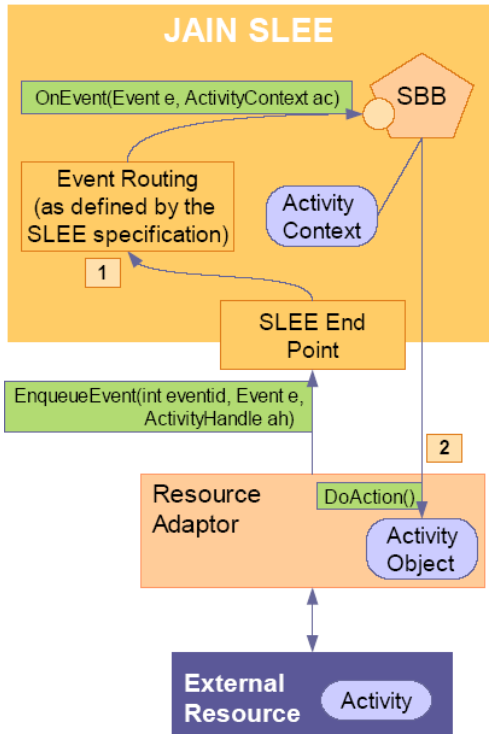

Figura 28 – Adaptador de Recursos

3.5.3.2 **Eventos**

Um evento representa uma ocorrência sujeita a processamento aplicacional. Os eventos podem ser originados por diversas fontes incluindo pilhas protocolares, pelo próprio contentor SLEE, ou por outro componente SBB. Um evento é uma abstracção usada para modular ocorrências que não estão associadas com nenhum processo computacional específico (por exemplo o evento resultante do envio da mensagem "SIP INVITE" a pedir o estabelecimento duma sessão multimédia entre duas entidades).

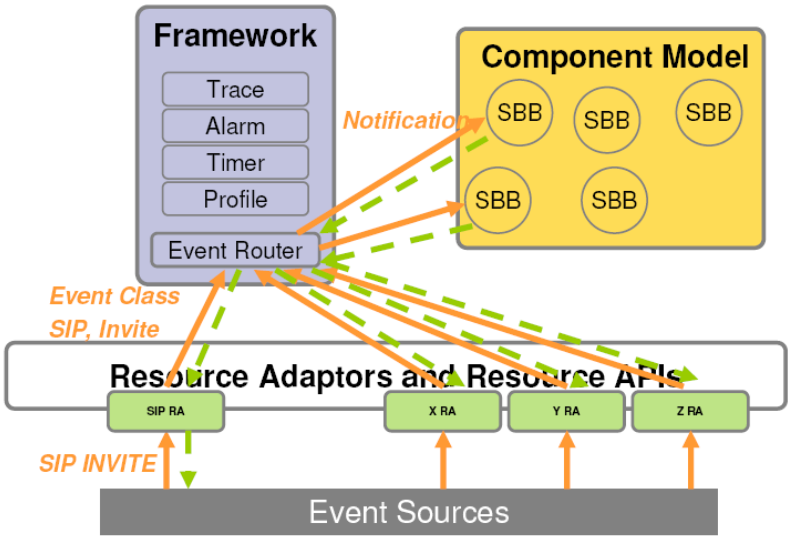

Figura 29 – "Roteamento" de Eventos [57]

3.5.3.3 Actividades e Activity Context

As actividades modelam uma máquina de estados subjacente a um determinado recurso. De acordo com as transições da máquina de estados, o recurso emite um fluxo de eventos. O RA disponibiliza aos SBB interessados uma representação Java desta actividade (Objecto Actividade).

O *Activity Context* é uma entidade lógica do SLEE que recebe e faz o "roteamento" de eventos (através do *Event Router*) para os componentes do SBB.

3.5.3.4 Service Building Block (SBB)

Componente reutilizável de software que envia e recebe eventos e, com base no seu estado e nos eventos recebidos, executa uma determinada lógica computacional. Um SBB é um componente com estado e, como tal, pode lembrar os resultados de computações anteriores e usá-los em computações posteriores. Um SBB pode ser composto por outros SBB para a construção de aplicações mais complexas com base em SBB existentes. Um serviço é então um conjunto de SBBs que comunicam entre si (Figura 30).

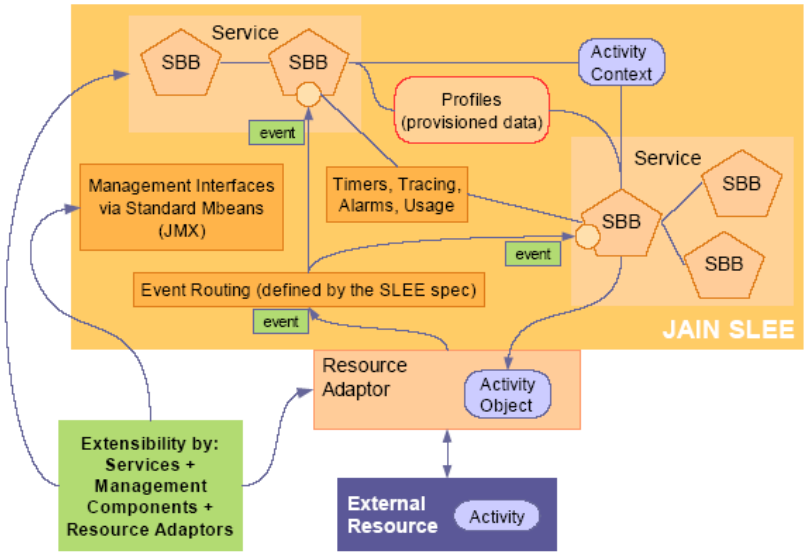

Figura 30 – Interacção entre os diferentes componentes do JSLEE

3.5.4 *SLEE Facilities*

O SLEE também é composto por um conjunto de APIs que facilitam na construção dos serviços ou dos RAs. São elas:

- *Timer Facility;*
- *Trace Facility;*
- *Alarm Facility;*
- *Usage Facility;*
- *Activity Context Naming Facility;*
- *Profile Facility;*

3.5.5 *J2EE vs JSLEE*

Comparando aplicações de J2EE (aplicações empresariais) com as aplicações de JSLEE (comunicações), destacam-se os seguintes aspectos:

- *Invocation*: Síncrono (pedido/resposta) vs. Assíncrono (orientado a eventos);
- *Event granularity*: Frequência baixa vs. Frequência alta;
- *Component lifecycle*: Persistente (guarda estado) vs. Transiente;
- *Data Sources*: Servidores com a cópia definitiva primária vs. Proveniente de várias fontes;
- *Computation*: Acesso intensivo a base de dados vs. Computação intensiva;
- *Transactions*: Base de dados vs. *Lightweight*;
- *Availability*: Dois a três 9s vs. Cinco 9s (menos de 6 minutes de *downtime* por ano);
- *Real-time*: Não vs. *Soft* (*near-real-time*);

- *Deployment*: Centralizado em *clusters* vs. Distribuição pela rede;

O JSLEE baseia-se nos contentores dos EJB, não sendo dependente do J2EE apesar de ser construído com base neste. Os serviços a correr no SLEE podem comunicar directamente com J2EE, invocando directamente um EJB. Em contraste, se um serviço a correr em J2EE precisa de comunicar com algum serviço do JSLEE terá que utilizar o *Java Connector Architecture* (JCA) para disparar um evento através de um RA (por exemplo, um RA para *Java Message Service*[9] (JMS)) para que este chegue ao Contentor JSLEE. Assim fica garantido a convergência dos Contentores JSLEE e J2EE, que resolvem os diferentes problemas dos operadores de rede (Figura 31). É a própria especificação JSLEE que define esta interacção, incluindo a implementação do conector SLEE para J2EE.

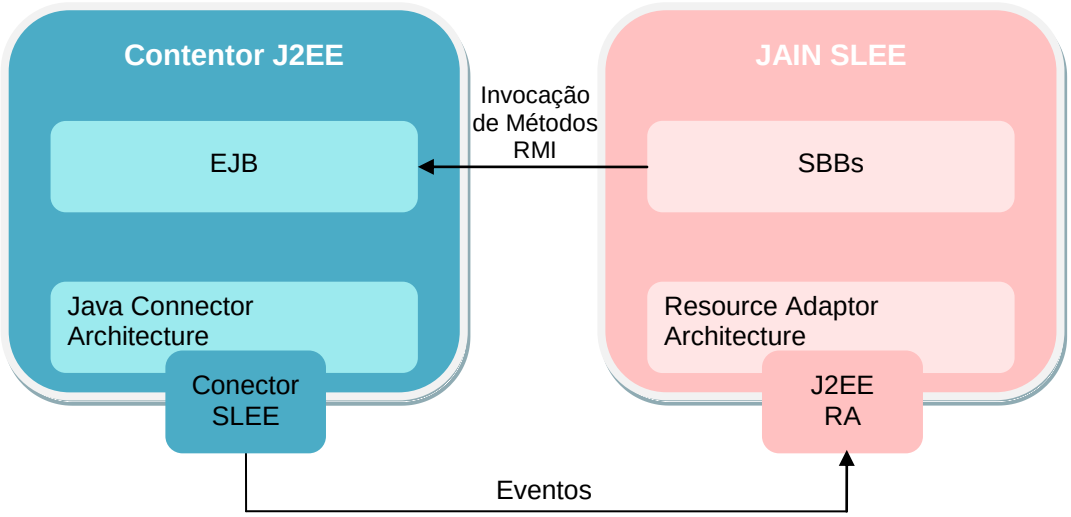

Figura 31 – Comunicação entre JSLEE e J2EE

[9] API Java para troca de mensagens entre aplicações.

Capitulo 4 – *Charging* em IMS

O IMS proporciona um ambiente favorável ao aparecimento de novos serviços e ainda a convergência das redes fixa e móvel. Como tal, o 3GPP introduziu uma nova *framework* de taxação genérica, para responder ao novo desafio de tarifar em IMS [59].

Hoje em dia, os operadores móveis deparam-se com mercados saturados nos países desenvolvidos e um decréscimo de receitas por utilizador resultado de uma forte competição. Especialmente com os serviços tradicionais, caso da voz e mensagens, a percentagem de receitas tem vindo a diminuir, devido a um aumento do interesse em serviços multimédia e comunicações ricas em novas funcionalidades. Exemplos são serviços de voz sobre comutação de pacotes, como VoIP, jogos *multi-player online*, partilha de *Media* e conteúdos, entre outros [60]. O IMS fornece as funcionalidades que permitem controlar estes serviços e promete reduzir os custos operacionais ao aumentar a flexibilidade e a simplicidade, baseado numa única infra-estrutura. Como o que se pretende é que estes serviços tenham sucesso, é necessário que estes suportem um sistema de taxação eficiente e flexível. Ao permanecer com uma aproximação *flat-rate*, não irá ser possível tarifar da forma mais correcta ou mesmo tarifar os novos serviços IMS, oportunidades que os competidores com modelos de tarifação orientados para o cliente que endereçam as necessidades individuais dos clientes deverão ter em conta. Por outro lado, como uma plataforma, o IMS oferece ao operador a possibilidade de utilizar uma solução de taxação única para todos os serviços.

A solução de taxação para IMS faz parte de uma *framework* genérica (3GPP *Release* 6), definida não só para IMS, pertencente ao nível do subsistema, mas também utilizada para tarifar ao nível de acesso e ainda ao nível dos serviços [61].

4.1 3GPP Charging: Princípios comuns e Arquitectura

Como foi referido anteriormente, a taxação em IMS não se encontra isolada dos restantes mecanismos de taxação já antes existentes no 3GPP. Os seus princípios gerais e a sua arquitectura foram adicionados a uma *framework* genérica de taxação, para deste modo ser possível dar resposta aos requisitos específicos da taxação em IMS.

Com a *release* 6 do 3GPP, foi criada uma *framework* comum para especificar a taxação tendo como objectivo descrever as funcionalidades gerais da taxação numa única especificação [61]. Serve assim, como documento modelo “umbrella” para todas as subsequentes especificações que possam existir para a taxação no 3GPP [59].

Este novo modelo de taxação teve como motivação, o aumento do número de tecnologias e serviços, cada um com uma nova especificação específica, tornando-se inviável.

Assim, em vez de se ter que definir todas as funcionalidades de taxação sempre que um novo serviço ou tecnologia surge, já que estas seriam independentes das especificações já existentes, o 3GPP identificou todas as funcionalidades lógicas de taxação comuns, que fornecem os aspectos diferentes das funcionalidades necessárias para todas as partes da rede 3GPP relevantes para a taxação, e combinou-as numa única arquitectura lógica (Figura 32).

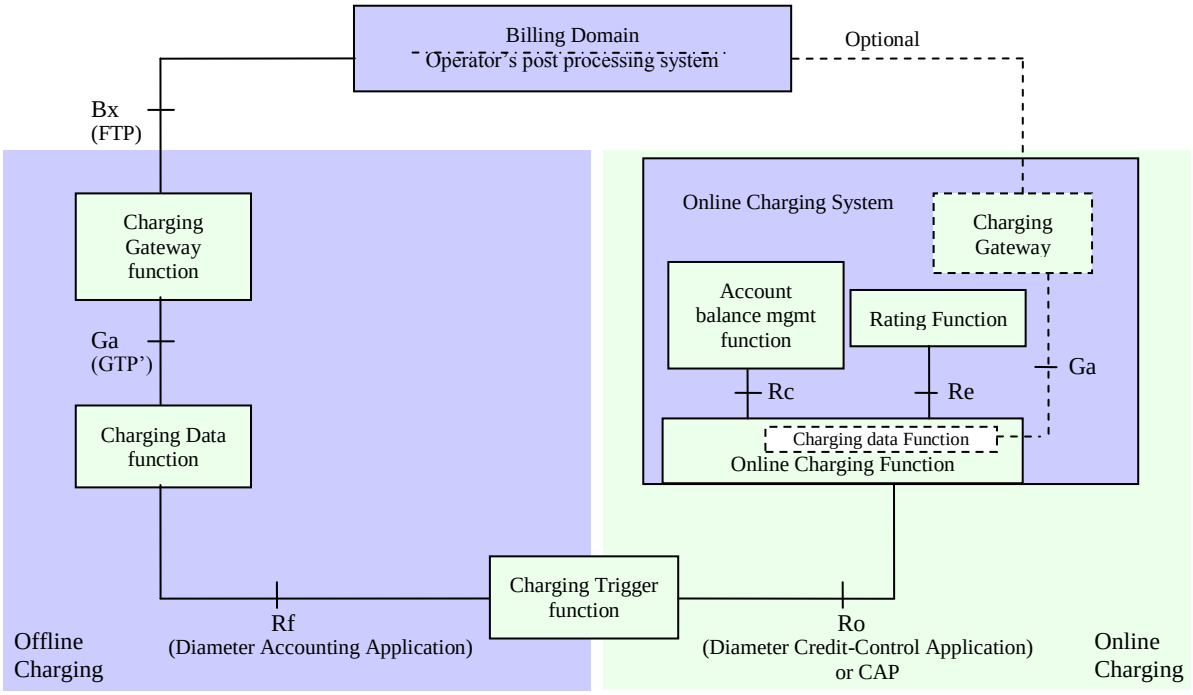

Figura 32 – Arquitectura simplificada da *framework* de *Charging* 3GPP *Release* 6 [59]

Dentro dos procedimentos de taxação da rede 3GPP, e observando a figura, é possível distinguir dois mecanismos distintos: taxação *offline* e *online*.

No caso *offline*, a utilização dos recursos da rede, é reportada ao *Billing Domain* (BD), através de uma série de funções lógicas (descritas mais à frente), após a utilização desses mesmos recursos já terem ocorrido. Como é possível concluir, este modo de proceder não interage directamente com o serviço a ser fornecido, não afectando a sua utilização. Apesar de muitas vezes ser utilizado para este propósito, os mecanismos de taxação *offline* não devem ser confundidos com tarifação pós-pago, dado que este último representa um método ou acordo de pagamento. Pelo contrário, a taxação *online* é um mecanismo onde a informação de tarifação pode afectar, em tempo real, o serviço oferecido, sendo necessário por este facto, uma interacção entre o mecanismo de tarifação com os elementos que controlam os recursos da rede. A autorização para a utilização desses recursos deverá ser

obtida previamente, isto é, antes do fornecimento de um determinado recurso da rede ocorrer. Este tipo de taxação vai então depender do saldo da conta do subscritor e do valor a taxar de acordo com o evento ocorrido. Taxação *online* é muitas vezes utilizada em serviços pré-pagos.

Dentro desta *framework*, é depois possível encontrar os documentos específicos para cada serviço e tecnologia. A estrutura completa com as diferentes especificações está ilustrada na figura seguinte [61]:

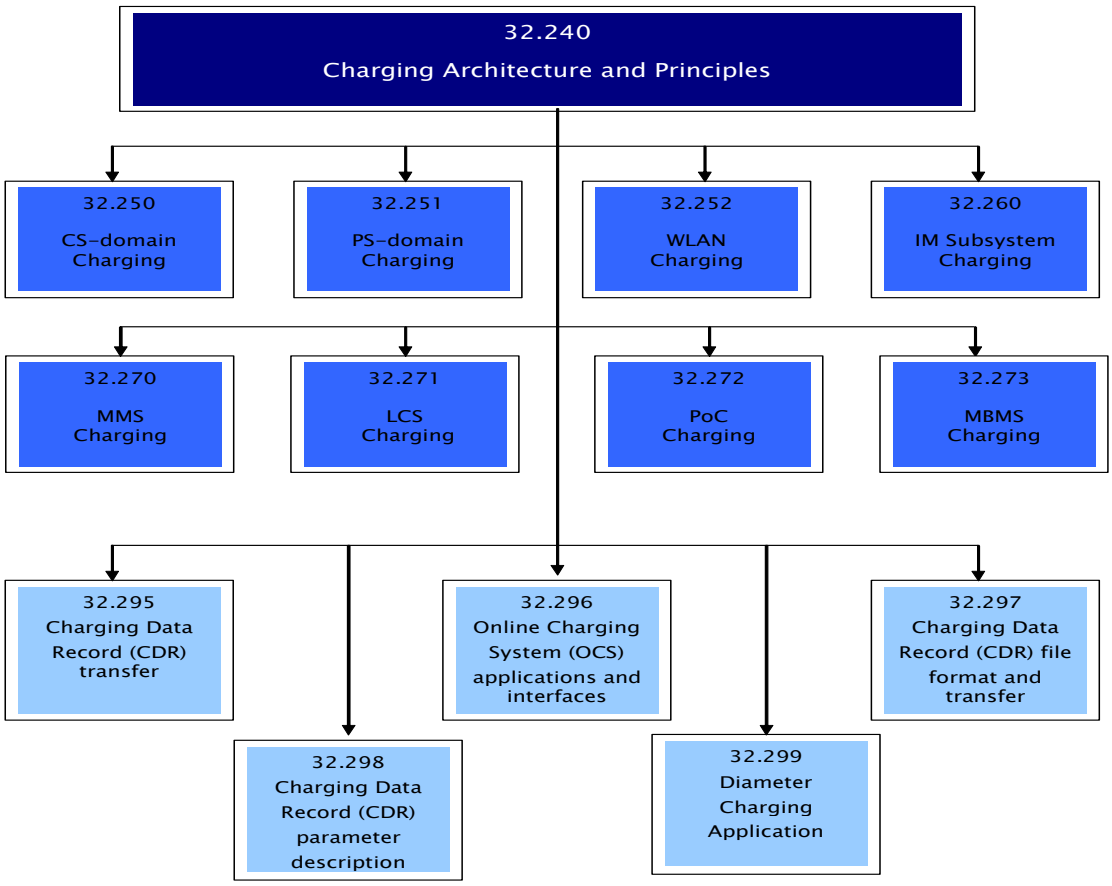

Figura 33 – Estrutura das especificações de *Charging* [61]

Estas especificações estão divididas em três grupos, descrevendo os mecanismos de taxação em três diferentes níveis de taxação, identificados pelo 3GPP:

- Nível de rede (CS, PS, WLAN);
- Nível de subsistema (IMS);
- Nível de serviço (por exemplo, PoC, serviços *multicast*, serviços de mensagens, etc.);

Como um todo, formam as chamadas especificações *middle tier*[10] [61].

[10] Termo utilizador para as especificações de taxação 3GPP que especificam as funcionalidades de taxação, online e offline.

4.2 **Taxação em IMS**

Baseado no que foi dito anteriormente, a especificação [62] descreve em detalhe os mecanismos de taxação *online* e *offline* em IMS, bem como os elementos deste subsistema envolvidos.

4.2.1 *Taxação Offline em IMS*

Olhando para a Figura 34, é possível observar três funções lógicas, pertencentes à arquitectura genérica de taxação do 3GPP *release* 6, responsáveis pela taxação *offline*: o *Charging Trigger Function* (CTF), o *Charging Data Function* (CDF) e o *Charging Gateway Function* (CGF).

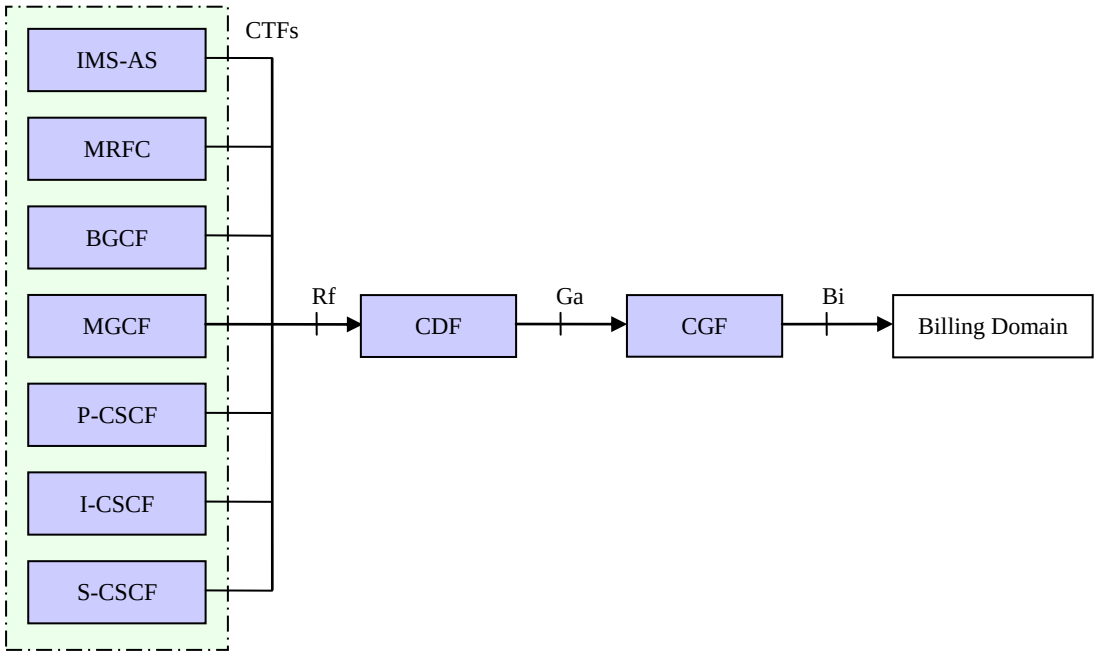

Figura 34 – Arquitectura Offline Charging IMS [62]

O CTF é um componente que se encontra integrado nos elementos da rede, e que permite gerar informação de tarifação baseando-se na observação da utilização dos recursos de rede, isto é, *chargeable events*. Estes eventos são as actividades do utilizador que utilizam e consumem os recursos da rede.

Após reconhecer um pedido a ser tarifado (os chamados *chargeable event*), o CTF gera um evento de taxação (*charging event*[11]), "data record" contendo um conjunto de informação que permite caracterizar o correspondente evento da rede. Através da utilização da aplicação de *accounting* do protocolo *Diameter* base [31], o evento de taxação é transferido para o CDF através da interface Rf.

[11] *chargeable event* são eventos (por exemplo, mensagens SIP) que podem gerar *charging events* que são enviados pelo CTF

As entidades funcionais IMS presentes na Figura 34 participam na taxação *offline* IMS através de um CTF integrado, monitorizando as mensagens de sinalização SIP e ISUP, definidas como *triggers* de eventos de taxação.

A informação relevante para taxação é então mapeada para mensagens *Diameter*, sendo colocada dentro de AVPs. As mensagens aplicadas para taxação *offline* do IMS são *accounting request* (ACR) e *accounting answer* (ACA), estando definidas no protocolo base *Diameter*. As mensagens ACR podem ser de três tipos [start], [interim] e [stop] para o caso de sessões SIP válidas e do tipo [event] para o caso de erros em sessões SIP ou para o caso de mensagens SIP que não geram sessões. As diferentes mensagens de sinalização que o operador poderá utilizar para gerar eventos de taxação estão listadas na especificação [62] (ver Tabela 5).

Tabela 5 – Mensagens de *Accounting* "trigadas" por métodos SIP ou mensagens ISUP para todos os nós IMS à excepção do MRFC e do AS [62]

Mensagem *Diameter*	**Método SIP/Mensagem ISUP**
ACR [Start]	SIP 200 OK a confirmar um pedido inicial SIP INVITE
	ISUP:ANM (aplicável ao MGCF)
ACR [Interim]	SIP 200 OK a confirmar uma mensagem SIP RE-INVITE ou SIP UPDATE [por exemplo, alteração dos *codecs* durante uma sessão já estabelecida]
	Expiração de um AVP [Acct-Interim-Interval[12]]
ACR [Stop]	SIP BYE (tanto em situações normais como anormais)
	ISUP:REL (aplicável ao MGCF)
ACR [Event]	SIP 200 OK confirmando uma mensagem SIP que não cria sessão, que são: SIP NOTIFY SIP MESSAGE SIP REGISTER SIP SUBSCRIBE
	SIP 3xx Resposta para redireccionamento
	SIP Final Response (4xx, 5xx or 6xx)
	SIP Final Response (4xx, 5xx or 6xx)
	SIP CANCEL
	I-CSCF completando uma Cx Query[13] que foi solicitada em resposta a um SIP INVITE

Tabela 6 – Mensagens de Accounting "despoletadas" por métodos SIP para o MRFC [62]

Mensagem *Diameter*	**Trigger**
ACR [Start]	SIP 200 OK confirmando um pedido SIP INVITE para inicializar uma sessão ad hoc multimédia de conferência
ACR [Interim]	SIP ACK confirmando um SIP INVITE para conectar o utilizador a uma sessão de conferência
	Expiração do AVP [Acct-Interim-Interval]
ACR [Stop]	Mensagem SIP BYE
	Respostas finais SIP com códigos de erros 4xx, 5xx ou 6xx indicando o fim de uma sessão a decorrer

Como é possível observar nas tabelas anteriores, os ASs não se enquadram em nenhuma das tabelas, pois depende de quem implementa o serviço. Mesmo assim, os ASs suportam

[12] Este AVP é enviado pelo servidor *Diameter* ao cliente, para este decidir como e quando produzir *accounting records*.

[13] Interface com o HSS para autorizar o utilizador e seleccionar um S-CSCF

os quatro tipos de ACR (Start/Interim/Stop/Event). A utilização do ACR [Start], [Interim] e [Stop] (*Session Charging*) versus ACR [Event] (*Event Charging*) depende dos serviços fornecidos pelo AS [62].

Estas mensagens são então enviadas para o CDF, podendo o seu endereço ser obtido através da informação presente na sinalização SIP (campo *P-Charging-Function-Addresses*) ou estar pré-configurado pelo operador, no elemento IMS respectivo.

O CDF utiliza a informação recebida para gerar *Charging Data Records* (CDRs) com uma estrutura e conteúdo normalizado pelo 3GPP. Através do protocolo GTP' (derivação para taxação do *GPRS Tunnelling Protocol* (GTP)), o CDF envia o CDR criado para o CGF (interface Ga). Este último, actua como uma *gateway* da rede 3GPP com o BD, tendo como tarefas realizar o processamento prévio do CDR recebido, isto é, validar, reformular (passar para ficheiro) e armazená-lo de forma persistente, utilizando, de seguida, a interface Bx para enviar o CDR para o BD. O BD, onde o *rating* acontece para a taxação *offline*, não se encontra definido pelo 3GPP, sendo da responsabilidade do operador realizar essa tarefa de acordo com as suas preferências.

4.2.2 *Taxação Online em IMS*

Como é possível observar-se na Figura 32, apenas duas entidades lógicas participam no processo de taxação *online*, o CTF e o *Online Charging System* (OCS).

Online Charging é um processo onde a autorização para a utilização dos recursos da rede deverá ser previamente obtida, isto é, antes do fornecimento de um determinado recurso da rede ocorrer. Esta autorização é obtida através do OCS após um pedido recebido por parte da rede, do CTF. Ao receber um pedido para a utilização dos recursos, a rede reúne a informação de tarifação relevante e gera um evento de tarifação em direcção ao OCS, isto tudo em tempo real. Como resposta, o OCS retorna a autorização apropriada para a utilização do recurso. A autorização poderá estar limitada a um determinado nível (exemplo, volume de dados ou duração), podendo ter de ser necessário, por esse facto, a renovação de tempo a tempo enquanto a utilização do recurso persistir [63].

O OCS é composto por três funções lógicas, o *Account Balance Management Function* (ABMF), o *Rating Function* (RF) e o *Online Changing Function* (OCF).

O ABMF é o módulo dentro do OCS onde se encontra o saldo da conta da subscrição. O RF, por seu lado, determina o valor a taxar ao serviço de acordo com o recursos de rede a serem utilizados. O OCF, única função que interage directamente com o CTF, ainda se subdivide em duas funções, sendo elas o *Session-Based Changing Function* (SBCF) e o

Event-Based Charging Function (EBCF). O primeiro tem como tarefa lidar com todos os pedidos de recursos baseados em sessões, enquanto o segundo é responsável pelas actividades dos utilizadores relativas a eventos. Ambas as funções deverão garantir a execução correcta das respectivas transacções de taxação, isto é, realizam o controlo sobre o consumo dos recursos da rede baseado nas sessões ou eventos, através dos pedidos de taxação *online Diameter* enviados pelos CTFs, realizando as acções correspondentes na conta do subscritor, tendo o auxilio do RF para determinar o que taxar.

Ao contrário do processo para taxação *offline*, o CTF que suporte taxação *online* deverá suportar interacções em tempo real com o OCS através da interface Ro. A utilização dos recursos de rede só deverá acontecer após a autorização do OCS. Para tal, terá de monitorizar o consumo dos recursos e ainda, como última funcionalidade, deverá conseguir terminar/bloquear o acesso ao recursos da rede no caso de não ser autorizado o seu acesso tanto nos pedido iniciais como nos pedidos adicionais (reserva de mais recursos numa sessão já a decorrer).

Tal como na interface Rf, o protocolo escolhido para a interface Ro é o *Diameter*. O 3GPP adoptou a *Diameter Credit-Control Application* [42] tendo sido estendida com o acréscimo de AVPs específicos para esta interface [64]. As mensagens *Diameter* trocadas entre o CTF e o OCF são *Credit-Control-Requests* (CCR) e *Credit-Control-Answers* (CCA).

O número de entidades funcionais que poderão desempenhar o papel de CTF para taxação *online* resume-se a três elementos: o S-CSCF, o MRFC e o AS (Figura 35).

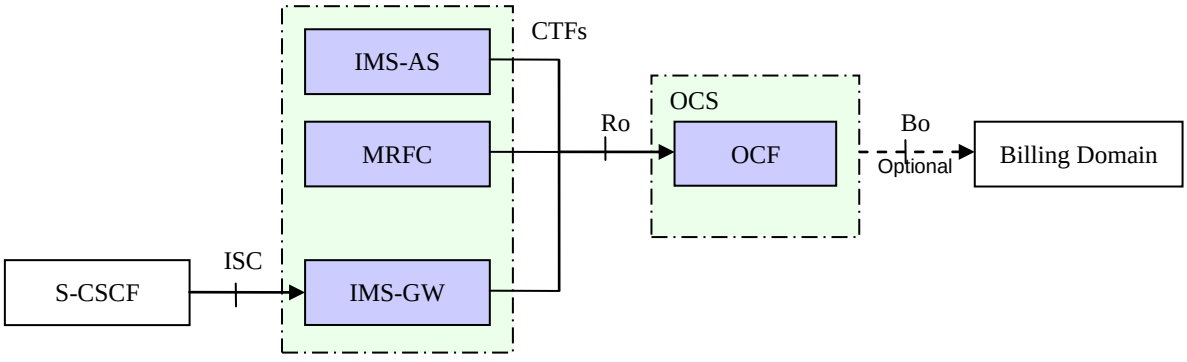

Figura 35 – Arquitectura *Online Charging* IMS [62]

Percebe-se pela figura, que o S-CSCF não interage directamente com o OCS. Essa função é desempenhada por uma *gateway*, a IMS *Gateway Function* (IMS-GW) para interacções de *credit-control* com o OCS. A IMS-GW actua como um CTF no que diz respeito à interacção com o OCS, mas para o S-CSCF, é visto como AS que interage utilizando o protocolo SIP. O IMS-GW tem como função traduzir a informação SIP recebida para *Diameter*. Este processo de taxação online é transparente para o S-CSCF sendo o IMS-GW

invocado utilizando a interface ISC. O IMS-GW é visto pelo S-CSCF como um qualquer outro serviço, sendo o pedido enviado para o IMS-GW através da definição dos iFCs que especificam quando é invocado o respectivo serviço (ver Anexo B onde é explicada a interacção entre o S-CSCF e o AS).

Para que as mensagens SIP cheguem aos ASs (aqui estou a incluir também o MRFC e o IMS-GW só para uma questão de leitura) terão de se encontrar definidos nos iFCs que eventos poderão ser considerados como válidos para taxação *online*. O comportamento de *triggering* de pedidos de *Online Charging* não se encontra especificado, dependendo do serviço a taxar.

Para o caso do IMS-GWF a lista dos eventos possíveis de taxação online são apresentados na Tabela 7. No caso do MRFC os *triggers* apenas se encontram definidos a partir da *release* 7 e no caso do AS os *triggers* ainda não se encontram definidos (é possível encontrar na norma [62] a indicação FFS (*For Further Study*)). Desta forma, o operador deverá criar os iFCs da forma que mais lhe convenha, tendo em mente o tipo de serviço.

Tabela 7 – Mensagens CCR "trigadas" por métodos SIP para o IMS-GWF [62]

Mensagem *Diameter*	Método SIP
CCR [Initial]	SIP INVITE (SCUR)
	SIP NOTIFY (ECUR)
	SIP MESSAGE (ECUR)
	SIP REGISTER (ECUR)
	SIP SUBSCRIBE (ECUR)
	SIP REFER (ECUR)
	SIP PUBLISH (ECUR)
CCR [Update]	SIP 200 OK confirmando um SIP INVITE, RE-INVITE ou SIP UPDATE [por exemplo, mudança nos components de *Media*] (SCUR)
	RE-INVITE ou SIP UPDATE [por exemplo, mudança nos components de *Media*] (SCUR)
	Expiração da quota reservada, o tempo válido expirou ou trigger de autorização (SCUR)
CCR [Terminate]	Mensagem SIP BYE (tanto para os casos de fim de sessões normais ou anormais) (SCUR)
	SIP 200 OK confirmando mensagens SIP não relacionados a sessões (ECUR)
	Abortar um processo de *set-up* de uma sessão SIP, utilizando um trigger interno, ou um SIP CANCEL.(SCUR/ECUR)
	Deregisto (SCUR/ECUR)
	Resposta Final SIP 2xx (incluindo respostas 202 a pedidos REFER, excepto SIP 200 OK) (ECUR)
	Respostas Final/Redireccionamento 3xx (SCUR/ECUR)
	Resposta Final SIP (4xx, 5xx ou 6xx), indicando algum erro num procedimento relacionado com o set-up de uma sessão (SCUR)
	Resposta Final SIP (4xx, 5xx ou 6xx), indicando algum erro num procedimento não relacionado com sessões (ECUR)
CCR [Event]	SIP NOTIFY (IEC)
	SIP MESSAGE (IEC)
	SIP REGISTER (IEC)
	SIP SUBSCRIBE (IEC)
	SIP REFER (IEC)
	SIP PUBLISH (IEC)
	Resposta Final SIP (4xx, 5xx ou 6xx), indicando algum erro num procedimento não relacionado com sessões (IEC)

Pela Tabela 7 é possível constatar que mensagens SIP não iniciais (exemplo, UPDATE) podem gerar eventos de taxação. O problema é que os pedidos "despoletados" apenas podem ser pedidos iniciais, como é o caso do INVITE. Para resolver esta situação, os ASs terão de se adicionar no caminho da sinalização, passando assim todos os pedidos subsequentes de sinalização pelo AS (ver protocolo SIP e o papel do cabeçalho *Record-Route*).

Para a interacção entre o CTF e o OCF através da interface Ro, é possível distinguir três diferentes tipos de taxação *online* que, de acordo com os eventos que chegam da rede, influenciam o tipo de mensagens de CCR trocadas por estes elementos:

- *Immediate Event Charging* (IEC);
- *Event Charging with Unit Reservation* (ECUR);
- *Session Charging with Unit Reservation* (SCUR);

Estas mensagens irão ser recebidas e tratadas pelas duas funções existentes no OCF, sendo os métodos de interacção IEC e ECUR, por um lado, da responsabilidade do EBCF, sendo a mensagem SCUR, por outro, da responsabilidade do SBCF.

No caso do IEC (Figura 36), as mensagens trocadas são do tipo CCR [event] e permitem, por exemplo, o débito directo de uma quantidade de unidades (p.ex. dinheiro, tempo) na conta do utilizador. Este tipo de operação tem como característica trocar apenas um par de mensagens CCR/CCA. São operações realizadas num só passo e que não relacionam com mais nenhum evento que possa ocorrer entre o CTF e o OCF.

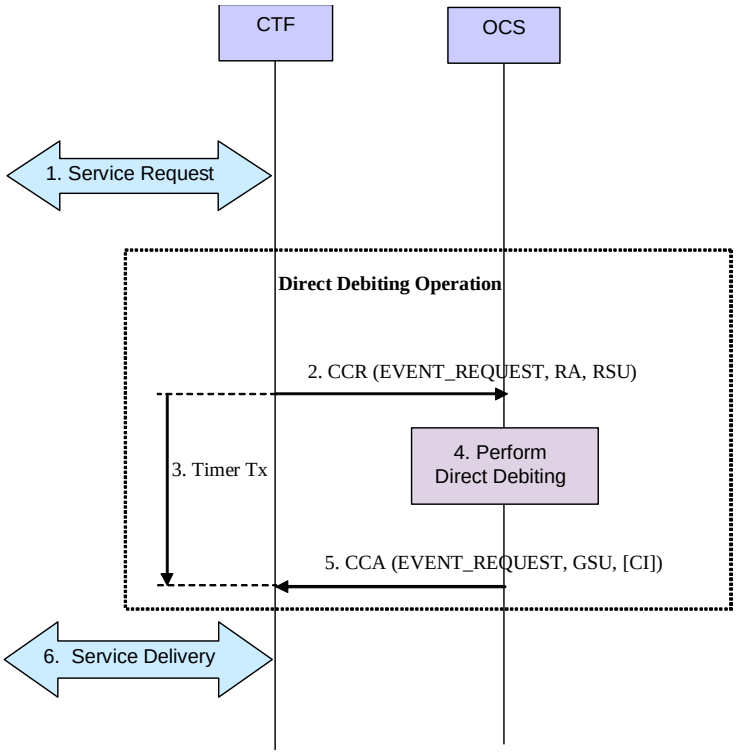

Figura 36 – Exemplo IEC para débito directo [64]

No caso do ECUR (Figura 37) e do SCUR (Figura 38), as mensagens enviadas pelo CTF podem ser do tipo CCR [initial] e CCR [terminate] (um par de mensagens CCR/CCA de cada tipo, por pedido), podendo o SCUR, como opção, enviar uma ou mais mensagens do tipo CCR [update], em situações em que a quantidade de unidades pedidas expirou ou é necessário reservar mais unidades. Nestas duas situações, ECUR e SCUR, e tal como o nome indica, ambos os processos efectuam uma reserva de unidades na conta do utilizador, isto é, o débito na conta do utilizador não é imediatamente executado, mas sim após o serviço ser realizado, e custo total ser calculado, que poderá não ser possível antemão (por exemplo, no caso de serviços em que a taxação é baseada no tempo que demora uma dada sessão). Com a mensagem CCR [initial], o CTF solicita uma quantia inicial de unidades que poderá ser autorizada ou não através da mensagem CCA equivalente. Este pedido inicia uma nova sessão de Credit-Control. Como mencionado anteriormente, no caso do método SCUR, é possível trocar diversas subsequentes mensagens CCR/CCA [update] quando a sessão decorre e os recursos são gastos. Quando a sessão do utilizador finalmente

termina ou o serviço baseado em eventos foi entregue, é enviado uma mensagem CCR [terminate] a terminar a sessão. Isto inclui o débito na conta do utilizador e a libertação de recursos reservados que não foram gastos.

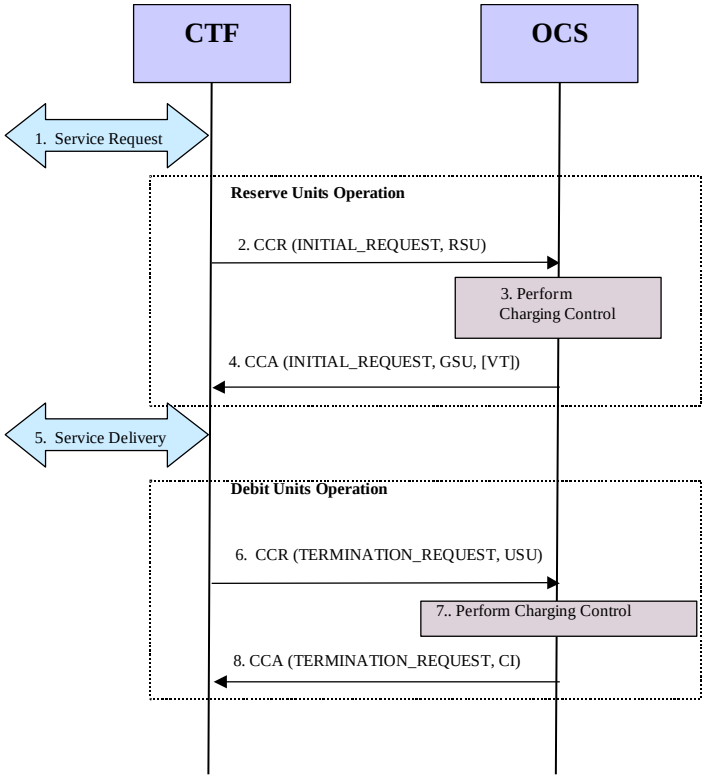

Figura 37 – Exemplo ECUR [64]

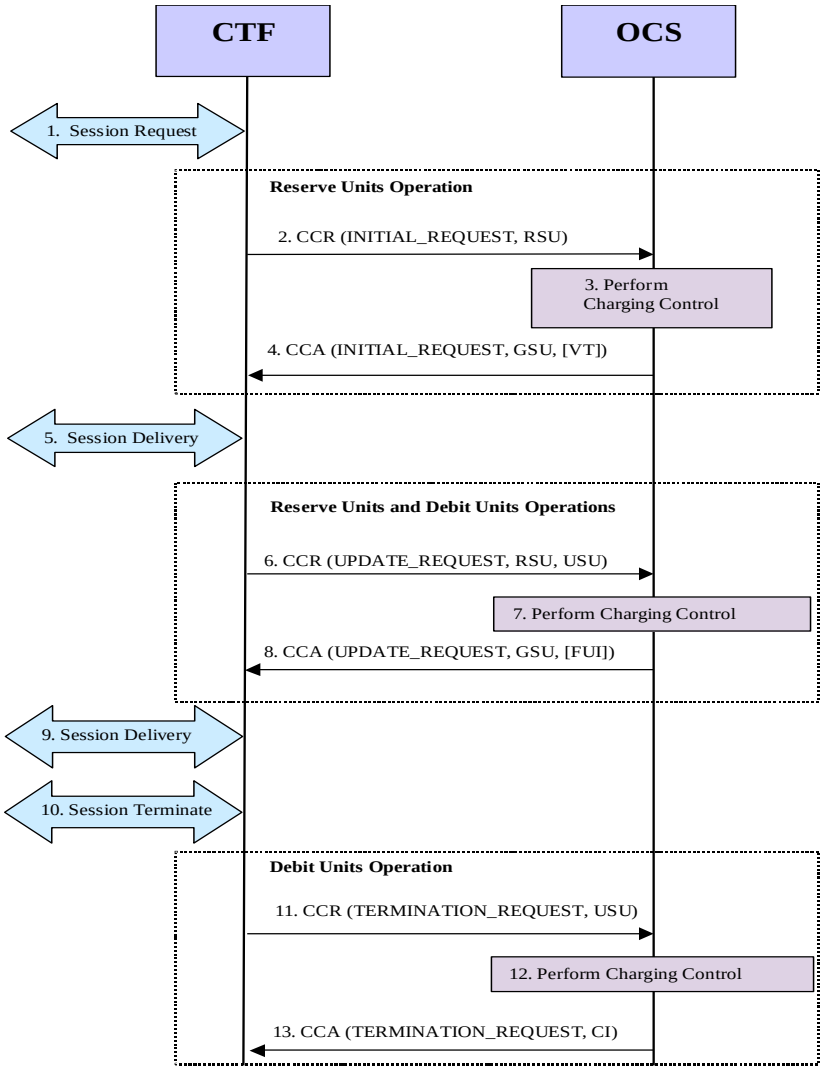

Figura 38 – Exemplo SCUR [64]

Como o CTF actua no caso de receber uma resposta negativa por parte do OCS é da responsabilidade do operador. Uma possibilidade seria o IMS-GWF actuar como um SIP B2BUA, enviando uma mensagem SIP BYE para os dois elementos terminais da sessão, deste modo terminando a sessão e libertando os recursos da rede. Para tal acontecer, o OCS inclui uma indicação na mensagem CCA (AVP especifico para este propósito *Final-Unit-Reservation*), quando as últimas unidades do utilizador se encontram disponíveis. Outra forma de actuar, é reencaminhar a sessão para um servidor que permita recarregar a conta de um utilizador.

O OCS poderá ainda gerar CDRs caso o operador deseje processar essa informação posteriormente. Nesta situação, o OCS poderá incluir um CDF de modo a gerar os CDRs que poderão conter a informação do que foi taxado, isto é, o resultado dos processos de taxação e *rating* por parte do OCS. O CGF, que nesta situação está presente no OCS, gera os ficheiros CDR e fornece esses ficheiros ao BD. Para o IMS, esta informação apenas se

encontrava disponível na *release* 5 do 3GPP. Apesar disso, no 3GPP *release* 6, toda a informação necessária para criar os S-CSCF-CDRs está incluída na mensagem CCR enviada pelo OCS pelo IMS-GWF.

4.2.3 *Extensões Diameter para taxação no IMS*

No caso de taxação em IMS, foram especificadas, como foi dito anteriormente, duas novas aplicações *Diameter* que estendem a aplicação Base. A interface Rf, definida em [64], utiliza a parte de *accounting* do protocolo base e o seu servidor actua no modo *stateless*. A aplicação Ro, por seu lado, definida também em [64], estende a *Diameter Credit Control Application* (DCCA) [42], que utiliza tanto a parte *stateful* como a *stateless* do modo de autorização do protocolo base (Figura 39).

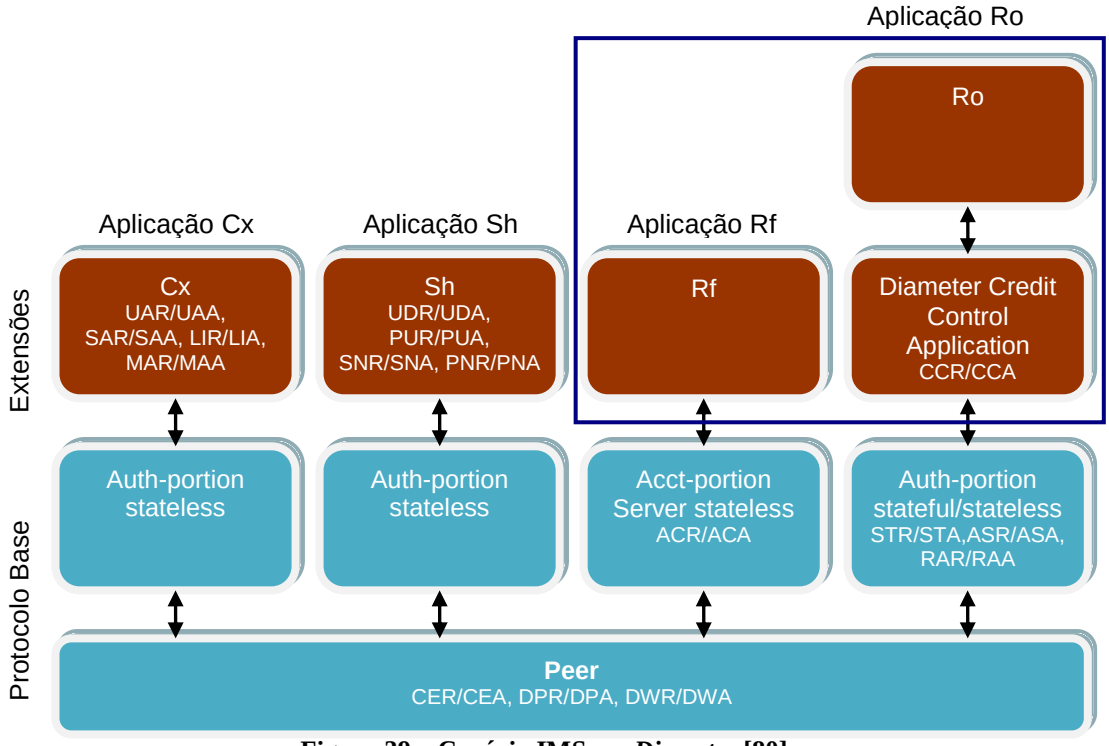

Figura 39 – Cenário IMS em *Diameter* [80]

4.2.4 *Policy and Charging Control (PCC)*

Devido ao grande aumento de serviços de dados e multimédia, os operadores deverão encontrar uma solução que lhes permita aumentar as receitas com os seus serviços, bem como fornecer diferentes tipos de serviços para atrair mais clientes. Com o acréscimo de utilizadores, os recursos da rede tornam-se limitados, tornando-se necessário encontrar um mecanismo que forneça diferentes níveis de QoS para os subscritores com planos de tarifação diversificados. Os operadores precisam de assegurar a reserva suficiente de recursos para um dado serviço, definindo serviços diferenciados de acordo com os níveis

de QoS subscritos para os utilizadores. Deste modo, os operadores podem direccionar os subscritores e os fornecedores de serviços para serviços com uma garantia de QoS.

Outra preocupação relevante para os operadores prende-se com o facto de no futuro próximo, o domínio dos serviços de voz tradicionais, ser ultrapassado por serviços de dados, sendo necessário encontrar uma forma de tarifação adequada a cada tipo de serviço. Para taxar de acordo com o conteúdo, é necessário para os operadores analisarem a informação presente nos pacotes de dados que chegam e saem para cada subscrição e distinguir o tipo de conteúdo como dados, áudio, vídeo ou mesmo texto. Esta informação é enviada para o sistema de *charging* onde os subscritores são tarifados de acordo com o conteúdo.

Só desta forma será possível os operadores recuperarem as perdas para as tecnologias tipo VoIP, onde o operador já não tem o domínio tanto da rede, como de todos os serviços.

O PCC proporciona uma solução a estes problemas. Através desta nova estrutura, definida na *release* 7, é possível integrar tanto a informação relativa à tarifação, como a políticas de QoS. Com o PCC, é possível reduzir o custo de OPEX[14] e CAPEX[15], simplificando a estrutura da rede e requerendo um menor número de manutenção e custos.

4.2.4.1 Integração do PCC em IMS

A estrutura do PCC é definida no 3GPP *release* 7. Baseado no *Flow Based Control* (FBC) definido na *release* 6, o PCC adopta ainda a função *Session-Based Local Policy* (SBLP). Deste modo, a integração de politicas de QoS e controlo de tarifação é alcançada.

4.2.4.2 QoS Policy Control

O QoS *policy control* baseado em sessões está definido no 3GPP *release* 5. De acordo com as diferentes aplicações existentes (voz, vídeo, etc.), bem como políticas de controlo locais definidas pelo operador, o IMS pode controlar os recursos de rede IP ocupados por uma aplicação específica. Por exemplo, o IMS pode controlar e gerir a largura de banda reservada a uma aplicação e gerir a sua prioridade.

O processo de controlo de recursos no IMS definido na *release* 5 é da seguinte forma: Durante o *setup* inicial da sessão (por exemplo, o início de uma chamada), o UE pede à rede a reserva de determinados parâmetros multimédia (como *codecs*, tipo de *Media*, e largura de banda), através do SDP contido na mensagem SIP.

[14] OPEX – *Operational Expenditure*, que significa o capital utilizado para manter os bens físicos de uma empresa.

O P-CSCF reencaminha de seguida esses parâmetros do *Session Description Protocol* (SDP) para o PDF, através da interface Gq. Como já foi explicado anteriormente, o PDF é na *release* 5 uma entidade lógica do P-CSCF e na *release* 6 é uma função *stand-alone* responsável pela gestão dos recursos da rede.

O PDF autoriza os parâmetros multimédia de acordo com as mensagens multimédia trocadas pelo utilizador e pelas políticas locais de QoS.

Após a autorização, os parâmetros multimédia autorizados são devolvidos ao UE e os recursos para configurar a largura de transmissão são reservados.

O PDF reencaminha esses parâmetros de controlo da QoS para o *Gateway GPRS Support Node* (GGSN), através da interface Go (caso particular de acesso GPRS, que foi o inicialmente previsto pelo 3GPP nas primeiras *releases* da arquitectura IMS).

Como dispositivo que executa as politicas de controlo de QoS, o GGSN analisa os endereços IP de origem e destino, controlando e filtrando a informação IP que circula. A informação de *policy* de QoS é controlada pelo PDF.

4.2.4.3 Flow Based Control (FBC)

A estrutura do FBC é definida pelo 3GPP *release* 6 [65]. Esta estrutura resolve o problema de tarifação ao nível da camada de rede/transporte. As interfaces Gy/Gz[16] são utilizadas para enviar a informação de tarifação tanto para os sistemas de tarifação *online* como *offline*, respectivamente. A informação de taxação inclui: o tipo de sessão, tempo do inicio da sessão, tempo do fim da sessão e a informação respeitante ao fluxo do serviço.

O processo é realizado da seguinte forma:

- O utilizador envia uma mensagem SIP para o P-CSCF;
- O P-CSCF notifica o *Charging Rule Function* (CRF) da informação relativa ao fluxo de dados, através da interface Rx, incluindo o identificador do fluxo IP, a largura de banda e valor de QoS;
- O CRF envia então essa informação para o GGSN, incluindo os detalhes do fluxo IP e a chave de tarifação (informação usada pelo sistema de taxação *online* e *offline* para questões de *rating*);
- O GGSN envia os dados necessários de tarifação para os sistemas de tarifação, de acordo com a chave de taxação.

[15] CAPEX – *Capital Expenditure*, que significa o capital utilizado para adquirir ou melhorar os bens físicos de uma empresa.

[16] Na perspectiva da arquitectura geral de *Charging* os pontos de referência Gy e Gz são funcionalmente equivalentes às interfaces Ro e Ga, respectivamente.

4.2.4.4 PCC – Solução de integração

Comparando as duas estruturas mencionadas anteriormente, quando o *Policy Control* e o FBC são utilizados como dois elementos isolados, estes têm as suas próprias interfaces e entidades funcionais. Apesar disso, ambas as entidades funcionais bem como o conteúdo das mensagens das interfaces são similares.

A separação do *Policy Control* e do FBC irá incrementar a complexidade da estrutura da rede e do fluxo das mensagens (devida à interligação entre os elementos de rede, tais como o GGSN, P-CSCF com as entidades como o CRF e o PDF), assim como uma degradação da performance e eficiência. Por este facto, o CRF e o PDF deverão ser integrados num único dispositivo, sendo a interface estendida de forma a suportar a interligação das duas funcionalidades.

Na Figura 40 encontra-se a solução a este problema, estando representada a *framework Policy and Charging Control* (PCC) que permite agregar as funcionalidades de controlo de taxação e dos recursos da rede. Esta arquitectura pretende ser genérica, podendo ser adoptada por arquitecturas diferentes do IMS. Nesta figura podemos observar a integração do PDF e o CRF num único elemento formando *o Policy and Charging Rule Function* (PCRF). O PCRF é o elemento que aplica as *policy rules* diferenciando-as por serviço e/ou sessão. É o PCRF que comunica com a entidade de rede, o *Policy and Charging Enforcement Function* (PCEF), impondo as decisões de controlo de admissão. O PCEF aplica então as decisões, controlando os recursos e efectuando o controlo de admissão, garantindo a QoS adequada para o tipo de serviço solicitado. Este elemento é o resultado da integração do *Policy Enforcement Point* (PEP) e o *Traffic Plane Function* (TPF). No caso de um acesso GPRS, por exemplo, o PCEF iria encontrar-se no GGSN, já que é o elemento da rede que controla o fluxo dos dados. Também presente na figura encontra-se o *Application Function* (AF), que é o elemento que faz os pedidos de reserva de recursos, recebendo ainda as notificações relativas a esta reserva e libertação de recursos. Fazendo um paralelismo para o IMS, já que o PCC pretende ser uma *framework* genérica, o AF corresponde ao P-CSCF.

As interfaces Gq e Rx são integradas formando a Rx+, e as interfaces Go e Gx são integradas formando a interface Gx+.

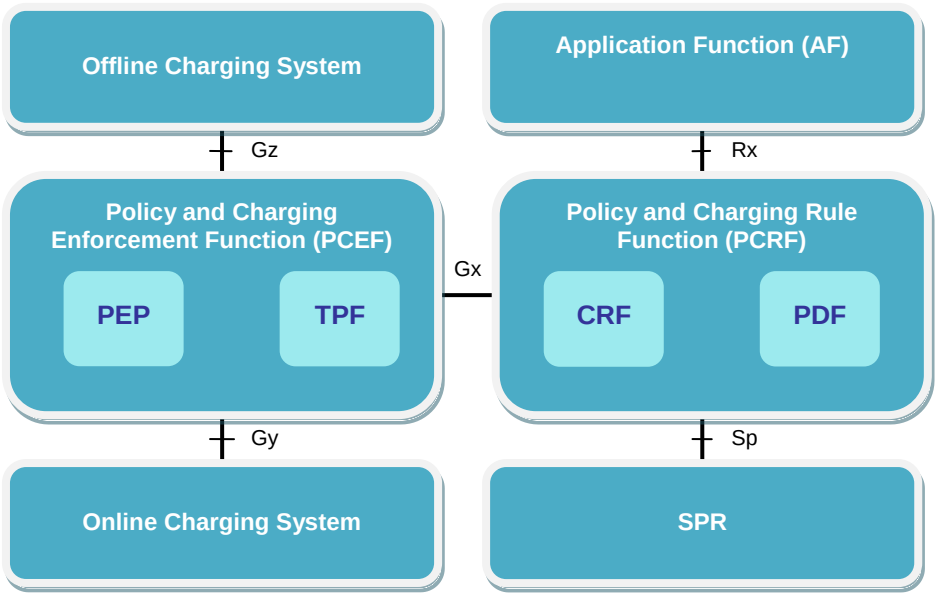

Figura 40 – Solução de integração PCC

Adicionalmente, foi definida a função lógica *Subscription Profile Repository* (SPR) que irá permitir armazenar o perfil de rede do utilizador e enviar essa informação para o PCRF através da interface Sp. A adição do SPR permite aumentar a capacidade do PCRF para controlar a QoS. Através deste novo elemento, o PCRF não permite apenas fornecer o controlo de serviços baseados na QoS, como ainda controla a informação de subscrição baseada na QoS (isto é, politicas de QoS baseadas na subscrição e não no serviço). Exemplos da informação fornecida pelo SPR serão a lista de serviços autorizados pelo utilizador e a informação da QoS permitido para o utilizador.

Capitulo 5 – Demonstrador

Todo o trabalho esteve enquadrado na criação dos elementos que iriam permitir realizar a taxação a dois serviços da PT Inovação no contexto das redes de próxima geração, o serviço Vídeo Portal e o serviço PC2Phone.

O serviço de Vídeo Portal tinha como funcionalidade, e tal como se percebe pelo nome do serviço, fornecer a um utilizador da arquitectura IMS um portal multimédia que lhe permitia visualizar vídeos.

Este portal consistia num leque de vídeos, cada um com o seu custo de visualização. Ao tentar visualizar os vídeos, o serviço de vídeo portal validaria o saldo do utilizador, deste modo autorizando ou não a utilização dos recursos da rede, e claro, a visualização do vídeo. Como é possível observar, e dado que visualização do vídeo depende do saldo do utilizador, estamos na presença de um serviço que realiza taxação *online*. Também se constata que, sendo o serviço taxado ao conteúdo e numa única transacção, estamos na presença de taxação *online* no modo IEC, isto é, a operação é realizada num só passo, quando o utilizador tentar aceder aos vídeo. Caso o saldo seja suficiente, o utilizador terá acesso ao vídeo sem mais nada lhe ser cobrado.

Pelo contrário, no caso do PC2Phone, estamos na presença de um serviço baseado em sessão, isto é, o objectivo deste serviço é taxar o utilizador minuto a minuto por realizar chamadas da rede IMS para a PSTN. Desta forma, o modo de taxação *online* para este serviço é baseado na reserva periódica de unidades até a chamada terminar ou o utilizador esgotar o seu saldo, taxação SCUR.

É importante referir que o trabalho desenvolvido, o Enabler de *Charging* e o RA *Diameter*, teve como principal função suportar a taxação dos serviços Vídeo Portal e PC2Phone, mas foi criado para ser genérico e deste modo puder ser aplicado a qualquer outro serviço que corra no SIP AS num ambiente JSLEE.

5.1 Arquitectura do Sistema

O serviço de Vídeo Portal (Figura 41) é um módulo/serviço utilizado sobre a plataforma SIP AS. Este sistema comunica com a plataforma Inovox (solução da PTIN que funciona como MRFC/MRFP) para possibilitar o toque de anúncios. Este serviço controla a plataforma Inovox e notifica o *Enabler* de *Charging*, para que o utilizador seja taxado pelos vídeos que escolhe. Este, por sua vez, comunica com o servidor de *Charging*, através do RA *Diameter*.

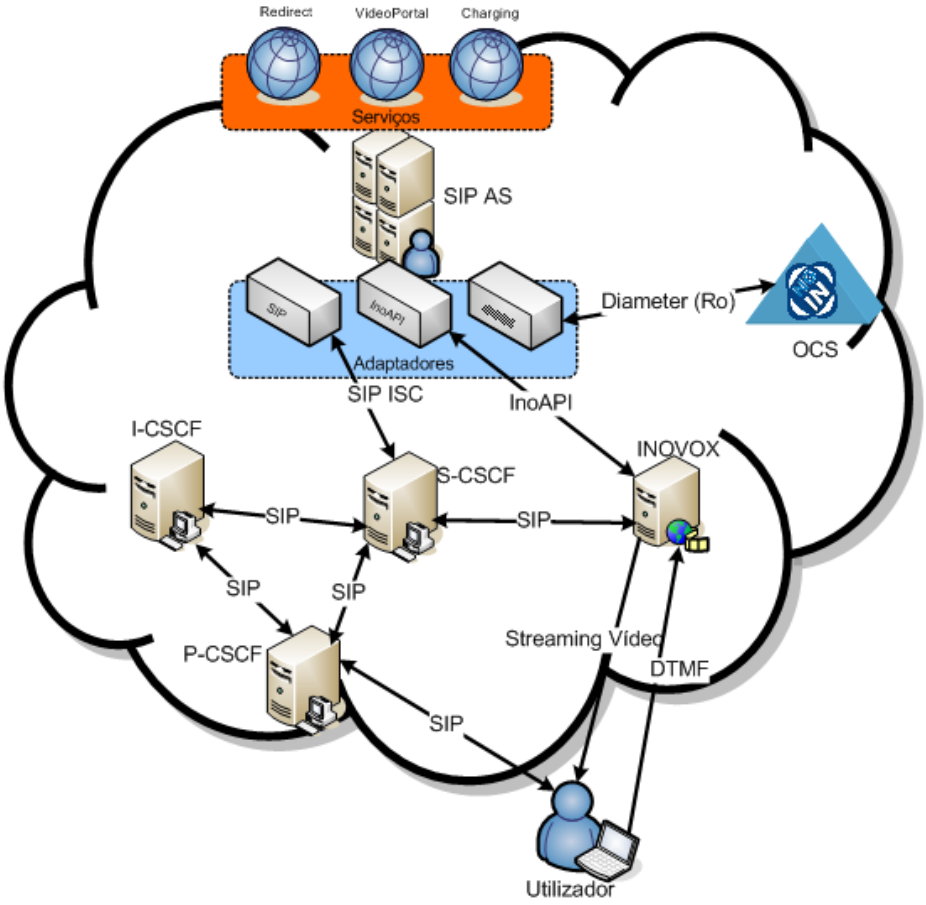

Figura 41 – Arquitectura do Vídeo Portal

Para aceder ao serviço de Vídeo Portal, o utilizador inicia uma sessão multimédia enviando um SIP INVITE para o endereço "sip:videoportal@ptinovacao.pt" (endereço configurável). Este pedido é reencaminhado pelo core IMS chegando ao AS, através dos *triggers* configurados no HSS. Ao recebê-lo, o AS que funciona como um servidor SIP *Redirect* (serviço *Redirect*) responde com um *302 Moved Temporarily* com o endereço onde está o Inovox. Através do RA da InoAPI (protocolo proprietário da PTIN) o Vídeo Portal controla os eventos que são enviados pela Inovox e, usando o *Enabler Diameter*, é controlada a tarifação aplicada a este serviço.

Durante a utilização do serviço de Vídeo Portal, o utilizador tem acesso a um menu inicial com possíveis escolhas de vídeos para visualização (Figura 42). Durante este período o utilizador pode efectuar escolhas relativas aos filmes existentes e este processo tem de ser validado pelo sistema de taxação. Mediante a informação proveniente do sistema, o serviço de Vídeo Portal fornece o vídeo pedido ou uma mensagem indicativa da impossibilidade de visualização do vídeo devido à inexistência de saldo. A escolha dos vídeos por parte do utilizador é realizada premindo no terminal o número do vídeo. Ao premir, o terminal envia no RTP o caracter DTMF com a escolha realizada. Na Figura 41 esse procedimento é

representado pela ligação entre o utilizador e o Inovox. Como alternativa, e caso suportasse, o cliente poderia enviar essa informação através de um pedido SIP INFO, estando o conteúdo no corpo da mensagem.

Figura 42 – Portal de Vídeo com a lista de opções

O serviço PC2Phone tem como finalidade taxar sessões a utilizadores na rede IMS que pretendam realizar chamadas com utilizadores da PSTN (Figura 43).

Ao iniciar uma sessão, o S-CSCF irá reencaminhar o pedido SIP INVITE para o serviço PC2Phone que irá proceder à reserva de unidades através do *Enabler* de *Charging*, caso o utilizador tenha saldo suficiente. Após a reserva inicial de unidades (neste serviço as unidades são tempo), o *Enabler* de *Charging* contem um *timer* que permite, de forma periódica, reservar mais unidades e indicar as que foram consumidas pelo utilizador ao servidor de *Charging*. Quando o utilizador desliga a chamada, as unidades reservadas e não utilizadas são novamente creditadas no saldo do utilizador.

Caso o utilizador fique sem saldo antes de terminar a chamada, o serviço terá de funcionar como um servidor SIP *Back2BackUA* para garantir o controlo da sessão, podendo desta forma desligar a chamada.

Como se pode observar na Figura 43 as chamadas estão a ser encaminhadas directamente do S-CSCF para a MGCF. Teoricamente, esta interacção deveria incluir um ponto intermédio, o BGCF, que teria como função realizar a escolha do MGCF. Como a rede montada apenas continha uma MGCF, o endereço deste elemento estava aprovisionado estaticamente no S-CSCF.

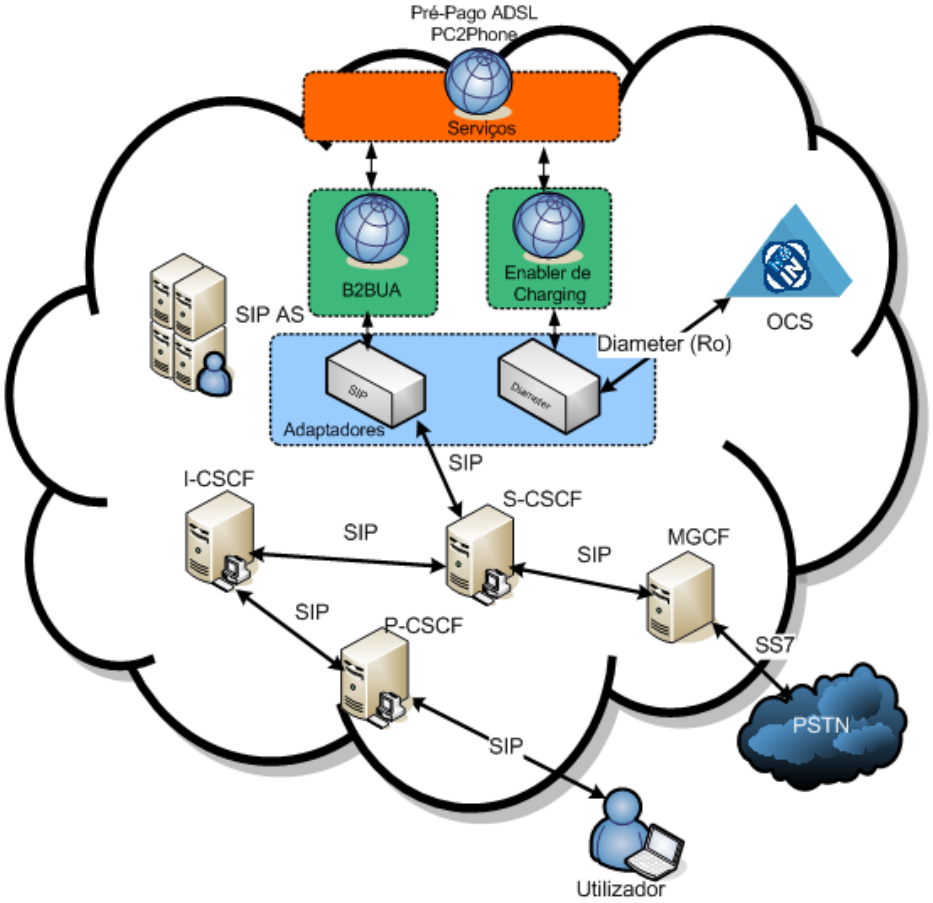

Figura 43 – Arquitectura Serviço PC2Phone

5.2 Ferramentas Utilizadas

Este módulo apresenta as ferramentas utilizadas para a criação do demonstrador, *Mobicents* e *Intelligent Diameter Stack* (IDS).

O *Mobicents* [68] é um AS baseado em eventos altamente escalável com um modelo de componentes robusto e um ambiente de execução *fault tolerant*. Trata-se da primeira, e para já única, plataforma VoIP *Open Source* a conseguir ser *compliant* com o standard JSLEE 1.0. No contexto das redes de próxima geração, o *Mobicents* enquadra-se como um ambiente de execução altamente performance para SDP e IMS.

A IDS é uma stack *Diameter*, implementada em Java, proprietária da PTIN. Esta ferramenta permitiu criar o RA *Diameter* na plataforma *Mobicents*, utilizado para implementar a interface Ro da arquitectura IMS de forma a interagir com o sistema de taxação.

5.2.1 *Mobicents*

O *Mobicents* é uma plataforma *Open Source* totalmente *compliant* com o JSLEE 1.0, implementando ainda algumas das funcionalidades propostas para o JSLEE 1.1.

Trata-se de uma plataforma que fornece às aplicações de telecomunicações um robusto ambiente de execução e modelo de componentes.

No domínio das redes de próxima geração, o *Mobicents* enquadra-se como uma plataforma de distribuição de serviços de elevada performance para SDPs e IMS. O Mobicents permite a composição de SBBs com funcionalidades importantes tais como controlo de chamada, controlo de custo, aprovisionamento dos dados do utilizador, administração e presença.

Ainda para *Mobicents*, encontra-se disponível a ferramenta gráfica EclipSLEE, que permite, de um modo simples, a criação e rápido desenvolvimento de serviços para JSLEE. Trata-se de uma ambiente de criação de serviços, disponível como *plug-in* para o Eclipse.

Para além das telecomunicações o *Mobicents* pode servir para outros problemas que necessitem de sinalização com elevado volume e baixa latência.

A implementação do *Mobicents* corre como um serviço do *JBoss* (Figura 44). O JSLEE não é uma especificação J2EE, contudo usa muitos dos seus componentes tal como *Java Management Extensions* (JMX[17]), JNDI. O *JBoss AS* é uma plataforma de construção de aplicações que fornece muitas destas facilidades, donde se destacam:

- *JBoss Cache*;
- JMX;
- JNDI;
- *JavaAssist*;
- *JBoss Clustering*.

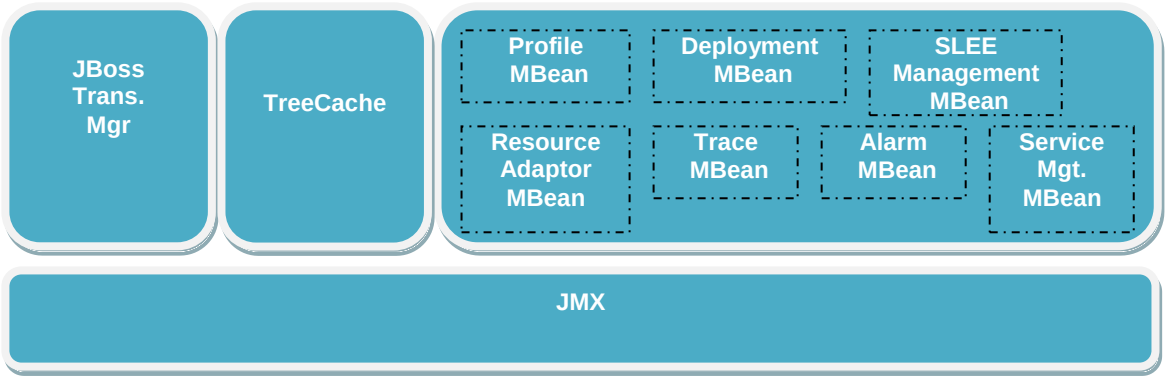

Figura 44 – Mobicents como serviço *JBoss*

[17] A tecnologia JMX fornece ferramentas para criar aplicações para gerir e monitorizar dispositivos, aplicações, serviços, etc.

5.2.2 *Intelligent Diameter Stack (IDS)*

A IDS é uma stack *Diameter* proprietária da PTIN com o principal objectivo de facilitar o *deployment* de interfaces e agentes que utilizam o protocolo *Diameter* como protocolo subjacente. Esta stack fornece mecanismos de *parsing* para transformar dados em/de mensagens *Diameter*. Ainda, e de forma transparente, fornece as diferentes técnicas inerentes dos protocolos AAA, como mecanismos de reencaminhamento, entre outros.

5.2.2.1 **Arquitectura IDS**

A arquitectura IDS pode dividir-se em 4 diferentes módulos:

- *Connection Manager;*
- *Base State Machine;*
- *Message Router;*
- *Application;*

5.2.2.1.1 **Connection Manager**

O *Connection Manager* é responsável por toda a comunicação do protocolo *Diameter*. Este componente utiliza uma aproximação *Non Blocking*, graças à Java NIO [67], para uma arquitectura escalável.

Este módulo é representado por uma *Singleton Class* que recebe o porto do servidor como argumento e lança uma *thread* específica para lidar com as tentativas de novas conexões.

Presente nesta classe existem dois conceitos distintos mas complementares, *Peer* e *Node*. Um *Peer* representa potenciais pontos de conexão (ver em [31] em detalhe o seu conceito) enquanto os *Nodes* representam uma conexão ponto-a-ponto *Diameter*.

Quando a conexão é estabelecida, um *Node* é criado e registado. O registo do *Node* é feito tanto no *Connection Manager* como num *selector*. Este *selector* (característico do Java NIO) é um objecto que representa um objecto constantemente a ser monitorizado para detectar alterações. Quando várias *sockets* se encontram registadas no *selector*, este *selector* observa alterações no estado das *sockets*, por exemplo, mensagens recebidas, *timeouts*, desconexões entre outros eventos. Quando "questionados", estes *selectors* transmitem as alterações que ocorreram desde a última alteração. O selector permite uma arquitectura escalável, independente das conexões apenas uma *thread* é usada, tendo esta, o papel de, em simultâneo, verificar novas mensagens para todas as conexões.

Este módulo realiza ainda o *parser* das mensagens de rede para uma classe genérica que representa a mensagem *Diameter*.

5.2.2.1.2 Base State Machine

O módulo *BaseStateMachine* é um representação do conjunto de regras e estados definidos no [31]. Este módulo contem uma *Queue* interna que recebe eventos, estando estes associados aos *Peers* conectados a esta aplicação.

Este módulo é ainda responsável por verificar o estado da conexão, isto é, verificar se a conexão encontra-se ligada e estimular esta conexão quando esta se encontra no estado *idle* (*keepalives*). Este mecanismo representa o *Device-Watchdog* presente no protocolo *Diameter*.

5.2.2.1.3 Message Router

O módulo *Message Router* representa os diferentes procedimentos de reencaminhamento existentes no protocolo *Diameter*. Este módulo contém a lógica de encaminhamento baseado no tipo de *role*, *realm* e aplicação, fazendo ainda o mapeamento entre os pedidos e respostas. De acordo com o *role*, o sistema *Diameter* aplica lógicas diferentes podendo funcionar como cliente/servidor ou como um agente *Diameter*. Os *realms* representam os domínios suportados pelo sistema [31].

É este módulo que identifica para onde é enviada a mensagem, isto é, baseando-se na informação presente na mensagem *Diameter*, o *Message Router* verifica que aplicação *Diameter* é responsável pela mensagem, tornando este processo transparente às diferentes aplicações.

5.2.2.1.4 Application

Este é o nível de operação mais elevado da *stack*. Todas as aplicações que utilizam a *stack* terão de estender uma das classes que herdam a classe *Application*. Estas classes definem quatro diferentes modos de operação, estando divididos em aplicações cliente/servidor para Autenticação/Autorização e *Accounting*. Na versão utilizada para a criação do demonstrador, apenas as aplicações de Autenticação/Autorização foram consideradas ainda não estando desenvolvidas as aplicações de *Accounting* (importantes, por exemplo, para a implementação da interface Rf).

A primeira classe a herdar a classe *Application* irá dar início a toda a configuração da stack, criando as diferentes *threads* para cada um dos módulos. Essa configuração é feita lendo um ficheiro propriedades, que indica o modo como a informação de configuração é obtida. Temos dois modos de operação, podendo a informação ser lida de um ficheiro XML ou através da utilização de uma *framework* proprietária da PTIN *eXtensible Architecture Framework* (XAF).

Para sessões *stateful* este módulo contém uma subclasse que contém métodos para manipular as sessões activas (por exemplo, remover sessões, verificar sessões que expiraram, etc.).

5.3 **Resource Adapter (RA) Diameter**

O RA *Diameter* foi um módulo/conector desenvolvido para a plataforma *VoIP Open Source Mobicents*. Tem como função interligar o *Mobicents* com um servidor de *Charging* através do protocolo *Diameter*, de acordo com a especificação IMS para a interface Ro. Deste modo, as suas principais funcionalidades são:

- Criação de um *Resource Adaptor* segundo a especificação JSLEE 1.0;
- Estabelecer uma ligação com um servidor de *Charging* utilizando o protocolo *Diameter*;
- Receber e tratar todas as mensagens enviadas pelo servidor de *Charging*;
- Transmitir as mensagens apropriadas ao SLEE sob a forma de eventos;
- Permitir aos serviços existentes no AS, isto é, aos *Service Building Blocks* (SBB), a capacidade de criar e gerir sessões de *Charging*, com diversas opções e modos de taxação.

5.3.1 *Objectivos*

Ao desenvolver o sistema de modo a interligar a plataforma *Mobicents* a um servidor de *Charging*, pretendia-se aumentar o leque de serviços disponíveis no SIP AS, dotando as aplicações presentes neste de capacidades de tarifação.

O principal objectivo do processo de desenvolvimento foi, então, o seguinte: Interligação do SIP AS a um servidor de *Charging* de acordo com a especificação IMS para a interface Ro.

Apesar da criação do *Enabler* de *Charging* (ver mais à frente), este RA pode ser utilizado por qualquer serviço que corra no SIP AS. O que o *Enabler* permitiu foi criar uma camada de abstracção no topo do RA, não sendo necessário aos serviços (por exemplo Vídeo Portal) perceberem de *Diameter* para taxarem o utilizador.

Ao criar o RA pretendeu-se criar um conector *Diameter* que fosse genérico e aplicável a qualquer interface *Diameter* do IMS. Apesar de nesta dissertação só fazer referência à interface Ro, a interface entre o SIP AS e o HSS também já se encontra implementada (interface Sh).

5.3.2 *Perspectiva Funcional*

5.3.2.1 Use-Case "Débito directo"

No caso de pedidos de taxação do tipo "Débito directo" (Figura 45), o serviço pretende debitar na conta de um dado utilizador uma quantidade fixa de unidades de crédito. Um exemplo de utilização será em serviços taxados ao evento, nos quais o cliente paga por efectuar determinadas acções específicas ou aceder a determinados serviços.

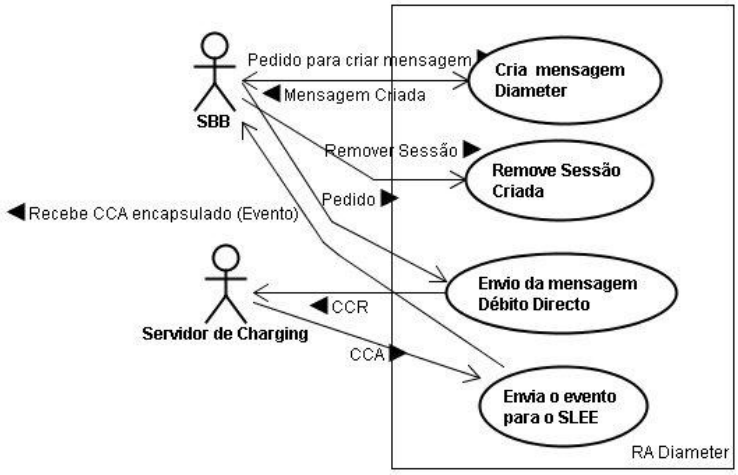

Figura 45 – *Use Case* "Débito Directo"

Inicialmente, o serviço invoca o método de criação de mensagens *Diameter* da interface oferecida pelo *Resource Adaptor*, inserindo como atributo a informação que pretende enviar na mensagem (débito directo, existindo três alternativas para efectuar a verificação de saldo em termos de uma quantia monetária, de tempo ou de informação em bytes). Como resposta, o RA fornece a mensagem já estruturada e que será enviada para o servidor de *Charging* (através de outro método do RA). Após a recepção da resposta do servidor de *Charging*, o RA encapsula a mensagem num evento, disparando-o para o SLEE, chegando a resposta ao objecto que iniciou o pedido. Para terminar, o serviço terá de remover a sessão criada anteriormente de modo a regressar ao estado inicial.

5.3.2.2 Use-Case "Verificação de Saldo"

No caso de pedidos do tipo "Verificação de Saldo" (Figura 46), o serviço pretende verificar a existência de saldo suficiente na conta de um dado utilizador para realizar uma dada operação. Um exemplo de utilização seria, por exemplo, antes de uma operação de débito. Um outro cenário seria um serviço puder verificar que funcionalidades apresentar a um utilizador, dependendo do seu saldo.

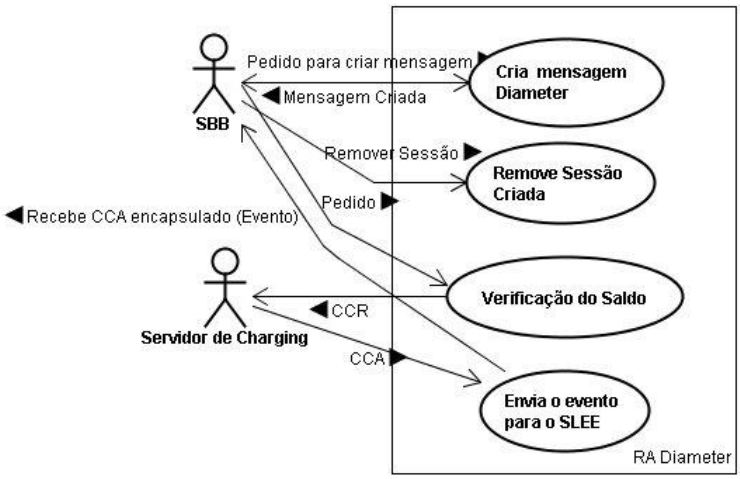

Figura 46 – *Use Case* "Verificação de Saldo"

Inicialmente, o serviço invoca o método de criação de mensagens *Diameter* da interface oferecida pelo *Resource Adaptor*, inserindo como atributo a informação que pretende enviar na mensagem (verificação de saldo, existindo três alternativas para efectuar a verificação de saldo em termos de uma quantia monetária, de tempo ou de informação em bytes). Como resposta, o RA fornece a mensagem já estruturada e que será enviada para o servidor de *Charging* (através de outro método do RA). Após a recepção da resposta do servidor de *Charging*, o RA encapsula a mensagem num evento, disparando-o para o SLEE, chegando a resposta ao serviço. Esta resposta indica ao serviço se o utilizador tem ou não saldo suficiente para aceder ao conteúdo pretendido. Para terminar, o serviço terá de remover a sessão criada anteriormente de modo a regressar ao estado inicial.

5.3.2.3 Use-Case "Custo de um Serviço"

No caso de pedidos do tipo "Custo de um Serviço" (Figura 47), o serviço no AS pretende saber qual o custo de um dado serviço a oferecer a um cliente. Um exemplo de utilização seria uma aplicação pedir ao servidor de *Charging* o custo de um ou mais serviços para de seguida puder verificar se o utilizador tem ou não saldo suficiente, através de um pedido do tipo "Verificação de Saldo".

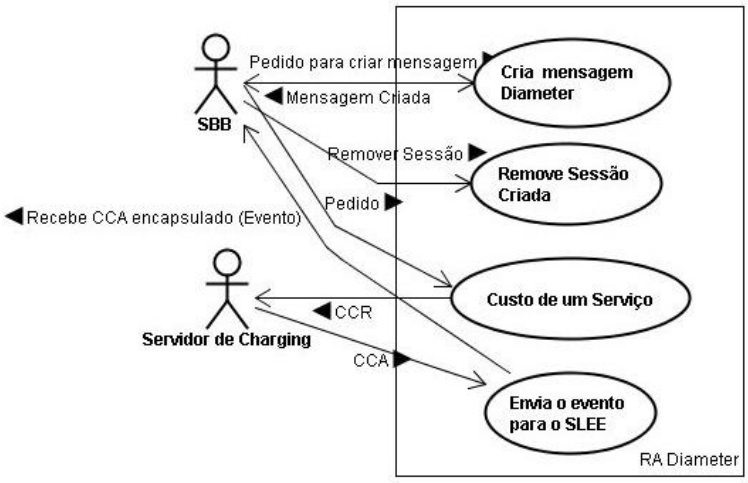

Figura 47 – *Use Case* "Custo de um Serviço"

Inicialmente, o serviço invoca o método de criação de mensagens *Diameter* da interface oferecida pelo *Resource Adaptor*, inserindo como atributo a informação que pretende enviar na mensagem (passando o nome do serviço cujo custo deseja conhecer). Como resposta, o RA fornece a mensagem já estruturada e que será enviada para o servidor de *Charging* (através de outro método do RA). Após a recepção da resposta do servidor de *Charging*, o RA encapsula a mensagem num evento, disparando-o para o SLEE, chegando a resposta ao serviço. Esta resposta indica então o custo do serviço pretendido. Para terminar, o serviço terá de remover a sessão criada anteriormente de modo a regressar ao estado inicial.

5.3.2.4 Use-Case "Crédito Directo"

No caso de pedidos do tipo "Crédito directo" (Figura 48), o serviço pretende creditar na conta de um dado utilizador uma quantidade fixa de unidades de crédito. Um exemplo de utilização será um serviço que permita ao utilizador carregar a sua conta, ou que por um outro motivo ofereça crédito a este.

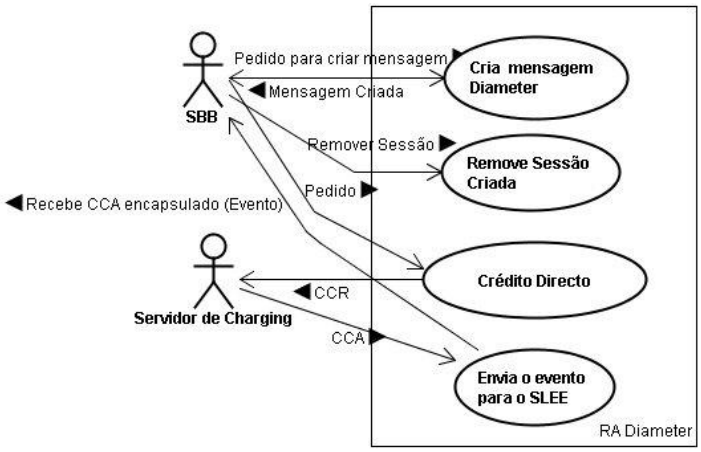

Figura 48 – *Use Case* "Crédito Directo"

Inicialmente, o serviço invoca o método de criação de mensagens *Diameter* da interface oferecida pelo *Resource Adaptor*, inserindo como atributo a informação que pretende enviar na mensagem (crédito directo, existindo três alternativas para efectuar a verificação de saldo em termos de uma quantia monetária, de tempo ou de informação em bytes). Como resposta, o RA fornece a mensagem já estruturada e que será enviada para o servidor de *Charging* (através de outro método do RA). Após a recepção da resposta do servidor de *Charging*, o RA encapsula a mensagem num evento, disparando-o para o SLEE, chegando a resposta ao serviço. Para terminar, o serviço terá de remover a sessão criada anteriormente de modo a regressar ao estado inicial.

5.3.2.5 Use-Case "Reserva de Unidades"

No caso de pedidos de taxação "Reserva de Unidades" (Figura 49), uma forma de taxação *online* do tipo SCUR, o serviço no AS pretende reservar uma determinada quantia de unidades de crédito para um dado serviço. Um exemplo de utilização será um cenário em que uma aplicação que baseie o seu modelo de negócios na duração ou consumo de utilização do serviço e deseja assegurar à partida que uma aplicação concorrente não utilizará o saldo do cliente necessário ao serviço.

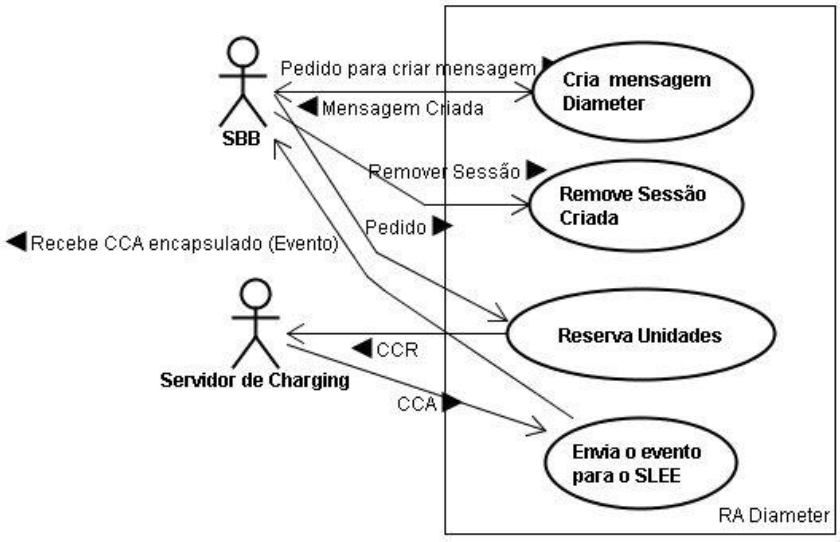

Figura 49 – *Use Case* "Reserva de Unidades"

Inicialmente, o serviço invoca o método de criação de mensagens *Diameter* da interface oferecida pelo *Resource Adaptor*, inserindo como atributo a informação que pretende enviar na mensagem (reserva unidades, existindo três alternativas para efectuar a verificação de saldo em termos de uma quantia monetária, de tempo ou de informação em bytes). Como resposta, o RA fornece a mensagem já estruturada e que será enviada para o servidor de *Charging* (através de outro método do RA). Após a recepção da resposta do servidor de *Charging*, o RA encapsula a mensagem num evento, disparando-o para o SLEE, chegando a resposta ao serviço. O serviço obtém da resposta a informação sobre a quantia de unidades reservada podendo esta ser a pedida, a máxima permitida pelo servidor ou a possível dado o saldo do utilizador.

Este tipo de taxação requer que a aplicação actualize o servidor de *Charging* com novos pedidos de "Reserva de Unidades" à medida que o serviço é usado, informando o servidor da reserva efectuada, de quantas unidades já foram consumidas e devem, deste modo, ser taxadas e, eventualmente, do fim da operação de taxação, não sendo necessário reservar mais saldo. Neste tipo de transacção é necessário armazenar o identificador da sessão, pois este terá de ser sempre o mesmo desde o início até ao fim da taxação.

Para terminar, o serviço terá de remover a sessão criada anteriormente de modo a regressar ao estado inicial.

5.3.3 *Perspectiva Lógica*

O *Resource Adaptor Diameter* está organizado em cinco *packages*, sendo possível distinguir três deles correspondentes às classes e interfaces do *Resource Adaptor* em si e dois correspondentes ao *Resource Adaptor Type*.

Dentro do RA, podemos encontrar então os *packages*:

- ***pt.ptinovacao.ra.diameter.ids***
- ***pt.ptinovacao.ra.diameter.util***
- ***pt.ptinovacao.ra.diameter.ra***

Dentro do *Resource Adaptor Type* temos os *packages*:

- ***pt.ptinovacao.ra.diameter.ratype***
- ***pt.ptinovacao.ra.diameter.ratype.activities***

5.3.3.1 Package pt.ptinovacao.ra.diameter.ids

O *package* ***pt.ptinovacao.ra.diameter.ids*** inclui apenas uma classe que corresponde ao cliente *Diameter* que será utilizado por este *Resource Adaptor* para enviar e receber as mensagens trocadas com o servidor de *Charging*. Trata-se da classe *DiameterClient* (Figura 50) que estende a classe *ClientAuthSession* da stack IDS.

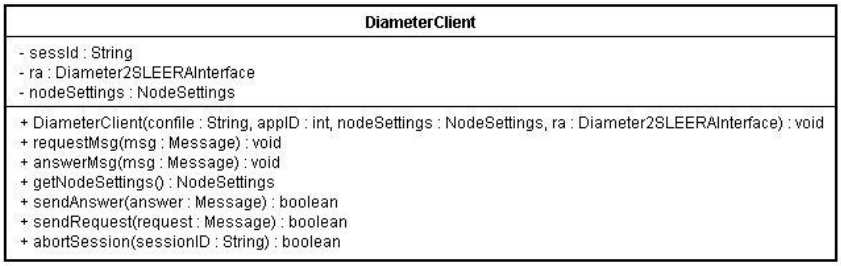

Figura 50 – Classe *pt.ptinovacao.ra.diameter.ids.DiameterClient*

5.3.3.2 Package pt.ptinovacao.ra.diameter.util

Tal como o nome indica, este *package* contem algumas classes e interfaces que permitem configurar e auxiliar o *Resource Adaptor*. Neste *package* é possível começar por observar a interface *Diameter2SLEERAInterface* (Figura 51), implementada pela classe *DiameterResourceAdaptor* (ver mais à frente) e que irá ser fornecida ao *DiameterClient* com dois métodos de forma a passar a informação deste para o RA.

Figura 51 – Interface *pt.ptinovacao.ra.diameter.util.Diameter2SLEERAInterface*

Dentro deste *package* é ainda possível observar a classe *NodeSettings* (Figura 52), que contem a informação de configuração do cliente *Diameter*.

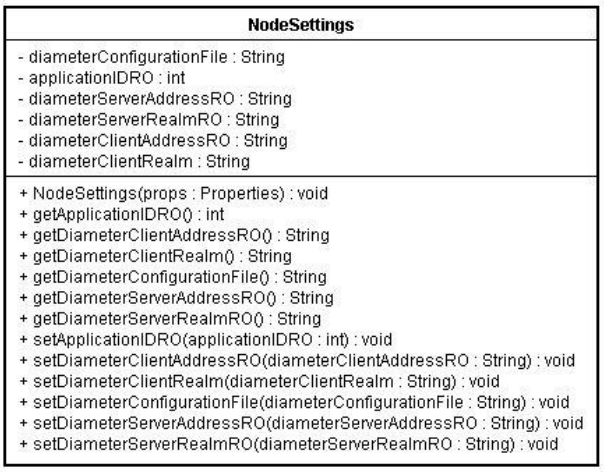

Figura 52 – Interface *pt.ptinovacao.ra.diameter.util.Diameter2SLEERAInterface*

Para terminar, este mesmo *package* contem ainda a classe *EventHandle* (Figura 53). Esta classe serve como chave na *hashtable associationMap*, da classe *DiameterResourceAdaptor*. Deste modo, através do *EventHandle* conseguimos obter o "event-class-name" que se encontra no "event-jar.xml" paro o evento correspondente (exemplo, 272R=pt.ptinovacao.diameter.stack.Credit-Control-Request).

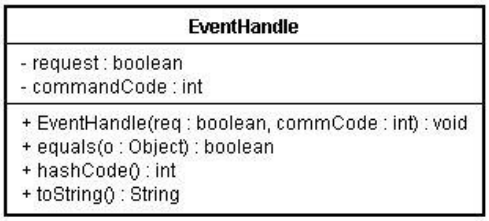

Figura 53 – Classe *pt.ptinovacao.ra.diameter.util.EventHandle*

5.3.3.3 Package pt.ptinovacao.ra.diameter.ra

O *package* ***pt.ptinovacao.ra.diameter.ra*** tem como funcionalidades não só a implementação do core do *Resource Adaptor* como também implementar todas as interfaces presentes dentro do *Resource Adaptor Type*, como por exemplo, a implementação da interface que irá permitir ao serviços acederem ao RA, permitir criar *ActivityContexts* entre os serviços e o RA e ainda a criação de actividades para cada aplicação *Diameter* distinta.

A Figura 54 permite ilustrar a estrutura de uma das classes dentro deste *package*, a *DiameterResourceAdaptor*. Além dos métodos ligados ao *lifecycle* do RA, temos ainda os métodos *onRequest*() e *onAnswer*() que implementam os métodos da interface *Diameter2SLEERAInterface*, sendo chamados pelo classe *DiameterClient*. Dentro desta classe é ainda possível observar o método *fireEventToSLEE* que envia os eventos para o SLEE e os métodos *get/setSession*() que permitem armazenar e aceder à informação que

contém as sessões activas. Relativamente aos atributos, temos o *sessions* onde são guardadas as sessões, o *Diameter client*, o *associationMap* (comentado anteriormente), o *raProvider* que implementa a interface disponível para os serviços entre outros.

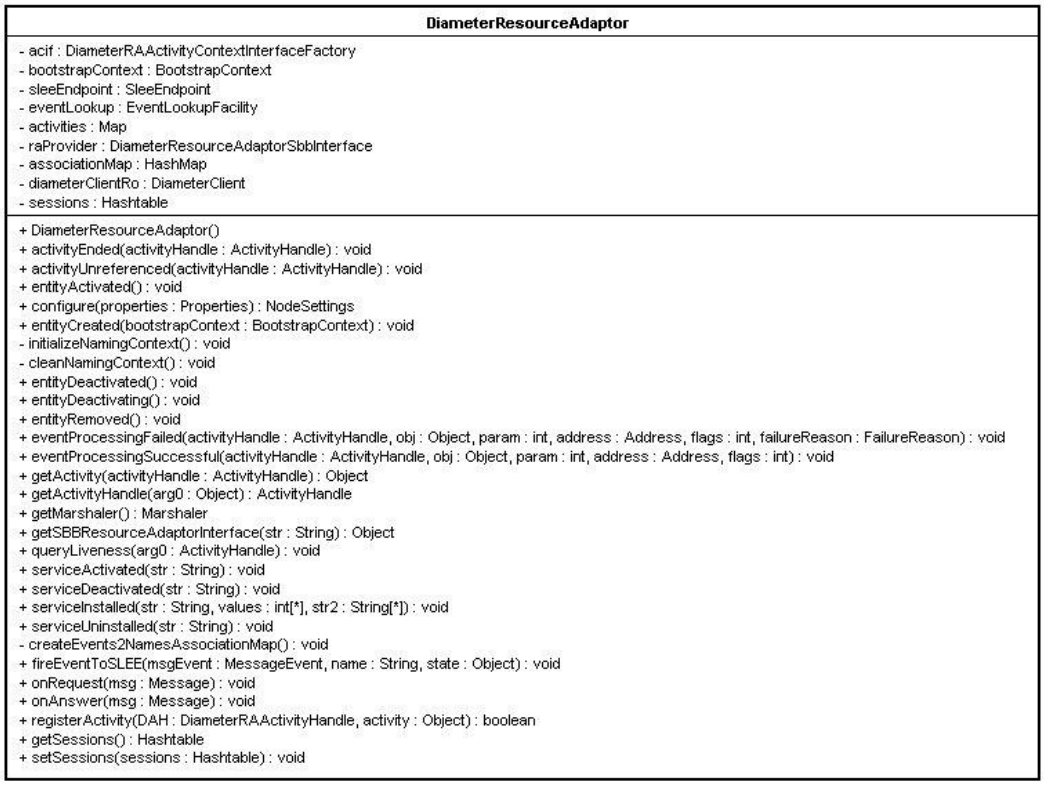

Figura 54 – Classe *pt.ptinovacao.ra.diameter.ra.DiameterResourceAdaptor*

Passando para a segunda classe presente neste módulo, temos a classe *DiameterRAProvider* (Figura 55) que tem como função implementar a interface que irá estar disponível para os serviços, isto é, a *DiameterResourceAdaptorSbbInterface*.

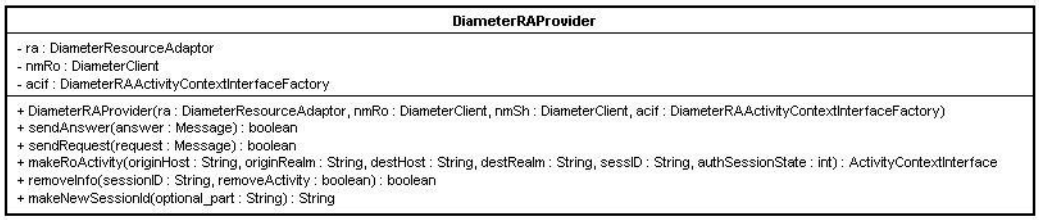

Figura 55 – Classe *pt.ptinovacao.ra.diameter.ra.DiameterRAProvider*

Para interligar os serviços e o RA, terá de ser criado um *ActivityContext* entre os dois elementos. Desta forma, o RA implementa a interface *DiameterRAActivityContextInterfaceFactory*, através da classe *DiameterRAActivityContextInterfaceFactoryImpl* (Figura 56).

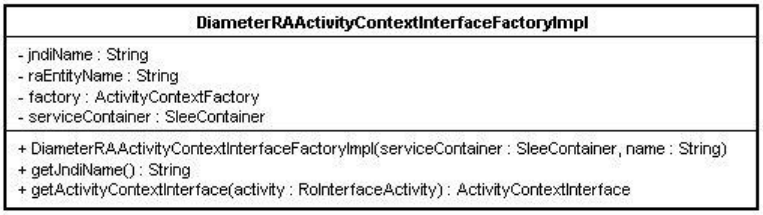

Figura 56 – Classe *pt.ptinovacao.ra.diameter.ra.DiameterRAActivityContextInterfaceFactoryImpl*

Depois de criado o *ActivityContext*, é possível criar as *Activities* que serão utilizadas para a comunicação e troca de mensagens entre os elementos. A classe *InterfaceActivityImpl* (Figura 57) permite criar uma *activity*, dentro do *DiameterRAProvider*. Deste modo, o SBB ao criar a actividade vai puder aceder aos diferentes métodos para criar as mensagens *Diameter*.

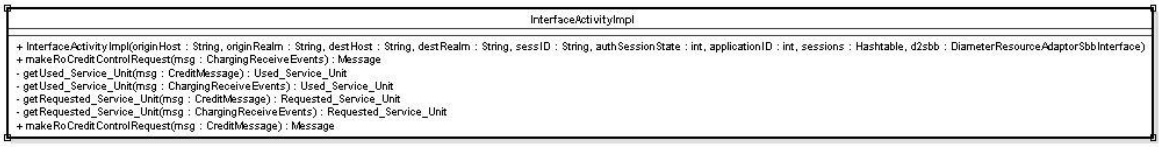

Figura 57 – Classe *pt.ptinovacao.ra.diameter.ra.InterfaceActivityImpl*

A classe da figura anterior implementa a interface *RoInterfaceActivity*, estendendo ainda a classe *DiameterActivityCommonPart* (Figura 58). Caso este RA tivesse ligado a diferentes tipos de actividade, por exemplo para a interface Sh, teríamos de implementar uma interface diferente da anterior referida. A classe *DiameterActivityCommonPart* permite separar da classe *InterfaceActivityImpl* a informação que é comum a todas as actividades, por exemplo a informação intrínseca ao protocolo *Diameter*.

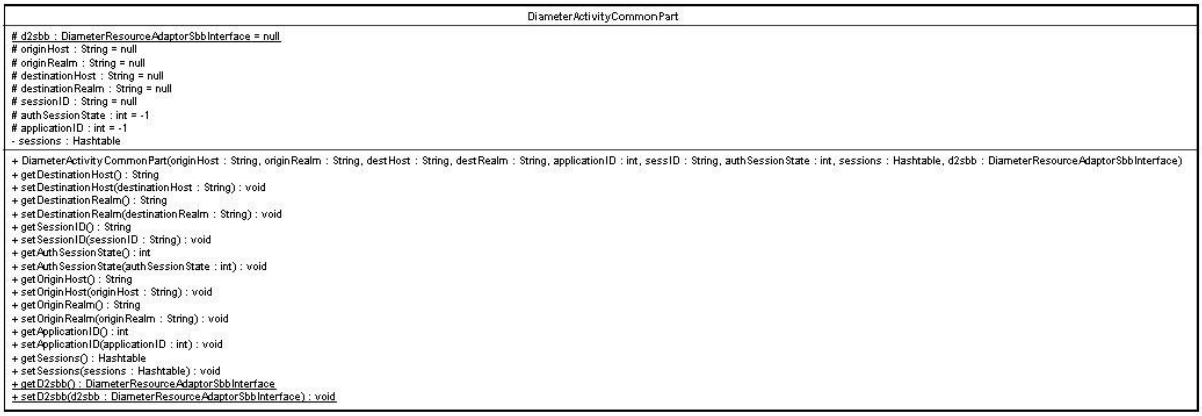

Figura 58 – Classe *pt.ptinovacao.ra.diameter.ra.DiameterActivityCommonPart*

Passando para a *DiameterRAActivityHandle* (Figura 59), é possível observar a classe que funciona como chave para identificar uma actividade. Tem como atributo a String *handle*, que armazena o *session-Id* da mensagem *Diameter*.

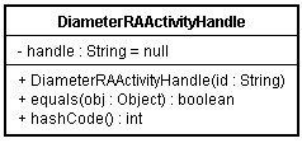

Figura 59 – Classe *pt.ptinovacao.ra.diameter.ra.DiameterRAActivityHandle*

Para concluir este *package*, temos para terminar a classe *MessageEventImpl* (Figura 60), que tem como objectivo implementar a interface *MessageEvent*, que representa o evento que é passado para os SBBs. Este evento contém a mensagem *Diameter* com o seu conteúdo.

Figura 60 – Classe *pt.ptinovacao.ra.diameter.ra.MessageEventImpl*

5.3.3.4 Package pt.ptinovacao.ra.diameter.ratype

O *package* ***pt.ptinovacao.ra.diameter.ratype*** contem quatro interfaces, sendo estas implementadas pelo *package pt.ptinovacao.ra.diameter.ra*.

Em primeiro, temos a interface *ActivitiesFactory* (Figura 61) que contem os métodos que permitem criar as actividades, neste caso as actividades para a interface Ro.

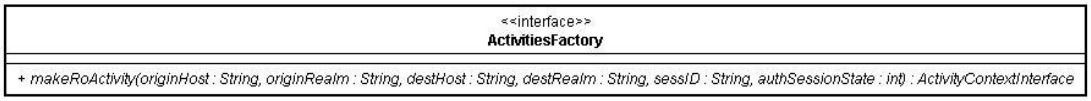

Figura 61 – Interface *pt.ptinovacao.ra.diameter.ratype.ActivitiesFactory*

Outra interface presente neste *package* é a *DiameterRAActivityContextInterfaceFactory* (Figura 62), que estará disponível para o *DiameterRAProvider* para este criar um *ActivityContextInterface* de acordo com a actividade correspondente. A partir do *ActivityContextInterface*, o SBB pode criar as diferentes actividades que quiser (ver conceito de Actividades e *Activity Context* na secção 3.5.3.3).

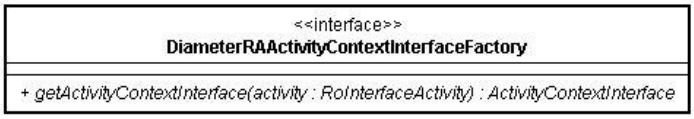

Figura 62 – Interface *pt.ptinovacao.ra.diameter.ratype.DiameterRAActivityContextInterfaceFactory*

Passando para a interface *DiameterResourceAdaptorSbbInterface* (Figura 63), estamos na presença da interface que irá estar disponível ao SBB para criar e enviar mensagens *Diameter*. Como foi dito, a interface *ActivitiesFactory* permite criar as actividades que irão enviar essas mensagens.

Figura 63 – Interface *pt.ptinovacao.ra.diameter.ratype.DiameterResourceAdaptorSbbInterface*

Podemos observar na figura anterior os métodos *sendRequest* e *sendAnswer* que irão estar disponíveis aos Sbbs para estes poderem enviarem mensagens *Diameter* iniciais e resposta, respectivamente. O método *removeInfo* permite remover uma determinada sessão e a respectiva actividade. O método *makeNewSessionId* permite criar uma nova sessão.

Para terminar, temos ainda dentro deste *package* a interface *MessageEvent* (Figura 64) que define o evento trocado entre o RA *Diameter* e os SBBs.

Figura 64 – Interface *pt.ptinovacao.ra.diameter.ratype.MessageEvent*

5.3.3.5 Package pt.ptinovacao.ra.diameter.ratype.activities

Tal como o nome indica, o *package* ***pt.ptinovacao.ra.diameter.ratype.activities*** representa as actividades possíveis de serem criadas por este RA. Neste momento temos apenas a possibilidade de criar uma actividade para a interface Ro. Para tal, temos a interface *RoInterfaceActivity* (Figura 65) que estende a interface *ActivityBaseInterface* (Figura 66) que contem a informação do protocolo base *Diameter*. Esta interface contém o método *makeRoCreditControlRequest* que permite criar uma mensagem *Diameter* de CC.

Figura 65 – Interface *pt.ptinovacao.ra.diameter.ratype.activities.RoInterfaceActivity*

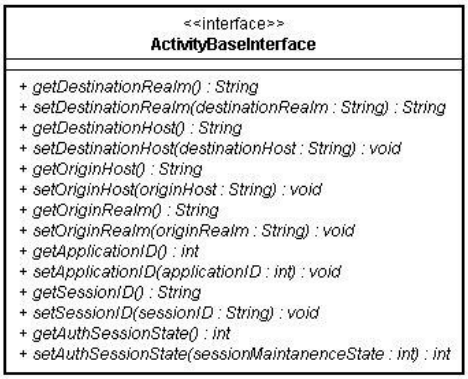

Figura 66 – Interface *pt.ptinovacao.ra.diameter.ratype.activities.ActivityBaseInterface*

5.3.4 *Perspectiva Física*

As Figura 67 e Figura 68 ilustram os diagramas de componentes e distribuição do sistema.

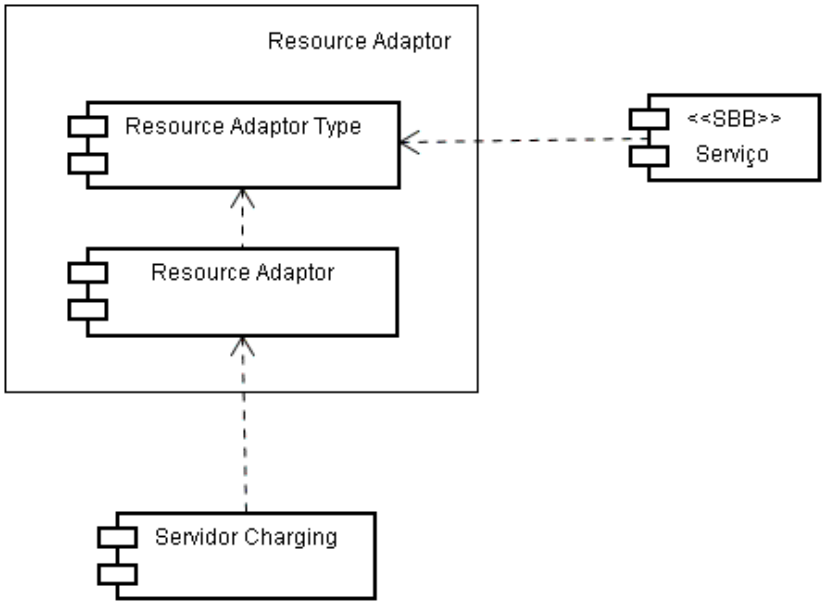

Figura 67 – Diagrama de componentes (internos e externos) do sistema

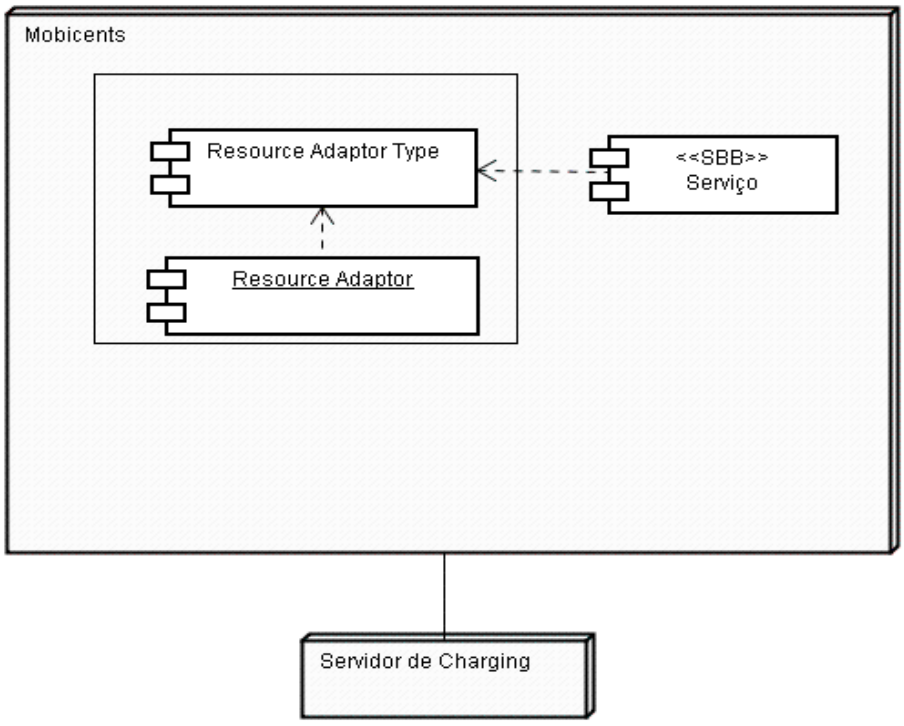

Figura 68 – Diagrama de distribuição do sistema

5.3.5 *Detalhes de Concepção*

5.3.5.1 Troca de mensagens com o servidor de Charging

O AS tem um papel de cliente perante o servidor de *Charging*. Deste modo, a troca de mensagens com este baseia-se em pedidos enviados (CCR) através dos métodos oferecidos pelo *Resource Adaptor* aos SBBs e nas respostas (CCA) do servidor, que depois são disparadas para o SLEE sob a forma de eventos (*MessageEvent*).

As mensagens CCR e CCA baseiam-se num conjunto de AVPs que deverão ser preenchidos quer pelo cliente quer pelo servidor de *Charging*, dependendo do tipo de mensagem e serviço pedido. Deste modo, o RA terá de preparar os pedidos para enviar ao servidor preenchendo uma série de AVPs no pedido a efectuar; no caso de recepção das respostas enviadas pelo servidor, o RA verifica os AVPs presentes na mensagem, reencaminhando estas mensagens para o SLEE.

As tabelas seguintes apresentam os diferentes AVPs e valores necessários para cada tipo de mensagem. Os AVPs *Mandatory* são obrigatórios em todas as mensagens segundo os [31] e [42]. Os AVPs *Required* são obrigatórios em alguns tipos de mensagens. Alguns AVPs devem apresentar um valor específico dependente do tipo de mensagem. Para uma descrição detalhada de cada AVP devem ser consultados os documentos [42] e [31].

Tabela 8 – AVPs necessários nas mensagens de débito automático de unidades

AVP	CCR	CCA
Session-Id	Mandatory	Mandatory
Result-Code	-	Mandatory
Origin-Host	Mandatory	Mandatory
Origin-Realm	Mandatory	Mandatory
Destination-Realm	Mandatory	-
Auth-Application-Id	Mandatory	Mandatory
Service-Context-Id	Mandatory	-
Service-Identifier	Mandatory	
Subscription-Id	Mandatory	
Event-Timestamp	Required	Required
CC-Request-Type	Mandatory Value set to EVENT_REQUEST	Mandatory Value set to EVENT_REQUEST
CC-Request-Number	Mandatory	Mandatory
Requested-Action	Required Value set to DIRECT-DEBITING	-
Subscription-Id	Required	-

Requested-Service-Unit	Required	-
Cost-Information	-	Required
Granted-Service-Unit	-	Required

Tabela 9 – AVPs necessários nas mensagens de verificação de existência de saldo

AVP	CCR	CCA
Session-Id	Mandatory	Mandatory
Result-Code	-	Mandatory
Origin-Host	Mandatory	Mandatory
Origin-Realm	Mandatory	Mandatory
Destination-Realm	Mandatory	-
Auth-Application-Id	Mandatory	Mandatory
Service-Context-Id	Mandatory	-
Subscription-Id	Mandatory	
Event-Timestamp	Required	Required
CC-Request-Type	Mandatory Value set to EVENT_REQUEST	Mandatory Value set to EVENT_REQUEST
CC-Request-Number	Mandatory	Mandatory
Requested-Action	Required Value set to CHECK-BALANCE	-
Subscription-Id	Required	-
Requested-Service-Unit	Required	-
Check-Balance-Result	-	Required Value set to ENOUGH_CREDIT or NO_CREDIT

Tabela 10 – AVPs necessários nas mensagens de crédito de novas unidades

AVP	CCR	CCA
Session-Id	Mandatory	Mandatory
Result-Code	-	Mandatory
Origin-Host	Mandatory	Mandatory
Origin-Realm	Mandatory	Mandatory
Destination-Realm	Mandatory	-
Auth-Application-Id	Mandatory	Mandatory
Service-Context-Id	Mandatory	-
Service-Identifier	Mandatory	
Subscription-Id	Mandatory	
Event-Timestamp	Required	Required
CC-Request-Type	Mandatory Value set to EVENT_REQUEST	Mandatory Value set to EVENT_REQUEST
CC-Request-Number	Mandatory	Mandatory
Requested-Action	Required	-

	Value set to REFUND-ACCOUNT	
Subscription-Id	Required	-
Requested-Service-Unit	Required	-
Cost-Information	-	Required

Tabela 11 – AVPs necessários nas mensagens de verificação do custo de um determinado serviço

AVP	CCR	CCA
Session-Id	Mandatory	Mandatory
Result-Code	-	Mandatory
Origin-Host	Mandatory	Mandatory
Origin-Realm	Mandatory	Mandatory
Destination-Realm	Mandatory	-
Auth-Application-Id	Mandatory	Mandatory
Service-Context-Id	Mandatory	-
Service-Identifier	Mandatory	
Subscription-Id	Mandatory	
Event-Timestamp	Required	Required
CC-Request-Type	Mandatory Value set to EVENT_REQUEST	Mandatory Value set to EVENT_REQUEST
CC-Request-Number	Mandatory	Mandatory
Requested-Action	Required Value set to PRICE-ENQUIRY	-
Subscription-Id	Required	-
Cost-Information	-	Required

Tabela 12 – AVPs necessários nas mensagens de reserva e débito de unidades

AVP	CCR	CCA
Session-Id	Mandatory	Mandatory
Result-Code	-	Mandatory
Origin-Host	Mandatory	Mandatory
Origin-Realm	Mandatory	Mandatory
Destination-Realm	Mandatory	-
Auth-Application-Id	Mandatory	Mandatory
Service-Context-Id	Mandatory	-
Service-Identifier	Mandatory	
Event-Timestamp	Required	Required
CC-Request-Type	Mandatory Value set to INITIAL_REQUEST, UPDATE_REQUEST or TERMINATE_REQUEST	Mandatory Value set to INITIAL_REQUEST, UPDATE_REQUEST or TERMINATE_REQUEST
CC-Request-Number	Mandatory	Mandatory
Subscription-Id	Mandatory	-

Requested-Service-Unit	Required	-
Used-Service-Unit	Required	-
Cost-Information	-	Required
Granted-Service-Unit	-	Required
Final-Unit-Indication	-	Required
Validity-Time	-	Required

5.4 **Enabler Charging**

Esta secção descreve o processo de desenvolvimento do sistema *Enabler* de *Charging*, abordando temas como a arquitectura, as opções de desenho, a interacção entre os diversos módulos constituintes e as dependências existentes para sistemas externos.

O *Enabler* de *Charging* é um módulo/serviço utilizado sobre a plataforma SIP AS. Este sistema comunica com os serviços existentes, fornecendo um processo de taxação adequado à lógica do serviço. Sendo assim, as suas principais funções são:

- Receber eventos de outros serviços relativos a acontecimentos passivos de serem taxados;
- Receber informação do utilizador desses serviços, do tipo de acção tomada e do tipo de serviço em questão de forma a aplicar a taxação correcta a cada caso;
- Comunicar com um servidor de *Charging* (apesar de ter sido utilizado um RA *Diameter* para comunicação com o servidor de *Charging*, era possível ter vários mecanismos, protocolos tendo o *Enabler* de *Charging* lógica adicional para interpretar o melhor mecanismo a utilizar);
- Informar o serviço acerca do resultado do processo de tarifação (o utilizador tem saldo suficiente para o serviço, o débito foi efectuado com sucesso, etc.).

5.4.1 *Objectivos*

Os principais objectivos do processo de desenvolvimento do sistema foram os seguintes:

- Criação de um serviço segundo a especificação JSLEE 1.0;
- Integração deste serviço com o Vídeo Portal e o serviço PC2Phone;
- Integração deste serviço com o servidor de *Charging* através do *Resource Adaptor Diameter* anteriormente apresentado;

5.4.1.1 Fundamentação e Abordagem, Restrições e Condicionantes

A abordagem tomada foi a de implementar um conjunto de eventos único que o *enabler* iria receber. Qualquer serviço que necessitasse ser taxado teria de enviar estes eventos consoante a acção desejada.

Por sua vez, o *enabler* usaria o RA *Diameter* para comunicar com o servidor de *Charging*. Esta invocação seria efectuada através da interface disponibilizada pelo RA (ver sub capítulo do RA, interface *DiameterResourceAdaptorSbbInterface*). A resposta do RA seria enviada para o *enabler* através de eventos. Da mesma forma, o *Enabler* de *Charging* também enviaria eventos para o serviço que enviou os pedidos de taxação.

5.4.2 *Perspectiva Funcional*

Para a integração deste sistema com o serviço de Vídeo Portal temos então duas actividades principais: o acesso de um utilizador ao serviço de Vídeo Portal e a escolha de uma opção durante o uso do serviço.

Para a integração deste sistema com o serviço PC2Phone temos três possíveis actividades, correspondentes ao pedido do utilizador IMS para o utilizador da PSTN, a reserva periódica de mais unidades (após as anteriores terem sido consumidas) e por último, claro, o desligar da chamada.

5.4.2.1 Use-case "Acesso ao serviço de Vídeo Portal"

Quando um utilizador acede ao serviço de Vídeo Portal é necessário verificar o saldo do mesmo, afim de confirmar a existência de saldo suficiente que permita a escolha de pelo menos uma das opções do serviço (Figura 69).

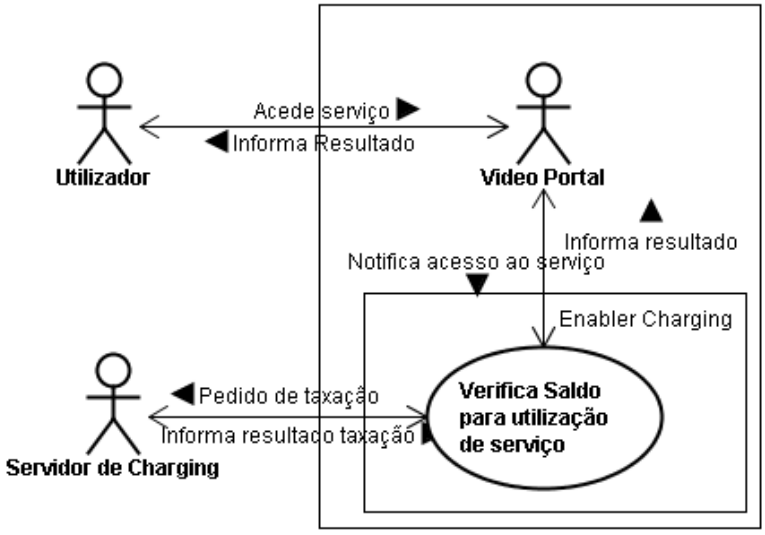

Figura 69 – *Use-Case* durante o acesso ao serviço de Vídeo Portal

5.4.2.2 Use-case "Escolha de Opção no serviço de Vídeo Portal"

Durante o uso do serviço de Vídeo Portal, o utilizador tem acesso a um menu inicial com possíveis escolhas de vídeos para visualização. Durante este período o utilizador pode efectuar escolhas relativas aos filmes existentes e este processo tem de ser validado pelo sistema de taxação. Mediante a informação proveniente do sistema, o serviço de Vídeo Portal fornece o vídeo pedido ou uma mensagem indicativa da impossibilidade de visualização do vídeo devido à inexistência de saldo (Figura 70).

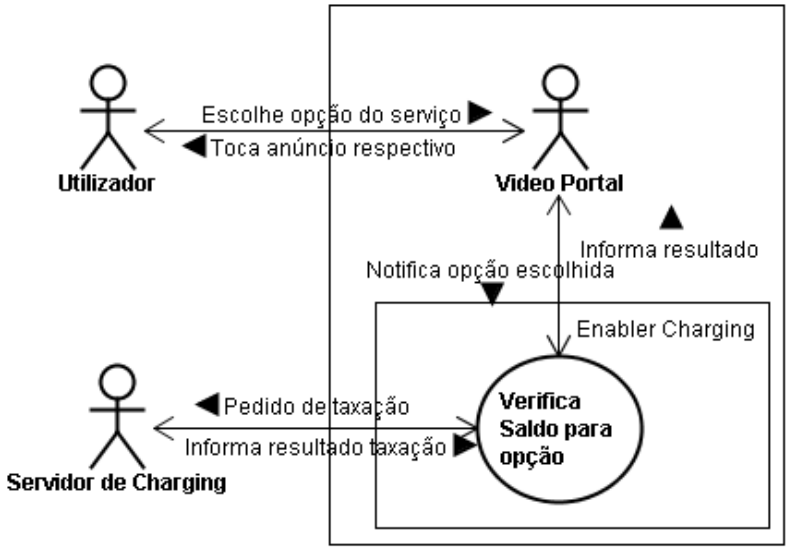

Figura 70 – *Use-Case* escolha de opção durante o uso do serviço de Vídeo Portal.

5.4.2.3 Use-case "Início da taxação da chamada"

A primeira actividade representa a inicialização da sessão por parte do utilizador, correspondente ao processo inicial de reserva de unidades (Figura 71). Caso o utilizador tenha saldo suficiente a chamada é reencaminhada para o utilizador da PSTN, dando-se início ao processo de taxação baseado em sessão. Pelo contrário, caso o saldo do utilizador seja inferior ao valor mínimo inicial necessário para a reserva de unidades (que neste caso correspondia a um minuto), a chamada seria cancelada.

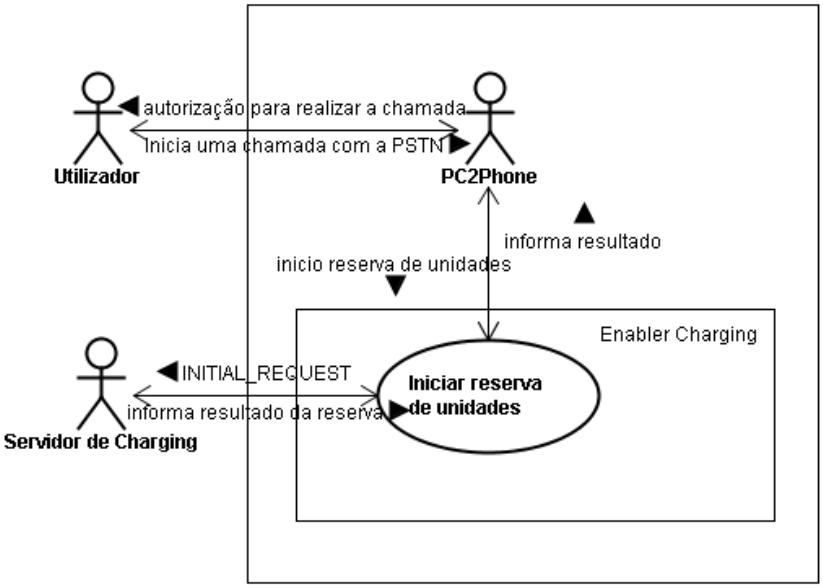

Figura 71 – Início da reserva de recursos

5.4.2.4 Use-case "Reserva periódica de unidades"

A segunda actividade do serviço PC2Phone corresponde ao processo de reserva de mais unidades e indicar ao servidor de *Charging* que as anteriores foram totalmente consumidas, isto é, após ter-se gasto o tempo anteriormente reservado temos de novamente reservar mais tempo. Caso o saldo do utilizador esteja a esgotar-se, o servidor de *Charging* envia uma notificação na resposta ao pedido do *Enabler* de *Charging* indicando que as unidades fornecidas são as últimas do utilizador, sendo o serviço responsável pelo cancelamento da chamada caso estas se esgotem.

Após a recepção da resposta do servidor de *Charging*, o *Enabler* envia uma mensagem a titulo informativo para o serviço PC2Phone, permitindo a este ter a percepção que, por exemplo, o saldo do utilizador está a terminar (Figura 72).

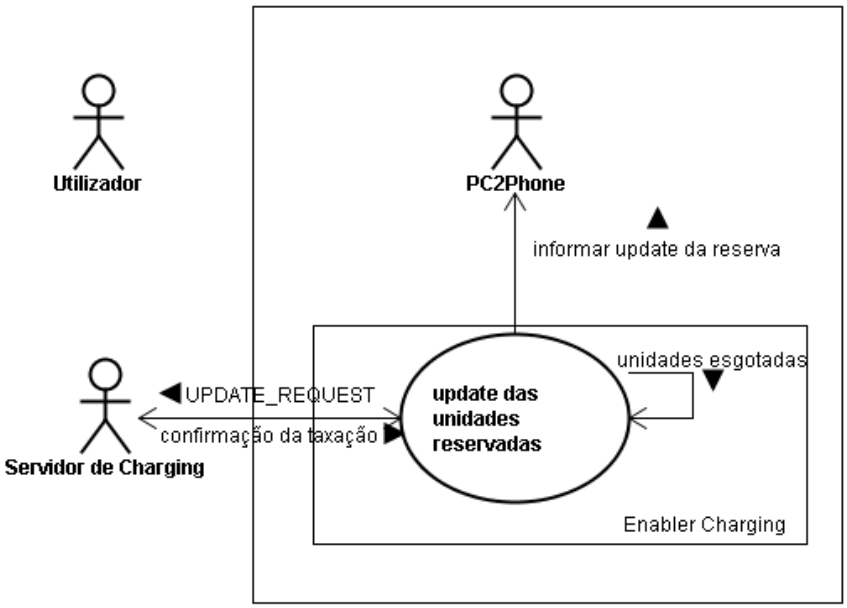

Figura 72 – *Update* das unidades reservadas

5.4.2.5 Use-case "Fim da chamada"

Esta última actividade representa o desligar da chamada por parte do utilizador. Durante este procedimento, a mensagem de sinalização (SIP BYE) chega ao serviço PC2Phone, que envia uma notificação para o *Enabler* de *Charging*, alertando-o para terminar o processo de taxação para aquela sessão. Para tal, o *Enabler* envia um pedido de taxação para o Servidor de *Charging* com a indicação das unidades reservadas não consumidas, permitindo ao servidor de *Charging* creditar estas unidades na conta do utilizador (Figura 73).

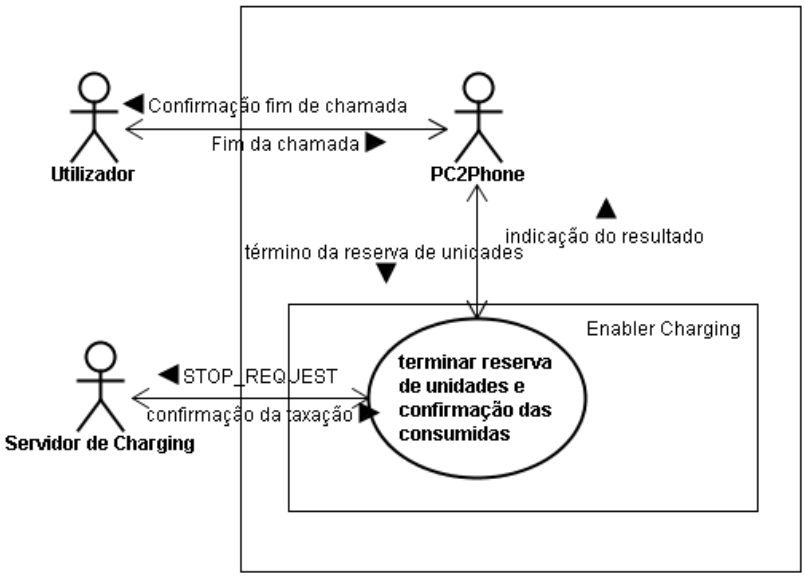

Figura 73 – Término da chamada

5.4.3 *Perspectiva Lógica*

O *Enabler* de *Charging* está organizado em dois *packages*, um que representa o próprio *Enabler* e outro correspondente aos eventos utilizados para comunicar com serviços que fazem uso deste.

Dentro do *Enabler*, podemos então encontrar o *package*:

- ***package pt.ptinovacao.sbb.sbb***

Dentro dos Eventos, temos o *package*:

- ***pt.ptinovacao.control.charging.events***

5.4.3.1 Package pt.ptinovacao.sbb.sbb

O *package* ***pt.ptinovacao.sbb.sbb*** contem duas classes com toda a lógica do *enabler*. A primeira classe, *BaseSbb* (Figura 74) contem os métodos que representam o *lifecycle* do SBB e ainda os métodos que permitem aceder às *facilities* do JSLEE.

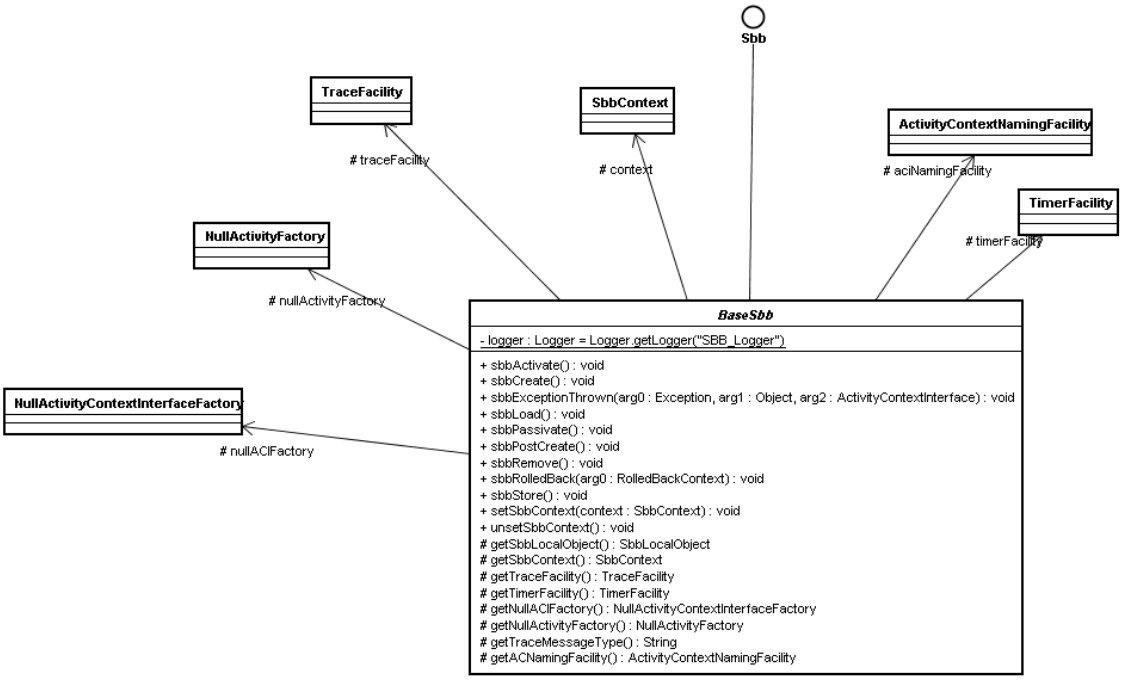

Figura 74 – Classe *pt.ptinovacao.sbb.sbb.BaseSbb*

A classe *BaseSbb* é herdada pela segunda classe presente neste *package*, a classe *ChargingSbb* (Figura 75).

Figura 75 – Classe *pt.ptinovacao.sbb.sbb.ChargingSbb*

Em primeiro lugar, a classe *ChargingSbb* contém a lógica e métodos que permitem tratar os eventos recebidos pelos serviços. Esses métodos, e tal como todos os representam eventos que possam ser tratados pelo SSB, terão de ter uma assinatura "on"+nome do evento. Pela figura anterior é possível constatar que o SBB suporta os eventos:

- *GiveCreditEvent*
- *ChargeServiceEvent*
- *GetServiceCostEvent*
- *CheckCreditEvent*
- *StartChargingEvent*
- *UpdateChargingEvent*
- *StopChargingEvent*
- *TimerEvent*
- *CreditControlAnswer*

Além deste requisito, para um SBB receber eventos terá de colocar no ficheiro "sbb-jar.xml" a lista de eventos suportados. Na lista anterior é possível distinguir três grupos de eventos. O primeiro grupo, constituído pelos primeiros sete eventos da lista anterior, correspondem aos eventos disponíveis aos serviços que queiram aceder às funcionalidades do *enabler*. O método *onTimerEvent* permite receber as notificações de *timers* lançados pela classe *ChargingSbb*. Por último, o *onCreditControl* permite receber os eventos lançados para o SLEE pelo RA *Diameter* (quando chega uma resposta do servidor de *Charging*).

Por outro lado, os métodos com a assinatura "fire"+nome do evento representam os eventos lançados pelo *Enabler* de *Charging* para os serviços. Estes métodos serão utilizados para enviar as respostas aos pedidos recebidos:

- *ChargingCreditOkEvent*
- *ChargingNoCreditEvent*
- *ChargingCheckCreditOkEvent*
- *ChargingCheckCreditOkEvent*
- *ChargingCheckNoCreditEvent*
- *ChargingEndCreditEvent*
- *ChargingGiveCreditACKEvent*
- *ChargingServiceCostEvent*
- *ChargingErrorEvent*

- *ChargingStartACKEvent*
- *ChargingStopACKEvent*
- *ChargingUpdateEvent*

Os *setters* e *getters* representam *Container Managed Persistence* (CMPs) que guardam informação, por instância de *Sbb*.

Entre os restantes métodos encontra-se o *getDiameterRaSbbInterface* que permite aceder à interface do RA *Diameter* para os *Sbb*s.

5.4.3.2 Package pt.ptinovacao.control.charging.events

O *package* ***pt.ptinovacao.control.charging.events*** contem os eventos já referidos anteriormente e que deverão estar acessíveis tanto para o *Enabler* de *Charging*, como para os serviços que o utilizam. Neste *package* estão incluídos os eventos tanto recebidos como lançados pelo *Enabler* de *Charging*.

5.4.4 *Perspectiva Física*

Encontra-se representado na Figura 76, e numa perspectiva física completa, as diferentes interacções entre os componentes do sistema que permitem suportar ambos os serviços.

Em particular para a taxação, alvo deste trabalho, o Sbb *Pre-Paid* comunica com o *Enabler* de *Charging*, criando uma camada de abstracção entre a comunicação SIP e os eventos lançados para o *Enabler* de *Charging*. Por seu lado, o *Enabler* de *Charging* utiliza o RA *Diameter* para comunicar com o servidor de *Charging*. Apesar desta abstracção, o papel desempenhado pelo SBB *Pre-Paid* poderia facilmente ser implementado pelos próprios serviços de Vídeo Portal e PC2Phone.

Na verdade, apenas os eventos trocados entre estes elementos funcionais vão variar, de modo que a perspectiva física não se irá alterar para qualquer outro serviço que necessite realizar taxação. Qualquer serviço que o faça, apenas terá de implementar a troca de eventos suportados pelo *Enabler* de *Charging*, tornando-se a sua adaptação extremamente simples e eficaz.

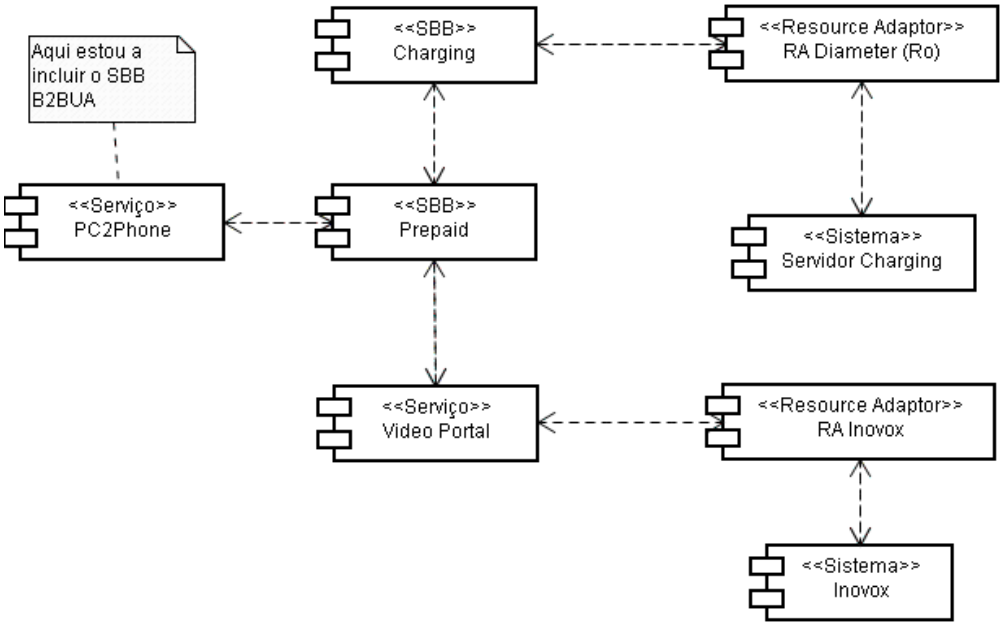

Figura 76 – Diagrama de componentes do sistema.

5.4.5 *Detalhes de Concepção*

É apresentado neste sub capítulo, a título de exemplo, as interacções entre os diferentes elementos do serviço Vídeo Portal.

Um dos elementos presentes nos diagramas de sequência das Figura 77 e Figura 78 é o "SLEE" que representa o ambiente de execução para onde são enviados e roteados os eventos assíncronos.

5.4.5.1 Acesso ao serviço de Vídeo Portal

Quando existe um acesso ao serviço de Vídeo Portal, o *Enabler* de *Charging* é informado e procede à avaliação do saldo do utilizador para verificar se o mesmo tem possibilidade de utilizar o serviço. Mediante a resposta do servidor de *Charging*, o sistema informa o serviço de Vídeo Portal para que o mesmo proceda ao toque do anúncio do vídeo respectivo, seja o menu de acesso do serviço ou vídeo indicativo da não possibilidade de utilização do serviço (Figura 77).

5.4.5.1.1 Processo de acesso ao serviço de Vídeo Portal

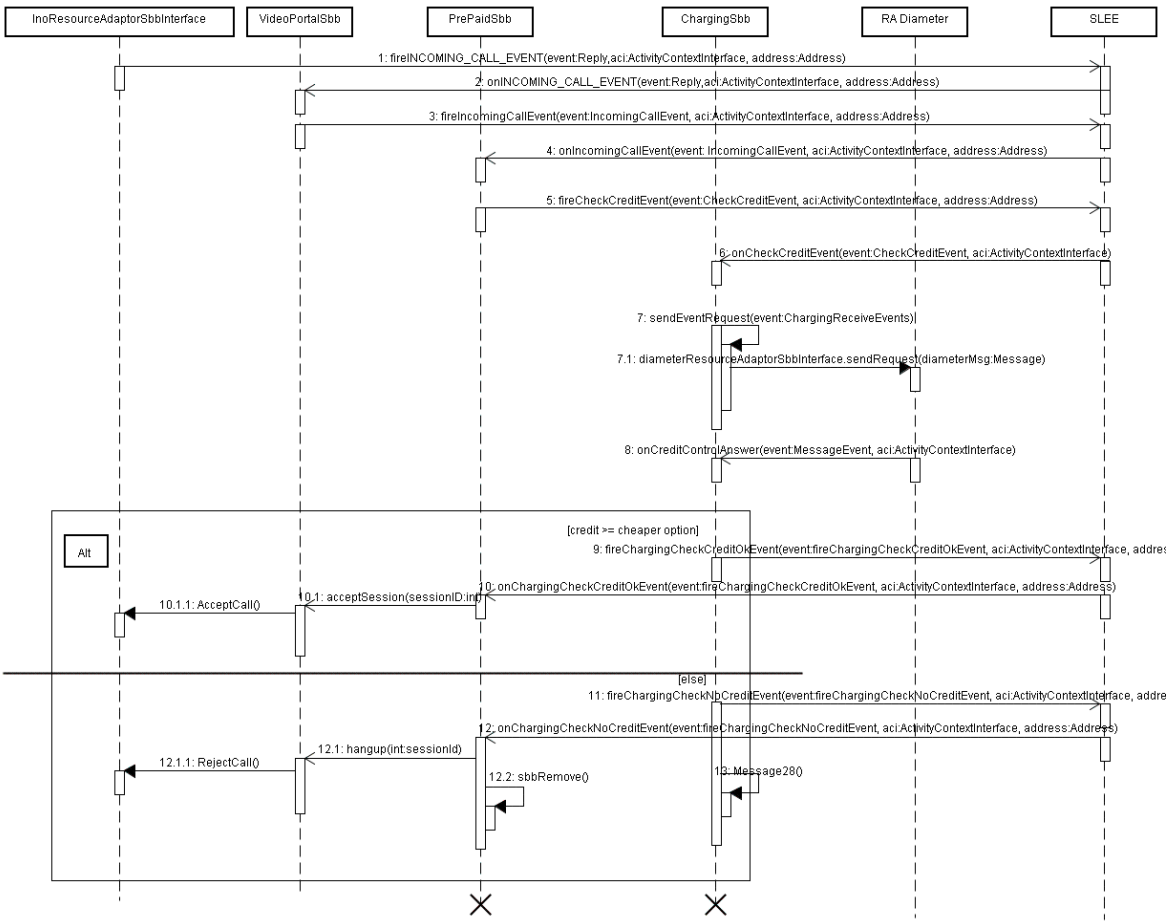

Figura 77 – Diagrama sequencial do processo de acesso ao serviço de Vídeo Portal.

5.4.5.2 Escolha de opção durante o uso do serviço de Vídeo Portal

Durante o uso do serviço de Vídeo Portal o utilizador pode escolher diferentes vídeos para visualizar. Este processo será acompanhado pelo sistema de forma a garantir que o utilizador tem saldo suficiente para a opção escolhida e neste caso garantir que a mesma é convenientemente tarifada. Caso não tenha saldo para essa opção ou caso o seu saldo seja inferior à opção mais barata o sistema encarrega-se de informar o serviço de Vídeo Portal deste facto (Figura 78).

5.4.5.2.1 Processo de escolha de opção durante o uso do serviço de Vídeo Portal

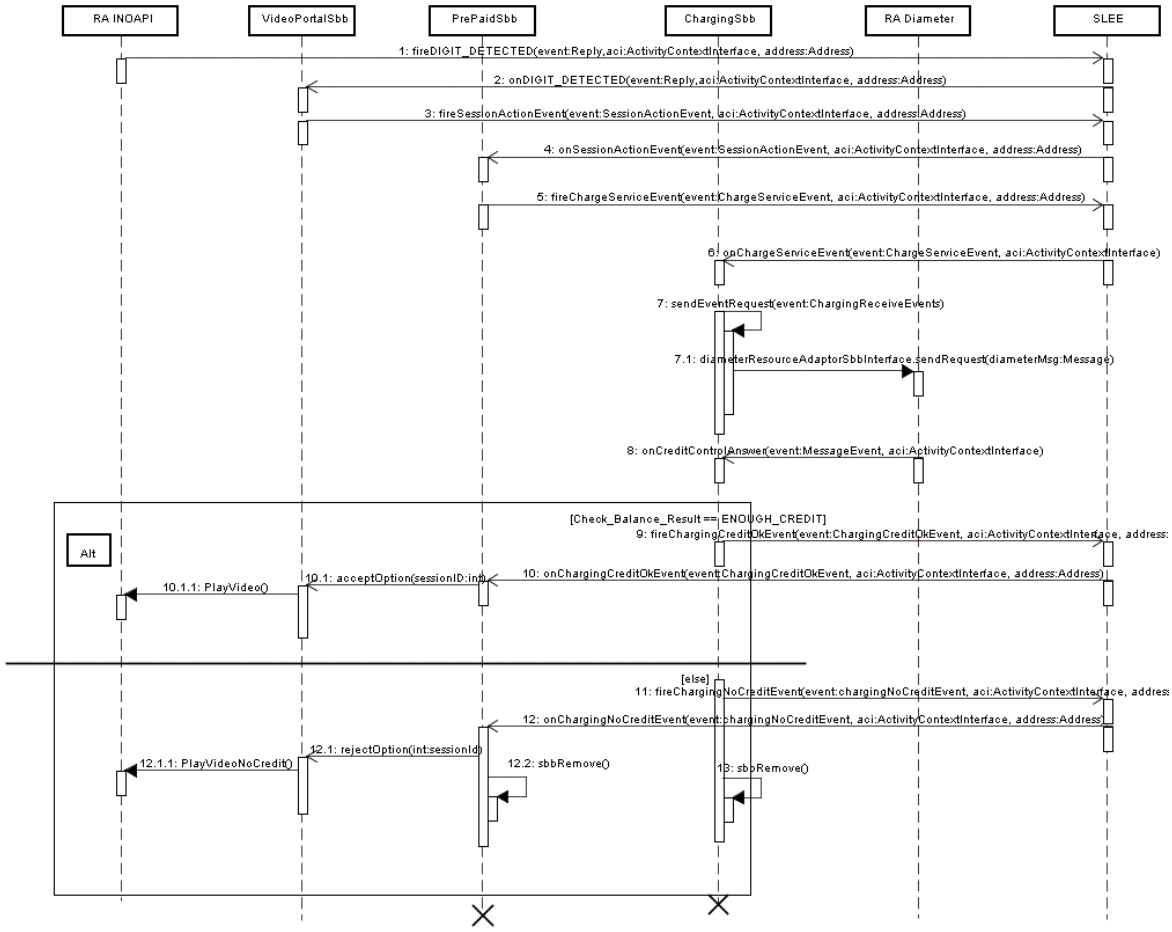

Figura 78 – Diagrama sequencial – Escolha de opção durante o uso do serviço de Vídeo Portal

5.4.6 *Avaliação do Trabalho Feito*

Após a implementação dos módulos anteriormente referidos, foram realizados alguns testes de funcionamento que permitiram a avaliar a solução desenvolvida. Estes testes

envolveram o uso do cliente SIP *eyebeam*, do core IMS e o servidor de taxação da arquitectura SH*ip*NET® da PTIN. Adicionalmente, esteve disponível um portal *web* que possibilitou a visualização/modificação do saldo dos utilizadores e o valor a taxar por cada serviço. Este portal permitiu confirmar as alterações do saldo ao utilizar os serviços.

Para avaliar o serviço Vídeo Portal, foi inicialmente aprovisionado o endereço "sip:videoportal@ptinovacao.pt" no elemento HSS. Este processo corresponde à criação de uma *Public Service Identity* (PSI) com um *trigger* para o AS onde está a correr o serviço de Vídeo Portal. O *trigger* é o elemento que permite ao S-CSCF perceber se é necessário contactar um AS. O *trigger* contem a informação que permite contactar o AS quando o utilizador liga para o "sip:videoportal@ptinovacao.pt" (ver Anexo B).

Como resultados dos testes, é possível observar algumas das mensagens trocadas pelo o AS e o servidor de *Charging* da PTIN (ver Tabela 13 e Tabela 14).

Tabela 13 – Exemplo de um pedido "Verificação de saldo"

```
<Credit_Control_Request>
        <fixed>
                Session_Id = "ccclient;1161253036;2;client"
        </fixed>
        <required>
                CC_Request_Number = "0"
                Destination_Realm = "sri.ptinovacao.pt"
                Origin_Host = "ccclient.ptinovacao.pt"
                Auth_Application_Id = "4"
                CC_Request_Type = "4"
                Origin_Realm = "ptinovacao.pt"
                Service_Context_Id = "video2@ptinovacao.pt"
        </required>
        <optional>
                <Subscription_Id>
                        <required>
                                Subscription_Id_Data = "user12@ptinovacao.pt"
                                Subscription_Id_Type = "3"
                        </required>
                </Subscription_Id>
                Requested_Action = "2"
                Event_Timestamp = "0"
                Service_Identifier = "4042"
        </optional>
</Credit_Control_Request>
```

A mensagem CCR presente na tabela anterior refere-se a um pedido de verificação de saldo entre o RA *Diameter* e o servidor de *Charging* da PTIN. Este tipo de mensagens, e tal como expõe a tabela, deverá ter presente o campo *CC-Request-Type* AVP com o valor igual a EVENT_REQUEST (4), incluir o *Requested-Action* AVP com o valor de CHECK_BALANCE (2) e o *Subscription-Id* AVP a identificar o utilizador. Este último inclui o *Subscription-Id-Data* AVP que contem a identificação do utilizador ("user12@ptinovacao.pt") e o *Subscription-Id-Type* AVP que define o tipo de identificador ("3" - endereço NAI (*Network Access Identifies*) [43])).

Ainda presente nesta mensagem, temos o *Service-Identifier* AVP que identifica o serviço Vídeo Portal (4042). O campo *Service-Context-Id* AVP tem presente o identificador do vídeo ("video2@ptinovacao.pt") no qual se pretende saber se o utilizador tem saldo suficiente.

O *CC-Request-Number* AVP identifica este pedido dentro de uma sessão. Como este pedido é do tipo evento (EVENT_REQUEST), o valor terá de ser 0.

Por último, para identificar o tipo de aplicação *Diameter Credit-Control Application* temos o *Auth-Application-Id* AVP com o valor 4 [42].

Tabela 14 – Exemplo de uma resposta "Verificação de saldo"

```
<Credit_Control_Answer>
        <fixed>
                Session_Id = "ccclient;1161253036;2;client"
        </fixed>
        <required>
                Result_Code = "2001"
                CC_Request_Number = "0"
                Origin_Host = "ccserver.ptinovacao.pt"
                Auth_Application_Id = "4"
                CC_Request_Type = "4"
                Origin_Realm = "ptinovacao.pt"
        </required>
        <optional>
                Check_Balance_Result = "0"
                Event_Timestamp = "0"
        </optional>
</Credit_Control_Answer>
```

A resposta à mensagem CCR encontra-se na Tabela 14. Esta mensagem tem a indicação se o utilizador tem saldo suficiente para visualizar o serviço. Nesta situação, e através da informação presente no *Check-Balance-Result* AVP, verifica-se que o utilizador tem saldo (0 = ENOUGH_CREDIT). Ainda importante, temos o *Result-Code* AVP com a indicação que foi realizado com sucesso todo o processo de verificação do saldo (2001 = DIAMETER_SUCCESS).

Para o Vídeo Portal, foram feitos testes com o acesso a vídeos grátis e vídeos pagos com custos distintos. Através do vídeo grátis foi possível visualizar, se mesmo sem saldo, o utilizador conseguia aceder ao vídeo. No caso dos vídeos pagos, observou-se a tentativa de aceder a vídeos com custos superiores ao saldo e o contrário, com custos inferiores ao saldo do utilizador.

No caso do serviço PC2Phone os utilizadores que realizam as chamadas com a PSTN terão de ter um *trigger* para o AS quando iniciam novas sessões. Como testes, foram realizadas várias chamadas com a PSTN variando o saldo do utilizador. Os primeiros testes foram realizados com utilizadores sem saldo, que desta forma não tinham permissão para realizar este tipo de chamadas. Como previsto, as chamadas foram sucessivamente rejeitadas.

Variando os testes, foram realizadas chamadas já com utilizadores com saldo. Nesta situação foi necessário testar se era realizado o débito correctamente no saldo do utilizador. Foi importante observar se a taxação era realizada periodicamente como planeada e ainda se as unidades reservadas mas não utilizadas eram creditadas na conta do utilizador. Para finalizar, foi testado o funcionamento do serviço quando o utilizador ficava sem saldo durante o decorrer de uma conversação.

Os resultados obtidos para ambos os serviços foram os esperados, não tendo ocorrido nenhum erro nos testes finais realizados.

Capitulo 6 – Sumário e Conclusão

Não há menor dúvida que o mundo das telecomunicações está numa grande evolução. A Internet está a desafiar as estruturas tradicionais de telecomunicações. A normalização do *IP Multimedia Subsystem* (IMS) representa a consequência natural face ao dilema de combinar os conceitos tradicionais, redes legadas, com as tecnologias emergentes VoIP e serviços Internet. O IMS pode ser considerado a última hipótese dos operadores para competirem com uma Internet aberta. A sua arquitectura está a influenciar a forma de pensar dos operadores ao nível das arquitecturas de serviços, desacoplando-as da camada de rede.

Esta nova camada de serviços deverá ver as restantes camadas inferiores de um modo *high level*, permitindo abstrair e esconder as complexidades da rede. Esses requisitos são preenchidos pelas *Service Delivery Platforms* (SDPs) que têm o objectivo de fornecer ambiente de execução que poderá ser utilizado por todos os serviços, fornecendo uma plataforma única para criação, aprovisionamento, *deployment* e controlo. Com a abstracção das complexidades da rede, os criadores de serviços vão puder criar serviços mais facilmente e rapidamente onde os serviços de voz, dados e vídeo poderão ser combinados compondo serviços multimédia mais atraentes. Através da análise do mercado constata-se já uma enorme variedade de implementações, vários vendedores com SDPs já disponíveis, sendo Java a tecnologia que prevalece, destacando-se dois ambientes de execução, J2EE (virado para a *Web*) e JSLEE (virado para telecomunicações).

A tecnologia JSLEE tem um potencial imenso. Vai modificar completamente o mercado das telecomunicações permitindo um acesso directo ao desenvolvimento de serviços por todos os actores de telecomunicações no mundo, independentemente dos sistemas. O grande senão, pelo menos para já, da tecnologia JSLEE é o seu estágio ainda um pouco imaturo e com poucos fornecedores no mercado com soluções comerciais.

Com o auxílio da *Open Mobile Aliance* (OMA), e ainda na camada aplicacional, vai ser possível assegurar a interoperabilidade entre os diferentes serviços, bem como ainda a interoperabilidade e independência da rede de acesso, dos fornecedores de serviços, entre outros. Com o objectivo de harmonizar as diferentes organizações de normalização numa única, vai ser possível um ambiente único para a criação de *Service Enablers* que depois poderão ser utilizados pelas SDPs, estando deste modo acessíveis aos criadores de serviços, abstraindo-os das questões de interoperabilidade.

Para garantir que os operadores consigam lucrar com aumento exponencial da diversidade e do número de serviços é necessário que existam novos modelos de taxação que primam pela eficiência e flexibilidade. É preciso perceber que os serviços de voz tradicionais estão a ser ultrapassados por serviços de dados, sendo necessário encontrar uma forma de tarifação adequada a cada tipo de serviço. O 3GPP definiu então uma *framework* de *charging* genérica que responde a essas necessidades, suportando uma taxação mais eficiente e flexível para a próxima vaga de serviços disponibilizados.

Com este projecto desenvolveu-se um conjunto de módulos que permitem tarifar, em tempo real, os utilizadores nas redes de próxima geração, de acordo com os serviços utilizados. Para tal, foi necessário inicialmente um estudo de todas especificações 3GPP e RFCs associados tanto à taxação como, obviamente, às redes de próxima geração. Este trabalho, apesar de ter como objectivo particular tarifar dois serviços específicos (Vídeo Portal e PC2Phone), foi realizado de tal forma genérico, que é facilmente integrado em novos serviços que estejam aprovisionados num ambiente JSLEE. Todos os objectivos propostos foram atingidos, tendo sido realizada com sucesso a integração com os restantes módulos dos serviços Vídeo Portal e PC2Phone.

Como trabalho futuro, existe a necessidade de implementar as funcionalidades de taxação *offline*, possibilitando novos modelos de negócio. Além de novas funcionalidades, era importante integrar o trabalho já realizado na solução da PTIN. Esta integração poderá ser facilmente realizada já que a solução da PTIN para plataformas de serviço em redes convergentes tem como ambiente de execução o JSLEE. Outro desafio, não tão prioritário, poderá passar pela evolução do trabalho realizado para a *release* 7 das especificações do 3GPP.

Apesar da imaturidade de todas estas novas tecnologias verificou-se com o trabalho realizado, que já é possível criar e disponibilizar serviços rapidamente. Com esta nova realidade e com os mecanismos adequados de taxação os operadores têm a possibilidade de obter as receitas que tanto anseiam e recuperarem as perdas para as tecnologias tipo VoIP e Internet.

Referências

[1] International Telecommunication Union (Setembro 2007):
http://www.itu.int

[2] ITU Telecommunication Standardization Sector (Setembro 2007):
http://www.itu.int/ITU-T

[3] Internet Engineering Task Force (Setembro 2007):
http://www.ietf.org/

[4] Overview of the IETF (Setembro 2007):
http://www.ietf.org/overview.html

[5] "A Mission Statement for the IETF", RFC3935, Outubro 2004.

[6] "The Internet Standards Process - Revision 3", RFC2026, Outubro 1996.

[7] 3rd Generation Partnership Project (Setembro 2007):
http://www.3gpp.org/

[8] Estrutura 3GPP (Setembro 2007):
http://www.3gpp.org/tb/home.htm

[9] 3rd Generation Partnership Project 2 (Setembro 2007):
http://www.3gpp2.org/

[10] European Telecommunications Standards Institute (Setembro 2007):
http://www.etsi.org

[11] Telecoms and Internet converged Services and Protocols for Advance Network (Setembro 2007):
http://www.etsi.org/tispan/

[12] Estrutura TISPAN (Setembro 2007):
http://www.etsi.org/tispan/#Structure

[13] Open Mobile Alliance (Setembro 2007):
http://www.openmobilealliance.org/

[14] "3GPP-IETF Standardization Collaboration", RFC3113, Junho 2001.

[15] "3GPP2-IETF Standardization Collaboration", RFC3131, Junho 2001.

[16] "OMA-IETF Standardization Collaboration", RFC3975, Janeiro 2005.

[17] Gonzalo Camarillo e Miguel Martin; "The 3G IP Multimedia Subsystem", Wiley, 2004.

[18] "Service requirements for the Internet Protocol (IP) multimedia core network subsystem", 3GPP TS 22.228, Stage 1, Release 7, Setembro 2007.

[19] "End-to-end Quality of Service (QoS) concept and architecture", 3GPP TS 23.207, Release 7, Junho 2007.

[20] “Quality of Service (QoS) concept and architecture”, 3GPP TS 23.107, Release 7, Junho 2007.

[21] “Network architecture”, 3GPP TS 23.002, Release 7, Outubro 2007.

[22] "IP Multimedia (IM) session handling; IM call model", 3GPP TS 23.218, Stage 2, Release 7, Dezembro 2006.

[23] “3G security; Access security for IP-based services”, 3GPP TS 33.203, Release 7, Outubro 2007.

[24] “Policy and charging control architecture”, 3GPP TS 23.203, Release 7, Setembro 2007.

[25] "SIP: Session Initiation Protocol", RFC3261, Junho 2002.

[26] “Media Gateway control protocol”, ITU-H248, Junho 2000.

[27] “Functional description of the message transfer part (MTP) of Signalling System No. 7”, ITU-Q.701, Março 1993.

[28] “RTP: A Transport Protocol for Real-Time Applications”, RFC3550, Julho 2003.

[29] “RTP Payload for DTMF Digits, Telephony Tones and Telephony Signals”, RFC2833, Maio 2000.

[30] Apresentação “IP Multimedia Subsystem(IMS) and Its Applications”, Jun-Won Lee, Nable Communications, Inc., Abril 2007.

[31] “Diameter Base Protocol”, RFC3588, Setembro 2003.

[32] “IP Multimedia Subsystem (IMS) Sh interface; Signalling flows and message contents”, 3GPP TS 29.328, Release 6, Setembro 2007.

[33] “IP Multimedia Call Control Protocol based on Session Initiation Protocol (SIP) and Session Description Protocol (SDP); Stage 3”, 3GPP, TS 24.229, Release 6, Setembro 2007.

[34] “Open Service Architecture (OSA) Application Programming Interface (API) - Part 1”, 3GPP TS 29.198, R99, Junho 2001.

[35] “Open Service Access (OSA); Stage 2”, 3GPP TS 23.198, Release 7, Março 2007.

[36] “Customized Applications for Mobile network Enhanced Logic (CAMEL); CAMEL Application Part (CAP) specification for IP Multimedia Subsystems (IMS)”, 3GPP TS 29.278, Release 7, Dezembro 2005.

[37] “Mobile Application Part (MAP) specification”, 3GPP TS 29.002, Release 7, Setembro 2007.

[38] “Uniform Resource Identifiers (URI): Generic Syntax”, RFC2396, Agosto 1998.

[39] “The tel URI for Telephone Numbers”, RFC3966, Dezembro 2004.

[40] “Interworking between Session Initiation Protocol (SIP) and Bearer Independent Call Control protocol or ISDN User Part”, ITU-Q.1912.5, Março 2004.

[41] “Diameter Session Initiation Protocol (SIP) Application”, RFC4740, Novembro 2006.

[42] “Diameter Credit-Control Application”, RFC4006, Agosto 2005.

[43] ”The Network Access Identifier”, RFC2486, Janeiro 1999.

[44] “A DNS RR for specifying the location of services (DNS SRV)”, RFC2782, Fevereiro 2000.

[45] “Diameter Mobile IPv4 Application”, RFC 4004, Agosto 2005.

[46] "IMS Multimedia Telephony Communication Service and Supplementary Services, Stage3", 3GPP TS 24.173, Release 7, Março 2007.

[47] “IP Multimedia Subsystem (IMS); Multimedia telephony; Media handling and interaction”, 3GPP TS 26.114, Release 7, Setembro 2007.

[48] “Voice Call Continuity (VCC) between Circuit Switched (CS) and IP Multimedia Subsystem (IMS); Stage 2”, 3GPP TS 23.206, Release 7, Setembro 2007.

[49] “Combining Circuit Switched (CS) and IP Multimedia Subsystem (IMS) services; Stage 2”, 3GPP TS 23279, Release 7, Setembro 2007.

[50] Moriana Group, “Service Delivery Platforms And Telecom Web Services”, Junho 2008.

[51] JSLEE (Setembro 2007):

http://jcp.org/en/jsr/detail?id=22

[52] OMA Enabler IMSinOMA (Setembro 2007):

http://www.openmobilealliance.org/release_program/ims_v1_0.html

[53] “Utilization of IMS capabilities Architecture”, OMA Enabler IMSinOMA, v1.0 Aprovada, Agosto 2005.

[54] White paper, The JAIN APIs: Integrated Network APIs for the Java Platform (Setembro 2007):

http://java.sun.com/products/jain/WP2002.pdf

[55] JAIN Initiative (Setembro 2007):

http://java.sun.com/products/jain

[56] APIs JAIN (Outubro 2007):

http://java.sun.com/products/jain/api_specs.html

[57] “JAIN SLEE Technology Overview”, Michael Maretzke, Maretzke.De

[58] "JAIN SLEE 1.0 Specification, Final Release".

[59] Ralph Kühne, Gerald Görmer, Morten Schläger e Georg Carle, “Charging in the IP Multimedia Subsystem: A Tutorial”, IEEE Communications Magazine, Julho 2007.

[60] Miikka Poikselkã, Georg Mayer, Hisham Khartabil, Ali Niemi, “The IMS IP Multimedia Concepts and Services in the Mobile Domain”, Wiley, 2004.

[61] “Telecommunication management; Charging management; Charging architecture and principles” 3GPP TS 32.240, Release 6, Outubro 2006.

[62] “Telecommunication management; Charging management; IP Multimedia Subsystem (IMS) charging”3GPP TS 32.260, Release 6, Março 2007.

[63] “Telecommunication management; Charging management; Online Charging System (OCS): Applications and interfaces”, 3GPP TS 32.296, Release 6, Outubro 2006.

[64] “Telecommunication management; Charging management; Diameter charging applications”, 3GPP TS 32.299, Release 6, Outubro 2007.

[65] “Overall high level functionality and architecture impacts of flow based charging; Stage 2” 3GPP TS 23.125, Release 6, Março 2006.

[66] “IP Multimedia (IM) Subsystem Cx and Dx Interfaces; Signalling flows and message contents”, 3GPP TS 29.228, Release 6, Setembro 2007.

[67] Java New I/O APIs (Setembro 2007):

http://java.sun.com/j2se/1.4.2/docs/guide/nio/,.

[68] Mobicents – The Open Source SLEE and SIP Server (Setembro 2007):

http://www.mobicents.org.

[69] "PSTN/ISDN simulation services; Originating Identification Presentation (OIP) and Originating Identification Restriction (OIR); Protocol specification", ETSI TS 183 007, v1.1.1, Maio 2006.

[70] "PSTN/ISDN simulation services; Terminating Identification Presentation (TIP) and Terminating Identification Restriction (TIR); Protocol specification", ETSI TS 183 008, v1.1.1, Maio 2006.

[71] “PSTN/ISDN simulation services: Explicit Communication Transfer (ECT); Protocol specification”, ETSI TS 183 029, v1.1.1, Abril 2006.

[72] "PSTN/ISDN simulation services: Communication HOLD (HOLD); Protocol specification”, ETSI TS 183 010, v1.2.1, Abril 2006.

[73] "PSTN/ISDN simulation services: Communication Diversion (CDIV); Protocol specification", ETSI TS 183 004, v1.1.1, Abril 2006.

[74] "PSTN/ISDN simulation services: Conference (CONF); Protocol specification", ETSI TS 183 005, v1.1.1, Maio 2006.

[75] “PSTN/ISDN simulation services: Message Waiting Indication (MWI); Protocol specification”, ETSI TS 183 006, v1.1.1, Março 2006.

[76] "PSTN/ISDN simulation services: Anonymous Communication Rejection (ACR) and Communication Barring (CB); Protocol specification", ETSI TS 183 011, v1.1.1, Maio 2006.

[77] ”The E.164 to Uniform Resource Identifiers (URI); Delegation Discovery System (DDDS) Application (ENUM)”, RFC 3761, Abril 2004.

[78] “SIP: Locating SIP Servers”, RFC 3263, Junho 2002.

[79] Apresentação “FOKUS Competence Center”, Prof. Dr. Thomas Magedanz, Fraunhofer Institute FOKUS.

[80] Apresentação “Tutorial 1: Implementing efficiently FMC and Triple Play”, Prof. Dr. Thomas Magedanz, Fraunhofer Institute FOKUS.

Anexo A – Organizações de Normalização

De modo a criar as Redes de Próxima Geração (em inglês, *Next Generation Networks* - NGNs) é necessário defini-las em diferentes fases de implementação. Não só é necessário definir a arquitectura global da rede, como é ainda essencial definir todos os elementos (funcionalidades e procedimentos) bem como as diferentes ligações entre eles, que levem à definição de uma rede que vá ao encontro dos requisitos do mercado.

Para tal, é necessário a existência de organizações, com diferentes responsabilidades, que colaborem e contribuam para o desenvolvimento e evolução global destas redes para uso comercial por parte dos operadores de telecomunicações.

Este anexo pretende apresentar alguns dos organismos que mais contribuem para o desenvolvimento das NGNs, descrevendo para cada organização o nível onde se insere o seu trabalho e ainda as colaborações existentes entre elas de modo a criar sistemas inter-operacionais.

1 International Telecommunication Union (ITU)

A ITU [1] foi criada em 1865 com o objectivo de resolver problemas no âmbito do tráfico primórdio das telecomunicações. Hoje, a ITU é uma organização internacional, onde membros de vários países e várias entidades comerciais se juntam para criar normas na área das telecomunicações. Esta organização encontra-se dividida em diferentes sectores, existindo um sector responsável pela criação de recomendações e normas de acordo com as necessidades do sector das telecomunicações designado por ITU-T [2]. Neste momento a ITU-T está concentrada na normalização das NGNs dando contribuições importantes através das suas recomendações (ITU-T *Recommendations*).

2 Internet Engineering Task Force (IETF)

O IETF [3] é um organismo internacional aberto ao público constituído por uma comunidade de operadores de telecomunicações, vendedores, investigadores. O seu trabalho em conjunto tem o intuito de contribuir para a evolução e melhoramento da arquitectura da Internet, desenvolvendo a maioria dos protocolos que são actualmente utilizados na Internet [4]. A missão desta organização encontra-se documentada em [5].

O IETF não pretende normalizar redes, elementos (nós) de arquitecturas ou mesmo arquitecturas que combinem diferentes protocolos. O seu objectivo é simplesmente

funcionar como uma "fábrica" de protocolos para a rede IP, que possam ser aproveitados para as interfaces entre os elementos.

O IETF encontra-se dividido em vários grupos de trabalho, estando organizados por diferentes áreas (exemplo, *routing*, transporte, segurança, etc.). Estes grupos de trabalho são formados por um número de voluntários que trabalham de modo individual, não representando as suas companhias quando trabalhando para o IETF.

A maioria das discussões técnicas dentro de um grupo de trabalho é feito via e-mail, reunindo-se apenas três vezes por ano, criando documentos técnicos que são o resultado de contribuições para a elaboração e publicação desses documentos. Após a revisão e publicação dos documentos criados o grupo de trabalho é dissolvida da organização.

Os grupos de trabalho são agrupados por áreas, sendo geridas por *Area Directors* (ADs), membros do *Internet Engineering Steering Group* (IESG).

Os documentos produzidos pelos grupos de trabalho são designados por *Internet-Drafts*. Quando terminados, são submetidos a uma revisão pelo IESG que decide posteriormente se o *Internet-Draft* deve ser publicado num novo *Request for Comments* (RFC). Este processo está documentado no [6].

Muitos dos protocolos especificados pelo IETF estão presentes nas NGNs, estando esta organização concentrada no desenvolvimento de extensões para estes protocolos, como é o caso do SIP e do Diameter, muito utilizados nestas redes, e apresentados nesta dissertação.

3 Third Generation Partnership Project (3GPP)

O 3GPP [7] foi um projecto criado em Dezembro de 1998 por cinco empresas de tecnologia móvel, a ETSI (Europa), ARIB/TTC (Japão), CCSA (China), ATIS (América do Norte) e TTA (Coreia do Sul), com o propósito inicial de desenvolver especificações para sistemas móveis de terceira geração baseados nas redes GSM (*Global System for Mobile communication*). Actualmente o 3GPP aumentou o seu âmbito, incorporando novas tarefas para incluir a manutenção e o desenvolvimento de especificações para GSM incluindo o suporte e evolução das redes móveis e tecnologias de acesso por pacotes.

Adicionalmente aos parceiros organizacionais, o 3GPP tem parceiros que representam o mercado e que proporcionam à organização requisitos que vão de encontro às necessidades do mercado. Nestes parceiros estão incluídos, entre outros, o UMTS Forum, 3G Americas, a GSM Association, a Global mobile Suppliers Association, o Forum TD-SCDMA, e o Forum IPv6.

O 3GPP encontra-se organizado em grupos designados por *Technical Specification Groups* (TSG) sendo estes geridos e supervisionados pela entidade *Project Co-ordination Group*

(PCG) dentro do 3GPP [8]. Actualmente o 3GPP está estruturado em quatro TSGs que aprovam as normas desenvolvidas, resultando documentos finais designados por *Technical Specifications* (TS) e *Technical Reports* (TR). Todas estas especificações resultantes do trabalho de cada uma das TSGs são agrupadas pelo 3GPP em *releases*.

A primeira versão do *core* IMS, utilizada nas redes NGNs, foi introduzida pelo 3GPP na *release* 5 em 2002, que deu a conhecer uma arquitectura base deste sistema para a convergência de serviços multimédia baseados em IP com as redes móveis. Mais tarde, em 2005, surgiu a *release* 6 que introduziu a este sistema algumas melhorias como PoC, GAN, MBMS, etc., bem como a operação com as redes de acesso WLAN. Também já finalizada encontra-se a *release* 7, que engloba já o trabalho desenvolvido pelo grupo TISPAN da ETSI para a evolução das NGNs, na convergência com as redes de acesso fixas, nomeadamente xDSL, para os serviços multimédia.

Desde que o 3GPP foi criado para a especificação técnica da redes de terceira geração baseada no sistema móvel europeu GSM, nasceu a necessidade de criar uma organização com a mesma missão que o 3GPP para os sistemas móveis da América do Norte e Ásia, baseado na normalização ANSI. Este foi o motivo que deu origem à organização 3GPP2 [9]. A organização e estrutura do 3GPP2 é muito idêntica ao do 3GPP e opera de forma muito similar. A versão do IMS do 3GPP2, designada por *Multimedia Domain* (MMD), é similar à definida pelo 3GPP, apresentando algumas diferenças [17]:

- A arquitectura 3GPP2 IMS está assente em *mobile* IP, enquanto a mobilidade no 3GPP foi pensada para GPRS;
- A descoberta do P-CSCF é realizada no 3GPP através da activação do *PDP context* e DHCP enquanto o 3GPP2 apenas suporta DHCP, já que a activação do *PDP context* é um procedimento específico do GPRS;
- Diferentes *codecs* de voz;
- *Home Subscriber Server*: o 3GPP HSS é uma agregação do 3GPP2 *Home AAA* e as suas bases de dados;
- Interface Sh: No 3GPP, alguma da informação fornecida por esta interface aos ASs não se encontra presente no 3GPP2;
- Etc.

4 European Telecommunication Standards Institute (ETSI)

O ETSI [10] é uma organização independente e sem lucros cuja missão é a produção de normas europeias nas telecomunicações, desenvolvendo igualmente actividades de pré-

normalização e normalização nas áreas das tecnologias da informação e da radiodifusão televisiva e sonora.

Sedeado em Sophia Antipolis, no sul de França, o ETSI integra mais de 900 membros, oriundos de cinquenta e cinco países, incluindo administrações, operadores de rede, prestadores de serviços, centros de investigação e consumidores. O programa de trabalho do Instituto é determinado pelos seus membros, que também são responsáveis pela aprovação das deliberações. Como resultado, as actividades do ETSI estão intimamente ligadas com as necessidades do mercado, expressas pelos seus membros.

O trabalho nesta organização encontra-se dividido em *Technical Committes* (TCs) com determinadas tarefas e responsabilidades.

Em Setembro de 2003, a ETSI juntou duas TCs: TIPHON e SPAN, dando origem a uma única TC designada por TISPAN [11] com principal ênfase para a convergência entre as redes móveis e fixas, utilizando o *core* IMS do 3GPP como elemento central no acesso a aplicações multimédia baseadas em IP aos utilizadores destas redes. Esta TC está organizada em oito grupos de trabalho com diferentes responsabilidades.

Em Dezembro de 2005 o TISPAN publicou a sua primeira *release* com a arquitectura global das NGNs e neste ano saiu a *release* 2 com principal foco á integração de novas redes de acesso e serviços (ex. IPTV, etc.).

5 Open Mobile Alliance (OMA)

A OMA [13] tem como missão facilitar a adopção global dos seus serviços móveis por parte dos utilizadores, especificando *service enablers* orientados para os requisitos do mercado que assegurem a interoperabilidade com os diferentes dispositivos móveis, regiões geográficas, fornecedores de serviços, operadores de redes móveis, e as próprias redes. Como resultado, permitem simultaneamente negócios mais virados para a inovação e diferenciação.

Com uma visão centrada no utilizador, a OMA assegura uma aceitação rápida e uma elevada proliferação dos serviços móveis, perdendo as empresas menos tempo com questões de integração e investigação aproveitando-o para aumentar a qualidade dos serviços.

A OMA foi formada em Junho de 2002, tendo sido criada com a consolidação da iniciativa *Open Mobile Architecture* e do *WAP Forum*. Desde essa data, novas organizações integraram este projecto, como são os casos, entre outros, do *Location Interoperability Forum* (LIF), da iniciativa *SyncML*, do *MMS Interoperability Group* (MMS-IOP), *Wireless*

Village, *Mobile Gaming Interoperability Forum* (MGIF), *Mobile Wireless Internet Forum* (MWIF).

A estrutura da OMA é constituída pelo *Technical Plenary*, responsável pela aprovação e manutenção das especificações técnicas da OMA. O *Technical Plenary* é organizado por um número de *Technical Working Groups* (TWGs) e por duas comissões.

Desta organização resultam *Release Packages*, que consistem num conjunto de especificações OMA, produzidos pelos TWGs.

A OMA define diferentes níveis de maturidade para as suas *releases*, sendo estas chamadas de *phases* (fases) na terminologia da OMA. Cada OMA *ReleasePackage* pode encontrar-se numa das seguintes fases:

- Fase 1: *Candidate Enabler Release* – Estado inicial da *release*.
- Fase 2: *Approved Enabler Release* – A *release* passou com sucesso os testes de interoperabilidade.
- Fase 3: OMA *Interoperability Release* – A *release* foi exaustivamente testada com outros OMA *service enablers* onde possa existir interoperabilidade.

Os objectivos da OMA facilmente se resumem a:

- Fornecer especificações técnicas baseadas nas exigências do mercado que promovam a modularidade, expansibilidade e consistência entre *enablers* para reduzir esforços na fase de implementação;
- Assegurar que as suas especificações forneçam interoperabilidade entre os diferentes dispositivos, fornecedores de serviços, operadores, e redes;
- Consolidar as diferentes actividades de normalização dentro da indústria de serviços móveis de dados; trabalhar juntamente com outras organizações e indústria para melhorar a interoperabilidade e diminuir os custos operacionais envolventes;
- Proporcionar benefícios aos membros da OMA de forma a elegerem participar activamente nesta organização;

6 Relação entre as Organizações

A Figura 79 ilustra as relações de cooperação entre as diferentes organizações de normalização.

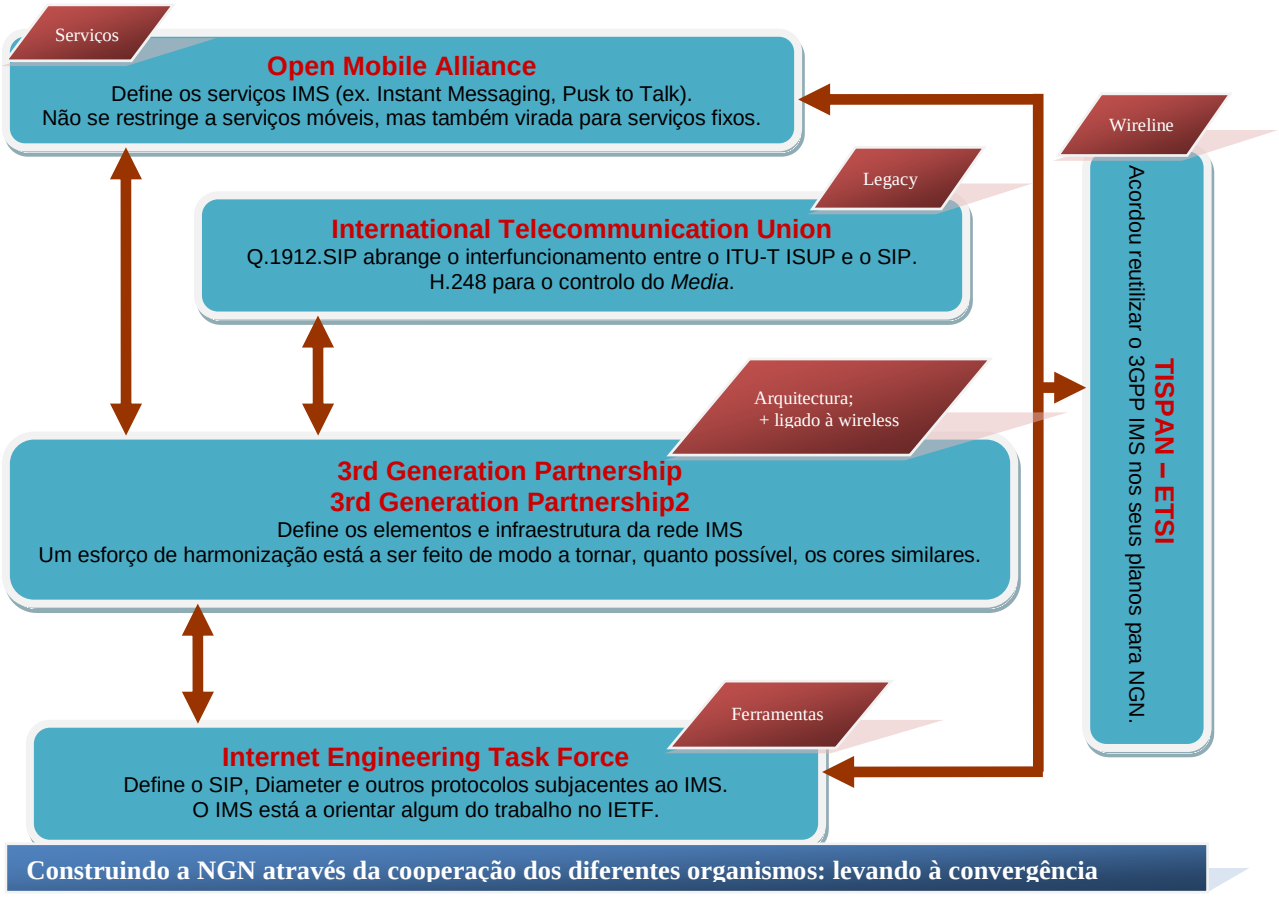

Figura 79 – Cooperação entre os organismos de normalização [80]

A NGN definida pelo TISPAN tem como base a rede IMS especificada na *release* 6 do 3GPP. Desta forma, é possível observar-se um trabalho em conjunto entre estas duas organizações com o objectivo de encontrarem uma solução comum, que permita responder às necessidades do TISPAN para a rede fixa como rede de acesso (Figura 80).

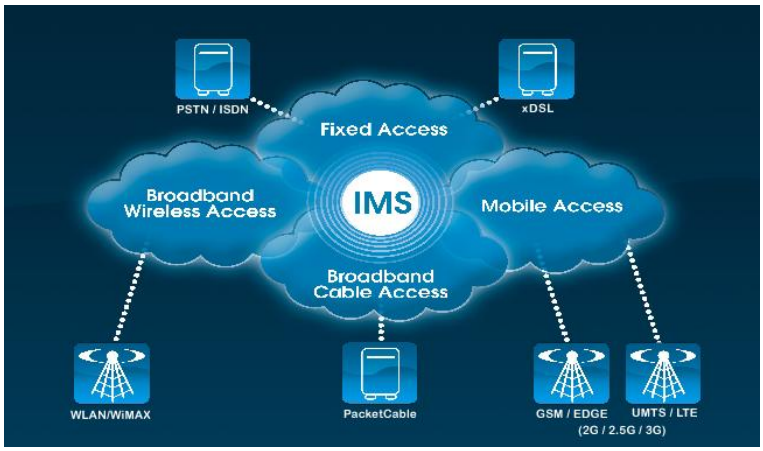

Figura 80 – Relação TISPAN/IMS [11]

Ao nível dos protocolos eleitos por estas duas organizações, é importante salientar a relação com o IETF, pois muitos dos protocolos presentes na arquitectura IMS foram desenvolvidos por esta organização, tendo agora o papel de contribuir com novos

protocolos bem como extensões para os já existentes, e que se encontram nas redes definidas pelo 3GPP e pelo TISPAN. Aliás, algumas das interfaces definidas pelo 3GPP estão agora a ser adoptadas pelo IETF, definindo aplicações genéricas que possam ser utilizadas por outras arquitecturas, à parte do IMS. Dois exemplos são a interface IMS Cx e Ro que resultaram nas aplicações IETF *Diameter Session Initiation Protocol (SIP) Application* [41] e *Diameter Credit-Control Application* [42], respectivamente.

A colaboração ente o 3GPP e o IETF está especificada em [14] e entre o 3GPP2 e o IETF em [15].

No caso da OMA, alguns dos *Technical Working Groups* usam o IMS como base para o desenvolvimento dos seus serviços. Em muitos casos, temos situações onde novos requisitos são necessários no IMS de modo a responder às exigências para a implementação de um novo serviço OMA. Devido a esse facto, foi criado um acordo entre o 3GPP/3GPP2 e a OMA de modo a integrar os requisitos encontrados pela OMA na arquitectura IMS. A OMA passa a fornecer os requisitos ao 3GPP evitando existir versões diferentes do IMS: 3GPP IMS e o IMS com as extensões para a OMA. Existindo uma organização única a gerir e manter as especificações do IMS é possível garantir a interoperabilidade entre as diferentes implementações do IMS dos diferentes vendedores.

Apesar do acordo, não existe uma distinção clara entre o IMS e os serviços em cima deste. Por exemplo, conferência pode ser considerado um serviço, mas é especificado pelo 3GPP como parte do IMS. No caso do serviço de presença a situação ainda é mais interessante, dado que tanto o 3GPP como a OMA encontram-se a trabalhar nesta área. Mesmo com ambos a trabalharem em questões similares, como é o caso de presença, o seu objectivo é ter especificações compatíveis.

Para assegurar que todos os serviços OMA possam usufruir do IMS (como especificado pelo 3GPP e o 3GPP2) foi criado pela OMA a IMSinOMA *Enabler Release Package*. Esta *release* contem os requisitos que permitem assegurar a interoperabilidade com a rede IMS, funcionando como um guia para a criação dos serviços interoperáveis.

Na mesma forma que o 3GPP e o 3GPP2, as especificações da OMA incluem referências a documentos do IETF devido à utilização dos seus protocolos e extensões. Esta colaboração encontra-se documentada em [16] e consiste apenas numa conjunto de engenheiros a trabalharem em colaboração, basicamente ao nível dos *Working Groups*.

No caso das redes legadas, e através da recomendação [40], o ITU é um dos responsáveis pela interligação entre o ISUP e o SIP para controlo de *Media*.

Anexo B – Interacção com a camada de serviços

Este anexo pretende descrever a interacção entre os ASs e a camada de controlo do IMS (Figura 81) como especificado em [21], [22]. Esta interacção é realizada utilizando a interface IMS *Service Control* (ISC).

Quando o S-CSCF recebe uma mensagem SIP do utilizador (através da *interface* Mw), o S-CSCF necessita avaliar para que AS(s) tem de enviar este evento, tomando a decisão com base em iFCs (*initial Filter Criteria*). Estes iFCs são informação armazenada no HSS e que constituem uma porção do perfil do utilizador que é enviada para o S-CSCF durante o processo de registo do utilizador na rede IMS, ou após a recepção de um pedido terminal para um utilizador que não se encontre registado. Após obter o perfil do utilizador do HSS, o S-CSCF obtém então os *filter criteria* (informação presente na mensagem SAR da interface Cx que liga o S-CSCF e o HSS). Esta informação é válida durante o tempo de vida do registo do utilizador ou até suceder uma mudança no perfil do utilizador.

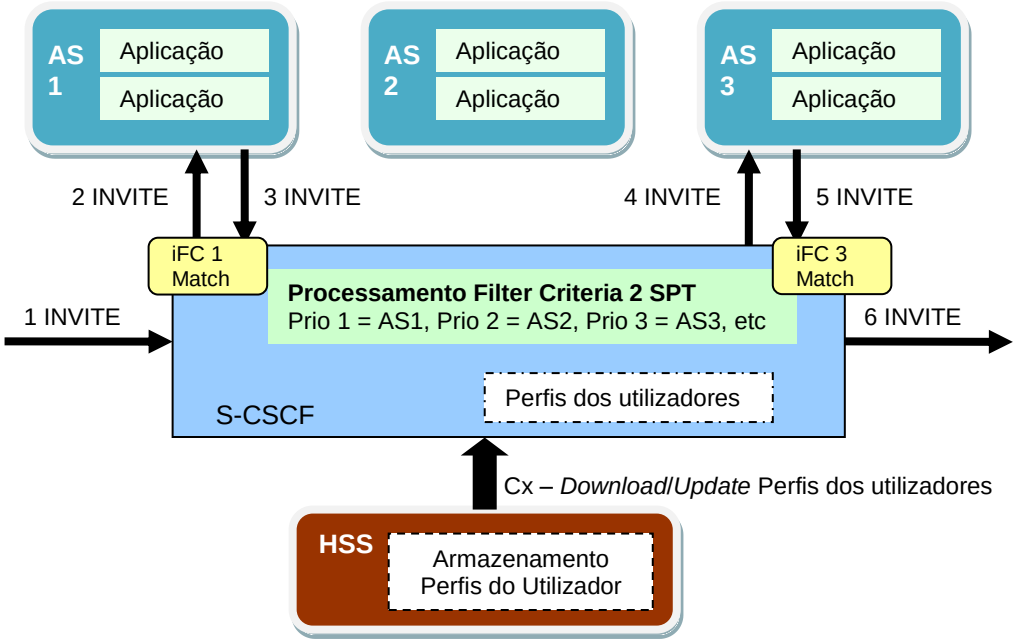

Figura 81 – Arquitectura de *triggers* para os AS

Os iFCs são dos elementos mais importantes da informação do utilizador armazenados na rede, porque são eles que determinam os serviços que serão prestados aos utilizadores. Um *Filter Criteria* contem a informação relativa ao utilizador que ajuda ao S-CSCF decidir qual ou quais os ASs a contactar (reencaminhamento do pedido SIP) para fornecer um ou mais serviços a um utilizador. Um conjunto de *Initial Filter Criteria* encontra-se

armazenado no HSS para cada utilizador, para cada aplicação ou serviço que o utilizador invocou. Cada grupo de *filter criteria* inclui um endereço de um AS, uma prioridade, *trigger points* e *service info*. Quando existem múltiplos *initial filter criteria* atribuídos ao mesmo utilizador, o S-CSCF avalia-os consoante a sua ordem numérica, isto é, um *initial filter criterion* com uma prioridade mais elevada será atendido depois de um com menor prioridade.

Cada *trigger point* funciona como uma expressão lógica composta por *Service Point Triggers* (SPTs) que são condições que a mensagem SIP, que chegou ao S-CSCF, terá de respeitar para o pedido ser reencaminhado para o AS correspondente deste iFC.

Estes SPTs podem então ser composto por:

- O valor do *Request-URI* da mensagem;
- O método do pedido SIP (por exemplo, INVITE, OPTIONS, SUBSCRIBE, etc.);
- A presença ou ausência de um cabeçalho SIP;
- Uma parcela ou a totalidade do conteúdo de um cabeçalho SIP;
- O tipo de sessão (S*ession-Case*), isto é, se o pedido SIP foi originado pelo utilizador a servido, ou se pelo contrário foi endereçado ao utilizador servidor (que pode estar ou não registado);
- A descrição da sessão, isto é, uma parcela ou totalidade de uma linha da informação do *Session Description Protocol* (SDP) presente no corpo da mensagem SIP;

Como foi dito anteriormente, e de forma a permitir o S-CSCF tratar os diferentes iFCs na sequência correcta, é necessário atribuir a cada um uma prioridade. Caso o S-CSCF não consiga atingir o AS, o S-CSCF deverá aplicar o *default handling* (comportamento em caso de erro) associado ao *trigger* (apenas presente no perfil do utilizador a partir da *release* 6 [66]). Este *default handling* tem como opções:

- Continuar a verificação dos restantes iFCs com menor prioridade presentes no perfil do utilizador;
- Abandonar a verificação dos restantes iFCs, e libertar o diálogo, respondendo com uma mensagem de erro para o utilizador;

De forma resumida, cada iFC presente no XML enviado ao S-CSCF contem a seguinte informação:

- Endereço do AS a contactar em caso de correspondência do iFC;
- Prioridade do iFC, que permitir sequenciar os iFCs (dois iFCs obviamente não poderão ter a mesma prioridade);

- *Trigger Points* compostos por uma ou mais instâncias de SPTs. Os SPTs podem ser ligados por meio de expressões lógicas;
- *Default Handling* (descrito anteriormente);
- Informação de serviço adicional que deverá ser adicionada ao corpo da mensagem antes de ser enviada para o AS (apenas no caso de mensagens REGISTER);

Anexo C – Identificação dos utilizadores[18]

No domínio IMS os utilizadores têm associadas identidades que se encontram relacionadas entre si.

Identidade Pública do Utilizador

Uma das identidades do utilizador é a identidade pública. A identidade pública permite identificar um utilizador podendo ter uma ou mais identidades públicas. Esta(s) identidade(s) são utilizadas por qualquer utilizador para solicitar uma comunicação com outros utilizadores. Por exemplo, poderá ser incluída num cartão de visita.

O IMS trás um conceito muito interessante: conjunto de identidades públicas implicitamente registadas. Numa mensagem SIP normal, cada identidade que precisa de se registar necessita de enviar uma mensagem SIP REGISTER. No sistema IMS, é possível registar várias identidades públicas em apenas uma mensagem, salvando tempo e largura de banda.

Características:

- Uma identidade pública deverá ter a forma de um SIP URL [25] e [38] ou de um TEL URL [39].
- Uma aplicação ISIM deverá armazenar, de uma forma segura, pelo menos uma identidade pública de utilizador (não deverá ser possível para um utilizador alterar a identidade pública), mas não é obrigatório que todas as identidades adicionais se encontrem armazenadas na aplicação ISIM.
- Uma identidade pública deverá ser registada implícita ou explicitamente antes do utilizador a puder utilizar para sessões IMS.
- Identidades públicas do utilizador não são autenticadas pela rede durante os registos.
- As identidades públicas do utilizador poderão ser utilizadas para identificar a informação do utilizador dentro do HSS.
- Caso o UICC não contiver uma aplicação ISIM, então uma identidade pública temporária deverá ser derivada do IMSI do USIM

Identidade Privada do Utilizador

[18] Baseado na *release* 6 do IMS

Todos os utilizadores IMS terão de estar associados a uma ou mais identidades privada do utilizador (Figura 82). Esta identidade é atribuída pelo operador e é utilizada, por exemplo, para Registos, Autorização, Autenticação, Administração e Taxação. Esta identidade tem o formato NAI (*Network Access Identifies*, definido em [43]). O formato de um NAI é "utilizador@operador.com".

Uma identidade privada funciona no domínio IMS de uma forma similar ao IMSI (*International Mobile Subscriber Identifier*) em GSM. Tal como o IMSI, que se encontra armazenado no USIM, a identidade privada não precisa de ser conhecida pelo utilizador, estando esta armazenada no ISIM.

Características:

- A identidade privada do utilizador não é utilizada para reencaminhamento de mensagens SIP.
- A identidade privada deverá estar presente em todos os pedidos de REGISTER.
- Uma aplicação ISIM deverá, de uma forma segura, armazenar a identidade privada. Não deverá ser possível para o utilizador modificar esta identidade.
- A identidade privada é uma identidade única definida pelo operador, que poderá ser utilizada pelo próprio operador, para identificar, de uma forma única, o utilizador numa perspectiva de rede.
- A identidade privada deverá ser "alocada" permanentemente a um utilizador (não é uma identidade dinâmica) e é válida para a duração da subscrição do utilizador.
- A identidade privada é utilizada para identificar a informação do utilizador (por exemplo, informação de autenticação) armazenada no HSS (usada, por exemplo, no processo de registo).
- A identidade privada poderá estar presente em registos de taxação.
- A identidade privada identifica uma subscrição, não o utilizador.
- A identidade privada é autenticada durante o registo de um utilizador (inclui re-registo e fim de registo).
- O HSS precisa de armazenar a identidade privada do utilizador.
- O S-CSCF precisa de obter e armazenar a identidade privada durante o registo e *unregistered termination*.
- Se o UICC não contiver uma aplicação ISIM, então a identidade privada do utilizador deverá ser derivada do IMSI do USIM, que permitirá identificar de uma forma única o utilizador dentro do operador.

Relação da identidade privada e pública do utilizador

O operador é responsável por atribuir a identidades privadas do utilizador e as identidades públicas;

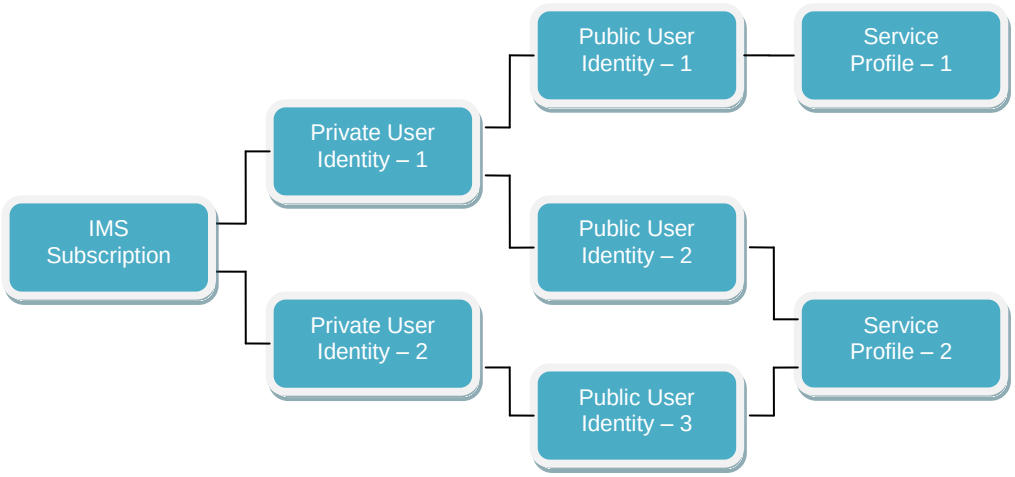

Figura 82 – Esquema com a relação das várias identidades

O *service profile* é independente do registo implícito, isto é, identidades públicas com diferentes *service profiles* diferentes poderão pertencer ao mesmo grupo de identidades implícitas. Cada utilizador apenas pode estar associado a um service profile, mas um service profile pode estar associado a mais do que uma identidade pública do utilizador.

IMPORTANTE: Todos os *service profile* ou identidades públicas que partilhem a mesma identidade privada estão associados ao mesmo S-CSCF.