



**Vitor António das
Neves Pinto**

Plataforma de serviços residenciais



**Vitor António das
Neves Pinto**

Plataforma de serviços residenciais

Dissertação apresentada à Universidade de Aveiro para cumprimento dos requisitos necessários à obtenção do grau de Mestre em Engenharia Electrónica e Telecomunicações, realizada sob a orientação científica do Prof. Doutor Francisco Fontes, professor auxiliar convidado do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro e do Prof. Doutor Rui Aguiar, professor auxiliar do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro.

o júri

presidente

Prof. Doutor João Matos

Professor associado do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

Prof. Doutor Renato Nunes

Professor auxiliar do Departamento de Engenharia Informática do Instituto Superior Técnico da Universidade Técnica de Lisboa

Prof. Doutor Francisco Fontes

Professor auxiliar convidado do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

Prof. Doutor Rui Aguiar

Professor auxiliar do Departamento de Electrónica, Telecomunicações e Informática da Universidade de Aveiro

agradecimentos

Ao Professor Francisco Fontes, por estar sempre disponível para esclarecer as dúvidas existentes e partilhar os seus conhecimentos. Revelou-se importante o espírito crítico demonstrado pelo mesmo na procura de mais e melhores soluções para os problemas encontrados.

Ao Professor Rui Aguiar, pela disponibilidade demonstrada em partilhar os seus conhecimentos e fornecer informação importante para o desenvolvimento da dissertação.

Ao Engenheiro Vitor Ribeiro, por ter facultado as condições necessárias ao desenvolvimento desta dissertação, bem como pela partilha do seu conhecimento.

Ao Instituto de Telecomunicações de Aveiro e seus colaboradores por me terem oferecido todas as condições e apoio necessários para o desenvolvimento desta dissertação.

À minha namorada, Luísa Silva, e à minha família pelo incansável apoio, paciência e motivação que sempre me deram durante o desenvolvimento da dissertação.

palavras-chave

OSGi, gateway de serviços, serviços residenciais, gateway residencial, e-Care, CWMP, TR-069, rede doméstica, rede de acesso.

resumo

O serviço de acesso à Internet tem sido, até agora, o principal catalisador para a difusão das chamadas ligações de banda larga junto do mercado residencial. O desejo de partilhar este serviço entre vários terminais da mesma residência, aliado a factores como a maturidade do protocolo IP, o aparecimento de tecnologias de rede que permitem a comunicação entre dispositivos em espaços reduzidos, a generalização de conteúdos multimédia em formato digital, entre outros, tem impulsionado o aparecimento das chamadas redes residenciais. Para além de interligarem entre si os mais variados tipos de dispositivos domésticos, estas redes asseguram ainda a comunicação destes mesmos dispositivos com entidades externas à residência. Existe, portanto agora a oportunidade para operadores de telecomunicações e provedores de serviços utilizarem a largura de banda disponível para disponibilizarem, através de uma única plataforma, toda uma nova geração de serviços a clientes residenciais. No entanto, garantir o correcto comportamento de todos estes novos serviços, lidar com as particularidades do ambiente residencial e do próprio utilizador residencial, impõe a operadores e provedores de serviços novos desafios.

Um dos principais desafios prende-se com a capacidade das actuais redes de acesso e domésticas para o transporte de vários tipos de serviços. Desta forma, na presente dissertação, são analisados cenários típicos de prestação de serviços residenciais, com ênfase nas infra estruturas de rede que suportam essa mesma prestação de serviços. Nesta, são também caracterizadas infra-estruturas de rede capazes de suportar a distribuição de um leque alargado de serviços residenciais, garantindo o correcto comportamento dos mesmos. A distribuição de vários tipos de serviços sobre infra-estruturas de rede partilhadas, requer também soluções que contemplem a configuração/gestão remota dos vários tipos de serviços e de mecanismos que possibilitem a auto configuração de dispositivos que residem na rede doméstica. Neste contexto a plataforma de serviços OSGi é apresentada. Por forma a concluir-se acerca da sua validade e aplicabilidade em cenários futuros de prestação de serviços residenciais, foi implementado um serviço de monitorização médica remota sobre a mesma. Esta, permitiu principalmente avaliar vantagens e desvantagens da plataforma OSGi, bem como indicar possíveis soluções que colmatam as desvantagens encontradas.

Esta dissertação apresenta portanto possíveis soluções para desafios com que operadores e prestadores de serviços se deparam em cenários de prestação de múltiplos tipos de serviços residenciais.

keywords

OSGi, services gateway, home services, home gateway, residential gateway, e-Care, CWMP, TR-069, home network, access network.

abstract

Internet access service has been, until now, the main reason to the current generalization of the so called broadband connections in the residential market. The will of sharing this same service between different terminals in the same house, together with other factors like the maturity of the IP protocol, the eruption of network technologies that allow the communication between devices in restricted environments, the generalization of multimedia contents distributed in a digital format, among others, has driven to the appearance of the so called residential networks. Besides providing the interconnection between the several types of residential devices, these networks also assure the communication of those same devices with entities outside the home. So this is the opportunity for telecommunications operators and service providers to use the available bandwidth to provide, using a single service platform, new kinds of services to residential costumers. Although, to assure the correct behaviour of that multitude of services, to deal with the uniqueness of the residential environment and of the residential user himself, challenges must be solved by operators and service providers.

One of the main challenges as to do with the ability of today's access and residential networks to transport different kinds of services. This way, in the present thesis, typical residential service providing scenarios are analyzed, with special detail on the network infra structures used on that same scenarios. Moreover, network infra structures that support the distribution of different kinds of residential services are also characterized, always considering that the expected behaviour of those same services is assured. The distribution of different types of services over shared networks, also requires solutions regarding the configuration/remote management of those same services, as well mechanisms that allow the auto configuration of domestic devices. In this context, the OSGi service platform is presented. In order to conclude about it's validity and applicability in future service providing scenarios, a remote medical monitoring service was implemented over OSGi. This implementation allowed mainly to judge the advantages and disadvantages of the OSGi platform, as well pointing some possible solutions that somehow minimize the disadvantages.

Therefore, this thesis presents possible solutions for challenges that operators and service providers will face in scenarios where is intended to provide to residential costumers a multitude of kinds of services.

Índice

Índice de figuras	17
Índice de tabelas	19
Acrónimos	21
Capítulo 1 – Introdução	29
1.1. Objectivos.....	32
1.2. Abordagem.....	32
1.3. Estrutura da dissertação.....	34
1.4. Trabalho preliminar.....	34
Capítulo 2 – Estruturas de rede de suporte a serviços	37
2.1. Modelo de referência de rede.....	38
2.1.1. Modelo de referência de rede completo.....	38
2.1.2. Modelo de referência da rede doméstica.....	40
2.2. Arquitecturas de rede e serviços típicos.....	44
2.2.3. Redes de acesso.....	44
2.2.3.1. Redes de acesso baseadas em tecnologias xDSL.....	44
2.2.3.2. Redes de acesso CATV.....	52
2.2.3.3. Redes de acesso baseadas em Ethernet.....	63
2.2.3.4. Outras redes de acesso.....	70
2.2.4. Redes residenciais.....	71
2.2.4.1. 10 Base-T, 100 Base-T.....	72
2.2.4.2. IEEE 802.11 a/b/g.....	72
2.2.4.3. HomePNA.....	73
2.2.4.4. HomePlug.....	74
2.3. Observações finais.....	75
Capítulo 3 – Plataformas de serviços residenciais	77
3.1. Serviços residenciais em redes de próxima geração.....	78
3.1.1. Serviços residenciais futuros.....	78
3.1.1.1. Telefonia.....	78
3.1.1.2. Aplicações baseadas em TV.....	78
3.1.1.2.1. Difusão digital de canais de TV e áudio.....	79
3.1.1.2.2. Vídeo/áudio on demand.....	79
3.1.1.2.3. Personal video recording.....	80

3.1.1.2.4	TV interactiva.....	80
3.1.1.3.	Serviços baseados em PC.....	80
3.1.1.3.1	Vídeo telefonia.....	81
3.1.1.3.2	Jogos multi jogador on-line.....	81
3.1.1.4.	Domótica.....	81
3.1.1.5.	Aplicações médicas.....	81
3.1.1.5.1	Serviços de diagnósticos remotos.....	82
3.1.1.5.2	Serviços de monitorização remota.....	82
3.1.1.5.3	Serviços de assistência médica remota.....	82
3.1.1.6.	Algumas considerações sobre serviços residenciais actuais e futuros.....	83
3.1.2.	Caracterização de redes de acesso de nova geração.....	83
3.1.3.	Caracterização de redes domésticas de nova geração.....	86
3.2.	Plataformas e gateways de serviços residenciais.....	89
3.2.1.	Plataformas de serviços áudio/vídeo.....	89
3.2.1.1.	Multimedia Home Platform.....	90
3.2.2.	Motivação e requisitos de uma plataforma de serviços residenciais.....	93
3.2.3.	A aliança OSGi.....	94
3.2.3.1.	Modelo OSGi do gateway de serviços residencial.....	95
3.2.3.2.	A plataforma OSGi.....	97
3.2.3.2.1	A framework OSGi.....	98
3.2.3.2.1.1	Ambiente de execução.....	99
3.2.3.2.1.2	Camada de modularização.....	100
3.2.3.2.1.3	Camada de gestão do ciclo de vida dos serviços.....	101
3.2.3.2.1.4	Registo de serviços.....	103
3.2.3.2.1.5	Camada de segurança.....	105
3.2.3.3.	Gestão remota da plataforma OSGi.....	106
3.2.3.4.	Serviços normalizados OSGi.....	107
3.2.3.4.1	Serviços da framework.....	108
3.2.3.4.2	Serviços de sistema.....	108
3.2.3.4.3	Serviços protocolares.....	109
3.2.3.4.4	Outros serviços.....	109
3.2.4.	A plataforma TR-069: CPE WAN Management Protocol.....	110
3.3.	Observações finais.....	115
Capítulo 4 – Validação de conceitos.....		117

4.1.	O projecto MUSE	117
4.1.1.	Plataformas e gateways de serviços residenciais no projecto MUSE.....	120
4.2.	Serviço de monitorização médica remota	121
4.2.1.	Enquadramento e descrição	121
4.2.2.	Opções e requisitos de implementação	123
4.2.2.1.	A implementação da framework OSGi OSCAR.....	124
4.2.2.2.	Bundles Base.....	125
4.2.2.3.	Outros bundles utilizados	125
4.2.2.4.	A especificação OSGi Device Access	127
4.2.3.	Módulos desenvolvidos	130
4.2.3.1.	Bundles OSGi.....	130
4.2.3.2.	Emulador de equipamento médico	140
4.2.3.3.	Gestor remoto da plataforma de serviços	141
4.2.3.4.	Servlet de submissão de dados médicos	144
4.2.4.	Validação dos módulos implementados.....	144
4.3.	Outros serviços e a plataforma OSGi.....	149
4.4.	Observações finais	150
	Capítulo 5 – Conclusões	153
	Referências	157
	Anexo I - Protocolos tributários da tecnologia ATM em ambientes xDSL	167
	Anexo II – Tabelas	169
	Anexo III – Implementações da plataforma OSGi	173

Índice de figuras

Figura 1: Modelo de referência de rede completo [16].....	38
Figura 2: Modelo da arquitectura de uma rede residencial [16].....	40
Figura 3: Configurações de redes residenciais [16].....	43
Figura 4: Espectro de frequências POTS e ADSL.....	45
Figura 5: Arquitectura de uma rede de acesso que usa a tecnologia ADSL na primeira milha .	46
Figura 6: Rede de acesso baseada na tecnologia xDSL a operar no modo PPPoE	49
Figura 7: Comparação dos débitos de transmissão obtidos para as tecnologias ADSL, ADSL2, READSL2 e ADSL2+ [65]	51
Figura 8: Arquitectura de uma rede de acesso CATV	56
Figura 9: Pilhas protocolares dos elementos de uma rede DOCSIS.....	60
Figura 10: Arquitectura de uma rede de acesso baseada em <i>Ethernet</i>	64
Figura 11: Pilhas protocolares dos elementos de uma rede de acesso baseada em <i>Ethernet</i>	68
Figura 12: Normas <i>Ethernet</i> [95].....	69
Figura 13: Partilha do serviço de acesso à Internet de banda larga. Baseado na figura 13 de [16]	71
Figura 14: Espectro das tecnologias POTS, RDIS, xDSL e HPNA 1.0, 2.0 e 3.0 [35].	73
Figura 15: Composição de um <i>gateway</i> residencial [16]	87
Figura 16: Rede doméstica multi-serviço. Baseado na figura 17 de [16].....	87
Figura 17: Entidades envolvidas na prestação de serviços através da plataforma MHP [106].	91
Figura 18: Arquitectura MHP [108].....	91
Figura 19: Possível modelo de um <i>gateway</i> de serviços residencial (OSGi) [112].....	96
Figura 20: Cooperação entre serviços e <i>bundles</i> [116].....	98
Figura 21: Camadas da <i>framework</i> OSGi [109].....	99
Figura 22: Modelo de delegação entre <i>class loaders</i> da <i>framework</i> OSGi [113].	101
Figura 23: Diagrama de estados de um <i>bundle</i> [113].....	101
Figura 24: Partilha de serviços entre <i>bundles</i> na plataforma OSGi [123]	104
Figura 25: Modelo de gestão remota definido nas especificações OSGi [112].....	106
Figura 26: Enquadramento do protocolo CWMP num ambiente xDSL [131]	111
Figura 27: Troca de mensagens entre um CPE e um ACS [131].....	113
Figura 28: Diferentes elementos da rede de acesso e agregação endereçados no projecto MUSE [9].	118
Figura 29: Organização interna do projecto MUSE, fase II [140].....	119

Figura 30: Interações entre as entidades envolvidas no serviço de monitorização médica remota.....	122
Figura 31: <i>Bundles</i> instalados por defeito na implementação OSGi OSCAR	125
Figura 32: Arquitectura JMX.....	126
Figura 33: Processo de refinamento de um <i>Device Service</i> [116]	129
Figura 34: Diagrama de classes e estrutura do <i>bundle Serial Service</i>	131
Figura 35: Interface <i>SerialService</i>	132
Figura 36: Diagrama de classes e estrutura do <i>bundle RXTXComm</i>	133
Figura 37: Diagrama de classes e estrutura do <i>bundle Driver Locator</i>	134
Figura 38: Diagrama de classes e estrutura do <i>bundle MedDrS</i>	136
Figura 39: Interface <i>MedService</i>	137
Figura 40: Diagrama de classes e estrutura do <i>bundle Medical Data Submitter</i>	138
Figura 41: Dependências entre os <i>bundles</i> intervenientes no serviço de monitorização médica remota.	139
Figura 42: GUI do emulador de equipamento médico	140
Figura 43: Formato da trama utilizada para envio de dados através da porta série	140
Figura 44: Interface <i>web</i> do gestor remoto	142
Figura 45: Diagrama de classes do gestor remoto da plataforma de serviços.....	143
Figura 46: <i>Bundles</i> activos na plataforma de serviços por defeito	145
Figura 47: Eventos na sequência da instalação do serviço de monitorização médica remota.	145
Figura 48: Registo do <i>SerialService</i> após a execução do emulador de equipamento médico	146
Figura 49: Processo de refinamento do <i>Device Service SerialService</i>	147
Figura 50: <i>Bundles</i> activos na <i>framework</i> concluído o processo de refinamento do DS <i>SerialService</i>	148
Figura 51: Armazenamento de valores médicos recebidos, submissão e visualização dos mesmos na base de dados.....	148
Figura 52: <i>Bundles</i> activos na plataforma de serviços após término do emulador de equipamento médico	149
Figura 53: Serviço BarkIDS [157].....	150
Figura 54: Estrutura em árvore de protocolos tributários da tecnologia ATM e possíveis encapsulamentos [160]	167

Índice de tabelas

Tabela 1: Taxas de transmissão obtidas com a tecnologia VDSL em função do comprimento do par de cobre [66].....	52
Tabela 2: Características da camada física das normas (Euro)DOCSIS 1.0 e 1.1.....	54
Tabela 3: Características da camada física da norma (Euro)DOCSIS 2.0	55
Tabela 4: Descrição da pilha protocolar usada pelo protocolo CWMP [131]	112
Tabela 5: Atributos de serviço EVC [91].	169
Tabela 6: Atributos de serviço UNI [91]	170
Tabela 7: Taxa/Alcance de tecnologias usadas em redes residenciais. Baseado na tabela 5 de [99] e [165].....	171

Acrónimos

AAA	<i>Authentication, Authorization, and Accounting</i>
AAL 5	<i>ATM Adaptation Layer 5</i>
ACS	<i>Auto-Configuration Server</i>
ADSL	<i>Asymmetric Digital Subscriber Line</i>
AES	<i>Advanced Encryption Algorithm</i>
ALG	<i>Application Layer Gateway</i>
ANACOM	<i>Autoridade Nacional de Comunicações</i>
API	<i>Application Programming Interface</i>
ARP	<i>Address Resolution Protocol</i>
ATA	<i>Analogue Terminal Adapter</i>
ATDMA	<i>Advanced Time Division Multiple Access</i>
ATM	<i>Asynchronous Transfer Mode</i>
ATM 25	<i>ATM 25.6-Mbps</i>
ATU-C	<i>ADSL Terminal Unit – Central office</i>
ATU-R	<i>ADSL Terminal Unit – Remote</i>
B-NT	<i>Broadband-Network Termination</i>
BRAS	<i>Broadband Remote Access Server</i>
CAC	<i>Call Admission Control</i>
CATV	<i>Community Access Television</i>
CDC	<i>Connected Device Configuration</i>
CE	<i>Customer Edge</i>
CE-R	<i>Customer Edge – Router</i>
CE-S	<i>Customer Edge – Switch</i>
CHAP	<i>Challenge Handshake Authentication Protocol</i>
CM	<i>Cable Modem</i>
CMS	<i>Call Management Server</i>
CMTS	<i>Cable Modem Termination System</i>
CPE	<i>Customer Premises Equipment</i>

CPN	<i>Customer Premises Network</i>
CSMA/CA	<i>Carrier Sense Multiple Access/Collision Avoidance</i>
CSMA/CD	<i>Carrier Sense Multiple Access/Collision Detection</i>
CWMP	<i>CPE WAN Management Protocol</i>
DA	<i>Device Access</i>
DHCP	<i>Dynamic Host Configuration Protocol</i>
DIX	<i>Digital Intel Xerox</i>
DLS	<i>Driver Locator Service</i>
DM	<i>Device Manager</i>
DNS	<i>Domain Name System</i>
DOCSIS	<i>Data Over Cable Service Interface Specification</i>
DrS	<i>Driver Service</i>
DS	<i>Device Service</i>
DSL	<i>Digital Subscriber Line</i>
DSLAM	<i>Digital Subscriber Line Access Multiplexer</i>
DVB	<i>Digital Video Broadcasting</i>
DVB-C	<i>Digital Video Broadcasting-Cable</i>
DVB-S	<i>Digital Video Broadcasting- Satellite</i>
DVB-T	<i>Digital Video Broadcasting-Terrestrial</i>
DVB-MC	<i>Digital Video Broadcasting-</i>
E/O	<i>Electro/Óptico</i>
E-LMI	<i>Ethernet – Local Management Interface</i>
E-MTA	<i>Embedded – Multimedia Terminal Adapter</i>
EFMC	<i>Ethernet in the First Mile over Copper</i>
EFMF	<i>Ethernet in the First Mile over Fibber</i>
EMS	<i>Element Management System</i>
EPON	<i>Ethernet Passive Optical Network</i>
ETSI	<i>European Telecommunications Standards Institute</i>
EUA	<i>Estados Unidos a América</i>

EuroDOCSIS	<i>European DOCSIS</i>
EVC	<i>Ethernet Virtual Connection</i>
F.O.	<i>Fibra Óptica</i>
FEC	<i>Forward Error Correction</i>
FMC	<i>Fixed Mobile Convergence</i>
FR-LMI	<i>Frame Relay – Local Management Interface</i>
FTP	<i>File Transfer Protocol</i>
FTTB	<i>Fibber To The Building</i>
FTTC	<i>Fibber To The Curb</i>
FTTCab	<i>Fibber To The Cabinet</i>
GSB	<i>Global System for Broadband</i>
HDSL	<i>High-bit-rate Digital Subscriber Line</i>
HDTV	<i>High Definition TV</i>
HFC	<i>Hybrid Fibber Coaxial</i>
HGI	<i>Home Gateway Initiative</i>
HTML	<i>Hyper Text Markup Language</i>
HomePNA	<i>Home Phoneline Networking Alliance</i>
HTTP	<i>HyperText Transfer Protocol</i>
HTTPS	<i>HyperText Transfer Protocol Secure</i>
IAD	<i>Integrated Access Device</i>
IEEE	<i>Institute of Electrical and Electronics Engineers</i>
IETF	<i>Internet Engineering Task Force</i>
IGD	<i>Internet Gateway Device</i>
ILMI	<i>Integrated Local Management Interface</i>
IM	<i>Instant Messaging</i>
IP	<i>Internet Protocol</i>
IPCP	<i>IP Control Protocol</i>
IPoA	<i>Internet Protocol over ATM</i>
IPTV	<i>Internet Protocol TeleVision</i>

IrDa	<i>Infrared Data association</i>
ISDN	<i>Integrated Service Digital Network</i>
IST	<i>Information Society Technologies</i>
ITU	<i>International Telecommunications Union</i>
ITU-T	<i>International Telecommunications Union - Telecommunications standardization sector</i>
J2ME	<i>Java2 Micro Edition</i>
J2SE	<i>Java2 Standard Edition</i>
JAR	<i>Java ARchive</i>
JDK	<i>Java Development Kit</i>
JMX	<i>Java Management eXtensions</i>
JVM	<i>Java Virtual Machine</i>
LAN	<i>Local Area Network</i>
LCP	<i>Link Control Protocol</i>
LLC	<i>Logical Link Control</i>
LSP	<i>Label Switched Path</i>
MAC	<i>Medium Access Control</i>
MGCP	<i>Media Gateway Control Protocol</i>
MEF	<i>Metro Ethernet Forum</i>
MHP	<i>Multimedia Home Platform</i>
MIB	<i>Management Information Base</i>
MIC	<i>Message Integrity Check</i>
MMS	<i>Multimedia Messaging System</i>
MPEG	<i>Moving Picture Experts Group</i>
MPEG-2	<i>Moving Picture Experts Group - 2</i>
MTA	<i>Multimedia Terminal Adapter</i>
MTU	<i>Maximum Transmission Unit</i>
MUSE	<i>Multi Service Access Everywhere</i>
MPLS	<i>Multi-Protocol Label Switching</i>
N-PE	<i>Network-Provider Edge</i>

NAT	<i>Network Address Translation</i>
NAPT	<i>Network Address Port Translation</i>
NGN	<i>Next Generation Networks</i>
NMS	<i>Network Management System</i>
NCS	<i>Network-based Call Signalling</i>
NT1	<i>Network Termination 1</i>
NT2	<i>Network Termination 2</i>
NTSC	<i>National Television Standards Committee</i>
OBR	<i>Oscar Bundle Repository</i>
OPEX	<i>Operational EXpenditure</i>
OSGi	<i>Open Services Gateway initiative</i>
OSI	<i>Open Systems Interconnection</i>
P2P	<i>Peer-to-Peer</i>
PAL	<i>Phase Alternating Line</i>
PAP	<i>Password Authentication Protocol</i>
PAT	<i>Port Address Translation</i>
PC	<i>Personal Computer</i>
PDA	<i>Personal Digital Assistant</i>
PLC	<i>Power Line Communications</i>
PHY	<i>PHYsical layer</i>
PPM	<i>Pulse Position Modulation</i>
PPP	<i>Point-to-Point Protocol</i>
PPPoA	<i>Point-to-Point Protocol Over ATM</i>
PPPoE	<i>Point-to-Point Protocol Over Ethernet</i>
POP3	<i>Post Office Protocol 3</i>
POTS	<i>Plain Old Telephone Service</i>
PSTN	<i>Public Switched Telephone Network</i>
PVC	<i>Permanent Virtual Connection</i>
PVR	<i>Personal Video Recording</i>

QoS	<i>Qualidade de Serviço</i>
QAM	<i>Quadrature Amplitude Modulation</i>
QPSK	<i>Quadrature Phase Shift Keying</i>
REDIS	<i>Rede Digital com Integração de Serviços</i>
READSL	<i>Reach Extended Asymmetric Digital Subscriber Line</i>
RFC	<i>Request For Comments</i>
RMI	<i>Remote Method Invocation</i>
RPC	<i>Remote Procedure Invocation</i>
RTP	<i>Real-time Transport Protocol</i>
RTSP	<i>Real Time Streaming Protocol</i>
C-VLAN	<i>Client – Virtual Local Area Network</i>
S-VLAN	<i>Stacked – Virtual Local Area Network</i>
SCART	<i>Syndicat des Constructeurs d'Appareils Radiorécepteurs et Téléviseurs</i>
SCDMA	<i>Synchronous Code Division Multiplex Access</i>
SDM	<i>Service Deployment Manager</i>
SDTV	<i>Standard Definition TV</i>
SHDSL	<i>Single-Pair High-Bit-Rate Digital Subscriber Line</i>
SIP	<i>Session Initiation Protocol</i>
SMS	<i>Service Management System ; Short Message Service</i>
SMTP	<i>Simple Mail Transfer Protocol</i>
SNAP	<i>Sub Network Access Protocol</i>
SNMP	<i>Simple Network Management Protocol</i>
SO	<i>Sistema Operativo</i>
SOA	<i>Service Oriented Architecture</i>
SOHO	<i>Small Office/Home Office</i>
SONET	<i>Synchronous Optical Network</i>
SP	<i>Sub Project</i>
SSL	<i>Secure Socket Layer</i>
ST	<i>Service Termination</i>

STB	<i>Set Top Box</i>
STP	<i>Spanning Tree Protocol</i>
STM-1	<i>Synchronous Transfer Mode 1</i>
TA	<i>Terminal Adaptation</i>
TCM	<i>Trellis Coded Modulation</i>
TCP	<i>Transmission Control Protocol</i>
TDMA	<i>Time Division Multiple Access</i>
TF	<i>Task Force</i>
TFTP	<i>Trivial File Transfer Protocol</i>
TISPAN	<i>Telecoms and Internet converged Services and Protocols for Advanced Networks</i>
TLS	<i>Transport Layer Security</i>
ToD	<i>Time of Day</i>
TUI	<i>Text User Interface</i>
TV	<i>TeleVision</i>
U-PE	<i>User – Provider Edge</i>
UDP	<i>User Datagram Protocol</i>
UMTS	<i>Universal Mobile Telecommunications System</i>
UNI	<i>User Network Interface</i>
URL	<i>Universal Resource Locator</i>
USB	<i>Universal Serial Bus</i>
UTP	<i>Unshielded Twisted Pair</i>
VC	<i>Virtual Circuit</i>
VCI	<i>Virtual Channel Identifier</i>
VDSL	<i>Very-high-bit-rate Digital Subscriber Line</i>
VID	<i>VLAN Identifier</i>
VLAN	<i>Virtual Local Area Network</i>
VoD	<i>Video on Demand</i>
VoIP	<i>Voice over Internet Protocol</i>
VPI	<i>Virtual Path Identifier</i>

VPLS	<i>Virtual Private LAN Service</i>
VPN	<i>Virtual Private Network</i>
WAN	<i>Wide Area Network</i>
WAR	<i>Web ARchive</i>
WP	<i>Work Package</i>
xDSL	<i>x Digital Subscriber Line</i>
XL PON	<i>eXtra Large Passive Optical Network</i>
XML	<i>Extensible Markup Language</i>

Capítulo 1 – Introdução

A autoridade nacional de comunicações portuguesa (ANACOM [1]), responsável pela regulação das comunicações postais e electrónicas em Portugal, reportou [2] no terceiro trimestre de 2006 um aumento de 24,1 % no número de clientes que subscrevem actualmente um serviço de acesso à Internet de banda larga fixo, crescimento esse relativo ao mesmo trimestre de 2005. Com este aumento, o número total de clientes que actualmente usufruem do referido serviço ronda os 1.400.000, o que se traduz numa taxa de penetração (n.º de clientes por cada 100 habitantes) de 13.3%. Esta taxa de crescimento específica para Portugal, não é mais do que um reflexo do que se verifica no resto da Europa e do Mundo, no que diz respeito à adesão de serviços de acesso à Internet em banda larga.

A generalização do serviço de acesso à Internet em banda larga proporcionou ao utilizador uma nova experiência e possibilitou-lhe, através do seu computador pessoal, o acesso a novos serviços como *email* e *web browsing*, bem como a serviços *on-line* como *home banking* ou *e-shopping*. Contudo a maturidade do protocolo IP, o aparecimento de tecnologias de rede que permitem a comunicação entre dispositivos em espaços reduzidos (por exemplo *WiFi*, *Bluetooth*, entre outras), o facto de os conteúdos multimédia (música, vídeo, fotografia) se apresentarem cada vez mais num formato digital (conteúdos esses que o utilizador deseja partilhar e usufruir em dispositivos mais amigáveis que o computador pessoal tais como a televisão ou o sistema de som) e a necessidade de partilhar a ligação de banda larga entre vários dispositivos (por exemplo, computadores pessoais, leitores de MP3, consolas de jogos, entre outros) impulsionaram o aparecimento das primeiras redes residenciais. Estas não são mais que redes

de comunicações heterogêneas que permitem, por um lado a interligação de vários tipos de dispositivos domésticos e por outro a comunicação destes com entidades externas à residência (através de um mesmo acesso de banda larga). Assim, a presença de uma rede doméstica facilita o acesso por parte do fornecedor de serviços a outros dispositivos domésticos para além do computador pessoal, dispositivos esses que podem ser usados para disponibilizar ao utilizador novos tipos de serviços residenciais. Para além do tradicional serviço de acesso à Internet, o utilizador passará a ter ao seu alcance um conjunto de novos serviços residenciais que inclui serviços como telefonia IP, vídeo-conferência, IPTV, *video on demand*, música *on demand*, *e-care*, domótica, tele-vigilância, entre muitos outros, sendo que todos eles serão distribuídos sobre uma mesma rede doméstica e de acesso, dando assim origem a conceitos como o *triple-play*¹ e *multi-play*².

Contudo, para que o utilizador tenha uma utilização satisfatória dos serviços, há requisitos qualitativos exigidos por este que têm de ser preenchidos, de onde se pode destacar a facilidade de utilização, integridade, disponibilidade e correcção. O cumprimento destes requisitos é tipicamente conseguido através da implementação de uma ou mais funcionalidades específicas nas redes de transporte, acesso e doméstica. Assim, no que diz respeito às redes domésticas, várias funcionalidades devem ser suportadas para que os referidos requisitos sejam cumpridos. Entre elas estão funcionalidades que permitam a distribuição simultânea de vários tipos de serviços (possivelmente provenientes de provedores de serviços distintos) sem que haja interferências entre estes, que garantam a integração na rede doméstica de dispositivos não compatíveis com as tecnologias de rede usadas (por exemplo TV, sistema de som, entre outros) por forma a maximizar o número de serviços possíveis e a facilitar a interface entre o serviço e o utilizador, que possibilitem a interacção entre esses serviços e que minimizem ou dispensem o utilizador de quaisquer tarefas de gestão e/ou configuração.

Da mesma forma, as redes de acesso e transporte têm também de sofrer algumas inovações visto que no passado estas eram dimensionadas apenas com o objectivo de fazer o transporte indiferenciado de dados. Para além disso, tipicamente diferentes tipos de serviço eram

¹ Disponibilidade de serviços de áudio (ex. telefonia), vídeo (ex. serviço de TV) e dados (ex. acesso à Internet), como três serviços totalmente independentes, sobre uma única rede de acesso de banda larga.

² Combinação de serviços de voz, vídeo e dados para a construção de novos serviços mais complexos (ex. TV interactiva). Ao contrário do que acontece com o *triple-play*, os serviços não são terminados em terminais específicos e dedicados a cada serviço, ou seja, vários terminais podem receber um mesmo serviço.

entregues ao cliente através de redes específicas para o efeito. Assim, actualmente estas redes estão a ser substituídas ou a sofrer melhoramentos que permitam a disponibilização de vários tipos de serviço (não só serviços que anteriormente eram transportados sobre redes específicas, mas também novos serviços que agora começam a surgir) sobre uma única rede. Com uma rede única para vários serviços é possível não só oferecer ao cliente a hipótese de usufruir num mesmo acesso à rede de um pacote de serviços mais rico, mas também baixar o preço por serviço, já que o custo de implementação, operação e manutenção dessa mesma rede passa a ser partilhado pelos vários intervenientes na cadeia de prestação de serviços. Esta nova geração de redes passará a estar dotada de funções que permitirão entre outros o transporte de uma larga gama de serviços sobre uma única rede, o acesso incondicional e em qualquer local por parte dos utilizadores a vários provedores de serviços sem que haja deterioração na qualidade do serviço (nomadismo), a auto-configuração de equipamentos que se situam não só na própria rede do operador, mas também nas instalações do cliente e o aprovisionamento e gestão de recursos.

Fundamental para que diferentes tipos de serviço sejam fornecidos sobre uma mesma rede (de transporte, acesso ou doméstica) é a implementação de mecanismos que permitam tornar os serviços (ou seja, as aplicações que implementam os serviços) totalmente agnósticos à rede que utilizam. Esta característica, própria das chamadas redes de próxima geração (RPG) ou NGN (*Next Generation Networks*), permitirá a criação de interfaces normalizadas para o acesso aos recursos disponibilizados pela rede, interfaces essas que, sendo totalmente abertas e estando acessíveis a todos, permitirão a criação de uma plataforma de serviços comum sobre a qual qualquer provedor de serviços e/ou aplicações com um mínimo de conhecimento das interfaces fornecidas pela plataforma, pode disponibilizar facilmente e rapidamente os seus serviços.

Organizações como o DSL Forum [3], HGI [4], o ETSI [5] e o ITU-T [6] através de actividades como o TISPAN [7] e o NGN [8] e também projectos de investigação e desenvolvimento no âmbito do IST, como o MUSE [9], têm por objectivo abordar algumas das questões levantadas ao longo desta introdução.

É, então, inegável que uma plataforma unificada não trará apenas vantagens aos provedores de serviços mas sim a todos os intervenientes na cadeia de valores da prestação de um determinado serviço, ou seja, aos provedores de conteúdos, aos provedores de serviços, aos operadores de rede e ao cliente. O provedor de serviços passa a fazer chegar os seus serviços a

mais clientes sem que para isso tenha de desenvolver várias vertentes de um mesmo serviço; o operador de rede vê o investimento na migração das suas redes ser compensado bem como a partilha dos custos de operação e manutenção; o cliente tem acesso a um maior leque de serviços a um preço menor. Estamos portanto perante uma situação em que todos os intervenientes têm algo a ganhar.

1.1. Objectivos

Com a presente dissertação pretendeu-se atingir os seguintes objectivos:

- Adquirir conhecimento sobre o estado da arte no que diz respeito a redes de acesso e domésticas utilizadas na prestação de serviços residenciais.
- Caracterizar estas redes de acesso e domésticas, no que diz respeito à tecnologia e capacidades de auto-configuração, autenticação, autorização, taxaço, entre outros.
- Determinar serviços residenciais típicos, bem como possíveis serviços residenciais a serem disponibilizados no futuro. Avaliar o impacto deste conjunto de serviços em redes de acesso e domésticas actuais e de próxima geração.
- Investigar soluções que permitam a criação de uma plataforma residencial multi-serviços, a qual contemple também um *gateway* multi-serviços que possibilite a instalação, gestão e configuração remota de serviços residenciais.
- Realizar um trabalho de implementação sobre a solução apresentada anteriormente, permitindo assim a validação prática da mesma.

1.2. Abordagem

O tema da presente dissertação aborda vários assuntos (redes de acesso e domésticas, serviços, plataformas de serviços, entre outros) pelo que o trabalho foi dividido em três fases fundamentais, sendo que o conhecimento adquirido numa fase é fundamental para o desenvolvimento da fase seguinte.

A primeira fase do trabalho pretendeu estabelecer o estado da arte no que diz respeito à prestação de serviços residenciais, bem como em relação às infra-estruturas de rede que suportam essa mesma prestação de serviços. O trabalho iniciou-se com o estudo das redes de

acesso de banda larga actualmente mais comuns no mercado residencial, estudo esse que culminou com a caracterização de três dessas redes. Contudo, uma vez que os serviços residenciais são tipicamente terminados em dispositivos que estão ligados à rede de acesso através de uma rede doméstica, um estudo semelhante foi também efectuado para o domínio das redes residenciais.

Esta infra-estrutura de rede, formada por uma rede de acesso e uma rede doméstica, permite a distribuição de alguns serviços residenciais, serviços estes que, na actualidade, poderão variar de acordo com o tipo de rede de acesso. Deste modo, para cada um dos tipos de rede acesso anteriormente estudados, foram ainda identificados e descritos conjuntos de serviços residenciais típicos.

Após a conclusão da primeira fase, iniciou-se uma segunda fase que teve como principal objectivo a identificação e descrição de uma plataforma multi-serviço que diminua toda a problemática associada à prestação de múltiplos serviços (instalação, gestão, configuração remota, entre outros) num ambiente residencial. Da mesma forma, era também um objectivo nesta fase caracterizar todo o ambiente de rede que servirá de suporte à referida plataforma, pelo que se iniciou a segunda fase do trabalho com a elaboração de uma lista de potenciais serviços a serem disponibilizados a clientes residenciais num futuro próximo. Para este trabalho foram fundamentais as visões do projecto MUSE e do consórcio DSL Forum. A análise dos requisitos impostos por estes serviços permitiu tirar conclusões acerca da capacidade das redes actuais (de acesso e domésticas) para efectuarem o transporte desses mesmos serviços e obter um conjunto de características desejáveis em redes de próxima geração que permitam a distribuição de múltiplos serviços.

Uma vez caracterizadas as redes de nova geração, sobre a qual vários serviços residenciais poderão ser disponibilizados, foi ainda apresentada uma solução que permite a implementação de um *gateway* multi-serviços, a ser incluído na plataforma de serviços e que permite a instalação, gestão e configuração remota de serviços residenciais. A segunda fase terminou com a descrição desta solução, análise de falhas da mesma e apresentação de soluções que permitam colmatar essas mesmas falhas.

A terceira e última fase da presente dissertação centrou-se na implementação de um serviço residencial sobre a plataforma de serviços, bem como na elaboração dos respectivos testes e avaliação. Esta fase teve por objectivo principal validar alguns dos conceitos apresentados durante as duas primeiras fases de trabalho.

1.3. Estrutura da dissertação

O texto desta dissertação encontra-se organizado em cinco capítulos, cujo conteúdo é brevemente descrito nos parágrafos seguintes.

O capítulo 2 começa por apresentar um modelo de referência de rede que irá ser utilizado como suporte ao longo da dissertação. A arquitectura, características e funcionamento de redes de acesso baseadas nas tecnologias xDSL (*x Digital Subscriber Line*), DOCSIS (*Data Over Cable Service Interface Specification*) e *Ethernet* são também descritos neste capítulo. Para cada um dos tipos de rede são ainda descritos os serviços residenciais que um cliente pode tipicamente usufruir. O capítulo encerra com a caracterização de uma rede doméstica actual.

No capítulo 3 é apresentado um conjunto de serviços e aplicações que poderão ser disponibilizados a clientes residenciais num futuro próximo, bem como o impacto destes serviços em futuras redes de acesso e domésticas. Neste capítulo são também apresentadas plataformas de serviço típicas, nomeadamente plataformas de serviços vídeo/áudio, bem como a plataforma OSGi. O capítulo termina com a descrição da plataforma TR-069 e o do protocolo *CPE WAN Management Protocol*.

O capítulo 4 apresenta a descrição, motivação e caracterização de um serviço de monitorização médica remota, desenvolvido no âmbito desta dissertação, com o objectivo de validar com um exercício prático a plataforma OSGi e o seu enquadramento numa plataforma de serviços residenciais. O enquadramento deste mesmo serviço no âmbito do projecto MUSE é também apresentado. O capítulo encerra com uma descrição pormenorizada dos módulos de *software* desenvolvidos sobre a plataforma OSGi e respectivos testes de validação.

No capítulo 5, são apresentadas as conclusões finais do trabalho realizado bem como propostas de trabalho futuro.

1.4. Trabalho preliminar

A presente dissertação aborda problemas e respectivas soluções com que os operadores de telecomunicações e provedores de serviços se deparam na implementação de uma plataforma de serviços residenciais. Porém, um conjunto de pesquisas, estudos e implementações previamente efectuado, e não aqui descrito, foi também fundamental para a conclusão desta mesma dissertação. Este trabalho preliminar centrou-se no chamado *gateway* residencial,

elemento fundamental na plataforma de serviços, e culminou num conjunto de publicações. Entre estas destaca-se o artigo e demonstração intitulada *“Demo of Triple Play Services with QoS in a Broadband Access Residential Gateway”* [10] efectuada na conferência IEEE Infocom 2006 [11], o artigo intitulado *“Pasarela residencial multiservicio con soporte de calidad garantizada para acceso de banda ancha”* [12] apresentado na conferência Telecom I+D 2006 [13], e mais recentemente o artigo intitulado *“Experiences with a QoS enabled Residential Gateway within a European multiservice broadband access environment”* [14], apresentado na conferência *Wired/Wireless Internet Communications* 2007 [15]. Assim, a presente dissertação é considerada uma extensão deste mesmo trabalho preliminar.

Capítulo 2 – Estruturas de rede de suporte a serviços

O serviço de acesso à Internet em banda larga foi até agora o principal impulsionador das subscrições de acessos de banda larga por parte de clientes residenciais. A crescente procura deste serviço tem levado os operadores de telecomunicações a modernizar as suas redes de acesso por forma a que cada vez maiores velocidades de transmissão sejam conseguidas e oferecidas aos seus clientes. Por outro lado, este aumento de largura de banda no serviço de acesso à Internet, gerou o aparecimento de redes domésticas simples cujo principal objectivo é a partilha do serviço de acesso à Internet entre vários terminais domésticos. Esta infra-estrutura de rede será o ponto de partida em direcção a uma infra-estrutura que suporte a distribuição de múltiplos serviços (rede multi-serviço) potencialmente provenientes de múltiplos provedores de serviços (rede multi-provedor).

Este capítulo apresenta o estado da arte no que diz respeito a infra-estruturas de rede de acesso de banda larga. Para cada tipo de rede analisada, é efectuada a descrição e caracterização dos métodos de auto-configuração, gestão, QoS (Qualidade de Serviço), autorização, autenticação e taxação aplicados na prestação de alguns serviços residenciais típicos. Da mesma forma, é também elaborado o estado da arte no que diz respeito a redes residenciais.

2.1. Modelo de referência de rede

Ao longo desta dissertação são feitas várias referências a entidades de rede que agrupam um determinado conjunto de funções. Assim, de forma a normalizar nesta dissertação a nomenclatura usada e as funções desempenhadas por cada um desses elementos, são apresentados na secção 2.1.1 o modelo de referência de rede completo e na secção 2.1.2 um modelo de referência da rede doméstica, considerado um subconjunto do primeiro modelo.

Deve ser claro que, em primeiro lugar, os blocos apresentados nos modelos não representam dispositivos físicos mas sim um conjunto de funcionalidades, o que significa que um só dispositivo físico pode conter vários blocos e portanto alguns dos pontos de referência apresentados no modelo de referência são internos a esse dispositivo. Em segundo lugar, alguns dos blocos podem estar ausentes numa arquitectura real, assim como cada bloco pode desempenhar mais ou menos funções quando se aplica o modelo a uma arquitectura de rede real.

2.1.1. Modelo de referência de rede completo

O modelo de referência da figura 1, construído com base em modelos apresentados em vários documentos do DSL Forum, é o modelo de referência de rede completo definido e adoptado pelo projecto MUSE.

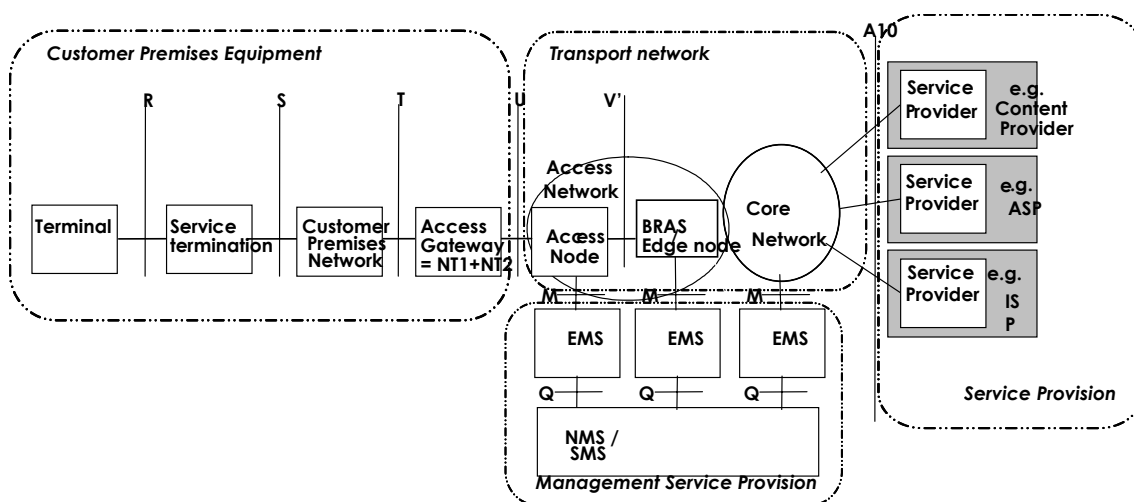


Figura 1: Modelo de referência de rede completo [16]

O modelo da figura 1 é constituído por vários blocos que por sua vez são agrupados em quatro grupos funcionais:

- *Customer premises equipment* – equipamento das instalações de cliente: conjunto de equipamento que reside nas instalações de um determinado cliente, também referido por rede doméstica. Este grupo funcional, os blocos que o constituem e os pontos de referência definidos entre eles serão descritos em maior detalhe na secção 2.1.2.
- *Transport network* – rede de transporte: confinada entre os pontos de referência U e A10, que a separam respectivamente das instalações do cliente e da rede de provisão de serviços, providencia a conectividade e transporte de tráfego entre os equipamentos do utilizador e um ou vários provedores de serviços. Pode ainda ser subdividida em *access network* e *core network*:
 - *Access network* – rede de acesso: constituída por troços de primeira milha que podem ser de diferentes tecnologias (por exemplo par de cobre, cabo coaxial, fibra) que são conectados a nós de acesso (*access nodes*) e funcionam como pontos de agregação do tráfego proveniente dos terminais dos clientes. A ligação entre a rede de acesso e a rede de núcleo (*core network*) é feita por um nó de fronteira (*edge node*) que funciona como um ponto de controlo de acesso à rede de núcleo para o tráfego que se apresenta no ponto de referência V’.
 - *Core network* – rede de núcleo: rede que transporta o tráfego de/para a rede de provisão de serviços (*service provision*).
- *Service provision* – provisão de serviços: formada por um conjunto de provedores de serviços, cada um dos quais pode fornecer um ou mais serviços. É separada da rede de transporte pelo ponto de referência A10 onde poderá também existir, apesar de não estar representado na figura 1, um ponto de controlo do acesso à rede de provisão de serviços, normalmente chamado de nó de fronteira de serviço (*service edge node*).
- *Management service provision* – gestão de aprovisionamento de serviços: efectua tarefas de configuração e gestão nos vários elementos da rede de transporte de acordo com os serviços que atravessam a rede. Esta entidade é constituída pelos seguintes elementos:
 - *Network Management System* (NMS) [17] – sistema de gestão de rede: responsável pela gestão de todos os elementos que constituem a rede de transporte. Todavia, em vez de fazer a gestão directa dos vários elementos de rede, instrui (através de interfaces definidas nos pontos de referência Q) os vários EMS (*Element Management Systems*) para o fazerem. Por sua vez o NMS pode interagir e ser instruído por um ou mais SMS (*Service Management Systems*).

- *Element Management Systems* (EMS) [17]– sistemas de gestão de elementos: responsáveis por efectuarem as tarefas de gestão nos vários tipos de elementos de rede de acordo com as instruções do NMS.

2.1.2. Modelo de referência da rede doméstica

A figura 2 apresenta em detalhe um modelo que se refere ao conjunto de dispositivos referido na figura 1 como *customer premises equipment*. Este modelo de referência, apresentado no âmbito do projecto MUSE, foi elaborado introduzindo algumas modificações (com o objectivo de o tornar o mais geral possível) a modelos já propostos no âmbito de organizações como o DSL Forum [18], actualmente o organismo mais importante no que toca a questões relacionadas com redes domésticas, e do ITU-T [19].

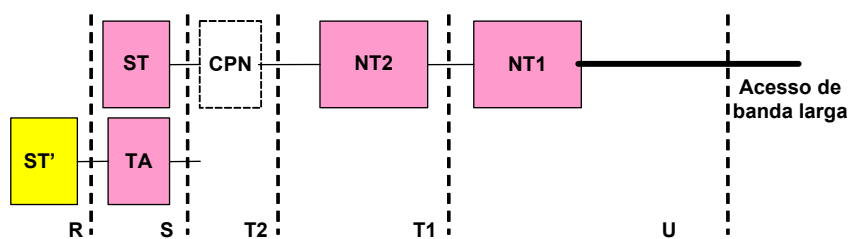


Figura 2: Modelo da arquitectura de uma rede residencial [16]

Na figura 2 estão representados vários blocos funcionais e interfaces (físicas e/ou lógicas) que no seu conjunto formam um possível modelo de representação de uma rede residencial. De seguida é feita uma breve descrição de cada um desses blocos:

- NT1 (*Network Termination 1*): este elemento termina ao nível físico a tecnologia usada na linha de acesso, ocultando as características desta para a rede residencial. Adicionalmente apresenta os dados recebidos à rede residencial através de uma interface normalizada, T1, o que normalmente se traduz numa interface compatível com as normas *Ethernet*. Um caso particular verifica-se quando a tecnologia ATM é usada na rede de acesso (por exemplo redes ADSL, ver secção 2.2.3.1). Nessa situação o bloco NT1 deverá também implementar as funções necessárias para lidar com esta tecnologia sendo assim feita a terminação das células ATM neste bloco e portanto as tramas do protocolo tributário são transmitidas transparentemente ao bloco NT2. Sintetizando, numa arquitectura actual o bloco funcional NT1 pode ser interpretado como um modem xDSL, DOCSIS ou de outra tecnologia.

- NT2 (*Network Termination 2*): este elemento poderá desempenhar um vasto leque de funções ao nível da camada de dados e/ou de rede (camadas 2 e/ou 3) do modelo definido no âmbito do organismo ISO [20], vulgarmente chamado de modelo OSI [21]. A seguir são referidas algumas delas:

- Transferir dados entre a rede residencial e a rede de acesso através de *Ethernet bridging* [22] ou através de encaminhamento IP transparente ou ainda através de uma função NAT/NAPT (*Network Address Translation/Network Address Port Translation*) [23]. Neste último caso, funções de *Application Layer Gateway*, que permitam a passagem de protocolos de sinalização como SIP (*Session Initiation Protocol*) [24], RTSP (*Real Time Streaming Protocol*) [25], FTP (*File Transfer Protocol*) [26], entre outros, através da função de NAPT, têm de ser implementadas no bloco NT2.
- Transferir tráfego dentro da rede residencial (entre as várias interfaces presentes no ponto de referência T2) através de *Ethernet bridging* ou de encaminhamento IP.
- Assegurar conectividade ao nível da camada de dados e/ou de rede entre as redes de acesso e residencial (por exemplo, estabelecendo sessões PPP (*Point-to-Point Protocol*) [27] com entidades na rede de acesso).
- Implementar funções de controlo de acesso entre rede de acesso e residencial (por exemplo, estabelecimento de sessões PPP para efeitos de autorização, *firewall*).
- Executar funções relacionadas com QoS, tais como classificação e marcação de pacotes/tramas, mecanismos de *scheduling*, entre outros.
- Efectuar tradução entre os mecanismos de QoS usados na rede doméstica (por exemplo, esquema de prioridade de tramas baseado em IEEE802.1p [28]) e de acesso (por exemplo, *diffserv* [29]).

O bloco NT2 poderá conter várias interfaces no ponto de referência T2, que permitirão a conexão de CPNs (ver ponto seguinte) de diferentes tecnologias. Importa no entanto referir que apenas tecnologias que permitem o transporte de tramas *Ethernet*, de uma forma transparente, são consideradas. Assim, para além de tecnologias que implementam as camadas física e MAC das normas IEEE802.3 [30] e IEEE802.11 [31], [32], [33], [34], tecnologias como HomePNA (*Home Phoneline Networking Alliance*) [35] ou HomePlug [36] são também válidas. O bloco NT2 é, neste caso, responsável por efectuar a conversão entre as várias tecnologias.

- CPN (*Customer Premises Network*): é uma *local area network* (LAN) que pode ser tão simples como uma ligação 10/100 Base-T [30], [37] entre o bloco NT2 e um bloco ST (ou TA), ou pode ser uma rede mais complexa com *hubs* e *switches* (é assumido que *routers* [38] não existem na CPN, uma vez que o bloco funcional NT2 já acumula as funções normalmente desempenhadas por um *router*). Como já referido, é contudo obrigatório que a tecnologia que constitui uma determinada CPN permita o transporte de tramas *Ethernet* de uma forma transparente aos blocos funcionais que a CPN pretende interligar. Da mesma forma, interfaces presentes nos pontos de referência T2 e S terão de ser compatíveis com estas tecnologias.
- ST (*Service Termination*): este bloco funcional representa o ponto de interface entre um determinado serviço e o utilizador. Funções relacionadas com o controlo do serviço (por exemplo, funções de sinalização) e com a terminação da ligação de dados relativa ao serviço são implementadas neste bloco. Os dados recebidos são depois processados e o seu conteúdo apresentado directamente ao utilizador sobre a forma de som, texto, imagem ou vídeo. Computadores pessoais, PDAs (*Personal Digital Assistant*) e telefones VoIP (*Voice over IP*) são exemplos de dispositivos que desempenham esta função.
- TA (*Terminal Adaptation*): este bloco poderá fazer a adaptação para uma outra tecnologia de rede digital (por exemplo, USB [39], IEEE1394 [40], Bluetooth [41]), de rede analógica (POTS) ou para uma interface específica (por exemplo SCART).
- ST' (*Service Termination*): este bloco funcional tem funções semelhantes ao bloco ST com a diferença de que será o ponto de interface entre um serviço e o utilizador após a adaptação.

O modelo apresentado na figura 2, pretende modelar redes domésticas actuais e futuras. A figura 3 apresenta algumas configurações de redes domésticas representadas de acordo com esse modelo. Estas configurações são também utilizadas para a introdução de alguma terminologia característica.

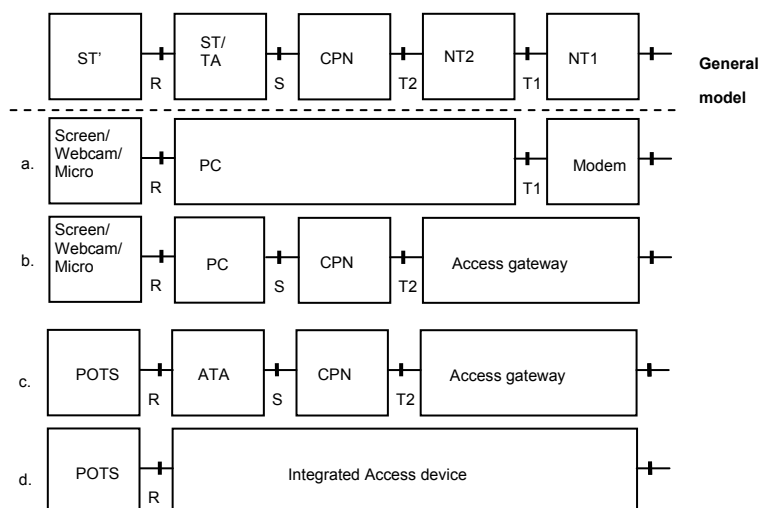


Figura 3: Configurações de redes residenciais [16]

A rede representada em *a*, é o caso mais simples onde um PC está directamente ligado a um modem através de uma interface *Ethernet*, sendo que a rede representada no modelo *b* evoluiu um pouco mais este conceito: em primeiro lugar os blocos funcionais NT1 e NT2 encontram-se implementados num mesmo dispositivo físico denominado de *access gateway* (*gateway* de acesso). Este não é mais que do que um *switch* ou *router* com um modem integrado, o que é bastante comum na actualidade. Em segundo lugar existe já uma rede local (representada pelo bloco CPN), o que permite que vários PCs partilhem os recursos disponibilizados pelo *gateway* de acesso. Esta é, eventualmente, a topologia de rede doméstica mais comum nos dias de hoje, onde vários utilizadores partilham um acesso à Internet em banda larga.

Por sua vez a configuração representada em *c*, permite ligar a uma CPN, onde poderão estar conectados outros dispositivos, um telefone POTS que possibilita ao utilizador fazer chamadas de voz, com um terminal analógico tradicional, através de uma rede IP. A adaptação de tecnologias e protocolos é feita através de um ATA (*Analogue Terminal Adapter*) (o dispositivo MTA (*Multimedia Terminal Adapter*) descrito na secção 2.2.3.2.3.1 é um tipo de ATA). Finalmente a topologia apresentada em *d* é algo semelhante à anterior, com a diferença de um dispositivo físico, denominado por *Integrated Access Device*, ser usado exclusivamente para o serviço de telefonia sobre IP. O dispositivo E-MTA (*Embedded MTA*), descrito nas normas PacketCable [42], é um tipo de IAD (*Integrated Access Device*).

2.2. Arquitecturas de rede e serviços típicos

A primeira parte desta secção descreve de uma forma breve e sucinta a arquitectura de algumas redes de acesso típicas, alguns serviços que normalmente são disponibilizados através dessas mesmas redes e finalmente como é feita a gestão e controlo desses mesmos serviços pelo operador. Independentemente da rede de acesso utilizada para fazer chegar os serviços ao cliente, estes são sempre terminados em equipamentos presentes nas instalações do cliente. Na segunda metade da presente secção são ainda apresentadas algumas configurações típicas de redes domésticas.

2.2.3. Redes de acesso

Actualmente as redes de acesso de banda larga destinadas ao mercado residencial e SOHO (*Small Office Home Office*) mais difundidas são sem sombra de dúvida as que usam as tecnologias xDSL e a tecnologia DOCSIS na primeira milha. Porém, alguns operadores de telecomunicações começaram já a implementar redes de acesso totalmente baseadas na tecnologia *Ethernet*, tal como definido no Metro Ethernet Forum [43]. Assim, nas subsecções seguintes é dada particular importância à arquitectura, mecanismos de gestão/controlo e aos serviços típicos de cada um dos três tipos de redes de acesso.

2.2.3.1. Redes de acesso baseadas em tecnologias xDSL

As tecnologias xDSL, cujas normas têm vindo a ser produzidas essencialmente pelos organismos DSL Forum [3], ANSI [44], ETSI [5] e ITU-T [6], fazem a transmissão digital de dados sobre um cabo de cobre que se estende desde as instalações do cliente a uma central de comutação do operador. Os vários tipos de tecnologia xDSL (HDSL, ADSL, ADSL2+, VDSL [46], entre outros), diferem entre si no débito de transmissão conseguido sobre esse mesmo cabo de cobre no sentido ascendente (das instalações do cliente para a rede do operador) e/ou descendente (da rede do operador para as instalações do cliente), não havendo portanto um grande impacto na restante arquitectura da rede de acesso conforme a tecnologia xDSL usada. Assim sendo, todo este capítulo se debruça sobre uma rede de acesso que usa na primeira milha a tecnologia ADSL, e nas secções 2.2.3.1.4 e 2.2.3.1.5 são feitas algumas observações acerca das tecnologias ADSL2 e VDSL, respectivamente.

2.2.3.1.1 Arquitectura de uma rede de acesso baseada em ADSL

A tecnologia ADSL é uma tecnologia ponto a ponto que permite a transmissão assimétrica (no máximo cerca 8Mbit/s no sentido descendente e cerca de 800 Kbit/s no sentido ascendente) digital de dados sobre um par de cobre (usualmente um cabo TVHV, primariamente usado na distribuição do serviço POTS), que não deve exceder os 3 Km de comprimento para os débitos acima enunciados. Ao contrário de normas do ITU-T como as normas V.34 [47] e V.90 [48], a norma ADSL [49] prevê que a transmissão de dados ao longo do par de cobre possa ser feita em simultâneo com a utilização do serviço de telefonia (POTS) ou RDIS (Rede Digital com Integração de Serviços). Tal é conseguido usando uma gama de frequências distinta da gama de frequências usada pelo serviço POTS (representada na figura 4) e RDIS.

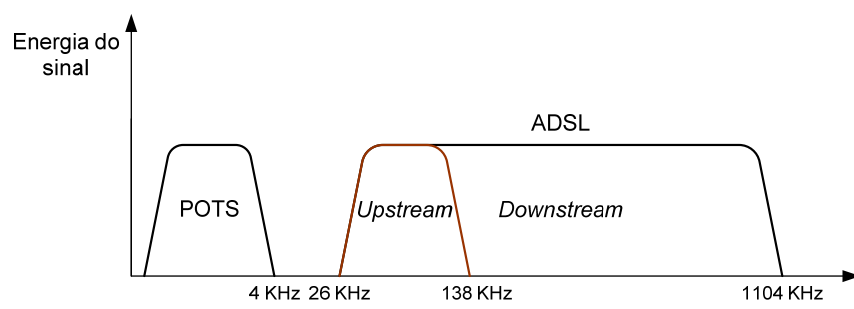


Figura 4: Espectro de frequências POTS e ADSL

No sentido descendente (*downstream*), ou seja, da rede do operador para as instalações do cliente, um filtro (*splitter*) integrado no modem ADSL (ou colocado antes deste) separa o sinal a ser enviado para o telefone POTS (frequências mais baixas – banda base) do sinal a ser enviado para o modem ADSL (frequências mais altas). Da mesma forma, no sentido ascendente (*upstream*), na central de comutação um filtro é responsável por separar o sinal proveniente do telefone POTS presente nas instalações do cliente (destinado à rede PSTN) do sinal proveniente do modem ADSL.

A figura 5 apresenta a arquitectura de uma rede de acesso que utiliza a tecnologia ADSL na primeira milha.

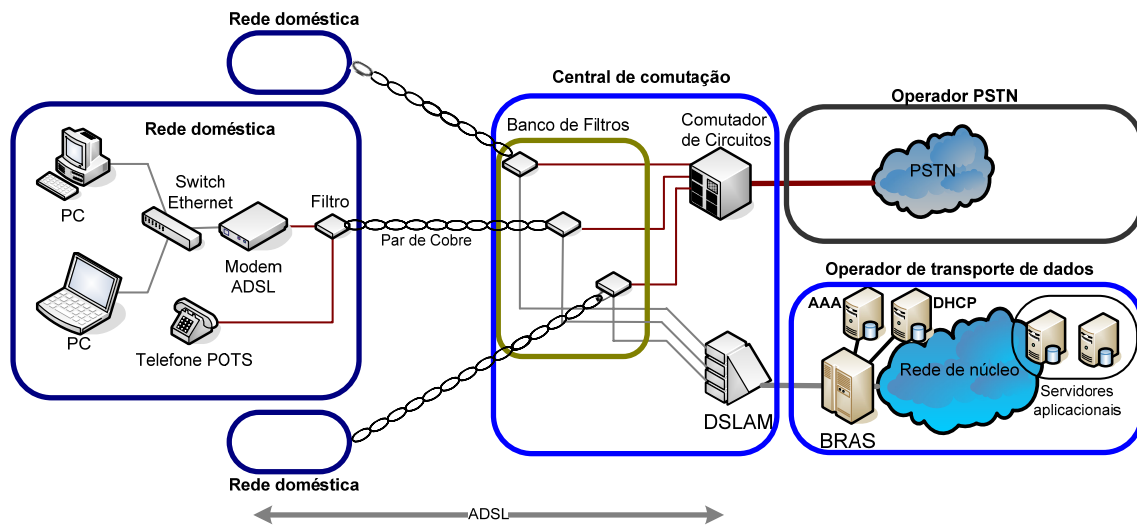


Figura 5: Arquitectura de uma rede de acesso que usa a tecnologia ADSL na primeira milha

De um modo geral a arquitectura é composta pelos modems ADSL (também denominados de ATU-R, *ADSL Transceiver Unit - Remote*) que residem nas instalações do cliente, filtros colocados nos extremos de cada par de cobre, nós de acesso denominados de DSLAMs (*Digital Subscriber Line Access Multiplexer*), que contêm as unidades ATU-C (*ADSL Transceiver Unit - Central*), colocados nas centrais de comutação e finalmente por nós fronteira denominados de servidores de acesso remoto de banda larga (BRAS).

Assim, os modems ADSL ligados através do par de cobre a um dado DSLAM e os elementos ATU-C desse mesmo DSLAM funcionam como um par, na medida em que são responsáveis por fazerem a descodificação/codificação dos dados transmitidos/recebidos pela unidade ATU que se encontra no outro extremo do par de cobre. Para além disso o modem ADSL é ainda responsável por entregar os dados recebidos do par de cobre aos terminais da rede doméstica, através de uma interface normalizada (por exemplo 10/100 Base-T, USB ou ATM25 [50]). Por sua vez, o DSLAM, após descodificar os dados recebidos das várias linhas de cobre que termina (descodificação essa feita pelos elementos ATU-C), é ainda responsável por multiplexar e enviar os dados recebidos para a rede do operador através de uma interface de alto débito (por exemplo através de uma interface ATM ou *Gigabit Ethernet*). Sucessivas operações de agregação são depois feitas ao tráfego enviado pelos vários DSLAMs na direcção ascendente. O agregado de tráfego é depois recebido pelo BRAS que para além de funcionar como ponto de entrada para a rede de núcleo do operador e de fazer o encaminhamento de tráfego para o seu destino, pode desempenhar também funções de AAA (*Authentication, Authorization, and Accounting*) de clientes e de auto-configuração de terminais, entre outras.

2.2.3.1.2 Serviços típicos

Nesta secção são descritos os serviços que normalmente são disponibilizados aos clientes que usufruem de um acesso de banda larga ADSL.

2.2.3.1.2.1 Serviço de telefonia (POTS, RDIS)

Este serviço inclui chamadas de voz entre terminais, através de uma rede clássica de banda estreita. Dependendo da técnica utilizada para transmitir o sinal de voz temos o serviço de telefonia POTS (sinal analógico) e o serviço de telefonia RDIS (sinal digital), sendo que para utilizar o serviço são necessários terminais (telefones) POTS e RDIS respectivamente. Em ambos os casos existem serviços complementares ao serviço de telefonia, tais como o reenaminhamento de chamadas, aviso de chamada em espera, identificação de chamadas, chamada em conferência, entre outros.

2.2.3.1.2.2 Acesso à Internet em banda larga

Este serviço dá aos utilizadores acesso à Internet. Tipicamente os operadores de rede oferecem aos clientes a possibilidade de escolha entre vários pacotes de serviços que se distinguem, entre outros factores, pela velocidade máxima de transmissão obtida na direcção descendente e ascendente, e também pelo volume de tráfego máximo gerado/recebido. Com este serviço o cliente tem normalmente acesso a um conjunto de outros de serviços, nomeadamente:

- *Web Browsing*: transferência de objectos alojados num servidor *Web* para o terminal do utilizador, usando para tal um *software* cliente (*Web Browser*).
- P2P (*Peer to Peer*): o termo *peer to peer* descreve aplicações através das quais os utilizadores podem usar a Internet para trocar ficheiros directamente entre si sem recorrer a um servidor mediador. Estas aplicações permitem a transferência de um único ficheiro cuja fonte é normalmente um conjunto de outros utilizadores que possuem esse mesmo ficheiro, sendo que são toleradas interrupções na transferência desse mesmo ficheiro.
- FTP (*File Transfer Protocol*): as aplicações de FTP permitem aos utilizadores copiarem ficheiros entre o seu sistema local e outro qualquer sistema remoto que esteja acessível através da rede. Normalmente um identificador de utilizador e respectiva *password* são necessários para o utilizador se ligar ao sistema remoto.

- *E-Mail*: esta aplicação refere-se à troca de mensagens entre utilizadores, que podem conter para além de texto outro tipo de conteúdos, através de servidores centrais (que podem pertencer ou não ao operador). Para tal são usados protocolos como o SMTP (*Simple Mail Transfer Protocol*), POP3 (*Post Office Protocol 3*) ou, no caso de se tratar de *Web mail*, o protocolo HTTP(S) [51].
- *Instant messaging*: as aplicações de IM (existem várias redes de IM) permitem ao utilizador verificar se outros utilizadores estão ligados à rede de IM e, caso estejam, permitem a troca de mensagens em tempo real com esses mesmos contactos. Algumas aplicações para além da troca de mensagens de texto permitem também a troca de mensagens de voz e a partilha de ficheiros.

2.2.3.1.3 Gestão e controlo de serviços

As normas das tecnologias xDSL apenas definem parâmetros relativos à camada física do modelo OSI, nada dizendo sobre a camada de dados. Todavia, organismos como o DSL Forum promovem o uso da tecnologia ATM sobre a tecnologia xDSL. Esta preferência pela tecnologia ATM acontece devido à universalidade e maturidade da tecnologia, às elevadas taxas de transmissão suportadas, ao suporte de QoS, à capacidade de auto-configuração (usando o protocolo de gestão ILMI [52]) e à capacidade de transporte de vários tipos de protocolos tributários (usando por exemplo a camada de adaptação AAL5 e um dos dois mecanismos definidos na RFC 2684 [53]). Da mesma forma, nada é especificado em relação aos protocolos a usar nas camadas protocolares superiores ao ATM, pelo que um conjunto variado de pilhas protocolares podem surgir, tal como é referido no Anexo I.

Assim, inicialmente, o DSL Forum apenas reconheceu no ponto de referência U (figura 1) a pilha protocolar PPP *over* ATM (PPPoA) [54]. Porém, vários operadores de rede, em muito devido a limitações inerentes a essa pilha protocolar, montaram as suas redes xDSL usando outras pilhas protocolares como PPPoE [55], IPoE [56] e IPoA [57], o que obrigou a que o DSL Forum passasse a reconhecer (entre outras) também estas pilhas protocolares na referida interface [58]. Destes quatro modos de funcionamento, o que usa a pilha protocolar PPPoE é actualmente o mais utilizado pelos operadores para disponibilizar o serviço de acesso à Internet. A figura 6 representa a pilha protocolar de cada um dos elementos de uma rede de acesso que usa tecnologia xDSL na primeira milha, quando o transporte de datagramas é feito usando o modo de operação PPPoE.

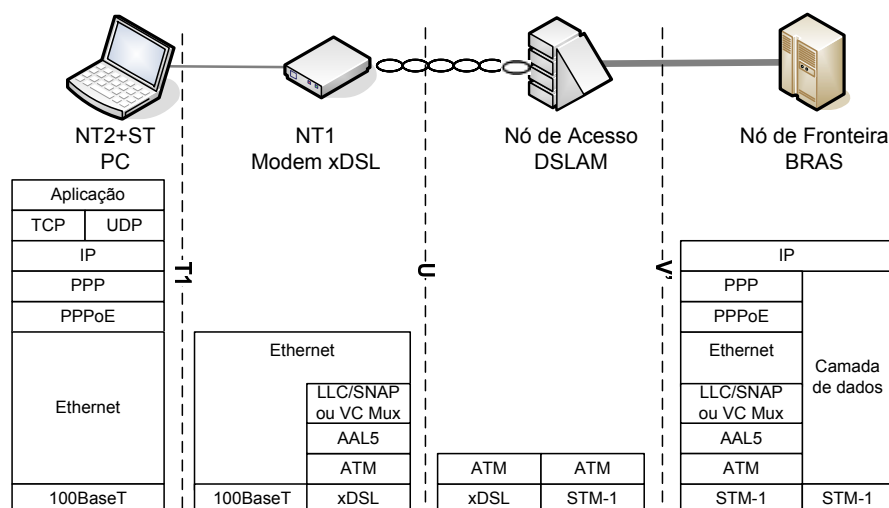


Figura 6: Rede de acesso baseada na tecnologia xDSL a operar no modo PPPoE

Neste modo de funcionamento o BRAS desempenha um papel importante em todo o processo de estabelecimento de conectividade e entrega do serviço ao CPE, servindo de elo de ligação com os mecanismos de autenticação, autorização e taxaço e estando também envolvido no último passo do processo auto-configurao do CPE (configuraoes relacionadas com a camada IP), como sero explicado mais a frente.

O processo de auto-configurao e centrado no CPE e acontece por fases. Ao longo destas o CPE vai adquirindo incrementalmente informao acerca da rede at que o sero e activado. O primeiro passo no processo de auto-configurao e o estabelecimento de conectividade fisica entre o modem xDSL e o DSLAM. Quando o modem e ligado, atraves de um mecanismo de treino, e determinada qual a taxa de transferencia maximo conseguida na ligao xDSL. Uma vez estabelecida a conectividade fisica entre o DSLAM e o modem, um processo opcional de configurao da camada ATM pode ser iniciado. Normalmente os operadores de rede fornecem ja os modems com a configurao da camada ATM efectuada. No entanto, caso o modem nao tenha sido pre-configurado, e possivel passar ao NT1, atraves do protocolo ILMI (*Integrated Local Management Interface*) que corre entre o modem e o DSLAM, toda a informao necessaria para descrever os PVCs (*Permanent Virtual Connection*) ATM (incluindo os parametros VCI (*Virtual Channel Identifier*) e VPI (*Virtual Path Identifier*)) a serem usados. O protocolo ILMI pode tambem ser usado para informar o NT1 acerca de qual dos metodos de encapsulamento [53] a usar para o protocolo tributario do ATM (*Ethernet*). Apes a configurao da camada ATM estar concluida, existe ja conectividade ao nivel da camada *Ethernet* entre o CPE e o BRAS, sendo possivel a partir deste momento estabelecer uma sessao PPPoE entre o modem (caso seja usado um modem/*router*) ou o terminal do cliente (caso seja

usado um modem a operar em modo de *bridging*, como representado na figura 6) e o BRAS.

O protocolo PPP engloba um conjunto de protocolos em que cada um desempenha uma função específica no processo de obtenção de conectividade. Através do protocolo LCP (*Link Control Protocol*) são negociadas várias opções relativas à sessão PPP (cifra, compressão, MTU, entre outros). Uma vez terminada esta fase de negociação, o utilizador pode provar a sua identidade (através da introdução de uma identificação de utilizador e uma senha) usando para o efeito um protocolo de autenticação (tipicamente o PAP [59] ou CHAP [60]). Esta fase de autenticação termina após o BRAS ter consultado o respectivo servidor de autenticação (AAA) onde o utilizador está registado e verificar que o utilizador é quem diz ser e que de facto tem autorização para aceder à rede do operador. Finalmente, através do protocolo IPCP (*IP Control Protocol*) o modem ou o terminal do cliente recebe um endereço IP (que o BRAS obtém após consultar um servidor de DHCP [61]). Informação de configuração adicional, como o endereço IP do servidor de DNS, pode também ser transmitida ao CPE usando extensões proprietárias ao protocolo IPCP [62].

Terminadas estas três fases, que constituem o estabelecimento de uma sessão PPP, está estabelecida a conectividade IP entre o CPE e o BRAS, sendo possível ao utilizador a partir deste momento aceder à rede do operador e portanto à Internet. Enquanto a sessão PPP se mantiver estabelecida o BRAS recolhe informação acerca do volume de tráfego gerado e recebido pelo cliente. Esta informação, após o término da sessão, é enviada conjuntamente com o tempo de duração da sessão para o sistema de *accounting* para possível posterior cobrança ao cliente. Caso o utilizador pretenda usar novamente o serviço, terá de iniciar uma nova sessão PPP.

Pode-se assim concluir que para o serviço de acesso à Internet, o uso da pilha protocolar PPPoE é suficiente para efectuar tarefas de auto-configuração, garantir segurança no uso do serviço, e efectuar tarefas de autenticação, autorização e taxaço.

2.2.3.1.4 ADSL2, READSL e ADSL2+

A tecnologia ADSL2 [63] foi especificamente desenhada para melhorar a taxa de transmissão e alcance conseguidos com a tecnologia ADSL, continuando a ser usada a mesma gama de frequências. Tal foi conseguido, essencialmente, através de melhoramentos na eficiência da modulação (uso de técnicas com um maior número de bits por símbolo), reduzindo o *overhead* introduzido pelas tramas DSL e usando algoritmos de processamento de sinal melhorados.

Assim, usando a mesma gama de frequências que a tecnologia ADSL (ver figura 4), a norma ADSL2 permite, em comprimentos curtos do par de cobre (menores que 1 Km), taxas de transmissão no sentido descendente da ordem dos 12 Mb/s e no sentido ascendente taxas de 1 Mb/s.

Um melhoramento adicional introduzido pelo ITU-T à norma ADSL2, e que é normalmente denominada de READSL2 (*Reach Extended ADSL2*), tem por objectivo estender o alcance do ADSL2 para além dos 5.5 Km. Tal é conseguido fazendo uma manipulação da potência do espectro de frequência.

Por sua vez a norma ADSL2+ [64] especifica o uso de frequências até aos 2.2 MHz (duplicando a gama de frequências usadas pelo ADSL e ADSL2) para a transmissão de dados na direcção descendente, o que resulta num aumento bastante significativo desta taxa de transmissão em linhas telefónicas com comprimento menor que 1.5 Km. A taxa de transmissão no sentido ascendente é semelhante à conseguida nos sistemas ADSL2, cerca de 1 Mb/s.

A figura 7 compara o alcance e débitos de transmissão conseguidos na direcção descendente para a tecnologia ADSL e as três tecnologias DSL descritas nesta secção em função do comprimento do par de cobre.

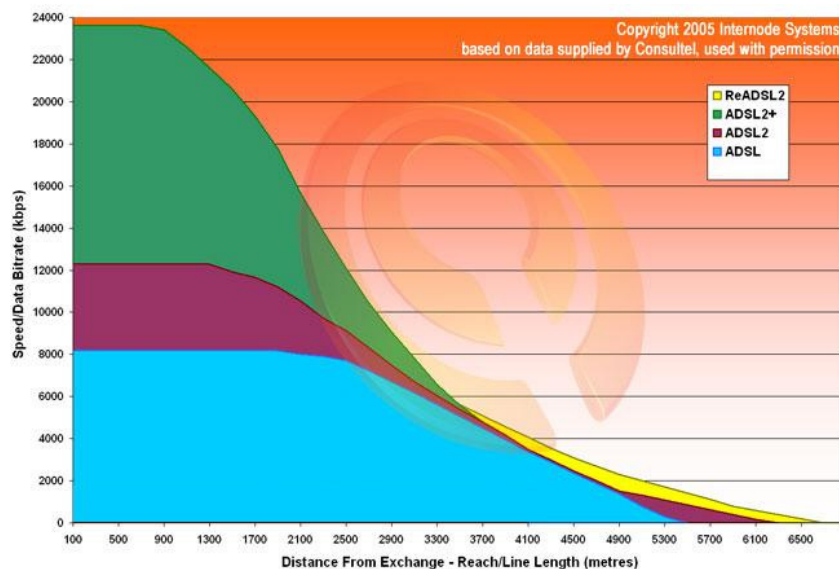


Figura 7: Comparação dos débitos de transmissão obtidos para as tecnologias ADSL, ADSL2, READSL2 e ADSL2+ [65]

2.2.3.1.5 VDSL

A tecnologia VDSL (*Very high speed Digital Subscriber Line*) [46], tem vindo a ser normalizada essencialmente pelos mesmos organismos que produziram as normas ADSL, e permite a transmissão de dados sobre um par de cobre a débitos até 58 Mb/s. Para se conseguirem tais taxas de transmissão sobre a linha telefónica, a gama de frequências usada para a comunicação foi estendida para lá dos 1.1 MHz ocupados pela tecnologia ADSL. A norma VDSL prevê o uso de frequências até aos 30 MHz mas, actualmente, apenas estão disponíveis especificações que permitem o uso do espectro até aos 12 MHz. Uma característica importante da tecnologia, derivada das limitações físicas existentes (essencialmente a atenuação a altas frequências verificada no par de cobre), consiste na limitação da distância do par cobre para cerca de 1.5 Km para as taxas de transmissão acima referidas. Devido a esta limitação de distância, a tecnologia VDSL é indicada para ser usada entre um armário de rua, ligado à rede núcleo por uma fibra óptica, e as instalações do cliente. Esta topologia é normalmente denominada de FTTCab (*Fiber To The Cabinet*). Em alternativa, a tecnologia VDSL pode ainda ser usada para servir clientes que habitem nas imediações da central de comutação.

É importante ainda referir que ao contrário da tecnologia ADSL, a tecnologia VDSL para além de um modo assimétrico possui também um modo de funcionamento simétrico. A tabela seguinte mostra as taxas de transmissão típicas em função do comprimento do par de cobre em ambos os modos de funcionamento.

Comprimento do par de cobre (m)	Débito na direcção descendente (Mb/s)	Débito na direcção ascendente (Mb/s)
300	52	6.4
300	26	26
1000	26	3.2
1000	13	13
1500	13	1.6

Tabela 1: Taxas de transmissão obtidas com a tecnologia VDSL em função do comprimento do par de cobre [66]

2.2.3.2. Redes de acesso CATV

As redes CATV (*Community Access Television*) foram inicialmente desenhadas para permitirem a distribuição de sinais analógicos de TV a vários clientes, o que até então era apenas possível usando o sistema de transmissão terrestre ou por satélite. Devido ao aparecimento de plataformas de transmissão digital como o DVB (*Digital Video Broadcasting*) e às vantagens que

estas apresentam face aos sistemas analógicos (por exemplo, o uso do *codec* MPEG-2 [67] diminui em grande escala a largura de banda necessária à transmissão de um dado canal de TV) os operadores têm vindo a migrar os seus sistemas de transmissão para plataformas digitais. Assim, cada vez mais a largura de banda, anteriormente ocupada por sinais de TV PAL ou NTSC, começa a estar disponível para a transmissão digital de dados. Neste contexto, surgiram as normas DOCSIS [68] que permitem criar sobre os sistemas CATV um “caminho” transparente ao protocolo IP (bem como a protocolos de suporte como o ARP, o DHCP, entre outros). Este, pode ser usado para a transferência de dados a altos débitos entre as instalações do cliente e as instalações do operador da rede de cabo (normalmente denominadas de *Head end*).

Existem actualmente dois grupos de normas, o grupo DOCSIS e o grupo EuroDOCSIS (estas últimas apresentam-se normalmente como um anexo à respectiva norma DOCSIS), ambos criados pelo consórcio CableLabs [69]. O primeiro grupo de normas está mais vocacionado para o mercado norte americano enquanto que as normas EuroDOCSIS são mais vocacionadas para o mercado europeu, mas nada impede que normas DOCSIS sejam implementadas por operadores europeus e vice-versa (a principal diferença entre os dois grupo de normas verifica-se na utilização do espectro de frequências). Existem actualmente três versões de cada um dos dois grupos de normas (ver secção 2.2.3.2.1), designadas de versão 1.0, 1.1 e 2.0 sendo que as duas últimas foram adoptadas como standard ITU-T [70] [71].

2.2.3.2.1 As normas DOCSIS

Existem actualmente três versões das normas DOCSIS: a versão 1.0, 1.1 e 2.0. As duas primeiras normas partilham as mesmas características ao nível da camada física, sendo que para migrar da versão 1.0 para a 1.1 apenas são necessárias actualizações de *firmware* ao CMTS (*Cable Modem Termination System*) e aos CMs (*Cable Modem*). Contudo para migrar da versão 1.1 para a 2.0, uma vez que esta última introduz várias inovações ao nível da camada física, é necessário também alterações ao *hardware* dos elementos de rede DOCSIS.

2.2.3.2.1.1 DOCSIS 1.0 e DOCSIS 1.1

A primeira especificação DOCSIS foi dirigida ao mercado residencial e teve como principal objectivo proporcionar a inter operabilidade de equipamentos produzidos por diferentes

fabricantes. A especificação apenas suporta a transmissão de dados em regime *best-effort* [72] e portanto não oferece qualquer mecanismo para garantir o desempenho de um dado serviço. Nesta especificação são ainda definidos mecanismos básicos que garantem a privacidade dos dados enviados e recebidos pelo CM [73], nomeadamente a cifra de tramas entre o CMTS e o CM e um mecanismo para obtenção e refrescamento de chaves (este mecanismo não define porém qualquer método de autenticação do CM perante o CMTS).

As taxas de transmissão são assimétricas, com a maior parte da largura de banda reservada para a direcção descendente, ou seja do CMTS para o CM, e permitiu aos operadores CATV implementarem rapidamente o serviço de acesso à Internet.

Após o enorme sucesso da primeira versão da norma DOCSIS, o consórcio CableLabs produziu a versão 1.1 que é compatível com a anterior versão 1.0. Esta nova versão introduziu a capacidade de controlo de QoS [74], o que permite a disponibilização de serviços avançados sobre a rede DOCSIS não possíveis com a transmissão de dados em *best effort*. Outras melhorias introduzidas em relação à versão 1.0 foram a introdução de suporte *multicast* e autenticação dos CMs baseada em certificados [75].

Por sua vez, as características da camada física mantiveram-se constantes nas duas primeiras versões da norma DOCSIS.

A tabela 2 resume as principais características da camada física das normas DOCSIS 1.0 e 1.1.

	Transmissão de dados do CM para o CMTS	Transmissão de dados do CMTS para o CM
Frequência	5-42 MHz (EUA) 5-65 MHz (Europa)	42-850 MHz (EUA) 65-850 MHz (Europa)
Largura de banda	200 KHz a 3.2 MHz	6 MHz (EUA) 8 MHz (Europa)
Modulação	QPSK ou 16-QAM	64-QAM ou 256-QAM
Taxa de transmissão	0.32 a 10.25 Mbps	27 a 36 Mbps
Modo de transmissão	Transmissão de rajadas de dados em intervalos de tempo (TDMA)	Transmissão contínua de dados recebidos por todos os CMs

Tabela 2: Características da camada física das normas (Euro)DOCSIS 1.0 e 1.1

2.2.3.2.1.2 DOCSIS 2.0

A natureza assimétrica das normas 1.x limita a capacidade de suporte de serviços que necessitam de simetria das taxas de transmissão (por exemplo vídeo conferência, aplicações p-2-p). Por este motivo e uma vez que a versão 1.1 da norma DOCSIS preencheu as lacunas existentes na versão 1.0 (essencialmente QoS e segurança), a norma DOCSIS 2.0 centrou-se no melhoramento de performance e na maior eficiência do uso da capacidade da rede. Introduziu alterações na camada física e particularmente nos mecanismos usados para transmissão de dados do CM para o CMTS (direcção ascendente) [76]. A versão 2.0 aumenta dramaticamente a taxa de transmissão ascendente proporcionando assim simetria nas taxas de transmissão.

A tabela seguinte resume as principais características da camada física da norma DOCSIS 2.0.

	Transmissão de dados do CM para o CMTS	Transmissão de dados do CMTS para o CM
Frequência	5-42 MHz (DOCSIS) 5-65 MHz (EuroDOCSIS)	42-850 MHz (DOCSIS) 65-850 MHz (EuroDOCSIS)
Largura de banda	200 KHz a 6.4 MHz (ATDMA) 1600 KHz a 6.4 MHz (SCDMA)	6 MHz (DOCSIS) 8 MHz (EuroDOCSIS)
Modulação	QPSK, 8, 16, 32, 64-QAM (ATDMA) QPSK, 8, 16, 32, 64, 128 TCM (SCDMA)	64-QAM ou 256-QAM
Taxa de transmissão	10.30 a 30.96 Mbps	27 a 36 Mbps
Modo de transmissão	Transmissão de rajadas de dados (ATDMA e/ou SCDMA) ³	Transmissão contínua de dados recebidos por todos os CMs

Tabela 3: Características da camada física da norma (Euro)DOCSIS 2.0

2.2.3.2.1.3 As normas EuroDOCSIS

As normas EuroDOCSIS diferem das normas DOCSIS essencialmente na gama de frequências usada para a transmissão de dados no sentido ascendente e descendente, bem como na largura espectral de cada canal descendente. Existem actualmente três versões da norma (1.0, 1.1 e 2.0), sendo que estas são apresentadas sobre a forma de anexo à norma

³ Os operadores podem implementar só uma das técnicas ou ambas (as duas técnicas são compatíveis e podem operar simultaneamente num mesmo canal), dependendo a escolha de um dado operador apenas dos requisitos ou preferências deste.

DOCSIS correspondente. A tabela 2 e a tabela 3 contemplam também as normas EuroDOCSIS.

2.2.3.2.2 Arquitectura de uma rede de acesso CATV

A arquitectura de uma rede CATV sobre a qual é feita transmissão de sinais de TV e transmissão de dados (conforme as normas DOCSIS) é apresentada na figura 8.

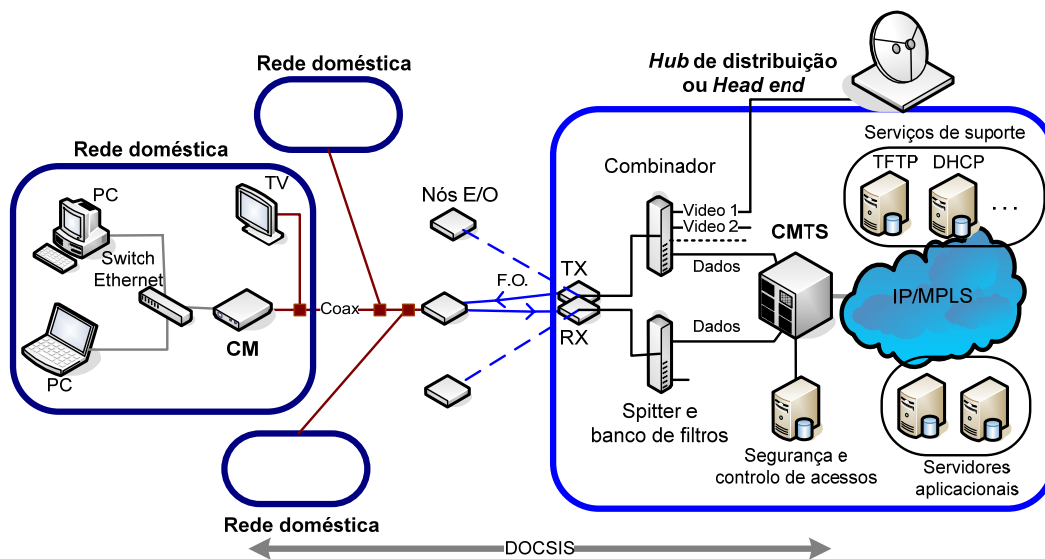


Figura 8: Arquitectura de uma rede de acesso CATV

Como se pode constatar pela figura acima, as redes CATV actuais são normalmente redes HFC (*Hybrid Fibber Coaxial*), visto que a primeira milha é constituída por um misto de fibra óptica e cabo coaxial.

Os elementos CM e CMTS, representados na figura 8, são definidos na arquitectura DOCSIS e de uma forma geral são as entidades que permitem a transmissão bidireccional de tráfego IP entre o *head end* e as instalações do cliente. O modem de cabo para além de desempenhar funções relacionadas com a camada física e de dados DOCSIS, opera ainda como uma *bridge* 802.1D [22], sendo o responsável pelo encaminhamento de tramas *Ethernet* entre os terminais e a rede de acesso. Por sua vez o CMTS, para além de desempenhar funções relacionadas com a camada física e de dados DOCSIS, opera ainda como uma *bridge* 802.1D ou em alternativa como um *router* [38]. É importante referir que o CMTS é não só responsável por encaminhar tramas de dados entre a rede HFC e a rede de transporte do operador, mas também entre os canais ascendentes e descendentes da rede HFC (o que permite a comunicação directa entre terminais em diferentes redes domésticas). Para além destas funções, o CMTS é ainda

responsável por participar no processo de auto-configuração do CM e respectivos terminais, participar no processo de registo e autenticação do modem (a autenticação do CM apenas é efectuada caso esteja implementada na rede pelo menos a versão (Euro)DOCSIS 1.1 [75]), desempenhar funções relacionadas com QoS (caso esteja implementada na rede pelo menos a versão (Euro)DOCSIS 1.1 [74]), efectuar tarefas de gestão no modem (via protocolo SNMP [77]), entre outras.

Para possibilitar a transmissão simultânea de canais de TV e de dados para as instalações dos clientes, um combinador presente no *head end* combina os sinais provenientes do CMTS (que podem conter dados que dizem respeito ao serviço de acesso à Internet, ao serviço de telefonia, entre outros) com os sinais de TV analógicos. Cada canal de TV analógico ocupa uma porção específica do espectro de frequências disponível (6MHz no sistema NTSC ou 8MHz no sistema PAL). Por sua vez os sinais digitais ocupam porções do espectro idênticas às ocupadas pelos canais de TV (6MHz no sistema DOCSIS e 8Mhz nos sistemas EuroDOCSIS), sendo que podem ser usados um ou vários canais para a transmissão de dados. O sinal resultante da combinação é depois difundido através da rede HFC até chegar às instalações de todos os clientes. Aí os sinais analógicos são descodificados em receptores de TV analógicos, enquanto que os sinais digitais são descodificados e terminados no CM.

É conveniente lembrar que o objectivo das especificações DOCSIS é o transporte bidireccional ao longo da rede HFC de pacotes IP. Na direcção descendente tal é conseguido encapsulando os pacotes IP em tramas *Ethernet* que depois são ainda encapsuladas em tramas MPEG-2 *Transport Stream*. Finalmente o sinal eléctrico resultante da conversão das tramas é enviado ao longo da rede HFC combinado com os sinais de TV, tal como referido acima. Uma característica importante das redes DOCSIS é o facto de no sentido descendente esta ser uma rede de difusão e portanto todos os CMs presentes na rede recebem as tramas MPEG-2 provenientes do *head end*. Isto obriga a que após a reconstrução de cada trama *Ethernet* (encapsulada em tramas MPEG-2) todos os CMs verifiquem se o endereço MAC de destino dessa mesma trama é o seu próprio endereço MAC (ou um endereço de *broadcast* ou de *multicast*). Em caso afirmativo a trama é passada à camada protocolar superior (ver secção 2.2.3.2.4) , caso contrário é descartada. Devido a esta característica os CMs recebem tramas que não se lhe são dirigidas, pelo que parte das tramas *Ethernet* é cifrada pelo CMTS antes de estas serem enviadas para a rede. Cada trama é depois decifrada apenas pelo CM que possui uma chave de decifra válida, obtida após troca de material criptográfico com o CMTS no

processo de auto-configuração do CM (ver secção 2.2.3.2.4).

Pode-se então concluir que na direcção descendente uma rede DOCSIS pode ser vista como uma rede ponto multiponto e toda a largura de banda disponível é partilhada entre CMs existentes na rede. Em redes que tenham implementada a norma DOCSIS 1.1 ou 2.0 [76] (normas que permitem QdS) esta partilha de largura de banda pode ser feita de uma maneira ponderada, para que determinados parâmetros de QdS pré acordados entre o cliente e operador possam ser respeitados. Para que a largura de banda disponível a cada cliente seja ampliada, basta que o operador dedique um maior número de canais à transmissão de dados o que, após uma redistribuição dos respectivos CMs pelos canais, resulta num menor número de CMs por cada canal.

Por sua vez, para a transmissão de dados no sentido ascendente está disponível uma pequena parte do espectro que é dividida em canais de largura variável. Ao contrário do que acontece na transmissão de dados do CMTS para os CMs, temos agora vários dispositivos (CMs) a transmitirem dados para uma só entidade (CMTS), usando um mesmo canal de transmissão. A norma DOCSIS resolve este problema dividindo o acesso ao canal de transmissão em intervalos de tempo, intervalos esses que são atribuídos a cada CM pelo CMTS (com base em reservas prévias ou através de contenção) para que estes possam transmitir os seus dados. Ao contrário do que acontece na direcção descendente, as tramas *Ethernet* enviadas pelos CMs não necessitam de ser encapsuladas em tramas MPEG-2 mas, todavia, parte das tramas é cifrada.

2.2.3.2.3 Serviços típicos

Nesta secção são descritos os serviços que normalmente são disponibilizados aos clientes que usufruem de um acesso de banda larga através de uma rede de acesso CATV.

2.2.3.2.3.1 Serviço de telefonia IP

O serviço de telefonia IP possibilita chamadas de voz, pessoa a pessoa (e outros serviços complementares) como acontece com o serviço POTS descrito na secção 2.2.3.1.2.1. A principal diferença reside no facto de o tráfego de voz e respectiva sinalização serem transportados sobre o protocolo IP.

Assim, após terem sido publicadas as normas DOCSIS 1.1, que permitem a transmissão de pacotes IP sobre a rede de cabo com QdS de uma forma segura, o consórcio CableLabs

iniciou a especificação das normas PacketCable que visam a entrega de serviços multimédia tempo real (por exemplo o serviço de telefonia IP). Estas, definem várias novas entidades que têm de ser adicionadas às redes CATV para o suporte dos referidos serviços. Entre elas estão as entidades MTA (*Multimedia Terminal Adaptor*) e CMS (*Call Management Server*) bem como um conjunto de *gateways* para a rede PSTN. A unidade MTA, peça fundamental da arquitectura, reside nas instalações do cliente e possui uma interface para um equipamento de voz (por exemplo telefone POTS) e uma outra interface para a rede HFC que permite a comunicação (via CM) com os elementos de controlo situados na rede (por exemplo CMS). *Codecs*, todas as funções de sinalização (as normas PacketCable definem o protocolo NCS (*Network-based Call Signalling*) [78], baseado no protocolo do IETF MGCP (*Media Gateway Control Protocol*) 1.0 [79], para este fim) e encapsulamento necessárias para o transporte de voz sobre IP (os dados relativos à voz são transportados sobre a rede em pacotes RTP [80]) e ainda funções de sinalização de QoS estão também incluídas no elemento MTA.

2.2.3.2.3.2 Acesso à Internet de banda larga

A norma DOCSIS 1.0 foi desenvolvida com o intuito de usar as redes CATV para a transmissão de dados sobre IP num modelo *best effort*. Assim, um dos primeiros serviços disponibilizados pelos operadores que implementaram esta norma foi precisamente o serviço de acesso à Internet (que continua a ser disponibilizado nas redes DOCSIS actuais). Este serviço é equivalente ao descrito na secção 2.2.3.1.2.2.

2.2.3.2.3.3 Serviço de difusão de TV

O serviço de difusão de TV através de uma rede CATV é baseado na difusão do tradicional serviço de TV recebido num televisor. A arquitectura de rede apresentada na secção 2.2.3.2.2 apenas menciona a difusão de canais de TV em formato analógico. No entanto, alguns operadores de CATV estão agora a introduzir nas suas redes sistemas de transmissão em formatos digitais. A norma MPEG-2 é actualmente a mais adoptada para a transmissão e codificação de conteúdos multimédia e permite a transmissão de múltiplos canais na mesma porção de espectro ocupada por um só canal de TV analógico (ou seja 6MHz ou 8MHz). Consoante a qualidade do serviço de TV digital, este pode ser classificado como SDTV (*Standard Definition TV*) ou HDTV (*High Definition TV*). Este último requer uma maior largura de banda por canal.

Uma desvantagem da migração para um sistema de transmissão digital é o facto de os receptores de TV actuais não estarem preparados para descodificar tais sinais. Por este motivo é necessário um dispositivo adicional que faça a conversão destes sinais para um sinal analógico passível de ser interpretado pelos receptores de TV tradicionais. Este dispositivo é normalmente denominado de STB (*Set Top Box*) e é normalizado pela iniciativa OpenCable do consórcio CableLabs.

2.2.3.2.4 Gestão e controlo de serviços

Ao contrário do que acontece com as normas xDSL, que especificam principalmente a camadas física da tecnologia, as normas DOCSIS especificam claramente as pilhas protocolares que cada um dos elementos especificados na arquitectura devem suportar. A figura 9 apresenta as pilhas protocolares implementadas nos principais elementos de uma arquitectura DOCSIS para o serviço de acesso à Internet.

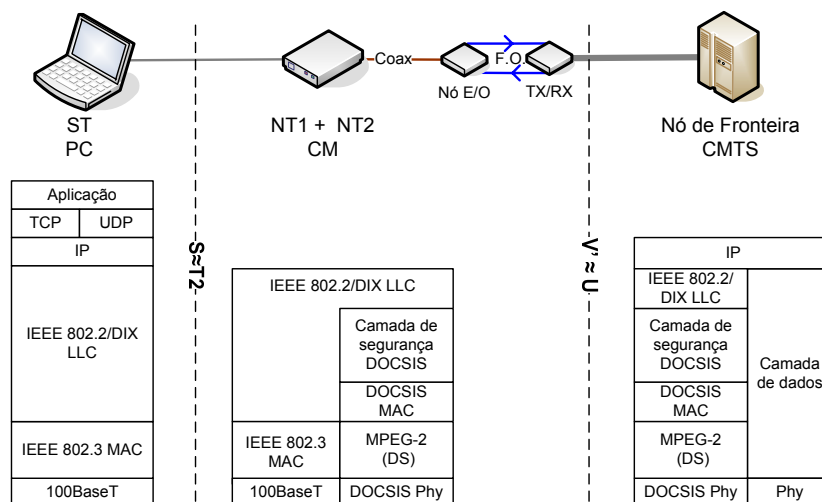


Figura 9: Pilhas protocolares dos elementos de uma rede DOCSIS

Como se pode constatar pela figura acima, o CM implementa as funcionalidades de uma *bridge* [22] com os datagramas IP a serem passados transparentemente entre a rede de acesso e a rede doméstica. Por sua vez o CMTS opera normalmente como um *router* IP e desempenha um papel fundamental não só no processo de autenticação, auto-configuração e gestão dos diversos CMs, mas também na transferência segura de dados entre o CMTS e os CMs.

À semelhança do que acontece na auto-configuração implementada nas redes xDSL, o processo de auto-configuração nas redes DOCSIS é faseado e em cada fase o CM vai adquirindo dados de configuração que lhe permitem passar à fase seguinte. Assim, quando o

CM é inicializado, começa por pesquisar em todos os canais (de 6MHz ou 8MHz) um canal que seja usado pelo CMTS para transmitir dados para os vários CMs (é também possível o canal a usar estar pré configurado no CM). Quando um desses canais é localizado, o CM espera o envio de mensagens específicas que o informam acerca dos canais e respectivos intervalos de tempo a serem usados para comunicar com o CMTS. Desta forma o CM, que acabou de se juntar à rede, encontra um intervalo de tempo de um dado canal ascendente que pode usar para se anunciar ao CMTS. Este anúncio gera uma troca de mensagens entre o CMTS e o CM, que permite ao CM receber informação de configuração relacionada com a camada física (ajustes de frequência de transmissão, potência de transmissão a usar, entre outros). No final desta troca de mensagens está concluída a configuração da camada física. Uma vez que não é necessário qualquer configuração na camada de dados, o CM inicia o processo de obtenção de conectividade IP. Para tal, um processo de DHCP é iniciado pelo CM através do qual obtém um endereço IP, o endereço IP de um servidor de TFTP [81], o nome do ficheiro de configuração que deverá ser descarregado do servidor de TFTP, o endereço IP do *default gateway*, o endereço IP de um servidor de ToD (*Time of Day*) [82], entre outros. É importante referir que uma vez que o CM opera em modo de *bridging*, o endereço IP atribuído ao CM apenas é usado para efectuar apenas tarefas de gestão e configuração.

Concluída a configuração da camada de rede, o CM estabelece uma ligação com um servidor de ToD afim de obter a data e hora correctas, seguida de uma ligação ao servidor de TFTP. Esta ligação ao servidor TFTP resulta no *download* de um ficheiro fundamental para a configuração do CM. Este ficheiro contém vários parâmetros que devem ser configurados no CM, tais como: frequência do novo canal em que o CM vai passar a receber tramas, frequência do novo canal que o CM deve usar para comunicar com o CMTS, número máximo de terminais conectados ao CM, parâmetros relacionados com a classe de serviço DOCSIS 1.0 em que o CM se encontra inserido (por exemplo ID da classe, máxima velocidade de transmissão descendente/ascendente, taxa de transmissão ascendente mínima garantida, entre outros) ou em alternativa (quando está implementada uma rede DOCSIS 1.1 ou 2.0) parâmetros relacionados com QoS (por exemplo a configuração dos classificadores de tráfego do CM). Opcionalmente podem também ser incluídos no ficheiro parâmetros relacionados com autenticação e segurança [73] [75] (por exemplo autenticação do CM, obtenção de chaves para cifra de tráfego, entre outros). Finalmente o ficheiro contém ainda MICs (*Message Integrity Check*) que permitem ao CM e ao CMTS verificarem a integridade do ficheiro descarregado.

Após o CM auto configurar-se de acordo com as variáveis contidas no ficheiro descarregado por TFTP, deve registar-se junto do CMTS. Para tal deve enviar para o CMTS uma mensagem com alguns dos parâmetros recebidos através do ficheiro, bem como os MICs contidos nesse mesmo ficheiro. Uma mensagem de sucesso é enviada pelo CMTS depois deste efectuar várias operações (sobre parâmetros recebidos) para confirmar o registo do CM. A partir do momento em que o CM recebe essa mensagem de sucesso está autorizado a enviar tráfego para a rede, a não ser que este tenha recebido através do ficheiro de configuração instruções para negociar chaves de cifra de tráfego com o CMTS e para se autenticar perante o CMTS (caso esteja implementada a norma (Euro)DOCSIS 1.1 ou 2.0). Nesta situação apenas após o CM estar autenticado e ter em sua posse as chaves de cifra de tráfego (obtidas após a troca de material criptográfico com CMTS, tal como especificado em [75]), é que este está autorizado a enviar tráfego proveniente da rede doméstica para a rede de acesso. Estas chaves de cifra (renovadas periodicamente) vão permitir ao CM decifrar o tráfego proveniente do CMTS endereçado a terminais da rede doméstica (ou ao próprio CM) e também cifrar todo o tráfego com destino ao CMTS.

Finalizado o processo de auto-configuração do CM, inicia-se o processo de configuração dos terminais, que consiste normalmente na obtenção de conectividade IP e na obtenção de informação adicional (por exemplo endereço IP do *default gateway*, endereço IP do servidor de DNS, entre outros) através do mecanismo DHCP. Uma vez concluída esta fase os utilizadores podem usufruir do serviço de acesso à Internet. Como referido em 2.2.3.1.2.2, muitos operadores utilizam o volume de tráfego gerado e/ou recebido no serviço de acesso à Internet para efectuarem a respectiva taxação. Na arquitectura DOCSIS esta informação é normalmente obtida através de consultas regulares a MIBs específicas implementadas no CM e cujos valores são registados para posterior processamento pelo sistema de taxação. Outros modelos de taxação mais complexos (por exemplo que tomem em conta várias classes de serviço) são normalmente baseados em medidas realizadas pelo CMTS.

Por sua vez a arquitectura PacketCable apesar de fazer uso de entidades DOCSIS define mecanismos de autenticação (autenticação mútua entre o MTA e as restantes entidades PacketCable), privacidade (diferentes cifras são usadas para a sinalização e para o tráfego de voz) e taxação (toda a informação utilizada em taxação é obtida unicamente de elementos de rede confiáveis que se situam exclusivamente na rede do operador) distintos dos especificados nas normas DOCSIS.

2.2.3.3. Redes de acesso baseadas em Ethernet

No modelo de referência apresentado na secção 2.1.2 é referido que o ponto de referência T2 apenas contempla interfaces compatíveis com a tecnologia *Ethernet*. Tal acontece porque actualmente esta tecnologia é a mais comum para a construção de LANs, quer em ambientes residenciais quer em ambientes comerciais. Assim, apesar de o tráfego proveniente das redes dos clientes se apresentar normalmente sobre a forma de pacotes IP transportados em tramas *Ethernet*, este é transportado ao longo da rede de acesso recorrendo a um conjunto de outros protocolos (por exemplo PPP, ATM, SONET, entre outros). Esta adição de protocolos aumenta o custo, a ineficiência e a complexidade na gestão, o que não acontece caso a tecnologia *Ethernet* seja estendida à rede do operador.

Assim, o IEEE e o MEF [43] formaram grupos de trabalho que visam promover a expansão da *Ethernet* para além das LANs. Enquanto o MEF pretende promover o uso da tecnologia junto dos operadores, criar modelos de serviço, entre outros, do trabalho do IEEE resultaram várias normas que procuram resolver limitações técnicas próprias de uma tecnologia originalmente desenvolvida para um ambiente de LAN. Com a norma IEEE802.3ah [83], o IEEE pretendeu ainda especificar o transporte de tramas *Ethernet* directamente sobre meios físicos característicos da primeira milha e até então não contemplados em normas IEEE802.3 (ver secção 2.2.3.3.4).

2.2.3.3.1 Arquitectura de uma rede de acesso Ethernet

Como já referido, as redes de acesso *Ethernet* pretendem também reutilizar meios físicos e topologias de rede comuns na primeira milha para o transporte directo de tramas *Ethernet* sobre esses mesmos meios físicos. A linha que dá conectividade física ao cliente pode estar conectada à rede do operador numa topologia ponto a ponto (por exemplo através de linhas de cobre, fibra óptica) ou numa topologia ponto multiponto (por exemplo EPON). A figura 10 apresenta a arquitectura de uma rede de acesso baseada em *Ethernet* em que ligações ponto a ponto são usadas entre a rede do operador e a rede doméstica.

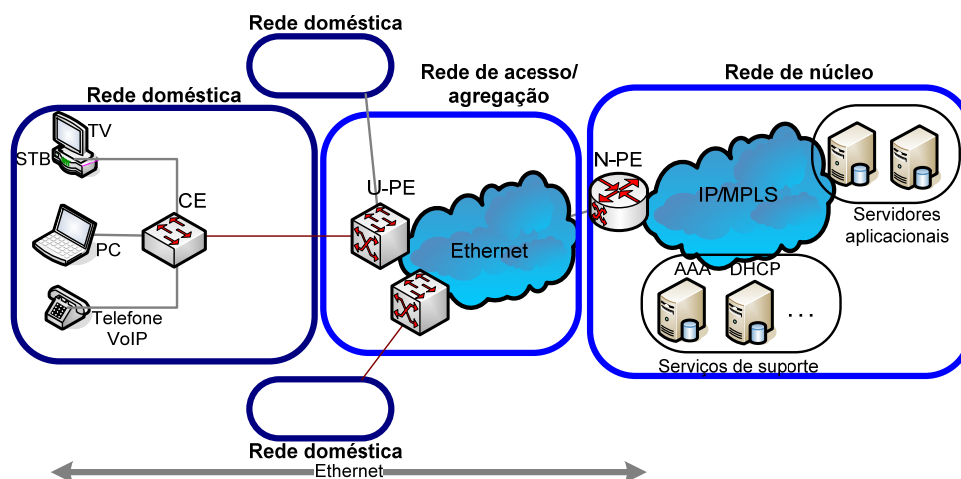


Figura 10: Arquitectura de uma rede de acesso baseada em *Ethernet*

Como se pode observar na figura acima, a arquitectura de rede é constituída essencialmente pelos elementos CE (*Customer Edge*), U-PE (*User-Provider Edge*) e N-PE (*Network Provider Edge*). Assim, os terminais do cliente recebem e enviam tramas *Ethernet* de e para a rede do operador através do CE, que funciona como ponto de interface entre a rede do operador e a rede doméstica. Este dispositivo pode desempenhar funções similares às de um *router* (neste caso o CE é denominado de CE-R (*Customer Edge Router*)) ou em alternativa pode ser um *switch Ethernet* (CE-S (*Customer Edge Switch*)). É importante referir que em ambos os casos as tramas que são enviadas para a rede do operador podem ser tramas *Ethernet* simples ou tramas com *tags* VLAN [84], sendo que neste último caso o CE poderá preencher informação acerca da prioridade das tramas (como especificado na adenda IEEE802.1p [28] à norma IEEE-802.1D [22]), acerca da VLAN (VID) a que as tramas pertencem ou acerca de ambas.

Por sua vez o dispositivo U-PE na sua essência é um *switch Ethernet*. Contudo, como este é o ponto de entrada para a rede do operador, funções adicionais são também desempenhadas neste elemento. Assim, para além de agregar o tráfego proveniente das várias redes domésticas, é também responsável por efectuar a identificação e marcação das tramas provenientes da primeira milha (nos seus vários formatos) antes destas serem encaminhadas para a rede do operador, isto para que seja possível a segregação e identificação do tráfego proveniente de cada cliente. Apesar da existência de outros métodos, a operação de marcação e classificação de tramas resulta normalmente na adição às tramas de uma *tag* S-VLAN (concatenada com uma possível *tag* C-VLAN já colocada pelo CE), tal como descrito na adenda IEEE 802.1ad [85] à norma IEEE802.1Q (método normalmente conhecido como Q-

in-Q). A informação contida na *tag* S-VLAN para além de possibilitar a segregação e identificação do tráfego de um dado cliente, pode também ser usada para inferir acerca do tratamento a nível de QoS que as tramas devem ter ao atravessar a rede do operador. Da mesma forma, as *tags* S-VLAN contidas nas tramas que chegam ao U-PE provenientes do N-PE, antes de serem removidas, são usadas por este para inferir acerca do seu destino (qual a rede doméstica a que se destinam) e acerca da sua ordem de envio. Um processo de aprendizagem de endereços MAC por parte de U-PE, como descrito em [22] e [84], pode também ser necessário para o encaminhamento de tramas. Finalmente, uma vez que o U-PE é o ponto de contacto com a rede do cliente este pode também desempenhar funções de policiamento, de CAC (*Call Admission Control*), filtragem de tramas e ainda funções relacionadas com a autenticação de utilizadores e/ou terminais, bem como com a auto-configuração destes últimos.

Por sua vez, a rede de agregação é normalmente constituída por um conjunto de dispositivos conectados de uma forma redundante, o que garante a robustez contra falhas de elementos de rede, requisito indispensável para uma rede de operador. Sendo uma rede *Ethernet* de grande dimensão a redundância e recuperação de falhas é conseguida através do STP (*Spanning Tree Protocol*) [86] [87]. Caso o operador o permita, a rede de agregação poderá também servir para o encaminhamento de tráfego local entre terminais em diferentes redes domésticas. O conjunto de CEs e toda a rede de agregação formam, o que é normalmente conhecido, como a rede metropolitana (neste caso como toda a tecnologia deste conjunto é *Ethernet*, dá-se o nome de rede metropolitana *Ethernet*).

Finalmente, o N-PE funciona como um ponto de demarcação entre uma rede totalmente baseada em *Ethernet* e uma rede que pode ser baseada em outras tecnologias. No caso apresentado na figura 10 a rede de transporte é baseada na tecnologia MPLS (*Multi-Protocol Label Switching*), pelo que as *tags* S-VLAN usadas na rede de agregação têm de ser removidas no N-PE e feita a tradução para LSPs (*Label Switched Paths*) MPLS. Para tramas que são enviadas da rede MPLS para a rede *Ethernet*, a operação inversa tem de ser efectuada (introdução de *tags* S-VLAN), o que normalmente requer que o N-PE seja também um ponto da rede onde os endereços MAC são aprendidos. À semelhança do que acontece com o U-PE, o N-PE é também interveniente na autenticação de utilizadores e/ou terminais bem como na auto-configuração destes últimos.

É importante referir que o facto de vários elementos de rede (U-PE, N-PE e principalmente

os *switches* da rede de agregação) terem de fazer a aprendizagem de todos os endereços MAC que pertencem a uma dada VLAN (o que no pior caso pode significar a aprendizagem dos endereços dos terminais das redes domésticas), pode conduzir à explosão das tabelas de endereços. Assim actualmente o IEEE está a trabalhar num mecanismo conhecido como Mac-in-Mac [88], que permite que as tramas que chegam ao U-PE sejam encapsuladas num novo cabeçalho cujos endereço MAC de origem e destino são respectivamente o endereço do U-PE e o endereço do ponto de saída da rede da trama. Desta forma nós intermédios apenas têm de conhecer dois endereços MAC para encaminharem a trama, o que representa uma melhoria significativa no tamanho das tabelas. Esta norma é compatível com a adenda IEEE802.1ad na medida em que mantém a possibilidade do uso da *tag* S-VLAN de 12 bits, mas permite também a expansão do número de VLANs possíveis através de um novo campo criado para este fim denominado de *iTag* (com um comprimento de 20 bits).

2.2.3.3.2 Serviços típicos

Uma das tarefas a que o MEF se propôs foi estabelecer normas para os serviços *Ethernet*, normas estas que são descritas do ponto de vista do CE, na medida em que a tecnologia e arquitectura da rede metropolitana são invisíveis para este. Definindo serviços desta forma a rede metropolitana pode evoluir independentemente da rede doméstica sem que haja uma interrupção nos serviços que um dado cliente subscreve actualmente. A definição de um serviço *Ethernet* assenta em dois conceitos fundamentais: o conceito de EVC (*Ethernet Virtual Connection*) e o conceito de UNI (*User Network Interface*). Este último é entendido como uma interface entre a rede do cliente (aqui o termo cliente refere-se tanto a um cliente residencial como a um provedor de serviços que usa os serviços *Ethernet* para fazer chegar os seus serviços aplicativos a clientes residenciais). Por sua vez o conceito de EVC é definido como a associação de dois ou mais UNIs (uma UNI pode pertencer a várias EVCs simultaneamente), sendo que no primeiro caso a EVC diz-se ser ponto a ponto e no segundo caso multiponto a multiponto. Uma EVC permite a transferência de tramas entre as UNIs que pertencem à EVC e evita que transferências de dados entre UNIs que não pertençam a uma mesma EVC sejam efectuadas, podendo portanto ser vista como um tipo de VPN (*Virtual Private Network*) implementada ao nível da camada 2. Numa EVC ponto a ponto, todas as tramas de serviço que partem de uma UNI são tipicamente entregues à outra UNI que pertence à EVC, porém, no caso de uma EVC multiponto a multiponto vários comportamentos podem ser verificados: tramas de um serviço *broadcast* ou *multicast*

tipicamente são replicadas e entregues às restantes UNIs da EVC. No caso de tramas de um serviço *unicast* estas podem ser replicadas e entregues a todas as UNIs que pertencem à EVC ou em alternativa os nós da rede podem “aprender” quais os endereços MAC que estão associados a cada UNI e entregar as tramas apenas às UNIs apropriadas. É importante referir, que uma EVC pode ser implementada numa rede *Ethernet* através do método Q-in-Q ou do método Mac-in-Mac ou em alternativa, quando é usada a tecnologia MPLS na rede de agregação, através de serviços VPLS (*Virtual Private LAN Service*) [89] [90].

Para especificar totalmente um serviço *Ethernet*, o operador para além de ter de definir o tipo de serviço (tipo de EVC), tem também de definir atributos associados às UNIs e atributos associados à EVC [91]. Estes atributos permitem especificar parâmetros como o meio físico sobre o qual as tramas vão ser trocadas, a lista de UNIs que pertencem à EVC, entre outros. A tabela 5 e a tabela 6, incluídas no Anexo II, apresentam os vários atributos definidos pelo MEF para a completa descrição de um serviço *Ethernet*. Após a definição do(s) serviço(s) *Ethernet* (definidos através de diferentes conjuntos de atributos) mais adequado(s) ao cliente, os mais variados serviços aplicativos podem ser disponibilizados ao cliente, desde que os dados referentes a esses mesmos serviços aplicativos sejam passíveis de serem transportados sob a forma de tramas *Ethernet*. Assim, qualquer um dos serviços descritos nas secções 2.2.3.1.2 e 2.2.3.2.3 que se encontrem nestas condições podem ser fornecidos a clientes através de uma rede de acesso *Ethernet*.

2.2.3.3.3 Gestão e controlo de serviços

Uma das vantagens em se estender a tecnologia *Ethernet* para além das LANs consiste no facto da sua característica *plug and play* poder continuar a ser utilizada. Desta forma, após estabelecida a conectividade ao nível da camada física, a conectividade ao nível da camada de dados é automaticamente estabelecida como especificado em [30], sendo portanto possível ao CE a partir desse momento comunicar com o U-PE. Todavia, como referido na secção anterior, um CE (com a respectiva UNI associada) apenas pode trocar tramas com outros CEs se existir um serviço *Ethernet* que o permita. Assim, com o objectivo de automatizar o processo de configuração do CE o MEF baseou-se no protocolo FR-LMI (*Frame Relay Local Management Interface*) [92] para desenvolver o protocolo E-LMI (*Ethernet Local Management Interface*) [93]. De uma maneira simples, este protocolo permite ao CE requerer e receber do operador informação (atributos UNI e EVC, instruções para a criação e destruição de EVCs,

entre outros) que possibilita que este auto configure um ou mais serviços *Ethernet* e assim possa enviar e receber tramas.

No entanto, um CE só deve receber a referida informação de configuração se estiver devidamente autorizado a tal. Ao contrário do que acontece num ambiente LAN, onde a autorização de acesso à rede (e da utilização dos recursos desta) por parte de um terminal está muitas vezes implícita no acesso físico que este tem (ou não) à infra-estrutura de rede, quando se pretende estender a tecnologia *Ethernet* à rede de acesso um mecanismo de autenticação deve ser implementado. O protocolo IEEE 802.1X [94] pode ser usado pelo operador para adicionar a capacidade de autenticação a uma rede de acesso baseada em *Ethernet*, sendo que o CE será a entidade IEEE 802.1X suplicante, o U-PE a entidade IEEE802.1X autenticador e o servidor de autenticação estará localizado algures na rede do operador. Este processo de autenticação para além de servir como controlo de acesso à rede, pode também ser usado pela rede para inferir acerca das configurações que devem ser transmitidas ao CE. É importante referir que este nível de autenticação apenas permite ao operador controlar o acesso por parte de um dado cliente (CE) a serviços *Ethernet* e não a serviços aplicacionais. O controlo de acesso a serviços aplicacionais, pode ser feito por exemplo se houver uma correspondência directa entre um dado serviço aplicacional e um dado serviço *Ethernet* (por exemplo um dado serviço aplicacional ser transportado sempre numa mesma EVC). Este mesmo princípio, pode ser usado pelo operador para o processo de taxação dos vários serviços aplicacionais que o cliente subscreve.

A figura 11 apresenta a pilha protocolar de uma rede de acesso baseada em *Ethernet*.

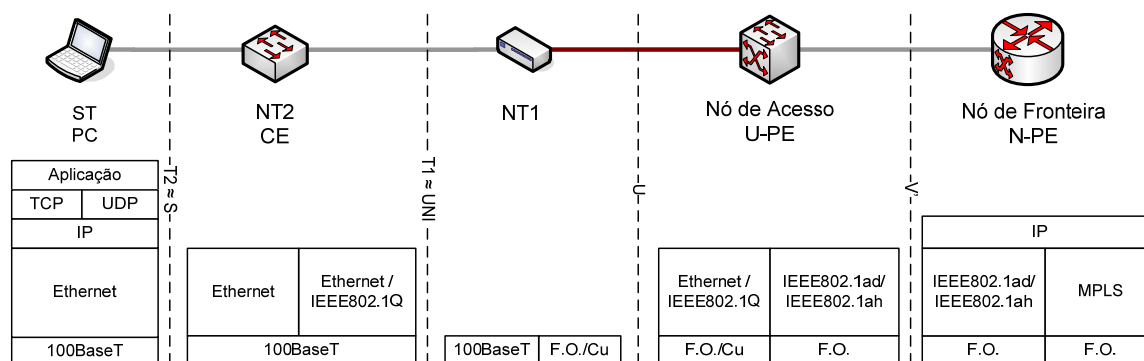


Figura 11: Pilhas protocolares dos elementos de uma rede de acesso baseada em *Ethernet*

Devido ao facto de a tecnologia *Ethernet* se estender por toda a rede de acesso, o N-PE e o terminal parecem estar no mesmo troço de LAN. Assim, após o CE se auto configurar, o terminal pode também auto configurar-se (por exemplo para aceder à Internet) usando o

mecanismo DHCP, como acontece normalmente nas redes locais.

O elemento NT1 (não representado na figura 10 por questões de simplicidade), faz a adaptação de tecnologias ao nível físico. Uma situação em que este elemento precisa de fazer a adaptação de tecnologias acontece quando uma EPON (*Ethernet Passive Optical Network*) é usada na primeira milha: o elemento NT1 é então responsável por fazer a ponte entre o domínio óptico e o domínio eléctrico.

2.2.3.3.4 As normas Ethernet

A família de normas IEEE que podem ser usadas na primeira milha de uma rede de acesso com a arquitectura apresentada ao longo deste capítulo é vasta. A figura 12 mostra a relação entre algumas dessas normas, bem como taxas de transmissão características e distâncias máximas para as quais essas taxas são válidas.

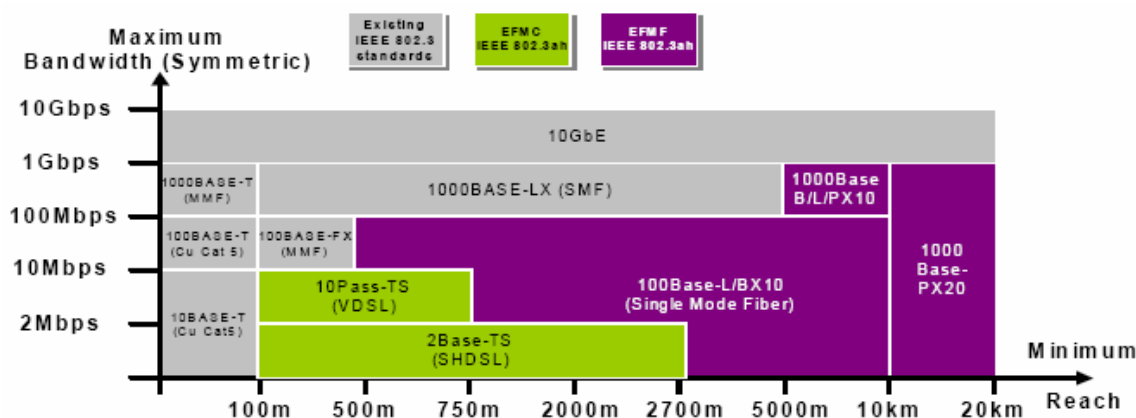


Figura 12: Normas *Ethernet* [95]

As normas 10Base-T e 100Base-T para o cobre e as normas 1000Base-T e 1000Base-LX para a fibra óptica, foram no passado responsáveis pela generalização da tecnologia *Ethernet* nas redes locais. Quando, nos últimos anos, a tecnologia *Ethernet* começou a ser apontada como uma possível solução para resolver o estrangulamento de largura de banda que existe na primeira milha da rede de acesso, o IEEE iniciou o desenvolvimento de um novo conjunto de normas para esse fim, cujo resultado foi a norma IEEE802.3ah. Esta, especifica novas formas de transportar tramas *Ethernet* sobre fibra óptica, EFMF (*Ethernet in the First Mile over Fiber*) e cabos de cobre, EFMC (*Ethernet in the First Mile over Copper*).

As novas normas EFMC introduzem o transporte directo de tramas *Ethernet* sobre o par de cobre usada pelo serviço POTS e ao nível da camada física reutilizam técnicas de modulação

DSL, nomeadamente as técnicas empregues nas tecnologias VDSL e SHDSL.

Por sua vez, as normas EFMF podem ser divididas em duas categorias: as normas que contemplam o transporte de tramas *Ethernet* em ligações de fibra óptica ponto a ponto, onde estão incluídas as normas 100Base-LX10, 1000Base-LX10, 100Base-BX10 e 1000Base-BX10 (as normas “LX” prevêem o uso de duas fibras ópticas, enquanto que as “BX” especificam o uso de uma só fibra óptica) e as que contemplam o transporte de tramas sobre PONs (*Passive Optical Networks*), onde estão incluídas as normas 1000Base-PX10 e 1000Base-PX20.

É importante referir que as diferentes normas (e topologias) aqui apresentadas são complementares e não concorrentes. As soluções sobre cobre (2Base-TS e 10Pass-TS) ajustam-se melhor a zonas residenciais e parques comerciais onde já existem pares de cobre usados para o serviço POTS, enquanto que soluções sobre fibra poderão ser mais adequadas a novas zonas comerciais e residenciais onde a fibra pode ser instalada no interior e exterior dos edifícios. Em alguns casos a combinação de tecnologias pode existir, dando origem a topologias do tipo FTTB (*Fiber To The Building*) ou FTTC (*Fiber To The Curb*), onde fibra óptica é estendida até à entrada de um edifício ou bairro residencial e daí a distribuição é feita usando cabos de cobre (por exemplo 100Base-T).

2.2.3.4. Outras redes de acesso

A secção 2.2.3 desta dissertação pretende dar uma visão geral da arquitectura e funcionamento das principais redes de acesso actualmente utilizadas pelos operadores de telecomunicações para fornecerem aos seus clientes residenciais os mais variados tipos de serviços. Obviamente que o leque de soluções possíveis não se esgota nas três arquitecturas aqui descritas. Contudo, as aqui apresentadas são representativas do que acontece actualmente no mercado e ambiente residencial.

Assim, redes de acesso baseadas em tecnologias sem fios como as tecnologias UMTS (*Universal Mobile Telecommunications System*) ou IEEE802.11x cujo âmbito de aplicação centra-se normalmente no terminal, não estão enquadradas com os objectivos desta dissertação. Ao contrário destas, a tecnologia PLC (*Power Line Communications*) começa agora a ganhar alguma importância no mercado dos acessos residenciais de banda larga. Porém, devido ao facto de as redes de acesso baseadas nesta tecnologia apresentarem desafios (e soluções) semelhantes aos apresentados em redes de acesso já descritas na presente dissertação (nomeadamente nas redes de acesso CATV), aparte das diferenças inerentes ao uso de meios físicos totalmente distintos,

a sua inclusão e descrição não traria qualquer valor acrescentado à presente dissertação.

2.2.4. Redes residenciais

A necessidade de partilhar o serviço de acesso à Internet em banda larga e outros recursos (por exemplo dados, *hardware*, etc.) entre vários terminais/utilizadores, a maturidade de protocolos de comunicação como o protocolo IP e o aparecimento de tecnologias especificamente concebidas para a transferência de dados em espaços reduzidos (por exemplo WiFi, HomePlug, HomePNA) têm sido os principais impulsionadores da evolução das redes domésticas.

Assim, passou-se de um cenário comum há alguns anos atrás, onde uma rede doméstica limitava-se a um simples modem ligado a um só computador através de um cabo USB, para uma rede doméstica mais complexa como a representada na figura 13. Esta configuração é eventualmente a mais comum nos dias de hoje e permite a partilha do serviço de acesso à Internet, bem como de outros recursos entre vários terminais.

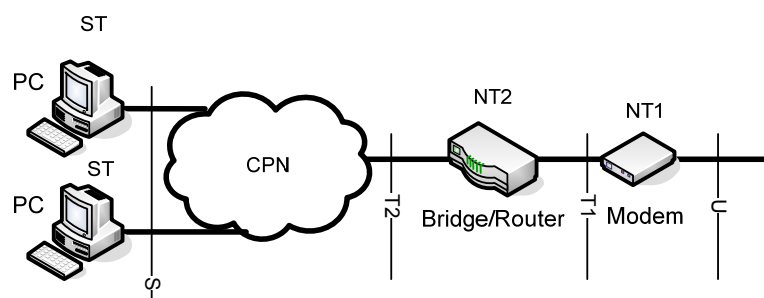


Figura 13: Partilha do serviço de acesso à Internet de banda larga. Baseado na figura 13 de [16]

Tipicamente o bloco NT2 desempenha a função de uma *bridge* IEEE802.1D ou a função de um *router*. O segundo caso é normalmente o preferido dos operadores, visto que o bloco NT2 para além de um *router* implementa também a funcionalidade NAPT (o que permite que os terminais na rede domésticas utilizem endereços IP privados). Assim, o bloco NT2 é responsável pelo encaminhamento ao nível da camada IP do tráfego entre terminais domésticos e do tráfego entre a rede doméstica e a rede do operador. Da rede do operador, este bloco recebe dados de configuração referentes à camada IP e ao serviço de acesso à Internet (por exemplo endereço IP do servidor de DNS) que este usa para se auto-configurar, tal como acontece na situação em que um terminal está directamente ligado ao modem.

Na figura acima, os blocos funcionais NT1 e NT2 estão implementados em dispositivos

físicos distintos e uma ligação *Ethernet* 10/100 BaseT é usada para ligar os dois dispositivos. Todavia, é também possível que ambos os blocos sejam implementados num único elemento físico, habitualmente denominado de *access gateway* - *gateway* de acesso.

A ligação física dos vários terminais ao bloco NT2 é feita através de uma rede local (representada pelo bloco CPN). Ao contrário do que acontece num ambiente empresarial, onde a tecnologia *Ethernet* é normalmente usada na construção de redes locais, num ambiente doméstico, devido aos requisitos muito próprios que este apresenta, a rede local pode ser constituída por um variado número de tecnologias, algumas das quais são brevemente descritas a seguir.

2.2.4.1. 10 Base-T, 100 Base-T

Como já referido na 2.2.3.3.4, a tecnologia mais utilizada actualmente em redes locais de computadores, é sem duvida a *Ethernet*. A tecnologia *Ethernet* encontra-se normalizada essencialmente em normas IEEE 802.3, nas quais são especificadas as camadas física e de dados referentes ao modelo OSI.

As normas 10Base-T e 100Base-T definem a implementação da tecnologia *Ethernet* sobre um cabo que contem dois ou mais pares de cobre entrançados e com as quais se conseguem taxas de transferência máximas de 10 Mb/s e 100 Mb/s respectivamente. Infra-estruturas de rede actuais baseadas nestas duas normas assentam normalmente numa distribuição estruturada de cabos *UTP category 5* entre um ponto central (que normalmente coincide com o local onde é feita a terminação do serviço de telecomunicações) e os vários terminais e dispositivos de rede (por exemplo *hubs*, *switches*, *routers*) a que se pretende fornecer conectividade.

2.2.4.2. IEEE 802.11 a/b/g

O conjunto de normas 802.11, desenvolvidas no âmbito do IEEE, apresenta especificações de interfaces sem fios entre um terminal e uma estação base ou, em alternativa, entre duas estações. De entre as várias normas desenvolvidas neste grupo de trabalho, as normas IEEE802.11 b/g são as que actualmente se encontram mais difundidas. Ambas as tecnologias operam na zona dos 2.4GHz, usam CSMA/CA como método de acesso ao meio e são inter operáveis. No entanto, devido ao facto de a tecnologia 802.11g alternar entre várias técnicas de modulação consoante as condições do meio (ao contrário da tecnologia 802.11b), maiores

taxas de transmissão são conseguidas com esta (uma taxa de transmissão máxima de 54 Mbps face aos 11Mbps da tecnologia IEEE802.11b).

Paralelamente à tecnologia 802.11b foi também desenvolvida a tecnologia 802.11a, com a qual se conseguem taxas de transmissão na ordem das conseguidas com a tecnologia 802.11g. Contudo, devido ao facto desta tecnologia operar nos 5GHz, taxas dessa ordem de grandeza apenas são conseguidas quando o receptor e o emissor se encontram em linha de vista, não existindo nenhum obstáculo entre estes. Por este motivo e pelo facto desta tecnologia não ser inter operável com as duas anteriores, a tecnologia 802.11a é actualmente preterida face às tecnologias 802.11 b/g.

2.2.4.3. HomePNA

O uso da cablagem telefónica que se encontra instalada em muitas das residências actuais para a comunicação entre computadores e outros dispositivos digitais, tem vindo a ser especificado por uma associação denominada de *Home Phoneline Network Alliance*. Essa proposta foi mais tarde adoptada pelo organismo ITU nas recomendações G.989.1, G.989.2, G.989.3 e G.9954. Existem actualmente quatro normas referentes a esta tecnologia, denominadas HomePNA 1.0, 2.0, 3.0 e 3.1, sendo que todas elas especificam um comportamento da camada de dados equivalente ao da tecnologia IEEE802.3, nomeadamente o uso de tramas compatíveis com as tramas *Ethernet* e o uso de um método de acesso ao meio físico baseado em CSMA/CD. Por este motivo pode-se classificar a tecnologia como “compatível” com a tecnologia *Ethernet*. As normas HomePNA definem o intervalo de frequência entre os 4 MHz e os 10 MHz como a gama a ser usada, não havendo assim qualquer interferência entre a tecnologia HomePNA e as tecnologias POTS e xDSL, tal como mostrado na figura seguinte:

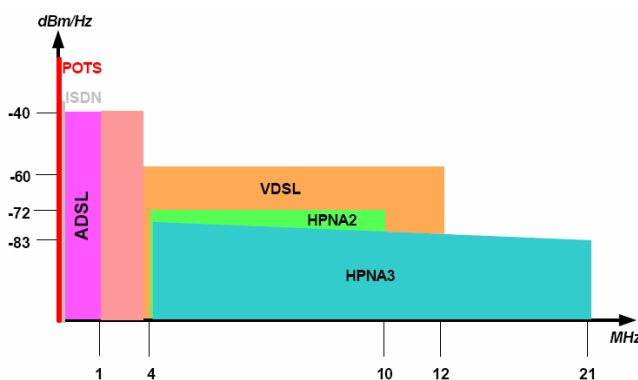


Figura 14: Espectro das tecnologias POTS, RDIS, xDSL e HPNA 1.0, 2.0 e 3.0 [35].

A versão 1.0 da norma HomePNA propôs o uso do formato de modulação PPM (*Pulse Position*

Modulation), o que se traduzia em taxas de transmissão de apenas 1 Mbps. Devido ao acelerado aumento da largura de banda disponibilizada aos utilizadores pelos operadores, rapidamente se verificou que esta taxa de transmissão era claramente insuficiente, o que resultou na publicação de uma segunda norma.

Com a norma HomePNA 2.0, à custa de técnicas de modulação mais complexas e de receptores que se adaptam às condições do canal em cada instante, são conseguidas taxas de transmissão máximas de 32 Mbps. A segunda norma introduz também um mecanismo de acesso ao meio baseado em prioridades (capaz de fornecer QoS relativa). Posteriormente à edição da norma 2.0, uma extensão nomeada de VoHPNA foi editada com o objectivo de garantir QoS em serviços de voz sobre a tecnologia HomePNA.

A versão 3.0 da norma, editada em 2003, além de definir mecanismos que permitem taxas de transmissão máximas de 128 Mbps, define também mecanismos de garantia de QoS (em oposição à versão 2.0 que apenas fornecia QoS relativa, baseado num esquema de prioridades) que permitem à tecnologia operar num cenário de *triple-play*. Uma outra inovação apresentada é o facto da tecnologia poder agora utilizar também a instalação de cabo coaxial para efectuar a transmissão de dados.

Finalmente, em Novembro de 2006, foi lançada a norma HomePNA 3.1 cuja principal inovação, em relação à versão 3.0, é uma taxa de transmissão máxima de 320Mbps.

2.2.4.4. HomePlug

As normas HomePlug especificam como criar uma LAN residencial usando como meio físico de transmissão a instalação eléctrica presente nessa mesma residência. Esta tecnologia, à semelhança da tecnologia HomePNA, foi definida para funcionar com tramas IEEE802.3 e portanto é também considerada “compatível” com a tecnologia *Ethernet*. Existem actualmente três normas referentes a esta tecnologia.

As normas 1.0 e *Turbo* utilizam uma técnica de acesso ao meio de transmissão que é uma variante da técnica CSMA/CA, o que permite um acesso ao meio de transmissão baseado na prioridade da trama que cada um dos nós pretende transmitir. Essa prioridade pode ser inferida por exemplo de tramas IEEE802.1Q, tal como especificado na norma IEEE802.1D. Ao nível da camada física a tecnologia funciona de uma forma adaptativa, ou seja, de acordo com as condições do canal de transmissão, diferentes técnicas de modulação são usadas, o

número de portadoras (que se situam na banda dos 4.5MHz aos 21MHz) varia e o FEC (*Forward Error Correction*) em uso também. Esta adaptação constante traduz-se numa taxa de transmissão máxima de 14 Mbps e 85Mbps para as versões 1.0 e *Turbo* respectivamente.

Por sua vez a norma HomePlug AV permite o uso desta tecnologia num ambiente *Triple-Play*. Para além de uma taxa de transmissão máxima na ordem dos 200Mbps, conseguida através do alargamento da gama de frequências usada (dos 2MHz aos 30MHz) e de técnicas de modulação mais complexas, esta versão implementa ainda mecanismos de garantia de QoS (requisito fundamental para a correcta operação num cenário *Triple-Play*). A norma prevê também a privacidade dos dados transmitidos em duas redes distintas mas que partilham uma mesma instalação eléctrica aplicando uma cifra AES (*Advanced Encryption Algorithm*) [96] de 128 bits a todos os dados transmitidos. Ao nível físico, mecanismos que evitam a radiação de potência de um adaptador para um outro que pertença a uma rede distinta garantem também isolamento e não interferência entre as duas redes.

A tabela 7, presente no Anexo II apresenta uma comparação entre as várias tecnologias no que diz respeito a taxas de transmissão, alcance (quando aplicável) e suporte de QoS. Importa referir que a eleição de uma dessas soluções como a preferida para um determinado utilizador depende essencialmente das necessidades de aplicação e do ambiente doméstico onde a solução vai ser aplicada.

2.3. Observações finais

A arquitectura xDSL permite o transporte de datagramas IP sobre uma infra-estrutura de rede inicialmente instalada e dimensionada unicamente para a disponibilização de um serviço de voz. Tal é conseguido à custa da introdução de várias camadas protocolares de adaptação que permitem o transporte de datagramas IP sobre essa infra-estrutura de rede, aproveitando assim os recursos disponíveis. Os serviços tipicamente disponibilizados a clientes residenciais através da tecnologia xDSL, centram-se essencialmente na telefonia e acesso à Internet em banda larga, sendo que os modelos de transporte, gestão, aprovisionamento, auto configuração, autorização, taxação, entre outros, são totalmente distintos para os dois tipos serviços.

À semelhança do que acontece nas redes xDSL, as redes DOCSIS permitem o transporte de pacotes IP sobre uma infra-estrutura de rede inicialmente instalada e dimensionada para outra

finalidade. Assim, a arquitectura DOCSIS permite o transporte de datagramas IP ao longo de uma rede HFC primariamente dimensionada para a difusão de canais de TV. Os serviços tipicamente disponibilizados, centram-se na difusão de canais de TV analógicos e/ou digitais, no serviço de telefonia IP e no serviço de acesso à Internet. Como acontece com o serviço de telefonia nas redes xDSL, diferentes modelos de transporte, gestão, aprovisionamento, autorização, auto-configuração, taxaço, entre outros, são aplicados ao serviço de difusão de TV. Por sua vez os serviços de telefonia IP e de acesso à Internet partilham modelos semelhantes.

Ao contrário das redes anteriores, as redes de acesso baseadas em *Ethernet* usam uma infraestrutura de rede totalmente dimensionada e concebida para o transporte de dados. Modelos de transporte, gestão, auto-configuração, aprovisionamento, autorização, taxaço, entre outros, são comuns a todos os serviços que são distribuídos através da rede. O facto de implementarem funcionalidades como *multicast*, suporte de QoS, entre outras, que nem sempre estão disponíveis nas redes xDSL e DOCSIS, tornam-nas uma solução ideal para a distribuição dos mais variados tipos de serviços (redes multi-serviço). Porém, uma vez que na maior parte dos casos a implementação deste tipo de redes requer um elevado investimento por parte dos operadores, prevê-se que as arquitecturas xDSL e DOCSIS continuem a ser sucessivamente melhoradas, tornando-se estas também redes multi-serviço. Outras limitações das redes de acesso *Ethernet* devem-se ao facto de basearem o seu funcionamento em protocolos e mecanismos desenvolvidos para a rede local. Escalabilidade, segurança e resiliência são algumas dessas limitações que se estão a abordar pela evolução dos protocolos e mecanismos já existentes.

Uma nota importante no que diz respeito ao modelo de prestação serviços é o facto de este assentar normalmente na exclusividade que um dado operador de rede tem sobre os seus clientes. Isto significa que um dado operador de rede é também o único prestador de serviços a que os clientes desse mesmo operador podem ter acesso, o que limita largamente o leque de serviços a que o cliente tem acesso.

Capítulo 3 – Plataformas de serviços residenciais

No capítulo anterior, após uma descrição e análise das principais redes de acesso usadas em ambientes residenciais, concluiu-se que existe actualmente alguma disparidade entre as capacidades e características de cada uma delas. Devido ao constante aumento da largura de banda que o cliente tem ao seu dispor e ao constante aparecimento de novos serviços de voz, dados e vídeo, algumas das redes de acesso têm sofrido e continuarão a sofrer um processo de evolução. Este processo visa essencialmente preparar as redes de acesso (ou pelo menos as que ainda não o estão) para a distribuição destes novos serviços, tornando-as redes multi-serviço. Da mesma forma, uma vez que a terminação destes serviços é efectuada na rede doméstica, esta tem também vindo a sofrer evoluções.

A primeira parte deste capítulo centra-se nestes processos de evolução de que as redes de acesso e domésticas estão a ser alvo, bem como nos requisitos que estas têm de preencher. Estes requisitos são essencialmente inferidos de uma lista de serviços residenciais, apresentada também neste capítulo, que serão disponibilizados num futuro próximo a clientes residenciais.

A segunda parte do capítulo centra-se em plataformas de serviços residenciais. Para além de uma breve descrição das plataformas de serviços mais comuns actualmente, são ainda apresentadas soluções que permitem a implementação de uma plataforma multi-serviço para ambientes residenciais.

3.1. Serviços residenciais em redes de próxima geração

Esta primeira secção pretende identificar algumas das características que as redes de acesso deverão apresentar no final do referido processo de evolução e que as tornarão adequadas para o transporte de múltiplos serviços. Para tal, são primeiro identificados alguns serviços tipo, a partir dos quais são inferidos os requisitos de rede necessários para o seu transporte. No final da secção são ainda apresentados desafios (e algumas soluções) que a distribuição destes novos serviços impõem às redes domésticas.

3.1.1. Serviços residenciais futuros

Esta secção apresenta uma lista de serviços e aplicações e respectiva descrição que serão disponibilizados (na verdade alguns estão já disponíveis) a clientes residenciais num futuro próximo. Esta lista não pretende ser exaustiva, pretende sim apresentar serviços que se consideram representativos dos vários tipos possíveis.

3.1.1.1. Telefonia

Os serviços de telefonia POTS e RDIS (já descritos na secção 2.2.3.1.2.1) são provavelmente os serviços telefonia mais comuns e são caracterizados por utilizarem redes próprias e muito específicas. Ao invés destes, os serviços de telefonia sobre IP têm vindo a ganhar popularidade essencialmente pelo facto de poderem ser implementados sobre uma qualquer rede que suporte o protocolo IP. Na secção 2.2.3.2.3.1 foi descrita a implementação deste serviço sobre as redes de acesso DOCSIS. Todavia, existem outras implementações mais comuns nomeadamente as que utilizam o protocolo de sinalização SIP [24].

Em ambos os casos, como já referido na secção 2.2.3.1.2.1, este serviço permite chamadas de voz entre dois terminais, sendo que outros serviços complementares ao serviço de telefonia podem também ser oferecidos.

3.1.1.2. Aplicações baseadas em TV

Esta secção descreve alguns serviços que podem ser oferecidos ao utilizador num receptor de TV.

3.1.1.2.1 Difusão digital de canais de TV e áudio

Este serviço é baseado no tradicional serviço de difusão de vários canais de TV (ou outros conteúdos audiovisuais *live*) recepcionados e descodificados num televisor. Na secção 2.2.3.2.3.3 é descrito um serviço de difusão de canais de TV sobre uma rede CATV, em formato digital e/ou analógico. No entanto, existem outras arquitecturas que permitem a transmissão de tramas MPEG-2 sobre uma qualquer rede de comutação de pacotes IP, normalmente conhecidas como arquitecturas IPTV. Devido ao facto de os conteúdos audiovisuais serem transmitidos num formato digital é necessária existência na residência do cliente de uma STB IP que converte os dados referentes a conteúdos audiovisuais em sinais analógicos passíveis de serem apresentados num receptor de TV comum.

Conjuntamente com conteúdos audiovisuais podem também ser transmitidos dados de outros serviços, como o serviço de guia de programas electrónico. Este serviço permite ao utilizador, por exemplo, consultar a programação de um dado canal, obter a informação de um dado programa, entre outros.

O serviço de difusão de áudio é semelhante ao serviço de difusão de TV, sendo que em vez de transmissão de conteúdos audiovisuais são apenas transmitidos conteúdos áudio *live* (por exemplo rádio). Os conteúdos áudio são primeiro codificados, através de uma das várias normas disponíveis (por exemplo MPEG Layer-3 [97]) e de seguida são transmitidos ao longo de uma rede de dados sobre a forma de pacotes IP até à residência do cliente, onde são descodificados e apresentados.

3.1.1.2.2 Vídeo/áudio on demand

O serviço descrito na secção anterior é caracterizado pelo facto de os conteúdos apresentados serem chamados de conteúdos *live*. O serviço de VoD (*Video on Demand*) difere neste ponto do serviço de difusão, sendo que no serviço de VoD é permitido ao cliente visualizar um dado conteúdo à sua escolha no instante que desejar. A escolha dos conteúdos que são possíveis visualizar num dado instante está obviamente limitada a uma biblioteca de conteúdos áudio/vídeo. Este serviço disponibiliza normalmente a possibilidade de actuar sobre a apresentação do conteúdo audiovisual através de controlos como *play*, *pause*, *fast forward* e *fast rewind*.

O serviço de *audio on demand* restringe os conteúdos transmitidos a conteúdos áudio.

3.1.1.2.3 Personal video recording

Este serviço pode ser descrito como um serviço de *Video on Demand* mas com capacidade de memorizar conteúdos, sendo que a origem dos conteúdos é normalmente o serviço de difusão de TV/áudio (por exemplo um programa de TV que o utilizador não pode ver em directo). Essa capacidade de memorização de conteúdos pode estar localizada em vários pontos da rede:

- No decodificador (STB) do cliente (equipado com disco rígido): neste caso o conteúdo é enviado, a pedido, para a STB do cliente e gravado no seu disco rígido.
- Na rede: nesta situação o conteúdo é enviado, a pedido, para um dado servidor na rede. O cliente pode depois visualizar o conteúdo através de um processo semelhante ao que acontece no serviço de VoD.

3.1.1.2.4 TV interactiva

A TV interactiva dá a possibilidade ao utilizador de combinar e usufruir de diferentes serviços, para além dos serviços de TV típicos, no receptor de TV. Isto é conseguido com a convergência entre os serviços de vídeo e outros serviços de dados. Alguns exemplos são: *web browsing* na TV, visualização de informação relativa ao serviço de telefonia na TV (por exemplo identificação de chamador, acesso à lista de chamadas efectuadas), serviço de notificação (por exemplo notificação de chegada de mensagens de *email*, notificação do início de um dado programa de TV). Uma outra vertente deste serviço é a possibilidade de o utilizador conseguir, de alguma forma, personalizar os conteúdos de que está a usufruir. A escolha do ângulo de visualização de uma partida de futebol, a participação numa votação acerca de qual o programa que será transmitido de seguida, entre outros, são exemplos desta outra vertente da interactividade. A existência de interactividade requer, normalmente, um canal de retorno da rede doméstica para o provedor de serviços.

3.1.1.3. Serviços baseados em PC

Esta secção descreve alguns serviços que podem ser oferecidos ao cliente que possua um computador pessoal. É importante referir que alguns dos serviços descritos na secção anterior, nomeadamente os serviços de difusão de TV, áudio e o serviço de vídeo/áudio *on demand*, podem também ser visualizados num computador pessoal, contudo, estes não serão de novo aqui descritos. Para além destes, o serviço de acesso à Internet, já descrito na secção 2.2.3.2.3.2, também não será descrito.

3.1.1.3.1 Vídeo telefonia

Este serviço possibilita uma conversação bidireccional entre dois utilizadores que para além da voz podem também usufruir de uma componente vídeo. O serviço poderá também ser oferecido ao cliente na TV, sendo que para isso é normalmente disponibilizada ao cliente uma STB específica para este serviço.

3.1.1.3.2 Jogos multi jogador on-line

Este serviço proporciona a vários utilizadores jogarem simultaneamente um mesmo jogo, utilizadores estes que podem estar geograficamente dispersos. Tradicionalmente este serviço utiliza uma arquitectura de comunicação do tipo cliente-servidor, arquitectura esta que apresenta algumas vantagens: uma única entidade central pode controlar todo o ambiente de jogo, os servidores recebem manutenção profissional, os servidores apresentam-se ao utilizador como entidades confiáveis, entre outras. Porém, a falta de escalabilidade, a quantidade de largura de banda e de capacidade de processamento que este modelo exige tem levado ao aparecimento de modelos baseados em arquitecturas *peer-to-peer*.

3.1.1.4. Domótica

A domótica refere-se à aplicação de técnicas de automação de edifícios a residências privadas com o objectivo de proporcionar maior conforto, segurança e economia aos habitantes das mesmas. Ao contrário dos serviços anteriores, que são destinados a um dado tipo terminal (TV ou PC), os serviços de domótica endereçam normalmente vários tipos de utilidades domésticas (por exemplo electrodomésticos, iluminação, sistemas de segurança, sistemas de climatização, entre outras), sendo que a comunicação entre estas e o elemento de rede onde reside a inteligência do serviço é conseguida através da rede doméstica e/ou da rede de acesso. Serviços como o controlo remoto de termóstatos por parte da companhia de energia possibilitando o deslocamento do consumo de energia para as horas de menor carga, controlo remoto de electrodomésticos por parte do utilizador, monitorização de sistemas de alarme e vigilância, são alguns dos serviços mais comuns na área da domótica.

3.1.1.5. Aplicações médicas

A área da saúde tem vindo a sofrer alterações importantes com o aparecimento de sensores e outros dispositivos inteligentes, ligações à Internet em banda larga e aplicações multimédia. Sensores inteligentes e com capacidades de comunicação podem agora ser usados por pacientes para desempenharem tarefas como a monitorização remota de parâmetros médicos

(por exemplo pressão arterial, ritmo cardíaco) ou a administração automática de medicação. Por sua vez a generalização de acessos de banda larga começa agora a possibilitar a integração destes sensores/dispositivos autónomos numa rede médica, que providencia o armazenamento dos dados recolhidos em repositórios seguros, a transmissão em tempo real das medições efectuadas e até o próprio controlo remoto desses mesmos sensores e dispositivos.

Este conjunto de factores têm vindo a gerar um novo conjunto de serviços que se podem classificar em três grupos:

3.1.1.5.1 Serviços de diagnósticos remotos

Este conjunto de serviços refere-se à possibilidade de um paciente ser diagnosticado acerca do seu estado de saúde sem estar na presença física de um médico. Tal é conseguido através da partilha de dados médicos (por exemplo imagens de alta resolução referentes a exames médicos), de medições efectuadas em tempo real ao paciente e de dados médicos estáticos (por exemplo historial médico do paciente). Este tipo de serviços possibilita a omnipresença de conhecimento médico especializado, reduz consideravelmente o tempo de diagnóstico e reduz também os custos normalmente associados a estes processos.

3.1.1.5.2 Serviços de monitorização remota

Estes serviços referem-se à colecta contínua de dados por sensores médicos usados pelo paciente e à análise destes dados para a identificação de condições de alarme. Este tipo de funcionalidade começou a ser primeiro usado em ambientes hospitalares com equipamento de monitorização normal. Todavia, o aparecimento de sensores de dimensão reduzida com capacidades de comunicação e a omnipresença de redes de comunicação que podem ser usadas por estes para transmitirem os dados referentes às suas leituras, tem gerado o aparecimento de serviços que proporcionam as mesmas funcionalidades no ambiente residencial. Quando um alarme é gerado, pessoal médico especializado pode avaliar os dados que provocaram esse alarme e decidir a acção a tomar (por exemplo enviar uma equipa médica a casa do paciente, enviar uma ambulância à residência deste para o transportar para o hospital, entre outros).

3.1.1.5.3 Serviços de assistência médica remota

Estes serviços referem-se à possibilidade de providenciar cuidados médicos contínuos mas não críticos a idosos, pacientes em recuperação pós operatória, pacientes que padeçam de

doenças terminais, entre outros, sem que para isso seja necessário que estes permaneçam numa unidade hospitalar. Este facto representa uma melhoria significativa na qualidade de vida destes pacientes, uma vez que podem continuar a usufruir do seu ambiente doméstico e social. Tal é conseguido associando a monitorização remota à operação remota do equipamento médico que providencia ao paciente esses cuidados médicos básicos.

3.1.1.6. Algumas considerações sobre serviços residenciais actuais e futuros

Importa referir que alguns dos serviços listados na secção 3.1.1, são já oferecidos ao utilizador (por exemplo a aplicação *Skype* [98] permite ao utilizador usufruir de um serviço de telefonia VoIP). No entanto, estes serviços são normalmente disponibilizados partilhando os mesmos recursos que o serviço de acesso à Internet (num modelo de *best effort*), não sendo possível oferecer quaisquer garantias de disponibilidade, robustez ou qualidade do serviço. Tal acontece porque, por um lado os provedores de serviço não têm qualquer relação de negócio com o operador de rede e por outro, porque algumas das redes de acesso actuais apenas suportam este modelo de serviço.

Assim, a secção seguinte apresenta algumas das características desejáveis em redes de acesso de próxima geração que permitirão, por exemplo oferecer ao cliente garantias na prestação de um dado serviço.

3.1.2. Caracterização de redes de acesso de nova geração

No modelo actual de prestação de serviços o provedor de serviços acumula normalmente a função de operador de rede. Até agora este modelo tem resultado, visto que o principal motivo pelo qual os clientes adquirem uma ligação de banda larga é o serviço de acesso à Internet. Contudo, facultar o acesso a novos serviços, alguns dos quais necessitam de enormes quantidades de largura de banda (por exemplo os serviços que envolvem transmissão de vídeo), é sem duvida um bom método para valorizar e aumentar a procura de largura de banda por parte dos clientes. Tal pode ser conseguido remodelando a rede de acesso de forma a permitir o acesso simultâneo por parte dos clientes a vários provedores de serviços, podendo porém ser mantida a relação que existe actualmente entre o cliente e o seu operador de rede. Este acesso de muitos para muitos é de interesse para o operador de rede pelos motivos já referidos, para os provedores de serviços que obtêm um novo canal de distribuição para os seus serviços e dos clientes que passam a ter acesso a um conjunto de serviços até então inatingíveis. Desta forma, os operadores de rede mantêm a exclusividade na prestação de

alguns serviços aos seus clientes (por exemplo serviço de acesso à Internet, serviço de telefonia) e “vendem” recursos de rede a terceiras entidades que apenas estão interessadas na prestação de serviços.

Contudo, para que os clientes se predisponham a pagar por esses novos serviços é necessário que se ofereçam garantias quanto à sua disponibilidade, previsibilidade, rapidez e robustez. Estas garantias podem ser conseguidas com a introdução de mecanismos de QoS em todo o caminho entre os provedores de serviços e o terminal onde o serviço vai ser terminado, o que inclui portanto toda a rede de acesso. De acordo com o grau de exigência que se pretende para cada um dos parâmetros acima referidos, a rede poderá necessitar da implementação de mecanismos que possibilitam QoS relativa (tratamento preferencial de um tipo de tráfego sobre outro) ou de mecanismos que possibilitam QoS garantida (onde existe garantia de parâmetros de QoS). Esta última, apesar de preferível, é mais complexa, visto que normalmente necessita de técnicas de reserva de recursos de rede.

Alguns dos serviços futuros requerem uma grande quantidade de recursos durante um curto espaço de tempo (por exemplo o serviço de VoD). De forma a evitar que estes recursos de rede estejam permanentemente reservados, é também desejável a implementação de mecanismos que permitem efectuar a alocação dinâmica de recursos. Apesar de esta funcionalidade requerer “inteligência” adicional na rede (mecanismos de controlo do uso de recursos, mecanismos de controlo e admissão de chamadas, entre outros), apresenta a vantagem de apenas haver ocupação de recursos de rede durante a sessão de um dado serviço, recursos estes que são libertados logo após o término da mesma. Importa referir que esta capacidade é ainda mais valiosa se a rede for capaz de detectar o início/término de uma sessão e efectuar automaticamente a respectiva reserva/libertação de recursos.

Uma outra característica desejável em redes de acesso de próxima geração é a presença de sistemas de monitorização da rede. Estes sistemas, através da realização de medições em pontos estratégicos da rede, permitem detectar situações de congestionamento e de sub aproveitamento de recursos da mesma. Isto pode permitir otimizar em tempo real a utilização de recursos de rede, o que é essencial para a garantia da QoS. Todavia, a utilidade destas medições não se esgota em tarefas de monitorização da QoS, visto que estas podem também ser usadas para detectar falhas de elementos de rede, construir perfis de tráfego necessários aos processos de planeamento e dimensionamento de rede, detectar ataques maliciosos ou intrusões que põem em causa a segurança e estabilidade da rede ou mesmo

servirem como entrada para sistemas de taxaço.

Em adição aos sistemas de monitorizaço de rede, a implementaço de mecanismos de *multicast* pode também proporcionar a optimizaço da utilizaço dos recursos na rede. Estes mecanismos permitem a entrega simultânea de uma mesma informaço a um conjunto de receptores, havendo apenas replicaço dessa mesma informaço nos troços de rede não comuns ao conjunto de receptores. Desta forma, na presença de serviços que se encaixem neste parâmetros (por exemplo serviço de difusão de canais de TV/áudio, jogos multi-jogador on-line) a implementaço de funcionalidades *multicast* na rede de acesso permite a poupança de recursos, nomeadamente de largura de banda.

Uma rede multi-serviço, que possibilita a um dado cliente aceder a vários provedores de serviço simultaneamente, irá requerer também alteraço nos modelos actuais de autenticaço, autorizaço e taxaço. O modelo empregue normalmente para o serviço de acesso à Internet (como descrito nas secçoões 2.2.3.1.3 e 2.2.3.2.4) deixa de ser eficaz num modelo multi-serviço essencialmente devido à introduço de serviços que requerem QoS e também devido ao facto de alguma da informaço do sistema de autenticaço, autorizaço e taxaço passar agora a ser também relevante para o provedor de serviços (para além de ser relevante para o operador de rede). Assim, o processo de autenticaço e autorizaço deverá ser dividido em autenticaço/autorizaço ao nível da rede (efectuado pelo operador de rede) e autenticaço/autorizaço ao nível do serviço (efectuado pelo(s) provedor(es) de serviços): a autenticaço/autorizaço efectuada pelo operador de rede permite a este identificar univocamente um cliente e daí inferir acerca dos recursos de rede (por exemplo largura de banda, QoS, endereçoamento, entre outros) de que este pode usufruir. Por sua vez a autenticaço/autorizaço efectuada pelo provedor de serviço é necessária para que este controle o acesso aos seus serviços e também para que este possa fornecer aos seus clientes serviços personalizados. Da mesma forma, a recolha de informaço de taxaço (necessária para posterior cobrança ao cliente), pode ser efectuada directamente pelos provedores de serviço ou em alternativa pelo próprio operador de rede que depois partilha a informaço recolhida com os respectivos provedores de serviço.

Este conjunto de características e funcionalidades permitirão que os mais variados tipos de serviços, provenientes de prestadores de serviços distintos, possam ser distribuídos sobre uma mesma rede de acesso.

3.1.3. Caracterização de redes domésticas de nova geração

Da análise efectuada na secção 2.2.4 verifica-se que actualmente as redes domésticas são dimensionadas essencialmente com o objectivo de partilhar um serviço de acesso à Internet de banda larga entre vários terminais que residem na rede doméstica. Uma vez que este serviço é normalmente disponibilizado a clientes residenciais num modelo de *best effort*, as redes residenciais são também dimensionadas à luz desse modelo. No entanto, quando se pretende disponibilizar diferentes serviços, com diferentes requisitos e sobre uma mesma rede doméstica, à semelhança do que acontece com as redes de acesso, novas capacidades terão de ser introduzidas. Estas novas capacidades, de uma maneira geral, poderão reflectir-se na adição de novas funcionalidades aos elementos já existentes e/ou na adição de novos elementos à rede doméstica apresentada na figura 13.

Um novo tipo de elemento a ser introduzido implementa as funcionalidades de *Terminal Adaptation* (ver modelo apresentado na secção 2.1.2). A introdução destes novos elementos na rede doméstica, resulta do facto de alguns terminais que implementam a interface entre o serviço e o utilizador (por exemplo TV, telefone POTS, entre outros) não serem compatíveis com a tecnologia *Ethernet/IP* (tecnologias consideradas na CPN). Estes terminais não podem portanto utilizar directamente os recursos e serviços disponibilizados através da rede doméstica. Como exemplo, um ATA desempenha a função de TA quando se pretende utilizar um telefone POTS num serviço de telefonia IP. Da mesma forma, uma STB desempenha a mesma função para o serviço de *Video on Demand*.

De uma forma geral um dispositivo que desempenhe as funções do bloco funcional TA, ou seja que processa informação proveniente da CPN referente a um dado serviço e a apresenta num formato passível de ser transmitido ao longo de uma rede não *Ethernet/IP* e/ou passível de ser reproduzido num dispositivo terminal, é denominado de *gateway* de serviço [16] [99]. Importa referir que nada impede que um dado dispositivo físico encerre no seu interior as funcionalidades de um modem, de um *router* e de um *gateway* de serviços e forme assim um *gateway* residencial multi-serviço.

Um *gateway* residencial pode então ser definido como um dispositivo físico que como mínimo implementa as funcionalidades do bloco NT2 [16] [100]. Contudo, um único dispositivo pode, para além do bloco funcional NT2, implementar um ou vários dos restantes blocos funcionais apresentados no modelo da figura 2, sendo esta situação expectável quando nos referimos a *gateway* residenciais. A figura seguinte apresenta a composição de um *gateway* residencial de

acordo com esta definição.

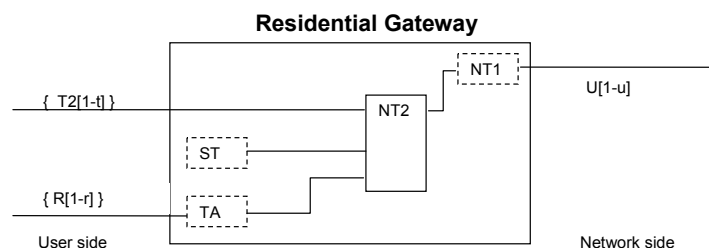


Figura 15: Composição de um *gateway* residencial [16]

O *gateway* residencial para além de vários blocos funcionais pode ainda apresentar até u interfaces do tipo U, que permitem a ligação do *gateway* a redes de acesso de diferentes tecnologias (embora apenas uma seja usada em simultâneo), até t interfaces do tipo T2, que permitem a ligação do *gateway* a CPNs de diferentes tecnologias e até r interfaces do tipo R, que conectam o *gateway* a dispositivos que implementem o bloco funcional ST'.

A figura 16 representa a arquitectura de uma rede doméstica multi-serviço onde vários *gateways* de serviços estão incorporados.

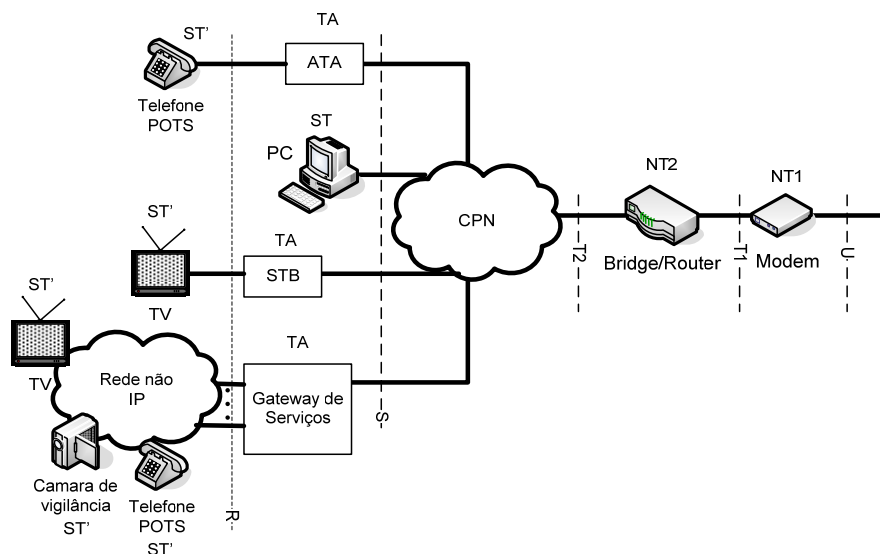


Figura 16: Rede doméstica multi-serviço. Baseado na figura 17 de [16]

Devido ao facto de a rede doméstica não ser estanque, na medida em que interage com a rede de acesso, algumas das características das redes de acesso de próxima geração são de alguma forma extensíveis à rede doméstica. Assim, se a rede de acesso deve permitir que vários provedores de serviços usem os recursos por ela disponibilizados para distribuírem os seus serviços, o mesmo terá que acontecer com a rede doméstica. Da mesma forma, a rede

doméstica deve permitir que vários tipos de serviços provenientes de vários provedores de serviços possam ser distribuídos sem que haja interferência entre estes.

Fundamental para esta última característica é a introdução de mecanismos de QoS na rede doméstica. O uso de QoS relativa evita a implementação de mecanismos complexos (por exemplo gestão de largura de banda, CAC, entre outros). Nesse sentido, organizações como o DSL Forum [99] suportam a implementação de mecanismos de QoS relativa baseada nas normas IEEE802.1Q [84], sendo que mecanismos de QoS que operam ao nível da camada IP e acima deverão ser transportados transparentemente entre o bloco NT2 e os blocos ST ou TA. Isto implica que o bloco NT2 funcione como o ponto onde é feita a tradução do esquema de marcação de pacotes/tramas usado na rede de acesso para tramas IEEE802.1Q e vice-versa. Para que a prioridade das tramas IEEE802.1Q seja respeitada ao longo da CPN, é necessário que as tecnologias de rede que a compõe suportem o transporte de tramas *Ethernet* e a priorização de tramas (o suporte desta característica para as tecnologias mais comuns pode ser consultado na tabela 7 do Anexo II).

Dada a vastidão de dispositivos e serviços possíveis, a configuração de novos dispositivos ou a reconfiguração de dispositivos já existentes inerentes à subscrição de novos serviços por parte do utilizador, fica normalmente fora do alcance dos utilizadores sem conhecimento técnico adequado. Por outro lado, a deslocação de pessoal técnico especializado às instalações do cliente acarretam elevados custos para os operadores e provedores de serviços. É portanto necessário reduzir o papel do utilizador neste processo, pelo que a configuração da rede doméstica deverá passar a ser efectuada pelos vários operadores (de rede e serviço) e com a mínima intervenção do utilizador. Dentro dos vários processos de configuração o de auto-configuração (local ou remota) desempenha um papel preponderante na correcta configuração de redes domésticas.

De uma forma geral a auto-configuração pode ser definida como o processo através do qual um elemento efectua autonomamente acções que lhe permitem obter informação de configuração relativa a um dado serviço subscrito pelo cliente [101]. Assim, a análise efectuada as longo da secção 2.2.3 permite afirmar que actualmente mecanismos básicos que permitem a auto-configuração de dispositivos que implementem os blocos funcionais NT1, NT2 e ST (para o serviço de acesso à Internet) estão já implementados. Contudo, para redes domésticas multi-serviço complexas, as suas funcionalidades têm de ser alargadas, bem como a sua aplicabilidade a outros tipos de dispositivos e serviços.

3.2. Plataformas e gateways de serviços residenciais

Como exposto ao longo da secção anterior, os *gateways* de serviço desempenham um papel fundamental em toda a plataforma de serviços. Para além de fazerem a adaptação dos dados recebidos através da rede de acesso para um formato passível de ser reproduzido em dispositivos que fazem a interface entre o utilizador e o serviço, são também o “último” elemento na rede doméstica sobre o qual o operador/provedor pode actuar para garantir a correcta operação de um determinado serviço.

A primeira parte desta secção descreve brevemente uma das plataformas de serviços mais comum actualmente, a qual inclui *gateways* de serviço dedicados. Em contraste com esta plataforma, uma plataforma de serviços que permita a distribuição de múltiplos tipos de serviços e permita a utilização de um único *gateway* de serviços para essa mesma variedade de serviços, apresenta claras vantagens. Assim, a restante secção apresenta soluções que permitem a implementação de um *gateway* multi-serviço.

3.2.1. Plataformas de serviços áudio/vídeo

As plataformas de distribuição de serviços áudio/vídeo, são actualmente o exemplo mais comum de uma plataforma de serviços. Neste tipo de plataformas dispositivos residenciais denominados de *Set Top Boxes* desempenham um papel fundamental na prestação do serviço possibilitando o acesso e a apresentação do serviço ao cliente, ou seja desempenhando a função de um *gateway* de serviço. À semelhança do que tem acontecido com toda a plataforma, também estes dispositivos têm vindo a evoluir. Assim, podemos classificar estes dispositivos em duas classes distintas: dispositivos que não necessitam de canal de retorno para o provedor de serviços e dispositivos que utilizam um canal de retorno. O primeiro tipo de dispositivo limita-se a receber os dados referentes ao serviço áudio/vídeo, através de um canal de *broadcast*, e a apresentá-los ao utilizador. Por sua vez, o segundo tipo de *gateway* de serviço permite também ao utilizador usufruir de serviços como o *video on demand* e o *personal video recording*. Dispositivos como os fabricados pela TiVo Inc. [102] ou pela Scientific Atlanta [103] utilizam o canal de retorno não só para transmitir comandos de *play*, *pause*, *fast forward* e *fast rewind* no serviço de VoD (ver secção 3.1.1.2.2), mas também para consultar *metadata* acerca de conteúdos transmitidos que, uma vez cruzada com as preferências do utilizador (por exemplo tipo de programa, actores, entre outros), resulta na gravação (num disco rígido interno)

automática de programas considerados de interesse para o utilizador.

Por sua vez, o aparecimento da televisão digital impulsionou o aparecimento da chamada televisão interactiva (ver secção 3.1.1.2.4) que, aos serviços tradicionais, adiciona serviços interactivos que impõem um *gateway* de serviço mais complexo que os acima descritos. Desta forma, estes novos serviços requerem um novo tipo de *gateway* de serviços áudio/vídeo que permita a execução das aplicações que implementam os referidos serviços interactivos. Uma vez que os provedores de serviços têm todo o interesse em que estas aplicações sejam executadas no maior número possível de STBs, estas estão geralmente equipadas com um *middleware*. Este, pode ser definido como uma camada de *software* intermédia que se situa entre o sistema operativo e a aplicação. Esta camada fornece uma API genérica, padronizada e bem definida que abstrai a especificidade e heterogeneidade do *hardware* e *software* (sistema operativo) de diferentes tipos de dispositivos. Quer isto dizer que todos os *gateways* de serviços áudio/vídeo, independentemente das suas características de *hardware* ou sistema operativo, que executem um dado *middleware* podem oferecer aos seus utilizadores os mesmos serviços interactivos.

Existem actualmente vários tipos de *middleware* disponíveis como por exemplo o OpenTV [104], o Microsoft TV [105] ou o MHP (*Multimedia Home Platform*) [106]. De entre estes destaca-se o MHP, que ao contrário dos outros dois, é um *middleware* não proprietário especificado pelo consórcio DVB (*Digital Video Broadcasting*) [107] e normalizado através do organismo ETSI [5].

3.2.1.1. Multimedia Home Platform

A especificação MHP, actualmente na versão 1.1 [108], centra-se essencialmente no *gateway* de serviços áudio/vídeo (por exemplo STB, TV digital integrada, PC multimédia), porém outras entidades são também considerados na cadeia de prestação de serviços. A figura 17 ilustra as várias entidades envolvidas na prestação de serviços através da plataforma MHP.

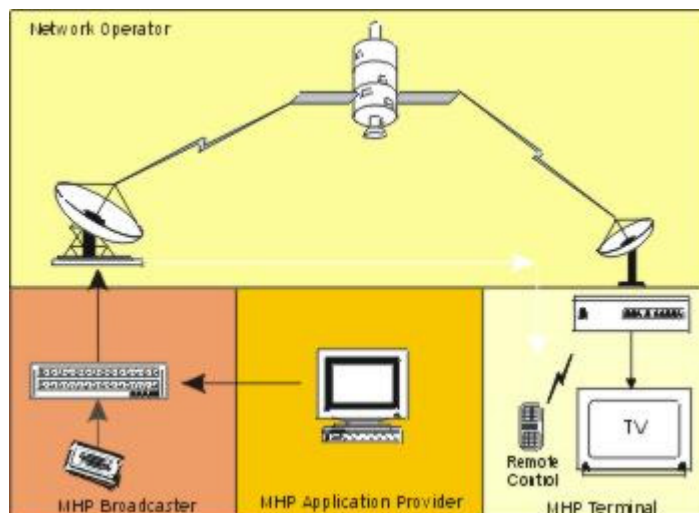


Figura 17: Entidades envolvidas na prestação de serviços através da plataforma MHP [106]

As aplicações que implementam os serviços interactivos são primeiro de tudo geradas por um ou mais provedor de aplicações (*Application Provider*). Exemplos de tais aplicações são jogos, serviços de informação (financeira, notícias), aplicações de *e-commerce* ou aplicações de *e-learning*. Tipicamente cada serviço interactivo está de alguma forma integrado num outro serviço que é oferecido ao cliente (por exemplo um canal de desporto, uma transmissão *live* de um dado evento). A integração da aplicação MHP no serviço disponibilizado ao cliente é efectuada pelo *MHP Broadcaster* que poderá acumular as funções de provedor de aplicações ou não. O conjunto de dados que diz respeito ao serviço e ao respectivo serviço interactivo é depois transmitido em tramas MPEG-2 através de uma rede de comunicações DVB (por exemplo DVB-S, DVB-C, DVB-T, entre outras), propriedade de um operador de rede (*Network Operator*), até ao terminal MHP do cliente onde as aplicações são executadas. A interface entre o utilizador e o serviço interactivo é tipicamente efectuada através da TV e de um controlo remoto que pode acumular funcionalidades de teclado e rato.

A figura 18 apresenta a arquitectura MHP que, de uma forma simples, é dividida em três camadas distintas.

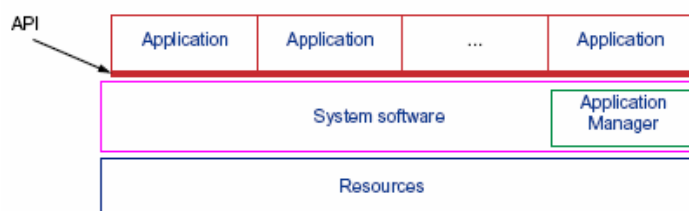


Figura 18: Arquitectura MHP [108]

A primeira camada contempla recursos de *hardware* e *software*, a segunda, designada por *software* de sistema (*system software*), implementa uma camada de abstracção (baseada na tecnologia *Java*) que reside entre os recursos e as aplicações (ou seja um *middleware*). O gestor de aplicações (*Application Manager*), que faz parte desta segunda camada, é responsável por gerir o ciclo de vida das aplicações MHP que constituem a terceira e última camada. De acordo com a existência ou não de um canal de retorno e com a possibilidade de acesso à Internet, a norma define três perfis MHP. Cada um destes, brevemente descritos de seguida, especifica os recursos que devem estar disponíveis num terminal MHP que implemente o respectivo perfil.

- Perfil de difusão enriquecido (*Enhanced Broadcasting Profile*): um *gateway* de serviço que implemente este perfil apenas suporta serviços interactivos que não necessitam de um canal de retorno.
- Perfil de difusão interactivo (*Interactive Broadcasting Profile*): estende o perfil anterior para serviços interactivos que necessitam de um canal de retorno.
- Perfil de acesso à Internet (*Internet Access Profile*): estende o perfil anterior permitindo o acesso a conteúdos e serviços na Internet.

Quando um serviço tem associado uma ou várias aplicações MHP, juntamente com os dados referentes a este, é difundida informação acerca das aplicações associadas. Esta informação inclui por exemplo o nome da aplicação, a localização dos seus ficheiros, argumentos que esta necessita para ser iniciada, entre outras. Assim, quando o utilizador comuta para o serviço, o gestor de aplicações termina a execução de todas as aplicações MHP associadas ao serviço anteriormente seleccionado e verifica toda a informação referente às novas aplicações a iniciar. Uma vez determinadas as aplicações a iniciar automaticamente com o novo serviço seleccionado, os dados referentes às mesmas são obtidos da *stream* MPEG-2 e as aplicações iniciadas. Aplicações associadas ao serviço mas não automaticamente iniciadas, são apenas iniciadas através de indicação explícita do utilizador. As aplicações mantêm-se em execução até que o serviço a quais estão associadas se mantenha activo.

A plataforma MHP permite portanto acrescentar novos serviços (interactivos) a serviços áudio/vídeo oferecidos através de uma rede de comunicações DVB. As aplicações que implementam estes novos serviços interactivos são automaticamente transferidas do provedor de serviços para o *gateway* de serviço áudio/vídeo do cliente e iniciadas automaticamente sem qualquer intervenção do utilizador.

3.2.2. Motivação e requisitos de uma plataforma de serviços residenciais

A abordagem por parte dos operadores e provedores de serviços até agora tem sido a de disponibilizar ao cliente um *gateway* por cada serviço. Como exemplo temos na plataforma de serviços vídeo/áudio, apresentada na secção anterior, a STB para serviços vídeo, os ATA para o serviço de VoIP, entre outros. Todavia, se estes *gateways* de serviço passarem a ser um ponto central para deposição de vários tipos de serviços (das aplicações que os implementam), tornando-se em *gateways* multi-serviço, um conjunto de vantagens pode daí advir, nomeadamente:

- Partilha de custos da plataforma de *hardware* do *gateway* de serviços: uma vez que diferentes serviços usarão essa plataforma, o custo da plataforma poderá ser repartido (directa ou indirectamente) pelos vários provedores de serviços que lá colocam os seus serviços.
- Possibilidade de interacção entre serviços: como as várias aplicações que implementam os serviços são executadas num mesmo ambiente, a interacção entre diferentes tipos de serviços passa a ser possível. Por exemplo o início da visualização de um filme (serviço vídeo) poderá causar a descida automática das persianas da sala de estar (serviço de domótica), sem qualquer intervenção do utilizador nesta última acção. Esta interacção entre serviços poderá dar origem a novos tipos de serviços até agora não possíveis.
- Centralização do processo de configuração de dispositivos/serviços: como referido na secção 3.1.3, com o aumento do número de serviços e dispositivos na rede doméstica, aumenta também a complexidade da sua configuração e gestão. A concentração de serviços e dispositivos facilita a criação de soluções de configuração e gestão normalizadas.
- Centralização do controlo e acesso a serviços para o utilizador: o utilizador passa também a ter uma interface centralizada e única para personalização dos seus serviços. Da mesma forma, na necessidade de acesso remoto a serviços por parte do utilizador, esse acesso poderá ser implementado de uma forma equivalente para todos os serviços, ou seja, o utilizador pode usar o mesmo método para aceder remotamente a serviços distintos.

Contudo, a facilidade de executar aplicações originárias das mais diversas fontes, com funcionalidades totalmente distintas e em diversas plataformas de *hardware* com diferentes características, não é uma tarefa trivial. Todavia, se à semelhança do que acontece na plataforma MHP, as aplicações forem construídas sobre um *middleware* existe a garantia das

mesmas poderem ser executadas em todos os ambientes em que este esteja presente. O ambiente doméstico é, no entanto, um ambiente muito específico e com requisitos especiais que esse *middleware* deve levar em conta. A seguir listam-se alguns desses requisitos:

- Os serviços podem ser adquiridos por um curto período de tempo, logo será vantajoso para o provedor de serviços um *middleware* que possibilite a gestão remota do ciclo de vida das aplicações que implementam esses serviços.
- Num mesmo ambiente irão ser executados serviços de diferentes provedores de serviços. Assim, o *middleware* deverá providenciar um modelo de programação consistente que proporcione uma coexistência pacífica entre os vários serviços.
- Da mesma forma, o *middleware* deve também evitar que um determinado serviço prejudique (intencionalmente ou não) a performance de outros serviços.
- Um mesmo provedor de serviços poderá também ter interesse em que vários dos seus serviços cooperem, pelo que os serviços não deverão ser executados num regime de isolamento total.

Apercebendo-se das vantagens que uma plataforma residencial e um *gateway* de serviços comum a vários serviços apresentam, um grupo de fabricantes, operadores de telecomunicações e companhias de *software* juntou-se com a finalidade de especificar um *middleware* que satisfizesse os requisitos acima enumerados. Em Maio de 2000 foram editadas as primeiras especificações desse *middleware*, normalmente conhecido por OSGi [109]. Nas secções seguintes este *middleware* ou *framework* é apresentado em detalhe.

3.2.3. A aliança OSGi

A aliança OSGi (anteriormente denominada de *Open Services Gateway Initiative*) foi fundada em Março de 1999, com o objectivo de criar as especificações de uma plataforma que permitisse a entrega remota de componentes de *software*. Tipicamente esta plataforma seria instalada em dispositivos com recursos computacionais limitados (ambientes *embedded*) e que pudessem operar como um *gateway* de serviços (por exemplo modems, *routers*, *set top boxes*, entre outros). A primeira especificação da plataforma, concluída em Maio de 2000, endereçava essencialmente o mercado residencial. Contudo, os seus atributos tornaram-na aplicável e atractiva para outros mercados, como o mercado das comunicações móveis, da electrónica de consumo, dos computadores pessoais e também para o mercado automóvel. Desde então a

especificação tem evoluído de forma a satisfazer os requisitos impostos pelos vários mercados onde a plataforma tem sido aplicada, indo já na sua quarta versão.

A primeira versão da especificação [110] centrou-se essencialmente na definição de uma *framework* (baseada na tecnologia *Java*) que preenche as necessidades de uma plataforma que está em permanente funcionamento, que suporta a alteração dinâmica do conjunto de aplicações em execução e que permite a partilha de código e serviços entre aplicações. Sobre a *framework*, foram também definidos alguns serviços normalizados, nomeadamente um *web server* e um serviço de *log*, serviços estes que podem ser usados pelas aplicações em execução na plataforma. A segunda versão da especificação [111] teve por principal objectivo alargar o número de cenários onde a plataforma pode ser aplicada. Para tal, alargou o leque de serviços normalizados, clarificou e melhorou a *framework* definida na primeira versão e introduziu ainda mecanismos que permitem a gestão remota de toda a plataforma. A terceira versão da especificação [112] pretendeu melhorar o suporte para plataformas de serviços móveis e de entretenimento (definindo novos serviços e não só) nomeadamente plataformas dirigidas ao mercado automóvel. Tal objectivo foi conseguido com sucesso ao ponto de a especificação ter sido adoptada por fabricantes de automóveis (por exemplo BMW Série 5). Esta versão introduz algumas arquitecturas de referência que pretendem clarificar a terminologia e contextualizar alguns serviços e capacidades da plataforma OSGi. Finalmente a quarta e última versão da especificação [113] apresenta significativas melhorias na *framework*, melhorias estas que implicaram a total rescrita da mesma. As melhorias visam essencialmente a extensão da *framework* para novas áreas nomeadamente a área dos dispositivos móveis.

Importa referir que as várias versões da especificação são compatíveis, na medida em que uma aplicação desenvolvida sobre uma plataforma que implementa a versão 1 da especificação, deverá funcionar também numa plataforma que implementa a versão 4.

3.2.3.1. Modelo OSGi do gateway de serviços residencial

Como já referido, as especificações da plataforma OSGi pretendem endereçar vários mercados, sendo que em cada um destes mercados existem cenários típicos onde a aplicação da plataforma se adequa. Apesar das especificações OSGi abordarem essencialmente a normalização do *middleware* que permite a entrega remota de aplicações, dada a importância do mercado residencial um modelo de referência para um *gateway* de serviços residencial e respectivo cenário de aplicação, são apresentados na versão três das especificações [112]. Este modelo, apresentado na figura 19, não é de forma alguma normativo, sendo apenas

apresentado para contextualização:

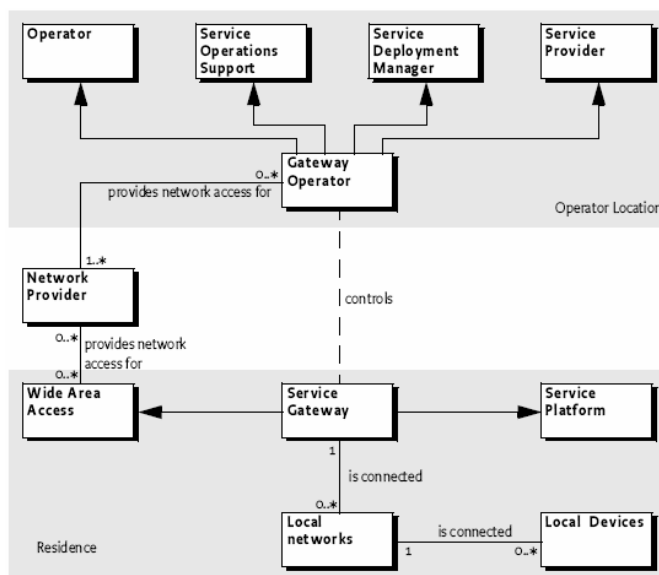


Figura 19: Possível modelo de um *gateway* de serviços residencial (OSGi) [112]

A figura acima apresenta várias entidades, algumas das quais podem ser traduzidas em dispositivos físicos enquanto outras apenas pretendem representar entidades funcionais. As entidades da figura 19 são brevemente descritas a seguir:

- *Service Gateway* – *gateway* de serviços: constituído por um servidor de plataformas OSGi que executa uma ou mais plataformas. Num ambiente residencial este pode também desempenhar as funções de *gateway* entre uma rede IP e uma rede local. Assim, o *gateway* de serviços poderá fazer uso dessas redes locais (na figura 19 referido por *local network(s)*) para comunicar com dispositivos locais (na figura 19 referidos por *local device(s)*) que enviam e/ou recebem informação da plataforma de serviços, e/ou enviam e/ou recebem informação do *gateway* IP. Por outro lado, o *gateway* de serviços poderá fazer uso do acesso a uma rede WAN (que tipicamente suporta o transporte de tráfego IP), disponibilizado por um operador de telecomunicações (na figura 19 referido por *network provider*), para comunicar com entidades terceiras localizadas fora da residência, como por exemplo com um operador de *gateway*.
- *Service Platform* – plataforma de serviços: este termo quando utilizado no contexto das especificações OSGi refere-se à instanciação de uma *Java Virtual Machine* [114], a uma *framework* OSGi e a um conjunto de componentes de *software* que são executadas sobre essa *framework*. A plataforma de serviços é o principal objecto das especificações OSGi, pelo que será descrita em maior detalhe ao longo desta secção.

- *Gateway Operator* – operador de *gateway*: é uma das entidades com a qual a plataforma de serviços comunica quando o *gateway* de serviços possui a capacidade de comunicar com entidades localizadas fora da residência. Tipicamente esta entidade actua sobre a plataforma de serviços em prol de outras entidades, tais como:
 - *Service Provider* – provedor de serviços: organização que disponibiliza componentes a serem instalados e executados na plataforma de serviços e que implementam uma dada funcionalidade útil ao utilizador.
 - *Service Deployment Manager* (SDM) – gestor de instalação de serviços: sistema que instala e gere as aplicações de um ou mais provedores de serviços.
 - *Service Operations Support* – operações de suporte a serviços: *hardware* ou *software* de suporte (que não reside na plataforma de serviços) necessário para executar uma dada aplicação.
 - *Operator* – operador: organização responsável pela plataforma de serviços. Como tal a sua principal função é controlar que SDMs estão autorizados a instalar aplicações na plataforma de serviços. Pode também efectuar tarefas de configuração e gestão na mesma.

3.2.3.2. A plataforma OSGi

A plataforma OSGi pode ser dividida em duas partes fundamentais: a *framework* e os serviços disponibilizados por essa *framework*. Antes de mais é apresentada a forma como no universo OSGi uma aplicação é entendida, dado que esta visão é fundamental para a compreensão das especificações, do funcionamento e das funcionalidades oferecidas por uma plataforma OSGi. Assim, no universo OSGi uma aplicação é entendida como um conjunto de módulos de *software* denominados de *bundles*. Cada um destes *bundles* implementa zero, uma ou várias das funcionalidades do conjunto total de funcionalidades oferecidas pela aplicação. Estas funcionalidades singulares têm na nomenclatura OSGi a designação de serviços. Assim, numa só frase, no domínio OSGi, uma aplicação é constituída por um conjunto de serviços (funcionalidades), serviços esses que são implementados por *bundles*.

Em concreto, um serviço na terminologia OSGi é definido [110] como um conjunto de classes *Java* que implementam uma certa funcionalidade, sendo que a especificação do serviço é feita através de uma interface *Java* que é depois implementada por uma ou várias classes. Importa referir que a definição de um serviço OSGi é suficientemente genérica para que tanto um servidor HTTP, como um dispositivo físico possam ser apresentados como um serviço.

Por sua vez um *bundle* pode ser definido [110] como um método de empacotamento para os serviços destinados a uma plataforma OSGi, método este que é reconhecido e “entendido” pela *framework* OSGi. Um *bundle* é então constituído por um conjunto de classes *Java* e possivelmente também por outros recursos (por exemplo ficheiros HTML, ícones, ficheiros de ajuda, entre outros) que no seu conjunto implementam zero, um ou vários serviços. Para além de dados relativos à implementação do serviço, o *bundle* transporta também informação que é usada pela *framework* para instalar e executar o *bundle*. Fisicamente um *bundle* é concretizado através de ficheiro *Java Archive file* – JAR [115].

O conceito de *bundle*, serviço e aplicação estão então directamente correlacionados. A figura 20 pretende ilustrar, com um exemplo, estes conceitos e introduzir a filosofia de partilha de serviços característica das plataformas OSGi.

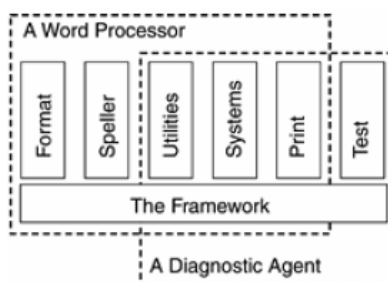


Figura 20: Cooperação entre serviços e *bundles* [116].

A figura 20 apresenta uma plataforma de serviços OSGi, com duas aplicações instaladas: uma aplicação de processamento de texto e uma aplicação de diagnósticos. A primeira aplicação é constituída por um conjunto de *bundles*, *bundles* estes que implementam vários serviços (funcionalidades) como o serviço de impressão, o serviço de formatação de texto, entre outros. Se o provedor de serviços responsável por estes *bundles* assim o desejar, pode permitir que outras aplicações usem os serviços implementados nos seus *bundles* (por exemplo a aplicação de diagnósticos poderá necessitar do serviço de impressão) evitando assim uma replicação de funções na plataforma. Por outro lado, a filosofia da plataforma OSGi permite também que o provedor de serviços adicione ou remova serviços à aplicação conforme o desejo do utilizador sem ter que para isso reinstalar toda a aplicação, e mais importante, sem perturbar os restantes serviços da plataforma e sem necessitar de reinicialização da *Java Virtual Machine*. Este exemplo apesar de simbólico ilustra bem as potencialidades da plataforma OSGi.

3.2.3.2.1 A framework OSGi

A *framework* OSGi é um ambiente normalizado que suporta a execução de serviços

implementados por *bundles*. As suas funções passam (mas não se limitam) pela disponibilização de mecanismos que permitem a gestão do ciclo de vida desses *bundles*, a partilha de código *Java* entre *bundles*, a partilha de serviços e a gestão remota de toda a plataforma. A *framework* é portanto bastante complexa, pelo que pode ser dividida em camadas, cada uma das quais contribui para o conjunto de funcionalidades que a *framework* apresenta. As camadas que constituem a *framework* são apresentadas na figura 21.

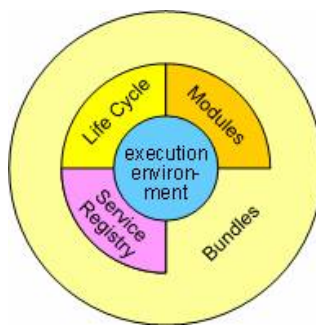


Figura 21: Camadas da *framework* OSGi [109].

A primeira camada, denominada de *Execution Environment* (ambiente de execução), caracteriza a plataforma *Java* usada para executar a *framework*. A segunda camada, denominada de *Modules* (camada de modularização), implementa, entre outros, mecanismos de carregamento de classes *Java* da *framework*. A terceira camada, conhecida por *Life Cycle* (ciclo de vida), define mecanismos que permitem controlar o ciclo de vida dos *bundles*. Finalmente, a camada *Service Registry* (registo de serviços) é responsável pela partilha de serviços entre *bundles*. Para além destas camadas existe ainda uma camada de segurança (não representada na figura) que intervém em todas as outras. As várias camadas são descritas em maior detalhe nas secções seguintes.

3.2.3.2.1.1 Ambiente de execução

A camada referente ao ambiente de execução da *framework* OSGi define os requisitos que a plataforma *Java* usada pela *framework* deve cumprir. Uma plataforma *Java* é constituída por uma JVM e APIs (conjuntos de classes *Java*), sendo que o ambiente de execução pretende especificar precisamente as APIs que permitem à *framework* operar correctamente. As duas primeiras versões das especificações apenas referem que a *framework* é definida com base na plataforma *Java 2* [117] (a *framework* pode também operar num ambiente JDK1.1 [118] com algumas limitações [111]). Por sua vez a terceira e quarta versões da especificação definem a partir da configuração/perfil CDC1.0/Foundation1.0 [119] da plataforma J2ME [120] um

ambiente de execução mínimo que suporta a *framework* e a implementação de serviços básicos.

3.2.3.2.1.2 Camada de modularização

Tipicamente numa plataforma *Java* duas aplicações distintas executadas simultaneamente são executadas por duas instanciações diferentes da JVM, o que garante isolamento entre as aplicações. Estas aplicações podem utilizar as classes da plataforma *Java* mas não podem partilhar classes entre si. Porém, uma plataforma OSGi é executada numa só instância da JVM o que permite que um *bundle* em execução nessa plataforma possa aceder a classes *Java* que não lhe pertencem. Contudo, esta partilha de classes pode não ser aceitável e tem de ser efectuada de uma forma controlada. Por este motivo a *framework*, e em particular a camada de modularização, implementa mecanismos que permitem a cada *bundle* ter o seu próprio *class loader* (objecto que carrega classes *Java* para a JVM) e que por defeito apenas pode carregar classes do próprio *bundle*. Para além das suas próprias classes, um *bundle* pode ainda aceder livremente a classes disponibilizadas pelo ambiente de execução, delegando para tal o carregamento dessas classes ao *class loader* do sistema.

Por sua vez o acesso a classes pertencentes a outros *bundles* é bastante mais controlado, mas não deixa de ser possível e de ter interesse. A partilha de classes entre *bundles* evita, por exemplo, que dois *bundles* distintos que necessitam de uma determinada biblioteca tenham ambos de a conter no seu ficheiro JAR. Desta forma, os *bundles* tornam-se mais compactos o que é uma clara vantagem quando estes são instalados na plataforma OSGi através de uma rede de comunicações. Desta forma, a *framework* permite a partilha de *packages* (conjunto de classes *Java*) entre *bundles*. Qualquer *bundle* pode, se assim o desejar, importar ou exportar *packages*. Exportar significa que as *packages* ficam disponíveis para outros *bundles* que as podem importar se delas necessitarem.

A camada de modularização define portanto um modelo de delegação de carregamento de classes entre *class loaders* que permite a um *bundle* aceder às suas próprias classes, a classes da plataforma *Java* (tipicamente contidas em *packages* do tipo *Java.**), a classes da *framework* (contidas em *packages* do tipo *org.osgi.framework*) e *packages* exportadas por outros *bundles*. A figura 22 representa o modelo de delegação de carregamento de classes que existe entre o *class loader* de cada *bundle*, o *class loader* da *framework* (na figura 22 representado por *System Bundle class loader*) e o *class loader* do sistema (na figura 22 *Parent/System class loader*), modelo este definido pela camada de modularização.

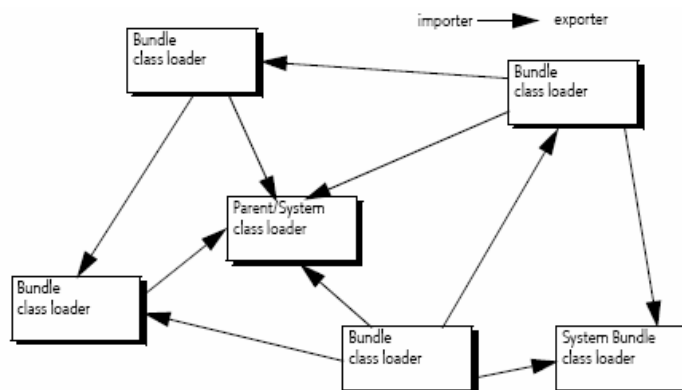


Figura 22: Modelo de delegação entre *class loaders* da *framework* OSGi [113].

A camada de modularização é activamente utilizada pela camada que implementa a gestão do ciclo de vida dos serviços, que é clarificada de seguida.

3.2.3.2.1.3 Camada de gestão do ciclo de vida dos serviços

Como já referido, um *bundle* apresenta-se sobre a forma de um ficheiro JAR. Este, para além de classes *Java* e outros recursos, contém também um ficheiro denominado de *Manifest* (*META-INF/MANIFEST.MF*). Este ficheiro é por norma incluído em todos os ficheiros de arquivo JAR (sejam *bundles* ou não) e contém informação acerca dos conteúdos do ficheiro JAR, informação essa que é guardada em cabeçalhos. Alguns desses cabeçalhos estão predefinidos na especificação dos ficheiros JAR [121], todavia as especificações OSGi definem novos cabeçalhos que transmitem à *framework* informação útil acerca do *bundle*, como por exemplo as *packages* que o *bundle* pretende exportar, as que pretende importar, versão do *bundle*, entre muitos outros. Esta informação é fundamental para a *framework* resolver, iniciar ou parar um *bundle*, ou por outras palavras, para gerir o ciclo de vida (conjunto de estados) de um *bundle*. A figura 23 apresenta os vários estados possíveis no ciclo de vida de um *bundle*.

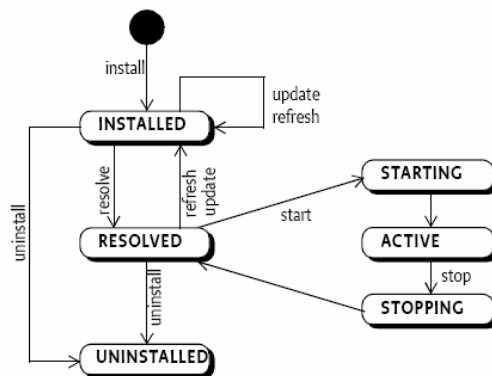


Figura 23: Diagrama de estados de um *bundle* [113].

O ciclo de vida de um *bundle* inicia-se com a sua instalação na plataforma. Para tal a *framework* disponibiliza métodos (invocados normalmente por um *Management bundle* – ver secção 3.2.3.3) que aceitam como parâmetro um URL (*Universal Resource Locator*) [122], URL este que indica a localização do ficheiro JAR a ser instalado. O ficheiro pode residir num servidor *web* remoto ou no sistema de ficheiros local. Após o ficheiro JAR ser transferido para a plataforma de serviços, a *framework* analisa os cabeçalhos presentes no ficheiro *Manifest*, bem como o código do *bundle* e instala-o na plataforma de serviços. Concluída esta operação o *bundle* entra no estado *Installed* (Instalado). A invocação do método *resolve*, inicia um processo através do qual a *framework* verifica se as *packages* que o *bundle* importa estão presentes na plataforma. Em caso afirmativo o *bundle* passa ao estado de *Resolved* (Resolvido). A partir deste momento, o *bundle* pode ser iniciado, o que acontece quando o método *Start* é invocado. Uma vez que a implementação do método usado para iniciar e parar cada *bundle* (métodos *Start* e *Stop*) difere de *bundle* para *bundle*, de acordo com as funcionalidades que cada *bundle* disponibiliza, a especificação OSGi apenas define (através de um interface *Java* denominada de *BundleActivator*) que os *bundles* poderão implementar um método para iniciar e um método para parar a sua própria execução. A classe *Java* que implementa estes métodos é depois referenciada num cabeçalho próprio do ficheiro *Manifest*.

Assim, o método *Start* é responsável pela alocação de recursos necessários à execução do *bundle*, lançamento de *threads*, inicialização de variáveis, registo de serviços (ver secção 3.2.3.2.1.4) que o *bundle* possa implementar, entre outros. Até que o método *Start* retorne (após executar as referidas tarefas) o *bundle* permanece no estado *Starting* (A iniciar). Quando o método retorna, o *bundle* transita automaticamente para o estado *Active* (Activo), estado em que permanece até o método *Stop* ser invocado. Quando este método é invocado qualquer *thread* lançada pelo *bundle* deve ser terminada e qualquer serviço implementado pelo *bundle* deve ser removido do registo de serviços. Após o retorno do método *Stop*, o *bundle* transita automaticamente para o estado resolvido (*resolved*). Uma vez neste estado o *bundle* pode ser actualizado ou removido: o processo de actualização suporta a migração de uma versão do *bundle* para uma mais recente. Por sua vez, a remoção implica a remoção física do *bundle* e de todos os recursos a ele associados da plataforma de serviços. Enquanto a operação de remoção está a ser efectuada o *bundle* encontra-se no estado não instalado (*Uninstalled*), assim que operação é concluída a *framework* apresenta-se como se o *bundle* nunca tivesse sido instalado.

Assim, a camada que implementa a gestão do ciclo de vida permite a um *bundle* que tenha permissões para efectuar tarefas administrativas (por exemplo *Management bundle*) controlar o ciclo de vida dos *bundles* que residem na *framework* e também a instalação de novos *bundles*.

3.2.3.2.1.4 Registo de serviços

Uma característica particular da plataforma de serviços OSGi, não encontrada numa plataforma *Java* comum, é a sua natureza dinâmica, ou seja, o conjunto de *bundles* instalado na plataforma pode variar num curto espaço de tempo. Portanto, para além de permitir a partilha estática de classes *Java* entre *bundles* (o que é suficiente para um ambiente *Java* regular), a plataforma OSGi permite também a partilha dinâmica de funcionalidades (serviços) entre *bundles*.

Um serviço é especificado por uma interface *Java* (que contém os métodos públicos do serviço) e implementado por uma ou mais classes *Java*. A interface que define o serviço e as classes que o implementam pertencem, tipicamente, a *packages* distintas sendo ambas incluídas num *bundle*. Uma vez o *bundle* na plataforma, é possível a este registar em qualquer instante (desde que no estado a iniciar ou no estado activo) o serviço que implementa junto do registo de serviços da *framework*. Para tal, através de um método específico, indica qual o nome da interface de serviço que implementa e um objecto de serviço (instanciação da classe *Java* que implementa a interface *Java* do serviço). Opcionalmente propriedades podem ser também incluídas no processo de registo do serviço. Uma propriedade não é mais que um par nome da propriedade/valor, cujo objectivo é facultar informação adicional acerca do serviço ao qual se refere. Considerando por exemplo um serviço de impressão, propriedades como a marca e modelo da impressora, a resolução máxima de impressão, o tipo de impressão, a localização da impressora descrevem em pormenor o serviço oferecido. Uma vez registado, o serviço passa a estar ao alcance dos restantes *bundles* em execução na plataforma. Importa referir que a *framework* permite que vários *bundles* registem diferentes implementações de um dado serviço. Fazendo a analogia com o exemplo do serviço de impressão apresentado acima, isto significa que os vários tipos de impressoras possíveis de conectar à plataforma OSGi, são representadas na plataforma de serviços por um serviço de impressão idêntico, mas com diferentes implementações e propriedades.

Desta forma, um *bundle* que pretenda utilizar um serviço registado na *framework* tem em primeiro lugar de conhecer a interface *Java* que define esse serviço. Conhecida essa interface, o *bundle* pode interrogar a registo de serviços acerca da existência de implementações desse

serviço. Por forma a especificar uma entre várias implementações a interrogação ao registo de serviços pode, para além da interface que descreve o serviço pretendido, conter um filtro que especifica o valor de umas ou mais propriedades. Em resposta a esta interrogação o *bundle* recebe zero, uma ou mais referências para objectos do serviço. Estas referências permitem ao *bundle* utilizar o serviço. Apesar de ser criada uma dependência entre o *bundle* que regista e o *bundle* que usa o serviço, nada impede que a partilha desse serviço termine em qualquer momento. Tal pode ocorrer porque o *bundle* que implementa o serviço indica explicitamente à *framework* que deseja terminar a partilha do serviço ou, em alternativa, porque o *bundle* acabou de transitar para o estado a parar. Ambos os casos implicam a remoção automática do respectivo serviço do registo de serviços.

Quando esta situação ocorre, o *bundle* que importa o serviço fica privado da sua utilização. Dependendo da “inteligência” que o *bundle* importador possui e da funcionalidade que pretende implementar, várias situações podem ocorrer: o *bundle* pode decidir que na ausência do serviço a sua execução deve ser terminada, pode tentar descobrir outra implementação do mesmo serviço na *framework* ou pode simplesmente continuar com a sua operação adaptando o seu comportamento à ausência do serviço. Em qualquer um dos casos é necessário que o *bundle* seja “notificado” que o serviço vai deixar de estar disponível. Tal é conseguido se o *bundle* “escutar” eventos gerados pela *framework*. Este tipo de eventos reportam o registo, remoção e alteração das propriedades de um serviço e podem ser usados por um *bundle* para descobrir a chegada, partida e alteração de serviços na plataforma.

A figura 24 apresenta uma sequência de acontecimentos numa plataforma de serviços OSGi, que pretende ilustrar os mecanismos de partilha de serviços entre *bundles*.

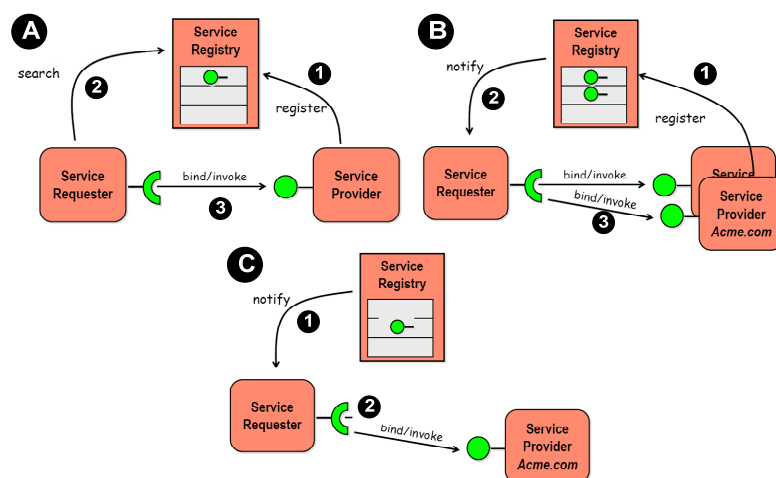


Figura 24: Partilha de serviços entre *bundles* na plataforma OSGi [123]

Os acontecimentos da figura 24 são de seguida descritos:

A	<ol style="list-style-type: none"> 1. Um <i>bundle</i> que implementa um dado serviço, aqui referido como serviço A, é instalado e activado na plataforma. O serviço é registado no registo de serviços. 2. Um outro <i>bundle</i>, que necessita do serviço A, é de seguida activado na plataforma. Afim de descobrir se algum dos serviços de que necessita está registado, o <i>bundle</i> interroga o registo de serviços. 3. Uma vez que o serviço A está registado, o <i>bundle</i> pode então invocar e utilizar o serviço A.
B	<ol style="list-style-type: none"> 1. Um novo <i>bundle</i> que implementa um outro serviço, aqui referido como serviço B, é instalado na plataforma. O serviço B é registado no registo de serviços. 2. Um evento é gerado para notificar, todos os <i>bundles</i> actualmente activos na plataforma, que um novo serviço foi registado. 3. Analisando o evento o <i>bundle</i> verifica que o serviço instalado é um serviço que também pretende utilizar, pelo que invoca o serviço B.
C	<ol style="list-style-type: none"> 1. O <i>bundle</i> que implementa o serviço A é removido da plataforma, pelo que é removido do registo. Esta remoção é anunciada através de um evento para os <i>bundles</i> na plataforma. 2. Através da análise do evento o <i>bundle</i> verifica que o serviço A vai ser removido e portanto liberta o serviço.

O facto de a plataforma OSGi permitir que um *bundle* utilize os serviços implementados por outros *bundles*, serviços esses que podem aparecer e desaparecer dinamicamente da plataforma, leva a que a plataforma OSGi seja muitas vezes descrita como uma plataforma cuja a arquitectura é dinâmica e orientada ao serviço SOA (*Service Oriented Architecture*) [124].

3.2.3.2.1.5 Camada de segurança

A camada de segurança é uma camada transversal a todas as outras na medida em que afecta o comportamento destas. As especificações da plataforma, no que diz respeito à segurança, são fortemente baseadas nas especificações de segurança da plataforma *Java 2* [125], pelo que não serão aqui expostas em grande detalhe. A arquitectura de segurança *Java 2* define o conceito de *Permission Classes* (classes de permissão), que protegem recursos (por exemplo um ficheiro) de acções específicas (por exemplo escrita, leitura). Estas classes tomam tipicamente o nome do recurso que protegem (por exemplo *File Permission Class*).

Assim as especificações OSGi definem dois tipos de classes de permissão:

- *Package Permission* – permissão de *package*: Como o próprio nome indica, esta permissão permite limitar a exportação e importação de *packages* apenas a *bundles* considerados de confiança.

- *Service Permission* – permissão de serviço: esta permissão visa proteger o registo e a obtenção de serviços, assegurando que apenas os *bundles* apropriados podem disponibilizar ou usar determinados serviços.

Na plataforma OSGi cada *bundle* tem um conjunto de permissões associadas, conjunto este que pode ser alterado em tempo real pelo *Management bundle* (ver secção 3.2.3.3).

3.2.3.3. Gestão remota da plataforma OSGi

No âmbito da plataforma de serviços OSGi, as actuais especificações definem como operações de gestão as seguintes tarefas:

- Gestão do ciclo de vida dos *bundles*: instalar, iniciar, actualizar, parar e remover *bundles*.
- Gestão da segurança: definição das permissões que cada *bundle* tem no que diz respeito à importação/exportação de *packages* e à partilha de serviços.
- Gestão da configuração: configuração da *framework*, bem como dos próprios *bundles*.
- Gestão de falhas: efectuar diagnósticos e corrigir de falhas.

Como mostrado na figura 19, num ambiente residencial, todas estas operações deverão ser efectuadas remotamente e por um operador de *gateway*. Esta característica iliba os utilizadores de qualquer tarefa de gestão da plataforma, sendo que a subscrição de serviços é a única tarefa de que os utilizadores se devem ocupar. Assim, a especificação OSGi define um modelo de gestão que faz uso das capacidades da plataforma de serviços para permitir a inter-operação entre plataformas de serviços e sistemas de gestão de uma forma não proprietária. A figura 25 apresenta as entidades definidas nesse modelo.

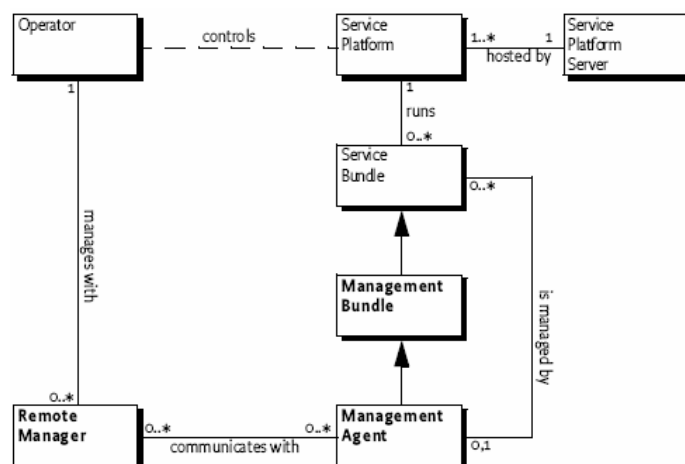


Figura 25: Modelo de gestão remota definido nas especificações OSGi [112].

A figura acima pode ser interpretada da seguinte forma: um operador de *gateway* é capaz de controlar uma plataforma de serviços através de um *Remote Manager* (gestor remoto) que comunica directamente com um *Management Agent* (agente de gestão, implementado na plataforma de serviços através de um *bundle* específico que possui permissões de administração, o *Management Bundle – bundle* de gestão). Por sua vez, este *bundle* de gestão é capaz de actuar sobre os vários *bundles* de serviço que são executados na plataforma e sobre a própria plataforma. Nesta podem coexistir vários agentes de gestão, cada um dos quais possibilita a comunicação com diferentes gestores remotos, sendo que num dado instante apenas um desses agentes pode efectuar tarefas de gestão na plataforma.

Um aspecto importante deste modelo é o facto de a especificação OSGi nada referir acerca do protocolo usado entre o gestor remoto e o agente de gestão. Em vez disso é definida uma API que está “visível” para o agente de gestão, agente este que é depois responsável por fazer a tradução entre os métodos implementados pelo protocolo de gestão usado e as funções da API. O propósito desta aproximação é permitir a interoperabilidade entre quaisquer sistemas de gestão (incluindo sistemas proprietários) e a plataforma de serviços.

3.2.3.4. Serviços normalizados OSGi

As especificações OSGi normalizam (definindo as suas interfaces *Java*) também um conjunto de serviços que deverão ser implementados sobre a forma de *bundles*. Importa referir que todos estes serviços são opcionais, na medida em que a sua disponibilidade na plataforma é considerada opcional e pode variar de instalação para instalação. No entanto, quando presentes, estes serviços standard podem ser utilizados pelos restantes *bundles* presentes na plataforma.

A lista de serviços normalizados tem vindo a ser alargada ao longo das várias versões da especificação e actualmente conta com cerca de vinte serviços e podem ser separados em quatro classes distintas: serviços da *framework*, serviços de sistema, serviços protocolares e outros serviços. Dada a extensão da lista de serviços, nas subsecções seguintes serão apenas apresentados alguns serviços de cada categoria. A lista completa dos serviços actualmente normalizados pode ser encontrada em [113] e [126].

3.2.3.4.1 Serviços da framework

Os serviços englobados nesta categoria actuam directamente sobre a operação da *framework*. Exemplos desses serviços são:

- *Permission Admin Service*: Como referido na secção 3.2.3.2.1.5, cada *bundle* em execução na plataforma tem um conjunto de permissões associadas. Estas permissões ditam por exemplo quais as operações que um dado *bundle* pode efectuar sobre um determinado ficheiro, quais as *packages* que pode importar, entre outras. Assim, o serviço de *Admin Permission* permite a um *Management Agent* administrar, de uma forma normalizada, permissões por defeito (que se aplicam a todos os *bundles*) bem como permissões particulares a determinados *bundles*.
- *Package Admin Service*: a secção 3.2.3.2.1.2 refere que *bundles* podem partilhar entre si *packages*, sendo assim criada uma dependência entre o *bundle* importador e o *bundle* exportador. Quando o *bundle* exportador é removido ou actualizado uma decisão tem de ser tomada no que diz respeito às *packages* exportadas por esse *bundle* e importadas por outros *bundles*. Assim, este serviço permite a um *Management Agent* obter informação acerca das *packages* partilhadas na *framework* e com base nessa informação criar políticas relativamente à partilha de *packages* entre *bundles*.
- *Start Level Service*: este serviço permite a um *Management Agent* controlar a ordem relativa de arranque e paragem dos *bundles* de uma plataforma de serviços, atribuindo a cada *bundle* um *start level*. O *Management Agent* pode, através deste serviço, alterar o *start level* de todos os *bundles* e também o *start level* actual da *framework* que inicia ou pára a execução dos *bundles*. Uma vez que apenas *bundles* com um *start level* menor ou igual ao *start level* actual da *framework* podem estar activos num determinado instante, este serviço permite ao *Management Agent* controlar em detalhe que *bundles* são iniciados e parados, bem como o instante em que tal acontece.

3.2.3.4.2 Serviços de sistema

Este tipo de serviços providencia o que se pode chamar serviços horizontais, necessários aos *bundles* instalados na plataforma. Alguns serviços deste tipo:

- *Log Service*: este serviço na verdade é composto por dois serviços: o *LogService* e o *LogReaderService*. O primeiro permite a um *bundle* o registo de informação de *log*, avisos, erros e semelhantes. O segundo permite a um *bundle* aceder a registos criados através do *LogService*.

- *Configuration Admin Service*: este serviço permite o envio de informação de configuração para um *bundle* que se encontra activo na plataforma de serviços. Esta informação é normalmente informação não disponível na altura em que o *bundle* foi desenvolvido, como por exemplo o número do porto IP a usar por um *bundle* que implementa um servidor HTTP.

3.2.3.4.3 Serviços protocolares

Estes serviços, como o próprio nome indica, traduzem um protocolo externo para um serviço OSGi. O serviço HTTP é um exemplo desse tipo de serviços.

- *HTTP Service*: o serviço HTTP é basicamente um serviço que permite a execução de *Servlets Java* [127]. Assim, *bundles* que contenham recursos baseados em tecnologias como HTML [128], XML [129] e *Servlets* (por exemplo uma interface com o utilizador) podem usar este serviço para tornarem esses recursos acessíveis através do protocolo HTTP.

3.2.3.4.4 Outros serviços

Esta secção apresenta alguns serviços que não se enquadram nas categorias de serviços OSGi anteriores.

- *XML Parser Service*: descrever dados através da linguagem XML (*Extensible Markup Language*) tem-se tornado uma prática comum nos últimos anos. Da mesma forma, é natural que os *bundles* usem também esta linguagem para descrever dados. Assim, um serviço que providencie um *XML Parser* aos vários *bundles* na plataforma evita replicação de funções nos vários *bundles*. Porém, nem todos os *parsers* são equivalente nas funções em que desempenham [130] e da mesma forma nem todos os *bundles* têm os mesmos requisitos em relação a um *parser*. Desta forma, este serviço permite a um *bundle* localizar na plataforma um serviço de *parser* com os requisitos desejados.
- *Device Access*: na verdade o *Device Access* não é considerado um serviço, mas sim apenas uma especificação. Esta especificação define como a interacção entre dispositivos externos, cuja presença pode variar ao longo do tempo, e a plataforma de serviços OSGi pode ser efectuada. Esta especificação foi utilizada nos desenvolvimentos efectuados sobre a plataforma OSGi e que estão descritos no capítulo 4 da presente dissertação. Assim, a especificação *Device Access* está exposta em detalhe no referido capítulo.

3.2.4. A plataforma TR-069: CPE WAN Management Protocol

Como constatado na secção 3.1.3, a variedade de dispositivos e serviços possíveis numa rede doméstica multi-serviço aumenta largamente a complexidade da sua gestão. Assim, o utilizador terá de ser libertado de qualquer tarefa relacionada com a sua gestão. A plataforma de serviços OSGi, descrita na secção anterior, oferece meios para que a gestão das aplicações, que implementam os serviços subscritos pelo utilizador, seja transferida para o domínio do operador. Contudo, muitas destas aplicações fazem uso de redes de comunicação para oferecerem ao utilizador os serviços pretendidos. Um serviço que depende dos recursos de uma rede de comunicações é normalmente conhecido como um *networked service* (todos os serviços apresentados na secção 3.1.1 são deste tipo). É então claro que a correcta operação destes serviços depende não só da aplicação que implementa o serviço mas também da rede de comunicações que este usa, ou seja, não só a aplicação deve ser gerida mas também os elementos de rede que poderão ter um impacto directo na performance do serviço.

As redes de acesso possuem desde há muito mecanismos que permitem este tipo de gestão. Contudo, o mesmo não é verdade no que se refere aos elementos de uma rede doméstica. Por este motivo o DSL Forum especificou toda uma plataforma para a gestão remota de equipamentos que estejam inseridos numa rede doméstica, normalmente conhecidos como CPEs (*Customer Premises Equipments*), a qual inclui o protocolo CWMP (*CPE WAN Management Protocol*) [131]. Este protocolo permite que o utilizador seja libertado de qualquer tarefa de gestão relacionada com os CPEs. Fazendo referência ao modelo apresentado na figura 2, um CPE poderá ser qualquer dispositivo que implemente um bloco funcional NT1, NT2, ST, TA ou vários destes blocos (por exemplo modem, *router*, *gateway* de acesso). O protocolo baseia-se num modelo cliente – servidor, onde o cliente é implementado no CPE e o servidor, denominado por ACS (*Auto-Configuration Server*), está localizado na rede do operador.

Como referido na secção 2.2.3, as redes de acesso actuais têm já mecanismos de auto-configuração que permitem a um CPE obter conectividade IP, pelo que este protocolo assume que o dispositivo que é objecto de gestão tem já conectividade IP com a rede do operador.

A figura 26 apresenta o enquadramento das entidades envolvidas no protocolo CWMP num ambiente xDSL:

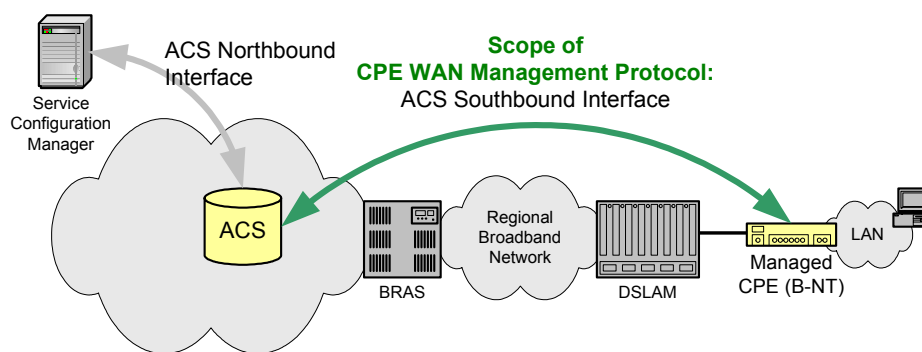


Figura 26: Enquadramento do protocolo CWMP num ambiente xDSL [131]

Como a figura 26 mostra, o ACS tem duas interfaces: a interface *southbound* e a interface *northbound*. A primeira permite a comunicação entre o ACS e o CPE (sendo portanto a interface em que o protocolo CWMP é usado), enquanto a segunda permite a deposição de configurações e ficheiros no ACS, bem como a obtenção de dados relativos a um dado CPE por parte de um gestor de configurações (na figura referido por *Service Configuration Manager*). Apesar de a figura apresentar como contextualização um cenário xDSL, esta plataforma pode ser usada sobre outro tipo de redes de acesso, uma vez que único requisito imposto é a existência de conectividade IP entre o ACS e o CPE.

Cada CPE apresenta um conjunto de pares nome/valor (parâmetros), que podem ser lidos e/ou escritos pelo ACS. Estes parâmetros estão normalmente organizados em grupos (objectos), objectos estes que por sua vez estão organizados de uma forma hierárquica. Este conceito de organização de parâmetros é semelhante ao usado numa MIB (*Management Information Base*) [132], SNMP (*Simple Network Management Protocol*) [133], pelo que o conjunto de parâmetros de um dado CPE será aqui referido como uma MIB.

O protocolo CWMP permite então as seguintes funcionalidades:

- Auto-configuração do CPE: esta funcionalidade permite o aprovisionamento automático de parâmetros do CPE (normalizados ou definidos pelos fabricantes). O primeiro pedido de aprovisionamento é iniciado pelo CPE após este obter conectividade IP. Consequentes operações de aprovisionamento podem ser iniciadas pelo CPE ou pelo ACS.
- Actualização de *software/firmware*: o protocolo também fornece métodos que permitem a transferência de ficheiros de *software/firmware* (com mecanismos que permitem identificar a versão a ser transferida e que permitem também o envio de uma notificação ao ACS do

sucesso ou falha da transferência) usando os protocolos HTTP, HTTPS [134], FTP ou TFTP. Para além de ficheiros individuais podem também ser transferidos pacotes de ficheiros juntamente com o conjunto de instruções que o CPE deve efectuar de forma a instalar o *software/firmware*. Opcionalmente estes pacotes de ficheiros podem ser digitalmente assinados por forma a garantir a sua integridade e autenticidade.

- Monitorização de performance e diagnósticos: o protocolo permite ao CPE transmitir para o ACS informação acerca da sua performance, estatísticas e estado actual, bem como informação que permite diagnosticar problemas de conectividade ou de serviço. Esta informação é baseada em parâmetros normalizados e/ou parâmetros definidos pelo fabricante do CPE.
- Segurança: o protocolo CWMP suporta o uso de SSL/TLS [135] nas comunicações entre o ACS e o CPE, o que permite não só a confidencialidade da comunicação e a integridade dos dados trocados, mas também uma autenticação mútua baseada em certificados entre o ACS e o CPE. Em alternativa o CPE pode também autenticar-se perante o ACS através de um segredo partilhado.
- Para implementar todas estas funcionalidades o protocolo CWMP assenta num conjunto de protocolos standard descritos na tabela 4.

Camada	Descrição
Aplicação de gestão CPE/ACS	Aplicação que utiliza o protocolo CWMP (não especificada na norma TR-069) no CPE e no ACS.
Métodos RPC	Conjunto de métodos RPC – <i>Remote Procedure Calls</i> , que são definidos pelo CWMP que permitem a leitura/escrita de parâmetros.
SOAP	Sintaxe baseada em XML aqui utilizada para encapsular RPCs sobre HTTP. A versão 1.1 [136] é requerida.
HTTP	Protocolo HTTP versão 1.1
SSL/TLS	O uso de SSL/TLS é recomendado mas não obrigatório. Estes protocolos adicionam segurança à ligação entre o ACS e o CPE. Caso seja usado o protocolo SSL a versão 3.0 é requerida, caso seja usado o protocolo TLS a versão 1.0 é requerida.
TCP/IP	Protocolo TCP/IP standard.

Tabela 4: Descrição da pilha protocolar usada pelo protocolo CWMP [131]

A figura 27 ilustra uma sessão típica entre um CPE e um ACS, onde o CPE obtém do ACS um conjunto de parâmetros.

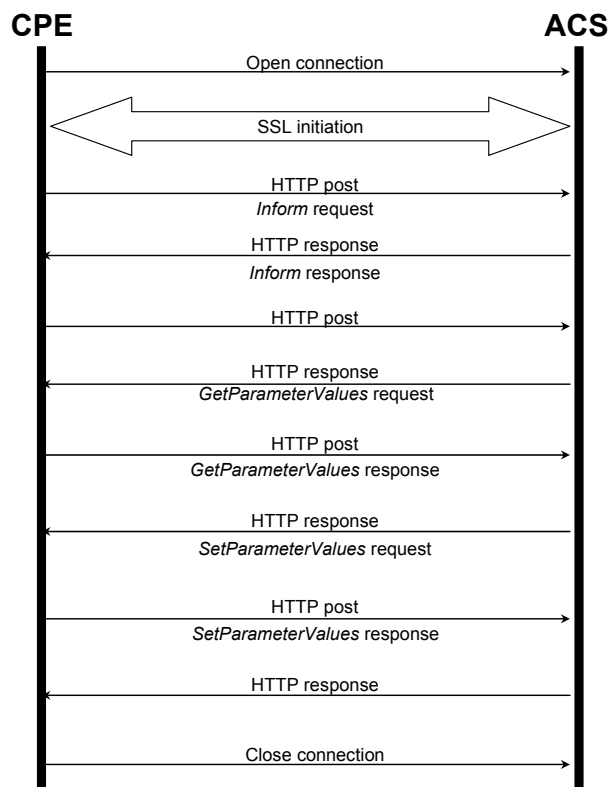


Figura 27: Troca de mensagens entre um CPE e um ACS [131].

Como mostrado na figura acima todas as mensagens são transmitidas sobre pacotes HTTP. Assim, a sessão inicia-se com o estabelecimento de uma ligação TCP por parte do CPE ao ACS, à qual se segue uma fase de negociação SSL/TLS opcional. Concluída essa negociação, é definido pela norma que o CPE deverá enviar uma mensagem de *Inform* para o ACS. Esta mensagem contém, entre outras, informação que identifica univocamente o CPE perante o ACS (por exemplo número de série), bem como informação que caracteriza o *hardware* e *software* do CPE. Como resposta o CPE recebe uma mensagem de *Inform Request*, que informa o CPE acerca de quantas mensagens poderá enviar/receber por cada método HTTP *post/response* (no exemplo figura 27 apenas uma mensagem é permitida). Como o CPE não tem nenhum pedido a efectuar ao ACS envia para este uma mensagem de HTTP *post* sem qualquer conteúdo. Por sua vez o ACS pretende obter o valor de alguns parâmetros actualmente configurados no CPE. Para isso envia uma mensagem de *GetParameterValues* onde são incluídos os parâmetros acerca dos quais o ACS deseja obter informação. O valor dos parâmetros pedidos são de seguida enviados pelo CPE para o ACS numa mensagem de *GetParameterValues Response*. Confrontando os valores actuais dos parâmetros enviados pelo CPE com o valor que esses parâmetros deveriam ter (através de uma configuração

armazenada numa base de dados a que o ACS tem acesso) e enviando uma mensagem de *SetParameterValues*, o ACS instrui o CPE para alterar o valor de alguns desses parâmetros. A confirmação da alteração dos valores dos parâmetros é obtida pelo ACS quando o CPE lhe envia uma mensagem de *SetParameterValues Response*. Uma vez que o CPE obteve já do ACS a configuração que necessitava, a recepção de uma mensagem de HTTP *response* sem conteúdo proveniente do ACS, indica ao CPE que este deve terminar a sessão.

A norma TR-069 para além de definir o protocolo CWMP e os métodos RPC que o CPE/ACS devem implementar, define também a versão 1.0 da MIB (acessível através dos métodos RPC) que um tipo específico de CPE, referido na terminologia DSL Forum como um IGD (*Internet Gateway Device*), dispositivo equivalente a um *gateway* de acesso, deverá implementar. Importa referir que a definição da MIB é suficientemente genérica para permitir que novos parâmetros ou mesmo objectos possam ser adicionados, por exemplo pelos fabricantes de equipamento. Esta MIB foi depois redefinida, clarificada e aumentada na versão 1.1, apresentada no documento TR-098 [137].

Com a versão 1.1. da MIB é então possível efectuar a auto-configuração de um conjunto de funcionalidades típicas de um *gateway* de acesso, nomeadamente: configuração da função de encaminhamento e *routing*, configuração da função de NAPT (caso o dispositivo opere como um *router*), configuração da tabela de *bridging* (caso dispositivo opere como uma *bridge*), configuração de classificadores de pacotes ou tramas, configuração de funções de policiamento, configuração de filas de espera, configuração da função de *scheduling*, configuração de ALGs, entre muitas outras. Analisando a extensa tabela de parâmetros apresentada na norma TR-098 facilmente se conclui que todas as funcionalidades típicas dos blocos funcionais NT2 e NT1, listadas na secção 2.1.2, são auto configuráveis através do protocolo CWMP se a versão 1.1 da MIB estiver implementada.

É importante, como nota final, vincar qual a principal vantagem do protocolo CWMP: o elemento gerido (CPE) contacta o elemento gestor (ACS) com a finalidade de obter toda a informação relevante para o seu processo de auto-configuração (ao contrário do que acontece com protocolos de gestão tradicionais como o SNMP). Isto significa, que um CPE que acabou de ser adquirido pelo cliente, após obter conectividade IP, pode auto configurar-se instantaneamente sem qualquer intervenção humana, o que é desejável e expectável em ambientes residenciais.

3.3. Observações finais

Novos serviços residenciais, que vão muito para além do tradicional acesso à Internet, começam agora a surgir e a serem disponibilizados pelos mais variados provedores de serviços. Porém, a correcta operação dos mesmos exige o suporte de funcionalidades específicas por parte das redes (de acesso e domésticas) sobre as quais estes mesmos serviços são distribuídos. Assim, no que diz respeito às redes de acesso, a possibilidade de vários provedores de serviços utilizarem uma mesma infra-estrutura de rede para distribuírem os seus serviços, suporte de QoS, sistemas de monitorização de rede, suporte de *multicast*, sistemas de autenticação, autorização e taxação mais sofisticados apresentam-se como exemplos de tais funcionalidades. Por sua vez, em relação às redes residenciais, funcionalidades necessárias nas redes de acesso como a QoS ou o uso simultâneo de uma infra-estrutura de rede por vários provedores de serviços, são características de alguma forma extensíveis também a redes domésticas futuras. Considerando a vastidão de dispositivos e serviços possíveis no ambiente residencial, a auto-configuração de equipamentos e aplicações é certamente a funcionalidade mais desejável e necessária num cenário de prestação de serviços residenciais futuro.

O facto de alguns dos terminais, que fazem a interface entre o serviço e o utilizador, não serem directamente compatíveis com as tecnologias *Ethernet/IP* (predominantemente usadas nas redes domésticas) implica a introdução na rede doméstica dos chamados *gateways* de serviços. A actual abordagem dos operadores e provedores de serviços em relação a estes dispositivos, passa pela instalação de *gateways* específicos a cada um dos serviços subscritos pelo cliente. Contudo, em oposição a esta abordagem, um *gateway* de serviços residenciais comum a vários serviços apresenta vantagens como a centralização dos processos de configuração de dispositivos e serviços, a partilha dos custos do *gateway* de serviços entre os vários provedores de serviços ou a possibilidade da construção de novos serviços baseados na interacção de outros serviços executados no mesmo *gateway* de serviços.

Considerando que a plataforma OSGi proporciona um ambiente de execução de serviços bem definido e documentado, permite a instalação e remoção de um serviço com uma perturbação mínima dos restantes serviços que se encontram na plataforma, encoraja a cooperação entre serviços, permite gestão remota do ciclo de vida e configuração desses mesmos serviços, esta apresenta-se como a melhor alternativa para integrar uma solução que contempla um *gateway* de serviços onde diferentes serviços, provenientes de diferentes fontes poderão ser depositados e executados. Todavia, a plataforma OSGi apenas contempla a gestão de

aplicações que implementam serviços subscritos pelo cliente, sendo que a correcta operação destes serviços requer normalmente também a gestão dos elementos da rede de comunicações utilizada por esses mesmos serviços. A rede de acesso encontra-se normalmente dotada de mecanismos que permitem efectuar esta tarefa, porém, o mesmo não é verdade em relação à rede doméstica. Assim, uma vez que o utilizador deverá ser ilibado de qualquer tarefa de gestão, o DSL Forum especificou o protocolo CWMP que possibilita a auto-configuração remota de equipamentos inseridos na rede doméstica. Pode-se então concluir que a presença de uma plataforma de serviços OSGi (alojada num *gateway* de serviços) na rede doméstica e o suporte do protocolo CWMP por parte dos vários elementos dessa mesma rede doméstica (incluindo o próprio *gateway* de serviços) permitem a entrega automatizada de serviços residenciais bem como a execução remota de todas as tarefas de gestão associadas a estes serviços.

Capítulo 4 – Validação de conceitos

A plataforma OSGi, apresentada no capítulo anterior, providencia mecanismos que permitem a execução, entrega e gestão remota de múltiplos tipos de serviços, sendo por isso ideal a sua inclusão num *gateway* de serviços. Por este motivo foi decidido que o protótipo de *gateway* residencial implementado no âmbito do projecto MUSE, no qual algum do trabalho da presente dissertação se enquadra, deveria suportar esta tecnologia.

Desta forma, a primeira secção deste capítulo apresenta o projecto MUSE, bem como os seus objectivos e enquadramento. Nesta primeira secção é ainda feita a contextualização de plataformas e *gateways* de serviços dentro do projecto MUSE.

As restantes secções deste capítulo são dedicadas ao enquadramento, descrição da implementação e avaliação de um serviço de monitorização médica remota implementado sobre a plataforma OSGi. Esta implementação pretende demonstrar as vantagens da plataforma OSGi e a sua aplicabilidade na construção de uma plataforma de serviços residenciais.

4.1. O projecto MUSE

O projecto MUSE (*Multi Service access Everywhere*) [9] é um projecto de investigação e desenvolvimento integrado no sexto programa quadro do IST (*Information Society Technologies*) [138] que conta com a participação de 34 parceiros, oriundos de 12 países europeus. Este projecto encontra-se dividido em duas fases (MUSE I e MUSE II), tendo a primeira decorrido

entre Janeiro de 2004 e Fevereiro de 2006 e estando a decorrer a segunda entre Janeiro de 2006 e Dezembro de 2007.

Os principais objectivos do projecto passam não só pela obtenção de uma visão consensual entre os vários participantes do que poderá ser uma futura rede de acesso/agregação de banda larga, mas também pela divulgação desta mesma rede junto de organismos de normalização como o ITU-T [6], ETSI-TISPAN [7], DSL Forum [3] e HGI [4]. Para além da elevada largura de banda (na ordem da centena de Mbit/s) a disponibilizar a cada assinante, características únicas e fundamentais desta rede futura serão os baixos custos de implementação e manutenção, a facilidade de distribuição de múltiplos serviços (rede multi-serviço) potencialmente disponibilizados por múltiplos provedores de serviços (rede multi-provedor), uma elevada taxa de penetração (deverá estar disponível a 85% da população europeia) e uma interacção adequada com a rede metropolitana e/ou de núcleo e também a rede residencial. A figura seguinte apresenta em pormenor os vários elementos desta rede futura que são objecto de estudo do projecto MUSE.

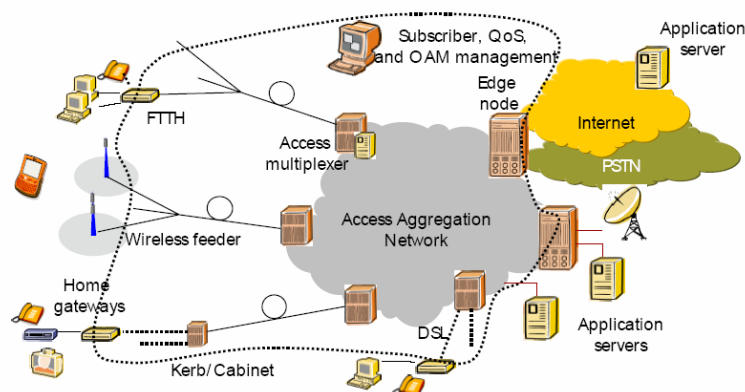


Figura 28: Diferentes elementos da rede de acesso e agregação endereçados no projecto MUSE [9].

Desta forma, no âmbito da primeira fase do projecto, para além da arquitectura de uma rede de acesso/agregação que permita a distribuição de serviços num regime multi-serviço/multi-provedor (com especial atenção aos nós de acesso e fronteira – na figura 28 *access multiplexer* e *edge node* respectivamente), estiveram também incluídos o estudo de tecnologias de primeira milha que possibilitem os débitos anunciados e a interacção da rede de acesso/agregação com a rede residencial (ou rede de SOHO, *Small Office Home Office*), com uma grande incidência no chamado *gateway* residencial (elemento através do qual é efectuada a referida interacção). Finalmente um conjunto de estudos técnico-económicos e um conjunto de testes em laboratório pretenderam validar economicamente e tecnologicamente, respectivamente, as

várias soluções encontradas. A primeira fase terminou com a descrição completa de uma arquitectura de rede que cumpre os requisitos impostos, bem como um conjunto de elementos base testados isoladamente em laboratório (plataforma de acesso⁴, tecnologias de primeira milha e *gateways* residenciais) que irão ser integrados e testados em conjunto na segunda fase.

Na segunda fase do projecto o âmbito foi alargado e contempla agora a adição de novas funcionalidades à arquitectura de rede especificada na primeira fase do projecto. Estas novas funcionalidades permitirão a distribuição de serviços multimédia avançados e também a convergência entre a rede fixa e móvel (FMC, *Fixed Mobile Convergence* [139]). A execução de testes laboratoriais integrados de todos os desenvolvimentos efectuados no projecto encerrarão a segunda fase do projecto.

Ao nível organizacional o projecto divide-se num conjunto de TFs (*Task Forces*) e SPs (*Sub Projects*), estes ainda divididos em vários WPs (*Work Packages*) alinhados com a área técnica de cada uma das TFs, como ilustrado na figura seguinte:

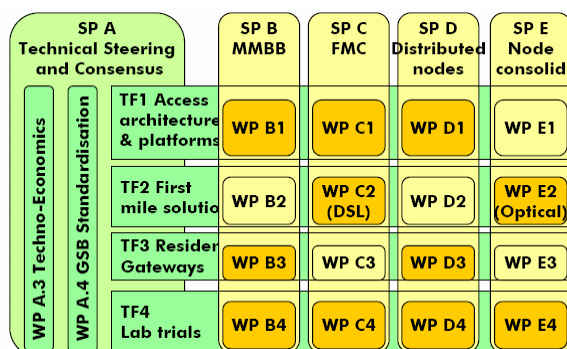


Figura 29: Organização interna do projecto MUSE, fase II [140]

O primeiro SP é denominado de SP A e é responsável por garantir a coerência técnica e o consenso de todo o projecto. Tal é conseguido com base nos estudos técnico-económicos (WPA.3), nas actividades que visam a promoção dos resultados obtidos pelo projecto junto de organismos de normalização (WPA.4 GSB, *Global System for Broadband*) e também com base nos resultados obtidos nas várias TFs. Por sua vez, cada um dos restantes SPs concentra-se num tópico de trabalho específico: o SP B dedica-se ao estudo das funcionalidades a serem incluídas nos elementos da plataforma de acesso e no próprio *gateway* residencial que

⁴ No contexto do projecto MUSE uma plataforma de acesso é definida como a combinação dos nós de acesso, dos nós fronteira e de todos os restantes elementos que fazem parte da rede de acesso/agregação e que permitem a distribuição de serviços.

permitirão a distribuição de serviços multimédia avançados, o SP C concentra-se na procura de soluções para o nó de acesso e para a primeira milha que permitam conectividade extremo a extremo e QoS num ambiente FMC, o SP D dedica-se ao estudo de cenários cuja tecnologia de primeira milha permite elevados débitos (tipicamente do tipo FTTx), com especial interesse no impacto nos nós de acesso e nos *gateways* residenciais e finalmente, o SP E promove estudos no sentido de baixar as despesas de operação da rede de acesso/agregação (OPEX, *Operational Expenditure*) através da implementação de uma XL PON (*eXtra Large Passive Optical Network*).

Os resultados obtidos em cada uma das áreas técnicas em cada um destes SPs são depois reportados, através da WP mais indicada, à TF dessa área técnica. Cada TF é depois responsável pela divulgação dos resultados obtidos e pela produção de contribuições para os organismos de normalização já referidos.

4.1.1. Plataformas e gateways de serviços residenciais no projecto MUSE

Apesar do estudo da rede doméstica se encontrar fora do âmbito do projecto MUSE, a definição e caracterização do *gateway* residencial (elemento responsável por fazer a interface entre a rede de acesso e a rede doméstica) encontra-se ainda dentro do âmbito do projecto. Como apresentado na secção 3.1.3, a definição de *gateway* residencial é bastante genérica e contempla a hipótese de este dispositivo implementar também a funcionalidade de um *gateway* de serviços. Assim, o protótipo desenvolvido durante a primeira fase do projecto MUSE (mais concretamente no âmbito da WP D3 [141]), contém a implementação de uma plataforma de serviços OSGi. Porém neste primeiro protótipo, a plataforma OSGi foi predominantemente usada para resolver problemas relacionados com a gestão e configuração remota do *gateway* residencial. Apesar de estas funções serem fundamentais para o correcto funcionamento do *gateway* e de a plataforma OSGi providenciar mecanismos que facilitam a execução das mesmas, não ficaram expostas de uma maneira clara todas as potencialidades da plataforma de serviços OSGi. Assim, para esta segunda fase, foi proposto o desenvolvimento de um serviço que demonstre as capacidades da plataforma de serviços OSGi e que irá ser incluído nos testes finais do *gateway* residencial. Esse serviço enquadra-se num cenário de aplicação médica e os pormenores da sua implementação e testes são descritos nas secções seguintes.

4.2. Serviço de monitorização médica remota

Esta secção pretende introduzir, enquadrar e descrever o serviço desenvolvido e que faz uso das potencialidades da plataforma OSGi. Para além do serviço, são também descritas funcionalidades, não directamente relacionadas com a plataforma de serviços OSGi, mas que completam todo o cenário em que o serviço se enquadra. Algumas opções relativamente à implementação dos vários módulos são ainda descritas nesta secção.

4.2.1. Enquadramento e descrição

Como descrito na secção 3.1.1.5, o aparecimento de sensores médicos de monitorização com capacidades de comunicação e a proliferação das ligações à Internet em banda larga têm vindo a possibilitar o aparecimento de um conjunto de serviços médicos até agora não possíveis. De entre estes novos serviços, destaco o serviço de monitorização remota que possibilita a um hospital (ou entidade semelhante) receber medições médicas elementares (por exemplo a pressão arterial, o ritmo cardíaco, entre outras) referentes a um dado paciente, sem que este esteja fisicamente presente. Para tal, o paciente simplesmente utiliza o dispositivo médico adequado no seu ambiente residencial e através de uma ligação à Internet em banda larga, os dados recolhidos por este dispositivo são enviados para a localização remota.

A figura 30 ilustra a utilidade deste tipo de serviços médicos: um paciente que foi submetido a uma intervenção cirúrgica, estará sujeito a um período de recuperação durante o qual deverá permanecer em repouso. Para além desta restrição, o processo de recuperação do paciente apenas requer que algumas medições médicas de rotina sejam efectuadas diariamente. Nestas condições não existe nenhuma vantagem adicional em manter o paciente no hospital durante o período de recuperação, pelo que este poderá recuperar da intervenção cirúrgica em sua casa. Caso o paciente possua na sua residência um *gateway* residencial que implemente uma plataforma de serviços OSGi, então equipamento simples mas capaz de efectuar medições médicas triviais pode ser instalado na casa do paciente de uma maneira rápida e cómoda. Tipicamente este tipo de equipamento médico apresenta interfaces de comunicação, também elas simples, tais como interfaces série (compatíveis com o standard RS-232 [142]), USB, IrDa [143] ou mesmo *Bluetooth* [41], pelo que poderá comunicar com o *gateway* residencial e com a plataforma de serviços OSGi de uma forma normalizada. Por sua vez, o *gateway* residencial, através da ligação de banda larga de que dispõe, poderá submeter os dados recebidos do

equipamento a uma base de dados hospitalar.

Desta forma, a instalação do equipamento médico na residência do paciente resume-se à sua ligação ao *gateway* residencial através de uma das interfaces de comunicação disponíveis. Uma vez efectuada esta tarefa, todos os *bundles* OSGi necessários para que o *gateway* comunique com o equipamento médico e submeta os dados recebidos do equipamento para uma base de dados (possivelmente alojada no hospital) podem ser instalados e activados remotamente, usando as facilidades de gestão remota da plataforma OSGi. Assim, pessoal especializado pode diariamente consultar as leituras efectuadas, sem que para isso tenha de manter o paciente no hospital ou o mesmo tenha de aí se deslocar. O equipamento irá permanecer na residência do paciente durante todo o processo de recuperação. Terminado o processo de recuperação, este é removido da residência do cliente e simultaneamente os *bundles*, previamente instalados, são removidos da plataforma OSGi presente no *gateway* residencial do paciente. A figura seguinte apresenta de uma forma sistematizada as entidades e as interacções entre as mesmas, para que o serviço descrito possa operar correctamente.

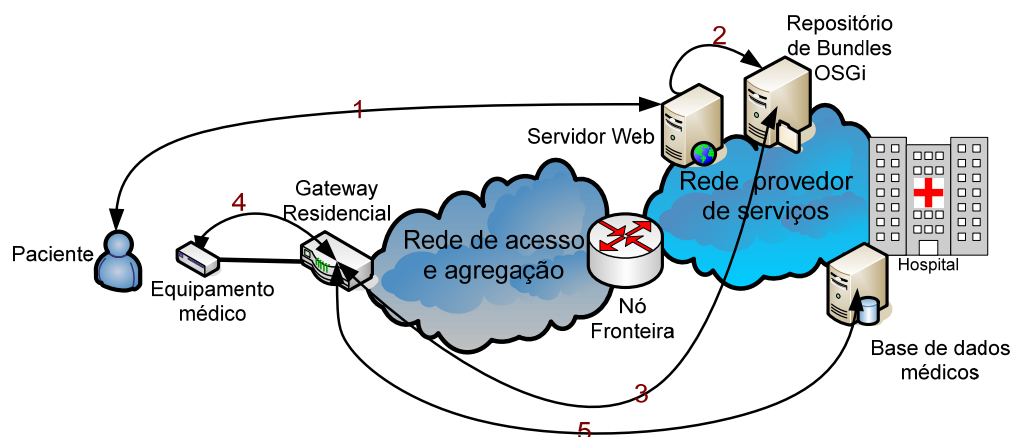


Figura 30: Interações entre as entidades envolvidas no serviço de monitorização médica remota

- 1 O equipamento médico é ligado ao *gateway* residencial. Uma vez efectuada esta operação, o provedor de serviços (possivelmente um provedor de serviços com que a unidade hospitalar mantém um acordo contratual) é contactado. Este ponto de contacto entre o cliente (paciente) e o provedor de serviços num caso real poderá ser efectuada de várias formas (ex. telefone, página *web*, etc.). Neste exercício assume-se que o ponto de contacto é concretizado sobre a forma de uma página *web* segura.
- 2 A referida página *web* apresenta um conjunto de opções que reflectem os vários tipos de equipamentos médicos que podem ser instalados, bem como tipos de medições a efectuar. A escolha de uma ou mais opções é traduzida em operações de gestão remotas, levadas a cabo por um gestor remoto da plataforma OSGi, que irão resultar na transferência, instalação e activação de *bundles* OSGi na plataforma de serviços do cliente.

- 3 De acordo com as opções seleccionadas, os *bundles* necessários são então transferidos de um repositório de *bundles* para a plataforma OSGi do cliente, onde são instalados e activados.
- 4 O equipamento médico pode assim estabelecer comunicação com o *gateway* residencial e transmite periodicamente para este as leituras médicas efectuadas.
- 5 O serviço previamente instalado sobre a forma de *bundles*, contém toda a informação necessária para estabelecer contacto (ex. URL, *login*, *password*) com a base de dados hospitalar e submeter para esta os dados recebidos do equipamento médico. Após o término da observação médica (que poderá levar dias, semanas ou meses) o equipamento médico poderá ser retirado da residência do paciente e os respectivos *bundles* removidos da plataforma de serviços correspondente.

4.2.2. Opções e requisitos de implementação

A descrição e enquadramento apresentados na secção anterior definem de uma forma geral as funcionalidades que o serviço de monitorização a desenvolver deverá ter. Porém, para se iniciar rapidamente a implementação dos vários módulos de *software*, as seguintes condições e opções foram consideradas:

- Equipamento médico a utilizar: ao início do processo de implementação não estava disponível um equipamento médico real. Assim, decidiu-se implementar uma aplicação em *Java* que é executada num PC e simula esse mesmo equipamento médico.
- Comunicação equipamento médico/*gateway* residencial: uma vez que a quantidade de dados a transferir entre os dois dispositivos é tipicamente baixa, a comunicação é efectuada através de uma porta série com o protocolo RS-232. Equipamentos que efectuem medições médicas simples apresentam tipicamente interfaces de transmissão de dados simples, como portas série, portas infravermelhos, entre outras.
- Comunicação *gateway* residencial/base de dados médicos: a base de dados (formato *mySQL* [144]) está acessível através de um *HTTP Servlet* implementado na tecnologia *Java*. Uma vez que a informação médica é confidencial, o protocolo *HTTPS* deverá ser usado na comunicação efectuada entre o *gateway* residencial e a base de dados. Uma outra opção seria a utilização do protocolo da camada de rede *IPSec*.
- Comunicação gestor remoto/agente de gestão da plataforma OSGi: as especificações *JMX (Java Management eXtensions)* [145] [146] definem uma arquitectura e uma API que no seu conjunto permite a qualquer aplicação *Java* ver os seus recursos (aplicações, serviços) geridos remotamente. Isto significa que com esta arquitectura a plataforma OSGi pode ser gerida remotamente, como referido na secção 3.2.3.3. Assim, a comunicação entre o gestor remoto

e o agente de gestão será efectuada recorrendo a esta tecnologia.

4.2.2.1. A implementação da framework OSGi OSCAR

Existem actualmente várias implementações das várias versões das especificações OSGi, sendo algumas *open source* e outras produtos comerciais (ver Anexo III). Pelo facto de a implementação OSCAR [147] ser do primeiro tipo, possuir uma grande comunidade de utilizadores e ter já disponível implementações de alguns dos serviços OSGi normalizados, bem como de outros tipos de serviços, foi decidido que esta seria a implementação a usar no âmbito da WP D3 do projecto MUSE. Uma vez que os desenvolvimentos efectuados serão incluídos nessa mesma WP, por questões de compatibilidade foi também utilizada a implementação OSCAR no trabalho desta dissertação.

A implementação OSCAR foi uma iniciativa de *Richard S. Hall* e tinha por principal objectivo construir uma plataforma de serviços *open-source*, compatível com as especificações OSGi. Para além de cumprir com as especificações OSGi, esta implementação deveria também resolver automaticamente as dependências (por exemplo transferindo-as de um repositório central de *bundles*) dos *bundles* a serem instalados. Para que a plataforma pudesse transferir e instalar *bundles* automaticamente, o autor criou o OBR (*Oscar Bundle Repository*) [148]. Este repositório consiste basicamente numa página *web* que aloja um recurso XML que descreve o conjunto de *bundles* que fazem parte do repositório. Esta descrição permite então que uma plataforma OSCAR consiga determinar as dependências (outros *bundles*) de um determinado *bundle*, as transfira e instale automaticamente. Inicialmente exclusivo da implementação OSCAR, rapidamente outras implementações começaram a manter repositórios de *bundles* baseados no formato OBR. Motivada pelo sucesso do OBR, a aliança OSGi produziu em 2005 uma RFC [149] que normaliza o repositório de *bundles*.

A última versão estável da implementação OSCAR, versão 1.0.5 (adoptada nos desenvolvimentos da presente dissertação), data de 13 de Maio de 2005 e implementa a versão 3 da especificação OSGi. Após o lançamento desta versão uma versão *alpha* 2.0 foi ainda lançada, mas entretanto o código que compunha esta versão foi doado ao projecto *Apache Felix* [150]. A implementação *Apache Felix* é portanto a continuação da implementação OSCAR. Actualmente ainda num processo de “incubação” na *The Apache Software Foundation* [151], o projecto pretende implementar uma plataforma OSGi completamente compatível com a versão 4 da especificação OSGi. Apesar de existir já uma implementação preliminar de

parte da norma, a inexistência de implementações de serviços OSGi normalizados torna a sua utilização um pouco limitada. Desta forma, uma migração para esta implementação será apenas aconselhada quando uma versão estável da mesma estiver disponível.

4.2.2.2. Bundles Base

A instalação base da plataforma OSCAR inclui apenas um pequeno número de *bundles*, tal como ilustrado na figura seguinte:

START LEVEL 1		ID	State	Level	Name
[0]	[Active]	[0]	System Bundle (1.0.5)
[1]	[Active]	[1]	Shell Service (1.0.2)
[2]	[Active]	[1]	Shell TUI (1.0.0)
[3]	[Active]	[1]	Bundle Repository (1.1.2)

Figura 31: *Bundles* instalados por defeito na implementação OSGi OSCAR

- *System Bundle*: Este *bundle* representa a própria *framework* OSGi e portanto disponibiliza todos os serviços e funções descritos na secção 3.2.3.2.1.
- *Shell Service*: de forma a interagir com o OSCAR é necessário uma *shell* que permita o envio de comandos para a *framework*. Uma vez que a especificação OSGi nada diz acerca disto, a implementação OSCAR define um serviço (disponibilizado por este *bundle*) para criar e executar comandos.
- *Shell TUI*: este *bundle* usa o serviço disponibilizado pelo *bundle Shell Service* para implementar um conjunto de comandos e criar assim uma interface textual através da qual se pode interagir com o OSCAR.
- *Bundle Repository*: *bundle* que permite o acesso ao OBR. O acesso ao OBR pode ser efectuado de duas formas: através de um serviço registado por este *bundle* (e que portanto pode ser usado por outros *bundles*) e através do comando “obr” também implementado pelo *Bundle Repository*.

4.2.2.3. Outros bundles utilizados

Como referido na secção 4.2.2, a comunicação entre o gestor remoto e o agente de gestão da plataforma forma é conseguida através da tecnologia JMX. De uma forma muito breve, a arquitectura JMX define três entidades: a entidade *Managed Bean* ou *Mbean*, a entidade *Mbean*

Server e os conectores/adaptadores. A figura 32 apresenta o enquadramento destes elementos na arquitectura JMX.

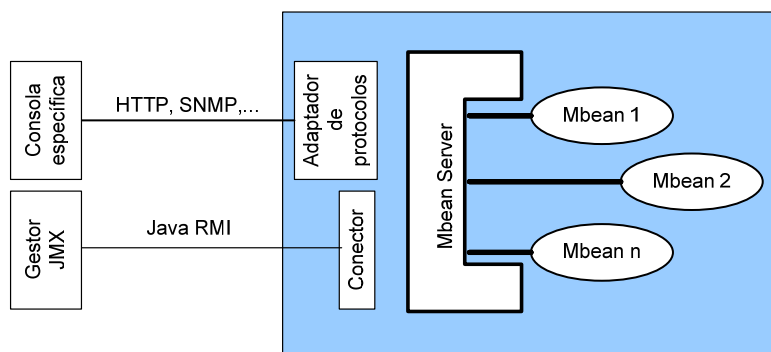


Figura 32: Arquitectura JMX

Um *Mbean* pode ser visto como sendo uma representação normalizada de um recurso que se pretende gerir, o *Mbean Server* actua sobre os *Mbeans* (que se registam no *Mbean Server*) e desempenha a função de agente de gestão da plataforma e finalmente os conectores/adaptadores permitem o acesso remoto ao *Mbean Server* por parte de um ou mais gestores remotos. Diferentes conectores/adaptadores implementam o acesso através de diferentes protocolos.

Por forma a gerir a plataforma OSGi OSCAR através desta tecnologia, é necessária a instalação de um conjunto de *bundles* que implementa uma “camada” JMX na mesma. Os *bundles* que constituem esse conjunto são brevemente descritos de seguida:

- *JMX-MX4J Agent Service*: este *bundle* implementa essencialmente um serviço que representa um *MbeanServer*, no qual os vários *Mbeans* são registados.
- *OSGi Mbean*: um dos recursos que se pretende gerir é a plataforma OSGi. Assim, este *bundle* inclui um *Mbean* que representa a plataforma de serviços OSGi.
- *Shell Mbean*: à semelhança do *bundle Shell TUI* este *bundle* utiliza o serviço disponibilizado pelo *bundle Shell Service* para construir um outro serviço que implementa comandos que permitem interagir com o OSCAR. A diferença, obviamente, reside no facto de estes comandos poderem ser invocados remotamente.
- *Service Introspector*: para além da plataforma OSGi, serviços implementados por outros *bundles* que vão sendo instalados na plataforma ao longo do tempo devem também poder ser geridos remotamente. Desta forma, este *bundle* implementa um serviço que analisa continuamente os serviços que são registados na plataforma e dessa análise retira informação

que possibilita a representação desses serviços sobre a forma de um *Mbean*.

- *Mbean factory*: este *bundle*, como o próprio nome indica, regista um serviço que representa uma *Mbean factory*. A sua função é construir *Mbeans* de acordo com dados recebidos do serviço *Service Introspector*.
- *Service notifier*: o serviço *Service notifier* escuta a introdução e remoção de serviços do registo de serviços da *framework* OSGi. De acordo com o evento ocorrido, o serviço *service notifier* regista ou remove os *Mbeans* respectivos (gerados pela *Mbean Factory* com informação recolhida pelo *Service Introspector*) do *Mbean Server*.
- *JMX RMI connector auth*: Este *bundle* implementa um conector que permite a interacção entre o gestor remoto e o agente de gestão da plataforma OSGi através da tecnologia RMI (*Remote Method Invocation*) [152].
- *JMX RMI registry*: de acordo com a definição da tecnologia RMI, para que seja possível interagir com a plataforma através de RMIs é necessária a presença de um *RMI registry*. Este *bundle* implementa um serviço que representa essa entidade.

Para além dos referidos *bundles*, um *bundle* que implementa o *Log Service* (ver secção 3.2.3.4.2) e um *bundle* que implementa o *Device Manager*, definido na especificação *Device Access* (ver secção 4.2.2.4), foram também utilizados. Ambos os *bundles* estão acessíveis através do OBR.

4.2.2.4. A especificação OSGi Device Access

O cenário descrito na secção 4.2.1 requer que o *gateway* de serviços (incluído no *gateway* residencial) comunique com um dispositivo externo ao próprio *gateway* residencial. Desta forma, a plataforma OSGi apresenta mecanismos que permitem precisamente comunicar e controlar dispositivos ligados através de uma dada tecnologia (por exemplo USB, RS-232, entre outras) ao *gateway* de serviços, mas que não fazem parte integrante do mesmo. Isto significa que a presença desses dispositivos pode variar ao longo do tempo. Este conjunto de mecanismos, brevemente descritos de seguida, é conhecido como DA (*Device Access*). Importa referir que estes mecanismos são considerados como opcionais e poderão não estar presentes em todas as implementações ou instalações da plataforma.

De uma forma sintética, a especificação DA permite a representação de um dispositivo físico na *framework* através de um serviço OSGi, o que significa que os *bundles* instalados na

plataforma “vêm” esse dispositivo físico apenas como mais um serviço registrado na plataforma. A presença desse serviço no registo de serviços da *framework* significa que o dispositivo está conectado ao *gateway* de serviços, enquanto que a ausência do serviço significa a ausência do dispositivo. Para além desta funcionalidade, a especificação DA define ainda um processo através do qual é possível encontrar o *driver* mais adequado para comunicar com cada dispositivo. No seu conjunto a especificação DA define quatro tipos de entidades: o DM (*Device Manager*), o(s) DS (*Device Service*)(s), o(s) DLS (*Driver Locator Service*)(s) e o(s) DrS (*Driver Service*)(s), cada uma das quais é descrita de seguida:

- *Device Service (DS)*: Como próprio nome indica, um DS é primeiro de tudo um serviço OSGi (e como tal é também um *bundle*) que é registado na *framework* (com um conjunto de propriedades específicas). Este serviço representa dispositivos físicos que podem ir de uma simples porta de comunicações (por exemplo uma porta série) a dispositivos mais complexos (por exemplo um modem conectado à porta série). Quando o dispositivo físico está presente, o DS correspondente é registado na *framework* e a operação inversa acontece quando o dispositivo é removido. Como todos os serviços na plataforma OSGi, um DS é definido através de uma interface *Java*, interface esta que apresenta métodos que permitem interagir com o dispositivo representado. Importa referir que no exemplo referido anteriormente dois DS seriam registados na *framework* quando o modem estivesse conectado à porta série: um DS que representa e possibilita a comunicação através de uma porta série e outro DS que representa e possibilita a comunicação com o modem.
- *Driver Service (DrS)*: importa desde já referir que o termo *driver* no contexto da especificação *Device Access* não se refere a *drivers* de baixo nível (normalmente ao nível do sistema operativo). A especificação DA assume que esse tipo de *drivers* altamente específicos ao sistema operativo em uso estejam devidamente instalados e configurados de antemão. Assim, no contexto DA, um *driver* é uma entidade que regista na *framework* um ou mais *Device Services* contidos no *bundle* que implementa o DrS e que pretendem refinar (oferecendo uma semântica mais rica e/ou um nível de abstracção mais elevado) um *Device Service* já registado na plataforma. No exemplo anterior o DS que possibilita a comunicação com o modem e que vem refinar o DS que representa a porta série, seria registado na plataforma por um DrS.
- *Driver Locator Service (DLS)*: um DLS (podem estar vários simultaneamente registados) sabe onde encontrar *bundles* que implementem um dado DrS necessário para refinar um DS.

- *Device Manager (DM)*: o DM é o coordenador de todas as acções efectuadas pelas entidades DS, DrS e DLS.

A figura seguinte representa o processo de refinamento de um DS e as várias interacções que ocorrem entre as entidades definidas na especificação *Device Access*.

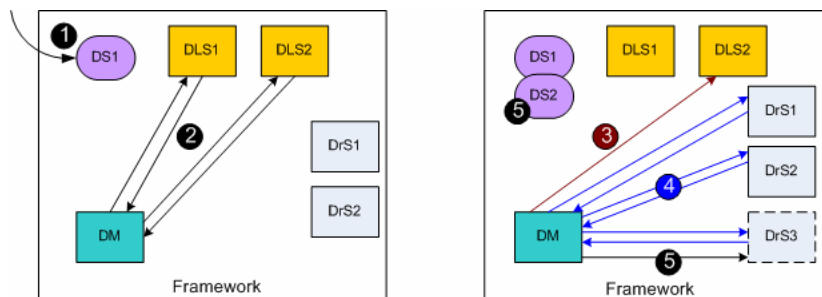


Figura 33: Processo de refinamento de um *Device Service* [116]

Um novo *Device Service* é registado na *framework* (figura 33 passo 1). O *Device Manager*, que está continuamente a observar o registo de *Device Services*, detecta que um novo DS (DS1) foi registado e portanto interroga todos os *Driver Locator Services* acerca do seu conhecimento de *Driver Services* que possam refinar o DS que acabou de se registar. Dos DLSs presentes, apenas o DLS2 responde afirmativamente (figura 33 passo 2). Com base nesta resposta, o DM interroga o DLS2 acerca da localização do DrS que contém o DS que o levou a responder afirmativamente à solicitação anterior. Este pedido resulta na instalação e activação do DrS3 na *framework* (figura 33 passo 3). Uma vez que todos os DrSs que podem refinar o DS1 já estão registados na plataforma, o DM questiona cada um deles acerca da sua capacidade para refinar o DS que acabou de se registar (figura 33 passo 4). Com base nas respostas obtidas o DM elege o DrS3 como o que deve refinar o DS1, o que resulta no registo de um novo *Device Service*, o DS2, por parte do DrS3 (figura 33 passo 5). Uma vez registado o DS2, um processo de refinamento semelhante ao que agora termina, poderá ser iniciado para este novo *Device Service*. Este processo irá continuar até que não haja DLSs que demonstrem conhecimento acerca de DrSs que possam refinar o DS que acabou de se registar.

A descrição anterior deixa todavia duas questões em aberto: Como é que um DLS determina que um dado DrS pode ou não refinar o DS em questão e como é o DM elege um entre os vários DrSs que mostram capacidade para refinar o DS. A resposta à primeira questão reside num conjunto de propriedades que todos os *Device Services* possuem quando são registados na *framework* e que estão disponíveis aos vários DLSs interrogados pelo DM. Exemplos dessas propriedades são a categoria do dispositivo que o DS representa (ex. série, USB), a classe do

dispositivo (por exemplo áudio, vídeo, entre outros), o modelo, o fabricante, bem como outras propriedades que podem ser definidas. Com base nestas propriedades cada DLS determina se algum dos DrSs acerca dos quais tem conhecimento pode ou não refinar o DS em questão.

A resposta à segunda questão, reside nesse conjunto de propriedades característico a cada categoria de *Device Service*: quando o DM interroga cada um dos DrSs acerca da sua capacidade para refinar o DS, cada um destes licita um valor numérico com base no número de propriedades cujo o valor coincide com o valor da mesma propriedade do DS a refinar. O DrS que licite o maior valor é eleito para registar o DS correspondente, sendo que no caso de dois ou mais licitarem um mesmo valor um desempate tem de ser efectuado. Para tal é consultado em primeiro lugar o valor de uma propriedade opcional que define a preferência de um dado serviço em relação a outros. Caso esta propriedade não esteja presente ou o seu valor seja igual entre os vários DrSs, é eleito o DrS que estiver à mais tempo registado na *framework*. Importa referir que este comportamento na escolha do DrS mais adequado, reflecte um comportamento por defeito. A entidade denominada de *Driver Selector Service* (introduzida apenas na segunda versão da especificação *Device Access* [111]) permite personalizar este comportamento.

4.2.3. Módulos desenvolvidos

Esta secção descreve a implementação dos vários módulos de *software* que compõem o cenário descrito na secção 4.2.1. A descrição encontra-se separada em três secções: a primeira refere-se a módulos de *software* implementados sobre a plataforma OSGi e portanto destinados a um *gateway* de serviços, a segunda refere-se a um módulo de *software* que é executado num PC directamente conectado ao *gateway* de serviços e finalmente a terceira refere-se a um módulo de *software* que será executado algures no domínio do provedor de serviços.

4.2.3.1. Bundles OSGi

Os *bundles* seguintes foram desenvolvidos sobre a implementação da plataforma OSGi OSCAR executada numa plataforma *Java* J2SE 5.0 [153]. Importa referir que nesta subsecção o termo “serviço” refere-se, a menos que o contrário seja referido explicitamente, a um serviço implementado por um dado *bundle* e não a um *networked service*.

4.2.3.1.1. Bundle Serial Service

Este *bundle* implementa um *Device Service* (descrito na especificação *Device Access*, secção 4.2.2.4) que representa a porta série do *gateway* de serviços. Assim, uma vez instalado e activo na plataforma de serviços, este *bundle* monitoriza a conexão/desconexão de dispositivos prontos a comunicar através da porta série do *gateway* e adiciona/remove ao registo de serviços da *framework* um serviço que representa essa mesma porta série. O registo deste serviço desencadeia um processo de refinamento do *Device Service* registado, coordenado pelo *Device Manager*, tal como descrito na secção 4.2.2.4. Porém, qualquer outro *bundle* presente na plataforma de serviços que deseje utilizar directamente a porta série poderá de imediato fazê-lo usando para tal o *Serial Service*. A figura seguinte representa o diagrama de classes e a estrutura do *bundle Serial Service*.

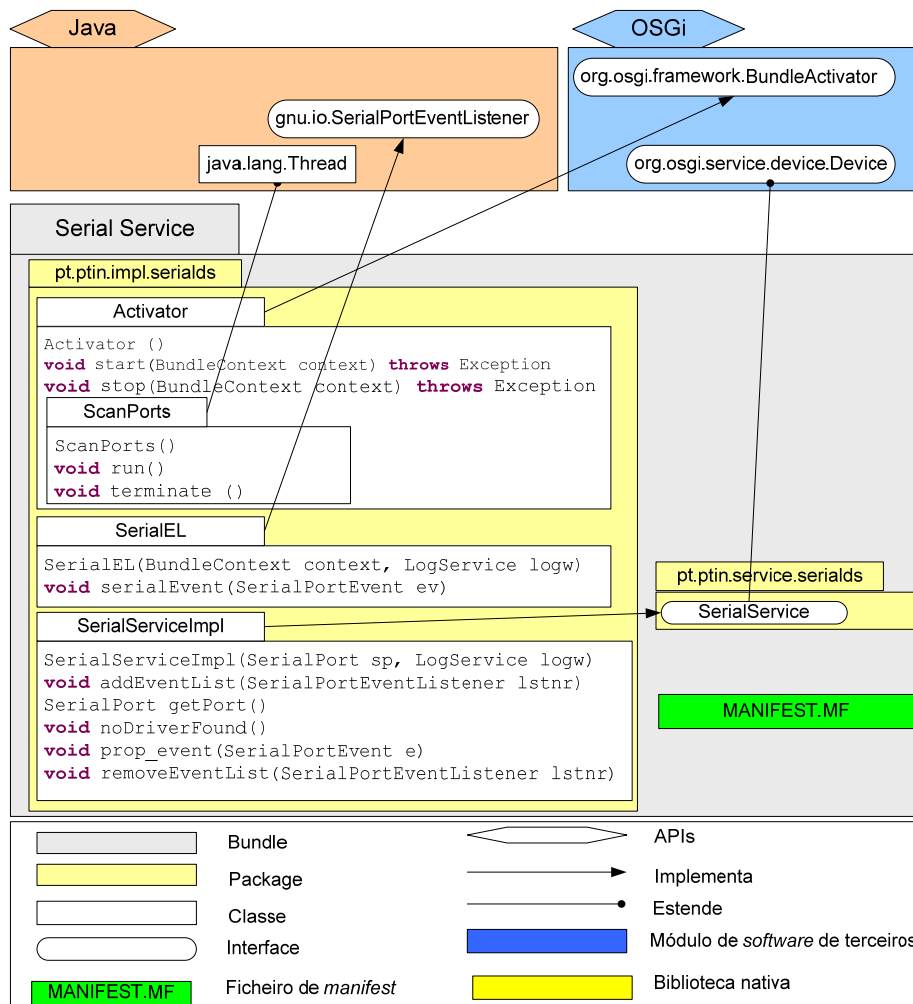


Figura 34: Diagrama de classes e estrutura do *bundle Serial Service*

Quando o *bundle* é activado (`pt.ptin.impl.serialids.Activator.start`) uma *thread*

(`pt.ptin.impl.serialds.Activator.ScanPorts`) é lançada. Esta *thread* procura, no *gateway* de serviços, todas as portas série disponíveis (não ocupadas por outras aplicações) e subscreve a recepção de eventos associados a essas mesmas portas. Quando um evento ocorre, este é analisado (`pt.ptin.impl.serialds.SerialEL.serialEvent(SerialPortEvent)`). Se da sua análise se concluir que um dispositivo foi conectado a uma das portas série do *gateway* de serviços, um *Device Service* (`pt.ptin.service.serialds.SerialService`) é registado na *framework* para representar a porta série associada ao evento (figura 33, passo 1). Assim, a partir deste momento outros *bundles* poderão aceder à porta série usando para isso o *Device Service* registado anteriormente. A interface que define o serviço *SerialService* é apresentada na figura seguinte:

```
public static final int MATCH_SERVICE_VENDOR=2;
public static final int MATCH_DEVICE_CATEGORY=1;

public interface SerialService extends Device {

    public SerialPort getPort();

    public void addEventList(SerialPortEventListener lstnr);

    public void removeEventList(SerialPortEventListener lstnr);

}
```

Figura 35: Interface *SerialService*

A interface `pt.ptin.service.serialds.SerialService` define métodos que permitem a outros *bundles* na plataforma obterem uma dada porta e subscreverem ou removerem a recepção de eventos que ocorram nessa mesma porta. Em adição a mesma interface define também um conjunto de valores inteiros que servirão como uma escala de licitação aos vários *Driver Services* que pretendam refinar este DS (ver secção 4.2.3.1.3).

Como se pode verificar na figura 34, o *bundle Serial Service* necessita de uma API *Java*, denominada de RXTX [154], não incluída por defeito no conjunto de APIs da plataforma *Java J2SE 5.0*. Uma vez que a presença desta API na plataforma *Java*, sobre a qual a plataforma *OSGi* é executada, é opcional um *bundle* que contém essa mesma API e a disponibiliza a outros *bundles* é necessário. Desta forma, foi criado um *bundle* denominado de *RXTXComm* que exporta a *package gnu.io* que por sua vez é importada pelo *bundle Serial Service*. O diagrama de classes e estrutura do *bundle RXTXComm* está representado na figura seguinte:

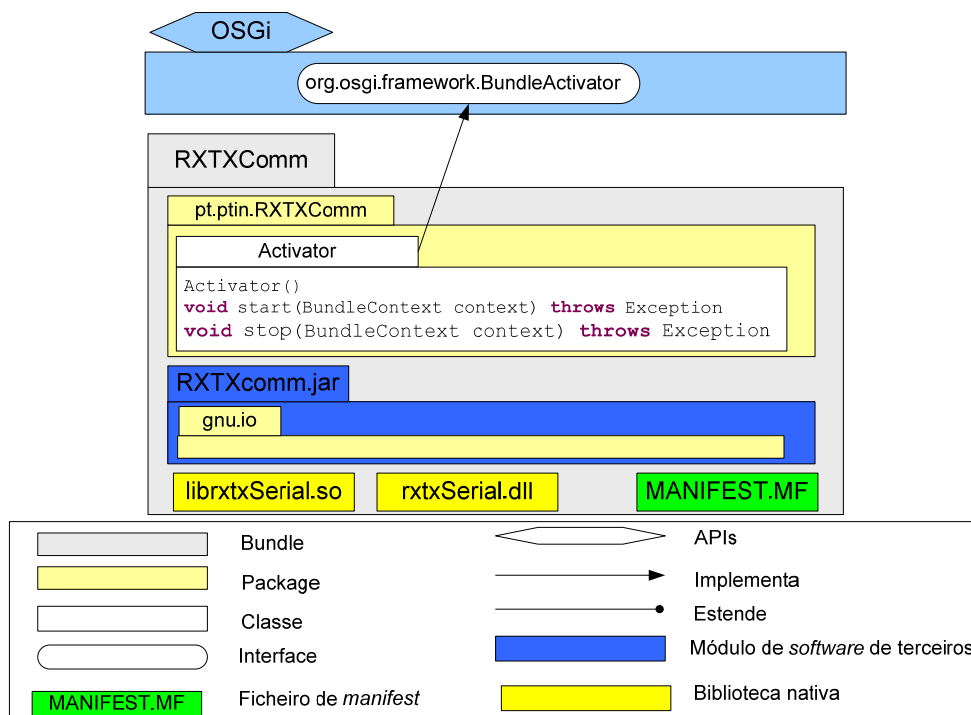


Figura 36: Diagrama de classes e estrutura do *bundle* RXTXComm

O *bundle* representado na figura 36 inclui o ficheiro *RXTXcomm.jar* que por sua vez inclui a *package* *gnu.io*. Esta, por sua vez, é exportada para a plataforma OSGi assim que o *bundle* entra no estado de *Resolved* (ver secção 3.2.3.2.1.3). A exportação da *package* é identificada através de uma declaração do tipo `Export-Package` no ficheiro *manifest* do *bundle*. Duas bibliotecas nativas, utilizadas na comunicação entre a API e o *hardware*, próprias ao SO (*Linux* e *Windows*) onde a plataforma *Java* é executada são também incluídas no *bundle*.

4.2.3.1.2. Bundle Driver Locator

Este *bundle*, como o próprio nome indica, implementa essencialmente um *Driver Locator Service* como descrito na secção 4.2.2.4. Durante o processo de inicialização do *bundle* (estado *starting*, figura 23) o conteúdo de um ficheiro alojado no OBR é utilizado para construir um dicionário (figura 37, classe `pt.ptin.impl.dlpti.DriverHash`) que contém informação acerca de todos os *Driver Services* que este DL conhece. Este ficheiro tem um formato próprio e através do qual cada um dos DrSs é descrito. Entre outros, este ficheiro inclui dados acerca da categoria de *Device Service* que o DrS pretende refinar, `DRIVER_ID` (como definido em `org.osgi.service.device.Constants.DRIVER_ID` [112]) e URL que localiza o *bundle* que implementa o DrS em questão. O diagrama de classes e estrutura do *bundle Driver Locator* é

apresentado na figura 37.

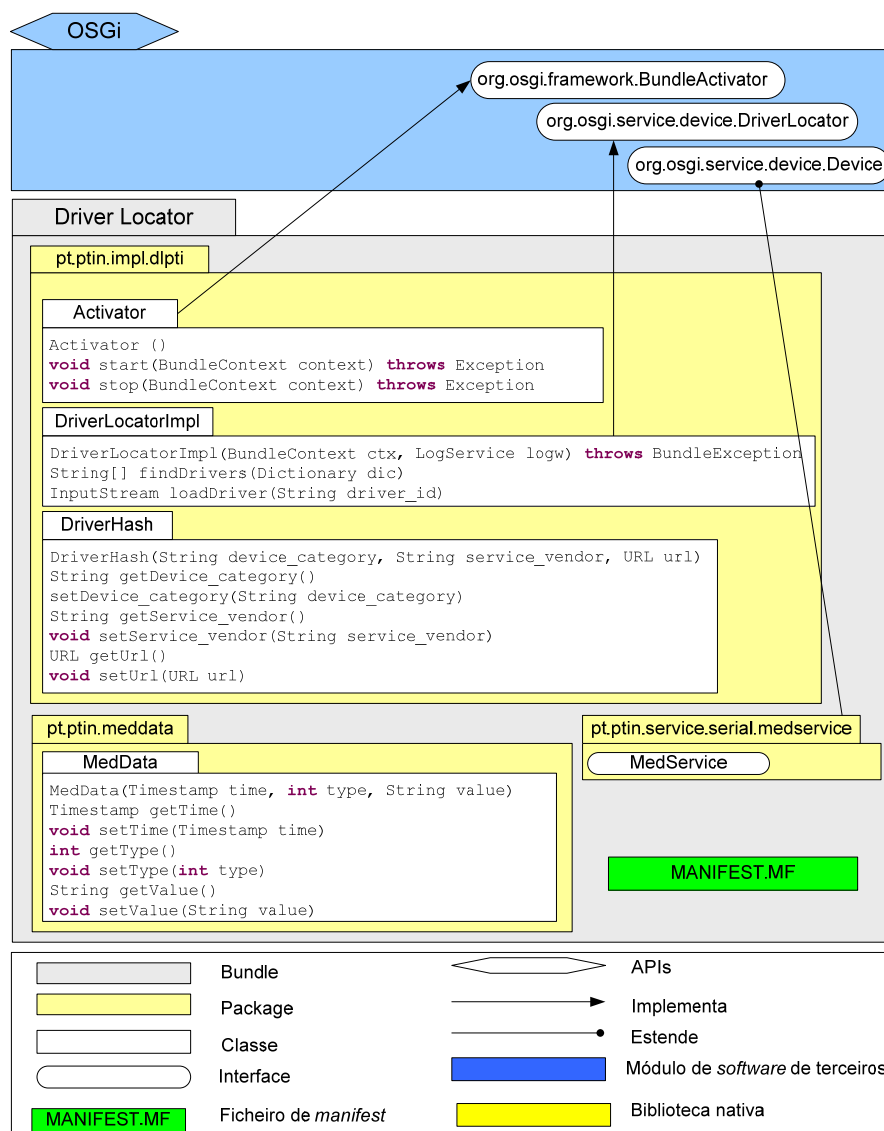


Figura 37: Diagrama de classes e estrutura do *bundle Driver Locator*

Para que um dado DLS seja reconhecido como tal pelo *Device Manager*, é necessário que este implemente a interface `org.osgi.service.device.DriverLocator` e registre na *framework* um serviço sobre essa mesma interface. Esta interface define dois métodos distintos que são invocados em diferentes instantes do processo de refinamento de um dado DS. Assim, aquando do registo de um novo DS na *framework*, o método `pt.ptin.impl.dlpti.DriverLocatorImpl.findDrivers(Dictionary)` é invocado pelo DM que obtém como retorno um conjunto de `DRIVER_IDS` (que representam univocamente um conjunto de DrSs) com capacidades para refinar o DS em questão (figura 33, passo 2). Por sua vez a invocação do método `pt.ptin.impl.dlpti.DriverLocatorImpl.loadDriver(String)`

por parte do DM, resulta na transferência e activação dos vários DrSs (que ainda não se encontrem neste estado) anteriormente apresentados como candidatos a refinar o DS (figura 33, passo 3).

Importa referir que a inclusão da *package* `pt.ptin.service.serial.medservice` (e exportação da mesma) no *bundle* *Driver Locator* possibilita a importação desta por parte de qualquer outro *bundle* presente na plataforma, sem que uma implementação do serviço definido pela interface `pt.ptin.service.serial.medservice.MedService` esteja de facto registada na *framework*. É assim possível, ao *bundle* que importa a *package*, transitar para o estado *active* independentemente da presença do serviço `MedService` e mais importante do que isso, adaptar o seu comportamento à presença/ausência do serviço. No cenário que se pretende implementar no âmbito da presente dissertação, tal comportamento é necessário ao *bundle* *Medical Data Submitter* (ver secção 4.2.3.1.4). Por sua vez a inclusão (e exportação) da *package* `pt.ptin.meddata` é justificada pelo facto de a interface `MedService` depender directamente da classe contida nesta mesma *package*.

4.2.3.1.3. Bundle MedDrS

O *bundle* *MedDrS* implementa um *Driver Service* de acordo com a especificação OSGi *Device Access*, DrS este identificado por um `DRIVER_ID` de valor `pt.ptin.serial.meddrs.0.0.5`. Sendo um DrS, este *bundle* implementa também um *Device Service* que, neste caso particular, pretende refinar o DS registado pelo *bundle* *Serial Service* para um DS denominado de *MedService* definido pela interface `pt.ptin.service.serial.medservice.MedService` (figura 39). Este DS representa então um dispositivo conectado à porta série do *gateway* de serviços da categoria *Serial Emulated Medical Equipment* (dispositivo emulado por uma aplicação descrita na secção 4.2.3.2.). Importa referir que o *bundle* *MedDrS* é automaticamente transferido e activado na *framework* pelo *Device Manager*, usando para isso informação obtida do *Driver Locator*. A figura 38 apresenta o diagrama de classes e estrutura deste *bundle*.

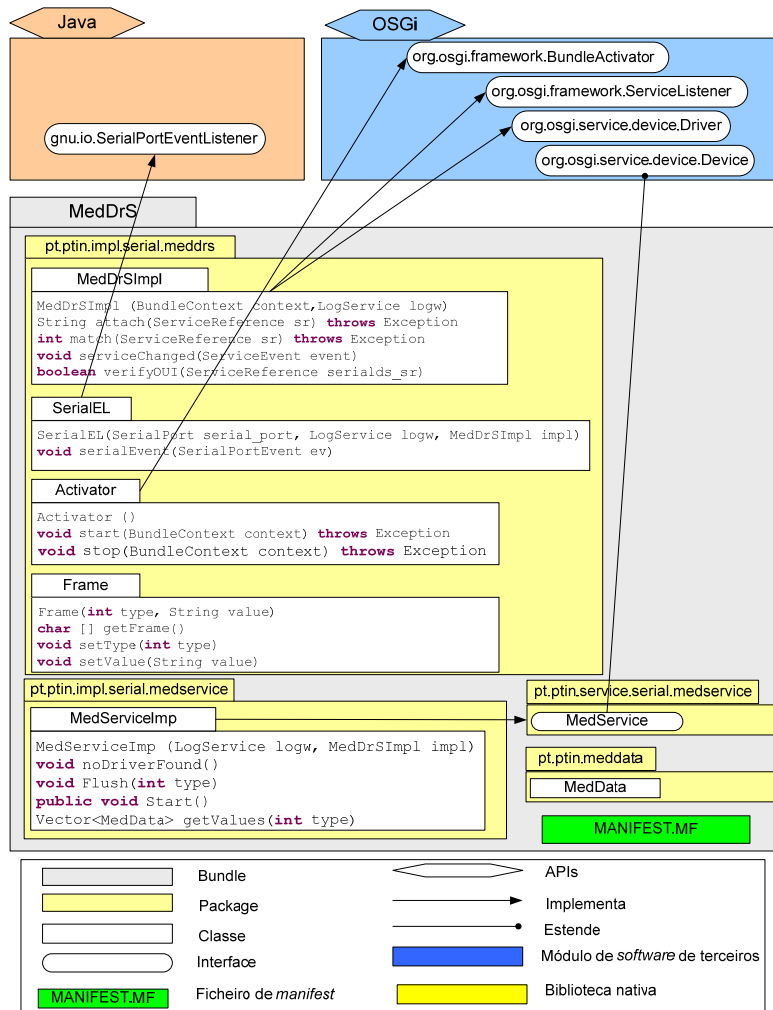


Figura 38: Diagrama de classes e estrutura do *bundle MedDrS*

Uma vez no estado activo, o DM invoca o método `pt.ptin.impl.serial.meddrs.MedDrSImpl.match(ServiceReference)` para que o DrS possa licitar um valor inteiro que reflecte a sua capacidade para refinar o DS em questão (figura 33, passo 4). No caso do *DS Serial Service* o valor licitado é baseado numa escala inteira (que pode variar entre 0 e 2) definida por esse mesmo serviço (figura 35). O valor 0 representa a incapacidade de refinar o DS, o valor 1 significa que a categoria do DS que se pretende refinar é semelhante à categoria do DrS e finalmente o valor 2 significa que para além da categoria, o DrS consegue também identificar o dispositivo conectado à porta série de uma forma unívoca. Ao contrário da comparação de categorias, esta identificação é efectuada de uma forma proprietária que se baseia no envio/recepção de uma mensagem (com o formato definido pela classe `pt.ptin.impl.serial.meddrs.Frame`, ver secção 4.2.3.2) através da porta série (o que implica um uso temporário do serviço *SerialService*) e respectiva análise do conteúdo da (possível) mensagem de resposta.

Caso o DrS implementado pelo *bundle MedService* licite o valor mais elevado, o DM (invocando o método `pt.ptin.impl.serial.meddrs.MedDrSImpl.attach(ServiceReference)`) garante o refinamento do DS *Serial Service* através do registo do DS *MedService* na *framework* (figura 33, passo 5). A figura 39 apresenta a interface que define o DS *MedService*.

```
public interface MedService extends Device{
    public void Start();
    public void Flush (int type);
    public Vector<MedData> getValues(int type);
}
```

Figura 39: Interface *MedService*

Este serviço fornece métodos que permitem iniciar o armazenamento dos dados recebidos pela porta série (`pt.ptin.service.serial.medservice.MedService.Start()`), descartar os dados recebidos (`pt.ptin.service.serial.medservice.MedService.Flush(int)`) e transmitir os dados recebidos até um certo instante a outro *bundle* que utilize este serviço (`pt.ptin.service.serial.medservice.MedService.getValues(int)`), sem que esse mesmo *bundle* tenha de saber todos os pormenores (por exemplo formato das tramas utilizadas) inerentes à comunicação com o dispositivo que recolhe as medições médicas.

A partir do momento que o DS *MedService* é registado este passa a utilizar o DS *Serial Service* até que seja removido do registo de serviços da *framework*. Como descrito na secção 4.2.3.1.1, o *Serial Service* implementa mecanismos que permitem a transmissão de eventos relativos à porta série para *bundles* que utilizem esse mesmo serviço. No caso do *bundle MedDrS* a análise deste tipo de eventos, fulcral no envio/recepção de mensagens através da porta série, é implementada na classe `pt.ptin.impl.serial.meddrs.SerialEL`. Outro tipo de eventos que o *bundle MedDrS* tem de escutar e processar são os eventos gerados pela *framework* relacionados com o registo e remoção de serviços. Este tipo de eventos permitem ao *bundle MedDrS* libertar atempadamente e de uma forma graciosa o serviço *SerialService*.

4.2.3.1.4. Bundle Medical Data Submitter

O *bundle Medical Data Submitter*, usando o serviço *MedService* implementado pelo *bundle MedDrS*, é responsável pela submissão das leituras efectuadas para a base de dados onde estas são armazenadas. Esta submissão é executada periodicamente (num intervalo de tempo configurável) e de uma forma segura utilizando o protocolo HTTPS [134]. Sobre a confidencialidade e integridade garantida pelo uso deste protocolo, um esquema de

autenticação baseado em nome de utilizador e palavra-chave é ainda implementado.

Uma vez transferido para a plataforma de serviços (através de uma indicação explícita enviada do gestor remoto para o agente de gestão da plataforma) e após as suas dependências terem sido resolvidas, o *bundle* transita para o estado activo. Após transitar para o estado activo o *bundle* ajusta o seu comportamento à presença ou ausência do serviço *MedService*, do qual depende para desempenhar a sua função. A figura 40 apresenta o diagrama de classes e estrutura do *bundle*.

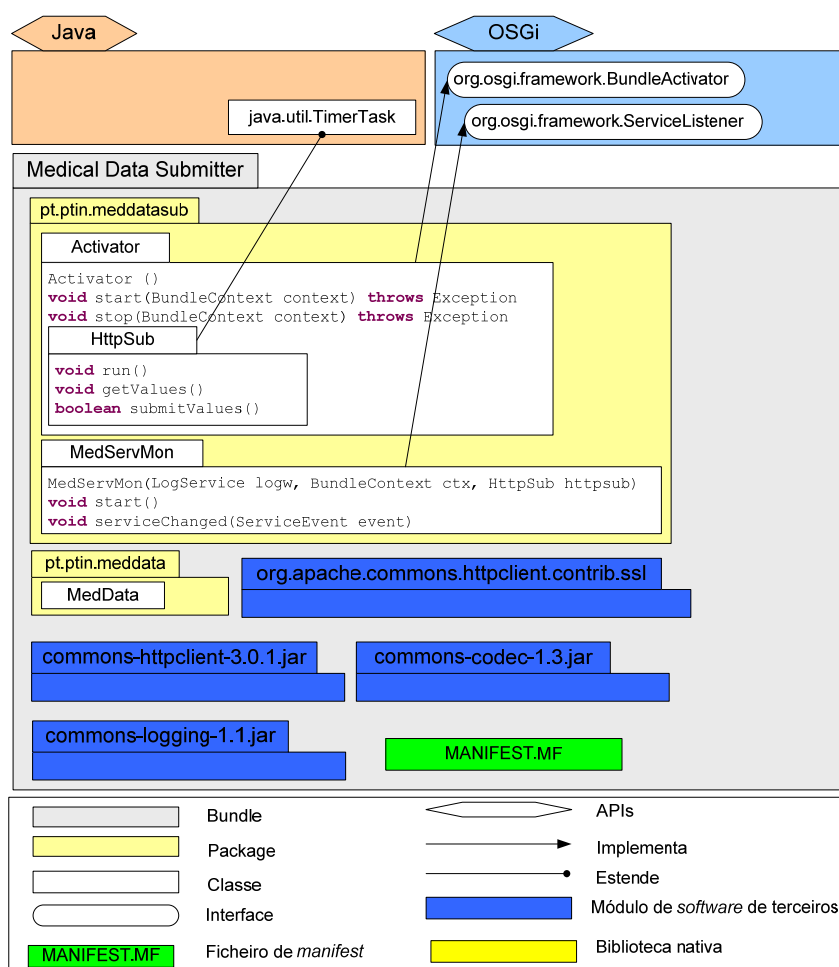


Figura 40: Diagrama de classes e estrutura do *bundle Medical Data Submitter*

Assim, se o *bundle* é activado na plataforma e o serviço *MedService* não se encontra registado na *framework*, o *bundle* fica num estado passivo em que apenas monitoriza (classe `pt.ptin.meddatasub.MedServMon`) os vários eventos de registo/remoção de serviços na plataforma. Se um desses eventos reportar que o serviço `pt.ptin.service.serial.medservice.MedService` acabou de se registar, o *bundle* automaticamente obtém do registo de serviços o serviço *MedService* e invoca o método

`pt.ptin.service.serial.medservice.MedService.Start()` deste serviço para que as leituras recebidas pela porta série comecem a ser armazenadas. A partir deste momento o *bundle* *Medical Data Submitter* inicia o processo periódico de submissão de dados (classe `pt.ptin.meddatasub.Activator.HttpSub`). Neste processo o *bundle* estabelece uma sessão HTTPS com o servidor *Web*, autentica-se com o respectivo nome de utilizador e palavra passe e submete todas as leituras efectuadas desde o instante de tempo em que a última submissão foi efectuada. A submissão periódica de dados cessa quando o próprio *bundle* for removido da plataforma (através de uma indicação explícita enviada do gestor remoto para o agente de gestão da plataforma) ou quando o serviço *MedService* for removido (desconexão do cabo que conecta o *gateway* de serviços ao equipamento médico). Neste último caso o *bundle* recebe uma notificação (evento gerado pela *framework*) que reporta o facto do serviço *MedService* estar prestes a ser removido da plataforma e retorna ao estado passivo em que apenas monitoriza os eventos gerados pela *framework*. Importa referir que toda a comunicação entre o *bundle* e o servidor *Web* é efectuada através de métodos disponibilizados por bibliotecas externas à API Java e à API OSGi (*Jakarta Commons http Client* [155]) pelo que o *bundle* *Medical Data Submitter* tem de incluir no seu ficheiro *jar* estas bibliotecas (na figura 40 elementos a azul).

A figura 41 apresenta as dependências entre *bundles* ao nível de *packages* e serviços OSGi.

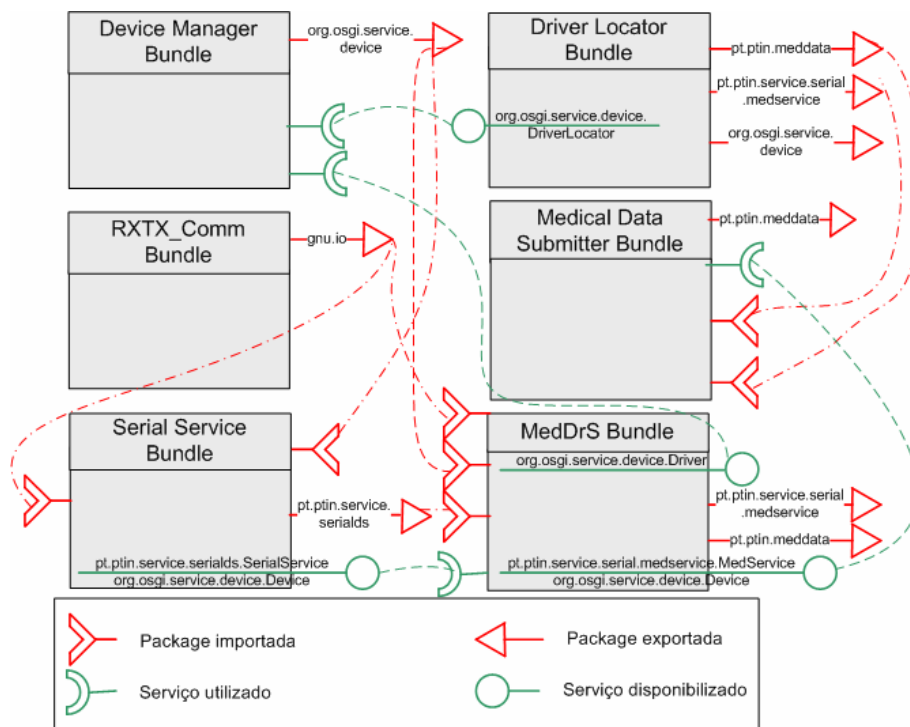


Figura 41: Dependências entre os *bundles* intervenientes no serviço de monitorização médica remota.

Existem vários *bundles* que contêm as mesmas *packages* (por exemplo os *bundles MedDrS* e *Driver Locator*), no entanto apenas uma é efectivamente exportada para a plataforma e depois importada pelos restantes *bundles*. No que diz respeito à partilha de serviços, verifica-se que existe um encadeamento na partilha de serviços OSGi entre os *bundles*. O *bundle Serial Service* disponibiliza o serviço *SerialService* que é utilizado pelo *bundle MedDrS*. Este, por sua vez, disponibiliza o serviço *MedService* ao *bundle Medical Data Submitter*. É esta colaboração entre todos os *bundles* que permite a correcta operação do serviço de monitorização médica remota.

4.2.3.2. Emulador de equipamento médico

O emulador de equipamento médico não é mais do que uma aplicação *Java* executada no PC conectado ao *gateway* de serviços através da porta série que emula um dispositivo de categoria *Serial Emulated Medical Equipment*. Esta aplicação, fazendo uso da biblioteca RXTX [154], envia pela porta série valores (introduzidos por um utilizador) que representam vários tipos de leituras médicas. A interface gráfica desta aplicação é apresentada na figura 42.

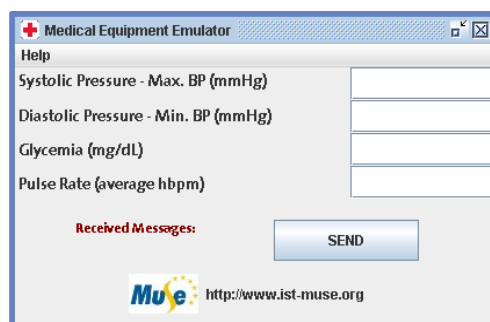


Figura 42: GUI do emulador de equipamento médico

Como representado na figura acima é possível introduzir medidas referentes à tensão arterial (sístole e diástole), à glicemia e ao ritmo cardíaco, sendo que o utilizador pode introduzir apenas um ou vários tipos de medida simultaneamente. Os valores introduzidos pelo utilizador, depois de pressionado o botão *send*, são formatados e enviados pela porta série numa trama cujo formato é apresentado na figura 43.

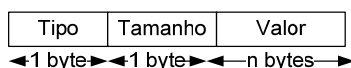


Figura 43: Formato da trama utilizada para envio de dados através da porta série

Desta forma, o campo Tipo codifica o tipo de dados que são transportados na presente trama (cada um dos quatro tipos de dados médicos mais um tipo reservado para mensagens), o

campo Tamanho transporta o tamanho total da trama em *bytes* e finalmente o campo Valor transporta o valor da medição médica ou o conteúdo da mensagem. Importa referir que o tipo de trama reservado à transmissão de mensagens é essencialmente utilizado no processo em que o *bundle MedDrS* pretende determinar se o dispositivo conectado à porta série é de facto um *Serial Emulated Medical Equipment* (secção 4.2.3.1.3).

4.2.3.3. Gestor remoto da plataforma de serviços

Como descrito na secção 3.2.3.3, um gestor remoto providencia ao operador (do *gateway* de serviços) mecanismos que lhe permitem efectuar operações de gestão sobre a plataforma de serviços. Das várias vertentes de gestão consideradas na especificação OSGi, o gestor remoto implementado no âmbito desta dissertação aborda apenas a vertente relacionada com a gestão do ciclo de vida de *bundles*. Importa contudo referir que ao contrário dos vários gestores remotos disponíveis, o gestor remoto implementado não obriga o operador a saber qual o conjunto de *bundles* necessários para disponibilizar um determinado serviço. Assim, a interface apenas apresenta os serviços disponíveis para subscrição, sendo que cada um destes serviços poderá implicar a activação de um ou mais *bundles* na plataforma de serviços. As dependências entre os vários *bundles* que formam um serviço são depois geridas automaticamente com a ajuda do OBR. Este facto permite que este gestor remoto seja não só usado pelo operador da plataforma de serviços, mas também pelo próprio utilizador que pode subscrever ou remover serviços como desejar sem uma intervenção directa do operador.

A figura 44 apresenta uma imagem da interface do gestor remoto implementado.

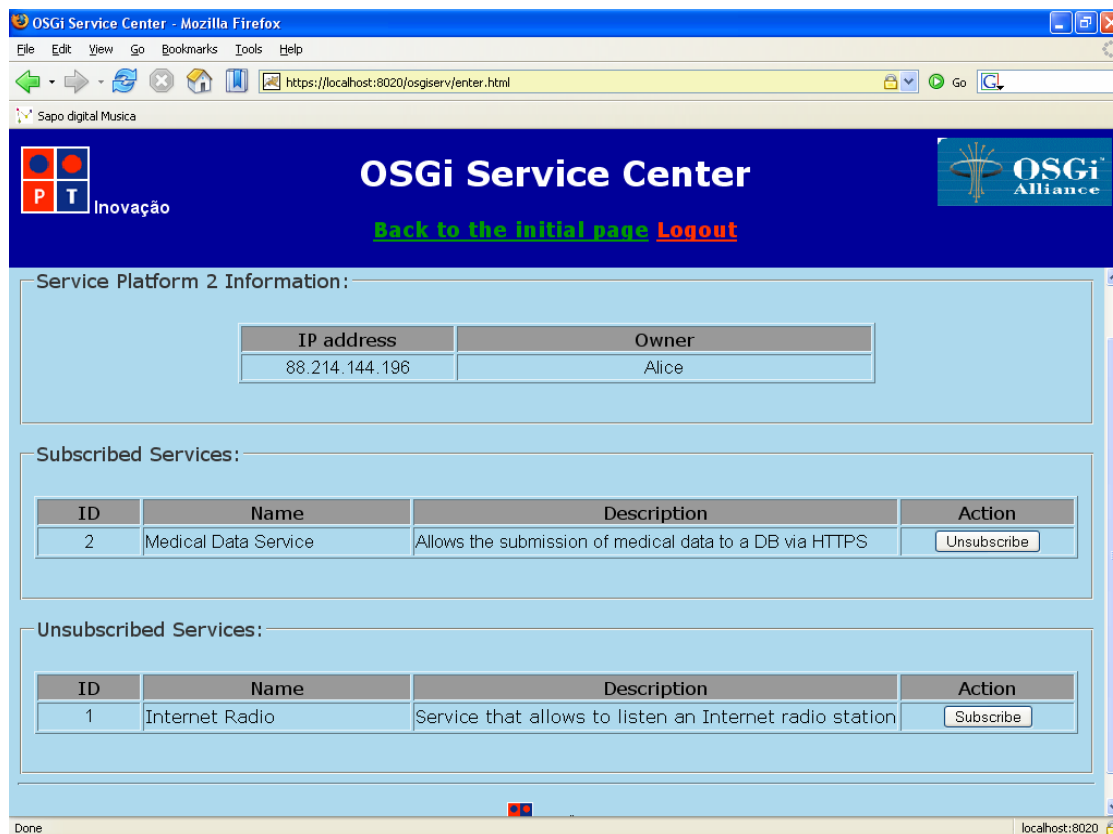


Figura 44: Interface *web* do gestor remoto

No caso particular da figura anterior dois serviços estão disponíveis para subscrição: o serviço de monitorização médica remota, desenvolvido no âmbito desta dissertação e um serviço que permite a audição de uma estação de rádio de Internet. Enquanto o primeiro serviço implica a activação de vários *bundles* na plataforma de serviços, o segundo necessita apenas da activação de um só *bundle*.

Funcionalmente, o gestor remoto pode ser dividido em dois módulos fundamentais: enquanto o primeiro lida com todos os pormenores relacionados com a comunicação com a plataforma OSGi, implementando para isso um gestor JMX, o segundo lida com a interacção entre o operador (do *gateway* de serviços) e o gestor JMX por meio de um conjunto de *Servlets Java*.

A figura 45 apresenta o diagrama de classes e estrutura do gestor remoto implementado.

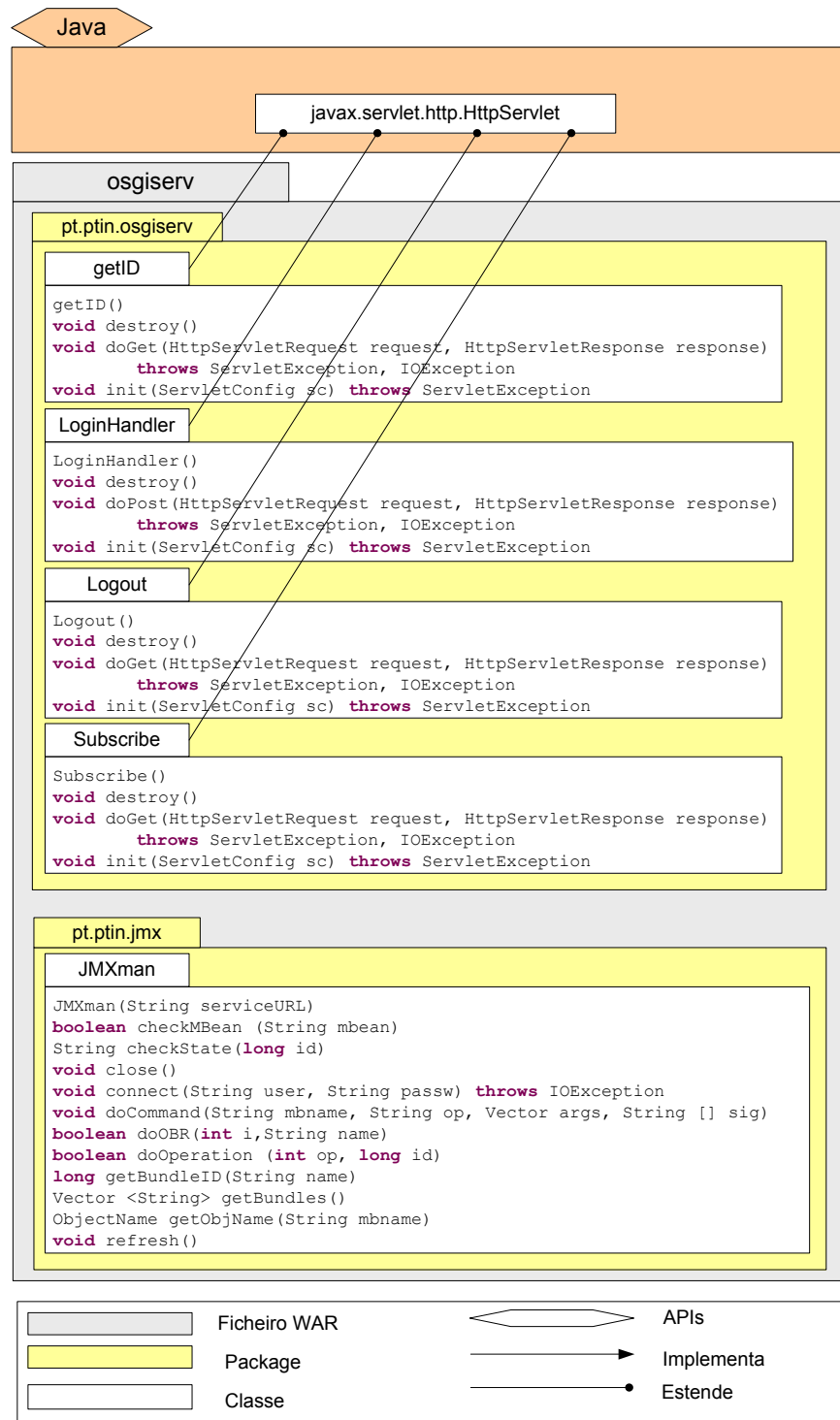


Figura 45: Diagrama de classes do gestor remoto da plataforma de serviços

O conjunto de *Servlets* para além de implementar a interface *Web*, que permite ao operador interagir com o gestor JMX (e por consequência com a plataforma OSGi), implementa também mecanismos que permitem a autenticação segura do utilizador (classes `pt.ptin.osgiserv.LoginHandler` e `pt.ptin.osgiserv.Logout`), a activação ou remoção de

serviços nas várias plataformas (classe `pt.ptin.osgiserv.Subscribe`) e a obtenção de dados referentes às várias plataformas de serviços (classe `pt.ptin.osgiserv.getID`). Importa referir que alguns destes dados (endereço IP, proprietário da plataforma, serviços disponíveis para subscrição) estão armazenados numa base de dados `mySQL` que serve de suporte ao conjunto de *Servlets*, enquanto outros (por exemplo serviços actualmente subscritos) são obtidos comunicando directamente com a plataforma de serviços, através do gestor `JMX`.

4.2.3.4. Servlet de submissão de dados médicos

Este *Servlet Java* comunica directamente com o *bundle Medical Data Submitter* instalado na plataforma de serviços através de uma ligação `HTTPS`. Este módulo é responsável por receber os dados enviados pelo *bundle* através da ligação `HTTPS` e caso as credenciais (nome de utilizador e palavra passe) apresentadas por este sejam válidas, efectuar submissão dos dados recebidos na base de dados. Para além da base de dados médica uma outra base de dados, que contém os tipos de medidas que cada utilizador pode submeter (perfil do utilizador), é utilizada por este *Servlet*. Isto significa que se um *bundle* tentar submeter medidas que não fazem parte do perfil associado às credenciais apresentadas, estas são ignoradas.

4.2.4. Validação dos módulos implementados

Esta secção apresenta uma breve descrição de um conjunto de resultados que permitiram validar experimentalmente os módulos de *software* desenvolvidos. Por forma a obterem-se estes resultados, o conjunto de procedimentos e interacções entre as diferentes entidades intervenientes no serviço, descrito na secção 4.2.1, foi adoptado.

A figura 46 apresenta o conjunto de *bundles* inicialmente instalados (e no estado activo) na plataforma de serviços, necessários ao cenário descrito na secção 4.2.1.

ID	State	Level	Name
[0]	[Active] [0]	System Bundle (1.0.5)
[1]	[Active] [1]	Shell Service (1.0.2)
[2]	[Active] [1]	Shell TUI (1.0.0)
[3]	[Active] [1]	Bundle Repository (1.1.2)
[4]	[Active] [1]	Log Service (1.0.0)
[5]	[Active] [1]	Log Reader PTi (0.0.3)
[6]	[Active] [1]	JMX-MX4J Agent Service (2.0.2)
[7]	[Active] [1]	JMX-MX4J http connector (2.0.3)
[8]	[Active] [1]	JMX rmi registry (2.0.4)
[10]	[Active] [1]	JMX rmi connector auth (2.0.5)
[11]	[Active] [1]	OSGi MBean (2.0.5)
[12]	[Active] [1]	Service Introspector (1.0.0)
[13]	[Active] [1]	MBean factory (2.0.0)
[14]	[Active] [1]	Service notifier (2.0.0)
[15]	[Active] [1]	Shell MBean (1.0.0)
[16]	[Active] [1]	Device Manager (1.0.0)
[17]	[Active] [1]	Driver Locator (0.0.4)

Figura 46: *Bundles* activos na plataforma de serviços por defeito

Os *bundles* de zero a três são instalados por defeito com a implementação OSCAR (secção 4.2.2.2), os *bundles* quatro e cinco permitem, respectivamente, que os restantes *bundles* escrevam mensagens de *Log* e que as mesmas sejam mostradas no ecrã, os *bundles* seis a quinze dotam a plataforma com a tecnologia JMX (secção 4.2.2.3) e finalmente os restantes *bundles* pertencem à especificação *Device Access* (secção 4.2.2.4). Admitindo um cenário comercial, este conjunto de *bundles* estaria instalado por defeito com o *gateway* de serviços que o cliente adquire.

O primeiro procedimento passa pela instalação (a partir da interface do gestor remoto) do serviço de monitorização médica remota. A instalação deste serviço resulta na sequência de eventos ilustrada na figura 47.

```

obr start "Medical Data Submitter"
Installing dependency: RXTXComm
Installing dependency: Serial Service
Installing: Medical Data Submitter
@: 2007-01-04 19:48:52.262 INFO entry by bundle with id #18
    Message:BundleEvent.STARTED
@: 2007-01-04 19:48:52.267 INFO entry by bundle with id #19
    Message:BundleEvent.STARTED
@: 2007-01-04 19:48:52.457 DEBUG entry by bundle with id #20
    Message:pt.ptin.service.serial.medservice.MedService not present at start time
@: 2007-01-04 19:48:52.461 INFO entry by bundle with id #20
    Message:BundleEvent.STARTED
@: 2007-01-04 19:48:52.596 DEBUG entry by bundle with id #19
    Message: Port: /dev/ttyS0 opened with success!
    
```

Figura 47: Eventos na sequência da instalação do serviço de monitorização médica remota

Ao instalar o *bundle Medical Data Submitter*, o mecanismo de dependências do OBR verifica que este depende de outro *bundle*, o *bundle Serial Service*, e que por sua vez este depende de um terceiro *bundle*. Assim, o processo de instalação do serviço resulta na instalação e activação de

três novos *bundles* na plataforma de serviços sendo que, no processo de activação, o *bundle Serial Service* apodera-se da única porta série disponível no *gateway* de serviços e o *bundle Medical Data Submitter* verifica se o serviço *MedService* está desde já disponível.

No passo seguinte do procedimento, a aplicação *Java* que emula o equipamento médico é executada. Esta acção causa o registo de um *Serial Service* como mostrado na figura 48.

```
@: 2007-01-04 20:00:15.538 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property: DEVICE_CATEGORY = Serial Port
@: 2007-01-04 20:00:15.538 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property: DEVICE_DESCRIPTION = Common Serial
    Port that supports RS-232 protocol
@: 2007-01-04 20:00:15.538 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property: service.description = Device Service
    that represents serial port /dev/ttyS0
@: 2007-01-04 20:00:15.538 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property: service.id = 31
@: 2007-01-04 20:00:15.538 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property: service.vendor = PTi
@: 2007-01-04 20:00:15.539 INFO entry by bundle with id #19
    Message:ServiceEvent.REGISTERED
    Service Reference:[pt.ptin.service.serialids.SerialService,
    org.osgi.service.device.Device]
@: 2007-01-04 20:00:15.539 DEBUG entry by bundle with id #19
    Message:Port: /dev/ttyS0 got plugged, serial device service registered!
```

Figura 48: Registo do *SerialService* após a execução do emulador de equipamento médico

O registo de um novo *Device Service* é detectado pelo *Device Manager* que inicia o processo de refinamento. A figura 49 apresenta os vários eventos gerados ao longo deste processo.

```

@: 2007-01-04 20:00:15.539 DEBUG entry by bundle with id #16
    Message:constructDriverDictionary
@: 2007-01-04 20:00:15.54 DEBUG entry by bundle with id #17
    Message:pt.ptin.impl.dlpti.DriverLocatorImpl knows pt.ptin.serial.meddrs.0.0.5
    Driver to refine Serial Port
@: 2007-01-04 20:00:15.565 INFO entry by bundle with id #16
    Message:***** New Driver Reference
    insa.device.services.devicemanager.DeviceManager@dc57db :
    [pt.ptin.service.serialids.SerialService,
    org.osgi.service.device.Device]:pt.ptin.serial.meddrs.0.0.5 :
@: 2007-01-04 20:00:15.581 INFO entry by bundle with id #21
    Message:BundleEvent.INSTALLED
@: 2007-01-04 20:00:15.594 INFO entry by bundle with id #21
    Message:BundleEvent.STARTED
@: 2007-01-04 20:00:15.594 INFO entry by bundle with id #21
    Message:ServiceEvent.REGISTERED
    Service Reference:[org.osgi.service.device.Driver]
@: 2007-01-04 20:00:15.596 INFO entry by bundle with id #16
    Message:***** New Driver Added
    insa.device.services.devicemanager.DriverReference@6f50a8[org.osgi.service.device
    Driver] : pt.ptin.impl.serial.meddrs.MedDrSImpl@157aa53
@: 2007-01-04 20:00:15.654 DEBUG entry by bundle with id #21
    Message:!!!!Driver pt.ptin.impl.serial.meddrs.MedDrSImpl matches ID!!!!!!
@: 2007-01-04 20:00:15.654 INFO entry by bundle with id #16
    Message:The best matching driver is
    insa.device.services.devicemanager.DriverReference@6f50a8
@: 2007-01-04 20:00:15.655 DEBUG entry by bundle with id #16
    Message:The Driver is attached to >>>>> [pt.ptin.service.serialids.SerialService,
    org.osgi.service.device.Device]
@: 2007-01-04 20:00:15.662 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property 0 : DEVICE_CATEGORY = Serial Emulated
    Medical Equipment
@: 2007-01-04 20:00:15.663 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property: DEVICE_DESCRIPTION = Emulated Medical
    Equipment
@: 2007-01-04 20:00:15.663 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property: DEVICE_MAKE = PTi
@: 2007-01-04 20:00:15.663 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property: service.description = Device Service
    that refines a Serial DS to a MUSE emulated Medical Equipment
@: 2007-01-04 20:00:15.663 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property: service.id = 33
@: 2007-01-04 20:00:15.663 DEBUG entry by bundle with id #16
    Message:getRegisteredProperties : Property: service.vendor = PTi
@: 2007-01-04 20:00:15.663 INFO entry by bundle with id #16
    Message:DeviceManager.deviceRegistered
@: 2007-01-04 20:00:15.663 DEBUG entry by bundle with id #16
    Message:constructDriverDictionary
@: 2007-01-04 20:00:15.664 INFO entry by bundle with id #16
    Message:***** New Driver Reference
    insa.device.services.devicemanager.DeviceManager@dc57db :
    [org.osgi.service.device.Device,
    pt.ptin.service.serial.medservice.MedService]:pt.ptin.serial.meddrs.0.0.5 :
    org.ungoverned.oscar.BundleContextImpl@c24c0 : [org.osgi.service.device.Driver]
@: 2007-01-04 20:00:15.664 DEBUG entry by bundle with id #16
    Message:constructDriverDictionary : drivers already installed found
    :pt.ptin.serial.meddrs.0.0.50
@: 2007-01-04 20:00:15.664 DEBUG entry by bundle with id #21
    Message:pt.ptin.impl.serial.meddrs.MedDrSImpl Driver bids 0
@: 2007-01-04 20:00:15.664 INFO entry by bundle with id #16
    Message:The best matching driver is null
@: 2007-01-04 20:00:15.664 DEBUG entry by bundle with id #21
    Message:pt.ptin.service.serial.medservice.MedService won't be further refined
    
```

Figura 49: Processo de refinamento do *Device Service SerialService*

O processo de refinamento inicia-se com o *Device Manager* a interrogar o *Driver Locator* acerca dos *Driver Services* que este conhece e que podem refinar o *Device Service SerialService*. Uma vez que o DL conhece um DrS nestas condições, o *bundle* que implementa esse DrS é descarregado e activado na plataforma (na figura 49 eventos a verde, grupo A). Uma vez

activo na plataforma, o *bundle* regista um DrS que é detectado pelo DM. Este novo DrS é de seguida interrogado pelo DM acerca da sua capacidade para refinar o *SerialService*. Uma vez que DrS consegue identificar univocamente o dispositivo conectado à porta série do *gateway* de serviços, este é eleito para registar um novo DS que refina o *SerialService* (na figura 49 eventos a azul, grupo B). Uma vez mais, o registo de um novo DS inicia um processamento de tentativa de refinamento por parte do DM. Todavia, uma vez que o DL não conhece mais nenhum DrS e que o DrS já activo na plataforma também não é capaz de refinar o DS em questão, o processo falha e o DS *MedService* não é refinado em maior detalhe (na figura 49 eventos a preto, grupo C). A figura 50 apresenta os *bundles* activos na *framework* (para além dos inicialmente instalados), concluído o processo de refinamento.

```
[ 18] [Active   ] [ 1] RXTXComm (1.0.0)
[ 19] [Active   ] [ 1] Serial Service (0.0.6)
[ 20] [Active   ] [ 1] Medical Data Submitter (0.0.3)
[ 21] [Active   ] [ 1] MedDrS (0.0.5)
```

Figura 50: *Bundles* activos na *framework* concluído o processo de refinamento do DS *SerialService*

Através de um evento lançado pela *framework* o *bundle Medical Data Submitter* é alertado para o registo do serviço *MedService*. Este serviço passa a ser então utilizado pelo *bundle* e a partir desse momento os dados recebidos pela porta série começam a ser armazenados pelo *bundle MedDrS*, para mais tarde serem submetidos para a base de dados médicos, como ilustrado na figura seguinte.

<pre>@: 2007-01-04 20:07:36.514 DEBUG entry by bundle with id #21 Message:TA_MAX= 110 values stored @: 2007-01-04 20:07:36.514 DEBUG entry by bundle with id #21 Message:TA_MIN= 85 values stored @: 2007-01-04 20:07:36.524 DEBUG entry by bundle with id #21 Message:GLYC= 75 values stored @: 2007-01-04 20:07:36.524 DEBUG entry by bundle with id #21 Message:RC= 80 values stored @: 2007-01-04 20:07:46.114 DEBUG entry by bundle with id #21 Message:GLYC= 77 values stored @: 2007-01-04 20:07:46.124 DEBUG entry by bundle with id #21 Message:RC= 85 values stored @: 2007-01-04 20:07:48.564 DEBUG entry by bundle with id #21 Message:GLYC= 80 values stored Jan 4, 2007 8:08:05 PM org.apache.commons.httpclient.auth.AuthChallengeProcessor selectAuthScheme @: 2007-01-04 20:08:05.394 DEBUG entry by bundle with id #20 Message:Set of data submitted to server with success</pre>	<table border="1"> <thead> <tr> <th>time</th> <th>value</th> <th>type</th> </tr> </thead> <tbody> <tr><td>2006-12-21 13:04:00</td><td>100</td><td>1</td></tr> <tr><td>2006-12-21 13:04:00</td><td>88</td><td>2</td></tr> <tr><td>2006-12-21 13:04:00</td><td>80</td><td>3</td></tr> <tr><td>2006-12-21 13:04:00</td><td>56</td><td>4</td></tr> <tr><td>2006-12-21 18:09:42</td><td>79</td><td>3</td></tr> <tr><td>2006-12-27 17:28:06</td><td>88</td><td>4</td></tr> <tr><td>2006-12-27 17:59:27</td><td>76</td><td>4</td></tr> <tr><td>2007-01-04 20:07:36</td><td>110</td><td>1</td></tr> <tr><td>2007-01-04 20:07:36</td><td>85</td><td>2</td></tr> <tr><td>2007-01-04 20:07:36</td><td>80</td><td>3</td></tr> <tr><td>2007-01-04 20:07:36</td><td>75</td><td>4</td></tr> <tr><td>2007-01-04 20:07:46</td><td>85</td><td>3</td></tr> <tr><td>2007-01-04 20:07:46</td><td>77</td><td>4</td></tr> <tr><td>2007-01-04 20:07:48</td><td>80</td><td>4</td></tr> </tbody> </table>	time	value	type	2006-12-21 13:04:00	100	1	2006-12-21 13:04:00	88	2	2006-12-21 13:04:00	80	3	2006-12-21 13:04:00	56	4	2006-12-21 18:09:42	79	3	2006-12-27 17:28:06	88	4	2006-12-27 17:59:27	76	4	2007-01-04 20:07:36	110	1	2007-01-04 20:07:36	85	2	2007-01-04 20:07:36	80	3	2007-01-04 20:07:36	75	4	2007-01-04 20:07:46	85	3	2007-01-04 20:07:46	77	4	2007-01-04 20:07:48	80	4
time	value	type																																												
2006-12-21 13:04:00	100	1																																												
2006-12-21 13:04:00	88	2																																												
2006-12-21 13:04:00	80	3																																												
2006-12-21 13:04:00	56	4																																												
2006-12-21 18:09:42	79	3																																												
2006-12-27 17:28:06	88	4																																												
2006-12-27 17:59:27	76	4																																												
2007-01-04 20:07:36	110	1																																												
2007-01-04 20:07:36	85	2																																												
2007-01-04 20:07:36	80	3																																												
2007-01-04 20:07:36	75	4																																												
2007-01-04 20:07:46	85	3																																												
2007-01-04 20:07:46	77	4																																												
2007-01-04 20:07:48	80	4																																												

Figura 51: Armazenamento de valores médicos recebidos, submissão e visualização dos mesmos na base de dados

Quando o *bundle Medical Data Submitter* pretende submeter os dados, obtém os valores

armazenados até esse instante pelo *bundle MedDrS* através do serviço *MedService*, autentica-se perante o *Servlet* de submissão de dados e executa o processo de submissão de dados.

O serviço continua funcional até que a aplicação que emula o equipamento médico seja terminada (o equivalente a desligar o cabo que interliga o PC onde a aplicação é executada ao *gateway* de serviços) ou então até que o *bundle Medical Data Submitter*, que implementa toda a “inteligência” do serviço, seja removido da plataforma. Na primeira situação, assim que o emulador é terminado, o DS *SerialService* é removido do registo de serviços (secção 4.2.3.1.1). Através de um evento gerado pela *framework*, o DM é informado desta alteração e ordena a remoção do DS *MedService* do registo de serviços e do próprio *bundle MedDrS* da plataforma de OSGi. No final desta sequência de acontecimentos os *bundles* que permanecem activos na plataforma estão representados na figura 52.

[17]	[Active]	[1]	Driver Locator (0.0.4)
[18]	[Active]	[1]	RXTXComm (1.0.0)
[19]	[Active]	[1]	Serial Service (0.0.6)
[20]	[Active]	[1]	Medical Data Submitter (0.0.3)

Figura 52: *Bundles* activos na plataforma de serviços após término do emulador de equipamento médico

O *bundle Medical Data Submitter*, que necessita do serviço *MedService* para operar correctamente, continua no estado activo a monitorizar o registo/remoção de serviços na *framework*. Este *bundle* apenas será removido quando uma operação de gestão remota com esse objectivo for efectuada. Por sua vez, os *bundles* dezoito e dezanove permanecerão na plataforma, mesmo após a remoção do *bundle Medical Data Submitter*. Desta forma, a instalação de um outro serviço (ou a reinstalação do serviço de monitorização médica remota) que necessite de utilizar a porta série não requer a reinstalação destes dois *bundles*. Outra opção perfeitamente válida, seria por exemplo estes dois *bundles* viram já instalados por defeito na plataforma de serviços, à semelhança do que acontece com os *bundles Device Manager* e *Driver Locator*.

4.3. Outros serviços e a plataforma OSGi

O serviço implementado no âmbito desta dissertação pode ser considerado um serviço relativamente simples, porém, o objectivo era mostrar como a arquitectura OSGi permite construir serviços complexos a partir de módulos de *software* simples (*bundles*). Com o objectivo de se ilustrar um pouco mais acerca do universo de serviços possíveis com a plataforma OSGi,

descreve-se de seguida brevemente um serviço na área da domótica e vigilância denominado de BarkIDS [156]. Três entidades distintas intervêm no seu funcionamento: o *gateway* residencial (onde é executada uma plataforma OSGi), um centro de controlo (para onde são enviados alarmes) e um terminal de utilizador (que recebe alarmes e efectua vídeo chamadas). A figura seguinte apresenta a relação entre as referidas entidades.

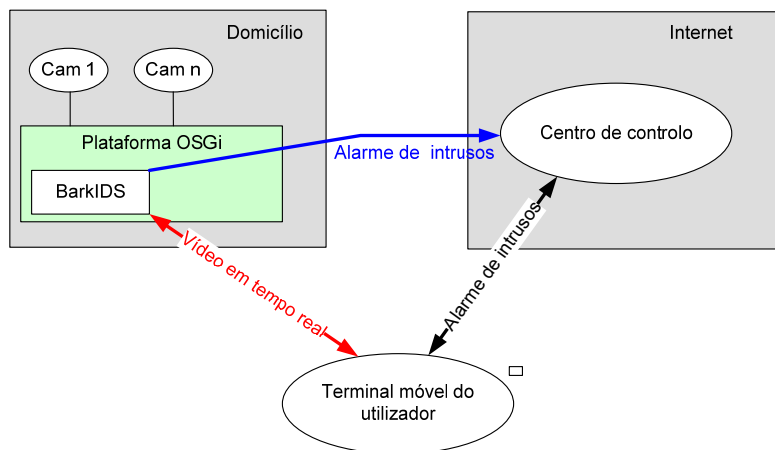


Figura 53: Serviço BarkIDS [157]

A aplicação que implementa o serviço BarkIDS foi totalmente implementada sobre a plataforma OSGi e inclui funções como processamento de imagem proveniente de câmaras USB ou IP, algoritmos de detecção de movimento e suporte de vídeo chamada com sinalização SIP. Quando é detectado movimento, um alarme é enviado para o centro de controlo, que por sua vez o reenvia para o terminal do utilizador num formato compatível (por exemplo SMS, MMS, entre outros) com o mesmo. O utilizador pode a qualquer momento efectuar uma vídeo chamada para o domicílio e receber um *stream* vídeo em tempo real de cada uma das câmaras de vigilância.

Serve esta descrição para concluir que a plataforma OSGi permite a implementação de, virtualmente, todo o tipo de serviços. O factor mais limitativo à implementação de um determinado serviço acaba por ser a capacidade do *hardware* do *gateway* de serviços onde a plataforma é executada.

4.4. Observações finais

Como demonstrado ao longo deste capítulo, a plataforma OSGi permite a implementação de serviços (mais concretamente das aplicações que os implementam) através da cooperação

entre *bundles*. Esta arquitectura modular permite que *bundles* que fazem parte de uma determinada aplicação possam ser partilhados e utilizados por várias aplicações. Este facto aliado ao facto de a linguagem *Java* permitir o rápido desenvolvimento de módulos de *software*, torna o tempo de desenvolvimento de novos serviços muito curto. A execução da plataforma OSGi numa *Java Virtual Machine* apresenta ainda a vantagem de uma mesma implementação da plataforma ser facilmente portada para *gateways* de serviços com diferentes arquitecturas e/ou sistemas operativos. Uma consequência directa da portabilidade da *framework* é a portabilidade dos próprios *bundles* que, com um mínimo de modificações, podem também ser utilizados nos diferentes tipos de *gateways* de serviços. De uma forma geral, isto significa que os provedores de serviços poderão desenvolver novos serviços muito rapidamente e apenas necessitam de manter uma única versão de cada serviço, em vez de manterem *n* versões para *n* tipos de *gateways* de serviços. Contudo, a necessidade de uma JVM apresenta também desvantagens. A principal é provavelmente a falta de dispositivos capazes de suportar esta tecnologia eficazmente, o que em muito é devido à resistência dos fabricantes de CPE's em incluírem esta tecnologia nos seus produtos. No entanto, este problema a pouco e pouco vai sendo ultrapassado através da banalização de dispositivos domésticos com elevados recursos de *hardware*, tais como consolas de jogos, computadores *media centre*, entre outros. O facto de estes dispositivos tipicamente usufruírem também de uma ligação de banda larga, torna-os um ponto ideal para a execução de uma plataforma de serviços OSGi.

Essa mesma ligação de banda larga desempenha também um ponto fundamental no sucesso da plataforma OSGi em ambientes residenciais. Se até agora estes serviços resumiam-se a pouco mais que o serviço de acesso à Internet, num futuro próximo o leque de serviços será extenso, diversificado e terá como denominador comum a tecnologia IP. A tecnologia OSGi permite a implementação de todos estes novos serviços o que permite, entre outros, resolver o problema da configuração e gestão remota (como demonstrado com a implementação efectuada) de uma tão grande variedade de serviços. Tal é conseguido por mecanismos disponibilizados pela plataforma e também através da centralização dos vários serviços num único elemento da rede doméstica (em oposição a ter-se um *gateway* específico a cada serviço). Contudo, as vantagens inerentes ao uso da plataforma OSGi, por si só, poderão não ser suficientes para garantir o sucesso de uma plataforma de serviços residenciais baseada nesta tecnologia. As próprias redes de acesso terão de ser capazes de transportar todos estes novos serviços de um modo satisfatório. Por outras palavras as rede de acesso, independentemente da sua tecnologia, deverão poder ser caracterizadas como o que foi denominado na secção

3.1.2 de redes de acesso de próxima geração. Não obstante, nada impede que *gateways* de serviços equipados com a tecnologia OSGi sejam desde já disponibilizados para a prestação de serviços residenciais cujo desempenho é satisfatório com as redes de acesso actuais. O facto de a plataforma OSGi não depender da tecnologia usada na rede de acesso, nem necessitar de redes de acesso que disponibilizem grandes larguras de banda para operar satisfatoriamente (por exemplo, um *bundle* tem tipicamente apenas alguns *Megabytes* de tamanho) facilita ainda mais esta adopção imediata da tecnologia OSGi.

Provavelmente a maior lacuna que a especificação OSGi ainda apresenta é facto de definir um ambiente multi-serviço mas não um ambiente multi-provedor. Quer isto dizer que vários provedores de serviços terão de obrigatoriamente gerir o seus serviços através de um mesmo gestor remoto, possivelmente propriedade de uma terceira entidade, um operador de *gateway*. Quer também isto dizer que não existe um isolamento na plataforma entre *bundles* que pertencem a diferentes provedores de serviços. Porém, soluções baseadas na virtualização da plataforma foram já propostas [158], implementadas e testadas com resultados satisfatórios, o que levará brevemente à sua inclusão na especificação OSGi.

Capítulo 5 – Conclusões

O trabalho apresentado nesta dissertação pretendeu abordar aspectos chave da prestação de serviços residenciais por parte de operadores de telecomunicações e provedores de serviços. Esta abordagem não se cingiu apenas à actualidade mas também a cenários futuros, sendo que para estes últimos foram antevistos problemas e sugeridas algumas soluções.

Assim, iniciou-se a presente dissertação analisando cenários actuais típicos da prestação de serviços a clientes residenciais por parte de operadores de telecomunicações. Nesta análise foram focados não só os diferentes serviços disponibilizados, mas também as infra-estruturas de rede (acesso e doméstica) que suportam a distribuição desses mesmos serviços, bem como métodos que permitem o transporte, gestão, auto-configuração, aprovisionamento, autorização, taxação, entre outros, desses mesmos serviços. Deste estudo concluiu-se que os serviços disponibilizados são ainda reduzidos e variam de cenário para cenário. Da mesma forma, os métodos utilizados para a distribuição, controlo e gestão de serviços variam também de cenário para cenário e por vezes de serviço para serviço dentro de um mesmo cenário. De uma forma geral pode-se dizer que existem disparidades entre os vários panoramas analisados, sendo que estas disparidades existem, em parte, devido a mecanismos inerentes às infra-estruturas de rede utilizadas em cada um dos cenários analisados.

Um factor comum que existe entre as várias arquitecturas analisadas, é o facto de actualmente uma grande largura de banda estar ao dispor do cliente residencial, largura de banda essa até agora usada quase em exclusividade para o serviço de acesso à Internet. Todavia, devido a esta grande disponibilidade de largura de banda, uma parafernália de novos serviços residenciais

tem vindo a surgir. Contudo, para que estes novos serviços apresentem o comportamento esperado, novas funcionalidades têm de ser implementadas nas infra-estruturas de rede utilizadas para a sua distribuição. Uma lista de possíveis serviços a serem disponibilizados num futuro próximo a clientes residenciais, permitiu inferir acerca de algumas dessas funcionalidades. Suporte de *multicast*, qualidade de serviço e sistemas de monitorização de rede, encontram-se entre as novas funções a serem suportadas numa rede multi-serviço. Adicionalmente, uma vez que estes serviços tipicamente serão disponibilizados por diferentes provedores de serviços (ao contrário do que acontece actualmente), a infra-estrutura de rede deverá também suportar o seu uso simultâneo por provedores de serviços distintos. Considerando esta diversidade de requisitos, as redes de acesso baseadas na tecnologia *Ethernet* apresentam-se como a solução a adoptar para a distribuição de múltiplos serviços. Contudo, considerando o investimento que a sua implementação requer, os operadores de telecomunicações optarão, provavelmente, por continuar a evoluir as suas redes baseadas nas tecnologias xDSL e DOCSIS até que estas satisfaçam esses mesmos requisitos.

O facto de um grande número de serviços passar a estar disponível ao cliente, introduz ainda um novo desafio aos operadores de rede e aos próprios provedores de serviços: a necessidade de implementação de mecanismos que permitam a configuração e gestão remota das aplicações e dispositivos que residem no ambiente doméstico e que são vitais para o correcto comportamento dos serviços. Tais mecanismos irão possibilitar a disponibilização de múltiplos serviços a clientes residenciais de uma forma escalável, correcta e eficaz.

Esta tarefa está porém dificultada devido ao facto de tipicamente os dispositivos onde os serviços são executados (*gateways* de serviço) estarem distribuídos por toda a rede doméstica. Em oposição a esta distribuição, um *gateway* de serviços residenciais comum a vários serviços, apresenta como vantagens não só a centralização dos processos de configuração e gestão de dispositivos/aplicações, mas também a partilha de custos do *gateway* de serviços entre os vários provedores de serviços que lá depositam os seus serviços, a possibilidade de se gerarem novos serviços resultantes da interacção entre serviços executados no mesmo *gateway* de serviços, entre outras. Todavia, esta abordagem centralizada apresenta também uma grande desvantagem: um ambiente que permita a execução de aplicações originárias das mais diversas fontes (diferentes provedores de serviços), com requisitos e funcionalidades completamente distintas tem de estar implementado nesse *gateway* de serviços centralizado. A plataforma OSGi implementa não só mecanismos que permitem anular esta desvantagem, mas também outros

mecanismos que permitem por exemplo a instalação e remoção de serviços sem perturbação dos restantes serviços em funcionamento no mesmo *gateway*, a gestão do ciclo de vida e configuração remotas desses mesmos serviços, entre outros.

Por forma a considerar a plataforma OSGi como uma opção válida para a construção de um *gateway* de serviços centralizado, foi efectuada com sucesso a implementação de um cenário de monitorização médica remota sobre a plataforma OSGi. Este cenário pretende mostrar não só as capacidades de gestão remota do ciclo de vida dos serviços (como desejado pelos provedores de serviços) implementados sobre a plataforma OSGi, mas também a sua capacidade para lidar com o ambiente doméstico que rodeia o *gateway* de serviços, por natureza extremamente dinâmico.

Porém, a plataforma de serviços OSGi apenas permite a gestão e configuração remota das aplicações que implementam os serviços, a configuração remota ou a auto-configuração de dispositivos da rede doméstica, também necessária para a correcta operação dos serviços, está (pelo menos de uma forma directa) fora do alcance da plataforma OSGi. Contudo, mecanismos como os implementados pelo protocolo CWMP do DSL Forum permitem colmatar esta falha.

Em jeito de conclusão final, pode-se então afirmar que o sucesso da distribuição de múltiplos serviços em ambientes residenciais, por parte de múltiplos provedores de serviços, passa pela introdução na rede doméstica de *gateways* de serviços centralizados com suporte para o protocolo CWMP e nos quais uma plataforma OSGi é executada.

Como trabalho futuro pretende-se implementar um gestor remoto da plataforma OSGi (a ser utilizado pelo exclusivamente pelo operador de *gateway*) que inclua, para além da gestão do ciclo de vida de *bundles*, mecanismos de configuração, de gestão de políticas de segurança e de detecção e correcção de falhas. Pretende-se também efectuar uma completa integração da plataforma OSGi com o protocolo CWMP. O objectivo desta integração é conseguir a auto-configuração remota de dispositivos domésticos (por exemplo o *gateway* residencial) dos quais serviços instalados sobre a plataforma OSGi podem depender para operar correctamente. Esta auto-configuração dos dispositivos deverá ser efectuada de uma forma síncrona com a instalação/remoção do serviço residencial da plataforma OSGi. Finalmente, pretende-se evoluir o serviço de monitorização médica remota por forma a que submissão de medidas médicas seja efectuada através de mensagens SIP, usando um esquema de *Instant Messaging* [159], em alternativa ao protocolo HTTPS.

Referências

- [1] ANACOM: Autoridade Nacional de Comunicações. URL: <http://www.anacom.pt/>
- [2] ANACOM, Serviço de transmissão de dados, “Informação estatística dos serviços de transmissão de dados, 3º trimestre de 2006”. Novembro 2006. URL: <http://www.anacom.pt/template12.jsp?categoryId=193563>
- [3] DSL Forum. URL: <http://www.dslforum.org/>
- [4] Home Gateway Initiative. URL: <http://www.homegatewayinitiative.org/>
- [5] ETSI: European Telecommunications Standards Institute. URL: <http://www.etsi.org/>
- [6] ITU-T: International Telecommunication Union – Telecommunications sector. URL: <http://www.itu.int/ITU-T>
- [7] ETSI-TISPAN: European Telecommunications Standards Institute – Telecoms & Internet converged Services & Protocols for Advanced Networks. URL: http://portal.etsi.org/portal_common/home.asp?tbkey1=TISPAN
- [8] ITU-NGN: International Telecommunication Union – Next Generation Network Global Standards Initiative. URL: <http://www.itu.int/ITU-T/ngn/index.phtml>
- [9] MUSE: Multi Service Access Everywhere. URL: <http://www.ist-muse.org/>
- [10] Francisco Valera, Jaime García, Carmen Guerrero, Vitor Manuel Ribeiro and Vitor Pinto, “Demo of Triple Play Services with QoS in a Broadband Access Residential Gateway“. 25th IEEE Infocom, Barcelona, Spain. April 2006.
- [11] 25th IEEE Infocom 2006, Barcelona, Spain. April 2006. URL: <http://www.ieee-infocom.org/2006/>
- [12] Francisco Valera, Jaime García, Vitor Pinto, Iván Vidal, Vitor Ribeiro, Carmen Guerrero, Arturo Azcorra, “Pasarela residencial multiservicio con soporte de calidad garantizada para acceso de banda ancha”. XVI Telecom I+D Conference, Madrid, Spain. November 2006.
- [13] XVI Telecom I+D Conference, Madrid, Spain. November 2006. URL: <http://www.telecom-id.com/>
- [14] Iván Vidal, Francisco Valera, Jaime García, Arturo Azcorra, Vitor Pinto, Vitor Ribeiro, “Experiences with a QoS enabled Residential Gateway within a European multiservice broadband access environment”. 5th International Conference on Wired/Wireless Internet Communications, Coimbra, Portugal. May 2007
- [15] 5th International Conference on Wired/Wireless Internet Communications, Coimbra, Portugal. May 2007. URL: <http://wwic2007.dei.uc.pt/>
- [16] MUSE project D TF3.2, “Detailed description of Residential Gateway and advanced features”. December 2005.
- [17] DSL Forum Technical Report TR-030, “ADSL EMS to NMS Functional Requirements, Protocols at the U Interface for Accessing Data Networks using ATM/DSL”. February 2000
- [18] DSL Forum Technical Report TR-058, “Multi-Service Architecture & Framework

- Requirements”. September 2003
- [19] ITU-T Recommendation G.995.1, “Overview of digital subscriber line (DSL) Recommendations”. February 2001
- [20] ISO: International Organization for Standardization. URL: <http://www.iso.org/>
- [21] ISO/IEC 7498-1, “Open Systems Interconnection – Basic Reference Model: The Basic Model”. 1994
- [22] IEEE Std 802.1D-2004, “IEEE standard for local and metropolitan area networks—Media access control (MAC) Bridges (Incorporates IEEE 802.1t-2001 and IEEE 802.1w)”. June 2004
- [23] P. Srisuresh, K. Egevang, “Traditional IP Network Address Translator”, IETF RFC 3022. January 2001.
- [24] A. Berger, D. Romascanu, “Session Initiation Protocol”, IETF RFC 3261. December 2003
- [25] H. Schulzrinne, A. Rao, R. Lanphier, “Real Time Streaming Protocol (RTSP)”, IETF RFC 2326. April 1998
- [26] J. Postel, J. Reynolds, “File Transfer Protocol (FTP)”, IETF RFC 959. October 1985
- [27] W. Simpson, Editor, “The Point-to-Point Protocol (PPP)”, IETF RFC 1661. July 1994
- [28] IEEE Std 802.1p, “Traffic Class Expediting and Dynamic Multicast Filtering. Addenda to the IEEE Std 802.1D-1998 (Annex H.2) standard”. 1998.
- [29] K. Nichols, S. Blake, F. Baker, D. Black, “Definition of the Differentiated Services Field (DS Field) in the Ipv4 and Ipv6 Headers”, IETF RFC 2474. December 1998
- S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, W. Weiss, “An Architecture for Differentiated Services”, IETF RFC 2475. December 1998
- [30] IEEE Std 802.3-2005, “LAN/MAN CSMA/CD Access Method”. June 2005
- [31] IEEE Std 802.11a, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications—Amendment 1: High-speed Physical Layer in the 5 GHz band”. 1999
- [32] IEEE Std 802.11b, “Supplement to 802.11-1999, Wireless LAN MAC and PHY specifications: Higher speed Physical Layer (PHY) extension in the 2.4 GHz band”. 1999
- [33] IEEE Std 802.11e, “Wireless Medium Access Control (MAC) and Physical Layer (PHY) specifications: Medium Access Control (MAC) Quality of Service (QoS) Enhancements”. November 2005
- [34] IEEE Std 802.11g, “Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications – Amendment 4: Further Higher-Speed Physical Layer Extension in the 2.4 GHz Band”. June 2003
- [35] CopperGate Communications, "HomePNA 3 Multimedia Home Networking For Triple Play Services". October 2005. URL: <http://www.dallasces.org/talks/nesin.oct.2005.pdf>

- [36] HomePlug. URL: <http://www.homeplug.org>
- [37] IEEE Std 802.3u-1995, “Supplement to Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications Media Access Control (MAC) Parameters, Physical Layer, Medium Attachment Units, and Repeater for 100 Mb/s Operation, Type 100BASE-T”. 1995
- [38] F. Baker, Editor, “Requirements for IP Version 4 Routers”, IETF RFC 1812. June 1995
- [39] Universal Serial Bus Implementers Forum, Inc. URL: <http://www.usb.org/home>
- [40] IEEE Std 1394-1995, “High Performance Serial Bus”. 1995
- [41] Bluetooth Special Interest Group. URL: <https://bluetooth.org/spec/>
- [42] PKT-TR-ARCH-V01-991201, “PacketCable™ 1.0 Architecture Framework Technical Report”. December 1999
- [43] MEF: Metro Ethernet Forum. URL: <http://www.metroethernetforum.org/EFMA.htm>
- [44] ANSI: American National Standards Institute. URL: <http://www.ansi.org/>
- [46] ETSI TS 101 270-2 V1.2.1 (2003-07), “Transmission and Multiplexing™; Access transmission systems on metallic access cables; Very High Speed Digital Subscriber Line (VDSL); Part 2: Transceiver specification”. July 2003
- [47] ITU-T Recommendation V.34, “A modem operating at data rates of up to 33 600 bit/s for use on the general switched telephone network and on leased point-to-point 2-wire telephone-type circuits”. February 1998
- [48] ITU-T Recommendation V.90, “A digital modem and analogue modem pair for use on the Public Switched Telephone Network (PSTN) at data rates of up to 56 000 bit/s downstream and up to 33 600 bit/s upstream”. September 1998
- [49] ITU-T Recommendation G.992.1, “Asymmetric digital subscriber line (ADSL) transceivers”. July 1997
- [50] The ATM Forum Technical Committee, “Physical Interface Specification for 25.6 Mb/s over Twisted Pair Cable, af-phy-0040.000”. November 1995
- [51] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, “Hypertext Transfer Protocol – HTTP/1.1”, IETF RFC 2616. June 1999
T. Berners-Lee, R. Fielding, H. Frystyk, “Hypertext Transfer Protocol – HTTP/1.0”, IETF RFC 1945. May 1996
- [52] The ATM Forum Technical Committee, “Integrated Local Management Interface (ILMI) Specification Version 4.0, af-ilmi-0065.000”. September 1996
- [53] D. Grossman, J. Heinanen, “Multiprotocol Encapsulation over ATM Adaptation Layer 5”, IETF RFC 2684. September 1999
- [54] G. Gross, M. Kaycee, A. Lin, A. Malis, J. Stephens, “PPP Over AAL5”, IETF RFC 2364. July 1998
- [55] L. Mamakos, J. Evarts, D. Carrel, D. Simone, R. Wheeler, “A Method for Transmitting PPP Over Ethernet (PPPoE)”, IETF RFC 2516. February 1999

- [56] Charles Hornig, “A Standard for the Transmission of IP Datagram’s over Ethernet Networks”, IETF RFC 894. April 1984
- [57] M. Laubach, “Classical IP and ARP over ATM”, IETF RFC 1577. January 1994
- [58] DSL Forum Technical Report TR-043, “Protocols at the U Interface for Accessing Data Networks using ATM/DSL”. August 2001
- [59] B. Lloyd, W. Simpson, “PPP Authentication Protocols”, IETF RFC 1334. October 1992
- [60] W. Simpson, “PPP Challenge Handshake Authentication Protocol (CHAP)”, IETF RFC 1994. August 1996
- [61] R. Droms, “Dynamic Host Configuration Protocol”, IETF RFC 2131. March 1997
- [62] S. Cobb, “PPP Internet Protocol Control Protocol Extensions for Name Server Addresses”, IETF RFC 1877. December 1995
- [63] ITU-T Recommendation G.992.3, “Asymmetric digital subscriber line transceivers 2 (ADSL2)”. January 2005
- [64] ITU-T Recommendation G.992.5, “Asymmetric Digital Subscriber Line (ADSL) transceivers – Extended bandwidth ADSL2 (ADSL2+)”. January 2005.
- [65] Internode Systems, “ADSL, ADSL2 and ADSL2+ Speeds and Reach Compared”. URL: <http://www.internode.on.net/adsl2/graph/>
- [66] Alcatel Telecommunications Review – 4th Quarter 2000, “VDSL, fiber-fast data transmission over copper pairs”. URL: <http://www.alcatel.com/doctypes/articlepaperlibrary/pdf/ATR2000Q4/gb/10antogb.pdf>
- [67] ISO/IEC TR 13818-1:1997, “Generic coding of moving pictures and associated audio information. Extension for real time interface for systems decoders”. 1997
- [68] CableLabs Certified Cable Modem project, “Data Over Cable Service Interface Specification”. URL: <http://www.cablemodem.com/>
- [69] CableLabs: Cable Television Laboratories, Inc. URL: <http://www.cablelabs.com/>
- [70] ITU-T Recommendation J.112, “Transmission systems for interactive cable television services”. March 1998
- [71] ITU-T Recommendation J.122, “Second-generation transmission systems for interactive cable television services – IP cable modems”. December 2002
- [72] ANSI/SCTE 22-1 2002, “Data-Over-Cable Service Interface Specification DOCSIS 1.0 Radio Frequency Interface (RFI)”. 2002
- [73] ANSI/SCTE 22-2 2002, “Data-Over-Cable Service Interface Specification DOCSIS 1.0 Baseline Privacy Interface (BPI)”. 2002
- [74] CM-SP-RFIV1.1-C01-050907, “Data-Over-Cable Service Interface Specifications DOCSIS 1.1, Radio Frequency Interface Specification”. September 2005
- [75] CM-SP-BPI+-I12-050812, “Data-Over-Cable Service Interface Specifications DOCSIS 1.1, Baseline Privacy Plus Interface Specification”. August 2005

- [76] CM-SP-RFIV2.0-I10-051209, “Data-Over-Cable Service Interface Specifications DOCSIS 2.0, Radio Frequency Interface Specification”. December 2005
- [77] J. Case, M. Fedor, M. Schoffstall, J. Davin, “Simple Network Management Protocol (SNMP)”, IETF RFC 1157. May 1990
- [78] PKT-SP-EC-MGCP-I11-050812, “PacketCable Network-Based Call Signalling Protocol Specification”. August 2005
- [79] F. Andreassen, B. Foster, “Media Gateway Control Protocol (MGCP) Version 1.0”, IETF RFC 3435. January 2003
- [80] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, “RTP: A Transport Protocol for Real-Time Applications”, IETF RFC 3550. July 2003
- [81] K. Sollins, “The TFTP Protocol (Revision 2)”, IETF 1350. July 1992
- [82] J. Postel, K. Harrenstien, “Time Protocol”, IETF RFC 0868. May 1983
- [83] IEEE Std 802.3ah.-2004, “Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications. Amendment: Media Access Control Parameters, Physical Layers, and Management Parameters for Subscriber Access Networks”. 2004
- [84] IEEE Std 802.1Q-2005, “IEEE Standards for Local and Metropolitan Area Networks, Virtual Bridged Local Area Networks”. December 2005
- [85] IEEE Std 802.1AD-2005, “IEEE Standard for Local and metropolitan area networks, Virtual Bridged Local Area Networks, Amendment 4: Provider Bridges”. December 2005
- [86] IEEE Std 802.1w, “Rapid Reconfiguration of Spanning Tree. Addenda to IEEE Std 802.1D-2004”. 2004
- [87] IEEE Std 802.1s, “Multiple Spanning Trees. Addenda to IEEE Std 802.1Q-2003”. 2003
- [88] IEEE 802.1ah, Draft 3.3, “Provider Backbone Bridges”. December 2006
- [89] M. Lasserre, V. Kompella, “Virtual Private LAN Service (VPLS) Using Label Distribution Protocol (LDP) Signalling”, IETF RFC 4762. January 2007
- [90] K. Kompella, Y. Rekhter, “Virtual Private LAN Service (VPLS) Using BGP for Auto-Discovery and Signalling”, IETF RFC 4761. January 2007
- [91] Metro Ethernet Forum, Technical Specification MEF 10, “Ethernet Services Attributes Phase 1”, November 2004
- [92] ITU-T Recommendation Q.933, “Digital Subscriber Signalling System No. 1 (DSS 1) – Signalling specifications for frame mode switched and permanent virtual connection control and status monitoring”. February 2003.
- [93] Metro Ethernet Forum, Technical Specification MEF 16, “Ethernet Local Management Interface (E-LMI)”. January 2006
- [94] IEEE 802.1X-2004, “IEEE Standards for Local and metropolitan area networks: Port-Based Network Access Control”. November 2004
- [95] Metro Ethernet Forum, “Ethernet in the First Mile over Point-to-Point Fibber – A

- Tutorial”. URL:
<http://www.metroethernetforum.org/PDFs/EFMA/EFM%20Optical%20Tutorial%20v3.pdf>
- [96] Joan Daemen, Steve Borg and Vincent Rijmen: “The Design of Rijndael: AES – The Advanced Encryption Standard”. 2002
- [97] ISO/IEC 11172-1:1995, “Information technology. Coding of moving pictures and associated audio for digital storage media at up to about 1,5 Mbit/s. Systems”. 1995
- [98] Skype. URL: <http://www.skype.com/>
- [99] DSL Forum Technical Report TR-094, “Multi-Service Delivery framework for Home Networks”. August 2004
- [100] Home Gateway Initiative, “Home Gateway Technical Requirements, Release 1.0”. July 2006
- [101] DSL Forum Technical Report TR-046, “Auto-Configuration Architecture & Framework”. February 2002
- [102] TiVo Inc. URL: <http://www.tivo.com>
- [103] Scientific Atlanta. URL: <http://www.sciatl.com/>
- [104] OpenTV middleware. URL: <http://www.opentv.com/>
- [105] Microsoft TV. URL: <http://www.microsoft.com/tv/default.msp>
- [106] Multimedia Home Platform. URL: <http://www.mhp.org/>
- [107] Digital Video Broadcasting project. URL: <http://www.dvb.org/>
- [108] ETSI TS 102 812 V1.2.1 (2003-06), “Digital Video Broadcasting (DVB); Multimedia Home Platform (MHP) Specification 1.1.1”. June 2006
- [109] OSGi Technology. URL: http://www.osgi.org/osgi_technology/index.asp?section=2
- [110] The OSGi Alliance, “OSGi Service Platform, Release 1”. May 2000. URL: http://www.osgi.org/osgi_technology/spec_download3.asp?Accept=Accept
- [111] The OSGi Alliance, “OSGi Service Platform, Release 2”. October 2001. URL: http://www.osgi.org/osgi_technology/spec_download3.asp?Accept=Accept
- [112] The OSGi Alliance, “OSGi Service Platform, Release 3”. March 2003. URL: http://www.osgi.org/osgi_technology/spec_download3.asp?Accept=Accept
- [113] The OSGi Alliance, “OSGi Service Platform, Release 4”. August 2005. URL: http://www.osgi.org/osgi_technology/spec_download3.asp?Accept=Accept
- [114] Tim Lindholm, Frank Yellin, “The Java Virtual Machine Specification – First Edition”, Addison Wesley. June 2006
- [115] Alan Sommerer, “The Java Archive (JAR) File Format”. September 1998. URL: <http://java.sun.com/developer/Books/javaprogramming/JAR/index.html>
- [116] Kirk Chen, Li Gong, “Programming Open Service Gateways with Java Embedded Server™ Technology”, Addison Wesley Professional. August 2001
- [117] Java Platform, Standard Edition (Java 2 Platform). URL: <http://java.sun.com/javase/index.jsp>

- [118] Java Platform 1.1 API. URL: <http://java.sun.com/products/archive/jdk/1.1/>
- [119] Foundation Profile. URL: <http://java.sun.com/products/foundation/>
- [120] J2ME, Java 2 Micro Edition. URL: <http://java.sun.com/j2me>
- [121] Manifest Format. URL: <http://java.sun.com/j2se/1.4.2/docs/guide/jar/jar.html>
- [122] T. Berners-Lee, R. Fielding, L. Masinter, “Uniform Resource Identifier (URI) – Generic Syntax”, IETF RFC 3986. January 2005
- [123] Humberto Cervantes, Didier Donsez, Richard Hall, “La plateforme dynamique de services OSGI™”. January 2006. URL: <http://www-adele.imag.fr/users/Didier.Donsez/cours/osgi.pdf/>
- [124] Roy W. Schulte, Yefim V. Natis, “SOA – Service Oriented Architecture”. April 2003. URL: http://www.gartner.com/DisplayDocument?doc_cd=114358&ref=g_fromdoc
- [125] Li Gong, “Java™ 2 Platform Security Architecture, Version 1.2”. URL: <http://java.sun.com/j2se/1.4.2/docs/guide/security/spec/security-spec.doc.html>
- [126] The OSGi Alliance, “OSGi Service Platform, Release 4 – Service Compendium”. August 2005. URL: http://www.osgi.org/osgi_technology/spec_download3.asp?Accept=Accept
- [127] Java Servlet Technology. URL: <http://java.sun.com/products/servlet/>
- [128] Dave Raggett, Arnaud Le Hors, Ian Jacobs, “HTML 4.01 Specification”, W3C Recommendation. December 1999. URL: <http://www.w3.org/TR/html401/>
- [129] Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, Eve Maler, François Yergeau, John Cowan, “Extensible Markup Language (XML) 1.1 (Second Edition)”, W3C Recommendation. September 2006. URL: <http://www.w3.org/TR/2006/REC-xml11-20060816/>
- [130] Java API for XML Processing (JAXP). URL: <http://java.sun.com/webservices/jaxp/>
- [131] DSL Forum Technical Report TR-069, “CPE WAN Management Protocol”. May 2004
- [132] R. Presuhn, J. Case, K. McCloghrie, M. Rose, S. Waldbusser, “Management Information Base (MIB) for the Simple Network Management Protocol (SNMP)”, IETF RFC 3418. December 2002
- [133] J. Case, M. Fedor, M. Schoffstall, J. Davin, “Simple Network Management Protocol (SNMP)”, IETF RFC 1157. May 1990
- [134] E. Rescorla, “HTTP over TLS”, IETF RFC 2818, May 2000
- [135] T. Dierks, C. Allen, “The TLS Protocol, Version 1.0”, IETF RFC 2246. January 1999
- [136] Don Box, David Ehnebuske, Gopal Kakivaya, Andrew Layman, Noah Mendelsohn, Henrik Frystyk Nielsen, Satish Thatte, Dave Winer, “Simple Object Access Protocol (SOAP) 1.1.”; W3C Recommendation. May 2000. URL: <http://www.w3.org/TR/2000/NOTE-SOAP-20000508>
- [137] DSL Forum Technical Report TR-098, “Internet Gateway Device Version 1.1 Data Model for TR-069”. September 2005
- [138] Information Society Technologies. URL: <http://cordis.europa.eu/ist/>

- [139] B.Melander, H. Mickelsson, Z. Ghebretensaé, K. Oberle, A.Sitek, B. Radier, V. Kulkarni: “FMC from a Fixed Access Perspective”. 15th IST Mobile & Wireless Communications Summit, Myconos, Greece, July 2006
- [140] Peter Vetter, “IST MUSE”. 2006. URL: http://www.ist-muse.org/Documents/MUSE_General_V03.pdf
- [141] MUSE Project D D3.2, “Gateway Solutions”. September 2005
- [142] Electronics Industries Association, “EIA Standard RS-232-C Interface Between Data Terminal Equipment and Data Communication Equipment Employing Serial Data Interchange”. August 1969.
- [143] Infrared Data Association. URL: <http://www.irda.org/>
- [144] MySQL AB. URL: <http://www.mysql.com/>
- [145] Java Community Process, “JSR 3, Java Management Extensions (JMX) Specification”. September 2000. URL: <http://www.jcp.org/en/jsr/detail?id=3>
- [146] Java Community Process, “JSR 160, Java Management Extensions (JMX) Remote API”. October 2003. URL: <http://www.jcp.org/en/jsr/detail?id=160>
- [147] OSCAR: An OSGi framework implementation. URL: <http://oscar.objectweb.org/>
- [148] OBR: OSCAR *Bundle* Repository. URL: <http://oscar-osgi.sourceforge.net/>
- [149] OSGi Alliance, “RFC-0112 *Bundle* Repository”, February 2006. URL: http://bundles.osgi.org/rfc-0112_BundleRepository.pdf
- [150] The Apache Software Foundation – Apache Felix Incubator. URL: <http://cwiki.apache.org/FELIX/index.html>
- [151] The Apache Software Foundation. URL: <http://www.apache.org/>
- [152] Java Remote Method Invocation specification. URL: <http://java.sun.com/j2se/1.5.0/docs/guide/rmi/spec/rmiTOC.html>
- [153] Java 2 Platform Standard Edition (J2SE) 5.0. URL: <http://java.sun.com/j2se/1.5/>
- [154] RXTX native lib. URL: <http://www.rxtx.org/>
- [155] Jakarta Commons HTTP Client. URL : <http://jakarta.apache.org/commons/httpclient/>
- [156] Barking Intruder Detection System. URL: http://barkids.forge.os4os.org/index_en.html
- [157] Rafael Barriuso Maicas, “BarkIDS: el servicio de videovigilancia del Hogar Digital”. 2004. URL: <http://internetng.dit.upm.es/Barkids.pdf>
- [158] Yvan Royon, Stephane Frenot, and Frederic Le Mouel: “Virtualization of Service Gateways in Multi-provider Environments”. 9th International SIGSOFT Symposium on Component-Based Software Engineering, Stockholm, Sweden. June 2006.
- [159] B. Campbell, J. Rosenberg, H. Schulzrinne, C. Huitema, D. Gurle, “Session Initiation Protocol (SIP) Extension for Instant Messaging”, IETF RFC 3428. December 2002
- [160] DSL Forum Technical Report TR-037, “Auto-Configuration for the Connection Between the DSL Broadband Network Termination (B-NT) and the Network using

- ATM”. March 2001
- [161] ISO/IEC TR 9577, “Protocol Identification in the Network Layer”. 1999
- [162] ITU-T Recommendation Q.2931, “Broadband Integrated Services Digital Network (B-ISDN); Digital Subscriber Signalling System No. 2 (DSS 2); User-Network Interface (UNI) Layer 3 Specification for Basic Call/Connection Control”. October 1995
- [163] M. Perez, F. Liaw, A. Mankin, E. Hoffman, D. Grossman, A. Malis, “ATM Signalling Support for IP over ATM”, IETF RFC 1755. February 1995
- [164] G. Gross, M. Kaycee, A. Lin, A. Malis, J. Stephens, “PPP Over AAL5”, IETF RFC 2364. July 1998
- [165] Solwise, “HomePlug 1.0, HomePlug Turbo and HomePlug AV performance and features comparison”. URL:
<http://www.solwiseforum.co.uk/downloads/files/solwise-homeplug-comparison.pdf>

Anexo II – Tabelas

Attribute	Type of Parameter Value
EVC Type ([91], Section 6.1)	Point-to-Point or Multipoint-to-Multipoint
EVC ID ([91], Section 6.2)	An arbitrary string, unique across the MEN, for the EVC supporting the service instance
UNI List ([91], Section 6.3)	A list of UNI Identifiers ([91], Section 7.1)
CE-VLAN ID Preservation ([91], Section 6.5.1)	Yes or No
CE-VLAN CoS Preservation ([91], Section 6.5.2)	Yes or No
Unicast Service Frame Delivery ([91], Section 6.4.1.1)	Discard, Deliver Unconditionally, or Deliver Conditionally. If Deliver Conditionally is used, then the conditions MUST be specified.
Multicast Service Frame Delivery ([91], Section 6.4.1.2)	Discard, Deliver Unconditionally, or Deliver Conditionally. If Deliver Conditionally is used, then the conditions MUST be specified.
Broadcast Service Frame Delivery ([91], Section 6.4.1.3)	Discard, Deliver Unconditionally, or Deliver Conditionally. If Deliver Conditionally is used, then the conditions MUST be specified.
Layer 2 Control Protocols Processing ([91], Section 6.6)	Entries from [91], Table 1 labelled Tunnel or Discard.
EVC Performance ([91], Section 6.7)	Performance objectives for Frame Delay Performance, Frame Delay Variation Performance, and Frame Loss Ratio Performance as specified in [91], Section 6.7

Tabela 5: Atributos de serviço EVC [91].

Attribute	Type of Parameter Value
UNI Identifier ([91], Section 7.1)	Any string
Physical Medium ([91], Section 7.2)	A Standard Ethernet PHY
Speed ([91], Section 7.2)	10 Mbps, 100 Mbps, 1 Gbps, or 10 Gbps
Mode ([91], Section 7.2)	Full Duplex or Auto-Negotiation
MAC Layer ([91], Section 7.3)	IEEE 802.3 – 2002
Service Multiplexing ([91], Section 7.4)	Yes or No
UNI EVC ID ([91], Section 7.5.2)	A string formed by the concatenation of the UNI ID and the EVC ID
CE-VLAN ID for untagged and priority tagged Service Frames ([91], Section 7.5.1)	A number in 1, 2, ..., 4094.
CE-VLAN ID/EVC Map ([91], Section 7.6)	Map as per [91], Section 7.6
Maximum Number of EVCs ([91], Section 7.7)	Integer ≥ 1

Bundling ([91], Section 7.8)	Yes or No
All to One Bundling ([91], Section 7.9)	Yes or No
Ingress Bandwidth Profile Per Ingress UNI ([91], Section 7.10.3)	No or parameters as defined in [91], Section 7.10.1
Ingress Bandwidth Profile Per EVC ([91], Section 7.10.4)	No or parameters as defined in [91], Section 7.10.1
Ingress Bandwidth Profile Per Class of Service Identifier ([91], Section 7.10.5)	No or parameters as defined in [91], Section 7.10.1
Layer 2 Control Protocols Processing ([91], Section 7.1.2)	Entries from [91], Table 1 with each being labelled Discard, Peer, or Pass to EVC

Tabela 6: Atributos de serviço UNI [91]

Tecnologia	Notas	750kbps+	3Mbps+	6Mbps+	9 Mbps+	12 Mbps+	19 Mbps+	50 Mbps+	90 Mbps+	Standard QoS
100bT Ethernet c/ cabo CAT5		✓	✓	✓	✓	✓	✓	✓	✓	802.1D Anexo H.2
10bT Ethernet c/ cabo CAT5		✓	✓	✓	✓	Não	Não	Não	Não	802.1D Anexo H.2
802.11b	1, 4, 7	✓ 40-60m ou 3 paredes	✓ 30-35m ou 1 parede	#, 1 Quarto s/ paredes	Não	Não	Não	Não	Não	802.11e
802.11g	1, 2, 4, 7	✓ 40-50m ou 2 paredes	✓ 40-50m ou 2 paredes	✓ 40-50m ou 2 paredes	✓ 30-35m ou 1 parede	✓ 1 sala, s/ paredes	# 1 sala, s/ paredes	Não	Não	802.11e
802.11a	1, 3, 4, 7	✓ 30-35m ou 1.5 parede	✓ 30-35m ou 1.5 parede	✓ 30-35m ou 1.5 parede	✓ 20-25m ou 1 parede	✓ 1 sala, s/ paredes	# 1 sala, s/ paredes	Não	Não	802.11e
HomePlug 1.0	1, 5, 8	✓	✓	%	Não	Não	Não	Não	Não	HomePlug QoS
HomePlug Turbo	1,5,8	✓	✓	✓	✓	✓	%	Não	Não	HomePlug QoS
HomePlug AV	1,5	✓	✓	✓	✓	✓	✓	✓	#	HomePlug QoS (implementa QdSG)
HPNA 2.0	1, 6, 8	✓	✓	%	Não	Não	Não	Não	Não	p/ VoHPNA apenas
HPNA 3.0	1, 6	✓	✓	✓	✓	✓	✓	✓	✓	HPNA3 QoS (implementa QdSG)

Tabela 7: Taxa/Alcance de tecnologias usadas em redes residenciais. Baseado na tabela 5 de [99] e [165].

Notas:

+ = as taxas de transmissão apresentadas têm em consideração o *overhead* físico, da camada de ligação dados e IP.

= esta taxa apenas pode ser conseguida em algumas casas.

% = esta taxa deverá ser conseguida na maioria das casas.

✓ = uma casa de tamanho mediano deverá ser conseguir ser coberta a esta taxa.

1 = o standard afirma que uma taxa mais elevada é possível atingir, no entanto normalmente não é possível atingir essas taxas.

2 = sem a presença da tecnologia 802.11b.

3 = a tecnologia 802.11a não é actualmente permitida fora dos Estado Unidos e do Japão.

4 = em gráfico, parede = parede interior; andar = 2 paredes interiores; parede exterior = 4 parede interiores.

5 = assume que o equipamento não é conectado através de um disjuntor diferencial ou UPS.

6 = assume que pelo menos é usado cabo do tipo CAT3.

7 = a performance do equipamento sem fios é altamente influenciada por materiais usados na construção, na posição de paredes, espelhos, lareiras, armários, mobília, presença de *Bluetooth*, telefones a operar nos 2.4 GHz, fornos de microondas, etc.

8 = HomePlug e HomePNA têm capacidades de classificação de tráfego, mas estas não estão acessíveis a aplicações.

Anexo III – Implementações da plataforma OSGi

- **Oscar**

Tipo		<i>open source</i>
Compatibilidade com a norma		Quase toda a <i>Release 3</i>
URL		http://oscar.objectweb.org/
Breve Descrição (da página Web)		<i>“Oscar is an open source implementation of the Open Services Gateway Initiative framework specification; the goal is to provide a completely compliant implementation of the OSGi framework specification. Oscar is currently compliant with a large portion of the OSGi 3 specifications, although certain compliance work still needs to be completed.”</i>
Repositório de Bundles		http://oscar-osgi.sourceforge.net/

- **Apache Felix**

Esta implementação aparece no seguimento da implementação OSCAR.

Tipo		<i>open source</i>
Compatibilidade com a norma		Parte da <i>Release 4</i>
URL		http://cwiki.apache.org/FELIX/index.html
Breve Descrição (da página Web)		<i>“Felix is a community effort to implement the OSGi R4 Platform, which includes the OSGi framework and standard services, as well as providing and supporting other interesting OSGi-related technologies. The ultimate goal is to provide a completely compliant implementation of the OSGi framework and standard services and to support a community around this technology. Felix currently implements a large portion of the OSGi 4 specification, but additional work is necessary for full compliance.”</i>
Repositório de Bundles		http://bundles.osgi.org/obr/browse?_xml=1&cmd=repository (Repositório da Aliança OSGi)

- **Knopflerfish**

Tipo		<i>open source</i>
Compatibilidade com a norma		Quase toda a <i>Release 4</i>
URL		http://www.knopflerfish.org/index.html
Breve Descrição (da página Web)		<i>“The Knopflerfish project is based on the Gatspace GDSP OSGi framework, which has been in development since 1999”</i>
Repositório de Bundles		http://www.knopflerfish.org/repo/index.html

- **ProSyst**

Tipo		Comercial
Compatibilidade com a norma		Release 4 (Certificado)
URL		http://www.prosyst.com/products/osgi_framework.html
Breve Descrição (da página Web)		<i>“mBedded Server Framework Package, an implementation of the OSGi Service Platform specification Release 4, is a Java Virtual Machine and operating system independent framework for deployment of applications and their components”</i>
Repositório de Bundles		http://www.prosyst.com/products/bundels.html

- **Avelink**

Tipo		Comercial
Compatibilidade com a norma		Release 3 (Certificado)
URL		http://www.avelink.com/OSGi/products/index.htm
Breve Descrição (da página Web)		<i>“Atinav offers aveLink Embedded Gateway, which is an implementation of OSGi service framework specification 3.0. aveLink Embedded Gateway is officially certified as an OSGi compliant product by the OSGi Alliance”</i>
Repositório de Bundles		Não disponível