

Architectural Support for ubiquitous access to multimedia content

by

Maria Teresa Pinto de Andrade

A Dissertation submitted in partial satisfaction of the
requirements for the degree of

Doctor of Philosophy

in

Telecommunications Engineering

to the

Department of Electrotechnical and Computing Engineering

of the

Faculty of Engineering

of the

University of Porto

Advisor: Professor Artur Pimenta Alves

2007

To my advisor,
Prof. Artur Pimenta Alves,
my family and my colleagues.

Acknowledgements

I want to express my gratitude to my advisor and to my colleagues at INESC Porto for their support, help and fruitful discussions and collaboration. In particular to my advisor, Prof. Pimenta Alves, for giving me the opportunity to participate in many different and interesting technological research projects and develop my engineering skills and knowledge. Special thanks are also endorsed to my colleagues at INESC Porto who, along with me, have formed the ENTHRONE's INESC team: Pedro Carvalho, Pedro Souto, Hélder Castro, Lucian Ciobanu, Catalin Calistru and Daniel Oancea. They have made possible to turn into reality all the ideas and concepts elaborated for ENTHRONE that were in line with the objectives of this thesis. Likewise to the ENTHRONE partners, with whom fruitful discussions were conducted and who have also given their contribution to the successful achievement of the proposed goals. In particular to Olimpiu Negru, the initiator and coordinator of the project, for his vision and for having supported and shared my ideas and beliefs for the project. To my family and friends who have always encouraged me, never doubting of '*a happy ending*'.

Abstract

Architectural Support for ubiquitous access to multimedia content

The work presented in this dissertation reflects the research, developments, experimental tests and analyses conducted by the candidate towards the obtention of feasible and optimal solutions for the problem of achieving QoS, context-aware and bandwidth-efficient transmission of audio-visual sources. The focus of this thesis is on designing and optimising the infra-structural support for the provision of ubiquitous and non-discriminatory access to multimedia content, looking at dynamic operation aspects. We developed a framework that combines the use of metadata with distributed technologies to achieve this goal. The use of metadata is instrumental to provide the support for content adaptation operations, which in turn is essential to provide the QoS-aware ubiquitous access to multimedia content regardless of coding formats, network resources availability or user terminal capabilities and preferences. This can be translated in the provision of context-aware multimedia services resulting in universal access to multimedia content.

The outcomes of this thesis include the specification of a middleware architecture and associated services to enable the provision of context-aware adaptable multimedia services in heterogeneous environments. The undertaken work has enabled the design and development of a reference architecture in the form of a set of services and interfaces based on standard emergent technologies, notably the MPEG-21 standards and Web Services technologies. This reference architecture can be used in a number of different environments, thus constituting an interoperable framework suiting different requirements and goals.

Keywords: Context awareness, content adaptation, ubiquitous access, interoperability, Quality of Service, metadata, ontology, reasoning, statistical inference, Bayesian analysis, probabilistic modeling, statistical multiplexing.

Brief meaning of definitions and abbreviations used throughout the text:

3GPP, 3rd Generation Partnership Project - The 3GPP is a collaboration agreement established in December 1998 between a number of telecommunications standards bodies.

Admin, Administrator - The user who has privileges to perform all kind of management tasks upon a system or subsystem.

API, Application Programming Interface - An API is an abstract concept expressed through a programming language, describing the functionality offered by a software system. It provides the means to describe how two computer programmes may interact. By specifying their own API, any software system or computer programme is providing the indications on how external software systems, applications or libraries may access and use their functionality.

ATM, Asynchronous Transfer Mode - Broadband transmission and switching technology.

AS, Autonomous System - Networking infrastructure managed by a single Network Operator.

A/V, Audio Visual - The term A/V is used to designate audiovisual sources.

BW, Bandwidth - In the context of multimedia digital services, it is used to describe the capacity of the communications channel, indicating the rate at which bits may be transmitted through the channel. It is also used to specify the amount of bits per second required to transmit a given signal over a network.

CBR, Constant Bit Rate - An encoded bit stream with a constant number of bits per unit of time.

CM, Content Manager - The subsystem responsible for the communication with the Content Provider for all tasks related with content browsing, searching, adaptation and management.

CP, Content Provider - A CP provides content either generated in real-time upon request or retrieved for a data base holding Digital Items.

DCMI, Dublin Core Metadata Initiative - The DCMI is an open organisation engaged in the development of open specifications for metadata to enhance the access to digital resources.

DI, Digital Item - As defined in the MPEG-21 specification, a DI is a structured object with a standard representation, constituting the fundamental unit of transaction. A DI contains multimedia content (resources in MPEG terminology), associated metadata, and structures representing relations between resources and metadata.

DIA, Digital Item Adaptation - DIA, part 7 of the MPEG-21 standard, specifies a set of tools that enable to adapt DIs in order to provide the user the best possible experience under the available conditions. This includes respecting user preferences terminal capabilities, network characteristics and conditions and natural environment characteristics.

DID, Digital Item Declaration - DID is an XML document expressed according to part 2 of the MPEG-21 standard (DID Language, DIDL), which specifies a set of abstract terms and concepts to construct an XML model for the declaration of DIs.

DiffServ, Differentiated Services - The differentiated services model, specified by the IETF, defines a networking architecture capable of offering different service levels or different quality assurances in end-to-end (E2E) communications. The model aggregates into distinct classes, individual flows that present requirements and characteris-

tics within the same range. It defines a set of building blocks from which a variety of aggregate behaviors can be expected to build an E2E quality of service connection.

DISP, Dispatcher - A subsystem implementing the decision-making layer of the content mediation platform.

DRM, Digital Rights Management - DRM defines the technologies that are used to provide secure content and ensure the protection of IPRs (Intellectual Property Rights) of the content's owners and of the usage rights of consumers.

DVB, Digital Video Broadcasting - The DVB consortium is composed of a number of European entities that play a key role in the A/V content value chain, notably Broadcasters, TV Producers, Telecom Operators and Consumer Electronics Manufacturers. The aim of the DVB group is to establish a number of open specifications for the advancement of digital television services in Europe, comprising interactive and mobile services. DVB has been the responsible for the issuing the standards for digital television over the air or terrestrial (DVB-T), using satellite communications (DVB-S and DVB-S2), through cable (DVB-C) and over the air to mobile or handheld devices (DVB-H). The DVB consortium has also delivered open interoperable specifications for the creation, reception and play-out of digital interactive television applications, the so-called MHP (Multimedia Home Platform) standard.

DVD, Digital Versatile Disc - Optical disk storage with a capacity of 4,7GByte. Especially used for storing digital video and audio, in particular digital movies.

EBU, European Broadcasting Union - The EBU is an association of national broadcasters, aiming at promoting cooperation between broadcasters and facilitating the exchange of audiovisual content.

EPG, Electronic Programming Guide - the EPG provides information concerning the contents and the scheduling of DVB streams.

ER, Event Reporting - ER is specified in part 11 of MPEG-21 and addresses the standardisation of performance metrics and interfaces for all reportable events in MPEG-

21. It provides the means of capturing and transporting these metrics and of defining interfaces that refer to Digital Items, Peers, environments, processes, transactions and Users. One example relates to monitoring the usage of copyrighted material. Another one that has been explored within the ENTHRONE project, is to instruct QoS probes on how to provide quality measurements and to pass those measurements between subsystems.

ETSI, European Telecommunications Standards Institute - This European body is officially responsible for standardization of Information and Communication Technologies (ICT) within Europe, including telecommunications, broadcasting and related areas.

(SP) FE, (Service Provider) Front End - The interface presented to the user to access the multimedia services.

FGS, Fine Granular Scalability - FGS is a scalable profile of the MPEG-4 video standard. It consists in a set of video coding tools that support different forms of scalability, namely quality (also designated as SNR, Signal to Noise Ratio), temporal, and hybrid temporal-SNR.

FTA, Free To Air - FTA is often used to designate the broadcasting of free TV programmes or services over terrestrial channels .

GOP, Group Of Pictures - Association of a repetitive number of images, providing support for random access to the coded video bit stream.

GPRS, General Packet Radio Service - GPRS is a packet-oriented radio transmission technology that allows to transmit services such as SMS (Short Message Service) or Internet data to GSM (Global System for Mobile Communications) mobile phones. GSM systems that use GPRS technology are designated as "*2.5G systems*".

HDTV, High Definition Television - HDTV refers to the transmission, usually broadcasting, of TV signals with higher spatial and temporal definition than the regular

TV broadcastings (PAL, NTSC, SECAM). HDTV signals have a 16:9 aspect ratio and usually their spatial resolution is twice that of standard TV. Temporal resolution may also double that of standard signals. It may use interlaced or progressive formats. Examples of resolutions are 1152i (interlaced: 1152 lines, 1440 pixels) or 720p (progressive: 720 lines, 1280 pixels).

HiperMAN, High Performance Radio Metropolitan Area Network - HiperMAN is an ETSI standard providing broadband fixed wireless access at radio frequencies between 2 GHz and 11 GHz primarily in the residential and small business user environments. It has been designed in cooperation with IEEE so that HiperMAN and a subset of IEEE 802.16 (WiMAX) may interoperate.

HTML, HyperText Markup Language - HTML is the markup language most extensively used to create Web pages. It is a language that uses labels, the *markup tags*, to describe the structure and semantics (up to some extent) of text-based information. An HTML file is a text file containing markup tags that provide Web browsers the indication on how to present that file.

HTTP, HyperText Transfer Protocol - HTTP is an application-level protocol used to exchange information between systems in the Internet. It was developed by the W3C and the IETF. Its current version, HTTP/1.1, has been published as IETF RFC 2616bis.

IEC, International Electrotechnical Commission - IEC is a global organization that prepares and publishes international standards for electrical, electronic and related technologies.

IEEE, Institute for Electrical and Electronics Engineers - The IEEE, pronounced *Eye-triple-E*, is a non-profit organization committed to the advancement of technology. IEEE's area of intervention encompasses all electro and information technologies as well as sciences. The IEEE is engaged in activities related with the specification of standards, scientific publications and conferences.

IETF, Internet Engineering Task Force - The IETF is an open international organization formed by researchers and experts, operators and vendors in the area of network engineering. Its mission is to produce protocol standards, best current practices and informational documents to improve the design, use and management of the Internet. The IETF is the body that defines standard Internet operating protocols such as TCP/IP.

IPMP, Intellectual Property Management and Protection - Part 4 of the MPEG-21 standard provides the means of expressing protection and governance (protection tools, rights expressions, digital signatures) over a specific part of a Digital Item hierarchy to be governed and/or protected.

IntServ, Integrated Services - IntServ is one of the architectural networking models defined by the IETF to provide end-to-end (E2E) QoS guarantees. It envisages the reservation of resources in each element network and domain participating in the establishment of the end-to-end network path. The reservation of resources is done using the protocol RSVP. IntServ can offer quality of service guarantees to individual flows by opposition to DiffServ (the other architectural model for E2E QoS services specified by IETF), which aggregates services with similar requirements into service classes.

ISO, International Standards Organisation - ISO is a non-governmental organization, constituting a network of the national standard institutes of 155 countries. ISO has its Central Secretariat in Geneva. The areas of intervention of ISO range from standards for traditional activities (eg. agriculture or construction), through mechanical engineering, medical devices, to the newest information technology developments, such as the digital coding of audio-visual signals for multimedia applications.

MANETs, Mobile Ad hoc Networks - Self-configuring wireless network composed of mobile routers and hosts. MANETs have arbitrary topology, as their mobile components can move randomly, joining or leaving the network in an arbitrary form. MANETs provide a viable alternative to access shared data where no fixed network infrastructure is available.

MHP, Multimedia Home Platform - MHP defines a middleware, comprising protocols, APIs and languages to allow interactive TV applications to be played in different TV set top boxes. It promotes the interconnection of peripherals and multimedia equipment via the in-home digital network.

MLE, Maximum Likelihood Estimation - MLE is a method used in Probability and Statistics, to determine, or estimate, the parameters of a probabilistic model that maximize the probability, or *likelihood*, of the sample data.

MPEG, Motion Picture Expert Group - Working group of ISO/IEC responsible for the development of video and audio encoding standards.

MPLS, Multi Protocol Label Switching - MPLS is a technology being specified by the IETF to enable the implementation of label switching on top of different packet-based link-level technologies. It includes procedures and protocols for the distribution of labels between routers and encapsulation. MPLS aims at providing an efficient solution to meet the requirements of next-generation Internet protocol (IP)based backbone networks, addressing aspects of QoS, scalability and routing.

Multicast - Multicast is a method to distribute the same information to a group of remotely located persons through the Internet. Instead of transmitting individually to each of those users a copy of the data (unicast transmission), it sends only once the data to a special address. One common situation in which it is used is when distributing real time audio and video to a set of hosts participating in a distributed A/V conference.

Multiplex - In digital TV broadcasting, a multiplex is a stream of all the digital data carrying one or more services within a single physical channel.

NIC, Network Interface Card - Hardware that allows computers and other end-user equipment to communicate through a network.

NM, Network Manager - A subsystem integrating the content mediation platform, responsible for the communication with the network-specific systems and with all tasks

related to networking aspects. NMs of adjacent Autonomous Systems exchange messages between them to request the allocation of resources and fulfilment of commitments agreed in cSLSs and pSLSs with the goal of providing QoS across distinct networks, environments and applications.

NO/NP, Network Operator/Provider - A business entity that offers networking services to their customers.

NVoD, Near Video on Demand - NVoD is the term used to designate a service offered to end users, allowing them to request and start watching a movie located in a remote video server, but only at specific times of the day, as scheduled by the provider of that NVoD service. Please see also the definition of VoD.

OTA, Over the Air - OTA is a term used to designate A/V services that are broadcasted using radio-frequency terrestrial channels.

OWL, Web Ontology Language - OWL is a W3C specification for creating and representing ontologies. OWL builds on RDF (Resource Description Framework), adding more vocabulary for describing properties and classes.

PDA, Personal Digital Assistant - PDA is a handheld (portable) consumer electronics device that provides scheduling and basic computing functionality.

PDF, probability distribution function - It provides a measure of the probability to every interval of the real numbers space.

Plug-ins - Plug-ins are software components that enable to extend the core functionality of applications or systems. They are typically developed at a latter stage, when new requirements are identified. They are then attached or integrated with the initial system to perform specific functionality needed to support the newly identified requirements.

POTS, Plain Old Telephone System - POTS is the term used to designate the traditional dial-up telephone system.

PPV, Pay Per View - PPV is one of the business models adopted within the digital television industry. This modality does not require the user to subscribe a monthly fee service. Rather, the user pays only the content or service he/she desires to watch.

PQoS, Perceived Quality of Service - PQoS refers to the quality of service as experienced by the end-user.

PSNR, Peak Signal to Noise Ratio - The PSNR is a value that provides the ratio between the maximum power of a signal and the power of the noise corrupting that signal. The PSNR is often used in video or image compression. It yields a quality measurement between the original and a compressed image. The higher the PSNR, the better the quality of the compressed image. The PSNR is usually expressed in a logarithmic decibel scale and is computed using the Mean Square Error (MSE) between the original image and the reconstructed one (from a compressed version of the original). The following are the mathematical expressions used to compute the MSE and PSNR:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-1} ||I(i, j) - K(i, j)||^2 \quad (0.1)$$

$$PSNR = 10 \cdot \log_{10} \frac{MAX_I^2}{MSE} \quad (0.2)$$

PVR, Personal Video Recorder - A PVR is a consumer equipment that allows users to store and playback video to and from a storage medium, as for example, a hard disk. While playing back the video, the user will be able to do operations such as pause, stop, start, fast-forward or fast-rewind.

QoS, Quality of Service - At the network level, QoS refers to the capability of a network to provide better service to selected network traffic over various technologies, which requires the provision of dedicated bandwidth, controlled jitter and latency and improved loss characteristics. When stated at the user level, it refers to the quality experienced by the user when consuming the content.

RDF, Resource Description Framework - RDF is a language developed for representing information about resources in the World Wide Web. It is particularly intended for representing metadata about Web resources, such as the title, author and modification date of a Web page, in *RDF Primer* [59].

RFC, Request For Comments - RFC is the term used in the IETF to designate the working documents that are used for the development of IETF specifications.

RSVP, ReSerVation Protocol - The RSVP is a signalling protocol used by network routers operating in distinct domains to exchange information concerning the reservation of network resources to meet the requirements of a QoS-enabled end-to-end connection as defined by the IntServ model. Version 1 of RSVP is specified in IETF RFC 2205.

R&D, Research and Development - R&D is the abbreviation used to refer to research work.

QM/P, Quality Meters/Probes - Device capable of extracting measures relative to the quality of a stream, in any point of its delivery path.

SDTV, Standard Definition TV - SDTV refers to the transmission of digital TV signals having a spatial and temporal resolution comparable to those of regular TV broadcasting systems (PAL, NTSC, SECAM). Aspect ratio of SDTV is 4:3 and spatial resolution is usually 704 pixels per 480 lines.

SLA, Service Level Agreement - An SLA is a formal contract between a Service Provider and a client, guaranteeing quantifiable network performance at defined levels. SLAs generally include the steps that should be performed by the Service Provider and the client on the event of failure. The SLA stipulates that the Service Provider guarantees that the agreed service will be available for a certain percentage of time and grants the client rights and remedies if the provider fails to meet the defined performance levels over the specified period.

SLS, Service Level Specification - An SLS constitutes the technical part of the SLA, i.e., the SLS contains the specification of the technical provisioning aspects of the

service defined in the SLA. The SLS is an integral part of the SLA. cSLS and cSLA are established between clients (end-users) and Service Providers, whereas a pSLS is established between two business entities.

SMG, Statistical Multiplexing Gain - In the context of TV broadcasting, statistical multiplexing allows to insert more TV channels in a radio channel having a given bandwidth which is smaller than the sum of the peak bit rates of individual TV programmes being multiplexed. SMG provides a measure of the differential between the sum of the individual peak bit rates and the channel's BW.

SMPTE, Society of Motion Picture and Television Engineers - The SMPTE is a technical society founded in 1916 with the mission of contributing to the development of the motion imaging industry. SMPTE is presently regarded as the most important body developing standards for the Television and Motion Picture industries. It carries out its goals by developing and publishing Industry Standards, Recommended Practices and Engineering Guidelines. It also promotes networking and dissemination of knowledge through its periodical scientific journal (the *SMPTE Motion Imaging Journal*) and by organising several industry-driven and scientific conferences.

SNR, Signal to Noise Ratio - SNR provides a measure of the degree of importance or visibility/ audibility of the noise in respect to the useful signal in audiovisual signals. It is defined as the ratio of a signal power to the noise power present in the signal. A high ratio is an indication of a signal with reduced relative noise level.

SOAP, Simple Object Access Protocol - SOAP is an XML and HTTP based protocol that enables the exchange of messages between software programmes running in different machines with different operating systems. It thus promote the interoperability in cross-platform programming.

SOA, Service Oriented Architecture - The term SOA is used to designate software systems that offer their functionality to external entities (usually other systems or applications), in the form of a set of services with standing, well-defined interfaces. SOA-based systems are seen as a collection of functionalities, which are implemented by

loosely coupled services. SOA-based systems are re-usable, in the sense that it should be possible to obtain new systems, from the viewpoint of delivered functionality, by differently combining their services.

SP, Service Provider - An SP is an entity that takes direct or indirect responsibility for business and operational aspects and builds a complete solution for the delivery of context-aware multimedia services.

TD, Terminal Device - An end-user multimedia-enabled device.

TDM, Terminal Device Manager - The subsystem responsible for the communication with the terminal device.

TVA, TV Anytime - The TVA forum is an association of entities working in the business of audiovisual applications with the goal of defining and promoting open specifications to enhance audiovisual interactive applications using local storage of consumer platforms. The forum has issued a metadata scheme, known as TVA, to be used in interactive digital TV applications that assist the end-user to navigate through and interact with the application while promoting interoperability across consumer platforms.

UDDI, Universal Description, Discovery, and Integration - UDDI is an XML-based registry specification that enables businesses and services worldwide to list themselves on the Internet. It uses W3C and IETF standards such as XML, HTTP, SOAP and Domain Name System (DNS) protocols.

UMA, Universal Multimedia Access - UMA is a concept by which any user equipped with any kind of multimedia-enabled device, using any type of network connection, is able to access and consume any kind of multimedia content at any time and anywhere.

UMTS, Universal Mobile Telecommunications System - Third generation (3G) cellular system standardised by ETSI and 3GPP organisations.

URI, Uniform Resource Identifier - URIs, are short strings that allow the identification of any kind of resources in the Web. URN (Uniform Resource Name) and URL (Uniform Resource Locator) are forms of implementing URIs. URLs are commonly used in Web terminology to indicate the addresses of Web sites and pages. Any form of implementing URIs have a well specified syntax and semantics as to provide a formal identification method. The IETF has published RFCs that provide those specifications.

UWB, Ultra Wide Bandwidth - UWB is a short-range radio technology that complements other longer-range radio technologies such as Wi-Fi, WiMAX, and UMTS. It is used to relay data from a host device to other devices located within a maximum of 10 meters distance. It is a technology suitable for high-capacity wireless personal area networks.

VBR, Variable Bit Rate - A VBR video bit stream is an encoded bit stream presenting a fairly constant quality along time, by using a variable number of bits per unit of time. The amount of bits used depends on the complexity or difficulty of the scene being encoded.

VoD, Video on Demand - VoD is an interactive application that allows end-users to select and consume at their will, upon request, a certain audiovisual content. Usually the A/V content is a protected movie, which means that the service is paid. Different modalities, from the business point of view, can be implemented for payment purposes (monthly subscription, PPV, etc).

XML, eXtensible Markup Language - XML is a text based language developed by the W3C with the objective of facilitating the publishing of electronic resources. XML is currently widely used to exchange data on the Web.

XSLT, eXtended Stylesheet Language Transformations - XSLT is a language specified by the W3C, to allow transforming one XML document into another XML document. XSLT is designed for use as part of XSL, which is a stylesheet language for XML

W3C, World Wide Web Consortium - The W3C is the most active and well-known standardisation body developing specifications for the use of the World Wide Web (WWW) resources. The W3C is responsible for the development of widely used standards on the Web such as HTTP, HTML or XML.

Web - The Web is the short term commonly used to designate the World Wide Web.

WiFi - Wi-Fi is a wireless local networking technology based on the IEEE 802.11 standards.

WiMAX - WiMAX is a broadband wireless technology based on IEEE 802.16 and ETSI HiperMAN standards. It enables the delivery of last mile wireless broadband access as an alternative to wired broadband like cable and DSL (Digital Subscriber Line).

WWW, World Wide Web - The WWW is the designation given to the infrastructure composed by all resources (content, computing, networking) available on the Internet.

Contents

List of Figures	xxi
List of Tables	xxiii
1 Introduction and Motivation	1
1.1 Introduction	1
1.2 Overview of the problem and motivation	3
1.3 Major contributions of this thesis and proposed solution	8
1.4 Structure of this thesis	12
2 Background Information	14
2.1 Digital Video Technologies	15
2.2 Distribution technologies	33
2.3 Metadata schemes	44
3 Review of the State of the Art	78
3.1 Context awareness and Content adaptation	79
3.2 Statistical Multiplexing	161
4 The why and how of Context-aware support	173
4.1 Concise statement of the problem	173
4.2 Justification	175
4.3 Discussion	177
5 Work developed and proposed solution	180
5.1 Concept	180
5.2 Relevant Usage Scenarios and Requirements	184
5.3 Platform functional architecture	197

5.4	The implementation of the EIMS	264
5.5	Context awareness in the EIMS	292
6	Validation of Results	297
7	Conclusions	306
7.1	Conclusion	306
7.2	Summary of Contributions	310
7.3	Future Research	312
	Bibliography	314
A	Related work previously developed	340
A.1	Statistical Multiplexing studies	340

List of Figures

1.1	The heterogeneity scenario	5
1.2	Open loop statistical multiplexing approach	7
1.3	The conceptual architecture of the content mediator platform	9
1.4	The use of the mediator reference architecture	10
2.1	OMG CORBA Reference Model Architecture	37
2.2	Invocation of a local object in CORBA	39
2.3	Web Services Conceptual Architecture	42
2.4	A simple example of metadata	47
2.5	Different description records for a resource showing the canopy over the main altar in St. Peter's Basilica made by Bernini	50
2.6	A simplified view of the structure of a partition of the MXF container	59
2.7	The large number of services received by the end-user in digital TV . .	63
2.8	Example of a bouquet in DVB	63
2.9	The METS structure	69
2.10	The cross-application role of DCMI	71
3.1	The role of context-aware content mediation platforms	80
3.2	Concept of a generic functional architecture	90
3.3	The need for higher level contexts	93
3.4	The SECAS functional architecture	99
3.5	Simplified representation of the CoGITO middleware	101
3.6	The use of CoBrA for context-aware applications	102
3.7	The functional blocks of the Context Toolkit	103
3.8	Components of the SOCAM architecture	104
3.9	Example of an MPEG-21 Digital Item	117
3.10	A graphical representation of the DID structure	127
3.11	Tools and concept of MPEG-21 DIA	128

3.12	Example of an excerpt of a Terminal UED	134
3.13	Example of an excerpt of a Network UED	135
3.14	Example of an AQoS metadata	136
3.15	Example of UCD metadata	137
3.16	Editing-like content adaptation using BSDL and gBS	138
3.17	The three different formats for RDF representation	145
3.18	Example of a CC/PP	157
5.1	The layered architecture of ENTHRONE	203
5.2	Actors interacting with the EIMS	207
5.3	Interoperation with heterogeneous Content Providers	210
5.4	Use Case Model of the EIMS	214
5.5	The evolution of the EIMS architecture from its original concept	242
5.6	The UML high-level block diagram of the EIMS architecture	243
5.7	The role of the NMs in an end-to-end scope	250
5.8	The role of the TDM	251
5.9	Collaboration Diagram for Session Initialization	254
5.10	Collaboration Diagram for DID Browsing/Searching	255
5.11	Collaboration Diagram for <i>Select DI</i>	256
5.12	Collaboration Diagram for <i>License Acquisition</i>	258
5.13	Collaboration Diagram for <i>Detect QoS fault</i>	260
5.14	Collaboration Diagram for <i>Acknowledge QoS fault</i>	261
5.15	Collaboration Diagram for <i>Recover from QoS fault request</i>	262
5.16	Collaboration Diagram for the <i>Upload DID</i> use case	263
5.17	<i>Dispatcher</i> subsystems and interfaces	268
5.18	<i>Service Manager</i> subsystems and interfaces	270
5.19	<i>Customer Service Manager</i> interfaces and main classes	272
5.20	Service-level to network-level conversion	277
5.21	<i>ADE</i> interfaces and subsystems	278
5.22	<i>StatSrv</i> interfaces and subsystems	282
5.23	<i>EIMS-CM@CP</i> interfaces and subsystems	287
5.24	<i>EIMS-CM@SP</i> interfaces and subsystems	288
5.25	<i>EIMS-TDM</i> interfaces and subsystems	291
6.1	Laboratory demonstration setup	300
6.2	Context providers and gathering of context	301
6.3	DI service response time	303
6.4	uploadDID service response time	304
6.5	Stats service execution time	305

List of Tables

2.1	The MPEG-2 TS Program Specific Information (PSI) tables	64
2.2	The DVB-SI tables	65
2.3	The DVB-SI tables (cont.)	66
2.4	The DCMI Element set	72
2.5	The DCMI Element set (cont.)	73
3.1	Types of content adaptation	111
3.2	Defining ontology elements in RDFS and OWL	172
5.1	Service Parameters for the "VoD" service	187
5.2	Service Parameters for the "Media File Downloading" Service	194
5.3	Brief Description of the EIMS major use cases	213
5.4	General description of the <i>Retrieve DI</i> use case	216
5.5	Main section of the <i>Retrieve DI</i> use case event flow	217
5.6	General description of the <i>Recover from QoS fault</i> use case	220
5.7	Main section of the <i>Recover from QoS fault</i> use case event flow	221
5.8	Main section of the <i>Recover from QoS fault</i> use case (cont.)	222
5.9	Main section of the <i>Recover from QoS fault</i> use case (cont.)	223
5.10	General description of the <i>Upload DID</i> use case	227
5.11	Main section of the <i>Upload DI</i> use case event flow	228
5.12	General description of the <i>Manage DID</i> use case	231
5.13	Main section of the <i>Manage DI</i> use case event flow	231
5.14	General description of the <i>Manage Service Provider</i> use case	233
5.15	Main section of the <i>Manage SP</i> use case event flow	234
5.16	General description of the <i>Manage Network Operator</i> use case	237
5.17	Main section of the <i>Manage NO</i> use case event flow	238
5.18	General description of the <i>Manage Content Provider</i> use case	240
5.19	Main section of the <i>Manage CP</i> use case event flow	241

5.20 Mapping contextual information to MPEG-21 DIA descriptors	296
A.1 Categories of TV sources	347

Chapter 1

Introduction and Motivation

1.1 Introduction

This section provides a brief overview of the work conducted within the context of this thesis, highlighting the main problems addressed and the contributions delivered.

It tries to guide the reader through this thesis, explaining the evolution it suffered since its conception until its finalisation. The work undertaken during the course of the years this thesis has endured, hasn't essentially diverged from its main original generic goal. Nevertheless its scientific and technical focus has moved away from the original intents to cope with the fast evolution that has occurred in the last years in the television and multimedia communications scenario.

The main generic goal of this thesis is to design and develop efficient techniques and approaches to support the delivery of quality and context-aware rich multimedia services. Initially the main area of application was centered around digital TV broadcasting services. The goal was to study forms of using as efficiently as possible the aggregated channel bandwidth trying to maximise the overall quality of

the delivered services and enabling as much diversity as possible taking advantage of new emergent multimedia formats.

As technology has evolved, we have witnessed a strong integration of digital television with Internet-like multimedia applications. On one hand, rich multimedia content typically found in Internet applications started to be incorporated or linked to digital television services. On the other hand, digital TV also moved away from its traditional channels into the IP world. It is now possible to watch TV, or any multimedia content for that matter, on a number of different terminal devices, either fixed or portable, and presenting quite diversified physical characteristics and processing power.

Contemporaneously, advances in digital transmission and networking technology have and will continue to offer increased throughput, functionality and performance, notably concerning quality assurances, moving away from the only best effort services to premium IP services. Recent wireless technologies such as Wi-Fi (based on IEEE 802.11 specifications), WiMAX (Worldwide Interoperability for Microwave Access) and more recently UWB (Ultra Wide Band), as well as mobile networking concepts such as seamless mobility and MANETs (Mobile Ad hoc Networks) or generally ad-hoc networks for personal communications are opening the doors for new multimedia applications where customisation assumes a vital importance. In addition the great progress observed in compression techniques of audiovisual content has enabled the consumption of multimedia content within a variety of new appealing applications on diversified multimedia-enabled devices. It also made possible to achieve higher levels of quality using the same bit rates or to spend much less bits to provide comparable quality. This has raised the expectations of users regarding quality and diversity of content.

Whilst this progress foresees the steady and continuing increase of trans-

mission bandwidth, history tells us that the more capacity there is, the more the demand exists. This is also true regarding hard disk capacity, the more we have on our machines, the more we use and need. Nonetheless, it seems to be consensual that the optimisation of channel resources has lost some of its relevance in favour of other aspects directly related with the diversification and customisation. Still, service providers that will be able to make a better use of their transmission capacity will be in a better position to meet the user's expectations. Public demand will continue to evolve towards more diversity, more quality and customisation, thus consuming all the available channel capacity.

In light of this evolution scenario, the objectives of the work were re-aligned, having acquired a broader scope that can be summarised as *the architectural support for the provision of context- and QoS-aware adaptable multimedia services*.

1.2 Overview of the problem and motivation

The main goal of this thesis was to develop the architectural support to enable the provision of high-quality and larger-choice multimedia services to the end-users, being able to react and adapt to different contexts of usage and varying conditions of that context. Accordingly, the work developed has focused on the following aspects:

- context-awareness;
- description schemes
- content adaptation enabling systems in distributed environments;
- provision of high-quality and larger-choice multimedia services to the end-users;

As technology continues to evolve at an accelerating rate, different multimedia-enabled devices aiming at the same field of application, but presenting diverse capabilities, are continuously being introduced in the market. To this reality adds the large volumes of multimedia content available on-line, presenting a multiplicity of formats and the ever growing acceptance of the network-centric paradigm from the general public. Increasingly more consumers, equipped with different devices and having diverse interests and preferences, *navigate* in the Internet or get connected to some network looking for different types of multimedia information. An access to content that meets user's expectations and demands must take into account all the different aspects of this heterogeneous scenario. A thorough characterization of both the content as well as of the context of usage is therefore needed to enable the selection and delivery of the content in the most adequate conditions, given the current context of usage. This includes content format diversity, terminal capabilities, network characteristics, user preferences and user surrounding environment. The characterization of all these factors requires the use of metadata. Currently, there is no universally accepted solution, which results in the existence of multiple formats (standard and proprietary) that are (partially) overlapping or that aim to cover distinct aspects. To achieve interoperability and transparent access to media content it is necessary to find a solution where multiple formats and models may coexist and where the information may be consistently used throughout the delivery chain, thus contributing to the enhancement of the users experience.

Providing QoS-aware ubiquitous access to multimedia content regardless of coding formats, network resources availability or user terminal capabilities and preferences, is still a challenge. This concept can be translated in the provision of context-aware multimedia services resulting in universal access to multimedia content.

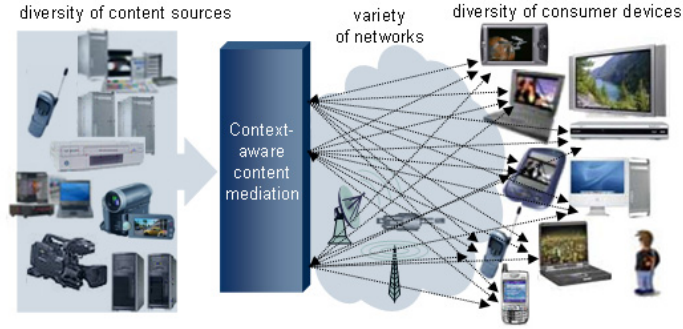


Figure 1.1. The heterogeneity scenario

Figure 1.1 presents this concept by illustrating the scenario of heterogeneity at the user, at the network and the content generation sides and the role of context-aware content mediation and distribution systems. Metadata is used to describe a-priori or at any instant the complete context where the content is to be or being consumed and the characteristics of the content itself. By context it is meant the capabilities of the user terminals, the characteristics and conditions of networks, the capabilities of content generation and conversion engines, the natural conditions of the user surrounding environment, the preferences of the user and the quality of the content as perceived by the user. Being in possession of this information, systems mediating the access to content are empowered to decide how the content should be delivered or be made available to each user at any instant.

Distributed technologies are essential to facilitate the inter-operation of the system with heterogeneous user terminal devices, with different network technologies, with varied context description mechanisms and with multiple encoding formats and content repositories.

High-quality broadcast, networked and storage audio-visual applications based on the MPEG-2 standard have become widely popular. The television broadcasting and the DVD industries have embraced this audio-visual compression technology for some years now. Although more bandwidth efficient compression schemes such as MPEG-4 and H.264 have already started to be used in some application domains, it is expected that broadcasters will extend the use of MPEG-2 for still a considerable number of years.

To provide wider choice of content yet guaranteeing the standards of quality and innovation and still obtaining profitable business, it is crucial to find ways of using efficiently the available transmission, storage and coding resources. The adoption of new generation of standards such as H.264/AVC [101], [108], or JPEG2000 [89], [190]. Yet, all the TV broadcasting industry has adopted the MPEG-2 standard and, although the quality of the MPEG-2 encoded video streams depends on implementation details of the compression tools, it is not expected to see significant improvements with this technology. We propose therefore to go one step further in the transmission chain and try to introduce improvements in the multiplex. We propose to use statistical inferencial techniques supported on the Bayes theory, VBR encoding schemes and video sources classification, in order to be able to predict the behaviour of video sources and use in a more efficient way the available transmission bandwidth.

Accordingly, and on a second plane, this thesis also addresses the optimization of the use of resources required to distribute high-quality video sources. This second aspect relies upon a probabilistic framework supported on the use of statistical inference techniques and on the Bayes theory. It builds on the use of video compression algorithms to achieve necessary reduction ratios in the sources' bandwidth and on the knowledge about their statistical and behaviour characteristics.

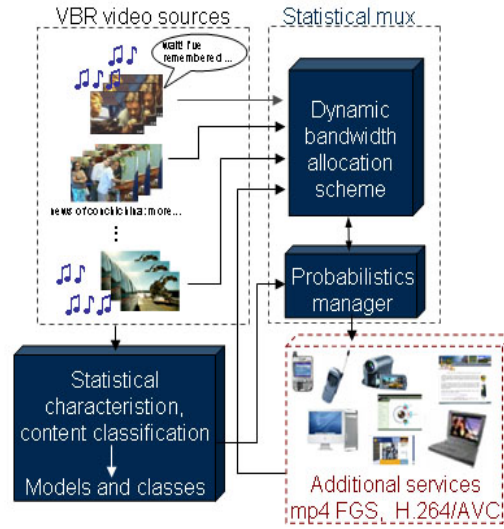


Figure 1.2. Open loop statistical multiplexing approach

It aims at finding optimal and innovative algorithms capable of making the best use of the available transmission bandwidth, while providing the best possible or required video quality. The generic application scenario is the combined transmission of several compressed good-quality video sources and other additional multimedia sources. The main idea, summarised in the diagram of figure 1.2, is to encode video sources in VBR mode and characterise them in terms of type of content and statistical behaviour.

Then, using an already established probabilistic framework listing a great number of statistical models identified against distinct characteristics of the video source and target quality, empower an intelligent multiplexer to maximize the available network bandwidth by predicting the probable amount of free space and filling it with extra services. Or using this released bandwidth to increase the number of bits spent to encode the most demanding sources, by giving back an indication of

available resources.

Two distinct application domains are envisaged: 1) television broadcasting, aiming the distribution of traditional TV programming together with added-value services such as news-on-demand, file downloading, VoD or corporate TV services, and also professional segment applications such as interconnection of television studios; 2) the Internet, targeting multimedia applications such as, media file downloading, VoD or NVoD, streaming of A/V clips, videoconference, etc. The type of networks that may support this kind of applications can be broadband wireless such as DVB-T/H, broadband wired networks such as DVB-C, ATM networks using fiber or cable, xDSL technologies using copper-wire infrastructures, satellite links using the DVB-S specification, etc.

1.3 Major contributions of this thesis and proposed solution

The main contribution of this thesis is the specification of a middleware architecture and associated services to enable the provision of context-aware adaptable multimedia services in heterogeneous environments. The work undertaken has enabled the design and development of a reference architecture in the form of a set of services and interfaces based on standard emergent technologies. This reference architecture, which concept is shown in figure 1.3, can be used in a number of different environments thus constituting an interoperable framework, suiting different requirements and goals. The development and implementation of this reference context-aware content mediation architecture took place within the framework of the European project ENTHRONE [31] and is designated as ENTHRONE Integrated Management Supervisor (EIMS). The EIMS is the content mediation platform con-

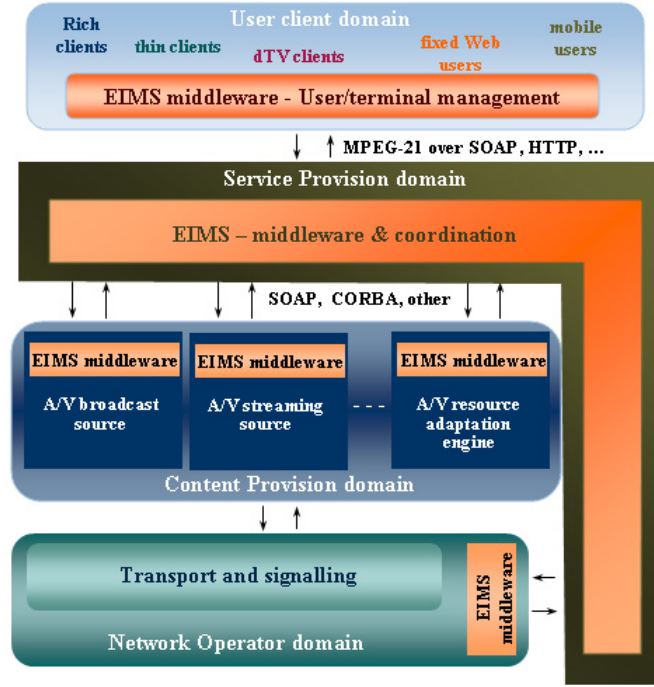


Figure 1.3. The conceptual architecture of the content mediator platform

ceived within the scope of this thesis to provide the architectural support for the QoS- and context-aware access to multimedia content.

Figure 1.4 illustrates its use within the multimedia content chain collaborating with existing participating entities. The diagram in figure 1.4 highlights the distributed nature of the content mediation platform with its deployment in and interaction with different key players in the multimedia value chain.

Whilst it is true that considerable amount of research has already been done in this area, since it involves a large number of topics, there hasn't yet been

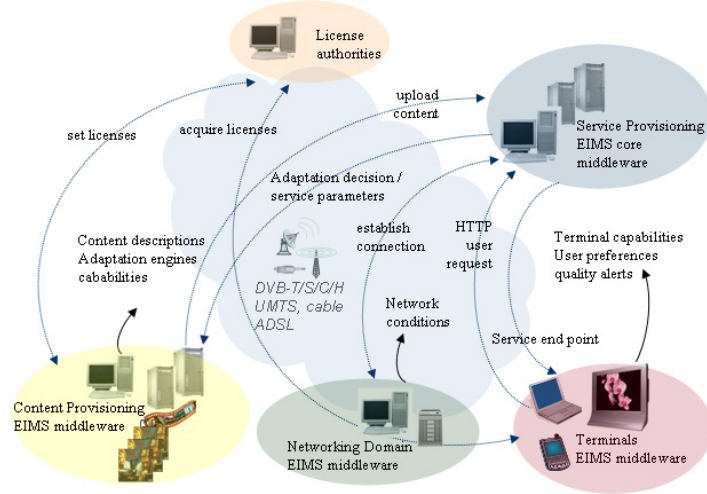


Figure 1.4. The use of the mediator reference architecture

proposed a complete open solution that addresses all the aspects of a quality and context aware provision of multimedia services. Important results have been achieved in the many different areas involved, without which the outcomes of this thesis would not have been possible. Nonetheless it is not a trivial task to put the different pieces working together towards common goals. Nor to define and develop a framework that enables to construct a modular *lego-approach* infrastructure where functionality can be added or removed as needed.

In parallel and relevant to this work, we have concentrated on the representation and use of context. In particular, the type and classification of contextual information has been studied and forms of representing it in a meaningful and useful way. For that purpose a thorough investigation was made concerning current work on context-awareness, including standard specifications for the representation of context and the use of ontologies. An ontology can be seen as a dictionary comprising a

set of concepts and statements describing those concepts and their relationships. It can be used to describe a domain of interest providing a view of real world situations occurring in that domain as a formal model that can be manipulated by machines. Ontologies provide the means to represent complex relationships between instances of the described concepts and support the use of rules expressing additional knowledge. In a given situation of consumption of multimedia content, the possibility of jointly using explicitly obtained contextual information, such as information collected by software modules running on the user terminal or in networking equipment, together with a high-level formal description of the domain, can enable the mediating system to perform better its role. It will empower that system with greater knowledge about the real world situation, putting it in a privilege position to take decisions regarding possible needed content adaptations and thus to better serve the interests of the user. Accordingly, we have investigated the use of ontologies as a form of knowledge representation formalism, based on which our system would be able to reason about extracted contextual information (low-level contexts) to automatically derive/infer higher level contexts.

In this thesis, as referred above, we have also been concerned with the efficient usage of network resources, notably in broadcast environments where the transmission medium is shared by a number of audiovisual services. This aspect also contributes to the provision of quality-aware services to the end-users, even if in particular scenarios. In this area, preliminary work was conducted towards the specification of a statistical framework based on inferential principles, notably the Bayes theory. The goal of this framework was to enable an intelligent multiplexer to distribute the total bandwidth channel among a number of video sources and to predict periods of time where the sum of the total bandwidth required to satisfy a minimum agreed level of quality across all sources would be less than the available

channel bandwidth. The unused bandwidth would then be used to transmit less demanding multimedia services based on new VBR (Variable Bit Rate) algorithms with high levels of granularity in the bit rate variation as is the case of the MPEG-4 FGS, *Fine Granular Scalability* algorithm. Whilst this line of work has not come to final results and conclusions, it has nevertheless formulated an innovative approach to the management of aggregated transmission bandwidth, enabling the transparent inclusion of additional multimedia services. It has laid down the foundations for further investigation work in this area.

1.4 Structure of this thesis

This thesis consists of seven chapters.

The present chapter introduces the problem addressed by the thesis and the proposed solutions, the performed studies and the undertaken research.

The second chapter briefly presents some background information on scientific fields of interest to this thesis but that are not directly addressed by its work. This is the case of video compression techniques, metadata schemes and distribution technologies.

The third chapter provides a description of the state-of-the-art concerning the topics more relevant to the fulfilment of this thesis' objectives and that were studied more deeply. Accordingly, it describes the recent developments achieved by the research community and the work done at the standardisation level on context awareness and multimedia content adaptation. Finally it also deals with the topic of probabilistic modeling for statistical multiplexing, which was the initial focus of this thesis and that it is relevant for an effective use of bandwidth resources.

The fourth chapter presents in a very clear and concise manner, the main problem addressed by the thesis, outlining its importance and demonstrating that it

still needs to be given an adequate answer.

The fifth chapter describes all the work developed by the proposer of this thesis, presenting a solution for the problem stated in chapter four.

The sixth chapter provides results obtained by performing both simulation with video sequences and carrying out laboratory tests with the developed algorithms and systems.

The seventh chapter resumes the main achievements and contributions of this thesis and provides directions for future work.

The last chapter contains all the references relevant to and used in this work.

Chapter 2

Background Information

This chapter provides a brief overview of topics and technologies that are essential to the successful achievement of this thesis' goals, but that were not the focus of the work. The information herein referred has been considered as background knowledge in the sense that the addressed topics have been studied and are used in this work, but haven't really been subjected to research.

Accordingly, this chapter describes pertinent aspects related with video compression techniques, metadata or description schemes and distribution technologies that are indeed essential for the deployment of context-aware networked audiovisual services. With respect to video compression, it is provided a summary of existing standard technologies, mostly those that are used within the traditional content distribution chains such as the case of the television industry, but also generically in multimedia network applications. The same applies to metadata, where an overview of the most well-established standardised specifications is provided. Nevertheless there are differences in the presentation of these two topics, as metadata is commonly divided into different classes and also because the development of the different standards and their current use is very much application-driven. And also

because metadata played a more important role in the studies and work developed during this thesis. It is unquestionable that nowadays, video encoding schemes are an essential part of any A/V or multimedia networked application, since they provide the source of the content. Within the context of this thesis they were used for that purpose, as in any other application, service or system. On the other hand, metadata has been more thoroughly studied and many different uses were made of metadata. In fact, metadata is intrinsically used in every aspect of our work, enabling the successful implementation of all functionality satisfying the identified requirements and proposed goals. Moreover, during the development of the work, it was necessary to make a number of decisions on how to use, combine and process metadata. This work has motivated the subsequent submission of technical contributions to the MPEG standardisation working group [35], [70], [132] and participation in special discussions groups created within that standardisation body.

Finally, concerning distribution technologies, it provides an overview of standards-based solutions concentrating on technologies that enable the implementation of a Service Oriented Architecture (SOA) approach.

2.1 Digital Video Technologies

2.1.1 Introduction

High-quality broadcast, networked and storage audio-visual applications based on the MPEG2 standard have become widely popular. The television broadcasting and the DVD industries have already embraced this audio-visual compression technology. Its use has now become familiar worldwide in a variety of environments, from custom consumer applications to professional TV broadcasting and video production. However, in spite of its popularity among the consumers, there is still much

to learn regarding the best way of using it and adjusting all the involved parameters in order to achieve the best compromise between quality, efficiency (profits) and content diversity.

Following the finalisation of the MPEG2 specification, further compression algorithms have emerged targeting common but also new segments of the audio visual scenario from multimedia Internet broadcasting to high definition TV and digital cinema. Such is the case of the MPEG-4 ISO/IEC standard [1] and the ITU-T H.264 specification [2]. They both present higher degree of flexibility in terms of bit rate and achieved quality and, in the case of MPEG-4, additional functionalities potentiating new forms of interaction to the end user with the video he is playing. This is essentially made possible by the fact that MPEG-4 is oriented to objects, unlike other algorithms that treat the video as a sequence of images, being the image the fundamental unit for processing. In MPEG-4, the image may be segmented into objects and different compression techniques applied to each, to yield better results. In addition the new generation of standards offer a number of varied profiles to suit different requirements of different applications and/or usage environments and even variability in time of those requirements. For example the H.264/SVC specification defines a scalable format that is extremely interesting to be used in environments where availability of network resources is not guaranteed, as is the case of cellular networks or more generally speaking of mobile environments. The JPEG2000 algorithm [89] is also a scalable format providing extremely high image quality. Its usage is being considered within the digital cinema industry. It provides the adequate functionality to be used within the different stages of the film industry from acquisition, through editing and post-production, to the distribution. The fundamental functionality and characteristics of the algorithms here referred will be described ahead in this section.

Video compression, and in particular the MPEG specifications, is indeed the key technology for the fast deployment of high-quality audio-visual services. Another very important factor for the success of these services is the availability of broadband networks capable of providing and guaranteeing a requested quality of service.

The future of audio-visual communication, namely digital television multimedia broadcasting, can be seen as the emergence of a new medium where the key components will be new interactive and participatory content, more diversity, more choice to the viewer and the ability to dynamically and individually compose each viewer's programming (structure the programme schedule at the viewer's will). And above all, more quality.

To provide a wider choice of content yet elevating the standards of quality and innovation and still guaranteeing profitable business, it is crucial to find ways of using efficiently the available transmission, storage and coding resources.

The new digital compression and signal processing algorithms have already opened the way, and still continue to do so, to the creation and manipulation of these new contents.

Very important aspects that will support the deployment and spread among the public of these new multimedia services have been recently investigated and are now being experimented in field-trials. Important R&D funding is still being directed to this area. Special attention is being given to the characteristics of the service as perceived by the end-user: reliability, high-degree of interactivity and intelligence, inter-operability, scalability and perceptual quality. The requirements at the end-user equipment necessary to exploit at full extent these characteristics are being studied and gradually introduced. The aspect of creation of new content to be used by advanced user interfaces is also being addressed: new authoring tools and platforms to develop interactive, adaptive, intelligent and creative services is another field of

research.

One of the key aspects of service characteristics has always been and will continue to be its quality. Without picture and sound quality all the other attributes will not have the strength to stand up for themselves. Another emergent key aspect is customisation. This includes both the ability to personalise the content and the services to the users's preferences and requirements as well to adapt it to suit characteristics and constraints of the environment where the user is consuming the services. Systems presenting this capability are usually designated as *context-aware* systems. By using contextual information, those systems are able to automatically adapt the content and/or the service so that it may be presented to the user in the best possible form according to the terminal the user is equipped with, the conditions of his/her network connection, the characteristics of his/her natural surrounding environment and also the user's preferences and special needs, thus contributing to enhance his/her quality of experience. Definition of context and considerations on how to use contextual information will be addressed further ahead in this thesis. To become a winner, it is clear that services and applications will have to give the best possible coverage to all the mentioned aspects.

It is therefore necessary to learn how to use as efficiently as possible the available resources, either bandwidth or compression tools. This is the third and probably basic aspect for the provision of high-quality networked audio-visual services. Intelligent management of both terminal and network resources, end-to-end perceptual quality support, intelligent gateways providing seamless connections between different networks and intelligent (trans)coding are fundamental issues to be addressed in order to achieve the desired goal.

We therefore argue that there are essentially three key aspects to be explored in order to deploy a successful TV or multimedia service:

- more choice
- more quality
- efficient use of resources

These essential aspects should be considered when designing a dTV or multimedia broadcasting system. All three are interrelated as they all contribute in some way to the same objective, which is the ability to enlarging the number of quality, offered services. More quality means better businesses' opportunities for the broadcaster and thus the (business) opportunity to enlarge their range of services; efficient use of resources will eventually make free channel bandwidth that could be used by a new service, thereby increasing the broadcaster's offered choice, or re-used by an existing service thus making possible to increase its quality.

2.1.2 The compression of video

Video data is essentially a sequence of pictures in time, with repetition rates of usually 24, 25 or 30 images per second. But it can even be more in the case of professional applications such as acquisition and post-production in TV environments or in the digital cinema industry. Each picture or frame is composed of a number of picture elements along the horizontal axis, forming lines of pixels, and a number of such lines along the vertical axis. The number of pixels in each line and the number of lines per picture may vary from application to application. In standard television systems, the normal spatial dimensions of a picture are 864 pixels¹ per 625 lines (but not all pixels are visible. It is usual to consider an active image area of 720 pels x 625 lines). The picture elements (pixels or pels) carry the value of brightness

¹the number 864 is obtained by considering a line period of 64×10^{-6} s and a sampling frequency of 13,5 MHz

and colour of the picture corresponding to the spatial location they appear. In the digital domain, pixels are represented by a digital word with a given length or number of bits. Typically the word length is 8 bits. But again, it may be more (10 or 12 bits long), notably in professional environments as mentioned before. As such, a picture can be seen as a matrix of $N \times M$ elements, each one with a D depth, and a video as a set of these matrices. When digital video is to be transmitted, each matrix is read or scanned from its first position until the last one, generating a sequence of serialised bits. Because the frame repetition rate is usually fixed (in Europe, television systems use 25 Hz or images per second, which corresponds to a picture period of 40 ms) the process of reading each matrix should take always the same amount of time. This means that, the larger the matrices, the more close in time bits will occur in the transmission channel, thus generating higher bit rates which require larger transmission bandwidth. Or, the larger the matrices the higher amount of storage space will be needed to store the video. Given the presented figures, it is easy to see that the transmission of digital video with spatial, temporal and quality characteristics normally found in the digital TV domain, will result in a large amount of data sent to the channel or stored in video servers. Hence the need to apply compression.

The importance of content and subjective quality

Compressed video is no more than a sequence of bits occurring more or less frequently depending on the target picture quality and on the difficulty of encoding the scenes (which in turn directly depends on the redundancy of images and scenes, mouvement and detail). Within the compression algorithm, quality greatly depends on the size of the quantization step in use (among other factors such as the accuracy of the motion estimation algorithm or the approach used to make the decisions

regarding the encoding mode to adopt). Likewise, compression gain strongly relies on the quantization process. It is in fact at this stage that occurs the major contribution to the reduction in bit rate. Therefore, it is here that work needs to be conducted to obtain a good balance between quality and bit rate.

Although a quite close and linear relationship can be found between the subjective quality of images and the objective metrics of picture quality, the PSNR (Peak Signal-to-Noise Ratio), this metric not always reflects accurately the degree of satisfaction of viewers. In fact, subjective quality, as the quality of the image perceived by the viewer, is frequently not confirmed by high values of PSNR. Factors that have great impact on the overall video quality are related to the content, spatial detail and motion and to the relative importance of each region of the image, of each image within a scene and of each scene within the sequence. Different and varying contents exhibit different needs during encoding. Some regions or pictures may prove to be more difficult to encode than others. Some may be more important for the overall video quality. Those regions and pictures are then critical for obtaining good subjective quality. They should therefore be allowed to consume more bits than the others.

These facts justify for themselves the need to:

- accurately study the relationship between type of content, number of bits spent and impact on the overall subjective quality;
- to collect statistical data from a representative collection set of video streams;
- to subjectively evaluate and classify those sequences in terms of type of content, activity, detail and picture quality;
- to cross the two types of information and obtain correlations between them;

- and finally, to build probabilistic models with appropriate and adjustable parameters, capable of accurately predict the evolution of video sources according to the initial characteristics of each scene.

This characterization can be performed at the scene and image levels. Clearly the first time scale has substantial advantages regarding the second one. The decisions made regarding the allocation of bandwidth will be valid for a longer period thus reducing the burden put on the resource allocation and monitoring mechanisms. A multiplexer responsible for the share of a certain channel bandwidth among a collection of video sources, if in possession of their characterization and/or parameters of the best matching probabilistic models, would have the ability to efficiently manage the existing resources and predict in advance variable requirements.

This is the goal of the work addressed in this thesis concerning statistical multiplexing:

- to study the behaviour of encoded video sources according to their content and subjective quality;
- to analyse existing correlations between number of bits spent and video content and quality;
- to build shared databases storing the extracted statistical information;
- to devise a generic architecture of a multiplexer that is capable of using that information and efficiently and dynamically share common channel resources.

But in fact these aspects are only means to achieve the ultimate goal: increased quality and efficiency in multiple digital video transmission.

2.1.3 VBR versus CBR encoding for transmission

The video compression algorithm specified in the MPEG2 standard allows economies in bandwidth while providing a high-quality service. While compression is not a completely reverse process, the recovered signal follows (in most cases, closely) the original material with more or less degradation, depending on the content of the pictures and on the number of bits used to encode the pictures.

The number of bits necessary to represent the video sequences is substantially reduced by exploiting temporal and spatial redundancy of moving images. Correlation between areas in the pictures and between successive pictures depends highly on the content of those images itself. Therefore the very same compression scheme applied to a varying content sequence, will either consume a variable number of bits maintaining a constant quality level, or will generate a constant bit rate flow providing different qualities along the duration of the sequence. Or, as a third alternative, will always consume the same number of bits, probably in excess for the majority of time, anticipating the needs of the most difficult scenes to encode - those with greater activity and higher spatial detail.

Using the MPEG2 encoding algorithm it is thus possible to follow one of the three following approaches, regarding transmission bit rate:

- constant bit rate, where the number of bits spent in each GOP is controlled to a mean value. This approach leads to varying picture quality or distortion;
- variable bit rate controlling the distortion. This approach is able to produce video sequences exhibiting constant quality level throughout its duration;
- constant bit rate using always the highest bit rate needed to satisfy at all times a minimum level of distortion;

In digital television broadcasting it is usually used a constant bit rate encoding scheme, setting the mean bit rate to a value where the distortion of the most difficult to encode scenes is only slightly perceived. Because these type of scenes have typically a rather small frequency of occurrence, this obviously means that most of the time, bandwidth is being wasted. Less difficult or easy to encode scenes are over-dimensioned.

Variable bit rate encoding can be seen as a distortion (or picture quality) controlled scheme, whereas constant bit rate is a source controlled scheme. VBR is also known as open-loop encoding and CBR as closed-loop feedback encoding. Parameters or aspects such as, bit rate, level of difficulty-to-encode (or alternatively type of content) and bit rate control algorithm, condition or interfere on the picture quality or alternatively are controlled by the distortion. The relationships are not linear and often is difficult to obtain the best compromise in closed loop encoding.

Besides the content of video sources, also the type of compression algorithm itself contributes to the inherently bursty nature of the compressed flow. The MPEG2 video compression algorithm encodes video sequences frame-by-frame adopting different encoding strategies according to a determined pattern. It generates sequences of intra (**I**), predicted (**P**) and bi-directional (**B**) compressed frames, whose frequency of occurrence is dictated by a compromise between compression efficiency, error robustness and random access. **B** frames usually consume less bits because they explore existing backwards and forwards temporal redundancy - one frame before and one frame after their own occurrence - in addition to spatial correlation. They are the ones therefore responsible for the major share of compression obtained. **I** frames are encoded without reference to any other frame, exploiting only redundancy existent in the frame itself. **I** frames are usually the most bit-consuming ones. Their role is to serve as anchors to re-synchronize the decoding process in the event of errors,

which otherwise would propagate indefinitely, and to provide entry points into the bit stream. **P** frames explore backwards and spatial redundancy. They lie in between **B** and **I** frames regarding compression efficiency and error robustness. This temporal encoding scheme thus imposes a bursty nature to the bit stream at the frame level. The variations in time of activity and detail within the same scene and the diversity of content between scenes are responsible for the burstiness at larger temporal scales - intra and inter scene variations.

The second alternative, constant bit rate transport, is the most commonly used because usually transmission channels, especially broadcast, are of fixed bandwidth. Also and not less important, because it greatly reduces the complexity of the control admission procedures and facilitates the operation and management of networks. CBR can be seen as a source rate control where the encoding parameters are adjusted to generate streams at a pre-determined rate. This usually leads to inconsistent picture quality, where images or scenes with more activity and/or detail exhibit perceptually annoying disturbances.

The third option, although providing constant or near-constant picture quality, is far from being ideal as valuable resources are wasted away. While this point may not appear to be so critical in consumer storage applications, it is definitely one of the great concerns in television broadcasting, networked applications and will also be very important in high volume archive systems. In this last field of application however, the use of the MPEG2 standard is far from having reached consensus among the film and television industry communities. Many argue that archived material must be kept in the original format with no degradation at all, which means without using any kind of lossy compression. This is still an open question.

On the other hand, in the television production and broadcasting arena,

MPEG2 has already consolidated its position. One of the challenges which broadcasters are still facing with, is the ability to dynamically share bandwidth among different programmes and thus maintaining the picture quality level constant throughout the whole duration of the programmes without incurring in extra expensive costs. This is only achievable with the first approach. Ideally, it would be desirable to allow true variable bit rate encoding, setting a target constant picture quality and spending as much bits as necessary to meet that target. However in practice, due to limitations in channel bandwidth and to network behaviour towards bursty traffic, it is still necessary to keep a certain control on the generation of bits. To obtain good results in terms of constancy in picture quality across a number of different sources multiplexed together in the same channel, that control should be sensible not only to the characteristics of the source, its present needs and current achievable perceptual quality, but also to the characteristics and needs of the other sources. These kind of considerations have a particular emphasis in the case of broadcasters, who are now at disadvantage compared with their cable and satellite counterparts. In order to be able to compete with cable and DBS (Digital Broadcasting Satellite), terrestrial broadcasters need to devise and implement a new business model, focussing on the possibility of delivering a multi-channel service, including high-quality services such as HDTV programming and Internet-like content. And they should do this by using only their already available frequency spectrum. They could become a *so appealing* wireless facilities-based operator, broadcasting directly to the consumer both free and subscription services, as well as the long-promised HDTV services. An efficient use of the available bandwidth channel without sacrificing the quality of the services, could be supported by a formal statistical methodology, capable of making inferences about the general expected behaviour of video sources. And, as a consequence, being able to give the support for the implementation of efficient strategies for dimensioning

the services and managing the spectrum.

Standardised video encoding algorithms

This section provides a very brief overview of the video encoding standards that have had and/or are having, the strongest impact on the advance of A/V and multimedia communications, from low-end Internet streaming applications to high-end professional digital cinema domain.

MPEG-2 The MPEG-2 video coding standard is also known as recommendation ITU-T H.262 [102]. This specification was developed at the beginning of the nineties, and accepted as an official ITU-T standard in 1994.

It was develop having as a main target, digital television applications, providing increased performance and widen the scope of application in relation to its predecessors. MPEG-2 can be considered a tremendous success as the enabling technology for digital television systems all over the world worldwide. It was also widely adopted in high-quality digital storage applications, notably the DVD.

Among other, the following requirements were identified, and then fulfilled, for the development of MPEG-2:

- Support graceful degradation and better error resilience to enable transmission over error-prone channels;
- Support interlaced scanning, to allow easy integration with analog TV;
- Support a wide range of bit rates and source formats.

MPEG-2 is a video coding format that achieves high compression ratios through the use of the following major encoding techniques:

- hybrid DPCM/DCT model;
- use of different types of images, namely Intra (I), Predictive (P) and Bi-directionally predicted (B);
- forward, backward and bidirectional motion compensation techniques;
- dual prime technique in P images;
- possibility of using motion vectors inside I frames;
- new scanning mode;
- use of field-pictures and frame-pictures to support simultaneously interlaced and non-interlaced formats.

In order to support the identified requirements, different profiles and levels were defined. In particular scalable formats were defined where scalability can be obtained adopting four different approaches and accordingly, support different application requirements:

- **Data Partitioning** - data in the bitstream is partitioned according to its importance in the reconstruction of the bitstream. More important blocks of data in the decoding process (headers, for example, or low order DCT coefficients) can be transmitted over prioritised channels.
- **SNR** - in Signal to Noise Ratio scalability, the coded bitstream is organised in layers yielding increasingly higher quality. The base layer is the only that can be independently decoded. It carries sufficient information to reconstruct a full spatial and temporal video sequence, but presenting reduced quality. Additional information is sent in enhancement layers. At the decoder side, this

enhancement information can be added to that of the base layer, yielding a better quality version of the video sequence.

- **Spatial**- the coded stream data is divided in layers, with the Base layer carrying sufficient information for the decoder to reconstruct a full quality and temporal video sequence, but presenting reduced spatial resolution. Additional data is sent in enhancement layers, enabling the decoder to reconstruct increasingly higher spatial resolution versions.
- **temporal** - similar to the spatial scalability, but within the temporal domain (the roles of the temporal and spatial domains are interchanged).

The levels and profiles defined in MPEG-2 are related to different sets of restrictions applied to encoding parameters and to different sets of coding tools, respectively. The most widely used combination is the *Main profile at Main Level* (MP@ML). This is the pair *profile-level* used in digital TV broadcasting to distribute the content to the end-users.

The idea behind profiles and levels is the one of simplifying the use of the standard in different application scenarios. Different pairs *profile-level* are associated to different applications. This can greatly simplify the life of manufacturers, who only need to incorporate in their equipment the minimum set of functionality as identified by the profile-level adequate for the application they are aiming at.

H.264 and MPEG-4 AVC Recommendation H.264 of ITU-T and part 10 of the MPEG-4 standard (Advanced Video Coding, AVC), constitute in fact a unique specification, that was jointly developed by the Video Coding Experts Group (VCEG) of ITU-T and MPEG. It is usually referred to as H.264/AVC [101], [189],[108].

The H.264/AVC specification was finalized in March 2003 and approved as an official standard by the ITU-T, in May 2003. In relation to its predecessors algorithms,

H.264/AVC achieves a significant improvement in rate-distortion efficiency, thereby yielding better quality at lower bit rates.

It is thus suitable for use in transmission channels with limited capacity, or without guarantees of constant network availability, as is the case of cellular networks (UMTS and GPRS) or, generally, best-effort networks (and thus also on the Internet). Nevertheless its scope of application is considerably large, from low-end applications, such as those running on the Internet, to high-end applications, such as DVD and HDTV, as it achieves considerably increased performance at any bit rate when comparing to other compression schemes.

Basically, its improved coding efficiency is obtained by the use of the use of a number of new or techniques, or variants of existing ones:

- Variable block-size motion compensation with small block sizes;
- Quarter-sample-accurate motion compensation;
- Motion vectors over picture boundaries;
- Multiple reference frames;
- Weighted prediction;
- Integer transform;
- Improved "skipped" and "direct" motion inference;
- Directional spatial prediction for intra coding;
- De-blocking filtering.

To suit a wide range of requirements, without introducing excessive complexity, three different profiles are defined:

- *Baseline*, delivering lower capability and increased error resilience support. It is suitable for applications such as videoconferencing or mobile video applications;
- *Main*, aiming at optimising the balance between quality and compression. It is adequate for high-end applications, using channels with guarantees of resources, as for example TV broadcasting;
- *Extended*, introducing new features, especially suitable for efficient streaming applications.

JPEG 2000 JPEG 2000 [190], [89], [124], also known by the acronym "*j2k*", is an image coding system standardised by ISO/IEC, based on wavelet compression technology. JPEG 2000 is a standard composed of 12 parts², some of which have already concluded the complete process of becoming an official standard. Other parts are in the final process of standardisation or even in the development phase.:

- Part 1, Core coding system;
- Part 2, Extensions;
- Part 3, Motion JPEG 2000;
- Part 4, Conformance;
- Part 5, Reference software;
- Part 6, Compound image file format;
- Part 8, JPSEC (JPEG 2000 Security);
- Part 9, JPIP (interactive protocols and API);

²In fact, and although the existence of a part with the number 12, only 11 parts exist, as part 7 was abandoned.

- Part 10, JP3D (volumetric imaging);
- Part 11, JPWL (wireless applications);
- Part 12, ISO Base Media File Format (common with MPEG-4).

Part 1 [89], defines the core of JPEG 2000, which includes the syntax of the j2k coded stream and the description of the encoding and decoding processes of j2k images. It also defines a basic file format called JP2 (image files with the JPEG2000 encoding format have the extension . JP2). This file format provides the means to include metadata, notably metadata required for the accurate rendering of the j2k image, such as colour space information. Part 1 of the standard is the essential part when implementing a basic j2k image codec.

The j2k specification defines a scalable or progressive bitstream format, allowing to extracted parts of data from the full bitstream, to reconstruct images with lower resolution or with lower bit rate, or even only regions of the image. This is achieved by dividing the original image and the coded stream in a hierarchy of levels:

- **textbftiles**, which are rectangular arrays corresponding to regions in the picture and that can be decoded by themselves;
- **titles** are decomposed into **levels of sub-bands** using a wavelet transform. Each successive level has approximately half of the horizontal and vertical resolutions of the previous level;
- sub-bands are divided into **code-blocks**, consisting in rectangular arrays of coefficients

Of interest within the A/V industry, is also part 3, which specifies a file format suitable for storing motion sequences of j2k images. This file format is called

MJ2 or MJP2 and it also provides the support for the inclusion of audio. MJ2 does not involve inter-frame coding: each frame is coded independently using JPEG 2000.

2.2 Distribution technologies

Middleware is essential to enable integration of different technologies in a flexible way to allow the cooperation between systems using diverse technology and data formats [153], [157], []. Diversity is a reality of our days and as technology continues to evolve, more and more different formats, protocols, hardware and software will come into use. In this context, middleware should be as flexible as possible while providing high levels of performance. Middleware layers are able to functionally bridge the gap between different application tools, allowing the integration and inter-operation of components using different technologies and data models. Distributed technologies such as Web Services [198], or CORBA [141], can be used to support the development of this middleware layer. With Web Services technology, which allows the development and publishing of services to be used on the Web, it is possible for virtually anyone using a Web browser to access and use those services. CORBA, defined by the Object Management Group (OMG) [139] is an open architecture that allows the access to distributed objects or applications. Both technologies provide standardised methods to describe the interfaces to the applications. In this section we will start by providing a brief description of characteristics and types of middleware. We will then present the main features and structure of some of the most widely used technologies.

2.2.1 Types of middleware

Distinct characteristics can be used to classify middleware technologies. In this section we highlight some that are more relevant for the fulfilment of this thesis objectives, from an implementation point of view.

RPC-based and message-based Middleware based on Remote Procedure Calls (RPC) [193], [81], allows to invoke remote services in a similar way as function calls in traditional programming languages. When a procedure in a local application calls a procedure in a remote application, local execution stops until the request returns. This is similar to what happens within a single application when a method is called. The use of *RPC-based* middleware can be recommended when local and remote applications need to frequently invoke functions belonging to each other and the amount of information to be exchanged is reduced. On the other hand, middleware that is based on the exchange of messages (i.e., *message-based*) implement a data queuing abstraction. Local applications post data to queues, which is retrieved and processed by remote applications. This type of interaction between inter-application methods is more interesting when the amount of information to be exchanged per interaction is rather high comparing to the number of interactions. Whereas RPC systems are object or procedure/function/methods oriented systems, messaging systems are typically information or document oriented.

Java RMI [134], [82] and CORBA [141] are examples of RPC-based middleware, whereas Web Services can be either RPC-based or message-based systems.

Language dependency Middleware systems can support the integration of applications written in different programming languages. This means that they allow applications to specify their external interfaces choosing one among a number of dif-

ferent languages. CORBA is a good example of a language-independent system. In CORBA, software objects can be remotely accessed once they have been declared as CORBA components. This is accomplished through the use of the CORBA Interface Definition Language (IDL), which supports different programming languages. There are other systems, such as the Java Platform Enterprise Edition (JEE) [183], based on the assumption that only one programming language is in use, in this case JAVA. However, JEE platform-based applications can be deployed into a variety of operating systems taking advantage of the JAVA portability.

Standard or proprietary This feature is quite important towards an effective and dynamic cooperation across distinct environments. The use of open standards can provide the means to easily scale the systems and integrate new technology. However, a standards-based approach may also present some drawbacks. On one hand, specifications tend to be quite complex as they try to incorporate as much functionality and flexibility as possible. On the other hand, it is virtually impossible for any standard to cover all existing requirements. In addition, standardisation processes usually require a considerable amount of time and effort. Proprietary solutions may overcome these problems and provide simpler implementations and deliver higher performance levels, although usually focusing in specific application areas and needs. CORBA is an example of a quite successful standards-based middleware. It is specified by the Open Management Group (OMG). Java EE is an industry-based standard developed by Sun Microsystems. The source code is made available through a Licensing Agreement scheme. DCOM on the other and is a proprietary middleware owned by Microsoft.

2.2.2 Brief description of some existing middleware technologies

DCOM DCOM, Distributed Component Object Model [32] is a proprietary technology of Microsoft. DCOM extends the Component Object Model (COM). Virtually any language can be used to create COM components and these can be used by a great number of programming languages and tools. Examples are the Java, Microsoft Visual C++ or Microsoft Visual Basic programming languages. DCOM defines how components and their clients interact, and it does so in a way such that the client and the component can connect without the need of any intermediary system component. The client is able to call methods in the remote component without any overhead. Critical design constraints of distributed software are fairly easy to work around, because the details of deployment are not specified in the source code. DCOM completely hides the location of a component, whether the component is part of the same process as the client or it is located in any other remote machine. In all cases, the way the client connects to a component and calls the component's methods is identical. DCOM provides mechanisms for location independency, which greatly simplify the task of distributing application components for optimum overall performance.

DCOM is thus a proprietary, RPC-based and language independent middleware technology that provides location-independency support.

CORBA The Common Object Request Broker Architecture (CORBA) [141], [136], is an open distributed object computing infrastructure standardised by the Object Management Group (OMG)[139]. CORBA is a language-independent technology based on the use of remote function calls. In addition CORBA follows a service-oriented approach by which the functionality delivered by the remotely accessible

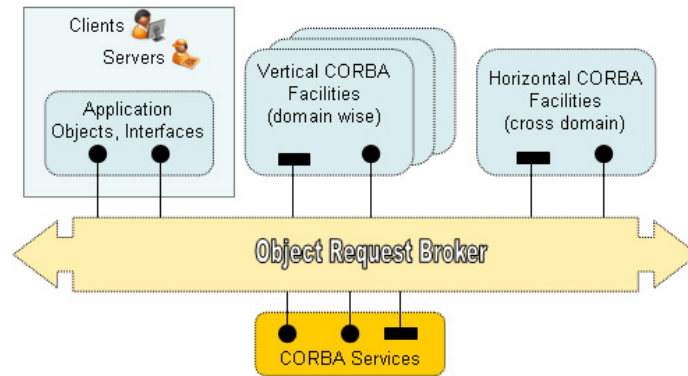


Figure 2.1. OMG CORBA Reference Model Architecture

methods is seen as services. Figure 2.1 presents the generic OMG reference model architecture. CORBA automates many common network programming tasks such as object registration, location, and activation; request de-multiplexing; framing and error-handling; parameter marshalling and de-marshalling; and operation dispatching.

In the CORBA model, an Object Request Broker (ORB) allows objects to publish their interfaces and thus make remotely available their services. It also allows client programs to locate these remote objects and request services from them. The model includes components that have a horizontal scope, i.e., components that provide functionality that can be used by many different application domains or different distributed applications in specific application domains. These are the Object Services, the Common Facilities and the Domain Interfaces components. Examples of Object Services, which are provided directly with the CORBA implementation and are extensively used by middleware client applications, are the *Naming* and the *Trading* services. These services provide location-independence functionality similar to a services directory listing. The Naming Service allows a client to find objects

based on their names, while the Trading service allows locating objects based on their properties.

Application Interfaces are developed specifically for a given application and as such are not standardised by the OMG. The Object Request Broker (ORB) component (which is the core of CORBA), provides as its core functionality, a mechanism for transparently communicating client requests to target object implementations. The ORB simplifies distributed programming by decoupling the client from the details of the method invocations. This makes client requests appear to be local procedure calls. When a client invokes an operation, the ORB is responsible for finding the object implementation, transparently activating it if necessary, delivering the request to the object, and returning any response to the caller.

Requests and responses between objects are delivered in a standard format defined by the Internet Inter-ORB Protocol (IIOP) [137]. The CORBA specification defines interoperability elements to support interaction between objects that are distributed across different CORBA-compliant ORBs. The main element of this interoperability framework is the General Inter-ORB Protocol (GIOP). It specifies the syntax and a set of message formats to exchange data between ORBs and is designed to work directly over any connection-oriented transport protocol. Another essential element is the Internet Inter-ORB Protocol (IIOP), which specifies how GIOP messages are exchanged using TCP/IP connections.

An important mechanism provided by the ORB is the CORBA Interface Definition Language (IDL), by which it is possible to separate the definition of the interface from its actual implementation. CORBA applications are composed of ob-

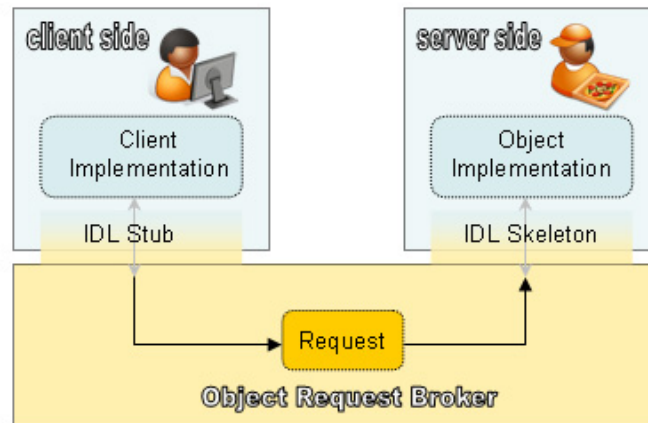


Figure 2.2. Invocation of a local object in CORBA

jects, which are individual units of running software providing specific functionality. Each object has its own interface defined using the OMG IDL. The interface specifies the syntax of the value delivered by the object to clients that invoke it. At the client side, applications should then use this IDL-based interface to specify the operation to be performed and to marshal the arguments passed in the service invocation. At the server side (called object) the same IDL interface is used to unmarshal the received arguments and to marshal the results of the operation to be sent to the calling object. Finally at the client side, it is used again to unmarshal the results. To perform all these operations, the IDL is compiled at both ends generating client *stubs* and *skeletons* objects. Figure 2.2 illustrates in a simple way, the described invocation process.

Stubs and skeletons act as proxies for clients and servers, respectively and provide the glue between the client and server applications and the ORB. The transformation between CORBA IDL definitions and the target programming language is automated by a CORBA IDL compiler. The use of a compiler reduces the po-

tential for inconsistencies between client stubs and server skeletons and increases opportunities for automated compiler optimizations.

Web Services Web Services [198] constitute an excellent way to enable inter-operation between applications on the Internet. This is partly due to the fact that Web Services are based on commonly known technologies such as HTTP and XML, as well as on proven and widely adopted approaches from other middleware technologies such as the case of the services-oriented approach of CORBA. They also present flexibility by supporting both RPC-oriented and message-oriented systems. In the last case the communication between applications is accomplished through the exchange of XML-based messages. Also the interfaces of applications and services are defined through the use of XML. This means that it is not necessary to use specialised compilers as in the case of the CORBA IDLs to build them.

A Web Service is an application or resource that can be accessed using standard Web protocols such as HTTP (HyperText Transfer Protocol)[185], [154], SMTP (Simple Mail Transfer protocol) [150], etc. Any type of application can be offered as a Web Service by exposing its functionality to the network. Typically a Web service describes its interface using a standard format and publishes this description in an interface repository.

The following is the definition of Web Services as agreed by the W3C:

”A Web Service is a software application identified by a URI, whose interfaces and binding are capable of being defined, described, and discovered by XML artifacts and that supports direct interactions with other software applications using XML-based messages via Internet-based protocols.”

Web Services technology is in fact a compilation of technologies: XML,

eXtensible Markup Language [208]; SOAP, Simple Object Access Protocol [135]; WSDL, Web Services Description Language [159]; and UDDI, Universal Description, Discovery, and Integration [1].

Through the registry service UDDI, which is an interface repository, listing available Web Services similar to a "Yellow Pages" catalogue for the Internet, it is possible for virtually anyone using a browser to access and use those services. The UDDI maintains a sort of information database of Web Services, with XML descriptions of functionality offered by the registered services. It registers services available on the Internet, allowing potential clients (Web-based applications) to discover and invoke them using SOAP messages. This is similar to the role of the "Yellow Pages" phonebook, which lists business, their products and services in a telephone directory, providing the required information to enable potential customers to contact them.

SOAP is the protocol for exchanging information in distributed environments defining a mechanism to pass commands and parameters between applications (clients and servers). SOAP is independent of the platform, object model, and programming language being used to develop those applications. SOAP is based on XML and it consists of three parts: 1) the envelope, which is the root XML element in the XML document tree representing the message. It provides the means to describe the content of the message and how to process it; 2) the header, which is a generic mechanism to add characteristics to the SOAP message; and 3) the body, which is the container of the mandatory information sent in the SOAP message.

Usually, SOAP messages use as transport mechanism the HTTP protocol. In fact, the SOAP specification includes elements to describe how to use the HTTP protocol. Nevertheless, as referred above, it is possible to carry SOAP messages using other

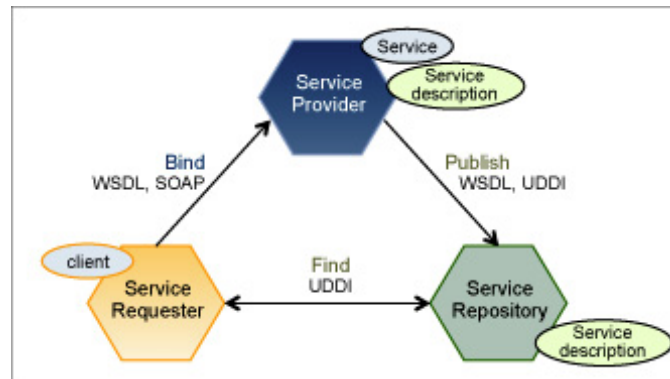


Figure 2.3. Web Services Conceptual Architecture

transport protocols such as SMTP or JMS (Java Message Service) [125].

WSDL is an open specification that provides standard means to describe Web Services. It provides in fact the description of the interface to a service. WSDL is an XML schema that describes services as collections of methods that are capable of exchanging messages over a network. A WSDL service definition is an XML document that provides the description of the interfaces made available by a service. It thus provides the necessary indications for remote clients to use those services - how to invoke the service (the service endpoint URI), the names of the available methods, the list of parameters, the expected return, etc. WSDL is extensible, allowing to describe services through their endpoints and messages, regardless of the message formats or the type of network protocols they use for communication.

Figure 2.3 presents the conceptual architecture of Web Services.

Java Enterprise Edition, JEE JEE [183] is a suit of technical specifications and communication APIs (Application Programming Interfaces), developed by Sun

Microsystems. It provides a standard description of how distributed object-oriented programs written in Java should be designed and developed and how the various Java components can communicate and interact.

Enterprise Java Beans (EJB) [182] is an important part of the JEE specification and describes a server-side component-based programming model. Using this model it is possible to build distributed applications by combining components developed with tools from different vendors. JEE describes various enterprise-wide services, including naming, transactions, component life cycle, and persistence, and how these services should be uniformly provided and accessed. JEE also supports Web Services thus enabling public methods of the session beans to be exposed as Web Services on applications servers conforming to the JEE specification³.

The Java EE platform provides a Java API for developing XML Web Services, known as JAX-WS [184]. Using the JAX-WS framework, it is possible to develop SOAP-based clients and end-points and to describe these as WSDL documents. And also to operate them across heterogeneous platforms (i.e., JAX-WS clients can invoke Web Services developed in different platforms and JAX-WS Web Service endpoints can be invoked by clients developed in other platforms.

conclusions

Due to its great flexibility and simplicity, Web Services is the preferred distributed technology to be used in the framework of this thesis. However, as Web Services technology has the capability of promoting the inter-operation between different middleware technologies, CORBA or simple RPC based mechanisms may

³It is also possible to use the non-standard modules such as the Apache AXIS server[11]

also be used. Notably, in specific cases where the frequency of messages exchanged between the objects is rather high and the amount of information is reduced [52]⁴.

Web Services are ideal for application integration both internally at the level of individual complex systems, as well as at a larger scope to interconnect software over the Internet. Web Services technologies are based on open standards recommended by the World Wide Web Consortium (W3C). The key Web Services technologies, SOAP, WSDL and UDDI, are all based on XML and, usually, the transport of SOAP messages is done using well-known transport protocols such as HTTP. In addition, Web Services are "*firewall-friendly*" which makes them suitable to be used in loosely-couple environments, such as the Internet, where different systems owned by distinct entities interoperate. Finally, and as well important, support for the implementation of Web Services is quite generalised, both in the industry as well as in the academic environments.

2.3 Metadata schemes

Metadata serves two purposes within the context of this thesis. On one side it is indispensable to enable locating networked multimedia content. On the other side, it is instrumental to allow the adapted consumption of the located content under specific conditions and characteristics of the usage environment. The recent and continuing advances in technology have made possible the development and proliferation in the consumer market of a variety of client devices capable of presenting multimedia content in many distinct ways. As such, the true access to the useful content is only possible if the content can be efficiently searched and queried and then retrieved and adapted for consumption under the specific usage environment

⁴It should nevertheless be emphasized that, although WS technologies are primarily message or document oriented, they can also be implemented adopting an RPC-based approach.

characteristics. These are the main roles of metadata within the scope of this thesis, notably for the development of the content mediation platform:

- to provide useful descriptions about the content so that content provision systems are able to perform efficient keyword or semantic-based searches;
- to provide indications concerning technical characteristics of the content thus empowering content mediation systems to devise required capabilities on terminals, networks and adaptation engines;
- to provide rich description of the context of usage, detailing as much as possible the characteristics and conditions of the consumption environment to assist the content adaptation decision.

Moreover, the implementation of the proposed approach for the statistical multiplexing of multimedia sources, also relies on the use of metadata. The statistical multiplexer uses information describing statistical properties of the A/V sources to be able to infer the behaviour of sources and accordingly optimise the use of available resources.

The today's abundance of archived digital multimedia content does not make it more accessible for use. The information explosion triggered by the emergence of the World Wide Web made enormous amounts of information available over the Internet, but at the same time the sheer size of this information turned search and retrieval of the useful information into an extremely difficult task. This is even truer when considering the non textual, multimedia information to which neither keyword indexing, nor advanced text mining techniques can be directly applied. As a result, many broadcasting corporations, news agencies, governmental institutions, etc are stuck with large digital multimedia archives that are of little use simply because most

of their content is unknown, and cannot be efficiently queried and managed.

There is therefore the need for new models for processing, organizing and indexing this data to enable searching, retrieval and delivery of the useful information. Models for multimedia content storage and retrieval must account for both context and content information, taking advantage of their dependencies for effective retrieval. Many research groups from all around the globe have been addressing the problem of indexing, categorizing, and searching large volumes of images and, more recently, audiovisual material.

Current solutions in use are extremely labor-intensive, requiring manual annotation of content. Manual annotation is extremely time-consuming and often subjective, often leading to incomplete and inconsistent annotations, preventing efficient retrieval of the useful content. Therefore researchers have been devoting their efforts to the study of techniques to automate the annotation process, by using advanced content analysis techniques for indexing the A/V data.

Approaches, as those based on machine-learning techniques, are being investigated to analyze and fuse sound, visual, and text information to automatically annotate multimedia data and to exploit the relationships among annotations, to improve search performance. Some approaches rely on the extraction of low-level features of the A/V content such as color, texture or shape, while others try to explore also higher-level characteristics of the content such as, recognizing particular types of objects, scenes or speakers.

A/V sequences can be analyzed by multiple systems such as, speech recognizers, audio analyzers, speaker identification systems, shot boundary detection systems, or text segmentation. The extracted information can then be stored together with the content, or not, in the form of metadata.

Metadata is commonly defined using its literal interpretation: *data about data*. Simply said, metadata is any kind of additional information concerning any type of content. Figure 2.4 illustrates this concept using as content the painting "The Incredulity of Saint Thomas" from (Michelangelo Merisi da) Caravaggio. In the example, metadata is shown as textual information.



Figure 2.4. A simple example of metadata

Metadata consists of a standardized or formalized structured format and a controlled vocabulary, enabling to describe stored content. Metadata often includes items like file type, file name, creator name, date of creation, etc. The main reason for metadata is to improve the management and access to information.

In the context of digital audiovisual objects, and according to its functional role, metadata can be classified into three categories (Wendler 1999 [204]):

- Descriptive, facilitating resource discovery and identification;
- Administrative, supporting resource management within a collection;
- Structural, binding together the components of complex information objects.

Descriptive metadata, as defined above, is a type of metadata related with the content, and referred as *content-related metadata*. In fact the above categorisation, due to presented definitions, inherently refers to only descriptions that are related with the content itself. The first class, descriptive content-related metadata, has been the one that has received more attention in the research community and also in commercial applications. One example is the Dublin Core Metadata Initiative (DCMI) [10], [83]. The structural aspects have more recently been subjected to strong activity, notably through the development of standards such as the MXF [13], [170] and MPEG-21 [94], [79] specifications.

However there is also another type of descriptive metadata, referred to as *contextual metadata*. It provides the means to describe the context of usage including the capabilities of encoders, transcoders (or generically speaking, adaptation engines) and terminals, the characteristics and conditions of networks, the real world environments and preferences of users. This type of metadata has already been mentioned in this manuscript and referred as being also instrumental for the successful implementation of the of the ubiquitous and QoS-aware access to multimedia content aiming at by this thesis.

Content-related descriptive metadata can be classified into two groups of descriptions:

- Low-level;
- High-level.

The first group includes all type of descriptions that can be extracted directly from the content. They are usually obtained in an automatic way by analysing the content. This kind of descriptions can be seen as intrinsic to the content in the sense that they provide measures (values, names) of characteristics of the encoded

bitstream. Within this group of descriptions it is common to find:

- colour or mouvement information, edges and textures in image or video content;
- timbral, spectral features, signal parameters such as fundamental frequency, pitch and rhythmic information in audio signals.

The second group, also called *context about the content descriptions*, addresses aspects of the content that cannot be directly extracted from it. They are generally obtained by manually annotating the content. As their generation usually requires the intervention of the user, they are prone to the subjective opinion of the annotator. This kind of descriptions provide semantic information about the content, i. e. an interpretation of the concept transmitted by the content. They are thus independent from the coding format but on the other hand are highly dependent on the person that creates them and her interpretation of the content, as well as on the context or application where the content is embedded in.

Using low-level descriptions to infer higher-level concepts is a step towards bridging the semantic gap (a very challenging topic currently being addressed within the Web 2.0 or Semantic Web paradigm). Low-level descriptions can be used in a more consistent manner. They remain always the same, provided the content does not change, regardless of the whole context the content is inserted. However they fail to provide a good description of the concepts and interpretations humans usually give to content and which they feel comfortable with when performing content queries.

On the other hand, the high-level or semantic-based descriptions provide better approximation of real world concepts but are highly subjective. Different applications and content providers may use different high-level descriptions for the same or similar contents. They depend both on the person that creates them and usually are application-dependent or context-dependent. For example, a descriptive

record of a visual resource containing a picture of the *Baldacchino*, the High Altar in St. Peter's Basilica made by the Italian baroc artist GianLorenzo Bernini (see Figure 2.5), can include many different labels as for example labels referring to the name of the work of art (Baldachino), to the creator (Bernini), to the type (sculpture) to the place (Rome, St. Peter's church) or to the period when it was created (baroc and/or year). If it was annotated for a programme concerning Bernini's life, then it will probably contain all this information. But then again, it may not. If it was annotated within the context of a programme about Rome or the Vatican, then it may have only information about the place. Accordingly, a query with the keywords "baroc", "Bernini" or "sculpture" will not locate this resource.

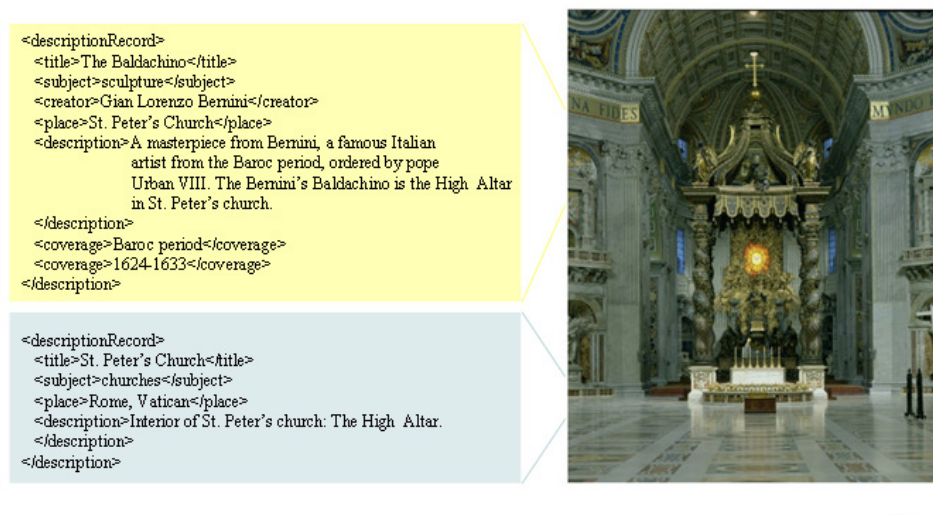


Figure 2.5. Different description records for a resource showing the canopy over the main altar in St. Peter's Basilica made by Bernini

Low-level descriptions have always the same meaning and depend only on the content itself. For example, a red car will always have the same (or at least,

similar) colour histogram whatever the picture or video sequence it is inserted in. However different high-level descriptions may be assigned to that picture or scene, as for example *formula 1 race* or *trip to London*.

Accordingly, it is easy to understand the benefits that may arise from jointly using the two types of descriptions. Moreover, the advantages that may come from the development of dependable systems that reason about the low-level descriptions to infer high-level concepts. This would indeed obviate the need for the intensive-labour work of manual annotators, while still providing descriptions closely meeting the way humans describe and search the content.

Most of the metadata schemes that have been or are being developed address the specification of high-level, content-related metadata (which includes the context of the content). Some also cover the definition of structural content-related metadata, especially in the TV industry as well as preferences and profiles of users. Very few are concerned with the non-user related contextual metadata (descriptions not directly related with the content but rather with terminals, networks or environments), although the generality and flexibility of some schemes could allow their use for this purpose. An even more restricted number of schemes specify description elements to express low-level features of the content.

Finally there is another aspect concerning metadata and standardised schemes. It refers to the way links are established between the metadata and the resources which the metadata refers to. Usually, as the process of generating metadata is different from the process of generating the content (although they could be conducted in parallel) and usually the two have different types, metadata records are kept in separated files. The link is obtained by including in the metadata record an element that uniquely identifies the resource it refers to. For example in networked resources, such as a Web page, this unique identifier could be the page URL [86]; in a library,

the library's catalogue records would contain the ISBN (International Standard Book Numbers). But metadata could also be embedded in the resource or carried along with the resource. Standards that are targeting environments where often resources are exchanged between applications or entities that need to process those resources, are the ones that are specifying the means to wrap together metadata and content.

The remainder of this section provides a brief overview of the most important metadata schemes that have been (or are being) standardized and that are in use within the audiovisual arena. Reference is also made to important specifications that have emerged within the library and cultural heritage domains. In addition to the metadata element sets, it is also referred initiatives to standardize the exchange of information between the repositories and the applications. As already stated at the beginning of this chapter, its purpose is to describe in a brief manner, technologies and tools that are important to the development of this thesis but that have not been a subject of study or research. Nevertheless, as metadata is crucial for the successful implementation of the context-aware access to multimedia content, some of the schemes here presented have been addressed in more detail. They have played an essential role in the design and development of the context-aware content mediation platform. This is the case of parts of the MPEG-7 and MPEG-21 standards [15], [94]

⁵ Accordingly they will be further described in more detail in the next section devoted to the review of the state-of-the-art. Likewise, due to its great importance in the Web 2.0 paradigm and consequently to the ubiquitous access to multimedia content, Resource Description Framework (RDF) and other W3C specifications [207] will also be analysed in the same section of this manuscript. W3C technologies occupy an important role towards the bridging between semantic-oriented environments and other

⁵In fact, within the course of the work that was conducted with the aim of implementing the solution proposed by this thesis - content mediation platform for the context-aware ubiquitous access to multimedia content - the work of MPEG-21 was closely followed. Moreover, several contributions were submitted to this forum.

more technical and low-level oriented.

2.3.1 MPEG-7

The Multimedia Description Framework (MPEG-7)[15] is a metadata standard specified by ISO/IEC, aiming at describing characteristics of multimedia content using XML Schema. MPEG-7 can be seen as a metadata standard that allows the representation of both content-related and contextual metadata. Moreover, at the content level, it defines both low-level and high-level description elements. Among the available metadata standards, it is the one that provides the most comprehensive set of audio-visual description tools.

These description tools cover a wide range of characteristics, from high-level features, in both a catalogue-oriented form (as for example, title, creator, date, etc.) as well as semantic-driven (descriptions about objects and events included or occurring in the A/V scenes or even about emotions that are likely to be expressed/motivated when consuming the content - this is sometimes refer as *the who, what, when, where and how* information about objects and events); to low-level features that describe the nature of the A/V binary content (such as colour associated with an image, mouvement associated with an A/V sequence or the pitch of a recorded instrument).

The low-level descriptions are addressed in the Visual and Audio parts of the MPEG-7 standard [91], [92], whereas the high-level metadata is addressed in the Description Schemes part of the standard [93].

The MPEG-7 standard defines the following five types of element or tools [99]:

- *Descriptors* (D) to represent features or characteristics of the content being described. A *Descriptor* is not high-level semantic descriptive information but

rather a structure with well defined syntax and semantics, providing name and values for single characteristics of the content (for example colour of a visual object or pitch of an audio signal);

- *Description Schemes* (DS) that enable to establish relationships between Descriptors or even between different Description Schemes. A DS may thus be composed of Ds and/or DSs. It contains descriptive information;
- A *Description Definition Language* (DDL) used to express Ds and DSs. It allows to create new elements or to extend or modify existing normalised DSs and Ds. The DDL is based on XML schema [208] and is specified as part 5 of the MPEG-7 standard [93];
- *System tools* to define the way descriptions and content are multiplexed. These tools address aspects such as synchronisation, transmission or file format;
- *Datatypes* to provide basic, generic and reusable description elements that can be employed by multiple Ds and DSs.

2.3.2 MPEG-21

The Multimedia Framework standard (MPEG-21)[94], [96], [79] is a set of open specifications that address precisely the aspects of interoperability among formats and systems aiming at enlarging the use of multimedia content in different consumption environments. MPEG-21 is the ISO/IEC standard currently under its final phase of development in MPEG. It focuses on the development of an extensive set of descriptions and tools to facilitate the transactions of multimedia content in heterogeneous network environments. MPEG-21 is a suite of specifications essentially targeted to the representation and use of contextual information. The standard is currently divided into 18 parts. The basic concepts of MPEG-21 are the User and

the Digital Item (DI). The User is any type of actor that manipulates content, be it a person or a system (e.g. subscriber, producer, provider, network). A DI is the smallest unit of content for transaction and consumption. A DI can be seen a virtual package of multimedia resources typically related to a certain subject (e.g. MPEG-2 video stream, MP3 file, a set of JPEG pictures, text file, etc.), together with associated descriptions (e.g. content-related metadata, rights descriptions). It also includes contextual information that can be used to improve the interaction with and consumption of that DI. Finally an essential part of the MPEG-21 DI is its Digital Item Declaration (DID) [96]. The DID is an XML document with syntax and semantics well defined, enabling the declaration of a structured, complex multimedia object (the DI), being itself a part of that same DI. The DID is itself metadata as it provides additional information on the DI. The DID is expressed in DIDL (Digital Item Declaration Language), which is a standardised XML-based language [96]. A DI is thus used to encapsulate as a virtual package or single unit, a variety of multimedia information (content and metadata). Resources can be directly included in the DID, but the most common is to place references inside the DID to external files or sources. A straightforward example of a DI is the case of a music album composed of a set of sound tracks and other information elements either in the form of resources or in the form of metadata, such as video clips of some of the tracks, images of the album's cover, an A/V interview with the performer and some photos, lyrics of the songs, the date of recording, the length of the songs, information about the author and/or about the performer such as a text biography, Web links, DRM descriptions, etc.

In addition to defining an XML-based model to express the structure of complex digital objects and describe its components, the MPEG-21 standard also specifies tools to use the contextual descriptions, for example to assist adaptation decision operations [98] and methods to provide a coarse control over the operations

that are performed upon the DI when interacting with it [95]. These methods (Digital Item Methods, DIM) can be automatically invoked and executed thus allowing to pre-define the operations performed upon the DI when interaction with it. On one hand, this provides the means for content creators or providers to specify the way the DI will be manipulated; on the other hand it provides a vehicle to promote interoperability across systems.

2.3.3 SMEF

The Standard Media Exchange Framework (SMEF) [17], has been defined by the BBC to support media asset management along the complete TV production chain, from the business viewpoint. SMEF uses a *data dictionary* and a set of *Entity Relationship Diagrams* to express the information generated in production, distribution and management of media assets, including the relationships between participating entities. Although not being an open standard endorsed by an official body, SMEF is publicly available via a free-licensing scheme and has received much attention from many key players in the content industry.

2.3.4 AAF

The Advanced Authoring Format (AAF), is a joint initiative of a number of large media organisations and software companies working together as the AAF Association ⁶[12]. It aims at providing a new media industry standard file format, designed to meet the requirements of information interchange needs that exist in today's media industry, notably within the authoring phases. Authoring is described as the process of creating multimedia content, including the related metadata. The AAF specification is specifically meant to operate within this part of the production

⁶The AAF Association was recently renamed as Advanced Media Workflow Association (AMWA).

process and does not support the delivery phase of multimedia content (e.g. play-out, broadcast).

In the authoring environment, it is common to use a wide range of source materials with distinct formats, as well as a varied number of tools presenting different characteristics, requirements and capabilities (e.g., cameras for the acquisition, PCs or special purpose editing stations, VCRs and data storage equipment, audio sources, scanners, etc.). This diversity often leads to the necessity of converting media into formats that can be used by the different applications and tools. AAF specification addresses these needs, facilitating the interchange and use of large collections of interrelated sets of metadata and essence. It defines a file format that is able to wrap all this information (audio and video essences and associated metadata), together with additional descriptive information to allow applications to unwrap this package. AAF makes use of SMPTE (Society of Motion Picture and Television Engineers) metadata and labels, defining a base set of the built in SMPTE classes that are likely to satisfy basic needs of applications. In addition it provides a mechanism to define new classes that provide specific functionality.

2.3.5 UMID

The Unique Material Identifier (UMID) is a metadata specification standardised by the SMPTE under the reference *SMPTE 330M* [173] with very specific and oriented goals. UMID is a unique identifier for audiovisual material. It can be generated automatically without reference to a central database⁷ It enables any A/V material to be identified uniquely from anywhere, even if the material has moved. It thus provides a unique reference for any kind of A/V content, being mostly used

⁷The SMPTE provides a Registration Authority (LLC), officially entitled to act as the provider of Unique Material Identifiers. It is possible to request on-line UMID registrations in the url <http://http://www.smpte-ra.org/>.

within the television broadcasting environments. Additionally, these unique identifiers provide the means to establish a link between the essence (video, audio, graphics, still images, etc) and the corresponding metadata, generating a time code for synchronising this data. UMIDs can be assigned to the content units with different levels of granularity, from complete A/V sequences (for example, a movie), through shots, up to the frame level. The UMID structure contains a set of components, covering different aspects or characteristics of the essence it refers to. There are two sorts of UMIDs: 1) the basic UMID, with the minimal components necessary for the unique identification, presenting a length of 32 bytes; and 2) the extended UMID with 64 bytes of length that, in addition, provides information on the creation time and date, recording location and names of the organisation and the author of the content (i. e., the signature metadata). UMIDs can also be used to keep control of the Intellectual Property Rights (IPR) that rule the usage of the essence.

2.3.6 MXF

The Multimedia Exchange Format (MXF), is an open file format targeted at the interchange of audiovisual material, with associated data and metadata [13]. MXF is a file format especially developed to be used as a container of source material and finished programmes in TV broadcasting and digital cinema environments. As opposed to AAF, this format was not developed with the aim of being used within the editing phases. Accordingly it is optimized for sequential reading and writing and is suitable for implementation in a wide range of professional A/V acquisition and storing devices, as well as in PC-based devices. MXF was developed by the Pro-MPEG Forum [152] and AAF Association [12] (or AMWA - see footnote 6) and has been standardised by the SMPTE. The *SMPTE 377M* standard [170] specifies the file structure and metadata of MXF. The MXF file structure enables to building



Figure 2.6. A simplified view of the structure of a partition of the MXF container

packages containing content and associated metadata. A very simple illustration of this structure is provided in Figure 2.6.

The MXF metadata is encoded using the KLV (Key-Length-Value) format, as specified by *SMPTE 336M-2001* [169], and employs data elements also specified by SMPTE, as for example, *SMPTE Universal Labels* [167], the SMPTE Metadata Dictionary [175] or the *SMPTE UMID* [173].

In addition to structural metadata, high-level descriptive information, also called as *context of the content*, can be included in the header. An initiative under SMPTE is currently undergoing to look at the specification of descriptive metadata designated as *SMPTE 380M Material Exchange Format (MXF) Descriptive Metadata Scheme-1* [174]. This metadata includes information such as title of the material, broadcasting time or production time.

The MXF essence container was designed as a generic container in order to accommodate different types of essence coding formats. Its format is specified in the *SMPTE 379M* standard [172]. The encapsulation of essence with a specific encoding format is accomplished by following specific mapping rules. Various essence mappings have already been defined among which a generic mapping of MPEG-2. Work is currently being finalised to define a mapping for the JPEG2000 encoding format [89], which is being regarded as the format to be adopted in digital cinema applications.

The MFX standard is intended to support file interoperability between content creation devices, servers and workstations, supporting the integration of file-based and streaming formats. By encapsulating or wrapping the content in a standardised format, it promotes the de-coupling of the coding format from the storage or transmission format.

MXF is a format that supports implementation of the so-called *non-destructive authoring* [213]. In the conventional approach, *designated as destructive authoring*, each editing stage incorporates directly into the incoming material all the effects and new editorial changes, usually without keeping record of those modifications nor any other additional information relevant for the process. The best chance is that the editorial metadata is stored in proprietary formats (sometimes it may even be some handwritten paragraphs in the editor or journalist logbook). This approach does not preserve the original materials or the information to arrive to them. On the other hand, MXF provides a package with a standardised format, where content and metadata can be wrapped together, as well as additional information linking the two. MXF promotes the adoption of the non-destructive authoring model, whereby content and metadata are kept aside and appropriately referenced as it happens with markup languages on the Web. Accordingly, MXF provides the mechanisms to allow re-usability of metadata within the production chain from the creation of the content to its transmission. MXF uses UMID metadata in its header to provide cross-references between essence and metadata.

2.3.7 GXF

The General eXchange Format was originally conceived for the interchange of simple camera shots over data networks and for its archival storage on data tape [40]. It Initially it provided the support for only JPEG video and uncompressed

audio. It was then extended to include additional compression types and to support compositions with video cuts, audio fades and metadata. It was standardised by the SMPTE under the reference *SMPTE 360M* [171].

MXF, AAF and GXF are specifications of file exchange formats for use in networked television production.

2.3.8 EBU P/Meta

The P/Meta specification is an initiative of the European Broadcasting Union (EBU) to incorporate the work of SMPTE, MPEG-7 and SMEF into a common metadata exchange specification that could be suitable to a number of different entities or business roles within the A/V or multimedia content chain: Producers, Broadcasters, Publishers and Archivers [75], [76]. In general terms the fundamental needs of these business entities comprehend the ability to identify and recognize media assets, the possibility to incorporate editorial and descriptive information, including forms of expressing the rights to use the material and of providing guidelines on how to use it (open and/or playback). To achieve this goal, the P/Meta metadata scheme defines the following elements:

- a number of *attributes* with complete semantic definition;
- *transaction sets* comprising its definition of purpose and content addressed. Each transaction set is built from attributes and from other sets;
- *reference data* (also known as 'enumerated values', 'code values' or 'controlled value sets') defined for specific attributes;
- a *syntax* and *notation* for set construction.

2.3.9 DVB-SI

The Service Information (SI) specification of the DVB standard, provides a set of elements for the full description of TV Broadcasting programmes. DVB-SI consists of *content-related context* and *structural* metadata. It provides high-level information concerning TV programmes such as genre, names of producer or actors, language, duration, time and date of transmission, channel and networks etc. (the *context of the content*). The nature and purpose of the DVB-SI metadata is twofold:

- structural metadata to provide the links between the different components of the TV programme and to allow TV receivers to synchronise, locate and correctly reassembled those transmitted components of the programme;
- metadata of the context of the content, including also textual semantic descriptions, that assist the user in selecting his programming and that can also be used to automatically program the receiver with the user choices preferences.

The DVB-SI metadata is formatted in different tables that are sent in multiplexed streams together with the A/V components of TV programmes. This multiplex has a structure according to the DVB specification [56] which in turn is based on the MPEG-2 Systems standard [88], more specifically on its Transport Stream (TS) syntax. The DVB-SI specification includes the four Program Specific Information (PSI) tables specified in MPEG-21 TS (see table 2.1) ⁸ and other additional nine specifically defined by DVB. The former carry information concerning a single multiplex (the multiplex where they are contained in). It should be noted that in

⁸The MPEG-2 TS specification defines a packet-oriented multiplexing structure for the transmission of information from different TV programmes. Information from one given programme is partitioned in fixed size packet (188 bytes) that are sent interleaved in time with other programmes without any fixed ordering. Accordingly, the operation of re-assembling individual programmes at the reception is only possible if packet identifiers are used and the way they are used is specified and known at the receiver. This is precisely the role of the PID (sent in the TS packet headers) and of some of the PSI tables (the PAT and the PMT).

the present scenario of digital TV broadcasting, there is an enormous number of TV channels that are broadcasted or transmitted, well exceeding the capacity of a single multiplex. As such, current transmission systems use several multiplexes or networks. Figure 2.7 illustrates this situation. The nine tables specified by DVB (see table 2.2) provide information concerning services and events occurring also in multiplexes other than the one they are inserted in. DVB-SI tables provide the means to TV Content and Service providers to be able to offer a large number of channels and services to their audiences using a complex network of multiplexes. The information contained therein allow TV receivers to identify and access any programme across a complex network.

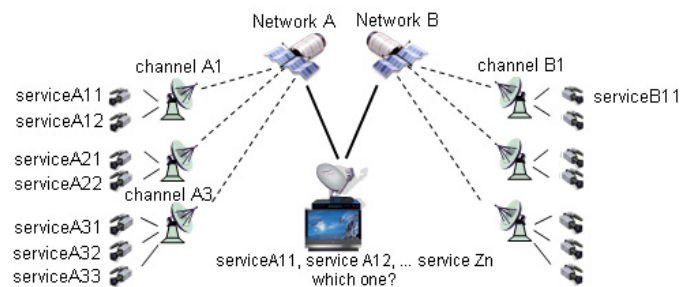


Figure 2.7. The large number of services received by the end-user in digital TV

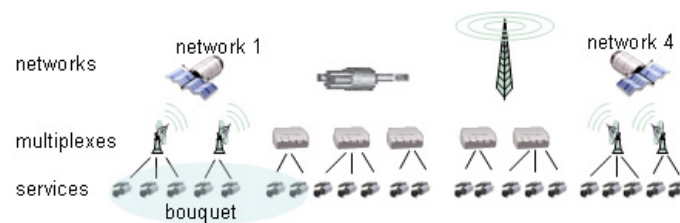


Figure 2.8. Example of a bouquet in DVB

Table 2.1. The MPEG-2 TS Program Specific Information (PSI) tables

Table name	Description / Purpose
PAT , Program Association Table	PAT information provides the means to associate the program number to the table PMT that carries the definition and additional information about that program. In each multiplex there is only one PAT, which is carried in MPEG-2 TS packets with the identifier 0 (PID = 0x0000).
PMT , Program Map Table	Each program in the multiplex must have its own PMT. This table provides the indication of the PIDs of packets that transport the components of the program it refers to. The PID of the TS packets that carry the PMT data is indicated in the PAT.
CAT , Conditional Access Table	This table carries information required to initiate the process of accessing to encrypted content. When present, it is always carried in packets with PID=0x0001. The information included in the CAT is proprietary (non-normative) and may vary according to the Conditional Access System used by the Service Provider.
NIT , Network Information Table	The NIT provides information describing the physical organization of the multiplexes/TSs carried via a given physical network, and the characteristics of the network itself (ex., RF channel frequencies). The unique identification of networks is accomplished by the assignment of individual network ID values.

Table 2.2. The DVB-SI tables

Table name	Description / Purpose
BAT , Bouquet Association Table	The BAT provides information regarding bouquets. A bouquet is a collection of services associated to a single service entity, which can thus be purchased as a single product. It may traverse the boundary of a network (see Figure 2.8). The BAT is sent in TS pacets with a PID value of 0x0011.
SDT , Service Description Table	The SDT describes a service that is carried in a given MPEG-2 TS. It provides information concerning the existence or not of extra descriptive data about that service (eventually sent in other tables) as for example scheduling information carried in the EIT. It also indicates whether the service is subjected to a conditional access system or not. SDT information is carried in TS packets with PID=0x0011
EIT , Event Information Table	This table carries information describing the occurrence of events in each service, in chronological order. It provides information concerning the scheduling (start time and duration) and status of events or programmes. Based on this information, receivers are able to build the Electronic Programming Guide (EPG). The EIT may also include content descriptors to rovide additional information on the programme such as genre classification. The DVB specifies that it is required to include in the multiplex as a minimum, information concerning the current and the next event in each service. However, Service Providers may include data concerning other future events. EIT data is sent in TS packets with PID = 0x0012.
TDT , Time and Date Table	The TDT table includes the UTC (Universal Time, Co-ordinated) time and the date. This information can be used as a reference by receivers when analysing the scheduling data extracted from the EIT. The TDT is transmitted in TS packets with a PID value of 0x0014.

Table 2.3. The DVB-SI tables (cont.)

TOT , Time Offset Table	This table carries the UTC-time and date information and local time offset. The TDT is transmitted in TS packets with a PID value of 0x0014.
RST , Running Status Table	If an event starts earlier or later than announced, receivers should be notified of that occurrence. The RST provides the means to carry the required information, enabling accurate and rapid updating of the timing status of specific events. It identifies the event for which scheduling changes have occurred, providing an indication of its current status.
ST , Stuffing Table	This table is used to invalidate tables that become obsolete.
DIT , Discontinuity Information Table	This table is used at transition points where the Service Information may suffer discontinuities. It allows to notify receivers of that situation.
SIT , Selection Information Table	The SIT is used to carry information about services and events only when they occur in a partial TS. It provides a summary of the SI data required to describe elementary streams (components of a service) in that partial multiplexed stream.

2.3.10 TVA

The TV Anytime metadata specification, was especially conceived to be used in the final phase of distribution of digital TV programmes. The use of TVA should assist the end-user while interacting with digital TV programming. The TVA metadata is a content-related metadata type in the sense that it describes features and semantics of the content. In addition the TVA metadata also provides metadata to describe the preferences of the user, thus allowing to build user profiles. These can be used in an automated way, enabling the end-user equipment to provide suggestions to the user.

The TVA metadata specification was developed by the TV-Anytime Forum [57] and is a part of a complex standard describing broadcast and on-line services (*Broadcast and On-line Services: Search, select and rightful use of content on personal storage systems, "TV-Anytime"*). The complete technical specification has been produced jointly by three international standardisation bodies, namely EBU, European Broadcasting Union, CENELEC, Comit European de Normalisation ELECTrotechnique, and ETSI, European Telecommunications Standards Institute) based on the work initially developed by the TV-Anytime Forum. It is currently published as a multi-part ETSI standard, where the first part (*Benchmark Features*) [49], provides a list and definition of all features of a complete TVA system. All the eight parts of the standard are available on-line at the ETSI Web site [51]. The

The three main areas addressed by TV-Anytime are *metadata*, *content referencing* (identification and location of content for delivery using the content referencing identifier or CRID), and "rights management" (rights management information, usage rules and security).

TV-Anytime metadata uses descriptors and description schemes from the MPEG-7 standard, notably to describe TV programmes and user preferences.

2.3.11 METS

The Metadata Encoding and Transmission Standard, defines an XML document format based on XML schema, for encoding metadata (descriptive, administrative, and structural) useful for the description and management of digital objects in digital libraries or repositories[138]. Moreover, the standard is meant to also assist the interchange or transaction of objects between different repositories, by defining a common data transfer syntax between repositories or between a repository and tools that process and present the objects. METS is developed under the initiative of the Digital Library Federation and is maintained by the MARC Standards Office of the Library of Congress. In general terms, the METS document format could be seen as providing the definition of a *wrapper* that concatenates digital object(s) in multiple formats. It provides a structure map documenting the organization of the digital object(s), as well as behaviors that act upon digital object(s). A METS document can be used in three different instances, accordingly assuming three distinct roles:

- Submission of digital objects, playing the role of a *Submission Information Package* (SIP);
- Archival of digital objects in the form of an *Archival Information Package* (AIP);
- dissemination of objects as a *Dissemination Information Package* (DIP).

The structure of a METS document is illustrated in Figure 2.9.

The *Header* contains metadata describing the METS document itself with information such as creator, editor, date of creation, etc. The sections designated as *Descriptive Metadata* and *Administrative Metadata*, provide the means to include, either directly or by reference, content-descriptive and administrative metadata respectively. Administrative metadata can be related with the document creation

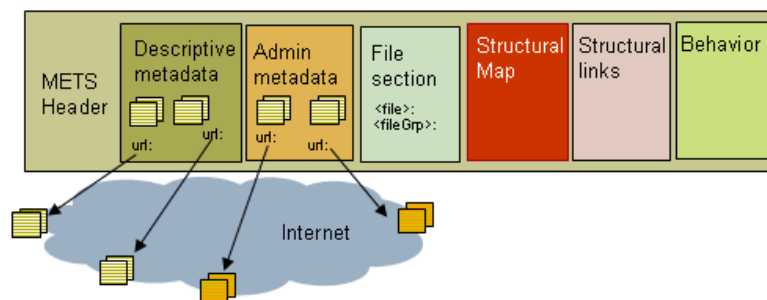


Figure 2.9. The METS structure

and/or storage date, intellectual property rights, metadata regarding the original source object(s) from which the digital library object derives and existing derivative relationships. The *File* section provides the list of all files that contain content components of the digital object. The section *Structural Map* outlines the hierarchical structure of the object, and links the elements of that structure to content files and metadata. The *Structural Links* section records the existence of hyperlinks between nodes in the hierarchy outlined in the Structural Map. The *Behavior* section can be used to associate executable behaviors with content in the METS object.

2.3.12 DCMI

The Dublin Core Metadata Initiative [83], [10] is one of the most widely used resource discovery metadata standards on the Internet. Developed and maintained by an open independent group, its descriptive element set was approved as an official standard by the International Standards Organisation (ISO) under the reference ISO 15836 [2]. It is a standard that provides the means to generate high-level content-related metadata (*the context of the content*).

This standard specifies the set of DCMI elements to be used to describe

resources with the single scope of facilitating the discovery of those resources on the Internet. This set of elements is quite small and eventually limited as the intention of the DCMI is just to provide a simple and yet effective way of locating resources. When this basic functionality is not sufficient, applications can use, together with them, other metadata standards able to fulfill their needs. In fact the aim of the DCMI is not to eliminate the use of other metadata schemes or replace their functionality. Instead it aims at providing the "glue" across systems using distinct standards enabling the establishment of multi-standard environment with a common "*core*"⁹. Figure 2.10 illustrates this concept. In this kind of environment, description records of resources would contain different mixtures of descriptive elements coming from distinct standards but, having always as a common ground, elements from the DC metadata element set.

The DC metadata standard provides two levels of description elements - Simple and Advanced or Qualified. The Simple Level defines fifteen basic elements. The Advanced Level incorporates a number of additional elements and a set of refinement elements designated as Qualifiers. Within the DC metadata scheme, elements assume the role of nouns. Nouns can have attributes which are indicated by the Qualifiers. The resources themselves are seen as the subject to which the nouns apply. The DCMI has also specified a Type Vocabulary containing the definition of a number of general purpose terms (for example, *Collection* defined as "An *aggregation of resources*") and a set of encoding schemes.

DC elements can be grouped into three categories according to the type of subject they refer to or the scope of the information they contain - *content*, *intellectual property* and *instantiation*. Table 2.4 lists the basic elements (defined in

⁹The word "core" in the Dublin Core metadata name refers precisely to this intended property: the set of elements to be defined should provide core functionality with a broad and generic scope of application, thus being cross-application by opposition to application-specific.

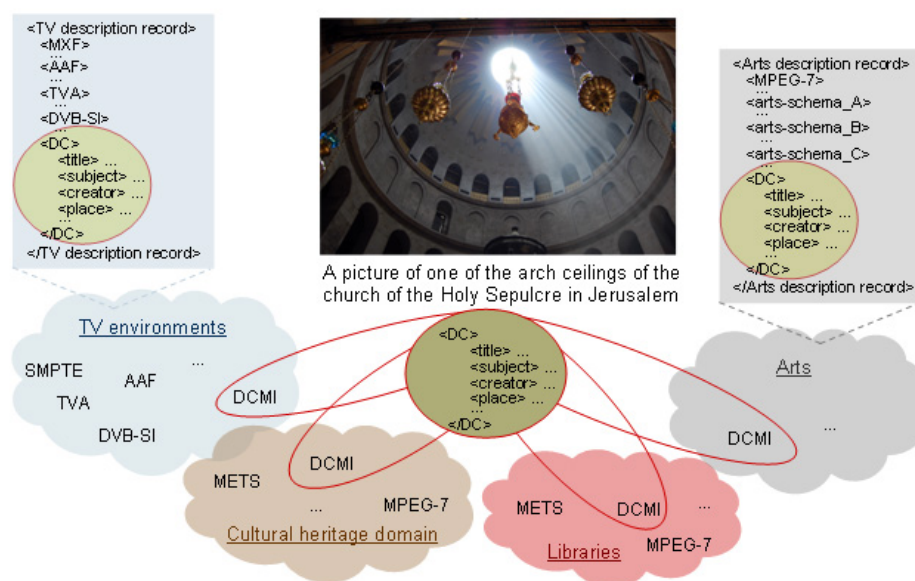


Figure 2.10. The cross-application role of DCMI

the Simple Level) grouped into these three classes and providing a brief overview of their meaning. The use of any element is optional and their ordering is random (not specified). The same element can appear more than once in each description.

A DC metadata record containing any of these elements and possibly elements from the Advance Level as well as qualifiers, can be expressed using HTML, XML or RDF.

2.3.13 RDF

The Resource Description Framework, is a declarative language that provides a standard way for using XML to represent metadata [200], [59]. It enables different metadata schemes to be expressed in the form of statements about properties of resources and about relationships between resources on the Web in a human-

Table 2.4. The DCMI Element set

Group	Element name	Overview
Content	Coverage	This element may describe both <i>spatial</i> and <i>temporal</i> characteristics of the intellectual content (i.e., the raw content represented by the resource being described). Spatial coverage refers to a physical region using names of places names or coordinates (longitude and latitude). Temporal coverage refers to the period the content refers to. For example, a painting from Caravaggio would have a temporal coverage "Baroc". It does not refer to the date when the resource was created but rather to what the resource is about.
	Description	A textual description of the content of the resource. It may include abstracts in case the resource being described is a document-like object, a synopsis of a movie or content descriptions in the case of visual resources.
	Relation	Provides the means to indicate links between resources. It allows to identify a second resource related to the resource being described and to indicate their relationship.
	Source	This element allows to include in the descriptive record of a given resource, metadata about a second resource from which the present resource is derived. It should only be used when it is considered important for discovery of the present resource.
	Subject	Provides a description of the topic of the content of the resource. It can be expressed as keywords, key phrases, or classification codes. Recommended best practice is to select a value from a controlled vocabulary or formal classification scheme.
	Title	The name given to the resource, usually by the Creator or Publisher.
	Type	It describes the category of the resource, such as home page, novel, poem, working paper, technical report, essay, dictionary, etc.

Table 2.5. The DCMI Element set (cont.)

Group	Element name	Overview
IPR	Contributor	Name of a person or entity who is not a Creator but who has made significant intellectual contributions to the creation of the resource. It could be for example the editor of a book or the illustrator.
	Creator	Creator or Author refers to the person or organization primarily responsible for creating the intellectual content of the resource (writers, artists, photographers, etc.) For example, the Creator of a resource containing a picture of the painting "The conversion of Saint Paul", would be "Caravaggio" (or the record description could have two Creator elements, the second with the name "Michelangelo Merisi").
	Publisher	The name of the entity responsible for making the resource available in its present form (e.g., a publishing house).
	Rights	This element could contain a statement concerning rights management, or an identifier linking either to a rights management statement or a service providing information about rights management for the resource.
Instantiation	Date	Provides the date of the creation or availability of the resource. The W3C recommends the use of date forms such as YYYY and YYYY-MM-DD that are defined in standard ISO 8601 [197].
	Format	This element provides information concerning the data format of the resource and, optionally, dimensions (e.g., size or duration). Its purpose is the one of assisting the identification of the software and/or hardware that might be needed to process and present the resource.
	Identifier	A string or number used to uniquely identify the resource. Examples include International Standard Book Numbers (ISBN) for books or URIs (URLs and URNs) for networked resources.
	Language	The indication of the language of the intellectual content of the resource.

understandable and machine-parsed form.

RDF only expresses the structure using XML, relaying on other metadata schemes for the definition of the actual semantics. The use of different metadata schemes inside an RDF document only requires that appropriate namespaces exist and have been declared in the document. This way the semantics of the used description elements can be readily understood by anyone or anything processing the document, by accessing the correspondent scheme. Below it is presented a very simple example of the use of RDF to express DC metadata.

```
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:dc="http://purl.org/dc/elements/1.1/">
  <rdf:Description rdf:about="http://www.artcyclopedia.com/">
    <dc:creator>John Malyon/Artcyclopedia</dc:creator>
    <dc:title>ArtCyclopedia: A Guide to Great Art on the Internet</dc:title>
    <dc:description>
      A Web site devoted to the discovery of Art resources on the Internet.
    </dc:description>
    <dc:date>2007</dc:date>
  </rdf:Description>
</rdf>
```

Resources can be almost anything that is uniquely identifiable by an Uniform Resource Identifier (URI) [85] (in plain terms, "provided that it has a Web address"). This means that it is possible to associate metadata with a Web page (as in the example above), a graphic, an audio file, a movie clip, etc. Accordingly, and although RDF has been essentially used to provide descriptions that fall into the category of *content-related*, it can be used to describe also the components of the context of usage.

The following elements are defined in RDF:

- properties;
- property-types;
- values
- descriptions.

RDF provides a syntax-independent model for describing resources in terms of their *properties* (attributes or characteristics). The properties associated with resources are identified by *property-types*, and have corresponding *values*. Property-types express the relationships of values associated with resources. A value may be an atomic element such as a text string or number, or it can be another resource, which in turn may have its own properties. A collection of these properties that refers to the same resource is called a *description*.

Brief Summary

It is evident that metadata has gained an increased importance over the last years within all areas of the multimedia content value chain.

Research groups have been working intensively in cooperation with standardisation bodies to produce flexible and complete specifications, able to satisfy the requirements of the different application areas. Although the market has been rather compartmented into application areas, such as cultural heritage, libraries or audiovisual production, initiatives have recently started pulling out efforts to merge or harmonise standards, usually relaying on the use of a common universal language, XML (eXtended Markup Language) [208]. As such, the strong competition that has existed so far within each sector in the current metadata marketplace, due to the great variety of application-specific metadata schemes, seems now to be crossing

borders and assuming a more global dimension.

As it was referred, the W3C has issued a number of open specifications that contribute to this globalisation, although its focus has been always on the use of resources available on the Internet. Nevertheless they are generic and flexible enough to be used in any other environment (just consider the example of XML or RDF). In fact, all of the described metadata specifications are based on XML schema, whereas RDF is being regarded as the foundation for metadata interoperability across different resource description communities.

Other standardisation bodies such ISO/IEC through its MPEG working group, are pulling efforts in order to arrive to common and complete specifications that may be able to satisfy the needs of multiple application areas while promoting interoperability across areas and re-usability of technologies. These two bodies seem to be the ones developing specifications with a broader range of application. Coordination between the two could drive to the obtention of better results in the management and sharing of multimedia resources.

The work of these bodies could also be of interest to the traditional A/V production industry, such as TV broadcasting or motion pictures (now entering the digital cinema era). If not by other reasons, at least due to the merging that we are witnessing between the TV and the Internet domains. However it is well known that the traditional A/V industry is quite conservative in respect to the adoption of IT-centric approaches and solutions developed within the Internet domain. Still, the SMPTE is clearly the major player in TV and d-cinema standards and contacts have already been established between this standardisation body and ISO/IEC.

From the above brief overview of different metadata specifications, it was also clear that some synergies, in terms of objectives, exist between MPEG-21 and METS and also up to some extent with MXF. All standards enable the declaration of the structure of complex digital objects. However, whereas MPEG-21 and METS do not define a container for the media resources (the essence in MXF terminology), the MXF specification does. Still, MPEG-21 is in the process of defining a file format to encapsulate all components of a digital object (including resources and metadata). On the other hand it seems that the MPEG-21 has broader objectives as it also provides tools for contextual metadata.

One thing is clear though : metadata has a tremendous importance towards the effective automated sharing of knowledge or of any digital resource, and the role of and need for open and flexible standards, can no longer be questioned. A successful implementation of the ubiquitous access concept will only be possible to achieve if metadata is used and shared in an open way.

Chapter 3

Review of the State of the Art

This section provides an overview of recent developments and standardisation efforts in scientific and technological topics that are related to the objectives of this thesis. The development of quality and context-aware networked audiovisual services in heterogeneous environments, addresses a plethora of topics. We will concentrate on the aspects of context-awareness and content adaptation techniques, which implicitly make use of metadata schemes and also distributed technologies and middleware layers to address interoperability issues. In terms of context-aware content adaptation, it presents established concepts in context-aware computing, context models and existing systems, architectures and frameworks for context-aware content adaptation. It addresses aspects related with the acquisition of context and forms of processing the contextual information and forms of reacting to context.

Aspects related with video compression techniques, metadata schemes and distribution technologies that, as referred above, are likewise essential for the deployment of context-aware networked audiovisual services, have been addressed in the previous section. As far as this thesis is concerned, they constitute background information in the sense that they have been studied and are used in this work but

they haven't really been subjected to research.

This section also includes a review of the topic of statistical video analysis and multiplexing. This topic is directly related with the first initial scope of the thesis, targeting the efficient use of channel bandwidth and network resources.

3.1 Context awareness and Content adaptation

Ultimately, content adaptation is all about the fulfilment of the concept *Universal Multimedia Access* (UMA). This concept has existed for many years. It has always been present in the minds of researchers who have devoted their efforts to the development of increasingly better, faster and efficient encoding schemes for multimedia content, network protocols and integration technologies. However it was only recently that this concept gained life of its own [155], [54], [195]. UMA, the Universal Multimedia Access, addresses the delivery of multimedia resources under different and varying network conditions, diverse terminal equipment capabilities and specific user or creator preferences and needs.

Why is adaptation needed? The many reasons for adaptation can in fact be melt down to one association: heterogeneity and globalisation. Our world is so diverse in itself and people and their consumption habits as well their inventions clearly reflect this diversity. As technology continues to evolve at an accelerating rate in the continuous search for better and faster solutions, hardware devices, computing systems and software will remain diverse and heterogeneous. There will always be the co-existence of different devices aiming the same field of application but presenting diverse capabilities. On the other hand, these technology advances have created the conditions for the transport and delivery of multimedia content across heterogeneous networks and its consumption in a variety of receiving devices with different processing and presentation capabilities. Increasingly more consumers equipped with

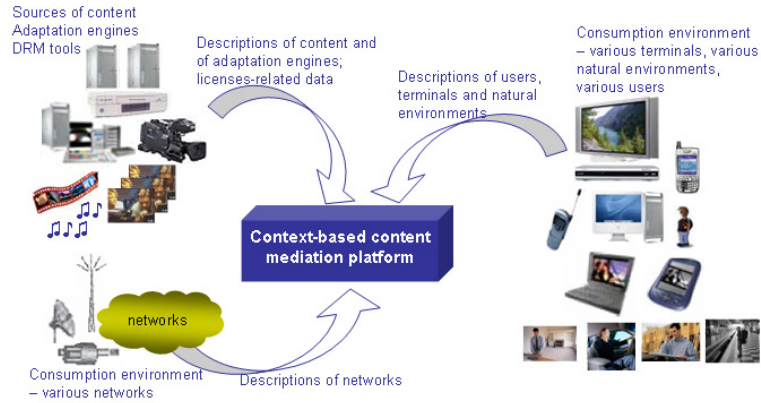


Figure 3.1. The role of context-aware content mediation platforms

different devices and having diverse interests and preferences *navigate* in the Internet or get connected to some network looking for different types of multimedia information.

This heterogeneity scenario and the role of adaptation-enabled content mediation platforms is illustrated in figure 3.1.

Adapting content can be seen as an efficient solution to allow the consumption of a (more or less) meaningful variation of that content, under network and environment conditions which prevented the previous consumption of the original content [41], [25], [26]. Adaptation is an operation performed upon the content with the purpose of modifying one or more of its features. This adapted version of the content, a variation of the original content, will thus have the adequate characteristics to allow its consumption on a particular device under particular network conditions and possibly satisfying specific user needs and preferences. Adaptation thus requires the availability of engines capable of manipulating and transforming the content. But a successful implementation of adaptation operations, and consequently the suc-

successful delivery of the UMA concept to end-users, requires also the availability of systems interoperating towards a common objective and of meaningful descriptions about the content and the environment and/or conditions under which the content will be consumed, the context. Those systems must be able to generate, collect and interpret these descriptions and accordingly make decisions regarding the need for and the type of adaptation.

Content adaptation has already gained a considerable importance in today's multimedia communications and will certainly become an essential functionality of any service, application or system in the near future. The continuing advances in technology are steadily emphasizing the great heterogeneity that exists today in devices, systems, services and applications. And likewise, it is bringing out the eager of consumers for more choice, more quality and more personalisation capabilities. Whereas research regarding techniques and approaches to operate varied forms of transformations upon the content is in a quite advanced state, the assessment of the degree of satisfaction of the users towards the delivered service is still a rather unexplored field. In order to provide effective content adaptation services that meet the real expectations of the users, it is essential to gain a deep understanding of context and be able to use it as efficiently as possible.

3.1.1 The use of context

Context-awareness can be defined as the ability of systems, devices or software programmes, to be aware of the characteristics and constraints of the environment and accordingly automatically perform a number of actions/operations to adapt to changes of the sensed environment.

The use of contextual information is instrumental for the successful implementation of useful and meaningful content adaptation operations. These, in turn,

are becoming extremely important for the implementation of systems and platforms that deliver multimedia services to the end-user. In the complex scenario that exists today in the multimedia content value chain, the possibility of automatically adapting content and applications is thus assuming increasing importance.

But to empower those systems to perform meaningful content adaptation operations that meet users expectations and satisfy their usage environment constraints, it is imperative that they use contextual information and take decisions based on that information. Indeed the implementation of content adaptation operations that meets users' expectations and satisfies their usage environment constraints, requires the use of contextual information to take decisions on how and when to adapt content. This section provides an overview of recent developments concerning the use of contextual information.

Services and platforms granting the access to content must be able to provide the intended value to its users independently of their location, type of terminal devices or networks being used or regardless of the format of the content. In addition, they should also respect user preferences, environmental conditions and content owners and usage rights.

One fundamental aspect to be considered, besides the availability of suitable descriptions about the content and enabling true interoperability, is the availability of descriptive information about the context of usage. Building context-aware applications enables the adaptation operations to be more meaningful and to better serve the interests and needs of end-users and applications.

Research on context awareness started more than a decade ago [16]. However it was only recently that this concept gained increased popularity among the multimedia research community and started to be addressed also at the standardisation level.

Research work around context-aware services was initiated especially in the mobile applications area. It was mainly focussed on processing information regarding the location and the type of device being used to receive and present the content. The work evolved, not only in the mobile communications area but also to other areas. Other types of information, collected through varied types of sensors, started also to be used. One example is the area of Human Computer Interfaces (*HCI*) where, information regarding the user such as, genre, age, user's emotions or user's disabilities, as well as environmental conditions, are used to adapt the application's interface to particular usages.

What is context-awareness?

Many different definitions can be found for context-aware applications, but the majority clearly relates them with adaptation operations. Accordingly, we can formulate the following definition of context, which agrees with most of the research work being developed in the last years:

Context-aware applications are those having the ability to detect, interpret and react to aspects of user's preferences and environment characteristics, device capabilities or network conditions, by dynamically changing or adapting their behaviour based on those aspects that describe the context of the application and the user.

Early definitions of context were limited or specific to a given application, as they were usually made by examples of the type of context information being used.

A more generic definition of context-aware applications can be used, without making any explicit reference to adaptation capabilities. Quoting [163]:

'A system is context-aware if it uses context to provide relevant information and/or services to the user'

Whereas this definition is sufficiently generic to include the previous definition or other possible ones, for the purpose of our work we will be in fact interested in the first one. In fact, the objective is to study forms of using contextual information to assist and enhance adaptation operations to ultimately improve the quality of the user experience. So being, this thesis will focus on the use of context to react to the characteristics and conditions of that context and accordingly trigger some kind of adaptation operation.

Nevertheless, the above definition is more generic as it does not depend on the type of additional information being used to deliver the service, but rather on the effects that information may cause on the behavior of the services and applications.

A. Dey [39] provides a good generic definition of context that can be applied no matter the type of information being used or the application in view. This is probably the most quoted definition of context:

Context is any information that can be used to characterize the situation of an entity. An entity is a person, place, or object that is considered relevant to the interaction between a user and an application, including the user and application themselves.

This definition implicitly states that any application using additional information in grade to condition the way the user interacts with the content, is a context-aware application.

What kind of contextual information is there?

Contextual information can be any kind of information that characterizes or provides additional information regarding any feature or condition of the complete delivery and consumption environment. Based on previous research [60], we can group this diversity of information into four main context classes: *Computing*,

User, Physical and Time.

The first group addresses the characteristics, capabilities and conditions of the resources that can or need to be used to provide the context-aware service. Although the word *Computing* could be interpreted as referring exclusively to the terminal device, in fact it has a broader scope as it refers to all resources involved in the delivery of the service - the terminal where the content is to be consumed, the network through which the content is to be delivered and any additional service or engine that may provide added value to the multimedia service. Accordingly we will rename this group to *Resources* context.

Below we provide some examples of contextual information falling into each of these four categories.

- *Resources Context* - description of the terminal in terms of its hardware platform, including any property such as processor, screen size or network interface; description of the terminal in terms of its software platform, such as operating system, software multimedia codecs installed or any other software application; description of the network such as maximum capacity, instantaneously available bandwidth or losses; description of multimedia servers, for example in terms of maximum number of simultaneous users or maximum throughput; description of transcoding engines in terms of their hardware and software platforms such as network interface or input/output formats allowed or bit rate range supported;
- *User Context* - description of the user general characteristics such as gender, nationality or age; description of preferences of the user related with the consumption of content such as the type of media or language preferred; description of the preferences of the user in terms of his/her interests, i.e., related with the

high-level semantics of the content, such as local news versus international news or action movies versus comedy; description of the user's emotions such as anxious versus relaxed or happy versus sad; description of the user status, such as online versus offline or sited versus walking; description of the history/log of actions performed by the user;

- *Physical Context*- description of the natural environment surrounding the user, such as lighting and sound conditions, temperature or location;
- *Time Context* - indication of the time at which variations in the context have occurred or scheduling of future events.

Besides these different types of contextual information that together can describe any entity involved in the delivery and consumption of the multimedia content, another dimension of the contextual information should be considered. This other dimension refers to the characteristics and nature of the contextual information itself, and not to the type of entity that it describes. In this dimension the following aspects should be considered:

- the *accuracy* or *level of confidence* of the contextual information;
- the *period of validity*. Contextual information may be static, thus having an unlimited validity period, or it may be dynamic and valid for only a given period of time. For example, general characteristics of the user are obviously static and do not required any additional information to be used. On the other hand, user's emotions will essentially be dynamic. Likewise, conditions of the natural environment such as lighting or background noise will also be dynamic.
- the *dependencies* on other type of contextual information. Reasoning about one type of contextual information may depend on other types of contextual

information.

Types of sensors

Contextual information is generated by sensors. Any hardware appliance, software component or entity capable of generating information describing some aspect of the context of usage can be designated as sensor. Sensors can be classified according to its nature, based on their mode of operation and according to the level of delivered functionality [106]. We present here this three-perspective categorization approach, for the sake of completeness. However, in our work we do not consider the last perspective. The reasons for that are provided below.

Traditionally sensors are seen as some hardware equipment that provides a measurement or information concerning some aspect of the context. For example, a camera can constitute a light sensor, providing indication of the level of brightness of the environment surrounding a user in a multimedia service. Other examples include microphones, Radio Frequency IDentification (RFID) badges, optical fiber vibration sensors, etc. This type of sensors are designated as *Physical Sensors*.

On the other hand, software components or systems are designated as *Logical Sensors*. Examples include low-level software modules installed in a PC that measure the amount of available memory or the actual throughput at the Network Interface Card (NIC) input.

These two types of sensors can be further distinguished by their mode of operation, being classified as *Passive* or *Active*. The former type of sensor, generates a measurement of some characteristic of the context, only when stimulated by some input signal. This input signal could be, for example, a sound with an intensity

higher than a given threshold. The latter, periodically generates a probe signal that enables it to detect changes in the context. Examples of active sensors are the radar or radio frequency sensors.

In any of these categories, sensors can provide directly their measurements or sensed context, in which case they are classified as *Simple*; or they can operate some kind of processing before passing the acquired information, being designated as *Smart*. We have however, some concerns regarding this last categorisation approach. We believe that it is wiser to de-correlate any processing functionality from the actual sensing capability. Sensors with embedded processing, will risk to lose their generality as they will perform some modification upon the sensed context that may suit only some applications. We believe that any processing functionality should belong to the application level, thus making all sensors application-independent and accordingly usable in any situation. Required processing, to meet specific requirements of different applications, should be implemented in the form of plug-ins by each different application. Still, the same plug-in or processing functionality may suit different applications' requirements and thus be used across applications. In any case, this approach allows sensors to be cross-application or application-independent. This concept is further explored in the next sub-section.

Using context

Most of the different types of contextual information described above, can be seen as low-level or basic contextual information, in the sense that they can be directly generated by some software or hardware appliance. Based on this basic contextual information, applications may formulate higher-level concepts.

This is precisely the meaning of context-awareness: *to make use of sensed context to build understanding at higher levels of conceptual abstractions.*

Usually, the knowledge and concepts that can be extracted, greatly depend on the application at hand. Concepts or knowledge that can be inferred through reasoning about the sensed context, are referred to as higher-level or implicit context.

From the several examples given in sub-section *What kind of contextual information is there?* , clearly only some can be seen as of high-level contextual information. This is the case, for example, of some descriptions within the User Context group, such as those concerning the emotions of the user or his/her physical state. This information can be inferred by analysing low-level contextual information obtained by imaging and sound sensors, for example. Location information, belonging to the Physical Context group, can be a low-level contextual information when expressed, for instance, by geographic coordinates. But it can also be a high-level context when referring to the type of physical space the user is in (e.g., indoor versus outdoor, train station, football stadium, etc.).

Context-aware applications must thus initially acquire the explicit or low-level contextual information and then process it and reason about it, to formulate concepts and make decisions on if and how, to react. Furthermore, the acquisition of context, at least of some types of contextual information, will be a continuous, periodic or not, process so that changes in the context may be perceived by the application. Of course the process of reasoning about the basic contextual information will also be a continuous process, that will be conducted whenever changes in the basic contextual information are detected. These three phases are sometimes des-

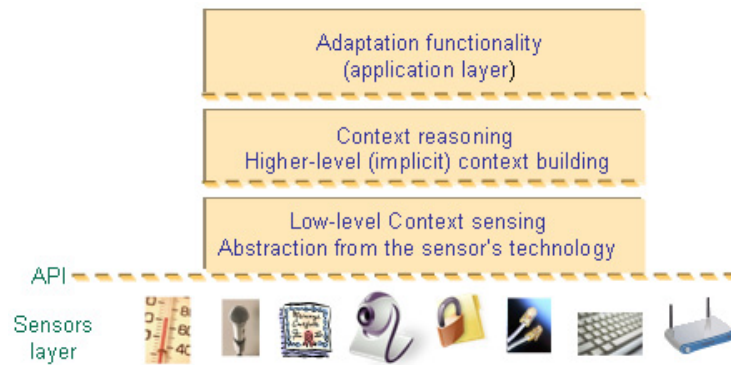


Figure 3.2. Concept of a generic functional architecture

ignated as *sensing the low-level context*, *generating higher-level context* and *sensing the context changes*.

Figure 3.2 presents the conceptual architecture of a platform that is able to implement these three phases and ultimately interact with content adaptation engines. This is a layered platform that enables combining multiple explicit contexts to build elaborated implicit contexts in an interoperable way. The external lower layer is instrumental to enable interoperability at the system level:

- Applications can be developed without the need to be aware of the details of the sensor devices in use and are thus independent from sensor technology;
- Different applications may use the same sensors and may make different uses of the same sensors and corresponding low-level sensed information;
- Sensors can be distributed and thus applications may profit from using explicit contextual information gathered in remote points.

Each layer can be further divided into functional modules. For example, the low-level context sensing layer may incorporate different modules acting as services, offering functionality to collect different types of low-level contextual information. Likewise, the context reasoning layer may provide various modules that reason about different sets of low-level contextual information. The existence of particular modules will be dependent on the type of adaptations offered by the application layer, which in turn should be dictated by the requirements of the application scenario.

Considerable research work has been conducted worldwide around context-awareness in various areas. One of the main challenges faced in context-aware computing has been, until recently, the lack of standardised models to represent, process and manage contextual information. Using proprietary formats and data models does not allow different systems to interoperate, and seriously limits the use of context in distributed heterogeneous environments.

Although more recently, standardisation bodies, such as the W3C (World Wide Web Consortium) and MPEG (Motion Picture Experts Group), have started to work on specifications to represent and exchange context, the mechanisms that are provided for establishing relationships among contextual information and constraints are still very limited. They provide efficient frameworks for developing moderately simple context-aware systems. However, they do not answer the needs of more complex and demanding context-aware applications, failing in providing the required support to represent additional information about context as referred above or for reasoning about the context. (Major standardisation efforts on-going in the area of context representation will be described in this section further ahead.)

Consolidated and trustworthy knowledge and high-level concepts are not easily obtained. It is not enough to just use as much sensed context as possible. It is necessary to know how to combine that information and how to extract conclusions depending on the application, on the recent past conditions and on the type and accuracy of available information. The same information may lead to different conclusions.

It is thus necessary that the system that is analysing the sensed context has this kind of additional knowledge [158]. Accordingly, reasoning about the sensed context usually requires the use of models with rules and relations, to allow combining and relating explicit context according to the application and situation. This concept is illustrated in Figure 3.3. Nevertheless, this may not be required in very simple situations. For example, in applications with a limited number of low-level contextual sources or adaptation alternatives. Or even when working with general-purpose sensors, that deliver data that is used essentially the same way across applications.

The use of ontologies and formal models for context

In an attempt to overcome the above referred limitations, researchers have been studying and proposing the use of ontologies to build context-aware systems. Ontologies provide the means to establish relationships between contextual information reasoning and formalising concepts. They thus enable to build complex context models based on knowledge. Moreover, the use of ontologies based on open formats and with well declarative semantics can be the vehicle to achieve interoperability, enabling different systems to reason about the available information, using the part relevant to their contexts.

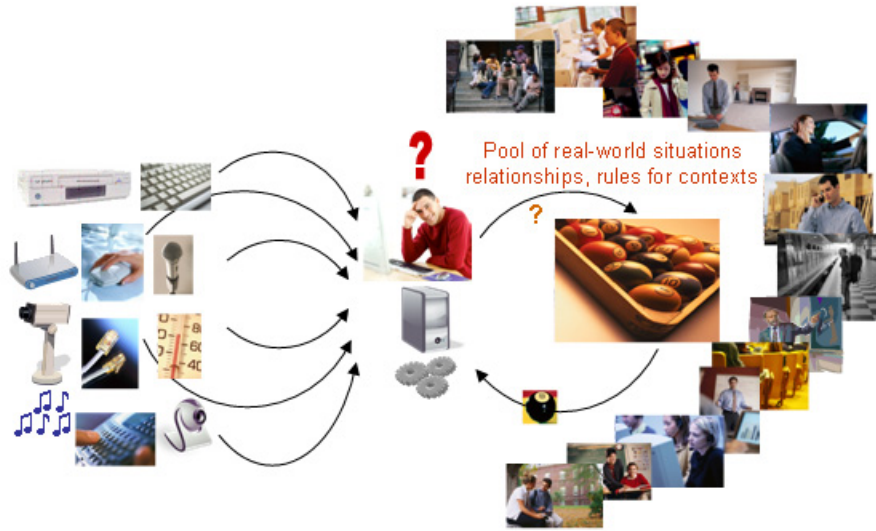


Figure 3.3. The need for higher level contexts

An ontology can be seen as collection of terms from a vocabulary providing the formalism to represent models of real world situations or domains with extended expressiveness. Its aim is to provide a common knowledge about domains, contexts or situations by formally describing the concepts and their relationships. In addition it should enable to apply reasoning techniques upon those models.

An ontology is meant to capture and convey into a formal model, common understanding and knowledge about a given domain and to enable automated processing or manipulation of that model. In practice, an ontology is a hierarchical description of a set of concepts, a set of properties and their relationships, and a set of inference rules [166].

A context ontology should be:

- *simple*;
- *flexible* and *scalable*, to easily extend for new requirements;
- *generic*, to be able to be used in various applications and contexts;
- *expressive*, to be able to represent complex situations.

According to [3] an ontology is *an explicit specification of some topic. It is a formal and declarative representation which includes the vocabulary (or names) for referring to the terms in that subject area and the logical statements that describe what the terms are, how they are related to each other, and how they can or cannot be related to each other. Ontologies therefore provide a vocabulary for representing and communicating knowledge about some topic and a set of relationships that hold among the terms in that vocabulary.*

[77] refers to an ontology as *an engineering artefact consisting of a vocabulary that describes a view of some domain; an explicit specification of the intended meaning of the vocabulary with a classification of concepts; and constraints that capture additional knowledge about the domain.*

According to the IEEE [67], *an ontology is similar to a dictionary or glossary, but with greater detail and structure that enables computers to process its content. An ontology consists of a set of concepts, axioms, and relationships that describe a domain of interest.*

Perhaps the most generic, as well as the most accepted definition of ontology is the one provided by Gruber [68]: *an ontology is a formal specification of a*

conceptualization.

Ontologies are being seen as the next level towards the fulfilment of the Semantic Web paradigm. As envisioned by the creator of the World Wide Web, Tim Berners-Lee [18], the goal of the Semantic Web or Web 2.0, is to achieve complete interoperability and seamless (automated) use of content or any resource available on the Web. To attain this objective, new tools and languages that provide sufficient expressiveness to describe resources in forms that may be usable in an automated way, are required. In fact the W3C has already developed specific ontology languages such as DAM-OIL (DARPA Agent Markup Language - Ontology Inference Layer) and OWL (Web Ontology Language) [127], to extend the features of their descriptive specifications such as XML [208], RDF (Resource Description Framework) [200] and RDF-S (RDF Schema) [19]. These specifications will be further described in section 3.1.3 when addressing standardisation work.

In the context of the Semantic Web, ontologies are foreseen as the vehicle to achieve seamless use of resources, enabling to automate the processing of and access to data. By providing machine-readable structured vocabularies, with definition of terms that characterise the available resources, as well as the description of relationships between those terms, it will be possible for software programs to interpret their meaning in an unambiguous way and thus infer higher-level concepts, matching real-world situations or data. Descriptive, semantic markup records that describe resources accessible on the Web, may include the terms defined in ontologies, thus closing the processing loop [205].

Although the aims of the Semantic Web do not explicitly refer to context-awareness, as already explained earlier in this manuscript, this is also an important

ingredient towards the seamless and universal use of resources available on the Web. Accordingly, the use of ontologies can be extended to context-awareness. In fact, the use of ontologies to provide efficient and comprehensive models to build interoperable context-aware systems seems indeed to be one of the most important aspects being addressed currently by researchers working in this area.

L. Ferreira Pires et. al [158], discuss the suitability of using ontologies for modeling context-aware services platforms. They investigate the use of Semantic Web technologies [207] to formalize the properties and structure of context information to guarantee common understanding.

The CoOL (Context Ontology Language) is an example of a context-oriented ontology approach recently developed. T. Strang et. al. [178] describe this context ontology language derived from their proposed approach to model context. They develop an Aspect-Scale-Context (ASC) model, where each aspect can have several scales to express some context information. The ASC model can be very interesting to implement Semantic Services discovery using context.

P. Yeow [211] presents an approach to context mediation by developing a model using the OWL (Web Ontology Language) [127]. The idea is to provide a common open model that can promote interoperability between systems using context information and operating in heterogeneous environments, such as the Web.

D. Preuveneers et al [37] propose an extensible context ontology to build context-aware services and allow their interoperability in an Ambient Intelligence environment. Their proposal is to define a core ontology based on OWL and built around 4 major entities, among which the user assumes the central role. This ontology defines concepts and their relationships for each of those four entities, trying to address commonly accepted basic needs of the majority of context-aware applications, but providing the flexibility to scale to new requirements.

Jun-Zhao Sun and J. Sauvola [181] define a context space and formulate a conceptual model for context-aware services. The idea is to build a methodology for designing context-aware services, providing guidelines on the selection of context information and adaptation alternatives. The model can also be used to assess developed context-aware services.

G. D. Abowd [71] developed work for context awareness applications in the area of Pervasive Computing. His research group at the College of Computing at Georgia Institute of Technology was involved in many different projects concerning context-aware applications in different domains, including personal location-aware services and context-aware services for the elderly people in the home environment. The driver of this group's research is the development of intelligent services for the elderly and/or physically impaired people. The initial focus of context-aware research was on the use of location information to build the context-aware services. The work evolved to the design of more generic platform to assist the implementation of context-aware applications. These outcomes will be further referred in the next section, when describing the Context Toolkit.

Architectures and Frameworks for context-awareness

In parallel to the recent activity on the use of ontologies to develop meaningful and efficient models to support the mediation of context, research groups have also engaged in the development of generic architectures and frameworks, in many cases using also the ontology-based model approach, to support the implementation of context-aware applications.

M. Keidi and A. Kemper [130] have investigated flexible architectures using Web Services technologies, to allow to dynamically extend the type of context information without the need to introduce modifications to the systems providing

the context-aware applications. They built a platform where all the pre and post processing of the context information is done by context plug-ins and context services, which communicate with the core system via Web Services. They defined their own XML schema to specify the syntax and semantics of the context information passed onto the platform. They did not contemplate the problem of the quality of the service nor the use of feedback to assess the results.

T. Chaari, F. Laforest and A. Celentano [186], also investigated the impact of using context information onto the core system delivering the services. They propose a flexible architecture that can easily be extended to use context information. They developed their work within the framework of the SECAS project (Simple Environment for Context-Aware Systems). The project aims at defining forms of capturing context and passing it to the application that will then use it to perform adaptation operations to suit the context of usage. One of the main concerns is to evaluate the impact of using contextual information at the application level. The SECAS architecture is illustrated in figure 3.4. This figure was obtained at the url http://www.dsi.unive.it/auce/docs/celentano_pcc06.pdf.

SECAS uses context brokers in charge of collecting contextual information, a context provider module that encapsulates the contextual information in an XML document and a context interpreter that maps low level contextual parameters values into a high-level representation. While this architectural approach is interesting from the view point of isolating the higher layers of the application from the modules that actually retrieve and process context, it requires each application or service to register itself, indicating what type of contextual information will be of value to it. This requires that applications are aware of the type of contextual information that can be generated and assumes that there are rather static relationships between

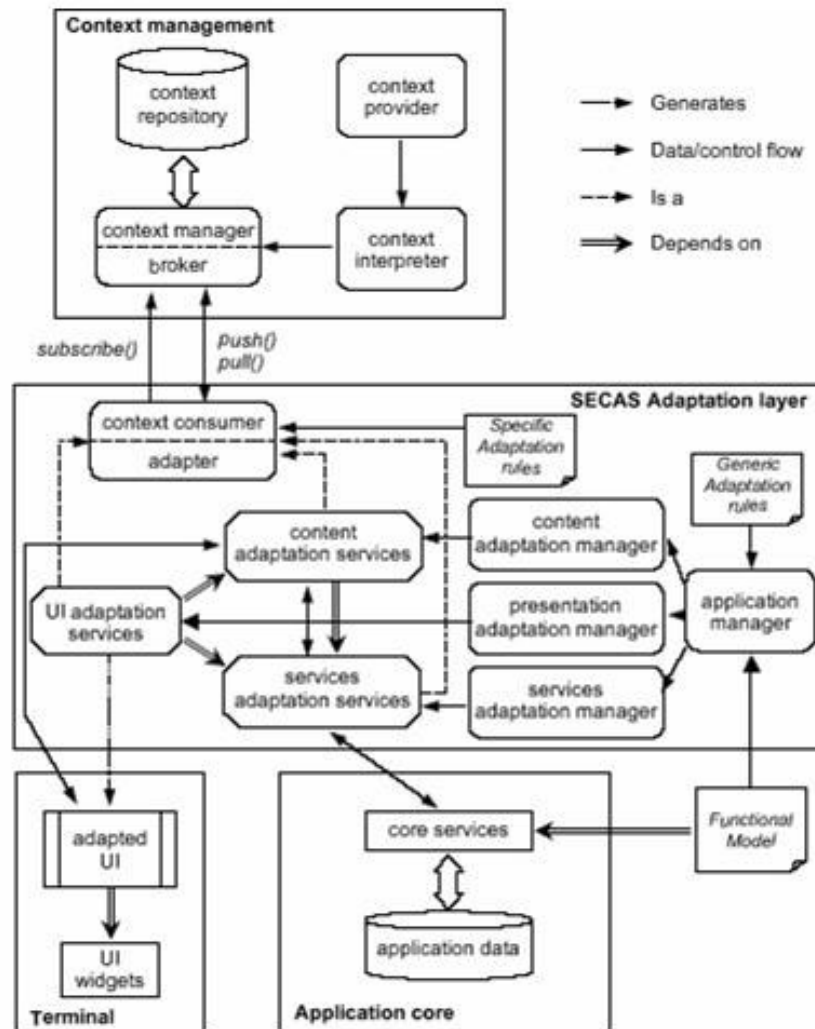


Figure 3.4. The SECAS functional architecture

context characteristics/conditions and the service being delivered. It somehow limits the flexibility of the application to react differently according to user profiles, for example, or even according to the type of content being used by the service or the available adaptation mechanisms. On the other hand, part of these limitations could be overcome by allowing applications to frequently update their registration parameters. However, this does not contribute to the performance of the system and still does not account for personalised adaptation.

D. Preuveneers et al report in [109] and [38] the work being conducted by their research group, the Distributed Systems and Computer Networks ("DistribNet") group of the Department of Computer Science at the Katholieke Universiteit Leuven. The group is involved in several research projects in this area, among which the projects CoDAMoS (Context-Driven Adaptation of Mobile Services) and CROSLOCIS (Creation of Smart Local City Services). In the first project they developed a middleware framework, designated as CoGITO (Context Gatherer, Interpreter and Transformer using Ontologies) [38], to allow the creation of context-aware mobile services in Ambient Intelligence environments, where it is assumed that a user is surrounded by other multimedia-enabled devices. Figure 3.5 shows a simplified view of the CoGITO's architecture.

The CoDAMoS's work focuses on the possibility to transfer services to nearby devices or to replace components to save resources. The middleware includes a specific context-aware layer that detects changes in the context of the user and accordingly initiates service re-location and/or replacement of components. In CROSLOCIS the main focus is the development of a service architecture that enables the creation of context-aware services, providing the mechanisms to collect, safely distribute and use contextual information respecting privacy issues.

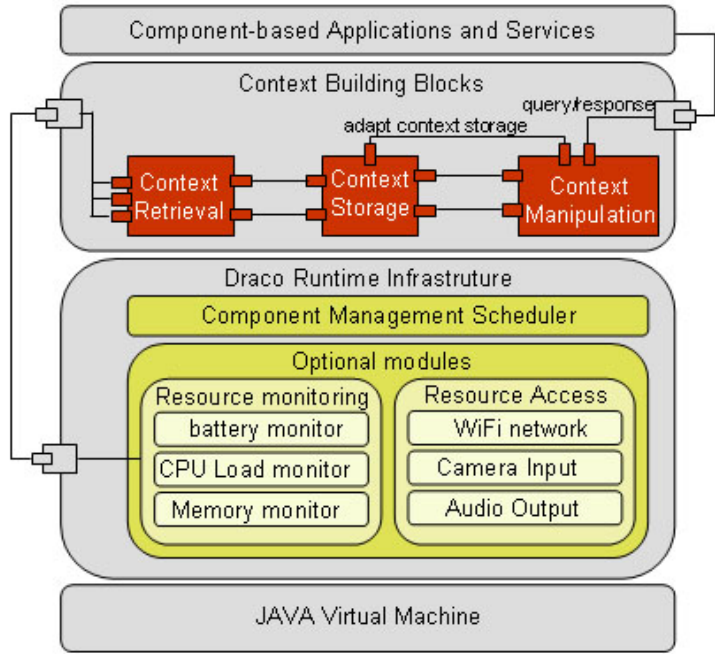


Figure 3.5. Simplified representation of the CoGITO middleware

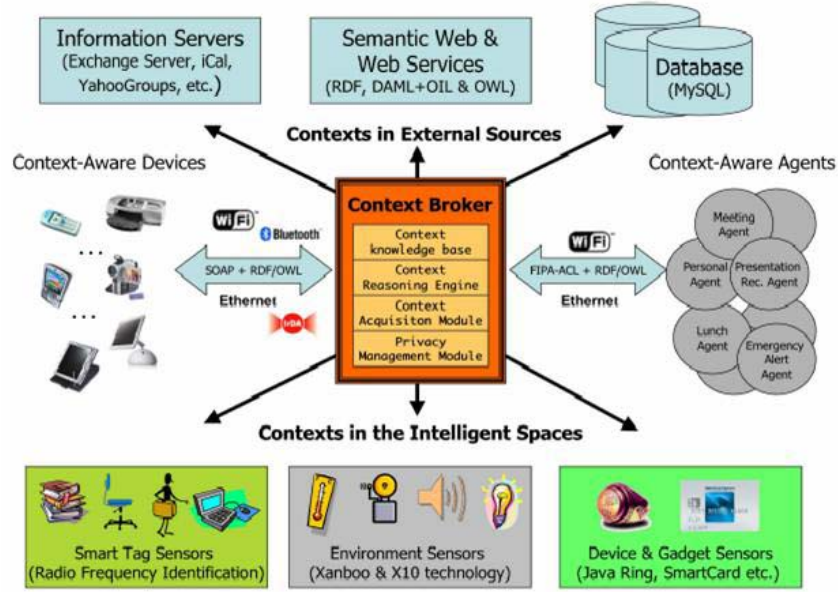


Figure 3.6. The use of CoBrA for context-aware applications

The Context Broker Architecture (CoBrA) project developed by the "ebiquity" group at the University of Maryland [130], is an agent based architecture that supports context-aware systems in smart spaces. It uses the Web Ontology Language (OWL)[127], [128], to define a specific ontology for context representation and modelling, enabling to describe persons, intentions and places. Figure 3.6 illustrates the use of CoBrA for the development of context-aware applications, showing its layered architecture.

An intelligent agent, designated as *context broker*, maintains a model of the context that can be shared by a community of agents, services, and devices present in the space. The system provides mechanisms to ensure privacy protections for the users. The project especially focuses on the intentions of the sensed users, and not

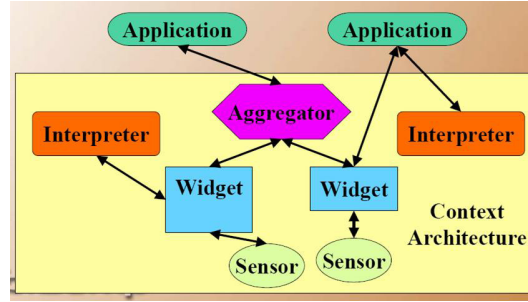


Figure 3.7. The functional blocks of the Context Toolkit

on describing systems or devices characteristics and capabilities.

The Context Toolkit [192] is a framework developed to assist the implementation of context-aware applications. It offers a distributed architecture that interconnects a series of software appliances, designated as context widgets. These software components work as plug-ins to contextual information sources, thus enabling applications to access different types of contextual information. They can be seen as services providing a uniform and seamless interface to various sensing devices, isolating the applications from the specificities of the context sensors. Some of the services offered by the Context Toolkit and implemented through the widgets, are the access to network-related contextual information through a network API or the storage of contextual information. Figure 3.7 presents the conceptual architecture of the Context Toolkit in terms of its functional components. This figure was obtained at the URL <http://www.vs.inf.ethz.ch/events/dag2001/slides/anind.pdf>.

The SOCAM framework (Service-Oriented Context-Aware Middleware) [69] defines a service-oriented architecture for building high-level context-aware applications using an ontology-based reasoning based on W3C specifications. The SOCAM platform is based on four major modules:

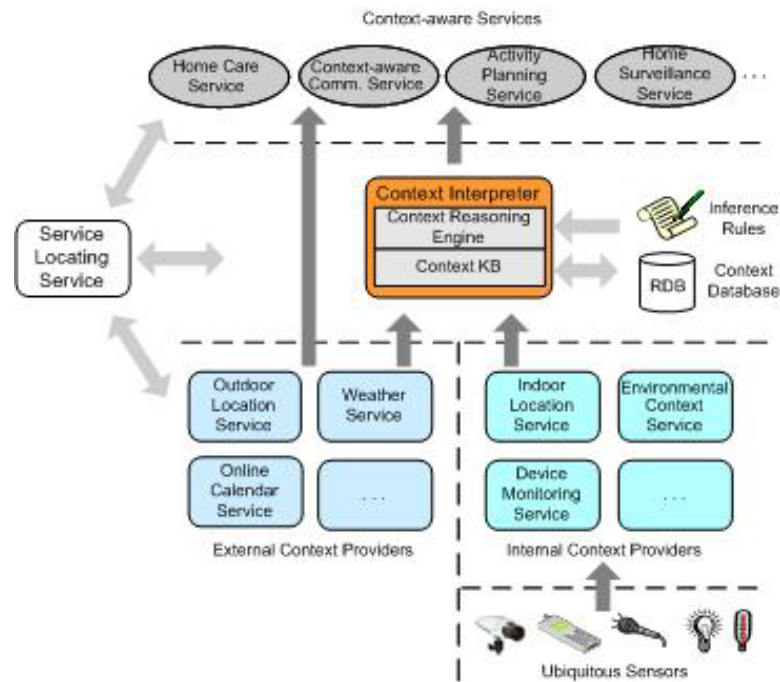


Figure 3.8. Components of the SOCAM architecture

1. a context provider;
2. a context interpreter;
3. a context database;
4. a service discovery engine.

Figure 3.8 presents the different components that integrate the SOCAM platform.¹

In SOCAM, the role of the context provider is to abstract the applications that process the contextual information from the devices that actually sensor the

¹Figure obtained at http://www.comp.nus.edu.sg/~gutao/gutao_NUS/SOCAM.htm.

low-level context. Applications can access a context provider service by querying the service discovery engine. The context interpreter conducts logic-reasoning to obtain a higher-level context, using a context ontology stored in the context database.

Several other research groups worldwide have and still are developing efforts to build intelligent systems that produce contextual information and react to context changes, focussing on designing middleware layers that allow interoperability with multiple external sensing devices. Such is the case of the project Oxygen at MIT Media Labs [129], the project CoolTown at HP Research [142] or the IBM project BlueSpace [164].

Most of the context-aware projects have been developed within the mobile services application area. They have primarily tried to explore context to improve usability aspects by sensing how the available devices were being used. Generally they reacted directly to the sensed low-level context. They usually lacked flexibility as they did not make use of ontologies, or made a rather static and restricted use of ontologies. Therefore, they did not explore the inter-relationships among different types of low-level contextual information and did not sufficiently address the aspects of interoperability, scalability and security/privacy. In fact, earlier research was typically application-centric, overlooking aspects of gathering of different types of contextual information from different spaces and interoperability in heterogeneous environments. Likewise, the aspects concerning security of content and context meta-data and ensuring the privacy of the user, have only recently started to be addressed. The new generation of projects is now focussing on these aspects relying on the use of standard ontology specifications, context representation frameworks and middleware layers. Such is the case of the CoBrA and SOCAM projects.

Summarising the current state of research in context-awareness: first gen-

eration of context-aware systems was mainly concerned with improving usability aspects using directly individual low-level or explicit contextual information, among which location assumed a central role. Second generation systems focussed on abstracting the application from the specificities of the process of sensing the low-level contextual information. They also explored the interrelations between different types of low-level contextual information, incorporating functionality to build higher-level or implicit context. This generation focussed on aspects of interoperability at the system level and on the gathering and use of multiple explicit contextual information. Third generation is now focussing on the aspects of inferring more complex implicit contexts using ontologies and taking into consideration security and privacy issues. Interoperability continues to be a central concern. Accordingly, current research is also looking at the aspects of context representation through the development of formal specification frameworks. Common ontologies are also a vehicle to enable interoperability from the semantic point of view. Different applications can use different sets of rules to reason about the same set of low-level contexts and using the same ontology.

Work has also been done regarding the assessment of the degree of quality with which context information is obtained and made available. T. Buchholz, A. Kpper and M. Schiffers [20] introduce the concept *Quality of Context* (QoC) and discuss its importance, especially in cases where context-aware services are being offered by a combination of different entities and collected from diversified sources. It should be possible to judge if a given entity involved with other in the delivery of a multimedia service, as for example a network operator in a news-on-demand service, is providing sufficiently high-quality context information to enable the Service Provider to meet the promises made to his customers.

To fully explore the benefits of using context information and enable further progress in this area, it is essential to acquire deeper knowledge about context. We can define *context-provision*, as a research field addressing aspects to enable the efficient use of context. Aspects such as understanding what context is, how it can be delivered and used and how to design and build the required architectural support. Context can be seen as any type of information that can be used to characterize the action of consuming content and all of its intervenients, be it persons, objects, systems, devices or applications. Without pretending to be exhaustive and conclusive, the following could be a list of subjects that could be addressed within this area of dynamic context-aware services and technologies for context provision:

- types of context information;
- availability of sensors;
- non-static environments;
- technology to extract context information;
- distributed generation and gathering of context information;
- management and ways of using the context information;
- standardised ways of representing the context information;
- dependencies between context information;
- quality, reliability and resolution of context information;
- relevance of context information towards the degree of user satisfaction;
- metrics for assessing the relevance of context and the degree of user satisfaction.

New and sophisticated forms of encoding the content, together with faster hybrid network technologies and integration middleware, have indeed the potential to deliver the content to a larger range of devices, reaching broader audiences. This thesis embraces this concept, by bringing together technologies and methodologies working in each of these fields. Although the precise aspect of content adaptation is essential to enable the ubiquitous access to content, this thesis does not provide research on that area specifically. It rather relies upon already conducted work and focuses on the aspects of context above described and on how to efficiently use technologies already available or under development, to perform the adequate adaptation to specific usage scenarios.

3.1.2 The forms of content adaptation

Adaptation may be performed in many different ways. The type of adaptation itself may differ, depending not only on the type of manipulation that is performed upon the content or metadata, but also depending on the time and place the adaptation is conducted. These factors can be combined between them, leading to a considerably large number of adaptation alternatives.

There are in fact many different ways and levels of classifying the different types of adaptation, but perhaps the one that better relates adaptation to the perceived result is the one that picks up on the type of manipulation that is performed upon the content. In this scheme it is possible to identify the following categories:

- *transcoding* between different formats;
- *scaling*, within the same encoding format but altering the quality by, for example, reducing the bit rate;

- *transmoding*, by transforming one media type into another such as for example a conversion text-to-speech;
- *filtering*, by which some components or parts of the content are removed.

Principally, performing adaptation operations involves the following three major steps:

1. Identifying the characteristics of both content and its context of usage (terminal capabilities, network characteristics and user preferences).
2. Then, deciding on when and what kind of adaptation needs to be applied, determining the correct adaptation parameters.
3. Finally, applying adequate adaptation operations by invoking relevant adaptation engines using the obtained adaptation parameters.

Any system, aiming at implementing efficient context-aware applications that use content adaptation, requires the availability of:

- Useful descriptions about the content
 - Semantic (type/genre of content, etc);
 - structural (bit rate, aspect ratio, encoding scheme, etc) data.
- Useful and comprehensive descriptions about the context
 - Terminal capabilities;
 - network characteristics;
 - user preferences;
 - natural usage environment characteristics;

- capabilities of the adaptation engines;
 - knowledge of the author restrictions and of rules to enforce digital rights.
- Content and metadata repositories or sources, which are easily accessible
 - Infrastructural support to convey all the concerning information and to enable communication between participating systems.

As stated above, adaptation operations can be classified in different forms. Three of the possible classification schemes, which are commonly agreed and used by the scientific community, concern the aspects related to:

- the instant when the adaptation is conducted;
- the location where the adaptation is performed;
- the type of manipulation performed upon the content.

Table 3.1 summarises the different types of adaptation that can be identified according to these three distinct classification schemes. In this table, the content manipulation is the type of classification that is significant from the viewpoint of perceived results. The first two types of adaptation essentially make an impact on the implementation aspects, and thus have to necessarily use some kind of adaptation classified according to the third type.

Table 3.1. Types of content adaptation

Classification scheme	Type of adaptation - main characteristics	
<i>Instant</i>	<i>Static or variation selection</i>	Must be performed at the server side. Several versions of the content are previously generated, trying to satisfy a range of client capabilities and preferences. Requires more storage capacity and incurs more data management effort. Simple processing, requiring limited processing power and complexity to decide the variation to select. No need to restrict to on-line availability of content adaptation engines.
	<i>Dynamic or on-line</i>	Can be performed at whichever location as long as there is capacity for it. Only one version of the content needs to be initially generated and kept in the repository. Requires more processing power. Requires the decision-taking process to communicate with the adaptation engine.
<i>Location</i>	<i>Server-based</i>	The client issues a request to the server, which needs to collect relevant context descriptions. The server is responsible for all decision-taking processes and adaptation operations. Allows either static or dynamic adaptation. May pose some concerns regarding client privacy issues. It increases the workload of (usually) overloaded servers.
	<i>Proxy-based</i>	The client issues a request to a proxy, which then connects to a server on behalf of the client and collects relevant context and content information from both sides. Takes advantage of the bandwidth between proxies and servers, which is usually high. May pose some problems regarding the enforcement of authors' decisions.
	<i>Client-based</i>	Not very common due to restricted processing capabilities. Does not promote the efficient use of network resources. On the other hand, it completely solves the problem of user privacy. <i>Client Knows Best:</i> - it has full information on the terminal's own capabilities, user preferences, access network conditions and all

<i>Location</i>	<i>Client-based</i>	usage context; - always up-to-date information; - it may use information that could not be disclosed to server due to privacy issues; - it needs to receive metadata, but if it is missing, it can always analyse the content; - it allows a fast introduction of new client devices with new capabilities that would otherwise be unrecognised for some time by server systems; - if problems occur after adaptation, location of the problem is obvious, which concentrates bug-fixing responsibility on the terminal.
<i>Content Manipulation</i>	<i>Transcoding</i>	Between different encoding formats.
	<i>Scaling</i>	Within the same encoding format but altering the quality for: - Bit rate (transrating) or the frame rate reduction. - Spatial resolution or colour depth reduction. - Key frame extraction. ⋮
	<i>Filtering</i>	Some components or parts of the content are removed for: - Removal of some objects in an MPEG-4 scene. - Removal of layers in a scalable stream. - Removal of speakers or background noise. - Text removal or outlining. ⋮
	<i>Transmoding</i>	Transforming one media type into another: - Video to image or video to audio or video to text. - Text to speech. - Table to plain text. - Language translation. ⋮

3.1.3 Supporting Technologies

This section describes the recent developments made in relevant international standardisation bodies, namely the W3C with its CC/PP (Composite Capability/Preferences Profiles) specification and ISO/IEC MPEG with the MPEG-7 and MPEG-21 DIA (Digital Item Adaptation) standards. MPEG-21 DIA, in addition to provide the support for standardized basic context descriptors essential for the implementation of content adaptation operations, also provides the means to describe how those descriptors contribute to the authorisation of adaptation operations. MPEG-21 is presently one of the most complete and flexible standards for multimedia content handling, addressing in particular in a comprehensively manner the aspects related with context description and content adaptation.

As such, this section will provide a succinct description of the most relevant DIA tools, indicating their functionality and scope of application. A description of the W3C standards is also included, due to the wide applicability of these standards and due to their considerable importance within the semantic Web paradigm. We start by making reference to MPEG-7, notably its part 5, Multimedia Description Schemes (MDS), given that it is probably the most comprehensive multimedia description standard. It has a wide scope of applicability, in particular in content adaptation applications. MPEG-7 MDS provides support for the description of user preferences and usage history, as well as for adaptation tools. Additionally, the MPEG-21 standard makes use of these kinds of MPEG-7 description tools within the same scope. It should be emphasized that this section does not have the goal to provide a full description of the referred standards. Instead, it aims at highlighting some specific features of each standard that can play an important role for the development of context-aware content adaptation systems. For a full description of each

standard, the reader should refer to official documentation and other publications given in the references' section of this document.

MPEG-7

MPEG-7 [26], [27] is a very comprehensive multimedia description standard that provides the mechanisms to describe information about the content (also referred to as contextual information of the content within the traditional content description environments such as libraries) and about the information contained within the content. The first class of descriptions includes, for example, the author of the content or its creation date. The second class includes in reality three different types of description capabilities: 1) low-level features of the content such as colour histogram, motion intensity or shot boundary detections; 2) structural information related with the encoding parameters such as coding scheme or aspect ratio format (also referred to as media characteristics); 3) high-level semantics describing the nature of the content (for example *quiet open-air scene with bright blue sky and green grass fields*).

Within the scope of content adaptation applications, MPEG-7 plays an important role in describing information about the content, notably the one related with the format of the content, i.e., with the technical parameters with which the content is encoded, the media characteristics. This kind of descriptions can also be used to describe the capabilities of the adaptation engines, as in fact what is necessary to know is the range of encoding parameters in which each specific adaptation engine can work with. Part 5 of the MPEG-7 standard (MPEG-7 MDS) [27] provides, among other, the framework for the description of this kind of features. For example, the possible variations that can be obtained from a given content can be

described using the MPEG-7 variation description tool. Moreover, one of the attributes of this tool, the variation relationship attribute, is able to specify the type of operation that needs to be performed to obtain a specific variation. Types of operations specified include transcoding, which in turn may involve modification of the spatio-temporal resolution, bit rate colour depth, transmoding or summarisation, among others. When the type of adaptation is a transcoding operation, applications may use the tool transcoding hints, which provide guidelines on how to implement the transcoding operation so that quality may be preserved or low-delay and low-complexity requirements may be met.

MPEG-21

MPEG-21 [94], [79] is the ISO/IEC standard currently under its final phase of development within MPEG. It focuses on the development of an extensive set of specifications, descriptions and tools to facilitate the transactions of multimedia content in heterogeneous network environments. The standard is currently divided in 18 parts (but actually composed only by 17, because part 13 no longer exists):

- Part 1 - Vision, Technologies and Strategy.
- Part 2 - Digital Item Declaration (DID).
- Part 3 - Digital Item Identification (DII)
- Part 4 - Intellectual Property Management and Protection Components (IPMP Components).
- Part 5 - Rights Expression Language (REL).
- Part 6 - Rights Data Dictionary (RDD).
- Part 7 - Digital Item Adaptation (DIA).

- Part 8 - Reference Software.
- Part 9 - File Format (FF).
- Part 10 - Digital Item Processing (DIP).
- Part 11 - Evaluation Tools for Persistent Association.
- Part 12 - Test Bed for MPEG-21 Resource Delivery.
- Part 13 - Scalable Video Coding (SVC).
- Part 14 - Conformance.
- Part 16 - Event Reporting (ER).
- Part 16 - Binary Format.
- Part 17 - Fragment Identification for MPEG Media Types.
- Part 18 - Digital Item Streaming.

In this section, we will describe only some parts of the standard, notably those that provide the most relevant functionality from the view point of context-awareness and ubiquitous access (parts 2, 3 and 7). The reader interested in obtaining more details on the rest of the standard may find useful information in [79] and [27].

The basic concepts of MPEG-21 are the User and the Digital Item (DI). The User is any type of actor that manipulates content, be it a person or a system (e.g. subscriber, producer, provider, network). A DI is the smallest unit of content for transaction and consumption, and at the conceptual level it can be seen as a package of multimedia resources related to a certain subject (e.g. MPEG-2 video stream, MP3 file, a set of JPEG pictures, text file, etc), together with associated descriptions (e.g. rights descriptions, other context descriptors, MPEG-7 [93] audio and video



Figure 3.9. Example of an MPEG-21 Digital Item

descriptors, etc), including the respective Digital Item Declaration (DID). Part 2 of the standard (Digital Item Declaration Language, DIDL) specifies a standardised method based on XML Schema for declaring the structure of the Digital Item [96]. Inside this XML document, the DID, standard MPEG-21 mechanisms are used to either convey resources and descriptions embedded directly in the DID or to provide information on the location of the resources to be fetched. The DID provides an indication of the composition of the DI in terms of media resources and metadata, as well as of its structuring in terms of the relationships among its several constituents. A DI is thus used to create the concept of a package or single unit, around a variety of multimedia information bounded by some common attribute, usually the topic they refer to. Figure 3.9 illustrates the concept of the MPEG-21 Digital Item.

The DIDL provides a number of entities or concepts, together with a well defined syntax and semantics, to allow to describe the complete structure of complex

digital objects as well as possible interactions with it. Among the complete set of defined entities - *Container, Item, Component, Anchor, Descriptor, Choice, Selection, Condition, Annotation, Assertion, Resource, Statement* - the following can be considered as the core to enable the use of MPEG-21 DID as a conceptual packaging format of complex digital objects. Their semantic meaning is here below indicated:

- *Resource*
 - a Resource corresponds to a digital representation of one media component of the object. It may, for example, be a CD quality file of the Requiem of Mozart or a PDF file with the bibliography of Mozart or the lyrics of the Requiem.
- *Component*
 - a Component wraps one or more resources. The condition to have more than one Resource under a single Component, is that those Resources need to be considered equivalent - same semantic meaning, but different digital representations. For example, the above referred CD quality version of Mozart's Requiem and an equivalent mp3 version of the same content.
- *Item*
 - an Item usually groups more than one Component, which refer to different contents, thus exhibiting a multi-part structure. An Item could be, for example, a TV programme having one video component with corresponding MPEG-2 MP@ML (Main Profile @ Main Level), one audio component with AC3 resource file and a text file with subtitles information in the form of specially marked up text with time stamps to be displayed during playback.

- *Descriptor*
 - conveys secondary information about Items, Components or Containers. The use of Descriptors is optional, but highly desirable. They allow to include metadata about the resources. A Descriptor may be resource-based (comprised of a single Component), or text-based (comprised of a single Statement).
- *Statement*
 - Statements are used to provide information concerning the format of the data included within Descriptors and encloses textual descriptive data within a Descriptor.
- *Container*
 - a Container can be seen as the higher level of the grouping mechanisms provided by the DID. It allows to bound together different single objects, which can already be complex objects, composed of various Resources and even different associations of single objects already grouped as a Container. It therefore provides the means to build very complex digital objects. One example could be a multi programme transport stream transmitted by a TV cable operator who aggregates a large number of digital TV programmes.

Inside the DID, these concepts are mapped into equivalent DIDL elements with syntax as follows:

- `<didl:Resource>`

`<element name="Resource" type="didl:ResourceType"`

```

        substitutionGroup="didmodel:Resource"/>
<complexType name="ResourceType" mixed="true">
  <complexContent mixed="true">
    <extension base="didmodel:ResourceType">
      <sequence>
        <any namespace="##any" processContents="lax" minOccurs="0"/>
      </sequence>
      <attribute name="mimeType" type="string" use="required"/>
      <attribute name="ref" type="anyURI"/>
      <attribute name="encoding" type="string"/>
      <attribute name="contentEncoding" type="NMTOKENS"/>
      <anyAttribute namespace="##other" processContents="lax"/>
    </extension>
  </complexContent>
</complexType>

```

- this element represents a media resource, which can be directly included in the DID (inline resources) or be referenced by a URI (external resources). In the first case, the format of the media resource must be indicated, using the appropriate *mimeType*. If the its type is text or XML, the media resource can be placed inside this element in its raw format; otherwise, it must be Base64 encoded. The attributes of this element are the *mimeType* and *ref* (with the location or endpoint of the resources, usually a URI). The resource element may only appear inside a Component element.

- Example:

```

<resource mimeType="image/jpeg"
  ref="http://inescporto.pt/Porto.jpg">

```

- **<didl:Component>**

```

<element name="Component" type="didl:ComponentType"
  substitutionGroup="didmodel:Component"/>
<complexType name="ComponentType">

```



```

<complexContent>
  <extension base="didmodel:ComponentType">
    <sequence>
      <element ref="didmodel:Descriptor" minOccurs="0"
        maxOccurs="unbounded"/>
      <element ref="didmodel:Resource" maxOccurs="unbounded"/>
    </sequence>
    <attributeGroup ref="didl:ID_ATTRS"/>
    <anyAttribute namespace="##other" processContents="lax"/>
  </extension>
</complexContent>
</complexType>

```

– a Component element may contain one or more Resource elements and any number of Descriptor elements.

– Example:

```

<Component>
  <Descriptor>
    <Statement mimeType="text/xml">
      <dii:Identifier>enthrone:did:exampleclip</dii:Identifier>
    </Statement>
  </Descriptor>
  <Resource mimeType="application/x-rtsp"
    ref="rtsp://enthrone.org/clip1"/>
</Component>

```

- **<didl:Item>**

```

<element name="Item" type="didl:ItemType"
  substitutionGroup="didmodel:Item"/>
<complexType name="ItemType">
  <complexContent>
    <extension base="didmodel:ItemType">
      <sequence>
        <element ref="didmodel:Descriptor"

```

```

        minOccurs="0" maxOccurs="unbounded"/>
<choice minOccurs="1" maxOccurs="1">
  <sequence>
    <element ref="didmodel:Item"/>
    <element ref="didmodel:Component"/>
  </sequence>
</choice>
</sequence>
<attributeGroup ref="didl:ID_ATTRS"/>
<anyAttribute namespace="##other"
  processContents="lax"/>
</extension>
</complexContent>
</complexType>

```

- Inside an Item element there may exist one or more Components or even more nested Items. Different Descriptor elements may also be placed inside the Item element.

- Example:

```

<Item>
  <Descriptor>
    <Statement mimeType="text/xml">
      <BasicDescription>
        This Item carries two sub-Items
      </BasicDescription>
    </Statement>
  </Descriptor>
</Item>
<Item>
  <Descriptor>
    <Statement mimeType="text/xml">
      <BasicDescription>
        The first sub-Item
      </BasicDescription>
    </Statement>
  </Descriptor>
  <Component>

```

```

        <Resource mimeType="application/x-rtsp"
            ref="rtsp://enthroned.org/clip1"/>
    </Component>
</Item>
    ....
</Item>

```

- **<didl:Descriptor>**

```

<element name="Descriptor" type="didl:DescriptorType"
    substitutionGroup="didmodel:Descriptor"/>
<complexType name="DescriptorType">
    <complexContent>
        <extension base="didmodel:DescriptorType">
            <sequence>
                <element ref="didmodel:Descriptor" minOccurs="0"
                    maxOccurs="unbounded"/>

                <choice>
                    <element ref="didmodel:Component"/>
                    <element ref="didmodel:Statement"/>
                </choice>
            </sequence>
            <attributeGroup ref="didl:ID_ATTRS"/>
            <anyAttribute namespace="##other"
                processContents="lax"/>
        </extension>
    </complexContent>
</complexType>

```

- a Descriptor provides additional information describing a given entity it refers to. This element may appear at any level, i. e., it may refer to Resources, to Components, Items or even Containers. Elements enclosed inside a higher hierarchical element, inherit the Descriptors of its parents. A Descriptor element may contain a textual description or it may use any kind of metadata scheme with the appropriate name space. A Descriptor

can also be a media resource (for example, a picture of a music CD cover, or a key frame of a video shot).

– Example:

```
<Descriptor id="AQoSData" eims:descriptorType="eims:ADT">
  <Statement mimeType="text/xml">
    <dia:DIA>
      <dia:DescriptionMetadata>
        <dia:ClassificationSchemeAlias alias="AQoS"
          href="urn:mpeg:mpeg21:2003:01-DIA-AdaptationQoS-CS-NS"/>
      </dia:DescriptionMetadata>
    </dia>
  </statement>
</Descriptor>
```

- **<didl:Statement>**

```
<element name="Statement" type="didl:StatementType"
  substitutionGroup="didmodel:Statement"/>
<complexType name="StatementType" mixed="true">
  <complexContent mixed="true">
    <extension base="didmodel:StatementType">
      <sequence>
        <any namespace="##any" processContents="lax"
          minOccurs="0"/>
      </sequence>
      <attribute name="mimeType" type="string"
        use="required"/>
      <attribute name="ref" type="anyURI"/>
      <attribute name="encoding" type="string"/>
      <attribute name="contentEncoding"
        type="NMTOKENS"/>
      <anyAttribute namespace="##other"
        processContents="lax"/>
    </extension>
  </complexContent>
</complexType>
```

- a Statement provides the mean to indicate the datatype of the information conveyed within textual-based Descriptors, which is done via the attribute "mimeType". Statements are thus used always inside textual-based Descriptors.
- Example:

```
<Descriptor>
  <Statement mimeType="text/xml">
    <dii:Identifier>enthroned:did:descriptor_1</dii:Identifier>
  </Statement>
</Descriptor>
```

- **<didl:Container>**

```
<element name="Container" type="didl:ContainerType"
        substitutionGroup="didmodel:Container"/>
<complexType name="ContainerType">
  <complexContent>
    <extension base="didmodel:ContainerType">
      <sequence>
        <element ref="didmodel:Descriptor" minOccurs="0"
                maxOccurs="unbounded"/>
        <element ref="didmodel:Item" minOccurs="0"
                maxOccurs="unbounded"/>
      </sequence>
      <attributeGroup ref="didl:ID_ATTRS"/>
      <anyAttribute namespace="##other" processContents="lax"/>
    </extension>
  </complexContent>
</complexType>
```

- The higher level of the DIDL element set is the Container. Inside a Container element one or more Component, Item and Descriptor elements may appear. Additionally, a Container element may include other Container elements.

– Example:

```

<Container>
  <!--This content DID or Container, refers to a DI that contains
  two Items. Each Item has two sub-Items. Each of these Item
  child has two Components, corresponding to two variations of
  the same content.-->
  <Descriptor>
    <!--This Descriptor applies to the composed DI. It provides
    the identification of the DI and a overall description
    applying to the two Items here contained.-->
    <Descriptor>
      <Statement mimeType="text/xml">
        <dii:Identifier>enthronedid:descriptor_0</dii:Identifier>
      </Statement>
    </Descriptor>
    <Statement mimeType="text/xml">
      <Title>Example DI</Title>
    </Statement>
  </Descriptor>
  <Item>
    ...
    <Component> ... </Component>
    ...
    <Component> ... </Component>
  </Item>
  <Item>
    ...
    <Component> ... </Component>
    ...
    <Component> ... </Component>
  </Item>
</Container>

```

An illustration of the structure of a DID containing all these elements, is presented in Figure 3.10. In that diagram, the DID does not have sub-levels, i.e., elements do not appear inside other elements of the same type (for example, an Item

does not appear inside another Item element).

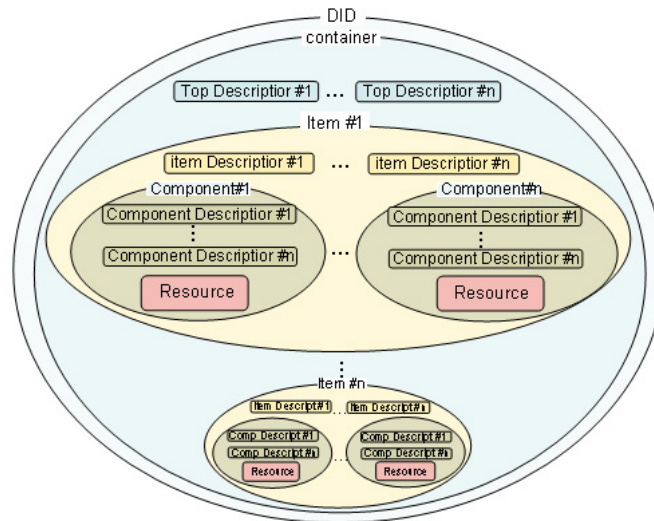


Figure 3.10. A graphical representation of the DID structure

Part 3 of the standard, Digital Item Identification (DII) [97] provides the mechanisms to uniquely identify Digital Items and parts thereof. It specifies the form to use identifiers to establish the link between DIs and associated information such as descriptive metadata or resources. It is an essential part of the standard towards the objective of flexibility and modularity seek by the standard, by which *“media resources and the Digital Item Declaration are fully separable, meaning that the representation of an Item can be communicated and processed in the absence of local copies of the associated media resources”* [94]. It allows to identify different types of DIs, resources inside the DIs, Description Schemes, Intellectual Property information, etc. Identification and description schemes that already exist can be used within the DII scope, otherwise new ones can be created and referred. Identifiers can be associated with Digital Items, or parts thereof, by including them inside the *Statement* element of the DID. Below it is provided an example of the use of DII.

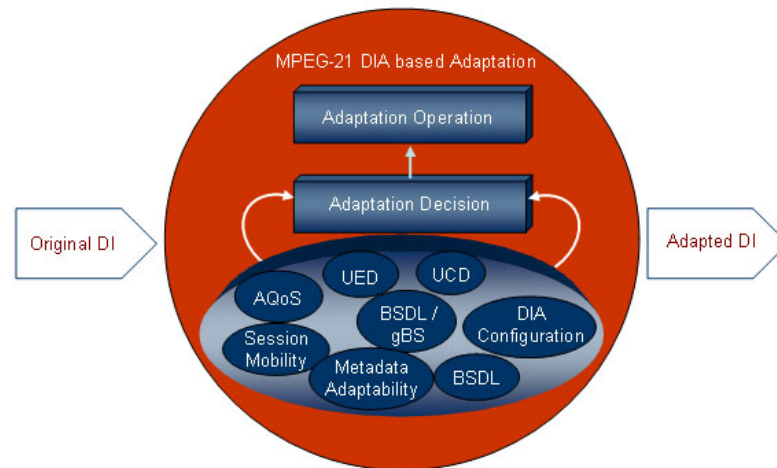


Figure 3.11. Tools and concept of MPEG-21 DIA

```

<Descriptor>
  <Statement mimeType="text/xml">
    <dii:Identifier>enthron:did:EN00XGS16062006DI</dii:Identifier>
  </Statement>
</Descriptor>

```

Part 7 of the standard (Digital Item Adaptation, DIA [98] provides a set of tools, allowing to describe the characteristics and capabilities of networks, terminals and environments, as well as preferences of users. In addition, the set of tools also provides the definition of the operations that can be performed upon the content and the result that can be expected. Figure 3.11 provides an illustration of the available tools of MPEG-21 DIA and its use for adaptation purposes.

Among others, specific adaptation tools and descriptions of the MPEG-21 DIA standard, such as Usage Environment Descriptor (UED), Adaptation Quality of Service (AQoS) and Universal Constraints Descriptor (UCD), define a set of descriptors and methodologies to describe the context of usage, the operations that can be

performed upon the content and the result that can be expected. Accordingly, these tools can be used to implement context-aware content adaptation systems. They can be used to analyse the current status of the consumption environment and decide upon the need to perform adaptation, including the type of adaptation to perform. The outcome of this process can be used to notify encoders or servers, receivers or intermediate adaptation engines, to adapt the stream to particular usage constraints or/and user preferences. The former can include, for example, the available network bandwidth or terminal display capabilities. The latter can, for example, indicate the removal of undesired objects from an MPEG-4 video scene or filtering out some of the media components of a DI. In addition, MPEG-21 DIA provides the means, in the form of declarative restrictions and conversion descriptions, to enable finer-grained control over the operations that can be performed when interacting with the content (playing, modifying, or adapting digital items). Systems adopting this approach, upon the same query, are able to produce different results according to the diverse usage environment constraints and/or user preferences/profiles. The concepts of the adaptation within MPEG-21 will be further detailed in section 5.2, which will present the layout of the functional architecture of an adaptation framework based on those concepts. As illustrated in figure 3.11, Part 7 of the MPEG-21 standard (DIA) specifies the following set of eight tools to assist adaptation operations:

- UED, Usage Environment Description tools
 - provide descriptions related to User Characteristics, Terminal Capabilities, Network Characteristics and Natural Environment Characteristics.
- AQoS, Terminal and Network Quality of Service tools
 - provide descriptions of QoS constraints, the adaptation operations capable of meeting those constraints and the effects of those adaptations upon the

DI in terms of its quality.

- UCD, Universal Constraints Description tools
 - allow describing constraints on adaptation operations and to have a control over the type of operations that are executed upon the content when interacting with it.
- BSDL (Bitstream Syntax Description Language) and gBS (generic Bitstream Syntax) tools
 - provide the means to describe the high-level structure of bitstreams using XML and XML Schemas. Their goal is to allow the manipulation of the bitstreams through the use of editing-style operations (e.g., data truncation) with the processor being format agnostic. They thus provide the means to manipulate the content at the bit stream syntax level.
- Metadata Adaptability tools
 - provide information that can be used to reduce the complexity of adapting the metadata contained in a DI, allowing filtering and scaling of metadata as well as the integration of XML instances.
- Session Mobility tools
 - provide the means to transfer state information regarding the consumption of a DI from one device to a second device.
- DIA Configuration tools
 - carry information required for the configuration of DI Adaptation Engines.

Among these, as referred above, the first three provide the core support for building context-aware content adaptation systems. They enable to describe both the characteristics of the usage environment, as well as the operations that can be performed on the multimedia content and the expected results. Their use enables the decision mechanism to select the adaptation operation that satisfies the constraints of the consumption environment, while maximising a given parameter or utility. This utility is most often the quality of the service or the quality of the user experience. These tools are briefly described hereafter. It is also included a description of the BSDL and gBS tools, since they provide support for the implementation of adaptation operations that are independent of the format of the resources to be adapted. As such, in spite of the fact that they are especially suited to be used with scalable formats, they can be used in a great variety of situations.

UED tools UED tools allow describing the characteristics of the environment in which the content is to be consumed, notably the capabilities of the terminal, the characteristics of the network and information regarding the user and his/her surrounding natural environment. These tools are grouped into four major classes, according to the entity they address:

1. *Terminal capabilities*: refers to the capabilities of the terminal where the content is to be consumed, in terms of the types of encoded formats that are supported - *codec capabilities* -, in terms of the display and audio output device characteristics and of several input devices - *input-output capabilities* - and finally physical characteristics of the terminal, such as the processing power, amount of available storage and memory, or data I/O capabilities - *device properties*.
2. *Network characteristics*: includes the description of the static attributes of

networks such as maximum channel capacity - *network capabilities* - and the description of parameters of the network that may vary dynamically along time, such as instantaneous available bandwidth, error rate or delay -*network conditions*. The former tools are used upon set up of the connection assisting in the selection of the optimum operation point, whereas the later are used to monitor the state of the service and update the initial set-up accordingly.

3. *User characteristics*: there are four different subsets of descriptions that fall into this category.

- *User Info*: a generic group of descriptors using MPEG-7 Description Schemes (DS), more precisely the *User Agent* DS. This group provides information on the user him/herself, as well as indications of general user preferences and usage history;
- *Presentation preferences*: a second subset of descriptors, related to the preferences of the user regarding the way the A/V information is presented. It includes the audio power and equaliser settings, video colour temperature, brightness and contrast or the user's focus of attention;
- *Accessibility Characteristics*: a group of descriptors carrying information concerning auditory or visual impairments of the user;
- *Location and status*: a subset of descriptors that provide information related to the instantaneous mobility status and destination of the user.

4. *Natural environment characteristics*: this kind of description tools provides information regarding the characteristics of the natural environment surrounding the user who wishes to consume the content. Using MPEG-7 description schemes, they provide information regarding the location and the time where the content is to be consumed and information describing AV attributes of

the usage environments, such as noise levels experienced in the surrounding environment or the light/illumination conditions.

Figure 3.12 and figure 3.13 show examples of excerpts of UED files describing characteristics of a terminal and the conditions of a network.

AQOS and UCD tools The main purpose of adapting the content is to try to provide the user with the best possible experience in terms of perceived quality. The perceived quality, despite being highly subjective and the need to take into consideration a number of factors pertaining to the user himself and his surrounding environment, is greatly conditioned by the instantaneous availability of network and terminal resources. To cope with varying network and terminal conditions, it is possible to envisage a number of different reactive measures or operations to perform upon the content in most cases.

The MPEG-21 DIA standard provides a mechanism that enables to decide which of those reactive measures to endorse in a given situation, whilst maximising the user experience in terms of perceived quality. Through the inclusion of descriptors that indicate the expected result in terms of quality for a set of different operations performed upon the content under certain context usage constraints, Adaptation Decision Engines (ADEs) can obtain the new best possible operation point and the correspondent transformation to perform.

This is accomplished through the AQoS and UCD description tools. The former provides the indication of different sets of encoding parameters and the resulting quality of the encoded bit stream for each of those sets. The latter enables

```

<?xml version="1.0" encoding="UTF-8" standalone="yes" ?>
<DIChoiceRequest xmlns="urn:visnet:contextMessage:01"
  xmlns:dia="urn:mpeg:mpeg21:2003:01-DIA-NS"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
  <ContextInformation id="ID000001">
    <dia:UsageEnvironmentProperty xsi:type="dia:TerminalsType"
      xmlns:dia="urn:mpeg:mpeg21:2003:01-DIA-NS"
      xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      <dia:Terminal>
        <dia:TerminalCapability xsi:type="dia:CodecCapabilitiesType"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
          <dia:Decoding xsi:type="dia:VideoCapabilitiesType"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <dia:Format href="urn:mpeg:mpeg7:cs:VisualCodingFormatCS:2001">
              <mpeg7:Name xml:lang="en" xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001">
                MPEG-4 Visual Simple Profile @ Level 1
              </mpeg7:Name>
            </dia:Format>
          </dia:Decoding>
          <dia:Decoding xsi:type="dia:AudioCapabilitiesType"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <dia:Format href="urn:mpeg:mpeg7:cs:AudioCodingFormatCS:2001">
              <mpeg7:Name xml:lang="en" xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001">
                MPEG-4 High Quality Audio Profile @ Level 1
              </mpeg7:Name>
            </dia:Format>
          </dia:Decoding>
        </dia:TerminalCapability>
        <dia:TerminalCapability xsi:type="dia:DisplaysType"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
          <dia:Display id="primary_display">
            <dia:DisplayCapability xsi:type="dia:DisplayCapabilityType"
              xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
              <dia:Mode refreshRate="30">
                <dia:Resolution horizontal="720" vertical="576" />
              </dia:Mode>
            </dia:DisplayCapability>
          </dia:Display>
        </dia:TerminalCapability>
      </dia:Terminal>
    </dia:UsageEnvironmentProperty>
  </ContextInformation>

```

Figure 3.12. Example of an excerpt of a Terminal UED

```

<ContextInformation id="ID000002">
  <dia:UsageEnvironmentProperty xsi:type="dia:NetworksType" id="uep0"
    xmlns:dia="urn:mpeg:mpeg21:2003:01-DIA-NS"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
    <dia:Network id="net1">
      <dia:networkCharacteristics xsi:type="dia:NetworkCapabilityType" id="nct01"
        maxCapacity="10000000" minGuaranteed="2678000"
        inSequenceDelivery="true" errorCorrection="true" errorDelivery="true"
        xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
      </dia:Network>
      <dia:Network id="net2">
        <dia:networkCharacteristics xsi:type="dia:NetworkCapabilityType" id="nct02"
          maxCapacity="10000000" minGuaranteed="3120000"
          inSequenceDelivery="true" errorCorrection="true" errorDelivery="true"
          xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
        </dia:Network>
        <dia:Network id="net3">
          <dia:networkCharacteristics xsi:type="dia:NetworkConditionType" id="nct03"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <dia:AvailableBandwidth average="2567000"/>
          </dia:NetworkCharacteristics>
        </dia:Network>
        <dia:Network id="net4">
          <dia:networkCharacteristics xsi:type="dia:NetworkConditionType" id="nct03"
            xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance">
            <dia:AvailableBandwidth average="2913000"/>
          </dia:NetworkCharacteristics>
        </dia:Network>
      </dia:UsageEnvironmentProperty>
    </ContextInformation>

```

Figure 3.13. Example of an excerpt of a Network UED

```

<DIA>
...
<Description xsi:type="AdaptationQoSType">
...
  <Module xsi:type="UtilityFunctionType">
...
    <Constraint IOPinRef="BANDWIDTH">
      <Vector>100000 500000 1000000 1500000</Vector>
    </Constraint>
    <Utility IOPinRef="DVQ"> <!-- Digital Video Quality-->
      <Vector>75 88 98 99</Vector>
    </Utility>
  </Module>
...
  <IOPin semantics?":AQoS:1.1.1" id="BANDWIDTH"/>
  <IOPin semantics="":AQoS:1.2.x" id="DVQ"/>
</Description>
</DIA>

```

Figure 3.14. Example of an AQoS metadata

the transformation of that information, together with the information about the current conditions of the usage context conveyed as UED, into the form of restrictions that can further be used by the adaptation decision engines. Figures 3.14 and 3.15 provide examples of AQoS and UCD metadata.

BSD tools In large-scale networked A/V applications ², when the quality drops in one of the "legs" of the end-to-end network path, it may be desirable to be able to adapt the content along the network, at the point where the problem had occurred. However, it may not be easy to find an adaptation engine with matching capabilities, at that exactly point (or even nearby). In broadcast applications or push services, where the content is sent simultaneously to multiple users, it would often be desirable to alter the characteristics of the video bit stream along the transmission chain, but only at specific points that would not interfere with users sharing the same content, but who were not experiencing loss of quality.

The accomplishment of this goal, requires the availability of content adaptation gate-

²By large-scale, we refer to multi-domain networks or a set of cascaded networks, each one constituting a different Autonomous System


```

<DIA>
...
<Description xsi:type="UCDType">
...
  <AdaptationUnitConstraints>
...
    <LimitConstraint>
      <!-- Horizontal Resolution constraint .>
      ...
      <Value> 720 </Value>
      <Operation operator=":SFO:38"> <!-- is less or equal to>
      ...
    </LimitConstraint>
    ...
  </AdaptationUnitConstraints>
</Description>
</DIA>

```

Figure 3.15. Example of UCD metadata

ways, placed along the distribution chain, which have the ability to process specific encoding formats. Given the large number of existing encoding formats and their variants, it may prove to be cumbersome to have such kind of media gateways. In addition, the solution is not easily expandable or scalable towards new encoding formats.

One much better approach is to enable the processing of encoded bit streams without the need to actually understand their specific encoding syntaxes, i.e. in a *format-agnostic* way. BSD tools of the MPEG-21 DIA provide the means to convey a description of the encoded bit stream using Extensible Markup Language (XML), and to perform the transformation of this description in the XML space. This can be done, for example, through the use of Extensible Stylesheet Language Transformation (XSLT). The adapted bit stream is then generated from the original bit stream, but using the modified description. Figure 3.16 illustrates this process.

To be able to generate universally understandable descriptions of specific encoding schemes, the standard has developed an XML-based language, the *Bitstream Syntax Description Language* (BSDL). To generate generic descriptions of

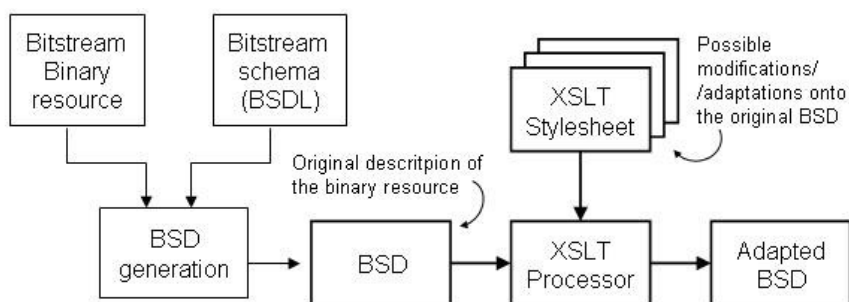


Figure 3.16. Editing-like content adaptation using BSDL and gBS

binary sources independently of the encoding format, the standard has specified the schema *Generic Bitstream Syntax* (gBS).

When using the BSDL tool, a Bitstream Syntax (BS) schema is generated at the encoder side, describing the encoding syntax of the bit stream. The BS schema then travels together with the bit stream, or is requested when needed. It essentially provides a high-level description of the structure or organisation of the bit stream. Therefore, it allows only for simple adaptation operations, such as filtering, truncation or removal of data.

More advanced forms are possible when using the gBS schema. This tool enables the use of semantic labels and the establishment of associations between those labels and syntactical elements being described. In addition, it provides the means to describe the bit stream in a hierarchical way.

BSD tools (BSDL and gBS) can thus be used with any kind of encoding format. It is however with scalable formats that their use is more interesting, as they may enable very simple operation. For example, when the constraints of the environment impose that a reduction of the bit rate is made, having a description of the syntax of the

scalable bit stream, greatly simplifies the role of the adaptation engine in identifying the layer(s) to be dropped.

Recent research addressing MPEG-21 Worldwide research groups are investigating the use of the tools being specified in the MPEG-21 standard to develop efficient adaptation engines and to improve the proposed approaches.

DANAE [30], an IST European Community (EC) co-funded project, addressed the dynamic and distributed adaptation of scalable multimedia content in a context-aware environment. The objective of DANAE was to specify and develop an advanced MPEG-21 infrastructure for such an environment, to permit flexible and dynamic media content adaptation, delivery and consumption. The goal was to enable end-to-end quality of multimedia services at a minimal cost for the end-user, with the integration of features and tools available from MPEG-4 and MPEG-7, and DRM support in the adaptation chain. Their system focused on DIA and DIP (Digital Item Processing - specified in part 10 of the MPEG-21 standard) and contributed to the MPEG-21 standardization effort. DANAE supports the application of DIP for DIA to adapt Digital Items in a globally, optimized and semantically aware way to allow to dynamically change the usage context, as well as to enable the adaptation of live content (the content may be delivered while still being created).

The ENTHRONE project [31], [121], [120], [8], [119], [118], [61], another IST project co-funded by the EC under Framework Programme 6, has developed a content mediation and monitoring platform based on the MPEG-21 specifications and distributed technologies³. The ENTHRONE platform features a core system designated as ENTHRONE Integrated Management Supervisor (EIMS), that main-

³In fact, as already indicated in the introduction to this manuscript, the work undertaken within this thesis, was partly developed within the ENTHRONE project. The author was one of the proponents of the project.

tains distributed data bases and repositories of MPEG-21 DIs and enables customised access to that content to multiple users, through heterogeneous environments and networks. The EIMS incorporates an adaptation decision engine (ADE) that uses MPEG-21 DIA tools, notably UED, UCD and AQoS. The ENTHRONE ADE transforms into UCD the UED it receives. Different UEDs are used: one to express the capabilities of terminals and the user preferences, generated at the consumption peer; another to express the capabilities of available adaptation engines; and a third one to express the condition of the network. The ADE thus transforms into a constraint representation the characteristics of the current usage environment. It then uses AQoS, describing the operations that can be performed upon the content (i.e., the possible variations of the content through their characteristics such as frame rate, aspect ratio or bit rate) and the result in terms of quality that can be obtained with each variation. ENTHRONE provides a complete framework for the generation, management and customised access to multimedia content using the MPEG-21 specifications (the subject of this thesis :-). The work of ENTHRONE in content adaptation has been so far limited to altering the encoding parameters of each media resource (adaptation operation "*scaling*" of type "*content manipulation*" as classified in table 3.1. However the project has initiated its second phase of activity. In this new period, ENTHRONE is extending the good results obtained in phase 1 in order to provide new forms of content adaptation and provide more flexibility in terms of distribution of all kind of resources (systems, devices, software programmes, A/V resources, etc.).

The Department of Information Technology (ITEC) of the Klagenfurt University [131] is actively contributing to the development of the MPEG-21 specifications, in particular those that aim to assist the context-aware adaptation of content. This research group has implemented MPEG-21 applications such as multimedia

adaptation tools (e.g., gBSDtoBin, BSDLink Webeditor) or the ViTooKi operating system that provides the support for describing terminal capabilities and user preferences using MPEG-21 [191].

The Catalan Integrated Project aims to develop an advanced Internet environment based on a Universal Multimedia Access (UMA) concept using MPEG-21 standard tools. The project intends to extensively employ the MPEG-21 and MPEG-7 standards for the adaptation decision taking procedures which will select the best adaptation of a specific content, taking into account network characteristics, terminal capabilities, and state of A/V content transcoding servers.

Other standardisation efforts

In addition to the tools and descriptions being specified in MPEG-21, other standardisation bodies have also conducted work towards the support of context-aware applications and the assistance of adaptation operations. That is the case of the World Wide Web Consortium (W3C) or the Internet Engineering Task Force (IETF), both of which have delivered related specifications in the last years. Content adaptation at the network edges is already being performed by content distributed mediators, in order to balance the network load or in the attempt to satisfy different user groups' profiles, redirecting requests based on geography or other profile characteristics. This is being achieved through the use of content distribution networks (CDNs), peer-to-peer (P2P) technologies and personalisation techniques.

In the mobile networking world, content filtering services are being added to most caches. Wireless network proxies transform both protocol and Web content, converting HyperText Markup Language (HTML) into Website Meta Language (WML) for small-screen displays and HyperText Transfer Protocol (HTTP) into

Wireless Datagram Protocol (WDP) for wireless delivery purposes. Adaptation of content, aiming at increased penetration, is already happening, in the mobile world. However, it still offers somewhat limited functionality, as it essentially accounts for bandwidth restrictions and device capabilities. The focus is still on the adaptation of the layout of the service, rather than on the content itself. The same is happening in the TV broadcasting world, aiming at the adaptation of interactive applications and Web-based content to multiple non-interoperable platforms.

IETF, the Internet Engineering Task Force, has produced the Internet Content Adaptation Protocol (iCAP) specification [84], which is basically an HTTP-based Remote Procedure Call protocol that allows clients to send HTTP messages to instruct iCAP servers, or generic Web servers, to perform some kind of operations on the content. iCAP servers are dedicated to perform specific tasks, and are thus expected to perform such tasks better, or more efficiently, than generic Web servers. The targeted adaptation operations were defined from a business perspective, and are seen as added-value services to the client. Examples of the envisaged adaptation operations are virus scanning, advertisement insertion and also some forms of content translation or content filtering.

The Mobile Alliance Forum has standardised the UaProf (User Agent Profile) specification [5], which provides an open vocabulary for WAP (Wireless Access Protocol) clients to communicate their capabilities to servers. It defines the data structure to describe client devices and to transport that information to the servers. This information may include hardware characteristics, such as screen size or type of keyboard, software characteristics, as for example type and version of browser, and also user preferences (e.g. sound enabled, colour choices, etc.). The idea is to empower the service providers with information that may assist them in customising

the service, up to a certain extent, to the end-user needs. The UAProf vocabulary has six components, as follows:

- *HardwarePlatform*: characteristics of the hardware of the terminal, including type of device, model, display size or memory;
- *SoftwarePlatform*: characteristics of the operating environment of the device.
- *NetworkCharacteristics* information about the network infrastructure, as for example bearer information;
- *BrowserUA*: identification of the browser application available on the device;
- *WapCharacteristicslists*: the WAP capabilities of the terminal;
- *PushCharacteristics*: push specifications of the device, such as maximum size of a push message.

World Wide Web Consortium (W3C)

The W3C is presently focussing on the development of technologies to implement the Semantic Web or Web 2.0 concept. The aim of the Semantic Web is to enable seamless access and automated processing to and of all resources available on the Web, using interchangeable descriptions that are both human and machine-readable. The accomplishment of this goal requires the use of common semantics to describe content, services or any resource (software or hardware) on the Web. As such, the work of W3C is very much connected with context awareness. The Resource Description Framework (RDF) [200], [59], the RDF Schema (RDFS) [19] and the Web Ontology Language (OWL) [127] are specifications of the W3C that constitute general description frameworks suitable to express metadata and hence,

context. RDF is a generic specification to represent metadata, in particular metadata related to resources on the Web, being strongly associated to the Semantic Web or Web 2.0 concepts.

RDF, the Resource Description Framework is an XML-based language to represent information in the Web. RDF has already been presented in section 2. In this section, some more details are provided, notably concerning the forms of using RDF.

As described in section 2, RDF defines a set of elements that enable to describe resources with attributes or characteristics (through the element *properties*) and to indicate the type of those attributes (through the element *property-type*). RDF thus provides the means to establish relationships between resources. Resources with the same *property-type* will have the same kind of characteristics. The proximity between them can then be assessed through the value of the *properties* elements.

RDF can be represented in one of three different formats:

1. XML syntax, which is the format commonly used to exchange RDF data between machines;
2. triple representation, consisting of sets of (Subject, Predicate, Object) values;
3. graph representation.

Figure 3.17 provides examples of the same metadata represented in RDF in each of its three possible formats. In this figure, the metadata is a simple sentence written in English, which is then represented using RDF in its triplet format (i), in the XML format (ii) and finally in the graph format (iii).

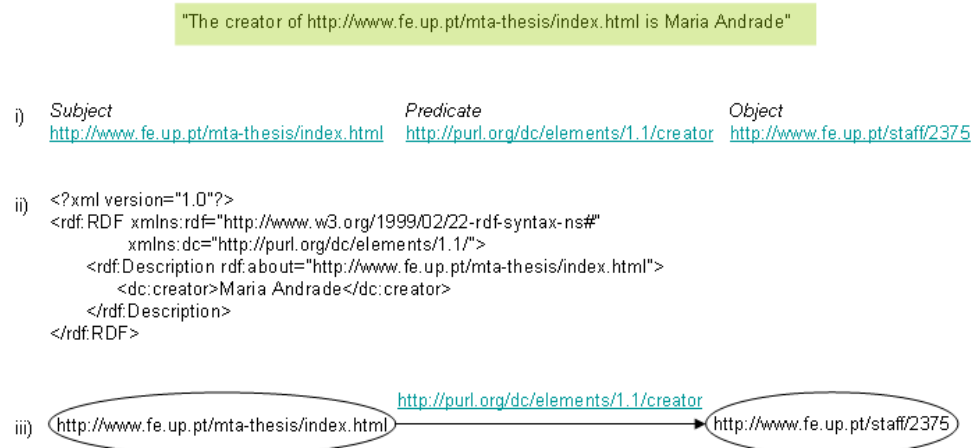


Figure 3.17. The three different formats for RDF representation

In the XML format (case ii) in the figure 3.17) the first line of the RDF document is the XML declaration. The XML declaration is followed by the root element of RDF documents (`rdf:RDF`). Namespaces are include to inform the origin for the definition of elements appearing in the RDF document. Accordingly, in the example, the `xmlns:rdf` namespace indicates that elements with the `rdf` prefix are from the namespace `http : //www.w3.org/1999/02/22 – rdf – syntax – ns`. Likewise, the `xmlns:dc` namespace specifies that elements with the `dc` prefix are from the namespace `http : //purl.org/dc/elements/1.1/`. This is the namespace of the Dublin Core Metadata Initiative (DCMI) [83], [10]. The `rdf:Description` element contains the description of the resource identified by the `rdf:about` attribute. The element `dc:creator` is a property of the resource.

RDFS, the RDF Schema allows to extend the scope of application of RDF, specifying how to use RDF to describe RDF vocabularies, (an hence, RDFS is also

designated as RDF Vocabulary Description Language [19]). Using RDFS it is possible to develop specific metadata vocabularies and to describe relationships among resources in terms of named properties and values.

As we saw above, RDF *properties* can be seen as attributes of resources and RDF *property-types*, as the relationships between resources. The combined use of these two elements can provide an indication of existence, and degree, of relationships between the resources. However, RDF does not provide the mechanisms for describing properties or relationships. RDF Schema solves this problem by defining classes and properties that can be used to describe other classes and properties. Using RDF Schema, it is possible, in the context of RDF, to declare new properties (attributes of the resources being described) as well as their semantics. This is, an RDF document can create an RDF Schema with the definition of properties of a given resource, as for example, title, director, producer, genre or duration, as well as the definition of the resources being described, as for example, TV programmes or movies.

OWL, the Web Ontology Language [127], [149], [128], [145], [78], is the W3C specification to develop ontologies. Ontologies, as described earlier in this section, provides the means to define terms representing real-world concepts and to describe the relationships between terms in a machine-understandable form. Those terms are then used to characterise resources on the Web.

In the context of the Semantic Web, ontologies are expected to be the key element towards the automated access to resources available on the Web. To achieve this goal, ontologies provide structured vocabularies describing relationships between different terms. If ontologies are created using open standards, any software

or hardware system (or human), will be able to process the information contained in the ontology, reason about it and unambiguously interpret it and infer the originally captured high-level concepts. [205].

OWL uses three main elements to create ontologies:

- classes;
- relationships;
- instances.

A *class* is used to represent a concept or entity of the real-world. For example, a *school* ontology, will most certainly have the classes *Student* and *Teacher*; or a *TV broadcasting* ontology will use the classes *Programme* or *Genre*. Classes are identified by a global unique identifier, which comprises a namespace and a string identifier. Namespaces are often Web locations, notably URLs, and the identifier, is usually a string with the name of the class. For example, in the case of the *TV broadcasting* ontology, the namespace could be "http://www.broadcastingCompany.com/ontology#", and the ID "Programme".

OWL can declare classes and organise these classes in a subclass hierarchy. OWL classes can be specified as logical combinations (intersections, unions, or complements) of other classes, or as enumerations of specified objects.

Relationships, as in RDF Schema, are captured using properties. OWL properties are very similar to RDF Schema properties: they can be hierarchically organised into sub-properties and can have domains and ranges. OWL classes are used to define those domains and ranges. The later can also be represented as externally defined datatypes (as for example, strings or integers). OWL provides two main types of properties: object and datatype properties. Object properties provide the means

to establish links between two classes of an ontology. Datatype properties enable to connect classes with (externally defined) datatypes. For example, let's consider the *TV broadcasting* ontology, and define the properties *hasTypeOf* and *hasTitle*. The property *hasTypeOf* relates the classes *Programme* and *Genre*, by having as domain the concept *Programme* and as range the concept *Genre*. The property *hasTitle* relates a class with a datatype, by having as domain the class *Programme* and as range, an XML Schema definition of a string (XSD:String).

Instances, or individuals, are analogous to object instances in object-oriented programming languages like Java or C++ (and hence are similar to instances of Java classes or to C++ objects instantiations). In an ontology, an instance is a representation of elements with actual data values. Considering the *TV broadcasting* ontology, **Weekly Magazine** can be an instance of the *Programme* concept (which is a class element of the ontology), and **News** an instance of the *Genre* concept.

Referring to the example of the *TV broadcasting* ontology, which is assumed to contain the class elements *Programme* and *Genre*, and the property elements *hasTypeOf* and *hasTitle*, excerpts of OWL code describing those elements and instances of the classes (**Weekly Magazine** and **News**, respectively instances of the *Programme* and *Genre* classes), are presented below:

```
<owl:Class rdf:ID="Programme"/>
<owl:Class rdf:ID="Genre"/>

<owl:ObjectProperty rdf:ID="hasTypeOf">
  <rdfs:domain rdf:resource="#Programme"/>
  <rdfs:range rdf:resource="#Genre"/>
</owl:ObjectProperty>

<owl:DatatypeProperty rdf:ID="hasTitle">
```

```

    <rdfs:domain rdf:resource="#Programme"/>
    <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
</owl:DatatypeProperty>

<Programme rdf:ID="Magazine">
  <hasTypeOf rdf:resource="#News"/>
  <hasTitle rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
    Weekly Magazine
  </hasTitle>
</Programme>

<Genre rdf:ID="News">
  <entitled rdf:datatype="http://www.w3.org/2001/XMLSchema#string">
    Generic International News Programmes
  </entitled>
</Genre>

```

Comparing with RDFS, OWL extends the use of properties by enabling the use of statements regarding properties and by imposing restrictions on how properties behave. For example, in OWL, properties can be declared to be transitive, symmetric or functional, or to be the inverse of or equivalent to another property. Restrictions on the behaviour of properties, for example value and cardinality, can be confined to individual classes. The fact that it is possible to assign characteristics to OWL properties, greatly assist the goal of modeling real world concepts or situations. And consequently, simplifying the role of software modules when reasoning about this information to infer the higher-level concepts. The benefits of this, can be made clear, through the following examples (extracted from [42] and [78]):

A) Consider an ontology with the following elements defined:

- of type class:
 - *Student*;

- *Parent*;
- *Friend*;
- *Age*.
- of type properties:
 - *hasAncestor*, stated as being *transitive*;
 - *hasFather*;
 - *hasFriend*, stated as being *symmetric*;
 - *hasChild*, stated as being *the inverse* of *hasParent*;
 - *hasAge*.

with the following instances and relationships:

- **Mike**, who *hasAncestor* **John**;
- **John**, who *hasAncestor* **Ali**;
- **Ali**, who *hasFriend* **John**;
- **Ali**, who *hasChild* **John**;

and with the following value and cardinality restrictions:

- *hasFather* property cannot have more than one instance (cardinality constraint);
- *hasAge* property cannot have a value higher than 150 (value constraint).

From this information, an OWL reasoner can deduce that:

1. Ali is an ancestor of Mike;
2. John is also a friend of Ali;

3. John's parent is Ali;
4. it is in the presence of an error if it detects more than one declaration of a *hasFather* relationship for the same instance of the class **Student**;
5. it is in the presence of an error if it detects the instance "200" of the concept **Age**.

B) Within the same domain, but comparing ontology elements that are possible to be denoted in RDF Schema and in OWL, as presented in table 3.2:

OWL enriches the existing XML, RDF and RDFS specifications with additional vocabulary and formal semantics. More specifically, the additional value introduced by OWL can be understood by enumerating the generic functionality and goal of each of these specifications, as follows:

- RDF provides a data model for assigning properties to resources and express simple relationships among them;
- RDFS adds a vocabulary (and rules on how to build new vocabularies) to describe properties and relationships (classes of RDF resources);
- OWL adds more vocabulary for describing properties and classes, enabling the creation of a hierarchy of levels and increased expressiveness and sophistication:
 - relations between classes;
 - cardinality and equality of classes;
 - enumerated classes;
 - richer description of properties and of their characteristics;
 - etc.

Three subsets or subclasses of OWL have been defined, with increased functionality, expressiveness and complexity:

- OWL Lite, satisfying the requirements for classification hierarchy and simple constraints;
- OWL DL (Description Logics), offering increased expressiveness while maintaining acceptable levels of complexity. It tries to obtain the best compromise between flexibility and simplicity, guaranteeing that all computational tasks can be performed within a finite and reasonable amount of time and that a conclusion will be reached. To observe these guarantees, it imposes some restrictions on OWL constructs, as for example, forbidding a OWL class to be an instance of another class (but allowing it to be a sub-class of a class);
- OWL Full, providing the maximum expressiveness and full compatibility with RDF (allows the use of all flexibility of RDF syntax) at at cost of increased complexity. OWL Full does not provide any guarantees of successfully reaching a conclusion within a bounded period of time.

Ontologies and conclusions defined and obtained within the domain of any one of these subclasses, will be also valid within the upper level subclasses domains.

OWL Lite is able to satisfy many of the requirements imposed by moderately complicated applications. Its use is considerably simpler than that of the other OWL subclasses, because it uses only some of the features of the complete set of OWL's capabilities, as for example:

- classes can only be named *superclasses* and cannot be arbitrary expressions;
- only certain kinds of restrictions can be used;
- equivalences can only be used between name classes;

- subclasses relationships can only be established between name classes;
- restrictions use only named classes;
- only 0 and 1 cardinality is allowed.

Since OWL builds on RDF and RDF Schema, its original (and primary) syntax is the RDF's XML syntax. However, due to the larger scope and richer semantic expressiveness and sophistication of OWL, this syntax may not provide the best form to create OWL documents, leading to difficulty to read documents (RDF syntax is very flexible). It should be noted that, whereas the primary use of OWL is to be machine-readable, it is also meant to be read and understood in an easy way by humans. Accordingly, other syntactic forms of OWL have been defined:

- XML based syntax, not aligned with the RDF conventions;
- abstract syntax, more compact and readable than XML;
- graphical syntax, based on UML (Unified Modeling Language)

An OWL document, or an OWL ontology, created with its primary syntax, is thus an RDF document. The root element of that document, the *rdf* : *RDF* element, specifies the namespaces used within the ontology. The document may then have a *owl* : *Ontology* element to include comments, versioning control or references to other ontologies. The element *owl* : *class* enables to define class elements, whereas properties are defined using the *owl* : *ObjectProperty* and *owl* : *DatatypeProperty*. Many other OWL syntactic elements exist to allow to define relationships, constraints, cardinality, etc. Whenever elements already exist in RDF or RDFS, they are used on OWL documents preceded by *rdf*: or *rdfs*:. This indicates that they already exist in the *rdf* or *rdfs* namespaces and, as such, their definition can be found

there. Below, we present an example of a OWL-Lite ontology within the school domain. The ontology expresses the concepts of students, graduate students, degrees and relationships therein. In particular the ontology captures the concept that a GraduateStudent is a student with a BA or BS degree.

```
<?xmlversion"1.0"?>
<rdf:RDF
  xmlns:owl="http://www.w3c.org/2002/07/owl#"
  xmlns:rdf="http://www.w3c.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3c.org/2000/01/rdf-schema#"
  xmlns:xsd="http://www.w3c.org/2001/XMLSchema#"
  <owl:Ontology rdf:about="">
    <rdfs:comment>An example OWL ontology</rdfs:comment>
    <owl:imports rdf:resource="http://www.adomain.org/degrees"/>
    <rdfs:label>Student Degree Ontology</rdfs:label>
  </owl:Ontology>
  <owl:class rdf:id="GraduateStudent">
    <rdfs:subclassOf>
      <owl:Restriction>
        <owl:onProperty>
          <owl:ObjectProperty rdf:about="#hasDegree"/>
        </owl:onProperty>
        <owl:someValueFrom>
          <owl:class>
            <owl:oneOf rdf:parseType="Collections">
              <Degree rdf:ID="BA"/>
              <Degree rdf:ID="BS"/>
            </owl:oneOf>
          </owl:class>
        </owl:someValueFrom>
      </owl:Restriction>
    </rdfs:subClassOf>
  </owl:class>
</rdf:RDF>
```

CC/PP The W3C has also delivered the Composite Capability/Preferences Profiles (CC/PP) specification [199], which defines an RDF-based framework for de-

scribing device capabilities and userpreferences.

CC/PP provides the means to specify client capabilities (i.e. the *user agent* information), as well as user preferences, using Uniform Resource Identifiers (URIs) [85] and RDF text sent in HTTP requests [185], [154]. The *user agent* specifies the preferences of the user in the header of the client HTTP request, such as versions of content or languages, and is empowered with negotiation capabilities.

Although the CC/PP specification was originally developed to be used mainly on wireless devices, it can be applied to any Web-enabled terminal. It uses RDF in its XML serialised format to exchange profiles between devices with information on the *user agent*'s capabilities and the user's preferences.

A CC/PP profile can be seen as a two-level tree containing components and attributes of those components. Components can be the hardware or the software platforms of the terminal, or any specific application running on top of those platforms. Different components may appear in one CC/PP profile. A reference to the schema that indicates the types of components that appear in that profile, as well as the rules concerning the attributes of those components, are included in the profile so that the recipient may correctly interpret and use the information there contained. As the CC/PP specification uses RDF, CC/PP profiles are composed of sets of "(*Subject*, *Predicate*, *Object*)".

The components (the *Subject*) have named attributes (the *Predicate*) and values for those attributes (the *Object*). The most common components in CC/PP refer to the hardware and software platforms of the client terminal device and are thus related to the capabilities of the terminal. There are, however, other type of components that can be described in a CC/PP profile, as for example the location. CC/PP uses a vocabulary to define the format and the language for specifying the names and

values of components and their attributes. Nevertheless, different CC/P profiles or applications may use different vocabularies. This is a particular feature of CC/PP that allows different applications to define and use particular vocabularies that suit their particular needs. One of the vocabularies that has been traditionally used in CC/PP profiles, is the UAProf in the mobile world. For other application areas, and in general terms, the W3C recommends the use of RDF to define vocabularies. Figure 3.18, illustrates an example of a CC/PP profile for a hypothetical PDA (Personal Digital Assistant).

Related R&D work The W3C is quite active in the field of context-aware applications, and as such is contributing to enable content adaptation. Many of the current specifications and approaches that are being studied or developed towards the implementation of content adaptation operations rely on the use of W3C technologies. M. H. Butler [21], provides a survey of technologies that can be used to enable devices to communicate their capabilities to the providers of services, enabling them to customize content for use. His work focussed on the use of W3C and Mobile Alliance technologies for use in mobile environments. He continued his work by designing a system that enables mobile clients to negotiate the format of the content being delivered to them, through the use of the CC/PP and UAProf specifications [22].

J. Gilbert and M. Butler [65], defined a data model to allow the use of profiles with multiple vocabularies, when using the CC/PP specification to allow mobile devices to communicate their capabilities to servers.

In spite of its wide applicability, the CC/PP model presents some limitations when addressing more complex context-aware scenarios. Although any kind of contextual information could in principle be described using CC/PP, as long as that context

```

<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3c.org/1999/02/22-rdf-syntax-ns#"
  xmlns:prf="http://www.inescporto.pt/test-profile-vocabulary#">
  <rdf:Description about="HardwarePlatform">
    <prf:Defaults
      Vendor="INESC"
      Model="INS2007"
      Type="PDA"
      Screensize="640x480x24"
      Keyboard="yes"
      Speaker="yes"
      Memory="32MB"
    </prf:Defaults>
  </rdf:Description>
  <rdf:Description about="SoftwarePlatform">
    <prf:Defaults
      OS="AdvancedINS"
      HTML version="4.0"
      JavaScriptVersion="4.0"
      WAPVersion="1.0"
    </prf:Defaults>
  </rdf:Description>
  <rdf:Description about="UserPreferences">
    <prf:Defaults
      Language="English"
    </prf:Defaults>
  </rdf:Description>
</rdf:RDF>

```

Figure 3.18. Example of a CC/PP

information could be described using RDF, it does not provide the mechanisms to carry additional information, such as temporal information or resolution of the contextual information. But the most important limitations can come from the fact that CC/PP Components and Attributes (or subtypes of them) have a limited set of values and a restricted syntax. For example CC/PP does not provide any support for expressing relationships or constraints. It also has some limitations regarding the type of information.

Conclusions

This section has provided an overview of major standardisation efforts that have emerged in the last years and that can be generically applied to the implementation of services and tools that contribute to the universal access to multimedia content (UMA, Universal Multimedia Access). Context-aware content adaptation systems provide the means to implement UMA and can thus be built based on the described technologies. Having carefully analysed these standards, it seems fair to say that the MPEG-21 standard is the most complete one. This is easily explained by the fact that the scope of MPEG-21 is much more than '*just*' adaptation or '*just*' context description:

- it defines a complete infrastructure for packaging, in a compact way, all the intervenient elements that contribute to the augmented access and use of multimedia content, such as content and context descriptions, content representation and composition, management of IPRs and digital rights, enforcement of digital rights or coarse control content consumption;
- provides a comprehensive framework for the use of context information and accordingly mediates the optimum access to content, deciding the need for adaptation;

- the scope of adaptation aims all types of content and the tools/descriptions aim all kinds of adaptation;
- makes use of a comprehensive set of descriptions (also through the use of MPEG-7) that are able to characterise virtually any of the entities usually cooperating for the delivery of the adapted multimedia service;
- it is flexible enough to scale to new requirements as technology progresses.

MPEG-21 provides the means to bridge the gap between being able to understand and use context and evaluate the quality of the result achieved by the adaptation mechanisms. In fact, the tool AQoS from the set of DIA tools, is a step towards that direction. Nevertheless, more work is needed to actually evaluate the results of adaptation-decision taking and of the adaptation itself, in terms of user expectations and degree of fulfilment of user experience satisfaction.

Likewise, further investigation is needed to fully understand context and how to optimally use it, especially in what concerns the generation of implicit or higher-level context.

In this particular aspect, it seems of utmost interest, to try to merge the work of the research community engaged on the use of ontologies to build contextual models, as described in section 2.1. Nevertheless, it should be emphasised that the W3C specifications are very important, as they provide the key technologies to build ontologies and to create and represent descriptions of any resource on the Web. But also due to their flexibility, their wide acceptance and already extensive usage. Cooperation between the MPEG and W3C would, as a minimum, enable the former to benefit from the larger market penetration and the latter from the extensive knowledge on multimedia processing.

The challenge still remains in understanding how to use those standard technologies in a broader scope, providing larger coverage of contextual information and delivering results that are closer to user expectations in different situations and conditions.

While it is true that standards are an effective vehicle to interoperability, thus being extremely important in today's heterogeneous scenario, true end-to-end interoperability can only be achieved and guaranteed with appropriate infrastructural support based on the use of standards and middleware technologies.

Due to the wide variety of existing content coding formats, metadata standards and multimedia-enabled end-user devices, it is not likely that one single system is able to take decisions regarding the need for and type of adaptation needed for every possible context and situation. As such, one essential aspect to be considered is the interoperability between systems that are used in different situations and can realize both generic and specific operations. This problem was addressed in the ENTHRONE project [31], [121], [119], [8], [120], [118], [61], through the development of the EIMS, the already mentioned middleware service-oriented infrastructure for ubiquitous content mediation and access. The ENTHRONE project has already entered its second two year phase. During phase 1, ENTHRONE has developed technology based on the MPEG-21 specifications, notably DIDL and DIA to enable service providers to deliver context-aware multimedia services. These developments will be thoroughly described in sections 5 and 6. Although it has implemented limited functionality, ENTHRONE has laid down the principles and the core infrastructural support for context-awareness content adaptation. As already mentioned in this manuscript, the work developed within the course of this thesis, has provided the foundations for ENTHRONE and part of that work was actually conducted within

the framework of ENTHRONE.

3.2 Statistical Multiplexing

3.2.1 Introduction

Statistical Multiplexing has been extensively studied by the research community worldwide as a mean to achieve important savings in bandwidth. In this context it has been especially directed towards video services over packet networks and television broadcasting.

With the advent of digital television, suppliers of TV programmes have to be able to offer content diversity and new additional services, such as internet access or interactive services, with as much quality as possible. And, if they want to remain competitive, they have to find ways of achieving this with the minimum extra costs, namely in channel bandwidth. It is therefore necessary to find ways of maximising the relation "*bandwidth efficiency versus quality*".

Traditional television transmission channels offer only *constant bit rate connections*, as it is the case of terrestrial or satellite broadcasting and cable networks (CATV). Video signals transmitted in this kind of channels are usually encoded at a constant bit rate, thus presenting variable picture quality throughout the duration of the programmes. The compression algorithm incorporates a mechanism to control and regulate the generation of bits. This mechanism uses a local buffer and continually monitors its fullness level. It prevents overflows and underflows to occur by adjusting the quantization step size as needed.

On the other hand, in the Internet, where A/V streaming and download applications are now consuming a great percentage of the available resources, variable

bit rate encoding schemes are becoming increasingly popular [101], [108], [54], [189], [116].

It is however possible, even in the type of TV transmission channels referred, to benefit from the advantages of a variable rate encoding mechanism. Exploring the variability in bit rate of individual video sources, increases the opportunity for improvements in statistical multiplexing gain. Therefore, even in fixed bandwidth transmission channels, it may be possible to obtain gains.

It takes only the aggregation, on a single flow, of a set of distinct variable bit rate encoded video sources (as for example, TV programmes or video clips downloaded on the Internet), through the use of a temporal statistical multiplexer before submitting that traffic to the transmission channel. The multiplexer dynamically distributes the total channel capacity among the several sources, taking advantage of the fact that independent video sources are highly uncorrelated. Bandwidth released by one source in periods of low activity, can be used by another source in a high-activity period. It is therefore possible to obtain a statistical gain which improve the relation "*quality versus efficiency*".

Typically, a 6 MHz RF analog channel supports the transmission of 6 MPEG2 TV sources, reserving 6 Mbit/s for each programme. This value has been considered as sufficient to guarantee good quality for all types of contents during most of the time, even for those more difficult to encode contents. However, these more demanding situations may not have a very high frequency of occurrence and, more importantly, they usually occur randomly, without correlations across different sources. Therefore this agreement is clearly not efficient: bandwidth is being wasted for a considerable period of the transmission time, while new or additional services are prevented from being transmitted.

On the other hand, if more channels are inserted in the multiplex, it will almost be certain that each programme will experience quality degradation, at least once during its transmission time (as they individual bit rates will have to smaller). This is, nevertheless, the scheme generally adopted by TV Service Providers. Our everyday life TV's experience confirms this fact: any subscriber of TV Cabo has certainly observed image dropping and annoying blocking artifacts that frequently appear on sports programming, high-quality documentary series or on PPV movies.

This approach is a *deterministic* multiplexing approach, where good QoS guarantees are maintained throughout the whole duration of TV programmes, at a cost of wasting a considerable amount of resources and leaving no extra space for additional services; or, where a more rational use of resources is made, at a cost of lowering the QoS guarantees.

Statistical multiplexing on the other hand, frees unused bandwidth, which can be used to allow a other sources to instantaneously increase its share of bandwidth during critical scenes, thus preventing picture quality degradation. Unused bandwidth may also be used to convey information from new sources.

3.2.2 Multiplexing approaches

Basically there are two different approaches to implement statistical multiplexing: open loop (no feedback) and closed loop feedback. The second method requires a feedback control path between the encoder and the multiplexer. This is the statistical multiplexing method that has already been put in practice for TV broadcasting.

In this method, the encoder is not allowed to operate in a truly VBR mode, but rather in a *VBR stepwise CBR* mode. The encoder may alter its bit rate to another constant value in a discrete manner, and only after being granted permission by the

multiplexer.

This mode works in a polling request-grant fashion. The statistical multiplexer has to be in constant communication with all encoders and has to continuously re-assign bandwidth to each one according to their requests. This imposes a rather heavy burden on the multiplexer and it prevents remote encoders operation, i. e., the encoding equipment and the multiplexer must be co-located. Otherwise, it would require the existence of transcoders near the multiplexer. This arrangement has obviously some disadvantages, notably regarding extra costs and the introduction of additional sources of errors in the chain.

This last concern (degradation of picture quality due to re-encoding) may however, have lost some of its relevance. Important advances in techniques for seamless transcoding have been developed [203], within the framework of the ATLANTIC project [156]⁴ and are being incorporated into commercial products.

Nevertheless, the extra costs and complexity, as well the introduced delay, continue to be an issue. Ideally, encoders should be set to operate in VBR mode, targeting a given constant quality, and the statistical multiplexing algorithm would work, as much as possible, independently.

3.2.3 The use of Probability Theory and Statistical Inference principles

The key issue in using statistics, either in communications engineering (and more specifically in the problem under investigation in this dissertation), or generically speaking, is to be able to make inferences about real-world events. As taken from the dictionary, *making inferences* is the *process of deriving a conclusion from*

⁴The ATLANTIC project was a successful EU funded project under the ACTS programme, which developed a networked MPEG-2 post-production TV studio. The proposer of this thesis participated in that project.

facts or premises. There are basically two main approaches to statistical inference - the *frequentist* and the *Bayesian*. They both can assume two forms - *parametric* and *non-parametric*.

Non-parametric statistical inference starts by imposing on the data a dependence (or independence) structure, without actually assuming any specific distribution. It is able to produce more robust statements, but it is often very difficult, if not impossible, to improve the model.

On the other hand, *parametric inference* starts by making assumptions about the distribution of the data, choosing a simple model with a small number of adjustable parameters. It then improves the model by adjusting the parameters based on our scientific knowledge.

The *frequentist approach*, also referred to as classical approach, uses an estimate of the unknown parameter to make inferences and addresses the uncertainty of the estimation by the frequency behaviour of the estimator. The Maximum Likelihood Estimator, *MLE*, is the most popular and widely used method of estimation.

The use of formal statistical modeling together with advanced statistical algorithms in communications engineering problems, has the potential of providing a powerful workbench for developing innovative and efficient control and management strategies. It is also suitable for making appropriate inference statements about the behaviour of systems and sources of data.

Statistical modeling includes the tasks of collecting data, extracting information from it, constructing adequate probabilistic models, quantifying and incorporating expert opinions, interpreting the model and the results and finally formulating the prediction of future events.

In essence, statistics aim ultimately at *formulating inferences*, which means *deriving*

conclusions from *facts* and/or *premises*.

The *facts* are the observed data and the *premises* can be represented by a probabilistic model of the system in observation. The *conclusions* or the *inferences* concern unobserved data, most frequently data that will occur in future events.

As such, in communications engineering, in particular in systems involved in the transmission of random data, the use of such framework can prove to bring important benefits for the efficient dimensioning of those systems, which in turn will be able to provide better quality and larger choice of services to the end-users.

In signal processing and in science in general, it is already a fundamental task to develop models for observed data and then to determine whether those models can conveniently describe the data. Data are quantitative measurements that we are able to collect upon the occurrence of events.

Whenever we perform experiments, make measurements or simply observe the Universe in action, we are collecting data. Then, having these data, we can use statistics - a set of mathematical tools - to extract information from the data. It is only then, using this information, that we are able to reach understanding.

But to be able to extract information from the data and with it to construct useful models, we need to have a good knowledge about the problem we are tackling and to know the properties of what we are looking for.

The problems that we are generally faced with, in communications engineering, and in particular the task addressed in this dissertation, involve random variables. For that reason, the information contained in the data sets that we will be looking into, is stochastic, meaning that it is unpredictable.

This seems almost as a contradiction, as the aim of the statistical framework is to

find a way to *predict* the behaviour of sources, which are the *unpredictable* random variables. In fact, with stochastic systems, no matter how much we might know about the past, it is not possible to predict with total accuracy the next value in a data set of a random or stochastic variable. It is however possible, to arrive to an understanding by analysing the experimental data sets, as, by definition, observed data is necessarily correct.

It is thus possible to construct models that are able to provide something definitive about stochastic information, more precisely the expected collective behavior of random variables. The model is able to quantify the expected behaviour and assess future values with respect to their probability of occurrence, which is generally useful to a certain extent.

A mathematical model for stochastic systems is expressed as a probability density function, *pdf*. The simplest way of understanding a *pdf* is to see it as the set of weights (*density*) of each possible value of a random variable. The weight or *density* of a value, indicates the probability that a value has to occur.

If the set of possible values that a random variable might assume is finite or countable, they constitute a *discrete pdf*. On the other hand, if the possible values are infinite, with adjacent values separated by an infinitesimal width, then they define a *continuous pdf*.

Probabilistic models in statistical analysis are the vehicle through which we relate the observed data with a scientific premise or hypothesis about real-world events. They describe the process that generates the set of observations and which, we believe, appropriately predicts future occurrences.

However, models are only an approximation and therefore, there may be several can-

didates to perform the task. It is thus common, for each problem under analysis, to consider several plausible models and then to assess which is the one that best describes the data for the problem under investigation. The model is able to transport the problem into a workable mathematical form and, therefore, it is only useful in practical systems, if it is characterised by a small set of configurable parameters.

In *Bayesian analysis* [4], [151], the statistical inference is obtained in the form of a *posteriori* distribution, which incorporates both the scientist beliefs and the observations in a well formed probabilistic framework. The model selection problem can be summarised by the *posteriori* distribution.

In a Bayesian framework it is therefore necessary to establish prior beliefs, in the form of probability distributions, on the models under consideration and on the parameters of each model. Sometimes the latter proves to be quite difficult. Another problem is the computation of the quantities that lead to, or allow, model selection - typically it involves the computation of integrals of large dimension that do not admit any closed form of analytical solution.

3.2.4 Probabilistic models of VBR video sources

Probabilistic models are a powerful tool to help to develop adjustable algorithms. They provide us with knowledge about the possible states of the players (in this case the video sources or television programmes) and therefore of their possible evolving states. This allows to predict future needs with some anticipation. Probabilistic models support rational decision-making using the principle of maximum expected utility and also support learning from incomplete data. They allow to combine prior knowledge with learning, thus providing a sound basis for decision making.

When the players are compressed video sources or TV programmes, probabilistic models, built upon previous knowledge and expressing uncertainty on the behaviour of the video source, should be used together with a content characterisation of the sources, and should be adaptable to new situations.

An impressive quantity of work has already been done in the area of video source modelling [126], [44], [43], [73], [34], [36], [23], [46], [115], [114], [126], [14], [104], [48], [146], [45], [160], [209], [162]. The approach that has been usually followed to find the appropriate family of probability models that best expresses the behaviour of video sources is the *frequentist* approach.

Based on training data, a probability family is chosen and then parameters of the model are adjusted in order to fit the observed data. Usually this is done at the frame time scale and the models and parameters are selected and assigned on a frame type basis.

The published work has almost invariably, addressed the modeling of MPEG1 VBR sources and, at a smaller extent, also MPEG-2 [102]. It is expected that the same principles, methodologies and conclusions will fully apply to MPEG-2 or other coding formats. However, to be able to perform a rigorous study and to apply with confidence the results obtained by simulation, it is advisable to work with actual MPEG-2 VBR sources and to devise appropriate models for these kind of video sources. It is also advisable to use, as training sets, experimental data obtained from sources addressing a wide range of characteristics, representative of real world TV programmes. Reference is made to the MPEG-2 encoding algorithm because it is the one presently being used within dTV applications. However, the same applies to other encoding algorithms, notably those which use is starting to spread on the Internet, as for example H.264/AVC [108], [101], which is also a candidate for future digital TV systems [53].

Several traffic models have been proposed for the characterization of compressed video sources. These models are usually directed towards intra and differentially encoded bit streams. The parameters of those models are frequently obtained by matching statistical characteristics of real video sources and the model under consideration.

Particular emphasis is usually given to matching the correlation structure of the sequence of number of bits per frame. It is known that correlation has a great impact in queueing performance of statistical multiplexers. Compressed video inherently exhibits different patterns of correlation. These are usually related to the type or content of the video source and to the compression algorithm.

The MPEG video coding algorithm uses both intra and predictive techniques. It generates a repetitive pattern of coded frames of different types: intra frames (**I**-frames), predictive frames (**P**-frames) and bi-directional predicted frames (**B**-frames). **I** frames are independently encoded, taking advantage of only spatial correlation within themselves. **P** and **B** frames are differentially encoded. **P** frames are predicted with respect to a past image. **B** frames are encoded using a prediction based on a previous and a future image. They thus take advantage of past and future temporal redundancy. Therefore, **B** frames usually achieve higher degrees of compression than **P** frames, and even higher than **I** frames.

When analysing compressed video sources with this kind of structure/pattern, the benefits of grouping frames of the same type to find the appropriate models for each type, become apparent. This thus leads to three different models, one for each frame type.

However, if two types of frames present similar statistical behaviour, or some form of correlation, between them, they can be put together in the same group. This is indeed the case with **P** and **B** frames. There is a strong correlation between the

two. On the other hand, **I** frames seem to be almost completely uncorrelated to the other two frame types. Accordingly, sometimes only two groups, and consequently two distribution probabilities and probability models, are considered.

In the vast literature on this subject, it is possible to find authors that use separate models for each type of frame and others who use only two models. In either case, the parameters of the models are often adjusted over time to capture the non-stationarity of video sources. The most widely referred probability distributions to describe the generation of bits per frame are:

- *log – normal* distribution, eventually with different parameters for each frame type or group;
- *gamma* or *weibull* distributions, whose parameters are obtained based on the *MLE* (Maximum Likelihood Estimation) for the distribution that fits better the observed data set.

The models which are commonly used to generate the probability distribution functions (*pdf*) that fit the experimental data set, fall into two main categories:

- *auto-regressive* (AR), *Markov* and *mixed AR-Markov*. For example, for the three different frame types, it could be used two *AR*(1) processes and a *Markovchain*; or three AR classes with time-varying coefficients.
- Time Expanded Samples (*TES*) models.

Although the mentioned *pdfs* are indeed the most commonly used for all frame types, some studies have concluded that, whereas **I** frames could be well described by them, that was not always the case with **P** and **B** frames.

Table 3.2. Defining ontology elements in RDFS and OWL

RDF Schema	OWL
Declare classes like <i>Country</i>, <i>Person</i>, <i>Student</i>, <i>Portuguese</i>; State that <i>Student</i> is a subclass of <i>Person</i>; State that Portugal and Spain are both instances of the class <i>Country</i> Declare <i>Nationality</i> as a property relating the classes <i>Person</i> (as its domain) and <i>Country</i> (as its range); State that <i>age</i> is a property, with <i>Person</i> as its domain, and <i>integer</i> as its range; State that Pedro is an instance of the class <i>Portuguese</i>, and that his <i>age</i> has value 48.	Declare classes like <i>Country</i>, <i>Person</i>, <i>Student</i>, <i>Portuguese</i>, <i>Stateless</i>, <i>MultipleNationals</i>; State that <i>Country</i> and <i>Person</i> are <i>disjoint</i> classes; State that Portugal and Spain are distinct individuals; Declare <i>HasCitizen</i> as the inverse property of <i>Nationality</i>; State that the class <i>Stateless</i> is defined precisely as those members of the class <i>Person</i> that have no values for the property <i>Nationality</i>; State that the class <i>MultipleNationals</i> is defined precisely as those members of the class <i>Person</i> that have at least 2 values for the property <i>Nationality</i>; State that the class <i>Portuguese</i> is defined precisely as those members of the class <i>Person</i> that have Portugal as a value of the property <i>Nationality</i>; State that <i>age</i> is a functional property.

Chapter 4

The why and how of Context-aware support

4.1 Concise statement of the problem

The problem addressed by this thesis is the ability of providing quality and context-aware multimedia services to end-users, while assuring efficient usage of resources.

The solution to this problem enables the fulfilment of the concept UMA or the realization of ubiquitous access. UMA translates as the access to any kind of digital content, using any kind of digital network technology, and allowing end-users to consume the content on any type of terminal device, wherever the user is. But this should not imply the need to confront the user with new and complicated technology details. It should rather expose users only to new resources or new ways of using and consuming those resources, maintaining essentially their usual mode of "doing things".

In the context of this thesis, *ubiquity* represents a concept similar to UMA, as defined

above. However, quite often, ubiquity, as in terms of ubiquitous computing, is defined as the merging of computers within the environments people inhabit, and thus within all moments of our everyday life (computers embedded in walls, furniture, home and office appliances, in anything on the streets, in public spaces or in the home, etc.). Moreover, these embedded computers should be connected via wireless technology, thus allowing to seamlessly create ad-hoc networks. This is the vision expressed by Mark Weiser [202], the pioneer of the term *ubiquitous computing*.

With the advent of the World Wide Web (WWW) and the proliferation in the mass market of a diversity of multimedia-enabled end-user devices, consumers increasingly expect to be able to access any kind of content anywhere and at any time, regardless of the capabilities of their terminals. To this reality adds the large volumes of multimedia content available on-line, presenting a multiplicity of formats and the ever growing acceptance of the network-centric paradigm from the general public. An access to content that meets user expectations and demands, must take into account all the different aspects of this heterogeneous scenario.

But this is not an easy task to achieve. Indeed, there are many different variables that play a role in this complex scenario and that should therefore, be taken into consideration on the road map to the ubiquitous access realization. On one side we have a large amount of content and services available out there to be searched, accessed and consumed. On the other end of the chain, we have a heterogeneous population of potential consumers, of multimedia-enabled devices used by those consumers and of real-world situations or environments where those consumers are involved in. In between, a number of different access network technologies satisfying the requirements for the transmission of multimedia content.

So, in fact, the initially stated problem is a complex one, involving a number of different individual problems, notably:

- identification, gathering and representation of useful and adequate low-level contexts;
- semantic description of high-level contexts and reasoning about low-level contextual information;
- suitability, feasibility and execution of meaningful content adaptations driven by context;
- interoperability layers to allow to build mediating systems that effectively promote the access to diversified content repositories, different network equipment and technologies and heterogeneous client devices;
- security and management of the digital rights, both of the owner of the content as well as of the user;
- user centered graphical user interfaces or generally aspects of human-computer interfaces (dynamically generated interfaces, multimodal interfaces).

4.2 Justification

We argue that the problem this thesis addresses hasn't yet been provided an efficient and complete solution. This is mainly due to the fact that its fulfilment involves research in many different areas and requires the combined use of multiple results and technologies. In chapter 3, "Review of the State of the Art", we have shown that substantial research has been, and still is being, performed in all of the areas of interest to this problem. However little intersection has been registered so

far between the different areas. It is not a trivial task to gather results and combine them in an integrated way to deliver increased functionality and widen the scope of application.

Accordingly, we have focussed our work on the provision of the architectural support for integration and interoperability. Although technology advances in each individual area are essential to the overall objective, if a proper combined use of that technology cannot be achieved, then its value will be minimum as far as it concerns that overall goal :

” Technology should not be valued for what it is but rather for what it does.”

Although, as we have seen in chapter 3, valuable research is being conducted worldwide and standards are starting to emerge, addressing the ubiquitous access challenge, there is still much work to do. In particular in what concerns the bridging between the human’s perceptions of the environment and real-world situations they are placed in, or of the quality of the content they are consuming, and the machine-readable representations/descriptions that are possible to build of those concepts.

Likewise important, and still not well studied nor clearly understood, is the influence of the different types of contexts to the successful achievement of an adaptation operation that meets user’s expectations and thus enables the ubiquitous user-centered access to networked multimedia content.

All these aspects need to be solved to effectively allow the seamless access of consumers, in particular non-technologist consumers, to all the possible variety of resources.

Accordingly, we have tried to address some of these issues, notably by studying the adequacy and availability of low-level context information and by investigat-

ing the semantic representation in machine-understandable form of real-world situations occurring in selected applications considered both useful and popular.

We have thus identified the need to study how to use, in a useful and adequate way, the directly acquired information about the context of usage (low-level contexts) and forms of representing it. Additionally we have identified the need for richer forms of describing real world situations or domain of interest (higher-level contexts). We have investigated the use of ontologies to build formal models providing the knowledge representation of real world situations.

The many international conferences dedicated to this problem clearly indicate the interest, usefulness and validity of conducting research in this area. Likewise, the involvement of standardisation bodies, as we have seen in chapter 3, developing tools and specifications that contribute to the overall aim of ubiquity, demonstrates the same. Moreover, dedicated working groups, as is the case of the Ubiquitous Web Domain group within the W3C [201], are being created to give the opportunity to researchers worldwide, to contribute in a consistent manner to the same objective.

4.3 Discussion

Why is it important to provide ubiquitous access to multimedia content?

We believe that the answer would be straightforward and pretty obvious to anyone, as indeed everyone would be able to profit from it.

Seamless access to any kind of content in this digital era is being regarded as a priority of many research programmes and administrations, as a means to improve the standards of education, services and aspects of social life for all citizens. But let us analyse it from two different perspectives: the providers or enablers of the eventual ubiquitous service and the users of the service.

Providers - Clearly the realisation of ubiquitous access allows content

providers to diversify the way they deliver their content, increasing their business opportunities. Traditional content providers may take advantage of multiple distribution channels and reach their customers at unusual time or places. This contributes to gain the fidelity of their audiences and even to gain new ones. On the other hand, large archives of multimedia content that could otherwise face the risk of becoming of no use, simply because they could not be queried and accessed with existing technology/solutions, will be able to be reached in disparate ways, thus enriching content owners or providers' business and expanding their offerings.

Users - the non-technologist user will be the one who will gain more with the fulfilment of the universal multimedia or ubiquitous access. And certainly, the vast majority of end-users fall into this category. Access to on-line information has already become indispensable in our daily working tasks in many different areas of activity.

Considering the present phenomenon of globalisation, a great number of professional activities and businesses are maintained across borders with people moving between sites and remotely conferencing and interacting with others, who possibly have different time zones and therefore are placed in different contexts with disparate network and technology access.

This need for seamless access to information is happening not only at work, but is rather becoming a constant presence and need in many activities of our everyday life. And, whereas in some cases it is already possible to gain easy access to information, or with some skilled knowledge in other cases, many times not even the talent of the most *technologist-aholics* can make it.

But again, it becomes extremely frustrating seeing or hearing our neighbour enjoying the benefits of knowledge (or pure entertainment) and not being able to do the same. Or not to be able to do something because we could not access the information we

needed to do it, and yet knowing that the information is out there.

Ubiquitous or universal multimedia access seems indeed to be a consensual commendable goal to pursuit. Interoperability and context-awareness can as well be easily justified as, in our view, they constitute means for the successful implementation of the former goal.

Interoperability means that Content Providers will be able to use and integrate new technology with their existing equipment. This will enable them to enlarge their offerings, as explained above, in a cost-effective way. It should be noted that the economic factor is extremely important, being the major driver for the decisions made by companies.

It also means that different end-user equipment will be able to interact with different content supplier's systems or among themselves, creating communities of users. And, of utmost importance, it enables different content suppliers to cooperate among themselves to provide richer services to the end-user.

Context-awareness provides the means for content mediation systems to be able to automatically adapt the content to suit different needs and situations. By devising efficient forms of using context, we are contributing to the development of systems that can automatically react to changes of the sensed context in ways that better meet the expectations of end-users. The fact that it enables automated operation envisages cost-effectiveness and opens the door for real-time operation.

Chapter 5

Work developed and proposed solution

5.1 Concept

The approach chosen to address the problem formulated in the previous section was to develop techniques for building the architectural support for quality and context-aware multimedia services. The adopted approach relies on the use of distributed technologies to build a Service Oriented Architecture (SOA) based on open standards to represent services, content, context and descriptions.

A complete framework was elaborated, from the identification of usage scenarios and elicitation of associated requirements, through the formal specification of the necessary functionality, mostly in the form of services, to the development of a prototype system and the conduction of tests. The main contribution of this work is the identification of generic services integrated in a common reference architecture to enable the implementation of adaptable, quality and context-aware multimedia services.

That common reference architecture was achieved as the ENTHRONE Integrated Management Supervisor (EIMS) software system [31], [121], [120], [119], [8], [118], [61]. The EIMS can be seen as a *context-aware* content management and mediation platform based on MPEG-21 [94], [96] and distributed technologies [198]. Its goal is to provide transparent access to multimedia content and services, offering a quality-controlled delivery.

To achieve this, it needs, on one hand, to be able to interpret content-related metadata, to locate the useful content and to learn its characteristics and the operations possible to be performed upon it. Likewise it needs to process and interpret context-related metadata to learn about the terminal and networks characteristics, as well as user preferences and possibly natural environment conditions. Only in possession of this information it is possible to adapt the content to meet the current context constraints. To operate in heterogeneous environments, it is desirable to have a single common framework to express and convey all this metadata. This common framework is the MPEG-21 standard.

The reason behind this choice is twofold: 1) the need to implement an interoperability layer to bridge across different content formats, repositories and databases; 2) the interest of handling in an integrated and consistent manner all types of information in play: content, content-related metadata and contextual information. As we have seen in section 3.1.3, MPEG-21 provides a mean to accomplish these aims.

Each unit of content meant to be made available to the end-user may consist of a number of resources of different media types. In order to be efficiently searched, it needs to be indexed and stored together with (or linked to) its descriptions. A mechanism is needed to enable the declaration of the parts that make up each content unit, and a publishing model must be set in place to allow handling the variety of

items that may be published to it.

Among the available emerging standards for content representation and description, MPEG-21 can be considered as one of the most advanced, complete and flexible, and therefore of great interest towards the objective of efficient use of multimedia repositories and universal multimedia access.

On the other hand, the EIMS needs to interact with a number of different entities, namely content providers, network operators, service providers and end-users, who are involved in the delivery of a multimedia service across networks. For this reason the EIMS is composed of different sub-systems, with well-defined functionality, each one interacting with one of those entities.

Functionality that is meant to be externally accessible is offered as services, either to other EIMS sub-systems, to other ENTHRONE systems or to external applications. The EIMS thus presents a services-oriented architecture (SOA), which facilitates its usage in different environments and contexts, and effectively enables the access of multiple, heterogeneous client devices to multiple heterogeneous sources of content. The Simple Object Access Protocol (SOAP)[135], is used for the exchange of messages between sub-systems or external applications. Web Services Description Language (WSDL) [159], is used for the specification of service interfaces. EIMS functionalities can be directly accessed and combined to support the requirements of different applications (such as unicast VoD, multicast NVoD, file downloading, real-time streaming, news-on-demand, broadcast iTV, etc).

In parallel, we have studied the usefulness and adequacy of using directly acquired information about the context of usage (low-level contexts) and forms of representing it. Additionally, we have investigated the use of ontologies to build formal models providing a knowledge representation of real world situations or domain

of interest (higher-level contexts).

This framework answers the needs of more complex and demanding context-aware applications by empowering our mediation system with a formal knowledge model, through which it would be able to automatically derive higher level contexts based on the sensed contextual information. The incorporation of semantic concepts and ontologies was evaluated as a mean to enrich low-level descriptions of the context of usage.

One of the contributions of this line of work, is the use of concepts of the Semantic Web (Web 2.0), implemented through the use of W3C specifications, notably OWL (Web Ontology Language) [127], with MPEG-21 tools. The advantage is the possibility to describe more accurately, real-world situations where the user is engaged in, through the use of semantics, while retaining the benefits of using MPEG-21 as an efficient representation format for complex multimedia objects, as well as its versatile and complete content adaptation framework as explained in section 3.1. Accordingly, adaptation decision mechanisms will be better informed to provide orientation towards more meaningful content adaptation operations that satisfy constraints of real-world situations and meet users' expectations.

Finally another addressed topic, relevant to the ubiquitous access to content, was the efficient use of resources in A/V networked applications. The adopted approach was the one of building a probabilistic framework and of generating a great number of statistical models capable of accurately describing a great number of video sources, and therefore offering the possibility of being able to predict, up to some extent, the behaviour of those sources in terms of bit rate demand against quality. This work was conducted prior to the development of the reference content mediating platform. Its interest to the platform was limited to a reduced number of

application scenarios, namely those involving the broadcasting of TV programmes. For practical reasons, and because it was not essential for the achievement of the proposed ENTHRONE results, it was not possible to progress the work in that direction. Accordingly, the corresponding developed work is not described here in this central implementation chapter, but are rather left to be described in the Annexes to this thesis.

This section thus provides the description of the functional design of the EIMS, followed by the description of the implementation of its modules in terms of methods and interfaces developed to deliver the identified functionality. It continues by reporting the work performed on the study of contextual information and the specification of an ontology suitable to provide sufficiently accurate descriptions of the identified situations and needs for content adaptation. Tests conducted and results obtained are presented in chapter 6.

5.2 Relevant Usage Scenarios and Requirements

This subsection provides a description of selected application scenarios and associated use cases, where adaptation of content to meet constraints imposed by the usage context, would allow to enhance the quality of the user experience. These scenarios were developed for the ENTHRONE project and some of them partially implemented.

A description of each selected scenario is provided, as well as the identification of different contexts or real world situations that may occur during the consumption of those applications. Each situation presents a different set of conditions and characteristics of the usage environment. Accordingly it identifies possible adaptation operations that could be applied to each situation, to satisfy the relevant constraint.

Although a great number of usage scenarios have been developed within ENTHRONE, many of them can be seen as a variant of another one, imposing essentially the same kind of requirements upon the EIMS. Accordingly, in this section, we will describe only two application scenarios that have been identified as of interest to provide useful feedback concerning the use of contextual information and the role of the EIMS towards the successful fulfilment of those scenarios. They provide us the motivation for the study and identification of relevant and useful contextual information, and the development of our content mediation platform. They provide the requirements from which the design of our system grows and the guidelines on how to build validation tests.

Even though the two selected scenarios may not be very different among them (in fact they share a lot of functionality), each one presents unique features that in turn impose specific requirements on the EIMS.

5.2.1 Video on Demand

Definition The Video on Demand (VoD) service is offered by an independent Service Provider. The SP manages its own group of subscribers and offers them an access portal. The SP maintains its own media assets as a data base and repository of Digital Items. The SP may not have the actual A/V content stored locally. As a minimum requirement, the SP asset repository and database will hold descriptions associated to each media asset and their location. A candidate format for holding this information is obviously the MPEG-21 specification. In that case, the SP repository would hold an MPEG-21 DID associated to each media asset and information extracted from those DIDs. Each media asset would then be referred to as an MPEG-21 Digital Item.

The content may reside in the original server (possibly owned by a Content Provider)

or it may even not exist physically, but rather be generated on the fly by a real-time encoder (possibly located at a CP premises). In addition, the SP may have agreements with other SPs, who themselves maintain a DID repository and database, in order to offer a wider choice to its customers. As a result of these agreements, the SP may mediate the access of its subscribers, also to the repositories of those collaborating SPs and CPs. It is the responsibility of the SP to ensure that the QoS level as agreed with its customer is maintained throughout the service lifetime, even in the case of a DI being supplied by a collaborating SP.

Generic description of the service The generic goal of this service is the one of offering a real-time Video On Demand service in both fixed and mobile environments where the access to content is controlled (using, for example, MPEG-21 REL[55]). "*Real-time*" implies that the DI (the selected A/V content) is to be streamed in real-time to the user, so that the user will be able to start watching the movie once the transmission begins (or in realistic terms, with a given, minimum latency). Moreover, it is assumed that the movie will not be stored in the user's terminal.

Users of this service can be equipped with terminals exhibiting different capabilities and may have various types of network connectivity where the conditions may vary along the duration of the service. Aspects of session mobility are addressed, as for example, the case of a user on the move equipped with some limited capability device, who arrives home and switches to a richer multimedia enabled device with better network connectivity.

This service allows registered customers to search for content based on free keywords. User can be equipped with any type of terminal, with the minimum requirement of having installed a Web browser and some type of A/V codec. Upon receiving a list of matching results, the user will be able to browse individual DIs to obtain more

Table 5.1. Service Parameters for the "VoD" service

Service type	Content type	Content format	Channel/network	Distribution mode	End-user terminal
Video on Demand	A/V	MPEG-21	xDSL	Pull	PC
		MPEG-4 file format	Cable	Pull	mobile phones
		MPEG-1/2/4 systems	UMTS		

information on them. Typically, high-level semantic descriptions, providing more details about the contents of the movie, thus helping the user to decide which one to select. Once the user has made his/her selection, the request is sent to the SP. The user will be able to select one among a number of possible quality levels for the delivery of the service. These levels are decided by the SP, who associates a price to each one of them. The service will then be initiated by the user after being notified by the SP that the contract has been agreed. During the service lifetime the user may switch terminal and continue to watch the movie from the point where he/she was.

Service Parameters Table 5.1 provides a list of possible parameters of the *Video on Demand* service. It provides a minimum set of alternatives in terms of content formats, networking technologies, distribution modes and terminal types that are suitable for the deployment of this service. Nevertheless, other different parameters could also be appropriate for this kind of service ¹.

¹each column should be read separately, providing a number of alternatives for the different aspects addressed.

Requirements The following list of requirements has been identified for the successful implementation of this service use case:

- the terminal needs to have network connectivity;
- the terminal needs to have installed a Web browser and one or more A/V codecs;
- the terminal should provide PVR functionality;
- the SP should be allowed to decide whether or not to use context-aware content adaptation;
- the SP may have agreements negotiated with other providers to allow its customers to retrieve content on remote locations;
- the SP is responsible for managing the access to remote resources by its subscribers. The contract established between the SP and each customer, defines the ability to access remote content;
- the location of the remote server needs to be known at the time the user selects a given file for download;
- communication capabilities need to be provided between the SP and the selected content server, to negotiate the availability of that server;
- the SP needs to have the means to find the delivery path and the resources availability on that path;
- the SP needs to obtain a description of the terminal's capabilities and the user preferences;

- the SP is expected to support the allocation of resources to fulfil the chosen QoS;
- the SP should support QoS monitoring and provide for reactive actions, should the terminal experience QoS degradation.

The added value introduced by EIMS:

- extension of the offer to other SPs and/or CPs domains content. An important benefit from the user point of view is that he/she will only subscribe to one single SP and does not need to know where the content he/she has selected is coming from;
- monitoring of the end-to-end QoS;
- support for heterogeneous terminals;
- support for different networking technologies;

Service use case walkthrough The different steps of a simple, demonstrative use case of the VoD service are described below. This use case already takes into consideration the role of the EIMS.

The user accesses the SP portal and registers him/herself. The EIMS receives the correspondent authentication information, identifying the profile of the user and creating a session associated to that user. The user can then browse through a list of recommended movies or he/she may submit a keyword-based query. In any situation, the user will be presented with a list of available movies. Each movie will be represented as an MPEG-21 DI with distinct high-level, content-based semantic annotations. He/she may browse individual movies (represented as Digital Items) to obtain more information about each DI to help him/her decide what to choose.

Once the user has reached a decision, he/she sends a request to the EIMS and is presented with a list of possible quality levels for the deployment of the service. Each quality level has a given price associated to it. It should be noted that, even in the case where the user selects the best service quality level, the usage environment conditions may not allow the EIMS to actually grant him/her the service with that chosen quality. The EIMS gathers required contextual information (from the terminal, networks and available adaptation engines) and decides if and what kind of adaptation is necessary and possible to be performed upon the selected DI, to acquire the adequate characteristics to be delivered to the user. The user is then notified of the conditions of the contract (which includes the final quality level and price). If in agreement, the EIMS finalises the set up of an appropriate QoS-enabled path and the user initiates the service by clicking on the link provided by the EIMS. Considering protected content, before the streaming begins, the user needs to acquire a valid license for the content he/she wants to consume. The EIMS mediates between the user and a License Server, the acquisition and download of that license.

Once the service begins, the EIMS starts supervising the quality level of the service. If the quality drops below the agreed value, the EIMS initiates reactive measures to minimize the problem. Possible reactive measures could include the temporary re-scaling of the video, reducing the bit rate thus potentially decreasing the occurrence of errors and loss of pictures at the receiver. This measure should be accompanied by a notification to the end-user and a consequent reduction of the service fee. Other alternative could be to switch to a different network path.

5.2.2 Media file downloading

Definition This service covers the delivery of content under a file format through a broadcast or Internet link for storage and later usage on the end user terminal. The

service can be operated on a "On-Demand" basis, but can also be a push service to several users who subscribed the service. A typical application in this service is the request for a movie, similar to a Video on Demand service, but for which the delivery can be more or less delayed according to the type of services the user has contracted with the Service Provider. As in the case of the VoD service, it is assumed that The SP manages its own group of subscribers, offering them an access portal. It manages its own repository of Digital Items (one DI in this case is a collection of one or more files with A/V or multimedia content, where all files share the same high-level semantic descriptions). In addition, the SP may establish contracts with other SPs and/or Content Providers (designated as collaborating SPs and/or CPs), thus enlarging the media asset that can be searched as accessed by its customers.

Generic description of the service This service assumes that the consumer will be able search and browse remote repositories and finally select a file for downloading. The user will then have access to a menu of options to select from, such as:

- scheduling of the download: download the selected content immediately, in an hour, in a day, in 48 hours etc;
- selection of the type of service in terms of quality/speed.

Each of these options will have a price associated to it. For example if a user wants to download content immediately via satellite, the price may be high. Whereas if the user may tolerate a delay of, for example, 48 hours the price could be significantly lower. In that case, the use of the satellite (or other transmission medium) can be scheduled differently, trying to use less busy periods of the day and even using a lower bit rate. This would saving bandwidth and "premium time" expenditure, thus enabling to cut down the price to a fraction of the *download immediately* scenario.

The provision of this service involves both aspects of end-to-end QoS provision at the network level, as well as adaptation of content towards the user preferences and/or terminal capabilities. This last aspect also contributes to the quality of service experienced by the user.

In the case of pushing content to content consumers (typically a multicast scenario), a number of files containing either video or other types of content, are stored in a server at the Service Provider site. The files have been selected by the SP using their customers profiles and represent, as much as possible, the content consumers preferences. The SP wishes to download this content to the content consumers' terminals, which may be PCs, STB (Set Top Boxes) or mobile terminals. The user is able to access the content and interact with it after it is completely downloaded to his/her terminal. To some degree, this service is similar to the provision of TV services, but without the stringent bandwidth requirements of a real time video service. Even though the content may be a video file, it can be downloaded to the content consumers terminal in due time and after the completion of the transmission, the user is able to watch it.

Off-line content adaptation may be performed (according to the Digital Item Adaptation (DIA) framework), before starting the transmission. This is, if necessary, depending on the usage environment characteristics, the A/V content may be adapted to suit those characteristics and a new file generated. Accordingly, it is assumed that the SP is aware of the network configuration, location of the consumers to whom the service is to be pushed, as well capabilities of the terminals they are equipped with. In this way, he is aware of the connection path(s). Before initiating the transmission, the SP needs to gather information about available network resources in the connection path. If there network resources availability is suitable for the volume of the information to transfer (i.e., if it does not take too long to download the file), the

appropriate transmission parameters are chosen and passed to the multicast software to start the transmission.

If the resources for some networks permit, the SP may start multiple transmissions according to the available bandwidth in the paths. The content will be provided on a carousel basis, to allow the downloading of content to consumers who may have missed the beginning of the transmission. The whole transmission will be IP or IP encapsulated over MPEG-2 TS, to enable content consumers connected through a DVB network to access the services.

Due to the nature of the service, the bandwidth required for the push service can be much smaller than that of a TV service. However, after the bandwidth has been defined, it has to be respected; otherwise the conditions established in the contract may not be satisfied or it may even happen that the data is received with errors.

Service Parameters Table 5.2 provides a list of possible parameters of the *Media File Downloading* service. It provides a set of alternatives in terms of content formats, networking technologies, distribution modes and terminal types that are suitable for the deployment of this service. Due to the broad scope of this service (anything can be downloadable!), a considerably large number of alternative formats are indicated and even more could be thought as being appropriate for this kind of service.

Requirements The following list of requirements has been identified for the successful implementation of this service use case:

- the end user terminal needs to have a file system;

Table 5.2. Service Parameters for the "Media File Downloading" Service

Service type	Content type	Content format	Channel/network	Distribution mode	End-user terminal
Media file downloading	A/V	MPEG-21	POTS	Push	PC
	Video	MPEG-4 file format	xDSL	Pull	Mobile phones
	Audio	MPEG-1/2/4 systems	Cable	Bi-directional	PDA
	Image	MPEG-1/2/4 video	OTA	Interactive	
	Graphics	MPEG-1/2/4 audio	UMTS	Fixed	
	Text	HTML	GPRS	Mobile	
	...	JPEG SVG ...			

- the terminal must be aware of when the requested content is delivered;
- some triggering mechanism is necessary at the terminal level to start accepting the file;
- applications should be allowed to store/retrieve content to/from the terminal persistent file system;
- the SP should be allowed to decide whether or not to use context-aware content adaptation;
- the SP may have agreements negotiated with other providers to allow its customers to retrieve content on remote locations;
- the SP is responsible for managing the access to remote resources by its subscribers. The contract established between the SP and each customer, defines the ability to access remote content;
- the location of the remote server needs to be known at the time the user selects a given file for download;
- communication capabilities need to be provided between the SP and the selected content server, to negotiate the availability of that server;
- the SP needs to have the means to find the delivery path and the resources availability on that path;
- the SP needs to obtain a description of the terminal's capabilities and the user preferences;
- the SP needs to have the mechanisms to schedule the delivery of the DI as indicated by the user and accordingly activate the delivery at the appropriate time.

The added value introduced by EIMS:

- seamless adaptation of the content to the user preferences, terminal capabilities, usage environment;
- access to multiple networked repositories of content without the need to know their locations;
- optimisation of transfer time;
- delayed downloading;
-

Service use case walkthrough A simple, demonstrative use case, considering the involvement of the EIMS can be described as follows:

- A user selects a DI² to download and the resource of that DI is a multiplexed MPEG-2 file (e.g. conformant to the MPEG-2 Program Stream syntax [88]). That DI contains a video and an audio MPEG- 2 elementary stream. The DI was chosen because the video resource matched the search criteria expressed by the user in semantics terms (the user does not need to know that the video is multiplexed or the actual format of the resource). However, through the use of the DIA description tool "CodecCapabilities" (a DIA descriptor tool belonging to the sub-class "Terminal Capabilities", which in turn belongs to the class "Usage Environment Description Tool"), the EIMS realizes that the users terminal does not have the capability to demultiplex MPEG-2 Program Streams. Therefore, the EIMS, upon collecting all

²each A/V content unit with a given semantic description is a DI. Different variations, in terms of audio or video encoded bit rate or encoding format, may exist stored in different files. However the user performs his/her search and selection based only on high-level semantic descriptions and not on technical specificities. For him/her, a DI is a downloadable file, with a given set of high-level, content-based semantic descriptions.

necessary contextual information (bandwidth availability on the network path, capabilities of the terminal and characteristics of available adaptation engines), decides the best possible adaptation, selecting the service parameters. It then invokes the appropriate adaptation engine to perform the correspondent adaptation operation upon the selected DI before download occurs.

Before the download takes place, the user has to purchase the content. The price will vary according to the downloading options chosen, and the intended usage of the content. The EIMS coordinates this process, interacting with the user and with a License Provider. At this point, a license, an MPEG-21 REL (Rights Expression Language)[55] license, for instance) is created and stored in a License Server, describing the usage rules the user is subject to, when accessing the content. Decryption key(s) necessary to get access to the content are contained in the license. Once the download is complete and the user wants to access the content, the license is obtained from the License Server through the EIMS and parsed on the Terminal. User intentions on the content are validated as regards to the usage rules stated in the license, the decryption key is extracted and access is granted.

5.3 Platform functional architecture

The main purpose of this chapter is to present an object oriented analysis and design of the context-aware content mediation platform. It is referred as the Enthroned Integrated Management Supervisor (EIMS).

This system was originally conceived as an interoperable platform able to mediate the ubiquitous access to networked multimedia content, and proposed to be implemented within the ENTHRONE project. The identification and characterisation of a number of real world application scenarios and services within Enthroned, among which the ones described in the previous section, has made possible to enhance the

proposed architecture in terms of required functionality as to support the identified application's requirements. It also made possible to deploy and test it within a complete end-to-end multimedia content value chain, and thus assess its adequacy for the targeted goal. Of utmost importance was the possibility to test the developed platform in a demonstration environment comprising user terminal able to consume MPEG-21 Digital Items, networking edge equipment and multimedia content sources and adaptation engines.

ENTHRONE has therefore provided significant contributions by 1) refining the functionality delivered by the platform; and 2) acting as a testbed, providing the means to assess the validity and adequacy of the chosen approaches and consequent developments.

Whereas the purpose of analysis is to investigate the problem, design emphasises on a logical solution, i.e. how the technology, in this case the Integrated Management Supervisor of ENTHRONE (EIMS), is able to fulfil the requirements. The analysis was done both in section 3, as well as in the previous subsection with the descriptions of relevant usage scenarios.

This sub-section provides the functional description of the developed platform using a UML (Unified Modelling Language) approach [103]. The UML is a notational system, with specified semantics for its notations, to model software systems using object-oriented concepts. It is derived from the notations of three object-oriented design and analysis methodologies, more specifically the *Grady Booch's* methodology for describing a set of objects and their relationships [66]; the *James Rumbaugh's* Object-Modeling Technique (OMT) [161]; and the *Ivar Jacobson's* approach for use case methodology [105]. UML is now an accepted standard of the Object Management Group (OMG) [140]. The process we have adopted for the design of our content

mediation platform involves the following steps:

- Requirements and usage scenarios analysis;
- Deployment models;
- Use cases definition and description (event flows);
- Collaboration and sequence diagrams.

This section provides the design and specification of the functional components of the EIMS and information on how these components interoperate to perform their intended tasks. The requirements extracted from the above described usage scenarios, enabled the definition of use cases that transform the requirements into a form that maps well to the software designer's environment (implementation). This design phase allowed the definition of logical software objects that have subsequently been implemented in an object-oriented language, more specifically JAVA.

Due to the iterative development methodology adopted, design has been refined along the development phase. Accordingly, the subsequent implementation, described in the next section, may not match exactly the functional specification here presented. Nonetheless, the most important use cases, included in the Use Case Model of the EIMS, remain essentially the same throughout the design phase onto the implementation phase. The Use Case model contains only the use cases that were thought to be more relevant from the architectural point of view, thus providing the foundation model of the EIMS. Building upon this model, further functionality has been gradually added.

We start by describing the logical view of EIMS in terms of its subsystems and layers. These elements present a logical partitioning of the system and almost do

not imply any particular deployment pattern by themselves. This logical description includes the user requirements identification, the use cases definition and description and the collaboration and sequence diagrams.

A clear understanding of the EIMS logical description requires that a brief overview of the complete ENTRHONE system is provided (in terms of actors and constituent sub-systems, from which the EIMS is the central one). We then proceed with the description of the possible deployment patterns of EIMS.

Although these two views are not completely independent (deployment typically does have some impact on the logical view), they are also not identical and therefore any question of possible deployment model is applicable to the EIMS as a whole, rather than to any particular module.

Depending on the business model in view, the deployment of the various modules of the EIMS can be partitioned or non-partitioned, distributed or centralised, rich or thin client based. We will provide further ahead some examples of deployment models for some of the described usage scenarios.

The primary goal of the ENTHRONE project is to provide a solution for the seamless access to multimedia content with end-to-end quality of service through integrated management of content, networks and terminals, respecting intellectual property rights (IPR). To achieve this goal, ENTHRONE provides an off-the-shelf infrastructure for end-to-end QoS management, where the EIMS plays the central role.

A Service Provider (SP) may build a complete system on top of ENTHRONE, but ENTHRONE does not provide all the components needed by a service provider. In particular, a SP has to add a Front-End through which a user will be able to access services. This is designated as Service Providers Front-End (SP-FE).

Although being highly dependent on the type of business model adopted [31], in EN-

THRONE we have assumed that the entity that makes the content available (in the form of DIs) for the end-user to search, browse and play with it, is always a Service Provider. Content Providers are seen as the entities that only produce content. Some Content Providers may assume also the role of Service Providers, while others will always rely on agreements with third-party Service Providers to expose their assets to the public.

Accordingly, it is assumed that the most straightforward way of using the EIMS is to install it, or install its core functionality, at a Service Provider site, who will need to establish agreements with Content Providers to mediate the access to content. Likewise, SPs also need to establish agreements with the other entities involved in the content value chain, if they want to deliver networked access to protected content to the end-user: Network Providers (NP), License Authorities (LA) and Consumer Electronics (CE) Manufacturers. The latter are not directly considered in ENTHRONE, as the interaction is made directly with the terminal equipment of the end-users. Nevertheless, a successful deployment of ENTHRONE may also involve off-line, *a-priori* agreements with CE manufacturers. This would allow to have some EIMS modules natively incorporated into the end-user devices.

While specifying the functionality and architecture of the EIMS, several assumptions were made, some of which have already been mentioned in the above paragraph. To make them explicit, they are enumerated bellow:

1. ENTHRONE is neither a Business nor an Operational Unit and does not assume any responsibility for these aspects.
2. The EIMS provides an off-the-shelf infrastructure solution, i. e. a system constructed from off-the-shelf tools, namely generic PCs, to support the E2E QoS management and ubiquitous content retrieval.

3. Whatever organization takes direct or indirect responsibility for business and operational aspects and builds a complete solution on the top of ENTHRONE is called a Service Provider (SP).
4. The Service Provider is a role, not necessarily a stand-alone organization.
5. The Service Provider ties together Network Operators, Content Providers and License Authorities in order to supply the E2E QoS and context-aware solution to End Users.
6. The Service Provider Front-End (SP-FE) is a supplementary subsystem, which communicates with ENTHRONE, and that together with ENTHRONE constitutes a complete system. Although its development could be sub-contracted to any third party organization, we still consider it as being owned by the Service Provider³

The EIMS actors

ENTHRONE has a layered architecture, with the EIMS as the top layer, implementing core functionality for the ubiquitous access to multimedia content. Figure 5.1 illustrates the layered architecture of ENTHRONE and the actors with which it interacts. That figure highlights the modular architecture of the EIMS, which comprises four distinct sub-systems with well defined functionality and with standing APIs to interact with external modules or actors. The EIMS orchestrates

³For practical reasons, the project has also developed its own SP-FE. Nevertheless, it is not the intention of ENTHRONE to design, nor of its EIMS platform to provide, the functionality of an access portal. This is left to the SP, as it is directly related with the business model the SP wants to adopt and to the way it wishes to present its services to the user. Moreover, it is foreseen as desirable from the SP side, that this functionality is not "imposed" by the EIMS, and that the SP is free to continue to use whatever system it had, before migrating to the EIMS.

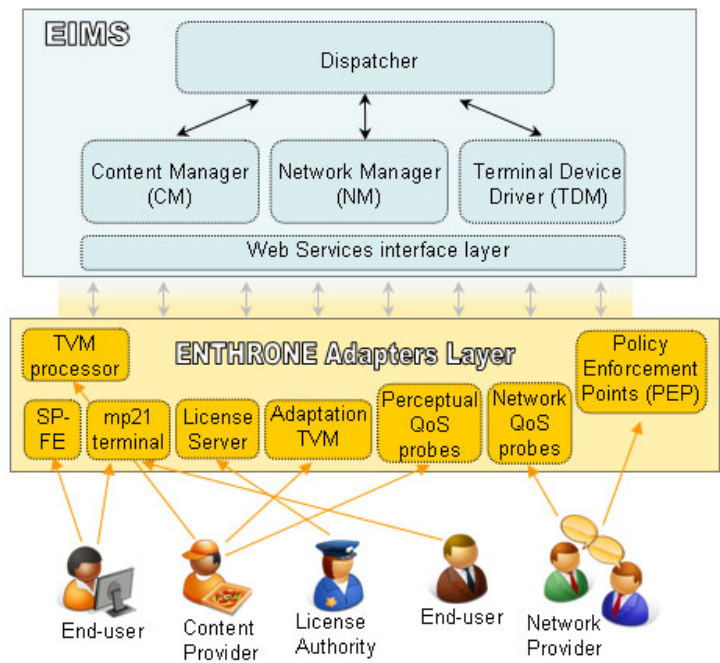


Figure 5.1. The layered architecture of ENTHRONE

the end-to-end QoS management process delegating the actual tasks to ENTHRONE adapters, such as end-user terminals, routers or TVM (Television and Multimedia) processors.

The diagram of Figure 5.2 shows five actors that have been identified at a high abstract level, as the people or things that interact with EIMS and therefore to whom or to what EIMS delivers value. The EIMS presents a modular and layered architecture as it can be seen in the diagram of Figure 5.2. Each sub-system has a well defined role. Whereas the EIMS-Dispatcher can be regarded as the brain of the system, the other sub-systems are meant to interact with the specific entities that participate in the delivery of networked multimedia services, as explained above. The role of each sub-system will be described in detail further ahead.

The main idea behind this high-level architecture is the one of obtaining a distributed system that can interact with the key players in the multimedia content value chain, being deployed in varied forms according to the services to be offered and the business models to be adopted. Moreover, this system should be able to offer additional functionality on top of existing equipment, enabling those entities to deliver added-value multimedia services to the end-user. In particular the following additional functionality was identified:

- publishing of multimedia content according to a common and open format in externally accessible (*queriable*) repositories;
- functionality to allow any user equipped with any kind of networked multimedia-enabled device to access the offered services, thus being able to search, browse and consume that published content;
- possibility of implementing multimedia services sensitive to the context of usage, therefore being able to collect and process contextual metadata coming

from users (preferences in terms of quality versus price), terminals (in terms of capabilities such as supported codecs or screen size) and networks (capacity and conditions);

- possibility to maintain a negotiation procedure between the user and the provider of the desired service
- offer the ability to monitor the offered service throughout the service lifetime, assessing the level of the agreed QoS or the context of usage characteristics and accordingly react to changes to quality degradations in the context of usage;
- allow different entities to share these tasks, maintaining an on-line cooperation through the exchange of information (mostly contextual descriptions) using open protocols and common data formats;
- possibility of achieving flexible forms of business model configurations, thereby allowing either one single entity to assemble all the business roles within the content value chain (generation of content, including Content Providers, License Authorities and Content Adapters, Network Operators and Service Providers) or having different physical entities for each of those roles.

The devised architecture as shown in Figure 5.2 seems to set up the path for providing an answer to these requirements. For each of the roles or entities in the content value chain a specific high-level complex module, designated of sub-system, has been included:

- *Dispatcher*, the master sub-system, directly communicating and offering functionality to Service Providers and serving user requests;
- *Content Manager*, *CM*, the sub-system responsible for managing the interactions related to content access and generation;

- *Network Manager, NM*, to allow the interoperation with Network Providers;
- *Terminal Device Manager, TDM* to provide a platform-independent interface to varied terminal devices.

These EIMS constituent sub-systems interoperate among themselves, and also interact with external entities, the *actors* of the system, to deliver the intended value. Actors are entities outside the system, persons or things, interacting with or stimulating it and using the value delivered by the system. Five major actors have been initially defined. The identification of the main value these actors expect to obtain from the system, has made possible to draw the core functionality of the EIMS, via *Use Cases*, and thus to arrive to an initial formal functional specification of the system. Below is the list and description of the EIMS main actors, followed by a description of the actions they are expected to maintain with the EIMS.

- Administrator (Admin)
- Service Provider Front End (SP-FE)
- Content Provider (CP)
- Network Operator/Provider (NO/NP)
- License Authority (LA)

The actor *Admin* represents the ENTHRONE system administrator, performing various house-keeping tasks. In particular, it performs operations typically under the responsibility of a system administrator, such as user registration and account management, as well as content management tasks, as for example, deletion of outdated

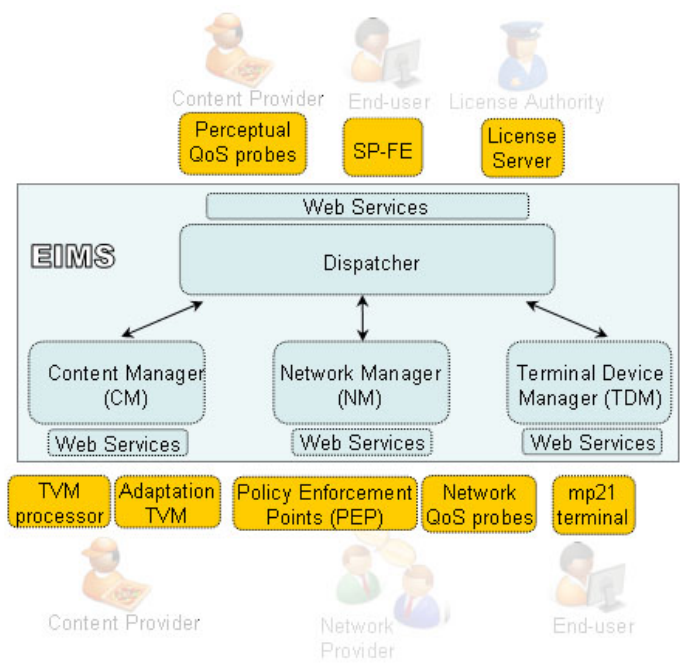


Figure 5.2. Actors interacting with the EIMS

content.

The *SP-FE* is a supplementary sub-system, which communicates with ENTHRONE on behalf of the end-user. The association of the SP-FE with all the ENTHRONE sub-systems, constitutes a complete system. The SP-FE is used by an end-user to access ENTHRONE services. An end-user represents a human actor who, while not directly interacting with the EIMS, is the ultimate beneficiary of the functionality delivered by EIMS, when retrieving Digital Items. The term Digital Item (DI), as it is used here, is defined in the MPEG-21 standard as described in section 3.1.3. It is via the SP-FE that the end-user initiates a request for the retrieval of a DI of a particular type he/she wants to consume. The SP-FE interacts with the EIMS exchanging information such as:

- user credentials;
- requests for DI search;
- list of DI choices;
- DI play request;
- PQoS measurement results;

A *Content Provider* represents a third-party system, which stores, manages and delivers DIs. DIs are presented to the end-user through the DI Browser, an ENTHRONE component belonging to the terminal. This component is described in [122]. DIs are owned by Content Providers but are published and made available to the end-users by Service Providers. Due to their relatively large number it is impractical to provide separate interfaces for each of them. Therefore, CPs will need to implement/utilize the ENTHRONE interfaces. This approach increases the flexibility and the scope of

application. CPs will be able to continue to use their legacy systems (data bases, repositories and even expensive content management systems). They will just need to implement a sort of *plug-in* offering functionality and characteristics specific to their systems on one side, while being compliant to ENTHRONE on the other side. This concept is illustrated in Figure 5.3. Interactions between the actor CP and EIMS include:

- registration of Content Providers. This action can be seen as sending an XML document with all the required data about the CP;
- upload of MPEG-21 Digital Item Declarations (DIDs) and possibly other type of metadata (such as TV-Anytime or MPEG-7);
- reception of requests to search DIs;
- pass list of DI choices (result of search);
- reception of directives related to adaptation;
- reception of commands related to the commencement of a *DI transfer*.

The *Network Provider* actor represents a content delivery infrastructure such as ADSL, cable, wireless, satellite, or terrestrial. One NP is seen as a networking Autonomous System (AS). Based on experience, it seems impossible to enforce the Network Providers /Autonomous Systems to use the ENTHRONE interfaces. Accordingly, it may be necessary to provide a separate adapter for each of them. The actor NP will interact with the EIMS in order to exchange information such as:

- negotiation and subscription of customer-level and provider-level SLSs through EQoS-SLS;

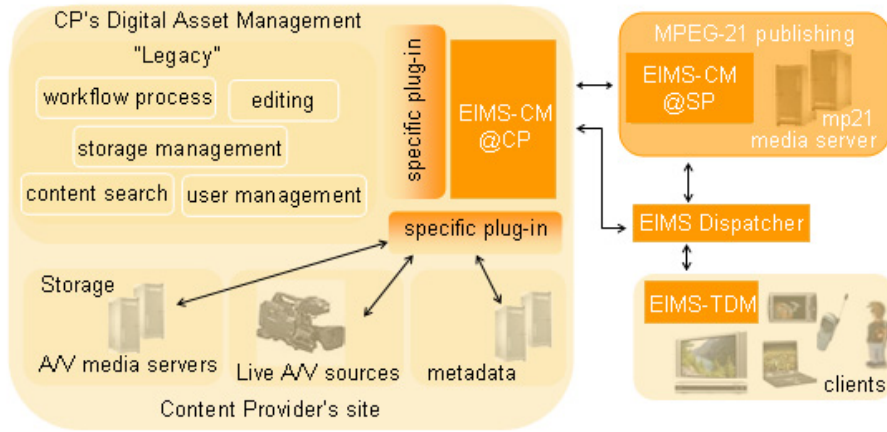


Figure 5.3. Interoperation with heterogeneous Content Providers

- SLS-related data for admission control logic;
- requests for resource provisioning from peering domains through ENTHRONE Resource Allocation (EQoS-RA);
- inter-domain topology-related information;
- monitoring information through ENTHRONE Quality of Service Resource Monitoring (EQoS-RM) suite of protocols.

The EQoS set of protocols has been designed and developed within the ENTHRONE project. The reader is encouraged to consult [31] to obtain more details on this suite of protocols that enable NPs to negotiate among themselves in a cascaded form to establish end-to-end QoS-enabled paths.

The *License Authority* represents an entity responsible for selling and issuing user's electronic permissions which allow users to consume Digital Items. Inter-

facing with a Certificate Authority could be heavily regulated and thus ENTHRONE will need to provide a separate adapter for each type. Interactions between the EIMS and the actor License Authority include the exchange of access rights related data such as:

- request to check if user has a license available/valid;
- license status with the indication of the user access rights;
- request to check rights for performing adaptation operations over the content;
- adaptation permissions related data;
- information related to unique identifiers;
- information regarding registered/available IPMP Tools providers.

Use Case Model

The above identified actors interact with the EIMS and consume the value delivered by it. The functionality required to deliver the value to implement the most relevant interactions are seen as the most relevant use cases, constituting the EIMS Use Case model. The model illustrates who or what will use EIMS, how will the system be used and the value EIMS is capable to deliver, i. e. what the system is able to do for its users. Figure 5.4 provides an overview of the EIMS Use Case model. It contains only the use cases that were thought to be more relevant from the architectural point of view, thus providing the foundation model of the EIMS. Building upon this model, further functionality has been gradually added. Use cases provide a formal way of describing and characterising the functionality delivered by the system. Moreover, they constitute the adequate vehicle to transform the requirements into a form suitable for object-based implementation. Table 5.3 provides a

brief description of each of these uses cases. Actors identified in Figure 5.2 are not the complete set of actors that interact with EIMS. An actor is an entity external to the system that typically stimulates the system with input events, or receives something from it. For a given use case, there is one initiator actor who generates the starting stimulus, and possibly several other participating actors. In Figure 5.4, only the initiator actors have been represented. These should not be directly compared against the actors that appear in Figure 5.1 because the two diagrams represent them at different abstraction levels. The EIMS Use Case model diagram show them at a lower level, thus already identifying the actual system from a given actor that will effectively interact with the EIMS and use the value delivered by EIMS. This was already reflected in figure 5.2. For example, a *PQoS Probe* is a system owned by Content Providers, placed at the A/V transmitting equipment or it may reside at the end-user terminal device, capable of performing QoS measurements and delivering notifications to EIMS. Because higher level actors may have several different equipment interacting with the EIMS (the lower-level actors), when describing the use cases for the sake of simplicity, we will always use the higher level actors. Accordingly, instead of using TVM or aTVM it will be used CP or instead of Policy Enforcement Points it will be used NP. The only exception will be for the equipment designated as *QoS Probes*. The reason for this is twofold: 1) this kind of equipment has several variants that can be owned by CPs, NPs or even to the end user; and 2) the QoS Probes are the initiating actor of one of the most important and innovative use cases towards the provision and maintenance of a QoS-aware service.

Table 5.3. Brief Description of the EIMS major use cases

Use case name	Value delivered by the system	Initiating actor
Retrieve DI	This use case is a complex one, incorporating other more simple use cases. It comprehends search of DIs, browsing of chosen DIs and finally selection of a DI and starting the delivery of that DI with QoS requirements respected.	SP-FE
Recover From QoS Fault	Monitoring and maintaining the requested QoS level during a DI delivery	QoS Probe
Upload DID	Transfer of DIDs and announcement of DIs	Content Provider
Manage DID	Management of DID repository (view, update, delete DIDs)	CP
EIMS Administration	General system management operations of the EIMS	Administrator
Manage SP	View, Update, Add or Delete SPs to the system	Administrator
Manage CP	View, Update, Add or Delete CPs to the system	Administrator
Manage NPs	View, Update, Add or Delete NPs to the system	Administrator

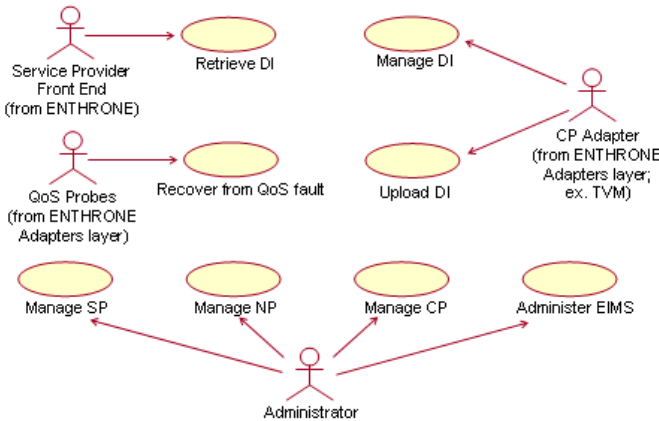


Figure 5.4. Use Case Model of the EIMS

Use case event flows

The successful implementation of each use case requires the interaction between modules, components and actors of the system and the exchange of data between them. This sub-section describes those interactions, the event flows, for each of the use cases present in the EIMS Use Case Model. For each one of them, two tables are presented: the first one provides a general description of the use case, identifying the initiating actor; the second one presents the typical course of events or actions that need to be executed for the implementation of the use case. Additionally, information is provided on possible variations from that typical sequence of actions.

Retrieve DI Use Case Event Flow This use case describes the retrieval of a Digital Item by a Service Provider Front-End (SP-FE) on behalf of an end-user. It is assumed that the end-user has authenticated himself to the SP-FE. It should be noted that in this use case, although technically the actor is the *Service Provider Front-End* because it is the external system that directly interacts with the EIMS,

the ultimate user of the value delivered by EIMS is the end-user or consumer.

Alternative Courses

Action 7: the list may eventually be empty. In this situation the use case either resumes to action 5 or terminates.

Action 8: the SP-FE may request that the DI be retrieved at a later time, and the system has to schedule the invocation of the cSLS at the requested time.

Action 10: if an adaptation is required, EIMS must contact an adequate adaptation engine and pass it the necessary adaptation information. It is expected that adaptation will mainly be performed at the content source (CP site). However, the EIMS is prepared to for the possibility of including DIA capabilities in the delivery chain.

Action 10: if this use case is being performed within the context of a *Media file downloading* service, EIMS must certify that the terminal has got enough storage capacity.

Action 11: if the EIMS is not able to setup a cSLS for delivery of the DI during the time interval requested and sends a notification to the TD.

Action 12: if this use case is being performed within the scope of a VoD service, the EIMS prompts the SP-FE to initiate streaming.

As it can be seen in the listing of the alternative courses, there are some variants for this use case depending on the type of envisaged service (as described in the previous subsection). For each, different assumptions/considerations are made:

Video on Demand - for this service, this use case describes retrieval of a DI in a video on demand service, provided by a Service Provider (SP). It is assumed

Table 5.4. General description of the *Retrieve DI* use case

Use Case:	Retrieve DI
Actors:	Service Provider Front End (SP-FE)
Purpose:	To enable the SP-FE on behalf of the end-user, to select and retrieve a DI to be consumed, according to the preferences of the user and the best possible quality under the present environment conditions.
Overview:	The SP-FE after having authenticated an end-user sends a request to the EIMS to initiate a session. The EIMS receives information regarding usage environment characteristics for that user, retrieved from a DB maintained by the SP or otherwise by directly requesting these to the user. The SP-FE sends a request to the EIMS on behalf of the end-user to search DIs satisfying some criteria or to browse a 'catalogue' of DIs. The EIMS performs a search into available DID repositories in order to locate content that meets the user request and sends back to the SP-FE the list of results. The search DI operation may be performed locally in the EIMS repository and database, in collaborating Content Providers databases and finally in remote Service Providers. The SP-FE indicates to the EIMS the end-user choice specifying the desired QoS and time of consumption. The EIMS checks that the user has a valid license to consume the chosen DI and, by analysing context and content descriptions contained in DIA descriptors, it decides if there is the need to perform adaptation. The EIMS negotiates a cSLS to be used to deliver the DI with the Network Operator to which the terminal is connected. It then invokes the negotiated cSLS and terminates the session.
Type:	Primary

Table 5.5. Main section of the *Retrieve DI* use case event flow

Typical Course of Events			
Actor Action (SP-FE)		System Response	
1	This use case begins when the SP-FE sends a session request to the EIMS.	2	Accepts the session request.
3	Sends to the EIMS the user credentials and usage environment descriptions (user preferences and terminal capabilities).	4	Receives and accordingly processes this information.
5	Sends to EIMS requests to search DIs satisfying some criteria or to browse a catalog of DIs.	6	Processes the requests and executes a <i>search DI</i> operation. This operation may be performed in three steps: • local search (into local EIMS repository and database); • Remote search in collaborating Content Providers; • Remote search in other Service Providers.
		7	Upon completion of the operations to search the DIs, the EIMS sends the list of results to the Terminal Device.
8	Sends to the EIMS the indication of the DI selected by the end-user, specifying the desired QoS and the time to initiate retrieval.	9	Checks that the end-user has a valid license to consume the chosen DI.
		10	Analyses DIA descriptors to decide if adaptation is required.
		11	Negotiates a cSLS with the Network Operator to which the terminal device is connected, for the retrieval of the DI.
		12	Invokes the negotiated cSLS, logs the service and terminates the session.

that the VoD service is subscription based, although it could also be PPV-based. In this case it would be necessary to address the acquisition of a new license. It is assumed that the SP manages its own bank of DIs but has agreements with other SPs to augment the offer to its subscribers (consumers).

TV broadcasting - in the context of this service, the use case describes viewing of a broadcasted TV programme. It is assumed that the terminal is in the coverage area of a TV Broadcasting service. There is no return channel. Interaction with the EIMS is rather limited when compared to other services, because many actions are performed locally by the terminal. Many operations of the described event flow do not apply.

Interactive Television (MHP) - the use case describes the retrieval of an MHP application. It is assumed that the user is watching some programme associated with MHP applications. Although two types of iTV services can be envisaged, differing on whether or not there is a return channel, we consider only the case with a return channel because it is the most interesting for the purpose of defining the EIMS architecture, allowing the interaction between the terminal and the EIMS during the service lifetime.

Media File Downloading - for this service, the use case describes downloading of a media file. It is assumed that the terminal has some storage device. It is also assumed that the service is offered *on-demand*, rather than on a push-basis. The latter is similar to a delayed downloading that is scheduled by the Service Provider. If the last leg is a DVB network, even if the SP-FE did not request delayed downloading, it may be preferable to first download the content and cache it at the SP, and only after that to schedule it for downloading. The alternative would be similar to VoD, in that a fixed bit-rate would have to be

requested along the complete path.

Recover from QoS Fault Use Case Event Flow This use case addresses the actions performed by the EIMS when degradation in the QoS is detected during the course of a DI delivery. The use case starts when a QoS Probe sends a QoS degradation alert to the EIMS. Depending on the actual service under which the DI delivery is occurring and on the cause of the degradation, different actions will be undertaken by the EIMS. These are explained below. It should be noted that a *push model* architecture was conceived for the EIMS rather than a *pull or poll model*. This means that it is not the EIMS that periodically needs to contact and interrogate external modules or adapters. Instead, whenever a change to the current *steady state* conditions the relevant module or adapter notifies the EIMS of the new conditions. Accordingly the EIMS is not overloaded with possibly unnecessary tasks. It should be considered that given the intentions of ENTHRONE to set up QoS-enabled paths by making use of existing networking technologies such as IntServ or DiffServ, the likelihood of quality degradation occurrences is expected to be rather low. Likewise, whereas it is possible and envisaged in some application scenarios, that the characteristics of the context of usage may vary throughout the duration of the service, the rate of occurrence of such variations is not expected to be as high as justifying the extra costs in terms of communication and processing load that the EIMS would incur if a poll-based approach would be implemented. Changes in the context of usage characteristics are more plausible to occur in mobile applications where, for example, lightning conditions or ambient noise of the natural environment surrounding the user may vary along the time.

This main course of events will have small variations according to the type of service being paid.

Table 5.6. General description of the *Recover from QoS fault* use case

Use Case:	Recoverfrom QoS Fault
Actors:	PQoS Probe
Purpose:	To enable the EIMS to take corrective measures during the delivery of a DI when degradation in the QoS is detected
Overview:	The use case starts when a QoS Probe sends a QoS degradation alert to EIMS. The EIMS identifies the QoS degradation cause and location (which could be the terminal, the network or the content provider) and notifies the terminal that the degradation has been registered and that corrective measures are being prepared. The EIMS applies QoS recovery rules according to the type of QoS degradation problem and type of ENTHRONE Service. In doing so, the EIMS may have to contact the terminal, the network or the content provider.
Type:	Primary

Table 5.7. Main section of the *Recover from QoS fault* use case event flow

Typical Course of Events			
Actor Action (SP-FE)		System Response	
1	The use case starts when a QoS Probe sends a QoS degradation alert to EIMS	2	The EIMS identifies the QoS degradation cause, which could be a problem with: 1. the Terminal Device itself; 2. the Terminal Device location; 3. one of the routes in the Network; 4. the Content Provider.
		3	The EIMS reflects the QoS degradation incident and its cause in the QoS Degradation Incident log
		4	The EIMS sends a message to the Terminal Device(s). This message should briefly acknowledge the problem and describe corrective measure(s) to be undertaken.
		5	The EIMS applies QoS recovery rules according to the type of QoS degradation problem and type of Service (some details are provided below).
		6	If QoS degradation is caused by the Terminal Device (itself or its location) or if it could be resolved by performing an adaptation in the terminal, IMS sends a more detailed explanatory message, including in it customer service contact information.
6.1	The Terminal Device analyses the message and activates the appropriate mechanisms.	7	If there is an alternative channel prepared for switch to, the EIMS will cause an automatic switch to this channel.

Table 5.8. Main section of the *Recover from QoS fault* use case (cont.)

		8	If the QoS degradation problem is caused by any of the Network Path legs or Content Provider the EIMS sends an emergency alert to the corresponding operator demanding to fix the problem. The acceptance or not to this alert depends on the contract that this CP has established.
8.1	The operator analyses the information and hopefully, initiates appropriate measures to fix the problem.	9	If the QoS degradation problem is caused by an intermediate Network in the Path from CP to consumer, an emergency alert is sent to the corresponding operator notifying about the occurrence of the problem. The Network operator may respond to this notification and fix the problem or reject it. In the latter case this event is recorded in a log file for future use in determining new pSLSs or request for compensation. This log file will be used at a later phase, in the re-negotiation of pSLSs and new routes.
9.1	The Network negotiates a new licensing agreement (a new cSLS).	10	If the QoS degradation has an objective cause such as weather conditions or Terminal Device location and there is an alternative path, which demands the End User's approval due to some change in the licensing agreement (e.g extra payment, another level of QoS, etc.), the EIMS initiates re-negotiation of the QoS agreement and sends a message to the terminal.

Table 5.9. Main section of the *Recover from QoS fault* use case (cont.)

10.1	The end-user interacts with the EIMS.	11	If the QoS degradation problem could be resolved by automatically adding a small amount of additional network resources at some Network Operator or Content Provider premises, the EIMS issues a corresponding request. Depending on the mutual agreement the NO or CP may accept or reject the request. It should be noted that this process is feasible only if the additional resources do not cause any major renegotiation of the pSLSs, which is a very time consuming process, far beyond the duration of a normal session.
11.1	The addressed operator analyses the message and hopefully initiates adequate measures.	12	If the QoS degradation problem could be resolved by a change in the content adaptation process at the Content Provider or Network Operator premises (in some special cases), the EIMS issues a corresponding request.
12.1	The addressed operator analyses the message and hopefully initiates the process of adaptation.		

Video on Demand - When a QoS degradation problem is detected the EIMS pauses the Terminal Device(s) experiencing this problem, and displays a message that briefly acknowledges the problem and informs about corrective measures being taken. All network resources allocated to this Terminal Device shall be guaranteed to be available whenever the content delivery is resumed. When and if the QoS degradation problem is fixed the EIMS will cause the Terminal Device to resume the content delivery, preferably from the point where the QoS degradation problem was first time detected. If the QoS degradation problem cannot be fixed the EIMS will gracefully suggest the End User to cancel the session (presumably without a charge). The EIMS could consider applying a flexible caching policy when a portion of content is proactively loaded at higher speed to a buffer located at some (presumably the closest) Network Operator premises.

TV Broadcasting - If a QoS degradation problem is discovered at the head-end side (e.g. TV Broadcaster premises) the EIMS will send proactively an informative message to all Terminal Devices that could be affected by this problem. If a QoS degradation problem is discovered at the network path between the play-out center of the broadcaster and the DVB-T transmitter:

- If the network belongs to the broadcaster, the EIMS sends an emergency alert to the broadcaster demanding to fix the problem and proactively sends an informative message to all Terminal Devices, which could be affected by this problem.
- If the network belongs to a Network Operator who was selected among others to provide connectivity between the play-out center and the DVB-T transmitter, the EIMS sends an emergency alert to the corresponding

Network Operator demanding to fix the problem and proceeds with the normal course of events as described in the main section for this use case.

Interactive TV (iTV) Service - Two types of Quality of Service shall be distinguished: a) *Streaming content QoS*, normally assessed in terms of quality of the video or sound components; and b) *Interactive content QoS*, which should be measured in terms of response time, as perceived by the End User. The Recover from QoS Fault event for a) is the same as for Video On Demand or Broadcasting depending on the delivery mechanism. For b) five possible approaches can be considered: 1) Adding additional bandwidth for iTV content delivered via broadcast; 2) Adding additional servers and/or point-to-point bandwidth for iTV content retrieved on-line; and 3) Switching to more aggressive compression (could still improve QoS at the price of making the Terminal Device CPU to work harder). This solution must be carefully evaluated due to the extra delay that it might introduce; 4) Automatic switching from on-line delivery to broadcast for highly-demanded iTV content; and 5) Applying various techniques of gradual rendering to improve End User perception of the system response time.

Media Files Downloading - in the case of a Media File Downloading service, the system's End-to-End Quality of Service is primarily measured in terms of download time. If a QoS degradation problem arises, switching to an alternative Network Path is the basic method to resolve this problem. However, this may imply the re-establishment of new pSLSs (through the invocation of the EQoS-Path protocol), if the existing pSLSs do not offer an appropriate alternative path. This would probably need to be followed by a re-provisioning of the network, which normally takes a considerable amount of time. Consequently this

solution may not be desirable unless an alternative route supported by the existing pSLSs is already set up. Under certain circumstances multiple Network Paths could be utilized simultaneously for delivery of different portions of the media file. These portions will need to be re-assembled at the Terminal Device side.

Upload DID Use Case Event Flow The DID Upload use case is related to media publishing. It allows the Content Provider to make available his content to the general public in the form of MPEG-21 DIDs. The value delivered by this use case allows CPs to transfer to an MPEG-21 database and repository DIDs for each unit of content they wish to make available to end-users through a Service Provider. As indicated earlier in this chapter, we have identified two different entities at the logical level or business role, as the entities responsible for the generation of content and for the distribution of content to the end users - the CP for the former and the SP for the latter. However, these two logic entities or business roles, can be assumed by the same physical entity.

For the purpose of the usage of the EIMS, the important thing is that it is assumed that these two roles exist in fact, and that CPs establish agreements with SPs running an EIMS in order to make their content available to end users. Whether these two roles are implemented by the same physical entity or by two different ones is not relevant.

For the *Upload DI* use case to occur, it is necessary that the EIMS administrator has registered the CP and created a CP user account (through one of the use cases presented in the Use Case Model of the EIMS). Using that account the CP can access the service *Upload DID* to transfer DIDs into a searchable MPEG-21 data base. The following two tables provide a summary of this use case in terms of the

Table 5.10. General description of the *Upload DID* use case

Use Case:	Upload DID
Actors:	Content Provider
Purpose:	To enable the Content Provider to transfer to the EIMS a DID for each unit of content he wishes to make available to end-users.
Overview:	The use case starts when the Content Provider logs in the system. Once properly identified, access is granted and the CP is able to view the list of current DID entries and to initiate the process of transferring new DIDs. He may be able to organize the provided DIDs in different categories, according for example the genre, media format or other attribute/characteristic. In this case, when initiating the upload process, the CP indicates to EIMS to which category the DIDs belong to. This depends however on the data model adopted for the EIMS data base and repository. Upon receiving a new DID, the EIMS parses the file to extract relevant information to be inserted in the local data base and to obtain the location of eventual additional files with metadata on the DI. It stores the DID in the local repository and requests the transfer of other metadata files identified as necessary to be analysed or/and stored in EIMS. The EIMS generates reports concerning the upload of DIDs. The reports are sent to the related CPs and to the EIMS administrator
Type:	Primary

value delivered and the flow of events or actions that need to take place between actors and system modules.

All content the CP wants to make available to the EIMS for search purposes must be in the format of MPEG-21 Digital Items. The CP must therefore generate DIDs for every transactional unit of content. The DID may contain a rich set of metadata such as media location, DRM usage rules, price information, high-level description (eg., title, author, genre, production date, etc), among others.

Some of this metadata is mandatory, as to allow the location of useful content upon

Table 5.11. Main section of the *Upload DI* use case event flow

Typical Course of Events			
Actor Action (CP)		System Response	
1	The use case starts when a CP, after having been properly logged into the system selects the option of <i>upload DIDs</i>	2	An interface is provided to the CP through which he is able to view DID entries in the local database and initiate the process of transfer new DIDs.
3	The CP selects the operation <i>upload DIDs now</i> , providing the EIMS with the required information.	4	A form is provided with fields for the CP to fill in. Presented fields could be the number of DIDs that are to be uploaded, the category to which they belong to or other high-level description, the location, etc.
5	The CP fills in the required information and selects <i>start</i> to initiate the transfer.	6	For each new DID, the EIMS parses the file and extract relevant information to store in its local database. If the DID points to other XML files containing metadata identified as necessary to process or to store locally, the EIMS sends a request to the CP to provide it those files.
7	The CP uploads the requested files.	8	The EIMS parses those files extracting identified data to be inserted into searchable tables and stores locally the DIDs.
		9	The EIMS presents an updated list of DIDs locally stored. If some errors have occur during the parsing of the received DIDs the EIMS sends an alert message to the CP.
10	If the upload of DIDs was successful, the CP logs out the system. Otherwise he may repeat from action 3, possibly after having corrected the identified errors.		

a user search request. Other metadata is optional, as for example, a synopsis of a movie, which could be presented to the user during browsing to help him/her select one among N different matches to his/her query. This has been well defined in the ENTHRONE project through the specification of the ENTHRONE DID data model. This model will be presented further ahead along this section. Although not constituting a development of this thesis, it was conceived through collaborative work, and it is an important piece for the clear understanding of the mode of operation of the complete content mediation platform (the EIMS).

This model is based on the DID and in fact it is compliant with the MPEG-21 DID schema. However, on one side it imposes some limitations or restrictions to the great flexibility of the standardised schema and on the other side it augments its scope by incorporating and ruling the use of additional metadata schemes, notably the TV Anytime scheme.

Information required to enable the EIMS to decide whether to adapt the DI and which adaptation to select, their effects upon the DI, relations to QoS and feasibility, should also be generated when creating the DID. This information may either be inserted directly in the DID or it may reside in another file pointed to in the DID. The EIMS, upon receiving the DID, must parse it to analyze and extract relevant information. This information or part of it is inserted in a relational database and the DID is stored in a local repository. The information to be placed in the tables is the one decided as appropriate to allow meaningful searches and, as indicate in the above paragraph, has been defined and identified in the ENTHRONE DID model. Contextual metadata and content-related metadata with technical descriptions about the content, are not needed for search purposes. Accordingly these data will be stored in a local repository and will a reference to it will be placed in the relational tables.

If the DID indicates that other type of metadata related to the DI is stored in other XML files (DIA information, MPEG-7 metadata, etc), the EIMS has to retrieve those XML documents, using the pointers it finds in the DID (the EIMS must request the transfer of those separated files). The *upload DI* use case has been designed as a manual operation, triggered by the actor Content Provider. However A similar use case, (enabling the transfer, parsing, values' extraction and storage of DIDs), but performing automatically without the intervention of the CP may also be developed. In fact, a similar approach had been implemented previously by the author of this thesis and other researchers [6] within the framework of another European project [29].

Manage DID Use Case Event Flow The *Manage DID* use case delivers value to allow the Content Provider to perform management operations on DIDs already uploaded by him to the EIMS database and repository. In particular it provides the functionality to view/read, update and delete DIDs. All content that the CP wishes to make available to the end-users through the Service Provider is announced as a DID file. It is assumed that a CP has a contract/agreement with the Service Provider running the EIMS. Similar to the *upload DI* use case, it is necessary that the EIMS administrator has registered the CP and created a CP user account. Below, in the next two tables, a general overview of the use case is provided as well as the sequence of operations that should be carried out when executing this use case.

The following three use cases are executed by the actor *EIMS Administrator*. Their purpose and the value delivered by each one of them is similar. However each one operates upon different entities: Service Providers, Network operators and Content Providers.

Table 5.12. General description of the *Manage DID* use case

Use Case:	Manage DID
Actors:	Content Provider
Purpose:	To enable the Content Provider to view/read, update and delete DID entries in the local EIMS database.
Overview:	The use case starts when a properly logged in Content Provider selects the option <i>Manage DID</i> . An interface is provided to the CP, with a menu having the available operations. The CP chooses one of them and continues the management operations.
Type:	Primary

Table 5.13. Main section of the *Manage DI* use case event flow

Typical Course of Events			
Actor Action (CP)		System Response	
1	The use case starts when a CP, after having been properly logged into the system selects the option of <i>manage DIDs</i>	2	An interface is provided to the CP with appropriate menus, through which the CP is able to browse/read, update and delete DID entries in the local database
3	The CP selects the operation he wants to perform and the DIDs upon which the operation is to be performed.	4	Accordingly, the EIMS presents to the CP a new screen to execute the desired operation.
5	The CP inserts necessary information.	6	The EIMS invokes the appropriate mechanisms to respond to the request.
		7	The EIMS presents an updated list of the DIDs locally stored.
8	The CP logs out the system.		

Manage Service Provider Use Case Event Flow This use case describes registration of a Service Provider (SP) in the EIMS and the maintenance of its account by the administrator of the platform. The information maintained by the EIMS on a SP should include fields such as the name, description credentials, list of Content Providers to whom the SP has agreements with, list of Network Operators to whom the SP has agreements, communication parameters, service delivery policy, etc. ⁴ A variant of this use case could be to dynamically discover SPs who are *ENTHRONE-compliant*, i.e., SPs who have an EIMS running at their premises, and prompt them to supply the required information. This way, everytime at the beginning of its operation, every EIMS would poll its surrounding environment to find out and register other EIMSs maintained by other SPs. This solution requires the existence of a registry mechanism such as the UDDI (Universal Description, Discovery, and Integration) specification. The tables below provide an overview of this use case and the sequence of operations that need to be executed to implement it. It should be noted that this use case presents three possible course of actions according to the management operations the actor wants to perform;

1. to register a new Service Provider into the system;
2. to remove an entry from the list of already registered SPs;
3. to modify data from an already registered SP.

Whereas the first table, with a brief overview of the use case, makes reference to all of the three possibilities, the second table only provides the sequence of steps pertaining to the first operation, for the sake of simplicity. Based on this information and that provided in the first table, the reader will easily draw the flow of events required to implement the other two operations.

⁴It should be noted that much of the details of this use case rely upon contractual and trust actions, as well as business model, that lie outside of the technical scope of the EIMS system.

Table 5.14. General description of the *Manage Service Provider* use case

Use Case:	Manage Service Provider
Actors:	EIMS Administrator
Purpose:	To enable the EIMS Administrator to register a new Service Provider into the system, to update information of an already registered SP or to remove an entry of a registered SP.
Overview:	The EIMS Administrator selects the <i>Service Provider Management</i> option through the EIMS console. From the list of available operations, the EIMS Administrator selects <i>Service Provider Registration</i>. A form appears with empty fields corresponding to the information that should be inserted (name, credentials, description, collaborating CPs, collaborating NPs, communication parameters, DI delivery policies, etc.). The EIMS Administrator fills in the requested info and selects the option <i>Submit</i>. From that point on, the filled-in credentials (username and password) will uniquely identify the newly registered Service Provider. The EIMS administrator may also choose to view a list of currently registered SPs and may select one of those. In that case the system will prompt him whether to delete that entry or to modify data on the selected entry. In the later case, the system presents in a new window an edit-able form with data concerning the selected SP. The Administrator performs the necessary modifications and exits by clicking either the <i>Submit changes and exit</i> button or the <i>Exit without saving</i> button.
Type:	Primary

Table 5.15. Main section of the *Manage SP* use case event flow

Typical Course of Events			
Actor Action (Administrator)		System Response	
1	This use case occurs when the EIMS Administrator needs to perform management operations concerning Service Providers. It starts when the Admin selects the <i>Service Provider Management</i> option in the EIMS console.	2	An interface is provided to the Admin with appropriate menus, listing the options available to the Admin for the management of SPs (view/list currently registered SPs, add new SP, update SP and delete SP entries in the local database).
3	The Admin selects the option <i>Service Provider Registration</i> .	4	The system provides an interface with the necessary fields (name, credentials, description, collaborating CPs, collaborating NPs, communication parameters, DI delivery policies, etc.).
5	The Admin fills in the required information and selects <i>Finalise</i> .	6	The system prompts the Admin to check the information.
7	The Admin checks the information and selects <i>Submit</i> or <i>Cancel</i> .	8	The system displays one of the messages: 'SP <i>name-of-new-SP</i> added to the list of currently registered SPs!' or 'List of currently registered SPs unchanged!'.

As referred above there are alternative sequences of steps to this main course of events:

- At any time before selecting *Submit*, the EIMS Administrator can select *Cancel*;
- Action 3 - The EIMS Administrator may select *Service Provider update* and the system provides him the list of currently registered SPs from which he may select one. The system may also present a keyword-based search interface. The Admin then proceeds to step 4;
- Action 3 - The EIMS Administrator may select *Remove Service Provider*, in which case the system presents him the list of currently registered SPs from where the Admin selects one. The system may also provide a keyword-based search interface. The systems prompts the Admin to confirm the operation, proceeding to step 7.

Manage Network Operator Use Case Event Flow The *Manage Network Operator* use case is similar to the previously described use case, but operating upon Network Operators (NOs). Accordingly, it offers functionality to register (add), update (modify the fields of an existing NO) and remove a NO in the EIMS data base. The information maintained by the EIMS on a NO should include fields such as the name, description, credentials, IDs of networks the NO operates, type of network, relevant host IP addresses and port numbers, adaptation capabilities per network, QoS measurement capabilities, etc. It should be noted that, as with the previous use case, many of the details of this use case rely upon legal and financial issues that lie outside of the technical scope of the EIMS system, including SLAs (Service Level Agreements). Although ENTHRONE has been concerned with these aspects, they have not been addressed within the work relevant to this thesis. Accordingly

they will not be discussed or considered here. For the purpose of the design and development of the EIMS, it is assumed that a NO may operate several networks, each of which may have different characteristics. A variant of this use case could be to dynamically discover NOs. This solution would require the use of a registry mechanism such as the UDDI (Universal Description, Discovery, and Integration) specification.

The above table presents the sequence of actions that are endorsed by the Administrator and the system for the execution of the operation *Network Operator Registration*. Below we present the possible alternatives to this main course of events, corresponding to the selection of one of the other two possible management operations:

- At any time before selecting *Submit*, the EIMS Administrator can select *Cancel*;
- Action 3 - The EIMS Administrator may select *Network Operator update* and the system provides him the list of currently registered NOs from which he may select one. The system may also present a keyword-based search interface. The Admin then proceeds to step 4;
- Action 3 - The EIMS Administrator may select *Remove Network Operator*, in which case the system presents him the list of currently registered NOs from where the Admin selects one. The system may also provide a keyword-based search interface. The systems prompts the Admin to confirm the operation, proceeding to step 7.

Manage Content Provider Use Case Event Flow This use case provides the functionality for registering a Content Provider (CP) in the EIMS and to perform management operations upon the CPs registered to the system. The information

Table 5.16. General description of the *Manage Network Operator* use case

Use Case:	Manage Network Operator
Actors:	EIMS Administrator
Purpose:	To enable the EIMS Administrator to register a new Network Operator into the system, to update information of an already registered NO or to remove an entry of a registered NO.
Overview:	The EIMS Administrator selects the <i>Network Operator Management</i> option through the EIMS console. From the list of available operations, the EIMS Administrator selects <i>Network Operator Registration</i>. A form appears with empty fields corresponding to the information that should be inserted (name, credentials, description, IDs and type of managed networks, hosts IP addresses and port numbers, adaptation capabilities, QoS measurement capabilities, etc.). The EIMS Administrator fills in the requested info and selects the option <i>Submit</i>. From that point on, the filled-in credentials (username and password) will uniquely identify the newly registered Network Operator. The EIMS administrator may also choose to view a list of currently registered NOs and may select one of those. In that case the system will prompt him whether to delete that entry or to modify data on the selected entry. In the later case, the system presents in a new window an edit-able form with data concerning the selected NO. The Administrator performs the necessary modifications and exits by clicking either the <i>Submit changes and exit</i> button or the <i>Exit without saving</i> button.
Type:	Primary

Table 5.17. Main section of the *Manage NO* use case event flow

Typical Course of Events			
Actor Action (Administrator)		System Response	
1	This use case occurs when the EIMS Administrator needs to perform management operations concerning Network Operators. It starts when the Admin selects the <i>Network Operator Management</i> option in the EIMS console.	2	An interface is provided to the Admin with appropriate menus, listing the options available for the management of NOs (view/list currently registered NOs, add new NO, update NO and delete NO entries in the local database).
3	The Admin selects the option <i>Network Operator Registration</i> .	4	The system provides an interface with the necessary fields (name, credentials, description, IDs of networks, type of networks, relevant IP address and port numbers, adaptation capabilities, QoS measurement capabilities, etc.).
5	The Admin fills in the required information and selects <i>Finalise</i> .	6	The system prompts the EIMS Administrator to check the information.
7	The Admin checks the information and selects <i>Submit</i> or <i>Cancel</i> .	8	The system displays one of the messages: 'NO <i>name-of-new-NO</i> added to the list of currently registered NOs!' or 'List of currently registered NOs unchanged!'.

maintained by the EIMS on a CP should include fields such as the name, description, credentials (e.g. user name and password), permissions, communication parameters, content adaptation policy and content protection policy. Credentials need to be authenticated to authorise the CP to initiate manual operations such as uploading of DIDs into the EIMS data base and repository. Information related to permissions and communication parameters could be used for an automatic DID Upload or for the execution of the use case Recover from QoS Fault. In addition to registering new Content Providers and similar to the two previous use cases, this use case also allows the EIMS Administrator to update parameters of a registered CP or to delete a previously registered CP. The overview of this use case and the course of events for its implementation are presented in the two following tables.

As happened in the two previously described use cases, variations of this flow of events may occur if the Administrator opts to perform an operation other than the *Content Provider Registration* indicated in step 3.

- At any time before selecting *Submit*, the EIMS Administrator can select *Cancel*;
- Action 3 - The EIMS Administrator may select *Content Provider update* and the system provides him the list of currently registered CPs from which he may select one. The system may also present a keyword-based search interface. The Admin then proceeds to step 4;
- Action 3 - The EIMS Administrator may select *Remove Content Provider*, in which case the system presents him the list of currently registered CPs from where the Admin selects one. The system may also provide a keyword-based search interface. The systems prompts the Admin to confirm the operation, proceeding to step 7.

Table 5.18. General description of the *Manage Content Provider* use case

Use Case:	Manage Content Provider
Actors:	EIMS Administrator
Purpose:	To enable the EIMS Administrator to register a new Content Provider (CP) into the system, to update information of an already registered CP or to remove an entry of a registered CP.
Overview:	The EIMS Administrator selects the <i>Content Provider Management</i> option through the EIMS console. From the list of available operations, the EIMS Administrator selects <i>Content Provider Registration</i>. A form appears with empty fields corresponding to the information that should be inserted (name, credentials, description, Ipermissions, communication parameters, content adaptation policy and content protection policy, etc.). The EIMS Administrator fills in the requested info and selects the option <i>Submit</i>. From that point on, the filled-in credentials (username and password) will uniquely identify the newly registered Content Provider. The EIMS administrator may also choose to view a list of currently registered CPs and may select one of those. In that case the system will prompt him whether to delete that entry or to modify data on the selected entry. In the later case, the system presents in a new window an edit-able form with data concerning the selected CP. The Administrator performs the necessary modifications and exits by clicking either the <i>Submit changes and exit</i> button or the <i>Exit without saving</i> button.
Type:	Primary

Table 5.19. Main section of the *Manage CP* use case event flow

Typical Course of Events			
Actor Action (Adminstrator)		System Response	
1	This use case occurs when the EIMS Administrator needs to perform management operations concerning Content Providers. It starts when the Admin selects the <i>Content Provider Management</i> option in the EIMS console.	2	An interface is provided to the Admin with appropriate menus, listing the options available for the management of CPs (view/list currently registered CPs, add new CP, update CP and delete CP entries in the local database).
3	The Admin selects the option <i>Content Provider Registration</i> .	4	The system provides an interface with the necessary fields (name, credentials, description, permissions, communication parameters, DI adaptation policy, content protection policy, etc.).
5	The Admin fills in the required information and selects <i>Finalise</i> .	6	The system prompts the EIMS Administrator to check the information.
7	The Admin checks the information and selects <i>Submit</i> or <i>Cancel</i> .	8	The system displays one of the messages: 'CP <i>name-of-new-CP</i> added to the list of currently registered CPs!' or 'List of currently registered CPs unchanged!'.

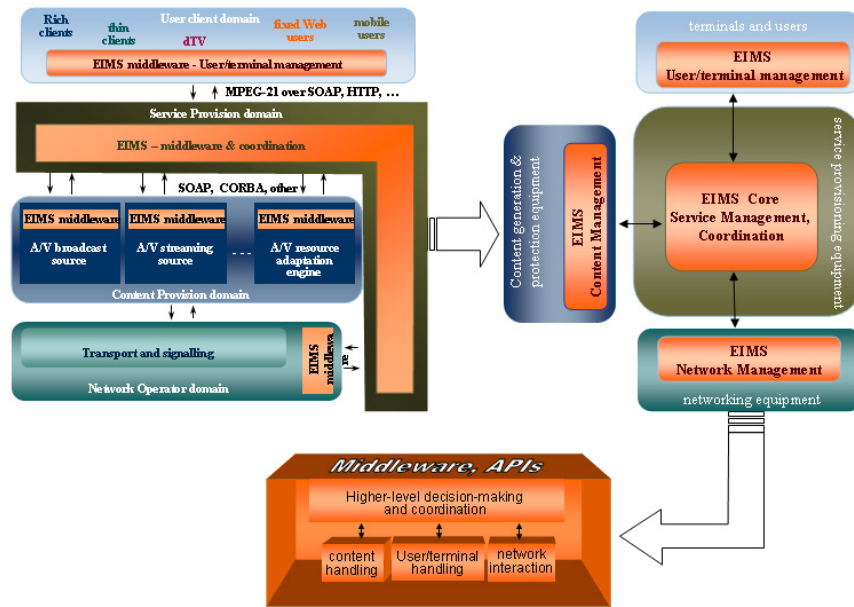


Figure 5.5. The evolution of the EIMS architecture from its original concept

EIMS design

The process of designing the high level architecture of the EIMS took place side-by-side with the identification and description of the Use Case Model of the system. The identified requirements considering different usage scenarios, drove the identification of needed functionality and of the actors using that functionality. This made possible to evolve from the conceptual architecture presented in Figure 1.3 to the high-level modular architecture presented in 5.2. This evolution is illustrated in Figure 5.5.

The rationale behind this architecture was the one of implementing the means to interact, as independently as possible, with the major players in the multimedia content value chain: Content Providers, Service Providers, Network Opera-

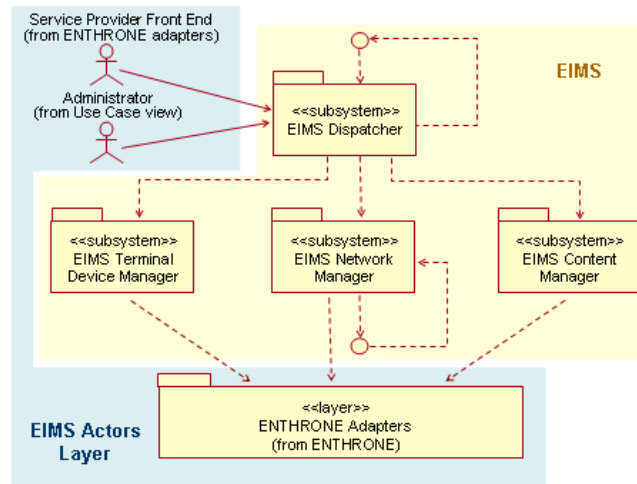


Figure 5.6. The UML high-level block diagram of the EIMS architecture

tors and End-users. Figure 5.6 presents the EIMS high-level architecture adopting a UML-like approach, highlighting the EIMS modules against its actors.

The platform comprises 4 subsystems. The connections between boxes represent interface and the orientation of the arrow indicates the dependencies, i.e. which subsystem imposes the interface. In formal terminology, a sub-system is an isolated, self-contained, functional entity that supplies a well-defined set of services and communicates with other subsystems through well-defined interfaces. It is thus usually, a complex module, composed of smaller operating units (modules or components). Layers can be used inside subsystems to represent a certain level of abstraction. Within the EIMS, different subsystems may have different number of layers. There is not the concept of end-to-end layers, i. e., it is not necessary to have the same number of layers within different sub-systems. This structure could be applied recursively, namely each layer could be represented in a form of collaborating subsystems,

which in turn could have layers, etc. Below it is presented an summary description of each EIMS subsystem in terms of delivered functionality.

EIMS Dispatcher The EIMS Dispatcher subsystem coordinates all the actions taken by the EIMS. In addition, its main specific responsibilities include the implementation of functionality to:

- Request and select the establishment of QoS-enabled paths;
- Perform service level monitoring;
- Decide upon the need to perform an adaptation, both initially when the user requests the retrieval of a DI, and during the actual delivery of a DI, in case the QoS falls below the acceptable threshold.

This subsystem provides functionality to receive and dispatch to other subsystems, users' requests for content and functionality related with the provision of a QoS and context-aware service to the end-user. So, in a first step, the EIMS Dispatcher offers *search* functionality. On a second step, once the user has decided the specific content he/she wants to consume, it offers *service-level* functionality.

Typically the first step involves receiving a user query and dispatching requests for content searches, locally or remotely, and sending back to the user a list of matching results. To accomplish this task the Dispatcher interacts with the EIMS-CM and possibly with other Dispatchers located at other SPs' premises.

In the second step, it involves the following actions: 1) negotiation of the level of quality the user is interested in; 2) coordination of the adaptation decision taking process, deciding the need for adaptation and, if necessary, invoking the adaptation operation; and 3) reservation the appropriate resources and monitoring the level of

quality throughout service lifetime.

The *service-level* step, thus includes functionality related to the *negotiation* of the service level required for the retrieval of DIs, as well as that related to service level *monitoring* during the DI delivery.

Negotiation of the service level typically involves performing resource reservation between two edges of the network, upon a request of an SP, in order to establish end-to-end QoS-enabled paths at the aggregate level between CP and consumers. This is essentially a request sent from the EIMS Dispatcher to the NO, through EQoS-pSLS⁵. It enables the establishment, if it still does not exist, of an aggregated communication channel between the Content Provider and a community of users served by a common access network⁶. Then, at the time of service invocation, it also involves the request for allocating the committed resources at the pSLS level (this is, inside the aggregated channel) through EQoS-RA (see footnote [5]), thus establishing a specific cSLS (customer SLS) that satisfies the requirements for providing to the user, the service he/she requested.

Service level *monitoring* aims at keeping track of the compliance of the level of service provided to the customers during the service lifetime, to the level initially agreed. The EIMS Dispatcher is responsible for auditing and reporting this, and for initiating reactive measures if necessary. It accomplish this by using information provided by monitoring agents (the already identified PQoS and network probes). It thus provides in-service verification of value-added services, verifying whether QoS

⁵EQoS is a suit of end-to-end signalling protocols developed within ENTHRONE. It allows SPs and NOs to exchange messages in a cascaded model to obtain the establishment of an end-to-end QoS-enabled path involving different entities or domains.

⁶A pSLS (provider Service Level Specification) is a contract that establishes the parameters and conditions of aggregated channels between a Content Provider and an access network serving a number of customers. More details are given on the use of SLSs, when describing the implementation of subsystem *NetSrvMgr*, in the next section

performance guarantees committed in the SLSs are in fact being met.

The accomplishment of the service level monitoring related operations by the Dispatcher encompasses locating the cause for the degradation of the QoS and proposing the necessary corrective actions, or keeping a log for future exploitation or compensation. It requires that the degradation of the QoS is detected and signalled to the Dispatcher by a system external to EIMS, designated as *QoS Probe*. This system may be located at the network, at the Content Provider premises or at the user terminal.

The list below, summarises the described functionality implemented by the EIMS Dispatcher:

1. Distributed content *search*:
 - locally;
 - in remote collaborating SPs.
2. service level support:
 - negotiation:
 - resource reservation request (at the pSLS level);
 - resource commitment request (inside the pSLS, at the level of cSLSs).
 - monitoring
 - locating the cause for quality degradation;
 - initiating appropriate reactive measures.

Finally, the EIMS Dispatcher is also responsible for making available a graphical user interface (GUI) for Administration purposes, and accordingly implement the required management functionality.

The EIMS Dispatcher can be seen as the core module of the EIMS coordinating all the actions of the other sub-systems. It is intended to be installed at the premises of the entity responsible for the provisioning of the service to the end-users. In ENTHRONE we have assumed this role would be played by Service Providers.

EIMS Content Manager The EIMS Content Manager (EIMS-CM) subsystem is responsible for keeping metadata on DIs, to support browsing and searching by consumers. This sub-system supports the Dispatcher in all operations related to the management of and interaction with DIs and DIDs.

The EIMS-CM is intended to cache available DIDs in order to improve the search operations and to allow these to be made in an interoperable way, regardless of the original formats used by the owners of the content in their servers or data bases. With this architecture, all searches for content are done in MPEG-21 DID repositories.

Whilst it will keep a repository of available DIDs, the CM will not provide or maintain servers or databases to store the actual resources pointed to by a DID. It will only maintain (and regularly update) a repository of DIDs. These can be replicas of DIDs of DIs owned by different Content Providers. The adoption of this approach enables the efficient interoperation with heterogeneous CPs, allowing them to keep running their own existing systems. This concept had already been referred earlier in this section and illustrated in Figure 5.3. A similar approach had also been proposed in previous work developed by the author[29], [6].

An EIMS-CM subsystem should be deployed at each ENTHRONE-compliant Content Provider. These may have contractual agreements with multiple Service Providers, and therefore their EIMS-CM may communicate with multiple EIMS Dispatchers. In order to further simplify this interoperation between CPs and SPs, the EIMS-CM

was split into two smaller-scope subsystems:

- the EIMS Content Manager at the Content Provider site (EIMS-CM@CP);
- the EIMS Content Manager at the Service Provider site (EIMS-CM@SP).

This approach enables to maintain the referred backwards-compatibility with legacy CPs' equipment, while introducing the novelty of using exclusively MPEG-21 repositories for content search. This way, the EIMS promotes the interoperability across existing systems and domains, potentiating the extended use of existing content and the provision of a better service to the end-user.

The EIMS-CM@CP provides functionality to assist the generation of MPEG-21 DIs, with the creation of the DID

The EIMS Network manager The IMS Network Manager (NM) subsystem is responsible for service level management at Network Providers (NP), and will be deployed at NPs that participate in an ENTHRONE-compliant network.

The EIMS Network Manager (EIMS-NM) subsystem is responsible for the EIMS tasks related to network resource allocation and quality monitoring at the network level. Essentially, the NM supports the negotiation of QoS-based network services. In addition, it also supports invocation and monitoring of these services, in order to ensure that the negotiated level of service is complied with. The negotiation of these services may be initiated by a Dispatcher, in which case their scope is end-to-end. They may also be negotiated between NMs, in which case their scope is local, i.e. within the boundaries of network domain or Autonomous System (AS). The latter kind of services must be previously established, so that the former can be negotiated: end-to-end services are built on top of a chain of domain-scoped services.

The NM offers services to the EIMS Dispatcher, allowing it to issue requests for:

- information concerning the conditions of the network;
- the reservation and establishment of QoS-enabled paths;
- the execution of some reactive measure to deal with situations of quality degradation on the network.

The EIMS NM is also responsible for other tasks related to service management, such as service planning, service monitoring and QoS-based route discovery and selection. This requires the EIMS Network Manager to have some knowledge of the resources and of control layers in a network. To accomplish its tasks the NM interacts with existing networking equipment, thus relaying on existing resource reservation and service management functionality of networks. It also relies on the use of cSLS/pSLS concept and on the EQoS suite of protocols, both developed in ENTHRONE.

As we saw above, the Dispatcher needs to be informed about the occurrence of situations of quality degradation. These can be detected in the terminal or along the network path by specialised modules (the *PQoS* and *NQoS* probes respectively). In the latter case, it is the NM the entity responsible for collecting appropriate measurements from the probes and pass them to the Dispatcher. In order not to overload the Dispatcher with unnecessary data, the NM works in a *push* mode. Accordingly, in this case, the NM acts as a client to the Dispatcher, using the corresponding service implemented by the Dispatcher only when it obtains a value below the agreed QoS.

The NM is the EIMS subsystem intended to be deployed at the Network Operators (NOs) premises, thus providing the bridge to the networking domain with all the associated "traditional" network equipment. This subsystem collaborates

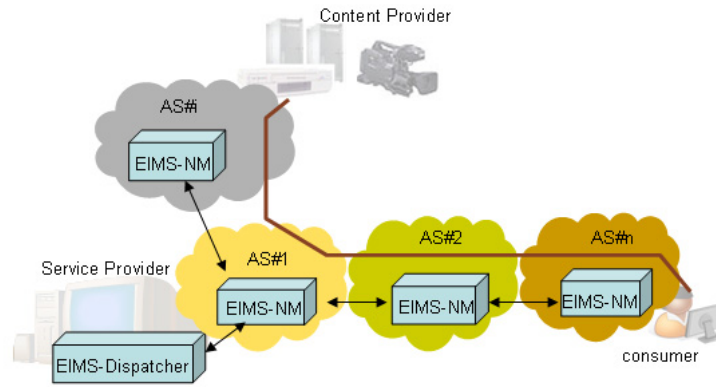


Figure 5.7. The role of the NMs in an end-to-end scope

with the EIMS Dispatcher to find a QoS-enabled delivery path for the DI selected by the end-user. This delivery path must be established on an end-to-end scale, to allow end-users to access and consume content stored in remote CP's repositories. Usually, the end-to-end path is the result of a concatenation of several network paths established across distinct Autonomous Systems (ASs). As illustrated in Figure 5.8, it is therefore required, that each cooperating AS has installed an EIMS-NM and that these communicate among themselves to provide the requested functionality on an end-to-end scale. This communication is achieved through the use of the EQoS suite of protocols, defined in the ENTHRONE project.

The EIMS Terminal Device Manager The Terminal Device Manager (TDM) subsystem encapsulates the decision making process made locally in the terminal device, being responsible for establishing the communication between the terminal device and the content mediation platform.

It provides a platform-neutral interface to various kinds of terminals. The TDM

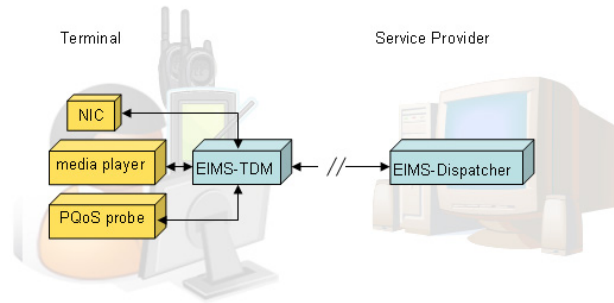


Figure 5.8. The role of the TDM

subsystem encapsulates functionally implemented locally by dedicated hardware of software components, such as Perceived QoS (PQoS) monitoring, performed by dedicated PQoS probes, and license management conducted by DRM-enabled players. It also provides the means to exchange information between the user and the SP domains (i.e., between the terminal and the Dispatcher), supplying the Dispatcher with PQoS alerts, information regarding terminal capabilities, user preferences and usage rights, which are essential for taking decisions regarding the provision and adaptation of content to the user. Figure 5.8 illustrates this role of the TDM.

The TDM's main responsibility is to report relevant terminal characteristics and PQoS information to the Dispatcher, which then takes various adaptation decisions based on the reported data.

In order to retrieve the relevant information, the TDM accesses various adapters, software or hardware components, which are part of the Terminal. This is done using generic Inter Process Communication (IPC) mechanisms, exchanging messages with a well defined syntax, based on XML and MPEG-21 DIDL.

Examples of adapters include a media player that provides descriptions related with terminal characteristics, a network interface card (NIC) to obtain instantaneous throughput, a License Manager (that may be included in a DRM-enabled media player) to provide information concerning DI licenses, or a PQoS probe to extract quality measures from the reconstructed stream. When any of these adapters needs to be configured to the characteristics of the service that has been agreed between the user and the SP, the TDM receives the relevant parameters from the Dispatcher and accordingly passes them to the appropriate adapters.

The following is the list of the most important types of information exchanged between the terminal and the SP, via the EIMS-TDM:

- terminal capabilities;
- user preferences;
- perceived QoS;
- licenses-related data;
- set up parameters for local adapters.

Collaborations per Use Case

UML *Collaboration Diagrams*, together with Sequence Diagrams, provide different views of the way in which objects interact in order to fulfil the requirements of a particular use case. While these two types of diagrams are functionally equivalent, showing the messages exchanged between objects and their temporal sequence, collaboration diagrams also show the actual links between the objects in a graphical way. Collaboration Diagrams are useful for showing how groups of objects work together to perform a piece of functionality. It is through their development, based on

the description of the use case, that the developer gains full insight of the required operations, exchange of messages and associated parameters, which are needed to deliver the intended use case value.

This section thus provides the UML Collaboration Diagrams that describe how the EIMS subsystems collaborate among them to successfully implement the use cases described in the previous subsection.

Simple use cases, that typically involve only the interaction of two entities - one EIMS sub-system and one external actor - are not included in this section. The corresponding Collaboration Diagram would not bring added-value to the tables include in subsection 5.3. The only exception is the *Upload DID* use case, because it is an instrumental use case for the design of the EIMS-CM subsystem.

Collaboration Diagrams for the *Retrieve DI* use case From the description of the *Retrieve DI* use case in Table 5.5, it is possible to identify the following steps:

1. Session initialization;
2. DID browsing/searching;
3. DI selection and adaptation decision;
4. License acquisition;
5. Negotiation of SLS;
6. Invocation of cSLS.

The execution of steps 5 and 6, greatly relies on the participation of networking equipment, external to the EIMS. This was explained above, when describing the functionality of each EIMS sub-system. As such, the execution of those steps,

from the EIMs involvement perspective, resumes to issuing a request to, or invoking a service from, an external module or system. For that reason, those steps will not be considered here. For the other steps, Collaboration Diagrams are provided bellow.

1. Collaboration Diagrams for the *Session Initialization* step

Figure 5.9 presents the collaborations established between the EIMS subsystems **Dispatcher** and **TDM**, for the execution of this step.

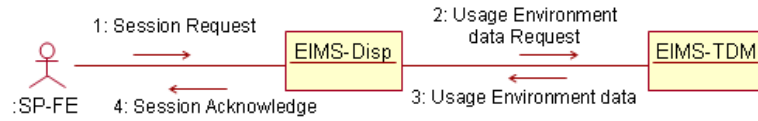


Figure 5.9. Collaboration Diagram for Session Initialization

This step is initiated by the actor **SP-FE** that directly communicates with the Dispatcher (EIMS-DISP) requesting the establishment of a session (**message 1**). The EIMS-DISP checks the credentials sent by the SP-FE and if accepted, sends a request to the TDM to obtain generic usage environment descriptions (**message 2**). Upon receiving these, the EIMS-DISP sets-up a session, storing this information and sending an acknowledgement to the SP-FE (**message 4**).

2. Collaboration Diagrams for the *DID Browsing/Searching* step

Figure 5.10 presents the collaborations established between different EIMS subsystems for the execution of this step.

The **SP-FE** initiates this step by sending the user query (a list of keywords inside the "Search DI" request) to the EIMS-DISP (**message 1**). The EIMS-DISP forwards the request (**message 2**) to the EIMS Content Manager (EIMS-

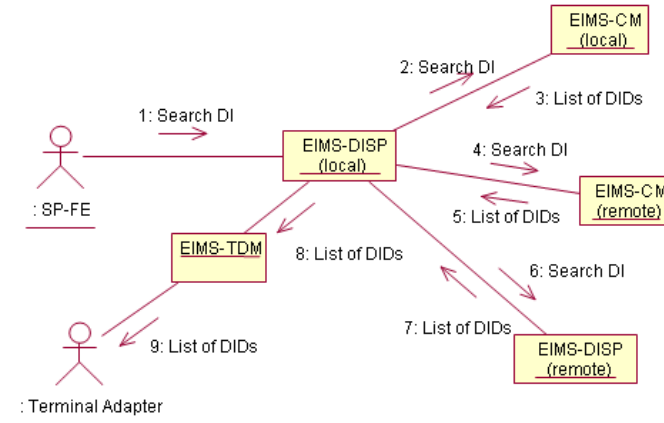


Figure 5.10. Collaboration Diagram for DID Browsing/Searching

CM). The EIMS-CM fulfils the request by searching local metadata, in MPEG-21 repositories. The outcome of this search is packaged as a list of DIDs (**message 3**) that the local EIMS-CM sends back to the EIMS-DISP.

If this list is empty, the EIMS-DISP sends a "Search DI" requests (**message 4**) to the EIMS-CM of Content Providers that had been previously registered in the system. The remote EIMS-CMs fulfil the request by searching their own repositories and return a list of DIDs found (**message 5**).

Again, if this list is empty, the EIMS-DISP forwards "Search DI" requests (**message 6**) to remote EIMS-DISPs, operated by previously registered SPs. Each of these will service the request, essentially as if it had been sent to them by a SP-FE. To prevent endless loops, a mechanism imposing a limit of time for the originally called Dispatcher to resume operation and send back to the user the list of results, or a maximum number of forwarding hops, is implemented. The result of processing the request is again sent as a list of DIDs (**message**

7) to the local EIMS-DISP. Whether the EIMS-DISP performs searches on remote CP or on remote SP, even if its previous searches lead to matching DIDs, is an issue to be handled during implementation. One possibility might be to allow the definition of different policies. Finally, the EIMS-DISP sends a list of DIDs (**message 8**) to the TDM, which forwards it (**message 9**) to the an ENTHRONE Adapter located on the terminal⁷. Note that the EIMS-TDM may make some transformation of the list it receives from the EIMS-DISP before forwarding it, in order to adapt it to the Terminal capabilities.

3. Collaboration Diagrams for *DI selection and adaptation decision step*

Figure 5.11 presents the collaborations established between different EIMS sub-systems for the execution of this step.

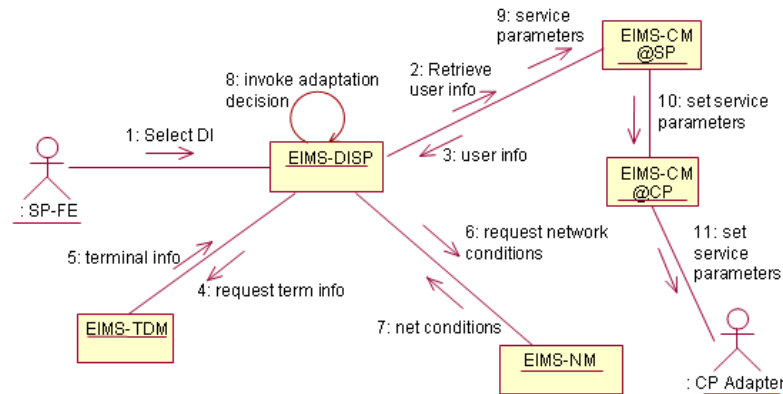


Figure 5.11. Collaboration Diagram for *Select DI*

This step may only be performed after a *DI Searching/Browsing* has occurred.

The end-user after having browsed through a list of DIs and having decided

⁷This particular adapter must be able to process and present MPEG-21 DIDs. As already mentioned, a special component, designated as DI Browser, was developed for this purpose.

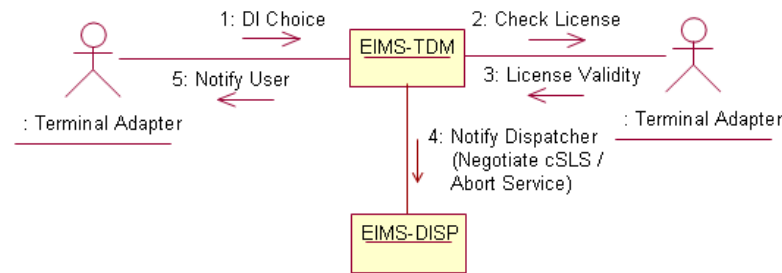
the content he/she would like to consume. When that happens, this step may then be initiated by the **SP-FE**, that sends a request for the selected DI to the Dispatcher (**message1**). The EIMS-DISP initiates the procedures to decide if it is necessary to adapt the selected DI. It first gathers required contextual information: it requests user-related information from the data base (**message 2**) and, if this record does not contain the necessary terminal descriptions, it also requests from the TDM, terminal-related data (**message 4**); it proceeds by requesting information about the network availability (**message 6**). Once in possession of all necessary metadata, the EIMS-DISP invokes inline mechanisms to perform the adaptation decision taking (**message 8**). When this is completed, it passes the obtained service parameters to the EIMS-CM@SP (**message 9**), which in turn passes them to the EIMS-CM@CP (**message 10**). This sub-system directly interacts with adapters at the Content Provider side, thereby having the means to set up the encoding parameters of an adaptation engine (an adapter located at the Content Provider site) (**message 11**).

4. Collaboration Diagrams for the *DI License Acquisition* step

Figure 5.12 presents the collaborations established between the EIMS subsystems *Dispatcher* and *TDM*, as well as terminal Adapters, for the execution of this step.

An adapter located in the terminal, sends to the EIMS-TDM a request (**message 1**) with the DI choice entered by the user. Before initiating the procedure for setting up the network path (i.e. the establishment of a cSLS for delivery of a DI), the IMS-TDM sends a requests to another adapter, typically a DRM-enabled media player or a dedicated License Manager module, (**message 2**) to verify if there is a valid license for the consumption of the requested DI.

Note that this verification may involve the exchange of messages with other EN-

Figure 5.12. Collaboration Diagram for *License Acquisition*

THRONE subsystems, but they are not represented in this diagram, because no EIMS subsystem participates in those exchanges.

Finally, the outcome of this process is sent back to the TDM (**message 3**). If this outcome is positive, the TDM sends a request to the EIMS-DISP to negotiate a cSLS. Otherwise, it notifies the EIMS-DISP to abort the transaction and also notifies the end user. He/she may then initiate the adequate procedure with a License Authority, to acquire a valid license.

Collaboration Diagrams for the *Recover from QoS Fault* use case Table 5.7 provides a description of the sequence of actions that are performed to implement the *Recover from QoS Fault* use case. From the information there contained, it is possible to identify the following steps:

1. Detect QoS Fault;
2. Acknowledge QoS Fault.
3. Recover from QoS Fault Request;
4. Notify the problem to the involved actors;

5. Propose corrective actions (new DIA parameters or small increment in the network resources that do not imply re-establishment of the pSLSs);
6. Log of the event in a file for future exploitation in re-establishment of pSLSs or for compensation, in case no actions are taken;
7. Negotiate and Acquire New License.

Similar to the explanation given to some of the steps involved in the execution of the *Retrieve DI* use case, the execution of steps 5 and 6 greatly relies on the operation of networking equipment, external to the EIMS. For that reason they will not be described here. The same happens with step 7 ut for different reasons: this step is identical to one appearing in *Retrieve DI* use case, which description was already presented there.

1. Collaboration Diagrams for the *Detect QoS fault* step

Figure 5.13 presents the collaborations established between different EIMS subsystems for the execution of this step.

The general idea behind this approach is that a QoS Fault event, a degradation on the quality of the service in relation to the values agreed upon service establishment, is detected by a QoS Meter, or QoS Probe, which can be located at different points of the transmission chain. More precisely:

- at the terminal, by a Perceptual QoS (PQoS) Meter (step 1);
- at any point of the distribution network, by a NQoS Meter (steps 5 and 6);
- at the Content Provider premises, by a PQoS Meter (step 10).

Although the end-to-end transmission path may be composed of different network elements, which in turn may use different QoS Meters, for the purpose

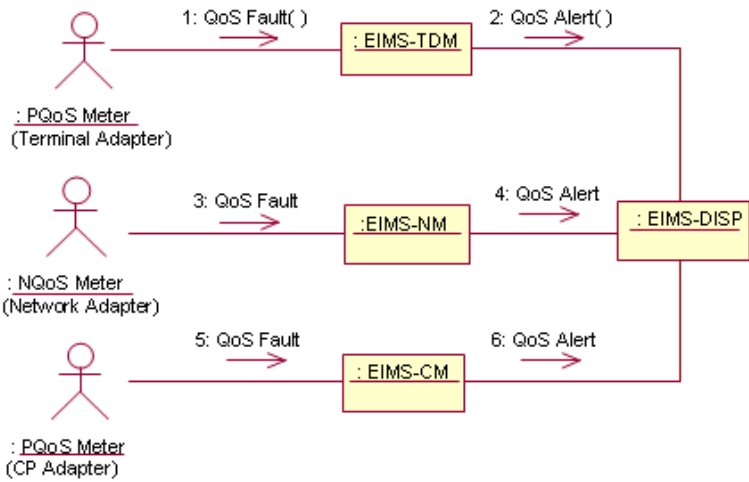


Figure 5.13. Collaboration Diagram for *Detect QoS fault*

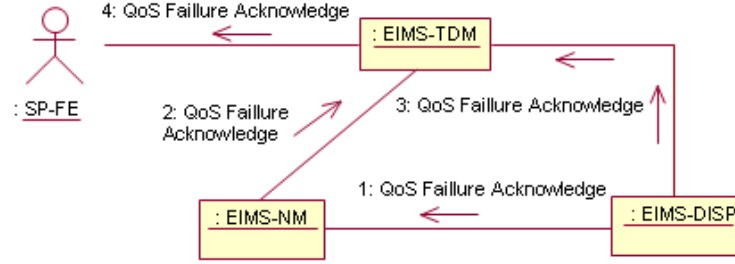
of describing this step, and in general the *Recover from QoS fault* use case, there is no need to make any distinction. The operations to be performed by the EIMS components will be the same, even if the contents of the exchanged messages may be different.

When a QoS fault event is detected a corresponding alert message is sent to the EIMS-DISP for centralized logging and problem analysis purpose.

item **Collaboration Diagrams for the *Acknowledge QoS fault* step**

Figure 5.14 presents the collaborations established between different EIMS sub-systems for the execution of this step.

When a quality drop occurs, a message should be sent to the user, to let him/her know that the problem was detected and that remedial actions are being provided. If the problem was detected by the terminal PQoS Meter, the corresponding acknowledgement message will be initiated locally by the

Figure 5.14. Collaboration Diagram for *Acknowledge QoS fault*

EIMS-TDM (**message 4**), otherwise it will be initiated by the IMS-DISP and sent to the TDM typically (**message 3**). In the latter case if the QoS Fault problem spawns across the whole network segment the message will be sent to all terminals which are affected by this problem, via the network (**messages 1 and 2**).

2. Collaboration Diagrams for the *Recover from QoS fault request* step

Figure 5.15 presents the collaborations established between different EIMS subsystems for the execution of this step.

Basically, all subsystems of the EIMS subsystems may participate in, and collaborate for, the fulfilment of this step. The Dispatcher is the coordinator of this step, by analysing the QoS alerts and deciding on possible reactive measures (**message 1**). As consequence of that decision, the EIMS-Disp may send a request to: a) the EIMS-TDM (**message 2**), which will try to implement locally remedial actions (**message 3**); b) to the network via a EIMS-NM (**message 4**), which will try to implement locally remedial actions (**message 5**) and/or which will propagate the request to other EIMS-NMs (**message 6**)⁸;

⁸Every EIMS-NM receiving a QoS Recovery Request, will essentially behave the same way as the

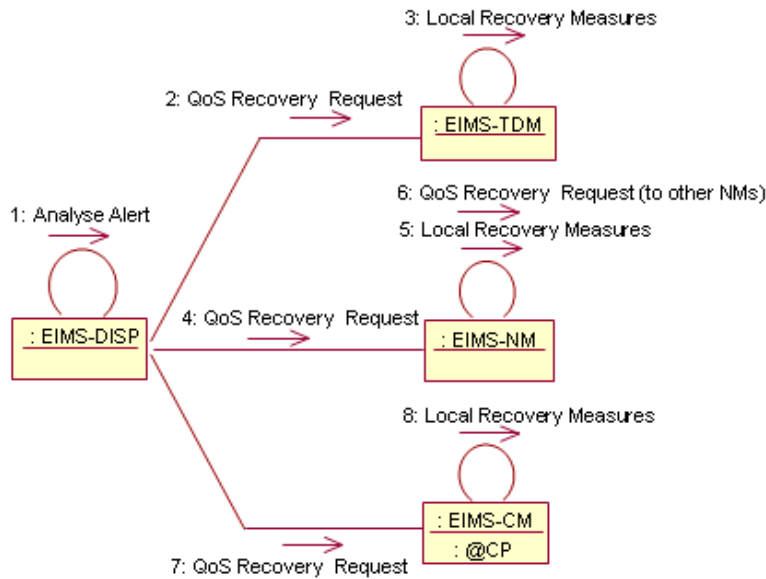


Figure 5.15. Collaboration Diagram for *Recover from QoS fault request*

c) to the EIMS-CM@CP (**message 7**), which will also try to locally implement remedial actions (**message 8**).

Collaboration Diagrams for the *Upload DID* use case Table 5.11 provides a description of the sequence of actions that are performed to implement the *Upload DID* use case. Although consisting in the execution of a considerably large number of steps, this use case involves only two entities the CP and the EIMS-CM. As such, it is possible to draw in a single diagram, all the collaborations established between these two entities, instead of having multiple diagrams, one for each step or event, as we did for the previous use cases.

1. Collaboration Diagrams for the *Upload DID* use case

calling module

Figure 5.16 presents the collaborations established between the EIMS subsystem *Content Manager* and a Content Provider Adapter⁹, for the execution of this step.

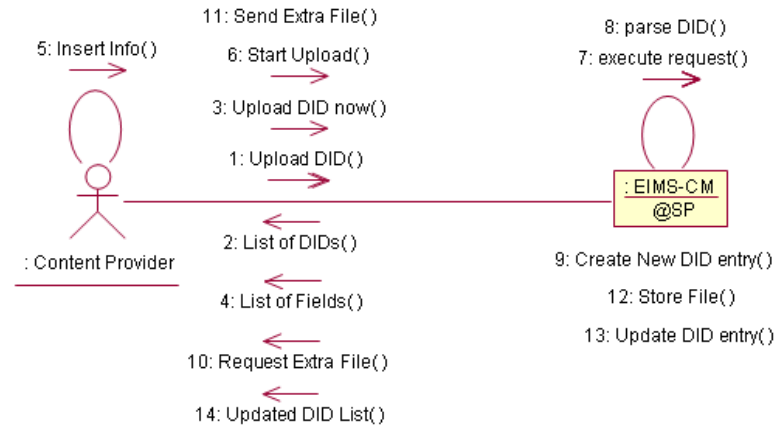


Figure 5.16. Collaboration Diagram for the *Upload DID* use case

Once properly authenticated and authorized, the CP selects from the EIMS-CM console or user interface, the operation "Upload DID" (**message 1**). The EIMS-CM presents the list of presently stored DIDs and the option to upload new DIDs (**message 2**). The CP selects that option (**message 3**), to which the EIMS-CM responds (**message 4**), presenting to the user a form with fields to fill (such as number of DIDs, type or category to which they belong to, location, etc). The CP then fills in that form with the appropriate information (**message 5**) and initiates the upload process of a DID it to the EIMS-CM (**message 6**). Upon reception of this information, the EIMS-CM prepares to receive and process the DID (**message 7**). For each DID it receives, the

⁹Although in the presented diagram, the role of initiating the Upload DID use case is assigned to a CP Adapter, this adapter can be seen as belonging to the EIMS Content Manager located at the CP's premises, the EIMS-CM@CP

EIMS-CM parses the XML file to analyze it and extract relevant information (**message 8**). It creates a new entry in the database (**message 9**) to insert this information. If the DID indicates that other type of metadata related to the DI is stored in other XML files at the CP premises (which could be, for example, a file with DIA information or MPEG-7 metadata), the EIMS-CM has to retrieve (or request the transfer of) those separated files, using the pointers it finds in the DID (**messages 10 and 11**). Each newly received file is processed and stored and the corresponding DID entry is updated (**messages 12 and 13**).

5.4 The implementation of the EIMS

The functional modules described in the previous section, were implemented as sets of services. In turn, each service was implemented as a set of Java objects, and deployed as Web Services. They thus offer service operations that can be externally accessed. In some cases those operations are used only inside the EIMS, between subsystems or even within the same subsystem; in other cases they are used by external actors. The open source projects Eclipse [58], and Apache Axis 1.2 [11], were used as the development and implementation platforms.

The communication between the EIMS subsystems is done using Web Services technologies, notably SOAP [135], and it is assumed that they use HTTP over SSL/TLS [33], [87], as the transport protocol. On the Web, given that the client universe is virtually open, SSL/TLS, which uses certificates, is used to authenticate only the server. Client authentication is usually done using passwords. On the other hand, the interactions between the EIMS Dispatcher and other EIMS subsystems are done in a more or less closed environment: the EIMS subsystems are operated by en-

entities that have previously established some form of contract among them. Therefore, we assume that in real implementations, Web-service clients are also authenticated using certificates. If needed, the different entities operating EIMS subsystems may exchange certificates, obtained from their usual authentication authorities, to set up a secure channel.

This section describes the implementation details of the main EIMS subsystems, which support the functionality identified and described in the previous section. Because the EIMS subsystems were implemented as sets of Web Services, WSDL [159] definitions of those services are also provided as an Annex.

Although the EIMS is composed of four main subsystems - *Dispatcher*, *Content Manager*, *Network Manager* and *Terminal Device Manager* - in this section, we will not present the implementation of the *Network Manager* subsystem. The reason for that is twofold:

- Firstly, because neither the author of this thesis, or colleague researchers from INESC Porto ¹⁰ participated actively in the implementation of this software subsystem;
- secondly, and as well important, because all the functionality required to support the context-aware access to multimedia content, can be provided by the other three subsystems, taking advantage of the network abstraction that is possible to achieve within the EIMS.

As already referred in the previous section, network availability and allocation of resources in the network, is performed using the pSLS/cSLS concept and

¹⁰INESC Porto is the research institute where the proponent of this thesis conducted her research work. INESC Porto was one of the main partners of the ENTHRONE project.

service planning approach of ENTHONE.

A more thorough explanation concerning the cSLS/pSLS and service planning approach, will be provided in this section, when referring to the network-related functionality of the Dispatcher. Nevertheless, a brief explanation can already be given here.

Service Level Specifications (SLS), are XML documents that describe characteristics and conditions of contracts established between an entity, notably a Service Provider, and a Network Operator, for the provision of QoS-enabled network channels. In order to be able to serve a community of users within the same region (i.e., served by the same access network), an SP may negotiate with the NO a large capacity channel for a large period (e.g., some weeks, or months), establishing with the NO a pSLS (provider SLS).

By performing an adequate planning, the SP may establish several pSLS with NOs, in such a way as to contract sufficient network capacity to potentially serve all its customers. Then, whenever the SP receives requests from end-users, instead of having to contact the NO, it uses a contracted pSLS to find out if it can deliver the requested service. The network resources needed to deliver that service, can also be allocated via the pSLS, by creating a customer SLS (cSLS) and deducting the used resources in the pSLS. This approach thus provides the SP an abstraction of the network during service negotiation with end-users.

5.4.1 The EIMS Dispatcher Implementation

The EIMS Dispatcher subsystem coordinates all the actions taken by the EIMS. Every Service Provider that wishes to offer ENTHRONE-compliant services must deploy an EIMS Dispatcher subsystem.

According to the functionality described in the previous section, from the point of

view of a consumer, the EIMS Dispatcher delivers functionality that can be grouped under two main services;

- Service discovery and location;
- Service negotiation and supervising.

The first service allows a consumer to search for content and services available through SPs in the form of MPEG-21 DIs. The latter supports the negotiation of the service level required for the consumption of a DI, as well as the monitoring of the service level provided during the DI's consumption.

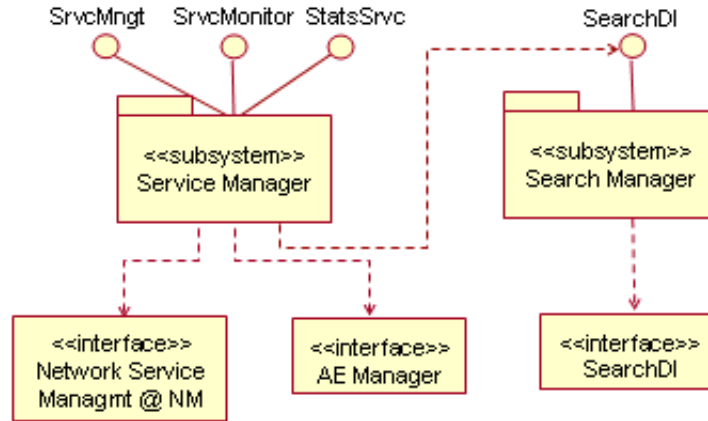
The two above-mentioned services offered by the Dispatcher, are implemented by two subsystems:

- *Search Manager*;
- *Service Manager*.

Figure 5.17 presents the service architecture of the Dispatcher, showing the interface provided and used by those services.

The *Search Manager* subsystem implements the discovery and location service, thus supporting searching of DIs and browsing of DIDs, via a well defined interface. As described in the functional description of the EIMS, these operations should be possible to be performed using a local cache, in remote Content Providers or even other Service Providers with which the initial Service Provider has agreements. In all these cases, the *Search Manager* uses the same interface.

The *Service Manager* subsystem supports all the service management functionality at the Service Provider (SP). As derived from the user requirements and the functional specification, service management operates at different levels. Accordingly, it is implemented through different, specific-scope, subsystems:

Figure 5.17. *Dispatcher* subsystems and interfaces

- *Customer Service Manager, CustSrvMgr*. This subsystem manages the service the SP provides to its customers, the content consumers. It supports not only service negotiation and invocation, but also service monitoring.
- *Content Service Manager, ContSrvMgr*. This subsystem provides the client-side functionality of the Content Provider service level management. The server-side functionality is exposed as a Web Service by each Content Provider, as part of the local *EIMS-CM@CP* (EIMS Content Manager @ Content Provider=).
- *Network service manager, NetSrvMgr*. This subsystem provides the client-side functionality required to perform service level management at the network side. Basically, it is responsible for negotiation and invocation of SLA-based network services.
- *Adaptation Decision Engine, ADE*. This subsystem takes DI adaptation decisions in order to provide the best quality given the usage environment constraints (on the terminal, on the network, on the the adaptation engines ca-

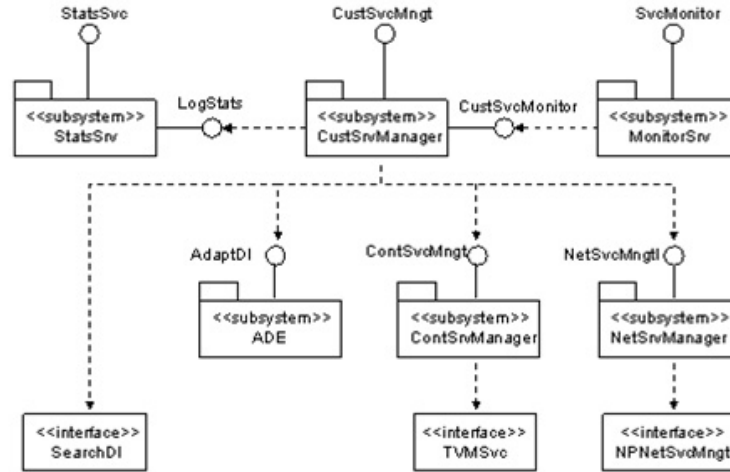
pabilities and user preferences). The ADE can be invoked both during service negotiation and during service delivery, in the case the measured QoS falls below the agreed level. In both cases it determines the set of service parameters that provide the best quality, given the above mentioned constraints.

- *Statistics service, StatsSrv.* This subsystem logs service requests, which can be later used by the Service Planning system for service forecasting. It provides statistics regarding consumption of DIs, thus allowing CPs and SPs to know which are the most popular DIs.
- *Monitoring service, MonitorSrv.* This subsystem acts mainly as a gateway through which the Service Manager receives QoS alerts sent by QoS probes. All the actual processing associated to monitoring is provided by the *Customer Service Manager* subsystem. The use of a specific subsystem to collect the information generated by QoS probes, enables the actual processing functionality to be independent from the protocols used by the probes to provide feedback of the service level.

Service Manager subsystem

The *Service Manager* subsystem is fundamental for the implementation of a context- and quality-aware multimedia service. It is the most complex subsystem of the complete EIMS, which needs to interact with a significant number of modules, sub-systems and entities. Its architecture, in terms of software modules and interfaces, is presented in Figure 5.18.

It should be noted that the service offered by an SP to its customers, is built on top of two other services: 1) a content service, providing the content and managed by a Content Provider; and 2) a network service, providing the transport

Figure 5.18. *Service Manager* subsystems and interfaces

of that content, from the CP to the consumer, and managed by a Network Operator. The SP will only be able to provide its service with QoS guarantees, if each of these two services are, as well, provided with QoS guarantees. Consequently, in order to provide QoS-enabled multimedia services to content consumers, the EIMS Dispatcher must itself support negotiation, invocation and monitoring of services with QoS guarantees from both CPs and NOs.

Whereas the SP is the provider of the multimedia service to consumers, it needs to use services of CPs and NOs to build that service. This means that the SP is, on one hand, a *service* from the end-user perspective, and is a *client* of both the CP and NO services, on top which the multimedia service is built.

This is the basis for the internal architecture conceived for implementing the *Service Manager*: these three dimensions or roles drove the architectural implementation design of the *Service Manager*. The conceived modular design of this subsystem,

comprises the six subsystems listed above, but having the following three, occupying a relevant position:

- Customer Service Manager (*CustSrvMngr*);
- Content Service Manager (*ContSrvMngr*);
- Network Service Manager (*NetSrvMngr*).

The use of these three subsystems enables the separation of service level monitoring according to the domain, and to implement a hierarchical approach. The former, allows detecting problems near their source, hopefully enabling the implementation of reactive measures before they affect the end-user. The latter favours an incremental and de-centralised solution, distributing the load and enabling increasingly elaborated remedial actions.

The *CustSrvMngr* is at the top of the hierarchy. Ultimately, it will have to handle all the problems that other service managers were not able to handle. Being at the top of the hierarchy, it can have a more comprehensive view of the state of the system. It thus may handle problems in one service by taking compensation actions in other services. For example, it could incorporate statistical multiplexing algorithms and dynamically re-assign bandwidth and/or changing encoding parameters of various streams sharing the same pSLS, according to their measured quality levels.

The *ADE* is also a very important subsystem of the *Service Manager*, providing a decisive contribution towards the provision of context- and quality-aware services.

The list of major functionality implemented in each of these subsystems is provided below.

Customer Service Manager subsystem The *CustSrvMgr*, the subsystem at the top level of the *Service Manager*, is illustrated in Figure 5.19.

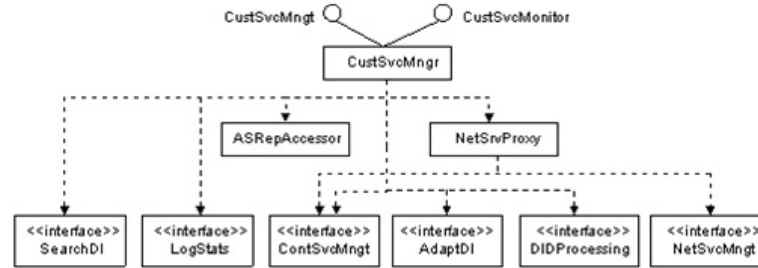


Figure 5.19. *Customer Service Manager* interfaces and main classes

The following service operations or methods are offered by the *CustSrvMgr* through its interfaces:

1. **SelectDI** - this operation is used to submit a service request. It is assumed that the user has already browsed through a list of DIDs upon submitting a query and receiving a list of matching results. In the case the request can be satisfied, the output is a DID with: 1) a service identifier assigned by the *CustSvcMgr*; 2) the service end point, i.e., the address where the DI is located; and 3) information for monitoring purposes. In the case of failure, the output is a DID with the reason why the service request could not be fulfilled, and suggestions to remedy the problem.

- **Synopsis:** this operation is used by a customer to submit a service request.
- **Input:**
 - identifier of the content/service being requested;
 - desired service class;

- Schedule for consumption;
- terminal location, including IP address and access network netmask;
- MPEG21 UED with terminal capabilities and network characteristics (of the terminal's access network).

- **Output:**

- customer service identifier;
- service end point;
- information for service monitoring.

2. **PlayDI** - This operation allows to invoke a service, which was previously negotiated via the *SelectDI* operation. In the case of success, the output may include a DID to be presented to the consumer, the service end point, i.e. the address the consumer should point to its player to consume the service, and information for monitoring purposes. In the case of failure, the output is a DID with the reason why the service request could not be fulfilled, and suggestions to remedy the problem.

- **Synopsis:** this operation is used to invoke a service previously negotiated using the *SelectDI* operation.

- **Input:**

- identifier of the service, generated during its negotiation phase, i.e. during the execution of a previous *SelectDI* operation;

- **Output:**

- service end point;
- information for service monitoring.

3. **NotifyDIConsumption** - this operation is used to notify the *CustSvcMngr* of the consumption of a DI cached at a terminal. This is useful in order to keep statistics on the demand of DIs. This is a one way operation, there is no output.

- **Synopsis:** this operation is used to notify the *CustSvcMngr* of the consumption of a DI cached at a terminal.

- **Input:**

- identifier of the DI being consumed;
- terminal location, including IP address and access network net mask;
- URL from where the DI was fetched.

4. **raisePQoSAlarm** - this operation allows QoS probes to send alerts to the *CustSrvMngr* on a push mode. Whenever a QoS probe measures a value below the agreed QoS for that service, it notifies the *CustSrvMngr*, enabling him to initiate remedial actions.

- **Synopsis:** this operation is used to notify the *CustSrvMngr* that a service it previously negotiated service is not satisfying the agreed QoS.

- **Input:**

- identifier of the service experiencing QoS degradation;
- the measured QoS value that triggered the alarm;
- UED with terminal capabilities and network characteristics (of the terminal's access network).

- **Output:**

- service end point;
- information for service monitoring.

5. **getNetUED** - this method generates a UED string containing the network conditions of the pipes that connect the input TVM to each of the terminal interfaces.

- **Signature:** `String getNetUED(String tvmPidURL);`
- **Input:** URL of the configuration service of the source of content;
- **Output:** UED string with Network Characteristics.

6. **createNetSLA** - this method creates a cSLA corresponding to a network service that supports the delivery of the selected content from the source to the terminal, satisfying the requirements of the adaptation decision taken by the ADE.

- **Signature:**
`public String createNetSLA(String tvmPidURLString,String adeDecision);`
- **Input:** URL of the configuration service of the source of content;
- **Output:** Identifier of the cSLA created.

7. **cancelNetSLA** - this method allows to cancel a previously generated cSLA.

- **Signature:** `public static void cancelNetSLA(String id);`
- **Input:** Identifier of the cSLA to be canceled.

8. **getServiceId** - this method allows to generate unique identifiers of services requested by customers.

- **Signature:** `public static String getServiceId();`
- **Output:** a unique identifier of a service.

9. **storeToDisk** - this method creates a file with the provided directory path and name and writes the input string to that file. Returns a handle to newly created file.

- **Signature:**
public static File storeToDisk(String dataContent, String dirPath, String fileName);
- **Input:** string of the data to write in the repository and the directory path and name of the file where the data should be written to;
- **Output:** a handle to a newly created file.

10. **retrieveFromDisk** - this method opens the file with the provided directory path and returns its content.

- **Signature:**
static String retrieveFromDisk(String storageDirPath,String fileName);
- **Input:** directory path and name of the file where the data should be read from;
- **Output:** a string with the contents of the input file.

11. **getMatchingServices** - this method returns the identifiers of active sources of content, delivering the input DI with the specified service type to the input access network.

- **Signature:**
static ArrayList<String> getMatchingServices(String aSFileName, String diID, String serviceType, String accessNetAddr);
- **Input:** Absolute name of the file containing the identifiers of agreed services, and information to match, which includes: the identifier of the DI

supported by the service, the service type and the IP address of the access network;

- **Output:** An array list with the service identifiers of matching active services.

It should be noted that the methods **getNetUED**, **createNetSLA** and **cancelNetSLA**, have correspondent methods implemented in the *NetSrvMgr*. They are implemented in the *CustSrvMgr* by the class *NetSrvProxy* that acts as a proxy of the *NetSrvMgr* subsystem. It isolates the *CustSrvMgr* class from the network details. The realm of the *CustSrvMgr* is essentially that of MPEG21, whereas the realm of the *NetSrvMgr* subsystem is that of the networks. Whereas the *CustSrvMgr* operates in the MPEG-21 world, dealing with MPEG-21 Usage Environment Descriptors (UED), the *NetSrvMgr* operates at the level of the network, dealing with cSLSs and pSLSs. It is therefore necessary at some point, to translate to and from MPEG-21 UEDs and SLAs/SLSs. Figure 5.20 illustrates this process.

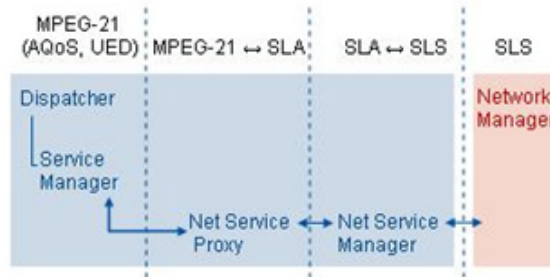


Figure 5.20. Service-level to network-level conversion

Adaptation Decision Engine subsystem The Adaptation Decision Engine (*ADE*) subsystem takes DI adaptation decisions in order to provide the best quality, given constraints on the terminal, network, adaptation engines (or TVMs) and user pref-

erences.

The *ADE* can be invoked both during service negotiation and during service delivery. The latter may happen when the measured QoS falls below the level agreed. In both cases it determines the set of service parameters that provide the best quality, given the above mentioned constraints. The *ADE* is based on MPEG-21 DIA and supports AQoS and UCD [121]. The core of the *ADE* is implemented an ADTE tool, developed by Hewlett Packard and which is part of MPEG21 reference software. Figure 5.21 presents the architecture of the *ADE* in terms of subsystems and interfaces

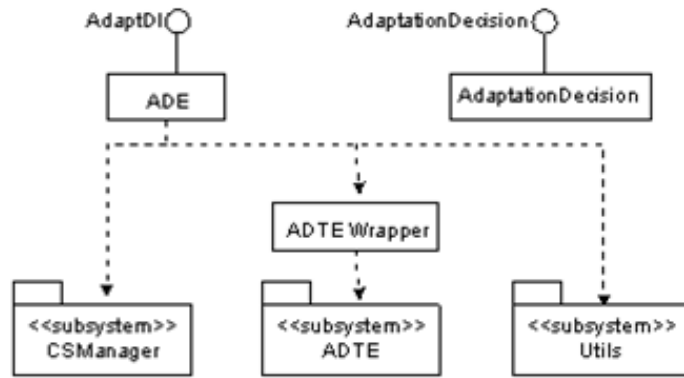


Figure 5.21. *ADE* interfaces and subsystems

When invoked, the ADE receives useful contextual and content-based descriptions (UED and AQoS). UEDs are transformed into UCDs, such that the capabilities of terminals and the user preferences, the characteristics of available adaptation engines and the capacity of the network may be transformed into constraints. The ADTE receives ordered sets of "*encoding parameters-quality values*" expressed in the AQoS description. It selects the first set that satisfies the constraints expressed in the UCD description. The ADE output is a set of "*name-value*" pairs (one for

each service parameter), selected among the AQoS descriptors originally provided. This output is used to configure the Adaptation Engine. This configuration is done using Web Services and associated SOAP messages.

The *ADE* interface provides the following operation:

- **decide** - this operation, implemented by the ADE, is used to request an adaptation decision satisfying the following constraints: the user desired service quality, the terminal capabilities, the network conditions and the AEs capabilities.
 - **Synopsis:** this operation is used to retrieve the best adaptation parameters given the current constraints.
 - **Input:**
 - * AQoS description;
 - * UED description received from the terminal;
 - * UED description provided by the *NetSrvMgr*;
 - * UED descriptions of the Adaptation Engine (AE);
 - * the user desired service quality.
 - **Output:**
 - * adaptation, or service, parameters to be subsequently passed to the AE.

Content Service Manager subsystem This subsystem is responsible for all service level management functionalities related to Content Providers. In particular it provides the means to request the implementation of remedial actions at the CP side. It passes service parameters obtained by an adaptation decision taking operation (the *decide* operation implemented by the ADE) to an Adaptation Engine (AE)

controller, in the EIMS-CM@CP. The following methods are implemented by this subsystem:

- **{config, getServices, start, stop}** - these methods are part of the class *AE-Controller*, which is essentially a Web Service client for dedicated Adaptation Engines controllers¹¹ This component should be seen as providing the functionality of a "plug-in", providing a bridge to specific equipment and being easily replaced.

Network Service Manager subsystem The *NetSrvMgr* is responsible for negotiation and invocation of SLA-based network services, and its implementation is greatly dictated by the network service level management approach adopted in EN-THRONE. Although this aspect was not part of the developments made within this thesis, it is worth to provide here a brief explanation to clearly understand the operation of the *Service Manager* and in particular of its *NetSrvMgr* subsystem.

For scalability reasons, the Service Provider, possibly based on service forecasting, negotiates contracts with Network Providers, for the establishment of large, point-to-point channels on a considerably large time scale (e.g., weeks, months). These channels, or virtual pipes, have specific QoS and bit rate guarantees, which are specified in a provider Service Level Specification (pSLS). The pSLS thus represents network resources that can be used by the SP to provide media services to its customers, during the time interval negotiated.

The pSLS is a fundamental concept to the EIMS, because it provides a very simple and well defined abstraction of the network. Network conditions and availability are

¹¹Within Enthrone, a specific AE Controller was developed by one of the partners. As such, this class was developed as to match the requirements imposed by that specific controller.

managed by the EIMS Dispatcher via its *NetSrvMngr*, using the pSLS.

This approach of network service planning and provisioning on a large time scale, is very interesting, because it avoids the need to negotiate with the network, every time a consumer requests a service:

- a pSLS describes the parameters of virtual pipes that connect Content Providers to a community of users (to an Access Networks, AN, serving a community of users). When a user, connected to a given AN #i, requests the consumption of a DI provided by a CP connected to AN #j, if there is already a pSLS connecting those two ANs, then all the EIMS needs to do, is to use that pSLS to know if the request can be accepted (and with which quality). It will also use it, to allocate the resources needed to provide the service, creating for that purpose a customer SLS (cSLS).

A cSLS, is another concept used by the *NetSrvMngr*, representing the network resources that are used to satisfy a particular service request. This request, in turn, is expressed as a customer Service Level Agreement (cSLA)).

Whereas a pSLS has typically a large lifespan, independently of the actual multimedia services that are being delivered by the SP, the cSLS lasts only for the duration of a particular service request.

Within this network service management approach, the main task of the *NetSrvMngr* is to negotiate and manage pSLSs. It issues cSLAs/cSLSs requested by the *CustSrvMngr* to provide multimedia QoS-enabled services to end-users.

To accomplish these tasks, the *NetSrvMngr* is composed of two main subsystems: the *SLA Request Handling* and the *pSLS/cSLS repository*. The *SLA Request Handling* handles cSLAs requested by the *CustSrvMngr*, using the information stored in the

pSLS/cSLS Repository. The following methods are made available by the *NetSvMngr* interface:

- **getpSLSState;**
- **createcSLA;**
- **getNetSLS;**
- **cancelcSLA.**

Statistics Manager subsystem The *StatsSrv* subsystem logs service requests from content consumers, both accepted and rejected. This information can be later used by the Service Planning system for service forecasting. It also allows the retrieval of the most popular DIs recently consumed. The architecture of this subsystem is illustrated in Figure 5.22.

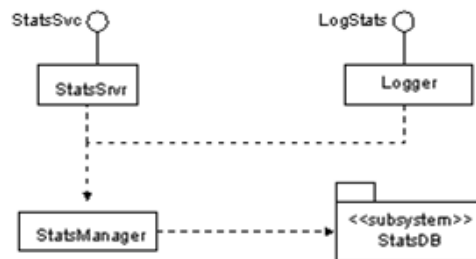


Figure 5.22. *StatSrv* interfaces and subsystems

The following are the operations implemented by the *StatsSrv*:

1. **getTop** - this operation, implemented by the *StatsSrv* interface, allows a CP or SP to request information on the most requested DIs, either from a

specific access subnetwork, or overall. A CP, as for example, a TV broadcaster delivering a digital TV interactive service, may use this information to decide the contents of the data carousel.

- **Synopsis:** this operation is used to request information on the most most requested DIs, either from a specific access subnetwork or overall.
- **Input:**
 - size of the "top" desired;
 - subnetwork address of interest.
- **Output:**
 - list with the top requested DIs, ranked in order of popularity.

2. **write** - this is a simple operation for writing a line to a log. As a side effect, the statistics Data Base may need to be updated.

- **Synopsis:** this operation is used to write a line to a log.
- **Input:**
 - A string containing the service ID;
 - the endpoint at which the content DI can be reached;
 - the IP address of the terminals access network.

The *Search Manager* subsystem

The implementation of the *Search Manager* subsystem of the Dispatcher, allows to isolate the modules and classes that actually perform the content searches (implemented inside the EIMS Content Manager subsystem), from the user request. This brings the following benefits:

- allows a more efficient management of multiple users' requests;
- provides an easy migration path, for example, for the support of new query formats;
- enables the SP to maintain more than one data base and corresponding access modules, or distributed data bases, while exposing one single interface to the end user.

In addition, it acts as a gateway between other modules of the Dispatcher and the EIMS Content Manager subsystem, thus simplifying the operation of the latter.

The essential operations implemented by this subsystem are offered via its *SearchDI* interface, as follows:

1. **SearchDI** - this operation is used to allow consumers to submit queries and receive back a list of useful results that matched the user's criteria. The request is passed to the EIMS data base access modules (resident on the EIMS-CM).
 - **Input:** XML string with a list of parameters and their values.
 - **Output:** DID with a list of items, one for each matching DI.
2. **getDI** - this operation allows to retrieve from the EIMS database, a specific DID corresponding to a given identified DI. This kind of operation is usually requested, when a user is browsing through a list of DIs and wishes to receive more information about one of those DIs. That extra information is contained in the DID and can be thus presented to the user.
 - **Input:** Digital Item ID, unique in the EIMS DB.
 - **Output:** the DID corresponding to the identified DI.

3. **getDIList**: This operation is similar to the **getDI** operation, except that, rather than allowing to request a single DID, it allows to request multiple DIDs.

- **Input**: array of Digital Item IDs.
- **Output**: a DID with an Item per Digital Item, which ID matched one of the input IDs.

5.4.2 The EIMS Content Manager Implementation

The *EIMS-CM* subsystem provides interoperation with Content Providers and thus is in charge, in first instance, of all aspects related to content:

- publishing and management of content and metadata;
- search for content;
- retrieval of content and metadata;
- configuration of sources of content;

In order to fulfil these tasks, it maintains a close cooperation with the EIMS-Dispatcher. As we have seen in the previous section, this sub-system has two different versions and consequently two different implementations - Content Manager at Service Provider (EIMS-CM@SP) and Content Manager at the Content Provider (EIMS-CM@CP). This approach was essentially motivated by the need of providing an easy and flexible mode of interfacing with Content Providers' existing equipment.

The EIMS-CM@CP implements functionality related to the generation of content according to the selected publishing format (i.e., MPEG-21 DI), and comprising all required content-related and contextual metadata. It also plays a role in the configuration of the sources of content, by passing the service parameters.

Sources of content that may be configured, are generically designated as Adaptation Engines¹². Finally, it also acts as the initiator of a media service, when that service is delivered by a real-time, live encoder.

The EIMS-CM@SP Performs the management of metadata and provides the support for content search. It allows metadata to be uploaded as compressed DIDs, via a SOAP interface. Information contained in the DID is extracted and indexed in a relational data base for enabling content search and retrieval. The DID is stored in an XML repository.

Search can be based on descriptive content-related metadata, as for example, TVA (TV Anytime), in which case the use issues a request with a list of keywords. Retrieval is usually done using MPEG-21 identifiers. Results are always passed as MPEG-21 DIDs.

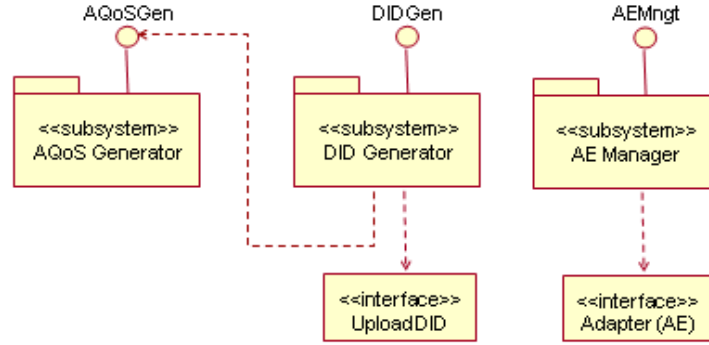
It should be noted that Content Providers and Service Providers are seen as roles and not necessarily as physical entities. This means that the same entity can play the two roles and thus that it will deploy the two versions of the EIMS-CM. For the sake of generality though, the two roles are seen as independent.

Figures 5.23 and 5.24 present the architecture of the two versions of the EIMS-CM, in terms of constituent subsystems and offered interfaces.

The **EIMS-CM@CP** includes two applications, the *AQoS Generator* and the *DID Generator*, to enable the generation of Adaptation QoS (AQoS) metadata and of MPEG-21 DIDs. They can be used as stand-alone applications or embedded in the EIMS architecture. In any case they provide always the same Web Services based interfaces.

The *DID Generator* can act as the client of the *AQoS Generator*, in order to insert

¹²Within the ENTHONE project, sources of content are designated as TV and Multimedia (TVM) processors. When it is possible to configure the TVM, it is designated as *aTVM* (adaptation TVM)

Figure 5.23. *EIMS-CM@CP* interfaces and subsystems

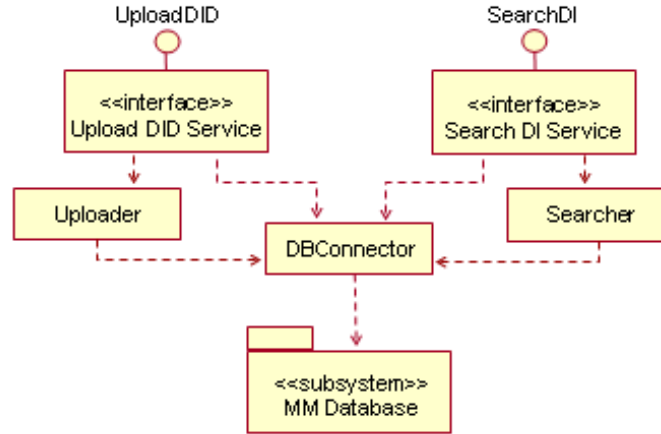
appropriate AQoS metadata into the DID being created.

The third subsystem implements functionality to configure and control Adaptation Engines. It offers services that are used by the EIMS-Dispatcher, more precisely by the *ContSrvManager* subsystem. We have described in the previous section the corresponding client methods, implemented in the *ContSrvManager*.

The *AQoS Generator* is a fundamental subsystem for the implementation of context and quality aware services. It provides the descriptions that related possible adaptation operations, service parameters and expected quality, thus enabling to implement a quality-based adaptation decision.

The *EIMS-CM@SP* implements functionality required for storing and retrieving data to and from the EIMS data base. For that purpose, it offers two interfaces, as illustrated in Figure 5.24 - *SearchDI* and *UploadDI* - and a data base system composed of a relational data base and an XML repository.

The interface *SearchDI* offers the same operations as the *SearchDI* interface of the *Search Manager* subsystem of the Dispatcher, as explained above in 5.4.1. However, it is the *EIMS-CM@SP* that implements the classes that provide the actual access

Figure 5.24. *EIMS-CM@SP* interfaces and subsystems

to the MPEG-21 database. This data base is part of the *EIMS-CM@SP*. It is maintained by the Service Provider, and is used for publishing the content as MPEG-21 DIs and for performing content searches.

Accordingly, the *EIMS-CM@SP* implements interfaces and corresponding classes that actually access the MPEG-21 data base. Bellow, we present the list of implemented operations and corresponding methods. For simplicity reasons, the operations of the *SearchDI* interface are not included in this list because they are identical to the operations provided by the *SearchDI* interface of the *Search Manager* subsystem of the Dispatcher (listed and described in section 5.4.1). Only the methods are included here.

1. **uploadDID**

- **Synopsis:** this operation is used to upload a DID onto the EIMS-CM data base.

- **Input:** an MPEG-21 DID.

2. **getDID**

- **Synopsis:** this operation can be used to download from the IMS-CM a DID previously uploaded.
- **Input:** the identifier of the DID to download;
- **Output:** the requested DID.

3. **searchDI** - this method is executed in two phases. In the first one, it makes use of the relational storage for finding DIs matching the user's request. The answer of the relational data base is a set of DI dentifiers. These are used in the second phase, the retrieval phase. In this phase, original items with the DI Identifiers obtained in the first phase are extracted from the XML repository.

- **Signature:** public String searchDI(String query);
- **Input:** XML string with a list of parameters and their values.
- **Output:** a DID with an item per matching DI.

4. **getDI**- this method access the XML repository and retrieves the DI matching a given DI Identifier.

- **Signature:** public String getDI(String diID);
- **Input:** a string with a DI Identifier.
- **Output:**a DID containing the matching item.

5. **getDIList** - this method access the XML repository and retrieves the set of DIs that match the provided DI Identifiers.

- **Signature:**public String[]getDIList(String[] ids);

- **Input:** string array of DI Identifiers.
 - **Output:** string array of DIDs containing the matching items.
6. **UploadDID** - this method parses the input DID and populates the relational DB for later processing of the search queries. It also stores in the XML repository, the items contained in the received DID. These are used in the retrieval phase of the operations of the Search DI interface.
- **Signature:** public String uploadDID(String did);
 - **Input:** XML string with the DID being uploaded;
 - **Output:** String indicating the outcome of the operation.
7. **getDID** - this method access the XML repository and returns the DID with the matching Identifier.
- **Signature:** public String getDID(String didID);
 - **Input:** String with a DID identifier
 - **Output:** the matching DID.

The EIMS Terminal Device Manager implementation

The EIMS TDM subsystem encapsulates the decision making process made locally in the terminal device regarding context management, notably, usage environment characteristics, QoS measurements and licenses-related data. It provides the means for the exchange of information between the user terminal and the EIMS subsystems responsible for the supervision and provisioning of the context and QoS-aware service. It supplies the EIMS Dispatcher with information regarding terminal capabilities, user preferences and usage rights, and the level of experienced QoS, which are essential for taking decisions regarding the provision and adaptation of

content to the user.

Depending on the capabilities of the terminal, the TDM might also encapsulate local adaptation operations.

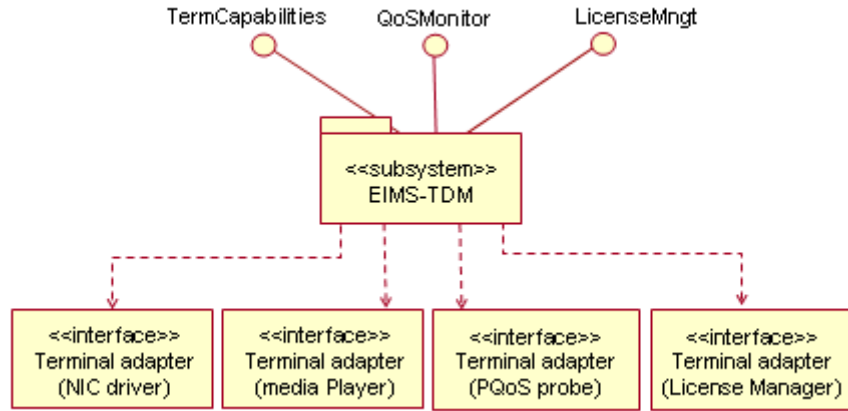


Figure 5.25. *EIMS-TDM* interfaces and subsystems

As illustrated in Figure 5.25, which presents its architecture, the TDM exposes the following interfaces:

- Terminal Capabilities (*TermCapabilities*);
- License Management (*LicenseMngt*);
- QoS Monitoring, (*QoSMonitor*).

The interface *TermCapabilities* is used during the *SelectDI* use case. The TDM cooperates with Terminal Adapters (media player, NIC, etc) to obtain the relevant information about the terminal capabilities and characteristics. It then generates a terminal UED which it sends to the Dispatcher.

The interface *LicenseMngt* implements operations to request to a specific adapter (License Manager) to verify if there are valid licenses on the terminal for the

consumption of a given DI, and accordingly notify the Dispatcher. These operations are performed within the course of the use cases *Select DI* and *Play DI*. When a license needs to be bought, the EIMS interacts with a License Provider on behalf of the user. In this case, the role of the TDM is to receive and store the acquired license on the terminal.

The interface *QoSMonitor* implements operations to allow collecting QoS alerts generated by a terminal Adapter, designated as *PQoS probe*¹³.

Depending on the mode of operation of the *PQoS probe*, the TDM may, or not, have to perform local processing. Usually, PQoS measurements are made on a periodically basis. Only the values that fall below and agreed threshold, should be considered as QoS alerts and accordingly sent to the Dispatcher.

The filtering of the measured values needs to be performed at the terminal side, so that the Dispatcher is not overloaded with unnecessary data. This filtering is done based on the initial QoS value agreed during service negotiation, and passed to the *PQoS probe* via the TDM.

Accordingly, the *QoSMonitor* also implements operations to allow the Dispatcher to pass initial PQoS values, needed to configure the *PQoS probe*.

5.5 Context awareness in the EIMS

From the above descriptions of the functionality and corresponding implementation of the EIMS subsystems, it is clear that the EIMS uses contextual information to assist service level monitoring.

Without entering into specificities pertaining to the implementation of each application scenario, the following can be considered as the metadata sources generically

¹³PQoS measurements could be also performed by the media player itself, which is already a terminal Adapter. For simplicity reasons, two adapters are considered, based not on physical entities but on the assumed role.

used by the EIMS:

- content or service providers and/or their equipment or adapters;
- terminal adapters;
- users;
- networking equipment and adapters;

The first source of metadata provides content-related metadata, as for example textual descriptions of the content included in the DID. But it also provides contextual metadata, notably the characteristics of Adaptation Engines (AEs) as follows:

- type of media supported;
- type of formats supported at the input and output;
- bit range operation point;
- aspect ratio range supported (in case of video sources).

All the other three sources provide contextual metadata, more precisely:

- capabilities of the terminal, notably the type of supported codec, screen size dimensions and network throughput at the terminal's NIC (network interface card), and perceived QoS measurements;
- preferences of the user in terms of preferred quality/price option;
- network resources availability and network QoS measurements.

So in fact, there are four types of context providers or sources of context: Content Providers, terminals, users and networks. The contextual information that they supply, can be seen as *low-level* or *sensed* context. They are represented by numeric or string values that can be used in direct comparisons with reference values. Some can be automatically acquired from appropriate sensors or adapters, while others may be manually supplied by humans. This is highly dependent on the availability of appropriate sensors on each context provider. It should be noted that, whereas contextual information pertaining to the context provider "user" is often a *high-level* context, in this case it is a *low-level* context, as it simply consists on a string with a certain value indicating the level of quality with which the user would like to consume the service.

These different types of contextual information, were the ones identified as required from the description of selected application scenarios. In particular, they enable the implementation of adaptation operations designated as *transcoding* and *scaling* from the "content manipulation" classification mode (please see table 3.1 in section 3.1.2). Obviously, the actual implementation of those adaptation operations, depends on the availability of adequate AEs. These two types of adaptations were identified as sufficient to support the requirements of the application scenarios and corresponding real-world situations. Nevertheless, the identified contextual could support other types of adaptation, as for example *filtering* and *transmoding*.

The objective of generating contextual information is to assist the adaptation decision taking in two situations: 1) when negotiating the service, upon a user request; and 2) throughout the service lifetime, whenever the sensed quality level is below the initially agreed value, or when changes in the sensed context require or allow an adaptation of the content. This last situation comprises the case referred in the VoD usage scenario (described in section 4), where a user, watching a movie

in his/her PDA (using a cellular transmission system) while traveling by train on his/her way back home, arrives home and transfers his/her session to a STB with an HDTV monitor and connected to a high-speed xDSL network.)

However, the goal is always the same: the provisioning of the best quality under the current usage context constraints and satisfying user preferences¹⁴.

Constraints represent restrictions to be applied to the complete set, or space, of service parameters' values. In turn, each set of parameters' values correspond to a given value in the resources' space, most often, a bit rate or bandwidth value. However, different sets of parameters may correspond to the same resource value that satisfies the constraints.

In that situation, it would be desirable to select the set of service parameters that yields the best quality. As such, it is necessary to establish relationships between quality, service parameter sets and constraints on the resources.

Moreover, to enable automated operation using those relationships those relationships should be described using a common format with well defined syntax and semantics. When the constraints and relationships are relatively simple, this can all be provided by the MPEG-21 DIA AQoS tool, in particular when used together with the MPEG-21 UCD tool, which allows to express the characteristics and conditions of the usage environment as constraints¹⁵.

Table 5.20 lists the MPEG-21 descriptions that can be used to represent the above listed contextual information and satisfy the adaptation requirements in specific situations or usage contexts, that may occur within the application scenarios described in section 5.1.

¹⁴We usually consider that the user preferences are part of the usage context. As such, they can be seen also as a constraint.

¹⁵For other more complex situations, ontologies can provide the means to express those more sophisticated concepts and relationships, and thus allow the use of similar types of sensed context to infer more complex higher level concepts.

Table 5.20. Mapping contextual information to MPEG-21 DIA descriptors

Usage Context / Situation	Nature/source	Group	Descriptor
Small screen at receiver.	Static constraints of the physical environment. This characteristic can be either inferred from the user request or be included in a terminal UED generated by some software module residing at the terminal.	Terminal	dia:TerminalCapability:DisplaysType:resolution
Network bandwidth limitation.	Static or dynamic conditions of the physical environment.	Network	dia:NetworkCapability:maxCapacity dia:NetworkCondition:AvailableBandwidth
Loss of quality at one terminal.	Dynamic characteristics of the content caused by some dynamic constraint of the usage environment. This information is provided by the terminal Adapter "PQoS probe".	Content	MPEG-7:MediaFormatType:QualityMeasure
Switching terminal type.	Dynamic characteristics of the context, notably change in terminal capabilities and network availability. This information is provided by terminal adapters and network adapters. .	Terminal Network	dia:TerminalCapability:DisplaysType:resolution dia:NetworkCapability:maxCapacity dia:NetworkCondition:AvailableBandwidth

Chapter 6

Validation of Results

Overview

This section presents the results obtained with demonstrations conducted in laboratory environment, as well as with performance tests, using the developed content mediation platform. The former aimed at proving the feasibility of the proposed approach and evaluating the extent at which the elicited requirements were being met.

The latter enabled to obtain some performance metrics, thus helping to assess latency introduced by the system, and eventually locate the source for major bottlenecks.

It should be however emphasized that these demonstrations and tests had to necessarily have a reduced scope, due mainly to four reasons:

- The high level of complexity of the developed infrastructure and possible deployment patterns;
- the necessity of generating a sufficiently comprehensive and meaningful set of content and associated MPEG-21 DIs representations;
- the lack of availability of multiple, distinct multimedia-enabled devices;

- the fact that the content mediation platform was developed within a collaborative research project, having had the direct involvement of other partners in its development.

Nevertheless it can be concluded that the major objectives were achieved. It was possible to successfully use the EIMS to provide: 1) the same multimedia services to different terminals, exhibiting different capabilities in terms of screen size and supported media codecs; 2) to provide services with different qualities and formats to the same terminal; 3) dynamic monitoring of the QoS during service lifetime, and the execution of remedial actions (adaptation decision and transrating of the video source using the newly selected service parameters) upon a simulated loss of quality due to network congestion.

In all three cases, latencies experienced by the system were within reasonably well accepted limits. The latency of the system in delivering useful results upon a user query and the time to take an adaptation decision were low and considered acceptable by the users of the system (in the order of 1-2 s).

However it was noticed that latency was heavily dependent on the size of the content DIDs and other XML documents necessary to process upon user requests (such as UED and AQoS).

The performance tests that were conducted, enabled only to have an estimation of the existence, extent and location of possible problems. It should be emphasized that the major concern when implementing the content mediator platform was not the efficiency in terms of speed, but rather efficiency in terms of supported functionality and flexibility. In other words, the specification and development of the platform favoured flexibility and scope rather than performance.

We believe that it is common understanding, that new concepts and ideas need first to prove that they can do the job. Only then, should they prove that they can do it

well.

Demonstrations

Several demonstrations were successfully conducted using the developed modules of the context mediation platform (i.e., the EIMS). Some were performed at public events (but with a limited audience, as it was the case of the demonstration performed during the annual audit of the ENTHRONE project or the one shown during a preparatory visit of the auditing panel of INESC Porto). Others, were conducted among the researchers involved in the project. In all cases the demonstrations occurred in controlled environments.

Figure 6.1 illustrates the setup of a demonstration performed at INESC Porto premises. That setup was used to demonstrate a very simple VoD application scenario with two different types of multimedia-enabled terminals.

Below is the description of the demonstration walkthrough:

- Two consumers, a Tablet PC and a PDA, initiate a search by connecting to the platform and sending a keyword-based query;
- The tablet PC using the DIBrowser, the PDA using a normal Web browser;
- They both receive a DID containing a list of matching DIs;
- They navigate through the DIDs, visualising embedded content and descriptions;
- Finally, they select a given DI for consumption indicating the preferred quality;
- The TDM checks if there is valid licenses and forwards the request to the EIMS at the server side;

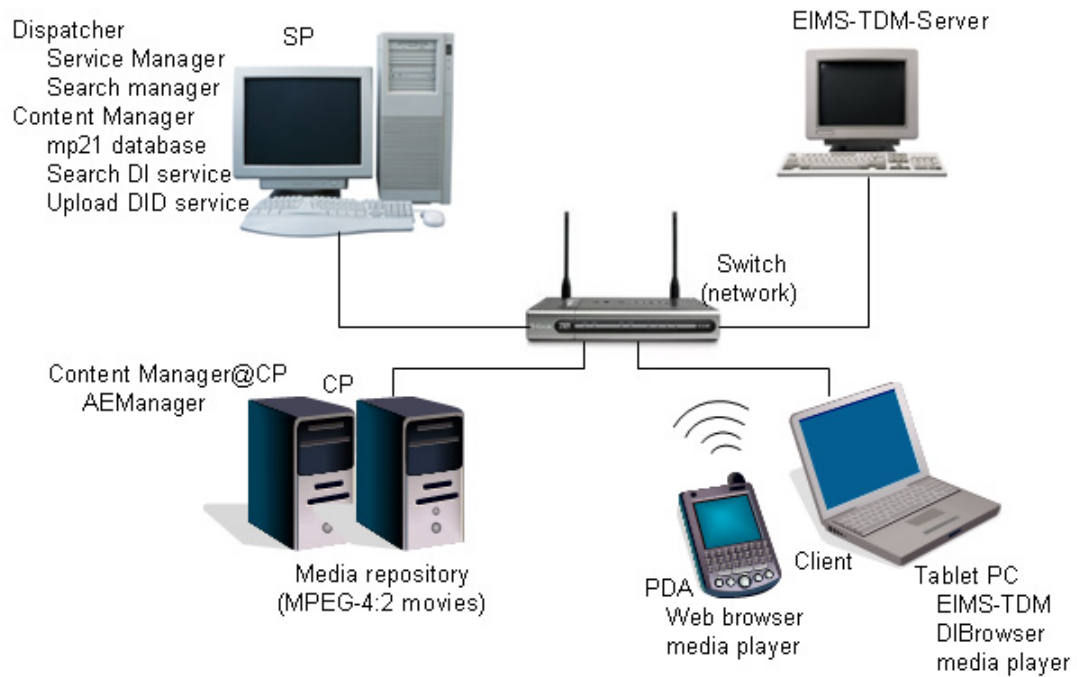


Figure 6.1. Laboratory demonstration setup

- the EIMS Service Manager subsystem of the Dispatcher gathers required contextual information;
- The EIMS Dispatcher determines the best service parameters given the capabilities of the terminal, network conditions and user preferences;
- The ADE is invoked to take a quality-based adaptation decision;
- Encoding parameters corresponding to the chosen operating point are passed to the AE Controller of the EIMS Content Manager subsystem at the CP site;
- adequate network resources are allocated in a previously contracted pSLS;
- The EIMS Service Manager passes the service end point to the terminal (the

url where to find the adapted content);

- The consumer receives the multimedia content adapted to the his/her context characteristics.

Figure 6.2 show the sources of context (or context providers) involved in this usage scenario and ilustates the process of gathering and using the generated context.

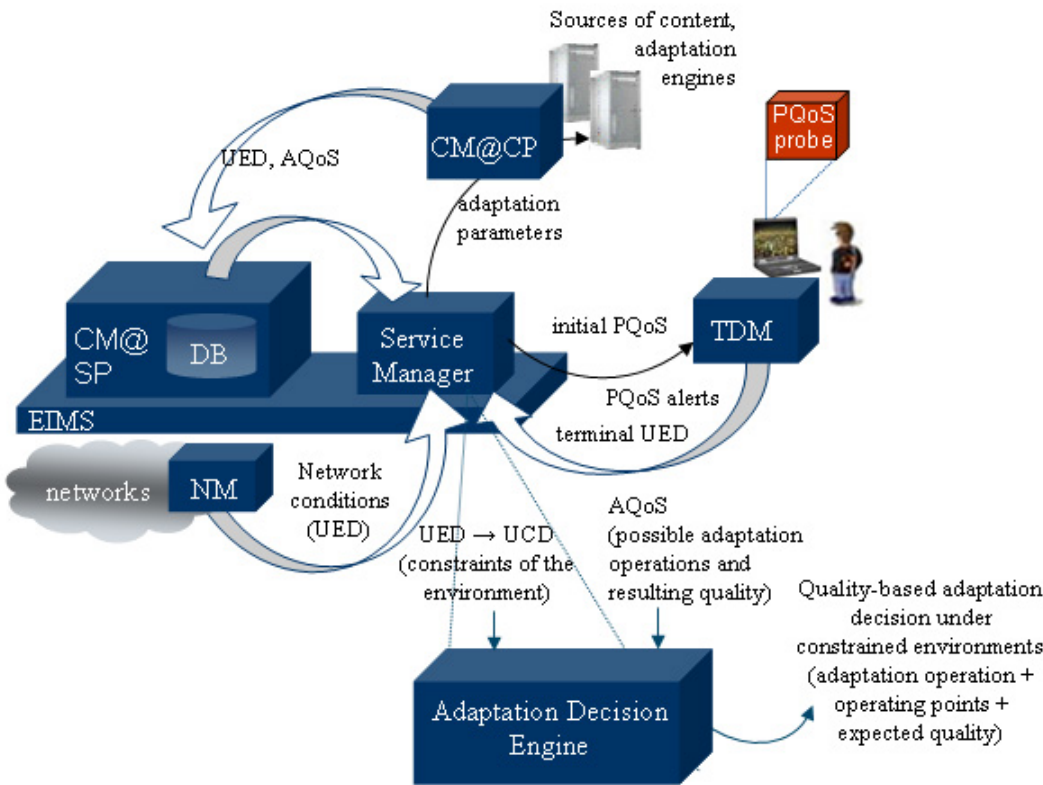


Figure 6.2. Context providers and gathering of context

Performance tests

To conduct performance tests, very simple dedicated clients were developed to invoke various services made available by the EIMS. Below we describe the client developed, designated as "*Testing client*" the tests conducted and results obtained.

Testing client is application developed with the purpose of simulating simultaneous accesses to EIMS services.

The application was developed in Java using the JAX-RPC technologies provided by the Java Web Services Developer Pack 1.5.

The EIMS subsystems offer several services to external entities or applications, to accomplish a number of operations, namely:

- Digital Item searching, DI selection, DI consumption, DID upload and download, retrieval of statistics about DI consumption and reconfiguration of AEs.

The developed application creates several threads, depending of the number of virtual clients that we intend to simulate accessing EIMS services. Each thread represents one virtual client and exchanges information with the EIMS services using SOAP messages. The code below, provides a mechanism to measure execution times of the invoked methods, i.e., the response time of EIMS to invoked services.

```
for(int i=0;i<a;i++) {
    Fir1 fir1 = newFir1(); //runnable class
    Thread myThread = new Thread(fir1);
    myThread.start();
```

Measured time is calculated from the moment when the first thread is created until the last thread finishes.

Tests were made for the following EIMS services: *selectDI*, *searchDI*, *getDI*, *uploadDID*, *getDID*, and *getTop*.

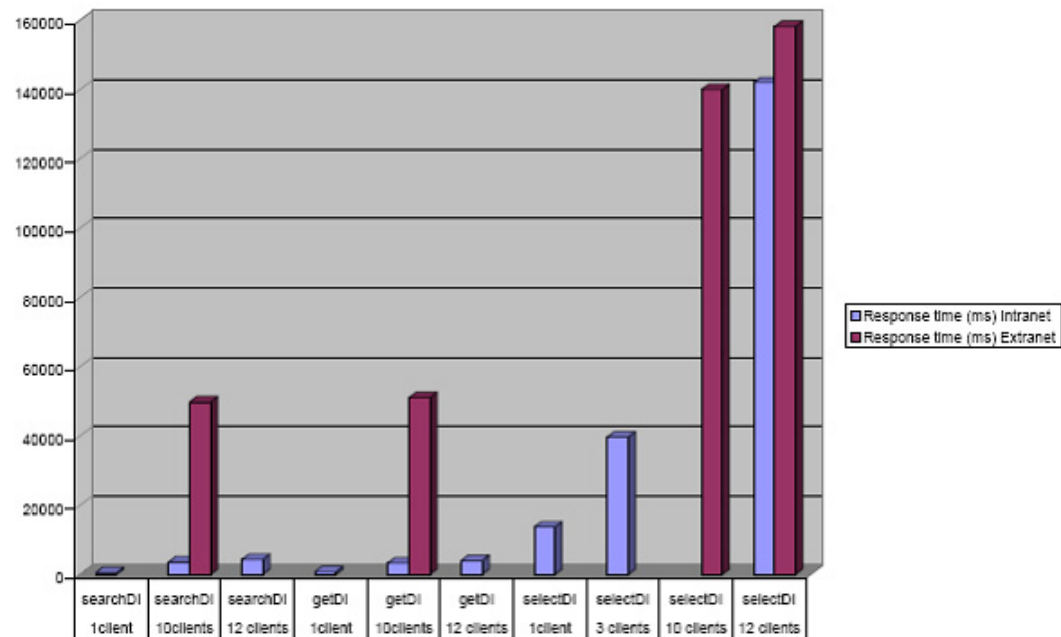


Figure 6.3. DI service response time

For the services *uploadDID* and *getDID*, a DID file of size 6KB was used. The diagrams below, presented in Figures 6.3, 6.4 and 6.5, show response times, when the services are invoked by a different number of simultaneous virtual clients. The tests were made with the EIMS running on a PC with x86 architecture, CPU Pentium IV at 3.2 GHz and with 512M Ram memory. Moreover, the tests were conducted with clients located both inside the same Intranet as the EIMS and outside. In the latter case an ISP services were used, offering an Internet access with 512Kbps downstream and 128Kbps upstream link.

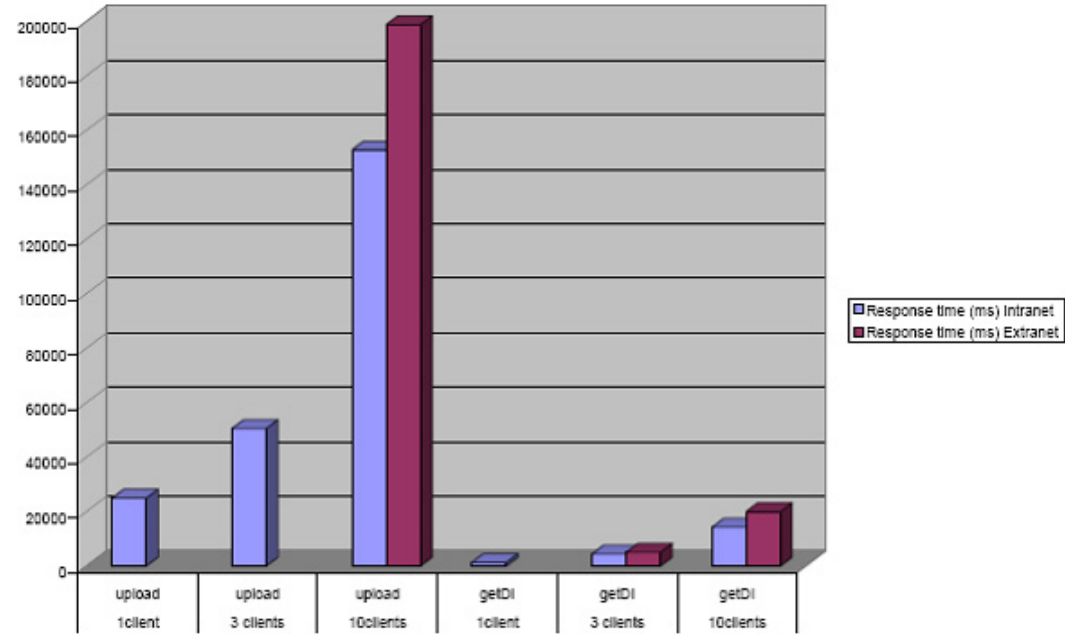


Figure 6.4. uploadDID service response time

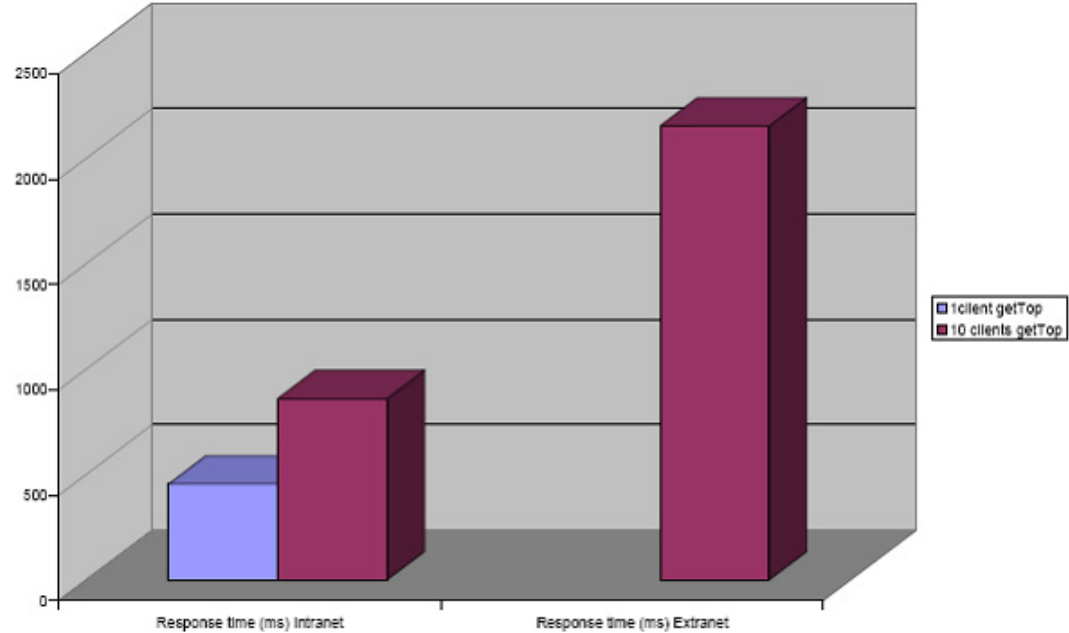


Figure 6.5. Stats service execution time

Chapter 7

Conclusions

7.1 Conclusion

The work reported in this dissertation can be seen as a an integration of studies, research, experiments and developments on different, but related, topics, where the "glue" is provided on one hand, by the common objective of contributing to the fulfilment of the ubiquitous access, and on the other hand by the use of a common tool - metadata.

Indeed, metadata plays an central role across the different axis of developed work:

- it is essential for effectively locating content in an automated way, thus potentiating new forms of use for any asset and widen the access to knowledge;
- it is instrumental within the process of adapting the discovered content to a format suitable to be consumed by the person or application that searched for it (and thus, also potentiating new forms of use of content ...);
- it enables to automate the operation and decisions of resource allocation mech-

anisms, potentiating a better use of shared resources;

- it is in fact, the vehicle through which it is possible to introduce some degree of automation in any kind of decision-taking processes.

As we had already concluded in this manuscript, upon presenting a review of the state-of-art in metadata schemes, metadata has become a core element of any system mediating the access to any kind of resource (content, metadata, services, applications, computing resources, etc).

Moreover, the need for standardised formats for representing metadata seems now to be unquestionable.

The reference architecture developed within this thesis, the EIMS, makes extensive use of both metadata and open standards. More precisely, the MPEG-21 framework and Web Services technologies.

The MPEG-21 framework serves two main purposes:

- common unified format: the use of Digital Items (in particular the DID) as a common intermediate format, by which two resources created under different, possibly incompatible formats, could be equally searched using the same search engines, their existence equally communicated to interested users, and then be eventually consumed under their original format, without the need to operate any kind of transformation upon the content.
- to provide a suite of standards-based tools to describe context of usage and assist in a consistent way, adaptation operations.

Within the EIMS, content is published according to the MPEG-21 DID model and extensive use is made of the adaptation tools of the standard, notably those specified in MPEG-21 DIA (Digital Item Adaptation). In particular, extended

use is made of the UED tool to characterize the context of usage, namely the terminal characteristics including network interface static characteristics and dynamic conditions, network characteristics and conditions, and characteristics of encoders and adaptation engines.

The AQoS tool is used together with the UCD tool to provide an automated quality-based adaptation decision under constrained environments. This is done at connection set-up, i.e., when selecting and negotiating the service to obtain the initial service parameters, as well as during service lifetime (dynamically).

Proper use is made of the AQoS metadata, via a database holding perceived quality measures obtained off-line using PQoS objective measurements. The matching between perceived quality values and the service parameters was obtaining using empirical data. This is a very sensitive aspect - only preliminary results were obtained and certainly more work could be performed in this area to obtain good models that relate service parameters with perceived quality.

The developed content mediation platform is universally accessible, and reusable in different environments and applications, as it offers its value via Web Services (WS). This provides the platform with a flexible and universal mode to interoperate with external devices or entities.

For example, the response of the ADE (an XML document) is captured by the coordinating sub-system of the EIMS (the Dispatcher) and redirected to remote Adaptation Engines using Web Services interfaces. Likewise, the perceived quality measurements made at the terminal by a third-party PQoS probe, or the network measurements, are pushed to the EIMS using Web Services. The same applies to the communication with network management systems to request the establishment of SLAs/SLSSs.

All external communication is accomplished using Web Services technolo-

gies. Internal communication may also be performed via WS; however, in this case, the use of WS is conditioned by a careful balance between performance and flexibility.

Essential to any networked platform operating in heterogeneous environments, is the use of communication middleware. This layer enables the system components to be deployed, replicated or divided at different locations, ensuring the communication and signalling between them and enabling the provision of remote added-value services. The use of Web Services (WS) came as a natural choice to satisfy these needs. It enabled the development of a service oriented architecture, facilitating the inter operation between the different components of the system, and the easy deployment in a wide range of scenarios.

The exchange of information is accomplished using SOAP (Simple Object Access Protocol), which is a technology that pairs well with MPEG-21, as both are based on XML. WS technologies provide an open and universal mode for client devices and external applications to access and use the platform services. As it is widely used across the Internet and universally supported, it enables the establishment of a variety of business models.

To the best of our knowledge, there are no asset management systems available, that completely base their functionality and operation on open standards, as is the case of the the MPEG-21 specifications and Web Services technologies.

The EIMS identifies all content, regardless of its original format, as MPEG-21 Digital Items, performs searches and locates the useful content using the DIDs, performs content adaptation decision using MPEG-21 DIA metadata, contained in the DIDs or obtained from the participating entities, and exchange them in standardised DIDs.

Moreover, the EIMS is a distributed system (which is the other central aspect of the EIMS together with MPEG-21) that can be deployed in diverse and flexible ways.

Likewise, the use of Web Services technologies, increases greatly its scope of application and promotes the universal access from a number of external terminals, devices and applications.

There is a considerable risk in specifying and developing such a system, due to the fact that it interacts with a diversified range of entities, devices and applications. This concern was addressed through the adoption of a modular and Service Oriented Architecture (SOA). It led to a flexible architecture capable of coping with, and easily upgrade to, new incoming requirements. In addition, it should be considered that the various entities with whom the EIMS interacts, come from various domains (Content Creation and Distribution, Content Protection, Networking, Consumer Electronics), with diverse skills and technology backgrounds, and where the same concept or fact will most probably be interpreted in different ways.

The great strength and risk of the EIMS is that it provides a complete solution. It is not just a system that is able to keep a data base and repository of DIDs/DIs and allow users to search/browse through it; it is not just a system that performs content adaptation decision based on user terminal characteristics; it is not just a system that performs service-level negotiation management; not just a system that monitors the QoS of the service; not just a system that performs the verification of licenses; Still, with improvements to be made!

7.2 Summary of Contributions

The following list presents the main contributions of this thesis:

1. specification of a reference architecture, in the form of a set of services and interfaces, to enable the provision of context-aware adaptable multimedia services:

- based on emergent open standards;
 - using off-the-shelve technology (i.e., PCs);
 - operating across heterogeneous environments at all levels: content, networks and terminals;
 - supporting interaction with all players in the multimedia content value chain;
 - enabling easy migration to new technology while promoting the interoperation with legacy-systems;
 - enabling re-usability of functionality and easy reconfiguration to suit different or new requirements;
 - setting the path for the fulfilment of the ubiquitous access concept.
2. implementation and test of an instance of the specified reference architecture based on requirements extracted from real world application scenarios, thus proving the feasibility of the adopted approach.
 3. elaboration of a methodology for the classification and representation of various types of context, relevant for the implementation of context-aware services.
 4. specification of the abstract functional architecture of an context-aware adaptation decision engine, using different types of contexts and relaying on the extensive use of semantics.
 5. identification of research directions to develop solutions for sophisticated and interoperable context-aware content adaptation requirements, by merging concepts and technologies of the Semantic Web with MPEG technologies.

6. Proposal of a probabilistic framework supported on the Bayes theory for statistical video multiplexing, adopting an *open-loop* approach and thus preventing the necessity of establishing a feedback path to the actual source of bits.

7.3 Future Research

Considering the results obtained and the topics and directions explored during the development of this thesis, the following is a list of aspects that are considered both as interesting and useful to be pursuit:

- to investigate and implement new forms of reactive measurements upon the detection of a quality degradation.
- to explore the use of distributed Adaptation at full potential, including distributed adaptation decisions and distributed adaptation operations, at any point of the content value chain. This poses considerable challenges regarding the availability and use of adaptation engines placed at the edges of Autonomous Systems.
- to combine the use of ontologies and higher level concepts with the low-level description framework of MPEG-21 to enable to design more complex context-aware adaptation situations.
- to explore the possibility of developing application-specific and/or domain-specific ontologies, aligned with corresponding profiles of the MPEG-21 DIA description tools.
- to develop content adaptation strategies that take into account, at a finer detail, Digital Rights Management (DRM) information.

Regarding the line of work dedicated to statistical multiplexing, specific directions of work are included in the section dedicated to that topic (included as an Annex for reasons already explained).

Bibliography

- [1] *Oasis specification: Universal Description, Discovery, and Integration, UDDI v3.0*, 2006. Available on-line at [http : //uddi.org/](http://uddi.org/).
- [2] ISO TC 46/SC. *ISO 15836:2003(E), Information and documentation The Dublin Core metadata element set*, Feb 2003. Available on-line at [http :
//www.niso.org/international/SC4/n515.pdf](http://www.niso.org/international/SC4/n515.pdf).
- [3] W. Pratt A. Farquhar, R. Fikes and J. Rice. Collaborative ontology construction for information integration. Research Report KSL-95-63, Knowledge Systems Laboratory Department of Computer Science, Stanford, August 1995. Also available as <http://www-ksl-svc.stanford.edu:5915/>.
- [4] H. Stern A. Gelman, J. Carlin and D. Rubin. *Bayesian Data Analysis*. 2000.
- [5] Open Mobbile Alliance. *UaProf, User Agent Profile, OMA Standard V2 0-20060206-A*, Feb 2006.
- [6] M. T. Andrade. Managing multimedia content and delivering services ... In *Proceedings of the International Conference Mobile HCI 2002*, Pisa, Italy, September 2002.
- [7] M. T. Andrade. A statistical framework to enlarge the potential of digital

- tv broadcasting. In *Proceedings of the International Symposium ISPA 2003*, Rome, Italy, September 2003.
- [8] M. T. Andrade. In *Proceedings of the Eumob 2006 Workshop, as part of the MobiCom 2006 conference*, Sardinia, Italy, September 2006.
- [9] M. T. Andrade and A. P. Alves. New perspectives for digital television,. In *Proceedings of the International Conference on Media Futures*, Florence, Italy, May 2001.
- [10] Ambjorn Naeve Pete Johnston Andy Powell, Mikael Nilsson and Thomas Baker. Dcmi abstract model, 2007. Available at [http : //dublincore.org/documents/abstract – model/](http://dublincore.org/documents/abstract-model/).
- [11] Axis Development Team Apache Software Foundation. *Apache Web Services project - Axis Page*. Home page at the url [http : //ws.apache.org/axis/](http://ws.apache.org/axis/).
- [12] Advanced Media Workflow Association. The amwa web site, 2007. Web site at [http : //www.aafassociation.org/](http://www.aafassociation.org/).
- [13] M. Beard B. Devlin, J. Wilkinson and P. Tudor. *The MXF Book*. Focalpress, Oxford, UK, 2006.
- [14] P. Sen G. Karlsson B. Maglaris, D. Anastassiou and J. Robbins. Performance models of statstical multiplexing in packet video communications. *IEEE Transactions on Communications*, 36(7):834–843, July 1988.
- [15] P. Salembier B. S. Manjunath and T. Sikora. *Introduction to MPEG-7: Multimedia Content Description Language*. John Wiley and Sons Ltd, 2002.
- [16] N. Adams B. Schilit. and R. Want. Context-aware computing applications. In

Proceedings of the IEEE Workshop on Mobile Computing Systems and Applications, pages 85–90, Santa Cruz, USA, December 1994.

- [17] BBC. The standard media exchange framework data model (smef-dm), 2000. Brief description and specification available at *http* : *//www.bbc.co.uk/guidelines/smef/*.
- [18] Tim Berners-Lee. Semantic web road map, 1998. Available in the url *http* : *//www.w3.org/DesignIssues/Semantic.html*.
- [19] Dan Brickley and R.V. Guha. RDF Vocabulary Description Language 1.0: RDF Schema. W3C Recommendation, 10 February 2004. Available at the url *http* : *//www.w3.org/TR/owl-semantics/*.
- [20] A. Kpper Buchholz and M. Schiffrs. Quality of context: What it is and by we need it. In *Proceedings of the 10th Workshop of the OpenView University Association, OVUA'03*, Geneva, Switzerland, July 2003.
- [21] M. H. Butler. Current technologies for device independence. Public report, Hewlett Packard Laboratories, Bristol, UK, April 2001.
- [22] M. H. Butler. Implementing content negotiation using cc/pp and wap uaprof. Public report, Hewlett Packard Laboratories, Bristol, UK, August 2001.
- [23] L. Lambadaris C. Huang, M. Devetsikiotis and A. Kaye. Modeling and simulation of self-similar variable bit rate compressed video: A unified approach. In *Proceedings of the ACM SIGCOMM conference*, 1995.
- [24] T. DeMartini C. Timmerer and H. Hellwagner. he mpeg-21 multimedia framework: Conversions and permissions. In *Proceedings of DACH Security 2006*, pages 225–235, Dsseldorf, Germany,, March 2006.

- [25] Pedro M. Carvalho. Multimedia content adaptation for universal access. Master's thesis, Faculdade de Engenharia da Universidade do Porto, Rua Dr. Roberto Frias, 4200-465 Porto, Portugal, July 2004.
- [26] S.-F. Chang and A. Vetro. Video adaptation: Concepts, technologies and open issues. *Proceedings of the IEEE*, 93(1):148–158, Jan 2005.
- [27] Leonardo Chiariglione. The mpeg home page. Accessible at the url *http : //www.chiariglione.org/mpeg/*.
- [28] D. M. Cohen and D. P. Heyman. A simulation study of video teleconferencing traffic in ATM networks. In *Proceedings of the IEEE INFOCOM'93 conference*, pages 894–901, 1993.
- [29] Contessa Consortium. The contessa web site, 2003. Web site at *http : //www.contessa.org/*.
- [30] DANAE Consortium. The danae project web site, Nov 2005. Web site in the url *http : //danae.rd.francetelecom.com/*.
- [31] Enthroner Consortium. Enthroner web site, 2006. Web site in the url *http : //www.enthrone.org/*.
- [32] Microsoft Corporation. Dcom technical overview, 1996. Available at the url *http : //msdn2.microsoft.com/en - us/library/ms809340.aspx*.
- [33] Netscape Corporation. *The SSL Protocol*, November 1996. Available on-line at the url *http : //wp.netscape.com/eng/ssl3/draft302.txt*.
- [34] A. Tabatabai D. Heyman and T. Lakshman. Statistical analysis of MPEG2-coded VBR video traffic. In *Proceedings of Sixth International Workshop on Packet Video, PV'94*, September 1994.

- [35] M. T. Andrade C. Timmerer D. Oancea, P. Carvalho and H. Hellwagner. Defect report on iso/iec 21000-15. Contribution to standardisation MPEG2007/M14502, ISO/IEC JTC1/SC29/WG11, San Jose, USA, April 2007.
- [36] A. Tabatabai D. P. Heyman and T. Lakshman. Statistical analysis and simulation study of video teleconferencing in ATM networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 2(1):49–59, March 1992.
- [37] Dennis Wagelaar Andy Georges Peter Rigole Tim Clerckx Yolande Berbers Karin Coninx Viviane Jonckers Davy Preuveneers, Jan Van den Bergh and Koen De Bosschere. Towards an extensible context ontology for ambient intelligence. In Panos Markopoulos, Berry Eggen, Emile Aarts, and James L. Crowley, editors, *Second European Symposium on Ambient Intelligence*, volume 3295 of *LNCS*, pages 148 – 159, Eindhoven, The Netherlands, Nov 8 - 11 2004. Springer.
- [38] Peter Rigole Dhouha Ayed Davy Preuveneers, Yves Vandewoude and Yolande Berbers. Context-aware adaptation for component-based pervasive computing systems. In *Advances in Pervasive Computing 2006, Adjunct Proceedings of the 4th International Conference on Pervasive Computing*, pages 125–128, May 2006.
- [39] Anind K. Dey. *Providing Architectural Support for Building Context-Aware Applications*. PhD thesis, Georgia Institute of Technology, 2000.
- [40] Bob Edge. Gxf the general exchange format. *EBU Technical Review*, (291), July 2002. Article available on-line at [http : //www.ebu.ch/en/technical/trev/trev291 – edge.pdf](http://www.ebu.ch/en/technical/trev/trev291-edge.pdf).

- [41] F. Pereira editor. Special issue on multimedia adaptation. *Signal Processing: Image Communication*, 18(8), Sept 2003.
- [42] Abdur-Rahman El-Sayed. *Semantic-Based Context-Aware Service Discovery in Pervasive-Computing Environments*. PhD thesis, University of Waterloo, Waterloo, Ontario, Canada,, 2006.
- [43] J. Enssle. Modelling and statistical multiplexing of VBR MPEG compressed video in ATM networks. In *Proceedings of the 4th Open Workshop On High Speed Networks conference*, pages 59–67, September 1994.
- [44] Aparone et. al. Statistical models for packet video communications. *Digital Signal Processing'91*, pages 654–659, 1991.
- [45] D. Reininger et. al. Statistical multiplexing of VBR MPEG compressed video on ATM networks. In *Proceedings of the IEEE INFOCOM'93 conference*, pages 919–926, March 1993.
- [46] Jabbari et. al. Statistical characterisation and block-based modeling pf motion-adaptive coded video. *IEEE Transactions on Circuits and Systems for Video Technology*, 3(3):199–207, June 1993.
- [47] T. Wiegand et al. Rate-constrained coder control and comparison of video coding standards. *IEEE Trans. on Circuits and Systems for Video Technology*, 13(7):688–703, July 2003.
- [48] Melamed et. alt. TES-based traffic modelling for performance evaluation of integrated networks. In *Proceedings of the IEEE INFOCOM'92 conference*, pages 75–84, April 1992.
- [49] ETSI. *ETSI TS 102 822-1 V1.3.1 (2006-01)*.

- [50] ETSI. *ETSI TS EN 300 468, Digital Video Broadcasting (DVB): Specification for Service Information (SI) in DVB Systems*, 2003.
- [51] ETSI. Etsi publications download area, 2007. Web site in the url *http : //www.etsi.org/services_products/freestandard/home.htm/*.
- [52] Olli Seppala Eugene A. Gryazin, Juha O. Tuominen. Heterogeneous middle-ware structures for mygrocer project. Tehnical report, Helsinki University of Technology, Dpt of Computer Science and Engineering, Interactive IT Laboratory, Espoo, Finland, 2002.
- [53] F. Ben Amara F. Kossentini and A. Jerbi. Iemerging h.264 standard supports broadcast video encoding. *EE Times*, January 2003.
- [54] editors F. Pereira et al. Special issue on universal multimedia access. *IEEE Signal Processing Magazine*, 20(2), March 2003. 2003.
- [55] International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC). *ISO/IEC 15938-5:2004, Information Technology - Multimedia Framework (MPEG-21), Part 5: Rights Expression Language*, 2004.
- [56] DVB Forum. *Digital Video Boadcasting Web site http : //www.dvb.org/*, 2007.
- [57] TV-Anytime Forum. The tv-anytime web site, 2007. Home page in the url *http : //www.tv - anytime.org/*.
- [58] Eclipse Foundation. Eclipse - an open development platform. Home page at the url *http : //www.eclipse.org/*.
- [59] Eric Miller Frank Manola and Brian McBride. *W3C Recommendatio: Resource*

- Description Framework (RDF) Primer*, February 2004. Available at [http :
//www.w3.org/TR/2004/REC - rdf - primer - 20040210/](http://www.w3.org/TR/2004/REC-rdf-primer-20040210/).
- [60] Chen G and Kotz D. A survey of context-aware mobile computing research,. Technical Report TR2000-381, Dartmouth College, Dept. of Computer Science, 2000.
- [61] H. Asgari G. Kormentzas, M.T. Andrade and C. Skianis. Service management enhancements to ims architecture. *International Journal of Network Management*, 17, 2007.
- [62] M. Garret and W. Willinger. Analysis, modelling and generation of self-similar VBR video traffic. In *Proceedings of the ACM SIGCOMM'94 conference*, 1994.
- [63] M. W. Garrett and M. Vetterli. Joint source/channel coding of statistically multiplexed real-time services on packet networks. *IEEE/ACM Transactions on Networking*, 1(1):71–80, February 1993.
- [64] M. W. Garrett and W. Willinger. Analysis, modeling, and generation of self-similar VBR video traffic. In *Proceedings of the ACM SIGCOMM'94*, pages 269–280, September 1994.
- [65] J. Gilbert and M. Butler. Using multiple namespaces in cc/pp and uaprof. Public report, Hewlett Packard Laboratories, Bristol, UK, January 2003.
- [66] Michael W. Engel Bobbi J. Young Jim Conallen Grady Booch, Robert A. Maksimchuk and Kelli A. Houston. *Object-Oriented Analysis and Design with Applications (3rd Edition)*. Addison-Wesley Professional, 3rd edition, 2007.
- [67] IEEE SUO WG (Standard Upper Ontology Working Group). Ieee p1600.1,

standard upper ontology working group (suo wg) home page, 2003. Available at the url [http : //suo.ieee.org/](http://suo.ieee.org/).

- [68] T. Gruber. A translation approach to portable ontology specifications. *Knowledge Acquisition*, 6(2):199–221, Feb 1993.
- [69] Pung H. K. Gu, T. and D. Q. Zhang. A service-oriented middleware for building context aware services. *Elsevier Journal of Network and Computer Applications (JNCA)*, 28(1):1–18, Jan 2005.
- [70] M. T. Andrade C. Timmerer H. Castro, P. Carvalho and H. Hellwagne. A did model for media streaming maf. Contribution to standardisation M14503, ISO/IEC JTC1/SC29/WG11, MPEG2007/M14503, San Jose, USA, April 2007.
- [71] M.R. Ebling H. W. Gellersen G. Hunt G.D. Abowd and H. Lei. Context-aware pervasive computing. *IEEE Wireless Communications*, 5(9):8–9, October 2002.
- [72] Reena Hanumansetty. Model based approach for context aware and adaptive user interface generation. Msc, Virginia Polytechnic Institute and State University, Falls Church, Virginia, USA, July 2004. Also available as [http : //scholar.lib.vt.edu/theses/available/etd – 08242004 – 120131/unrestricted/MastersThesis.pdf](http://scholar.lib.vt.edu/theses/available/etd-08242004-120131/unrestricted/MastersThesis.pdf).
- [73] B. Haskell and A. Reibman. Multiplexing of variable rate encoded streams. *IEEE Transactions on Circuits and Systems for Video Technology*, 4(4):417–424, August 1994.
- [74] Mihaela van der Schaar Hayder M. Radha and Yingwei Chen. The mpeg-4 fine-grained scalable video coding method for multimedia streaming over ip. *IEEE Transactions on Multimedia*, 3(1):53–68, March 2003.

- [75] Richard Hopper. Ebu project group p/meta metadata exchange standards. *EBU Technical Review*, (284), Sept 2000. Article available on-line at <https://www.ebu.ch/en/technical/trev/trev284-hopper.pdf>.
- [76] Richard Hopper. Ebu project group p/meta metadata exchange scheme v1.0. *EBU Technical Review*, (290), April 2002. Article available on-line at <https://www.ebu.ch/en/technical/trev/trev290-hopper.pdf>.
- [77] Ian Horrocks. Ontologies and the semantic web. The University of Manchester.
- [78] Ian Horrocks, Peter F. Patel-Schneider, and Frank van Harmelen. From *SHIQ* and RDF to OWL: The making of a web ontology language. *J. of Web Semantics*, 1(1):7–26, 2003.
- [79] R. Walle I. Burnett, F. Pereira and R. Koenen. *The MPEG-21 Book*. John Wiley and Sons Ltd, 2006.
- [80] IETF. *Dublin Core Metadata for Resource Discovery, Internet RFC 2413*, Sept 1998. Available on-line at <http://www.ietf.org/rfc/rfc2413.txt>.
- [81] IETF. *RPC: Remote Procedure Call Protocol Specification, Internet RFC 1050*, April 1998. Available on-line at <http://www.ietf.org/rfc/rfc1050.txt>.
- [82] Sun Microsystems Inc. *The Java Tutorials - Trail: RMI*, 2007. Home page in the url <http://java.sun.com/docs/books/tutorial/rmi/index.html>.
- [83] Dublin Core Metadata Initiative. The dcmi web site, 2007. Web site at <http://dublincore.org/>.
- [84] J. Elson Internet Engineering Task Force (IETF) Networking Group and A. Cerpa. *Request for Comments: 3507, Internet Content Adaptation Protocol (ICAP)*, April 2003.

- [85] R. Fielding Internet Engineering Task Force (IETF) Networking Group, T. Berners-Lee and L. Masinter. *RFC 2396, Uniform Resource Identifiers (URI): Generic Syntax*, August 1998.
- [86] R. Fielding Internet Engineering Task Force (IETF) Networking Group, T. Berners-Lee and M. McCahill. *RFC 1738, Uniform Resource Locators (URL)*, December 1994.
- [87] T. Dierks Internet Engineering Task Force (IETF) Networking Group and E. Rescorla. *Request for Comments: 4346, The Transport Layer Security (TLS) Protocol Version 1.1*, April 2006.
- [88] ISO/IEC. *ISO/IEC 13818-1:2000, Information Technology - Generic Coding of Moving Pictures and Associated Audio Information: Systems*, 2000, 1995.
- [89] ISO/IEC. *ISO/IEC 15444-1:2000, The JPEG 2000 standard, Part 1: Core coding system*, December 2000.
- [90] ISO/IEC. *ISO/IEC 15938-2:2002, Information Technology - Multimedia Content Description Interface - Part 2: Description Definition Language (DDL)*, April 2002.
- [91] ISO/IEC. *ISO/IEC 15938-3:2002, Information Technology - Multimedia Content Description Interface - Part 3: Visual*, Jun 2002.
- [92] ISO/IEC. *ISO/IEC 15938-4:2002, Information Technology - Multimedia Content Description Interface - Part 4: Audio*, Jul 2002.
- [93] ISO/IEC. *ISO/IEC 15938-5:2003, Information Technology - Multimedia Content Description Interface - Part 5: Multimedia Description Schemes (MDS)*, May 2003.

- [94] ISO/IEC. *MPEG-21 part-1: ISO/IEC TR 21000-1:2004, Information Technology - Multimedia Framework - part 1: Vision, Technologies and Strategy*, November 2004.
- [95] ISO/IEC. *MPEG-21 part-10: ISO/IEC 21000-2, Information Technology - Multimedia Framework - part 10: Digital Item Processing*, October 2004.
- [96] ISO/IEC. *MPEG-21 part-2: ISO/IEC 21000-2, Information Technology - Multimedia Framework - part 2: Digital Item Declaration Language*, October 2004.
- [97] ISO/IEC. *MPEG-21 part-3: ISO/IEC 21000-3, Information Technology - Multimedia Framework - part 2: Digital Item Identification*, November 2004.
- [98] ISO/IEC. *MPEG-21 part-7: ISO/IEC 21000-2, Information Technology - Multimedia Framework - part 1: Digital Item Adaptation*, October 2004.
- [99] ISO/IEC. *ISO/IEC 15938-10:2005, Information Technology - Multimedia Content Description Interface - Part 10: Schema Definition, Version 1*, April 2005.
- [100] ISO/IEC. *MPEG-21 part-7: ISO/IEC 21000-7:2004/Amd 1:2006, Information Technology - Multimedia Framework - Part 7: Digital Item Adaptation, Amendment 1: DIA Conversions and Permissions*, March 2006.
- [101] ITU-T and ISO/IEC JTC 1. *ISO/IEC 1449610:2003 Coding of Audiovisual Objects Part 10: Advanced Video Coding (AVC) and ITU-T Recommendation H.264: Advanced video coding for generic audiovisual services*, 2003.
- [102] ITU-T and ISO/IEC. *ITU-T Recommendation H.262 (ISO/IEC 13818-2), "Information Technology - Generic Coding of moving pictures and associated audio: part 2, video*, April 1995.

- [103] James Rumbaugh Ivar Jacobson, Grady Booch. *The Unified Modeling Language Reference Manual, (2nd Edition)*. The Addison-Wesley Object Technology Series, 2005.
- [104] T. Balsells J. Zamora J. Mata, S. Sallent and A. Kolt. Statistical models for MPEG video standard. In *Proceedings of the IEE EUSIPCO'94 conference*, volume 1, pages 624–627, September 1994.
- [105] Ivar Jacobson. *Object-Oriented Software Engineering: A Use Case Driven Approach*. Addison-Wesley, 1992.
- [106] Seiie Jang. *Implementation of Context-Aware Application Model in Smart Environments*. PhD thesis, Gwangju Institute of Science and Technology, 2005. Available on-line at the url [http : //mase.itc.nagoya – u.ac.jp/ jang/paper/Dissertation\(sjang\)eng.pdf](http://mase.itc.nagoya-u.ac.jp/jang/paper/Dissertation(sjang)eng.pdf).
- [107] Paul W. Jones. Efficient jpeg 2000 vbr compression with true constant perceived quality. *SMPTE Motion Imaging Journal*, July 2007.
- [108] Peter List Detlev Marpe Matthias Narroschke Fernando Pereira Thomas Stockhammer Jrn Ostermann, Jan Bormans and Thomas Wedi. Video coding with h.264/avc: Tools, performance, and complexity. *IEEE Circuits and Systems Magazine*, pages 7–28, 2004.
- [109] Peter Rigole Julien Pauty, Davy Preuveneers and Yolande Berbers. Research challenges in mobile and context-aware service development. In *Proceedings of Workshop on Future Research Challenges for Software and Services (FRCSS 2006)*, April 2006.
- [110] Radu Jurca and Boi Faltings. Reputation-based service level agreements for

- web services. *Service Oriented Computing, Lecture Notes in Computer Science*, 3826:396–409, 2005.
- [111] Anasawee Kaenampornpan and Eamonn O'Neill. Modelling context: An activity theory approach. *Lecture Notes in Computer Science*, 3295:367–374, October 2004. Also available as [http : //citeseer.ist.psu.edu/731314.html](http://citeseer.ist.psu.edu/731314.html).
- [112] J. G. K.E., Kupper and P.J. Murray. Eai and web services - a simple guide, 2001. Available as documentation of CGI and Cape Clear Software through the url.
- [113] E.Papaioannou D.Vergados K.Karpouzis, I.Maglogiannis and A.Rouskas. Integrating heterogeneous multimedia content for personalized interactive entertainment using mpeg-21. *IEEE Journal of Multimedia Communications*, 2006.
- [114] Marwan Krunz and Herman Hughes. A traffic model for mpeg-coded vbr streams. In *Proceedings of the ACM SIGMETRICS'95 conference*, pages 47–55, March 1995.
- [115] Marwan Krunz and Armand Makowski. Modeling video traffic using m/g/infinity input processes: A compromise between markovian and lrd models. *IEEE Journal on Selected Areas in Communications (JSAC)*, 16(5):733–748, June 1998.
- [116] Weiping Li. Overview of fine granularity scalability in mpeg-4 video standard. *IEEE Trans. on Circuits and Systems for Video Technology*, 11(3):301–317, March 2001.
- [117] J. F. Arnold M. R. Frater and P. Tan. Joint source/channel coding of statistically multiplexed real-time services on packet networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 4(6):521–526, December 1994.

- [118] H. Castro C. Alberti C. Calistru M. T. Andrade, P. Carvalho and P. de Cuetos. A unified data model and system support for the context-aware access to multimedia content. In *Proceedings of the WMSRM'07*, Aachen, Germany, March 2007.
- [119] H. Castro P. Carvalho B. Feiten M. T. Andrade, P. Bretillon. Context-aware content adaptation: a systems approach. In *Proceedings of the Eumob 2006 Workshop, as part of the MobiCom 2006 conference*, Sardinia, Italy, September 2006.
- [120] P. Carvalho H. Castro L. Ciobanu B. Feiten M. T. Andrade, P. Souto. Dynamic service management in heterogeneous environments using mpeg-21 dia for multimedia content adaptation. In *Proceedings of the Next Generation Networking Middleware'06*, Coimbra, Portugal, May 2006.
- [121] P. M. Carvalho B. Feiten M. T. Andrade, P. F. Souto and I. Wolf. Enhancing the access to networked multimedia content. In *Proceedings of the 7th International Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS 2006)*, Seoul, South Korea, 19-21 April, April 2006.
- [122] P. Carvaho M. T. Andrade, G. Ciobanu. A browser application for the consumption of mpeg-21 digital items. In *Proceedings of the*, April 2007.
- [123] L. F. Pires M. Van Sinderen P. D. Costa and D. Rios. Services platforms for context-aware applications. 2004.
- [124] A. Bilgin M. P. Boliek M. W. Marcellin, J. Gormish. An overview of jpeg-2000. In *Proceedings of IEEE Data Compression Conference*, pages 523–541, 2000.
- [125] Jrgen Thelin Martin Erzberger. Jms and web services. Technical report, Softwired AG, Sept 2002.

- [126] Ron Sass Marwan Krunz and Herman D. Hughes. Statistical characteristics and multiplexing of MPEG streams. In *Proceedings of the INFOCOM'95*, pages 455–462, 1995.
- [127] Deborah L. McGuinness and Frank van Harmelen. *Recommendation W3C: Web Ontology Language Overview*, February 2004. Available on-line at [http : //www.w3.org/TR/2004/REC – owl – features – 20040210/](http://www.w3.org/TR/2004/REC-owl-features-20040210/).
- [128] Chris Welty Michael K. Smith and Deborah McGuinness. OWL Web Ontology Language guide. W3C Recommendation, 10 February 2004. Available at the url [http : //www.w3.org/TR/owl – guide/](http://www.w3.org/TR/owl-guide/).
- [129] MIT. Overview of the oxygen project, 2005. Web site in the url [http :](http://oxygen.csail.mit.edu/Overview.html) [//oxygen.csail.mit.edu/Overview.html](http://oxygen.csail.mit.edu/Overview.html).
- [130] M.Keidi and A.Kemper. Towards context-aware adaptable web services. In ACM, editor, *Proceedings of the conference WWW2004*, 2004.
- [131] Austria Multimedia Communication Group at the university of Klagenfurt. The multimedia communication web site, 2006. Web site in the url [http :](http://www.uni-klu.ac.at/tewi/inf/itec/mmc/index.html) [//www.uni – klu.ac.at/tewi/inf/itec/mmc/index.html](http://www.uni-klu.ac.at/tewi/inf/itec/mmc/index.html).
- [132] M. T. Andrade P. Cuetos S. Battista N. Forel, C. Alberti and A. Benjelloun-touim. New maf for consideration: Multimedia streaming with qos, (msqos). Contribution to standardisation M13167, ISO/IEC JTC1/SC29/WG11, MPEG2006/M13167, March 2006.
- [133] P. Nasiopoulos and R. Kreidieh Ward. Effective multi-program broadcasting of pre-recorded video using vbr mpeg-2 coding. *IEEE Transactions on Broadcasting*, (3), September 2002.

- [134] Sun Developer Network. *Remote Method Invocation Home Page*, 2007. Home page in the url *http* : *//java.sun.com/javase/technologies/core/basic/rmi/index.jsp*.
- [135] Yves Lafon Nilo Mitra. *W3C Recommendation: SOAP Version 1.2 Part 0: Primer (Second Edition)*, April 2007. Available at *http* : *//www.w3.org/TR/2007/REC-soap12-part0-20070427/*.
- [136] Inc. Object Management Group. *Common Object Request Broker Architecture: Core Specification, V3.0.3*, March 2004.
- [137] Inc. Object Management Group. *Internet Inter-ORB Protocol (IIOP)*, March 2004.
- [138] The Library of Congress. the metadata encoding and transmission standard, the official web site, 2007. Home page at *http* : *//www.loc.gov/standards/mets/*.
- [139] Object Management Group (OMG). The home page of the omg web site, 2007. Available at the url *http* : *//www.uml.org/*.
- [140] Object Management Group (OMG). The unified modelling language, uml resource page, 2007. Information available in the url *http* : *//www.uml.org/uml/*.
- [141] Open Management Group (OMG). *CORBA, XML and XMI Resource Page*, 2007. Home page in the url *http* : *//www.omg.org/technology/xml/index.htm*.
- [142] P. Goddi P. Debaty and A. Vorbau. Integrating the physical world with the

- web to enable context-enhanced services. Technical Report HPL-2003-192, HP, 2003.
- [143] P. Gautier P. Kesteren and V. Ruol. Improved statistical multiplexing performance through the application of digital video impairment. *SMPTE Journal*, May 2002.
- [144] Luca Di Gaspero Stefano Mizzaro Ivan Scagnetto Andrea Selva Luca Vassena Paolo Coppola, Vincenzo Della Mea and Paolo Zandegiacomo Rizio. Information filtering and retrieving of context-aware applications within the mobe framework. In *Proceedings of the International Workshop on Context-Based Information Retrieval (CIR-05), Paris, France, 2005*.
- [145] Peter Patel-Schneider and Ian Horrocks. OWL 1.1 Web Ontology Language overview. W3C Member Submission, 19 December 2006. Available at the url <http://www.w3.org/Submission/owl11-overview/>.
- [146] J. Perez and J. Berger. Expected posteriori prior distributions for model selection, March 2000.
- [147] Adrian Petcu. Frodo: A framework for open/distributed constraint optimization. Technical Report 2006/001, EPFL, 2006. Also available as <http://liawww.epfl.ch/frodo/>.
- [148] Adrian Petcu. Recent advances in dynamic, distributed constraint optimization. Technical Report 2006/006, Swiss Federal Institute of Technology, EPFL, 2006. Also available as <http://liawww.epfl.ch/frodo/>.
- [149] Patrick Hayes Peter F. Patel-Schneider and Ian Horrocks. OWL Web Ontology Language semantics and abstract syntax. W3C Recommendation, 10 February 2004. Available at the url <http://www.w3.org/TR/owl-semantics/>.

- [150] Jonathan B. Postel. *RFC 821: Simple Mail Transfer Protocol - SMTP*, 1982. Available at the IETF Web site in the url *http : //www.ietf.org/rfc/rfc821.txt*.
- [151] S. James Press. *Subjective and Objective Bayesian Statistics, 2nd edition*. Wiley Series in Probability and Statistics. 2nd edition, 2003.
- [152] Pro-MPEG. The pro-mpeg forum web site, 2007. Web site in the url *http : //www.pro - mpeg.org*.
- [153] Liang Cheng Qinag Wang. Aware ware: an adaptation middleware for heterogeneous environments. In *Proceedings of the IEEE International Conference on Communications*, volume 3, pages 1406–1410, June 2004.
- [154] J. Mogul H. Frystyk T. Berners-Lee R. Fielding, J. Gettys. *RFC 2068: Hypertext Transfer Protocol – HTTP/1.1*, 1997. Available at the IETF Web site in the url *http : //www.ietf.org/rfc/rfc2068.txt*.
- [155] C.-S. Li R. Mohan, J. Smith. Adapting multimedia internet content for universal access. *IEEE Transactions on Multimedia*, pages 104–114, March 1999.
- [156] J. Ruela L. Teixeira R. Storey, A. Alves and T. Andrade. The ATLANTIC news studio: Reference model and field trial. In *Proceedings of the European Conference on Multimedia Applications and Techniques (ECMAST'97)*, pages 67–78, Milan, Italy, May 1997. Available on-line at the url *http : //www.fe.up.pt/ mandrade/research.html*.
- [157] Markus Endler Ricardo da Rocha. Evolutionary and efficient context management in heterogeneous environments. In *Proceedings of the 3rd international workshop on Middleware for pervasive and ad-hoc computing*, volume 115, pages 1–7, Grenoble, France, 2005. ACM Press New York, NY, USA.

- [158] Dockhorn Costa P. Guizzardi G. Ferreira Pires L.-Pereira Filho J. van Sinderen M. Rios, D. Using ontologies for modeling context-aware services platforms. In *Workshop on Ontologies to Complement Software Architectures*, October 2003.
- [159] Arthur Ryman Sanjiva Weerawarana Roberto Chinnici, Jean-Jacques Moreau. *Web Services Description Language (WSDL) Version 2.0 Part 1: Core Language*, 2007. W3C Recommendation available in the url [http :
//www.w3.org/TR/2007/REC – wsdl20 – 20070626/](http://www.w3.org/TR/2007/REC-wsdl20-20070626/).
- [160] O. Rose. Statistical properties of MPEG video traffic and their impact on traffic modeling in ATM systems. Technical Report 101, University of Wuerzburg, Institute of Computer Science, February 1995.
- [161] James Rumbaugh. *OMT Insights: Perspective on Modeling from the Journal of Object-Oriented Programming*. SIGS Books, 1996.
- [162] V. Bannai S. Agrawal, C. Barry and L. Kazovsky. Characterization, modeling and multiplexing of real-time mpeg-ii video, 2002.
- [163] G. Pereira Filho S. Pokraev, P. D. Costa and M. Zuidweg. Context-aware services; state-of-the-art. Technical Report TI/RS/2003/137, Telematica Instituut, 2003.
- [164] P. Chou S. Yoshihama and D. Wong. Managing behaviour of intelligent environments. In *Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerComm 2003)*, Dalas, Fort Worth, Texas, USA, 2003.
- [165] Guido M. Schuster and Aggelos Katsaggelos. *Rate-Distortion Based Video*

Compression: Optimal Video Frame Compression and Object Boundary Encoding. Kluwer Academic Publishers, 1997.

- [166] Ming Dong Shiyong Lu and Farshad Fotouhi. The semantic web: opportunities and challenges for next-generation web applications. Information Research Report 4, Department of Computer Science Wayne State University, Detroit, MI 48202, July 2002. Also available as <http://www.informationr.net/ir/>.
- [167] SMPTE. *SMPTE 298M:1997: SMPTE Universal Labels*, 1997.
- [168] SMPTE. *SMPTE RP 205-2000 Application of Unique Material Identifiers in Production and Broadcast Environments*, 2000. This standard is an accompanying recommendation to the UMID standard, providing best practices when using UMID in TV broadcasting.
- [169] SMPTE. *SMPTE 366M-2001: Key Length Value encoding (KLV)*, 2001.
- [170] SMPTE. *SMPTE 377M, 2003: Material Exchange Format (MXF) - File Format Specification*, 2003.
- [171] SMPTE. *SMPTE 360M-2004: General Exchange Format (GXF)*, 2004.
- [172] SMPTE. *SMPTE 379M-2004 : Material Exchange Format (MXF) - TelevisionMXF Generic Container*, 2004.
- [173] SMPTE. *SMPTE Standard 330M-2004 for Television Unique Material Identifier (UMID)*, 2004.
- [174] SMPTE. *SMPTE 380M: Material Exchange Format (MXF) - Descriptive Metadata Scheme -1*, 2007. Specification under development.

- [175] SMPTE. *SMPTE RP 210.10-2007 SMPTE Metadata Dictionary Registry of Metadata Element Descriptions*, 2007. Also available as [http : //www.smptra.org/mdd/index.html](http://www.smptra.org/mdd/index.html).
- [176] SMPTE. The smpte web site, 2007. Web site in the url [http : //www.smpte.org](http://www.smpte.org).
- [177] R. E. Kurt Stirewalt and G. D. Abowd. Composition property analysis: a new strategy for model checking user-interface designs. Technical Report MSU-CSE-99-30, Department of Computer Science, Michigan State University, 1999.
- [178] Linnhoff-Popien C. Frank K. Strang, T. Cool: A context ontology language to enable contextual interoperability. In Dameure-I. Hagimont D. Stefani, J.B., editor, *Proceedings of 4th IFIP WG 6.1 International Conference on Distributed Applications and Interoperable Systems (DAIS2003)*, volume 2893 of *Lecture Notes in Computer Science (LNCS)*, page 236247, Paris, France, 2003. Springer Verlag.
- [179] A. Stuart and J. K. Ord. *Advance Theory of Statistics: Classical Inference and Relationship*, volume 2. 5th edition, 1991.
- [180] T Sullivan, G.J.; Wiegand. Rate-distortion optimization for video compression. *IEEE Signal Processing Magazine*, 15(6):74–90, Nov 1998.
- [181] Jun-Zhao Sun and Jaakko Sauvola. Towards a conceptual model for context-aware adaptive services. In *Proceedings of the 4th International Conference on Parallel and Distributed Computing, Applications and Technologies*, 2003.
- [182] Inc. Sun Microsystems. *Sun Developer Network Site - Enterprise JavaBeans Technology White Papers*, 2007. Web site at [http : //java.sun.com/javaee/overview/whitepapers/ejb.jsp](http://java.sun.com/javaee/overview/whitepapers/ejb.jsp).

- [183] Inc. Sun Microsystems. *Sun Developer Network Site - Java EE at a Glance*, 2007. Web site at <http://java.sun.com/javaee/>.
- [184] Inc. Sun Microsystems. *Sun Developer Network Site - Web Services Overview*, 2007. Web site at <http://java.sun.com/webservices/>.
- [185] H. Frystyk T. Berners-Lee, R. Fielding. *RFC 1945: Hypertext Transfer Protocol – HTTP/1.0*, 1996. Available at the IETF Web site in the url <http://www.ietf.org/rfc/rfc1945.txt>.
- [186] A. Celentano T. Chaari, F. Laforest. Adaptation in context-aware pervasive information systems: The secas project. *International Journal of Pervasive Computing and Communications*, June 2006.
- [187] K. Luyten T. Clerckx and K. Coninx. Generating context-sensitive multiple device interfaces from design. 2004.
- [188] H. K. Pung T. Gu and D. Q. Zhang. A bayesian approach for dealing with uncertain contexts. In G. Kotsis, editor, *Proceedings of the 2nd International Conference on Pervasive Computing*. Austrian Computer Society, 2004.
- [189] G. Bjntegaard T. Wiegand, G. Sullivan and A. Luthra. Overview of the h.264/avc video coding standard. *IEEE Trans. on Circuits and Systems for Video Technology*, 13(7):560–576, July 2003.
- [190] David S. Taubman and Michael W. Marcellin. *JPEG 2000: Image Compression Fundamentals, Standards and Practice*. Kluwer International Series in Engineering and Computer Science, 2004.
- [191] C. Timmerer and H. Hellwagner. Interoperable adaptive multimedia communication. *IEEE Multimedia Magazine*, 12(1):74–79, Jan-Mar 2005.

- [192] Berkeley University. Context toolkit, 2001. Also available as *http : //www.cs.berkeley.edu/ dey/context.html*.
- [193] Carnegie Mellon University. *Remote Procedure Call - Technology Roadmap*, 2007. Available on-line at the url *http : //www.sei.cmu.edu/str/descriptions/rpc.html*.
- [194] Michael Uschold, Peter Clark, Fred Dickey, Casey Fung, Sonia Smith, Stephen Uczekaj Michael Wilke, Sean Bechhofer, and Ian Horrocks. A semantic infosphere. In Dieter Fensel, Katia Sycara, and John Mylopoulos, editors, *Proc. of the 2nd International Semantic Web Conference (ISWC 2003)*, Lecture Notes in Computer Science, pages 882–896. Springer, 2003. Presentation available from the url *http : //www.cs.man.ac.uk/ horrocks/Slides/ISWC – Presentation – SemanticFiltering.pdf*.
- [195] A. Vetro. Mpeg-21 digital item adaptation: Enabling universal multimedia access. *IEEE Multimedia*, 11(1):84–87, Jan-March 2004.
- [196] R. Virk. An architecture for content syndication, routing and distributed publishing, 2001. Available at *http : //www.cambridgedocs.com*.
- [197] W3C. *Date and Time Formats (based on ISO 8601)*, W3C Technical Note, Sept 1997. Available on-line at *http : //www.w3.org/TR/NOTE – datetime*.
- [198] W3C. *Web Services specifications*, 2002. Home page of W3C Web Services in the url *http : //www.w3.org/2002/ws/*.
- [199] W3C. *Recommendation W3C: Composite Capability / Preference Profiles (CC/PP): Structure and vocabularies 1.0*, Jan 2004.

- [200] W3C. *Recommendation W3C: Resource Description Framework (RDF): Concepts and abstract syntax*, Feb 2004.
- [201] W3C. *Ubiquitous Web Domain*, 2007. Also available as [http : //www.w3.org/UbiWeb](http://www.w3.org/UbiWeb).
- [202] Mark Weiser. The computer for the 21st century. *Scientific American*, 3(265):94–95, 1991.
- [203] N. Wells. Transparent concatenation of MPEG compression. *EBU Technical Review*, 1998.
- [204] Robin Wendler. Metadata in the library. *Harvard University Library Notes*, (1286):4–5, Jul-Aug 1999. Also available on-line at [http : //hul.harvard.edu/publications/hul_notes_pdfs/HULN₁286.pdf](http://hul.harvard.edu/publications/hul_notes_pdfs/HULN1286.pdf).
- [205] Tim Berners-Lee with Mark Fischetti. *Weaving the Web : The Original Design and Ultimate Destiny of the World Wide Web by its Inventor*. Harper Collins Publishers, San Francisco, 1st edition, 1999.
- [206] W.Lum and F.Lau. Context-aware decision engine for content adaptation. *IEEE Journal of Pervasive Computing*, 27(2):97–111, May 2002.
- [207] W3C World Wide Web Consortium. Semantic web technologies, owl web ontology language overview, w3c recommendation, 2004. Available on-line [http : //www.w3c.org/TR/2004/REC – owl – features – 20040210/](http://www.w3c.org/TR/2004/REC-owl-features-20040210/).
- [208] W3C World Wide Web Consortium. Semantic web technologies, w3c web site, 2007. Home page in the url [http : //www.w3c.org/](http://www.w3c.org/).
- [209] R. Chen X. Wang and J. Liu. Monte carlo bayesian signal processing for wireless communications. *Journal of VLSI Signal Processing*, (30):89–105, 2002.

- [210] Xiaokang Yang and Nam Ling. Statistical multiplexing based on MPEG-4 fine granularity scalability coding. *The Journal of VLSI Signal Processing*, 42(1):69–77, January 2006.
- [211] Philip Tan Eik Yeow. Context mediation: Ontology modeling using web ontology language (owl). Working Paper CISL 2004-11, Massachusetts Institute of Technology, MIT, Composite Information Systems Laboratory (CISL), Cambridge, MA 02142, Nov 2004.
- [212] Sang-Jo Yoo. Efficient traffic prediction scheme for real-time vbr mpeg video transmission over high-speed networks. *IEEE Transactions on Broadcasting*, (1), March 2002.
- [213] Masahiro Shibata Makoto Yamamoto Atsushi Narumi Yutaka Kaneko, Shinya Takeuchi and Shunji Sunasaki. A program production system using id and file-data over ip networks. *SMPTE Motion Imaging Journal*, April 2005.

Appendix A

Related work previously developed

A.1 Statistical Multiplexing studies

A.1.1 Brief Introduction to the work performed

In section 3, devoted to the description of the state-of-the-art, we saw that conventional transmission channels used in digital TV (dTV) broadcasting, are fixed-bandwidth channels and that, usually, multiplexers operate in a deterministic way. Nonetheless, we have also described the work conducted by researchers towards the development of statistical multiplexing algorithms, both in terms of the probabilistic modeling studies as well as multiplexing approaches. However, very little references can be found in the literature regarding the actual use of statistical multiplexing in real dTV services and, consequently, on the actual multiplexing gains that can be obtained.

Usually, the goal of broadcasters while using statistical multiplexers, is to

maximise one of the the following relations:

- ”*quality versus efficiency*”, which means ”more quality at the same transmission price”;
- ”*Quantity versus efficiency*”, translating into ”more programmes at the same transmission price”;

or even a mixture of the two. In the majority of the cases, the goal is to obtain the best compromise for the first relation. In many cases, they also seek to obtain a combination of the two: statistical multiplexing is used to save bandwidth, part of which can be re-used to prevent loss of quality in periods of high-activity in any of the existing programmes; the other part can be used to include one extra source, thus eventually increasing the broadcaster’s revenues, or even be relinquished, thus diminishing the transmission costs.

As explained in section 3, the traditional *deterministic multiplexing* approach does not allow to obtain a good compromise for the first relation, and is inefficient, in terms of the quality of programmes, when trying to maximise the second one.

On the other hand, *statistical multiplexing* schemes can potentially achieve good compromises in all three situations.

Our work concerning statistical multiplexing differed from previous work in both the approach and goals.

As referred above, usual goals of broadcasters, when using statistical multiplexing, include the possibility of enlarging the programming choice they offer to their customers. However, this is usually done by dimensioning their multiplex, a-priori, with $N+m$ sources, knowing that their stat mux will be able to save B_s bandwidth, enough to include the m extra sources. The new multiplex with $N'=(N+m)$ sources, will

then behave as a normal multiplex. This means that, although part of the bandwidth will in fact be dynamically distributed across the N' sources, the inclusion of new sources is not dynamic. It needs to be decided and fixed, a-priori, or at least, at some specific time.

Our goal, which is also appropriate for Internet-based applications, is to obtain a more flexible framework, where the decision of including the additional m sources, or any other number, including zero, can be made on the fly.

Our approach also differs from that of previous work on statistical multiplexing. Whereas the majority of work has been concentrated on the so-called "*closed-loop feedback*" mode (explained in section 3), our work targets the "*open-loop*" approach. Moreover, a novel use of the Bayes theory is proposed, together with a knowledge based approach, by which a data base with probabilistic models that translate the behaviour of classes of content, is used to infer the occurrence of periods of low activity in each source (corresponding to periods of low bit rate).

Statistical multiplexing allows to dynamically change the share of bandwidth assigned to each source, according to their own instantaneous needs. This unique feature can be used to simultaneously, or alternatively, increase the quality and the quantity of video streams transmitted in shared media such as digital television broadcasting and the Internet.

Accordingly, the following new perspectives can be expected to be opened to the television broadcasting industry [9], [7]:

- the effective transmission bandwidth of existing services in the multiplexer is reduced, matching more closely the actual needs of the source to achieve a fairly constant quality;
- channel capacity is thus made available to accommodate new services;

- in a re-distribution point, such as a regional opt-out center, a programme may be seamlessly substituted by another one (for instance, regional programming replacing the national one), even with un-matching bit rates. The re-distribution of bandwidth is automatically performed.
- when a certain channel finishes its programming, it can be simply withdrawn from the multiplex. Its share of bandwidth is transparently assigned to a new source or simply distributed among the other already existing sources, thus improving their quality.
- bandwidth of a set of services sharing the same transmission channel may be periodically (for example, at the beginning of a new programme) or continuously redistributed, according to the programme's content.

Likewise, in Internet media applications (e.g., media file downloading, VoD or NVoD, streaming of A/V clips, etc), foreseen benefits include:

- efficient use of shared resources, which benefits all users;
- possibility of increasing the average quality of the more demanding sources of content, without jeopardising the quality of other streams;
- private content creators will have increased chances of distributing their content and reaching larger audiences.

Statistical multiplexing of open-loop (VBR) encoded video sources, allows efficient allocation of bandwidth in a multi-channel digital TV service. It thus provides the means for broadcasters to offer increased quality and to be able to efficiently manage, their complete network. It also frees channel capacity allowing the introduction of new additional services such as data broadcasting and IP-based data services. This is essential to the introduction of improved and richer programming including

interactive television.

However, statistical multiplexing as opposed to deterministic, has an increased probability of losses. Although independent video sources are highly uncorrelated, there is always the probability, however small it may be, of multiple sources to simultaneously transmit at their peak rates. When that happens, the multiplexer buffer may overflow, causing losses. Simulation studies are therefore necessary to help quantifying the amount of losses and the actual statistical multiplexing gain.

To run these simulations, models describing the bit rate behavior of VBR video sources are needed to provide input stimulus to the system. To this aim, it would be extremely interesting to conduct a detailed characterisation of representative sets of compressed video sources, leading to the creation of a data base of probabilistic models. This would provide a knowledge base, translating the behavior of real-world television programmes or Internet video sources, which could be used to formulate and implement efficient statistical multiplexing algorithms. Better results could even be achieved, if that statistical characterization would be complemented with a content-based classification (with classes matching representative types of content, such as "news", "sports", "action movie" or "video-clip") and crossing the two types of information to obtain correlations between the two.

A.1.2 Statistical Framework

Characterisation and Classification of video sources

In our work we have explored the use of the Bayes theory and proposed a novel method of multiplex performance and bandwidth optimization. The approach is to design a multiplexer that uses probabilistic models, supported on a Bayesian framework, to predict the bit rate behaviour of video sources by using features ex-

tracted, or declared, from the sources.

Features used, are those that characterize the nature of the video, such as type of content (using amount of mouvement, detail and activity), the genre of programme, characteristics of the compression algorithm, duration of the video, etc.

These features are then used to obtain from a database, the most appropriate model for the source, or to build a new one, thus allowing to predict the amount of resources needed to be allocated for the transmission of that source.

This approach is suitable to be applied to a scenario of combined transmission of multiple audio visual sources such as the case of a digital television broadcasting service, but also to that of Internet A/V streaming and download applications.

The number of bits necessary to encode a given video source at a certain quality level, is highly dependable on the type of content [212], [133], [143] [126], [162]. Different types of content present different degrees of difficulty to encode and this can vary from video source to video source, from scene to scene in a given programme or even within a scene or shot. It is thus expected that different scenes will exhibit different bit rate requirements to achieve the same quality and that different sources overall will also require different amount of bits.

This means that it is possible to anticipate different behaviours of video sources regarding their bit rate requirements. As such, a classification of sources by type of content, can help to identify models that may describe reasonably well, the bit rate behaviour of sources.

In this context, our framework for obtaining a characterisation of video sources bases itself on subjective and objective metrics. We use the definition of *criticality*, an objective metric of the number of bits per pixel at the output of the VBR encoder. The *criticality* is directly related to the objective quality of the source, the *SNR* (Signal to Noise Ratio). This metric indicates the number of bits necessary to obtain a certain

objective quality. It depends only on the content itself and therefore on the degree of difficulty to encode that content.

Before actually performing the categorisation of sources, we have identified a number of relevant categories of TV programmes likely to exhibit different types of content and activity. These categories were based on the programmes classification provided by DVB-SI [50], to identify and describe the genre of TV programme being transmitted. Table A.1 gives the list of the categories we have selected. Those were thought to be representative of usual TV programming while possessing sufficiently different statistical characteristics.

We propose to conduct the analysis of video sources, as a two-step process, the first of which consists on an hybrid procedure, combining subjective and objective measures as follows:

- visualisation of the video source and their classification into one classe according to the amount of mouvement and detail. This procedure should be assisted of simple objective measures of the activity between images - *intra scene* - and between scenes *inter scene*.

This should lead to the obtention of a bi-dimensional complexity matrix, with one dimension being the measure of mouvement and the other the measure of detail. Items that exhibit small amount of mouvement and detail will be placed in the top left corner of the matrix, whereas those with high levels will be assigned to the lower right corner.

- measurement of the *criticality* of each group of sources placed in each element of the matrix. To accomplish this step, it is necessary to previously encode each source in VBR mode, constant quality. A measure of the *criticality* is performed and then, its mean value and standard deviation are calculated.

Table A.1. Categories of TV sources

Movies	adventure, western, war comedy romance thriller science fiction
News	general weather interviews documentary
Shows	talk shows contests variety
Sports	football, basketball tennis athletics motor sports
Arts, culture, social	rock/pop video clips opera, ballet musical fashion
Education, Science	nature technology medicine
Leisure	tourism, travel
Publicity	

Once this hybrid analysis is performed, a statistical analysis is conducted for each group of sources placed on the same matrix element. The proposed statistical analysis, is based on a Bayesian framework that will be explained in the next section. Essentially, the idea is to take measurements regarding the number of bits spent to encode the sources, on a GOP-basis (Group Of Pictures).

It would also be possible to add a third dimension to the bi-dimensional matrix generated in the first step of the analysis. The third dimension would be referred to the level of quality with which the sources had been encoded. This would lead to the definition of profiles, each of which would be characterised by a measure of *criticality* and a level of *quality*. In order to simplify the experiments, it is though possible to target the same level of quality for all sources, ignoring this third dimension.

A.1.3 Problem formulation within a Bayesian Framework

Knowing that source α has a certain number of characteristics, will it be possible to predict its behaviour in terms of deviation of its bit rate from an expected target bit rate?

Our objective is to be able to develop statistical multiplexing techniques capable of anticipating the amount of unused bandwidth within the multiplex, in respect to a certain mean value. Regarding an expected mean bit rate thought as necessary to achieve a target quality, knowing the characteristics of the sources, the goal is to be able to identify the probability that the bit rate be under that value - we call to this situation the *occurrence of a valley*. This goal can be pursued through the development of a statistical model incorporating prior knowledge obtained from sets of training data and using efficient numerical model selection techniques such as the method *Markov Chain Monte Carlo* (MCMC) [4], [151], [209].

The statistical model is based on the Bayes theory, applying statistical inferencial techniques aiming at obtaining the most likely model M that describes that random variable given a set of training data D .

In Bayesian analysis [4], it is necessary to establish prior beliefs in the form of probability distributions. The statistical inference is obtained in the form of a posteriori distribution, incorporating both the prior knowledge and the observations, capable of determining the state of the random variable. The model selection is summarised by the posteriori distribution.

Formally this can be seen as the problem of maximisation of a probability densiy function:

$$\begin{aligned}
 P(M|D) &= \frac{P(D|M) \times P(M)}{P(D)} \Rightarrow \\
 \max_M P(M|D) &= \max_M \left[\frac{P(D|M) \times P(M)}{P(D)} \right] \Rightarrow \\
 \max_M P(M|D) &= \max_M \left[\log \frac{P(D|M) \times P(M)}{P(D)} \right] \Leftrightarrow \\
 \max_M P(M|D) &= \\
 &= \max_M [\log P(D|M) + \log P(M) - \log P(D)]
 \end{aligned}$$

The probabilistic model includes a random variable and an unknown parameter, for which extensive sets of experimental data need to be collected. The random variable represents *valleys* in the GOP bit rate. At instant t_i it can therefore assume only one out of two values: 1, indicating the occurrence of a valley, or 0 indicating that a instant t_i a valley was not detect.

- random variable V - occurrence of a valley - with two possible states;

- unknown parameter Dev_{GOP} - the deviation of the GOP bit rate relative to the expected mean.

The statistical characterization of VBR bit streams provides the necessary ingredients to address the problem of predicting the bit rate behaviour as an inferential process.

It provides the sample data, facilitates the selection of appropriate family of probability distributions and allows the formulation of inductive statements relative to hypothesis states and the sample data. These statements are in reality posteriori probabilities formulated under the Bayesian approach.

Prior knowledge about the bit rate behaviour, is included in the statistical model in the form of a prior probability distribution $p(\theta)$ in light of the sample data. Variable V indicates the occurrence or not of a valley at a certain time t_1 . Only two hypotheses can be evaluated at instant t_1 - either there is the beginning of a valley ($V_{t_1, \Delta_{GOP}} = 1$) or not ($V_{t_1, \Delta_{GOP}} = 0$). The two possible states are inferred through the statistics extracted from the bit stream. Dev_{GOP} is the bit rate deviation from the expected mean GOP bit rate. It takes a complete GOP period (Δ_{GOP}) to decide whether there is a valley or not. Given a prior distribution for Dev_{GOP} , $p(Dev_{GOP})$, and the likelihood function $\mathcal{L}(V|Dev_{GOP})$, the posteriori distribution for Dev_{GOP} according to Bayes rule is given by:

$$P(Dev_{GOP}|V) \propto p(Dev_{GOP})\mathcal{L}(V|Dev_{GOP})$$

The probability that a valley occurs at instant t_2 within a period Δt after the occurrence t_1 of the last valley, and the probability that it does not occur, given the bit rate behaviour within Δt , are:

$$\mathbf{V}_{t_1, t_2} = \mathbf{1} : \quad (t_2 = t_1 + \Delta t, \quad \Delta t \neq 0)$$

$$P(V_{t_1, t_2} = 1 \mid V_{t_1} = 0, Dev_{\Delta t}) =$$

$$P(Dev_{\Delta t} \mid V_{t_1}=0, V_{t_1,t_2}=1) \times P(V_{t_1,t_2}=1 \mid V_{t_1}=0) = P(Dev_{\Delta t} \mid V_{t_1,t_2}=1) \times P(V_{t_1,t_2}=1 \mid V_{t_1}=0)$$

$$\mathbf{V}_{t_1,t_2} = \mathbf{0}:$$

$$P(V_{t_1,t_2}=0 \mid V_{t_1}=0, Dev_{\Delta t}) = P(Dev_{\Delta t} \mid V_{t_1}=0, V_{t_1,t_2}=0) \times P(V_{t_1,t_2}=0 \mid V_{t_1}=0) = P(Dev_{\Delta t} \mid V_{t_1,t_2}=0) \times P(V_{t_1,t_2}=0 \mid V_{t_1}=0)$$

(Note : two valleys can not occur contiguously.)

These formulas can be further processed and used to compute the *posteriori log – likelihood* ratio between the two cases. Using the probability distribution $p(\theta)$ for the period where bit rate valleys are not detected to compute $P(V=v)$, will allow to find the optimal solution to the problem of declaring the occurrence or not of a valley.

A.1.4 Some preliminary statistical results

Some sequences encoded in MPEG2 VBR with DVD quality have already been used to extract some statistics. The objective with those measures is to be able to formulate initial approaches to the problem of characterizing the occurrence of valleys in the mean GOP bit rate and thus inferring suitable probabilistic models.

Those sequences consisted of a set of 1193 GOPs of a movie presenting some variation in the type of content throughout their duration. It was also used some sequences from the film "Gladiator" and a real-estate publicity spot. Several histograms have been generated according to the proposed in the workplan. Both the Weibull and the Gama distributions were used as models having provided good fits for the spacing among valleys and occurrence of valleys. The exponential and the beta distributions also seemed to be possible candidates for the priors. More deep

study needs to be performed.

Regarding the amount of bandwidth expected to be gained by using dynamically multiplexed VBR sources, the results obtained with an aggregation of 4 sources are presented in figure ???. For approximately equivalent qualities, CBR encoding uses almost always considerably more bandwidth than VBR. With 4 sources it is possible to obtain gains up to 7 Mbit/s (*circa* 30%) with an average of around 2,9Mbit/s (*circa* 12%).

A.1.5 Further work

The developed work has enabled the formulation of an approach based on the Bayes theory and classification of video sources, to design a statistical multiplexer suitable to be used not only in dTV broadcasting, but also in Internet-based applications. The work concentrated on the definition of a framework to describe MPEG-2 VBR video sources through reference probabilistic models, and to allow to select the most adequate probabilistic models using a formal Bayes procedure.

Although this line of work had to be discontinued, for reasons already explained, we believe it would be interesting to continue this work, concentrating on the Internet application scenario. This domain is where exists more variability on both network resource conditions and content formats and requirements. This potentiates a wider use of a statistical multiplexer adopting the formulated principles.

Further work could therefore be conducted within the proposed guidelines, starting by analysing a large number of video sources typically found in Internet-based applications and then building the complexity matrix referred above. Model selection could then be performed for the identified classes through a formal Bayes

procedure. This would allow to create a knowledge base, holding information suitable to be used by a statistical multiplexer. To overcome the main obstacle to the implementation of the Bayes methodology, which is the high computational requirements, it would be desirable to make use of numerical methods such as the Markov Chain Monte Carlo simulation technique [209] to approximate samples from the joint posteriori distribution.

Another interesting direction to follow, would be to try to explore the combined use of this statistical framework with semantic descriptions of sources, probabilistic models and context (more precisely, the instantaneous bandwidth availability and capabilities of terminals).