eman ta zabal zazu

U.P.V. E.H.U.

Computer Science Faculty. Computer Languages and Systems Department

A Semantic Middleware to enhance current Multimedia Retrieval Systems with Content-based functionalities

, a dissertation submitted to the Department of Computer Languages and Systems of the Computer Science Faculty of the University of the Basque Country in partial fulfillment of the requirements for the degree of Doctor of Philosophy

by Gorka Marcos Ortego

This dissertation is supported by the following advisors
Dra. María Aranzazu Illarramendi Echabe and Dr. Julián Flórez Esnal

Donostia-San Sebastián,
2011

ISBN: 978-84-694-6824-1

Contents

List of Tables

List of Figures

Acknowledgment

Esta tesis ha sido posible por el soporte, la dedicación, el saber hacer y el tesón de mis dos directores de tesis , Arantza y Julián, y por la confianza y el apoyo que Vicomtech-IK4 ha depositado en mí.

En estos años, he ido tejiendo este trabajo en estrecha relación con muchas personas. Esta tesis ha sido sin duda posible gracias a ellos. En los proyectos que me han servido para validar este trabajo, he tenido la suerte de trabajar con más de 60 expertos de distintos ámbitos y países. Estoy especialmente agradecido a aquellos que generosamente han compartido su conocimiento y debo recordar especialmente al equipo de compañeros de ETB. En Vicomtech-IK4, también he estado muy bien acompañado. No habría podido realizar esta tesis sin la ayuda y generosidad de Ivanjou, Kevin, Tim y Jorge. Y nunca me podré olvidar del apoyo incondicional que en todo momento me han dado Igor, y con él, todos mis compañeros del departamento, con los azules a la vanguardia. Ahí va, de nuevo, un mila esker para todos vosotros. Y Petra, esto te incluye también a ti.

Regarding some key contributions that I received, I can't forget the kindness of the Professor Ray Larson and the generosity of the NTUA team granting me access to their reasoner. Phivos, thank you once again. I also want to thank Oliver (Fraunhofer-HHI) and Sergio (University of Brescia) for the time they kindly devoted to review this work.

Y los amigos de las cuadrillas de Donosti y Bilbao y "de" Vicomtech-IK4. Gracias por estar ahí y por haber compartido conmigo los progresos y desesperaciones.

A Iñigo, Maider, Laia y toda la familia de allá y de acá. En este camino me habéis apoyado, comprendido y ayudado en todo lo que habéis podido, sin preguntas ni condiciones. Soy feliz por seguir compartiendo el CAMINO con vosotros.

Aita, Ama... esto, como tantas cosas, lo empezamos juntos. Sin vosotros, no hubiera sido posible. ¡ Gracias !

Beizama, Febrero de 2011.

Summary

This work reviews the information retrieval theory and focuses on the revolution experimented in that field promoted by the digitalization and the widespread use of the multimedia information. After analyzing the trends and promising results in the main disciplines surrounding the content-based information retrieval field, this thesis proposes a reference model for Multimedia Information Retrieval that aims to contextualize the thesis contributions. According to this reference model, this work proposes an architecture for a component named "Semantic Middleware" that aims to centralize the main semantic services to be provided during the indexing, storage, search, retrieval and consume of the multimedia elements. This architecture has been designed from a pragmatic point of view, aiming to facilitate the enhancement of the current systems with content-based functionalities. The architecture proposal includes a set of key design criteria for a right deployment. In order to validate this thesis, two real complementary deployments have been performed and reported in this work.

Part I

MOTIVATION AND CONTEXT OF THE THESIS

1 Introduction

This chapters aims to describe the scope of this thesis, its motivation and context. The chapter ends with a section that aims to facilitate the reading of this document.

1.1 Scope of this thesis

Looking up in the IT business and computer industry dictionaries, is possible to find diverse definitions of the term “Middleware”. One extended definition is the one proposed by Kavanagh and Thite (2009):

> *general term for any programming that serves to “glue together” or mediate between two separate and often already existing programs. A common application of middleware is to allow programs written for access to a particular database to access other databases.*

In a coherent way with this statement, we define the term **“Semantic Middleware”** as that **piece of software that semantically glues together different existing programs that co-exist with a common target**. Specifically, this thesis work is related to the semantic tying among the different modules or components that are frequently part of complex multimedia information retrieval and management systems.

Therefore, this thesis, based on a diagnosis of the semantic needs of those systems, propose a generic architecture to define a middleware that fulfills those needs in a pragmatic, feasible and beneficial way from the programmatic point of view.

Along the following sections, we establish and define the problems and motivation behind this work. Once this has been clarified, we describe the structure of this thesis work at the end of the chapter. This structure has been defined to expose and clarify the different details about the technological context, the definition, implementation and validation of this work.

1.2 Problem identification and motivation of this thesis

In this section we firstly present the revolution experienced by the media industry in the last years. Then, we describe the impact of such revolution from the perspective of the scope of this thesis work.

1.2.1 New needs in the content creation and consumption industry

During the last decade there have been different phenomenons that have led to a deep and huge revolution in the way the content is created, managed and exploited. In the following, we highlight the most significative ones:

- First of all, the **digitalization** of the content. The content is not just what we can find inside a tape, book or disc that is stored in a specific shelf of an archive, but an entity per se. The disappearance of the physical part of the content has increased its protagonism. The expects and needs of the users have changed. In many situations, the user is not looking just for an identifier, reference number or a title (as in the traditional libraries) but for a content that contains a specific piece of information, sentence, image or piece of audio.

- Closely related to the previous item, the evolution of the information management and retrieval systems has led to the **appearance of a new generation of products**, such as the MAMs or "Media Asset Management". These products are not merely repositories of digital assets, but also aim to digitalize the whole workflow of the content creation, generation and exploitation. We would like to highlight three phenomenons linked to the digitalization of those process. First of all, the migration from tape-based archives to digital libraries accessible on the Intranet has changed the way the metadata (information about the content) is generated (Avilés et al., 2005). The metadata is not just generated in a specific point of the workflow and is related to many different aspects of the asset (e.g., legal, internal, technical). Secondly, and related to the previous one, the content is not just a unique entity, but a set of entities linked (e.g., the video, several audio tracks, the script, the metadata). It is only this set of entities the one that conform what was previously understood as content. And finally, and fulfilling the premonitions of Serb (1997), the roles of the people working in organizations that handle content, have significantly changed. For instance while in a broadcaster almost all the annotations were handled by the people working in the archive, nowadays, due to the presence of the MAM systems, the journalists have the main role in the generation of those metadata. This has an impact in the coherence and soundness of that information.

- The maturity achieved by the technologies for the storage and data network has significantly contributed to the mentioned digitalization and therefore in the accessability of the content. Nowadays the manufacturers of media asset management systems include high resolution video storage solutions that are feasible for small media producers.

- The explosion in the generation of content is not only due to a specific factor but to a set of factors: the globalization of the society, the democratization of the digital devices, the appearance of new communication platforms and the consequent increase of media companies, etc. All these factors have definitely contributed to the current situation, as the work of Pastra and Piperidis (2006) corroborates. There is a need of handling or controlling this digital content explosion. This is mainly due to the fact that the explosion has occurred in a relatively short period of time, and many organizations have not been able to either adapt the way they deal with the content, or modify the business model in such a way that the establishment of new ways of dealing with this content is a feasible task.

- The new communication channels (Internet, mobile networks), have also contributed to this new scenario, where the content is created, accessed and shared by users that were not active part of the content life-cycle. We are not only referring to the generation of content made by final users (i.e., prosuming) but also to the fact that the investment required to make the content accessible to the general public has decreased significantly. And the involvement of such public, usually leads to the generation of new content associated to that content.

This new context surrounding the generation and consumption of content and its metadata has implied and will imply in the next years deep changes in almost all the business processes of the media industry. This thesis focuses on a specific aspect of this revolution. We are concerned about the way the content should be managed in this context in order to take advantage of the semantic richness of the content itself. In this sense, this thesis is a contribution in between the information retrieval systems that have been adopted by the industry and the achievements of the scientific field "Multimedia Semantics". We cover this issue in the following subsection.

1.2.2 Context and contributions of this thesis

As we have stated, our contribution aims to support the media industry in order to increase the semantic exploitation of their content. In order to do this from a pragmatic perspective, this work is located in between the technology acquired by the industry during the digitalization and the achievements of the scientific field "Multimedia Semantics".

Regarding the systems that drive the media storage and retrieval in the industry, we highlight the following facts:

- Independently of the domain, the technology massively employed is the relational database together within search algorithms of different nature. This technology is mature, well established and, in fact, as we have previously stated, has been and is one of the main drivers of the digitalization process.

- Beside this, in those sectors with the highest amount of media content generated (e.g. entertainment industry), most of the systems are proprietary or customized solution (Datamonitor-Analysists, 2007; Multimedia-Research-Group, 2004).

- Regarding the employment of common structure for the modeling of the information, excepting some niche sectors (e.g., libraries), there is an important lack of homogeneity. Although there are multiple standards coming from different forums, most of the companies organize their information following their own internal criteria. This was one of the main conclusions of the professional Workshop of Annotations and Metadata models for Audiovisual/Multimedia hold in the context of the CHORUS forum (Metadata-Professional-workshop, 2007) and is also supported by the book of Cox et al. (2006).

- We have had the chance to know the systems of seven Spanish broadcasters (being five of them local broadcasters) and all the major content producers of the Basque Audiovisual Cluster [1]. Many of them have already faced this digitalization process, investing very important amounts of money, but in most of the cases they have just replicated in a digital way the organization schemas that they had in their analogue archives, without taking use of the opportunities of the digitalization. In some cases, they are already facing the upgrade and customization of those systems in order to include some preliminary semantic functionalities (e.g., automatic query expansion based on synonyms).

In this context, and coming from the search, retrieval and image analysis communities, a new scientific community has been devoted to the improvement of the multimedia content retrieval by employing content-based functionalities. This community, frequently tagged as "Multimedia Semantics", aims, according to the definition of Giorgos Stamou and Stefanos Kollias (Furht, 2006), to *deal with the*

[1] http://www.eikencluster.com/

question how to conceptually index, search and retrieve the digital multimedia content, which means how to extract and represent the semantics of the content of the multimedia raw data in a human and machine-understandable way.

As we present later, this community is providing very interesting and promising results aligned with that aim. They are bringing new means of extracting information out of the multimedia content. The correct storage of this information combined with new search techniques are presented as the basis for the scenarios of the future multimedia information retrieval.

In this context, our aim is not the contribution to the generation of those new systems in the long term, but the adaptation of the current retrieval and storage technologies in order to increase their performance by the gradually integration of the emerging content-based features. We are not proposing a revolutionary paradigm for multimedia retrieval but a **straightforward approach based on the deployment of a middleware** to enrich current Multimedia Information Retrieval (MIR) systems with successful semantic applications that benefit from the understanding of the multimedia asset. This middleware is a three layered semantic middleware that has been designed to provide semantic services needed by different content-based applications involved in conventional multimedia retrieval workflow. The main feature of **this middleware** is that it **centralizes the semantic knowledge and the provision of semantic services** in the system. Below, we summarize the main advantages of our proposal:

- *Outsourcing.* The middleware facilitates the integration into existing systems since the semantic services are outsourced from the retrieval engine(s).
- *Uniqueness.* The middleware avoids current semantic duplicities imposed by the employment of satellite applications (e.g., content-based recommendation, ontology-based clustering). This simplifies the work of knowledge engineers, since the upgrading of the knowledge representation of the domain is performed in a single place.
- *Semantic interoperability.* The middleware includes a semantic representation of the knowledge which is format-agnostic. In those cases where the middleware is working with components or information sources that employ different formats or languages, the architecture of the middleware provides simple mechanisms to perform the needed adaptations and carry out the upgrades derived from the evolution of each of the peers.

We have also contextualized this middleware **within a global multimedia reference model** and provided a **set of key design parameters** for its correct deployment. Finally, we have validated this middleware by the **implementation of two deployments in two real complementary scenarios** belonging to different industrial sectors.

1.3 How to read this thesis work

In the chapter following this introductory one (Chapter 2), we take care of the technological context of this thesis. In that chapter, we introduce the basis of the Multimedia Information Retrieval (MIR) theory. Once this is clarified, we cover three scientific and technological issues directly related to this thesis work: (i) the initiatives that have similar aims or similar approaches to this work, (ii) a review of the main multimedia metadata models and (iii) a summary of key contributions in the field of the content-based retrieval. With this chapter, the introductory part ends.

In the **second part of this thesis work** we deal with the contributions of this thesis, which are summarized in Table 1.1.

To acquire a complete understanding of this semantic middleware, its design and validation, the reader may turn to the Chapter 3, which covers the following issues:

- First of all, in order to contextualize this thesis, we propose **a reference model for multimedia information retrieval**. This can be found in Section 3.1.
- Once we have defined the context of this thesis, the middleware or main contribution of this thesis work, is described. Section 3.2 covers the **definition of each of the layers** of the proposed architecture for this middleware.
- With the aim to support a right deployment of this middleware in a real system Section 3.3 includes a list of **key criteria** to have in mind.

At the end of this part, Chapter 4 provides an overview of some other minor contributions in this thesis.

Table 1.1: Summary of contributions and information about their location in the report

CONTRIBUTION	LOCATION	PAGE
MIR Reference Model Proposal	Sec. 3.1	59
Semantic Middleware: definition	Sec. 3.2	63
Semantic Middleware: deployment design criteria	Sec. 3.2	63
USE CASE WIDE: deployment implementation	Ch. 5	79
USE CASE RUSHES: deployment implementation	Ch. 6	123
DMS-1 OWL ontology	Sec. 4.1	71
Contribution to JPSearch Standardization activity	Sec. 4.2	73
Summary of Publications	Sec. 7.3	157

In the third part of this thesis, we cover the validation of the proposed semantic middleware architecture. We include two real deployments of the middleware: WIDE (Chapter 5) and RUSHES (Chapter 6). For each of the use cases the following aspects are covered:

- **Contextualization:** Each semantic middleware deployment has been implemented within a global system. In order to contextualize the development of the semantic middleware, we include a description of the global system , the motivation that led to its development and its architecture.
- **Functionalities:** Once the context of the semantic middleware has been described, a description of the functionalities implemented by the middleware is included.
- **Mapping with the proposed middleware:** The middleware proposed in the second part of this thesis, has three lines: Browsing Line, Search Line and Storage Line (described in Figures 3.2 and 3.3)). Both scenarios cover the *Browsing Line* included in that reference architecture . Regarding the other two lines, the validation scenarios are complementary. On the one hand, the first deployment provides a wide range of services for the *Search Line* in order to cover different information sources. On the other hand the second scenario implements different services devoted to support the *Storage Line*, in order to provide advanced indexing mechanisms of the multimedia assets. Therefore, the combination of the scenarios provide a global overview of the provision of services for the three lines of the reference model for multimedia retrieval.
- **Identification of design criteria:** In a coherent view with the requirements identified in Section 3.3, we detail the decisions taken for each semantic middleware.
- **Implementation details:** We identify the main implementation details for each of the three layers that compose each semantic middleware.
- **Validation:** Finally we provide information about the validation of both semantic middleware.

The **fourth part** of this thesis work, covers the summary of the conclusions of this thesis, the main future work action lines started and our publications (Chapter 7).

At the end of this thesis, there are two parts including some annexes and the list of referenced bibliography.

2 Technological Context

This chapter aims to provide a focused review of the technological context of this thesis work. In the following section, we provide a general introduction to the Information Retrieval (IR) field followed by a description of the impact in the field promoted by the targeting of the multimedia assets. We also cover the significant role played by the Multimedia Analysis community in such revolution. Accordingly with this, we include a brief description of the current context, techniques and challenges of that community.

After this introductory section, we concentrate on three topics that are deeply related to the semantic middleware that we present. First of all we provide a description of the IR field. Then we include a section focused on the relevant contributions found in the literature that propose a system, architecture or middleware to promote and enable the usage of content-based functionalities in retrieval systems. Each contribution is reviewed and the main differences with respect to this thesis work are highlighted.

Furthermore, in order to understand the important efforts that the scientific and industrial communities are spending in facilitating the management, sharing and retrieval of multimedia assets, a review of the main metadata models is performed. From our point of view, due to the fact that the model is the main element to provide the semantic services targeted, an understanding of the current context of such models is a key criteria for a successful enhancement of a MIR system with content-based functionalities.

Finally, in order to grasp the current status of the semantic-aware techniques being developed by the scientific community regarding different stages of the MIR process, a summary of several main contributions and key surveys in the content-based multimedia information retrieval field is provided.

Figure 2.1 supports graphically the structure of this chapter. In this Figure we show how the deployment of a semantic middleware (Section 2.2), the employment of multimedia metadata models (Section 2.3) and the integration of content-based techniques (Section 2.4) are key contributions to current MIR systems (Section 2.1) with content-based functionalities.

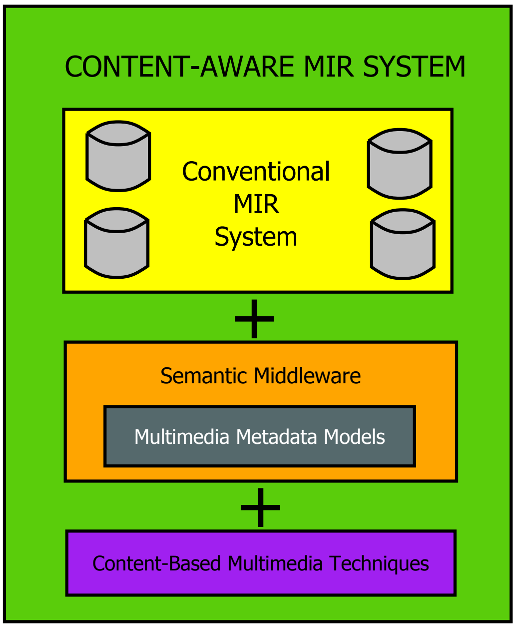


Figure 2.1: Graphical summary of the Technological Context Chapter

2.1 Multimedia in the information retrieval theory

Information Retrieval (IR) can be understood as the field related to the storage, organization, and searching of collections of data. But behind this simple definition, there is a little confusion. As Styltsvig (2006), based on the work of Lancaster (1968); van Rijsbergen (1979), remembers, Information Retrieval systems do not actually retrieve information, but rather documents from which the information can be obtained if they are read and understood. To be more precise, that which is being retrieved is the system's internal description of the documents, thus as the process of fetching the documents being represented is a separate process. Despite this loose definition, information retrieval is the term commonly used to refer to this kind of process, and thus, whenever we used the term **Information Retrieval**, it refers to this text-document-description retrieval definition. Moreover, whenever the type of document that is retrieved is not only a text-document but any kind of digital asset, we employ the term **Multimedia Information Retrieval**.

In the following subsections, we first try to clarify the distinction between Information Retrieval versus Data Retrieval. Secondly, we present a short summary of the history of IR. After this, we provide an overview of the IR models. Finally, we include a deeper analysis of the state of the art in multimedia retrieval. This analysis includes some introductory concepts about the content analysis field, which, as will be stated, is a key agent in the development of MIR systems.

2.1.1 Information versus Data retrieval

We find the inclusion of the distinction between Information and Data retrieval proposed by Baeza-Yates and Ribeiro-Neto (1999) in the context of this thesis convenient. The term Data retrieval should be employed whenever the main objective is the determination of which documents of a collection contain the keywords that the user employed in a query. However, most frequently, that it is not enough to satisfy the user information need. In fact, the user of an IR system is concerned more with retrieving information about a subject than with retrieving data which satisfies a given query. A data retrieval language aims at retrieving all objects which satisfy clearly defined conditions such as those in a regular expression or in a relational algebra expression.Thus, for a data retrieval system, a single erroneous object among a thousand retrieved objects means total failure. For an information retrieval system, however, the retrieved objects might be inaccurate and small errors are likely to go unnoticed. The main reason for this difference is that information retrieval usually deals with natural language text which is not always well structured and could be semantically ambiguous. On the other hand, a data retrieval system (such as a relational database) deals with data that has a well defined structure and semantics.

This difference is even more evident in the case of this work, when the assets to be retrieved are multimedia ones. This, as stated Cusumano (2005) is even noticed in the attitude of the user, which is usually more tolerant to the lack of precision of the systems.

2.1.2 Summary of a long history

Singhal (2001) from Google refers to the Sumerians (3000 BC) to locate the beginning of the practice of archive information. Professor Larson (2010) also mentions the Sumerians but goes even backwards, considering that the mnemonic systems probably developed in prehistoric times can also be considered a form of mental IR. Although it is the aim of this thesis to discuss the origin of the Information Retrieval, we share the idea that the need to archive and retrieve information became more and more important during the centuries. Even more, with the invention of the paper and the printing press. The computers also were employed for this aim.

The article “As We May Think” written by Bush (1945) is considered as the beginning of the automatic access to large amounts of data stored. In the fifties, several works were developed about the basic idea of searching and finding text with a computer. The work of Luhn (1957) is one of the key references of that period. During the next decade, several key developments in the field happened. Most notable were the SMART system developed by Salton (1971), first at Harvard University and later at Cornell University.

Based on the work of this decade, during the 1970s and 1980s many developments were built. Several models for retrieving documents were developed and the progress in all the steps of the retrieval process were important. The experiments were tested on small text collections (several thousand articles) available to researchers. This lack of large collections was solved with the 1992 Text Retrieval Conference or TREC, which established objective methodologies and measurements for information retrieval systems, that are employed nowadays.

The algorithms developed during those decades were the first ones to be employed for searching the World Wide Web from 1996 to 1998. However, the powerful provided by the cross linkage available on the web led to the implementation of new approaches, which are out of the scope of this thesis.

As we analyze later in 2.1.4, the explosion of the multimedia asset has led, in the recent years to a new revolution in the field.

2.1.3 Information Retrieval Models: classical and modern

As Larson (2010) defines, a model for IR is a specific and distinct approach for the text processing and the ranking algorithms of the system. A shared agreement in the key literature in IR is that the main classic information retrieval models are the following: Boolean, Vector Space, and Probabilistic. In addition there are many systems that are hybrids of two or more of these models (e.g., a vector system with Boolean result limiting features).

The earliest retrieval model is the Boolean model, described in the work of Gudivada et al. (1997),f and is based on Boolean logic. Most of the earliest commercial search services, local search engines or individual Web sites implement this model. The Boolean model is a set-oriented model, where sets of documents are defined by the presence or absence of an individual index term. If the term is there, and the logic of the boolean (AND, OR...) query is fulfilled, the document is retrieved. Boolean systems have several disadvantages. Perhaps the most serious is that there is no inherent notion of ranking.

The vector space model, deeply described in the work of Salton et al. (1975), represents a document as a vector of terms. Vector space IR systems base implement ranking algorithms according to how close together the vector of the query and the vector of all the documents are. So, the ranking is a kind of similarity measure based on the terms employed in the query and in the

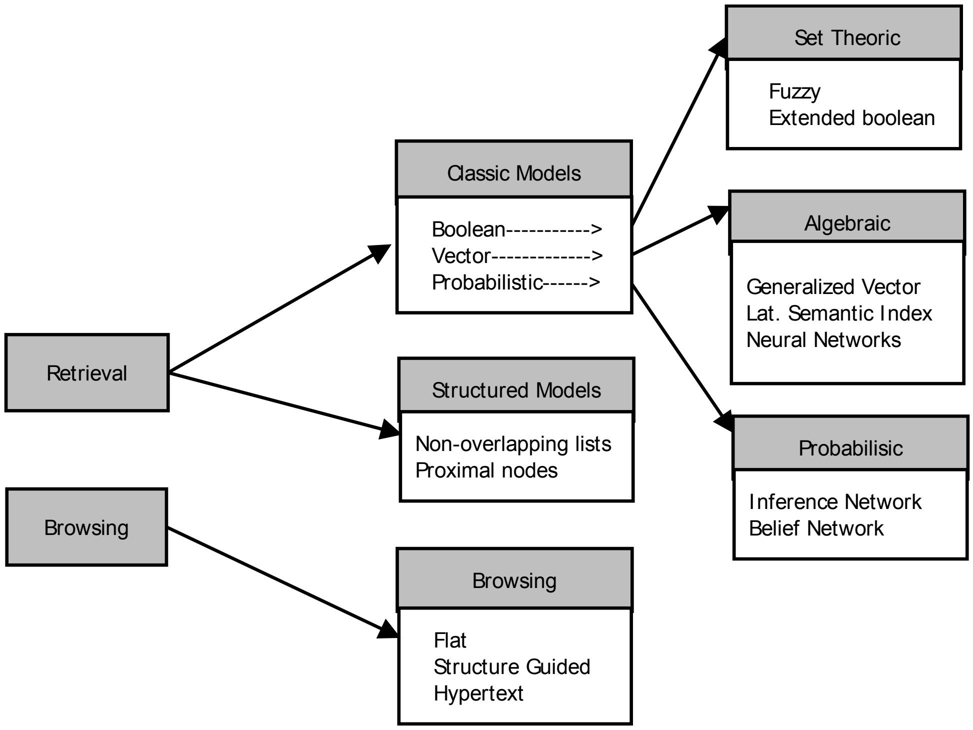


Figure 2.2: Model classification in modern Information Retrieval

documents archived.

The probabilistic model is based on what is called the Probabilistic Ranking Principle (PRP): the documents of a collection should be ranked by decreasing probability of their relevance to a query (Robertson, 1997). Relevance is therefore defined as a subjective assessment by a given user or machine of the value or utility of a particular document in satisfying a particular need for information.

Baeza-Yates and Ribeiro-Neto (1999) go further and, as can be seen in Figure 2.2, beside the classical models, which are deeply explained, two new models are included for the retrieval and three for the browsing.

The structured models are aware and make use of certain knowledge of the structure of the document. Inside this category we distinguish the non-overlapping lists and Proximal Nodes approaches. First of all, regarding the non-overlapping lists approach, Burkowski (1992) proposes to divide the whole text of each document in non-overlapping text regions which are collected in a list. Since there are multiple ways to divide a text in non-overlapping regions, multiple lists are generated. According to this, a book may be composed of a list of all the chapters, a list of all the sections and a list of all the subsections. While the text regions in the same (flat) list have no overlapping, text regions from distinct lists might overlap. And, once these lists are created the approach is similar to the one employed in the vector-space, but applying it for each list. Secondly, the proximal nodes models proposed by Navarro and Baeza-Yates (1997) propose a model which allows the definition of independent hierarchical (non-flat) indexing structures over the same document text. Each of these indexing structures is a strict hierarchy composed of chapters, sections, paragraphs, pages, and lines which are called nodes. To each of these nodes is associated a text region. Further, two distinct hierarchies might refer to overlapping text regions. Given a user query which refers to distinct hierarchies, the compiled answer is formed by nodes which all come from only one of them. Thus, an answer cannot be composed of nodes which come from two distinct hierarchies (which allows for faster query processing at the expense of less expressiveness).

Finally, regarding the browsing models, Baeza-Yates and Ribeiro-Neto (1999) define three approaches: flat, structure guided browsing and hypertext. First of all, with the **flat model** the idea here is that the user explores a document space which has a flat organization. For instance, the documents might be represented as dots in a (two dimensional) plan or as elements in a (single dimension) list. The user then glances here and there looking for information within the documents visited. Secondly, the **structure guided browsing** tries to facilitate the task of browsing organizing the documents in a structure such as a directory. Directories are hierarchies of classes which group documents covering related topics. Finally, the **hypertext** is a high level interactive navigational structure which allows us to browse text non-sequentially on a computer screen. It consists basically of nodes which are correlated by directed links in a graph structure.

2.1.4 Multimedia Information Retrieval (MIR)

The explosion of multimedia content caused by the digitalization and the convergence of the technology has conducted to a new revolution in the information retrieval. This revolution has led to new trends and techniques for very diverse aspects of the retrieval (Tse, 2008): object representation, architecture for storage systems, data compression techniques, statistical placement of discs, scheduling methods for disks requests, multimedia pipelining and stream dependent caching among many others.

However, the main impact of this revolution in this work is related to the way the metadata is created and made accessible for the search and retrieval. In this context, the mentioned revolution has impacted on two scientific fields, blurring the boundaries between them: Multimedia Information Retrieval (MIR) and Image and Video Analysis.

On the one hand, in the IR field, the inclusion of the multimedia assets in the information retrieval implies new means for the storage, retrieval, transportation, and presentation of data with very heterogeneous features such as text, video, images, graphs, and audio. Baeza-Yates and Ribeiro-Neto (1999) in their book about the modern concept of Information Retrieval already include several chapters focused on the techniques and approaches to retrieve multimedia assets, as an emerging particularity of IR. The motivation behind this new activity is due to the fact that traditional IR techniques are very efficient from the performance and precision point of view when the fundamental unit is the textual document and the search is based on text and carried out over simple data types. However, in the case of multimedia information retrieval the underlying data model, the query language, and the access and storage mechanisms *must be able to support objects with a very complex structure*. Furthermore, the scientific work devoted to establish the foundations of the next generation of multimedia information retrieval systems, such as the remarkable contribution of Meghini et al. (2001), are slowly having an impact in the commercial products. An example of this preliminary deployment of such concepts is the last version of the multimedia database of Oracle, which is able to handle and perform some operations on new object types (e.g., DICOM images from the medical sector).

On the other hand, the image and video community has spent remarkable efforts during the last years to promote what they coin as "Content or semantic based visual/multimedia information retrieval" (CBVIR) (Lew et al., 2006; Naphade and Huang, 2002). According to Zhang (2006), CBVIR has already a history of fifteen years, but it is in the last years when the focus has moved from extraction of low-level features from the multimedia assets (e.g., dominant colour in an image) to the resolution/minimization of the semantic gap (e.g., person recognition). The community is devoted to a higher level of semantic abstraction. This is called by the author as Semantic-based Visual Information Retrieval, and is leading to the application of such technologies for the enhancement of current multimedia management and retrieval systems. Among the processes being improved, we may highlight the followings: indexing and retrieval, higher-level interpretation, video summarization, object classification and annotation, and object recognition. We also want to notice that in all these disciplines, the presence of the technologies developed by the semantic web community has been significantly increasing during the last years. The book edited by Stamou and Kollias (2005) provides a very complete summary of the efforts made by the community to perform the semantic analysis required for the multimedia information retrieval.

In this section, we try to provide an introductory explanation to some of the most relevant techniques and current challenges for the main disciplines involved in that task. The image, video and audio components are treated independently. For each of them, we identify the key research issues and trends. In Section 2.4, we include a brief summary of the state the art in content based information retrieval in the context of this thesis. That summary has a more technical and specific perspective.

Image retrieval

When the retrieval is about images, the metadata is not the only valuable piece of information, but also, thanks to the image analysis techniques, of the features of the images (Zhang and Izquierdo, 2008). According to Eakins and Graham (1999), depending on the features of the image you employ for the retrieval, the queries can be classified into three levels, being each of the level of different complexity:

- Level 1: Primitive features such as colour, texture, shape, or the spatial location of image elements.
- Level 2: Derived features involving some degree of logical inference about the identity of the objects depicted in the image.
- Level 3: Abstract attributes involving a significant amount of high-level reasoning about the meaning and purpose of the objects or scenes depicted.

The first level of queries is the set of queries that are more easily solved. All the information is gathered in the image and, therefore, there is not need of any external intelligent resource. This type of queries is relatively easy to solve but is largely limited to specialist applications. Levels 2 and 3, which are in fact most widely demanded, together are commonly referred to as semantic-based visual information retrieval. However, there is an important gap between Level 1 and 2, referred to as the semantic gap (Smeulders et al., 2002).

The bridging of this semantic gap is the main objective of most of the research activity of the scientific community working in image retrieval. Zhang and Izquierdo (2008) group the efforts of this community according to two main classifications. The first classification is based on the features exploited for the retrieval. The second classification is made based according to the different retrieval paradigms existing in the literature.

Concerning the first classification, these research works imply that general visual information representation schemes that are employed to design image retrieval algorithms can be categorized into the following three classes:

- Textual feature-based.This is based on the written metadata available about the image, and is concerned with the classical retrieval technology already stated.
- Visual feature-based. The paradigm behind this is to represent images or video clips using their low-level attributes, such as color, texture, shape, sketch and spatial layout, motion, audio features, which can be

automatically extracted from the multimedia content themselves. There are many examples of this preliminary approach in the literature. For instance, the pioneer QBIC (Faloutsos et al., 1994) and the more recent PicHunter (Cox et al., 2000).

- Combined textual-visual feature-based methods. Many researchers have investigated the possibility of combining the text-based and content-based retrieval. For instance, the iFind (Lu et al., 2000) system approaches the problem first by constructing a semantic network on top of an image database, and then using relevant feedback based on low-level features to further improve the semantic network and update the weights linking images with keywords. Other approaches (e.g., the work of Shi and Manduchi (2003); Zhang and Izquierdo (2007)) just treat each feature individually and fuse the lists to obtain the final results. However, this is still a remain challenge for the community.

Regarding the second classification, related to the retrieval paradigms, the main categories are the following:

- Region based representations. According to the current state of the art in image analysis, it is difficult to go beyond the extraction of middle level features. And these middle level features usually are not referred to the whole image, but just to a part of it. For that reason, many research works seek to use a combination of regional descriptions to represent an image, because it is much more feasible to link those middle level features (e.g., vegetation, sky) to regions. In our opinion a reference work in the literature is the work performed by Papadopoulos et al. (2007) in the ITI institute in Greece .

- Fusion of multiple features. This category compiles those works related to the joint exploitation of different features of multimedia content. The motivation behind this thesis is that different features and their respective similarity measures are not designed to be combined naturally and straightforwardly in a meaningful way. A large number of different features can be used to obtain content representations that could potentially capture or describe semantically meaningful objects in images. The challenge in this type of work is the appropriate selection of those features. Zhang and Izquierdo (2007) provide a review of this kind of approaches.

- Probabilistic inference for context exploitation. The current imprecision of the image analysis algorithms (Santini, 2003) and the aim to approach to the way the human brain behaves are the main motivation of this kind of approaches, where statistical methods are employed to learn and train algorithms. Popular techniques related to storing and enforcing high-level information include neural networks, expert systems, fuzzy logic, decision trees, static and dynamic Bayesian networks, factor graphs, Markov random fields, etc. A comprehensive literature review on these topics can be found in the work of Naphade and Huang (2002).

- User relevance feedback. These approaches make use of the last stage of a retrieval process, to employ the users' judgement to influence the previous steps. By doing so, the retrieval systems accept the user as the central actor, which implies accepting the users' interactions with information as the central process. Rui et al. (1998) and Crucianu et al. (2004) provide a complete introduction and short survey to this kind of techniques.

Video retrieval

From the visual perspective, a video can be understood as a consecutive set of images or frames. However, the techniques for the automatic extraction of features out of a video have some peculiarities that have not been covered in the previous section. In this section we cover some particular research challenges and subfields that are particular of video retrieval. First of all, the **shot boundary detection or scene segmentation**, that can be understood as a continuous action in an image sequence (Han et al., 2000). This is one of the first steps to be applied in video processing. As a consequence, a video is divided into a set of sub-videos or shots. There are different types of transitions between shots. Depending on the transition, there are more suitable techniques. Geetha and Narayanan (2008) provide an introduction to six families of algorithms. In general this is a well solved issue being the results on the shot boundary detection competition of TRECVID conference very appealing (close to 100% of precision).

Once the shots have been identified, it is very suitable to extract the set of most representative frames in the shot. These frame or frames are named **Key-frames**. Once this is done most of the techniques mentioned for image retrieval can be applied to that shot straightforwardly. The simplest techniques are static (e.g., select the central frame of the shot) but there are also very challenging unsupervised approaches e.g., (Hafner et al., 1995; Hauptmann et al., 2003) that automatically select the most suitable key-frame according to the established parameters.

Another distinction with respect with the image analysis is the existence of the spatio-temporal relationship. Some relevant works on this are devoted to the extraction of **motion descriptors** (Smeulders et al., 2002) and **temporal texture**(Ngo et al., 2003).

The content of the video, specially if it is long, can be organized using **clustering techniques**. The similar shots or frames are grouped on the same cluster, simplifying the analysis and understanding of the video. Here, again, the approaches are multiple (e.g., hierarchical clustering (Fan et al., 2004) and spectral clustering (Chasanis et al., 2008) among others).

Regarding the indexing and retrieval techniques, the approaches are in general similar to the image retrieval. However, we would like to mention some innovative approaches that make use of the peculiarities of the video in order to enhance the **automatic annotation** of the asset. Hanjalic (2005) has developed a system for adaptive extraction of highlights from a sport video based on excitement modeling. Feldmann et al. (2008) have employed the motion of the camera for the automatic detection and modeling of flat surfaces. Vasconcelos and Lippman (1997) integrated shot length along with global motion activity to characterize the video stream with properties such as violence, sex or profanity.

Finally we should notice that the videos and the multimedia assets, are composed by one or more essences or modalities. The research works for indexing and retrieval of multimedia assets employ more and more the different modalities present in the asset to extract as rich annotations as possible in what is called **multimodal analysis** (Lai et al., 2002; Snoek et al., 2007; Wu et al., 2005).

Audio retrieval

The retrieval of Audio retrieval is a very wide scientific field and is not fully aligned with the scope of this thesis. Therefore, it is completely out of our expertise and in our validation scenarios we have never tackled this issue. Since the audio is not covered in the rest of this thesis work, we do not go into detail of this subject. As an introduction to the field, we would like to name some of the multiple disciplines and key references behind the audio retrieval. The challenges and the techniques employed for the retrieval of music (Byrd and Crawford, 2002; Ellis, 2006; Klapuri, 2004), notated music (Hoos et al., 2001) or human spoken audio (Peinado, 2006; Rabiner and Juang, 1993) are totally different. In the work of (Spanias et al., 2007) and (Zoelzer, 2008) the reader may find support for a deeper understanding of the field.

2.2 Approaches for the semantic improvement in the multimedia retrieval workflow

In this section we aim to cover the relevant work in the literature related to the improvement of the retrieval of multimedia assets by employing semantic-aware technologies but from a global perspective. This means that the work that we present in this section does not employ semantic technologies to improve just some aspect of the retrieval process but, in a similar way to our approach, aims to tackle this issue from a broader perspective.

This is, to our vision, the closest related work to our approach. According to this, we dedicate a different subsection for each of them. Each subsection includes not only the description of the work, but also the main differences among the work presented by the authors and the work we present here.

2.2.1 An intelligent media framework for Multimedia Content

Bürger (2006) and Günter et al. (2007) also face the gap between the heterogeneity of the information and the users. They state that the main motivation behind their work is that in the current multimedia management systems, users are supported by a wide range of features which are traditionally based on full text search and metadata queries. However, generating metadata is an error-prune and work-intensive task, that for multimedia content cannot yet be made fully automatically. In this context, they define the Intelligent Media Framework (IMF) to formalize and manage the semantic connections across the system, semi-automatic annotation tools to index multiple incoming streams, information databases and audiovisual archives, and a recommender system to analyse and visualise consumer feedback that is delivered over a back channel system (Messina et al., 2006). This framework provides the following services to the rest of the components of the system.

- Services to create, annotate and manage the intelligent media assets that make up the show under real-time conditions. These services operate on a metadata level and do not actually store any raw video streams (they rather reference these so called essence).
- A service to manage and deliver information about the staged events (e.g., the schedule of the contests and races, the participating athletes, the results).
- A service to manage and deliver information on the way how a live broadcast of a sporting event is presented (e.g., which types of switching concepts are available and used in a certain concept of a show, which basic dramaturgic concepts are appropriate according to the disposition of the production team and/or a predefined mood of the show).
- A service to access the vocabularies and the terms of the controlled vocabulary constituting the knowledge base of the live staging domain.
- A messaging system to support the real-time aspects of the staging process by offering subscription methods to other subsystems.

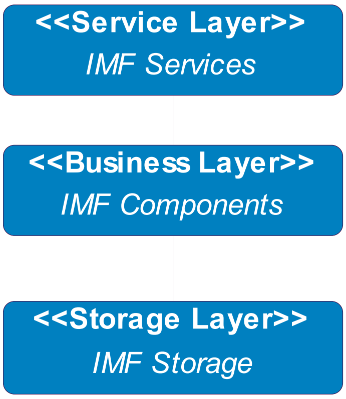


Figure 2.3: Architecture of the Intelligent Media Framework Component

This set of objectives makes the Intelligent Media Framework and its applications very close to the our semantic middleware and specially to the deployment of the RUSHES system. Beside this, its architecture (see Figure 2.3 is also based on a combination of a classical three-tier architecture with the principles of Service Oriented Architectures (SOA). The main responsibilities are addressed in different layers:

- The Service Layer: In a similar way to the semantic middleware gateway that we present in Section 3.2, the services layer consists of the services provided by the IMF to the other building blocks of the whole system and external systems. These building blocks include components responsible for the manipulation, semantic enrichment and recommendation of data.
- The Business Layer: this layer is designed to interact with the services and therefore is in charge of handling the data in specified incoming data formats. This includes the data formats specified by the IMF as well as standard data formats such as MPEG-7 or NewsML.
- The Storage Layer: This layer is responsible for the transformation of data into the data formats specified by the IMF data model and secondly for the provision of a persistence layer for the whole system.

The lower the layer, the more differences are detected with respect to the approach we present. The IMF relies on the storage layer for the transformation of the data into the data formats specified by the IMF data model. This implies that the storage of the semantic model (passive role) is not the main mission of the layer but the transformation of the external data into a kind of internal representation according to a model (active role). This is due to the fact that the IMF relies on a specific data format to provide the mentioned semantic services. In this approach, the assets are not just a passive representation of information but complex objects. Those complex objects are called by the author Knowledge

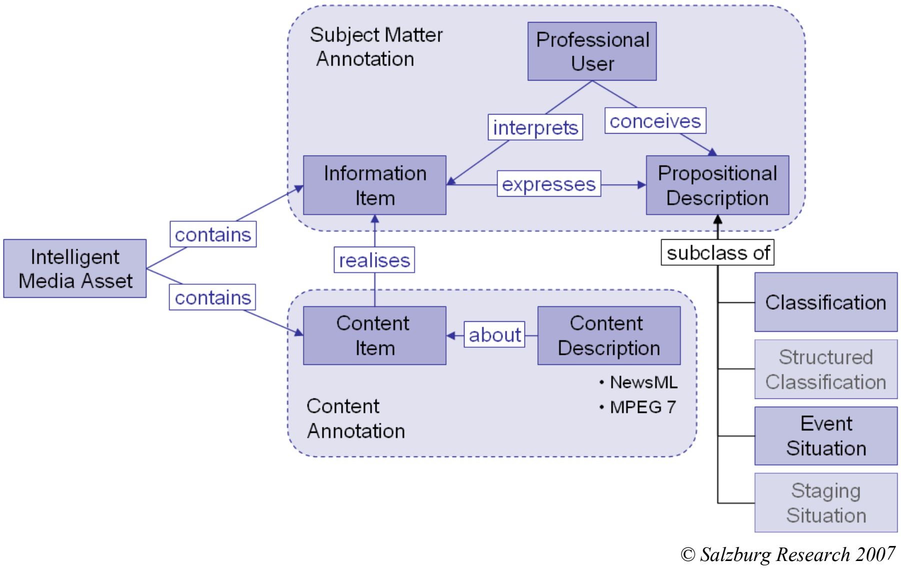


Figure 2.4: Model of the Knowledge Content Objects of the IMS

Content Objects and their model can be seen in Figure 2.4. The core parts of this model are content annotations which provide information about the essence (i.e. the raw video stream) and subject matter annotations which provide information about the subject matter of the essence.

To summarize, the IMF, in the context of the work we present here, can be understood as a kind of semantic middleware which is designed to work in a specific environment where the multimedia assets are mapped into a new set of multimedia assets that are able to perform by themselves some semantic operations. This fact and the consequences in the design and implementation derived from it are the main differences with this work. In our opinion, this approach is not compatible with the motivation behind the work we present here. The main reasons for this are the performance and cost consequences of such replication and the techniques employed for the storage and management of the Knowledge Content Objects.

2.2.2 Information Mediation Layer: a new component for the digital libraries architecture

Candela et al. (2006), in the context of a larger effort dedicated to the definition of a reference model for the digital library (Candela et al., 2007), describe the motivation and scope of the introduction of a new layer: “The Information Mediation Layer”.

Their work relies on the idea that Digital Libraries are often built by exploiting

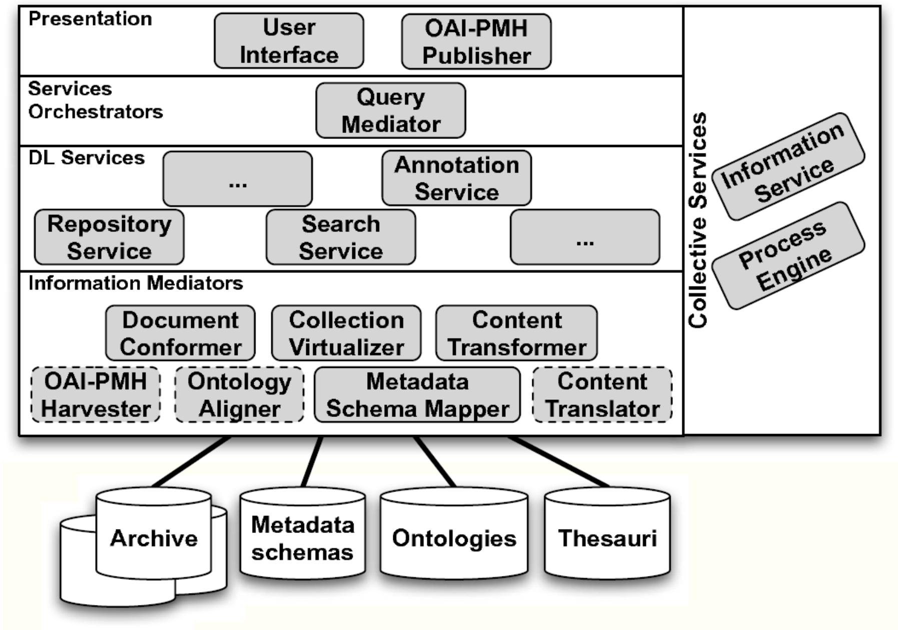


Figure 2.5: Reference Model for the Digital Libraries

already existing resources. According to them, the most frequently shared resources are the documents of the archives, but many other type of resources, such as authority files[1], thesauri, language dependent resources, ontologies, classification systems [2], and gazetteers [3]. Those resources are mainly created by third-parties and are heterogenous.

The need of handling this heterogeneity is the main motivation of their work, the information mediation layer which is graphically summarized in Figure 2.5.

This layer implements the services required for the provision of virtual views of what they name information spaces. The main idea is to improve the access to the information by homogenizing it. The mediators that compose this area may be classified according to the following:

- Information organization: These mediators are related to the semantic representation of the information organization aspects. They may be related to the problem of the heterogeneity (e.g., the provider of a virtual view of the information object model, which is able to provide information about the multiple object manifestations, the object composition) or the problems raised by large volumes of data (e.g., the provider of a virtual collection view, which is able to organize the information space in multiple sets of objects, each capable to meet a different need).

[1] http://authorities.loc.gov/
[2] http://www.oclc.org/dewey/
[3] http://middleware.alexandria.ucsb.edu/client/gaz/adl/index.jsp

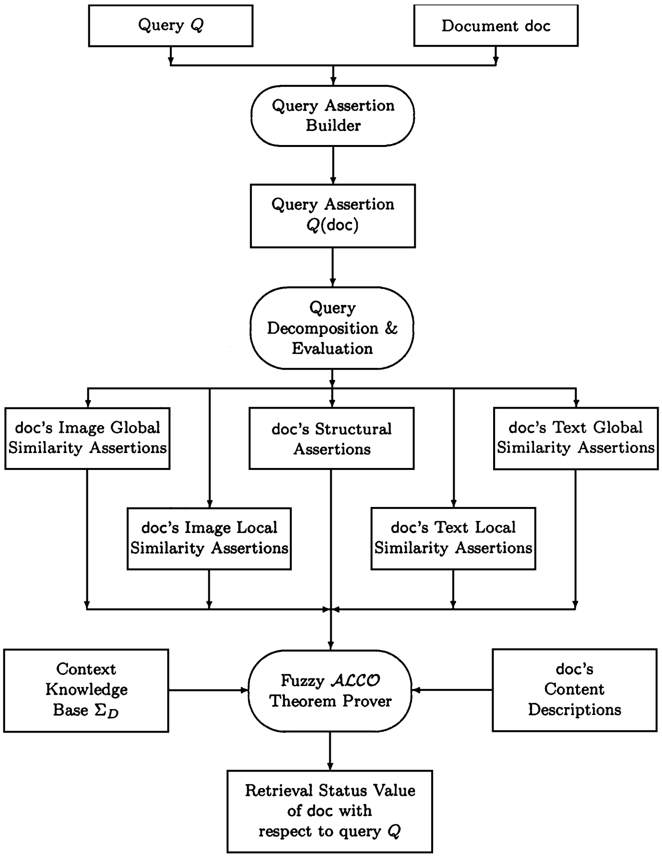


Figure 2.6: Query Decomposition process

- Object manifestation: This kind of mediators provides a manifestation view. The manifestation is the way through which the content of an information object is perceived by the user. The functionalities provided by the services of this area are: (i) to access the manifestation while hiding details about its storage, and (ii) the dynamic generation of alternative and more profitable manifestation formats.

- Metadata object manifestation: This class of mediators provides a metadata view. The functionalities provided by the services of this area are: (i) the presentation of the metadata in a required format, and (ii) the dynamic generation of new metadata.

Thus, the Information Mediator Layer has a number of services that implement the corresponding mediation functionality. Some of these services are mandatory in any Digital Library (DL) system while others depend on the specific DL application area.

Regarding the contextualizing of Candela et al. (2006) contribution with respect to this thesis, their approach is in general very similar. However, the lack of technical details of the mediators or information about any implementation makes difficult the identification of the similarities and differences.

2.2.3 A model for multimedia information retrieval

The work of Meghini et al. (2001) is a very remarkable contribution in the literature that handles the problem from a generic perspective. This theoretic work results in a conceptual model that, according to the authors, encompasses in a unified and coherent perspective the many efforts that are being produced under the label of MIR.

The model is formulated in terms of a fuzzy description logic, which plays a twofold role: (i) it directly models semantics-based retrieval, and (ii) it offers an ideal framework for the integration of the multimedia and multidimensional aspects of retrieval mentioned above. This scope is the reason because we have included Meghini work in this section, in spite of the fact that the nature of this work, as we do state later, is intrinsically different. Figure 2.6 aims to graphically summarize the approach followed by the model to address the query processing. The model presents a decomposition technique that reduces query evaluation to the processing of simpler requests, each of which can be solved by means of widely known methods for text and image retrieval, and semantic processing. Each of the steps in the process has been expressed mathematically according to the mentioned fuzzy description logic. Therefore the semantics, the current state of the art in multimedia querying and the peculiarities of the multimedia retrieval have been taken into account by the authors over the whole process.

This model shares our motivation of bringing semantics to MIR systems. However, while our aim is to support system managers to complete their current facilities, the scope of this model is to define guidelines for the design of systems that are able to provide a generalized retrieval service. This fact and the nature of the model are the most significant differences with respect to the work we present here.

2.2.4 A three layer infomediation architecture

Kerschberg and Weishar (2000) in their article about Conceptual Models and Architectures for Advanced Information Systems present an approach about how conceptual modeling of information resources can be used to integrate information obtained from multiple data sources, including both internal and external data.

Their work is based on a three-layer Reference Architecture consisting of various types of mediation services, including facilitation and brokerage services,

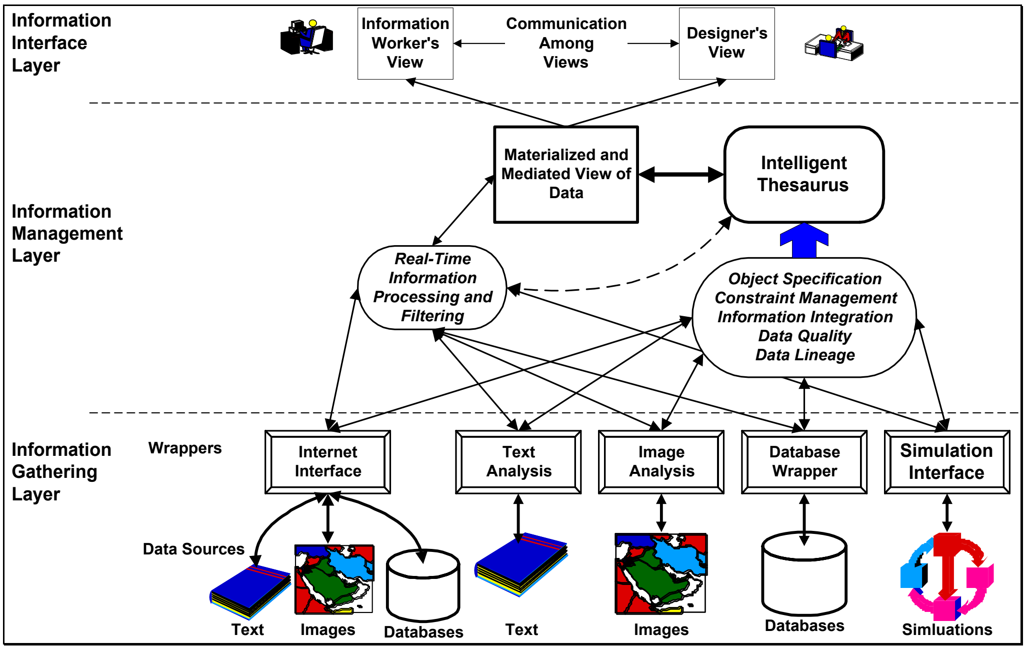


Figure 2.7: Layered Information Architecture & Processes

mediation and integration services, and wrapping and data access services. Although their work is domain agnostic, in Figure 2.7 a particularization for the logistics domain is shown.

The upper layer Information Interface layer, is the layer in charge of providing the users with the available information. This layer must support scalable organizatino, browsing and search. Some of the services provided by that layer are the intelligent thesaurus or yellow pages.

The intermediate layer Information Management layer, is responsible for the semantic integration, replication and catching of the information gathered by all the information sources.

Finally, the bottom layer Information Gathering layer is responsible for collecting and correlating the information from many incomplete, inconsistent, and heterogeneous repositories.

This short summary is enough to understand the main differences of this approach with this thesis work. On the one hand, the focus of their work is devoted in information integration. On the other hand, their approach relies on the adaptation and replication of that information, instead of the provision of semantic services (e.g., terminological mapping, negotiation resources) to a main system in order to perform searches over external repositories.

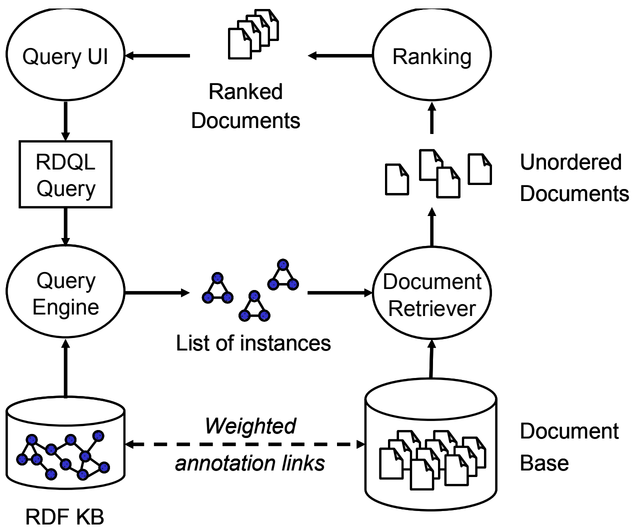


Figure 2.8: View on ontology-based information retrieval

2.2.5 Ontology Based Information retrieval

The work of Castells et al. (2007) is a relevant example of what can be understood as ontology based retrieval system. Their proposal is a retrieval model meant for the exploitation of full-fledged domain ontologies and knowledge bases, supporting semantic search in document repositories. Castells et al. (2007) in a coherent view with their understanding of semantic information retrieval (see Figure 2.8), assume that each information source includes a knowledge base (KB) which was built using one or several domain ontologies that describe concepts appearing in the document text. The concepts and instances in the KB are linked to the documents by means of explicit, non-embedded annotations to the documents.

This work and its promising results shares the motivation of this thesis work and provides semantic services over different steps of the multimedia retrieval (e.g., query processing, result ranking). However, it relies on a very specific particularization: the existence of a knowledge base expressed in an XML format (in this example RDF or Resource Definition Framework) for each of the information sources. This is the main difference with this thesis work and is clearly against our will to facilitate the integration of the semantic techniques in current multimedia asset management systems.

2.2.6 Ontology-enriched semantic space for Video Search

Wei and Ngo (2007) share our aim of diminishing the semantic gap between the low-level features available for the multimedia assets due to the analysis algorithms and the high-level features demanded by the users. They propose a novel model, namely Ontology-enriched Semantic Space (OSS), to provide a computable platform for modeling and reasoning concepts in a linear space. According to the authors, OSS enlightens the possibility of answering conceptual questions such as a high coverage of semantic space with minimal set of concepts, and the set of concepts to be developed for video search.

The basis of their work is a simplification, in the sense of performance and computational resources consumed, of the comparison of concept pairs. The OSS is composed by a semantic space that is linearly constructed to model the available set of concepts. The expressive power of OSS is linguistically spanned with a set of basis concepts, which is easier to generalize, not only to the available concept detectors but also to the unseen concepts.

The main implications of this simplifications are the following:

- Query disambiguation: OSS facilitates the interpretation of the terms of the query of the user.
- Query Concept Mapping: The comparison between the concepts is done by ensuring the global consistency.
- Multi modality fusion: OSS is a key element for the generation of concept clusters, and the authors demonstrate that those clusters allow to effectively fuse the outcomes of concept based retrieval (visual) and text based retrieval (keywords).
- Scalability: OSS facilitates the selection of concept detectors (e.g., face recognition) that result to be more useful for query answering in a domain.

We have included the work of Wei and Ngo in this section not because we consider that it can be understood as a pure semantic middleware in a retrieval system but because its defines a semantic infrastructure generic enough to be employed to support several steps of the information retrieval process. The authors do not focus of the integration of their technology into existing systems. However, they describe their work as a useful semantic resources to empower the analysis of the content, the implementation of navigation mechanisms (e.g., cluster construction) and the mapping of the queries into the internal vocabulary.

2.2.7 MPEG-7 driven multimedia retrieval

While there are profuse bibliography related to the development and use of the MPEG-7 standard (Dasiopoulou et al. (2010) provide an extensive state of the art just on MPEG-7 ontologies), it is not easy to find relevant references that employ the MPEG-7 to bo beyond the building of a specific solution, generating a framework or whole retrieval system.

In that scare bibliography references, the work of Schallauer et al. (2006), a Description Infrastructure for Audiovisual Media Processing Based on MPEG-7 and some complementary reports of the same authors Bailer and Schallauer (2006); Bailer et al. (2007) are, from the perspective of this thesis, remarkable contributions.

Schallauer et al. (2006) work tackles, from a generic perspective, a big set of aspects related to one of the key steps of the multimedia retrieval process, the multimedia processing. According to this, their system is able to import audiovisual data into the system and to perform and control automatic content-analysis tools which extract a number of low- and mid-level metadata. But going beyond that, as Figure 2.9 reflects, their contribution includes also the following components:

- A Manual Documentation Component used for textual descriptions and description of high-level semantic information, which cannot be extracted automatically.
- A Search component for query formulation and result presentation, which provides search options for both textual and content-based queries.
- A backend infrastructure providing storage and search functionalities

As a result of this, they propose a complete open (MPEG-7 based) multimedia retrieval system that has been designed taking into account the difficulties and peculiarities of the multimedia indexing.

This work shares the objective of the thesis work we present here but it presents a difference: it does not complement a system, in fact it implements a new one. However, if we focus on the multimedia processing component, for one of the deployments that we present here, we share not only the approach but also the ontology employed, the DAVP profile of MPEG-7. While Schallauer et al. (2006) employ this ontology and a complete query, search and storage machinery for that ontology, this thesis is more generic. Even for the case that part of our middleware is composed by MPEG-7 ontologies, the storage and query facilities are shared with the rest of the ontologies present in the middleware.

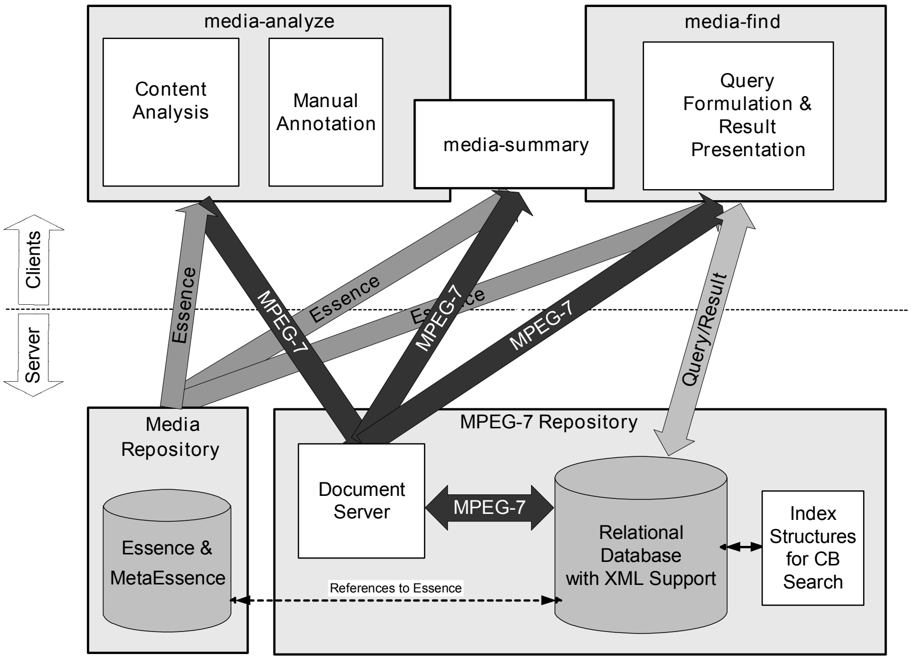


Figure 2.9: System overview of the infrastructure components for multimedia description

2.3 Metadata models for multimedia

Bailer and Schallauer (2008) provide an overview of the role of the metadata in the audiovisual media production process. The state one premise fully shared with us: although there are multiple multimedia metadata standards, no single standard fulfills all requirements required in complex real live applications. Both, the middleware that we present in Section 3.2 and its deployments (see Chapters 5 and 6) rely on this assertion. In this section we provide an overview of relevant references in the field of metadata models for the management (i.e. indexing, processing, searching and so on) of multimedia assets.

In order to facilitate the comprehension of the different standards, their differences and complementarities, we provide a summary of the different types of metadata.

2.3.1 Types of multimedia metadata

There are many different types of metadata (Cox et al., 2006; Smith and Schirling, 2006). Not all of them are involved in the search process. However, the digitalization has led to a system convergence in the companies and in order to enhance the retrieval systems with content-based features, usually all the metadata requirements of the company have to be taken into account.

Following the approach of the researchers of the Joanneum Research Institution Bailer and Schallauer (2008), we can classify the metadata according to three main parameters: the source of the metadata and its properties.

Types of metadata according to the source

- *Capture.* The capture metadata is mainly related to the technical description of the asset and is created together with the asset. Some examples of this metadata are the DMS-1 (SMPTE 380M-2004 - Descriptive Metadata Scheme - 1), annotations provided by some broadcast cameras, the Exif (Exchangeable image file format) information and so on.
- *Legacy and Related Information.* This metadata, sometimes generated even before the assets itself, makes reference to the legal aspects of the assets (e.g. production contracts) and audiovisual material that is related to the asset (e.g. interview to the creator).
- *Manual Annotation* This metadata is very rich from the semantic point of view, but very costly. In a professional environment, this information is reliable and valuable.
- *Content Analysis* This source is derived from automatic analysis of the content in order to extract metadata describing it. This metadata can be related to very low level features (e.g. histograms of a key-frame), middle level features (e.g. face identification) or high level features (e.g. face recognition). The problem of extracting semantics from the low and middle level features is known as the semantic gap (Santini and Jain, 1998) and is still not satisfactorily solved for open domains (Hauptmann et al., 2007). This type of metadata is therefore more unprecise, but extremely cheaper than the produced by the manual annotation.
- *Text and semantic analysis.*This includes recognition of references to named entities (e.g. persons, organisations, places) as well as linking them to ontological entities, the detection of topics and the classification of content segments and linking content to legacy or related information.

Types of metadata according to the properties The nature of an asset is usually complex. In a professional environment, an asset is composed of different essences (e.g. several audio tracks, subtitles and so on). Each essence consist on a dynamic representation of information that usually changes over the time. Taken this into account, we can distinguish between the following types of metadata according to its properties.

- *Scope.* A metadata unit may refer to the whole asset or just a segment of one of the components of the assets. It can apply to a spatial, temporal or spatiotemporal segment of the content. The same metadata elements may exist in different scopes, such as the title of a movie and the title of a scene.

- *Data type.* The datatypes of the metadata may be diverse. First of all it can be either textual or numerical. The textual metadata can be free text, discrete set of values (e.g. thesauri, ontologies). The numerical metadata can be composed of integer numbers, vectors, and so on.

- *Time Dependency.* Some metadata changes in the time (i.e. dynamic metadata) while other pieces metadata are not altered (i.e. static metadata).

- *Spatial dependency.* This is the same that in the previous case but for the spatial component.

- *Modality channel dependency.* Some metadata affects the whole asset while some affects fully or partially just to one of the modalities of the asset (e.g. audio).

- *Context dependency.* There are metadata that depend highly in the context in order to provide a meaningful interpretation. For instance, classifying a segment as “frightening” is fully context dependent.

2.3.2 EBU P/Meta

The EBU or European Broadcasting Union self-defines as “the largest association of national broadcasters in the world, built to promote cooperation between broadcasters and facilitate the exchange of audiovisual content”. This has had an impact in the work they have done regarding the metadata models and schemas. This work has been mainly focused on the exchange of metadata. This activity started in 1999, based on other works already in progress at the British Broadcasting Corporation (BBC) on the schema SMEF (Standard Media Exchange Format) and the RAI (Radio Televisione Italiana).

This work, tagged as P/Meta (EBU-Technical-Department, 2001), is a flat list of metadata entries focused on the commercial programme exchanged between broadcasters. P/Meta defines syntactical rules that must be followed when the generation of the metadata is done.

From the technological point of view, P/Meta does not constrain any implementation, since it does not go beyond the definition of the terms. It can be “materialized” as XML documents or Word docs or embedded in MXF (SMPTE 377-1-2009 Material Exchange Format). As other schemas, it uses numerical codes for attributes and standard values. This facilitates the machine manipulation and the multilingual aspect.

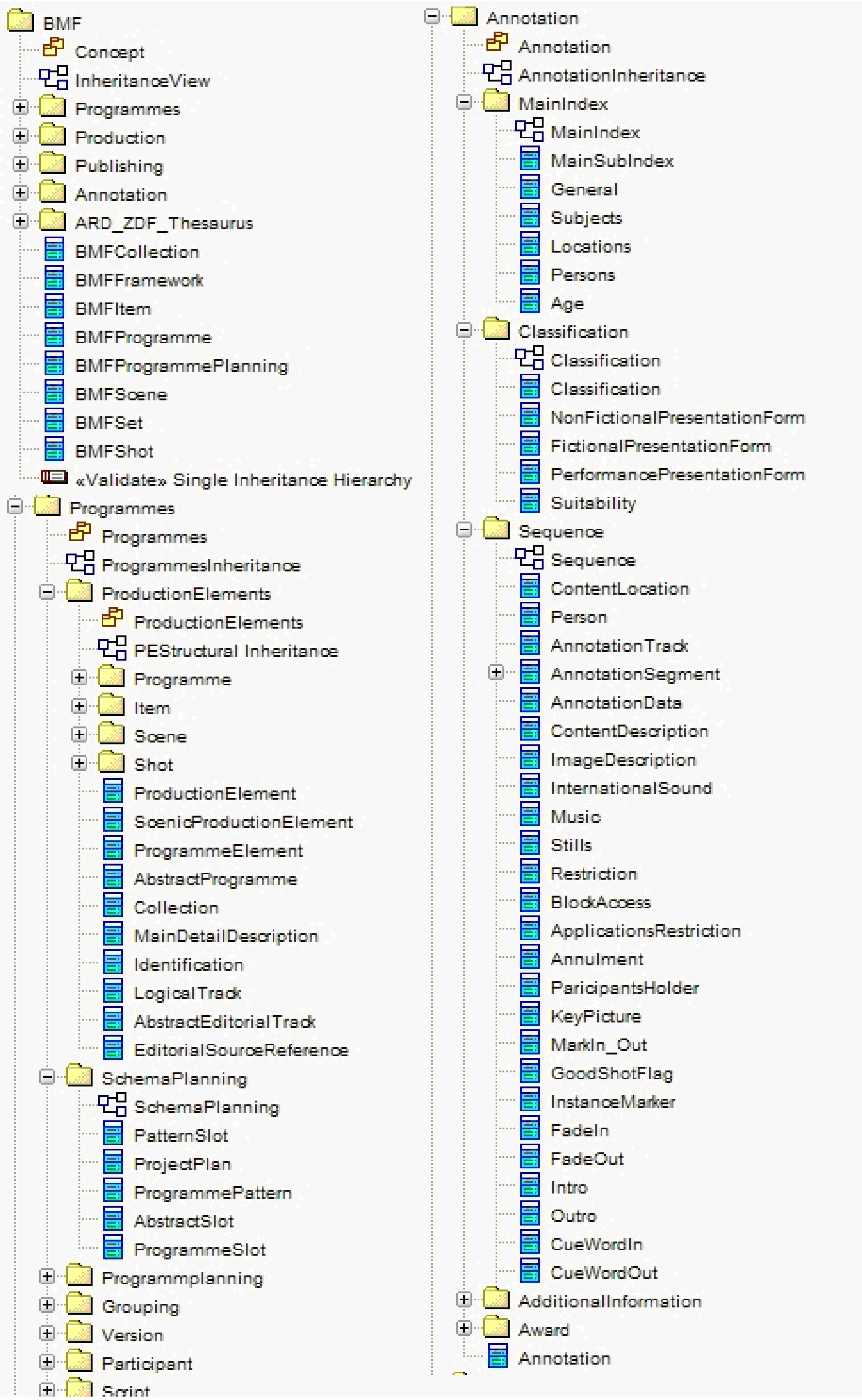


Figure 2.10: BMF root nodes

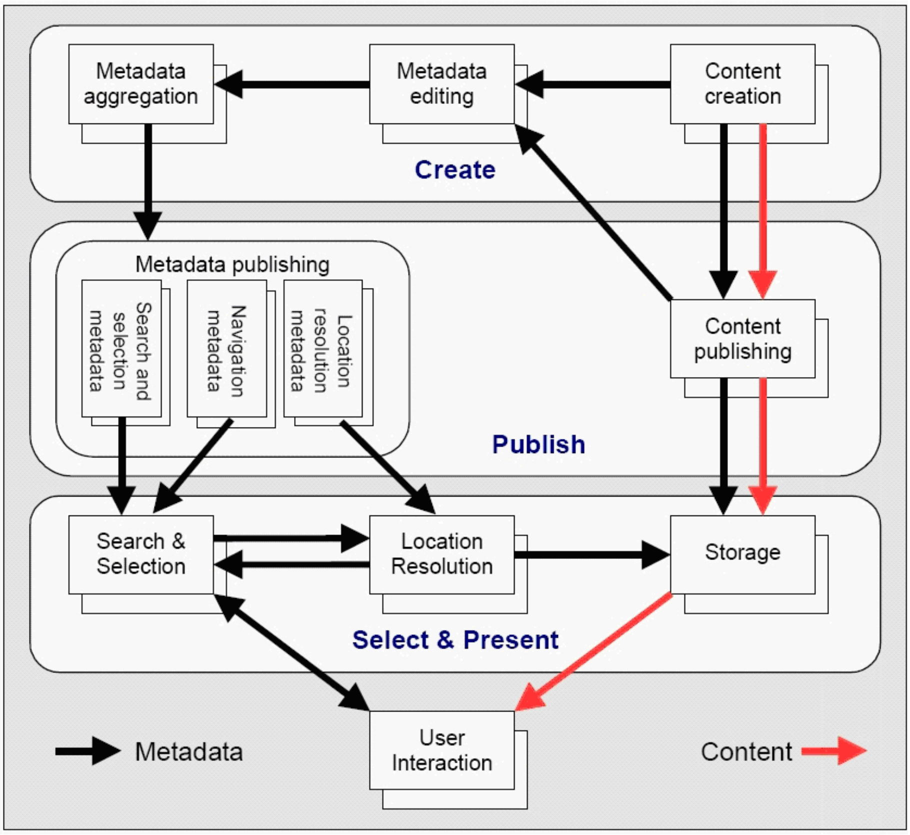


Figure 2.11: TVAnytime Metadata Model framework

2.3.3 Standard Media Exchange Format - SMEF

SMEF (BBC, 2000) is a standard for metadata modeling defined by the British Broadcast Corporation. It covers the indexing of the assets from a very wide perspective, going from the asset itself (media object) to the shot level and the editorial objects (programmes). While P Meta was mainly defined for the exchange, SMEF was defined for internal usage in the corporation.

2.3.4 Broadcast Exchange Metadata format - BMF

The Institut für Rundfunktechnik GmbH (IRT) has developed the Broadcast exchange Metadata Format (BMF) that defines an uniform, generic model for metadata in the field of television production.

As can be seen in Figure 2.10 the main nodes of the BMF schema are Programme, Production, Publishing, Annotation and the Thesaurus. In the same Figure two of the nodes more related to this work have been included.

From the point of view of the building of a middleware, the standard is focused on the exchange of material and therefore it does not cover very deeply the description of the essence itself. For instance, it does not handle the identification of the key frames of a video.

2.3.5 Dublin Core

Dublin Core (Dublin-Core, 2007) is a list of simplified metadata fields that can be used to annotate different media types and formats. For instance, Dublin Core can be used for broadcast essences, films, images, magazines and so on.

The set of main elements defined by the standard is composed of the following 13 items: contributor, coverage, creator, date, description, format, identifier, language, publisher, relation rights, source, title and type.

Each one of these descriptive elements can be modified by a qualifier (that can be seen as pairs of attribute/value). Thus, the annotation consist of a set of descriptives, each one of them with a list of qualifiers.

Dublin Core is a compact solution usually employed with catalogues with a non high complexity.

2.3.6 TV Anytime

TVAnytime or TVA (TV-Anytime, 2007) is a synchronized set of specifications established by the TV-Anytime Forum. TVA features enable the search, selection, acquisition and rightful use of content on local and/or remote personal storage systems from both broadcast and online services.

These specifications compose a framework based around a data model and a common metadata representation format. Besides this, there are some specifications to build some tools (i.e. Metadata, Rights management) on the top of that data model and representation format.

In the context of this thesis, the most interesting part of TVA is the 3rd part of the standard, which is the one that describes the mentioned Metadata Model that can be used to build the semantic middleware knowledge base (see Section 3.2).

Figure 2.11 clearly identifies the separation of the processing of metadata and content while at the same time it illustrates the parallels between the processing of metadata and content. Both content and metadata go through the creation, publication and consumption (select & present). However, both suffer different modifications during those processes.

According to this, TVA Metadata Model distinguishes between the Creation, Publication and Selection and Presentation of the item.

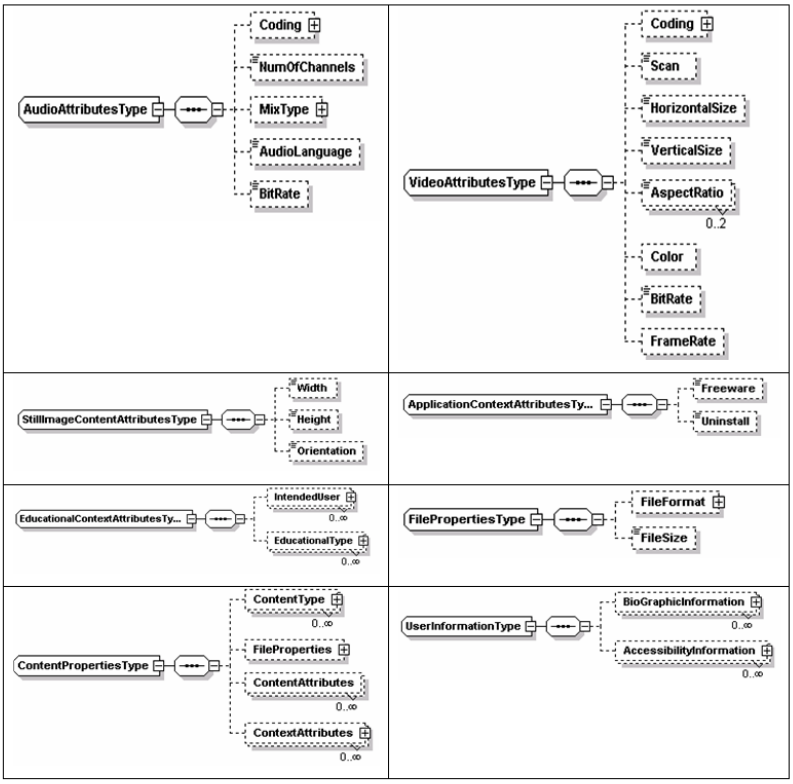


Figure 2.12: TVAnytime metadata model summary

It is important to remark that in the areas of content packages and targeting TVA relies on the data types defined by the family of standards MPEG-21. These data types can be checked in the Digital Item Adaptation (DIA) (ISO, 2007) specification and the Digital Item Declaration (DID) (ISO, 2005b) specification.

As the previous standards, TVA generates a schema where the agreed concepts and relationships are included. Due to the size of this schema, we include in the Figure 2.12 the parts of the schema more aligned with this thesis.

2.3.7 MPEG-7

In this section we tackle one of the most mentioned multimedia metadata models in the literature: MPEG-7. Due to the relevance of this standard in the context of this thesis, we review existing approaches that customize and query it and the available ontologies that model it.

Description

The ISO/IEC standard Multimedia Content Description Interface (MPEG-7) (Bailer and Schallauer, 2008; ISO, 2001) has been defined as a format for the description of multimedia content in a wide range of applications. MPEG-7 provides a powerful combination of description schemes (DS) and descriptors (D). MPEG-7 descriptors represent properties of the content description, while description schemes are containers for descriptors and other description schemes. For the definition of description schemes and descriptors the Description Definition Language (DDL) is used, which is an extension of XML Schema. MPEG-7 descriptions can be either represented as XML or textual format named TeM or in a binary format named BiM.

A core part of MPEG-7 are the Multimedia Description Schemes (MDS),which provide support for the description of media information, creation and production information, content structure, usage of content, semantics, navigation and access, content organisation and user interaction. The structuring tools are especially flexible and allow the description of content on different levels of granularity. In addition to this, the Audio and Visual parts define low and middle level descriptors for these modalities. Figure 2.13 shows the Multimedia Description Schemas classified according to the functionality they implement. At the lower level of that Figure, the basic elements (schema tools, mathematical structures, linking structures ...) can be found. In the middle left part we see the content management and content description tools, which are presented here as separate entities. However, they are interrelated and may be partially included in each other. Then we have the sets of tools related to the organization of the content, navigation of the content and the interaction with the user.

MPEG-7 profiles and the DAVP profile

MPEG-7 covers a very wide scope. In order to tackle the needs of some specific applications, the standard allows the profiling (ISO, 2005a; Troncy et al., 2006). Profiles have been proposed as a means of reducing the complexity of MPEG-7 descriptions. As specified in the standard, the definition of a profile consists of three parts, namely: i) description tool selection, i.e. the definition of the subset of description tools to be included in the profile, ii) description tool constraints, i.e. definition of constraints on the description tools such as restrictions on the cardinality of elements or on the use of attributes, and iii) semantic constraints that further describe the use of the description tools in the context of the profile.

This profiling is a very interesting issue from the scope of the work we present here, since allows the retrieval system developers to use this standard either fully or partially. Some of the profiles that have been standardized are:

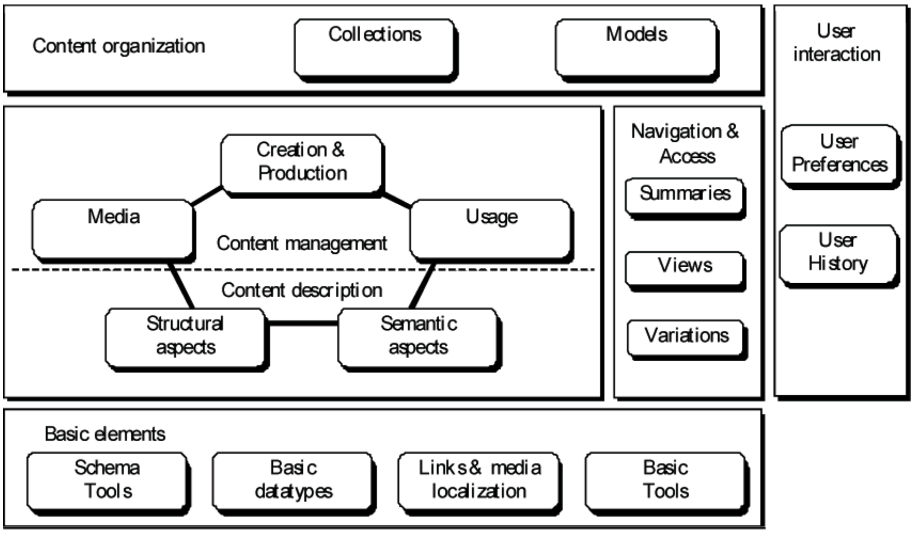


Figure 2.13: Multimedia Description Schemes

- *Simple Metadata Profile (SMP).* Describes simple metadata tagging for single instances of multimedia clips.
- *User Description Profile (UDP).* Describes the personal preferences and usage patterns of users of multimedia content. Descriptions of users' preferences enables automatic discovery, selection and recommendation or recording of multimedia content.
- *Core Description Profile (CDP).* Describes general multimedia content such as images, videos, audio and collections. Furthermore, content collections spanning various media types can also be described.

Following the same methodology, Bailer and Schallauer (2006); Bailer et al. (2007) have provided a Detailed Audio Visual (DAVP) profile focused on the audiovisual production, archiving, search and retrieval and media monitoring. It allows describing image, audio, video and audiovisual content.

The following are some of the key features of this profile:

- *Structural description of the content.*The scope of a description may vary from whole media items to small spatial, temporal or spatiotemporal fragments of the media item. The definition of these fragments must be flexible enough, to allow fragments that are based on audiovisual features (such as image regions representing objects or shots of a video), any higher-level features (e.g. scenes in a video) or manually defined by an annotator. This includes descriptions of different kinds of modalities, descriptions produced with different tools, such as results from automatic content-analysis, semantic interpretation and manual annotation. The latter are mainly in textual form, but it is nonetheless beneficial to structure these instead of having simple free text annotations.

- *Description of visual and audio features and signal properties.* In many search and retrieval systems, query by example is an important query paradigm. As a prerequisite, the content description must include visual/audio feature descriptions. These feature descriptions may also be required for semantic information extraction algorithms. Many approaches rely on the low and middle level features that can be extracted automatically from audiovisual content. Especially in the archive application area, the description of the condition of the audiovisual material (e.g. using the audio signal quality descriptor) is an important requirement. Media, creation and usage information: These kinds of metadata, which are usually global in the sense that they refer to a complete content, are commonly used in the envisaged application areas and often the only ones available in legacy metadata information.

- *Summaries.* Efficient browsing and visualization of descriptions of multimedia content is an important requirement in many applications. Summaries, used in connection with the full content descriptions, are a very valuable tool for this purpose.

From our point of view, DAVP is a key contribution in the context of the work we present here. It is based on an open solution and fully focused on the implementation of the content-based features to enable multimedia semantic retrieval. It is also remarkable that it favours the usage of a model promoted by the scientific community by simplifying the usage of the big set of descriptors provided by the whole MPEG-7.

MPEG-7 ontologies

MPEG-7 is mainly expressed in XML, leaving part of the semantics implicit in the language. There is a significant number of works in the literature that aims to formalize the MPEG-7 descriptions in a more powerful language to enhance the interoperability aspects of the metadata.

The work of Dasiopoulou et al. (2010) provides a recent roundup of this set of efforts, summarized in the table 2.1

Querying MPEG-7

The definition of a query language for MPEG-7 has been a very active field in the last period (Adistambha et al., 2007; Döller, 2008; Gruhne et al., 2007).

Multimedia Ontology	Representation Language	MPEG-7 Coverage	Ontology Design	Application Context
Harmony	OWL Full	structure, visual	monolithic	analysis & annotation
aceMedia	RDFS	structure, visual	modular	analysis & annotation
SmartWeb	OWL	structure, visual	modular	analysis & annotation
BOEMIE	OWL DL	structure, visual & audio	modular	analysis & annotation
DS-MIRF	OWL DL	entire MDS,	modular	mpeg7 xml to rdf
Rhizomik	OWL DL	entire MPEG-7	monolithic	mpeg7 xml to rdf
COMM	OWL DL	structure, visual	modular	analysis & annotation

Table 2.1: Summary of the state of the art MPEG-7 based multimedia ontologies

This activity has been reinforced and promoted by the MPEG standardization committee, by instantiating a call for proposal (N8220) for an MPEG-7 query format (MP7QF) and specified a set of requirements (N8219) to unify the access to MPEG-7 multimedia databases. This access is based on a set of precise input parameters for describing the search criteria and a set of output parameters for describing the result sets. This is understood as MPEG-7 query format or MP7QF.

According to the requirements set by the standardization committee, MP7QF format will be determined by an *Input Query Format*, an *Output Query Format* and some *Query Management Tools*. Figure 2.14 provide an example of an input query proposed by Döller (2008) to express a query by example request based on the predominant color of the asset.

Therefore, if the work in progress described in the mentioned literature goes on, in a short period of time, the standard will help the community to make progress towards interoperable multimedia databases. According to our vision, this will neither immediately nor totally replace the retrieval engines of the companies. However, in our opinion, it will help to complement those systems with content-based features.

2.3.8 SMPTE Descriptive Metadata

The Society of Motion Picture and Television Engineers (SMPTE) started working on the handling of metadata in 1998, trying to harmonize the exchange of Media assets as bit streams.

The SMPTE Metadata dictionary is described in the SMPTE Standard 335M,

```
<m7qf:Query timeout="200">
 <m7qf:Input>
   <m7qf:RsPresentation maxPageEntries="5">
     ()
   </m7qf:RsPresentation>
   <QueryCondition>
     <m7qf:Feature id="dc1">
       <m7qf:VisualD xsi:type="DominantColorType">
         <ColorSpace type="RGB"/>
         <SpatialCoherency>28</SpatialCoherency>
         <Value>
           <Percentage>12</Percentage>
           <Index>1 1 1</Index>
           <ColorVariance>1 1 1</ColorVariance>
         </Value>
       </m7qf:VisualD>
     </m7qf:Feature>
     <m7qf:QueryExpression>
       <m7qf:SingleSearch xsi:type="m7qf:QueryByFeatureDescriptionType"
           featureIDREF="dc1" exactMatch="false" />
     </m7qf:QueryExpression>
   </QueryCondition>
 </m7qf:Input>
</m7qf:Query>
```

Figure 2.14: Query by Example based on an MPEG-7 descriptor

standardized as Recommended Practice 210 (last review in 2007) (SMTPE, 2007).

The Dictionary lists more than 1700 metadata elements, being each entry uniquely registered. The items of the list are tagged as “nouns”, “adjectives”, “adverbs”, “verbs” and so on and can be used to create what they call annotation “sentences”.

The Dictionary includes items for different uses. For example, it lists items that identify “The absolute position of the subject depicted in the essence”, or The Digital Geographic Information Exchange Standard (DIGEST) geo-referenced coordinate system used at image capture” or “An unordered list of strong references to Rights sets”, “Enhancement or modification to the audio essence”, “Information about data essence compression”. Through combination of such diverse but well categorized items, the final metadata is generated.

As can be seen, the dictionary covers very different topics that are related to the scope of this thesis work. They are grouped into identifiers and locators, administration, interpretive, parametric, process, spatial-temporal and experimental.

Content class	# of containers	# of subcontainers	# of elements	Description
PBCoreIntellectualContent	9	0	16	intellectual content of a media asset or resource
PBCoreIntellectualProperty	4	0	7	creation, creators, usage, permissions, constraints,...
PBCoreInstantiation	1	3	28	nature of the media asset
PBCoreExtensions	1	0	2	Extensions crafted by organizations outside of the PBCore Project

Table 2.2: Description of PB Content Classes

2.3.9 PB Core

According to the promoters of PB Core (White et al., 2003) the PBCore (Public Broadcasting Metadata Dictionary) was created by the public broadcasting community in the United States of America for use by public broadcasters and related communities. Initial development funding for PBCore was provided by the Corporation for Public Broadcasting.

The PBCore is built on the foundation of the Dublin Core (ISO 15836), an international standard for resource discovery, and has been reviewed by the Dublin Core Metadata Initiative Usage Board.

At the reporting period, PB Core has 53 elements arranged in 15 containers and 3 sub-containers, all organized under 4 content classes, which are reflected in Table 2.2.

2.3.10 MXF-DMS1

The Material Exchange Format (Kienast et al., 2007; Wilkinson, 2003) is now an established SMPTE standard (SMPTE standards 377M-394M), defining the specification of a file format for the wrapping and transport of essence and metadata in a single container. Material Exchange Format is an open binary file format targeted at the interchange of captured, ingested, finished or almost finished audio-visual material with associated data and metadata. It was designed and implemented with the aim of improving file-based interoperability between servers, workstations and other content devices. MXF files are efficiently stored on various types of media and transported on several transportation links. MXF has got strong support in the industry which can be seen by a number of

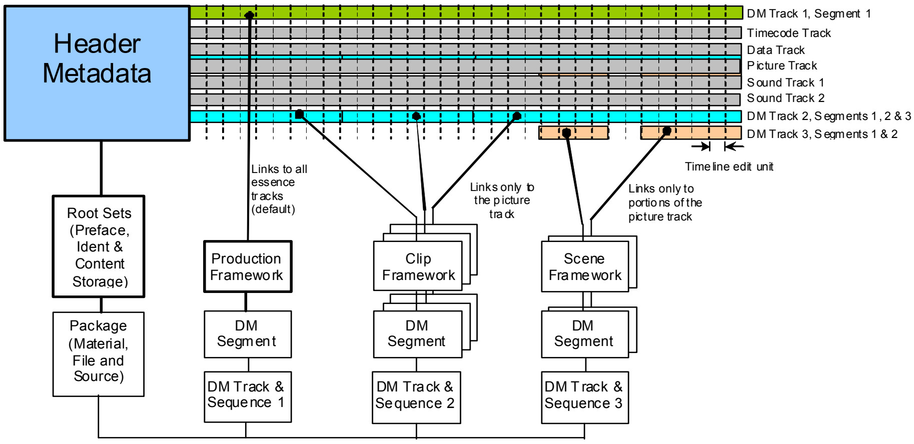


Figure 2.15: Descriptive Metadata Frameworks and their Relationship to the Content of an MXF File Body

available software SDKs described in this short summary.

Support for technical metadata is built directly into MXF specification. In order to provide enough flexibility to deal with different kinds of descriptive metadata, a plugin mechanism for descriptive metadata (DM) is defined. The description of the DM can be found in the structural header metadata (see Figure 2.15). Each item has associated with it a descriptor of the essence. Descriptive metadata tracks may be related to a timeline, an event or a static track.

Descriptive metadata schemes are identified by universal labels in the value field of the DM schemas batch properties. This is recorded in the preface set of the structural metadata. In this ways decoders can detect the existence of descriptive metadata on an early stage and can react, if it is a known schema.

The MXF-DMS-1 (formerly know as Geneva Scheme) is the most important schema in this context. In the DMS-1 specification, several parties agreed upon the terms "production", "clip", and "scene". According to the standard,

- **Production Framework:** Compiles the set of metadata related to the identification and ownership details of the audio-visual content. MXF considers that during the "Production", the metadata is always related to the complete MXF file. Thus, this framework does not deal with the annotation of partial segments. Some of the metadata sets included in the Production Framework are: Publicator, Awards, Classification, Contract, Image Format, Caption description, Episodic Item, Setting Period and Rights.

- **Clip Framework:** As Figure 2.16 summarizes, the Clip Framework Handles the sets of descriptive metadata and the properties related to the caption and creation of information of the independent audio-visual clips. In MXF a clip is an essence container that may comprise of a number of interleaved

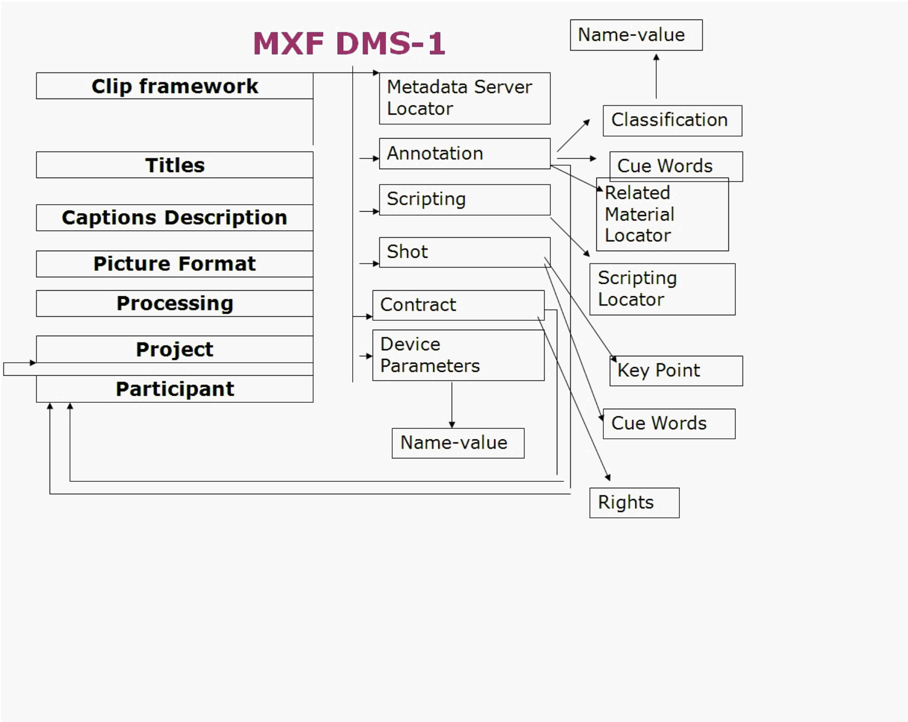


Figure 2.16: Summary of Clip Framework Schema

audio, video, or data essence elements. Some of the metadata sets covered by this framework are: Shot, Device Parameter, Rights, Classification, Scripting, Processing and Project.

- **Scene Framework:** Contains descriptive metadata sets and properties that describe actions and events within individual scenes of the audio-visual content. Scenes may overlap and they may relate to a point in time rather than having duration.Some of the metadata sets are Setting Period, Annotation, Person, Organisation, Location, Shot, and Titles.

The combination of the above logical frameworks (production, scene and clip) constitutes a powerful and shared descriptive annotation model that has been adopted by manufacturers of professional equipment over the whole workflow.

2.3.11 Extensible Metadata Platform XMP

The Extensible Metadata Platform (XMP) (Adobe-Systems-Incorporated, 2008) is a specification promoted by Adobe for the creation, processing, and interchange of metadata, for a wide variety of applications. The specification is composed of three parts. The first one is related to the representation model of basic metadata. It takes care of the way the metadata can be organized and the serialization of

Property	Value type	Category	Description
xmpRights:Certificate	URL	External	Online rights management certificate.
xmpRights:Marked	Boolean	External	Indicates that this is a rights-managed resource.
xmpRights:Owner	bag ProperName	External	An unordered array specifying the legal owner(s) of a resource.
xmpRights:UsageTerms	Lang Alt	External	Text instructions on how a resource can be legally used.
xmpRights:WebStatement	URL	External	The location of a web page describing the owner and/or rights statement for this resource.

Table 2.3: XMP Rights Management Schema

the model into RDF. The second part provides a detailed set of properties and descriptions for the metadata schemas, including general-purpose schemas such as Dublin Core, and special-purpose schemas for Adobe applications such as Photoshop. From the perspective of this thesis work the most interesting issue is that it also provides information on extending existing schemas and creating new schemas. Finally the third part covers technical details about the packaging and storage of the serialized metadata.

XMP already proposes a set of standarized schemas. Some of them are generic (e.g. Dublin Core Schema, Rights Managment schema summarized in Table 2.3, Media Management Schema, Dynamic media schema and so on) and some of them are specialized (e.g. Adobe PDF Schema, Camera raw schema and Exif schemas among others). Regarding the extensibility of the schemas, the standard allows either to create an schema from the scratch or to increase/modify any existing schema. The metadata guidelines (Metadata-Working-Group, 2009) proposed by the Metadata Working Group promoted by a core group of key companies in the media sector provide a deep overview of the relationship between XMP with other key standards.

2.3.12 Other Standards

In the following paragraphs we include some metadata definition initiatives less relevant either due to its disparity with the context of this thesis or due to the lack of usage. However, their inclusion helps the global understanding of the status of metadata models for the multimedia professional environments.

Advanced Authoring Framework - AAF The Advanced Authoring Format (AAF) (AAF, 2004; Kienast et al., 2007), promoted by the Advanced Media Workflow Association (AMWA), is a format defined for the professional exchange of information in the domain of the video authoring and post-production. It is composed by a set of specifications either adopted by the SMPTE or in the process of being adopted.

The project is divided into three main activity lines: specifications of the interfaces, specification of the applications and data models. This last line is the most relevant to the context of this thesis work. However, it is still a work in progress.

Marc21 (Ahronheim, 2002) is the acronym for Machine-Readable cataloguing. It defines a data format that emerged from a Library of Congress-led initiative that began thirty years ago. It provides the mechanism by which computers exchange, use, and interprets bibliographic information, and its data elements make up the foundation of most library catalogues used today.

IFTA[4], International Federation of Television Archives published in 1992 a minimum data list of 22 fields for cataloguing broadcast materials. There are three main areas that group the fields: Identification (8 fields), Technical (9 fields) and Legal (5 fields).

International Press Telecommunications Council (IPTC) schemas The IPTC has developed a set of standards related to the niche of the press industry. The standards mainly related to the exchange of news, covering different issues as its structure, exchange aspects and so on. For some specific topics (e.g. sports, events, ...) specific specifications have been added to the family of standards(IPTC, 2009).

2.3.13 Criteria to choose the best standard

In spite of the existence of multiple standards for the creation, manipulation and querying of the metadata, most of the proprietary solutions in the media industry are proprietary or customized solutions (Multimedia-Research-Group, 2004). We share Bailer and Schallauer (2008) opinion about that one of the possible answers to this situation is that in most of the cases, any standard covers all the needs neither of a system nor of a company.

Our approach therefore is based on a dynamic combination of different models, either standard of proprietary. But, regarding the selection of those standards, which are the key criteria that must be taken into account?, and how suit the mentioned standard fit those criteria?. The definition of the scope of the usage of the model is a key criterion. As there is not a unique model that is the best for all the purposes, it is very important to defined a weighted list of requirements for the model. In the following paragraphs, as a summary of this section, we provide some of the requirements that may compose that weighted list.

[4] http://www.fiatifta.org

- *Domain.* The domain of the system that will employ the model is obviously a key parameter for the selection of the domain. However, the more close is the model to a specific domain (e.g. MARC-21 for libraries, IPTC standards for news) the more difficult is to employ it for other domain. The main barriers usually are that either a significant part of the model becomes irrelevant or that a big part of the model is not flexible enough to cover the requirements of this domain. The XMP model is an example of the opposite scenario. It has been designed to be employed in a very open range of applications and domains.

- *Degree of the search.* Obviously, all the standards cover in a sufficient manner the identification of the assets that allow its searchability. However, all P/Meta, MPEG-7 and DMS-1 allow to include many descriptive elements (i.e. Editorial Title, Publishing title,...) either for the whole asset of for parts of it (e.g., shot). This feature enables to implement more powerful or deep search algorithms.

- *Relation with third parties / applications.* The application that will integrate the model, may interact with external entities (e.g., organizations) or other components in the organization (e.g., billing system). Some standards optimize or facilitate that interoperability. In some cases due to the intrinsic definition of the standard (e.g. the standards defined for news exchange by IPTC) and in other cases due to the adoption of the standard by the global industry (e.g. DMS-1 standard of MXF).

- *Management of rights.* In many multimedia applications, the management of the rights is a key aspect. Concerning this issue, some of the standards that handle this feature with more powerful mechanisms are P/Meta and BMF. As Figure 2.3 shows, XMP is an example of non powerful modeling for the management of rights.

- *Content-based.* In those cases where the application that employs the model requires the management of features derived from the content analysis, there are some models that will provide better support than others. If the application needs to handle the low level features extracted during the analysis process, MPEG-7 standard is undoubtedly the best option (Sikora, 2001). If the application needs to handle middle-high level features but related to a very diverse set of units of the assets (i.e. the whole asset, a shot, a frame, a segment of a frame, an audio fragment...), other standards as DMS-1, with complex resources for asset decomposition, can be complete enough to cover the requirements.

- *Degree of innovation in the search.* When the approach for the search requires the employment of techniques emerging from the scientific community, the selection of an standard as MPEG-7, due to its flexibility and predominance in the scientific literature, is the best option.

2.4 Content-based retrieval, a multidisciplinary challenge

As has been stated in the introductory chapter (see Chapter 1), one of the main objectives of this thesis work is to facilitate the integration of the content-based functionalities in the current multimedia retrieval system.

The existing work regarding the implementation of such functionalities is a very wide field that covers multiples disciplines and scientific communities. The compiling of the main approaches for each of the content-based functionalities that can be integrated in a retrieval workflow it is out of the scope. However, due to the relevance of this work and its closeness with the aim of our middleware, in the following paragraphs we provide a short summary of the state of the art. In order to do this, we highlight some key references that provide a global overview.

Content-based multimedia technology

As Naphade and Huang (2002) stated, the extraction of the Semantics from Audiovisual Content is perhaps the final frontier in Multimedia Retrieval. With this statement we are not arguing that the analysis of the multimedia is the only content-based functionality to be implemented, but the most important. The progress made in the automatic or semi-automatic content based analysis, allows the development of other content-based functionalities that promotes the progress towards the content-aware multimedia information retrieval systems.

The book of Zhang (2006) provides a complete overview of the content-based multimedia paradigm. The work is organized to tackle the complexity of moving from features to semantics, the image and video annotations, the new mechanisms of human computer interaction and the models and tools for semantic retrieval.

The work of Lew et al. (2006) is a significant contribution to the comprehension of the state of the art in Multimedia Information Retrieval. They review more than one hundred recent articles on the topic, discussing their role according to the research directions. Their work covers browsing and search paradigms, user studies, affective computing, learning, semantic queries, new features and media types,high performance indexing, and evaluation techniques.

In our opinion, a key aspect to understand the present and future of the Multimedia Information retrieval is to understand the semantic gap problem. According to Smeulders et al. (2002), the semantic gap can be understood as "the lack of coincidence between the information that one can extract from the visual data and the interpretation that the same data have for a user in a given situation". Haas et al. (2007) provide a survey on the work done and to be done to reduce this semantic gap is provided. Hauptmann et al. (2007) provide a nice dissertation about how the management of a big set of high-level concepts in a retrieval system will help to the reduction of such semantic gap.

The book edited by Stamou and Kollias (2005) provides a very complete summary of the efforts made by the community to employ the MPEG standards (MPEG-7 and MPEG-21) and the tools coming from the semantic web to perform the semantic analysis required for the multimedia information retrieval.

One of the most active research fields in multimedia retrieval is the one related to the automatic indexing of the assets. The comparison between the early reports of Brunelli et al. 1996; 1999 and the latest surveys on the field (Divakaran, 2008; Snoek and Worring, 2005) show the progress achieved during the last decade. This progress has led the community to the implementation of more advanced techniques for multimedia retrieval (Datta et al., 2008; Eidenberger, 2003).

Other fields that are also having an active role towards the content-based multimedia retrieval paradigm are the query parsing and expansion (Bhogal et al., 2007) approaches, the knowledge representation theory(Bloehdorn et al., 2004) and ranking and re-ranking techniques (Natsev et al., 2007; Ruthven and Lalmas, 2003).

Although not fully focused in multimedia, the PhD report of Styltsvig (2006) provides a deep overview of the employment of ontologies for the semantic improvement in retrieval techniques.

Going through the different works presented and their achievements results provides a very optimistic scenario where the progress of the scientific community is prepared to be fully or at least partially migrated to real retrieval workflows. As was stated in the introduction chapter, these workflows have very pressing needs with respect to the integration of content-based functionalities. In the following chapters we describe how some of these features are integrated in two retrieval workflows.

Part II

CONTRIBUTION OF THE THESIS

3 Semantic Middleware to enhance multimedia information retrieval systems

This chapter compiles our contributions related to the definition of a semantic middleware to enhance the MIR systems. In order to achieve this we firstly define a reference model for the whole MIR process. Then, we locate this thesis work within this reference model. Then we describe an architecture to implement that middleware and finally, we define a set of key criteria to correctly deploy such architecture into a real system.

3.1 Multimedia Information Retrieval Reference Model

In order to contextualize this thesis, in this section we establish a reference model for information retrieval. This is done to clarify the role and the relationship of the middleware that we present with the other components of a retrieval system. This is a key issue, since, as we have already stated, we assume that the implementation of content-based functionalities implies, in different steps of the retrieval process, the management and understanding of the semantics of the domain. According to this, **we conceive the proposed semantic middleware as the main provider of the semantic information and services required by the different components and applications, existing or to be added, in complex multimedia information retrieval systems**. And these applications and modules may perform very different tasks in the retrieval process.

The reference model that we present here, is an extension of the adaptation made by Ray Larson of the model proposed by Dagobert Soergel. In Soergel (1974) proposed a model that he called ISAR: Information Storage and Retrieval system (see Figure 3.1). Under this definition, he included those systems whose aim was the retrieving of a document (or another retrieval object) relevant to a concept, assuming that a concept could be a complex entity that incudes the relationships among different concepts and terms. He distinguished 2 lines, the searching line and the storage line.

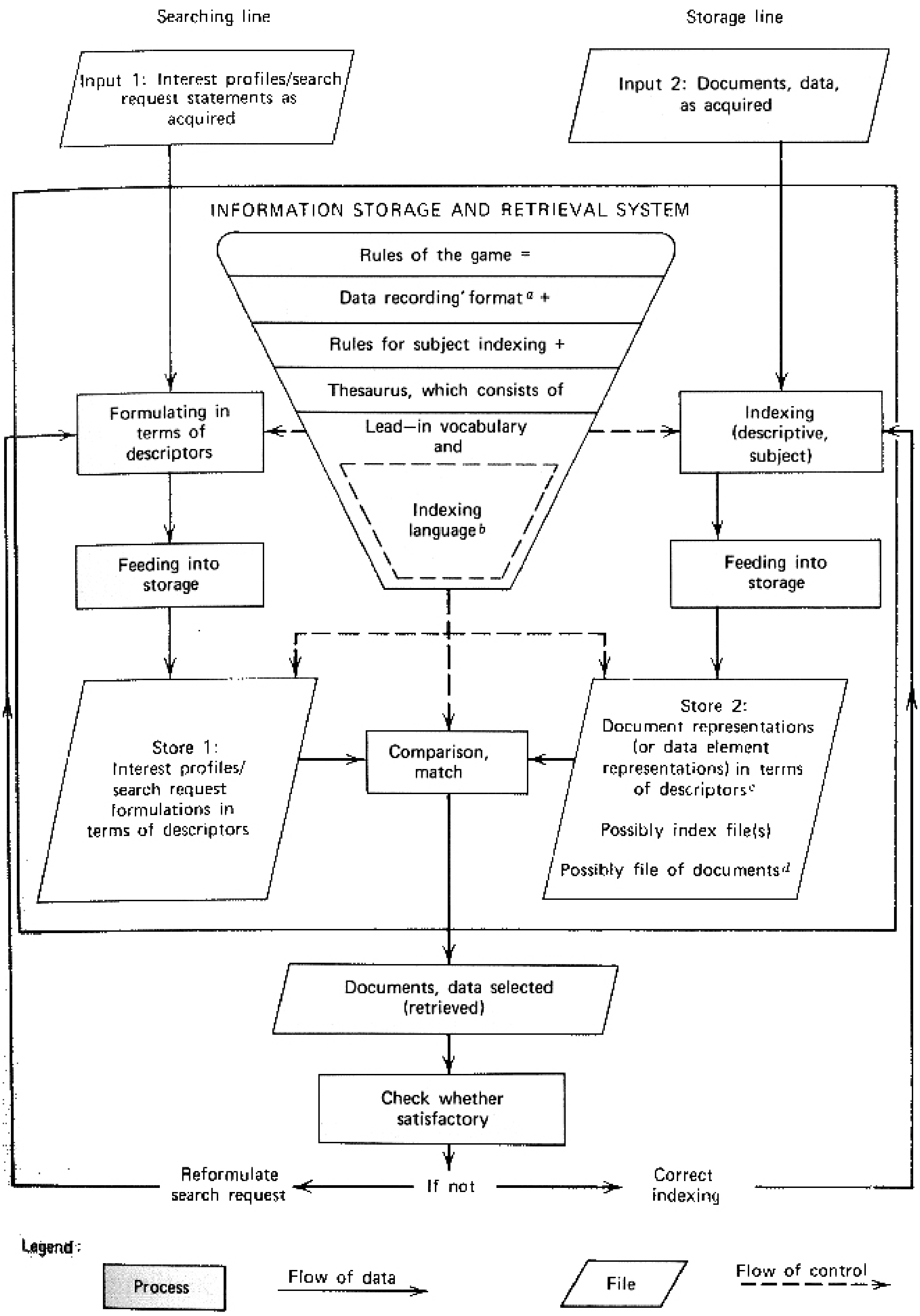


Figure 3.1: Information Retrieval and Storage Reference Model by Soergel

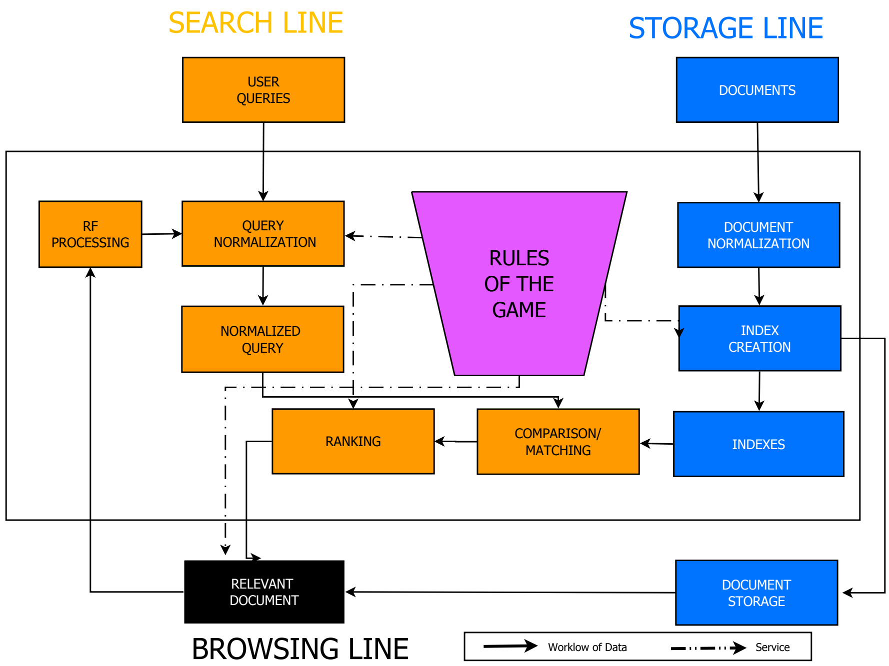


Figure 3.2: Information Retrieval Reference Model

Since that work, Ray Larson from Berkeley has done several adaptations in order to update the model to the current state of the art. In Figure 3.2 we include a graphical representation that combines the modifications performed by Larson (Larson, 2007, 2010) and our contribution. Some of the main Larson contributions are the simplification of the model and the inclusion of the relevance feedback functionality. Moreover, in order to include the distinction between information retrieval and information browsing proposed by Baeza-Yates and Ribeiro-Neto (1999), we have added the *Browsing Line* in that model.

From the perspective of this thesis work, the semantic middleware that we propose occupies the central block named "Rules of the Game". The arrows that leave this block represent the semantic services that can be provided to the different processes of the information retrieval. The services are represented as broken arrows because they are not mandatorily provided. In complex systems, each process may imply one or more modules and a module may occupy several processes. Table 3.1 contains some examples of those services that this middleware is aimed to provide or at least semantically support. For each example (e.g. query processing support), relevant literature is referenced.

Table 3.1: Examples of semantic services to be provided in a MIR system

Process	**Support service offered by the middleware**
Query Normalization	Support for query building (Baer et al., 2001) Query processing (Arens et al., 1996) Query expansion (Bhogal et al., 2007) Query mapping and federation (Tzitzikas et al., 2002)
Index creation	Video indexing (Snoek et al., 2007) Text indexing (Köhler et al., 2006) Image indexing (Wang et al., 2006) Multimodal indexing (Simou et al., 2005) Support for manual annotation (Piasecki and Beran, 2009) Semantic modeling of multimedia standards MPEG-7, DMS-1... (Troncy et al., 2006)
Ranking	Semantic based ranking (Shamsfard et al., 2006) Document organization (Kim and Seo, 1991)
Relevant Documents	Recommendation systems (Yu et al., 2009) Concept-based visualization (Luo, 2007) Semantic visualization techniques (Faaborg, 2003) Relevance feedback techniques (Ruthven and Lalmas, 2003)

3.2 Semantic Middleware, a three Layered Architecture

In this section we describe the generic architecture of the middleware that is to play the role "Rules of the game" of the previously presented reference model. This middleware aims to belong to a multimedia storage and retrieval system that integrates some of the content-based functionalities shown in Chapter 2.

Along the section we include a short generic description of the architecture. After this, we provide an analysis of each of the layers that constitute the architecture.

3.2.1 Requirements of the middleware

First of all, based on our experience and the requirements expressed in the two mentioned scenarios, we provide a summary of the objectives and requirements that lead us to propose this middleware architecture.

- *System agnostic.* The middleware is to be deployed in very different systems (e.g., a search engine able to retrieve multimedia assets from different sources, a professional Media Asset Management for high resolution videos in a company, and so on).
- *Domain agnostic.* The middleware may be deployed in very different domains.
- *Do not replace, but enrich retrieval and storage approaches.* The middleware has to play the role of semantic service provider in a system, without replacing key tasks of the system, such as the storage and retrieval of information. According to our humble experience, nowadays this replacement is neither feasible nor advisable in most of the current systems and companies.
- *Seamless integration.* In a coherent view with the previous item, the architecture has to enable such as seamless integration with the existing systems as possible.
- *Clear upgrading and scaling mechanisms* The maintenance of this middleware can employ different people from different disciplines. The architecture should facilitate its maintenance.
- *Generic multipurpose semantic service provider.* As has been reflected in Chapter 2, there are many tools and techniques that provide different content-based functionalities. The middleware obtained after the deployment of the architecture has to be flexible enough to attend very diverse semantic demands from those components.

- *Empower the multimedia indexing processes.*

- *Wide range of semantic knowledge.* From the semantic point of view, the information required by the modules demanding services of the middleware, may belong to very diverse degrees of semantic richness (e.g., vectors belonging to colour histogram, semantic distance between two concepts of the domain, and so on). The middleware architecture has to provide mechanisms for the representation and management of this semantic disparity.

- *Semantic Uniqueness.* The middleware architecture has to end semantic duplicities and semantic redundancy imposed by the employment of satellite applications in a system that may employ their own semantic information (e.g., content-based recommendation module, ontology-based clustering).

- *Ensure Semantic Interoperability.* The middleware is likely to work with components, tools or information sources that employ different formats or languages. The middleware architecture has to allow interoperability and has to be able to react to the upgrades derived from the evolution of each of the mentioned elements.

3.2.2 Middleware Architecture

The architecture proposed for the middleware (see Figure 3.3) is based on a classical approach to the development of software applications: data, business logic, and presentation.

The three layers are as follows:

- **Semantic Middleware Knowledge Base (SMD KB)**: This layer gathers all the semantic information about the domain of the application.

- **Semantic Middleware Intelligence Engine (SMD IE)**: This layer is made up of a set of interrelated software elements that are able to perform atomic operations over the semantic information gathered in the SMD KB.

- **Semantic Middleware Gateway (SMD GW)**: This layer is mainly composed of a light-weighted set of interfaces that offer customized services to the different modules of system.

Following we provide a more detailed description of each layer.

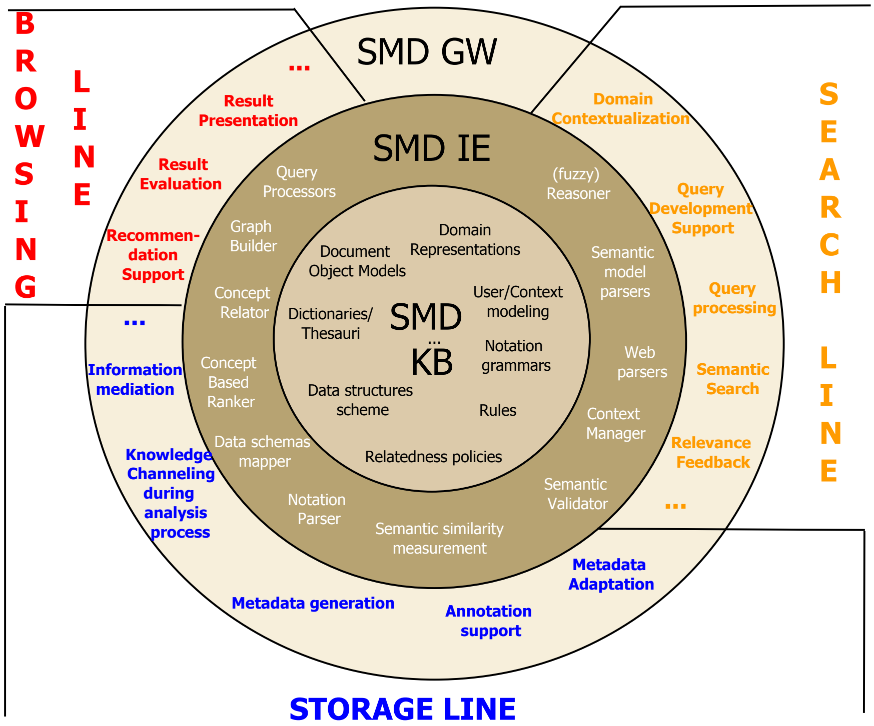


Figure 3.3: SMD three layered architecture

3.2.3 Semantic Middleware Knowledge Base (SMD KB)

This layer groups all the semantic passive resources in the middleware. Any semantic request received by the middleware will imply at least one request to any or several of the resources available in this layer.

Therefore, it is, undoubtedly, the most critical layer of the architecture. It is the key entity to avoid semantic duplicity and is usually highly influenced by work done in the standardization forums.

Some of the abstract entities that may shape this layer can be seen in Figure 3.3. In the following we highlight some of them: Document object models, domain representations, notation grammars, resources to enable the mapping between different terminologies, dictionaries and thesauri, semantic representation of information exchange formats, rules for conducting the multimedia indexing, user and context modeling and polices for the definition of relatedness among documents.

3.2.4 Semantic Middleware Intelligence Engine (SMD IE)

This intermediate layer is in between the SMD KB (a passive layer from the processing point of view) and the SMD GW (the layer that acts as the semantic services front-end).

This layer is in charge of the semantic interoperability and is responsible for the implementation of the logic to be performed on the top of the SMD KB from a service agnostic perspective. According to our definition of this layer, it is composed by a set of software items that perform some specific operation across the semantic information. We call these software items Processing Elements (PEs).

Some examples of PEs functionalities implemented by the PEs named in 3.3 are: Inference of new knowledge, query processing, linkage among concepts, building of partial or full graphical views of the domain, terms translation/mapping, semantic/similarity based ranking algorithms, context managers, format parsers, semantic validation, mapping between data structures, fuzziness reasoning and notation parsing.

This layer is also very likely to embed external developments from the scientific (e.g. fuzzy logic reasoners) and the software community (e.g. APIs for the management of graphs).

3.2.5 Semantic Middleware Gateway (SMD GW)

This is the outer layer and is responsible for the communications with the rest of the components of the information retrieval and browsing system. It represents the front-end employed by the retrieval system to allow the outsourcing of the semantic services. We call those services the Support Processes (SPs). In 3.3 we include, as an example, the SPs implemented by us in the scenarios included in this article. The services have been divided according to the *Search*, *Storage* and *Browsing* lines of our reference model. Among those services we may find SPs that are provided due to the presence and actions of a final user (online SPs) or SPs that enable functionalities carried out before the user arrives in the system (offline SPs).

- Regarding the functionalities offered by the online SPs we highlight the following: Recommendation for the query development, query processing functionalities, support to enrich or simplify the manual annotation of multimedia assets, facilities to enable concept-based graphical navigation, terminology adaptations, semantic support for query negotiation among different information sources and result ranking and document mapping.
- With respect to the offline SPs this layer may offer functionalities for the

following processes: integration of the information provided by the analysis modules during the automatic or semiautomatic indexing of the assets, extraction of new knowledge by applying (fuzzy) reasoning techniques and periodic reporting of potential updates in the representation of the domain in the SMD KB.

In the following section, some key criteria are included for a right implementation of each of the layers of the middleware.

3.3 Key design criteria

In the following subsections we provide some key criteria in order to carry out a correct deployment of a semantic middleware in a specific scenario or system.

3.3.1 Semantic Middleware Knowledge Base (SMD KB)

In order to carry out an optimal design of a SMD KB, the following issues are critical.

- Identification of the domain. The domain is related to the information that is or could be required by the modules of the retrieval system, including the forthcoming ones. It is quite common to try to cover all the domain of the organization, which is to the detriment of other technical criteria such as performance and maintenance.

- People in charge of the knowledge base. The selection of the right people in the organization for the design and maintenance of the knowledge base is crucial. Whenever possible, the establishment of a team of at least two knowledge engineers is the best option. The list of skills of the team must include: deep and global knowledge of the nature of the content handled by the organization and the needs of their users, expertise in semantic formalization, multimedia metadata modeling, technical skills to understand the implications of the design and, depending on the domain, first hand information about the trends and technical road-map of the sector. During the design process a deep relation with experts in semantic techniques in charge of the SMD IE is strongly recommended.

- Technology of the SMD KB. The nature of the technology to be employed in the knowledge base can be different. We may find, among others, simple thesauri contained in text files, object oriented schemas or complex and deeply interrelated ontologies. The nature and complexity of the information system to be enhanced is the key factor to selecting the most appropriate technology. In our experience, in the field of the multimedia information retrieval, ontologies are the best option for the implementation of the SMD KB core. This is due to the following reasons. On the one hand, as we describe in Chapter 2, most of the successful techniques for the implementation of functionalities that make use of the content of the multimedia employ semantic models that are easily reproduced with ontologies. On the other hand, the domain covered by the system is usually composed of different inter-related subdomains (e.g., user context, domain of the multimedia content, internal data formats of the organization). The semantics handled by the ontologies and the ontology mapping facilities are rich enough to support the needs of the SMD KB.

- Composition of the knowledge base. The knowledge base can be composed of a set of interrelated or unrelated items (e.g., a set of mapped ontologies, a separated grammar notation, rule files, and so on). Even considering the maintenance cost of interrelated resources, the ideal scenario should avoid isolated semantic resources. As we state later, a combined solution of both scenarios, having certain resources isolated and the rest deeply interrelated, is also possible.

- Reusing shared knowledge. In many cases information systems are fed by external content, usually coming from a specific sector. The SMD KB should make use, whenever feasible, of the standardization efforts made by the key agents of that sector or by the scientific community.

- Documentation. The generation of the SMD knowledge base must be profusely documented. The meaning of each semantic unit must be unequivocally defined so as to avoid problems in the usage and maintenance of the system. Certain exceptions might be allowed, specially in those cases where part of the SMD KB is automatically enriched by a third party (application, users, etc.).

- Update. The update of the SMD module can be caused by external factors (e.g., upgrade in an standard), or internal factors (e.g., statistical information extracted from the users activity). The key issue here is the definition of the right updating procedures and approaches (e.g., supervised update of the knowledge base, consistency checking methodologies, clear definition of the responsible of each piece of information in the knowledge base and precise mechanisms to detect the impact of any change on the knowledge base in the SMD behaviour). In our experience, the availability of an stable tool to browse the KB pieces and update them is highly recommended.

3.3.2 Semantic Middleware Intelligence Engine (SMD IE)

The key issues for the design and implementation of this layer are the following:

- Maintainability. In order to ensure the maintainability of the layer, PEs should implement atomic operations and should ignore the application status. Therefore, PEs are expected to belong to different abstraction levels, having PEs that, while keeping a service independent philosophy, make use of other PEs to carry out their tasks.

- Exchange information structures. The definition of the information structures to be shared among the PEs is a key issue. The information structures shared among them (e.g., user profile, semantic graph, and document representation) should be designed to facilitate their interconnection. This implies that the structures should be flexible enough to cover the needs not only of the existing services but also of the forthcoming ones. Our recommendation is to employ an object oriented approach since it supports the inheritance between the entities and the scalability. This can result in different implementations: Java objects, XML documents, and so on.

- Making use of available resources. The use of the available tools is highly recommended (e.g., ontology reasoners, graph visualization libraries). The Semantic Web community has been very prolific in the development of tools for parsing, inferring and reasoning over semantic structures (Stamou and Kollias, 2005). If some PEs of the SMD IE employ any external system, the impact of this dependency on the global system should be minimized during the design process. This criterion will lead to a more careful definition of the set of structures employed for the information exchange among PEs, achieving a SMD IE ready for the obsoleteness of certain applications or the appearance of new tools.

- PEs network interoperability. The design of the SMD IE should avoid constraints derived from the multiplatform and multi language implementations. If the performance of one of the PEs requires that is implemented in a different programming language, the SMD IE must provide the communication mechanisms required to ensure the interoperability among them. One of example of this is the work of Pan (2005)), where the author aims to employ the most efficient reasoner.

- Execution synchronization. PEs that carry out tasks of an over-long or unpredictable duration should implement asynchronous communication mechanisms. These PEs can be invoked by other PEs or by the SMD GW and its execution should not block other processing flows.

3.3.3 Semantic Middleware Gateway (SMD GW)

We highlight the following key aspects hidden behind this abstract entity.

- Flexibility. The technology to be employed should be mainly determined by the requirements of the components of the system which the middleware would interact with. This usually leads to the employment of state of the art networking technology (e.g., SOAP).

- PEs invocation. Each SP of the SMD GW should be defined in order to be able to perform parallel invocations of the PEs. This should be done depending on the service required and the configuration parameters on execution time.

- Synchronization and status management of each SP. PEs usually perform operations of unpredictable duration, and, specially for operations performed off-line, they may take too long. The SMD GW must have synchronization and failure detection mechanisms for the PEs requests and each SP must implement procedures to manage and report the status of its task. This allows the integration of the SMD with conventional status monitoring mechanisms of the professional information retrieval system.

- Global status management. Only under very few circumstances should the SPs provided by the SMD GW handle information about what is happening in the system. Although it is quite common that a Support Processor of the SMD GW is kept alive and invoked several times during a specific operation (i.e., the automatic annotation performed on the same multimedia asset by multiple analysis modules) this is not recommended. In spite of this, there should exist a third component -either from the IR system or external- that would handle this status and the signalling to be exchanged with the SMD GW to ensure that the SP is performed correctly. Otherwise the complexity integration of the SMD GW would increase significantly and the approach would not fit the philosophy behind the whole middleware architecture.

4 Other contributions

This chapter summarizes two minor contributions derived from the work performed during the design, implementation and validation of the semantic middleware described in the previous chapter.

Those contributions, described in the sections of this chapter, are summarized in the following:

- We have developed, to our knowledge, the first OWL implementation of the DMS-1 standard metadata schema. Section 4.1 provides a deeper explanation of this minor contribution.
- The semantic middleware architecture and its integration in the WIDE system was employed to contribute to the validation and completion of the image search architecture of the Jpsearch JPEG normative (Dufaux et al., 2007) during the standardization process. Section 4.2 details this contribution.

4.1 Implementation of the first DMS-1 OWL ontology

In the deployment of the semantic middleware performed in RUSHES, following the criteria expressed in Section 3.3, the SMD knowledge base was designed by interrelating different ontologies in order to represent the domain of the retrieval application. The core or central ontology was one developed by us that represented partially the DMS-1 standard (see Section refstar:dms1), which is part of the Material Exchange Format (MXF) standard defined by the broadcast industrial community. The model was implemented in OWL (Grau and Motik, 2008) using the Protégé OWL editor (Knublauch et al., 2004).

The motivation behind this ontology was to implement the descriptive structure of the multimedia assets and the metadata about them. In 4.1 the reader may see a partial view of the implemented standard in RUSHES.

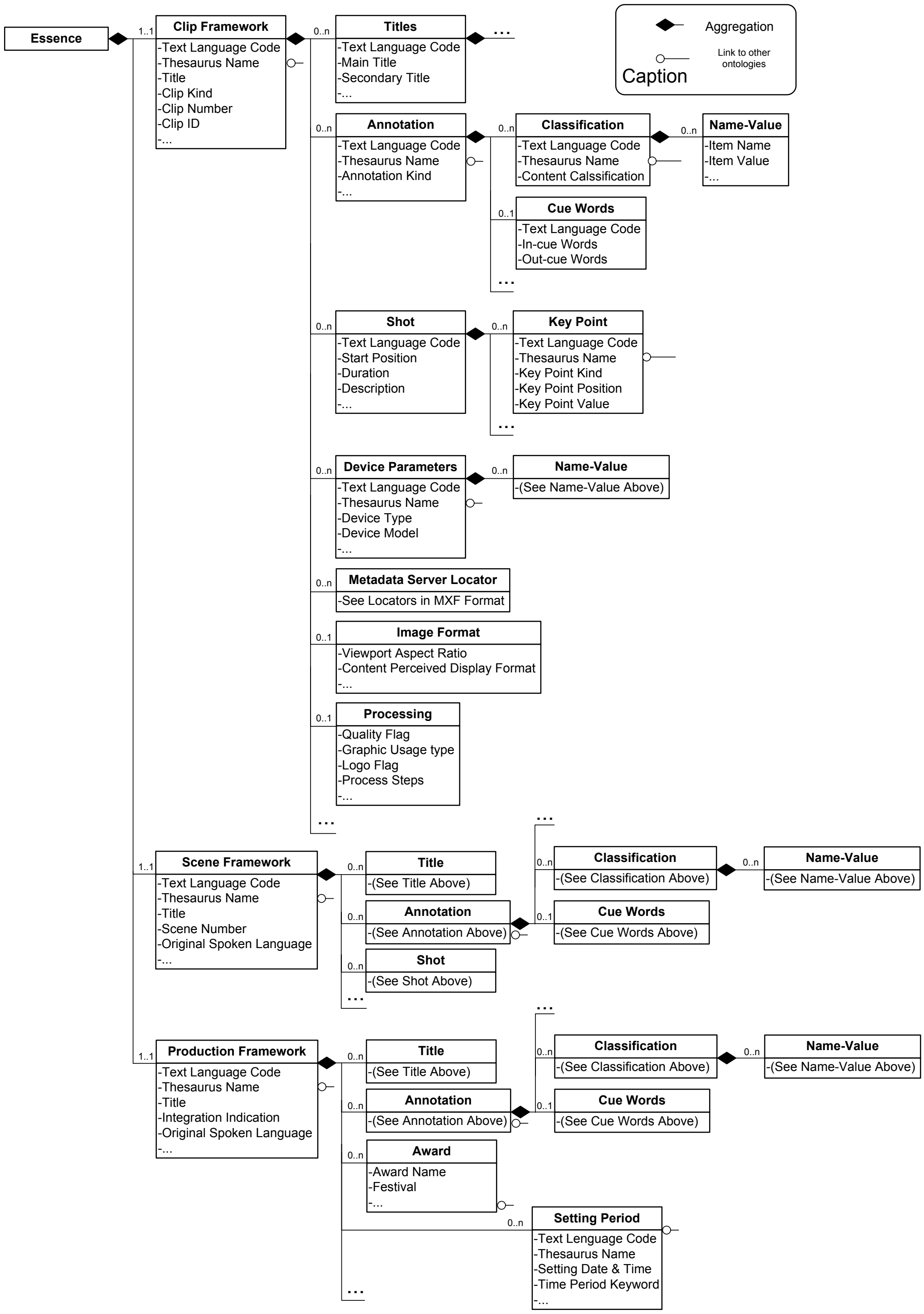


Figure 4.1: Fragment of the implemented DMS-1 schema based on the aggregation relation

As has been stated before, the standard groups the annotations according to three logical entities: the production perspective of the asset, the asset as a clip or continuous essence element or element interval and, finally, the different scenes or "actions and events within individual parts of the audiovisual content". For each one of the scenes, different metadata sets of properties were defined. For each of them, the cardinality and mandatory level were provided.

The sets of metadata and properties can be applied to various frameworks (Production, Clip and Scene) and their nature is diverse. For example the standard provides sets of metadata about the titles of the asset, awards, events, information about the device parameters, file, formats and rights among many others.

Regarding the temporal decomposition, the ontology, accordingly to the standard, models also the concept "Shot". This entity is related to some of the frameworks and allows to define specific annotations for a shot of any of the essences of the asset (video, audio, and so on). For each of the shots, the duration, start position, and description are provided. And if needed, keypoints or some extra annotations can be attached to the shot.

In order to add the specific metadata, the ontology models as a concept what the standard defines as the property set named "Annotation". This concept is directly linked to different thesauri. This feature makes the model a powerful tool to capture the metadata for the different pieces of the asset.

A deeper description of this ontology can be found in (Marcos et al., 2009).

4.2 Contribution to the validation of the general framework and architecture of JPSearch standardization activity

JPSearch is an standardization activity promoted by the JPEG (Joint Photographic Experts Group) forum that as Dufaux et al. (2007) summarizes aims to provide a standard for interoperability for image search and retrieval systems. More specifically, it targets the definition of interfaces and protocols for the exchange of data between systems and devices.

The Part-1 of JPSearch, named "Framework and system components" provides a global view of JPSearch (ISO/IEC, 2007). This part, after reviewing the new role of the user, the definition of a set of use cases and describing the overall search and management process, describes the introduction of the JPSearch architecture. The proposed architecture, shown in Figure 4.2, is composed of 4 layers: user layer, query layer, management layer and content layer.

As part of the standardization activity, the working group launched a call for

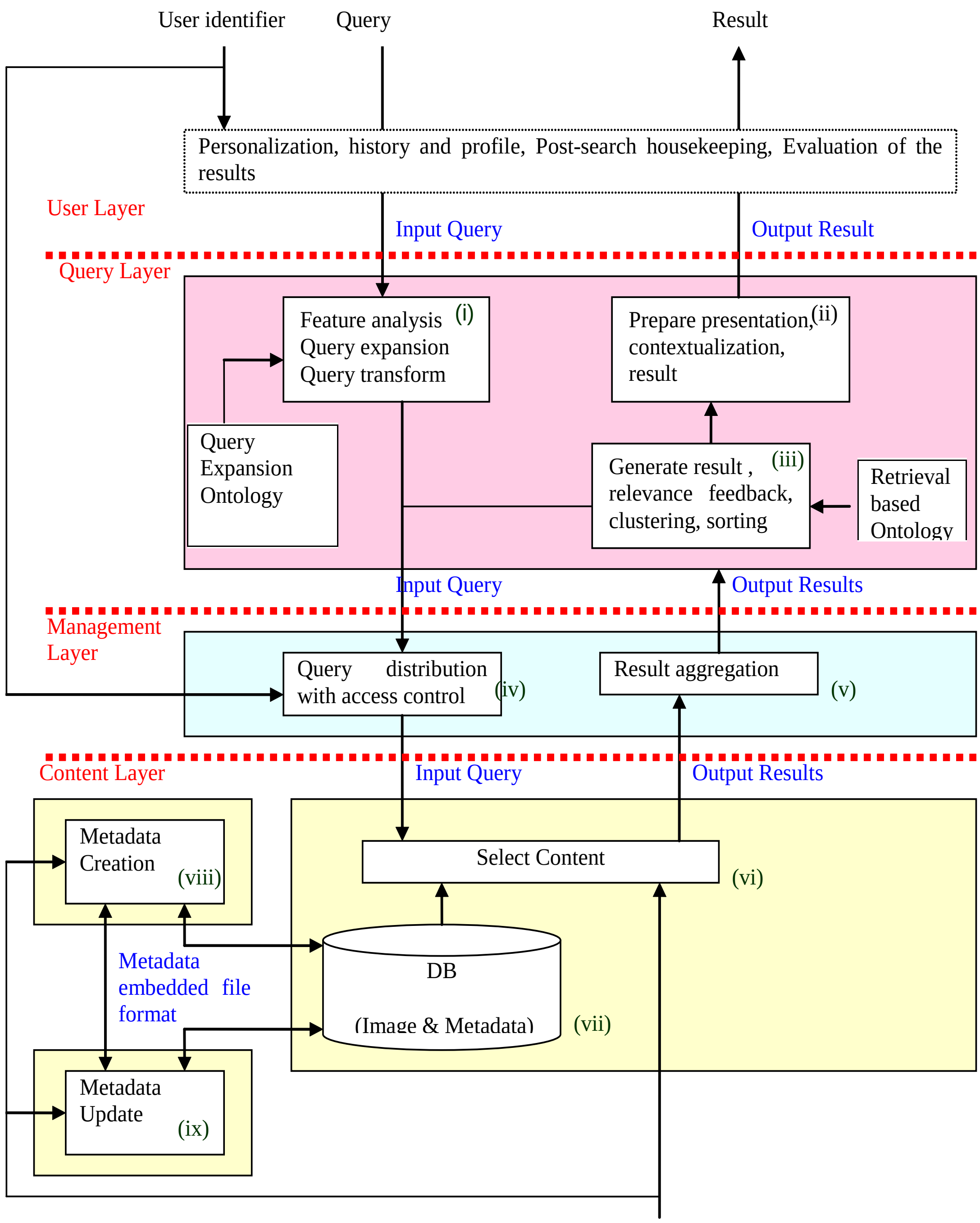


Figure 4.2: JPSearch Architecture

technologies to validate the proposed architecture with respect to innovative and emerging systems that were able to handle the storage and retrieval of images. The WIDE system and specially the deployment of the semantic middleware that we present in Section 3.2 were the basis for the contribution that we sent to the JPSearch standardization working group (Marcos and Jiménez, 2007).

Part III

VALIDATION, DEPLOYMENT IN REAL SCENARIOS

5 WIDE use case: Semantic Middleware for multimedia retrieval from multiple sources used by a multidisciplinary team in a car industry domain

This chapter covers the validation performed of the model presented in Section 3.2 by its deployment on a multisource multimedia retrieval system in the domain of the car industry. With respect to the reference model for content-based multimedia information retrieval that we have presented, WIDE deployment covers the “Search Line” and the “Browsing Line”.

In the following sections we cover the following subjects:

- As an introduction to the WIDE SMD, the whole WIDE system is briefly described. The main motivation behind this is to explain the global architecture of the system. This aims to be helpful to understand the motivation behind the services provided by the WIDE SMD, named Meta Level.

- Once the whole system has been described, we review the functionalities and services provided by the deployment of semantic middleware.

- We also cover the criteria handled during the design process and the final decisions taken. This is done considering the generic criteria explained in Section 3.3.

- With the aim to provide some guidelines for further deployment, we provide some details about the implementation.

- Finally, in the last section we summarize the WIDE SMD validation process and results.

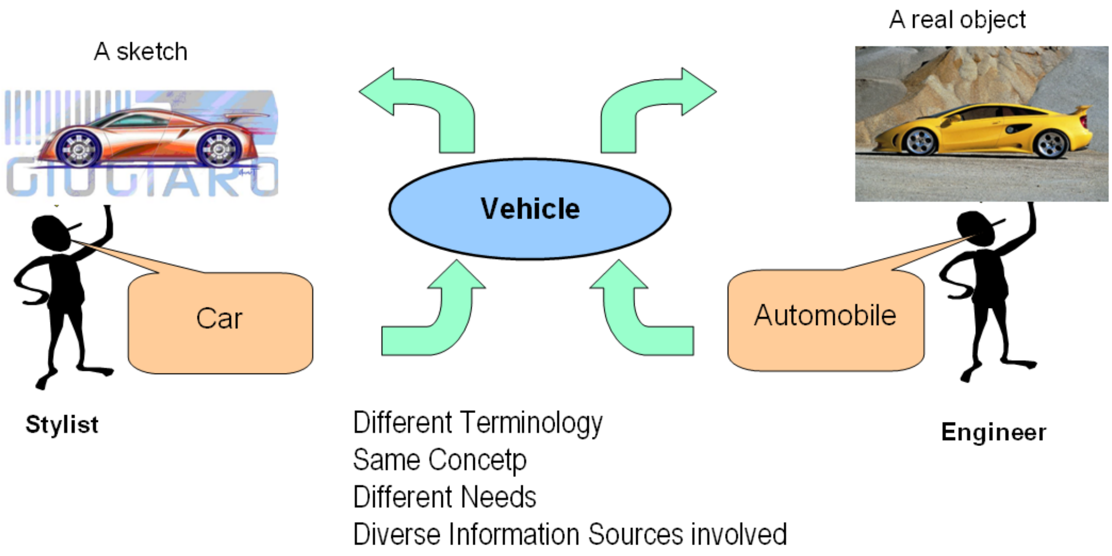


Figure 5.1: WIDE problem statement

5.1 WIDE system

This system was designed, implemented and validated in the context of a research project of the 5th Framework Programme of the European Commission [1]. In (Marcos et al., 2005d; Sevilmis et al., 2005) the reader may find a deeper understanding of the system. In this subsection we cover the motivation that led to the system implementation, the main features of the system and its architecture.

5.1.1 Motivation of the system

WIDE aimed to provide the basis for the improvement of the quality and efficiency of innovative product design. The design process involves **different types of user communities** that employ different languages to access the multimedia data stored in **diverse information sources**. These sources also handle different semantics. And in many cases, as shows Figure 5.1, the **needs behind the users are different**.

Beside this, **the complexity of the access** to very complete proprietary systems represents in many cases a barrier for their employment. This was the situation identified by Schenck, member of the WIDE consortium and one of the European leaders in engine testing, for the ASAM-ODS information

[1] WIDE project(IST-2001-34417) website http://www.ist-wide.info

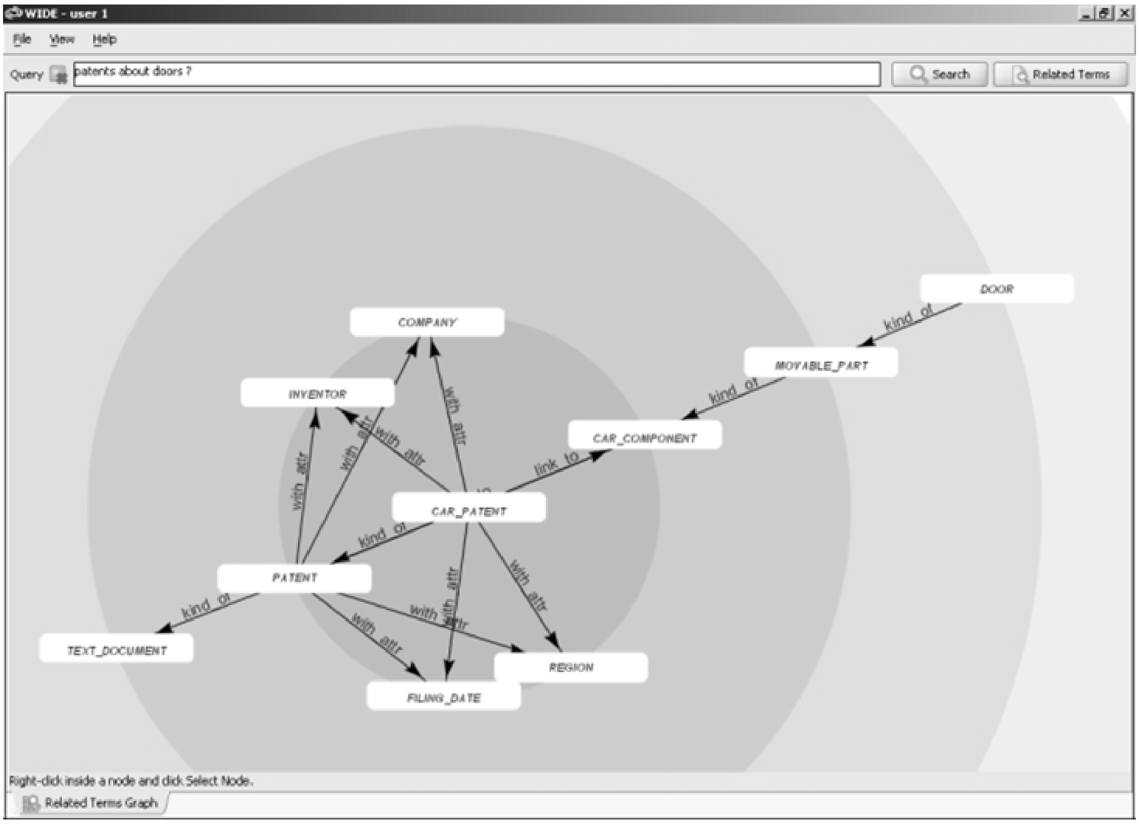


Figure 5.2: Screenshot of WIDE visual tool for domain browsing

sources[2]. The support in the access to these information sources by allowing complementary access mechanisms was another motivation behind the system implementation.

Finally, the information retrieval systems employed by the car domain organizations in the project were not **flexible enough in order to access external information sources**, which value was continuously increasing. An important uniformity lack in information access was clearly identified.

5.1.2 Objectives of the system

In this context, WIDE applied Semantic Web (SW) technologies and methods in an integrated, scalable and reconfigurable information and knowledge retrieval and sharing system. The key to improving innovative design lied in the idea of achieving better multimedia information retrieval tools to **enhance the inter-working of multi-disciplinary design teams**. WIDE system was designed

[2]The ASAM-ODS is the Open Data Services standard developed by the Association for Standardisation of Automation and Measuring Systems (Thelen et al., 2005)

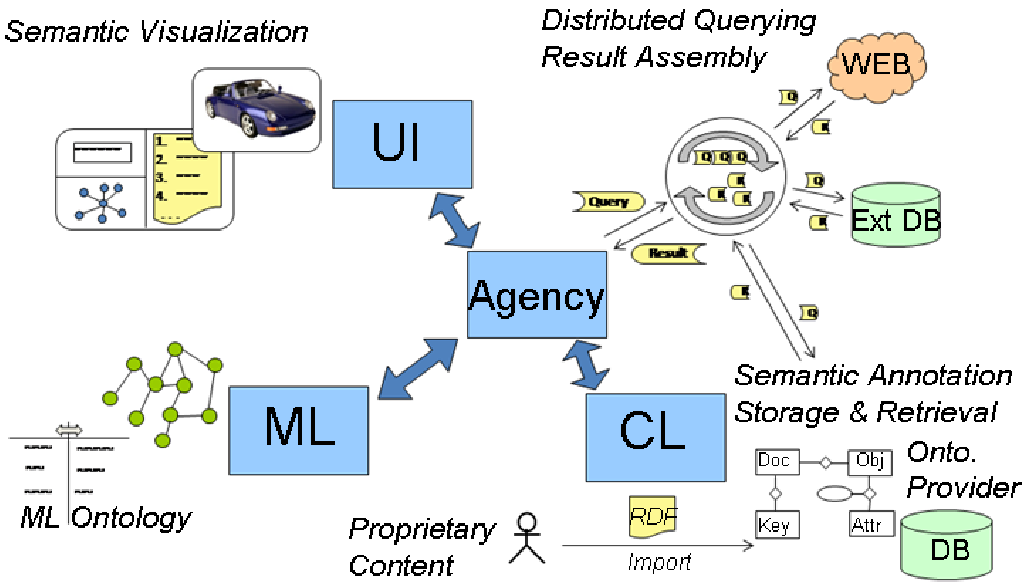


Figure 5.3: Architecture of WIDE system

to enhance the effective and efficient inter-working of industrial designers and engineers by offering a natural and coherent environment for identifying information needs, finding and accessing different information sources, receiving and viewing information from the different sources and relating the results to the current state of the designing process.

The **machine-understandable semantics of diverse information sources** and the integration of the SW techniques in the current approach for multimedia information retrieval were key objectives in WIDE.

The project also explored advanced visualization approaches (see Figure 5.2) that (i) made used of the content-based functionalities and (ii) integrated novel information sources fully aware of the semantics of the content.

Regarding the retrieval engines tackled by the system we highlight the following: an SQL relational database of images and documents, an API to perform web searches, the querying interface of a semantic RDF repository, and an API to access a system compliant with the ODS (Open Data Services) standard developed by the Association for Standardisation of Automation and Measuring Systems (ASAM) (Thelen et al., 2005).

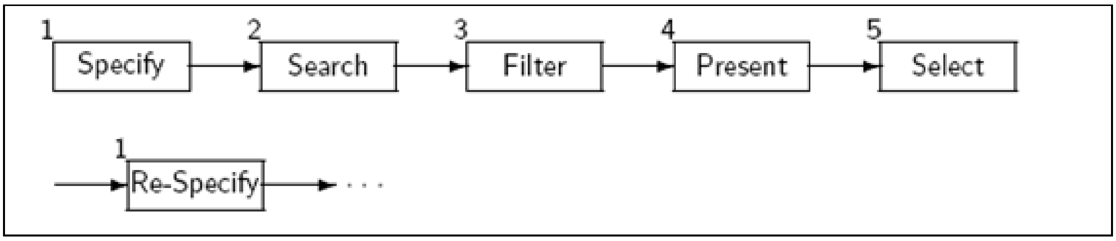


Figure 5.4: Classical search workflow

5.1.3 System architecture

In this section we briefly describe the system architecture in WIDE. Figure 5.3 depicts the main components of the WIDE Semantic based Information System and their relations.

The system is mainly composed by four subsystems: the User Interface (UI), **the SMD deployment called Meta Level (ML)**, the Agency and the Content Level. The UI (User Interface) is the graphical front-end (see Figure 5.3 that is in charge of handling all the aspects dealing with the user interface (domain browsing, query building, results visualization, and so on). The CL (Content Level) represents the internal information of the company. In the project RDFS information sources, relational databases, internet websites and proprietary product management systems were tested. The Agency or agent platform is the search engine of the system, performing the intermediation between the user actions and the querying and access functionalities offered by the set of information sources.

5.1.4 Search Workflow in WIDE

This section describes a new interaction paradigm proposed within the WIDE project. This paradigm, deeply explained in (Smithers et al., 2004) has been defined to cover the needs of information in the engineering design activity, where different people work collaboratively during different steps of a global process.

The WIDE approach was a new interaction paradigm based on the semantic enrichment of the retrieval process. In order to understand this, it is necessary to have in mind the classical workflow for search engines that can be seen in Figure 5.4. In this model the flow information is characteristically linear and for that reason could have a compromised efficiency. Moreover, its exactness could be improved by a cyclic approach in which semantics take a big role to specialize the information retrieval process.

Figure 5.5 represents the WIDE search workflow. That workflow has been

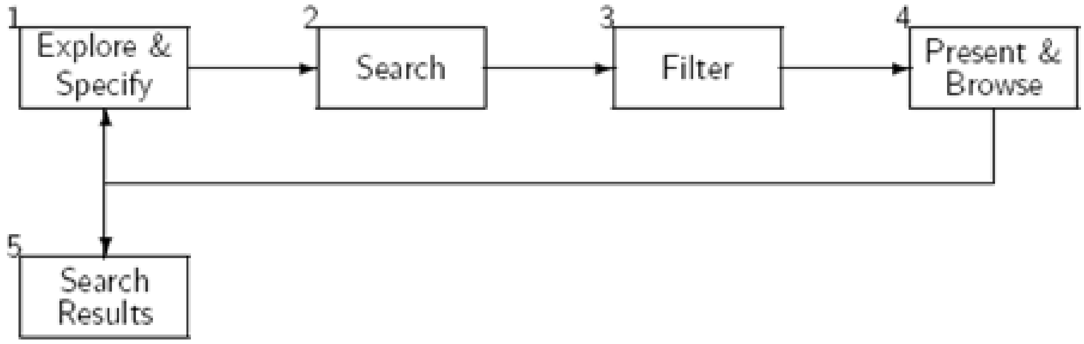


Figure 5.5: Search model implemented in WIDE

developed to better support effective and efficient information retrieval and re-use by users involved in a design process who do not have strong prior knowledge of the organization and structure of the information sources they need to access.

The first step, *Explore and Specify*, includes two sub-processes. The first one represents the system facilities that allow the user to explore and browse graphically his/her domain of interest in order to discover what might form possibly good search specifications. A change of the user's mind at this very first step implies a considerable optimization of the search process. The *Specify* sub-process allows the user either to type the target query or to drag and drop concepts from the explored graph. The *Search and the Filter* processes do not differ too much from the classical workflow. However, the first one can use all the peculiar advantages of the Semantic Web based search engines and the second one has a wider range of criteria (information about the user, knowledge about the domain of the search, semantic comprehension of the query, etc.) that facilitates the filtering. Once the results have been collected, filtered and ranked by the system, they are Presented in the WIDE user interface (UI) that facilitates its *Semantic Browsing*. This interface lets the user not only get the final results (fifth step) but also to use the shown information to refine the query and go back to the first step of the process. Thus, the model defined is circular.

Moreover for the search experience, the WIDE system provides the user with some online collaboration facilities in order to interact with other users.

5.1.5 Meta-Level, the SMD in WIDE

As has been reflected in Figure 5.3, the deployment of the SMD in WIDE is called Meta-Level (ML). The Agency platform, is not only the module responsible for the search over the multiple information sources but also the component in charge of the communication among the modules. As that Figure shows, the communication with the UI is done through the Agency.

The ML provides services to the UI front-end and the search agency in order to implement the semantic driven search workflow described in the subsection 5.1.4.

5.2 Description of WIDE SMD

In this section we provide a description of the WIDE SMD. In order to do this, we include the following sections: (i) First of all, the functionalities of the SMD are described, (ii) then those functionalities are contextualized in one scenario and (iii) finally a summary of the provided services is included.

5.2.1 WIDE SMD functionalities

As has been stated, WIDE employs agent technology to tackle different information sources, offering the results through a unique front-end. The semantic middleware, which is described in (Marcos et al., 2007), is in charge of the semantic operations and services needed by that front-end.

In the following we provide an explanation of those services that the WIDE SMD provides to the rest of the components of the services to enable the content-based retrieval paradigm. In order to facilitate the comprehension and contextualization of the services, the functionalities are grouped according the following categories: "What to ask", "How to ask", "Keeping the semantics all over the process" and "Taking advantage of user knowledge".

What to ask

This section aims to tackle those initial situations where the user needs to search for something but he/she has not a clear understanding of what he/she has to ask in order to get the needed information.

Provide information about the domain. The SMD generates an initial graph that is displayed by the UI as soon as the user is logged in the system. This graph permits the user to learn about the concepts of the domain, their attributes and the relationships among them. Therefore, browsing this information in an intuitive way, the user is able to shape his/her domain of interest. Additionally, the user has the possibility to drag the concepts from the graph and drop them in the query bar.

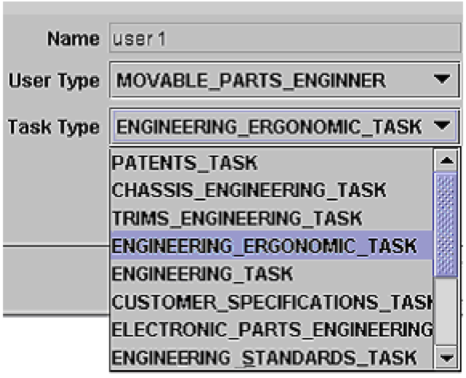


Figure 5.6: Selection Panel for Task Types

Personalized view of the domain. Thanks to the SMD, the WIDE system keeps information about the profile of the user and about the work (process, process step and task) he/she is carrying out. Figure 5.6 covers a partial view of the tasks handled by the system. Each task is related to a set of relevant concepts of the domain. This information is employed not only to show personalized views of the domain but also to perform more accurate query processing and results ranking algorithms.

Allow the user to ask the system for information about the domain. For that, WIDE SMD provides a mechanism that generates graphs that reflect the sub domain of interest of the user. Thus, the user is able to type some "terms" and the system infers and processes the SMD KB, generating an appropriate and customized graph that shows those terms, their relations, the intermediate concepts, the neighbor concepts, and so on. This procedure permits the user to learn about the possible intermediate concepts that could exist, perceiving a broader understanding of the domain, and to filter the area of knowledge, focusing on his/her domain of interest.

How to ask

Neither WIDE nor the ML tackle the multilingual difficulties that arise when users speaking different languages use the same application. Their scope covers only the aspects that are fully related to the terminology.

Although an ontology can be defined as a formal, explicit specification of a shared conceptualization, it is a matter of fact that in the real life every individual uses his/her own terminology.

In the following paragraphs the services provided by the WIDE SMD to tackle this situation are described.

The domain modeled by the ontology contains not only the concepts

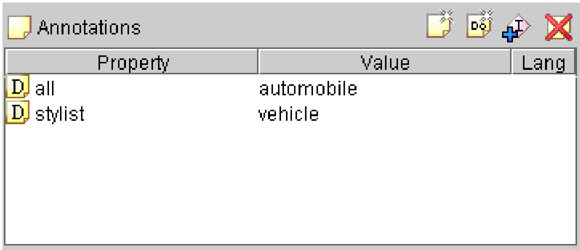


Figure 5.7: Protégé 2000 Annotation Panel

of the domain but also their mapping into the user terminology. In WIDE, the Knowledge Engineer (KE) is responsible for the maintenance of an ontology of “user types”. Each type of user represents a group of users that share a terminology. For instance, in a design company two clear groups of users are the engineers and the stylists (see Figure 5.7). The KE is able to create synonyms for each of the concepts of the domain ontology, and map these synonyms to the different user types represented in the user ontology.

The system should hide the internal terminology. WIDE SMD hides the internal terminology at two levels:

1. The first level is based on the groups of synonyms defined in the domain ontology. Therefore the system is able to map forward and backwards the user terminology into the internal consensual terminology. This allows the system to keep the terminology employed by the user to define the query till the presentation of the results. The constraint of this method is that the system is able to hide only the terms typed by the user.

2. The second level is based on personal dictionaries that, for each user, store the particular terms that he/she has used in all his/her successful searches. Thus, thanks to the SMD, the system is able to hide the internal terminology in a search process, even when those terms have not been typed in that search session.

Keeping the semantics all over the process

The establishment of a content-based retrieval framework requires a deeper understanding of the domain than in the classical systems. WIDE SMD covers this aspect by:

- WIDE SMD has knowledge about the semantics of the domain.
- The system is somehow aware of the context of the search.

- The information sources allow queries expressed using a machine understandable language. It is possible to express not only keyword based queries, but enquires with semantic clauses. In WIDE, the search platform employs an extension of the Resource Query Language (RQL) (Karvounarakis et al., 2002) as the internal language.

The system should transform the human language into machine language. In a coherent view with WIDE objectives, WIDE SMD implements a mechanism that, without handling full natural language, is far away from the structured interfaces (see Figure 5.14).

This approach is carried out by the Meta Level and is based on three main processes: a) a notation grammar to parse the user query; b) a content-awareness query expansion approach and c) a parser that creates the final queries written in RQL, as required by the search platform. Section 5.4 details the implementation of those processes.

The system should present to the user all the available and useful knowledge. Depending on the nature of the knowledge or information repositories, the system should implement the needed mechanisms in order to extract all the available information and present the most suitable information to the user.

WIDE SMD collects the results linking them and their metadata to the nodes of a graph that shows the concepts and the relationships involved in the search process.

However, it may happen that the results collecting is not possible, since the system uses some unstructured information sources (e.g. Google). In this case, there is no way to find out what the results represent. And so, those results are attached to a node called "Unclassified Results".

The WIDE system also implements what is called "The semantic path". This is used to support the user in the results navigation with some accurate information about those results. For instance, if the user asks for pictures about cars with brand either Audi or Seat, the system retrieves several pictures included within a graph like the one showed in Figure 5.8. Then, when the user clicks on the node named "Picture", the whole list of pictures appears, and for each of these pictures the system assigns a semantic path. Thus, if the system knows that the first picture is an image of a car with a brand that is Audi, it highlights in red colour the following nodes of the graph: picture, car, brand and Audi. Thus, the system represents graphically the knowledge retrieved from the domain ontology and the information sources, allowing the user to identify and filter easily what each result represents.

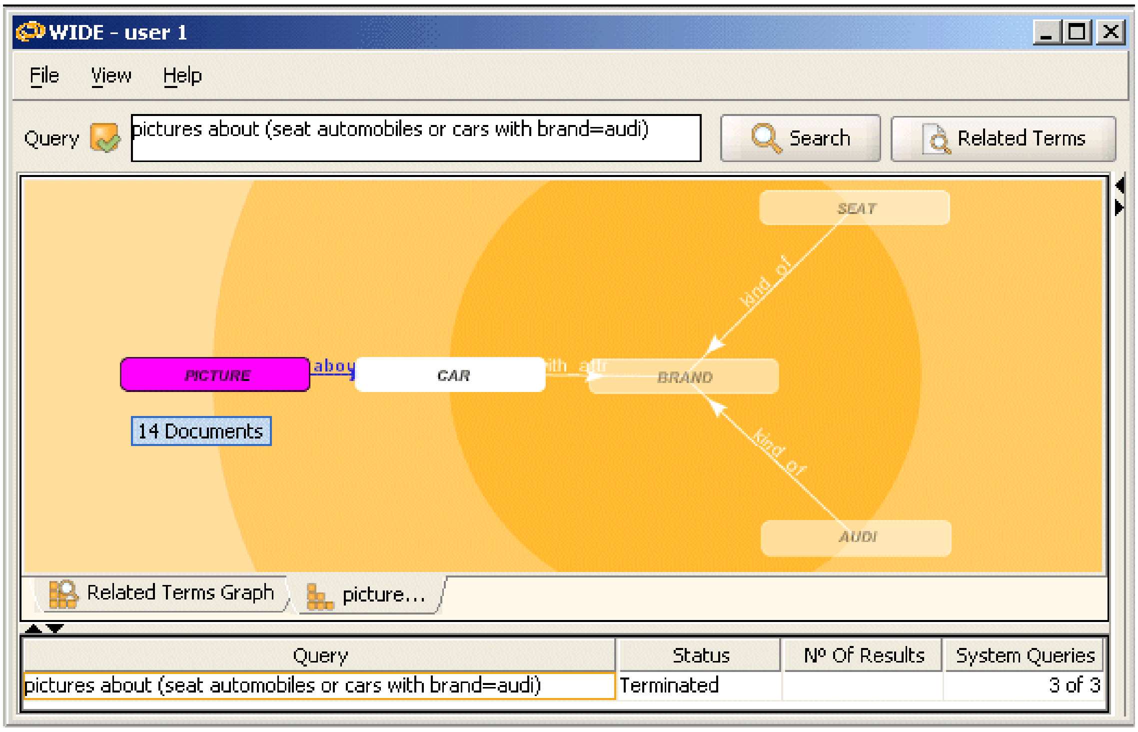


Figure 5.8: Results browsing in WIDE

Taking advantage of user lessons

An advanced Semantic Web system should avoid wasting the feedback generated by the daily users' experiences. Using this knowledge means not only to improve the system itself, but also the level of the acceptance and satisfaction of the users.

This section proposes several approaches to achieve this scenario.

Systems that listen to user problems. WIDE SMD stores some logs that are reported to the KE in order to improve the modeling of the domain. The information covers mainly the unknown concepts employed by the user and the unanswered queries.

Systems that remember their users experience. WIDE SMD stores the preferred synonyms of the user, avoiding showing him/her the internal terminology in the future.

5.2.2 SMD functionalities in an scenario

In this section we present a simplified use example of our approach. The symbol (*) is used to remark where the functionalities implemented in the ML architecture described are used.

Paul, a designer, logs into the system because he needs to look for inspiration in order to design a new door for a new car that is being prototyped in his company. In the logging process he is shown (*) the different processes usually done in his company and specially the ones related to his profile. He chooses “Station Wagon door design” and then a graph with the main concepts involved in the task appears in the screen (*). As he is used to work in this task, he avoids to browse, query and expand this graph (*) and he types in the query bar “Images of Maserati doors”.

What happens internally is that the system checks if the query fits with the notation grammar (*). Once this has been checked, some terminological translations may be done (*) according to Paul’s profile and personal dictionary (*). After that, the system is ready to create several system queries out of the user query (*). The result of this process is a set of system queries like “Images of doors of cars with brand = Maserati” or “Image of doors with brand = Maserati” which are transformed into RQL (*) and submitted to the agents, which apply them to the Information Sources.

Once the results arrive, they are ranked (*), evaluated (*) and graphically displayed making use of all the semantic information available in each result.

5.2.3 Summary of services

We summarize here the functionalities offered by the WIDE SMD.

- Context and user query awareness retrieval: the SMD KB models the different users, their profile and the tasks they carry out. Although this is not a service by itself, this information is employed to enhance the behaviour of the rest of the services described below.

- Support for query development and query validation: based on a notation grammar the SMD provides support for the construction of semi-natural queries (e.g. “Patents about doors older than 5 years”). The SMD, according to a predefined grammar, validates the syntactical rightness of the query and provides the user with recommendation tools to facilitate the query completion while he or she is typing it.

- Semantic query interpretation and expansion: Using a parser for the notation grammar a syntactical interpretation of a well formed query is performed. Making use of the context information and applying semantic techniques (reasoning, distance metrics), the SMD provides different versions for the query with different expansion degrees. For instance, the query “Pictures of seats cars” is expanded to: “IMAGES ABOUT CAR WITH BRAND = SEAT” and “IMAGES ABOUT SEAT CARS”.

- Query normalization: the SMD translates the terms of the query into the internal terminology (e.g. pictures is mapped into IMAGES) and the semi-natural language query is transformed into a query expressed in the language employed by the search agents, which is a variation of RQL(Karvounarakis et al., 2002).

- Support for Query negotiation: The federation of the query over different search engines is driven by rules and performed by a set the search agents (Nitto et al., 2002). During this process, the agents require semantic information (e.g., synonyms and term mapping resources, similarity metrics and disambiguation). The SMD provides them the semantic resources required during the query negotiation and adaptation phases.

- Results ranking and presentation: the ranking of the results is based on a weighted discrimination of the metadata available for the result and the information of the SMD about the context of the user and the task he/she is performing. Regarding the visualization, the SMD is responsible for the graph building that enables a concept-based representation of the results. This allows the user to easily identify the results of the different information sources that provide the results.

- Customization of the information presentation: the front-end employs the SMD services to hide the internal terminology to the user in different moments of the interaction. This service enables an automatic translation of the information by employing the terminology of the community the user belongs to and by using his/her personal preferences. This is applied for the domain browsing, the recommendations provided for the query completion, and the metadata of the results.

5.3 Key design criteria of the SMD

In this section we provide an overview of the solutions adopted for the design, implementation and maintenance of the WIDE SMD. We cover the three layers independently, following the structure and criteria presented in Section 3.3.

5.3.1 WIDE SMD KB

First of all, we describe the solutions adopted for the key criteria related to the SMD KB.

People in charge of the knowledge base: The team in charge of the SMD KB was composed of two engineers, both of them with a long track record of expertise in engineering of car bodies and engines respectively. Moreover, one of them belonged to standardization committees in that sector.

Identification of the domain: The domain was initially divided into the following subdomains: users, tasks and information. Concerning the users, based on the organization of the company (i.e., nature of the work and information access), more than 20 types of users were initially identified and hierarchically classified. This hierarchy simplified the definition of the terminology of each user type, by applying inheritance to the subtypes with respect to the information available for the supertypes.

With respect to the tasks, following the real workflow of the design and testing processes, the tasks were interrelated using temporal and dependency constraints. The initial number of tasks identified was reduced to a smaller set of tasks (about 35), selecting only those that had certain peculiarities from the search perspective. This was done to avoid performance and maintenance problems derived from the management of tasks that, although intrinsically different, were identical to the perspective of the search requirements.

Finally, the hardest task was the identification and definition of the information domain. The approach most successfully implemented was the following: First of all, a collection of representative multimedia assets were collected. This set was mainly composed of patents from different sources, images of a very diverse nature (from high quality images for inspiration to sketches), 3D objects, graphs, test results, normative for engine testing and car design. Secondly, and ignoring the existing metadata of the sources the data came from (e.g., internal database, ODS system and DVD library), knowledge engineers made a first identification of two lists: set of search criteria and set of relevant targeted pieces of information. After this, those sets were discussed and enriched with those experts on those tasks that had the most demanding search needs. Finally, a coherent object oriented model was built including the identified pieces of information.

Technology of the SMD KB: The technology employed in the WIDE SMD KB was diverse. First of all, the already identified subdomains were mapped to OWL (Ontology Web Language) (Grau and Motik, 2008) ontologies. Seven ontologies were developed. One ontology was devoted to the users, another for the domain of the car industry, three more related to search context (processes, process steps and tasks) and finally, one developed to reflect the subjective feelings related to the inspiration process of the designers (e.g., “Looking for aggressive animals”, “vintage objects”). In total the system handled more than one thousand interrelated concepts.

Due to the demanding requirements of the online services and the state of the art of the ontology accessing technology during the development phase, the ontologies were parsed and stored in a relational database by a self-developed component. This component performed off-line all the pre-processing required over the ontology to allow efficient usage of the information during the provision of the online services.

Beside the mentioned ontologies, the SMD KB also included a Backus Naur form notation (Knuth, 1964) to enable the semi-natural language processing.

Composition of the knowledge base. The main relationships among the concepts of the different subdomains were the following: First of all, the user's types were related to the task the users were involved in, allowing different involvement degrees. Each type of user had a dictionary. This dictionary was mapped to the internal terminology employed in the domain ontology. Then, the different tasks types were interrelated to the concepts of the domain ontology, weighting the relevance of the different concepts for each of the tasks performed. This was done with the goals of: firstly, optimizing the query processing (e.g., "Rolls Royce" was linked to the brand of the manufacturer of a car or a plane engine depending on the type and context of the user) and secondly, presenting the results according to the ranking. Thus, the results with a lower semantic distance to the context were prioritized.

The parser of the notation grammar employed to process the query was also linked with the terms of the information ontology. As a result, the SMD was able to provide query completion support (e.g., If the user typed "Patents about" and waited longer than 2 seconds, the SMD sent a list of possible items that may be patented to the front-end).

Reusing shared knowledge. The domain of the car and engines design and testing is very broad. For instance, diverse incompatible and very dynamic classifications for the segments of the cars (e.g., station wagon) coming from different official and unofficial sources were identified at the designing stage. Although important sets of standard terminology and schemas (e.g., ODS) were included, most of the information gathered in the KB had to be developed from scratch by the knowledge engineers in order to achieve the project aims.

Documentation and Update. The object properties of the ontologies were hierarchically defined. The top level of the property hierarchy provided information about the usage of the property in the retrieval system. Therefore, the update procedure for the ontology was done by analyzing the different properties of the concept to be updated. Depending on the superproperty the property belonged to, an update procedure was defined. The update procedure included guidelines, pre-checks to be carry out or software adaptations that must be required.

5.3.2 WIDE SMD IE

Regarding the criteria used in the deployment of the WIDE SMD IE, we highlight the following:

Maintainability. The approach followed for the definition of the PEs was the conventional object oriented software design (Schach, 2007). For instance, behind the *Graph Constructor PE* there was a set of 10 interfaces that generated a graph starting from a set of concepts, but with different input parameters.

Exchange information structures. Reaching an stable definition of the structures of information in WIDE SMD IE was crucial. Some of the software entities most employed and exchanged by the PEs were: Node, Graph, Term (semantic or syntactic), TermList, Query, Result, Context and User.

Making use of available resources. The external resources used by the PEs in the WIDE SMD KB were the RACER reasoner (Haarslev and Möller, 2001), used mainly for consistency check during the off-line processes of the SMD, an OWL API (Bechhofer et al., 2003), and a combination of JavaCC and JJTree to automatically build the notation grammar parser required by the *BNF Syntactic Paser* and *Semantic Interpreter and Query Expansion* PEs. The Google Sets functionality (http://labs.google.com/sets) was employed by the *External Resource Handler PE* to automatically detect lack of completeness in the SMD KB. Thanks to the thesauri information provided by Google Sets, the system was able to detect that with the query “Sketches of Kia logos”, the user was looking for the logo of a car brand even without having Kia identified in the SMD KB as a car brand.

PEs network interoperability. All the PEs in the WIDE SMD IE were either standalone applications remotely invoked or standard Java Interfaces.

Time management. Critical PEs provide asynchronous communication mechanisms.

5.3.3 WIDE SMD GW

Flexibility: The offline SPs offered by the SMD were standalone. The online SPs were implemented with Java RMI. The decision was taken during the design stage due to the fact that all the components the SMD supports to, were developed in Java and some of them running on different servers. Nowadays, a SOAP solution would have been more appropriate.

PEs invocation: In order to increase the performance, all the PEs were installed in the same machine that the SMB KB. However this was not always the case for the SMD GW SPs.

Synchronization and status management of each service: The communication between the SMD GW SPs and the PEs was mainly asynchronous.

Global status management: Each SMD GW SP could be queried about each status at any time. A set of error and status codes allowed the control of the search status, which were shown to the user.

5.4 Implementation details of the SMD

As has been explained in the subsection 5.1.4, the WIDE SMD supported the user in all the steps of what we called "Circular Search" approach and over the online collaboration mechanisms.

In a coherent view with Chapter 3, the ML architecture is shown in the Figure 5.9.

In the following subsections, the implementation details of each of the layers is tackled.

5.4.1 ML-KB

This section describes the ML KB which was composed of every piece of knowledge used by the Processing Elements to carry out their different atomic tasks. That ML KB allowed the Support Processes to interact with the different WIDE subsystems all over "the WIDE cycle".

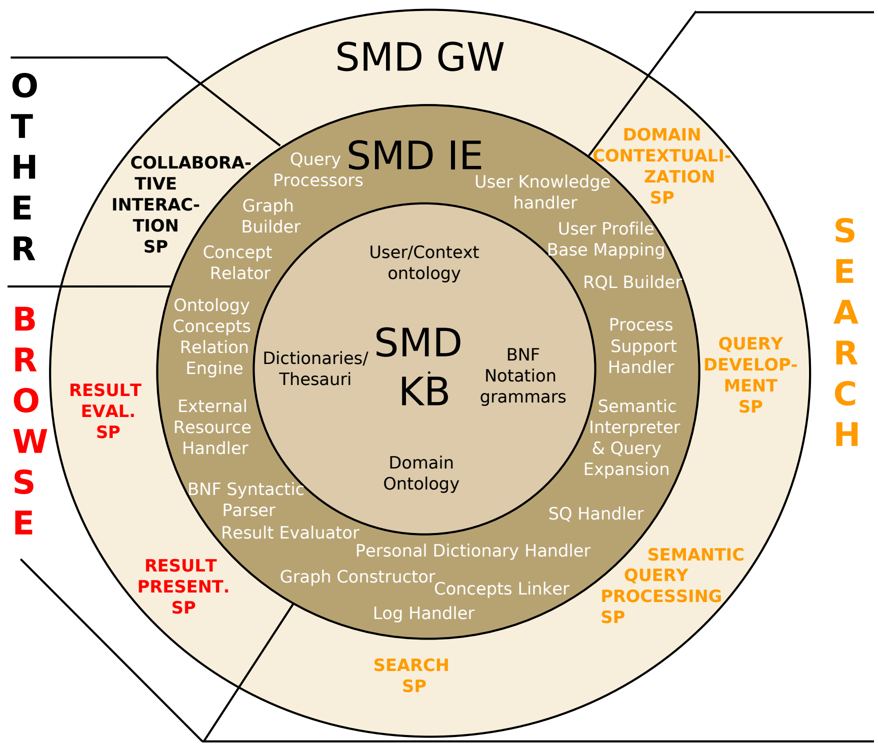


Figure 5.9: Meta Level Architecture

The ML-KB was composed by a set of seven OWL ontologies and a Backus Naur form (Knuth, 1964) notation employed to enable the semi-natural language processing.

ML ontologies

ML was composed of 7 interrelated ontologies that allow the ML to support the WIDE system all over the search process.

ML ontologies Language. The language used to edit ML ontologies was OWL (Ontology Web Language) (Grau and Motik, 2008). OWL facilitates greater machine interpretability of Web content than that supported by XML, RDF, and RDF Schema (RDF-S) by providing additional vocabulary along with a formal semantics.

ML ontologies Edition. The Edition of ML ontologies was performed with the Ontology Editor Protégé. The main characteristics of Protégé are the followings: (i) its community of users is composed of thousands of people, (ii) it is open-source, multiplatform (Java) and (iii) it is able to provide an extensible architecture for the creation of customized knowledge-based applications.

ML ontologies Storage. Once the ontologies were edited it was needed to ensure an appropriate access to the information represented with those sets of classes and relationships. Our preliminary research was to test the feasibility of the different available tools. Firstly, the RDF repository of Sesame (Broekstra et al., 2002), which is RQL query-able, was used. Secondly, ontology APIs and Reasoners were tested.

These experiences conducted to some conclusions that we highlight here:

- The reasoners and APIs of that period were focused on the management and inference over the instances. During the development of this work, concerning the class level, these tools provided quite simple information. These tools provided solution to problems of low complexity such as answering the following queries: Is 'A' a subclass of 'B'?, which are the subclasses of 'A'?. However, there was no way to retrieve more or less straightly and with an affordable cost from the time consumption point of view, answer to a question like Is 'A' related to 'B' and if so, how?. To answer these kind of questions implied to develop recursive algorithms which performance decreased exponentially when the size of the ontology gets closer to the needs of the real applications.
- The size of the ontologies was the second problem that arose during the research. Some of the available tools, were not able to work at all or in a proper way with ontologies that have hundreds of concepts and relationships.
- Another undesirable situation with the available tools was that while in the edition of an ontology some restrictions were carried out over the relationships (for example, to restrict the cardinality), it was not possible to have different restrictions over the same property depending on the range of that property. Thus, there was no way to specify that "A car HAS 4 or more wheels but 1 and only 1 steering wheel". This made difficult to make use of the transitivity of the properties, since the reasoners were not able to analyse the following clause "A car has_powertrain and a powertrain has_engineblock engine_block", being both has_powetrain and has_engineblock transitive properties and extract the following conclusion: "The engine_block is contained by a car".

The conclusions gathered from our research work , led us to implement an optimized SQL repository that represented the basis of the knowledge gathered by the ontologies, but implementing some additional mechanism that also allowed the storage of some pre-inferred information.

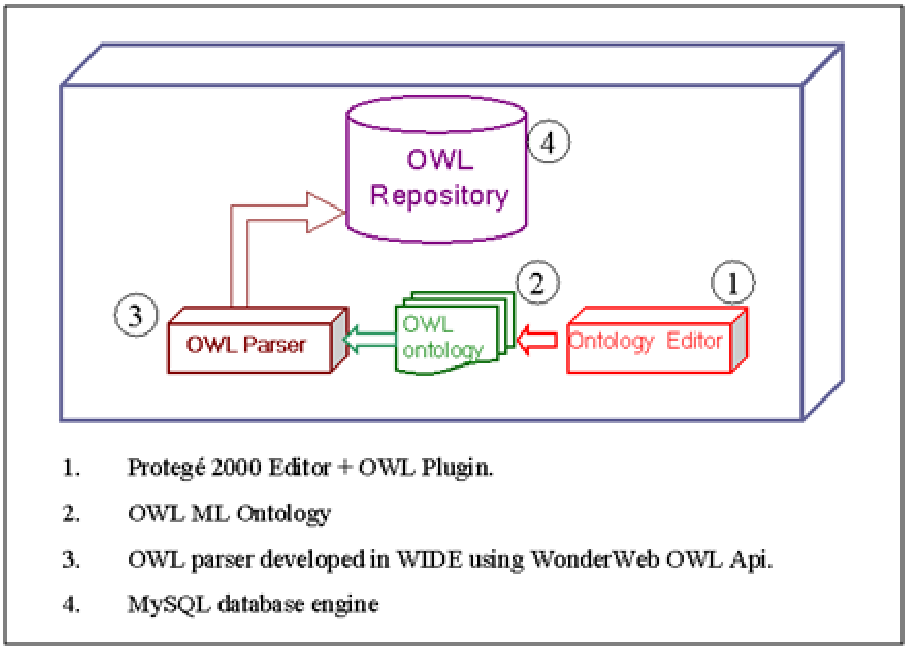


Figure 5.10: Ontologies in WIDE ML KB

This OWL repository (OWL-rep from here on) was based on a MySQL database and is sketched in Figure 5.10.

According to the image, the repository was filled using an OWL API developed by the WonderWeb IST project (Bechhofer et al., 2003).

This inference process was carried out automatically before the WIDE system was launched, and was executed when the KE made any change in the ontologies. The result of this inference process was the fulfilment of the above mentioned OWL-rep. This process, needed up to 5 minutes on a Pentium 1.2 GHz and 512MB of RAM to parse a set of ontologies with around 1000 entities

The purpose of each table in this OWL-rep is described in annex V.

ML ontologies. The seven ontologies developed for the ML KB were the following:

- Car Industry Domain Ontology (from now on ML Domain Ontology): This ontology was the main of all the ML ontologies and was modeled employing the knowledge of both Industrial Partners (ItalDesign and Schenck).

- Content Type Ontology: This ontology modeled the domain of the multimedia assets retrieved by the system. Some of those content types were: documents, pictures, drawings, sketches, CAD models, etc. (see Figure 5.11).

- Description Ontology: This ontology was modeled to represent the different descriptions presented in the scenarios. This ontology aimed to represent the subjective feelings of the design process.

- User Types Ontology: The User Types ontology represented the different user profiles of the company. This information was used to provide the appropriate terminology mapping for each user profile.

- Process Ontology: This ontology contained the knowledge of the different processes carried out in the scenarios defined by the industrial partners. This ontology was highly related to the one that follows.

- Process Steps Ontology: This ontology represented the different Process Steps of the scenarios, together with the relationships among them. This ontology was related not only with the Task Type Ontology, but also with the ML Domain Ontology, since for each process step the main concepts involved in that step were specified.

- Task Types Ontologies: This ontology represents the different tasks, understood as atomic pieces of work, of the scenarios. For each task the most important concepts involved were also specified. Thus the Task Types Ontology was also related to the ML Domain Ontology.

The first three ontologies were generic and if there had been any available international standards, those would have been used. The other ontologies were modeled ad-hoc in order to represent the knowledge related to the different user profiles of the industrial partners and the distinct Processes, Process Steps and Tasks carried out by those users.

BNF Grammar

As we detail in the subsection 5.4.2, the User Query process carried out in order to generate the different System Queries that employed the search process was made using a parser based on a BNF Grammar.

BNF is a formal notation to describe the syntax of a grammar of a language. The definition of a grammar like that allows the generation of parsers that are able to decide if the usage of that language is correct or even to extract useful information about the meaning of that usage.

In Annex V the reader is able to find the ML BNF grammar. Additionally, some important hints about this grammar are included here.

The motivation behind the definition of the BNF grammar was the need of supporting a semantic interpretation of the user query. The aim was to avoid the

- CONTENT TYPE ONTOLOGY SYSTEM TERM
  - CONCEPTUAL DATA
  - CONTENT TYPE
    - AUDIO
    - DOCUMENT
      - APPLICATION DOCUMENT
      - IMAGE DOCUMENT
        - DIAGRAM
          - DIAGRAM 3D
        - DRAWING
        - PHOTO
        - PICTURE
        - SKETCH
      - MODEL DOCUMENT
        - CAD MODEL
          - CAD MODEL 3D
        - PACKAGE MODEL
      - TEXT DOCUMENT
        - CUSTOMER SPECIFICATION
        - ERGONOMIC CODES
        - LEGISLATIVE REGULATION
          - DIRECTIVE
          - HOMOLOGATION
          - LAW
          - RECOMMENDATION
        - PATENT
          - CAR PATENT
            - BRAND PATENT
            - DESIGN MODEL PATENT
            - GENERAL PATENT
        - STRUCTURAL ANALYSIS
        - TEST
          - EMISSION TEST
          - IMPACT TEST
          - PERFORMANCE TEST
        - TEST RESULT
    - VIDEO

Figure 5.11: Screenshot of ContentType ML Ontology

usage of a structured user interface that forced the user to specify the concept that was the target of the search plus some attribute/value pairs and keywords.

This grammar was produced after having analysed the structure of the expression typically used when querying multimedia systems. Although, as any grammar, it did not cover every possible scenario, it can be stated that the achieved one was quite powerful and flexible, being their main components and characteristics the following:

- This grammar allowed binary and logical operators.
- The BNF ML Grammar was scalable.
- Its main components were: qualifying lists, qualifyings, connectors, logical operators, attributes, values, units and keywords.

5.4.2 ML-IE

As has been stated, ML Processing Elements (PEs from now on) can be considered as black boxes that perform atomic tasks over the information gathered in the ML KB. In the following paragraphs, the main PEs implemented in the WIDE SMD deployment are described.

Ontology Concepts Relation Engine

This PE was in charge of the reasoning process of the OWL-Rep in order to find out if the relation about concepts. For instance, the Ontology Concepts Relation Engine had to find the relation and the nature of that relation between the concept “Golf” an the concept “door”.

The Ontology Concepts Relation Engine implemented an optimized sql based search algorithm over the OWL-REP that, when received 2 terms that had been modeled within the Ontology, found out if both terms were related, retrieving for each case useful and needed information.

Figure 5.12 shows some of the relations that this PE was able to infer in real time.

Although the name of those relationships was quite intuitive, in the following we include the description of some of them in order to provide a more clear idea of the information retrieved by this PE. These descriptions assume that the PE has been invoked in order to find the relation of the terms ’A’ and ’B’:

- HasAnAncestorDirectlyPointedByA: This kind of relation was retrieved when A and B were Classes and when there was an ObjectProperty that had one

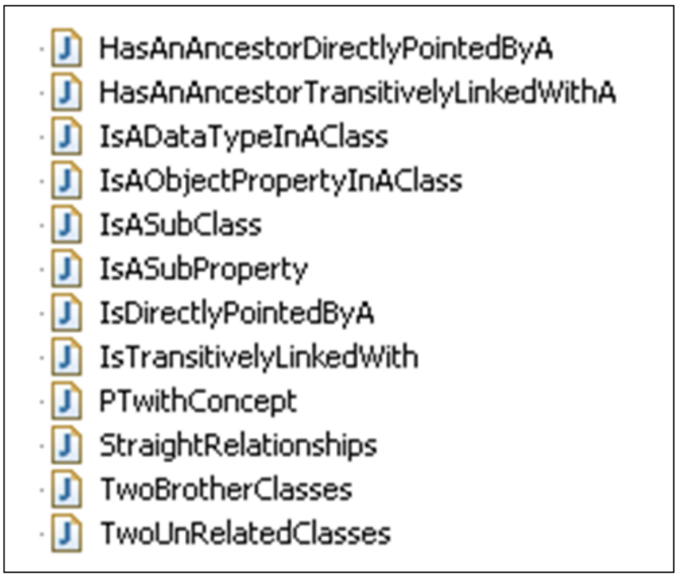


Figure 5.12: Overview of relationships hierarchy

of the superclasses of B as the domain whereas its range was composed of A. Besides the relation, the ML Ontology Concepts relation engine also retrieved the list of all the classes that were in between the domain of the property and the class B.

- HasAnAncestorTransitivelyLinkedWithA: The PE retrieved this kind of relation when one of the superclasses of the Class B were in the domain of a transitive property which range was composed of a class that belongeds also to the domain of that property. So, the transitivity of that property jumped from one class to the other until it reached the class A. ML Ontology Concept Relation Engine retrieved all the useful information, which included two lists of concepts that contained the list of ancestors of B and the list of intermediate concepts respectively.

- IsADataTypeInAClass: This relation was retrieved by the PE when B was a DataType which domain was A.

- IsAnObjectPropertyInAClass: Only when B was an ObjectProperty and A belonged to its domain. This relation was generated by the reported PE. If there was any restriction on the range of the property B, the list of classes was also retrieves.

- IsTransitivelyLinkedWithA: In the following we list the requirements associated to this relation: (i) A and B were both classes, (ii) B was in the domain of a transitive property that had in its range a class that was also in the domain of the same transitive property, and so on until (iii) finding a

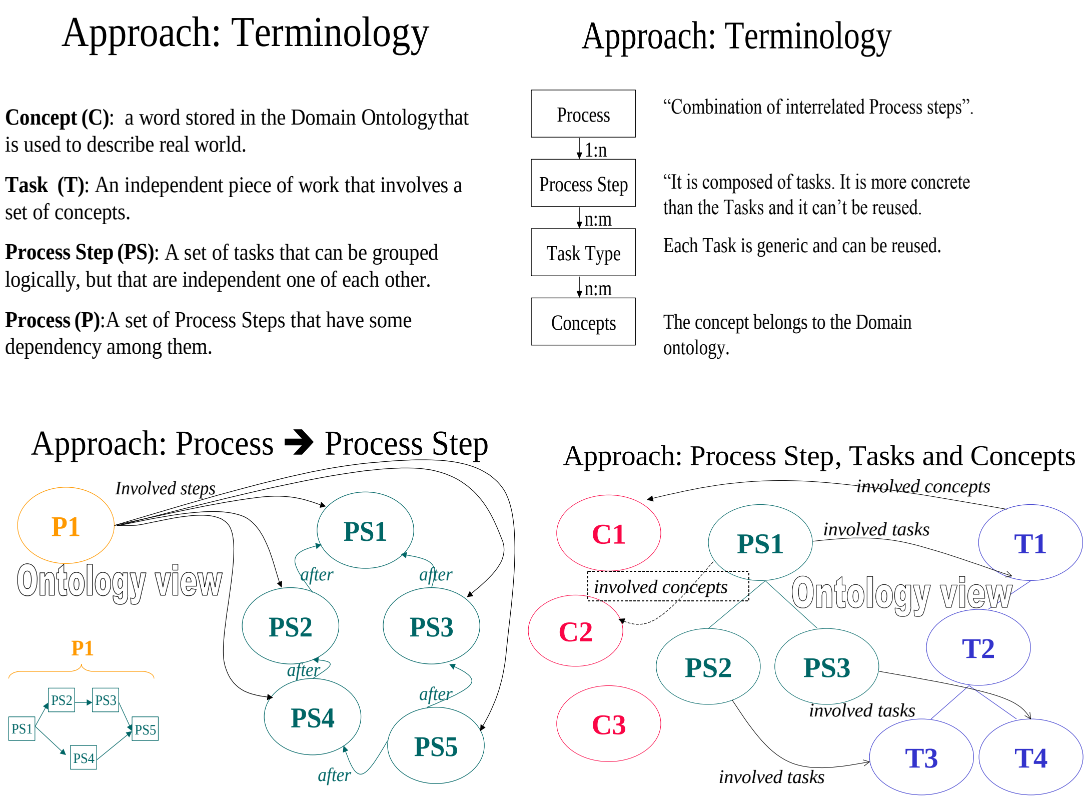


Figure 5.13: ML approach for Process Context Management

property which domain was A. The list of concepts that were involed in that link was also retrieved.

- PTWithConcept: Whenever B was a Class that belonged to the ContentType ML ontology (e.g. picture) and A was a Class that belonged to the ML Domain Ontology, this relation was generated and retrieved by the reporting PE.

Process Support Handler

Every user logged in WIDE had to specify the process and task he/she was performing. The WIDE approach, as can be seen in Figure 5.13, considered that a process was composed by a set of process steps, with some constraints among them. And each one of those process steps was broken down into several tasks, which were related to some sample queries. Besides this, every process step or task was linked with the ML ontology concepts that were involved in their execution.

In a coherent view with this approach, this PE implemented the following main roles:

1. The “Process Support Handler” was responsible for transferring the knowledge edited by the KE from the OWL-Repository to the UI via the Agency. This was done by the generation of XML files each time the system was launched. A simplification of what one of these files could be can be found in the Annex V.

2. Its second main role was to support the ML with the information about processes once the user started his/her interaction with the system. For instance, when the ML had to process an user query that was executed within a context (i.e. a specifi process, process step and task) this PE was responsible for providing the list of the main concepts involved in that context.

User Knowledge Handler

This Processing Element was in charge of retrieving the needed information for each user. This piece of information was composed of the allowed tasks and “language” (terminology) related to the user profile of the user together with his/her personal dictionary.

User Profile Based Mapping

The User Profile Based Mapping was responsible for the terminology mapping made by the ML at very different moments. This allowed to handle the different user’s terminology, which was one of the requirements of the system.

This User Profile Based Mapping PE allowed the mapping between the user terms and the ML internal terms, hiding the user the terminology used by the system.

In summary, the role of this PE was to translate a term into another one, whenever it was possible and appropriate, based on the type of user. This was carried out using the knowledge stored in the OWL-Rep. This knowledge came from the ML Ontologies where the KE user annotation properties to specify the different synonyms for each user type.

Personal Dictionary Handler

The ML Ontologies had only one internal term for each concept. However there are also some synonyms to take into account the different terminologies for each

Figure 5.14: Input field of WIDE user interface front-end

user profile.

In order to hide the internal terminology to the user, the ML learnt the user-preferred terms to use them in the future.

This knowledge was stored for each user in what is called “Personal Dictionary” and the “Personal Dictionary Handler” was responsible for using and updating this dictionary.

As those personal dictionaries were stored in the CL, this PE had to invoke the Agency in order to access and modify these dictionaries.

BNF Syntactic Parser

This Processing Element checked if a string was compliant with the BNF notation (see annex V to see the grammar implemented).

Once the string was checked the following was performed:

- If the string was not compliant with the notation, the parser generated an error, which was transmitted to the user via the User Interface (see Figure 5.14).
- If the string was compliant, the parser generated a tree that represented the interpretation of that string according to the notation. The query shown in Figure 5.14 would generate the tree interpretation shown in Figure 5.15.

The parser reported error messages to let the user know the rightness of its query from the syntactic point of view. In order to do that, the UI showed either an OK status symbol or a non-OK one.

Semantic Coherent Parser

The input needed to invoke this parser was the tree generated by the PE BNF Syntactic Parser reported in subsection 5.4.2. The role of the Semantic Coherent Parser was to enrich this tree with the information extracted in the OWL-Rep. According to this, and using also some other Processing Elements as the ones related to the terminology translation, this parser enriched each term of the tree

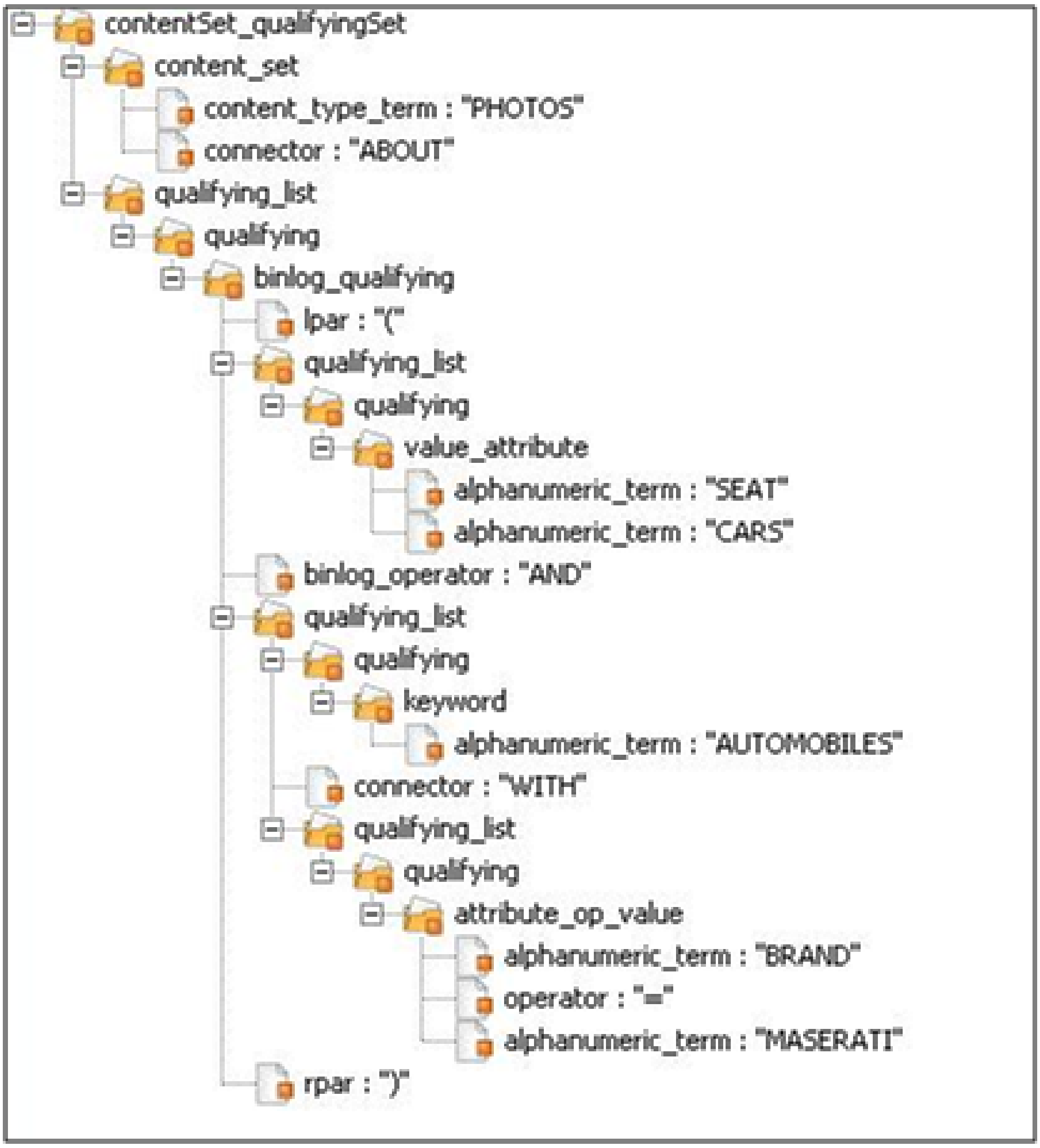


Figure 5.15: ASF interpretation of the query

with extra-information as for example: (i) with new synonym terms taking into account the user terminology, (ii) tagging the terms that were not known by the ML Ontology, (iii) tagging the connectors, logical and binary operators, etc.

Besides this enrichment of the tree, the parser checked the semantic coherence of the tree. For instance, a query like “pictures of documents of /ldots” would retrieve a semantic error message saying that *“It seems that you are specifying a type of content related to another type of content”*. This message was transmitted to the UI, which was responsible for the refinement query process.

Semantic Interpreter & Query Expansion

This processing element carries out one of the most critical tasks in the ML. The input of this process was an Abstract Syntax Tree (AST) that was obtained after a successful syntactical parsing and which semantic coherence had been checked. This AST represened the user query. This tree had also a complete information about the terms that composed the tree, taking into account what was represented in the ontology.

The output of that process was a set of trees that represented the different System Queries. In the following paragraphs we explain what was done by the PE to generate that output.

The expansion process was based on the interpretation of the user query. This interpretation was based on the structure of the query typed (according to the BNF grammar) by the user and on the inference of the OWL-Rep. Besides this, some rules were defined in order to apply some replacements whenever this PE considered it was needed.

The following examples were quite representative of this process. For each one of the examples, both the original pattern and final result are included, together with some explanations when needed.

Example # 1

```
Input:
Interpretation of the input AST tree:
 "DOCUMENT about TEST of (AUDI CAR and CYLINDER > 8)"
L0 - userInput
  L1 - contentSet\_qualifyingSet
    L2 - content\_set
      L3 - content\_type\_term : "DOCUMENT"
      L3 - connector : "ABOUT"
    L2 - qualifying\_list
```

```
      L3 - qualifying
        L4 - keyword
          L5 - alphanumeric\_term : "TEST"
      L3 - connector : "OF"
      L3 - qualifying\_list
        L4 - qualifying
          L5 - value\_attribute
            L6 - alphanumeric\_term : "AUDI"
            L6 - alphanumeric\_term : "CAR"
        L4 - connector : "WITH"
        L4 - qualifying\_list
          L5 - qualifying
            L6 - attribute\_op\_value
              L7 - alphanumeric\_term : "CYLINDER"
              L7 - operator : ">"
              L7 - numeric\_term : "8"

Output:
Interpretation of the input AST tree:
 "DOCUMENT about TEST of CAR with (AUDI BRAND and CYLINDER > 8)"
L0 - userInput
  L1 - contentSet\_qualifyingSet
    L2 - content\_set
      L3 - content\_type\_term : "DOCUMENT"
      L3 - connector : "ABOUT"
    L2 - qualifying\_list
      L3 - qualifying
        L4 - keyword
          L5 - alphanumeric\_term : "TEST"
      L3 - connector : "OF"
      L3 - qualifying\_list
        L4 - qualifying
          L5 - keyword
            L6 - alphanumeric\_term : "CAR"
        L4 - connector : "WITH"
        L4 - qualifying\_list
          L5 - qualifying
            L6 - binlog\_qualifying
              L7 - lpar : "("
              L7 - qualifying\_list
                L8 - qualifying
                  L9 - value\_attribute
                    L10 - alphanumeric\_term : "AUDI"
```

```
              L10 - alphanumeric\_term : "BRAND"
        L7 - binlog\_operator : "AND"
        L7 - qualifying\_list
          L8 - qualifying
            L9 - attribute\_op\_value
              L10 - alphanumeric\_term : "CYLINDER"
              L10 - operator : ">"
              L10 - numeric\_term : "8"
        L7 - rpar : ")"
```

Explanation: As is sketched in this example, the reporting PE was able to modify the input tree in order to generate a new output tree which expressed the domain knowledge stored in the ML. This implied the inference of the relations among the concepts involved in the input tree, and the interpretation of those relations. Finally, this interpretation was transformed into a new output tree. This inference was based on some of the different PEs that were analysed in this section, and they could use not only the knowledge of the ML Ontologies, but also some external knowledge as for example the Google Sets.

Example # 2

```
Input:
Interpretation of the input AST tree:
 "cars with (brand = f?* and cylinders = 8)
L0 - userInput
  L1 - contentSet\_qualifyingSet
    L2 - content\_set
      L3 - content\_type\_term : "CAR"
      L3 - connector : "WITH"
    L2 - qualifying\_list
      L3 - qualifying
        L4 - binlog\_qualifying
          L5 - lpar : "("
          L5 - qualifying\_list
            L6 - qualifying
              L7 - attribute\_op\_value
                L8 - alphanumeric\_term : "BRAND"
                L8 - operator : "="
                L8 - wildcard\_term : "F?*"
          L5 - binlog\_operator : "OR"
          L5 - qualifying\_list
            L6 - qualifying
              L7 - attribute\_op\_value
```

```
                L8 - alphanumeric\_term : "CYLINDER"
                L8 - operator : "="
                L8 - numeric\_term : "8"
          L5 - rpar : ")"

Output:
L0 - userInput
  L1 - qualifyingSet
    L2 - qualifying\_list
      L3 - qualifying
        L4 - keyword
          L5 - alphanumeric\_term : "CAR"
      L3 - connector : "WITH"
      L3 - qualifying\_list
        L4 - qualifying
          L5 - binlog\_qualifying
            L6 - lpar : "("
            L6 - qualifying\_list
              L7 - qualifying
                L8 - binlog\_qualifying
                  L9 - lpar : "("
                  L9 - qualifying\_list
                    L10 - qualifying
                      L11 - value\_attribute
                        L12 - alphanumeric\_term : "FORD"
                        L12 - alphanumeric\_term : "BRAND"
                  L9 - binlog\_operator : "OR"
                  L9 - qualifying\_list
                    L10 - qualifying
                      L11 - value\_attribute
                        L12 - alphanumeric\_term : "FERRARI"
                        L12 - alphanumeric\_term : "BRAND"
                  L9 - binlog\_operator : "OR"
                  L9 - qualifying\_list
                    L10 - qualifying
                      L11 - value\_attribute
                        L12 - alphanumeric\_term : "FIAT"
                        L12 - alphanumeric\_term : "BRAND"
                  L9 - rpar : ")"
            L6 - binlog\_operator : "AND"
            L6 - qualifying\_list
              L7 - qualifying
                L8 - attribute\_op\_value
```

```
            L9 - alphanumeric\_term : "CYLINDER"
            L9 - operator : "="
            L9 - numeric\_term : "8"
      L6 - rpar : ")"
```

Explanation: This example aims to show some of the expansion mechanisms implemented in the ML in order to expand the queries according to the semantic interpretation of the query typed by the user. Thus, in the example included, we show the interpretation made by the ML concerning the wildcard term “F*” specified by the user.

Graph Constructor

This PE can be considered as a black box that generated the graphs that were visualized by the user.

In the following we include (i) the input needed by this PE, (ii) the different outputs generated depending on the different inputs and (iii) the internal process carried out in order to transform the input information into the output.

Input: This PE needed the following information:

- The list of concepts that had be shown in the graph.
- The list of concepts that were related to the context of the graph.
- Level of expansion wished.

Output: The output was a Graph. The complexity of this graph was configurable using the level of expansion specified. In fact, the graphs shown to the user in order to let him explore the domain were more complex and bigger than the ones related to the visualization of the results. One example of this Graph Notation can be found in Annex V.

Internal Process: This PE interacted, among others, with the Ontology Concepts Relation Engine, to find out the relations of the concepts that must appear in the graph. Then the concepts were drawn in the graph depending on the input information. The information about the context was used to make a deeper expansion of some parts of the graph, in order to highlight the most important concepts.

RQL Query Builder

The analysis and process of the queries carried out by the ML was based on the output of the trees generated by the Syntactic Parser based on the ML BNF Notation.

However the language chosen in the WIDE system in order to exchange the queries between the ML and the Agency was RQL. Later on, the Agency, analysed this RQL queries to map and apply them into the different information sources available in the project.

According to this, the PE RQL query Builder was responsible for the translation of the system queries generated by the ML into the language RQL.

Thus, an input like the following would generate a query as the one represented in Figure 5.16.

```
>L0 - userInput
>  L1 - contentSet\_qualifyingSet
>    L2 - content\_set
>      L3 - content\_type\_term : "DOCUMENT"
>      L3 - connector : "ABOUT"
>    L2 - qualifying\_list
>      L3 - qualifying
>        L4 - keyword
>          L5 - alphanumeric\_term : "TEST"
>      L3 - connector : "OF"
>      L3 - qualifying\_list
>        L4 - qualifying
>          L5 - keyword
>            L6 - alphanumeric\_term : "CAR"
>        L4 - connector : "WITH"
>        L4 - qualifying\_list
>          L5 - qualifying
>            L6 - binlog\_qualifying
>              L7 - lpar : "("
>              L7 - qualifying\_list
>                L8 - qualifying
>                  L9 - value\_attribute
>                    L10 - alphanumeric\_term : "AUDI"
>                    L10 - alphanumeric\_term : "BRAND"
>              L7 - binlog\_operator : "AND"
>              L7 - qualifying\_list
>                L8 - qualifying
>                  L9 - attribute\_op\_value
```

```
SELECT pt, mc, c1, c2, c3
  FROM {pt : $pt} @p {mc : $mc}, {rc1 } @w_a1 {c1 : $c1 }, {rc2} @w_a2 {c2 : $c2},
        {rc3 } @w_v1 {v1 : Literal}, {rc4} @w_a3 {c3 : $c3}, {rc5} @w_v2 {v2 : Literal}
 WHERE @p = "has_info_about" AND ($pt = "DOCUMENT")
     AND $mc = "TEST" AND mc = rc1 AND @w_a1 = "link_to" AND $c1 = "CAR"
     AND ((c1 = rc2 AND @w_a2 = "with_attr" AND $c2 = "BRAND"
       AND c2 = rc3 AND @w_v1 = "with_value" AND v1 = "AUDI")
      AND (c1 = rc4 AND @w_a3 = "with_attr" AND $c3 = "CYLINDER"
       AND c3 = rc5 AND @w_v2 = ">" AND v2 = "8"))
```

Figure 5.16: Example of RQL System Query

```
>                    L10 - alphanumeric\_term : "CYLINDER"
>                    L10 - operator : ">"
>                    L10 - numeric\_term : "8"
>              L7 - rpar : ")"
```

SQ Handler

This PE was in charge of the management of the different system queries generated by the ML. Each system query (SQ) had information about (i)the user query that generated it, (ii) the terms involved in that SQ, (iii) the expansion carried out in the ML and (iv) the results that are retrieved for them.

Some other useful piece of information handled by this PE was related to the Semantic Path. Each result of the graph was related to a specific Semantic Path, depending on the System query that generated this result. According to this, the SQ Handler, was responsible for the generation, storage and assignation of this Semantic Path. So, the SQ Handler preserved and updated this information all over the process.

Result Evaluator

This Processing Elements was in charge of analysing the results that were retrieved by the Agency (Annex V) includes a description of the return provided by the Agency). First of all, this PE linked each result with the appropriate SQ. After that, the Result Evaluator analysed the information retrieved within the result.This information was different depending on the information source that provided the result. However, the structure of the information was common and by the time the result arrived in the ML, this common structure was already built by the Agency.

The Result Evaluator assigned the result, if proceeded, to the appropriate node of the graph. However if the ML concluded that the retrieved result was an

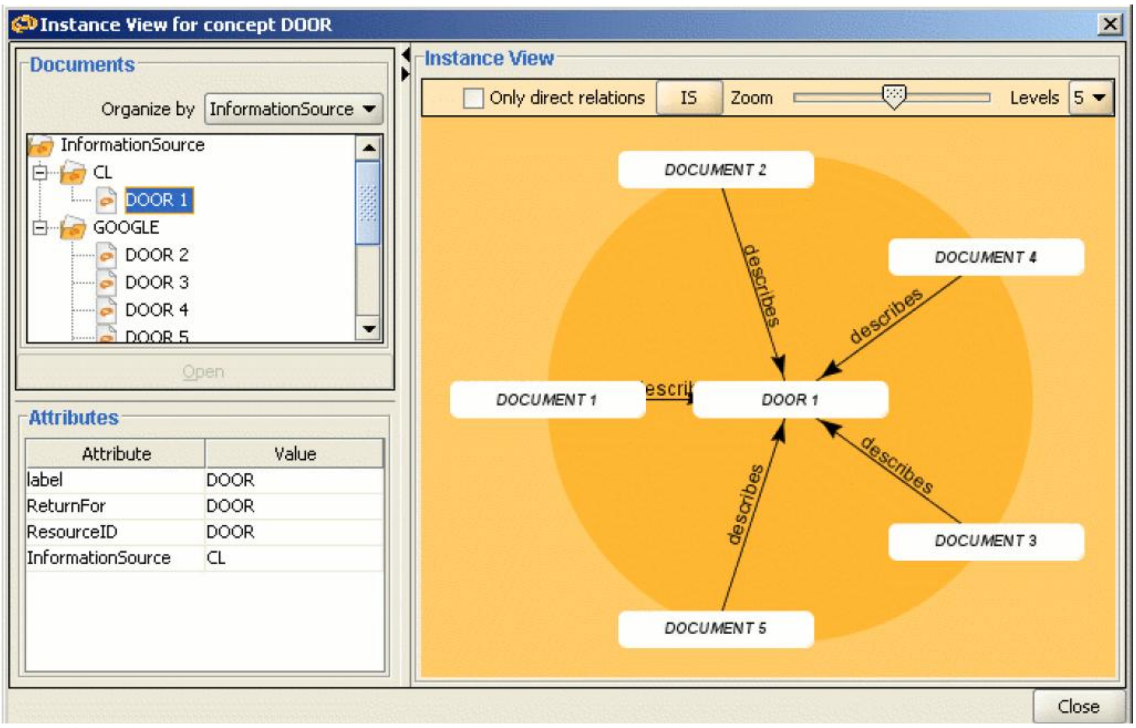


Figure 5.17: Visualization of the instance view

intermediate concept in the Information Source, it did not assign the result to any node. These kinds of results were visualized in the Instance View (see Figure 5.17), which allowed the user to easily identify to which concept belonged each instance.

Result Ranker

This Processing Element was invoked just before sending the results to the User Interface. Its role was to rank the results in the graph. This ranking was based on several criteria. Some of the most important were the following:

- The relevance of the metadata of each result. This metadata was compared against the information of the context (process, process step, task, concepts involved in the task, etc.). The more related they were, the highest the rank was.

- The similarity between the understanding of the Information Source and the ML one. As the agency carried out a mapping between the terminology of the CL and the ML, it could happen that the ML had a different interpretation of the same concept. Thus, the reporting PE used this similarity distance to rank the results within each node of the graph.

- When available, the context information sent together with some results, was used in order to evaluate the similarity of the context with the context of the same concept in the ML. This was used to perform a more accurate filter than the terminological mapping made by the Agency using the synonym lists provided by the ML and the IS. So, if the Agency mapping mechanism made possible the retrieval of a result of an “automobile” when asking about a “car”, the analysis of this context allowed to measure how close was the interpretation of “AUTOMOBILE” to the ML interpretation of “CAR”.

This PE was also responsible for filtering the results with the aim of deleting the information that was superfluous from the user point of view. For instance, the information about the context of a term (the surrounding concepts according to the IS interpretation of that term) was not shown to the user. Thus, the RDF fragments that denoted this extra-information were removed from the results by the reporting PE.

External Resource Handler

One of the main objectives of the WIDE project was to test the semantic emerging technologies in order to analyze their efficiency and some possible lacks or needs of the current status. According to this philosophy, the idea was to access different semantic information sources. However, the state of the art of the available information sources made no possible to have any IS different from the one developed within the project.

In spite of this, during the last period of the project, Google Labs made available a thesauri quite powerful, named GoogleSets. For instance, when this Thesauri is invoked asking about the term “PORSCHE”, it retrieves a list of what it understands as similar terms: SEAT, AUDI, MERCEDES, etc.

Thus, whenever the ML did not know anything about a term specified by the user (i.e. KIA), it tried to find out more information about it. According to this, if the user asked something about a “KIA”, the reporting PE accessed this external resource via the Agency and retrieved the list of “similar terms”. Then, it checked the ML OWL-rep in order to find out anything about them. In case it discovered that some of the terms were known and the Knowledge Engineer had also considered that they were somehow related, the PE reached a positive conclusion, which was the output of this PE.

Ontology Concept expansion

One of the facilities provided by the UI to the user is the possibility of selecting one of the nodes of the graph, and expand it.

This processing element, "Ontology Concept Expansion" was in charge of receiving the name of the node that had to be expanded and infer the OWL-Rep in order to expand it.

This expansion was based on the concepts that were close to it in the graph, as its subclasses, the classes that belonged to the range of the properties where that class was in the domain, and so on.

Log Handler

One of the objectives of the ML was to learn from the user experience. In order to do this, this PE had an important role, since it was in charge of storing some incidences that occurred during the interaction of the user with the system. For example, this handler stored information about the unknown concepts that were included by the user queries, the preferred terminology by the user, the invocations made to the GoogleSets, the queries that do not get any result, etc.

This information was used by the Knowledge Engineer in order to update the ML Ontologies.

Concepts Linker

This PE was in charge of the linkage of two concepts. In order to do that, it used the Ontology Concepts Relation Engine to find out the relation between those concepts. Once this was done, it tried to translate this into something that was more easily understood by the user.

For instance, if the Concepts Linker PE was asked to relate "car" and "engine", once it has learnt that the relation among car and engine was "has_engine", the Concepts Linker retrieved the following string "car WITH engine".

5.4.3 ML-GW

This section takes care of the last kind of entities that compose the ML Architecture: the Support Processes. Those processes, as has been previously stated, were in charge of the interaction with the rest of the WIDE subsystems in order to make available to the user the service that the WIDE tool is aimed to provide.

Paying attention to the lines detailed in the reference model, six Support Processes (SPs) were developed:

- The system was able to learn about the context (task carried out by the user, his/her profile, the process he/she could be trying to realize, etc) giving to the user the information about the domain that could be of his/her interest. The first SP describes the contribution of the ML subsystem concerning this issue.

- The second SP was focused on the support offered by the system in order to help him in the development of a query that expresses the information he/she is needing at that specific moment.
- The third SP was related to the process carried out by the system in order to process, interpret, semantically understand and expand the query defined by the user.
- The Search Support Process was focused on the ML contribution to the process that started when the queries generated by the ML were sent to the Agency and finished when the results were retrieved.
- The process that took care of the Results Evaluation was also supported by the ML.
- After the internal evaluation of the results, the system prepared the visualization of the information retrieved.

Domain Contextualization Support Process

This SP was focused on supporting the user in the learning and querying process of the domain. The user was able to see the graph of his/her domain of interest (based on the task and process selected, if any) but also ask about the relation among several concepts. This last action implied the generation of a new graph (which is translated into the generation of another graph). Then, he/she was also able to interact with the shown graphs in order to expand (ask for more information) some nodes.

In order to support this, the reporting SP made use, among others, of the following Processing Elements:

- "Graph Generator": this was used to generate both kind of graphs that were used in this process.
- "Ontology Concepts Relation Engine": which was used in order to feed the "Graph Generator".
- "User Knowledge Handler, Personal Dictionary Handler and User Profile Based Mapping": to facilitate the translation among the user and internal terminology.

- “Ontology Concept Expansion”: to expand a graph, paying special attention to a specific node of the graph. Query Development Support Process

Query Development Support Process

This Support process made use of the same facilities used by the previous one, plus the “Concepts Linker”. The usage of this Processing Element made possible the drag and drop facility that allowed the user to drag a node and drop it in the query bar. This provided an automatic completion of the query by including the dragged concept in a semantically coherent way.

Semantic Query Processing Support Process

The Semantic Query Processing SP was in charge of the process that started when the user query was received by the Meta Level and finished when the different System Queries (SQs) generated by the ML were sent to the Agency.

- First of all the ML checked if the user query was syntactically correct, according to the BNF grammar. This was done with the “BNF Syntactic Parser”. The output of this process was an AST tree.
- Once this was done, the ML translated the terms of this tree according to the internal terminology. This was carried out interacting with the “Personal Dictionary Handler” and the “User Profile Based Mapping”.
- Once the terms of the AST tree were described using internal terminology, the Semantic Query Processing Support Process analysed this tree but from a semantic point of view. This was done by the “Semantic Coherent Parser” and the “Semantic Interpreter & Query Expansions” PEs. During this process the processing elements were (i) checking if the query was coherent, (ii) analysing which terms are known by the ML Ontologies and (iii) trying to expand the queries into different system queries, (iv) and finally transforming the input AST tree into several trees.
- Finally, using the “RQL Query Builder” PE, the SP parsed the generated trees in order to create the final RQL queries that were sent to the Agency.

Search Support Process

The process tackled by this SP started when the System Queries generated by the ML were sent to the Agency and finalized when the results reached the Meta Level. During this process, the role of the ML was limited to the assistance to the

Agency in the terminological mapping among the ML internal terminology and the terminology of each Information Source. This work was mainly realized using the "Synonym Provider Processing Element". For instance, the ML could send together with the queries or in an asynchronous way a list of synonyms for the term "car": "automobile" and "passenger vehicle".

Result Evaluation Support Process

When the results were retrieved, the Result Evaluation Support Process made use of some Processing Elements ("Result Ranker", "Result Evaluator", "SQ Handler", "Ontology Concepts Relation Engine" and "Process Support Handler") in order to analyse and rank the results. The ranking mechanisms are described in the subsection 5.4.2, which details the processing element that carried out this evaluation.

Result Presentation Support Process

This process was in charge of the following tasks:

- Generation of the graph with the main concepts of the search. In order to do this, the SP, used the "Graph Generator Processing Element".
- Assignment of the results to the ranked specific node of the graph. This was done thanks to the use of the following PEs: "SQs Handler" and "Result Evaluator".
- The preparation of the semantic path. For each result, a semantic path was included to show the main concepts each result was related to. The "SQ Handler" Processing Element was used to complete this task.

Collaborative Interaction Support Process

This seventh service is included here, but is not considered as part of the list of the core services. This service implemented in WIDE was related to the support for collaborative work among different users. It covered the sharing of documents and the remote control and sharing of the application user interfaces. It therefore was not directly related to our field: Multimedia Information Retrieval and the reference model that we provided.

The collaborative interaction support process was mainly implemented in the UI and Agency Subsystems. However the ML Processing Element "User Profile Based Mapping" was used in order to allow the graph sharing among the users.

Thus, when a user graph had to be shown by several users, the ML provided the terminology mapping in order to hide each user terminology.

5.5 WIDE SMD Evaluation

The validation of the Meta Level was done through its integration in the WIDE system. Regarding the evaluation of its behaviour, it was performed following an strategy based on three action lines: (i) extraction of performance features, (ii) evaluation of the whole WIDE system by end users and (iii) evaluation of the opinion of the knowledge engineers regarding the benefits and drawbacks of the Meta Level.

Regarding the performance of the Meta Level the following evaluation framework was used: a conventional PC with 512MB of RAM memory and 1.2 GHz where different batch files allowed the launching of each of the evaluation tests. In that framework, the following Meta Level KB items were installed: a set of 7 interrelated ontologies with more than one thousand concepts and a BNF grammar of more than 200 lines (excluded comments). Regarding the off-line services, the Meta Level was able to infer new knowledge out of the ontologies and parse all the information into the SQL repository in about 5 minutes and a half, and a java parser was generated for the BNF grammar in less than 1 minute. Concerning the online services, we summarize here some of the most significative aspects:

- The average time required to perform a query expansion, limiting the number of system queries created out of each query to 4 was 1,4 seconds. Part of this time (0.09 seconds) was devoted to the validation of the query with respect to the BNF grammar.The number of queries employed in the test was 250.
- The average time used for the transformation of a system query into the RQL queries employed by the Agents Platform was 0.22 seconds. The number of queries employed in the test were 1000.
- The average time for the generation of a graph showing the interrelations of a set of 9 concepts was 3.2 seconds. Two hundred sets were randomly generated from a list of 1200 concepts, out of which 150 were unknown to the system. The average graph created had 19 concepts and 32 relationships.
- The average requested time to rank a set of 50 results coming from three different information sources was 1.1 seconds. The number of result sets randomly generated out of real result sets was 200.

All the presented results are just related to the performance of the Meta Level and do not include the communication time required to receive the request or send the answer to the other modules of the system.

Regarding the opinion of the final users, a deep evaluation of the whole WIDE system with more than 20 users was performed in the two companies involved in the project. The evaluation implied the conduction of real experimentation sessions and the fulfillment of opinion questionnaires. Most of the functionalities included in the evaluation were not directly provided by the Meta Level but semantically supported by it (e.g. utility of the results retrieved, query development facilities and so on). The results of this evaluation showed a globally satisfactory feeling. As a summary, we include some of the most significative results: the time reduction to find the required information was 25% for engineers and 10% for designers, whereas the graphical representation of the results based on its nature and their interrelations, and the new opportunity to tackle different information sources at once were positively evaluated.

Regarding the evaluation of the professional users, i.e. knowledge engineers, they highlighted the feasibility and convenience of centralizing the semantic information in a unique component that provided all the semantic resources required by any other module. Concerning the implementation details, the simplicity of the use and update of the ontologies with respect to the achieved added value was appreciated. On the other hand, the upgrade and management of the BNF grammar was highlighted as one of the most expensive tasks regarding the maintainability of the Meta Level.

6 RUSHES use case: Semantic Middleware to enable automatic analysis techniques in large repositories of un-edited material in the domain of a broadcaster

This chapter is focused on the second deployment of the SMD proposed in this thesis. The description of the successful integration of this deployment in a real scenario aims to show the feasibility of the proposed architecture and the exposed criteria for its design. Providing the main implementation details which also aim to provide some useful guidelines for those people in charge of the media asset management systems who have to face the extension of their current systems with content-based functionalities.

According to the reference model for content-based multimedia information retrieval that we have presented (see Chapter 3), the services provided by the WIDE SMD (see Chapter 5) were mainly related to the "Search Line" and the "Browsing Line". However, the SMD deployment that we present in this chapter is mainly devoted to the semantic enhancement of the "Storage Line". Therefore, this chapter provides design and implementation details of an SMD that aims to improve aspects such as the ingesting of metadata coming from external sources, automatic and semi-automatic metadata extraction, novel methods for manual annotation and so on.

In this introductory section we describe the system, its motivation, objectives and architecture.

The rest of the chapter is structured as follows:

- First of all and with the aim to contextualize this work, the whole RUSHES system, its motivation, objectives and architecture are briefly described.
- Once we have highlighted the position of the RUSHES SMD in the global architecture, we provide a description of the main functionalities and

services provided by the SMD.

- After this, we describe the criteria handled during the design process.
- Next to this, and with the purpose of supporting similar deployments, we provide some implementation details.
- Finally, we summarize the evaluation and validation processes.

6.1 RUSHES system

RUSHES was a project within the EU's 6th Framework Program [1] related to the knowledge extraction based on semantic content annotation, scalable multimedia cataloguing, interactive navigation and innovative retrieval techniques. Specifically, the RUSHES system designed, implemented and validated a system for indexing, accessing and delivering raw, unedited audio-visual footage known in broadcasting industry as “rushes”. In order to accomplish this, many different technologies were required, ranging from multimedia analysis to multimedia search, to user interfaces, as well as to supporting models for taxonomies and metadata. Thus, the project tested and validated a proof of concept of the incoming semantic driven multimedia retrieval.

6.1.1 Motivation of the system

As we have stated in the introduction of this work, a massive spread of videos in the video databases means that a quick and efficient search for the right kind of content is posing an increasing challenge. In the broadcast industry as well- as TV companies know very well- up to several hundred hours of video material must be captured and catalogued to enable rapid access when it comes to reprocessing. Once the material is processed and used, most of the original content, remains unedited. The existing audiovisual raw media database contains invaluable resources for both professional and home users. However, the lack of enough high-level annotation that can be made both human readable and suitable for machine processing limits their usage and value.

In this context, the main motivation brought by RUSHES was the development of seamless indexing, search and retrieval of content, specifically applied to archives of raw media material (rushes), to ease in-house postproduction in both professional and home environments.

[1] RUSHES European project (FP6-045189) http://www.rushes-project.eu/

Therefore, RUSHES focused on the extraction of knowledge from raw video material and the semantic structuring of raw media repositories. The main reason to do this was that, due to the cost of the annotation, the raw annotation material is usually not employed and, in the best of the cases, relegated to the deepest servers of the MAM. Any improvement in this situation would lead to new business opportunities and enrich the current postproduction process.

6.1.2 Objectives of the system

The central objectives of the RUSHES project were:

- Provide a system for the processing of rushes, semantic annotation and cross media retrieval based on existing sources.
- Enable indexing, search and retrieval of rushes archives to ease in-house postproduction or reuse in a media professional environment.
- Provide a new approach to handle the multimedia annotations to promote the forthcoming content-based multimedia information retrieval.

Regarding the technical objectives we highlight,

- To develop techniques for automatic low-level metadata generation and content indexing.
- To apply learning and reasoning techniques for automatic semantic inference from RUSHES.
- To get an automatic semantic annotation and content organization on top of a defined metadata model, using multimodal analysis, inferencing through reasoning tools and semantic context matching. The goal is to enhance the metadata extraction to obtain a relevance feedback in order to improve the annotation process.
- To conduct multimodal analysis for semantic inference.
- To use relevance feedback for semi-automatic semantic-based annotation.
- To identify key human factors issues in relation to accessibility of new digital content services, with particular emphasis on inclusive accessibility, independently of user's capabilities.
- To develop smart interfaces for metadata creation, media retrieval and repository navigation, based on semantic representations, automatic media summarisation and content-based clustering based on extracted features.

6.1.3 System architecture

RUSHES architecture was based on a Service-Oriented Architecture (SOA) with loosely coupled services. The architecture was constituted by a number of service domains, one of those, the SMD we present here. Each service represented some vital functionality to a RUSHES System. Functionalities exposed through services included: storage, content processing, training of classification models, searching, and manual data annotation.

Figure 6.1 shows the high-level service domains in the RUSHES architecture. The boxes represent service domains, and the arrows show the dependencies between the domains. The service domains were characterized by certain high-level requirements which defined the responsibility of each domain. The service interfaces were usable without any knowledge of the underlying implementation of the component(s) exposing the service(s).

With the purpose of understanding the relation of the RUSHES SMD deployment, **also called Metadata Model or MDM**, we provide a brief overview of the main service domains by means of a description of how content and users interact with the RUSHES architecture.

In order to connect a user to multimedia content, the content had first to be analyzed and stored in a searchable format. A user could then get access to multimedia material through querying the RUSHES system via a user application. In the following description service domains are introduced in **Capitalized Bold**.

The main entry point for content into the RUSHES system was via **Multimedia Access** which was a layer built on top of a Content Providers' existing storage system shown as **Content Provider Multimedia**. A **Connector** found and detected the multimedia content to be inserted into the RUSHES system and pushed it to **Content Capture and Refinement (CCR)** which analysed the content and extracted a content description with the semantic support of the **MetaData Model**. As part of the content extraction some metadata needed to be persisted in storage (for example thumbnails or summaries to be displayed by the User Interface), and this was shown as **System Generated Multimedia**. The content descriptions were then indexed by one or more search engine components in the **Search Engine domain**.

The main entry point for users was the **User Interface** where users could do manual annotation, pose queries, browse, and give feedback on content. Queries were directed to **Query and Result Refinements (QRR**) which took care of the communication with the Search Engine and merged and returned results to the User Interface. Query and Result Refinements made use of CCR to process multimedia queries and generate content descriptions according to the same MetaData Model which was used prior to indexation by the Search Engine.

Based on feedback from the user, parts of the system could be updated

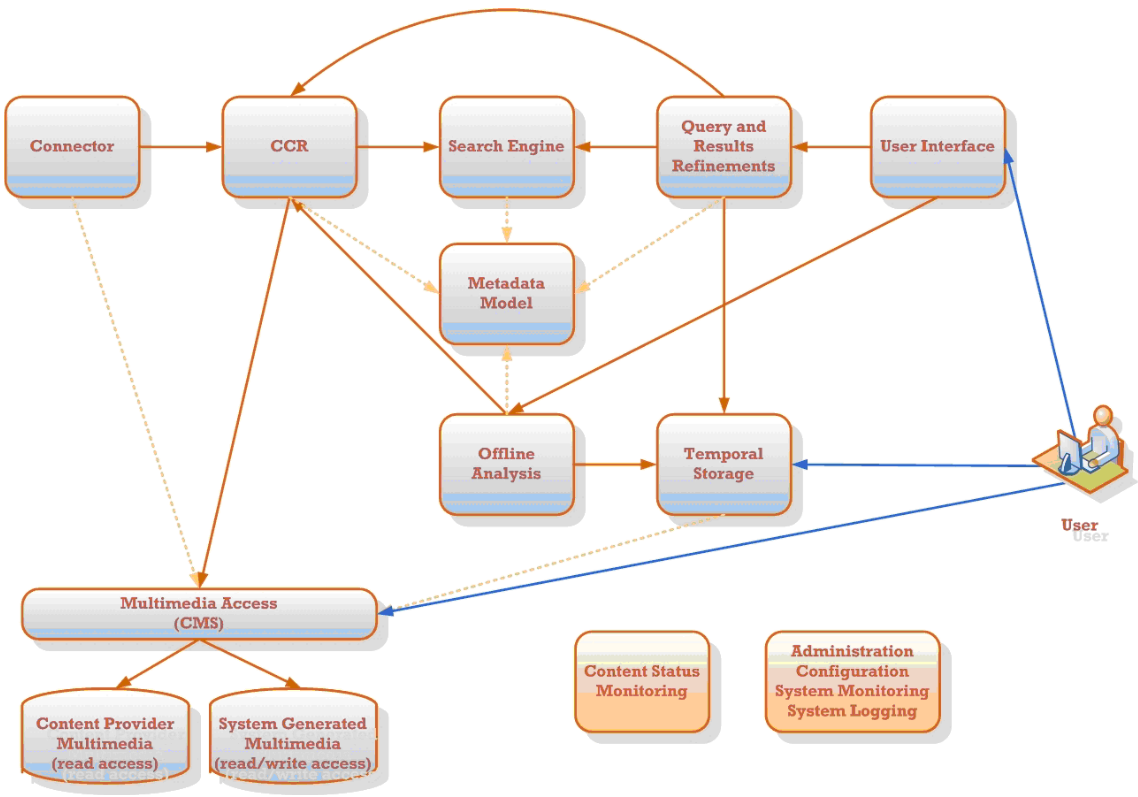


Figure 6.1: Architecture of RUSHES system

(for example by retraining multimedia classifiers). This training was done by components in **Offline Analysis**. The **Temporal Storage** facilitated the temporary storage of content material allowing different components to access it within the same process.

The **Content Status Monitoring** domain and the **Administration, Configuration, System Monitoring, System Logging** domain represented the Administrative Layer composed of services that were pertinent for ease of use in an industry strength system.

6.1.4 Metadata Model, the SMD in RUSHES

Metadata Model is the name of the deployment of the SMD in the RUSHES project (see Figure 6.1). From the semantic perspective it is the key element in the architecture, interacting with many other modules to provide the services and information that those components could require.

6.2 Description of the RUSHES SMD

In this section we provide a global overview of the functionalities provided by the RUSHES SMD in order to achieve the objectives of the system presented in the previous section. In order to do this, we provide a global overview of the functionalities followed by their contextualization in a scenario and a summary of the provided services.

6.2.1 RUSHES SMD functionalities

In the following subsections, we list the functionalities of the SMD.

To prepare the multimedia assets to be retrieved

If we wish to support the content-based multimedia information retrieval paradigm, one of the first issues to solve is to ensure the availability of information about the content or metadata. This is usually a complicated and expensive task. In RUSHES is even more difficult, since the material employed is un-edited and the information known about it is very poor.

In this context, RUSHES SMD had a key role for the asset indexing process. This process, represented in our reference model in the “Storage Line”, started when a multimedia asset arrived in the system and ended when the system had stored this asset and all the metadata available or generated during that process in a searchable format.

In the following paragraphs, and using Figure 6.3 as graphical support, we include some of the functionalities provided by the RUSHES SMD to enable this indexing process.

- To gather existing metadata: Even for the case of the raw material, by the time the asset arrives in the system (step (1) in Figure 6.3), some metadata is available. Due to the disparity of the metadata schemas employed in the professional broadcasters systems (Multimedia-Research-Group, 2004), this metadata can be expressed in diverse formats. The RUSHES SMD was responsible for the gathering of this metadata (step (2) in Figure 6.3) and its adaptation to the internal format of the system.

- To provide semantic information during the video analysis: Once a new un-indexed video was notified by the Connector Domain, the Content Capture and Refinement (CCR) started with an execution graph the analysis process to automatically generate metadata. In order to do that, some of the

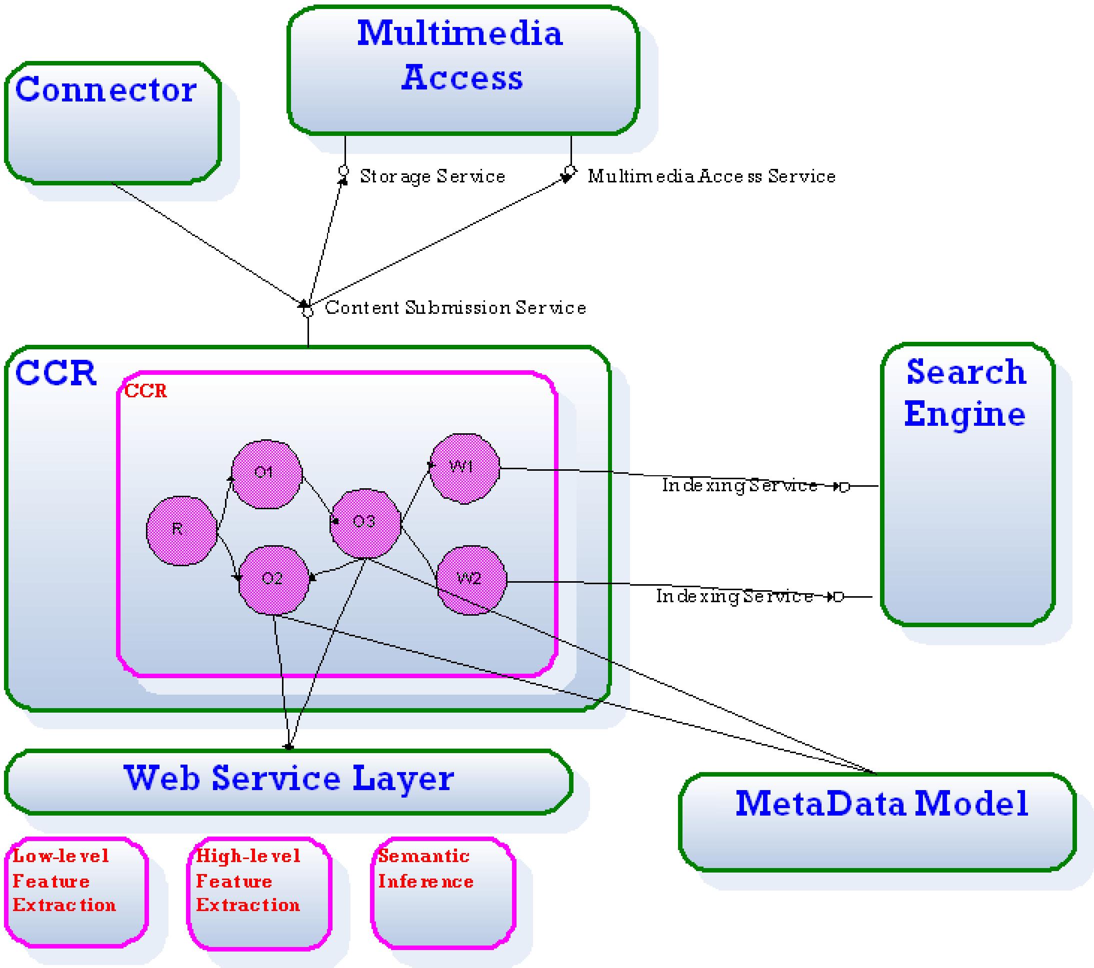


Figure 6.2: Logical architecture of the CCR Service Domain

operators that composed the CCR (i.e. Multimodal Analysis) needed as an input semantic information provided by previous modules (step (3) in Figure 6.3). In RUSHES this information was centralized in the SMD (called MDM). Figure 6.2 shows the CCR domain and its relation with the Metadata Model.

- To generate information storage and metadata generation during the analysis process: According to the information previously stated, the CCR provided an integrated environment for processing multimedia content such as image, audio and video through the implementation of an execution graph. The RUSHES SMD was continuously invoked by such graph. As a result of this, the SMD was able to perform the previous functionality and, beyond that, it stored all the information known about that asset according to the internal semantic model (step (3) in Figure 6.3). Using this information compilation and thanks to the technology employed in the SMD KB (see subsection 6.4.1), the SMD was able to generate new metadata by applying fuzzy reasoning techniques (step (4) in Figure 6.3).

- To make the information available: The final motivation of the metadata extraction was its usage by the user. In order to do this, this metadata was required by different modules. First of all, the search engine, which indexed the metadata to optimize its accessing and to provide advanced retrieval mechanisms (e.g. scene based retrieval), the annotation tool that was employed to enrich manually the metadata and diverse clustering-based result navigation tools. The SMD was responsible for the provision of the information in such a way that the requirements of each of the modules was achieved (step (5) in Figure 6.3).

To enhance the user experience

- Recommendation based retrieval: RUSHES system implemented an engine that provided recommendations to the user according to the results of the query. The recommendation was based on two approaches: other user actions and concept based. For this last option, RUSHES SMD provided the semantic information required by the recommendation engine to perform its task.
- Annotation Wizard: The annotation tool of the system included an annotation wizard that facilitated the completion of the annotations based on the information modeled in the RUSHES SMD.
- Relevance Feedback: The user interaction was employed to improve the existing annotations in the system. The Relevance Feedback module of the system, was partially based on the knowledge of the domain. This information was provided by the SMD.

6.2.2 SMD functionalities in an scenario

In the following we include a use case integrated in an scenario in order to highlight the SMD services mentioned from a more generic perspective.

Jess is a journalist, working for a broadcasting company, and on assignment in Paris documenting the protests and demonstrations going on. He returns with an extended volume of footage compiling of both videos and photographs. Some seconds of the video are employed for the news just after Jess arrives with the information. The complete video goes into the RUSHES system in order to ensure its automatic preservation and availability for the future.

Once Jess introduces the memory card of its Panasonic PS2 professional camera in the indexing station, not only the video and the audio is available,

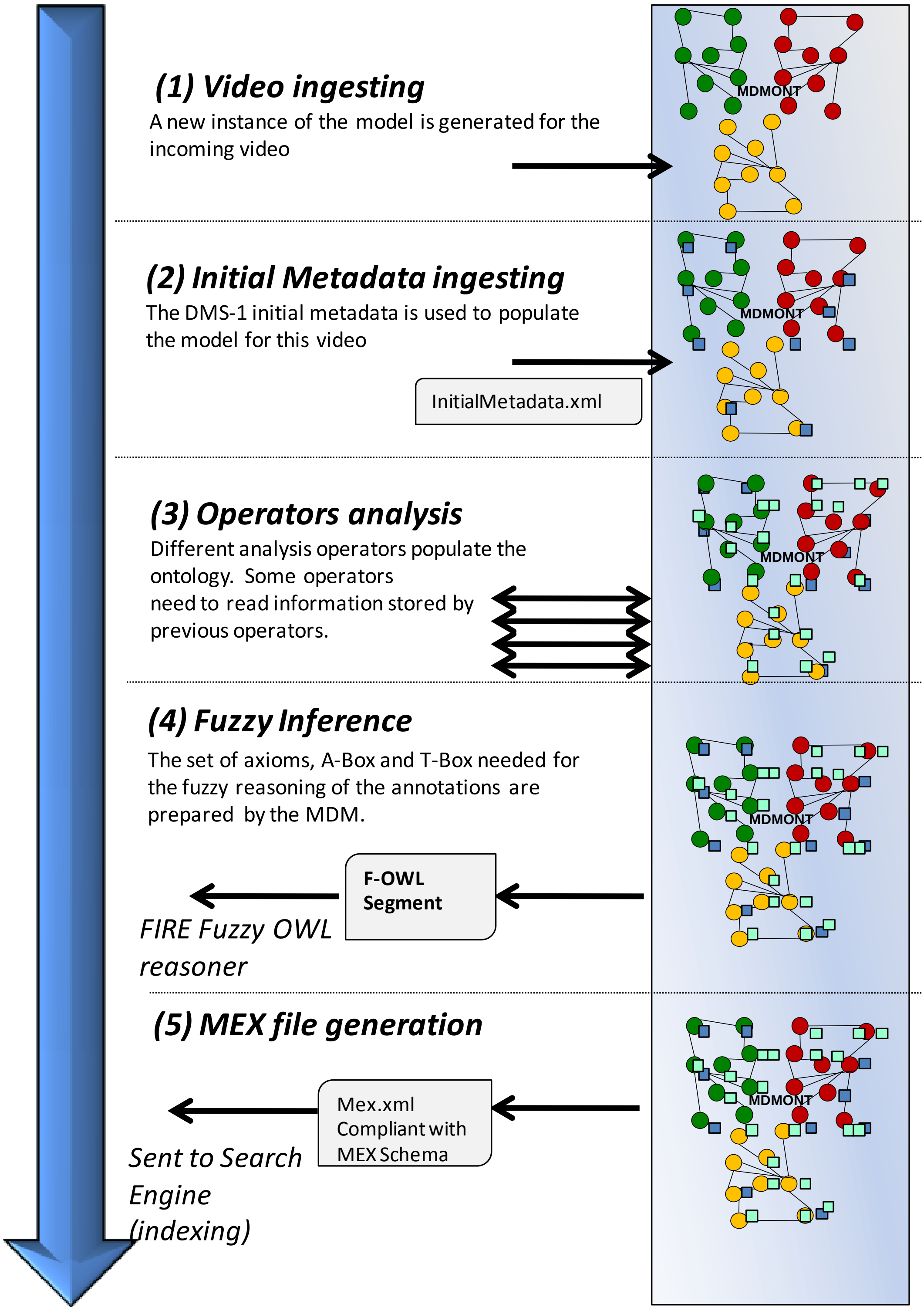


Figure 6.3: Information storage and metadata generation in RUSHES

but also the metadata embedded by the camera. This metadata is composed of DMS-1 (Smith and Schirling, 2006) compliant annotations and embedded in a MXF (Material Exchange Format) (Haas and Mayer, 2001) container. The RUSHES SMD is in charge of parsing the metadata provided by the PS2 camera of the organization into the internal model.

Regarding the video and audio generated by the Panasonic cameras, they are processed by the analysis algorithms of the broadcaster. For example, the OCR text detector, a module to split the video into shots and detect their key-frames and the module to provide the probability of average number of faces in a shot, and so on. During this process, the RUSHES SMD **stores and relates the information provided by those algorithms** following the internal semantic model of the system. Beside this, the SMD also **provides the input needed by the face detector module** (e.g. list of the identifiers of the key-frames). At the end of the analysis process, the S**MD employs a fuzzy reasoner to complete the annotations** (e.g. based on the number of faces, the absence of vegetation provided by the vegetation classifier, and the presence of text on the bottom area of the screen, the SMD annotates certain segments of the videos with the tag "Demonstration front" with a certain probability, which is based on the a weighted combination of the probabilities of the rest of the parameters.

The SMD **homogenizes the semantic annotations** (e.g. all the annotations are expressed according to time codes, avoiding the heterogeneity of the different analysis modules that expresses the annotations according to the second, number of frame, ...). Metadata is synchronized by an external module and then the SMD is asked to **send the annotations to the professional engine(s) that must handle them**. In this case, this component is the FAST ESP engine, which is the key search-component in the Media Asset Management of the Jess' Broadcaster. The SMD **parses only the searchable annotations into a MEX** (Multimedia Exhange Format) file. This MEX file, which is an XML document, is compliant with the schema defined in the search engine to index the annotations. This transforms the video into a retrievable document in the system.

Six months later, Leticia, a journalist employs the RUSHES search interface to look for demonstrations and retrieves the asset that Jess generated. During this retrieval process, the SMD has indirectly helped Leticia by **supporting a recommendation system** in the user interface that has suggested her to employ the term "demonstration head" to refine the search. Once she has found the video, following the company policy, she spends some minutes annotating it in order to increase its visibility. An **annotation wizard** supported by the SMD helps her to complete the location of the demonstration and the names of the most relevant people of its front.

6.2.3 Summary of services

The main services provided by the SMD were:

- Metadata Adaptation and Information Mediation. First of all, the video ingesting and analysis were performed. As a result, diverse annotations belonging to different semantic abstraction levels were extracted and indexed by the search engine. During this process, the SMD carried out parsing processes to convert external data schemes (e.g. DMS-1 metadata (Wilkinson, 2003) coming from a Panasonic camera) or datatypes (e.g. timestamps) into the internal ones defined in the system.

- Knowledge Channeling during Multimedia Analysis. The incorporation of analysis modules to the search engine was the main source of changes from the architectural point of view. An external module was in charge of an execution graph that invoked the different analysis modules. The SMD supported this task by carrying out the conduction of the knowledge during the analysis process. This process involved expert modules (low-level analysis operators, concept detectors, and Bayesian network classifiers) that worked at different steps of the video analysis process. The information generated by some modules was needed by the remaining ones. Some of them needed additional pieces of the semantic model in order to perform their analysis. During this process, the SMD was responsible for the persistency and availability of the intermediate information generated (e.g. extracted low-level features or representative key-frames) and the semantic metadata obtained (e.g. a list of faces recognized for each key-frame).

  Furthermore, the SMD stored every piece of information generated for each video, preserving the semantic meaning of it by linking it with the semantic entities gathered in its knowledge base.

- Fuzzy Reasoning based Metadata Generation. Once the analysis of the video was finished, the SMD semantic repository was populated with all the information generated by the different operators. This information was related to the structure of the asset (e.g. number of tracks that composed the asset, main shots of the video, representative key-frames and so on) and with the content (e.g. number of faces present in each frame, type of audio, an essence of vegetation in a shot, and so on). These annotations, often linked to a confidence value, were inferred by the SMD in order to extract new annotations.

- Annotation tool, Recommendation and Relevance Feedback. A wizard for self completion of the annotation tool, the recommendation system of the search interface and the relevance feedback modules implemented in the system employed the semantic information modeled in the SMD KB to perform their tasks.

6.3 Key design criteria of the SMD

In the following subsections, in a coherent view with the ones stated in Section 3.3, we summarize the criteria taken into account during the design process of the RUSHES SMD.

6.3.1 RUSHES SMD KB

The RUSHES SMD KB was mainly composed by a set of rules used during the mentioned fuzzy reasoning and a set of interrelated ontologies that covered the news domain and the technical and descriptive aspects of the multimedia assets. Regarding the key design criteria, we highlight the following.

People in charge of the knowledge base: The responsible for the SMD KB was the archive manager of the broadcaster. His work was supported by a broadcast systems engineer. In the design step the cooperation of a multimedia expert was required.

Identification of the domain: The domain of the RUSHES SMD KB was composed by three distinct subdomains. First of all, the domain of the content (e.g. news content). Secondly the domain related to the technical description of the multimedia assets. This domain was related to the services provided by the SMD during the analysis of the video. Finally, a third domain, not considered at the first steps of the design process, was the domain of the descriptive understanding of the assets. The comprehension of the organizational requirements showed that the technical description of the multimedia should be complemented with a more ambitious descriptive language that enabled the interaction with other processes of the organization apart from the retrieval.

Technology of the SMD KB: Following the same criteria than in the WIDE SMD deployment, the language chosen for the information modeling was in RUSHES SMD KB was OWL-DL. Regarding the rules, they were expressed according the syntax employed by FIRE, the fuzzy reasoner used and described by Simou and Kollias (2007).

Reusing shared knowledge: For the domain related to the content, the first approach was to integrate an extension of the LSCOM Lite ontology proposed by Neo et al. (2006) with the thesauri that the organization already employed for the content annotation and classification. However, the final version of the ontology was quite far from the LSCOM lite implementation and much more close to the organization reality.

For the technical description of the multimedia assets, the SMD KB in RUSHES incorporated an OWL implementation of the detailed A/V profile (DAVP) of MPEG-7 proposed in the following publications (Bailer and Schallauer, 2006; Bailer et al., 2007).

Finally, the generic description of the assets was done following the industry design standard SMPTE 380:DMS-1 (Descriptive Metadata Schema) that belongs to the MXF (Material eXchange Format) normative family. This was done by implementing the first OWL ontology for that schema (Marcos et al., 2009).

Composition of the knowledge base: Regarding the composition of the Knowledge base, the three ontologies were mapped in order to provide the services while keeping a global perspective. On one hand, the link of the DMS-1 ontology with the MPEG-7 was done through the temporal decomposition of the profile, where the two ontologies intersected. On the other hand, the mapping between the DMS-1 ontology and the domain model was realized by linking the dictionaries proposed by the SMPTE 380:DMS-1 and the main concepts of the domain ontology.

Update: Some of the PEs were highly dependent on the implementation of the SMD KB, specially on the DAVP ontology. A file registered the dependencies between the PEs and the ontologies of the SMD-KB. This file provided information about the type and degree of the dependency and was checked before applying any change.

6.3.2 RUSHES SMD IE

RUSHES PEs were mainly related to the effective completion and parsing of the ontologies (based on the Protégé and Jena APIs), the fuzzy reasoning over the information, the building and transformation of the information exchange formats, and the parsing of the XML files of the different incoming pieces of information (e.g., DMS-1 compliant metadata coming from the camera, information provided by the shot boundary segmentation module, and so on).

Maintainability: RUSHES PEs were developed following a similar approach to the defined for RUSHES SMD IE. For instance, the *XML Transformer PE* encapsulated more than 20 PEs to handle the XML formats that were parsed and stored in the SMD KB.

Exchange information structures : SMD IE PEs (e.g. *Data schemas mapper*, *Fuzziness Handler*, *Semantic Integrator*) employed mainly three set of structures for information exchange. First of all, the regular set of structures imposed by the conventional ontology APIs. Secondly, the specific format for the A-Box and T-Box imposed by the fuzzy inference engine. Finally the format employed for the communication between the components of the whole system. That format was called MEX (Multimedia Exchange Format) and was a customized XML version of the DMS-1 standard.

Make use of available resources: the *Ontology Handler* and the *Concept Relator*, among other PEs, integrated Protégé and Jena APIs for the SMD KB parsing. The *XML Transformer* employed XML processing APIs for the management of the information structures, and the *FIRE fuzzy reasoner* was based on the FIRE software (Simou and Kollias, 2007).

PEs network interoperability: all the PEs were implemented in Java and they provided asynchronous communication mechanisms.

6.3.3 RUSHES SMD GW

RUSHES SMD GW made available the following PSs: information conduction (storage and provision) during the analysis process, ensure semantic interoperability among the modules by performing the semantic translations between different information exchange structures that were required, semantically support for the recommendation system of the annotation tool.

Flexibility: Both the offline and online SPs provided by the SMD GW were based on a service oriented architecture and implemented with SOAP.

PEs invocation: In order to increase the performance, and assuming that the real time analysis of the videos was not affordable, all the PEs were installed in the same machine that the SMD KB. However this is not always the case for the SMD GW PSs.

Synchronization and status management of each service and global status management: The main constraints were related to the service offered during the analysis of a video. In that specific service, the SMD GW kept a process alive for each of the video that was in charge of the synchronization and communication with all the PEs and external modules involved during the analysis process. This information was made available for external entities that could require it (control console, UI component).

6.4 Implementation details of the SMD

This section covers the technical implementation of the SMD in RUSHES. Figure 6.4 shows a graphical description of the components of this SMD deployment. The services of the SMD GW layer are grouped in a coherent view with the reference model expressed in 3.2.

In the following subsections, the implementation details of each of the layers is tackled.

6.4.1 MDM KB

RUSHES SMD KB was composed by three interrelated ontologies and a set of axioms employed by the fuzzy reasoner.

In this subsection we cover some common implementation details of the ontologies. After this, we provide a brief description of each of the ontologies. Finally, we include an overview of the axioms.

Ontologies: implementation details

In the RUSHES system, the model was implemented in OWL (Grau and Motik, 2008) using the Protégé OWL editor (Knublauch et al., 2004) (see Figure 6.5).

OWL standard did not cover the requirements related to the storage of the fuzziness information. However, RUSHES SMD had to handle the uncertainty provided by the analysis modules in order to employ it later in the fuzzy reasoner. In order to solve this issue, the annotations resource of the OWL language was employed. According to this, Figure 6.6 shows a tiny ontology, where the annotations were used to express that “The key frame instance named as "10392" contains an instance of Face named as "Tony Blair" with a probability of 0,78”.

The three ontologies were related to different aspects of the domain of the application. First of all, an ontology that descriptively modeled the multimedia

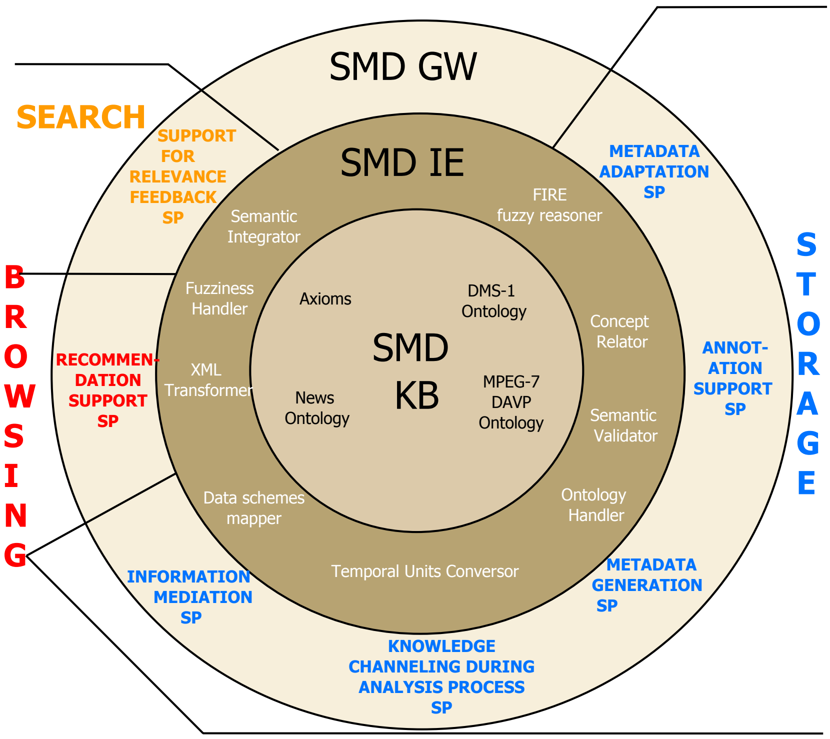


Figure 6.4: RUSHES SMD architecture

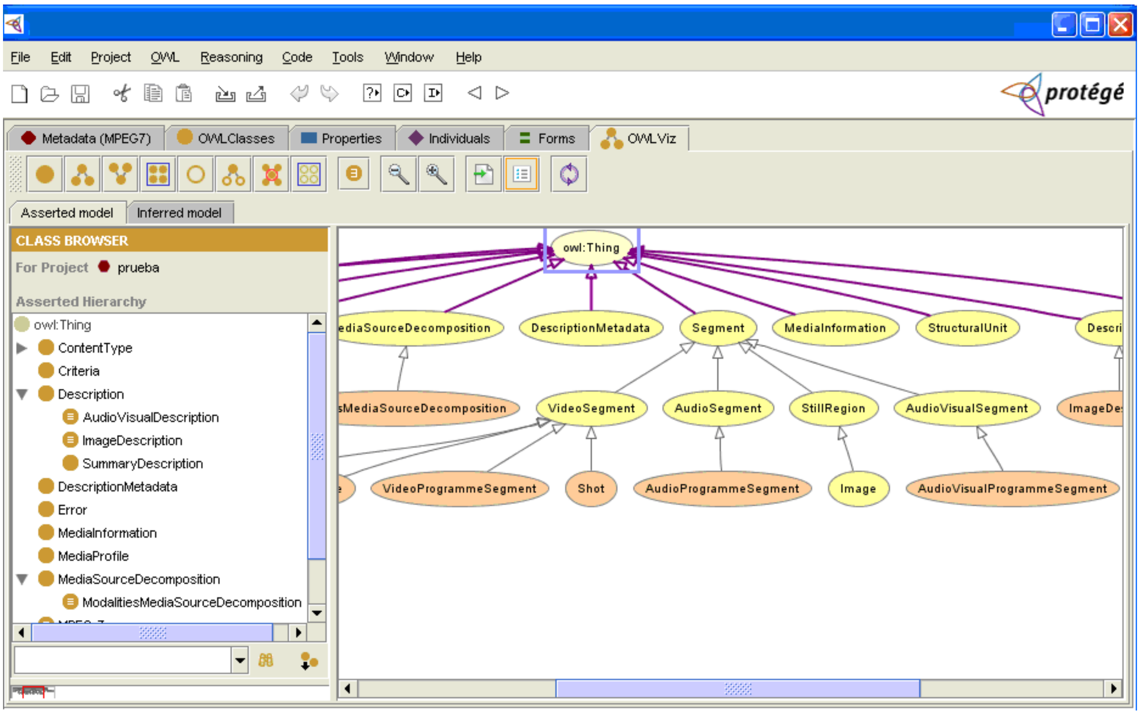


Figure 6.5: Protégé OWL editor

```
<?xml version="1.0"?>
<rdf:RDF
  xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns:owl="http://www.w3.org/2002/07/owl#"
  xmlns="http://www.owl-ontologies.com/Ontology1196180634.owl#"
 xml:base="http://www.owl-ontologies.com/Ontology1196180634.owl">
 <owl:Ontology rdf:about=""/>
 <owl:Class rdf:ID="Face"/>
 <owl:Class rdf:ID="Video_Keyframe"/>
 <owl:ObjectProperty rdf:ID="contains">
  <rdfs:range rdf:resource="#Face"/>
  <rdfs:domain rdf:resource="#Video_Keyframe"/>
 </owl:ObjectProperty>
  <owl:DatatypeProperty rdf:ID="name">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#string"/>
  <rdfs:domain>
   <owl:Class>
    <owl:unionOf rdf:parseType="Collection">
     <owl:Class rdf:about="#Video_Keyframe"/>
     <owl:Class rdf:about="#Face"/>
    </owl:unionOf>
   </owl:Class>
  </rdfs:domain>
 </owl:DatatypeProperty>
  <owl:DatatypeProperty rdf:ID="withAProbability">
  <rdfs:range rdf:resource="http://www.w3.org/2001/XMLSchema#float"/>
  <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#AnnotationProperty"/>
 </owl:DatatypeProperty>
 <Video_Keyframe rdf:ID="keyframe_10392">
  <withAProbability rdf:datatype="http://www.w3.org/2001/XMLSchema#float"
  >0.78</withAProbability>
  <name xml:lang="en">10392</name>
  <contains>
   <Face rdf:ID="face_tonyBlair">
    <name xml:lang="es">Tony Blair</name>
   </Face>
  </contains>
 </Video_Keyframe>
</rdf:RDF>
```

Figure 6.6: Approach to express the fuzziness by employing annotations

items of the broadcaster following the DMS-1 standard. The second ontology was related to the domain covered by the content itself (i.e. news, sports...). And finally, an ontology that covered the descriptive and technical decomposition of the multimedia items.

Descriptive Metadata Scheme (DMS-1) Ontology

The motivation behind this ontology was to implement the descriptive structure of the multimedia assets and the metadata about them. In (Marcos et al., 2009) the reader may find a description of this ontology while in subsection 2.3.10 an overview of the DMS-1 standard is provided.

As has been stated before, the standard groups the annotations according to three logical entities: the production perspective of the asset, the asset as a clip or continuous essence element or element interval and, finally, the different scenes or "actions and events within individual parts of the audiovisual content". In RUSHES SMD, for each one of the scenes, different metadata sets of properties were defined. For each of them, the cardinality and mandatory level were provided.

The sets of metadata and properties were applied to various frameworks (Production, Clip and Scene) and their nature was diverse. For example the standard provided sets of metadata about the titles of the asset, awards, events, information about the device parameters, file, formats and rights among many others.

Regarding the temporal decomposition, the ontology, accordingly to the standard, modeled also the concept "Shot". This entity was related to some of the frameworks and allowed to define specific annotations for a shot of any of the essences of the asset (video, audio, and so on). For each shot, the duration, start position, and description were provided. And if needed, key points or some extra annotations were attached to the shot.

In order to add the specific metadata, the ontology modeled as a concept what the standard names as "Annotation", a property set. This concept was directly linked to different thesauri. This made the model a powerful tool to capture the metadata for the different pieces of the asset.

Figure 4.1 provided a partial view of the DMS fragment implemented by us in RUSHES.

Domain Ontology

In the RUSHES system, the information domain was defined by the raw content repository of the project. This repository was created according to a specific set of scenarios defined by a broadcaster. These scenarios and the content were related to the daily activity of the preparation of the news.

The ontology developed in the project was an extension of a well known ontology about news: the LSCOM Lite ontology proposed by Neo et al. (2006). The extension of the ontology was driven by all the information that could be detected by the analysis modules and by the thesauri that were provided to the journalists in the manual annotation tool.

Low Level Ontology

In order to support the conduction of the information during the analysis process, the MDM modeled the multimedia assets from the video analysis theory perspective. Therefore, the model had to represent the low-level features (e.g. histograms, descriptors, etc.).

Due to the requirements of the RUSHES system, this work only tackled the visual description of the content. In order to achieve this, the MDM incorporated an OWL implementation of the detailed A/V profile (DAVP) proposed in (Bailer and Schallauer, 2006; Bailer et al., 2007).

Ontology Mapping

The mapping between the DMS-1 ontology and the domain model was realized linking the dictionaries proposed by the SMPTE 380:DMS-1 and the concepts of the domain ontology.

The linkage of DAVP ontology with the DMS-1 ontology, was done through the temporal decomposition of the profile, where the two ontologies intersected.

Axioms

The fuzzy reasoner needed (i) a Terminology box, which was required for representing the knowledge and it defined the concepts, the roles, and the individuals and (ii) an Assertional Formalism (ABox) that had the assertions and the declaration of axioms. While the TBOX and the ABOX were generated on real time using the information gathered in the previously described ontologies, the axioms had to be predefined.

In the following, we include a fragment of the axioms employed for the identification of a keyframe of a demonstration. The axioms assumed that a keyframe with more than ten faces, in a non-vegetal environment and with a presence of some text in the bottom or middle-bottom of the image, could be tagged with a certain probability as the head of a demonstration. The probabilities of the assertions (ABOX) were handled by the fuzzy reasoner in order to generate new individuals according the defined axioms.

```
(implies demonstrationhead demonstration)
(implies demonstrationhead (and  crowdinstreet
(some has-textposition (or bottom midd-botton))))
(equivalent morethantenfaces (and keyframe (at_least
10 has_faces)
(equivalent street (and non-vegetation outdoor))
(equivalent crowdinstreet (and morethantenfaces street ))
```

6.4.2 MDM IE

The MDM IE was composed by a set of PEs that were mainly devoted to the management (reading, writing, validation, inference...) of the ontologies stored in the MDM KB and the fuzzy reasoning over the annotations stored during the analysis process. In the following paragraphs we describe the main PEs implemented in the middleware.

Ontology Handler

An instance of this PE was generated for each video that was analyzed. Therfore, when several videos were analyzed concurrently, there were various instances of the MDM running. When the video arrived in the system, this PE was invoked and an empty model was loaded in memory. This model was populated with the information coming from the different analysis steps. The usage of this PE warrantied that the annotations generated for one video were not merged with the rest of the videos being analyzed.

Concept Relator

This PE was implemented with the Jena API (McBride, 2002). This API was extended since, the methods implemented by this engine were mainly related to search and navigation of the concepts and instances of the model. The extension performed was due to the need to reduce the amount of time required by the API to browse and search the concepts of the model (T-Box), when the search criteria complexity increased. For instance, following a similar approach to the WIDE SMD, inference methods were implemented to enhance the Jena API in order to retrieve all the “intermediate”concepts and their subclasses that link the concepts A and B.

Semantic Validator

The semantic validator employed a Pellet Reasoner (Sirin et al., 2007) to check the consistency of the ontology each time the model was enriched with new metadata. The coherence of the introduced annotation was checked. Whenever any unconsistency was detected, the annotations were removed and therefore discarded.

Temporal Units Conversor

This PE was in charge of the homogenization of the temporal units the annotations were referred to. Some annotations were related to a period of time, to an interval of frames or to a unique frame. This PE adapted any temporal reference arriving into the SMD into the Time Code promoted by the SMPTE (2008). The temporal references leaving the SMD were provided according to the needs of the destiny module.

Data Schemas Mapper

This PE was in charge of the management of many diverse inputs received by the MDM. This covered many different types of XML and text files. The existence of this PE was coherent with the philosophy behind our middleware. All the analysis modules the SMD interacted with, provided the information according to their own nature. Thus, different algorithms from different software providers were employed without requiring any external adaptation. This was specially useful, since most of the analysis modules available from the video analysis community provided very simple and diverse output formats.

XML Transformer

This PE was in charge of performing XML transformations. It was mainly employed to transform the MEX file that were indexed by the search engines into other XML formats required by other components such as the annotation tool or the result visualization interface.

The PE was implemented in Java. It was able to apply either transformation stylesheets or to create object oriented structures from XML Schemas and then use those structures to perform the transformation.

Fuzziness Handler

This PE was in charge of the preparation of the information over which the reasoning had to be performed. First of all, and depending on the input parameters provided by the software entity invoking this PE, it employed other PEs to extract from the ontology a subgraph with all the classes and individuals involved in the reasoning.

Then, it went through this set of concepts and individuals and the rules stored in the SMD KB and created the list of concepts, individuals, probabilities and axioms that composed the A-Box and the T-Box required by the fuzzy engine. As has been explained in subsection 6.4.1, the fuzziness information was extracted from the annotations.

Fire Fuzzy Reasoner

This PE was responsible for the invocation of the employed reasoner and the management of the communication. The reasoner employed is the FIRE fuzzy reasoner (Simou and Kollias, 2007; Stoilos et al., 2007) provided by NTUA Image Group.

6.4.3 MDM GW

The MDM GW was composed by a set of web-services that exposed the services mentioned in section 6.2. The MDM GW was implemented as a standalone windows server and was able to attend parallel invocations from the different modules of the system. The server was stateless for all the services but the ones related to the analysis of the ingested videos. During that process, the server had to preserve the information of the different analysis operations performed on each video. In order to do that, an instance of part of the MDM GW was created to attend the requests related to each video.

In Figure 6.7, a partial view of this set of interfaces is shown. For example, the mdm2ccr represents the services provided by the middleware to the CCR module, in order to ensure the knowledge conduction during the video analysis process. In that view the functionalities are grouped according to the unit of information they were related to: the whole asset, a video segment, a cluster, a keyframe and so on. The methods available through these classes were used by the CCR module to store the information generated by the multimedia analysis algorithms in the MDM KB.

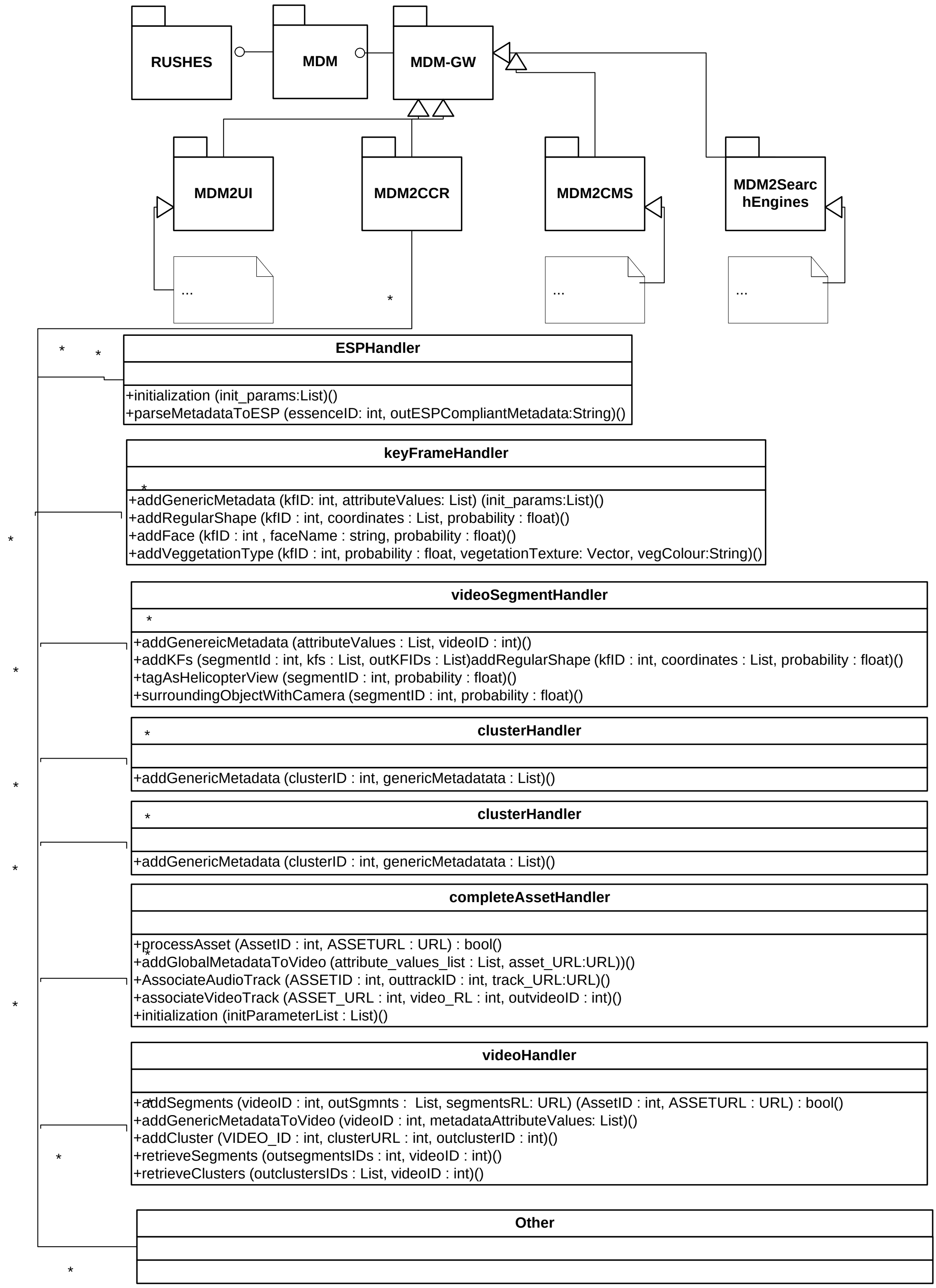


Figure 6.7: Partial view of the MDM GW interfaces.

6.5 RUSHES SMD Evaluation

The feasibility and convenience of the deployment of the MDM RUSHES was validated and evaluated as follows. Concerning the feasibility, due to the provision of the services using standardized and well established multi-platform techniques, the MDM was successfully and seamlessly integrated into the RUSHES system. This integration required the interaction with other components such as the FAST commercial ESP search engine, a commercial database, the system interfaces (search and annotation), and the analysis operators, which were developed with different programming languages and executed on different operating systems.

In the following items we summarize some aspects related to the performance of the integrated MDM for a testbed of 70 unedited videos with a total duration of more than 18 hours (the results do not include the time needed for the communication with the rest of the modules of the system):

- During the analysis process the MDM was invoked an average of 1.3 times per analysis operator. However, the waiting time due to the MDM operation for the whole analysis process was, in average, less than the 0.6% of the whole analysis period.
- The average time used by the MDM to generate a MEX file according to the format required by the ESP engine was 4 seconds.
- The generation of new information based on the fuzzy reasoning required as average 0.5 seconds for each analyzed video shot.
- The provision of a list of recommendations for the annotations required less than 0.34 seconds.

Regarding suitability, the evaluation of the MDM was mainly carried out through a discussion session with the Broadcast Engineering Department of ETB (Basque Public Broadcaster). The main conclusions are outlined in the following items:

- They corroborated that the middleware on the one hand, allowed a seamless integration with their existing MAM system and on the other hand, it provided the functionality of managing semantic aspects.
- The fact that the MDM did not replace any of the currently available professional solutions for information storage but complemented them was positively considered.
- The channeling of knowledge during the analysis process was understood as a needed functionality according to the progress expected in the

Multimedia Analysis processing techniques. However, regarding the MDM some additional features were expected. The service implemented in the MDM gathered the information coming from the different analysis algorithms, and once all the operators had finished, the information was made available for the rest of the components of the system. However, in many cases (e.g. news), journalists may require the content before this process ends. It would be very useful to be able to provide partial results of the analysis (e.g. shot boundary information), even if the whole process is not yet finished.

- The results provided by the fuzzy reasoner did not allow to extract any conclusion. This was mainly due to the fact that most of the analysis operators were not able to provide the information about the probability of success. Therefore, the transformation of this small amount of information into new interesting annotations wihtin the dataset was not a straightforward task. In Chapter 7 we tackle a new approach to make use of the benefits of a semantic fuzzy reasoner to generate new metadata.
- Their lack of previous expertise about the semantic technologies employed within MDM IE layer was considered as an startup barrier. However, it was also understood that the intermediate layer (MDM IE) was to be upgraded only on a long term basis.

The results of the global system evaluation can be found in (Sarris, 2009).

Part IV

CONCLUSIONS AND FUTURE WORK

7 Conclusions and future work

This chapter is mainly devoted to the summarization of the contributions of this work and the description of its future action lines. The chapter concludes with a list of our publications in the field.

7.1 Summary of Conclusions

In the previous chapters we described the current status of the multimedia information retrieval, highlighting the drawbacks and unbridged gaps. In such context, we advocated for the noteworthiness of content-based functionalities on the top of the existing multimedia search and retrieval systems, from a pragmatic and integration oriented point of view. Thus, we have provided an overview of the incoming and emerging content based techniques and we have contextualized them within the current trends and challenges in the multimedia information retrieval field.

With the aim of promoting this goal, we **presented a reference model for the retrieval process**, which distinguishes three lines, and, where the provision of what we call semantic services is outsourced to a sole module, the semantic middleware. We **proposed a generic architecture** to implement that middleware. The definition of this architecture has been driven by the aim of integrating those content-based functionalities in the existing multimedia information technology. Therefore, the architecture avoids the replacement of the well-established technologies for the massive storage and efficient recovery algorithms, which constitute the core of most of the multimedia information and management systems. Beside this, the existence of this middleware, reinforces the semantic interoperability of the systems and avoids the problem of semantic redundancy. This problem is mainly due to the gradual integration of modules or subsystems that require certain semantic information in order to perform their task. The lack of a unique semantic middleware may lead to a system with semantic information replicated, in a coherent or incoherent way, in different locations of the system and under different administrators.

Beside this definition, based on the experience acquired from applying the architecture, we **included a set of key issues for the correct deployment of each of the layers that compose the middleware**. These set of key issues have been **mapped to the real detail implementations of two fully distinct and complementary validation scenarios**. The combination of both scenarios covered a wide range of processes involved in the storage and retrieval of multimedia assets including the content-based potential.

According to this, we consider that **this work supports the development of a new generation of multimedia retrieval systems**. This support is driven by a fully pragmatic point of view, reinforced by the following assertions:

- According to the review of the trend in retrieval systems, the SMD avoids, in our opinion, the incoming threaten of the semantic redundancy. This is achieved by assembling the modeling of the domain in a sole component.

- The middleware enables the incorporation of the emerging content-based features throughout the whole retrieval process (*Storage*, *Search*, and *Browsing* lines).

- It facilitates the gradual and scalable incorporation of content-based features in current system, without imposing the replacement of current retrieval techniques. This favours a seamless integration with the existing technologies.

- It provides a single internal semantic representation that facilitates semantic interoperability with external entities.

- The definition of a set of key design parameters and the validation of the middleware by its deployment in two different and complementary domains facilitates the development of new deployments over diverse domains. Based on our experience we highlight the following domains of interest: professional content producers, document management systems in ERPs (Enterprise Resource Planning), cultural digital libraries and public administrations.

As a result of this, in our opinion, this work is a valuable contribution to support the managers of complex multimedia information management and retrieval systems in the task we have faced: A gradual and seamless enhancement of those systems in order to implement content-based functionalities.

7.2 Future work

This section is devoted to the future action lines that will follow the work presented in this PhD report. In the following we provide a summary of those action lines. Then we include a description of the preliminary results of those lines that have been already started.

7.2.1 Architecture for semi-automatic multimedia analysis

As to future work, we aim to explore the usage of the semantic middleware to automatically generate annotations of the multimedia assets. We started this work in the RUSHES scenario, by providing the knowledge conduction during the information extraction and by applying fuzzy reasoning to the information provided by those modules in order to extract new information.

In the future, we intend to improve this annotation generation process by increasing the scope of the usage of the semantic middleware. We want to increase the cooperation between a middleware deployment and analysis modules, establishing an iterative communication. The main idea is to employ the knowledge gathered by the semantic middleware deployment in a similar way to the functionality implemented in the RUSHES system, to guide the analysis process. In Figure 7.1 we provide an sketch with the global view of the proposed architecture.

The establishment of the iterative communication mentioned, has been initially implemented through a working plan that is composed by two main action lines.

The first action line is related to the establishment of **techniques for the dynamic management of the analysis workflow** based on the knowledge gathered in the semantic middleware. According to this, the middleware will support the decisions taken during the analysis process. The procedure is based on a set of rules that are able to handle the fuzziness of the annotations provided by the analysis modules and gathered in the semantic middleware. The current developments combine the following elements:

- *Drools* is a business rule management system (BRMS) written in Java. The main reason to use it is that it includes a chaining inference based rules engine that supports the JSR-94 standard for the construction, maintenance, and enforcement of business or logic policies. This allow us to define the logic of the analysis process by establishing a set of nodes (states) and rules. The definitive flow is determined on real time by the information coming from java applications that are invoked from Drools. In our case this information comes from the semantic middleware, which is described in the following.

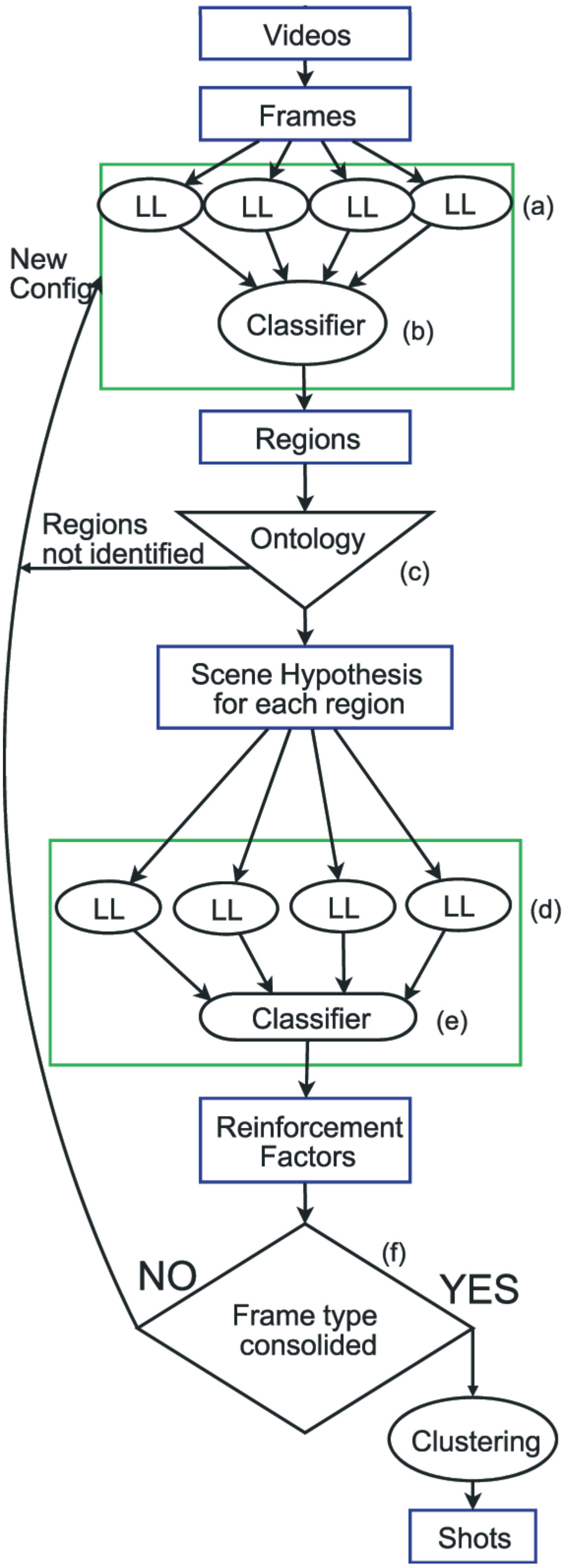


Figure 7.1: Proposed Architecture for semi-automatic multimedia analysis by hypothesis reinforcement

Figure 7.2: Preliminary results of the classification process

- And a deployment of a *Semantic Middleware* fully compliant with the work presented here. In order to provide the information required for the determination of the work flow (i.e. hypothesis reinforcement) it includes an ontology to store the information provided by the analysis modules and a reasoner in order to infer new information.

The main objective targeted by the mentioned dynamic management is therefore the definition of a framework and a methodology for its seamless application to different domains. This is due to the fact that current content analysis techniques are, in most of the cases, highly dependent of the domain the content is related to. Therefore, the management of the workflow must be dynamic not only in the sense of changing during the analysis process, but also in the sense of being capable to be employed in different domains with low adaptation costs.

The second action is related to the information shared by the middleware and the analysis modules. Till now in the described scenario of RUSHES we have successfully implemented a communication between the modules and the middleware, where the middleware acts as a passive actor. Its main role is the gathering and provision of the information received and requested.

At the reporting period we are working to reinforce the role of the middleware in that communication. We believe that in some cases, the middleware should be employed for the management of the analysis performed by the modules. For example, a module may focus on the definition and identification of regions and their predominant colours based on the output of a classifier. According to the output of that module, the classifier may be re-invoked in order to perform a new classification based on a different set of parameters.

Some preliminary experiments of the analysis process are visually represented in Figure 7.2 and deeply explained in Olaizola et al. (2009).

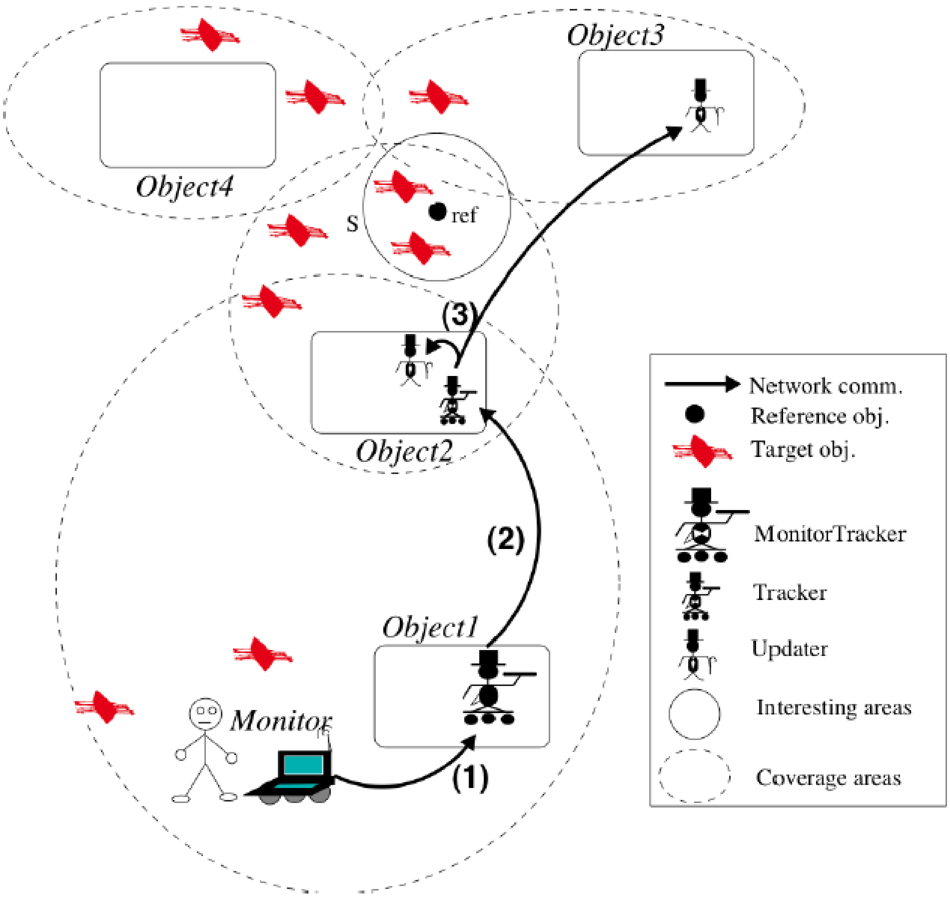


Figure 7.3: Architecture for a location aware system for monitoring sports events

7.2.2 Content-based retrieval functionalities in broadcast production

We have started a new research line based on the approach presented here for the inclusion of the content-based features. The main objective is to migrate this approach to the semantic-based handling of multiple cameras that are employed for the production and broadcasting of an event. The motivation is the simplification of the process that the technical directors do in order to select the most appropriate camera to shot a specific scene or happening. The combination of the camera parameters and the information of the location allow the deployment of advanced retrieval algorithms that semi-automatizes the selection of the appropriate camera and view.

In Figure 7.3 we include a graphical representation of the architecture proposed for a specific scenario (i.e. a rowing event), where technical directors may indicate interesting objects or geographic areas in run-time and the system is in charge of selecting the best shots from the cameras in the scenario.

7.3 Summary of publications

This section summarizes our publications, distinguishing between those that are directly related to our contributions, those that are related to the preliminary results of the future work and other publications in the field.

7.3.1 Publication related to the contributions of this thesis

In the following subsections we summarize the publications related to the contributions of this thesis work.

Semantic middleware definition and multimedia reference model

The proposed reference model for Multimedia Information Retrieval, the definition of the middleware and the design criteria is published in:

- (Marcos et al., 2011)
  Article: Marcos, G.; Illarramendi, A.; Olaizola, I. G. & Flórez, J. A Middleware to enhance current Multimedia Retrieval Systems with Content-based functionalities (to be published) Multimedia Systems, 2011

Although this article also covers the design criteria of both deployments, it is only included once.

Validation of the Semantic Middleware in the WIDE Use Case

The following publications are related to the description of the semantic middleware implemented in the WIDE use case described in Chapter 5.

- (Marcos et al., 2007)
  Article: Marcos, G.; Smithers, T.; Jiménez, I. & Toro, C. Meta Level: Enabler for semantic Steered Multimedia Retrieval in an Industrial Design Domain Systems Science, 2007, 2, 15-22

- (Marcos et al., 2005b)
  Techreport: Marcos, G.; Jiménez, I. & Smithers, T. D20 Technical Report of the Meta Level (WIDE_IST_2001_34417), 2005

Validation of the Semantic Middleware in the RUSHES Use Case

The publications describing the implementation of the semantic middleware in the RUSHES use case, described in 6, are the following:

- (Marcos et al., 2008)
  Inproceedings: Marcos, G.; Krämer, P.; Illarramendi, A.; Olaizola, I. G. & Flórez, J. Semantic Middleware to Enhance Multimedia Retrieval in a Broadcaster Semantic Multimedia, Third International Conference on Semantic and Digital Media Technologies, SAMT 2008 Koblenz, Germany, December 2008, Proceedings, Springer-Verlag Berlin Heidelberg, 2008, 74-88

- (Marcos, 2007)
  Techreport: Marcos, G. D8 RUSHES Metadata model specification Vicomtech, 2007

Implementation of the first DMS-1 OWL ontology

The following paper deals with the implementation details of the SMD KB of RUSHES which is driven by the implementation in OWL of the DMS-1 OWL ontology.

- (Marcos et al., 2009)
  Inproceedings: Marcos, G.; Alonso, K.; Illarramendi, A.; Olaizola, I. G. & Flórez, J. DMS-1 driven Data Model to enable a Semantic Middleware for Multimedia Information Retrieval in a Broadcaster Proceedings of the 4th International Workshop on Semantic Media Adaptation and Personalization., 2009

Contribution to the JPSearch standardization activity

In the following we provide the details of the contribution to the standardization activity of the JPSearch described in 4.2.

- (Marcos and Jiménez, 2007)
  Techreport: Marcos, G. & Jiménez, I. Contribution to JPSEARCH Standardization activity: Mapping on JPSearch architecture of the WIDE Image Search and Retrieval System. Doc Number: wg1n4394 Vicomtech, 2007.

7.3.2 Publications of the future work

The following is a publication that describes the approach for the extension of the semantic middleware in order to enhance its role during the media analysis:

- (Olaizola et al., 2009)
  Inproceedings: Olaizola, I. G.; Marcos, G.; Krämer, P.; Flórez, J. & Sierra, B. Architecture for semi-automatic multimedia analysis by hypothesis reinforcement IEEE International Symposium on Broadband Multimedia Systems and Broadcasting, 2009

This work is mainly addressed by the researcher Igor G. Olaizola.

And regarding the second action line for the future work, the following describes the first results published:

- (Ilarri et al., 2010)
  Inproceedings: Ilarri, S.; Mena, E.; Illarramendi, A. & Marcos, G. A Location-Aware System for Monitoring Sport Events Eight International Conference on Advances in Mobile Computing & Multimedia (MoMM 2010), Paris (France), ACM Press, ISBN 978-1-4503-0440-5, Austrian Computer Society (OCG), ISBN 978-3-85403-273-1, 2010, 305-312

7.3.3 Other publications in the field

In this section we cover other publications that either (i) are related to the systems implemented in the WIDE use case (not only the semantic middleware) or (ii) are related to the technologies employed for the implementation or (iii) are preliminary steps of the future work estimated for the work.

WIDE system

In the following we list the publications related not only with the WIDE SMD but with the whole WIDE system:

- (Marcos et al., 2005d)
  Inproceedings: Marcos, G.; Smithers, T.; Jiménez, I.; Posada, J.; Stork, A.; Pianciamore, M.; Castro, R.; Marca, S. D.; Mauri, M.; Selvini, P.; Sevilmis, N.; Thelen, B. & Zecchino, V. A Semantic Web based Approach to Multimedia Retrieval Fourth International Workshop on Content-Based Multimedia Indexing (CBMI05), 2005

- (Sevilmis et al., 2005)
  Inproceedings: Sevilmis, N.; Stork, A.; Smithers, T.; Posada, J.; Pianciamore, M.; Castro, R.; Jiménez, I.; Marcos, G.; Mauri, M.; Selvini, P.; Thelen, B. & Zecchino, V. Knowledge Sharing by Information Retrieval in the Semantic Web. ESWC, 2005, 471-485

- (Stork et al., 2005)
  Inproceedings: Stork, A.; Sevilmis, N.; Smithers, T.; Posada, J.; Pianciamore, M.; Castro, R.; Jiménez, I.; Marcos, G.; Mauri, M.; Selvini, P. & Thelen, B. A Semantic Web Approach to CE 11th International Conference on Concurrent Enterprising: Integrated Engineering of Products, Services and Organisations, 2005, 193-200

- (Thelen et al., 2005)
  Inproceedings: Thelen, B.; Sevilmis, N.; Stork, A.; Castro, R.; Jiménez, I.; Marcos, G.; Posada, J.; Smithers, T.; Mauri, M.; Pianciamore, M.; Selvini, P. & Zecchino, V. Information Management on the Basis of Semantic-Web Techniques, or A Google for Developers VDI-Verlag - Gesellschaft Fahrzeug- und Verkehrstechnik: Erprobung und Simulation in der Fahrzeugentwicklung: Mess- und Versuchstechnik, 2005, 167-180

- (Smithers et al., 2004)
  Inproceedings:Smithers, T.; Posada, J.; Stork, A.; Pianciamore, M.; Ferreira, N.; Grimm, S.; Jiménez, I.; Marca, S. D.; Marcos, G.; Mauri, M.; Selvini, P.; Sevilmis, N.; Thelen, B. & Zecchino, V. Information Management and Knowledge Sharing in WIDE. EWIMT, 2004

Technologies involved in implementation of the semantic middleware deployments

In the following list we include those publications related to the technologies employed for the deployments of the semantic middleware in the use cases.

- (Marcos et al., 2005c)
  Inproceedings: Marcos, G.; Lamsfus, C.; Eskudero, H. & Linaza, M. T. Semantic based Querying and Retrieving Information for Artistic Expressions: The Art-e-Fact Ontology Fourth International Workshop on Content-Based Multimedia Indexing (CBMI05), 2005

- (Marcos et al., 2005a)
  Inproceedings: Marcos, G.; Eskudero, H.; Lamsfus, C. & Linaza, M. T. Data Retrieval from a Cultural Knowledge Database Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS), 2005

- (Lamsfus et al., 2004)
  Inproceedings: Lamsfus, C.; Karagiannis, G.; Sotiropoulou, S.; Eskudero, H.; Marcos, G.; Linaza, M. T. & Daniilia, S. The art-E-fact Ontology: a Possible Contribution to CIDOC CRM 10th joined meeting between the CIDOC CRM Special Interest Group and the ISO/TC46/SC4/WG9, 2004, 1-5

Part V

ANNEXES

OWL-Rep structure

In the following we describe the structure of the OWL-Rep database mentioned in the subsection 5.4.1. The figure included in this annex provides a graphical summary of this structure.

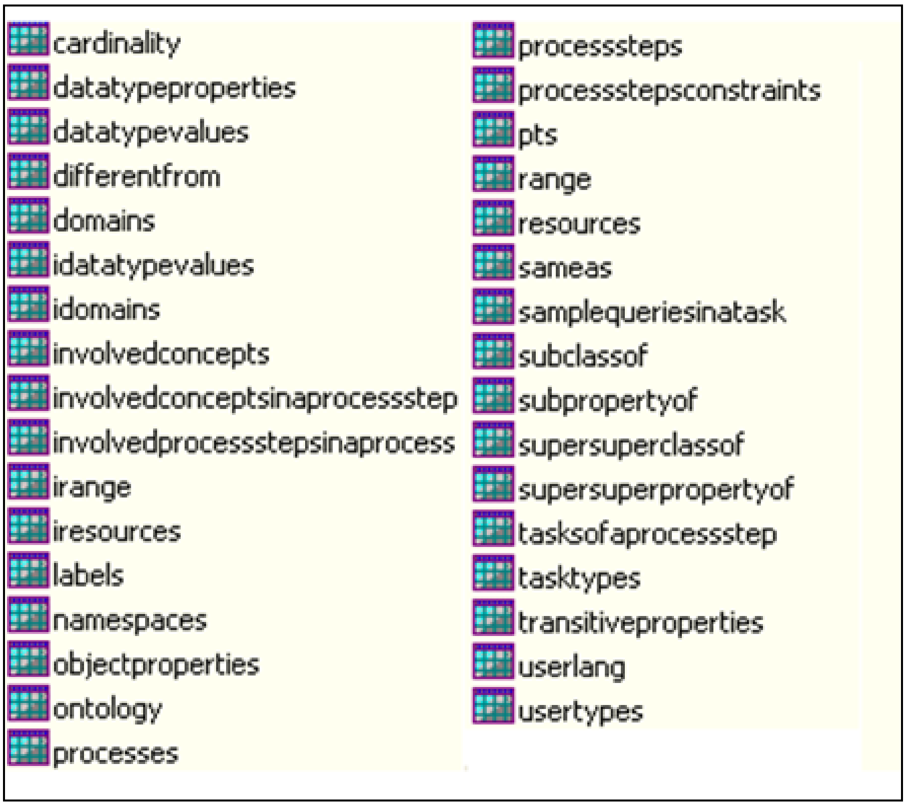
Table resources This table was fed with all the resources of the Ontology. Under the definition of “resource” the classes the objectProperties and the dataTypeProperties were compiled. This table was the core of the repository since it indexed the primary key of every term in the Ontology.

ObjectProperties This table compiled all the information needed in order to interpret an objectProperty properly. It included information about its cardinality, its symmetry, about if it was transitive, inversefunctional, and so on. This table also had control over all the “modified” objectProperties, since each time the KW made a modification of the cardinality of the property, or whatever, it compiled the new information, without loosing the link with the original property.

DataTypeProperties The structure was really similar to the previous one. The objective also was the same, but the target entity was the set of DataTypeProperties.

Cardinality: This table compiled the different cardinalities that could be found in the Ontology. A cardinality covered the “atleast”, “atmost”, “isrequired” and “ismultiple” values. Thus, once a property was related to a specific cardinality, its maximum and minimum value , its mandatory or non mandatory nature were specified.

Domain: This table was used to link the properties (both DataTypeProperties and ObjectProperties) with their domain.



IDomains The role of this table was the same as the previous one. However its size is bigger since it also took into account the properties that a class inherits of its superclasses.

Range: This table stored the information about the ranges of the ObjectProperties.

Iranges: As iDomains, this table took into account the ranges of the properties inherited by a class.

TransitiveProperties: This table handles the transitivity of the properties that linked the classes. Due to the fact that it was generated in advance, its existence increased the performance of the inference processes carried out during the search process. The transitivity of each property was treated taking into account the properties hierarchy.

According to this, if we had: “CAR” ==> (has_powertrain) “POWERTRAIN” and “POWERTRAIN” ==> (has _engine) “ENGINE”, if both properties (“has_powertrain” and “has_engine”) were subproperties of a common transitive property (“has”), the ML understood that “CAR” has “ENGINE” and that there was one concept (“POWERTRAIN”) in between. This kind of inference was the one used to fulfill the reporting table.

SubClassOf This table allowed the MetaLevel to browse the Ontology hierarchy, since it related each class with its superclasses.

SubPropertyOf This table pursued the same objective than the previous one, but it collected information about the hierarchy of the properties.

Ontology This table contained general information about the ontology.

Namespaces This table allowed the ML to distinguish among the concepts based on their namespace.

Labels This table collected the labels that stored the synonyms that corresponded to each user profile terminology.

SuperSuperClass This table stored for each class the top level/s classes of the branches of the graph it belonged to. This was used in order to have at every moment full control of to which ontology or subontology a concept was related to.

SuperSuperProperty Thanks to the use of this table, the ML was able to know which is/are the “older ancestor/S” of a property. This piece of information was really useful in order to treat the inheritance in a proper way.

Sameas Other of the OWL contributions was the possibility to link classes that the KE may consider were equivalent. This table, that stored those “similarity relations”, allowed the ML to infer those equivalences. The equivalence was easy and intuitively edited with the Protégé.

DifferentFrom This table had a parallel objective with regard to the previous one, but focusing on the “different” feature of the classes.

DataTypeProperties, DataTypePropertiesValue, iDataTypeProperties These tables stored the information of the different datatype properties defined in the ML Ontology and their value (if any). They have information about the type of those properties: string, float, int, etc.

TaskTypes: This table was fulfilled with the information got after an inference process that extracted the task types defined in the ML Ontology.

InvolvedConcepts: This table related each tasktype with the main concepts involved in that task.

UserTypes: This table contained the different user profiles of the users of the WIDE System.

UserLang: This table specified for each type of user, the appropriate language. This allowed the implementation of the different dictionary and terminology mapping facilities.

Pts: The main role of this table was the storage of the different Content Types that exist in the ontology.

Processes, ProcessesSteps, Processstepsinaprocess and processsteps constraints These tables had information about the different processes and processteps, specifying which process steps belong to which process and defining flow constraints among those process steps.

Involvedconceptsinaproesstep: This table related each process step with the main concepts involved in that process step.

Taskofaprocesstep: This table specified the different tasks that had to be performed to carry out each process step.

Samplequeries: The current table stored the sample queries that the KE had modeled in the ontology for each task.

BNF Grammar Notation

In the following we provide a partial view of the BNF grammar employed in the WIDE SMD KB.

BNF GRAMMAR - Non Terminal values

```
erInput>          ::= <content_type> <qualifying_list> #contentSet_qualifyingSet
                     | <qualifying_list> #qualifyingSet

ntent_type>       ::= <request_term> <content_type_connector> #request_set
                     | <content_type_term> <content_type_list> #content_set

ntent_type_list>  ::= <binlogop> <content_type_term> <content_type_list>
                     | <content_type_connector>

nlogop>           ::= <or_binlogop>
                     | <and_binlogop>

nnector>          ::= <content_type_connector>
                     | <qualifying_connector>

alifying_list>    ::= <qualifying> [ <connector> <qualifying_list> ]

alifying>         ::= <lparenthesis> <qualifying_list> <binlogop_qualifying_list> <rparenthesis>
                       #binlog_qualifying
                     | <attribute> <operator> <value> <unit> #attribute_op_value_unit
                     | <attribute> <operator> <value> #attribute_op_value
                     | <value> <attribute> #value_attribute
                     | <keyword>

nlogop_qualifying_list>  ::= <or_binlogop> <or_qualifying_list>
                           | <and_binlogop> <and_qualifying_list>

_qualifying_list>   ::= <qualifying_list> [ <or_binlogop> <or_qualifying_list> ]

d_qualifying_list> ::= <qualifying_list> [ <and_binlogop> <and_qualifying_list> ]

ntent_type_term> ::= <alphanumeric_term>

ribute>           ::= <alphanumeric_term>

ue>               ::= <asterisk>
                     | <numeric_term>
                     | <alphanumeric_term>
                     | <wildcard_term>

t>                ::= <alphanumeric_term>

yword>            ::= <alphanumeric_term>
```

<u>**BNF GRAMMAR - Terminal values**</u>

```
quest_term>          ::= "ALL" | "INFORMATION" | "WHAT"

ntent_type_connector>    ::= "ABOUT" | "FOR" | "OF" | "WITH" | "WITHIN" #connector

alifying_connector>        ::= "ABOUT" | "AS" | "FOR" | "OF" | "WITH" | "WITHIN"
                              | "LIKE" | "ON" | "ONTO" | "OVER" #connector

erator>                ::= "=" | "EQUALS"
                          | "!=" | "NOT_EQUALS"
                          | "<" | "LESS_THAN" | "MINOR_THAN"
                          | "<=" | "LESS_OR_EQUAL_THAN" | "MINOR_OR_EQUAL_THAN"
                          | ">" | "MORE_THAN" | "GREATER_THAN" | "MAJOR_THAN"
                          | ">=" | "MORE_OR_EQUAL_THAN" | "GREATER_OR_EQUAL_THAN"
                          | "MAJOR_OR_EQUAL_THAN"

_binlogop>             ::= "OR" #binlog_operator

d_binlogop>            ::= "AND" #binlog_operator

renthesis>             ::= "(" #lpar

arenthesis>            ::= ")" #rpar

terisk>                ::= "*" | "ANY"

meric_term>            ::= ["0"-"9","+","-"] ( ["0"-"9","+","-","/",".",",",":"] )*

hanumeric_term>  ::= ["A"-"Z","0"-"9","+","-"] ( ["A"-"Z","0"-"9","_","+","-","/","."] )*

dcard_term>          ::= ( ["A"-"Z","0"-"9","+","-","?","*"] | "<ANY>" )
                          ( ["A"-"Z","0"-"9","_","+","-","/",".",",",":","?","*"] | "<ANY>" )*
```

Process Support File

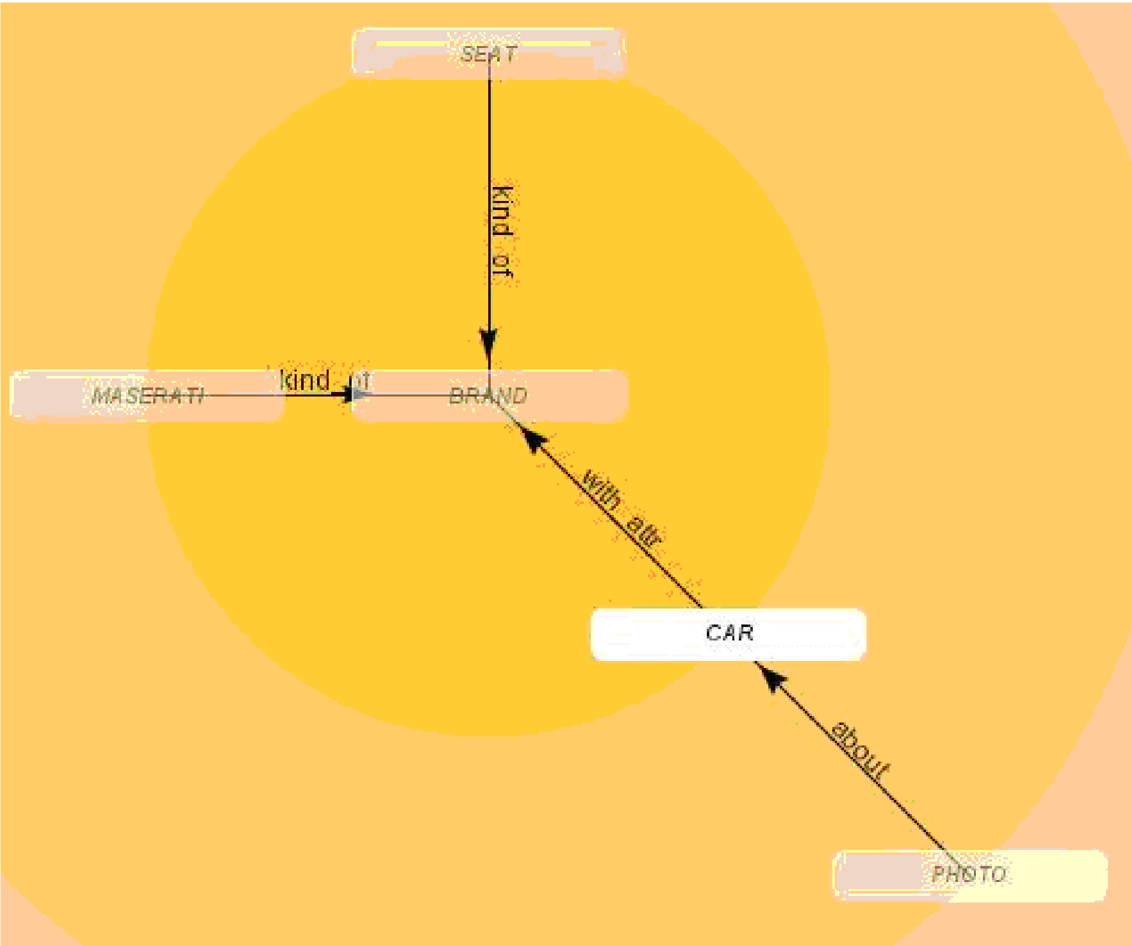
```
<processData>
    <processes>
        <process name="ENGINEERING_PROCESS"/>
        <process name="STYLING_PROCESS"/>
        <process name="DESIGN_PROCESS"/>
        <process name="FRONT_BUMPER_DESIGN_PROCESS">
                <process_step name="FRONT_BUMPER_STYLE_MODEL_VALIDATION">
                        <conditions/>
                </process_step>
                <process_step name="FRONT_BUMPER_STYLE_DEFINITION">
                        <conditions/>
                        <task name="INTERNAL_STANDARDS_TASK"/>
                        <task name="STYLING_PACKAGING_TASK"/>
                        <task name="INTERNAL_PHOTO_DATABASES_TASK"/>
                        <task name="INTERNAL_STANDARDS_TASK">
                                <query_sample query="'legal requirements'"/>
                        </task>
                        <task name="STYLING_PACKAGING_TASK">
                                <query_sample query="'external dimensions'"/>
                        </task>
                        <task name="INTERNAL_PHOTO_DATABASES_TASK">
                                <query_sample query="aggressive animals"/>
                                <query_sample query="logos of brands of cars"/>
                                <query_sample query="pictures of 'front view' cars"/>
                        </task>
                </process_step>
                <process_step name="FRONT_BUMPER_FEASIBILITY_ANALYSIS">
                        <conditions/>
                </process_step>
        </process>
    </processes>
    <querySamples>
        <task name="task1">
                <query_sample query="picture about car"/>
                <query_sample query="document about emission_standard"/>
        </task>
    </querySamples>
</processData>
```

Graph Format

This annex includes the internal representation of the graph shown in the figure shown in the following.

As the nomenclature is based on XML, it is quite intuitive. The main concept entity is used in order to inform the UI which is the most important node in the graph.

```
<graph>
  <nodes>
     <node id="4" name="PICTURE"/>
     <node id="3" name="BRAND"/>
     <node id="1" name="MASERATI"/>
     <node id="2" name="CAR"/>
  </nodes>
  <relations>
     <relation src="2" dst="3" label="with_attr"/>
     <relation src="1" dst="3" label="kind_of"/>
     <relation src="4" dst="2" label="about"/>
  </relations>
  <mainnode id="2"/>
</graph>
```


SEAT
kind_of
MASERATI
kind_of
BRAND
with_attr
CAR
about
PHOTO

Result Format

This annex includes a fragment of a return (result retrieved by the Agency from the ISs) .

In the following we include some highlights about the interpretation of this return:

- In the first section of the RDF fragment we can see a result retrieved by the Agency. The interpretation of the information source concerning this concept can be seen in the RDF:datatype. This result is composed of a set of bindings. Some of those bindings represent the metadata related to that result in the IS whereas some metadata specify some useful information concerning the way that result was retrieved. For instance the ResultFor binding gives information about which was the concept asked in the query that implied the retrieval of this concept. The binding named “Information Source” is added by the Agency in order to specify which IS produced this result.

- The second piece of fragment represents some context information for the term. This is only retrieved from some ISs, and aims to represent the concepts that, according to that ISs, are semantically close to the concept the result belongs to.

- Finally, the third piece of fragment represents the relations among the different results, according to the information gathered from each information source.


```
<rdf:RDF
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rs="http://jena.hpl.hp.com/2003/03/result-set#">
<rs:ResultSet rdf:about="http://ist-wide.info#resultset">
<rs:size>3</rs:size>
<rs:solution rdf:datatype="WIDE_known:MotorVehicle">
<rs:ResultSolution>
<rs:binding>
```

```
<variable rdf:datatype="WIDE_known#meta">InformationSource</variable>
<value>CL</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_known#meta">ReturnFor</variable>
<value>CAR</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_known#meta">ResourceID</variable>
<value>Provider_Ontology_Final_00162</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_unknown">family</variable>
<value>Car</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_known">MODEL</variable>
<value>Bora</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_unknown">project</variable>
<value>Production_Car</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_known">SEGMENT</variable>
<value>Coupe</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_known">YEAR</variable>
<value>1971</value>
</rs:binding>
</rs:ResultSolution>
</rs:solution>
<rs:solution rdf:datatype="WIDE_known#meta:context_info">
<rs:ResultSolution>
<rs:binding>
<variable rdf:datatype="WIDE_known#meta">baseConcept</variable>
<value>MotorVehicle</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_known#meta">Neighbour</variable>
<value>ManufacturedAggregat</value>
</rs:binding>
```

```
<rs:binding>
<variable rdf:datatype="WIDE_known#meta">Neighbour</variable>
<value>ManufacturedAggregate</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_known#meta">Neighbour</variable>
<value>VehicleCategory</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_known#meta">Neighbour</variable>
<value>EmissionCategory</value>
</rs:binding>
</rs:ResultSolution>
</rs:solution>
<rs:solution rdf:datatype="WIDE_known#meta:relation">
<rs:ResultSolution>
<rs:binding>
<variable rdf:datatype="WIDE_known#meta">SourceResourceID</variable>
<value>Provider_Ontology_Final_00158</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_known#meta">DestinationResourceID</variabl
<value>Provider_Ontology_Final_00150</value>
</rs:binding>
<rs:binding>
<variable rdf:datatype="WIDE_known#meta">Name</variable>
<value>describes</value>
</rs:binding>
</rs:ResultSolution>
</rs:solution>
</rs:ResultSet>
</rdf:RDF>
```

Part VI

BIBLIOGRAPHY

Bibliography

AAF (2004). Advanced authoring format : low-level container specification v1.0.1.

Adistambha, K., Doeller, M., Tous, R., Gruhne, M., Sano, M., Tsinaraki, C., Christodoulakis, S., Yoon, K., Ritz, C. H., and Burnett, I. S. (2007). The mpeg-7 query format: a new standard in progress for multimedia query by content. In *Proc. International Symposium on Communications and Information Technologies ISCIT '07*, pages 479–484.

Adobe-Systems-Incorporated (2008). Xmp specification part 2. standard schemas.

Ahronheim, J. R. (2002). *Cataloging the Web : Metadata, AACR, and MARC 21*. The Scarecrow Press, Inc.

Arens, Y., Knoblock, C. A., and Shen, W.-M. (1996). Query reformulation for dynamic information integration. *J. Intell. Inf. Syst.*, 6(2-3):99–130.

Avilés, J. A. G., León, B., Sanders, K., and Harrison, J. (2005). Journalists at digital television newsrooms in britain and spain: workflow and multi-skilling in a competitive environment. *Journalism Studies*, 5(1):87–100.

Baer, D., Groenewoud, P., Kapetanios, E., and Keuser, S. (2001). A semantics based interactive query formulation technique. In *Proceedings of the Second International Workshop on User Interfaces to Data Intensive Systems (UIDIS'01)*, page 43, Washington, DC, USA. IEEE Computer Society.

Baeza-Yates, R. A. and Ribeiro-Neto, B. A. (1999). *Modern Information Retrieval*. ACM Press / Addison-Wesley.

Bailer, W. and Schallauer, P. (2006). The detailed audiovisual profile: Enabling interoperability between mpeg-7 based systems. In *Proceedings of 12th Multimedia Modellinig Conference*, pages 217–224, Beijing.

Bailer, W. and Schallauer, P. (2008). *Multimedia Semantics - The Role of Metadata*, chapter Metadata in the Audiovisual Media Production Process, pages 65–84. Springer Berlin / Heidelberg.

Bailer, W., Schallauer, P., and Neuschmied, H. (2007). Mpeg-7 detailed audiovisual profile. Technical report, Joanneum Research.

BBC (2000). Standard media exchange framework (smef) data model 1.5.

Bechhofer, S., Volz, R., and Lord, P. W. (2003). Cooking the semantic web with the owl api. In *International Semantic Web Conference*, pages 659–675.

Bhogal, J., Macfarlane, A., and Smith, P. (2007). A review of ontology based query expansion. *Information Processing and Management: an International Journal*, 43(4):866–886.

Bloehdorn, S., Petridis, K., Simou, N., Tzouvaras, V., Avrithis, Y., Handschuh, S., Kompatsiaris, Y., Staab, S., and Strintzis, M. G. (2004). Knowledge representation for semantic multimedia content analysis and reasoning. In *Proceedings of the European Workshop on the Integration of Knowledge, Semantics and Digital Media Technology (EWIMT)*.

Broekstra, J., Kampman, A., and van Harmelen, F. (2002). Sesame: A generic architecture for storing and querying RDF and RDF Schema. In *ISWC 2002: Proceedings of the First International Semantic Web Conference, Sardinia, Italy*, pages 54–68.

Brunelli, R., Mich, O., and Modena, C. M. (1996). A survey on video indexing. *Journal of Visual Communications and Image Representation*, 10:78–112.

Brunelli, R., Mich, O., and Modena, C. M. (1999). A survey on the automatic indexing of video data„ ,. *Journal of Visual Communication and Image Representation*, 10(2):78–112.

Bürger, T. (2006). An intelligent media framework for multimedia content. In *Proceedings of International Workshop on Semantic Web Annotations for Multimedia (SWAMM)*.

Burkowski, F. J. (1992). Retrieval activities in a database consisting of heterogeneous collections of structured text. pages 112–125.

Bush, V. (1945). As we may think. *The Atlantic Monthly*.

Byrd, D. and Crawford, T. (2002). Problems of music information retrieval in the real world. *Inf. Process. Manage.*, 38(2):249–272.

Candela, L., Castelli, D., and Pagano, P. (2007). A reference architecture for digital library systems: Principles and applications. In Thanos, C., Borri, F., and Candela, L., editors, *DELOS Conference*, volume 4877 of *Lecture Notes in Computer Science*, pages 22–35. Springer.

Candela, L., Castelli, D., Pagano, P., and Simi, M. (2006). The digital library information mediator layer. In Agosti, M. and Thanos, C., editors, *IRCDL*, pages 29–36. DELOS: a Network of Excellence on Digital Libraries.

Castells, P., Fernández, M., and Vallet, D. (2007). An adaptation of the vector-space model for ontology-based information retrieval. *IEEE Trans. Knowl. Data Eng.*, 19:261–272.

Chasanis, V., Likas, A., and Galatsanos, N. (2008). Video rushes summarization using spectral clustering and sequence alignment. In *TVS '08: Proceedings of the 2nd ACM TRECVid Video Summarization Workshop*, pages 75–79, New York, NY, USA. ACM.

Cox, I. J., Miller, M. L., Minka, T. P., Papathomas, T. V., and Yianilos, P. N. (2000). The bayesian image retrieval system, pichunter: Theory, implementation and psychophysical experiments. *IEEE transactions on image processing*, 9:20–37.

Cox, M., tadic, L., and Mulder, E. (2006). *Descriptive Metadata for Television*. Focal Press.

Crucianu, M., Ferecatu, M., and Boujemaa, N. (2004). Relevance feedback for image retrieval: a short survey. In *In State of the Art in Audiovisual Content-Based Retrieval, Information Universal Access and Interaction including Datamodels and Languages (DELOS2 Report*.

Cusumano, M. A. (2005). Google: what it is and what it is not. *Commun. ACM*, 48(2):15–17.

Dasiopoulou, S., Tzouvaras, V., Kompatsiaris, I., and Strintzis, M. (2010). Enquiring mpeg-7 based ontologies. *Multimedia Tools and Applications*, 46(2):331–370.

Datamonitor-Analysists (2007). The evolving broadcast value chain, 2006 - 2012. Technical report, Datamonitor.

Datta, R., Joshi, D., Li, J., and Wang, J. Z. (2008). Image retrieval: Ideas, influences, and trends of the new age. *ACM Comput. Surv.*, 40(2):1–60.

Divakaran, A. (2008). *Multimedia Content Analysis: Theory and Applications*. Springer Publishing Company, Incorporated.

Döller, M. (2008). *Multimedia Semantics - The Role of Metadata*, chapter Specification of an MPEG-7 Query Format, pages 195–218. Springer Berlin / Heidelberg.

Dublin-Core (2007). The dublin core metadata element set: An american national standard. Technical report, The Dublin Core Metadata Element Set: An American National Standard.

Dufaux, F., Ansorge, M., and Ebrahimi, T. (2007). Overview of JPSearch: a Standard for Image Search and Retrieval. In *5th International Workshop on Content-based Multimedia Indexing (CBMI-2007)*.

Eakins, J. P. and Graham, M. E. (1999). Content-based image retrieval: A report to the jisc technology applications programme. Technical report, Institute for Image Data Research, University of Northumbria at Newcastle.

EBU-Technical-Department (2001). P/meta (metadata exchange standards. Technical report, EBU Technical Department.

Eidenberger, H. (2003). A new perspective on visual information retrieval. In *SPIE Electronic Imaging Symposium*. SPIE.

Ellis, D. P. (2006). Extracting information from music audio. *Commun. ACM*, 49(8):32–37.

Faaborg, A. J. (2003). *A Goal-Oriented User Interface for Personalized Semantic Search*. PhD thesis, B.A. Information Science Cornell University.

Faloutsos, C., Equitz, W., Flickner, M., Niblack, W., Petkovic, D., and Barber, R. (1994). Efficient and effective querying by image content. *Journal of Intelligent Information Systems*, 3:231–262.

Fan, J., Elmagarmid, A. K., Zhu, X., Aref, W. G., and Wu, L. (2004). Classview: hierarchical video shot classification, indexing, and accessing. *IEEE Transactions on Multimedia*, 6(1):70–86.

Feldmann, I., Waizenegger, W., and Schreer, O. (2008). Extraction of 3d scene structure for semantic annotation and retrieval of unedited video. In *MMSP*, pages 82–87.

Furht, B. (2006). *Encyclopedia of Multimedia*. Springer.

Geetha, P. and Narayanan, V. (2008). A survey of content-based video retrieval. *Journal of Computer Science*, 4:474–486.

Grau, B. C. and Motik, B. (2008). Owl 1.1 web ontology language: Model-theoretic semantics. Technical report, Oxford University.

Gruhne, M., Tous, R., Delgado, J., Doeller, M., and Kosch, H. (2007). Mp7qf: An mpeg-7 query format. In *AXMEDIS '07: Proceedings of the Third International Conference on Automated Production of Cross Media Content for Multi-Channel Distribution*, pages 15–18, Washington, DC, USA. IEEE Computer Society.

Gudivada, V. N., Raghavan, V. V., Grosky, W. I., and Kasanagottu, R. (1997). Information retrieval on the world wide web. *IEEE Internet Computing*, 1(5):58–68.

Günter, G., Behrendt, V., Bürger, T., Westenthaler, R., and Glachs, D. (2007). Basic specification of the intelligent media framework. Technical report, Salzburg Research.

Haarslev, V. and Möller, R. (2001). Racer system description. In Goré, R., Leitsch, A., and Nipkow, T., editors, *International Joint Conference on Automated Reasoning, IJCAR'2001, June 18-23, Siena, Italy*, pages 701–705. Springer-Verlag.

Haas, W., Bailer, W., and Hausenblas, M. (2007). Media Semantics. *ÖGAI Journal*, 26(1):24–29.

Haas, W. and Mayer, H. (2001). Mpeg and its relevance for content-based multimedia retrieval. *J. UCS*, 7(6):530–547.

Hafner, J., Sawhney, H. S., Equitz, W., Flickner, M., and Niblack, W. (1995). Efficient color histogram indexing for quadratic form distance functions. *IEEE Trans. Pattern Anal. Mach. Intell.*, 17(7):729–736.

Han, S., Yoon, K., and Kweon, I. (2000). A new technique for shot detection and key frames selection in histogram space. In *Proceedings of the 12th Workshop on Image Processing and Image Understanding (IPIU)*.

Hanjalic, A. (2005). Adaptive extraction of highlights from a sport video based on excitement modeling. *IEEE Transactions on Multimedia*, 7(6):1114–1122.

Hauptmann, A., Yan, R., and Lin, W.-H. (2007). How many high-level concepts will fill the semantic gap in video retrieval? In *CVIR*.

Hauptmann, E. G., Jin, R., and Ng, T. D. (2003). Video retrieval using speech and image information. In *In Storage and Retrieval for Multimedia Databases 2003, EI'03 Electronic Imaging*.

Hoos, H. H., Renz, K., and Görg, M. (2001). GUIDO/MIR — an experimental musical information retrieval system based on GUIDO Music Notation. In *Proc. 2nd International Symposium on Music Information Retrieval (ISMIR 2001)*, pages 41–50. Indiana University, Bloomington (IN), USA.

Ilarri, S., Mena, E., Illarramendi, A., and Marcos, G. (2010). A location-aware system for monitoring sport events. In *Eight International Conference on Advances in Mobile Computing & Multimedia (MoMM 2010), Paris (France)*, pages 305–312. ACM Press, ISBN 978-1-4503-0440-5, Austrian Computer Society (OCG), ISBN 978-3-85403-273-1. Selected for special issue.

IPTC, I. P. T. C. (2009). Iptc standards. news ml g2.

ISO (2001). Mpeg-7. multimedia content description interface. iso/iec 15938.

ISO (2005a). Information technology – multimedia content description interface – part 11: Mpeg-7 profile schemas iso/iec tr 15938.

ISO (2005b). Iso-iec 21000-2 information technology - multimedia framework - part 2 digital item declaration (did).

ISO (2007). Iso/iec 21000-7 information technology - multimedia framework - part 7: Digital item adaptation (dia).

ISO/IEC (2007). Iso/iec jtc1/sc29/wg1 wg1n4083, "iso/iec pdtr 24800-1: Framework and system components". Technical report, ISO/IEC.

Karvounarakis, G., Alexaki, S., Christophides, V., Plexousakis, D., and Scholl, M. (2002). Rql: a declarative query language for rdf. In *Proceedings of the 11th international conference on World Wide Web*, pages 592–603, New York, NY, USA. ACM.

Kavanagh, M. J. and Thite, M. (2009). *Human resource information systems : basics, applications, and future directions / [edited by] Michael J. Kavanagh, Mohan Thite*. Sage, Los Angeles :.

Kerschberg, L. and Weishar, D. (2000). Conceptual models and architectures for advanced information systems. *Applied Intelligence*, 13(2):149–164.

Kienast, G., Zeiner, H., Hofmair, P., Schlatte, R., Thallinger, G., Bürger, T., Denissenko, D., and Wahler, A. (2007). Representation techniques for multimedia objects. Technical report, Joanneum Research.

Kim, W. and Seo, J. (1991). Classifying schematic and data heterogeneity in multidatabase systems. *Computer*, 24(12):12–18.

Klapuri, A. (2004). *Signal Processing Methods for the Automatic Transcription of Music*. PhD thesis, Tampere University of Technology, Finnland.

Knublauch, H., Fergerson, R. W., Noy, N. F., and Musen, M. A. (2004). The protégé owl plugin: An open development environment for semantic web applications. In *International Semantic Web Conference*, pages 229–243.

Knuth, D. E. (1964). backus normal form vs. backus naur form. *Commun. ACM*, 7(12):735–736.

Köhler, J., Philippi, S., Specht, M., and Rüegg, A. (2006). Ontology based text indexing and querying for the semantic web. *Know.-Based Syst.*, 19(8):744–754.

Lai, W.-C., Chang, C., Chang, E., Cheng, K.-T., and Crandell, M. (2002). Pbir-mm: multimodal image retrieval and annotation. In *MULTIMEDIA '02: Proceedings of the tenth ACM international conference on Multimedia*, pages 421–422, New York, NY, USA. ACM Press.

Lamsfus, C., Karagiannis, G., Sotiropoulou, S., Eskudero, H., Marcos, G., Linaza, M. T., and Daniilia, S. (2004). The art-e-fact ontology: a possible contribution to cidoc crm. In *10th joined meeting between the CIDOC CRM Special Interest Group and the ISO/TC46/SC4/WG9*, pages 1–5, Nuremberg, Germany.

Lancaster, F. W. (1968). *Information Retrieval Systems: Characteristics, Testing and Evaluation*. Wiley, New York.

Larson, R. (2007). Principles of information retrieval. Technical report, University of California, Berkeley. School of Information.

Larson, R. R. (2010). *Encyclopedia of Library and Information Sciences, Third Edition*, chapter Information Retrieval Systems. Marcia J. Bates and Mary Niles Maack.

Lew, M. S., Sebe, N., Djeraba, C., and Jain, R. (2006). Content-based multimedia information retrieval: State of the art and challenges. *ACM Trans. Multimedia Comput. Commun. Appl.*, 2(1):1–19.

Lu, Y., Hu, C., Zhu, X., Zhang, H., and Yang, Q. (2000). A unified framework for semantics and feature based relevance feedback in image retrieval systems. In *Image Retrieval Systems, ACM Multimedia*, pages 31–37.

Luhn, H. P. (1957). A statistical approach to the mechanized encoding and searching of literary information. *IBM Journal of Research and Development*, 1(4):309–317.

Luo, H. (2007). *Concept-based large-scale video database browsing and retrieval via visualization*. PhD thesis, The University of North Carolina at Charlotte, Charlotte, NC, USA. Adviser-Fan, Jianping.

Marcos, G. (2007). D8 rushes metadata model specification (rushes fp6 045189). Technical report, Vicomtech.

Marcos, G., Alonso, K., Illarramendi, A., Olaizola, I. G., and Flórez, J. (2009). Dms-1 driven data model to enable a semantic middleware for multimedia information retrieval in a broadcaster. In *Proceedings of the 4th International Workshop on Semantic Media Adaptation and Personalization.*

Marcos, G., Eskudero, H., Lamsfus, C., and Linaza, M. T. (2005a). Data retrieval from a cultural knowledge database. In *Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS)*, Montreux/Switzerland.

Marcos, G., Illarramendi, A., Olaizola, I. G., and Flórez, J. (2011). A middleware to enhance current multimedia retrieval systems with content-based functionalities (to be published). *Multimedia Systems*.

Marcos, G. and Jiménez, I. (2007). Contribution to jpsearch standardization activity: Mapping on jpsearch architecture of the wide image search and retrieval system. doc number: wg1n4394. Technical report, Vicomtech.

Marcos, G., Jiménez, I., and Smithers, T. (2005b). D20 technical report of the meta level (wide ist 2001 34417). Technical report, Vicomtech.

Marcos, G., Krämer, P., Illarramendi, A., Olaizola, I. G., and Flórez, J. (2008). Semantic middleware to enhance multimedia retrieval in a broadcaster. In Luke, D., Hardman, L., Hauptmann, A., Paulus, D., and Staab, S., editors, *Semantic Multimedia, Third International Conference on Semantic and Digital Media Technologies, SAMT 2008 Koblenz, Germany, December 2008, Proceedings*, pages 74–88. Springer-Verlag Berlin Heidelberg.

Marcos, G., Lamsfus, C., Eskudero, H., and Linaza, M. T. (2005c). Semantic based querying and retrieving information for artistic expressions: The art-e-fact ontology. In *Fourth International Workshop on Content-Based Multimedia Indexing (CBMI05)*, Riga, Latvia.

Marcos, G., Smithers, T., Jiménez, I., and Toro, C. (2007). Meta level: Enabler for semantic steered multimedia retrieval in an industrial design domain. *Systems Science*, 2:15–22.

Marcos, G., Smithers, T., Jiménez, I., Posada, J., Stork, A., Pianciamore, M., Castro, R., Marca, S. D., Mauri, M., Selvini, P., Sevilmis, N., Thelen, B., and Zecchino, V. (2005d). A semantic web based approach to multimedia retrieval. In *Fourth International Workshop on Content-Based Multimedia Indexing (CBMI05)*, Riga, Latvia.

McBride, B. (2002). Jena: a semantic web toolkit. *Internet Computing, IEEE*, 6(6):55–59.

Meghini, C., Sebastiani, F., and Straccia, U. (2001). A model of multimedia information retrieval. *J. ACM*, 48(5):909–970.

Messina, A., Boch, L., Dimino, G., Bailer, W., Schallauer, P., Allasia, W., Groppo, M., Vigilante, M., and Basili, R. (2006). Creating rich metadata in the tv broadcast archives environment: The prestospace project. In *Proc. Second International Conference on Automated Production of Cross Media Content for Multi-Channel Distribution AXMEDIS '06*, pages 193–200.

Metadata-Professional-workshop (2007). Workshop "Annotations and Metadata models for Audiovisual/Multimedia Content" IRT, Munich, 21 - 22 November 2007.

Metadata-Working-Group (2009). Guidelines for handling metadata. v 1.0.1.

Multimedia-Research-Group (2004). Digital media asset management & workflow management in the broadcast industry: Survey & analysis. Technical report, Research And Markets.

Naphade, M. R. and Huang, T. S. (2002). Extracting semantics from audiovisual content: The final frontier in multimedia retrieval. *IEEE Trans. on Neural Networks*, 13(4):793–810.

Natsev, A. P., Haubold, A., Tešić, J., Xie, L., and Yan, R. (2007). Semantic concept-based query expansion and re-ranking for multimedia retrieval. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, pages 991–1000, New York, NY, USA. ACM.

Navarro, G. and Baeza-Yates, R. (1997). Proximal nodes: a model to query document databases by content and structure. *ACM Transactions on Information Systems*, 15(4):400–435.

Neo, S.-Y., Zhao, J., Kan, M.-Y., and Chua, T.-S. (2006). Video retrieval using high level features: Exploiting query matching and confidence-based weighting. In *Image and Video Retrieval*, pages 143–152. Springer Berlin / Heidelberg.

Ngo, C.-W., chuen Pong, T., and Zhang, H.-J. (2003). Motion analysis and segmentation through spatio-temporal slices processing. *IEEE Trans. Image Processing*, 12:341–355.

Nitto, E. D., Pianciamore, M., and Selvini, P. (2002). The role of agents in knowledge management. In *WOA*, pages 29–34.

Olaizola, I. G., Marcos, G., Krämer, P., Flórez, J., and Sierra, B. (2009). Architecture for semi-automatic multimedia analysis by hypothesis reinforcement. In *IEEE International Symposium on Broadband Multimedia Systems and Broadcasting*.

Pan, Z. (2005). Benchmarking dl reasoners using realistic ontologies. In *OWLED*.

Papadopoulos, G. T., Mezaris, V., Kompatsiaris, I., and Strintzis, M. G. (2007). Combining global and local information for knowledge-assisted image analysis and classification. *EURASIP J. Adv. Signal Process*, 2007(2):18–18.

Pastra, K. and Piperidis, S. (2006). Video Search: New Challenges in the Pervasive Digital Video Era. *Journal of Virtual Reality and Broadcasting*, 3(11).

Peinado, A. (2006). *Speech recognition over digital channels : robustness and standards.* ENGINEERING TE 13 PEI.

Piasecki, M. and Beran, B. (2009). A semantic annotation tool for hydrologic sciences. *Earth Science Informatics*, 2(3):157–168.

Rabiner, L. and Juang, B.-H. (1993). *Fundamentals of speech recognition*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Robertson, S. E. (1997). The probability ranking principle in ir. pages 281–286.

Rui, Y., Huang, T. S., Ortega, M., and Mehrotra, S. (1998). Relevance feedback: a power tool for interactive content-based image retrieval. *Circuits and Systems for Video Technology, IEEE Transactions on*, 8(5):644–655.

Ruthven, I. and Lalmas, M. (2003). A survey on the use of relevance feedback for information access systems. *Knowl. Eng. Rev.*, 18(2):95–145.

Salton, G. (1971). *The SMART Retrieval System—Experiments in Automatic Document Processing*. Prentice-Hall, Inc., Upper Saddle River, NJ, USA.

Salton, G., Wong, A., and Yang, C. S. (1975). A vector space model for automatic indexing. *Commun. ACM*, 18(11):613–620.

Santini, S. (2003). *Annotation for the Semantic Web*, chapter Semantics without Annotation. IOS Press.

Santini, S. and Jain, R. (1998). Beyond query by example. In *Proc. IEEE Second Workshop on Multimedia Signal Processing*, pages 3–8.

Sarris, N. (2009). Rushes-project: D5.5 report on validation and performance evaluation. Technical report, Athens Technology Centre.

Schach, S. R. (2007). *Object-Oriented and Classical Software Engineering*. McGraw-Hill, Inc., New York, NY, USA.

Schallauer, P., Bailer, W., and Thallinger, G. (2006). A description infrastructure for audiovisual media processing systems based on mpeg-7. *Journal of Universal Knowledge Management*, 1(1):26–35.

Serb, K. (1997). Towards the digital archive-a paradigm shift in exploiting mediaassets. In *Broadcasting Convention*.

Sevilmis, N., Stork, A., Smithers, T., Posada, J., Pianciamore, M., Castro, R., Jiménez, I., Marcos, G., Mauri, M., Selvini, P., Thelen, B., and Zecchino, V. (2005). Knowledge sharing by information retrieval in the semantic web. In *ESWC*, pages 471–485.

Shamsfard, M., Nematzadeh, A., and Motiee, S. (2006). Orank: An ontology based system for ranking documents. *International Journal of Computer Science*, 1:225–231.

Shi, X. and Manduchi, R. (2003). A study on Bayes feature fusion for image classification. In *Proceedings of the IEEE Workshop on Statistical Analysis in Computer Vision*, Madison, Wisconsin.

Sikora, T. (2001). The mpeg-7 visual standard for content description-an overview. 11(6):696–702.

Simou, N. and Kollias, S. (2007). Fire : A fuzzy reasoning engine for impecise knowledge. In *K-Space PhD Workshop*. K-Space PhD Students Workshop, Berlin, Germany.

Simou, N., Saathoff, C., Dasiopoulou, S., Spyrou, E., Voisine, N., Tzouvaras, V., Kompatsiaris, I., Avrithis, Y. S., and Staab, S. (2005). An ontology infrastructure for multimedia reasoning. In Atzori, L., Giusto, D. D., Leonardi, R., and Pereira, F., editors, *VLBV*, volume 3893 of *Lecture Notes in Computer Science*, pages 51–60. Springer.

Singhal, A. (2001). Modern information retrieval: A brief overview. *Bulletin of the IEEE Computer Society Technical Committee on Data Engineering*, 24(4):35–42.

Sirin, E., Parsia, B., Grau, B. C., Kalyanpur, A., and Katz, Y. (2007). Pellet: A practical owl-dl reasoner. *Web Semantics: Science, Services and Agents on the World Wide Web*, 5(2):51–53.

Smeulders, A. W. M., Worring, M., Santini, S., Gupta, A., and Jain, R. (2002). Content-based image retrieval at the end of the early years. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(12):1349–1380.

Smith, P. J. and Schirling (2006). Metadata standards roundup. *IEEE Multimedia*, 13:84–88.

Smithers, T., Posada, J., Stork, A., Pianciamore, M., Ferreira, N., Grimm, S., Jiménez, I., Marca, S. D., Marcos, G., Mauri, M., Selvini, P., Sevilmis, N., Thelen, B., and Zecchino, V. (2004). Information management and knowledge sharing in wide. In *EWIMT*.

SMPTE (2008). Smpte 12m-1-2008 television: Time and control code.

SMTPE (2007). Metadata dictionary registry of metadata element descriptions. Technical report, SMTPE.

Snoek, C. G. M., Huurnink, B., Hollink, L., de Rijke, M., Schreiber, G., and Worring, M. (2007). Adding semantics to detectors for video retrieval. *IEEE Trans. on Multimedia*, 9(5):975–986.

Snoek, C. G. M. and Worring, M. (2005). Multimodal video indexing: A review of the state-of-the-art. *Multimedia Tools Appl.*, 25(1):5–35.

Soergel, D. (1974). *Indexing Languages and Thesauri: Construction and Maintenance*. John Wiley & Sons Inc.

Spanias, A., Painter, T., and Atti, V. (2007). *Audio Signal Processing and Coding*. Wiley-Interscience.

Stamou, G. and Kollias, S. (2005). *Multimedia Content and the Semantic Web: Standards, Methods and Tools*. John Wiley & Sons.

Stoilos, G., Stamou, G., Pan, J., Tzouvaras, V., and Horrocks, I. (2007). Reasoning with very expressive fuzzy description logics. *Journal of Artificial Intelligence Research*, 30:273–320.

Stork, A., Sevilmis, N., Smithers, T., Posada, J., Pianciamore, M., Castro, R., Jiménez, I., Marcos, G., Mauri, M., Selvini, P., and Thelen, B. (2005). A semantic web approach to ce. In *11th International Conference on Concurrent Enterprising: Integrated Engineering of Products, Services and Organisations*, pages 193–200, University BW Munich, Germany.

Styltsvig, H. B. (2006). *Ontology-based Information Retrieval*. PhD thesis, Computer Science Roskilde University, Denmark.

Thelen, B., Sevilmis, N., Stork, A., Castro, R., Jiménez, I., Marcos, G., Posada, J., Smithers, T., Mauri, M., Pianciamore, M., Selvini, P., and Zecchino, V. (2005). Information management on the basis of semantic-web techniques, or a google for developers. In *VDI-Verlag - Gesellschaft Fahrzeug- und Verkehrstechnik: Erprobung und Simulation in der Fahrzeugentwicklung: Mess- und Versuchstechnik*, pages 167–180, Würzburg. VDI-Verlag, 2005.

Troncy, R., Bailer, W., Hausenblas, M., Hofmair, P., and Schlatte, R. (2006). Enabling multimedia metadata interoperability by defining formal semantics of mpeg-7 profiles. In *SAMT*, pages 41–55.

Tse, P. K. C. (2008). *Multimedia Information Storage and Retrieval:Techniques and Technologies*. IGI Publishing.

TV-Anytime, F. (2007). Ts 102 822-2 broadcast and on-line services: Search, select, and rightful use of content on personal storage systems ("tv-anytime"); part 2: System description.

Tzitzikas, Y., Spyratos, N., and Constantopoulos, P. (2002). Query translation for mediators over ontology-based information sources. In *Proceedings of the Second Hellenic Conference on AI*, pages 423–436, London, UK. Springer-Verlag.

van Rijsbergen, C. J. (1979). *Information Retrieval*. Butterworths, London, 2 edition.

Vasconcelos, N. and Lippman, A. (1997). Towards semantically meaningful feature spaces for the characterization of video content. In *ICIP '97: Proceedings of the 1997 International Conference on Image Processing (ICIP '97) 3-Volume Set-Volume 1*, page 25, Washington, DC, USA. IEEE Computer Society.

Wang, H., Liu, S., and Chia, L.-T. (2006). Does ontology help in image retrieval?: a comparison between keyword, text ontology and multi-modality ontology approaches. In *ACM Multimedia*, pages 109–112.

Wei, X.-Y. and Ngo, C.-W. (2007). Ontology-enriched semantic space for video search. In *MULTIMEDIA '07: Proceedings of the 15th international conference on Multimedia*, pages 981–990, New York, NY, USA. ACM.

White, A. M., Baker, A., Bloss, M., Burrows, P. E., Efthimiadis, E. N., Brooks, M., MacCarn, D., Shepard, T., and Twohill, C. (2003). Pb core — the public broadcasting metadata initiative: progress report. In *DCMI '03: Proceedings of the 2003 international conference on Dublin Core and metadata applications*, pages 1–10. Dublin Core Metadata Initiative.

Wilkinson, J. (2003). Material exchange format (mxf) descriptive metadata scheme - 1 (standard, dynamic). Technical report, Pro-MPEG.

Wu, Y., Chang, E. Y., and Tseng, B. L. (2005). Multimodal metadata fusion using causal strength. In *MULTIMEDIA '05: Proceedings of the 13th annual ACM international conference on Multimedia*, pages 872–881, New York, NY, USA. ACM Press.

Yu, Y., Kim, J., Shin, K., and Jo, G. S. (2009). Recommendation system using location-based ontology on wireless internet: An example of collective intelligence by using 'mashup' applications. *Expert Syst. Appl.*, 36(9):11675–11681.

Zhang, Q. and Izquierdo, E. (2007). Combining low-level features for semantic extraction in image retrieval. *EURASIP Journal on Advances in Signal Processing*, 1:12.

Zhang, Q. and Izquierdo, E. (2008). Visual information retrieval. In *Encyclopedia of Multimedia*, pages 971–979. Springer Science+Buisiness Media, LLC, 2nd edition.

Zhang, Y.-j. (2006). *Semantic-Based Visual Information Retrieval*. IRM Press.

Zoelzer, U. (2008). *Digital Audio Signal Processing*. John Wiley & Sons Software.