An extensible platform for semantic classification and retrieval of multimedia resources

Paolo Pellegrino, Fulvio Corno

Politecnico di Torino, C.so Duca degli Abruzzi 24,
10129, Torino, Italy
{paolo.pellegrino, fulvio.corno}@polito.it

Abstract

This paper introduces a possible solution to the problem of semantic indexing, searching and retrieving heterogeneous resources, from textual as in most of modern search engines, to multimedia. The idea of "anchor" as information unit is here introduced to view resources from different perspectives and to access existing resources and metadata archives. Moreover, the platform uses an ontology as a conceptual representation of a well-defined domain in order to semantically classify and retrieve anchors (and the related resources). Specifically, the architecture of the proposed platform aims at being as modular and easily extensible as possible, in order to permit the inclusion of state-of-the-art techniques for the classification and retrieval of multimedia resources. Eventually, the adoption of Web Services as interface technology facilitates the exposition of the semantic functionalities and of content management to web application designers and users without any additional overload on the content creation and maintenance workflow.

1. Introduction

Resources, exponentially growing in number on the Internet, are slowly but increasingly shifting from textual to multi-media. The increase in storage capacity and computational power, as well as in connectivity and bandwidth, permits and stimulates the creation of multimedia digital archives, both for sake of resource preservation and for ease of fruition. However, current methodologies for classification of multimedia resources are still mostly based on tags and metadata which are usually manually added to the resources during the archival phases. Approaches oriented to the automatic discovery or extraction of information from multimedia resources are of course under study, but still rather inefficient and immature, especially if compared with the consolidated algorithms and technologies which are currently applied to textual documents. In addition, the Semantic Web initiative is pushing even forward the classic knowledge paradigms, shifting mainly from a keyword-based view of web documents to a more articulated representation of knowledge, based on concepts and relations between concepts (ontologies). These new technologies are recently growing in popularity and seem very promising, but are still mostly exploited for textual resources only. In effect, text is the simplest form of knowledge representation which can be both understood by humans and quickly processed by machines. So, while text itself is already efficaciously used as knowledge representation for machines, digital multimedia resources like images and videos cannot in general be used as they are for efficient classification and retrieval. Some intensive processing or manual metadata creation is necessary to extract and associate semantic information to a given multimedia resource.

In this context, the proposed work aims at providing a flexible, lightweight and ready-to-use platform for the semantic classification, search and retrieval of heterogeneous resources. Particularly, the proposed approach keeps in high consideration the reuse of existing digital repositories and metadata, and provides means for the exploitation and testing of the recent knowledge extraction technologies applied to multimedia resources. A simple information container and collector, called anchor, is therefore introduced as information unit to wrap knowledge about given resources and to permit a classification based on the concepts defined in the domain ontology. It is therefore possible to exploit semantic based technologies, commonly used for textual resources, even for an enhanced fruition of multimedia content, both in terms of classification and search for the desired resources in the available repositories and in terms of retrieval and composition of the requested material in a suitable form for the users. In fact, semantic-oriented technologies can be easily and uniformly integrated in the platform to transparently enable web application designers to enhance their applications with little effort, guaranteeing an improved flexibility and resource fruition to the users.

In the next sections of this paper, starting from section 2, related works are presented, followed by a description of the design principles in section 3, and the architectural design in section 4. Section 5 presents the tests performed, and eventually, in section 6, the entire work is summarized as a conclusion.

2 Related Works

Three main areas may be individuated that particularly pertain to this work and substantially involve technologies for extraction of knowledge from multimedia resources, metadata exploitation and merging, and knowledge representation issues.

Various projects have recently proposed solutions to the problem of maintaining repositories of digital resources. The cataloguing of resources is generally handled manually or semi-automatically. Usually, each repository contains only a specific type of resource, as it may happen for books, videos, etc., and specific metadata schemas are used for the classification. Some approaches try to automatically extract and exploit audiovisual features for the classification, such as the SCHEMA project [6], and Caliph & Emir [10] which are mainly focused on images.

The DSpace project [7] proposes the archival of heterogeneous resources through the exploitation of new emerging standards for resource identification and handling such as DOI and HANDLE [8]. Here, however, the attention is less focused on the semantics of the resource content. The Simile project [9] exploits the DSpace work to efficiently manage distributed metadata, in form of RDF triples, so requiring the conversion of existing schemas into this W3C recommended format (or equivalent).

Instead, our approach aims at providing an easily manageable, extensible and ready-to-use semantic-oriented platform. The existing H-DOSE platform has already been successfully and easily integrated into a number of different applications (Moodle [11], Muffin [12]), so offering semantic services along with those already existent ones. Although specific tests are yet to be published, the authors believe that the improvements added to this platform are worth the effort, and are therefore explained in this paper, starting from the next section.

For what concerns metadata, the necessity for diversification and fine-grained levels of detail in various cases has brought to the creation of the most diverse schemas for the collection of document metadata. However, it is extremely difficult to find a single schema that is both popular and capable of describing any type of resource. In fact, the existing metadata schemas are either too generic, like the Dublin Core (DC) [5], too specific like the MPEG-7 [13]. Half-way approaches are not yet widely accepted as standards, like the Harmony project for the ABC ontology [4], which integrates DC with MPEG-7 to provide a detailed description of an audiovisual resource. The main difficulty in similar cases resides in identifying possible mappings between elements which are similar in the joined schemas, as well as in managing the increased complexity of the new schema. It becomes therefore difficult to reconduct metadata of existing digital archives to a new schema, while many tags of composite schemas may remain empty even for newly created archives unless manually filled in, operation which is often unscalable. While the adoption of a unique, simple and widespread standard would still be desirable, in this project a flexible approach has been adopted. Particularly, as explained in the next sections, the adoption of a simple XML container has been chosen. In this way, different schemas may be easily included independently and therefore reused as they are. This also means that available software and technologies that can parse and use existing schemas may be reused in the platform with minimal adaptation.

3 Design Principles

The platform architecture is based on the existing H-DOSE platform [1]. H-DOSE currently provides semantic functionalities for web applications through an easy to access interface, allowing rapid inclusion of services into the existing development workflow and trying to maximize the benefit/cost ratio for the inclusion of semantics in web applications. In particular, H-DOSE is focused on semantic search and indexing services, providing means for classifying a textual web resource with respect to a conceptual model represented as an ontology. It also transparently stores conceptual information (annotations) about indexed resources and retrieves such resources in response to user queries, according to the semantic similarity between the queries and the annotated resources. The relations specified in the ontology allow to expand the search of documents through correlated concepts, as explained in [2]. The main functionalities that are offered are therefore semantic indexing, search and deep-search of textual resources.

The new architecture aims at maintaining backward compatibility and the same functionalities of the existing platform, while adding support for semantic classification and retrieval of heterogeneous resources. The main novelty proposed here is therefore the introduction of a modular substructure that allows to easily extend the semantic capabilities of the platform with state-of-the-art algorithms suitable for the classification of any given type of resource. The semantic framework which collates the whole architecture can then be exploited to retrieve resources with a minimal effort basing simply on the conceptual relevance of the resource content. Particularly, two elements play a relevant role in the architecture of the new multimedia framework and will therefore be described in more details in the next sections: anchors, which associate resources to descriptions – rather than using the textual document itself as a self-describing search object –, and mapping modules, which exploit the semantics of given descriptions to generate links to the

domain ontology (annotations). The rest of the components mainly covers automatic creation of anchors, management of semantic annotations and resource retrieval.

3.1 Anchors and Resources

Generally, any multimedia resource may be "described" from different perspectives or views, basing on a precise context, or level of detail, subcomponents, etc. For instance, a video can be considered as a whole or as a temporal sequence of chapters. Some particular frame could also be described in detail, and so on. Hence, resources, which are usually stored as atomic items, can be considered as more versatile sources of information. For this reason, each perspective of a given resource is associated to one *anchor* (Figure 1), which substitutes the resource as information unit. Each anchor is composed by *two type of elements*: the first one indicates the *target* location of the resource and the precise part of interest of the resource, so "physically" defining the perspective, e.g., as a temporal or spatial restriction, or as a text fragment, etc.; the second type of elements instead, may appear multiple times in an anchor and represents a semantic *description* about the so defined perspective. Both of these types of elements are actually simple containers which are to be customized depending on the particular necessities. So a simple perspective which refers to a web resource as a whole may be simply targeted by specifying its URL, whereas a spatiotemporal restriction of an audiovisual clip requires more articulated target details. In any case, the anchor target element should simply specify the necessary information to permit the retrieval of the targeted resource perspective. On the contrary, the intended use of the description elements is the collection of data, metadata or features useful for the classification of the anchor. So, for instance, plain text may be used as description for audiovisual resources in order to exploit textual classifiers. Furthermore, Dublin Core (DC) metadata (or other more specific schemas) can be used to indicate additional information like author, date of creation, etc., which provide multidimensional views for the given anchors. Eventually, descriptions specific for certain types of media can be adopted for classification with feature- or case-
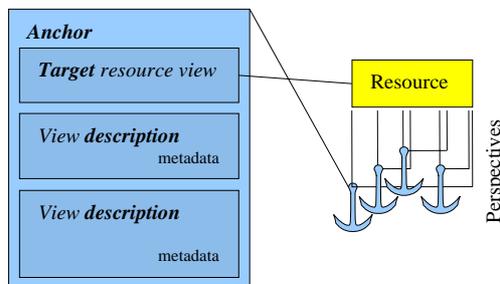
based approaches. Customized targets and descriptions are parsed through simple ad-hoc modules that can be easily added as platform extensions. Similarly, classification algorithms can be added to the platform to fully exploit the flexibility introduced with the descriptions. In this case, the main task is the interpretation of a description basing on the conceptual domain provided as an ontology to the platform. As a result, each description may be associated to one or more concepts defined in the ontology, so contributing to the classification of the container anchor. To summarize, the anchors (information units) may be used as generic yet uniform containers for various forms of descriptions, referring to the targeted resource perspective.

An XML descriptor is used to collect various anchors for the same resource, or even for multiple resources. The XML descriptors can therefore be used as a sort of distributed repository, the stored information being located in the same location of the described resources. Alternatively, the descriptors can be used as a central local cache of existing metadata coming from the resource archives, that therefore remain separated from the related anchors. The platform keeps track of all the annotated descriptors, anchors and of the described resources, so that they can be easily retrieved or updated.

For example, a target element for a simple fragment of an XHTML web page can be simply expressed as follows:

```
<target type="urn:dose:target:xhtml"
src="http://www.eg.org/index.html#/body/h1" />
```

In this case a `type` attribute specifies how to handle and parse the target element as well as how to eventually retrieve the resource specified through the `src` attribute.

Instead, for a simple video the MPEG-7 schema could be used to specify which part of it we are focusing on:

```
<target type="urn:dose:target:mpeg7"
  src="http://www.example.org/video.mpg">
  <Mpeg7 xmlns="urn:mpeg:mpeg7:schema:2001"
    xmlns:xsi=
    http://www.w3.org/2001/XMLSchema-instance
    xmlns:mpeg7="urn:mpeg:mpeg7:schema:2001">
    <Description xsi:type="ContentEntityType">
      <MultimediaContent
          xsi:type="AudioVisualType">
      <AudioVisual>
        <TemporalDecomposition>
          <AudioVisualSegment id="cha01">
            <MediaTime>
              <!--from beginning...-->
              <MediaTimePoint>
                T00:00:00:00F25
              </MediaTimePoint>
              <!--to 12'25sec & 17/25sec-->
              <MediaDuration>
                PT0H12M40S17N25F
              </MediaDuration>
            </MediaTime>
          </AudioVisualSegment>
        </TemporalDecomposition>
      </AudioVisual>
      </MultimediaContent>
    </Description>
  </Mpeg7>
</target>
```


Figure 1: Anchors and resources

In this case an appropriate module is necessary to parse the MPEG-7 elements and to possibly retrieve the desired resource fragment(s).

Similar examples can be proposed for description elements. For instance, in a description element it is possible to simply specify plain inline text:

```
<description
  type="urn:dose:description:plainText"/>
  This is a sample textual description
</description>
```

Differently than target elements, description elements are used to classify an anchor. So, a specific module will parse and handle specific types of descriptions and exploit their content to associate the anchor to concepts in the domain ontology (which abstracts the relevant concepts from the content of the archived resources).

In this example, the Dublin Core schema is exploited:

```
<description
  type="urn:dose:description:DublinCore"
  xmlns:dc=http://purl.org/dc/elements/1.1/>
  <dc:title xml:lang="en">
    Wildlife and nature
  </dc:title>
  <dc:author>John Doe</dc:author>
  <dc:type>Documentary</dc:type>
  <dc:language>en</dc:language>
</description>
```

Other well-known schemas may of course be exploited, such as MARC [14], CIDOC [15] and the Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH) [16].

3.2 Anchor Life-cycle

The role of the anchors in the platform is to serve as identifiers for resource perspectives and as collectors of metadata for indexing and search. Their use in the platform is explained in this section, and basically covers the creation of anchors and descriptors, the indexing phase, including the creation of semantic annotations, and the search phase, which describes the composition of

advanced queries and the retrieval of anchors and resources through the previously indexed annotations.

First of all, the anchors must be created (step 1 in Figure 2 on the left). An automatic procedure may generate anchors by extracting metadata from a given resource (step 2 and 3). As explained in more detail in the next sections, the Resource Inspector is one of the types of modules that can be extended and seamlessly integrated with new implementations capable of generating descriptors (i.e., anchor containers) for different resource types. The main idea here is the provision of a service capable of automatically index a given (and supported) resource. A centralized repository guarantees that resources are not indexed multiple times, unless modified since the last indexing operation. In such a case the older anchors and their annotations must be updated or removed.

Once some anchors have been created, they can be examined for indexing (step 4). For each anchor, the process of indexing is based on the creation of weighted links (generally referred to as semantic annotations) to the concepts of the domain ontology (Figure 3). The annotations are firstly created for each description contained in a given anchor and then combined to yield the final weighted links (spectrum) for such anchor (steps 5 and 6 in Figure 2). As the descriptions are generally variegated in type and content, a number of different modules (which can be easily added as extensions to the platform) perform the mapping between metadata and ontology concepts in order to create the annotations. Particularly each mapping module can implement a different strategy and exploit specific "samples" attached to the ontology concepts. For instance, textual descriptions may be mapped through a tf/idf algorithm exploiting multilingual synsets for each concept (i.e., set of words that identify the concept)[2][3]. Instead, descriptions with visual features could be mapped through a case-based
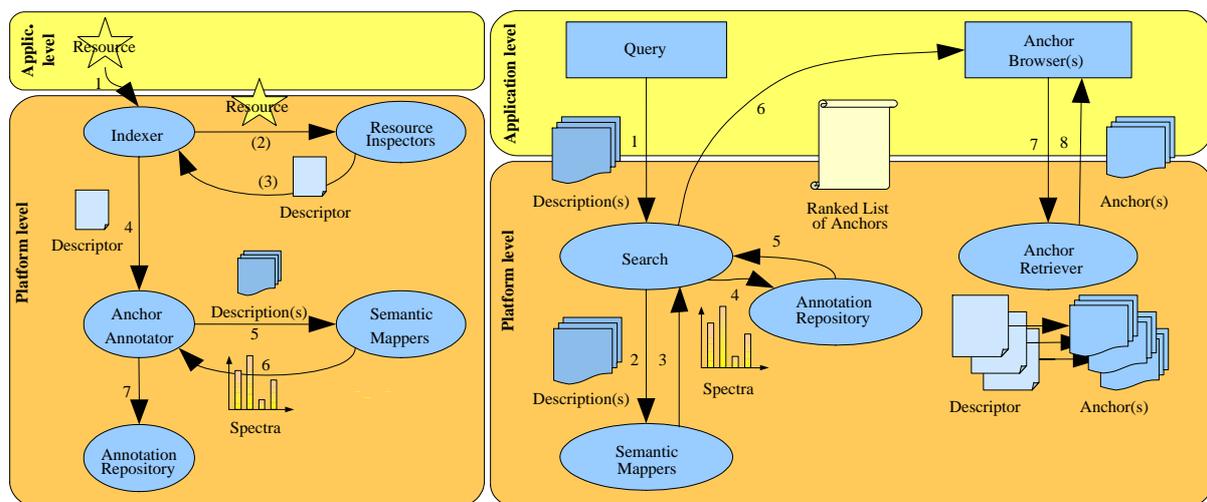

Figure 2: Indexing (left) and Search (right)

reasoning technique, and so on.

In the end, the semantic annotations (spectrum) for each anchor are stored in the annotation repository for further retrieval (step 7). Particularly, a semantic annotation for an anchor is composed by the URL of the container descriptor, the unique anchor ID within such descriptor and the weights that correlate the anchor to each ontology. This last operation concludes the indexing phase.

The search process starts from a user query. In the search phase, as shown in Figure 2 on the right, the user query is converted into a logical expression of descriptions (step 1 in the figure) similar to those normally included in the anchors. In this way, it is possible to support different forms of query, not necessarily textual. For instance, one might exploit a specific description type to represent a user's sketch, another one to represent audio features extracted from a just recorded humming, and so on. These descriptions, representing the query, are then mapped to the ontology concepts by the platform search engine (steps 2 and 3) through query oriented mapping strategies (which may actually be the same used in the indexing phase). In addition, thanks to the relation between concepts, it is possible to infer correlated concepts so that a wider range of related resources can be retrieved.

Once a definitive spectrum has been computed by merging all the description spectra, a spectrum-based distance function is utilized to rank the annotated anchors (steps 4 and 5). The list of anchors (which actually corresponds to a list of descriptor URLs including anchor IDs), is then returned to the web application (step 6). At this point the textual links to the anchors can already be displayed as they are, leaving to the user the burden of retrieving the anchor and the related resources. As an alternative, the platform may be requested to retrieve the anchors from the container descriptors and the resources

are then obtained by exploiting the information included in each target element. In this way a more practical preview can be proposed to the user, even as a combination of multiple multimedia channels (steps 7 and 8). In effect, at the application level, in addition to the existing textual visualization of search results, the new architecture also allows the automatic composition of clips or snapshots from the retrieved result. So, eventually, the composition of the fragments retrieved through the anchor targets, especially when heterogeneous, can be arranged according to the context and to the user preferences. For instance, a query for images could either return a grid of thumbnails or a slideshow, whereas textual documents may be shortly described in a list, possibly indicating the most relevant fragments. An appropriate form or interface will permit the user to personalize his/her semantic-based multimedia fruition environment so as to improve comfort and satisfaction.

4 Architecture

In this section, the logical deployment of the new platform is discussed with respect to the type of offered services. The most relevant component of the platform are identified and described and the tasks performed by each service are explained in the context of typical usage scenarios.

The entire architecture can be logically divided in three main functional levels. Each level includes a set of web services (based on the standard SOAP) or modules, from the most user oriented (indexing, and search) to the most platform specific ones (database management, etc.).

The topmost layer exposes the main services offering indexing and search functionalities to web application developers wishing to include semantics into their works. The interfaces for such services are inherited from the H-
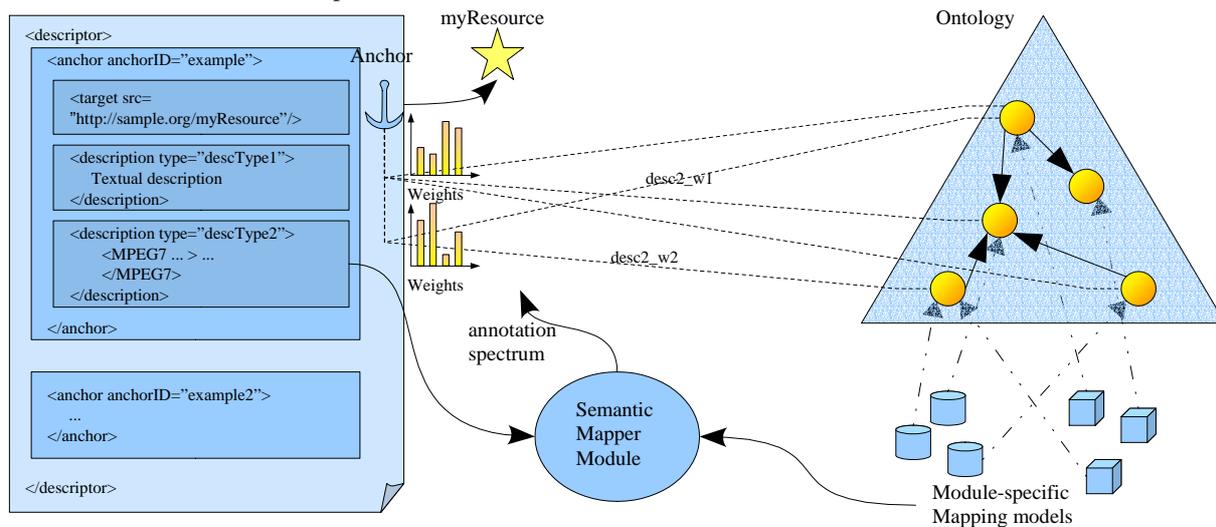

Figure 3: Creation of annotations through a Semantic Mapper Module

DOSE platform to guarantee the maximum possible compatibility with the older architecture, so facilitating the migration to the one proposed here. However, additional functionalities are provided to the user, especially for what concerns the management of non-textual resources.

The kernel layer includes the modules and services that actually implement most of the techniques for semantic classification, search and retrieval. General interfaces are also provided to ensure the necessary functionalities in all the extended modules. So, for instance, every Semantic Mapper implementation must receive descriptions and return a spectrum.

The last layer hides the management of complex data to the higher levels. So, data stored and retrieved through storage devices (databases, files, etc.) are wrapped around simpler functions or services.

In the next sections, the most significant components of the platform are described in more details. Subsequently module extension issues are discussed.

4.1. Indexing

The indexing service receives requests for the classification of a given resource. Its primary task is to obtain a descriptor whose anchors are to be annotated. If the resource is not already a descriptor, it is passed to any specific Resource Inspector module that is possibly capable of automatically or semi-automatically generating a valid descriptor. If no descriptor can be obtained, the indexing is interrupted for the requested resource and the error is logged for further debug.

4.2. Resource Inspectors

Each Resource Inspector is an implementation of a simple predefined interface, which defines the functionalities for this type of modules. Two main tasks should be provided by any implementation, given a resource: to decide whether it can process it; to generate a descriptor (if no error occurs).

The first task is not strictly necessary, but is intended to facilitate and quicken the selection of the appropriate implementation for a resource.

The second task is actually responsible for most of the work. Basically, a resource inspector is fed with the URI of a resource (that should be the resource to be indexed) and returns a valid descriptor containing a series of anchors describing whichever parts of the resource that may be relevant (or even the whole resource).

The module may exploit external libraries, such as filters, to extract features and fill appropriate descriptions. This type of modules is in effect expected to become rather complex, especially in case of multimedia resources, where, for example, one might wish to automatically process a video stream, its audio or even its

captions to split the entire sequence in smaller clips, each identified by an anchor (the target element may be used to specify the desired restriction). Indeed, similar scenarios require the integration of state-of-the-art techniques from numerous research areas, just not to cite the relatively simpler issue of understanding the various resource formats.

Easier is the case of retrieving special metadata from the resource or from linked resources that have been provided during the authoring phase, i.e., directly by the creator of the resource. These sources of information should actually represent a rather reliable description of the resource, even if they are often incomplete or inaccurate, especially when inadequate tools are available in the authoring phase.

In any case, the information captured from a given resource (perspective) should be appropriately organized in anchor descriptions with the precise purpose of providing relevant hints about how to classify the anchor with respect to the domain ontology and to the mapping modules that will be used for such descriptions.

One may of course also decide not to depend on this "descriptor factory" by creating the descriptor(s) through some other external tools, and then feeding the descriptions directly to the indexing service in place of the resources.

4.3. Semantic Mappers

The mapping modules (or Semantic Mappers, SM) are used to associate anchor descriptions to one or more ontology concepts, according to a predefined interface. Specifically, the information stored in a description is used by a SM to determine how strongly the description is associated to each ontology concept. These weighted associations, also known as semantic annotations, are represented as a conceptual spectrum (a sort of histogram which indicates how much each concept is correlated to a given object) and classify an anchor description according to the semantics of its content.

An advantage of using spectra for managing sets of annotations is the possibility of combining different spectra into a single spectrum. So, the weights corresponding to the same concept can be simply averaged among the given spectra.

Each description element may contain metadata in the desired format and can be exploited by all the SMs that provide support for it. Consequently, more than one spectrum can be obtained for each description.

The techniques used to compute a spectrum given a set of descriptions may be various: each implementation may actually employ the most diverse methodologies and external aids. For instance, in the case of textual descriptions a tf/idf strategy may be utilized in conjunction with multilingual synsets linked to the

ontology concepts, as explained in [3]. Instead, for descriptions containing visual features, case-based reasoning techniques might be useful to determine the most appropriate topics. In effect, the modularity of the platform presented in this paper permits to explore and integrate innovative approaches with relatively little effort.

4.4. Search Engine

The search service receives queries as descriptions and returns a ranked list of anchors, each identified by the container descriptor URL followed by the anchor XPath or ID. In more details, the entire process proceeds as follows. A web application designer proposes to the user one or more input fields and interfaces that collect the user query information and convert it in one or more descriptions, depending on the type and complexity of the query components. In effect, the descriptions can be composed in a logical expression that is then passed to the search service. It exploits the appropriate Semantic Mappers to convert the descriptions in spectra, which are then used to retrieve the most relevant anchors through the Annotation Repository. Eventually, the anchors are ranked also taking into account the logical expression.

4.5. Anchor Retriever

Once a search has been performed, the anchor retriever service can be used to extract the requested anchors from the respective descriptors, so that the web application designer can actually access the targeted resources and compose a response for the user. Precisely, the target element extracted from an anchor specifies the exact part of the resource that should be shown to the user. The web application designer might however summarize it in different ways according to the resource type.

4.6. Extensibility directions

An insight on the components that are directly involved in the platform extensibility is here presented. Depending on the necessity, it is possible to add functionalities even adding a new single implementation for just one module. The reuse of the existing modules is in fact possible to support articulated scenarios.

A new description type is generally necessary when a new metadata schema is to be used. In this case it may be sufficient to implement an appropriate Description Handler module that wraps the data into an internal format, which is exploited by a Semantic Mapper for classification. In this way multiple schemas can be transparently managed by various Semantic Mapper modules.

If the platform does not yet provide Semantic Mapper modules capable of handling the new description types, an ad-hoc implementation of a Semantic Mapper is probably necessary, in order to efficaciously exploit all the information for the creation of annotations towards the ontology concepts. Also, a distinction can be made between query related descriptions and resource oriented descriptions, so that it is possible to appropriately enhance simple user's queries in the former case, for instance by inferring correlated concepts through the domain ontology.

Additionally, different target types can be specified in order to support the most various resource formats. In fact, one may desire to consider only a well defined segment of a resource, be it spatial, temporal or both. In these cases, a new Target Handler implementation can manage, other than the predefined tags, even the most complicated structures. For instance, it is possible to support the MPEG-7 standard to specify spatiotemporal segments of video resources, which could then prove to be very useful to an anchor browser on the application side. Similarly it would even be possible to specify resources that have no real URL, as in the case of real world objects, e.g., a book on a particular shelf. Depending on the implementation, one may decide to either provide methods to retrieve the actual (part of) resource or to leave this burden to the calling program.

Eventually, a new implementation of a Resource Inspector can be useful to automatically generate a descriptor given a resource. In this case the programmer should be aware of all of the above modules, in order to meet all the schema specifications for the content of the target and description elements. As in the case of the Semantic Mappers, external filters or programming libraries may facilitate the integration and reuse of state-of-the-art techniques.

5. Test case

The platform has been tested using videos from the Open Video (OV) Project [17], which collects and makes available a repository of digitized video content for the digital video, multimedia retrieval, digital library, and other research communities. The unavailability of video test-sets classified with a well defined ontology motivated the creation of an ontology suitable for a subset of the OV archive, namely the NASA K-16 Science Education Programs Special Collection. At the time the collections were analyzed, this was the biggest subset spanning on a reasonably restricted domain (Table 1). The textual descriptions associated to each video have been automatically extracted to create one anchor for each video, for a total of 555 anchors. The most frequent words have then been exploited to manually generate the ontology and the extra information (i.e., sysnsets) necessary to map text to concepts with a tf/idf-based Semantic Mapper. The ontology contains 169 concepts, mostly about meteorology. In these preliminary tests, the

platform classified the anchors producing 1685 annotations to 105 out of the 169 ontology concepts. Actually, some anchors could not be annotated because their descriptions were either empty or contained words which have not been considered for concept mapping. In effect, the ontology covers a somewhat heterogeneous domain, yet it still lacks a few branches, uncorrelated with meteorology, which would cover the anchors lacking of annotations.

Table 1 Videos in the Open Video Project collection (2005/06/12)

Collection subset	Videos
Internet Moving Images Archive	1121
NASA K-16 Science Education Programs	*555*
The Informedia Project at Carnegie Mellon University	321
CHI Video Retrospective	121
University of Maryland HCIL Open House Video Reports	52
Digital Himalaya Project	34
2001 TREC Video Retrieval Test Collection	26

Search results are show in general good recall due to the relations defined in the ontology, but are of course strongly biased, as the ontology has been created upon the dataset. They are therefore not reported here. Ongoing work is focused on improving the ontology so that it can be shared for other researchers and on integrating feature-based technologies for the enhancement of video classification and retrieval.

6. Conclusions

In this paper we have proposed a modular implementation of a semantic platform that can be exploited for any resource type. The idea of anchor is here introduced as information unit. Each anchor may be used to represent a particular aspect (or perspective, view) of a given resource through any desired set of metadata, without requiring burdensome metadata conversions or adaptations. Each particular resource perspective is defined within an anchor through a customizable target element which allows to seamlessly reuse multiple digital archives as they are, independently on their format or location. Automatic or semi-automatic approaches are supported to generate anchors given a resource and multiple mapping strategies can be exploited to associate anchors to ontology concepts. The resulting architecture can therefore offer semantically enhanced fruition of existing memories and resource repositories through unified semantic indexing, search and retrieval services. Eventually, the entire platform uses widely adopted web technologies such as web services and standard formats to expose a minimal complexity to a web application designer whishing to propose these semantic enabled services to the end users.

Preliminary tests have been conducted with a set of videos annotated through textual descriptions and seem encouraging. Further testing is under development and will include the adoption of feature-based techniques for semantic classification and retrieval of images, videos and audio resources. The ontology developed during this work will be publicly released to stimulate further research, as well the platform source code.

7. References

[1] D. Bonino, A. Bosca, F. Corno, L. Farinetti, F. Pescarmona, *H-DOSE: an Holistic Distributed Open Semantic Elaboration Platform*, SWAP2004: 1st Italian Semantic Web Workshop 10th December 2004, Ancona, Italy

[2] D. Bonino, F. Corno, L. Farinetti, *Domain Specific Searches using Conceptual Spectra*, ICTAI 2004 the IEEE International Conference on Tools with Artificial Intelligence,15-17 Nov 2004, Boca Raton, Florida, USA, pp.680-687

[3] D. Bonino, F. Corno, L. Farinetti, A. Ferrato, *Multilingual Semantic Elaboration in the DOSE platform*, SAC 2004, ACM Symposium on Applied Computing, March 14-17, 2004, Nicosia, Cyprus

[4] C. Lagoze, J.Hunter, *The ABC Ontology and Model*, (Version3), Journal of Digital Information, Special Issue - selected papers from Dublin Core 2001 Conference

[5] Dublin Core Metadata Initiative, http://purl.org/dc/

[6] V. Mezaris, H. Doulaverakis, S. Herrmann, B. Lehane,N. O'Connor, I. Kompatsiaris, M. G. Strintzis, *The SCHEMA Reference System: An Extensible Modular System for Content-Based Information Retrieval*, Proc. Workshop on Image Analysis for Multimedia Interactive Services (WIAMIS), April 2005, Montreux, Switzerland

[7] Tansley, R., Bass, M. and Smith, M., *DSpace as an Open Archival Information System: Current Status and Future Directions*, Lecture Notes in Computer Science: Research and Advanced Technology for Digital Libraries LNCS 2769. pp. 446-460, 2003

[8] Sam X. Sun, *Establishing Persistent Identity Using the Handle System*, Tenth International World Wide Web Conference, Hong Kong, May 2001

[9] Simile: Semantic Interoperability of Metadata and Information in unLike Environments, http://simile.mit.edu/

[10] Mathias Lux, Jutta Becker and Harald Krottmaier, *Caliph & Emir: Semantic Annotation and Retrieval in Personal Digital Photo Libraries*, Proceedings of CAiSE '03 Forum at 15th Conference on Advanced Information Systems Engineering, p. 85-89, June 16th-20th 2003, Velden, Austria

[11] Moodle, http://moodle.org/

[12] Muffin, http://muffin.doit.org/

[13] Moving Picture Expert Group Standards, http://www.chiariglione.org/mpeg/

[14] MARC Standards, Library of Congress Network Development and MARC Standards Office, http://lcweb.loc.gov/marc/marc.html

[15] CIDOC Conceptual Reference Model (CRM), http://cidoc.ics.forth.gr/index.html

[16] Open Archives Initiative Protocol for Metadata Harvesting (OAI-PMH), http://www.openarchives.org/

[17] The Open Video Project, http://www.open-video.org/