COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

COCHIN – 682022

2010

Seminar Report

On

Semantic Digital Library

Submitted By

Varghese S. Chooralil

In partial fulfillment of the requirement for the award of

Degree of Master of Technology (M.Tech)

In

Software Engineering

DEPARTMENT OF COMPUTER SCIENCE

COCHIN UNIVERSITY OF SCIENCE AND TECHNOLOGY

COCHIN – 682022

Certificate

This is to certify that the Seminar report entitled " Semantic Digital Library ", submitted by Varghese S Semester I, in the partial fulfillment of the requirement for the award of M.Tech. Degree in Software Engineering is a bonafide record of the Seminar presented by him in the academic year 2011.

Dr. Sumam Mary Idicula	*Dr. K Paulose Jacob*
Seminar Guide	*Head of the Department*

ACKNOWLEDGEMENT

I express our profound gratitude to the Head of Department Dr. K Paulose Jacob for allowing me to proceed with the seminar and also for giving me full freedom to access the lab facilities.

My heartfelt thanks to my guide Dr. Sumam Mary Idicula for taking time and helping me through my seminar. She has been a constant source of encouragement without which the seminar might not have been completed on time. I am very grateful for her guidance.

I am also thankful to Mr. Santhosh Kumar, Lecturer, Department of Computer Science, for helping me with my seminar. His ideas and thoughts have been of great importance.

ABSTRACT

Almost every type of information can be represented in digital form, including text, pictures, musical works, computer programs, databases, models and designs, video programs, and compound works combining many types of information. Here I am covering the working of digital library along with its limitations. Its limitations are overcome by Semantic Digital library.

I propose a service-oriented architecture that explicitly includes a semantic layer which provides primitive services to the applications built on top of the digital library. As part of this layer, a specific component is described: the PIRATES framework. This module assists end users to complete several tasks concerning the retrieval of the most relevant content with respect to a description of their information needs

Keywords- Semantic, digital library, repositoary, semantic web,archive

Contents

1. INTRODUCTION

Improvements in digitization have led, in the last decades, to a huge evolution in the way digital libraries and archives are conceived, designed, and used. Both the transition of library materials from traditional to digital formats and the large (and continuously growing) availability of digital content pose new challenges. More sophisticated software tools are needed to meet the expectations of users, which are often high due to the classical information overload problem. Searching everything everywhere is becoming a habit also in digital libraries, but finding exactly what it is needed remains a very hard job . Data interoperability and sharing is another issue that must be faced when developing tools concerning digital library: often, contents and archives should be shared across different platforms and applications, usually by means of a Web-based infrastructure.

we propose a service-oriented architecture that explicitly includes a semantic layer which provides primitive services to the applications built on top of the digital library. As part of this layer, a specific component is described: the PIRATES framework. This module assists end users to complete several tasks concerning the retrieval of the most relevant content with respect to a description of their information needs (a search query, a user profile, etc.). Techniques of user modeling, adaptive personalization, and knowledge representation are exploited to build the PIRATES services in order to fill the gap existing between traditional and semantic digital libraries.

2. Overview of the Digital Library System

2.1 The structure of information and sets of digital objects

This report gives an overview of the concepts as background to the more detailed explanation and the technical information. The purpose of the information architecture is to represent the riches and variety of library information, using the building blocks of the digital library system. From a computing view, the digital library is built up from simple components, notably digital objects. A digital object is a way of structuring information in digital form, some of which may be metadata, and includes a unique identifier, called a handle. However, the information in the digital library is far from simple. A single work may have many parts, a complex internal structure, and one or more arbitrary relationships to other works. To represent the complexity of information in the digital library, several digital objects may be grouped together. This is called a set of digital objects. All digital objects have the same basic form, but the structure of a set of digital objects depends upon the information it represents.

The different types of material in a digital library, information can be divided into categories, e.g.: text with SGML mark-up, World Wide Web objects, computer programs, or digitized radio programs. Within each category, rules and conventions describe how to organize the information as sets of digital objects. For example, specific rules will describe how to represent a digitized radio program. For each category, the rules describe the digital objects that are used to represent material in the library, how each is represented, how they are grouped as a set of digital objects, the internal structure of each digital object, the associated metadata, and the conventions for naming the digital objects. A user interface that is aware of the rules and conventions applying to certain categories of information is able to interpret the structure of the set of digital objects. Complex information can be presented without the user having any knowledge of the complexity. Since the user interface recognizes how material is represented, it can provide unsophisticated users with Digital library objects**.**

In the digital library, information is stored as "digital objects". A primitive idea of a digital object is that it is just a set of bits, but this idea is too simple. The content of even the most basic digital object has some structure, and information, such as intellectual property rights, must be associated with the digital object. Figure 2 shows that a digital object in a repository has two parts, content and associated data, sometimes called "metadata".

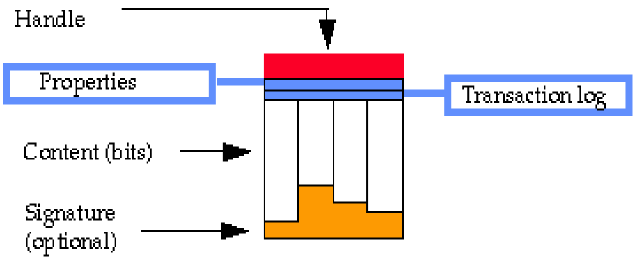

Figure 1. Parts of a digital object

To enable the content to represent useful information, its type must be known. Thus part of the content may be of type text (perhaps encoded in a mark-up language), while another part may be of type audio. A single digital object may contain many types of content. It turns out that arbitrarily complex data types can be constructed from a few basic types, notably bit-sequences, handles and other digital objects. By combining these in various combinations, any digital content can be represented.

To manage valuable intellectual property, certain metadata is required. This is shown in the figure. It always includes a unique identifier (the handle). It may also include properties such as rights and access methods. One property states whether a digital object is mutable, in that it may be altered after being placed in a repository. Another is a digital signature or other method of validating that an object has not been changed. Frequently, it is useful to keep a log of all transactions associated with each digital object.

2.2 Components of the Digital Library System

The digital library framework permits many different computer systems to coexist. The key components are shown in the figure below. They run on a variety of computer systems connected by a computer network, such as the Internet.

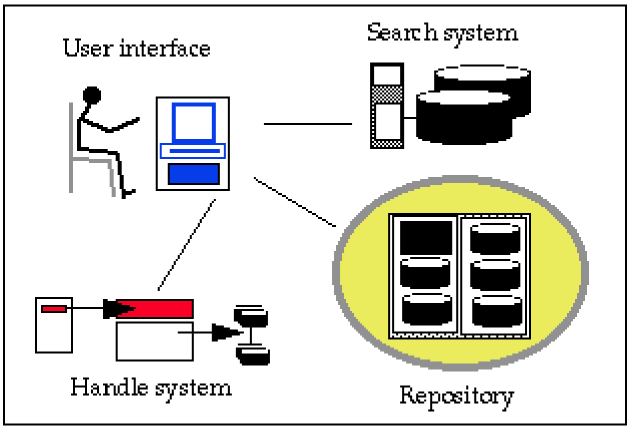

Figure 2. Major system components

User interfaces

Both the pilot and the prototype have two user interfaces: one for the users of the library, the other for the librarians and system administrators who manage the collections. Each user interface is in two parts. A standard Internet browser is used for the actual interactions with the user. This can be Netscape Navigator, Microsoft's Internet Explorer, or the Grail browser developed by our colleagues at CNRI. The browser connects to client services, which provide intermediary functions between the browser and the other parts of the system. The client services allow the user to decide where to search and what to retrieve; they interpret information structured as digital objects; they negotiate terms and conditions, manage relationships between digital objects, remember the state of the interaction, and convert among the protocols used by the various parts of the system.

Repository

Repositories store and manage digital objects and other information. A large digital library may have many repositories of various types, including modern repositories, legacy databases, and Web servers. Section 4 of this report describes the pilot repository that we have implemented and enhancements planned for the prototype. The interface to this repository is called the repository access protocol (RAP). Features of RAP are explicit recognition of rights and permissions that need to be satisfied before a client can access a digital object, support for a very general range of disseminations of digital objects, and an open architecture with well defined interfaces. Repositories must look after the information they hold A repository stores digital objects, both the content and the metadata. A digital object as stored in a repository may be very different from the digital object that is made available to users' computers. Different repositories will have very different internal organizations, but for each digital object every repository will have a properties record, which holds attributes of the object, and a transaction log. Since digital objects contain valuable intellectual property, the stored form of a digital object within the repository includes information that allows for it to be managed within economic and social frameworks. The repository maintains this information, provides basic reference information, and provides security to ensure that only valid operations are carried out on the digital objects.

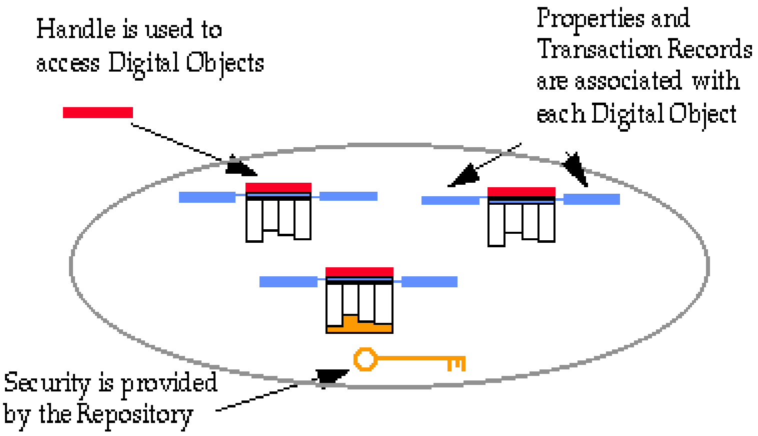
Handle is used to access Digital Objects
Properties and Transaction Records are associated with each Digital Object
Security is provided by the Repository

Figure 3. A repository

The internal organization of a repository and the way that digital objects are stored are hidden from the user. A simple protocol is provided for interactions with the repository. This protocol is called the "repository access protocol." The basic commands in this protocol are those to access a digital object and its metadata, and the service request to disseminate a digital object. In addition there are commands to add and delete digital objects.

Handle system

Handles are general purpose identifiers that can be used to identify Internet resources, such as digital objects, over long periods of time and to manage materials stored in any repository or database. CNRI's handle system is a computer system that provides a distributed directory service for identifiers (handles) for Internet resources. When used with the repository, the handle system receives as input a handle for a digital object and returns the identifier of the repository where the object is stored.

3. An example of how the Digital Library support a user's query

To understand the function of these system components, here is an example of how they allow a user to carry out a simple query. Suppose that a user is looking for a digitized photograph showing both President Calvin Coolidge and President Herbert Hoover. The interaction could pass through the following stages.

The first stage is to search for digitized photographs that fit the required criteria. The client services provide the user's browser with a form for searching. The user fills in the form with a search query, asking for photographs of Coolidge and Hoover. The completed form is sent to the client services. The client services translate the query into the formats and protocols required by the search system. For example, the search system may use Z39.50. The client services conduct a Z39.50 session with the search system and obtain a list of the digital objects that satisfy the query. Each digital object is identified by its handle.

The next stage is for the user to select a digitized photograph to view. The client services present the user's browser with the list of digital objects found through the search system (currently as an html page with links to click). The user selects the required photograph.

The third stage is retrieval of the digitized photograph. The client services send the handle of the chosen photograph to the handle system, which returns the address of the repository. The client services pass the handle to the repository, using the RAP protocol. Several versions of the photograph may be stored in the repository as a set of digital objects, identified by the handle. The client services select one, perhaps a small thumbnail, and requests it from the repository. All RAP transactions pass through an explicit terms and conditions step. Checking the terms and conditions associated with this digital object may need negotiation between the client services and the repository, or direct interaction with the user.

Finally, the digitized photograph that was chosen is delivered from the repository, via the client services, to the user's browser and displayed on the screen.

4. The Information Architecture

4.1 Outline of the Information Architecture

4.1.1 The structure of information in a digital library

Interactions, such as the query described above, require that information in a digital library be organized effectively. Within the library, information is stored as basic units of digital information, e.g., a digitized map, a section of text, a Web page, a scanned photograph, etc. In digital form, each basic unit is a sequence of bits, but users often want to refer to material at a higher level of abstraction than the individual item. Common English terms, such as a "report", a "computer program", or an "opera" can refer to many items that are variants of each other. They may have different formats, minor differences of content, different usage restrictions, and so on, but for some purposes users are willing to consider them as equivalent.

The issues to be addressed in structuring information include the following.

- Digital materials are frequently related to other materials by relationships such as part/whole, sequence, etc. For example, a digitized text may consist of pages, chapters, front matter, an index, illustrations, and so on. In the World Wide Web, a typical item may include several pages of text, with embedded images, and links to other information. A single computer program is assembled from many files, both source and binary, with complex rules of inclusion. Materials belong to collections. These may be collections in the traditional, custodial sense; they may be the on-line groupings provided by a publisher; or they may be the pages maintained by a Webmaster.
- The same item may be stored in several digital formats. Sometimes, these formats are exactly equivalent and it is possible to convert from one to the other (e.g., an uncompressed image and the same image stored with a loss-less compression). At other times, the different formats contain different information (e.g., differing representations of a page of text in SGML and PostScript formats).
- Because digital objects are easy to change, different versions are created continually. (Some organizations change their Web home page several times per month.) Versions may differ by a single bit or may be very different. When existing material is converted to digital form, the same physical item may be converted several times. For example, a scanned photograph may have a high resolution archival version, a medium quality version, and a thumbnail.
- Each element of digital information may have different rights and permissions associated with it.
- The manner in which the user wishes to access material may depend upon the characteristics of computer systems and networks, and the size of the material. For example, a user connected to the digital library over a high speed network may have a different pattern of work from the same user when using a dial-up line.

The information architecture described here provides a general approach to organizing the material within the digital library in such a manner that computer programs can understand the structure of the material and carry out the interactions that the user wishes.

4.1.2 Basic principles

The information architecture is motivated by the following basic principles:

- Users and their applications programs must be given flexibility. Since users explore material in almost every conceivable manner, the organization of information should not be biased by expectations about how users will approach the material, their level of expertise, or the sequence in which items will be accessed.
- Collections must be straightforward to manage. In digital libraries, as in all libraries, comparatively small professional staffs manage very large collections of material. The architecture must allow the staff to concentrate on curatorial aspects, and free them from routine tasks wherever possible.
- The information architecture must reflect the economic, social, and legal frameworks developing in the information infrastructure. In particular it must recognize that information is valuable, subject to terms and conditions, and is transmitted over insecure networks that cross national boundaries. These considerations are a driving force behind the technical framework which underlies the architecture.

4.2 Data types, structural metadata, and meta-objects

The information architecture is based on three simple concepts: data types, structural metadata, and meta-objects. A data type describes technical properties of data, such as format, or method of processing. Structural metadata is metadata that describes the types, versions, relationships and other characteristics of digital materials. A meta-object is an object that provides references to a set of digital objects. In its simplest form, a meta-object is a list of handles of other digital objects. For example, a poetry anthology might be represented by one digital object per poem. A meta-object for the anthology is a digital object that lists all the poems. An important example of a meta-object is a digital object that lists all converted versions of a specific physical item.

As part of the pilot system, with colleagues at the Library of Congress, we developed specifications of structural metadata and meta-objects for two categories of material, scanned photographs and digitized texts. For the prototype we plan to extend these specifications to other categories of material.

In developing these rules for each category of material, certain guidelines were applied to all categories.

1. **All data is given an explicit data type**

   Each item of data has an associated data type. The type specifies that the data has a certain format (e.g., the data is in the JPEG format), should be processed in a specific way (e.g., a computer program is written in the C programming language), or has a specific organization (e.g, a section of text has been marked up with SGML tags).

2. **All metadata is encoded explicitly**

   All metadata that is needed to manage the collection or to provide access is coded explicitly. In particular, no semantic information is included in any name that is not encoded separately as metadata. (This can be contrasted with computer file systems, where semantic information is often embedded in file names, such as ".txt" indicating a text file.)

3. **Handles are given to individual items of intellectual property**

   Whenever an item of information might be used on its own, it is given its own handle and made into a separate digital object. By having its own handle, an item may be accessed independently. This provides maximum long-term control and flexibility. For example, if a digitized text contains illustrations that could potentially be used independently, each illustration is made into a separate digital object with its own handle.

4. **Meta-objects are used to aggregate digital objects**

   In a digital library, the full metadata about a single piece of information may exist in several places within a repository and also in external catalogs, indexes, or finding aids. Maintaining links to all the metadata is a huge task, and therefore the architecture does not require them. Much is gained from having a meta-object for each item that provides links to all versions of the item and to all structural metadata. External bibliographic records can then refer to the meta-object and not need to know details of a set of digital objects.

5. **Handles are used to identify items listed in meta-objects**

   A meta-object contains a list. We use handles to identify the items of these lists. This provides a robust, flexible structure that allows subsequent reorganization of the collection with minimal effort.

The interpretation of these rules is often a matter of judgment, with a trade-off between a powerful representation of information, which is flexible in use but laborious to manage, and a simpler representation. Ultimately such decisions can not be dictated by the architecture or the system designers. They must be made by the curators who are knowledgeable about the material and responsible for managing it. The system provides straightforward methods for curators to decide how best to manage collections.

5. An Example of the Use of Meta-objects

5.1 Scanned photographs in Digital Library

Scanned photographs are a simple category of material that illustrates the general principles of how to use meta-objects. In the National Digital Library Program, most of the photographs to be scanned are single items, but there are numerous interesting cases to consider, including sets of photographs, and large photographs and posters that are scanned in sections.

With colleagues from the Library of Congress, we have developed guidelines for representing each scanned photograph as a set of digital objects linked through a meta-object.

5.2 Digital objects for a scanned photograph

When a typical photograph is scanned, three or more versions are produced. In NDLP terminology, they are called a low resolution "thumbnail", an intermediate resolution "access" image, and a high resolution "reference" image. Separate digital objects are created for each individual version. They each contain metadata specific to the version and the data bits for the image. To describe the photograph and its digitized versions, a meta-object is created. It contains metadata that is common to all versions of the photograph and handles for the three separate versions. Thus the scanned photograph is represented by a set of four digital objects.

5.3 Digital objects for individual versions

The digital object for each individual version of a scanned photograph has the following information:

- **Key metadata.** Key metadata is metadata contained in the digital object that is used to manage the object in a networked environment. It includes the handle, and the rights and permissions associated with the digital object.
- **Structural metadata.** This includes other metadata associated with the specific version. It includes fields for description, owner, handle of meta-object, data size, data type (e.g., "jpg"), version number, description, date deposited, use (e.g., "thumbnail"), and the date of last revision.
- **Image data.** This is the image data.

Meta-object

The digital object for the meta-object has the following information:

- **Key metadata.** The key metadata is metadata contained in the digital object that is used to manage the object in a networked environment. It includes the handle, and the rights and permissions associated with the digital object.
- **Structural metadata.** This is metadata that applies to the original photograph and to all the versions. It includes a description, the owner, the number of versions, the date deposited, the use ("meta-object"), and the date of last

revision. If bibliographic information were to be included, it would be added to this part of the meta-object.

- **Data about each version.** For each of the three scanned versions (e.g., the thumbnail), there is a package of information including the handle of the version, and the relationship among the versions.

The usual manner of access to the photograph is to begin with the meta-object and from there to select one of the individual versions. However, to permit a user to go directly to a specific version, some information is duplicated across objects. In particular, the rights and permissions are an integral part of every digital object.

6. Handles for scanned photographs

At a early stage of processing a collection, the NDLP's procedure is to give a control identifier to each item that is digitized, converted, or otherwise prepared for the library. For example, a scanned image of a photograph from the Coolidge Consumerism compilation has the identifier: 3a16116r.jpg.

This control identifier is an example of a semantic name. The form of the identifier conveys information about the item. For example, "r.jpg" indicates an image intended for reference, in the jpeg format. This is convenient for processing, but, for long term identification, semantic names are fraught with danger and violate one of the guidelines given above. Therefore, in the digital library system, we encode such semantic information explicitly as metadata, which is stored in digital objects, and replace the control identifiers by handles, which provide a unique, persistent, location independent name for each item. An example of a handle is:

loc.ndlp.amrlp/3a16116

This particular example is the handle of the meta-object that lists the various versions of the original object. The following terminology is used in describing handles:

"loc.ndlp.amrlp" is the naming authority

"3a16116" is a locally unique string

For convenience in processing, the scanned versions of the same photograph are distinguished by sequence numbers. For example, the two following handles refer to different versions of the same photograph. (For example, the first handle might refer to the reference version, the second to a small thumbnail.)

loc.ndlp.amrlp/3a16116.1
loc.ndlp.amrlp/3a16116.2

Using the string "3a16116" from the control identifier as part of the handle is for mnemonic convenience only. Any string could be used and totally different strings could be used for the separate versions. However, this convention is convenient for managing the collection. The following diagram shows the use of the meta-object:

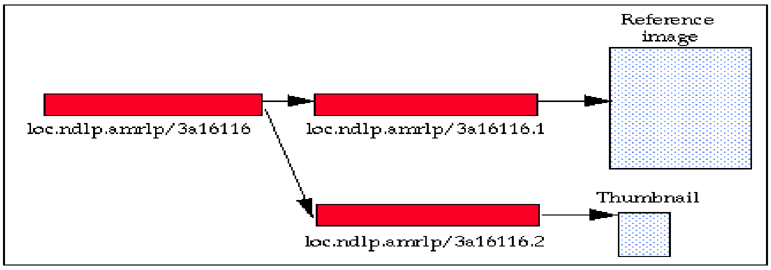

Figure 4 A meta-object used to identify two version of a scanned photograph

The handle to the meta-object, "loc.ndlp.amrlp/3a16116", permanently identifies the set of scanned images made from this single photograph. The scanned photograph can be referenced by this handle, for example, in MARC records, shelf lists, external bibliographies, and any other place where a name is needed that can be relied on for the long term.

7. Depositing a scanned photograph

To deposit a scanned photograph in the repository is partly a professional task carried out by library staff and partly automated. The beginning point is a set of files received from the contractor doing the scanning, each with a control identifier. The following tasks require professional attention:

- Selection of the material that will be made into each digital object.
- Specification of the metadata for those fields that require judgment.

The actual creation and depositing of the set of digital objects in the repository and the registration of handles in the handle system is carried out by a computer program. The following operations are carried out automatically:

- Creation of the meta-object and the links to other digital objects.
- Depositing the digital objects in the repository.
- Registering the handles in the handle system.

Access to a scanned photograph

Deposit of a set of digital objects is one basic operation on the set of digital objects that represent a single scanned photograph. Other basic operations concern access. These are discussed in more detail in the later section on repositories. For the scanned photograph category, the access conventions are:

- Bibliographic entries in search systems refer to the scanned photograph by the handle of the meta- object.
- If a user requests a summary of the photograph, the "thumbnail" image is provided.
- If the user requests access to the photograph without specifying which version, the "access" image is provided.

- . **Semantic Digital Library**

  It propose a service-oriented architecture that explicitly includes a semantic layer which provides primitive services to the applications built on top of the digital library. As part of this layer, a specific component is described: the PIRATES framework. This module assists end users to complete several tasks concerning the retrieval of the most relevant content with respect to a description of their information needs (a search query, a user profile, etc.). Techniques of user modeling, adaptive personalization, and knowledge representation are exploited to build the PIRATES services in order to fill the gap existing between traditional and semantic digital libraries. we are designing and developing a digital platform capable of maintaining the semantic meaning of each digital object and its content, of maintaining its origin and authenticity, and of retaining its interrelatedness.

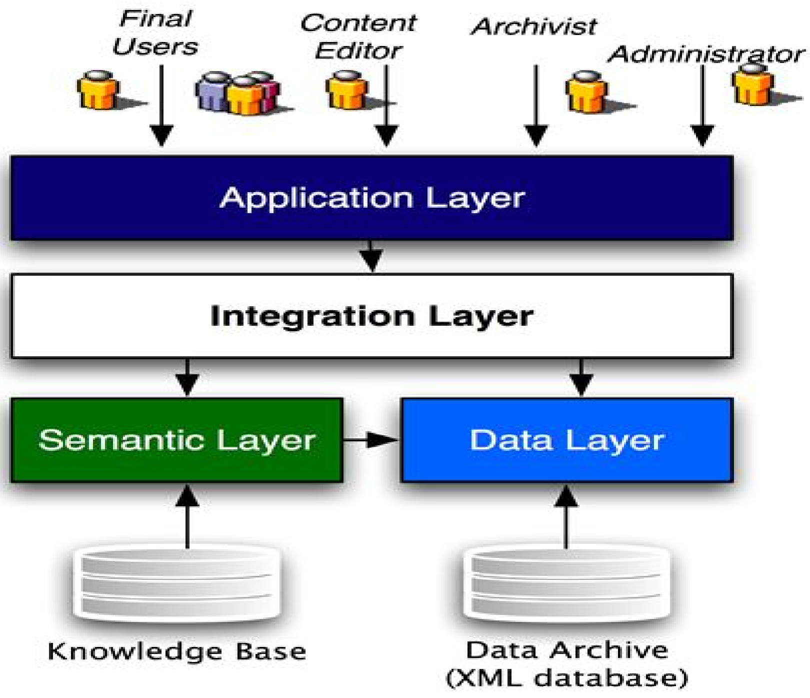

Figure 5 An Architecture of semantic Library

At the bottom of the architecture, digital archives are stored in XML databases by a core layer devoted to data storage and persistence (Data Layer). The access to these archives can be provided directly by the Data Layer services, or by a Semantic Layer which exploits a Knowledge Base in order to find (suggest) the most relevant content fulfilling the user needs (e.g., a user query submitted from a search engine). The Integration Layer forms the “architectural glue” that brings the digital library beyond the scope of a single application, unifying the interfaces of different subsystems into the same interoperable environment. The standard set of Web service technologies (XML, SOAP, WSDL) provides the means to describe, locate, and invoke a Web Service. The Integration layer uses a component called Enterprise Service Bus, which implements the orchestration of different application-specific services in order to integrate these applications in the digital library. Finally, at the top of the architecture, the Application Layer hosts the programs used by the digital library users (archivists, content editors, final users, etc.) in order to edit, publish and access the contents stored in the archives.

8. Towards Semantic

This layer exposes its services to the user applications through the Enterprise Service Bus located in the Integration layer. Two main components characterize the Semantic layer:

- PIRATES Framework, which communicates with a Knowledge Base in order to retrieve or suggest potentially relevant information from the archives. This framework provides primitive services to automatically classify, annotate and recommend specific content using techniques based on natural language processing. PIRATES is composed of three components, a Cognitive Filtering Tools module, an Automatic Tagger, and a Knowledge Base Builder.
- Meta Search Engine, which exploits the document annotations provided by PIRATES in order to recommend similar contents with respect to those retrieved by a traditional search engine fulfilling user queries. This module can also be used for refining a user query which has not provided enough results (query reformulation).

The presence of the Semantic Layer is aimed at improving the information access mechanism by empowering its environment by semantic services.

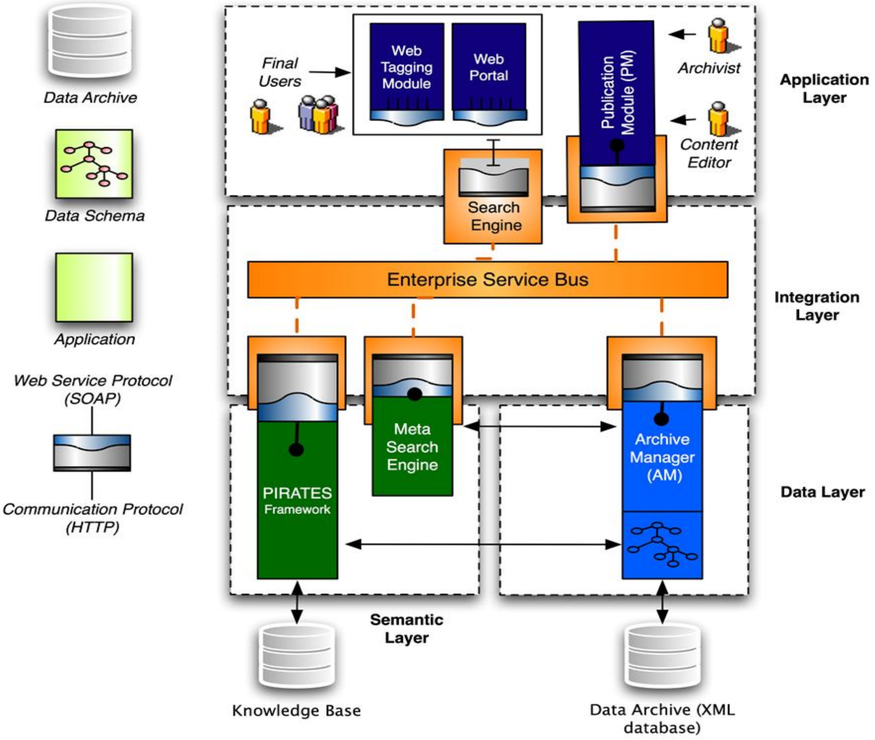

Figure 6 Accessing Digital Library Archives by means of "Semantic Services"

Semantically enabled technologies are expected to bring a number of benefits to the users of digital libraries such as helping people to find relevant information more efficiently, giving better access to that information, and aiding the sharing of knowledge within the user community. Starting from these motivations, in this section we outline an approach for adding semantics to archives using tags suggested by an automatic system (the PIRATES framework) that is based on information extraction techniques. Before introducing PIRATES, we describe the ways an archivist can exploit tagging services to annotate a digital content item. We discuss also some notable limitations inherent to the use of manual tags that lead us to propose an automatic approach to tagging.

8.1 Adding Semantics to Digital Archives: the Tagging Approach

Tagging is a textual annotation technique based on meta-data information (i.e., tags). A tag is a keyword users use to annotate a content, in order to organize knowledge, describe it, correlate it with other contents, or simply to retrieve it easily in future searches. The tagging activity may be manual if it is provided by a human user, or automatic if is it generated by a dedicated software. Archivists can employ tags differently because they can be guided by different tasks. Typically, tagging is used with the explicit intent of:

1. classifying content by means of a corpus of concepts that are familiar to the archivist (e.g., taxonomies, thesauri, or any bag of keywords representing meaningful categories for him/her)
2. summarizing resource content by means of a short list of keywords representing the user-generated content description
3. expressing a polarity judgment about a content by means of proper adjectives provided as tags (e.g., “sad”, “wonderful”)
4. correlating tagged resources with people and their skills such as the level of expertise, the reputation, or the importance of a person mentioned in the resource content (e.g., “guru”, “geek”, “vip”, “bill-gates”, etc.)
5. creating dichotomous classification criteria in order to describe resources as belonging or not belonging to a particular category (e.g., “clinical”/“not-clinical”, “statistical”/“not-statistical’, “accepted”/“rejected”, and so on)
6. providing temporal information to a resource, for example annotating the date of an event related to that resource.

To some extent, all these forms of tagging express a classification intent targeted to establish effective schemata for organizing the knowledge and facilitating content retrieval.

Tagging allows users to determine suitable labels for their resources freely without relying on any predetermined vocabulary or hierarchy . Moreover, tags can be very effective for serendipitous browsing of a digital archive of documents (or bookmarks) in order to find relevant information. Hence people tag the content with their own vocabulary and ultimately their mental models in order to facilitate the process of recall. Besides with these potential benefits, manual tags suffer with some of notable limitations

- Ambiguity: with an uncontrolled vocabulary, many tags can be ambiguous. Indeed in tags we can find the same ambiguity that we find in natural language (e.g., homonymy, polysemy, synonymy, spelling mistakes, disambiguation, words which have more than one common spelling or morphology etc.).
- Undistinguished concerns: social tagging systems do not enforce, or even propose, a schema for distinguishing the purpose of a meta-data value. Tags might be variously,

subject descriptors, genres, self-reminders, tangential remarks (such as colors or years, especially for non-textual information such as pictures) or proper names.

- Independence of terms: social tagging does not provide relationships to connect and relate different terms: each tag is independent of the others and no inference is possible. In other words, the structure of a tag system is “flat”.
- Effort: systematically (and consistently) tagging Web resources is tedious, error prone and rather wearying.

In order to alleviate these limitations, we propose an automated approach that assists the user when (s)he tags a Web resource. A software system analyzes the textual document and provides suggestions/recommendations for new tags by exploiting information extraction tools and ontologies. Using this approach, we try to achieve two different goals:

- use a controlled, ontology-based vocabulary, not necessarily present in the original Web resource, in order to classify it as result of the automatic tagging process; our vocabulary is a structured form of knowledge representation (the ontology) and provides entities (classes), instances and relations (is-a links between entities).
- reduce the manual effort required to tag a Web resource

8.2 The PIRATES Framework: Merging Cognitive Filtering and Semantic Services

This section presents the PIRATES framework: a Personalized Intelligent Recommender and Annotator TEStbed for text-based content retrieval and classification. Using an integrated set of tools, this framework lets the users experiment, customize, and personalize the way they retrieve, filter and organize the large amount of information available on the Web. Furthermore, the PIRATES framework undertakes a novel approach that automates typical manual tasks, such as content annotation and tagging, by means of personalized tags recommendations and other forms of textual annotations (e.g., key-phrases).

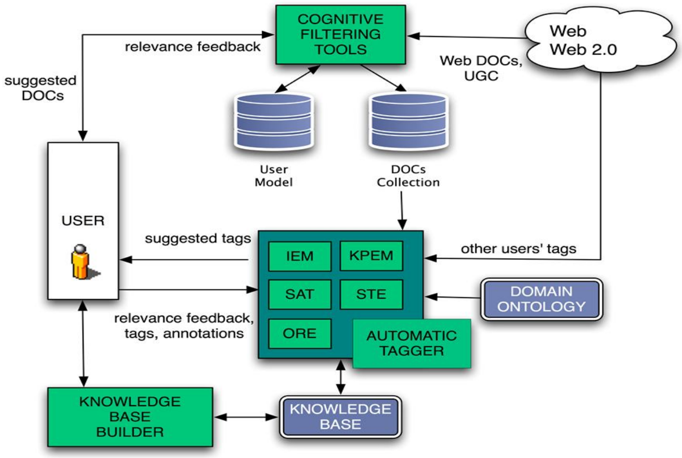

Figure 7: PIRATES main modules.

The PIRATES architecture, shown in Figure 7, is formed by three major components:

- The Cognitive Filtering Tools module implements the IFT (Information Filtering Tool) module. The IFT algorithm is used to build representations of user interests (IFT user models), to provide mechanisms of relevance feedback and to classify the textual content of a document belonging to an incoming stream of documents. The classification process produces evaluations of the relevance (in the sense of topicality) of a document according to a specific model of user interests represented by semantic networks. Semantic networks are built in a supervised way, by integrating a priori knowledge provided by domain experts, expressed as a training set constituted by keywords, short excerpt of textual description, and links to relevant documents. This knowledge is then encoded by the IFT algorithm into a vector of weighted terms and connections, linking terms which co-occur in the training set.
- The Automatic Tagger module implements several modules devoted to automatically annotating an incoming stream of text (the content of a document) by means of tag recommendations: the sub module IEM (Information Extraction Module) suggests entity, names, and dates, KPEM (Key-Phrases Extraction Module) key-phrases, SAT (Sentiment Analysis Tool) polarity judgments, STE (Social Tagger Engine) tags used by a community of Web 2.0 users, while ORE (Ontology Reasoner Engine) extracts tags from an ontology. The user can choose the combination of annotator modules to exploit in order to obtain suggestions for tags.
- The Knowledge Base Builder module organizes documents in a Knowledge Base repository and produces annotated documents.

The PIRATES framework operates on a set of input documents stored in the Information Base (IB) repository and suggests personalized tags and other forms of textual annotations (e.g., key-phrases) in order to classify them. The original documents are then annotated with these tags forming the Knowledge Base (KB) repository.

Our main goal in integrating PIRATES to empower information access, allowing users to find new relevant contents easily and automatically support them when categorizing documents by means of keywords (tags) in a personalized and adaptive way. We have designed PIRATES keeping in mind several applications where it can provide innovative adaptive tools enhancing user capabilities: in e-learning portals for supporting the tutor and teacher activities in monitoring student performance, behavior, and participation; in knowledge management contexts (including scholarly publication repositories and digital libraries for supporting document filtering and classification and for alerting users in a personalized way about new posts or document uploads relevant to their individual interests.

Although the PIRATES framework is still a theoretical model, a prototype version has been already developed, integrating the IFT subsystem and, at the same time, implementing two different KPEM algorithms (respectively domain dependent and domain independent), and a preliminary version of both the IEM module and the ORE module. In particular, the ORE module does inference over a local ontology written in OWL format, using a reasoning mechanism based on is-a relationship between the ontology concepts.

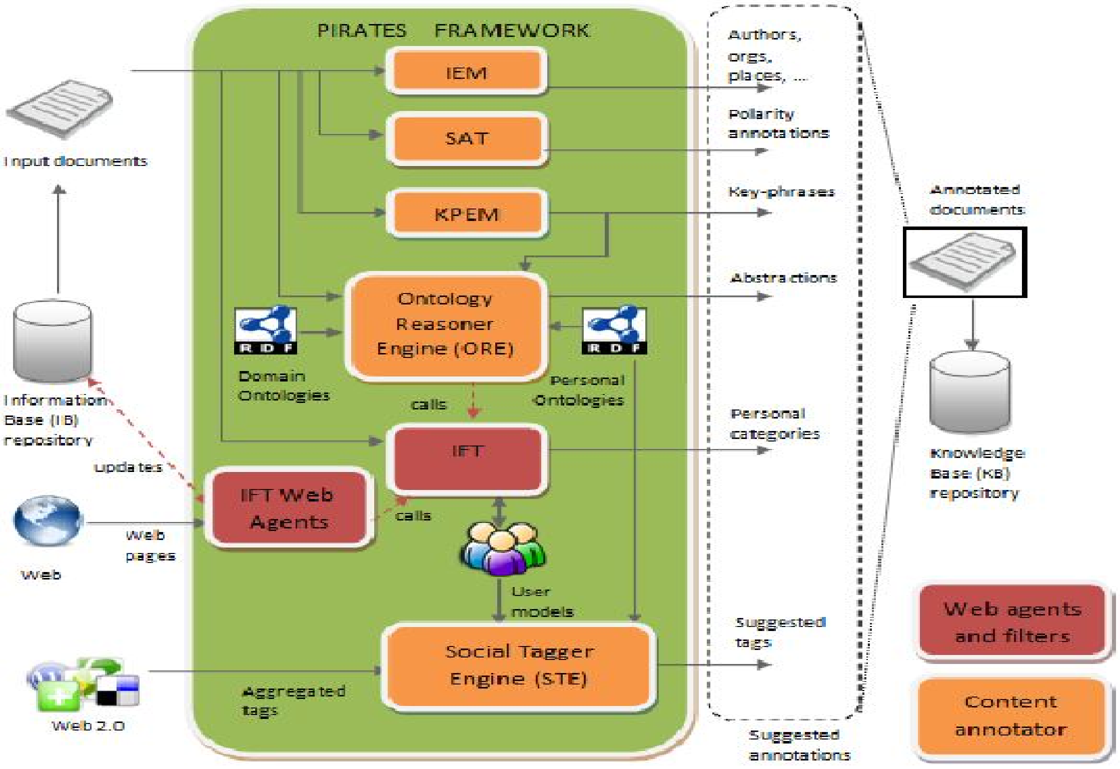

Figure 8 PIRATES Framework

On the left hand side all the possible input sources are shown: Single textual Documents, specific IB repositories which can be contained within an E-Learning or Knowledge management Environment. The righthandside shows the annotations automatically suggested to the user and the resulting KB repository. The main module can be categorized into two categories: Modules that discover new content (Web Agents and Filters),and Modules that extract information from these new contents(Content Annotators).

9. CONCLUSION

It is a proposed architecture to introduce semantic layer on the top of the digital library aimed at better addressing the changes in the final users information needs and improving the effectiveness of the information access.

In this seminar report we have proposed a service-oriented architecture for the digital library that explicitly integrates a semantic layer. The integration of semantic services is aimed at better addressing changes in final users' information needs and improving the effectiveness of information access. To support this new semantic layer, we have designed a framework based on adaptive and personalized services, distinguishing the digital library from an old-fashioned DBMS/structured archive system. Giving access to the semantics of contents helps to realize the vision of a semantic digital library, which is possibly one of the most innovative evolutions in current digital libraries.

10.REFERENCES

Architecture for Information in Digital Libraries, William Y. Arms , Christophe Blanchi ,Edward A. Overly ,Corporation for National Research Initiatives ,Reston, Virginia D-Lib Magazine, February 1997.

Toward Semantic Digital Libraries: Exploiting Web2.0 and Semantic Services in Cultural Heritage, Andrea Baruzzo, Paolo Casoto , Prasad Challapalli, Antonina Dattolo, Nirmala Pudota, Carlo Tasso
Department of Mathematics and Computer Science - University of Udine, Italy,Journal of Digital Information, Vol 10, No 6 (2009).

Semantic Digital Libraries, Sebastian Ryszard Kurk , Adam Westerki and Ewelina Kruk.

Post-proceedings of the Vth Italian Research conference on Digital Libraries,IRCDL,2009