YOUFILE: Ontology based system for document smart indexing

Luca MAINETTI, Roberto PAIANO, Alberto BUCCIERO, Anna Lisa GUIDO, Stefano SANTO SABATO, Luciano CAPONE, Ugo BARCHETTI, Enrico PULIMENO
University of Salento
Via per Monteroni 73100 Lecce
Italy
Tel. +39 0832 297305

ABSTRACT

The huge amount of files of different types (from photos, personal documents, to accounting or fiscal documents) which everyone stores in his own personal computer or in other hardware device (such as internet virtual space, external hard drives, etc.) risk, with the passing of time, to go lost. The founding idea of the eSCI project (eStore of Captured Information) is just to avoid the accidental loss of files, proposing to develop a web portal which allows the user to safely store his own documents and to retrieve them whenever he wants through a simple internet connection. Within the scope of this research project, the reached objective was double: first to assist the user in the document indexing activity, leading him to define the right keywords able to univocally characterize each file; in this way the user doesn’t index the document only though its file name but also providing more precise information about its content. This allows to characterize the document according to a semantic meaning which will make its retrieval easier. At the same time, we would develop a software architecture that should be able to automatically generate the right data entry forms needed to index each particular class of documents, avoiding the expensive activities of the evolutive maintenance of the application. We made use of ontologies to express the real semantic value that each document could hide, simplifying their indexing and retrieval.

Keywords: On the fly form generation, Ontology, Electronic documents, ontology and database synchronization, Document classification

1. INTRODUCTION

The presence of huge paper archives and the need of their frequent consultation is cause of complex problems for a working organization. One of the more consolidated solution is the conversion of these archives from paper to electronic format, a solution which provides several advantages in terms of space optimization, use flexibility and retrieval easiness of the same documents. Very often (for example for the fax documents), the quality of the original doesn’t allow the use of the Optical Character Recognition techniques (OCR), so such documents are converted in simple raster image without any possibility of electronic retrieve of their content. Similarly, also the documents directly generated as electronic (digital images, word processor documents, files produced by specific software applications) contain several information very diversified which don’t allow an easy aggregations of them in groups of similar documents and their retrieve after they have been archived.

To overcome the limits previously mentioned, in time, they has been developed several classification systems which support user in the categorization of his own folders of documents; they go from professional systems used for example in digital libraries to those more “personal” as for example the file tagging system which equips Windows Vista.

These classification systems involve, however, an additional conceptual and operative effort, as it’s needed to specify unique information to add and link to the image of the acquired documents (for paper documents) and/or to the electronic ones. This information, retrieved from the content of the original text, allows to index the image and to qualify it inside the data base resulting by the final transformation of the paper archive into electronic data. The phase of design and definition of the “indexes” is very delicate, because it has to be performed “a priori” and it will be the essential condition for an optimal research inside the electronic documents archive.

If we think that a dossier corresponding to a request for founds or loan can be composed by more documents each sent with a different date, we easily understand the difficulty of the traditional paper archive personnel when it’s needed, for example for legal problems, to recompose all the documents related to the same dossier.

It’s then necessary a study phase in order to identify the document’s indexing criteria, able to assure, for the future, the possibility of an unambiguous document research, also correlated to all those documents of the same file. On the basis of this criteria, it will be possible to view, through appropriate consultation techniques of the data base, the entire sequence of documents received in time, related to a specific object, or it will possible to group the documents by a specific temporal period, customer and so on.

Typically the phase of keywords definition is in charge to the company providing the recording service and it’s an onerous and at the same time not much paying activity. According to these assumptions we were brought to think to a centralized system (realized as web application) able not only to provide its users recording space in a “virtual hard drive” fashion, but also to support to autonomously perform the delicate preliminary phase of indexing also the less technically skilled users. It’s on this basis that the eSCI project has been carried out.
2. eSCI PROJECT

The eSCI project (eStore of Captured Information) is a research project lead by Memar Monteassegni, Italian company working for more than 20 year in the field of paper and digital storage and GSA-Lab (Graphics and Software Architectures Laboratory) of the University of Salento.

One of the main objectives of this project is to develop a Web portal, called YouFile, that will allow users to directly interact with the company in order to electronically store digital documents or to activate its services related to the paper documents recording, providing tools for the auto-configuration of their own archives, OCR and Information Capture services etc. Through the YouFile portal, the user will select the desired service of both electronic and paper record.

The main expected result is to free the company of all those phases as customers’ search and acquisition as well as the indexing of their documents, allowing them, without any company operator help, to request paper document storage service, to request document digitalization, to store electronic documents and to organize them into folders in order to retrieve them in a fast and secure way.

eSCI project is divided into three main research themes, the first is connected to the definition of a storage hardware and software architecture able to support many electronic repository and a great amount of e-documents, the second concerns the automation of the paper storage process, booked or activated though the web application, the third regards the development of a system which assists the web users in the creation of their own electronic virtual folders and in the following definition of the indexes for each e-document class.

It’s just the last one, i.e. the development of assistive systems helping the web user in the creation and maintenance of their digital folders, the main target of the project. In fact, as previously said, the main difficulty in these systems is the complete characterization of every document, with appropriate keywords, to be able to perform a targeted effective and fast search.

3. RELATED WORKS

The basic idea of the project is not too far from the concept of digital library [1] born in the sphere of biblioeconomy and then widely developed in time.

Several are the examples of digital libraries, for example the Vatican Library or the Digital Library at the University of Michigan. However there is a rightful distinction between digital libraries and digital archives. Differently from the digital libraries, the digital archives were born to hold digital documents directly generated by whom performs their upload, moreover the documents are stored in groups and not as single items: the user performing the file storage, organizes its contents following his own logic, grouping, for example, in different folder the documents that share the same content. Another problem is the privacy issue: while in digital libraries the contents could be accessible also to user groups (even if restricted), in digital archives, typically, every user can access only his own contents.

However, either digital libraries or digital archives share the same main problem, the document indexing issue: it has to be performed in order to facilitate as much as possible the e-document retrieval.

While in the digital libraries the indexing problem can be faced up thought the definition of a set of keywords more or less specific for a defined argument, as in the most cases the document are limited to a specific and well known sector, in the digital archives field it is impossible to think to predefined keywords because the types of the documents is diversified and each user could define different keywords for the same document’s type.

Document indexing practice (either for single documents or groups) has very far origins: the first was the “faceted” classification system, devised in 1930 by the Indian Ranganathan and which subsequently evolved as indexing standard even if without asserting itself [2].

In the last few years the trend is to perform indexing starting from a first phase of knowledge extraction, realized by more or less sophisticated pattern recognition techniques: for example [3].

In recent times several portals have been developed to store personal digital documents (mainly photographs and video). Let’s think for example at YouTube (www.youtube.com) or to the photo sharing portal Flickr (www.flickr.com).

The proliferation of such portals, following the Web 2.0 trend of information sharing, is not at all suitable to store documents containing sensible data as private and personal documents which a user would like to store in a secret and secure place.

Moreover, those indexing system are based only on simple user self defined tags and, even if in Web 2.0 applications it is useful and powerful feature, it seems to be not much usable in a system which has one of its main target in assisting users in the correctly indexing of their documents by providing a set of keywords that should be able to entirely characterize them.

4. OPEN ISSUES AND MOTIVATION

The company typically manages the phase of definition of the search keys through one or more interviews with the customer who comments his documents. The phase of analysis is oriented to the definition of a model for the specific document class. This model is used to define the most important unique data in each document class in order to let the company to retrieve the right document when its owner ask for it. Traditionally the phase of association of the search keys to each document is performed through the support of a software application which has a certain set of data entry forms, developed “ad hoc”, one for each document class.

The complexity of the recording and indexing management application is mainly due to the large amount of data entry forms that must be developed from scratch in order to characterize every different type of document. This reflects on:

- an overhead in the maintenance of the archive management application that must be continuously evolved in order to update the set of supported document models with the new documents types that, periodically, is needed to add;
- a cost for the company that is permanently tied to a software house for such maintenance;
- a weak reactivity with respect to the consumers, which often has to wait to index their document belonging to new and not yet supported class, while the indexing application is being updated;
- although we can hypothesize forms of customized indexing, where the same customer takes care of the search keys definition of his own digital archives, this would generally produce two negative effects:
  - the proliferation of heterogeneous document classification systems discouraging any standardization;
  - it wouldn’t assist the customer in the generation and maintenance of his own archives, not valuing, de facto, the experience in the document management field that the company has laboriously consolidated in the course of the years and that it would have to represent a added value of the entire system.
In order to achieve such result, it’s important that the needed information for the data entry form generation are expressed as concepts with a strict semantic meaning. The ontological bases seem to be the better means for the archiving of this kind of concepts.

5. PROPOSED SOLUTION

In the traditional archive document systems, the information about the documents to index, is acquired through data entry form statically generated. This feature represents an important design constrain because it is not much scalable in architectures that make use of many document types. It would be indeed extremely complex and expensive to develop new interfaces whenever it’s required to define a new document class. Another limit of a static form generation system is surely represented by the updating issues of already defined models. In fact, the simple modification of a field would imply the reprogramming of the whole interface and the database structure.

We advocate that the solution, in order to make the indexing systems more flexible and effective, can be the use of ontologies where to keep the company know-how about the best indexing strategy for each document class and a proper interface generator able to dynamically propose users the search keys more suitable to describe every specific document.

We show in Fig. 1 the logical architecture of the implemented tool.

Fig. 1 - Architecture Logic Model

The figure above (Fig. 1) shows two main subsystems:

- Storage repository of document classes, developed through the ontology language OWL (ontology web language): here are stored the document classed description in a XML file over the server file system.
- Instance Repository of concepts/document that represents the real archived documents: this is a real data base containing the instances of every document uploaded in the portal virtual hard drive, as well as every keyword’s value selected by the user to fully characterize his document.

One of the main problem of this approach is the need to keep synchronized the knowledge base of the document models and the ontology.

It’s important to maintain this synchronization because every concept (expressed in the ontology) has to be translated in one or more tables inside the database. When the user requests the uploading and indexing of a new document belonging to a particular document class, the ontology supplies the key fields needed to its description. The Jena libraries are useful to manage the ontology models written in OWL language [8]. Through these libraries, the information in OWL format, regarding the dynamic form generation, are translated in XML format and successively processed by a parser able to interpret them and generate the corresponding Form/GUI for the data entry of the search key values. When the user fills the key fields with the information describing his document, this data are sent to the system that takes care to save these information into the instance database.

Next document search performed by the user will obtain information both from knowledge base and the data base. Knowledge base provides information needed to generate the form through which leading the user in the desired document class selection, while the database will keep track of every document uploaded together to the set of keywords used to characterize and qualify it.

6. KNOWLEDGE BASE

Design Methodologies

In the international scientific panorama there are several design methodologies for ontologies (Uschold & King methodology [4]; Gruninger & Fox methodology [5]; Methontology [6]). The methodology used in this paper is that proposed by Stanford University [7]: it was parallely born with the tool Protégé, an open source editor of ontologies. It is possible to download this tool at the http://protege.stanford.edu/. The tool allows to represent ontology in visual way and to obtain its formal representation. The methodological steps foreseen by the Stanford University methodology are enough general and therefore easily adaptable to whatever particular application domain.

We choose the methodology proposed by the Stanford University for two main reasons:

- The methodology is enough detailed: thanks to the formulation of the questions, of the list of the terms and to the following distinction among concepts and relationships within that terms, the methodology is a precise guide in the definition of the knowledge base.
- The methodology was born together to the Protégé editor: it is a reliable tool to design ontology. The editor (and the methodology) is supported by an ardent community of developers and users. The creators of the methodology are, therefore, constantly available to furnish some very useful practical suggestions in the modelling of particular situations.

Motivation about the use of a knowledge base

The idea to use a knowledge base to support the archiving of the digital documents, was born to answer to one of the main goals of the project that is to give the possibility to the user to store some documents in way as close as possible to its way of reasoning to facilitate the following recovery of the same document. To do this it is essential to represent the reasoning made up by the user to archive his own documents and this means to reproduce the semantic connections that he thinks in order to store his own documents. Just this mechanism is at the base of the semantic web, therefore the definition of the ontology, that captures in formal way the semantic connections among the representative concepts of the several documents that a consumer would like to archive, seems the solution most suitable to our goals.

Our basic idea is to define a set of characteristics (concepts), that are in common between a set of documents, linking together the main characteristics of these documents in order to define the overall characteristics of a well defined typology of document (i.e. document class).

Besides, every typology of document that will be represented in the ontology won’t be characterized only by information that tightly characterize the document (such as creation date, author etc.) but it can be also characterized by other information, not specific of the document, but referring to a particular semantic connection made up by the user (concepts that does not characterize the document) (Fig.2).
In fig. 2, for example, the document class 1 is characterized by both the concepts “C1” and “C3” that characterize the document (C) and of the concepts “N1” and “N2” that don’t characterize the document (N). The document class 2, instead, is characterized by both the concepts “C2” and “C3” that don’t characterize the document and by the concepts “N2” and “N4” that don’t characterize the document.

As we can see, the concept N2 and the concept C3 are common to the two typology of document represented.

The ontology is not only a simple “data descriptor” but a descriptor of possible semantic link between data. The ontology is a useful guideline for the user in the hard work of the document classification.

Besides, always to answer to the requirements of the project, the consumer will be facilitated, through the knowledge base, in the following retrieval of the documents: he is supposed to follow the same reasoning already done for the filing of his own documents in phase of retrieval.

Hypothesis of work analyzed

In the following, we discuss three different approaches analyzed in order to define a knowledge base for our goals.

Ontological representation of all possible concepts that a user may define for a specific document.

In the realization of the ontology we may think to define all possible concepts and sub concepts tied up to a specific matter.

Following this approach, in our case we have to define, for all the document classes individualized, as many subclasses as the possible documents that a customer would like to insert.

The approach has been therefore rejected for 2 principal reasons:

• The massive structure of documents that we want to manage is very tall and therefore very tall will be the number of subclasses to define;

• To foresee in advance all the possible cases for every type of document is impossible, therefore it would be possible that customer would be forced to continually ask to the company for ontology updates: this, if from a side increases the work load for the operator, from the other side brings the customer to abandon without a lot of facility the portal immediately not finding answer to his need of archiving.

Definition of a scheme that allows to semantically characterize the documents with the persistence inside the knowledge base.

Discarded the first hypothesis, the focus of the realization of the knowledge base moves from the concept to the metadata: it is possible to define, therefore, an ontology that doesn't describe the specific concepts of every typology of document but the metadata that allow the client to describe his own typology of document.

In other words, a basic structure is defined through which the customer is driven in to perform the classification (and the following recovery) of his own document.

With this approach the knowledge base will semantically define a series of concepts connecting each other that allow to describe the document and the possible semantics' relationships that tie the document to other concepts and that allow to characterize it.

At this point, supposing to want to store a picture, the knowledge base will represent the concept of picture connected, for instance, to the concept of landscape or to that of event (to tie the picture to a particular event of his own life).

The specific picture of the customer will be simply an instance of these concepts semantically correlated between them. If another user wants to archive, for example, a picture related to a "trip in mountain in summer" he would instance exactly the same concepts but giving them his own value.

With this approach to every documents indexed by the customer, it will correspond the instance of one or more concepts inside the knowledge base and therefore the persistence of the documents will directly be realized inside the knowledge base.

Also this hypothesis doesn't result realizable: the portal is designed to keep a great number of documents. The huge number of instances for every typology of document will make the management of the ontology difficult from the technological point of view.

Definition of a scheme that allows to semantically characterize the documents equipping with a layer of persistence inside the database.

The adopted solution has been to combine the undisputed advantages tied to the definition of an ontology whose focus is on the metadata and not on the concept, with the advantages coming from the use of a data base that will constitute, therefore, the layer of persistence. In this way it can be fully exploited, from one side the syntactic and semantics expressiveness of the knowledge bases and from the other side the technologies already consolidated within the data base field to manage a massive structure of data. The logical scheme of this architecture can be so represented (Fig. 3)
Every attribute is characterized by different properties that define it, such as, for example, the dimension and the data type: these attributes are very important for the on the fly data entry form generation.

Some attributes of a document can be semantically put together with other attributes, for instance it is possible to think about the personal data. Every document, could ask for different personal data for every different person, and therefore it is necessary to subsequently specialize the concept of person distinguishing among personal data useful for the different typologies of document.

Therefore, the followed approach has been that to individualize 3 macros categories of concepts:

<table>
<thead>
<tr>
<th>Concepts</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Attributes</td>
<td>The concept is made up by several sub concepts; this concept gathers all the possible attributes that can be individualized in the several typologies of documents examined. In turn every attribute is an ontological class that can contain other subclasses</td>
</tr>
<tr>
<td>Documents</td>
<td>Subsequently divided in subclasses it defines in every subclass the possible typologies of documents. In turn every document can contain other subclasses</td>
</tr>
<tr>
<td>Support</td>
<td>It defines all that concepts that are not tightly tied to the document but that allow to specify better the its characteristics.</td>
</tr>
</tbody>
</table>

Table 1 Different categories of concepts

Through a deeper analysis it has emerged that, actually, for every document it is possible to individualize a series of concepts correlated to it. Such concepts are not necessarily concepts that characterize the document or they would be able to link together, as in the case quoted before "person" attributes tied up semantically each other.

For this reason, with a bottom-up approach, we realized a generic ontological class "concepts" inside which we created two sisters subclasses: the subclass "Principal" in which several documents are defined and the subclass "Secondary" in which the concepts that do not characterize the document are defined. Of course, well defined object properties will allow, where necessary, to semantically connect Primary and Secondary concepts.

Definition of the ontological properties

We define three different types of properties:

- The object property that links together documents and attributes that define the documents. The name of these properties is "has attributeName" and they have as range the class that represents the attribute previously defined.
- The object property that links together primary and secondary concepts. Secondary concepts can group together several attributes or concepts in order to better define (and through property that does not characterize the document) a document.
- Property that characterize in a specific way each attribute of the document (attribute data type, length etc.)

Property restriction

All the restriction defined belongs to the type "owl:hasValues" and they allows to define the value that a well specific property assumes in a well defined context. We use restriction to define the properties that characterize each attribute.

Adding instances

The last step of the methodology proposed by the Stanford University, adopted by us, is to add, if necessary, some instances. As underlined previously the instances in our knowledge base are useful to add a layer of semantic to the concept defined in terms of classes and properties: the index keys will be stored in the database. In this way the ontology is complete and it will be the input for the tool that automatically generates the data entry form.

Definition of the mapping with knowledge base

To realize the persistence layer we studied a very simple algorithm that allows to produce in automatic the database where the instances of the knowledge base will be stored. We don't define here all the details of the algorithm: it is enough to underline that every class that represents a different typologies of document become a table whose attributes are represented by the Object Property that they connect the document to the relative subclasses of the class "attributes". In the mapping we managed through opportune N:M relationships the existing relationships among primary and secondary concepts.

Update the knowledge base: methodological guidelines.

Using the designed and implemented portal it is possible that, to answer to the customers’ requirements, It could be needed to modify the knowledge base.

It is possible to modify the knowledge base under three points of view:

- Add/delete an attribute to a document already defined in the knowledge base.
- Add/delete a semantic relationship between a document and a concept that does not characterize the document (the concept may be a new concept or an already existing concept);
- Add a new typology of document.

It is clear that the domain expert must update the knowledge base. The domain expert is responsible to modify the knowledge base coherently with the old one.

Update the data base

For each change to the knowledge base it must correspond a change to the data base. The change for a side must protect information already in the knowledge base and for the other side must allow to insert the new information according to the new knowledge base.

An example

In order to understand the realization of the knowledge base, we show an example. We start from the concept of PICTURE. When the user wants to store its pictures, the knowledge base will drive the user in order to insert information that will make simpler the retrieval of the uploaded files. The document in the knowledge base will be the PICTURE document and it will be represented as a subclass of the “Document” class. The picture is characterized by the year, the period of the day, position and the camera used to take a picture and so on. The picture may be colour or black and white and may have a well defined orientation (portrait or landscape).

It is clear that the listed concept, except the year, orientation and colour are concept that does not characterize the picture but they are very helpful in order to allow the user to remember its own keywords.

The secondary concepts here listed may be in common with several typology of document: we can think to a drawing or to another image that a user may want to archive.
The secondary concepts are defined one time and they are used again when necessary. An example of secondary concepts is the protagonist of a picture.

7. CONCLUSIONS

To archive a document doesn't mean to preserve reams of printed paper but it means, by now, to preserve electronic documents. This is true both for the companies and for the private citizen that use less and less the printed paper and always more digital documents that can be of several type: from photo, video, word processor to the fiscal documents. Become necessary to provide a tool that allows the consumer to archive safely digital documents of various type. The basic idea of the eSCI project is to realize an efficient and simple to use tool that allow both the indexing and the archiving of electronic documents. The proposed architecture is constituted by two key points that provide an high added value for the company:

1. the possibility to generate on the fly the data entry form;
2. the possibility to provide through an ontological support, a set of keywords for every typology of document that drives the user in the classification of his own documents

The on the fly data entry form generation reduces the need to ask to the IT expert (inside or outside the company) in order to add a new typology of document and/or to modify the way of indexing it: the activity becomes, instead, an updating of the knowledge base that can be made with extreme simplicity and in less time and above all inside the company, without asking to external software house. Current technology makes simple to read the ontological file that represent the knowledge base in order to generate on the fly the data entry forms.

It is clear that it is very important to synchronize the knowledge base with the underlying database: it contains the keywords of the user and it will ensure the retrieval of the information and of the files connected with the keywords.

The use of the knowledge bases for the representation of the documents allows, besides, to make explicit the know-how already acquired by the company in the field of the document indexing, avoiding, in this way, fragmentations of the know-how among the several divisions specialized in a well specific set of documents. The proposed architecture allows, besides, to transform the know-how acquired by the company in a service that the company offers to the customer. Thanks to the formal and precise description of each document that the company allows to store, the data entry forms helps the final user in the indexing of his own documents without any need of company’s personnel intervention.

8. Acknowledgments

We would thank Tommaso Mezzina and the Memar Monteasaggi company for the tangible support. Moreover, particular thanks to the Apulia Region that has founded eSCI project.

9. References

[8] W3C (February, 10 2004) “OWL Web Ontology language Reference” W3C