A Method for Maintaining Document Consistency Based on Similarity Contents

Farid Meziane* and Yacine Rezgui**

University of Salford, Salford M5 4WT, UK
{f.meziane,y.rezgui}@salford.ac.uk

Abstract. The advent of the WWW and distributed information systems have made it possible to share documents between different users and organisations. However, this has created many problems related to the security, accessibility, right and most importantly the consistency of documents. It is important that the people involved have access to the most up-to-date version of the documents, retrieve the correct documents and should be able to update the documents repository in such a way that his or her documents are known to others. In this paper we propose a method for organising, storing and retrieving documents based on similarity contents. The method uses techniques based on information retrieval, document summarisation and term extraction and indexing. This methodology is developed for the E-Cognos project which aims at developing tools for the management and sharing of documents in the construction domain.

1 Introduction

The main activity of most PC users is about creating, managing, deleting and retrieving electronic documents. Thanks to the existing file management systems, this organisation is performed using hierarchical structures. These structures categorise documents using their properties. For example, we would create a file "lecture1.ppt" in the subdirectory "Lectures" which is itself a subdirectory of the "Object-oriented Design" subdirectory. In fact we are associating some semantics to the created file. It is a lecture for the "Object-Oriented Design" module. However, using strict hierarchical filing can make it hard for users to perform the following operations [5]: *File documents:* documents can appear in only one place; *Manage documents:* locations in the hierarchy is used for organisational and management purposes; *Locate documents:* Document may be filed according to one criterion but retrieved according to another; *Share documents:* different structures for different people. The task becomes even more complex when we are dealing with various documents of one or many organisations. The problem becomes even more complex if the WWW is used as the place to exchange and organise these documents. Another major problem faced with shared documents

* School of Sciences
** Information Systems Institute

is consistency whereby everybody interested in the document should be aware of any changes to the document. Some systems have been developed to address some of these issues. The Presto system [4,5] aims at creating placeless documents and attempt to create a more natural and fluid forms of interaction with a document space. DocMan [2] is a document management system which supports cooperative preparation, exchange and distribution of documents. The system particularly stressed on the loss of work done simultaneously on a document and access restrictions. The Zelig System [3] was developed for managing multiple representation documents.

In this paper we present a methodology for maintaining document consistency using similarity content. This methodology is developed for the E-Cognos project which aims at developing tools for the management and sharing of documents in the construction domain. The approach is based on generic principles related to information retrieval and knowledge management. The aim of this project is to exploit these principles to develop an approach that will support consistency across large knowledge repositories maintained in a heterogeneous and distributed collaborative business environment. The approach is based on a solid theoretical foundation, and will be deployed in a real business environment. The remaining of the paper is organised as follows: in section 2 we present the motivation and the background of the project. Section 3 will define the different types of document considered in this project. In section 4 we present the generic model of the methodology used for poorly structured documents. Sections 5 and 6 presents variants of the methodology for documents with text formatting structure and highly structured documents. We end the paper with a short conclusion.

2 Background and Motivation

Numerous documents of diverse nature are involved in the construction domain. These documents are of two types: drawings and written documents. Drawings are the straightforward media to convey most of the information needed by construction companies and include a lot of information that can be hard to put into words. They are usually more formal and comprehensive than text information. Moreover written documents are complementary to drawings, they are the traditional support of an engineering project description. Some of them such as building codes, examples of technical solutions, computation rules define the legal context of a project. Others like technical specifications documents or bill of quantities are generated by the engineering activities and often have a contractual importance.

The documents generated within the entire life cycle of a construction project, and especially during the design stage, need to be of quality in order to provide a reliable basis for contractors to perform their construction activities. Documents of quality are obtained by ensuring, during their production, a coherent and consistent structuring both on the logical and physical side. This structuring is

relevant in the sense that the semantics of a document can be efficiently mastered and thus correctly described.

Moreover, a document has not only to be self consistent but needs also to be consistent with the entire project documentary base as well as the construction standard and regulation base. Furthermore, many practitioners and researchers in the construction domain have recognised the limitations of current approaches to managing the knowledge relating to and arising from a project in a distributive collaborative environment. Among the reasons for these limitations are:

- Much knowledge, of necessity, resides in the minds of the individuals working within the domain.
- The intent behind decisions is often not recorded or documented.
- The knowledge gained during a project is often poorly organised and buried in details. Hence, it becomes difficult to compile and disseminate useful knowledge to other projects.
- People frequently move from one project to another, making it difficult to track those involved in decision making.

Knowledge in the construction domain can be classified into the following three categories:

- *Domain knowledge:* this forms the overall information context. It includes administrative information (e.g. zoning regulations, planning permission), standards, technical rules, product databases, etc. This information is, in principle, available to all companies, and is partly stored in electronic databases.
- *Corporate knowledge:* this is company specific, and is the intellectual capital of the firm. It resides both formally in company records and informally through the skilled processes of the firm. It also comprises knowledge about the personal skills, and project experience of the employees and cross-organisational knowledge. The latter covers knowledge involved in business relationships with other partners, including clients, architects, engineering companies, and contractors.
- *Project knowledge:* this is the potential for usable knowledge and is the source of much of the knowledge identified above. It comprises both knowledge each company has about the project and the knowledge that is created by the interaction between firms. It is not held in a form that promotes reuse (e.g. solutions to technical problems, or in avoiding repeated mistakes), thus companies and partnerships are generally unable to capitalise on this potential for creating knowledge.

This overall context has often resulted in knowledge redundancy and inconsistencies, business process inefficiencies, and change control and regulatory compliance problems. Moreover, the introduction of new national regulations, or amendments made to existing ones, are often not handled effectively within organisations and projects.

3 Documents and Their Logical Representation

A document is a transitional and changing object defined within a precise stage of the Project Life Cycle. Generally, a document is related to many elaborated documents of the Project Documentary Database. A document has one or many authors. It is described by general attributes such as a Code, an Index, a Designation, a Date of creation and a list of the document's Authors. Ideally, a list of document versions also keeps memory of any amendments made to the document during its lifecycle. An indexing system may be associated to the document. A document is submitted for approval according to a defined circuit of examiners representing diverse technical or legal entities. Each examiner issues a statement that enables the document to be approved, rejected or approved under reservation.

Documents have also been traditionally represented using a set of key words. These key words or indices can either be manually defined by a user with a good knowledge of the semantics of the document, or extracted automatically from the text of the document using proven Information Retrieval (IR) techniques.

A document has a logical and a physical structure, which are both used to convey in the best possible way its internal semantics. The physical structure of a document is described using a properly defined syntax supported by one or several software tools.

Each document should have ideally metadata attached to it. A possible solution for describing metadata is through RDF *(Resource Description Framework - a development based on XML)* that provides with a simple common model for describing metadata on the Web. It consists of a description of nodes and attached attribute/value pairs. Nodes represent any web resource, i.e. Uniform Resource Identifier (URI), which includes URL *(Uniform Resource Locator)*. Attributes are properties of nodes and their values are text strings or other nodes.

Documents are classified based on their inherent nature and the structure they exhibit taking into account the specificities of the domain; the construction sector in the E-Cognos project. Three classes of documents have been identified, namely: *poorly structured documents, documents with a clear physical structure,* and *highly structured documents*.

Poorly structured documents are composed of text with no formal structure. These constitute the vast majority of construction documentation. Documents are treated here simply as black boxes. The set of amendments to this category of documents include modifying the content of the document and deleting the document.

Documents with a text formatting structure are documents that are tagged using HTML, or at best XML but without reference to a Document Type Definition (DTD). A physical structure in the form of a hierarchical tree, or hypertext link of nodes can, in theory, be easily generated from this representation. This structure offers a variety of possibilities in terms of text retrieval. These documents include direct references to other documents/document sections. The set of amendments to this category of documents include inserting a new docu-

ment element (heading, paragraph, section, etc.), deleting/modifying an existing heading in a document.

Highly structured documents are instances of an XML-based meta-language. These documents have a semantic structure that can easily be used as a basis for text queries and retrieval. Ideally, we can envisage that all the documentation that is used and produced be an instance of a specific XML DTD over which users can exercise control over its internal semantics. These documents include naturally direct references to other documents/document sections. The set of amendments for this category of documents include: adding a new DTD element to the DTD language, instantiating a new DTD element within a document and deleting the instance of a DTD element within a document.

4 System Description

The general framework of the methodology, as shown in Figure 1, is for poorly structured documents. The methodology is composed of 7 steps and these are described in the following subsections. Step 0 is the entry point to the system. It can be the submission of a new document or the re-submission of a modified document. Both instances will go through the same process. A logical document is used for searching and other document related operations. A Physical document is only retrieved on users requests.

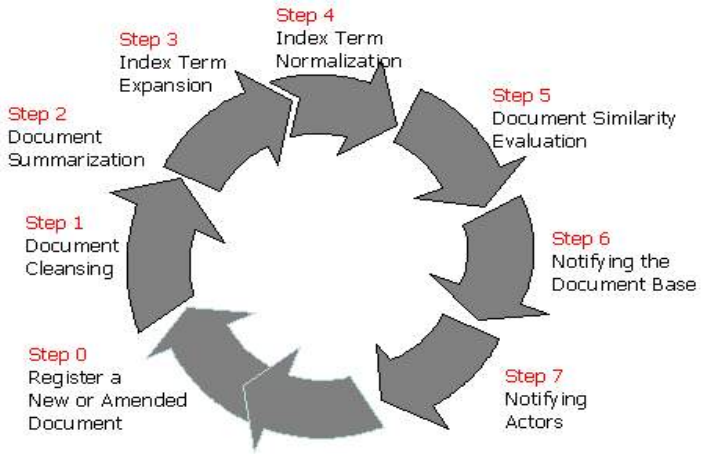

Fig. 1. System Overview for Poorly Structured Documents

4.1 Document Cleansing Module

This step aims at reducing the document to a textual description by eliminating non-discriminating words. The resulting document contains mainly nouns and

association of nouns that carry most of the documents semantics. A cleansed document reduces drastically text complexity allowing better performance in document retrieval and processing. This involves the following tasks:

- Lexical analysis of the text in order to treat digits, hyphens, punctuation marks, and the case of letters. This reduces the initial document into a subset of words that are potential candidates for index terms
- Elimination of stopwords to filter out words with very low discrimination values for retrieval purposes.
- Stemming the remaining words with the objective of removing affixes (prefixes and suffixes) and preventing the retrieval of documents containing syntactic variations of query terms, e.g. use, using, used, usage, etc.

4.2 Document Summarization

This step aims at providing a logical view of a document through summarization via a set of semantically relevant keywords. These are referred to, in this stage, as index terms. The purpose is to gradually move from a full text representation of the document to a higher-level representation. This module is composed of the following tasks:

Index Terms Extraction In order to reduce the complexity of the text, as well as the resulting computational costs, the index terms to be retained are:

- All the nouns from the cleansed text. It is in fact argued that most, if not all, of the semantics of a text document is carried out by nouns as opposed to verbs, adjectives, and adverbs.
- Noun groups (non-elementary index terms) co-occurring with a null syntactic distance (number of words between the two nouns is null).

Extracting the Structure of the Document This stage will only be possible if the document has been produced using a document formatting language, including RTF, SGML, HTML and XML. Each node of the resulting hierarchical structure will have an identifier that will be used to track nodes, their parents and children. A node might contain other elements including references within or outside the scope of the document, figures, tables, formulas, etc.

Establishing the Inverted File Structure of the Text The purpose of an inverted file structure is to track the position of each index term occurrence in the text. The positions of index term occurrences can be tracked either on a word or character basis, or on a physical position basis (by pointing to node identifiers for example). The latter technique, referred to as Block Addressing, can only be used where a physical structure of a document is available. It presents the advantage of reducing space storage requirements. Documents with no clearly defined physical structure will make use of word addressing. Two activities are involved in this stage:

- Determining the raw frequency of each index term in the text; This is referred to as intra-clustering similarity in the Vector Model [1, 6, 7]. This aims at determining the number of times a term is mentioned in a document, as well as the location in the document of the term occurrence.
- Determining the number of documents in which each index term appears. This aims at counting the number of documents of the Document Knowledge Base (Project Knowledge Base and / or Corporate Knowledge base in which the term appears).

Calculating the Index Term Weight for the Document The purpose here is to quantify the degree of importance in terms of semantics that the index term has over the document. It is proposed that the formula defined for the Vector Model [1, 6, 7] will be used.

4.3 Index Terms Expansion and Normalization Using a Construction Thesaurus and Ontology

This steps aims at normalizing the index terms obtained from the previous stage by using either direct ontology concept mapping wherever possible, or indirect ontology mapping by using a thesaurus as described in Figure 2. If no direct mapping exist between the initial index term and the list of ontology concepts then the thesaurus is used to provide synonyms for each term. The synonyms are used for indirect mapping. It is important to emphasise that the ontology is the structure that is used to convey semantics and maintain knowledge consistency across the project, corporate and domain layers. As such, the concepts of the ontology are the unique reference for the E-Cognos platform.

Ontology Concept Expansion Based on Concept to Concept Relationship It is proposed in this methodology that the retained concepts be expanded based on their ontological direct relationships. We distinguish three main types of relationships:

- Generalisation/Specialisation Relationships
- Composition/Aggregation Relationship
- Concept association with varying semantics

Ontology Concepts Weighting The ontology concepts resulting from a direct document index term mapping, indirect index term mapping, or ontology concept expansion need re-weighting. The following approach is proposed:

1. In case of a direct mapping the weight of the document index term is applied as such to the ontology concept.
2. In case of indirect mapping or concept expansion, it is proposed that a correlation factor be applied to the initial document index term weighting. The correlation factor is obtained by the cosine of the angle between the Index Term Vector and the Ontology Concept Vector. This is based on a technique used in Query Expansion Based on Similarity Thesaurus [8].

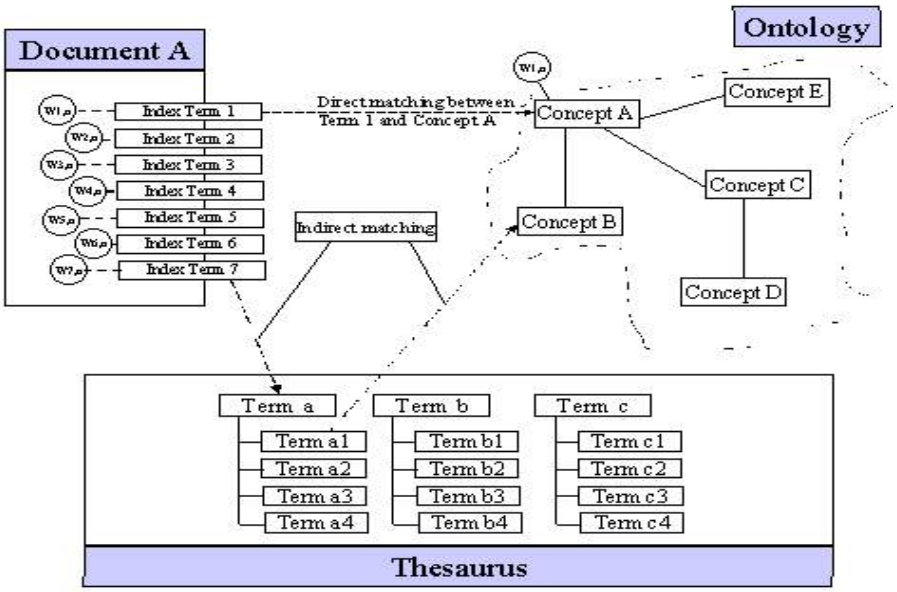

Fig. 2. Index Terms Mapping Against the Ontology

4.4 Document Similarity Evaluation

The purpose of this step is to compare the similarity between a newly uploaded/processed document with the remaining documents (or knowledge items) stored in the various knowledge repositories.

Document Similarity Calculation Against all the Document Set The purpose here is to provide a function that evaluates the similarity between two documents. We adopt the following approach:

Let t be the number of index terms in the system and k_i a generic index term. $K = \{k_1, k_2, \ldots, k_i\}$ is the set of all index terms. A weight $W_{i,j} > 0$ is associated with each index term k_i of a document d_j . If an index term does not appear in the document text then $W_{i,j} = 0$. Therefore, with a document is associated an index term vector:

$$\overrightarrow{d}_j = (W_{1,j}, W_{2,j}, \ldots, W_{t,j})$$

Based on the document index term vector above, two documents d_i and d_j are represented as t-dimensional vectors. The approach adopted by the Vector Model to evaluate the similarity between a query and a document by measuring the correlation between their index term vectors is used. Furthermore, the similarity between two given documents will be measured by the correlation between

their index term vectors. This correlation can be quantified by the cosine between these two vectors. $sim(d_i, d_j)$ varies between 0 and 1. An example of an illustration of the matrix is given in Table 1

	Doc A	Doc B	Doc C	Doc D	Doc E	Doc F	Doc G
Doc A	1	sim(B,A)	sim(C,A)	sim(D,A)	sim(E,A)	sim(F,A)	sim(G,A)
Doc B	sim(A,B)	1	sim(C,B)	sim(D,B)	sim(E,B)	sim(F,B)	sim(G,B)
Doc C	sim(A,C)	sim(B,C)	1	sim(D,C)	sim(E,C)	sim(F,C)	sim(G,C)
Doc D	sim(A,D)	sim(B,D)	sim(C,D)	1	sim(E,D)	sim(F,D)	sim(G,D)
Doc E	sim(A,E)	sim(B,E)	sim(C,E)	sim(D,E)	1	sim(F,E)	sim(G,E)
Doc F	sim(A,F)	sim(B,F)	sim(C,F)	sim(D,F)	sim(E,F)	1	sim(G,F)
Doc G	sim(A,G)	sim(B,G)	sim(C,G)	sim(D,G)	sim(D,G)	sim(E,G)	1

Table 1. An example of a Document Similarity Matrix

Establishing Document Clusters Based on the Similarity Table The purpose here is to propose clusters of documents based on their degree of similarity. These clusters can directly be generated from the Document Similarity Matrix proposed in the previous section.

4.5 Notifying the Constituents of the Document Base

The purpose of this step is to notify relevant documents of the knowledge base, based on the nearest cluster(s), the potential risk of inconsistency that might exist as a result of a new event (upload of a new document, amendment to an existing document, etc.).

4.6 Notifying Relevant Actors

The purpose of this stage is for each potentially inconsistent document to notify actors who have subscribed an interest into the document (including authors) about this last event, and its potential degree of inconsistency. This notification will be materialised by the sending of an XML-based description of the metadata of the newly created or amended document to all the concerned actors.

5 Maintaining Document Consistency Based on Document Explicit Relationships

This case applies to documents that have a clear physical structure and make use of hypertext navigational and cross-referencing links. Hypertext allows non-sequential browsing and editing of text. It can be represented as a network of nodes that are correlated by direct links in a graph structure. Each node is

associated with a block of text that can represent a paragraph, chapter, section, or even a web page. Two related nodes are connected one to the other by a direct link, which correlates the text associated with these two nodes. This is explicitly described in the text by a special tag, or a highlighted portion of the text. Figure 3 describes the application of the method for this type of documents. The proposed rules to apply are as follows:

Rule 1: if a node is amended then the node in question as well as the recursive parents should be flagged as potentially inconsistent. For instance, if paragraph P4 of document B in Figure 3 is amended then chapter b4 of document B(Containment Relationship) and chapter C3 of document A (Referencing Relationship) are potentially inconsistent, and should be flagged as such.

Rule 2: if a node is amended then the external nodes that are referencing it might be potentially inconsistent. These external nodes should be flagged as potentially inconsistent as they do reference an amended node. We consider that it is up to the author to look after the consistency of the internal references of the node being modified.

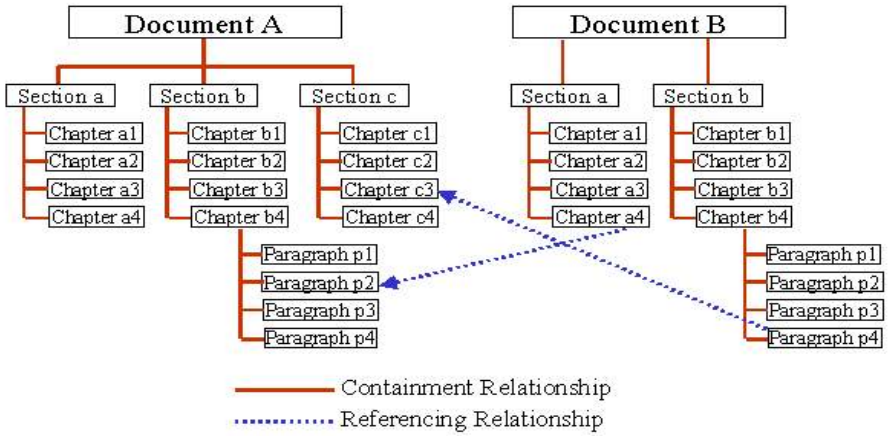

Fig. 3. Relationship Types in a Structured Hypertext Document

6 Maintaining Consistency of Highly Structured Documents

Highly structured documents are best represented by XML DTD compliant documents. XML documents allow human and machine-readable semantics mark-up. XML allows users to define new tags and impose data validation on them. This raises the problem of having unified and standardised definitions of tags used across documents. In that respect, it is highly recommendable to use a

standardised DTD for authoring XML documents. This is already an area of intense activity (AECXML, bcXML, etc.). It is recommended in this approach that the elements of a given XML DTD be interpreted semantically by indexing them, by the author of the DTD, to the concepts of the ontology. Concepts that highly describe the semantics of the contents of the instance of the DTD element are selected and retained by the DTD author(s) as this is a knowledge intensive activity. Therefore, each DTD element will be associated and indexed to a set of ontology concepts, as described in Figure 4.

In the same way, each ontology concept will be associated with a set of indexing DTD elements. Let us take an example:

DTD_Element_A1 has ontology indexes: *(Ont_Con_1,Ont_Con_3, Ont_Con_4)*

DTD_Element_A2 has ontology indexes: *(Ont_Con_2, Ont_Con_4, Ont_Con_6)*

DTD_Element_A3 has ontology indexes: *(Ont_Con_4, Ont_Con_6, Ont_Con_8)*

If an instance of a *DTD_Element* is amended, then the ontology concepts that index this element will be used to characterize this amendment. A simple but not very effective approach is to flag all documents that index the same ontology concepts of a document/or DTD Element instance that has been amended as potentially inconsistent. Using this approach, instances of *DTD_Element_A2* and *DTD_Element_A3* will be flagged as potentially inconsistent following an amendment to *DTD_Element_A1*.

A further step, which makes use of a more sophisticated approach, will attempt to retain only the instances from the flagged DTD elements that contain or reference the same ontology concept instance. This implies that the E-Cognos platform maintains instances of ontology concepts throughout the system.

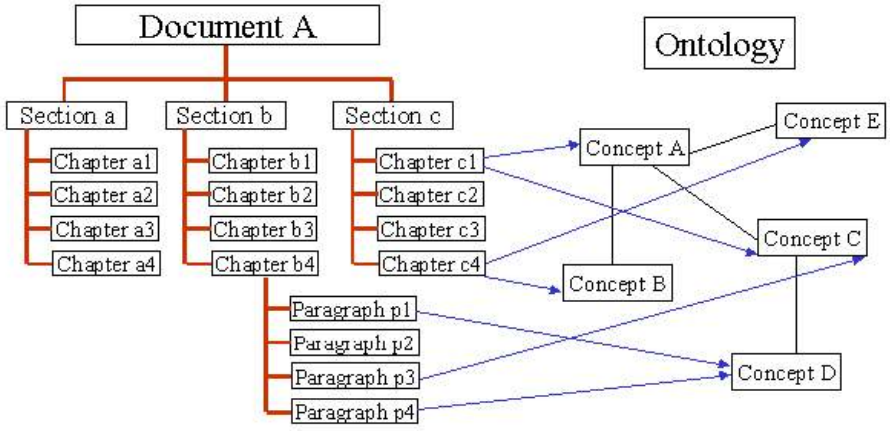

Fig. 4. XML Elements Indexing to the Construction Ontology

7 Conclusion

The work presented in this paper is an initial attempt to specify a methodology for maintaining document consistency across the knowledge repositories of the knowledge, corporate and domain layers of the construction domain. The methodology uses generic principles related to information retrieval and knowledge management that can be incorporated into an approach that supports consistency across large knowledge repositories maintained in a heterogeneous and distributed collaborative business environment. E-Cognos aims at exploiting those principles to develop such an approach based on a solid theoretical foundation, and to deploy it in a real business environment in the context of the project partners. The methodology will be used in the construction domain. However, the model is generic and should be applicable to any other domain. Few changes might be necessary to take into account the nature of the document of the new domain and the use of another ontology which structure may influence some processes of the proposed model. A web-based implementation will be used for the E-Cognos project and this methodology will be implemented using Java and related technologies. It is also worth mentioning that the proposed methods assume that all documents have been authored using a common natural language. Moreover, the multi-lingual aspect of documents has not been addressed at the moment but will be addressed at a later stage.

Acknowledgements

The authors as well as the E-Cognos Consortium would like to acknowledge the financial support of the European Commission under the IST programme.

References

1. R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. Addison Wesley, 1999.
2. A. Bäcker and U. Busbach. Docman: A document management system for cooperation support. In *Proceedings of the Hawaii International Conference on System Sciences (HICSS-29)*, pages 82–91, 1996.
3. A. Celentano, S. Pozzi, and D. Toppeta. A multiple presentation document management system. In *Proceedings of the 10th Annual Conference on Systems Documentation*, pages 63–71, 1992.
4. P. Dourish, W.K. Edwards, A. Lamarca, and M. Salisbury. Presto: An experimental architecture for fluid interactive document space. *ACM Transactions on Computer-Human Interaction*, 6(2):133–161, 1999.
5. P. Dourish *et al.* Extending document management systems with user-specific active properties. *ACM Transaction on Information Systems*, 18(2):140–170, 2000.
6. G. Salton and M. Lesk. Computer evaluation of indexing and text processing. *Journal of the ACM*, 15(1):8–36, 1968.
7. G. Salton and C. Yang. On the specification of term values in automatic indexing. *Journal of Documentation*, (29):351–372, 1973.
8. Q. Yonggang and H. Frei. Concept based query expansion. In *Proceedings of the 16th Annual International Conference ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 160–169, 1993.