# Applications of the Semantic Web for document retrieval

Andreas Eberhart
International University in Germany
D-76646 Bruchsal
E-mail: eberhart@i-u.de

## 1. Introduction

Finding the right information in the Internet or corporate Intranet remains one of the biggest problems in our digital everyday world. Most full-text search engines offer the user the ability to combine search terms with Boolean expressions and to limit the search space. A popular augmentation of this approach is to categorize the area of interest à la Yahoo. This definitely yields a high precision of queries like: give me a list of all Canadian universities, but obviously, due to the organizational and administrative overhead involved, the categories are very coarse grained. Furthermore, one has to make a lot of compromises when designing such a taxonomy of categories and it will definitely not fit everybody's view.

We also face these problems in the area of online learning. The idea here is to define a fixed set of metadata for the documents stored in the document repository. The Learning Object Metadata standard, for instance, suggests storing keywords, the technical document format, difficulty level, etc. for each unit. This information is very useful for the management of content, however, when used for a search interface, too many search parameters confuse the user.

We observe some fundamental problems: Information retrieval must have a more personalized character with the system being able to leverage context information. A search for a JDBC tutorial should be treated differently if it is issued by a software professional or by an undergraduate student. The interface into the system should be flexible and not only allow a fixed set of attributes to look for. Finally, the system should not be restricted to only one source of information in order to avoid the problems mentioned with a Yahoo-like approach.

## 2. Points of interest

### From keywords and taxonomies to ontologies

We think that there are two aspects of using keywords and taxonomies that are noteworthy. The first point is that it is crucial that every user of a metadata-based content management system shares the same interpretation of the taxonomy terms and the keywords. This is a noteworthy point even if it is implicitly clear to us, since natural language is used. Secondly, a content management system could be viewed as an application that uses a very simple ontology about categories, subcategories, keywords, documents, etc. Document management and retrieval systems implicitly share this ontology and implement it in the application logic..

It is quite clear that a knowledge supported retrieval system is only useful if it bases on a large ontology and if it has access to a large base of tagged content. We believe that this ontology does not have to provide a deep understanding of the domain being searched. After all we want to build a smart librarian that retrieves a document with the answer, not an expert that produces an answer. The "shallow" nature of the ontology should enable the integration of information from several external sources. For instance, an ontology on Java by Sun Microsystems could be used by an online learning system on distributed applications.

From our point of view, it is better to sacrifice some level of detail and expressiveness in ontologies if it makes them easier to integrate and more useable, rather than having islands of complex ontologies for isolated applications. The lessons we will learn from integrating simple ontologies will then also be the foundation for more complex endeavors.

### Agent scenario

Today's search engines work in a brute force fashion. Let us look at how we obtain information in our daily lives: an

important aspect could be characterized under the term "ask the expert". We know that Jim is the database guru in our company, therefore he could probably point us to a good tutorial on JDBC. What is important here is that Jim also knows me, thus he knows which level of difficulty would be appropriate. If someone encounters a software setup problem while working on a term project, the right person to ask would probably be an experienced computer user who is taking the same class. Chances are that this person has already encountered and solved the same problem. We would probably also find the relevant information on the Internet, but in such a personalized environment, the search precision is much higher. We believe that relevant context information such as the user's background and experience can be exchanged in a flexible way using semantic web techniques.

People usually engage in a conversation where the experts tries to find out more. A user saying: I am having trouble setting up software X might prompt the expert's question: which operating system are you working on? Finally this interaction is ended with the expert providing an answer, pointing to a document or another expert, or saying: I don't know. Obviously these processes are very complex and several quite fuzzy heuristics are involved in every step. However, we feel that even a partial mapping of this model to a search agent system will solve many of the shortcomings of traditional systems.

**Integration of information services**

Another interesting point in this agent based information retrieval example is the integration of traditional information systems. The information which student is enrolled in which course is probably stored in a university ERP system. Data from the ERP system can be highly relevant when students search for classes. Work done in the area of semantic description of web services will be a valuable foundation for our application example.