# ENRICHING ONTOLOGIES FOR IMPROVED ACCESS TO WEB DOCUMENTS*

Thorsten Kurz Kilian Stoffel
*University of Neuchâtel, IIUN*
*Pierre-à-Mazel 7, 2001 Neuchâtel, Switzerland*

**ABSTRACT**

The complex hierarchies of ontologies make it often difficult for human users to locate classes within them. We present in this paper a method to enrich classes of an existing light-weight ontology with additional keyword attributes that are learned from instances of these classes. Further we show a method to use these attributes for improving the recall of queries and for facilitating access to the classes in a result set after a query.

**KEYWORDS**

Semantic Web, DAML, Ontologies, Classification, Navigation

## 1. INTRODUCTION

Ontologies play a key role in the Semantic Web [Hendler 2001] effort and the efficient annotation and classification of data is a crucial for its success. Ontologies are used for storing and for retrieving information: First, during the creation of the meta data for a document. This process can be seen as a classical classification process. For each document the subsuming class has to be found. Second, when making queries to a document repository using an ontology, the classes satisfying the criteria defined by the query have to be found and then the instances (documents) of these classes have to be returned. Ideally both types of classification are performed automatically. However automatic classification is still prone to errors. Therefore there is still a need for interacting with a user in order to fulfil these classification tasks. With the drastically increasing amount of electronically available data, this process has to be made as efficient as possible to keep data repositories up to date.

In this paper we will present a framework that improves the user interaction with document repositories that use ontological annotations. Throughout this paper we use the Open Directory Project (ODP) as an actual example for problems with large ontologies and for how our methods improve the access to such ontologies.

ODP is basically a light-weight ontology in the form of a subclass taxonomy. The ODP ontology relies on a collaborative effort of more than over 50.000 human editors. The resulting ontology might be ideal from the point of view of an individual editor but there is no guaranty that somebody else will share this point of view and be able to find a class by browsing down step by step starting at the root.

Browsing down to a class in an ontology consists of making a series of selections, each of those being a further restriction on the set of remaining classes. On the top level of the ODP the world is split into the classes like: *Art*, *Business*, *Computers*… After choosing of one of these classes, i.e. *Computers*, we find on the next level all classes, that are – according to the editor - subsumed by the concept *Computers*: *Algorithms*, *Artificial Intelligence*,… Further, more restrictive classes are selected until the selection has been narrowed down to the searched class. Thereby one has to follow the path that is imposed by the editors of the ontology, there is no place for personal shortcuts or alternative paths. This applies not only to the ODP but also to other hierarchical navigation structures such as e.g. in Yahoo.

* This work was supported by the Swiss National Research Foundation: grant number 5003-057750/1

In order to improve the navigation we propose to enrich the original classes of the base ontology and to create a virtual ontology that contains classes and attributes extracted from the content of the documents classified under the base ontology. We tested our approach be creating a system that applies our approach to one of the top classes of the OPD base ontology namely the business class.

The rest of the paper has the following structure: In the following Section 2 our approach to enrich ontologies with additional attributes, in Section 3 we discuss the application of such enriched ontologies, and we finish in Section 4 with the conclusion.

# 2. ENRICHING ONTOLOGIES

## Ontology Languages and the ODP Ontology

There exist two major research initiatives that promoted the investigation of the applicability of ontologies to the WWW: On-To-Knowledge [Decker 1999, Fensel 2001] and DAML [Hender 2000] have been very successful and have a major impact on the way ontologies have to be seen these days. In the DAML Darpa Project a new language was defined called DAML+OIL [DAML+OIL 2001]. It builds on earlier W3C standards such as RDF and RDF Schema [RDF], and extends these languages with richer modelling primitives as commonly found in frame-based languages. This is the language we use for modelling and representation of our enriched ontology. With respect to our base ontology, we rely on the RDF version of the Open Directory Project (ODP) [ODP], that consists basically of classes that have class names and that are interrelated through subclass relations.

## Enriching by adding Attributes

Our goal of making ontologies more suitable and easier accessible for classification implies that their classes should become easier to find and easier to disambiguate. One way to achieve this is by supporting both, direct lookup of a class, and browsing through relations [Stoffel 1998].

In the ODP ontology there exists initially only one attribute per class, the class name. The class names are composed of the names of the parent classes plus its own name and reflect a path in the hierarchy. All components of the name are meaningful and necessary for the identification of the class as e.g.:

`Business/Industries/Manufacturing/Materials/Nanomaterials`

We introduced a new attribute “keywords” for collecting characterical keywords that can be used as synonym for locating a class and as a disambiguation criterium.

The initial values for the keyword attribute were taken from the components of the class names. Albeit this being a useful start set, these keywords are far from complete in characterising the classes, which is why we aimed for enriching our ontology with additional keyword from external sources. Since the ODP is already in use for classification and provides both, classes and instances (documents), it is possible to use the documents as training data to learn more about the corresponding classes.

In our system we used a crawler to retrieve instances, i.e. web documents, that were classified under the ODP business classes, and extracted keywords from these documents. As strategy for extracting this information we restricted ourselves to the <META> tags in these documents. We observed that the keywords in the <META> tag are used to a large extend correctly, however there are also misplaced or even misused keywords present. Therefore we filtered the keywords and suppressed all keywords whose frequency was below the average frequency of all the keywords for a class. The remaining keywords were added as values of the keyword attribute to the corresponding classes.

# 3. APPLICATION OF AN ENRICHED ONTOLOGY

In our system we use the keywords attribute of the enriched ODP ontology for retrieving classes matching a query and for manual disambiguation in case that there a several classes in the result set. Instead of displaying them as a flat list, the classes of the result set are presented in a hierarchical structure that is

obtained by using the original ontology, and keeping only those branches of the hierarchy that contain a class of the result set. This is especially useful for large result sets with more than 10 classes.

This means for selecting a specific class one still has to start at the root node of a ontology and then follow the subclass relations down to the class. And such a path can still be rather long and imply difficult decisions at each node, since it is often hard to predict which further subclasses will hide behind a class.

However it is often possible for a human user to describe or at least to recognize characteristic keywords of a class. Thus the set of the keywords attributes from the classes in the result set can be used to specify step by step a virtual class by adding keywords, that will converge finally to one of the classes of the result set. Attributes can be used to select either a set of all classes containing them or a set of all classes not containing them. Opposed to hierarchical browsing, where a choice affects only the direct subclasses of the current class in the hierarchy, here a choice is influencing the whole result set and therefore also all subclasses at every level and branch of the hierarchy. That means inclusions or exclusions of keywords have a bigger impact on the size of the remaining result set than simple subclass selections.

For this secondary navigation structure only distinctive keywords are used, i.e. keywords that are present in some of the classes of the remaining result set, but not in all of them. After each inclusion or exclusion the remaining result set is narrowed down and the list of keywords is updated accordingly.. (Cf. Fig. 1). This method allows to "jump" often several levels with a single selection and thus simplifies navigation considerably.

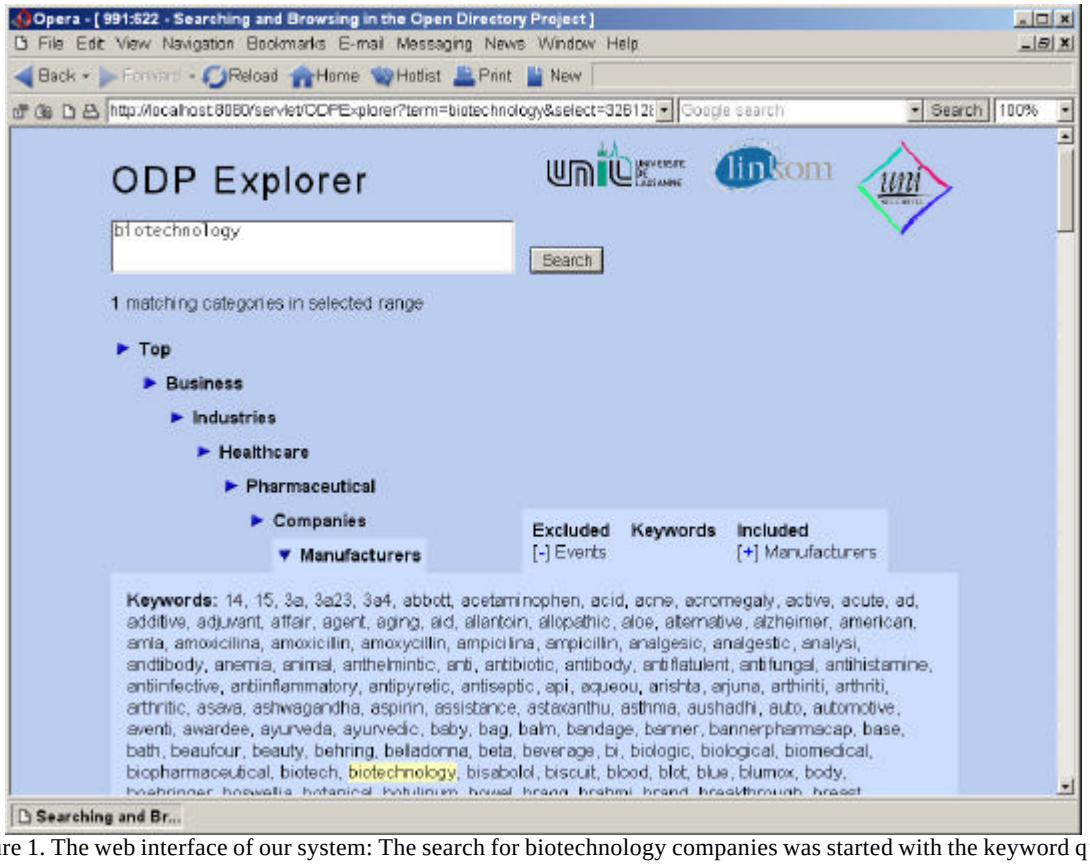


Figure 1. The web interface of our system: The search for biotechnology companies was started with the keyword query for "biotechnology". The following exclusion of the keyword "Events" and the inclusion of "Manufactures" allowed for a fast and efficient access to the searched class. In the middle left of the screen the traditional hierarchical access path is visible in parallel. Below is the list of keywords that were added to the class.

The enhancements that we have described above are maintained in parallel to the original ontology. The original ontology serves as the common ground on which knowledge can be exchanged with other systems or agents, while the enhancements are part of a virtual ontology that exists on top of the original ontology. The content of the virtual ontology provides an improved access to the underlying ontology, but does not to appear or infer with the knowledge exchange with other systems. The virtual ontology is maintained together

with the original ontology in a light-weight knowledge base [Stoffel 1999] that we have created for this purpose.

## 4. CONCLUSION

In our system we have first extended the ODP ontology, then added content learned from documents on the web and finally successfully used this knowledge to provide an improved access to the classes of the original ontology. Our tests have shown that through the parallel navigation, which allows the inclusion or exclusion of attributes and thus constructing a virtual concept, the manual selection and disambiguation process can be greatly simplified. Often a large result set that is distributed over many distant branches in the ontology can be reduced efficiently by the inclusion or exclusion of specific keyword attribute, which is especially useful, if the conventional taxonomic navigation provided either too many choices at once or the parent class names did not allow to conclude on their underlying subclasses. However in some cases when there are to many unique attributes present in the result set, the secondary navigation structure contains too much information and looses thereby its usefulness. We are addressing this issue by our ongoing work on a better algorithm for selection the most relevant and distinctive attributes for the remaining classes in the result set.

## ACKNOWLEDGEMENTS

We would like to thank Adrien Battistolo and André Lang from Linkom for implementing the crawler and providing the additional keywords for the ODP classes.

## REFERENCES

DAML+OIL, 2001. Reference description of the daml+oil (march2001) ontology markup language. *http://www.daml.org/2001/03/reference.html.*

Decker, S. et al., 1999. Ontobroker: Ontology based access to distributed and semi-structured information. In R. Meersman et al., editor, DS-8: *Semantic Issues in Multimedia Systems*. Kluwer Academic Publisher, 1999.

Fensel, D., 2001. *Ontologies: Silver Bullet for Knowledge Management and Electronic Commerce*. Springer-Verlag, Berlin.

Hendler, J., 2000. DARPA Agent Mark Up Language (DAML), Available: *http://www.darpa.mil/iso/DAML/DAML_ISO_World.ppt* .

Hendler, J. et al., 2001, The Semantic Web: When the Internet gets Smart, *Scientific American May 2001*

ODP Open Directory Project: *http://www.dmoz.org*

RDF *http://www.w3.org/RDF/*

Stoffel, K. et al. 1998. "Query Building Using Multiple Attribute Hierarchies". *Proc. 1998 AMIA Annual Fall Symposium.*

Stoffel, K. and Hendler, J. 1999. "PARKA-DB: Back-End Technology for High Performance Knowledge Representation Systems ". I*EEE Expert: Intelligent Systems and Their Applications Special Issue on The Use of Ontologies*.