Ontobroker: The Very High Idea

Dieter Fensel, Stefan Decker, Michael Erdmann, and Rudi Studer

University of Karlsruhe, Institute AIFB, 76128 Karlsruhe, Germany
Email: {fensel, decker, erdmann, studer}@aifb.uni-karlsruhe.de,
http://www.aifb.uni-karlsruhe.de/WBS/broker

Abstract

The World Wide Web (WWW) is currently one of the most important electronic information sources. However, its query interfaces and the provided reasoning services are rather limited. Ontobroker consists of a number of languages and tools that enhance query access and inference service of the WWW. The technique is based on the use of *ontologies*. Ontologies are applied to annotate web documents and to provide query access and inference service that deal with the semantics of the presented information. In consequence, intelligent *brokering* services for web documents can be achieved without requiring to change the semiformal nature of web documents.

Introduction

The World Wide Web (WWW) contains huge amounts of knowledge about almost all subjects you can think of. HTML documents enriched by multi-media applications provide knowledge in different representations (i.e., text, graphics, animated pictures, video, sound, virtual reality, etc.). Hypertext links between web documents represent relationships between different knowledge entities. Based on the HTML standard, browsers are available that present the material to users and use the HTML-links to browse through distributed information and knowledge units. However, retrieving information from the web is only weakly supported. Actually, the main query answering services the web provides are keyword-based search facilities carried out by different search engines, web crawlers, web indices, man-made web catalogues etc.

(Luke et al., 1997) propose *ontologies* to improve the query answering support of the "knowledge base" WWW. Ontologies are discussed in the literature as a means to support knowledge sharing and reuse (cf. Friedman Noy & Hafner, 1997). This approach to reuse is based on the assumption that if a modeling scheme—i.e. an *ontology*—is explicitly specified and agreed upon by a number of agents, it is then possible for them to share and reuse knowledge. We use the *metaphor of a newsgroup* to define the role of such an ontology. It is used by a group of people who share a common subject and a related point of view on this subject. Thus it allows them to annotate their documents to provide an *intelligent brokering service* that enables informed access to their web documents.

We designed and implemented some tools necessary to enable the use of ontologies for enhancing the web. We developed a broker architecture called Ontobroker with three core elements: a query interface for formulating queries, an inference engine used to derive answers, and a webcrawler used to collect the required knowledge from the web. We provide a *representation* language for formulating ontologies. A subset of it is used to formulate queries, i.e. to define the *query language*. An *annotation* language is offered to enable knowledge providers to enrich web documents with ontological information. The strength of our approach is the tight coupling of informal, semiformal and formal information and knowledge. This supports their maintenance and provides a service that can be used more generally for the purpose of *knowledge management* and for integrating knowledge-based reasoning and semiformal representation of documents.

This paper is organized as follows. First we provide the motivation for our approach. Then we sketch the languages and tools used to represent ontologies, formulate queries, and annotate web documents with ontological information. Finally a discussion of the possibilities and limitations of Ontobroker as well as related work and conclusions are given.

The Bottlenecks of the WWW

The WWW provides huge amounts of information in informal and semi-structured representations. This is one of the key factors that enabled its incredible success story. The representation formalisms are simple and retain a high degree of freedom in how to present the information. In consequence, we strictly follow the basic design paradigm. However, freedom in information representation and simple representation formalisms cause serious bottlenecks in accessing information from the web. Basically there are two different search techniques available at the moment: human browsing through textual and graphical representations following hyperlinks and keyword based search engines that retrieve further hyperlinks for this browsing process. The query answering and inference service of the WWW is very limited when compared to relational or deductive databases that enable precise queries and inference service for deriving new knowledge. In the following we will discuss some examples that illustrate limitations of current WWW access.

- Imagine that you want to find out about the research subjects of a researcher named *Smith* or *Feather*. Consulting a search engine will result with a huge set of pages containing the key word *Feather*. Preciseness, recall, and presentation are limited. Even if the pages of

the person are identified it requires a significant human search effort to investigate these pages until the page that contains the required information has been found.

- The format of query responses is a list of hyperlinks and textual and graphical information that is denoted by them. It requires human browsing and reading to extract the relevant information from these information sources. Remember, we were looking for the research subjects of Mr. *Feather*. We would like to get a list of research topics like: "*World Wide Web, Ontologies, Knowledge Acquisition, Software Engineering*". However, it requires further human extraction to retrieve this information. This burdens web users with an additional loss of time and seriously limits information retrieval by automatic agents that miss all common sense knowledge required to extract such informations from textual representations.
- Still, the above mentioned problems are rather trivial compared to queries that refer to the content of several pages. Imagine that you want to find the research subjects of a research group. You have to figure out whether this is written on a central page or whether each researcher enumerates them on his pages. Then you have to determine all members of this research group and go through all their pages. The required search effort and lack of recall make such queries impractical for a large, distributed and heterogeneous groups of people (i.e., web sources).
- Finally, each current retrieval service can only retrieve information that is represented by the WWW. This sounds trivially true, but it significantly limits query answering capability. Imagine that *Feather* writes on his homepage that he cooperates with another researcher *E. Motta* on investigating formal specifications of problem-solving methods. However, you will completely miss this information for *E. Motta* if he does not repeat the information (with the reverse direction) on his homepage and you are only consulting his page. However, an answering mechanism that can make use of the implicit symmetry of cooperation could provide you with this answer.

Summing up our discussion we identify the issues that we will improve with our approach: We use semantic information for guiding the query answering process; enable answers with a well-defined syntax and semantics that can directly be understood and further processed by automatic agents or other software tools; enable a homogeneous access to information that is physically distributed and heterogeneously represented in the WWW; provide information that is not directly represented as facts in the WWW but which can be derived from other facts and some background knowledge. Subsequently we will discuss the different languages and tools that are provided for this purpose.

The Query Interface

The query formalism is oriented toward a frame-based representation of ontologies that defines the notion of

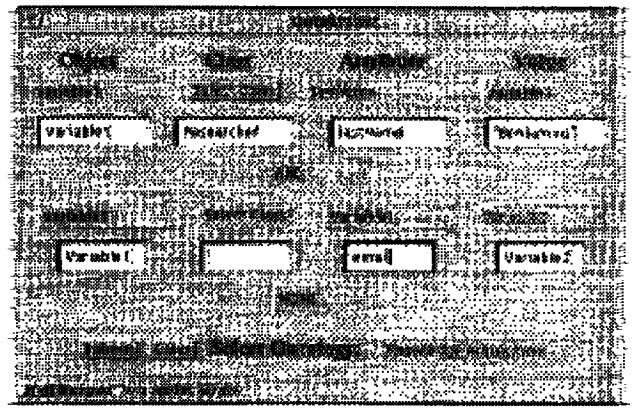

Fig. 1 The advanced query interface.

instances, classes, attributes and values. The generic scheme for this is

O:*C*[*A*->>*V*]

meaning that the object *O* is an instance of the class *C* with an attribute *A* that has a certain value *V*. The structure of the query language can be exploited to provide a tabular query interface as shown in Figure 1 which asks for the researchers with last name *Benjamins* and their email addresses. The result is shown in Figure 2.

Ontobroker can also be used to collect distributed information. The query in Figure 3 collects all research topics of the members of the research group on knowledge-based systems at the Institute AIFB, i.e. it retrieves the research topics of a research group that are distributed at the different homepages of the researcher.

Another possibility is to query the knowledge base for information about the ontology itself, e.g. the query

FORALL *Att*, *T* <- *Researcher*[*Att*=>>*T*]

asks for all attributes of the class *Researcher* and their associated classes.

We also need support for selecting classes and attributes from the ontology. To allow the selection of classes, the ontology has to be presented in an appropriate manner. Usually a ontology can be represented as a large hierarchy of concepts. In regard to the handling of this hierarchy a user has at least two requirements: first he wants to scan the vicinity of a certain class looking for classes better suitable to formulate a certain query. Second a user needs an overview over the whole hierarchy to allow an quick and

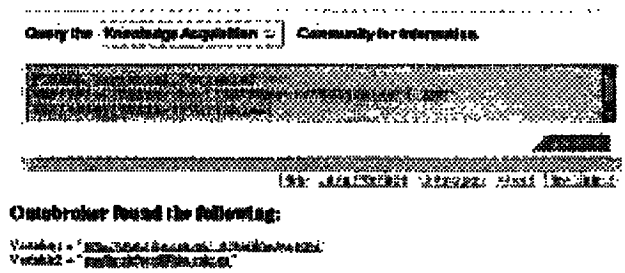

Fig. 2 The result of a query.

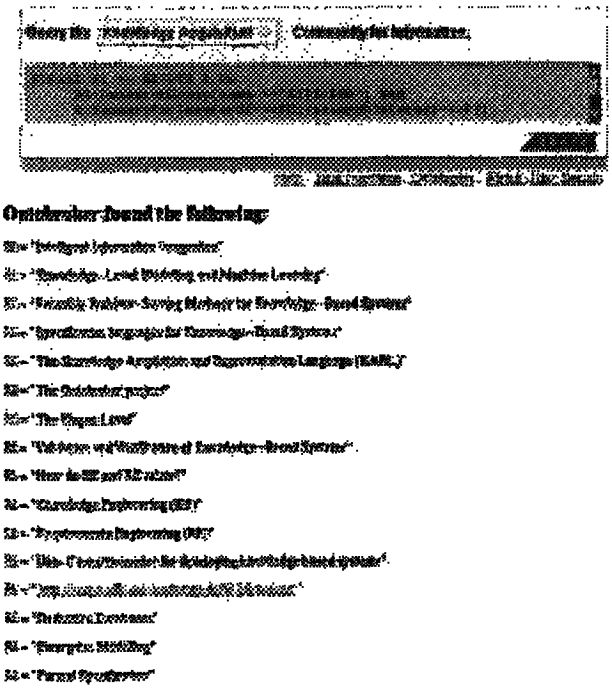

Fig. 3 The result of a query.

easy navigation from one class in the hierarchy to another class. These requirements are met by a presentation scheme based on Hyperbolic Geometry (Lamping, Rao & Pirolli, 1995): classes in the center are depicted with a large circle, whereas classes at the border of the surrounding circle are only marked with a small circle (see Figure 4). The

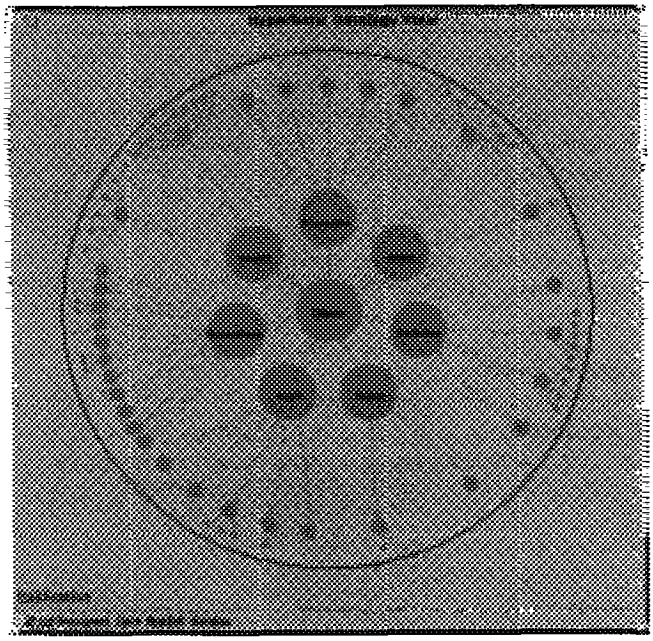

Fig. 4 The hyperbolic ontology view.

visualization techniques allows a quick navigation to classes far away from the center as well as a closer examination of classes and their vicinity. When a user selects a class from the hyperbolic ontology view, the class name appears in the class field and the user can select one of the attributes from the attribute choice menu because the pre-selected class determines the possible attributes. Based on these interfaces Ontobroker automatically derives the query in textual form and presents the result of the query (see Figure 2 and Figure 3).

The Ontology

The basic support we want to provide is query answering about elements of an ontology. The ontology may be described by taxonomies and rules. Since there are effective and efficient query evaluation procedures for Horn-logic like languages we based our inference engine on Horn logic. However, simple horn logic is not appropriate from an epistemological point of view for two reasons:

- First, the epistemological primitives of simple predicate logic (of which Horn logic is a subset) are not rich enough to support adequate representations of ontologies. Usually, ontologies are defined via concepts or classes, is-a relationships, attributes, further relationships, and axioms. Therefore an adequate language for defining the ontology has to provide modeling primitives for these concepts. *Frame-Logic* (Kifer, Lausen & Wu, 1995) provides such modeling primitives and integrates them into a logical framework providing a Horn logic subset. Furthermore, in contrast to most Description Logics, expressing the ontology in Frame-Logic allows for queries that directly use parts of the ontology as first class citizens. That is, not only instances and their values but also concept and attribute names can be provided as answers by means of variable substitutions.
- Second, it is often very artificial to express logical relationships via Horn clauses. We provide a much richer language to define complex expressions and internally translate them via the Lloyd-Topor transformation into normal logic programs. Examples of non Horn rules are given in Table 1.

The concept hierarchy consists of elementary expressions declaring subclass relationships. The attribute definitions declare attributes of concepts and the valid types that a value of an attribute must have. The rule in table 1 ensures that whenever a person is known to have a publication then the publication also has an author who is the particular person and vice versa. This kind of rule completes the knowledge and frees a knowledge provider to provide the same information at different places reducing development as well as maintenance efforts.

The Annotation

Knowledge contained in the WWW is generally formulated

using HTML. Therefore, we developed an extension to the HTML syntax to enable the ontological annotation of web pages.[1] We will only provide the general idea. More details can be found at http://www.aifb.uni-karlsruhe.de/WBS/broker. The idea behind our approach is to take HTML as a starting point and to add only few ontologically relevant tags. With these minor changes to the original HTML pages the knowledge contained in the page is annotated and made accessible as facts to the Ontobroker. This approach allows the knowledge providers to annotate their web pages gradually, i.e. they do not have to completely formalize the knowledge contained therein. Further, the pages remain readable by standard browsers like Netscape Navigator or MS Explorer. Thus there is no need to keep several different sources up-to-date and consistent, reducing development as well as maintenance efforts considerably. All factual ontological information is contained in the HTML page itself. We provide three different epistemological primitives to annotate ontological information in web documents:

- An object identified by an URL (Uniform Resource Locator) can be defined as an instance of a certain class.
- The value of an object's attribute can be set.
- A relationship between two or more objects may be established.

Discussions of the Approach

Providing information and knowledge via the Ontobroker requires two time-consuming activities: designing an ontology and annotating web documents.

Designing ontologies is a time consuming activity because it aims for a formal and consensual model of some aspect of reality. However, building such a model pays off in several dimensions beyond merely improving the web presentation of documents. It can be used by companies and organizations as a reference model for their internal data and information (cf. Kühn & Abecker, 1997). It can be used by standardization committees to establish standards for representing information about some areas (Benjamins & Fensel, 1998). Therefore, such effort pay back beyond simple improving web accessibility.

Annotating web documents with ontological information is much easier. A trained person with some basic HTML knowledge is able to annotate ca. five pages an hour (ca. one thousand per month). Still, we would like to provide a more sophisticated tool that supports this process. Currently, annotations have to be written with text editors. However, as for the query interface one could make use of a graphical representation of the ontology and use it for a click-and-paste process in producing annotation. Another possibility for stable web sources is to replace the annotation effort by deriving wrappers which extract this information (cf. Ashish & Knoblock. 1997). Such a wrapper can be used to directly derive the factual knowledge that is used by the inference engine of Ontobroker. In this scenario a wrapper replaces the annotation process and the process of translating annotations into facts.

Finally, we decided to design our annotation language as a minor extension of HTML because most documents on the web use this formalism. However, the W3C is currently developing the *resource description framework* (RDF). This format can be used to add meta information to documents, i.e. to include semantical information about documents. That approach shows a number of similarities with Ontobroker because both approaches aim at machine-readable content information and enable automated processing of web resources. However, in Ontobroker the annotation information is tightly integrated into HTML. This reduces redundancy of information on a web page to a minimum. Meta data defined in RDF have to be provided on an extra page or en bloc inside of a web-page. Therefore, elements from a web page like text fragments or links cannot directly be annotated with semantics. These elements must be repeated so that they can be enriched with meta-information. This design decision may cause significant problems for maintaining web documents due to the redundancy of the information. However, when a final version of RDF is recommended by the W3C it will be an easy task to implement a wrapper that automatically generates RDF definitions from annotations in Ontobroker. Therefore, we will join this standard enabling other agents to read our meta information. Generating automatically RDF descriptions makes the annotated knowledge available to agents and brokering services that search the web for information. That is, this knowledge may not only be used

[1] We did not make use of the *extensible Markup Language (XML)* to define our annotation language because many existing HTML pages are not well-formed XML documents and because the required extension is minor.

Table 1. Some Ontology Definitions

Concept Hierarchy	Attribute Definitions	Rules

Object[].
Person :: Object.
Employee :: Person.
AcademicStaff :: Employee.
Researcher :: AcademicStaff.
Publication::Object.
	Person[
	firstName =>> STRING;
	lastName =>> STRING;
	eMail =>> STRING;
	...
	publication =>> Publication].
	Employee[
	affiliation =>> Organization;
	...].
		FORALL Person1, Publication1
		Publication1:Publication
		[author ->> Person1]
		<->
		Person1:Person
		[publication ->> Publication1].

by Ontobroker to answer direct questions of a human user but it will also be available for all automated search mechanisms that can read RDF and can make use of an ontology (cf. Ambite & Knoblock, 1997).

Providing automated access for other search agents is essential because we view Ontobroker only as a first step into the direction of a *knowledge web*. Establishing several of such brokering services with different ontologies providing semantical descriptions of their information contents requires to free the clients from directly contacting these brokers. Instead he will make use of a customized search agent that consults the different brokering services for query answering. The ontology used by a broker is its *competence description*, i.e. it clarifies the topics it can provide knowledge about. The human user may only want to contact a final and small selection of brokering services that are returned by its personalized search agent.

Conclusions and Related Work

In this paper we introduced methods and tools for enhancing the Web. We proposed ontologies as a means to annotate WWW documents. Ontobroker includes a query interface for formulating queries, an inference engine for deriving answers to the posed queries, and a web crawler for searching through the various subnets and translating the ontological annotations into facts for the inference engine. Ontobroker is the basis for realizing the Knowledge Acquisition Initiative $(KA)^2$ (Benjamins & Fensel, 1998) and for developing a knowledge management system for industrial designers in regard to ergonomic questions. In the latter project, the same knowledge may be used by users, i.e. industrial designers, and as input and output for inference processes of the system. This twofold use of the same piece of knowledge is enabled through the tight coupling of semiformal and formal knowledge in Ontobroker.

The approach closest to ours is SHOE, which introduced the idea of using ontologies to annotate information in the WWW (Luke et al., 1997). HTML pages are annotated via ontologies to support information retrieval based on semantic information. However, there are major differences in the underlying philosophy: In SHOE, providers of information can introduce arbitrary extensions to a given ontology. Furthermore, no central provider index is defined. As a consequence, when specifying a query the client may not know all the ontological terms which have been used to annotate the HTML pages and the web crawler may miss knowledge fragments because it cannot parse the entire WWW. Thus the answer may miss important information and the web crawler may miss knowledge bits. In contrast, Ontobroker relies on the notion of an *ontogroup* defining a group of Web users who agree on an ontology for a given subject. Therefore, both the information providers and the clients have complete knowledge of the available ontological terms. In addition, the provider index of the Ontocrawler provides a complete collection of all annotated HTML pages. Thus, Ontobroker can deliver complete answers to the posed queries. The philosophy of Ontobroker is also tailored to homogeneous intranet applications, e.g. in the context of knowledge management within an enterprise. SHOE and Ontobroker also differ with respect to their inferencing capabilities. SHOE uses description logic as its basic formalism and currently offers rather limited inferencing capabilities. Ontobroker relies on Frame-Logic and supports rather complex inferencing for query answering.

One can situate Ontobroker in the general context of approaches that support the integration of *distributed* and *heterogeneous* information sources using a *mediator* (Wiederhold & Genesereth, 1997) that translates user queries into sub-queries for the different information sources and integrates the sub-answers. Wrappers and content descriptions of information sources provide the connection of an information source to the mediator. However, most of these approaches assume that the information sources have a stable syntactical structure that a wrapper can use to extract semantic information. Given the heterogeneity of any large collection of web pages, this assumption is often not fulfilled. However, wrappers and annotation-based approaches are complementary.

Acknowledgments. We thank Jürgen Angele, Richard Benjamins, and Asun Gomez-Perez for their cooperations.

References

Ambite, J.L. and Knoblock, C.A. 1997. Agents for Information Gathering. *IEEE Expert, Sept*ember/October.

Ashish, N. and Knoblock, C.A. 1997. Semi-automatic Wrapper Generation for Internet Information Sources. In *Proceedings of the IFCIS Conference on Cooperative Information Systems (CoopIS)*, Charlston, South Carolina.

Benjamins, V. R. and Fensel, D. 1998. Community is Knowledge! In $(KA)^2$. In *Proceedings of the 11th Banff Knowledge Acquisition for Knowledge-Based System Workshop (KAW'98)*, Banff, Canada.

Friedman Noy, N. and Hafner, C.D. 1997. The State of the Art in Ontology Design. *AI Magazine*, 18(3):53—74.

Kifer, M., Lausen, G., and Wu, J. 1995. Logical Foundations of Object-Oriented and Frame-Based Languages. *Journal of the ACM*, 42.

Kühn, O. and Abecker, A. 1997. Corporate Memories for Knowledge Management in Industrial Practice: Prospects and Challenges. *Journal of Universal Computer Science, Special Issue on Information Technology for Knowledge Management*, Springer Science Online, *3(8)*.

Lamping, L., Rao, R., and Pirolli, P. 1995. A Focus+Context Technique Based on Hyperbolic Geometry for Visualizing Large Hierarchies. In *Proceedings of the ACM SIGCHI Conference on Human Factors in Computing Systems.*

Luke, S., Spector, L., Rager, D., and Hendler, J. 1997. Ontology-based Web Agents. In P*roceedings of First International Conference on Autonomous Agents.*

Wiederhold, G. and Genesereth, M. 1997. The Conceptual Basis for Mediation Services. *IEEE Expert*, September/ October, pp. 38—47.