Journal of Universal Computer Science, vol. 9, no. 7 (2003), 682-696
submitted: 3/2/03, accepted: 21/7/03, appeared: 28/7/03

SemanticMiner – Ontology-Based Knowledge Retrieval[1]

Eddie Moench
ontoprise GmbH,Amalienbadstr. 36, D-76227 Karlsruhe
moench@ontoprise.de

Mike Ullrich
ontoprise GmbH, Amalienbadstr. 36, D-76227 Karlsruhe
mike.ullrich@ontoprise.de

Hans-Peter Schnurr
ontoprise GmbH, Amalienbadstr. 36, D-76227 Karlsruhe
schnurr@ontoprise.de

Juergen Angele
ontoprise GmbH, Amalienbadstr. 36, D-76227 Karlsruhe
angele@ontoprise.de

Abstract: During the analysis of knowledge processes in enterprises it often turns out that simple access to existing enterprise knowledge which is covered in documents is not possible. To enable access to a company's document and data stocks *Information Retrieval (IR)* technologies play a central role. In the following we describe the underlying theory of the SemanticMiner system, including methods and technologies as well as continuing approaches to obtain *Knowledge Retrieval (KR)* by dint of semantic technologies.
Key Words: information retrieval, knowledge management, knowledge representation, ontology, logic
Category: E.1, H.3.0, H.3.3, I.2.0, I.2.1, I.2.3, I.2.4

1 Definition and Scope of Ontologies

Within this chapter will give a general definition and our interpretation of ontologies. Therefore we will discuss the scope of this technology and of adjacent technologies. We credit the ongoing, unfortunately tergiversating discussion and the divergence of the standardization efforts of ontologies and Topic Maps.

1.1 From Lightweight Semantics to Ontologies

As illustrated in figure 1 we will explain the three main roots of the evolutionary tree of which ontologies have been evolved. Furthermore we will show how and

[1] Part of this work was carried out within the EC sponsored (grant IST-2001-34038) dot.kom project (http://www.dot-kom.org).

why logic programs on top of ontologies give wings to the knowledge representation model to climb another evolutionary step.

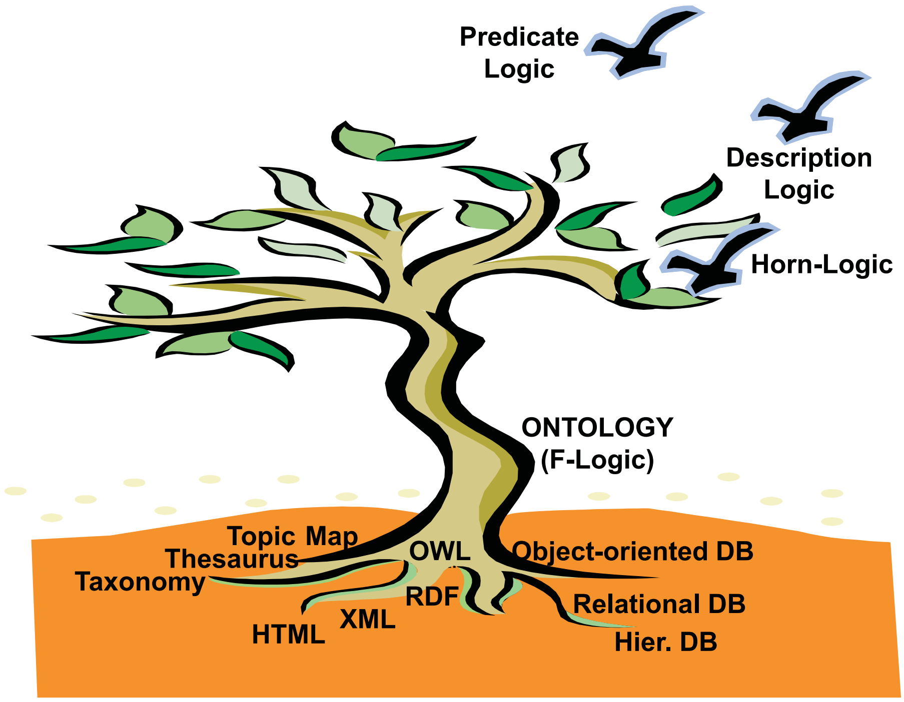

Figure 1: Evolution towards Ontology

1.1.1 Taxonomy

Definition 1 Taxonomy. Taxonomy is a hierarchy of terms [Bru01].

Even ancient biologists tried to categorize flora and fauna. The most famous Software, which uses Taxonomies, is the Windows (File) Explorer from Microsoft.

1.1.2 Thesaurus

Thesaurus originated from the library domain. It represents a terminology to a certain domain. Apart from the hierarchy there exist a fixed set of predefined relations between the objects: e.g. similarity and synonymy. Microsoft Word's thesaurus for different languages is it's most well-known application.

1.1.3 Topic Map

Topic Map is an ISO standard on XML basis.

Definition 2 Topic Map. A Topic Map essentially consists of topics (abstract things), associations, scopes (ranges of validity for Topics) and assigned documents outside of the Topic Map (occurrences) (c.f. [Hof01]).

Topic Maps are offered by large number of vendors. Most well-known applications exist within the area of information retrieval, visualization and navigation. The standard only describes the structure of the Topic Map. Neither a common data model nor a standard query language is defined. Query languages and many extensions were individually realized by vendors.

1.1.4 Ontology

First usage of the term ontology was by Aristotle, meaning as much as the science of being.

Definition 3 Ontology. In computer science we define ontology as "an explicit specification of a (shared) conceptualisation" [Gru93].

This definition is quite general. We will extend and specify our interpretation of ontology later. At this point we would like to stress, that the ontology is the most expressive model discussed so far. All features of taxonomies, thesauri and Topic Maps can be expressed in ontologies.

In order to transfer a Topic Map into an ontology, the hierarchy has to be checked whether it is a true inheritance hierarchy ("is-a" instead of "has-part"). Has-part relationships can be expressed via a relation between two concepts. Some features from Topic Maps cannot be directly transformed (e.g. scopes have to be transferred into relations). Ontologies offer the possibility of separating schema (meta model) and contents, thus enabling performant mass data operations.

Additionally and probably most important, the ontology brings a powerful set of rules, which can be used to formulate mappings to other ontologies, constraints, negations and logical functions as well as mathematical operations and further functions [KLW95].

By means of the query language ontologies can be queried with the same language used for modelling in arbitrary directions. E.g. which are the sub concepts of person? Which company offers which products? Which persons over 30 years know about a certain topic?

1.2 Web Representation Languages

The standardization of web technologies is driven mainly by the World Wide Web Consortium (W3C).

1.2.1 HTML

Hypertext mark-up language was invented in the early 90ties by Tim Berners-Lee (et al.), now head of the W3C. Hypertext is a presentation language, with the possibility to be displayed on any system and with hyperlinks connecting other HTML-documents[2]. Yet a link does not have a meaning. The problem with HTML is that the information provided is not machine processable. It's like a color-fax, which can only be read and interpreted by humans.

1.2.2 XML(s)

Definition 4 XML. XML is a "metalanguage which describes web data and its structure (unlike HTML, which describes how data should be presented)"[3].

Within XMLs a schema for XML can be defined. In the last year many domain specific standards based on XML have been developed[4] and XML has also become famous as configuration files for applications and state of the art applications use XML to exchange data with other applications [5].

1.2.3 RDF(s)

Definition 5 RDF(s). With RDF the semantics of data, which is expressed in XML, can be specified in a standardized and interoperable manner [Fik].

RDF statements consist of triples: a resource (is a unique resource identifier, e.g. a URL), a property (like author) and a value. These parts represent subject, predicate and object [Bra]. RDFs again is the schema for RDF.

1.2.4 DAML+OIL and OWL

DAML+OIL and OWL alike define a basic ontology vocabulary. Additionally to RDF, DAML allows to specify data types, ranges, a non-exclusive Boolean combination of classes and axioms like disjoint, inverse or transitive concepts [OO].

DAML+OIL has been developed by DARPA. Currently DAML and all other efforts have been canalized towards OWL, which again is powered by the W3C and is currently request for comments.

---

[2] http://www.ideafinder.com/history/inventions/story069.htm
[3] http://www.auburn.edu/helpdesk/glossary/xml.html
[4] http://www.xml.org
[5] Web Services, http://www.w3.org/2002/ws/

1.2.5 F-Logic Ontology

F-Logic covers most parts of OWL (see section 1.4.2) and additionally allows specifying axioms freely. E.g. you can express: "If a person writes a book, which has a topic, he is an expert for that topic." Additionally F-Logic uses the same syntactical constructs for both modelling and querying the ontology.

1.3 Database Systems

A database system aims at separating data from the application. Even though ontologies do not (yet) cover all functionalities of database systems (e.g. transaction management), they are richer in means of the underlying model. In this chapter we therefore look at the conceptual model of different databases.

1.3.1 Database Concepts

In order to understand databases, the term Entity Relationsship (ER) model has to be defined first.

Definition 6 ER-model. An ER-model consists of entities (an object, like a person), relationships (e.g. the relation between a person and a company) and attributes (e.g. haircolor) (c.f. [ERM]).

It is characteristic for databases that the schema (the column-titles) is separated from the data (the rows).

In object oriented databases the model has been extended to cover e.g. inheritance and class hierarchies. This can be useful for example, if there are groups of entities which have different attributes. E.g. only students out of persons have a matriculation number.

1.3.2 Ontologies

Coming from Object-oriented databases, ontologies add the ability of Inferencing. Inferencing means to automatically generate new facts (implicit facts), which are derived by means of logical conclusions. New facts can be discovered by the consequent usage of rules over existing facts (c.f. [MUS03]). Ontologies provide, supplementary to the support of navigation, much more powerful possibilities of modelling, which enable additional functionality for the knowledge model [SM01].

Relational databases can be imported and thus handled as "flat" ontologies, object-oriented databases could be imported as well (currently there is no tool support for this, due to the weak distribution of such systems). The ontology schema can be mapped onto a database schema or another ontology. The data of the database is the available within the ontology as instances. The database systems are queried on demand, only when the information is necessary to answer a corresponding query [MUS03].

1.4 Logic and Inferencing

In order for the Semantic Web to become true, a logic component is necessary to enable automated conclusions. So far we have discussed various approaches for representation of data, information and knowledge. Logic builds the foundation to enable execution above such models.

1.4.1 Predicate Logic

"In logic, as in grammar, a subject is what we make an assertion about, and a predicate is what we assert about the subject. When the subject of the sentence is an individual object (like Socrates in "Socrates is mortal"), then we are using first order logic. When the subject is another predicate (like being mortal in "Being mortal is tragic"), then we are using second order logic or higher order logic." (from [Sub]). In the following we will limit our discussion to first order logic.

1.4.2 Description Logic

Description Logic is a subset of Predicate Logic. It allows to specify a terminological hierarchy using a restricted set of first order logic formulas. Therefore it is well suited for modelling. The main usage of Description Logics's inferencing mechanisms is classification and subsumption[6]. Latest research proposes that it is possible and even more efficient to transfer Description Logic into Horn Logic Programms. There are only smaller parts which cannot be translated into Horn Logic, while the performance of Horn Logic systems is a magnitude better than on Description Logic systems [GHVD03].

1.4.3 Horn Logic

Horn Logic is another subset of Predicate Logic. Basically speaking Horn clauses are rules or implicational constraints. This is also the basis for the programming language PROLOG, which unfortunately doesn't come with well-founded semantics [Heg].

There is an intersection between Description Logic and Horn Logic, yet large parts of Horn Logic cannot be expressed in Description Logic and some parts of Description Logic cannot be modelled in Horn Logic.

[6] http://www.semanticweb.org/inference.html

1.4.4 F-Logic

For the IR system SemanticMiner we use the *F-Logic* (Frame-Logic) language. F-Logic is an instantiation of Horn Logic by Kifer and Lausen [KLW95].

OntoBroker [DEFS99], which serves as back-end for the SemanticMiner system, is the first commercial implementation of F-Logic, where performance issues were the most important design issue.

1.5 The omniscient-paradigm

By the utilization of an ontology one automatically accepts the "omniscient"-paradigm, which is derived from a traditional approach of cognition in social systems.

Definition 7 omniscient-paradigm. Knowledge is hereby represented and organized in only one structure, completely independent of by whom, how, where and why this knowledge was created originally. The nowadays arising approach of "distributed intelligence" is on the other hand based on the assumption, that knowledge is always and indivisibly connected with different so called contexts, like for instance individuals, groups, time periods, and places, and therefore not capable of being central organized: Accordingly, knowledge is context specific [NSB00].

It has to be mentioned as well that the user group of the aimed knowledge-based system has to be agreed on the ontology [Gru95]. By the usage of this formalism, ambiguousness will be prevented.

2 Information Retrieval

For the idea respectively the domain of information retrieval (IR), there exists no general accepted definition nor delimitation. From the historical point of view, IR has been developed to improve the (re)locating of research publications. Even if this area remains still in main focus of IR, the domain and the objects, with which IR is dealing, as well as the conceptual formulation have broadened. A description can be found at the Fachgruppe Information Retrieval of the Gesellschaft für Informatik [Fuh96]:

"Information Retrieval takes information systems into account in respect of their role they play within the process of knowledge transfer, from the human knowledge producer to the information demander". Thus the target of IR is to prepare and offer stored data (texts, structured data, pictures, facts, etc.) in a way, that they can be retrieved, regarding a concrete information need and a problem specific search strategy, in the most precise and complete way.

2.1 Quality Appraisal of IR systems: Recall and Precision

The units most commonly used for the measurement of the assessment of the goodness of IR-systems are *Recall* and *Precision.* According to these two measures, the search with an IR-system is estimated on basis of the delivered search result. The term relevance of a document servers therefore as basis.

A set of different definitions of the term relevance are to be found in [Kai93] for example. We will introduce the definition of *relevance* according to [CLvRC98]:

Definition 8 Relevance. If a user wants to retrieve a document to a query, then this document is seen to be relevant to this query.

Now, the two measures recall and precision can be defined [BYRN99]:

Definition 9 Recall. Recall constitutes the measure for the completeness of the retrieval result and is defined through the ratio of retrieved, relevant documents and the total number of available, relevant documents in the corpus.

More precise: Given is an information need I and a query q of a user. Then the recall is calculated by

$$req(q, I) = \frac{|\mathcal{R}(q, I)|}{|\mathcal{R}(I)|}, \tag{1}$$

whereas $|\mathcal{R}(I)|$ indicates the quantity of all relevant documents to the information need I and $|\mathcal{R}(q, I)|$ the quantity of documents that have been retrieved with the query q and which are relevant to the information need I (c.f. figure 2).

The range of the recall value goes from zero to one. A recall of zero is given for the worse result, whereas a recall of one is given for the best possible result.

Definition 10 Precision. Precision serves for the measurement of the accuracy of a retrieval result and as well as an indicator for the ability of an IR system not to deliver irrelevant documents. Precision is defined as the ratio of the retrieved, relevant documents to the total number of all retrieved documents.

More precise: Given is an information need I and a query q of a user. Then the precision is calculated by

$$pres(q, I) = \frac{|\mathcal{R}(q, I)|}{|\mathcal{E}(q)|}, \tag{2}$$

whereas $|\mathcal{R}(q, I)|$ indicates the quantity of all relevant documents to the information need I and $|\mathcal{E}(q)|$ the quantity of all retrieved documents (c.f. figure 2).

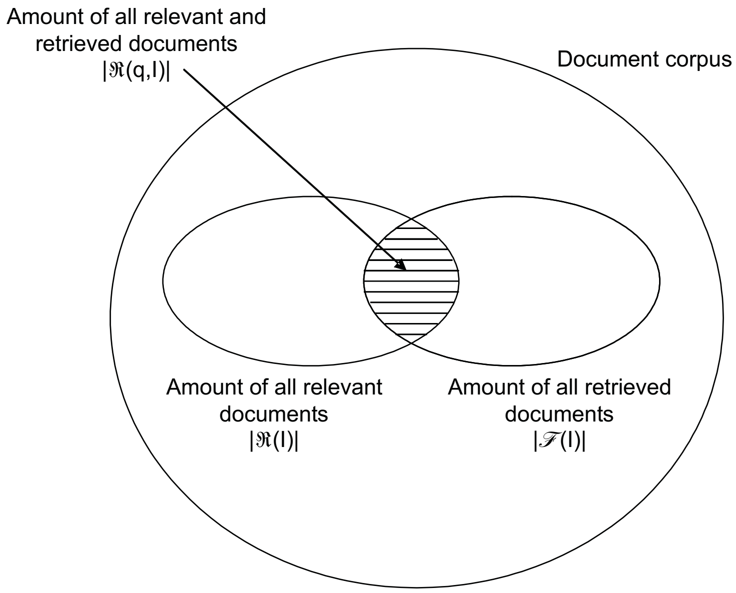

Figure 2: Recall and precision to a given sample information need

The range of the precision value goes also from zero to one. The aim is as well to maximize the value of precision.

It only makes sense looking at *both* measures. Recall for instance, leaves the number of irrelevant delivered documents unconsidered. Thus the value of recall can easily be set to 1, by returning all documents in the corpus to any query. Regarding this case, the precision value would be very low of course. The individual contemplation of precision on the other hand would tell you nothing about the completeness of the retrieval results. Precision alone could be maximized by returning only very few documents.

For a search with a high claim for completeness of the search results as well as enforcing a linear ordering of the retrieved documents – which is the case in the SemanticMiner system – , one has to focus on maximizing a high recall value. Thus we are keeping a major attention on this measure inside the SemanticMiner system (see also chapter 3.1 with more on this issue).

Average precision versus recall figures are useful for comparing the retrieval performance of distinct retrieval algorithms over a set of example queries. However, there are situations in which we would like to compare the retrieval performance of our retrieval algorithms with the individual queries. Thus single measures which combine recall and precision might be of interest (e.g. F-measure, the harmonic meaning). Nevertheless, we want to focus on both measures separately and show how ontologies can be used to improve each of these measures, which will implicit improve single measure values as well, though they will not be taken into account explicit in this paper.

3 Knowledge Retrieval — Semantic Information Retrieval

The annual proceeded TREC (Text REtrieval Conference) conferences serves as an indicator for the retrieval quality of the state-of-the-art ad-hoc IR systems[7] The purpose of TREC is to support research within the information retrieval community by providing the infrastructure necessary for large-scale evaluation of text retrieval methodologies. This implies the testing of the quality of implementations of current algorithms within IR. [Har00] compares the results of the participated ad-hoc IR systems over the last years. It shows up, that since 1996 there has been a stagnation registered within the ad-hoc IR systems regarding the retrieval quality (recall and precision). From this can be concluded that after the actual state of research the development of retrieval and indexation algorithms is exhausted.

3.1 Dependence of the Retrieval Quality on the Query

The quality of an ad-hoc search service in the sense of recall and precision is largely dependent on the actual query. This characteristic was proven practically on ad-hoc IR systems by [Har00]: Different ad-hoc IR systems at TREC have been compared on a pro-query basis. It showed up that an ad-hoc IR system can produce a very high quality result on a certain query, while the same system is performing very badly on different queries, compared to other ad-hoc IR systems.

3.2 Alteration of the Query

The aim of alteration of the query subsists in the adaptivity of the vocabulary of the user to the IR system. This idea has been taken up by many researchers following up with the approach of an automatic alteration of a query. The approach is widespread and popular. There exist a lot of alteration modification algorithms in the literature, e.g. [BMS98].

Definition 11 Query Modification. Query modification is the automatic alteration of a query by reason of additional knowledge (thesaurus, relevance feedback, statistics, etc.) with the aim to obtain better retrieval results. Thereby the danger of the so-called *query drift* is given, being the danger that the altered query does no longer reflect the original information need.

[7] With ad-hoc search a completely automatic search is understood.

In literature one can also find other terms for query modification, like *query extension* or *query reformulation*.

Our approach is different from the known alteration modification algorithms in the way, that the query modification is completely decoupled from the document corpus and the extension of the query possesses universally valid status — as described in chapter 1.5. By this, we are able to avoid the danger of the query drift, as described above.

3.3 Query Expansion

The dependency of the retrieval quality on the query supports our motivation in the SemanticMiner system to lay the focus on the query for an ad-hoc search service. The underlying *query expansion* approach is able to attach ontological knowledge to the query ad-hoc IR system and thus improve the quality of the produced results. On the one hand this leads to an improve of the recall values, because more relevant documents are found by the quantitative (and certainly qualitative) raising of the search terms. On the other hand no general statement can be given on the precision values, due to the fact that with the raising of the amount of relevant documents found to the information need I and the query q, i.e. $|\mathcal{R}(q, I)|$, the amount of all found documents with query q is raising as well, i.e. $|\mathcal{E}(q)|$. However, typically the user is only looking at the top 10 to 20 documents of a search retrieval. Therefore we introduce precision at n — another measure for IR — regarding to [Coo97].

Definition 12 Precision at n. Recall and precision are measures for the entire hitlist. They do not account for the quality of ranking the hits in the hitlist. Users want the retrieved documents to be ranked according to their relevance to the query instead of just being returned as a set. The most relevant hits must be in the top few documents returned for a query. Relevance ranking can be measured by computing precision at different cut-off points. For example, if the top 10 documents are all relevant to the query and the next ten are all nonrelevant, we have 100% precision at a cut off of 10 documents but a 50% precision at a cut off of 20 documents. Relevance ranking in this hitlist is very good since all relevant documents are all above the nonrelevant ones. Sometimes the term recall at n is used informally to refer to the actual number of relevant documents up to that point in the hitlist, i.e., recall at n is the same as (n * precision at n).

The qualitative ranking function combined with the query expansion by ontological knowledge within the SemanticMiner system lead to a substantial increase of the "subjective" (for the the user relevant) precision values — with regard to precision at 10 to precision at 20. This is due to the fact that documents with a high term conformance of *all* query terms experience the highest ranking.

We can as well conclude from [Har00] that the change of an ad-hoc search service during the performance of a query is definitely making sense. The advantage of the SemanticMiner system is that the underlying ad-hoc IR systems are transparent for the system and can be interchanged or supplemented.

3.4 Semantic of the Query

Another lack of general IR approaches lies in the fact that they are scarcely performing a pure syntactical search for terms regardless of the meaning of the words in the documents. Thus, this leads to a large number of hits, containing also documents in which the term was used in a different meaning. Furthermore, it is impossible to perform a search for similar terms respectively containment nor generalization while using such statistical approaches.

During the last three decades, there has been ongoing discussion on whether to focus on support of Natural Language Processing (NLP) with syntactical or semantical technologies. Both sides discussed and propelled approaches. It showed up ever more clearly that both technologies and in particular the interaction between statistic approaches and semantic modelling represent the most promising starting points for the advancement of the NLP.

3.5 Collocation Analysis with Integration of Structured Data

Through the combination of a search request as textual information with (semi-) structured information (e.g. lists, databases, meta data) and logical rule cohesion the performance of the presented approaches (c.f. 3.4) is further increased. The overall goal is to detach essential knowledge contents from the document corpus and present concrete answers, instead of providing a result list of links to documents containing the content.

In the SemanticMiner system this happens by way of *collocations*.

Definition 13 Collocation. A collocation (in our sense) is a significant occurance of two patterns (word forms) in a common context (direct neighborhood). Collocation analysis is a statistical approach (not syntactical). Examples are (dog : bark) or (dark : night).

By building correlation lists from databases which could be taken from an arbitrary source (e.g. a human-resources-system), it is then possible by means of collocation analysis to identify an expert to a specific topic in an enterprise for instance, based on completely unstructured information. The collocation used is ([search term + query expansion] : [data]). Other examples of the usage of collocation analysis is to unweave knowledge lacks over a list of topics or to generate competitor overviews from a company listing at New York Stock Exchange.

3.6 Deduction

As described in section 1, additional benefit of ontologies consist in their nature to allow derivations and evaluations of the above described rule-based interrelations by means of the inference engine OntoBroker. As descibed in 1.3.2, implicit knowledge will thereby be likewise interrogated and represented - made explicit. Thus, for the SemanticMiner system, this implies that all information derived by rules (i.e. has been available only implicit) will be represented as explicit information. Additionally all materializable rules will be materialized during start-up of the system. This means that after the evaluation of all these rules the generated instances are available as if they were not derived by rules. This technique speeds up the response time of the inferencing kernel by factors up to 70. The end user of the system is therefore not able to differentiate, if the information presented to him existed explicitly or has been derived by means of *deduction* and *"'inferencing rules"'*.

4 Future Work: Integration of Information Extraction

Information Extraction (IE) could be integrated by enhancing OntoMat-Annotizer (S-CREAM) [HSC02] to use the OntoBroker system as storage back-end (cf. figure 3, left upper corner). The advantage herein lies in the ability of using and applying the power of inferencing for the learned instances. For example, if the IE system discovers a new instance and this instance is immediately added to the OntoBroker system, all rules will immediately grasp. An application example could be the detection of new virus instances by IE. When these instances are added to Ontobroker, web administrators could be immediately warned to install a new patch on the infected system. In Ontobroker this can be accomplished with the following rule:

```
FORALL Virus, System, Patch Alert(Virus, Patch) <-
Infection(Virus,System) AND PatchAvailable(Virus, System, Patch).
```

The meaning of this rule can be paraphrased as follows: "If there is a known system in the enterprise which is infected by the virus detected and if there is a patch for this virus, then alert the user - e.g. system administrator - to install the corresponding patch for his system and show him where to find it."

5 Conclusion

As described above, the combination of semantic technologies and IR approaches, how it is converted within the SemanticMiner system, offers thus a clear benefit. The use of the Knowledge Retrieval system produces high-quality search results in practice and reduces the time spent on searching for information needed.

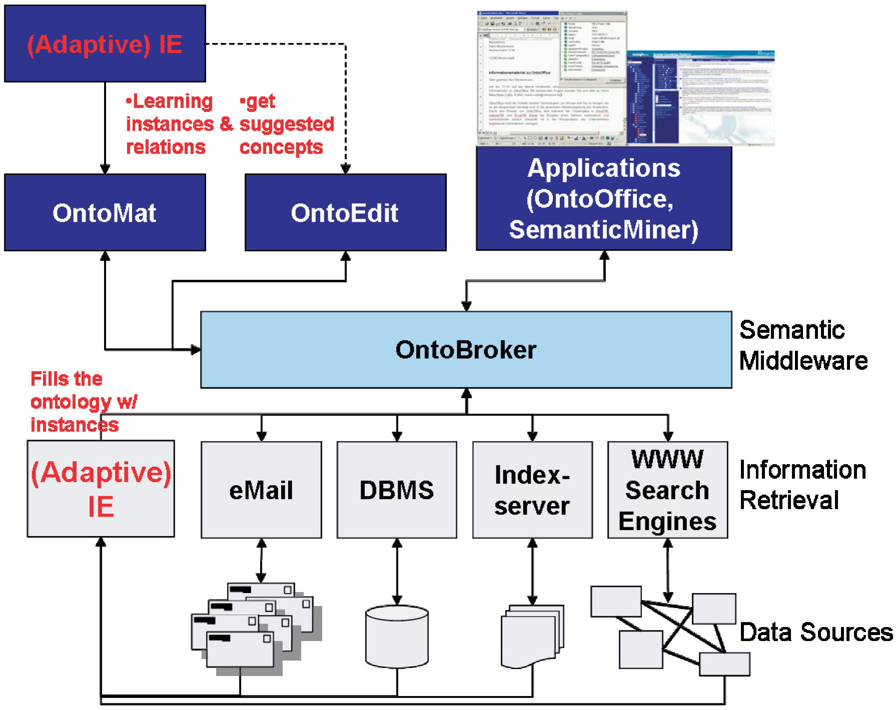

Figure 3: Architecture of IE Integration into the ontoprise Framework

Furthermore, by the addition of IE instances, relations, and concepts can be learned (semi)automatically. The newly created instances and relations would then be accessible through the API of OntoBroker. Thus all applications based on the OntoBroker system such as OntoOffice or SemanticMiner as described above could use and benefit of the output of the IE tools.

References

[BMS98] C. Buckley, M.Mandra, and A. Singhal. Improving Automatic Query Expansion. In *21st ACM SIGIR International Conference on Research and Development in Information Retrieval*, pages 206–214, 1998.

[Bra] Tim Bray. Divine Metadata for the Web. http://www.xml.com/pub/a/2001/01/24/rdf.html?page=2#rdf.

[Bru01] Bernd Bruegge. Einfhrung in die Informatik II, 2001. http://wwwbruegge.in.tum.de/teaching/ss01/Info2/vorlesung/folien.

[BYRN99] R. Baeza-Yates and B. Ribeiro-Neto. *Modern Information Retrieval*. ACM Press, New York, Addison-Wesley, 1999.

[CLvRC98] F. Crestani, M. Lalmas, C.J. van Rijsbergen, and I. Campbell. Is this Document Relevant? ... Probably - A Survey of Probablistic Models in

Information Retrieval. *ACM Computing Surveys*, 30:528–552, December 1998.

[Coo97] W. S. Cooper. On Selecting a Measure of Retrieval Effectiveness. In K. S. Jones and P. Willett, editors, *Readings in Information Retrieval*. Morgan Kaufmann, 1997.

[DEFS99] S. Decker, M. Erdmann, D. Fensel, and R. Studer. Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In R. Meersman et al., editor, *Database Semantics: Semantic Issues in Multimedia Systems*. Kluwer Academic, 1999.

[ERM] Das Entity-Relationship-Modell. http://www.ph-ludwigsburg.de/mathematik/lehre/ws0203db/skript/021114/script/DBERModell.htm.

[Fik] Richard Fikes. Ressource Description Framework (RDF). http://www.stanford.edu/class/cs222/slides2/RDF.PDF.

[Fuh96] N. Fuhr. Ziele und Aufgaben der Fachgruppe Information Retrieval, January 1996. http://ls6-www.informatik.uni-dortmund.de/ir/fgir/mitgliedschaft/brochure2.html.

[GHVD03] Benjamin N. Grosof, Ian Horrocks, Raphael Volz, and Stefan Decker. Description logic programs: Combining logic programs with description logic. In *Proceedings of WWW 2003*, pages 48–57, 2003.

[Gru93] T.R. Gruber. A translation approach to portable ontology specifications. *Knowledge Aquisition*, 5:199–220, 1993.

[Gru95] T.R. Gruber. Towards principles for the design of ontologies used for knowledge sharing. *International Journal of Human-Computer Studies*, 43:907–928, 1995.

[Har00] D. Harman. What We Have Learned, and not learned, from TREC. In *BCS-IRSG: 22nd Annual Colloquium on IR Research*, pages 2–20, April 2000. http://irsg.eu.org/irsg2000online/papers/harman.htm.

[Heg] Stephen Hegner. Horn Clauses and Feature-Structure Logic: Principles and Unification Algorithms. http://www.cs.umu.se/ hegner/Publications/PDF/lli93.pdf.

[Hof01] Tobias Hofman. Topic Maps, 2001. http://weblogs.medien.uni-weimar.de/topicmaps/about.

[HSC02] Siegfried Handschuh, Steffen Staab, and Fabio Ciravegna. S-CREAM - Semi-automatic CREAtion of Metadata. In *Proceedings of the 13th International Conference on Knowledge Engineering and Knowledge Management (EKAW02)*, 2002.

[Kai93] A. Kaiser. *Computer-untersttzes Indexieren in Intelligenten Information-Retrieval Systemen. Ein Relevanz Feedback orientierter Ansatz zur Informationserschliessung in unformatierten Datenbanken*. PhD thesis, Wirtschaftsuniversitt Wien, 1993.

[KLW95] M. Kifer, G. Lausen, and J. Wu. Logical Foundations of Object-Oriented and Frame-Based Languages. *Journal of the ACM*, 42:741–843, 1995.

[MUS03] Andreas Maier, Mike Ullrich, and Hans-Peter Schnurr. Ontology-based Information Integration in the Automotive Industry. Technical report, ontoprise whitepaper series, 2003.

[NSB00] S. Neumann, L. Schuurmans, and M. Bonifacio. Verteilte Systeme im Wissensmanagement. *Information Management und Consulting*, 15:75–82, 2000.

[OO] Roxane Ouellet and Uche Ogbuji. Introduction to DAML: Part II. http://www.xml.com/pub/a/2002/03/13/daml.html.

[SM01] S. Staab and A. Maedche. Knowledge Portals: Ontologies at Work. *AI Magazine*, 2(21), 2001.

[Sub] Peter Suber. Predicate Logic Terms. http://www.earlham.edu/ peters/courses/log/terms3.htm.