The Ontological Engineering Initiative $(KA)^2$

V. Richard Benjamins
University of Amsterdam, SWI, Roetersstraat 15, 1018 WB Amsterdam, The Netherlands
Dieter Fensel
University of Karlsruhe, AIFB, 76128 Karlsruhe, Germany

Abstract. The **K***nowledge* **A***nnotation Initiative of the* **K***nowledge* **A***cquisition Community,* $(KA)^2$ is an initiative to develop an ontology that models the knowledge acquisition community (its researchers, topics, products, etc.). This ontology will form the basis to annotate WWW documents of the KA community in order to enable intelligent access to these documents. $(KA)^2$ is an open joint-initiative where the participants are actively involved in (i) a distributive ontological engineering process to model the knowledge acquisition community (a domain ontology), and (ii) annotating web pages relevant for the KA community (the instances of the domain ontology).

$(KA)^2$ aims at intelligent knowledge retrieval from the Web and automatic derivation of "new" knowledge. In other words, it aims at knowledge-based reasoning on the Web, as opposed to information retrieval. Another objective of the initiative is to get better insight in distributive ontological engineering processes.

1 Introduction and motivation

The $(KA)^2$ initiative[1] has three major motivations and contributions. First, the World-Wide Web can be seen as the largest knowledge base ever (even bigger than CYC [1]). However, the amount of inferencing and deduction of new knowledge on the WWW is very limited. Current search engines (like Altavista or Yahoo) are mostly key-word based and basically do *information retrieval.* This leads, as everybody might have experienced, to answers containing overwhelming amounts of references to web documents. In other words, there is a clear information overload [2]. Although search engines get increasingly smarter, we expect that there will be a limit to such keyword-based information retrieval. An alternative approach concerns so-called *ontology-based* knowledge access or retrieval. An ontology refers to a commonly agreed conceptualization of some domain. One of the issues $(KA)^2$ aims to investigate, is the power and role of ontologies in intelligent access to information on the Web. In this sense, $(KA)^2$ hopes to contribute to the solution of a significant problem.

A second motivation of the $(KA)^2$ initiative relates to ontological engineering. Ontologies attract nowadays much attention of a variety of research communities [3], illustrating the fact that ontologies are considered useful for many applications. The notion of ontology, however, has been somewhat diluted lately. Many specific domain models (e.g. taxonomies) are currently called ontologies, regardless of the fact that these ontologies might only reflect the opinion of one or several persons, and basically only contain classes and sub-classes (and no axioms). Building a *consensual* and *rich* ontology is, however, not an easy task as it requires agreement of different people on different

[1] The URL of the $(KA)^2$ homepage is
`http://www.aifb.uni-karlsruhe.de/WBS/broker/KA2.html`

aspects. Concerning the KA ontology for example, in the Dutch university system, a Ph.D. student can officially only be supervised by a full professor, which would give rise to the ontological axiom: If X is supervisor of Y and Y is a Ph.D. student, then X is a full professor. In Spain, on the other hand, a Ph.D. student can be supervised by either a full professor or a doctor, making the axiom above invalid. $(KA)^2$ is an international initiative whose aim is to build a consensual ontology in a distributive way. A contribution of $(KA)^2$ is that it can be viewed as a large-scale experiment in collaborative, distributive ontology construction.

A third motivation of $(KA)^2$ is to have a clear insight in the groups and topics of the knowledge acquisition research community. To come up with a commonly agreed conceptualization and classification of the work and the people active in the KA community, is an important contribution in itself. Moreover, if this knowledge is easily and intelligently accessible, it could be very helpful to stimulate cooperations between different groups, to unite forces and to prevent repetitions of work.

The structure of this paper is as follows. In Section 2, we mention a disclaimer of our initiative, restricting its scope for feasibility reasons. In Section 3, we describe the approach to achieve the initiative's objectives. Section 4 discusses Ontobroker, which includes an ontology-based web-crawler. In Section 5, we briefly sketch the organizational structure of the initiative. Finally, in Section 6 we conclude the paper.

2 Feasibility of the approach and deliberative restrictions

As outlined above, one of the objectives of $(KA)^2$ is to turn the WWW from a knowledge base into a knowledge-based system, using an ontology and by developing an interpreter. However, it is infeasible and unthinkable (and even undesirable) that the whole World-Wide Web community would agree on one unique ontology. This would imply that all people shared the same view on the world. Nothing is less true.

Therefore, we used the metaphor of a newsgroup: a group of people that share a common subject and a related point of view on this subject [4]. This allows people – we call them an *ontogroup* – to annotate their web pages based on a shared ontology[2] to enable automatic inference.

In $(KA)^2$, we are defining such an ontogroup as the knowledge acquisition community. The web sites of the KA community form a sub-web of the WWW, and we think it is feasible to come up – in a *distributive* and *collaborative* way – with a KA-community accepted view on the KA world.

3 The approach

There are three main issues involved in the initiative (see Figure 1). (i) The knowledge acquisition community has to built its own ontology. (ii) The community has to fill this ontology with instances by annotating the relevant web pages. (iii) Given a query, a web-crawler has to access the web pages and use the ontology to infer answers. Depending on how rich the ontology is (e.g. the amount axioms allowing inferencing), the web-crawler can also deduce "new" information that is not explicitly stored on the Web. Notice that such inferencing is very common in knowledge-based systems, but not at all for web search engines.

[2]Actually, "shared ontology" is a pleonasm since an ontology is by definition shared. We write it here to stress the consensus aspect of an ontology, which is not always a characteristic of existing ontologies.

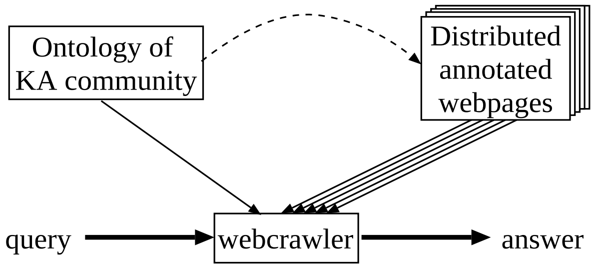

Figure 1: Overview of the $(KA)^2$ initiative. Relevant web pages of the knowledge acquisition community are annotated in terms of the KA ontology (dotted, bowed arrow). A web-crawler gets information from the ontology and from the web pages (the instances of the ontology) and based on that, it can deduce "intelligent" answers to queries.

3.1 Distributive ontological engineering

In order to come up with a consensual ontology of the knowledge acquisition community, we build the ontology as a collaborative joint-effort of the whole KA community. This requires that the ontology can be easily inspected, browsed and downloaded. These requirements have lead us to use the Ontolingua server [5]. Ontolingua is an interactive environment especially useful for updating, maintaining and browser ontologies. Ontolingua ontologies can be translated to several different languages, including Prolog, CORBA's IDL [6], CLIPS, LOOM [7], KIF, Epikit [8].

The current version of the ontology can be viewed at the European mirror site in Madrid of the Ontolingua server of Stanford University (http://www-ksl-svc-lia.dia.fi.upm.es:5915/). Login as "ontologias-ka2" with password "adieu007". The ontology for the KA community consists of seven related ontologies: an organization ontology, a project ontology, a person ontology, a research-topic ontology, a publication ontology, an event ontology, and a research-product ontology.

Ontological primitives Ontologies built in Ontolingua use the Frame Ontology [9], which is written in KIF (Knowledge Interchange Format) [10]. The Frame Ontology is, as its name suggests, a frame-based language which includes primitives such as classes, sub-classes, attributes, values, relations and axioms. Related ontologies can be connected to each other by inclusion.

Design decisions Before starting to build the (current version of the) ontology of the KA community, we took several ontological design decisions in line with the goal of the $(KA)^2$ initiative.

- The ontology should be simple. If we want to have maximal participation of the active research groups in the KA community, then using the ontology should be easy and straightforward, otherwise there is the danger to get lost in details. It should be – as much as possible – unambiguous for the provider agents to model particular instances (provider agents provide the initiative with knowledge by annotating their web pages, see Section 5). This made us decide to not use some of the already existing ontologies in Ontolingua, but rather to start from scratch focusing only on the most relevant concepts. Later versions of the ontology can be more extensive.
- The ontology should be modular and allow reusability. The ontology is currently divided in seven separate, but closely related, ontologies. Each of these

```
ResearchTopic
  Reuse
  PSM
  Ontologies
  ValidationAndVerification
  SpecificationLanguages
  KAMethodologies
    SisyphusII
    SisyphusIII
    SisyphusIV
  AgentOrientedApproaches
  KAFromNL
  KnowledgeManagement
    CorporateMemories
    EnterpriseModeling
  DistributedModelingOverTheInternet
  KAthroughMachineLearning
  RippleDownRules
  KAthroughConceptualGraphs
```

Table 1: The research topics in Knowledge Acquisition covered by the $(KA)^2$ initiative. Indentation denotes the *is-a* relation.

can be reused in other ontologies. In particular, several of these sub-ontologies are not specific for the KA community, but could be used for modeling any research community. Examples are the ontologies for “person”, “publication” and “organization”.

- The ontology should have high “visibility” and be easy accessible. This made us take the decision to use the Ontolingua server.

A distributive joint-effort Building the ontology is a collaborative and distributed process of the KA community. As an example, we sketch here how we are collaboratively developing the sub-ontology about “research topics” We identified 13 active research topics within the KA community, which are modeled as instances of the class “research-topic”. The topics appear in Table 1, where indentation represents the *is-a* relation between a topic and its sub-topics.

We characterize each of these research topics with an ontology, which can be seen as a scheme to be completed. This scheme has been established in a plenary meeting during the 9th KEML workshop in Germany in January 1998 (Knowledge Engineering: Method and Languages) and has (informally) the structure as shown in Table 2.

In order to come up with a high-quality ontology, we invited experts of the different subfields of KA to actively participate. For example, if some research group works on “verification and validation”, then that group could complete the sub-ontology of the research-topic ontology about V&V. At the moment (March, 1998), 15 groups (30 experts) are distributively working on providing the instances for the research-topics ontology. The task of each group is to complete the ontology (scheme) for his or her respective research topic. As an example, we show in Table 3 a possible instantiation of the research-topic “ontologies”.

```
Class: research-topic
Attributes:
    Name: <string>
    Description: <text>
    Approaches: <set-of keyword>
    Research-groups: <set-of research-group>
    Researchers: <set-of researcher>
    Related-topics: <set-of research-topic>
    Sub-topics: <set-of research-topic>
    Events: <set-of events>
    Journals: <set-of journal>
    Projects: <set-of project>
    Application-areas: <text>
    Products: <set-of product>
    Bibliographies: <set-of HTML-link>
    Mailing-lists: <set-of mailing-list>
    Webpages: <set-of HTML-link>
    International-funding-agencies: <funding-agency>
    National-funding-agencies: <funding-agency>
    Author-of-ontology: <set-of researcher>
    Date-of-last-modification: <date>
```

Table 2: Provisional ontology to characterize “research-topics”.

```
Class: research-topic: ontologies
Attributes:
    Name: ontologies
    Description: "Concerned with developing reusable and sharable
                  knowledge, originally static domain knowledge"
    Approaches: formal ontologies, conceptual ontologies,
                implemented ontologies, ...
    Research-groups: Stanford, SWI, AIFB, UPM, ISI, CNR, ...
    Researchers: Mike Uschold, Asun Gomez-Perez, Gertjan van
                 Heijst, Adam Farquhar, Nicola Guarino, ...
    Related-topics: Problem-solving methods, software engineering,
                    knowledge representation
    Sub-topics: description-logics, ...
    Events: 1997 Spring symposium on ontologies, ECAI workshop on
            formal ontologies, FOIS'98, ...
    Journals: IJHCS, KER, IEEE-Expert, ...
    Projects: Ontolingua, Plinius, Kactus, Methontology,
              (KA)2, GRASP, ...
    Application-areas: knowledge management, text understanding, ...
    Products: the Ontolingua Server, ODE, VOID, ...
    Bibliographies: http://www.kr.org/top/bibliography.html
    Mailing-lists: ontology@cs.umbc.edu, kaw@swi.psy.uva.nl
    Webpages: http://www.medg.lcs.mit.edu/doyle/top/
    International-funding-agencies: EC
    National-funding-agencies: DARPA (USA)
    Author-of-ontology: <Richard Benjamins>
    Date-of-last-modification: <Feb 12 1998>
```

Table 3: Example of instantiation of the research topic “ontologies”.

Representation of the ontology As will be presented in Section 4, our web-crawler reasons with Frame Logic (FLogic [11]). This means that the Ontolingua ontology also has to be available in FLogic. We deliberatively did not choose for doing the collaborative ontological engineering process in FLogic for two reasons. (i) Ontolingua comes with an integrated environment to develop ontologies, which is not the case for FLogic. (ii) Ontolingua is well known, which enhances the visibility of the ontology and of the initiative.

Ontolingua comes with an "ontology editor" that allows developers to input classes, sub-classes, attributes, values, axioms, etc. in a structured way, and the editor automatically generates Ontolingua code. Although the ontology editor helps, many people may have experienced that building an ontology from scratch in Ontolingua is daunting, not in the last place because of slow network connections. Experience has shown that the Ontolingua editor is better suited for checking, maintaining and modifying the ontology than for building an ontology from scratch. Therefore, an alternative strategy is to build ontologies off-line, and then import them into Ontolingua. However, writing Ontolingua code is not a comfortable level for persons to work with, that is, it is too close to the symbol level. To overcome this problem, ODE [12] has been developed (Ontological Design Environment) and it allows developers to specify their ontology at a conceptual level by means of completing tables (see Table 4). These tables are then automatically translated into Ontolingua code, which can be included in the ontology at the Ontolingua server.

A consequence of the decision to use both Ontolingua and FLogic is that we have to provide translators to establish a formal connection between the two. Basically there are two possibilities. (1) Translators from ODE to both Ontolingua and FLogic. Equivalence between the two is guaranteed by always modifying the ontology in ODE. (2) A translator from Ontolingua to FLogic. If in addition a translator from FLogic to Ontolingua is built, then it becomes also possible to inspect the instances if the ontology at the Ontolingua server. Notice that the current instances of the ontology have been entered manually, but in the course of the initiative they will be collected from the distributed web pages of the KA community.

Examples of the ontology As mentioned above, the KA ontology currently comprises seven different ontologies (about organizations, projects, persons, research-topics, publications, events and research-products). We have to stress that these represent the current version of the ontology. It is the aim of $(KA)^2$ to come up with a consensual version. In the following, we show global overviews of two sub-ontologies: the Person-ontology and the Publication-ontology.

The Person-ontology defines the types of persons working in academic environments, along with their characteristics. This ontology defines 10 classes and 23 relations. The overview does not show which classes the relations connect (but it can be browsed at Ontolingua Server). Indentation denotes the is-a relation.

```
Class hierarchy (10 classes defined):

    Person
       Employee
          Academic-Staff
             Lecturer
             Researcher
          Administrative-Staff
             Secretary
```

Concept name	Synonym	Description	Instances	Attributes
Person				Photo First-name Last-name Address Email ...
Researcher			Nicola Guarino Enric Plaza Tom Gruber Rose Dieng Stefan Decker Nigel Shadbolt ...	Research-interest Member-of Cooperates-with ...
PhD-Student			Mariano Fernandez Lopez Remco Straatman	Has-Supervisor ...
...	...	...	...	...

Table 4: Using tables to specify an ontology. A small part of how to specify the "person-ontology".

```
            Technical-Staff
      Student
         Phd-Student

23 relations defined:

  Address, Affiliation, Cooperates-With, Editor-Of, Email, First-Name,
  Has-Publication, Head-Of-Group, Head-Of-Project, Last-Name,
  Member-Of-Organization, Member-Of-Program-Committee,
  Member-Of-Research-Group, Middle-Initial, Organizer-Of-Chair-Of,
  Person-Name, Photo, Research-Interest, Secretary-Of, Studies-At,
  Supervises, Supervisor, Works-At-Project
```

The Publication-ontology defines – in 13 classes and 28 relations – the usual bibliographic entities and attributes. We tried, however, to keep it manageable.

```
Class hierarchy (13 classes defined):

    On-Line-Publication
    Publication
       Article
          Article-In-Book
          Conference-Paper
          Journal-Article
          Technical-Report
          Workshop-Paper
       Book
       Journal
          IEEE-Expert
          IJHCS
          Special-Issue

28 relations defined:

  Abstract, Book-Editor, Conference-Proceedings-Title,
```

```
Contains-Article-In-Book, Contains-Article-In-Journal,
Describes-Project, First-Page, Has-Author, Has-Publisher, In-Book,
In-Conference, In-Journal, In-Organization, In-Workshop,
Journal-Editor, Journal-Number, Journal-Publisher, Journal-Year,
Last-Page, On-Line-Version, On-Line-Version-Of, Publication-Title,
Publication-Year, Technical-Report-Number, Technical-Report-Series,
Type, Volume, Workshop-Proceedings-Title
```

3.2 The annotation language

The problem with information retrieval from the Web is that there is no commonly used syntax for representing semantics. Current search engines are therefore restricted to keyword-based search, and retrieve information by syntactically matching input words with words appearing in web documents. This "keywordness" is the reason for the overwhelming amount of (also) irrelevant answers on a query.

Basically, the cause of the problem is that HTML does not allow to specify semantics. For the purpose of $(KA)^2$ it suffices to simply add one new attribute to the anchor tag of HTML: the *onto* attribute. This attribute does not affect the visualization of HTML documents by standard web browsers such as Netscape or Explorer. The only thing that the onto attribute does, is that it makes visible valuable pieces of knowledge for the web-crawler – in the same way as (only) glittering objects in the world are visible for a crow. This small extension of HTML has been chosen to keep annotation as simple as possible to lower the threshold for participants of the initiative. Also, it enables the direct usage (actually, reuse) of textual knowledge already in the body of the anchor, as well as of further information provided by the other anchor attributes. This prevents the knowledge provider from representing the same piece of information twice. In our case, this simple solution suffices because only *factual* ontological knowledge is contained in HTML pages [13].[3]

Figure 2 illustrates fragments of an example web page annotated with the `onto` attribute. For example, `page` in ⟨`a ONTO="page[address=body]"`⟩ refers to the URL of the web page. `Body` refers to what follows and what is within the scope of the anchor, i.e. until the closing ⟨`/a`⟩. `Address` is a class of the KA ontology.

4 Ontobroker

Having discussed the KA ontology and the annotated web pages, in this section, we will present a brokering service that uses that knowledge to make intelligent deduction. The ontology-based brokering service Ontobroker[4] consists of three main elements: a

[3]The W3C –the international *World-Wide Web Consortium* for developing and promoting standards for the web– currently introduces the eXtensible Markup Language (XML) (http://www.w3.org/TR/PR-xml-971208) as a new standard for expressing the structure of web documents, and the resource description framework (RDF) (http://www.w3.org/Metadata/RDF/Group/WD-rdf-syntax) for describing the semantics of web documents. Meta-data defined in RDF have to be provided on an extra page or in a bloc inside a web-page. Therefore, elements of a web page such as text fragments or links cannot directly be annotated with semantics, but must be repeated in order to be enriched with meta-information. This design decision may cause problems for maintaining web documents due to the redundancy of the information. It is our intention to – when a final version of RDF is recommended by the W3C – implement a wrapper that automatically generates RDF definitions from our annotations (cf. [13] for more details).

[4]The URL of Ontobroker is `http://www.aifb.uni-karlsruhe.de/WBS/broker/`

```
----------------------------------------------------------------------
<html>
<head><TITLE> Richard Benjamins </TITLE>
<a ONTO="page:Researcher"> </a>
</head>

<H1> <A HREF="pictures/id-rich.gif">
<IMG  align=middle SRC="pictures/richard.gif"></A>
<a ONTO="page[photo=href]"
HREF="http://www.iiia.csic.es/~richard/pictures/richard.gif" ></a>
<a ONTO="page[firstName=body]">Richard</a>
<a ONTO="page[lastName=body]">Benjamins </a>
</h1> <p>

<A ONTO="page[affiliation=body]" HREF="#card">
Artificial Intelligence Research Institute (IIIA)</A> -
<a href="http://www.csic.es/">CSIC</a>, Barcelona, Spain <br>
and <br>
<A ONTO="page[affiliation=body]" HREF="http://www.swi.psy.uva.nl/">
Dept. of Social Science Informatics (SWI)</A>
-
<A HREF="http://www.uva.nl/uva/english/">UvA</A>, Amsterdam, the
Netherlands

<DL>
<DT><STRONG><A HREF="../../IIIA.html">IIIA</A> -
<a ONTO="page[address=body]">
Artificial Intelligence Research Institute </STRONG>
<DT><EM>CSIC - Spanish Scientific Research Council</EM>
<DT>Campus UAB
<DT>08193 Bellaterra, Barcelona, Spain </a>
<DT><IMG SRC="gifs/tel.gif">
    voice: +34-3-580 95 70
<DT><IMG SRC="gifs/fax.gif">
    fax: +34-3-580 96 61
<DT><IMG SRC="gifs/email.gif">
    Email:<A HREF="mailto:richard@iiia.csic.es" ONTO="page[email=href]">
          richard@iiia.csic.es</A>
<DT>URL: <A HREF="http://www.iiia.csic.es/~richard/">
         http://www.iiia.csic.es/~richard</A>
</DL></font>

</body>
</html>
----------------------------------------------------------------------
```

Figure 2: Example web page annotated with the ONTO attribute. `Page` in ⟨`a ONTO="page[address=body]"`⟩ refers to the URL of the page. `Body` refers to what follows and what is within the scope of the anchor, i.e. until the closing ⟨`/a`⟩. `Address` is a class of the KA ontology.

```
-------------------------------------------------------------
http://www.iiia.csic.es/~richard/index.html
http://www.iiia.csic.es/~richard/activities.html
http://www.iiia.csic.es/~richard/interests.html
http://www.iiia.csic.es/~richard/projects.html
http://www.iiia.csic.es/~richard/publications/pub-type.html
http://www.iiia.csic.es/~richard/cv/cv.html
-------------------------------------------------------------
```

Figure 3: An O-page of a knowledge provider agent.

web-crawler (called Ontocrawler), an inference engine and a query interface. Each of these elements is accompanied by a formalization language: the annotation language for annotating web documents with ontological information, the representation language for specifying ontologies (inside Ontobroker), and the query language for formulating queries. Notice that, although we use Ontobroker for $(\mathrm{KA})^2$, it is not specific for this initiative. Given any ontology and correspondingly annotated web pages, Ontobroker can deliver its brokering service.

Ontocrawler First, Ontocrawler searches through a fragment of the WWW that is annotated – using the annotation language – according to a particular ontology (in our case, the KA ontology) and collects the annotated knowledge fragments. Second, it translates the annotated knowledge fragments into facts formulated in the representation language. Neither the inference engine nor the querying client have to be aware of the syntactical way, the facts are represented on the web in the annotation language. Ontocrawler provides this abstraction mechanism. Only the knowledge provider has to use the annotation language.

In order to become a provider of an ontologically annotated knowledge chunk on the WWW, one has to do two things:

1. Define an O-page and register the page's URL at the provider index of Ontocrawler. This O-page contains a sub-index that specifies all URLs of the annotated web documents of that provider. Figure 3 gives an example.

2. Use the annotation language and an ontology of the Ontobroker to annotate the concerned web documents.

Inference engine The inference engine receives the query of a client and uses two information sources for deriving an answer: the ontology chosen by the client (the KA ontology, in our case) and the facts that were found by Ontocrawler on the WWW. The basic inference mechanism of the inference engine is the derivation of a minimal model of a set of Horn clauses (see [13] for more details). However, the language for representing ontologies is syntactically enriched. First, ideas of [14] were used to get rid of some of the limitations of Horn Logic, without requiring a new inference mechanism. Second, languages with richer epistemological primitives than predicate logic are provided. Frame logic [11] is used as the representation language for ontologies inside Ontobroker. It incorporates objects, relations, attributes, classes, and is-subclass-of and is-element-of relationships within a first-order semantic framework.

Query interface The broker communicates with clients who ask for some knowledge using web browsers like Netscape and Explorer. The query interface of Ontobroker comprises several active HTML pages and cgi-scripts that are executed by the browser of the client. The client selects the KA ontology to formulate his query. The answer of the broker will be based on this ontology and on the web documents that have been annotated using this ontology (only if an O-page has been registered, of course). The query language is a subset of the representation language customized for formulating queries.

The query formalism is oriented towards Frame-Logic syntax, that defines the notion of instances, classes, attributes and values. The generic schema 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. At each position in the above schema variables, constants or arbitrary expressions can be used. In the following we will provide some example queries to illustrate our approach.

```
FORALL R <- R:Researcher.
```

This query asks for all known objects, which are instances of the class researcher. Because the object identifier of a researcher is his/her homepage-URL, this query would result in a large list of URLs. This is one of the simplest possible queries. However, usually we are not interested in all researchers, instead we are interested in information about researchers with certain properties, e.g., we want to know the homepage, the last name and the email address of all researchers with first name “Richard”. To achieve this we can use the following query:

```
FORALL Obj, LN, EM <- Obj:Researcher[firstName->>"Richard";
lastName->>LN; email->>EM].
```

The Ontobroker gives the following answer (actually, there is only one researcher in the knowledge base whose first name is “Richard”):

```
Obj = "http://www.iiia.csic.es/~richard/index.html"
LN = "Benjamins"
EM = "mailto:richard@iiia.csic.es"
```

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. Figure 4 shows part of the answer of Ontocrawler. At the top left, the client has chosen to query the Knowledge Acquisition community. A bit lower, one can see the query, and below that the answer of Ontobroker appears (`Att` denotes “attribute” and `T` the type of the value of the attribute).

Since it is not easy for users to use FLogic, Ontobroker has a user-friendly interface that shows the ontology as a hyperbolic graph of the classes. Clicking on the classes in the graph immediately gives access to the attributes of a class which then can be used as elements of the query. The logical language is hidden behind a tabular query interface that allows to formulate most queries by selecting key parameters from the table fields. The query interface also supports the logical combinations of queries to compose more complex queries (AND, NOT, OR, etc.). More details can be found in [13] or directly via http://www.aifb.uni-karlsruhe.de/WBS/broker.

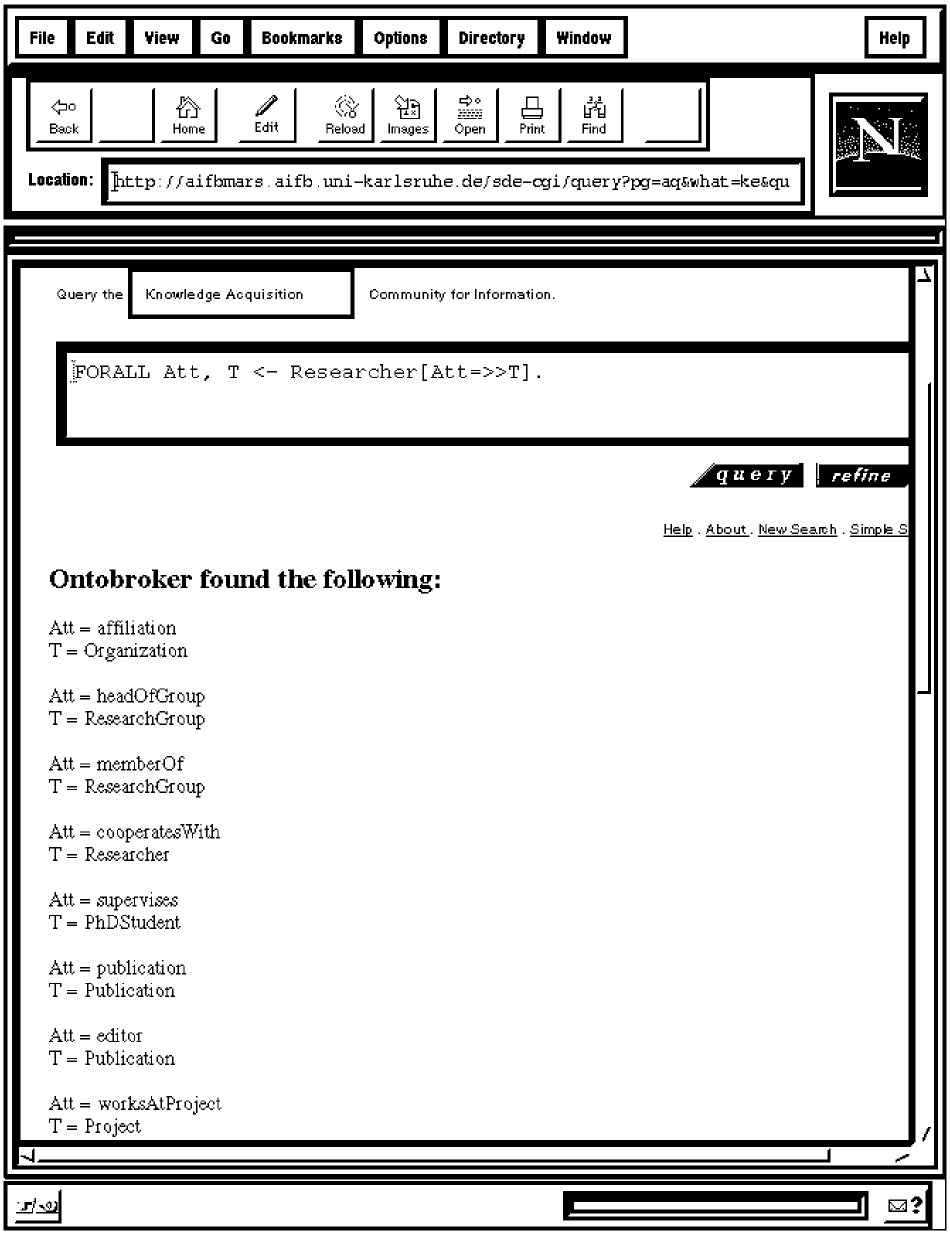

Figure 4: Ontobroker in action.

5 Organization of $(KA)^2$

$(KA)^2$ is organized as a community of several types of agents. Each type has well-defined responsibilities in order to get the $(KA)^2$ initiative started, keep it going, assure its scientific content, make it a global collaborative effort and attract industrial interest: coordinating agents, provider agents, ontopic agents, wise agents and business agents.

Coordinating agents The *coordinating agents* are responsible for the daily matters of the initiative. There are 6 of these agents. The *ontology agent* is responsible for keeping the KA ontology always up-to-date at the Ontolingua server. The *webtool agent* takes care of the web issues involved in the communication between the agents such as setting up a mailing list and a mail archive, as well as providing web tools to collaboratively work on the same ontology. The *managing agent* is responsible for the collaborative ontological engineering process for building the KA ontology, and for the overall process of the initiative. The *recruiting agent* tries to convince KA groups to participate in the initiative (he might make you an offer you can't refuse). The *annotation agent* coordinates the process of annotating web pages, and the *ontobroker agent* is responsible for keeping the Ontobroker up and working. Finally, the *"window on USA" agent* informs the initiative on related events, initiatives and work in the USA.

Wise agents Wise agents are concerned with the scientific issues involved in the initiative. They give high-level steering and suggestions concerning whether the initiative is going into the right direction.

Provider agents Provider agents provide the initiative with instances of the ontology. In other words, they have to annotate their web pages. Currently, around 25 researcher (mainly from Europe) have committed themselves to participate as provider agents. The recruiting agent is responsible for attracting more researchers and groups.

Ontopic agents Ontopic agents are research groups that contribute to the ontological engineering process to establish a consensual ontology of the KA community. There are 15 groups of ontopic agents, each group being responsible for a particular research topic of the KA ontology.

Business agents The *business agent* is responsible for exploring the possibility of external funding of the initiative and raising the interest of possible interested industries.

6 Conclusions

In this paper, we presented an initiative – $(\mathrm{KA})^2$ – whose goal is to enable knowledge-based reasoning on (a subpart of) the WWW, using an ontology. The subpart concerns the web pages of the KA community, and many research groups and researchers are already involved. To achieve the objectives of $(\mathrm{KA})^2$ three things are needed: (1) an ontology of the KA community, (2) annotated web pages in terms of the ontology, and (3) an ontology-based web-crawler to perform reasoning. Constructing the KA ontology will be a collaborative and distributed process for which the Ontolingua server has been chosen. The instances of the ontology are provided distributively by KA researchers through annotating their relevant web pages.

Our initiative implements a joint activity to establish a common notion of how to describe products, issues and agents of a community. Clearly our experiences are immediately applicable to other kinds of standardization attempts. Especially, our initiative deals with issues that are similar to questions in establishing organizational memories and *knowledge management* in general (cf. [15]). How to use ontologies to describe heterogeneous documents and how to organize a decentralized process that establishes such an ontology are key problems in these areas. Therefore, we expect significant contributions to these areas from our experiences.

One of the objectives of $(\mathrm{KA})^2$ is to investigate is the power and role of ontologies in intelligent access to information on the Web. We therefore think that applying these ideas in an industrial or commercial setting could be interesting. To stay close to the $(\mathrm{KA})^2$ initiative, think for example about the usefulness of such knowledge-based reasoning capabilities for scientific publishers. In general, the potential advantages of more intelligent reasoning on the WWW are enormous.

The current status of the (KA)^2 initiative is that all provider agents have to annotate their web pages, using the ontology (stimulated and supported by a dedicated person). However, using Machine Learning techniques it should be possible to automatically learn the instances from the web pages using the KA ontology as background knowledge. In a more distant future, it may also become possible to learn, derive or mine (parts of) the ontology (semi) automatically. For instance, statistical and ML techniques could be used to identify the most frequently occurring concepts at pages of the KA community, and could then try to cluster them. These clusters could then suggest a basic structure or starting point for the ontology. This is not so much of interest for our current initiative, but it is extremely valuable if our initiative shows that ontology-based knowledge retrieval and reasoning is a good alternative for keyword-based information retrieval. In general, it is undoable to build large ontologies as a collaborative process as we do for (KA)^2. In our initiative, however, it is worth the effort because we are still investigating the role of ontologies on the Internet.

There is a huge research effort going on about meta-data for web-documents (e.g., XML, RDF, WebSQL, Dublin Core[5]). More recently, there are also several projects that use ontologies together with meta-data to improve information retrieval (e.g., SHOE, Ontology Markup Language, Conceptual Knowledge Markup Language). Most of these projects relate in some way or antother to (KA)^2. We already mention that the Resource Description Format (RDF) may provide an alternative syntax for writing ontological annotations of web documents. The Dublin Core Initiative provides an ontology consisting of 15 core concepts for meta descriptions of online documents. This initiative has the distributed and heterogeneous ontology development process in common with (KA)^2, but differs in its aims. The level of formalization is lower and the scope is more limited. Describing online documents is only a subset of (KA)^2 which rather aims at describing research topics, groups, projects, and persons. Clearly both initiatives could benefit from each other. See http://www.aifb.uni-karlsruhe.de/WBS/broker/inhalt-wp.html for brief overviews of these related projects.

Finally, some numerical facts about the initiative (March 1998): the ontology currently comprises 80 classes, 27 axioms and 100 attributes (not including the attributes of the sub-ontology ontology of research topic, as it is still under development, see section on research topics). The facts base contains about 400 facts of the 30 researchers that have their pages annotated.

Acknowledgment

We would like to thank all current participants of the initiative: Andreas Abecker, Hans Akkermans, Nathalie Aussenac, Robin Boswell, Frances Brazier, B. Chandrasekaran, Paul Compton, Susan Craw, Stefan Decker, Rose Dieng, Michael Erdmann, Henrik Eriksson, Brian Gaines, Fernando Gomez, Asuncion Gomez-Perez, Robert Gordon, Udo Hahn, James Hendler, Knut Hinkelman, Catholijn Jonker, Rob Kremer, Dickson Lukose, Maillet-Contoz, Frank Maurer, Annejet Meijler, Tim Menzies, Enrico Motta, Mark Musen, Christine Pierret, Enric Plaza (thanks for the title of this paper), Guus Schreiber, Nigel Shadbolt, Derek Sleeman, Maarten van Someren, Rudi Studer, Bill Swartout, Annette ten Teije, Jan Treur, Frank van Harmelen, Sean Wallis, Bob Wielinga, Niek Wijngaards. Further we would also like to thank all participants of EKAW'97 and KEML'98 for their input. Finally, we are grateful to the anonymous re-

[5]URL of the Dublin Core project is http://purl.oclc.org/metadata/dublin_core/.

viewers for their helpful comments. Richard Benjamins is supported by the Netherlands Computer Science Research Foundation with financial support from the Netherlands Organisation for Scientific Research (NWO), and by the European Commission through a TMR grant.

References

[1] D. B. Lenat and R. V. Guha. *Building large knowledge-based systems. Representation and inference in the Cyc project.* Addison-Wesley, Reading, Massachusetts, 1990.

[2] D. O'Leary. The internet, intranets, and the AI renaissance. *IEEE Computer*, 30(1):71–78, 1997.

[3] N. Guarino and R. Poli. The role of ontology in the information technology. *International Journal of Human-Computer Studies*, 43(5/6):623–965, 1995. Special issue on ontology.

[4] D. Fensel, M. Erdmann, and R. Studer. Ontology groups: Semantically enriched subnets of the www. In *Proceedings of the International Workshop Intelligent Information Integration during the 21st German Annual Conference on Artificial Intelligence*, Freiburg, 1997.

[5] A. Farquhar, R. Fikes, and J. Rice. The ontolingua server: a tool for collaborative ontology construction. *International Journal of Human-Computer Studies*, 46(May):707–728, 1997.

[6] R. Orfali, D. Harkey, and J. Edwards, editors. *The Essential Distributed Objects Survival Guide.* John Wiley & Sons, New York, 1996.

[7] R. MacGregor. Inside the LOOM classifier. *SIGART Bulletin*, 2(3):70–76, June 1991.

[8] M. R. Genesereth, editor. *The Epikit manual.* Epistmemics, Inc, Palo Alto, CA, 1992.

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

[10] M. R. Genesereth and R. E. Fikes. Knowledge interchange format, version 3.0, reference manual. Technical report, Logic-92-1, Computer Science Dept., Stanford University, 1992. http://www.cs.umbc.edu/kse/.

[11] M. Kifer, G. Lausen, and J. Wu. Logical foundations of object-oriented and frame-based languages. *Journal of the ACM*, 1995.

[12] A. Gomez-Perez, A. Fernandez, and M. De Vicente. Towards a method to conceptualize domain ontologies. In *Working notes of the workshop on Ontological Engineering, ECAI'96*, pages 41–52. ECCAI, 1996.

[13] D. Fensel, S. Decker, M. Erdmann, and R. Studer. Ontobroker: The very high idea. In *Proceedings of the 11th International Flairs Conference (FLAIRS-98)*, Sanibal Island, Florida, 1998.

[14] J. W. Lloyd and R. W. Topor. Making prolog more expressive. *Journal of Logic Programming*, 3:225–240, 1984.

[15] O. Kühn and A. Abecker. Corporate memories for knowledge management in industrial practice: Prospects and challenges. *Journal of Universal Computer Science, Special Issue on Information Technology for Knowledge Management*, 3(8), 1997.