Knowledge Management through Ontologies

V. Richard Benjamins
SWI
University of Amsterdam
The Netherlands
richard@swi.psy.uva.nl

Dieter Fensel
AIFB
University of Karlsruhe
Germany
dfe@aifb.uni-karlsruhe.de

Asunción Gómez Pérez
DIA
Technical University of Madrid
Spain
asun@delicias.dia.fi.upm.es

Abstract

Most enterprises agree that knowledge is an essential asset for success and survival on a increasingly competitive and global market. This awareness is one of the main reasons for the exponential growth of knowledge management in the past decade. Our approach to knowledge management is based on ontologies, and makes knowledge assets intelligently accessible to people in organizations. Most company-vital knowledge resides in the heads of people, and thus successful knowledge management does not only consider technical aspects, but also social ones. In this paper, we describe an approach to intelligent knowledge management that explicitly takes into account the social issues involved. The proof of concept is given by a large-scale initiative involving knowledge management of a virtual organization.

1 Introduction

According to Information Week [APH98] "the business problem that knowledge management is designed to solve is that knowledge acquired through experience doesn't get reused because it isn't shared in a formal way." Because this can be any kind of knowledge – tacit, documented, procedural, etc. – the term knowledge management may refer to such various things [Wii94, O'L98] as corporate memories and instincts, expert systems, document managing systems, learning organizations [vHvdSK96], etc.

Knowledge management is not a product in itself, nor a solution that organizations can buy off-the-shelf or assemble from various components. It is a process implemented over a period of time, which has as much to do with human relationships as it does with business practice and information technology (IT). The process of managing knowledge involves the following actions:

- Knowledge gathering: acquisition and collection of the knowledge to be managed.
- Knowledge organization and structuring: imposing a structure on the knowledge acquired in order to manage it effectively.
- Knowledge refinement: correcting, updating, adding, deleting knowledge, in short: maintaining knowledge.
- Knowledge distribution: bringing the knowledge to the professionals who need it.

We can distinguish between two types of knowledge management systems: vertical and horizontal systems. Vertical systems are developed for one particular kind of business situation. Such systems are highly effective and have proven their value. Often, vertical systems are developed inside a company and are highly situation specific. Therefore, such systems are of little value for other business situations. Horizontal knowledge management systems are general systems that can be applied to a variety of business situations. They are frameworks that can be instantiated to particular situations (see [APH98] for a discussion of five of such systems: Wincite, Intraspect, ChannelManager, BackWeb, and KnowledgeX).

In this paper, we present a horizontal approach to knowledge management that is grounded in research on knowledge engineering. Knowledge engineering is a field that – during the past 15 years – has been concerned with capturing, analyzing, organizing, struc-

Proc. of the 2nd Int. Conf. on Practical Aspects of Knowledge Management (PAKM98)
Basel, Switzerland, 29-30 Oct. 1998, (U. Reimer ed.)
http://sunsite.informatik.rwth-aachen.de/Publications/CEUR-WS/Vol-13/

turing, representing, manipulating and maintaining knowledge in order to obtain intelligent solutions for hard problems [SBF98, O'L97]. It is therefore no surprise that knowledge engineering methodologies and techniques can be of high value for knowledge management, which is exactly concerned with the issues mentioned above in a business environment [SAA+99].

In order for our approach to work in a particular organization, we assume that it has an Intranet/Extranet or access to the Internet and that each member of the organization has a browser. In addition, the approach requires that the knowledge of interest is available in HTML pages on the net. Many companies already have an intranet, which is an easy to use infrastructure that gives companies access to a large variety of Internet techniques. Therefore, for users already familiar with browsers, our approach has a short learning curve.

This paper is organized as follows. In Section 2, we outline the technology underlying our approach. In Section 3, we present an application of our approach for a virtual organization: the knowledge acquisition research community. We indicate how this case study relates to a business context. In Section 4, we identify a number of possible dangers to successful implementation of knowledge management systems. Finally, Section 5 concludes the paper by putting it in context and relating it to the "Knowledge Chain" of knowledge management.

2 An ontology-based approach to knowledge management

Our approach comprises three main subtasks: (1) ontological engineering to build an ontology of the subject matter, (2) characterizing the knowledge in terms of the ontology, and (3) providing intelligent access to the knowledge. In a sense, this is reminiscent of relational database technology, where the ontology would correspond to the data model, the characterization would correspond to the instances (data) contained in the database, and access would take place through SQL. We will show, however, that our approach is significantly different from centralized databases, especially with respect to distributiveness and intelligence. Our approach captures distributive, rather than centralized knowledge. The knowledge is directly accessed at its original location (in HTML pages) rather than be separately input to a database. The approach allows to "discover" knowledge that is not explicitly known, but that can be deduced based on general knowledge (captured in the ontology). For example, in the context of human resource management, if in some company only senior managers can lead projects, and Mr. Paton is project leader, then we can deduce that Mr. Paton is a senior manager, even though this is nowhere stated explicitly.

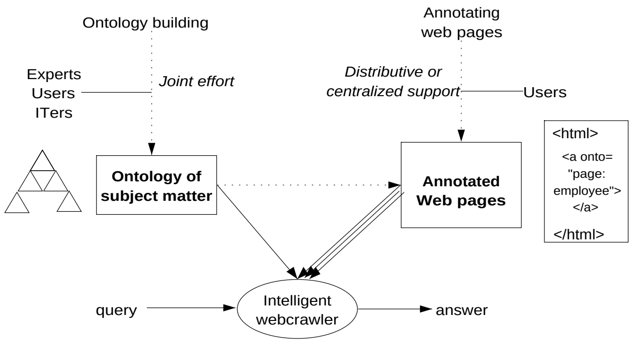

Figure 1: The approach.

Figure 1 gives a general overview of the approach. An ontology of the subject matter has to built, which is used to characterize the subject matter (i.e. to fill the ontology with instances). An intelligent web crawler receives a query in terms of the ontology, consults the subject matter (the instances), interprets them using the ontology and generates an answer. The instances (the actual knowledge to be managed) are distributed over different HTML pages (of an intranet or the Internet).

2.1 Ontological Engineering

An ontology is a shared and common understanding of some domain that can be communicated across people and computers [Gru93, Gua95, UG96, vSW97]. Ontologies can therefore be shared and reused among different applications [FFR97], which is one of the main reasons why ontologies are popular nowadays. An ontology can be defined as a formal, explicit specification of a shared conceptualization [Gru93, Bor97]. "Conceptualization" refers to an abstract model of some phenomenon in the world by having identified the relevant concepts of that phenomenon. "Explicit" means that the type of concepts used, and the constraints on their use are explicitly defined. "Formal" refers to the fact that the ontology should be machine readable. "Shared" reflects the notion that an ontology captures consensual knowledge, that is, it is not private to some individual, but accepted by a group. An ontology describes the subject matter using the notions of concepts, instances, relations, functions, and axioms. Concepts in the ontology are organized in taxonomies through which inheritance mechanisms can be applied.

In order to come up with a consensual ontology of some domain, it is important that the people who have to use the ontology have a positive attitude towards it. Dictating the use of a particular ontology to people to which they have not contributed, is not likely to succeed. Preferably, an ontology is constructed in a

```
----------------------------------------
Concept: Component

Relation: Part-of
Number of arguments: 2
Type of argument #1: component
Type of argument #2: component
----------------------------------------
```

Figure 2: Part of a physical device ontology.

collaborative effort of domain experts, representatives of end users and IT specialists. Such a joint effort requires (1) the use of a methodology that guides the ontology development process and (2) tools to inspect browse, codify, modify and down-load the ontology. Examples of such methodologies include METHONTOLOGY [FGJ97, GP98], Uschold's and Gruninger's methodology [UG96] and that of Gruninger and Fox [GF95]. The tool we use is the Ontology Server [FFR97], which is an interactive environment especially useful for updating, maintaining and browser ontologies. Ontolingua ontologies can be translated to different languages, including Prolog, CORBA's IDL [OHE96], CLIPS, LOOM [Mac91], KIF, Epikit [Gen92]. Ontologies built in Ontolingua use the Frame Ontology [Gru93], which is written in KIF (Knowledge Interchange Format) [GF92]. 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.

As an example, consider the context of the automobile industry. Here, the ontology would include, among others, terms related to mechanical and hydraulic devices. In the mechanical device ontology, examples of classes are "cylinder", "crankshaft" and "engine". An example of a binary relation is "part-of", which could be used to say that the cylinder is part-of the engine. The hydraulic device ontology could include the class "pipe" and the ternary relation "connection" to express that two mechanical devices are connected by a given kind of pipe. Note that the terms "cylinder", "crankshaft" and "engine" will be part of an ontology in the domain of "mechanical devices", while the concept "component" and the relation "part-of" will belong to a meta-ontology, applicable to any kind of physical device. Figure 2 and Figure 3 illustrate respectively part of a physical device ontology and part of a mechanical device ontology.

In a human resource management context, classes could be "employee", "manager", "project leader", "skill", "area of expertise". Applied to a concrete company, an ontology can fulfill the role of an "enterprise knowledge map".

```
----------------------------------------
Concept: Cylinder
Subclass-of: Component
Part-of: Engine

Concept: Crankshaft
Subclass-of: Component
Part-of: Engine

Concept: Engine
Subclass-of: Component
----------------------------------------
```

Figure 3: Part of a mechanical device ontology.

```
----------------------------------------
<html>
<head><TITLE> Mr. Paton </TITLE>
<a ONTO="page:ProjectLeader"> </a>
</head>
<body>
.....
<a ONTO="page[lastName=body]">Paton</a>
.....
</body>
</html>
----------------------------------------
```

Figure 4: A simple extension to HTML. The `onto` attribute allows to express ontological information in HTML pages.

2.2 Characterizing the knowledge

As already mentioned briefly, in our approach, the knowledge to be managed is distributively organized in HTML pages (e.g. in a company's intranet or on the WWW). The relevant knowledge can thus be maintained distributively by different persons (the responsible persons for the respective HTML pages). The subject matter knowledge within the HTML pages is annotated using the ontology as a scheme for expressing meta-data. For example, in the human resource management domain, the homepage of Mr. Paton would state that he is a project leader. We thus add meta-data to make this explicit. In our approach we do this by extending HTML with a new attribute of the "anchor" tag: the `onto` attribute. Figure 4 gives a simple illustration.

The HTML code in Figure 4 states that the URL of the page containing the information represents a `ProjectLeader` (a term defined in the ontology). `Page` in ⟨`a ONTO="page:ProjectLeader"`⟩ 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`⟩. The `onto` attribute does not affect the visualization of HTML documents in standard web browsers such as Netscape or Explorer. The only thing that it does, is that it makes visible the subject matter knowledge for the intelligent web crawler. This small extension of HTML has been chosen to keep annotation as simple as possible. Also, it enables the direct usage (actually, reuse) of textual knowledge already in the body of the anchor. This prevents the knowledge annotater from representing the same piece of information again (the text `Paton` appearing as the value of meta-data `onto` above, is the same text as is visualized in the browser). This simple solution suffices for our approach because the HTML pages only contain factual knowledge [FDES98].

2.3 Intelligent knowledge retrieval

Having discussed the ontology and the annotated HTML pages, we will now turn to using this knowledge for intelligent retrieval. We use the ontology-based brokering service Ontobroker[1], which consists of three main elements: a web crawler (called Ontocrawler), an inference engine and a query interface [FDES98].

First, Ontocrawler searches through the annotated pages (e.g. on an intranet) and collects the annotated knowledge fragments. Second, it translates the annotated knowledge fragments into facts formulated in the representation language used by Ontobroker. Neither the inference engine nor the querying user have to be aware of the syntactical way in which the facts are represented on the Internet. Only the annotaters have to use the annotation language.

The inference engine receives the query of a user and exploits two information sources for deriving an answer: the ontology of the subject matter and the facts that were found by Ontocrawler. The basic inference mechanism of the inference engine is the derivation of a minimal model of a set of Horn clauses (see [FDES98] for details). This resembles intelligent reasoning as known in Knowledge-Based Systems, with the difference that the instances of the knowledge base are now distributed over the different HTML pages. The query interface of Ontobroker consists of a hyperbolic visualization of the ontology and a table format in which the user can easily compose queries (see Figure 7). This prevents the user from having to know all the classes and attributes of the ontology.

3 Proof of concept: $(KA)^2$

In order to investigate the feasibility of our approach, we are performing a large-scale initiative on the Web, where the subject matter is the scientific knowledge acquisition community: the Knowledge Annotation Initiative of the Knowledge Acquisition Community[2]: $(KA)^2$. We describe thus a virtual organization consisting of researchers, universities, projects, publications, etc. The information resides at the World-Wide Web in the homepages of the KA researchers where they publish information about their affiliation, projects, publications, research interests, etc. [BF98].

From a concrete knowledge management point of view, the $(KA)^2$ initiative is not an esoteric, academic toy example. Imagine a large multinational with thousands of employees world wide. For such a large organization, effective human resource management (HRM) is of vital importance. However, finding "who knows what" in large organizations has always been a time-intensive process. A knowledge management system that allows to find adequate people based on their skills, experience and area of expertise would certainly be of high value. For large companies that have an organization-wide intranet, our approach is a real possibility to enhance the HRM task. It allows improvement of the precision, recall and presentation of the results of searches on an intranet or the WWW.

Notice, however, that the fact that $(KA)^2$ is naturally related to the HRM task, does not imply that it is limited to this knowledge management task. In principle, the subject matter of our approach can concern any kind of company-vital knowledge that need to be managed more effectively.

3.1 Ontological engineering in $(KA)^2$

In $(KA)^2$, we build an ontology of the KA community (cf. an "enterprise knowledge map"). Since an ontology should capture consensual knowledge, in $(KA)^2$, several researchers cooperate together – at different locations – to construct the ontology. In this way, we ensure that the ontology will be accepted by a majority of KA researchers. The current 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. The current version of the ontology can be viewed at the European mirror site in Madrid of the Ontology Server of Stanford University[3]. Login as "ontologias-ka2" with password "adieu007", and then load one of the seven sub-ontologies of the KA community. For illustration purpose, we include here examples of two sub-ontologies of the KA ontology: 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 Ontol-

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

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

[3] URL is http://www-ksl-svc-lia.dia.fi.upm.es:5915/

ogy Server). Indentation denotes the subclass-of relation.

```
Class hierarchy (10 classes defined):

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

```
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 Annotating pages in $(\mathrm{KA})^2$

Annotating HTML pages in $(\mathrm{KA})^2$ means that each participating researcher in the KA community has to annotate the relevant knowledge in his or her homepage environment. Figure 5 illustrates fragments of an annotated homepage of a researcher using the `onto` attribute. `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. Figure 6 illustrates the annotation of a publication. The annotation process looks like a tedious and error-prone task. Our experience is that it takes roughly one hour to annotate five pages. At the Ontobroker site, an annotation checker is available, and if needed, personal support can be given. In spite of the amount of work involved, there is one important factor that may make people be willing to annotate their homepages, and that is self-publicity. By annotating pages, researchers make themselves more visible to others, which enhances the likelihood that others will use and refer to their work, which – in the academic world – is a good thing. In Section 5, we come back to this issue.

3.3 Querying the KA community

In $(\mathrm{KA})^2$, in order for Ontocrawler to collect the knowledge from HTML pages, researchers have to register their pages. That is, they have to tell Ontocrawler which URLs it needs to visit. Once that is done, intelligent knowledge retrieval is possible. Users are freed from knowing the specific querying language through a user interface comprising a hyperbolic visualization of the ontology linked with a table interface (see Figure 7 and Figure 8). In the hyperbolic view, the ontology can be moved around with the effect that concepts dragged to the center are enlarged while peripheral concepts are reduced in size. If the user clicks on a concept, it is passed to the table in Figure 8. Specific attributes of the selected concepts can now be chosen (such as "lastname" and "email"). In this way, users can compose their query by browsing and clicking, with a minimum amount of typing. The table also allows the construction of composite queries using conjunctives such as `and`, `or`, `and not`, `or not`.

We can for instance ask for all researchers in the KA community. The answer would not only include researchers who have their homepage annotated, but also additional researchers that cooperate with these researchers. The ontology defines cooperation between researchers, which enables the following deduction: if X cooperates with Y then X and Y must be researchers. Ontobroker uses this type information, not for consistency checking (which would not be a very good idea in an open web environment), but for abductively deriving new facts (i.e. Y is also a researcher). This example illustrates that it is possible to access knowledge that is not explicitly represented, which is an important advantage of our approach compared to

```
---------------------------------------------------------------------
<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 5: 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.

```
------------------------------------------------------------------------------------------
                                   <a name="Benjamins:97a" ONTO="name:JournalArticle"></a>
                                   <a name="Benjamins:97a" ONTO="name[author=href]"
                                   href="http://www.iiia.csic.es/~richard/index.html">
V. R. Benjamins                                                                       </a>
and
                                   <a name="Benjamins:97a" ONTO="name[author=body]">
M. Aben                                                                               </a>
,                                  <a name="Benjamins:97a" ONTO="name[title=body]">
Structure-Preserving KBS Development through
Reusable Libraries: a Case-Study in Diagnosis.                                        </a>
                                   <a name="Benjamins:97a" ONTO="name[journal=body]">
IJHCS                                                                                 </a>
International Journal of Human-Computer Studies, Vol 47, pages
                                   <a name="Benjamins:97a" ONTO="name[firstPage=body]">
259                                                                                   </a>
-                                  <a name="Benjamins:97a" ONTO="name[lastPage=body]">
288                                                                                   </a>
,                                  <a name="Benjamins:97a" ONTO="name[year=body]">
1997                                                                                  </a>
                                   <a name="Benjamins:97a" ONTO="name[onlineVersion=href]"
HREF="http://www.iiia.csic.es/~richard/postscripts/ijhcs.ps">
(draft version)                                                                       </a>
------------------------------------------------------------------------------------------
```

Figure 6: Example of an annotated publication. All values of the ONTO attribute belong to the ontology of the knowledge acquisition community. The actual knowledge (the instances) representing the publication appears at the left-hand side, the right part contains the annotation code.

keyword-based search. We could also ask for all researchers that have worked together in some project, or for abstracts of all papers on a particular topic. More examples of queries to the knowledge acquisition community can be obtained through Ontobroker's homepage.

3.4 Some facts

The current version (July, 1998) of the ontology contains 80 classes, 27 axioms and 100 attributes, which are used to annotate 1000 facts of 17 researchers.

4 Feasibility of knowledge management systems

In order to say something about the feasibility of a horizontal knowledge management system such as we have described, we have to consider the risks involved. Risks come from various resources, and we will discuss them resource-wise; technological risks, and social and organizational risks.

4.1 Technological risks

From a technology point of view, there are several factors that endanger the success of our knowledge management approach.

- First of all, such an initiative is likely to fail without dedicated tools to support the tasks involved. In particular tools are needed for (1) constructing and maintaining the ontology, (2) annotating information sources and (3) querying them (see Figure 9). Currently, we use ODE [BFGPGP98] (ontological design environment), which allows one to specify ontologies at the conceptual level by completing tables, rather than at the implementation level. From these tables, ODE is able to generate the Ontolingua code of the ontology. We need, however, to complement this with more support. For instance, Webonto [Dom98] enables collaborative construction of ontologies over the WWW. Concerning the annotation process, we would need a tool that visualizes both the ontology and the HTML page to be annotated. Selecting a fragment of the HTML page and then clicking on a term of the ontology should have to effect to include the corresponding `onto` attribute/value in the HTML page.
- Similarly, tools are needed for updating knowledge, both at the instance level, where researchers annotate their personal pages, as well at the ontology level. Changes to the ontology might have dramatic consequences for updating the annotations in HTML pages, especially in pages that are annotated with an ontology term that becomes obsolete. We do not have a crystallized answer for this problem yet, but it certainly forms a risk to be considered. One possibility would be to use so-called XML[4] "name spaces" that let you include in a document (then an XML page rather than

[4] URL of XML is

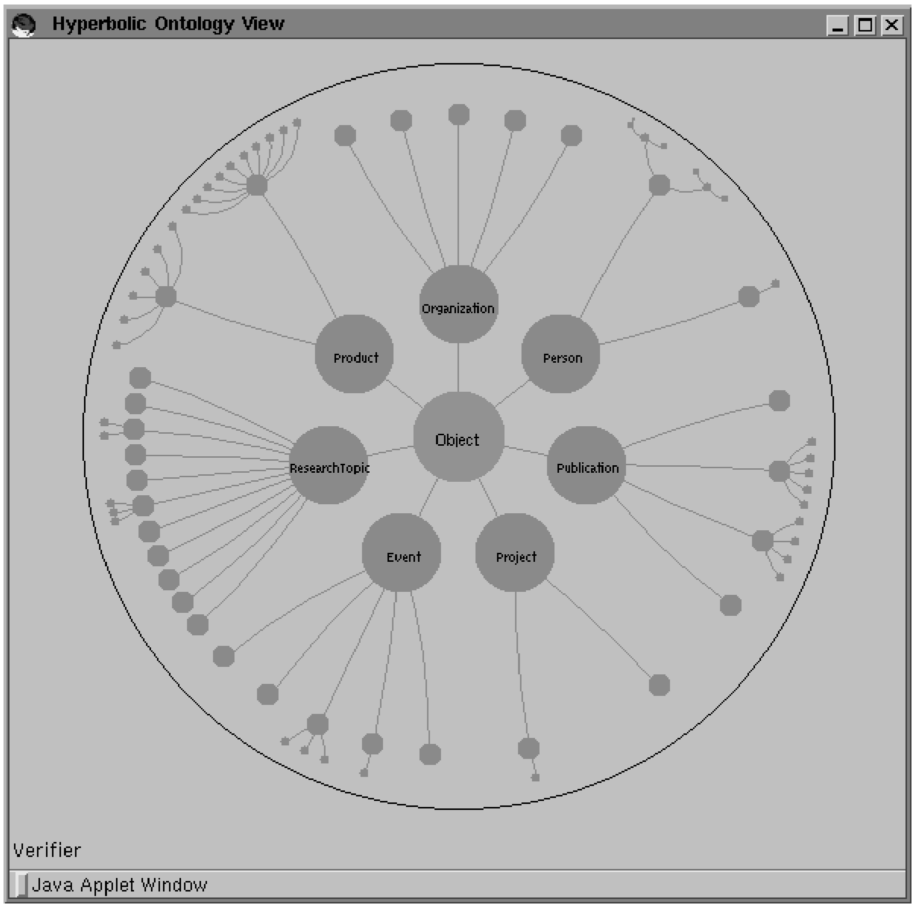

Figure 7: The hyperbolic query interface. Clicking on a node makes the corresponding class appear in the table interface of Figure 8.

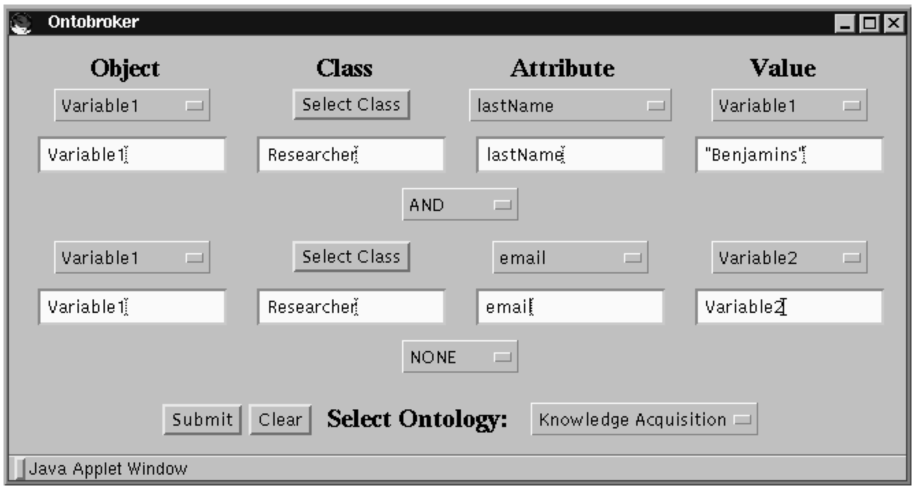

Figure 8: The table query interface.

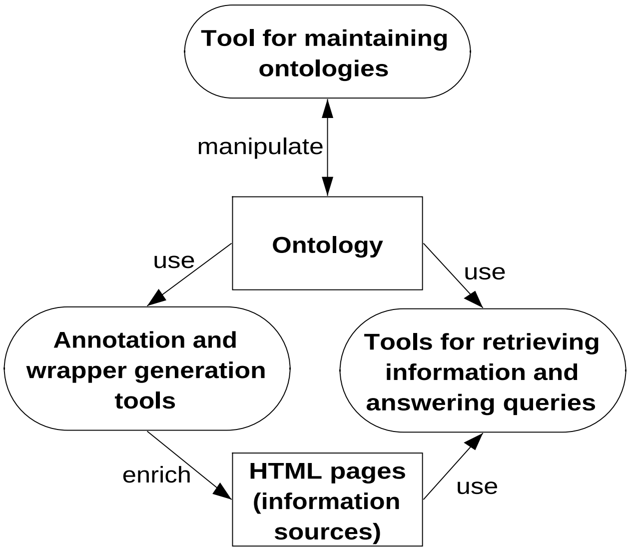

Figure 9: Tools to support knowledge management.

HTML) where the definition of a terms comes from.

- What happens when the knowledge is spread over ten thousands of HTML pages? Apart from the updating problem (see above) also the intelligent reasoning part might become a problem. This is a familiar problem in KBS research, when algorithms developed and tested on toy domains have to scale-up to real world applications.
- How does our simple extension to HTML relate to new technologies for the Web, that might make HTML obsolete? The W3C – the international *World-Wide Web Consortium* for developing and promoting standards for the Web – currently introduces the eXtensible Markup Language (XML) as a new standard for expressing the structure of web documents, and the Resource Description Framework[5] (RDF) for describing the semantics of web documents. When a final version of RDF is recommended by the W3C, we will implement a wrapper that automatically generates RDF definitions from our annotations [FDES98].

4.2 Social and organizational risks

- Without participating researchers, the $(KA)^2$ initiative would certainly fail. However, the nature of the initiative is such that participation is rewarding. It is a self-promoting activity. That is, researchers are better of if they participate because other researchers and outsiders can better and more easily find their work.

http://www.w3.org/XML/
[5]URL of RDF is
http://www.w3.org/Metadata/RDF/Group/WD-rdf-syntax

- In many companies, the mentality is competitive rather than collaborative. In other words: "If my colleague wins, then I loose." And: "If I make my knowledge available to others, then others will profit from that, and there will be a risk that they outperform me." This mentality is a real threat to success of knowledge management initiatives. Increasingly more companies become aware that a collaborative mentality leads to better results than competitive thinking [Cov89]. Organizations can stimulate collaborative thinking by changing the incentive system (such as making it financially rewarding to share knowledge).
- Given the high workload of today's employees, it may be easily felt that contributing to a knowledge management effort is a waste of time, or at least does not have priority. This is killing for any knowledge management initiative. Organizations should therefore reward knowledge management contributions equally as results that lead to direct profits. In addition, an effort should be made to reuse existing documents such that knowledge workers do not have the impression that they have to duplicate knowledge. There exist already tools to generate HTML pages from a variety of other formats (MSWord, Email, etc.).

5 Discussion and conclusion

5.1 Summary

In this article, we presented a knowledge engineering approach to knowledge management, which is based on many years of experience in dealing with knowledge. If we relate our work to the four knowledge management actions mentioned in the introduction, we get the following:

- Knowledge gathering is performed from existing HTML pages (knowledge annotation).
- Knowledge organization and structuring is done through an ontology (ontological engineering).
- Knowledge refinement is performed distributively by each worker (update annotations).
- Knowledge distribution is done by a web crawler that gives intelligent access to the knowledge that is "managed". This is a pull approach where users take the initiative when they need knowledge. However, the work presented here could as well be used for a push approach.

5.2 A social effort

We noted that knowledge management essentially involves people, and therefore any knowledge management effort is doomed to fail if human factors are not taken seriously. Knowledge management only works if people cooperate and are willing to share their knowledge. One way to stimulate sharing of knowledge is to change the incentive system accordingly.

5.3 The knowledge chain

An important framework in knowledge management is the so-called "Knowledge Chain" [Kou97] which refers to the adaptability of an organization to an ever changing market. The knowledge chain consists of four stages which are walked through in a circular way.

- Internal Awareness refers to the organization's ability to understand itself in terms of the skills and competencies that it possesses, and not so much in terms of its products.
- Internal responsiveness is concerned with the translation of internal awareness (skills and competencies) into teams with the skills and tools to bring a product to market.
- External responsiveness makes the difference for the organization's success or failure. It is the organization's ability to take quick and adequate decisions based on a corporate instinct, rather than to go through a long bureaucratic process before acting.
- External awareness represents an organization's ability to understand how the market perceives the value associated with its products and services as well as the changing directions and requirements of its markets. When coupled with internal awareness, external awareness may lead to entirely new markets.

Our approach contributes directly to the first two stages: internal awareness and responsiveness. An ontological engineering process, as is part of our approach, results in a knowledge map of the organization. This "map" certainly contributes to the internal awareness of an organization. The annotation process provides all instances of the knowledge map. Concerning the organization's internal responsiveness, if each employee annotates his or her homepage with skills, competencies and areas of expertise, it will be easy to find quickly and accurately the right persons for forming adequate teams.

5.4 Ontology-based versus keyword-based retrieval

One could argue that, if all the knowledge is available in HTML documents, then why use an ontology to annotate the information in the pages? After all, the annotation effort is considerable. Why not use general search engines for keyword-based searching through the HTML pages? As everybody might have experienced, keyword-based search easily leads to an overwhelming amount of answers (references to web documents). In other words, there is an information overload [O'L97], which makes it hard to find exactly what one is looking for and to get rid of nonsense (with respect to the query). Although search engines get increasingly smarter, we expect that there will be a limit to such keyword-based information retrieval. Moreover, current keyword-based search approaches do not allow to present information collected from distributive locations in a coherent way to users, since there is no knowledge of how the retrieved information relates to each other. Ontology-based retrieval does allow for this, through the ontology. Finally, the ontology-based approach allows to access implicit knowledge, which is definitely beyond the capacity of keyword-based approaches.

To reduce the annotation effort, machine learning techniques can be used that exploit ontologies to automatically classify textual information [CDF+98]. Moreover, wrappers can be built that extract the semantics of web documents based on regularities in their structure, format and content. Again, machine learning techniques can be used to semi-automatically build such wrappers [AK97, KWD97]. Clearly, this is an important research line to embark on.

5.5 Related work

There is a huge research effort going on about meta-data for web documents (e.g., XML, RDF, WebSQL, Dublin Core). 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 another to our approach and to $(KA)^2$ in particular. We already mentioned that the Resource Description Framework (RDF) may provide an alternative syntax for writing ontological annotations 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. See http://www.aifb.uni-karlsruhe.de/WBS/broker/inhalt-wp.html for brief overviews of these related projects and links to their homepages.

Acknowledgment

We thank Stefan Decker for his technical contribution to Ontobroker. Richard Benjamins is supported by the Netherlands Computer Science Research Foundation with financial support from the Netherlands Organization for Scientific Research (NWO).

References

[AK97] N. Ashish and C. A. Knoblock. Semi-automatic wrapper generation for internet information sources. In *Proceedings of the IFCIS Conference on Cooperative Information Systems (CoopIS)*, Charlston, South Carolina, 1997.

[APH98] J. Angus, J. Patel, and J. Harty. Knowledge management: Great concept ... but what is it. *Information Week*, 1998.

[BF98] V. R. Benjamins and D. Fensel. The ontological engineering initiative $(\mathrm{KA})^2$. In N. Guarino, editor, *Formal Ontology in Information Systems*, pages 287–301. IOS Press, 1998.

[BFGPGP98] M. Blázquez, M. Fernández, J. M. García-Pinar, and A. Gómez-Pérez. Building ontologies at the knowledge level using the ontology design environment. In *Proceedings of the Eleventh Workshop on Knowledge Acquisition, Modeling and Management, KAW'98*, Banff, Canada, 1998.

[Bor97] W. N. Borst. *Construction of Engineering Ontologies*. PhD thesis, University of Twente, Enschede, 1997.

[CDF+98] M. Craven, D. DiPasquo, D. Freitag, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery. Learning to extract symbolic knowledge from the world wide web. In *Proceedings of the 15th National Conference on AI (AAAI-98)*, Madison, Wisconsin, 1998.

[Cov89] R. Covey. *The seven habits of highly effective people*. Simon & Schuster, Inc., New York, 1989.

[Dom98] J. Domingue. Tadzebao and webonto: Discussing, browsing, and editing ontologies on the web. In *Proceedings of the Eleventh Workshop on Knowledge Acquisition, Modeling and Management, KAW'98*, Banff, Canada, 1998.

[FDES98] 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.

[FFR97] 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.

[FGJ97] M. Fernandez, A. Gómez Pérez, and N. Juristo. METHONTOLOGY: From ontological art towards ontological engineering. In *Spring Symposium Series on Ontological Engineering*, Stanford, 1997. AAAI Press.

[Gen92] M. R. Genesereth, editor. *The Epikit manual*. Epistmemics, Inc, Palo Alto, CA, 1992.

[GF92] 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/.

[GF95] M. Gruninger and M. Fox. Methodology for the design and evaluation of ontologies. In *Proceedings of the Workshop on Basic Ontological Issues in Knowledge Sharing held in conjunction with IJCAI-95*, Montreal, Canada, 1995.

[GP98] A. Gómez-Pérez. Knowledge sharing and reuse. In J. Liebowitz, editor, *The Handbook of Applied Expert Systems*. CRC, 1998.

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

[Gua95] N. Guarino. Formal ontology, conceptual analysis and knowledge representation. *International Journal of Human-Computer Studies*, 43(5/6):625–640, 1995. Special issue on The Role of Formal Ontology in the Information Technology.

[Kou97] Thomas Koulopoulos. Hooking into the knowledge chain. White paper at KMWorld.com, 1997.

[KWD97] N. Kushmerick, D. Weld, and R. Doorenbos. Wrapper induction for information extraction. In *Proceedings of the 15th International Joint Conference on AI (IJCAI-97)*, pages 729–735, Nagoya, Japan, 1997.

[Mac91] R. MacGregor. Inside the LOOM classifier. *SIGART Bulletin*, 2(3):70–76, June 1991.

[OHE96] R. Orfali, D. Harkey, and J. Edwards, editors. *The Essential Distributed Objects Survival Guide*. John Wiley & Sons, New York, 1996.

[O'L97] D. O'Leary. The internet, intranets, and the AI renaissance. *IEEE Computer*, 30(1):71–78, 1997.

[O'L98] D. O'Leary. Knowledge management: Taming the information beasts. *IEEE Intelligent Systems*, 13(3):30–48, 1998. Special Issue with three contributions.

[SAA+99] A. Th. Schreiber, J. M. Akkermans, A. A. Anjewierden, R. de Hoog, N. R. Shadbolt, W. Van de Velde, and B. J. Wielinga. *Engineering and Managing Knowledge, The CommonKADS methodology*. to appear, 1999.

[SBF98] R. Studer, V. R. Benjamins, and D. Fensel. Knowledge engineering, principles and methods. *Data and Knowledge Engineering*, 25:161–197, 1998.

[UG96] M. Uschold and M. Gruninger. Ontologies: principles, methods, and applications. *Knowledge Engineering Review*, 11(2):93–155, 1996.

[vHvdSK96] G. van Heijst, R. van de Spek, and E. Kruizinga. Organizing coporate memories. In B. R. Gaines and M. A. Musen, editors, *Proceedings of the 10th Banff Knowledge Acquisition for Knowledge-Based Systems Workshop*, pages 42.1–42.18, Alberta, Canada, 1996. SRDG Publications, University of Calgary.

[vSW97] G. van Heijst, A. T. Schreiber, and B. J. Wielinga. Using explicit ontologies in KBS development. *International Journal of Human-Computer Studies*, 46(2/3):183–292, 1997.

[Wii94] K. M. Wiig. *Knowledge Management, The central management focus for intelligent-acting organizations*. Schema Press Ltd., Arlington, TX, 1994.