OntoWeb – a Semantic Web Community Portal

Peter Spyns[2], Daniel Oberle[1], Raphael Volz[1,3], Jijuan Zheng[2], Mustafa Jarrar[2], York Sure[1], Rudi Studer[1,3], and Robert Meersman[2]

[1] Institute AIFB, University of Karlsruhe, D-76128 Karlsruhe, Germany
lastname@aifb.uni-karlsruhe.de

[2] STAR Lab, Vrije Universiteit Brussels,
Pleinlaan 2 Gebouw G-10, B-1050 Brussel
first.lastname@vub.ac.be

[3] FZI - Research Center for Information Technologies
Haid-und-Neu-Strasse 10-14, D-76131 Karlsruhe, Germany
lastname@fzi.de

Abstract. This paper describes a semantic portal through which knowledge can be gathered, stored, secured and accessed by members of a certain community. In particular, this portal takes into account companies and research institutes participating in the E.U. funded thematic network called OntoWeb. Ontology-based annotation of information is a prerequisite in order to offer the possibility of knowledge retrieval and extraction. The usage of well-defined semantics allows for the knowledge exchange between different OntoWeb community members. Thus, members are able to publish annotated information on the web, which is then crawled by a syndicator and stored in the portal's knowledge base. The backbone of the portal architecture consists of a knowledge base in which the ontology and the instances are stored and maintained. In addition, ontology-boosted query mechanisms and presentation facilities are provided.

1 Introduction

Although an ubiquitous and overwhelming amount of information is available at a snap of one's fingers, knowledge is not so easily retrievable. For knowledge is the result of an information processing activity. Knowledge has become a valuable asset for companies and institutions (or so-called communities in general) to such a degree that specific mechanisms have been put into place for the provision of high quality knowledge. Storing and aggregating knowledge may be one important aspect; accessing and finding appropriate knowledge is just as important. After all, how can one benefit from the knowledge available if one cannot find and retrieve it?

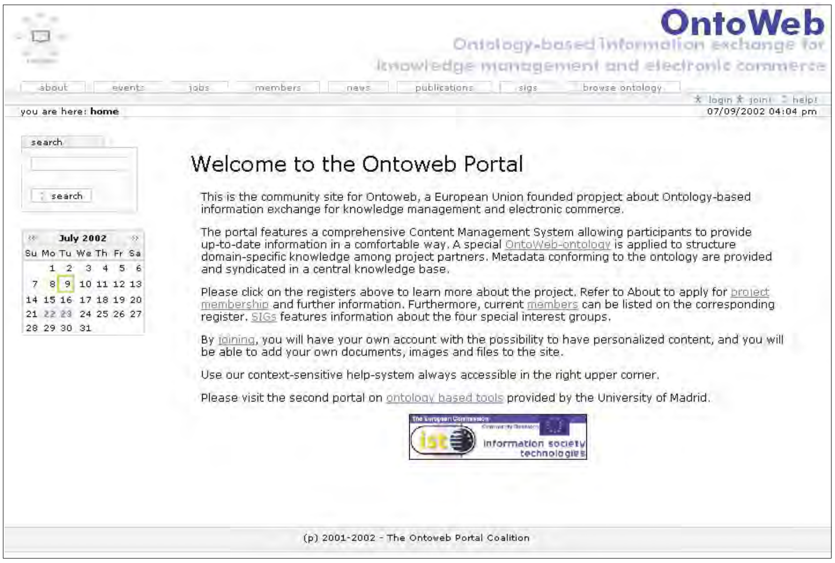
In this paper, work in progress on a semantic portal[4] is described through which knowledge can be gathered, stored, secured and accessed by members of a certain community (c.q. companies and research institutes working in the field of the Semantic Web

[4] The portal with its current test content can be accessed on http://ontoweb.aifb.uni-karlsruhe.de or http://starpc14.vub.ac.be:8000/OntoWeb/Browse/index.html.

and participating in the E.U. funded thematic network called OntoWeb [4]). It is an open community, i.e. new members can join at any time. The positive effects of the existence of such a portal are multiple. Only the most important ones will be mentioned. At a first stage, the portal serves as an inventory of knowledge available in the community. In the case of an Internet portal, knowledge has been made available outside of the organization of the original producer or owner. E.g., members of the community get a good overview of the skills and profiles of the various community members. In the case of an intranet, it may stimulate the communication between departments of a same company and support the local (technology) innovation management process.

Turning information into knowledge that suits the above mentioned situation, requires a shared conceptualization of the domain in question. In the present OntoWeb case, the domain spans a conceptualization of the OntoWeb organization (e.g., companies, research institutions, special interest groups etc.), of various kinds of documents (e.g., meeting minutes, deliverables, papers etc.), of events and their organizations (e.g., conferences, workshops, internal meetings etc.), of scientific results and material (e.g., cases, programs, etc.), and so forth. A formal version of such a shared conceptualization is commonly called an ontology [14]. When relating specific terms to concepts, a controlled vocabulary or some other common terminological framework can be created.

Ontology-based annotation of the community information is a prerequisite in order to offer the possibility of knowledge retrieval and extraction (also known as conceptual or intelligent search — cf. [11] as an example). The usage of well-defined semantics allows for the knowledge exchange between different OntoWeb community members. Members can publish annotated information on the web, which is then crawled by a syndicator and stored in the portal knowledge base.



Welcome to the Ontoweb Portal

This is the community site for Ontoweb, a European Union founded propject about Ontology-based information exchange for knowledge management and electronic commerce.

The portal features a comprehensive Content Management System allowing participants to provide up-to-date information in a comfortable way. A special OntoWeb-ontology is applied to structure domain-specific knowledge among project partners. Metadata conforming to the ontology are provided and syndicated in a central knowledge base.

Please click on the registers above to learn more about the project. Refer to About to apply for project membership and further information. Furthermore, current members can be listed on the corresponding register. SIGs features information about the four special interest groups.

By joining, you will have your own account with the possibility to have personalized content, and you will be able to add your own documents, images and files to the site.

Use our context-sensitive help-system always accessible in the right upper corner.

Please visit the second portal on ontology based tools provided by the University of Madrid.

(p) 2001-2002 - The Ontoweb Portal Coalition

Fig. 1. www.ontoweb.org - the OntoWeb portal

The paper is structured as follows: In section 2, we describe how information is semantically annotated on-site, crawled and subsequently aggregated (or syndicated) into a common database (cf. 2.1). An alternative is that the community members upload annotated information themselves. Therefore, we define a model for a publication workflow in subsection 2.2 and discuss the integration in the portal in 2.2. Section 3 deals with how the content (or community knowledge) can be accessed. By pointing and clicking, a user can browse the concepts of the ontology and the related instances (cf. subsection 3.1). He/She can enter one or more search terms in a query box (cf. subsection 3.2) or a form (cf. subsection 3.2). A short overview of related work is presented in section 4 before the future work on the OntoWeb semantic portal is sketched in section 5. Finally, section 6 contains some concluding remarks.

2 Content Provision

Basically, there are two ways of providing content to the OntoWeb portal. First, there is the syndication mechanism, automatically gathering metadata from participating sites. Second, the portal allows for content provision itself. Both possibilities are discussed in subsections 2.1 and 2.2, respectively.

2.1 Content Syndication

The portal allows centralized access to distributed information that has been provided by participants on their own sites. To facilitate this, participants can enrich resources located outside of the portal with metadata according to the shared OntoWeb ontology. This annotation process can be supported semi-automatically by the Ontomat Annotizer tool [9] for instance.

As depicted in Figure 2, syndicating information from participants is done by replicating their metadata. The information finds its way in the so-called DOGMA Server [10] which exploits a relational DBMS for storing and can be queried by users (cf. section 3 for a detailed discussion). Within the portal, authenticated users may generate content objects on their behalf (cf. subsection 2.2). As we use Zope [5] as underlying technology, such objects are stored in its respective database (so-called ZODB). Besides, metadata, both conforming to Dublin Core [16] as well as to the Ontoweb ontology, are generated for all the portal's objects. This can be achieved easily as all metadata are stored within Zope's own database. When adding new content to the portal, users have the possibility to supply metadata accordingly.

2.2 Content Objects

We acknowledge the fact that some members might not be able to publish data on the web on their own due to corporate restrictions or other reasons. Therefore OntoWeb participants staff members are provided with a personal space to create and manage content for the portal. To facilitate this, the portal includes a fully-fledged content management

[5] cf. http://www.zope.org

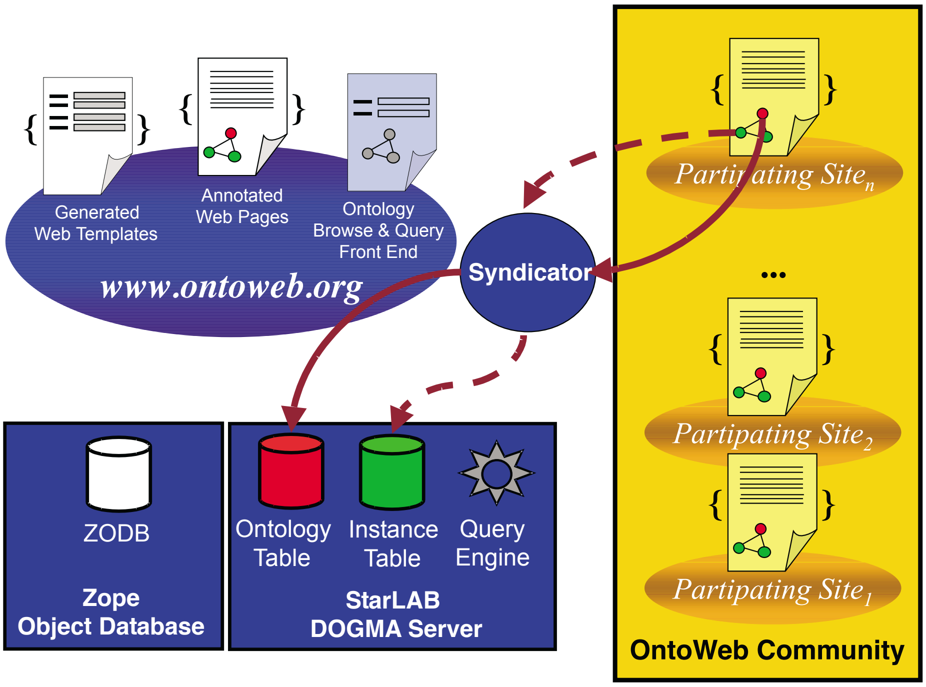


Fig. 2. Content Syndication

system. Additionally, all content created within the portal is automatically associated with the predefined OntoWeb design to achieve an integrated visual experience with a consistent appearance. In the personal space people can provide the following types of content:

- HTML-documents
- arbitrary files and folders
- selected predefined content types based on ontological concepts: Publications, News, Events, Scientific Events, Jobs, etc.

If a member chooses to create new content based on the predefined content types, appropriate metadata is automatically generated. Second, all content is associated with standard Dublin Core metadata to keep track of publishing information such as date of creation, last modification, authorship and subject classification.

Process Model for Publishing Workflows As mentioned in section 1, OntoWeb is an open community posing additional constraints since data that is (re)published through the portal could be provided by arbitrary people. In order to guarantee quality of data in such an environment, an additional model regulating the publishing process is required, which prevents foreseeable misuses. To support this requirement the established portal architecture was extended with a workflow component which regulates the publishing process. In the following we will begin with introducing the concept of a publishing workflow in general. Afterwards we explain how we instantiated this generic component in OntoWeb.

A publishing workflow is the series of interactions that should happen to complete the task of publishing data. Business organizations have many kinds of workflow. Our notion of workflow is centered around tasks. Workflows consist of several tasks and several transitions between these tasks. Additionally, workflows have the following characteristics: (i) they might involve several people, (ii) they might take a long time, (iii) they vary significantly in organizations and in the computer applications supporting these organizations respectively, (iv) sometimes information must be kept across states, and last but not least, (v) the communication between people must be supported in order to facilitate decision making. Thus, a workflow component must be customizable. It must support the assignment of tasks to (possibly multiple) individual users. In our architecture these users are grouped into roles. Tasks are represented within a workflow as a set of transitions which cause state changes. Each object in the system is assigned a state, which corresponds to the current position within the workflow and can be used to determine the possible transitions that can validly be applied to the object. This state is persistent supporting the second characteristic mentioned above. Due to the individuality of workflows within organizations and applications we propose a generic component that supports the creation and customization of several workflows. In fact, each concept in the ontology, which – as you might recall – is used to capture structured data within a portal, can be assigned a different workflow with different states, transitions and task assignments. As mentioned above, sometimes data is required to be kept across states[6]. To model this behavior, the state machine underlying our workflow model needs to keep information that "remembers" the past veto. Thus, variables are attached to objects and used to provide persistent information that transcends states. Within our approach, variables also serve the purpose of establishing a simple form of communication between the involved parties. Thus, each transition can attach comments to support the decision made by future actors. Also metadata like the time and initiator of a transition is kept within the system.

Workflows in OntoWeb Figure 3 depicts the default workflow within OntoWeb. There are three states: private, pending, and published. In the private state the respective object is only visible to the user himself, the pending state makes it visible to reviewers. In the published state, a given object is visible to all (possibly anonymous) users of the portal. If a user creates a new object[7], it is in private state. If the user has either a reviewer or a manager role the published state is immediately available through the publish transition. For normal users such a transition is not available. Instead, the object can only be sent for a review leading to the pending state. In the pending state either managers or reviewers can force the transition into the published state (by applying the

[6] For example, envision the process of passing bills in legislature, a bill might be allowed to be revised and resubmitted once it is vetoed, but only if it has been vetoed once. If it is vetoed a second time, it is rejected forever.

[7] Currently only within the portal, the content syndicated from other OntoWeb member web sites and within the databases is "trusted". We assume that this kind of data already went through some kind of review.

transition “publish”) or retract the object leading back to the private state. The reject transition deletes the object completely. When an object is in the private state, only the user who created it and users with manager roles can view and change it. Once an object is in published state, the modification by the user who created it resets the object into pending state, thus the modification must be reviewed again. This does not apply to modifications by site managers.

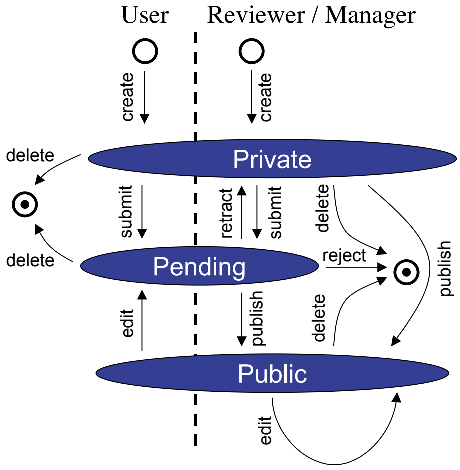


Fig. 3. OntoWeb Publishing workflow

3 Content Accessing

The hypothesis is that using an ontology results in an improved query refinement compared with a conventional keyword-based search. A (partial) validation of this hypothesis can be found in [8]. The browse and query facility has been developed as a highly generic system that offers exploration of the available information at the conceptual level. The semantic relationships are exploited to navigate through the application domain. As it concerns a shared ”mental map”, users are able to locate and find the desired information more rapidly. The main distinctions made when presenting the information to a user are between the sub- and superconcepts and the literal and non-literal properties of the different concepts. Currently, the user interface is work in progress.

3.1 Browsing

When browsing the semantic portal one can distinguish between browsing instances or instance details. In the case of browsing instance overviews, the portal displays collections of instances according to the user’s selection. When viewing instance details, the user is presented with detailed information on a particular instance. Links to related instances are grouped according to the community ontology.

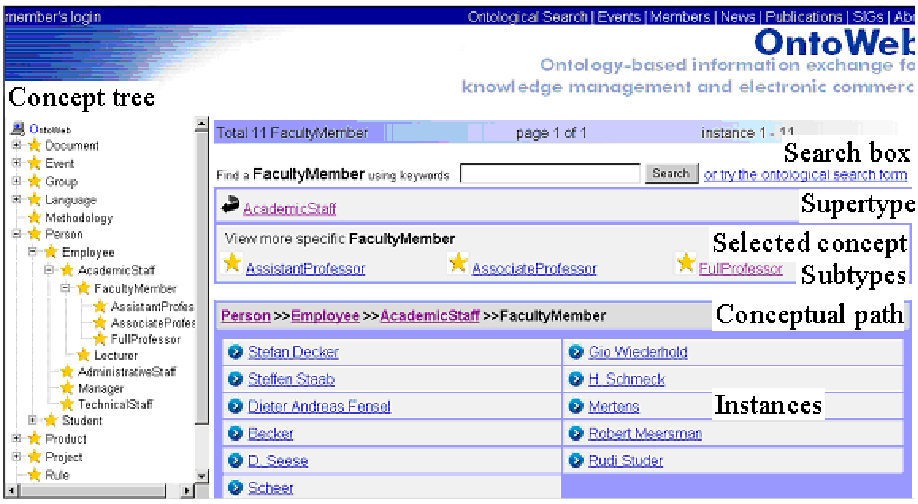


Fig. 4. Instance overview

Instance overview The hierarchical organization of the different concepts in the ontology is represented by a dynamic tree (see Figure 4). A user can view instances belonging to a concept from the tree (in the left pane) by expanding the tree nodes and clicking the concept of interest. The instances of this concept will then be displayed (in the right pane). By moving up and down the concept tree, a user can generalize or specialize instances. By clicking on a subtype (of the tree or in the conceptual path), the query precision should improve. This is because the instances of the supertype (i.e. the concept originally selected), including all the instances of its subtypes that do not belong to the subtype newly selected, are excluded from the result. Generalization (i.e. moving up one level in the hierarchy or clicking on the supertype displayed) on the other hand broadens the scope of the query, exploiting the concept hierarchy to expand the query to all instances of the siblings (and their subtypes) of the concept originally of interest to the user (cf. also [2].

Instance details When viewing the detailed information for a particular instance, a distinction is made between literal and non-literal properties of concepts. While the literal properties or attributes provide a user with detailed information, the non-literal properties or relationships with other concepts (and their instances) are shown as hyperlinks, enabling a user to jump to instances of related concepts. Attributes are displayed at the top of the page. These concern e.g., in the case of a person, the name, telephone number and email. . . All the relevant conceptual relationships are displayed in the lower part of the page (with an overview in the middle). They point to instances of related concepts presented at the bottom of the page that are grouped by relationship (cf. Figure 5).

3.2 Querying

Next to the browsing of the ontology and related instances, a user may opt at any moment to enter one or more search terms. This can be considered as a conceptually driven form of interactive query refinement.

Term based The portal offers a keyword based global search. The instances retrieved are presented to the user grouped by links pointing to the instance details page. The concept tree and conceptual path pane are dynamically adapted to the query results. When a user enters multiple keywords, the engine searches for paths between instances containing the different keywords and, if found, presents these paths to the user (cf. Figure 6). When a query is executed from an instance overview page, the results only include instances of the previously selected concept (and its subtypes).

Template based The form-based search allows for the construction of query paths across the ontology. A user is presented with a search form containing text boxes in which attribute values can be specified. Buttons labelled with a concept give access to other forms that can be used to specify related instances. For each node in the path, a user can add restrictions on the property values. The labels for the input boxes and the buttons are dynamically adapted (cf. Figure 7). They represent the shared definition of the current concept (shown on the titlebar).

4 Related Work

Using an ontology to support the access of content has been discussed before. E.g., the so-called Yahoo-a-lizer [6] transforms a knowledge base into a set of XML pages that are structured like the term hierarchy of Yahoo. These XML-files are translated via an XSL-stylesheet into ordinary HTML. Within Ontobroker-based web portals [5], a Hyperbolic View Applet allows for graphical access to an ontology and its knowledge base. Another related work is KAON Portal[8] which takes an ontology and creates a standard Web interface out of it. The OntoSeek [8] prototype uses a linguistic ontology and structured content representations to search yellow pages and product catalogs.

Given the difficulties with managing complex Web content, several papers tried to facilitate database technology to simplify the creation and maintenance of data-intensive web-sites. OntoWeb implements our framework for a SEmantic portAL, viz. SEAL [12], that relies on standard Semantic Web technologies. Other systems, such as ARANEUS [13] and AutoWeb [3], take a declarative approach, i.e. they introduce their own data models and query languages, although all approaches share the idea to provide high-level descriptions of web-sites by distinct orthogonal dimensions. The idea of leveraging mediation technologies for the acquisition of data is also found in approaches like Strudel [7] and Tiramisu [1], they propose a separation according to the aforementioned task profiles as well. Strudel does not concern the aspects of site

[8] cf. http://kaon.semanticweb.org/Portal

maintenance and personalization. It is actually only an implementation tool, not a management system.

The importance of conceptual indexing for information retrieval has been acknowledged since quite some time in the medical information processing field [15,**?**]. However, from our point of view the OntoWeb portal is rather unique with respect to the collection of methods used and the functionality provided.

5 Future Work

A next important step to take is to enter a significantly large amount of real life data in the instance base so that a truly useful knowledge base is created. Before doing that, an update of the ontology is foreseen as well. As a direct result, multiple inheritance will be allowed (and displayed in the ”tree” and conceptual path panes). As a general consideration, the user interface will be refined as well. After these steps, a large-scale assessment on the strengths and flaws (also as perceived by end-users) of the portal becomes possible.

Other topics for future work include semantic bookmarks. A semantic bookmark can be considered as stored query of the ontology and instance base as well as over the object base of the portal. The results can be enhanced by taking into account the concept- and property-hierarchies. Bookmarks already there can be combined conjunctively or disjunctively and so on. Another envisioned improvement are so-called push-services. Such notify the user if a certain resource has been changed.

6 Conclusion

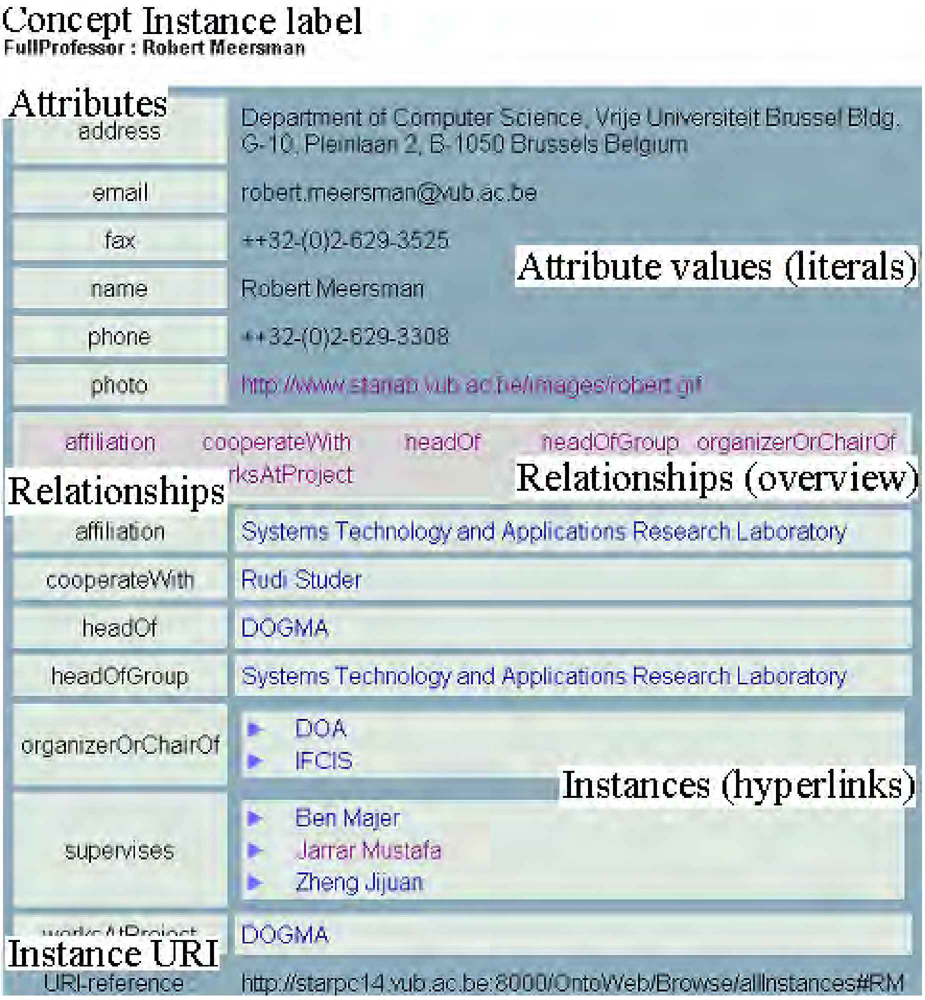
In this paper, a semantic portal has been presented. In particular, the components for content provision and access have been discussed in detail. It is our believe that the OntoWeb members will benefit from this portal in terms of a higher quality knowledge exchange in the semantic web community. As such, the portal serves as practical illustration and application of the scientific ideas put forward by the community members.

Acknowledgment:We like to thank Ben Majer (V.U.B. — STAR Lab) for his fruitful discussions and implementation work. Parts of the research presented here have been funded by the E.U. Thematic Network OntoWeb (IST-2000-25056), the V.U.B. and AIFB internal research grants.

References

1. C. R. Anderson, A. Y. Levy, and D. S. Weld. Declarative web site management with Tiramisu. In *ACM SIGMOD Workshop on the Web and Databases - WebDB99*, pages 19–24, 1999.
2. A. Aronson and T. Rindflesch. Query expansion using the UMLS. In R. Masys, editor, *Proceedings of the AMIA Annual Fall Symposium 97, JAMIA Suppl*, pages 485–489. AMIA, 1997.

3. S. Ceri, P. Fraternali, and A. Bongio. Web modeling language (WebML): a modeling language for designing web sites. In *WWW9 Conference, Amsterdam, May 2000*, 2000.
4. OntoWeb Consortium. Ontology-based information exchange for knowledge management and electronic commerce - IST-2000-25056. http://www.ontoweb.org, 2001.
5. S. Decker, M. Erdmann, D. Fensel, and R. Studer. Ontobroker: Ontology Based Access to Distributed and Semi-Structured Information. In R. Meersman et al., editors, *Database Semantics: Semantic Issues in Multimedia Systems*, pages 351–369. Kluwer Academic Publisher, 1999.
6. Michael Erdmann. *Ontologien zur konzeptuellen Modellierung der Semantik von XML*. Isbn: 3831126356, University of Karlsruhe, 10 2001.
7. M. F. Fernandez, D. Florescu, A. Y. Levy, and D. Suciu. Declarative specification of web sites with Strudel. *VLDB Journal*, 9(1):38–55, 2000.
8. N. Guarino, C. Masolo, and G. Vetere. Ontoseek: Content-based access to the web. *IEEE Intelligent Systems*, May-June4-5:70–80, 1999.
9. S. Handschuh and S. Staab. Authoring and annotation of web pages in CREAM. In *The Eleventh International World Wide Web Conference (WWW2002), Honolulu, Hawaii, USA 7-11 May*, 2002. To appear.
10. M. Jarrar and R. Meersman. An Architecture and Toolset for Practical Ontology Engineering and Deployment: the DOGMA Approach. In *Proceedings of the International Conference on Ontologies, Databases and Applications of Semantics (ODBase 02)*, 2002. (in press).
11. H. Lowe, I. Antipov, W. Hersh, and C. Arnott Smith. Towards knowledge-based retrieval of medical images. the role of semantic indexing, image content representation and knowledge-based retrieval. In C. Chute, editor, *Proceedings of the 1998 AMIA Annual Fall Symposium*, pages 882–886. AMIA, Henley & Belfus, Philadelphia, 1998.
12. A. Maedche, S. Staab, R. Studer, Y. Sure, and R. Volz. Seal — tying up information integration and web site management by ontologies. *IEEE Data Engineering Bulletin*, 25(1):10–17, March 2002.
13. G. Mecca, P. Merialdo, P. Atzeni, and V. Crescenzi. The (short) Araneus guide to website development. In *Second Intern. Workshop on the Web and Databases (WebDB'99) in conjunction with SIGMOD'99*, May 1999.
14. R. Meersman and M. Jarrar. Scalability and reusable in ontology modeling. In *Proceedings of the International conference on Infrastructure for e-Business, e-Education, e-Science, and e-Medicine (SSGRR2002s)*, 2002. (in press).
15. T. Rindflesch and A. Aronson. Semantic processing in information retrieval. In C. Safran, editor, *Seventeenth Annual Symposium on Computer Applications in Medical Care (SCAMC 93)*, pages 611–615. McGraw-Hill Inc., New York, 1993.
16. S. Weibel, J. Kunze, C. Lagoze, and M. Wolf. *Dublin Core Metadata for Resource Discovery*. Number 2413 in IETF. The Internet Society, September 1998.



Concept Instance label

FullProfessor : Robert Meersman

Attributes

Attribute values (literals)

address	Department of Computer Science, Vrije Universiteit Brussel Bldg. G-10, Pleinlaan 2, B-1050 Brussels Belgium
email	robert.meersman@vub.ac.be
fax	++32-(0)2-629-3525
name	Robert Meersman
phone	++32-(0)2-629-3308
photo	http://www.starlab.vub.ac.be/images/robert.gif

Relationships (overview)

affiliation cooperateWith headOf headOfGroup organizerOrChairOf rksAtProject

Relationships

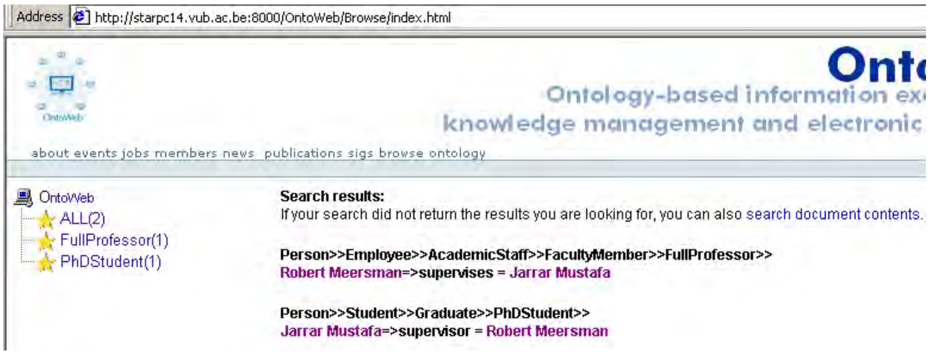
Instances (hyperlinks)

affiliation	Systems Technology and Applications Research Laboratory
cooperateWith	Rudi Studer
headOf	DOGMA
headOfGroup	Systems Technology and Applications Research Laboratory
organizerOrChairOf	▶ DOA ▶ IFCIS
supervises	▶ Ben Majer ▶ Jarrar Mustafa ▶ Zheng Jijuan
worksAtProject	DOGMA

Instance URI

URI-reference	http://starpc14.vub.ac.be:8000/OntoWeb/Browse/allInstances#RM

Fig. 5. Instance details



Address http://starpc14.vub.ac.be:8000/OntoWeb/Browse/index.html

Onto
Ontology-based information ex
knowledge management and electronic

about events jobs members news publications sigs browse ontology

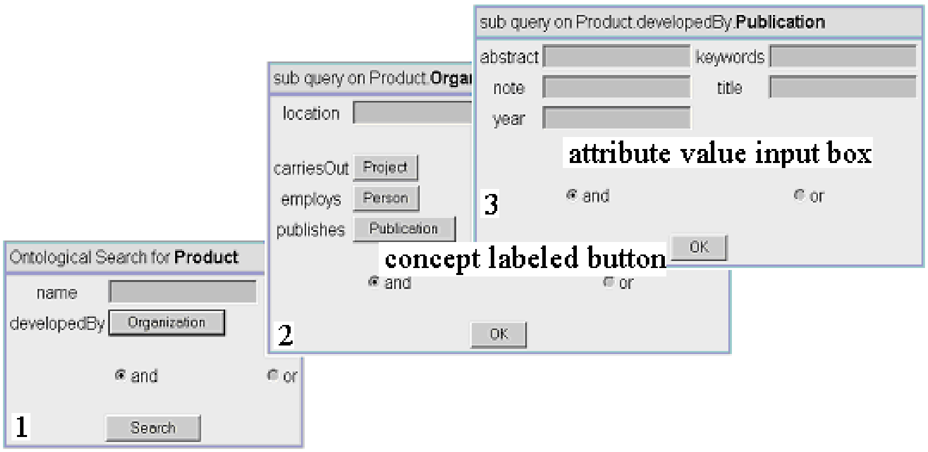
OntoWeb
- ALL(2)
- FullProfessor(1)
- PhDStudent(1)

Search results:
If your search did not return the results you are looking for, you can also search document contents.

Person>>Employee>>AcademicStaff>>FacultyMember>>FullProfessor>>
Robert Meersman=>supervises = Jarrar Mustafa

Person>>Student>>Graduate>>PhDStudent>>
Jarrar Mustafa=>supervisor = Robert Meersman

Fig. 6. Keyword based semantic search results



1 Ontological Search for **Product**
name
developedBy Organization
● and ○ or
Search

2 sub query on Product.**Orga**
location
carriesOut Project
employs Person
publishes Publication
● and ○ or
OK

3 sub query on Product.developedBy.**Publication**
abstract keywords
note title
year
● and ○ or
OK

attribute value input box

concept labeled button

Fig. 7. Semantic query form