"Web Portal": Complete ontology and portal

Mustafa Jarrar, Ben Majer, Robert Meersman, Peter Spyns
VUB STARLab, Pleinlaan 2 1050 Brussel

{Ben.Majer,Mjarrar,Robert.Meersman,Peter.Spyns}@vub.ac.be, www.starlab.vub.ac.be

Rudi Studer, York Sure, Raphael Volz
Institute AIFB, University of Karlsruhe, 76128 Karlsruhe, Germany

{studer,sure,volz}@aifb.uni-karlsruhe.de

Identifier	6.3
Class	**Deliverable**
Version	**1.0**
Version date	**30/05/2002**
Status	**final**
Distribution	**Public**
Responsible Partner	**AIFB, VUB**

OntoWeb Consortium

This document is part of a research project funded by the IST Programme of the Commission of the European Communities as project number IST-2000-29243.

Vrije Universiteit Amsterdam (VU)-Coordinator
Faculty of Sciences,
Division of Mathematics and Computer Science
De Boelelaan 1081a
1081 HV Amsterdam, the Netherlands
Fax and Answering machine: +31-(0)84-872 27 22
Mobil phone: +31-(0)6-51850619
Contactperson: Dieter Fensel
E-mail: dieter@cs.vu.nl

Revision Information

Revision date	Version	Changes
28.5.2002	0.1	AIFB Changes included
28.5.2002	0.2	VUB changes included

Abstract

This report describes deliverable 6.3: complete ontology and portal. The main goal of the portal is to provide the OntoWeb community members with an integrated access to all kinds of information related to the OntoWeb network. The current version of the OntoWeb portal supports both (conceptual) browsing and querying of the information stored in the portal, as well as content provision and access.
A syndicator will crawl the community members websites, annotated using the ontology, extract the annotation and store it in the knowledge base of the portal. The user is then able to browse and query the information stored in the portal. Querying of the information can be done using either ontology-based search forms or keywords. The system exploits the concepts and relationships from the ontology to allow the user to interact with this information on a conceptual level providing the user with a “mental map” of the available information.

Content Table

Content

Introduction

In this report, deliverable 6.3 (complete ontology and portal) is described. The web portal is a central part of the OntoWeb network where the technical possibilities of ontologies and the semantic web are put into practice. The main goal of the portal is to provide the OntoWeb community members with an integrated access to all kinds of information related to the OntoWeb network in general (members, researchers…) and the research aspects in particular (projects, events…). The current version of the OntoWeb portal offers both browsing and querying of the information as well as content provision and access, and is explained in more detail in the following sections. During the next phase of the project this portal will be further enhanced with extended functionalities such as advanced annotation tools.

Semantic portals

Ontology-based annotation of the available information is a prerequisite in order to offer retrieval of information on a semantical level. The backbone of the portal architecture consists of a knowledge base in which the ontology and the instances are stored and maintained. On top of this knowledge base, presentation and querying facilities are provided that take advantage of the ontology for the retrieval and presentation of information to the user

A semantic portal, through its usage of an ontology offers a higher quality of communication between publisher and consumer through the delivery of semantically processable information. The usage of well-defined semantics allows for the exchange of annotated information between different OntoWeb community members. Members can publish annotated information on the web, which is then crawled by the syndicator and stored in the portal knowledge base. Users of the portal can then access this information by taking advantage of the shared conceptualisation offered by the ontology. This shared conceptualisation can be seen as a “mental map” of the information contained in the portal and as such help the user with locating and processing this information.

Use of the ontology results in an improved query refinement compared with a keyword-based search offered by a conventional web portal thanks to the concept and relationship hierarchies. Also the semantic relationships are exploited to navigate through the mental model of the application domain. As it concerns a shared model the hypothesis is that

users will not get lost, and will be able to find more rapidly the desired information. By moving up and down the concept tree the user is able to generalise or specialise the type of instances that are presented. By moving down the hierarchy the precision of the instances retrieved is improved since those instances of the supertype that do not belong to the subtype currently selected are excluded from the result effectively narrowing down the scope of the query. Generalization on the other hand broadens the scope of the query, resulting in an improved recall when the concept hierarchy is exploited to expand the query to (recursively) include all subtypes when retrieving the instances of interest to the user.

Browse and query facility

Approach

The browse and query facility has been developed as a highly generic system which offers exploration of the available information at the conceptual level to the user of the portal. The reason to do so was that the ontology was still under construction at the time of development. Lack of a mature ontology meant that we could not target the ontology directly but instead had to rely on the ontology's meta-model. The main distinctions made in organising the presentation of the information to the user are therefore between the sub and super classes and the literal and non-literal properties of the different concepts.

Browsing

Browsing of the semantic portal can be separated into browsing of instance overviews and instance details. In the case of browsing instance overviews, the portal displays collections of instances according to the users selection. When viewing an instance details', the user is presented with the detailed information of a particular instance. Links to related instances are presented, grouped according to the community ontology.

Instance overviews

The hierarchical organisation of the different classes or concepts in the ontology is used to construct a dynamic tree allowing the user to browse the ontology and instances (see Figure1). The 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 to the user in the right pane.

Figure 1: Instance overview.

The user can decide to generalise or specialise the instance overview to a super or sub concept of the concept on display, restrict the number of instances of the concept presented even further using either a keyword or a form-based search or view the detailed information of a particular instance (see Figure 1).

Instance details

When viewing the detailed information for a particular instance, presented at a conceptual level, the information is organised according to the literal and non-literal properties of the classes in the ontology. While the literal properties or attributes of an instance provide the user with detailed information about an instance, the non-literal properties or relations with other instances, represented as links, enable the user to surf between related instances.
At the top of the instance details page the attributes of the selected instance are displayed. These attributes provide detailed information about the instance such as in the case of a person name, telephone number and email…

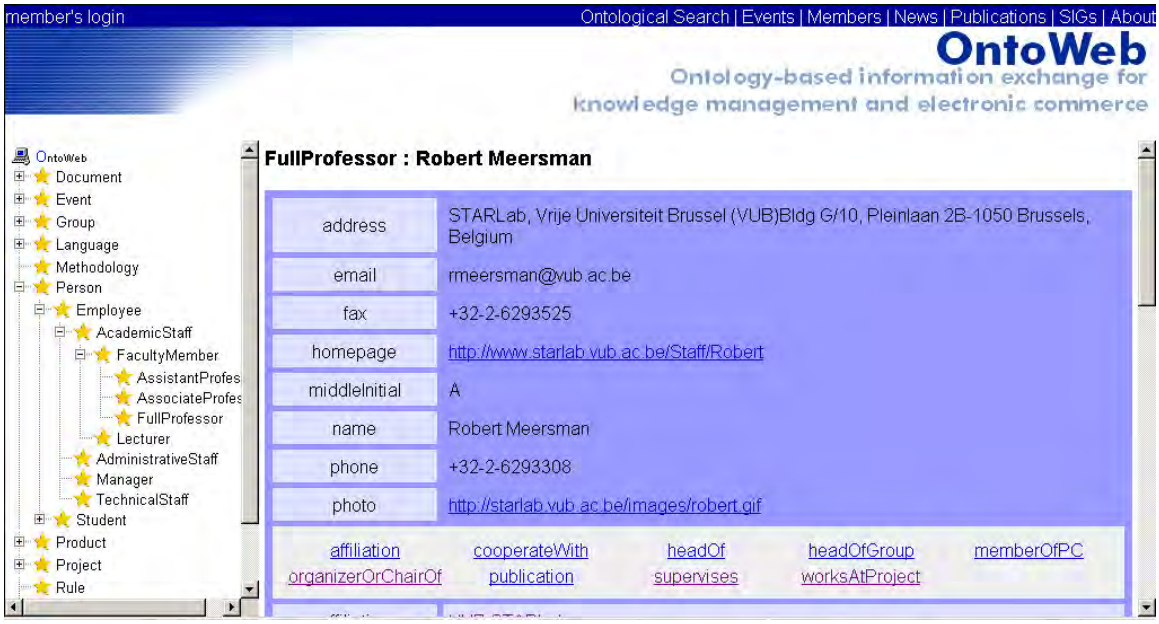

Figure 2: Instance details, literals.

Below the attributes, are displayed all the different conceptual relationships between the selected instance with other instances. These relationships are implemented as hyperlinks and point to other instances presented at the bottom of the page. The instances are grouped together according to the type of their relation with the selected instance (see figure 3).

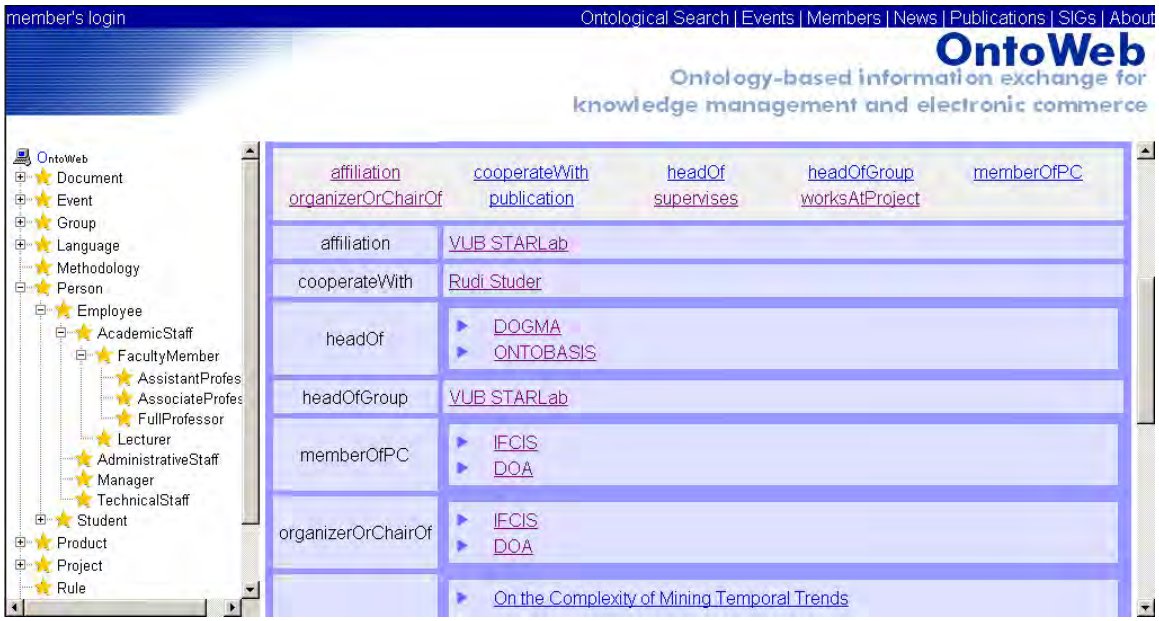

Figure 3: Instance details, resources.

Querying

By selecting a concept from the tree or through the use of the links to sub and super concepts in the instance overview page, the user expresses an interest in a particular type

of instances. The user can then further restrict the number of instances through the use of the search functions, both a keyword based and a form-based search is available.

Keyword-based search

The portal offers a keyword based global search. The instances retrieved are presented to the user with the concept type for each instance and a link giving access to the instance details page. Concepts from the ontology for which instances are included in the search result will be displayed when the key-word search results are sent to the user, allowing the user to narrow down the search results to only those concepts, which are of interest to her (see Figure 4). When the user enters multiple keywords, the engine will search for paths between instances containing the different keywords and if found present these paths to the user. When a search is executed from an instance overview page, the results are automatically narrowed down to those instances belonging to the concept the user is currently viewing.

Figure 4: Keyword based search, results.

Form-based search

The form-based search allows for the construction of query paths across the ontology. The user is presented with a search form containing textboxes in which property values of the instances to be retrieved can be specified. Also, the user is able to specify the instances with which the instances searched for are related. Buttons labeled with the concept type give access to other forms that can be used to specify related instances. For each node in the path the user can place additional restrictions on the property values allowed in the results.

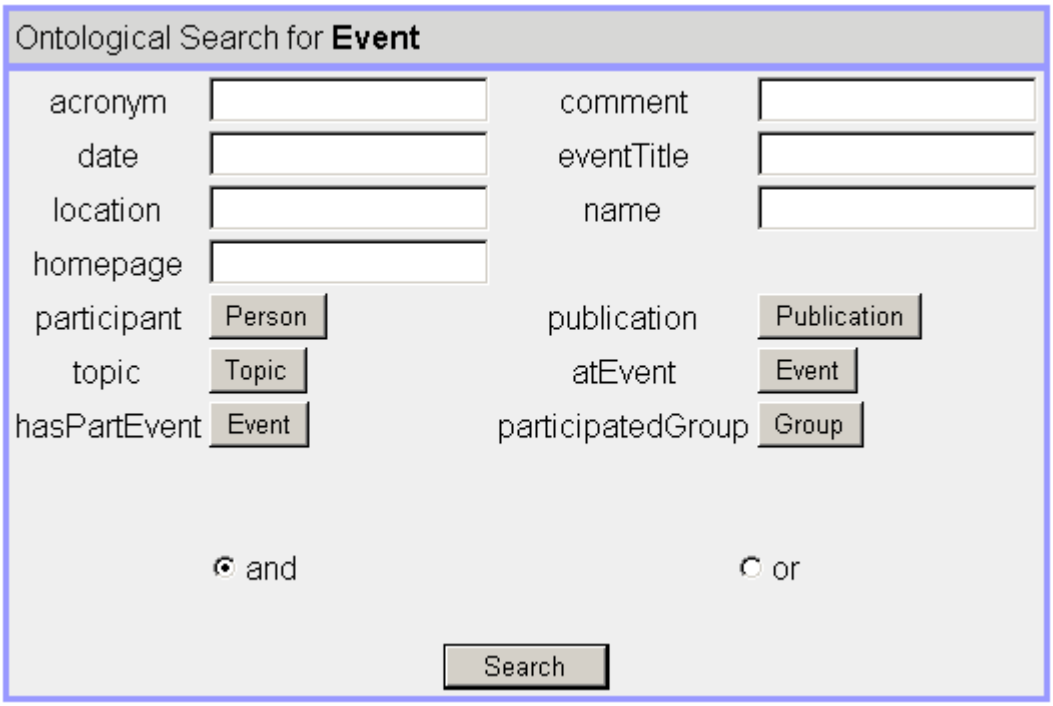

Figure 5: Query form.

Information provision

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. Syndicating information from participants is done by replicating their metadata. As depicted in

figure 6, both ontology and instances can be syndicated into the ontology base and the instance base in DOGMA server and can be accessed via the search and browse system.

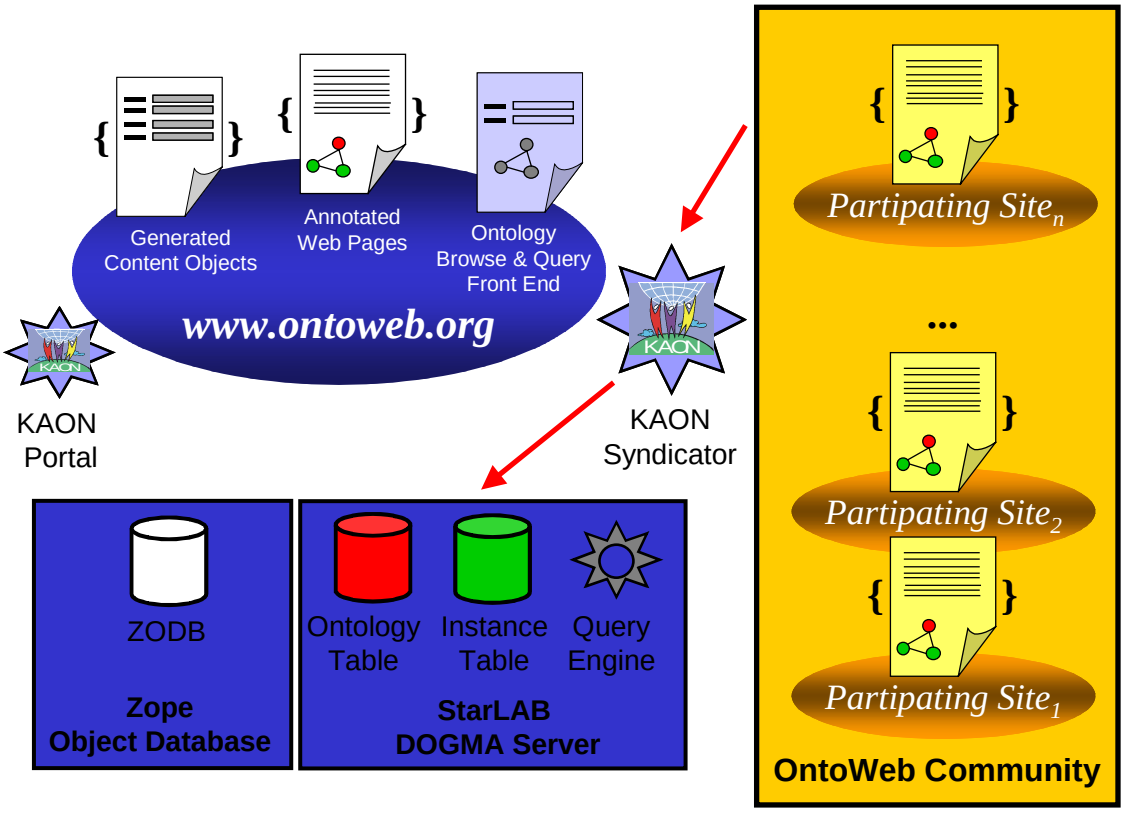

Figure 6: Content Syndication

Providing Content in the portal

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 system.

Additionally all content created within the portal is automatically associated with the predefined OntoWeb design to achieve a 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

-

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.

Publishing workflow

OntoWeb is an open community. Open communities pose 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 of private member data is required, which prevents foreseeable misuses. To support this requirement the system 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. 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. 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.

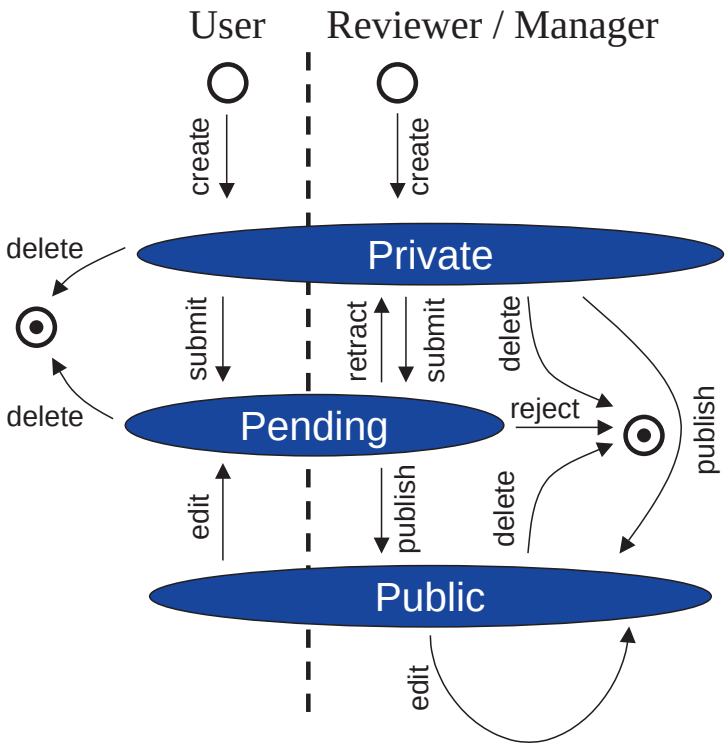

Figure 7: Publishing workflow

Figure 7 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[1] the object 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 do the transition to the published state (by applying the 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.

Information access

The browse and query facility is positioned on the meta-data level. Providing annotation (RDF) of the individual pages of the semantic portal itself is therefore not appropriate since these pages are a visualization of meta-information about (an)other web page(s). Instead the semantic portal provides a single file containing all the community meta-data stored in the portal, retrievable through a single url [/OntoWeb/Browse/AllInstances#].

[1] 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.

Additionally content provided by members in their personal spaces is also published as RDF according to the ontology. Only the metadata relevant to the particular piece of content is published.

```
+ <OntoWeb:FullProfessor rdf:ID="HSchmeck">
+ <OntoWeb:FullProfessor rdf:ID="Mertens">
- <OntoWeb:FullProfessor rdf:ID="RM">
    <OntoWeb:address>STARLab, Vrije Universiteit Brussel (VUB)Bldg G/10, Pleinlaan 2B-1050 Brussels,
      Belgium</OntoWeb:address>
    <OntoWeb:affiliation rdf:resource="#STARLab" />
    <OntoWeb:cooperateWith rdf:resource="#RStuder" />
    <OntoWeb:email>rmeersman@vub.ac.be</OntoWeb:email>
    <OntoWeb:fax>+32-2-6293525</OntoWeb:fax>
    <OntoWeb:headOf rdf:resource="#DOGMA" />
    <OntoWeb:headOf rdf:resource="#ONTOBASIS" />
    <OntoWeb:headOfGroup rdf:resource="#STARLab" />
    <OntoWeb:homepage>http://www.starlab.vub.ac.be/Staff/Robert</OntoWeb:homepage>
    <OntoWeb:memberOfPC rdf:resource="#IFCIS" />
    <OntoWeb:memberOfPC rdf:resource="#DOA" />
    <OntoWeb:middleInitial>A</OntoWeb:middleInitial>
    <OntoWeb:name>Robert Meersman</OntoWeb:name>
    <OntoWeb:organizerOrChairOf rdf:resource="#IFCIS" />
    <OntoWeb:organizerOrChairOf rdf:resource="#DOA" />
    <OntoWeb:phone>+32-2-6293308</OntoWeb:phone>
```

Figure 8: RDF instance data, extract.

Advanced services

As noted above, the availability of the mature ontology will allow for the development of more advanced services that take direct advantage of the ontology's semantics. Advanced services provide added value to the portal's information by processing the information into customized collections of instance data and offering these to interested agents. Examples of such advanced services would be a news service or a directory service. Through a news service, agents would be able to retrieve the latest news about the portal's community such as new members or upcoming events. This information can then be integrated into another website or a calendar application used by a community member. The directory service would allow for agents to easily lookup contact details for a particular company, research group or project.

Workshop

During the OntoWeb3 meeting in Sardinia, a workshop dedicated to the portal will be organised. During the workshop the participants will be familiarised with the semantic portal and explain the content provision efforts required from the community's members to enable the syndication process.
The workshop will also be used to establish an agreement among the OntoWeb community about the knowledge modeled in the ontology. A discussion will be organised about the contents of the ontology questions will be tackled such as which concepts and properties to

model and include in the ontology in order to satisfy the needs, requirements and views of the individual participants as much as possible.