D 1.4.3 Technology Road Show Report

Knud Möller (DERI, NUIG) & Ellen Schulten (UIBK)

with contributions from all participants of the
First International Semantic Web Technology Road Show

Abstract.
EU-IST Network of Excellence (NoE) IST-2004-507482 KWEB
Deliverable D1.4.3 (WP1.4)
Deliverable 1.4.3 reports on the First International Semantic Web Technology Road show. This show - a joint activity of ESWS and KWEB- took place on May 12th -13th, 2004 in Heracleion, Crete. In addition to describing the technologies that were demonstrated, the report presents crucial issues and findings that were identified in the interesting discussions that took place during and after the road show. The report finalizes with two recommendations. Recommendation 1 is the development of a template for the categorization of semantic applications. This template should allow for a swift and generic understanding of semantic tools and applications. Recommendation 2 is a call out to the KWEB partners to explore the possibility to jointly develop a web-based repository for semantic applications. These tentative recommendations could be further explored in Del 1.4.1, *The Technology Roadmap*.

Document Identifier:	KWEB/2004/D1.4.3/v0.5_Quality Controlled
Class Deliverable:	KWEB EU-IST-2004-507482
Version:	Final
Date:	Sept 08, 2004
State:	Ready for Dissemination
Distribution:	Public

Knowledge Web 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-2004-507482.

University of Innsbruck (UIBK) – Coordinator
Institute of Computer Science,
Technikerstrasse 13
A-6020 Innsbruck
Austria
Fax: +43(0)5125079872
Phone: +43(0)5125076485/88
Contact person: Dieter Fensel
E-mail address: dieter.fensel@uibk.ac.at

École Polythechnique Fédérale de Lausanne (EPFL)
Computer Science Department. Swiss Federal Institute of Technology
IN (Ecublens), CH-1015 Lausanne.
Switzerland
Fax: +41 21 6935225
Phone: +41 21 6932738
Contact person: Boi Faltings
E-mail address: boi.faltings@epfl.ch

France Telecom (FT)
4 Rue du Clos Courtel
35512 Cesson Sévigné
France. PO Box 91226
Fax: +33 2 99124098
Phone: +33 2 99124223
Contact person : Alain Leger
E-mail address: alain.leger@rd.francetelecom.com

Freie Universität Berlin (FU Berlin)
Takustrasse, 9
14195 Berlin
Germany
Fax: +49 30 83875220
Phone: +49 30 83875223
Contact person: Robert Tolksdorf
E-mail address: tolk@inf.fu-berlin.de

Free University of Bozen-Bolzano (FUB)
Piazza Domenicani 3
39100 Bolzano
Italy
Fax: +39 0471 315649
Phone: +39 0471 315642
Contact person: Enrico Franconi
E-mail address: franconi@inf.unibz.it

Institut National de Recherche en Informatique et en Automatique (INRIA)
ZIRST - 655 avenue de l'Europe - Montbonnot Saint Martin
38334 Saint-Ismier
France
Fax: +33 4 7661 5207
Phone: +33 4 7661 5366
Contact person: Jérôme Euzenat
E-mail address: Jerome.Euzenat@inrialpes.fr

Centre for Research and Technology Hellas / Informatics and Telematics Institute (ITI-CERTH)
1st km Thermi – Panorama road
57001 Thermi-Thessaloniki
Greece. Po Box 361
Fax: +30-2310-464164
Phone: +30-2310-464160
Contact person: Michael G. Strintzis
E-mail address: strintzi@iti.gr

Learning Lab Lower Saxony (L3S)
Expo Plaza 1
30539 Hannover
Germany
Fax: +49-511-7629779
Phone: +49-511-76219711
Contact person: Wolfgang Nejdl
E-mail address: nejdl@learninglab.de

National University of Ireland Galway (NUIG)
National University of Ireland. Science and Technology Building. University Road
Galway
Ireland
Fax: +353 91 526388
Phone: +353 87 6826940

The Open University (OU)
Knowledge Media Institute. The Open University
Milton Keynes, MK7 6AA
United Kingdom.
Fax: +44 1908 653169
Phone: +44 1908 653506
Contact person: Enrico Motta

Contact person: Christoph Bussler
E-mail address: chris.bussler@deri.ie

E-mail address: e.motta@open.ac.uk

Universidad Politécnica de Madrid (UPM)
Campus de Montegancedo sn
28660 Boadilla del Monte
Spain
Fax: +34-913524819
Phone: +34-913367439
Contact person: Asunción Gómez Pérez
E-mail address: asun@fi.upm.es

University of Karlsruhe (UKARL)
Institut für Angewandte Informatik und Formale Beschreibungsverfahren – AIFB. Universität Karlsruhe
D-76128 Karlsruhe
Germany
Fax: +49 721 6086580
Phone: +49 721 6083923
Contact person: Rudi Studer
E-mail address: studer@aifb.uni-karlsruhe.de

University of Liverpool (UniLiv)
Chadwick Building, Peach Street
L697ZF Liverpool
United Kingdom
Fax: +44(151)7943715
Phone: +44(151)7943667
Contact person: Michael Wooldridge
E-mail address: M.J.Wooldridge@csc.liv.ac.uk

University of Manchester (UoM)
Room 2.32. Kilburn Building, Department of Computer Science, University of Manchester, Oxford Road
Manchester, M13 9PL
United Kingdom
Fax: +44 161 2756204
Phone: +44 161 2756248
Contact person: Carole Goble
E-mail address: carole@cs.man.ac.uk

University of Sheffield (USFD)
Regent Court, 211 Portobello street
S14DP Sheffield
United Kingdom
Fax: +44 114 2221810
Phone: +44 114 2221891
Contact person: Hamish Cunningham
E-mail address: hamish@dcs.shef.ac.uk

University of Trento (UniTn)
Via Sommarive 14
38050 Trento
Italy
Fax: +39 0461 882093
Phone: +39 0461 881533
Contact person: Fausto Giunchiglia
E-mail address: fausto@dit.unitn.it

Vrije Universiteit Amsterdam (VUA)
De Boelelaan 1081a
1081HV. Amsterdam
The Netherlands
Fax: +31842214294
Phone: +31204447731
Contact person: Frank van Harmelen
E-mail address: Frank.van.Harmelen@cs.vu.nl

Vrije Universiteit Brussel (VUB)
Pleinlaan 2, Building G10
1050 Brussels
Belgium
Fax: +32 2 6293308
Phone: +32 2 6293308
Contact person: Robert Meersman
E-mail address: robert.meersman@vub.ac.be

Changes

Version	Date	Author	Changes
0.1	11-05-2004	Ellen Schulten	Start
0.2	24-05-2004	Knud Möller	Sections KIM, SWAN, Balto, Gate Museum F
0.3	04-06-2004	Ellen Schulten	Integration
0.4	04-07-2004	Knud Möller	Review & added info
0.4QC	04-09-2004	Ying Ding	Quality control; Comments
0.5	08-09-2004	Ellen Schulten	Incorporating changes from QC comment

Executive Summary

Hereafter KWEB

Hereafter WP1.4

In the Industry Pillar of Knowledge Web, Work Package 1.4 is dedicated to the promotion of ontology technologies. The mission of WP1.4 is to create awareness on how semantic web technologies could help organizations to deliver new products and services and how these technologies could create new business value. We hereby proudly present deliverable 1.4.3, the first contribution to this mission.

This deliverable reports on the First International Semantic Web Technology Road Show, a joint effort of ESWS and KWEB held on May 12-14, 2004 in Heracleion, Crete. The objective of the report is (1) to describe the innovative technologies that were demonstrated during this very first semantic technology road show and (2) to identify critical issues that need to be dealt with if these technologies would be implemented in a business environment.

In this report we make a distinction between semantic applications versus semantic tools — in other words, a distinction between semantic technologies for the end-user versus semantic technologies that support the development of semantic applications.

A Semantic Tool:
Smart help for a semantic expert

A Semantic Application:
Smart software for an end-user

Following the most significant observations of the road show:

- The progress in the development of semantic tools is currently shifting away from pure technological improvement and starts to seriously include improving end-user functionalities such as visualization and easy-to-use features.

- The progress made in the development of semantic applications is impressive both in quantity (a larger growth than tools) and in quality (the applications are often already implemented in a professional environment).

- Maintenance and support are—according to most developers—the major thresholds that prevent industries and organizations to deploy semantic technologies.

- Whereas the semantic research community is well informed about the semantic tools that are being developed, and of the direction the development is taking, the knowledge on the semantic applications is less disseminated.

Based on our observations, we argue in this report that maintenance and support are actually not the very first implementation issues to address. Industries and organizations lack first and foremost the sheer knowledge of the existence of semantic applications and their possibilities. When drafting the report, we experienced ourselves a related issue; the understanding and comparison of these very diverse applications is hindered because of the absence of a generic means to categorize and describe them.

2004: The Baby Boom of the Semantic Applications. Let's anticipate!

Hence, we put forward:

(1) A recommendation to develop a semantic application identification template which enumerates central features of the technology in a simple Yes/No list. The identification template should allow for a swift and generic understanding of the applications.
(2) a call out to KWEB partners to explore the possibility within KWEB for the development of a user driven web-based repository for semantic applications.

We propose that these tentative recommendations will be further explored in Del. 1.4.2; *The Technology Roadmap.*

Contents

Executive Summary 5

1. Introduction 8

2. Semantic Applications versus Semantic Tools 9

3. Semantic applications presented at the road show 10

3.1 Balto, a semantic annotation and clarification tool 11
3.2 Melvil- A search engine based on semantic nets 14
3.3 MuseumFinland 17
3.4 Promoottori, a Semantic Photograph Search and Browsing System 20
3.5 KIM Platform 22
3.6 SWAN 25

4. Semantic tools presented at the road show 28

4.1 The CO-ODE Project, an Environment for OWL based on Protégé and OilEd.. 28
4.2. ezOWL – Visual OWL Editor. 30
4.3 GATE 33
4.4 OWLCHESTRA: Facilitating the Development and Publishing of Small-Scale Web Ontologies 35

5. Concluding Remarks 37

6. Annex: A First Categorization of Semantic Applications and Tools 40

6. 1 A Semantic Application Identification Template 40
6. 2 A Semantic Tool Identification Template 42

1. Introduction

The First International Semantic Web Technology Road Show, held on May 12-14, 2004 in Heracleion, Crete, was in many respects a promising and exiting event. First and foremost, all demonstrated tools and applications were actually working. Not only that, they were also looking good! Contrary to previous years where the technological kernel of a tool was the major focus of a developer, user friendliness and easy-to-use features are issues that start to be addressed seriously. And in addition to the high quality of the tools and applications presented, the atmosphere was very inspiring and full of dialog. Every demonstrator and visitor was more than willing to share experiences, to discuss pros and cons and to think through possible future developments.

With the aid of written papers and the verbal explanations of the developers, we describe in chapter 2 and 3 the applications and tools that were presented. In chapter 4, we conclude the report with generic observations, findings and issues that arose at the discussions at the road show, at the KWEB meeting thereafter, and when reflecting back on the event.

We intend this report to be readable for a broad audience and we tried to provide a very simple description of each technology (next to mere technical features and expert information). In doing so, we encountered the following issue: there does not exist a generally accepted way to define and categorize semantic technologies. What is more, the range of the concept ´semantic tools and applications` is very broad, and the technologies that were presented were addressing an equally wide range of solutions and user groups. In chapter 6, we put forward two recommendations to facilitate the understanding of this broad range of semantic technologies. We do need to emphasize, that these recommendations, grew out of writing this report and do need further validation. Hence chapter 6 should be considered as an annex to support future research.

We kindly request all participants on feedback, comments and add-ons on the representation of their input and we would like to thank everybody for their contribution.

2. Semantic Applications versus Semantic Tools

When a research institute presents a semantic technology to an organization or a commercial entity, the first question will be: *'But what can it actually do for us?'* In this context, it is useful to make an explicit distinction between semantic applications and semantic tools. A semantic application is an end-user application based on semantic technologies, a semantic tool is a tool (usually some sort of software) for developers of semantic applications. The tool supports the development of semantic applications (and is, by the way, not necessarily a semantic application itself).

Semantic Technologies:
Distinguish between tools and applications

´I do not care about
the making of the vacuum cleaner.
Get me one!!'

Apart from a few border-cases, this distinction seems rather obvious and straightforward. At the same time, it is often not explicitly utilized. We suggest it should. For novices, it is confusing to discuss for example, an ontology editor next to a smart desktop organizer. Especially at this point in time, it is useful to establish a clear separation between applications and tools; whereas the first generation of semantic technologies focused on the development of tools for building the foundations for semantic applications, currently more and more actual applications are being developed.

Another reason for an explicit separation between applications and tools is that both possess different key features for their actual use. For example, whereas user friendliness is crucial for the adoption of an application by an end-user, it is ´just` an added value for a developer. For the latter, the actual functionality of the technology is key. Hence, the understanding of semantic technologies by novices will be facilitated by making the distinction between tools and applications. During the writing of the report, we developed two basic templates where we captured the main features of each category (see attachments). The templates could be utilized in future work on the dissemination of knowledge on semantic tools and applications. This issue will be addressed in more detail in chapter 6.

3. Semantic applications presented at the road show

As argued in the previous section, we make in this report an explicit distinction between semantic applications on the one side and semantic tools one the other side. A semantic application is defined as an end-user application based on semantic technologies whereas a semantic tool is a tool to enhance the development of semantic applications. For the promotion of semantic technologies to the industrial community, semantic applications obviously possess an enormous potential strength; an application can actually show in an applied area how semantic technologies can contribute to the amelioration of our day-to-day life.

During the Road show, 6 of the 9 semantic technologies showed, fell in the category of semantic applications, much more than the writers of this report had expected based on their experience of previous semantic technology demonstrations. Furthermore, it became clear that the demonstrators of the applications had other applications to show, but that they had confined themselves for this particular show to one choice. Most surprising, however, was that nearly all of the semantic applications were already functioning in real-life conditions and in a professional situation.

Underneath is following a portrayal of each semantic application demonstrated. In order to facilitate, we address for each application:

1. an intro;
2. a motivation;
3. the underlying principles and distinguishing features**;**
4. a discussion and a reviewing of the critical implementation factors

For the reader without a technological back ground and with an interest in the industrial use of the applications, especially section 3 and 4 might be of interest. In section 3, possible and interesting areas of application come into sight. In section 4, we review the viability of implementing the application in a professional context.

3.1 Balto, a semantic annotation and clarification tool

Gerard Bleeker, Pat Croke, Fionnuala Foy, Kieran Hannon, Cliodhna Hurst, Ann Johnston, Aine Leddy, Sean O'Riain
Hewlett Packard /DERI Líon - Digital Enterprise Research Institute European Software Centre, Galway
{gerard.bleeker, pat.croke, kieran.hannon, cliodhna.hurst, ann.johnston, sean.oriain}@hp.com,{fionnuala.foy, aine.leddy}@deri.ie

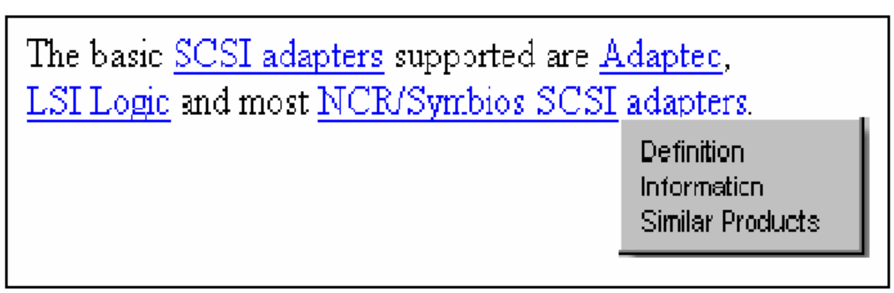
3.1.1 Introduction

Balto is a system to enhance the browsing of product description pages in HP's On-Line Services (OLS).

3.1.2 Motivation

The technical and business knowledge of the customers of HP is of a wide variety. This poses a problem for the description of the products that are online for sale; which level of knowledge should HP assume? A detailed technical description will be useless for a business manager, a focus on the business processes will not satisfy the technical manager, and providing all information will probably not be clear for anyone.

The product description pages of HP On-Line Service currently contain a large amount of technical terms, and Balto's aim is to aid users with varying technical backgrounds to understand and make greatest use of these pages. To achieve this, the system automatically highlights product names and key technical and business terms and provides them with pull-down menus offering additional information about the product or term through a web service. The following picture shows an excerpt from a product page, enriched with highlights. By clicking on the term "NCR/Symbios SCSI adapters", the user is provided with a pull-down menu:



An extract from a product description page enhanced by Balto

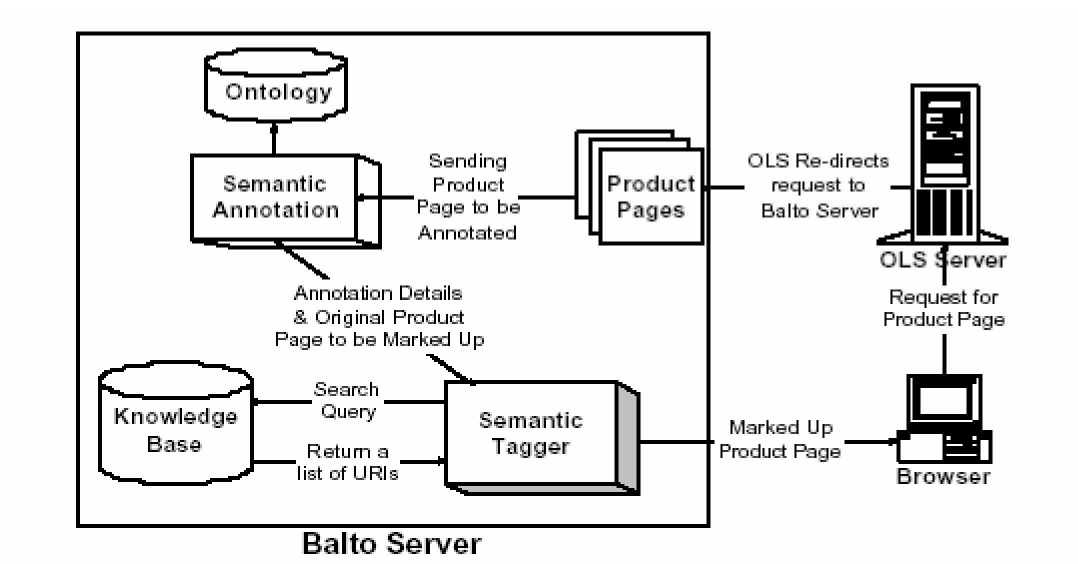
3.1.3 Principles and distinguishing features

The annotation process is performed with the aid of the GATE platform (see the section in this report on GATE). In particular, Balto uses a combination of gazetteer lists with product names and technical terms and a set of pattern matching rules using GATE's JAPE engine.

To further aid the semantic annotation process, Balto also defines an ontology for the product types and concepts relevant for the OLS, as well as the relationships between these concepts. The ontology covers a wide variety of concepts, ranging from product system requirement such as ‘processors’, ‘hard disk’, to more specific product information such as ‘spreadsheets or ‘text editors’.

A corresponding knowledge base contains instances of the ontology's concepts accompanied by the web services available for the product (e.g. further information, similar products, etc.). For instance, the concept ‘SCSI adapters’ may have web services directing the user to web sites explaining ‘SCSI adapters’ in greater detail. As a result, an instance ‘SCSI adapters’ is stored in the knowledge base along with the URIs to its web services. Therefore this knowledge base includes very general information relating to all products, along with very specific information that relates to particular products.

The general architecture of Balto can be seen as follows:



The semantic annotation is performed 'on the fly' by semantic annotator based on GATE technology. The annotated pages then are further tagged by semantic tagger in order to provide the drop-down functionality to the user on their browser listing a set of URIs. The tagging approach used here exists a little difference comparing wit AKT's Magpie application, where Magpie provides the drop-down functionality on client side, while Balto creates directly on the server side. The benefit of doing so can save one step of sending request of a redirection to a particular web services from client to server. When all semantically annotated entities are tagged, the product page is sent to the user for browsing. The user can select a relevant term from the text, choose a web service from the resulting drop-down menu and view extra information about the selected term.

Another semantic feature of the system is that annotated information is stored in RDF triples. The subject of the triple is the term on the product page. The predicate of the triple is the concept in the ontology which the subject is related to. The object of the triple is the list of URIs in the knowledge base. Therefore annotation stored in RDF triple makes the product page further processable and understandable to machines.

3.1.4 Discussion and critical implementation factors

The project is targeted at a very interesting domain – product purchasing, where lots of problems can be smoothed by semantic web technologies. This project clearly shows the value of the semantic web technologies, per se ontology, to this specific domain problem.

This project is based on one predefined ontology, therefore it does not touch the ontology mediation and ontology versioning problem. We can see that it is a pretty naïve solution which might work very well for one specific need. In the long run, if HP would like to provide such services for other companies which have their own defined ontologies, mediation of the ontologies will be the main challenge to face.

Versioning of the ontology is the problem difficult to escape since information is always changing. How to handle the annotation based on different versions of the ontologies, how to provide backward consistency of different versions of the ontologies and their corresponding instances will be another challenge to face.

The advantage of this approach is that very good results can be made within a comparably short period of time.

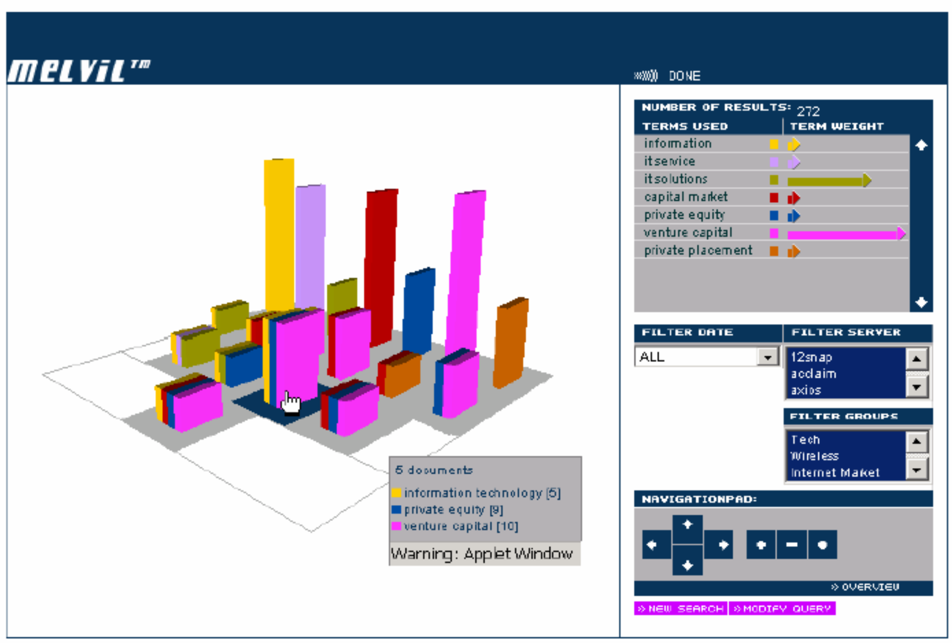
3.2 Melvil- A search engine based on semantic nets

Daniel Dögl,
uma information technology AG, Austria.
daniel.doegl@uma.at

3.2.1 Introduction

Melvil™ is a web-based search engine, specially adapted for specific knowledge areas and extended by a powerful 3D visualization. Its search technology is based on semantic networks, that are saved in XML topic map format (.xtm).

A screenshot illustrating Melvil:



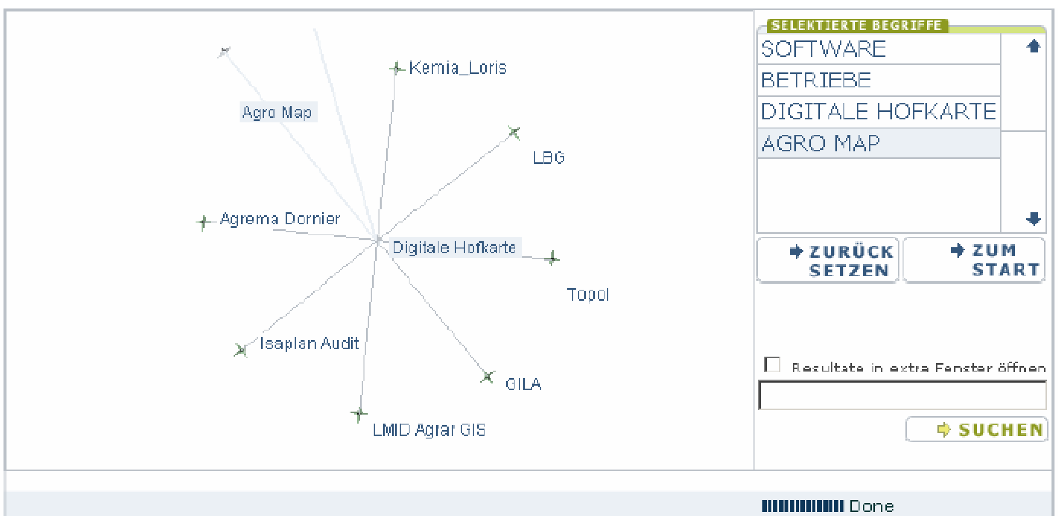
3D Result view with filter and navigation options

3.2.2 Principles and distinguishing features

For every topic area to be searched a semantic network is created. This is comprised of a hierarchical organization of terms whose relationships are defined. Each term, which is woven into this "knowledge world", is enriched with an extra search pattern. The search sample contains further spelling of the same world, synonyms or the same term in other languages. This collection of terms is called a "concept".

When looking for a concept all the defined spelling variations, synonyms, grammatical flexions or translations into other languages are automatically included. Therefore, it is possible to find the results in various languages, utilizing concepts that are targeted for multiple languages by entering a search query in a language known to the system.

Melvil™ also combines the advantages of concept-based searches and full text searches: A search query can be divided into concept and full text search automatically due to Melvil™'s Concept Recognition feature: When looking for a specific term, the Melvil™ Concept Recognition checks whether the search term matches a concept in the associated semantic net. The actual search query then consists of the concepts that have been identified as well as any parts of the query itself – defined as full text search queries. This way a single query can return results in different languages. In addition the semantic network can be visualized in 3D. This way the user can easily become acquainted with the contents of a specific knowledge area, before the actual search begins.



Formulation of a query using 3D Visualization and an additional text field with the Melvil™ concept matching function.

Users can click on concepts – depicted as knots – and navigate through the sub-concepts that are then shown, playfully learning the topics that can be searched for. Further, concepts can be linked to documents in order to enable the user to enter the knowledge base directly. The user can formulate his query either be simply entering a query in a text field but also by selecting concept representations in the 3D visualization. All the functions described here can be used individually or combined as required. For query formulation operators like wildcards or phrasal search are supported as well.

Before finally sending the query, the user can make use of the Melvil™ Prefiltering feature and exclude certain documents or servers. The search can be restricted to documents that fulfill certain qualities or carry specific meta data: e.g. the results should not be older than a defined date.

After having assembled the final query and the Prefiltering settings, the user can choose between two ways of result representation. Depending on the goal of the application, the results can be simply listed in a web page or presented in a well structured 3D visualization.

The 3D result visualization (Melvil™ Data Analysis) allows the user to reduce the found documents to the most relevant for his needs using meta-data filtering tools. Meta-data filter tools like age, length or source server are offered. In addition the results can be restricted to those that contain a certain choice of the search terms. The terms, which were searched on, can also be prioritized in order to rearrange the results.

Another crucial feature of Melvil™ is the ability to search database contents. To enable this feature, the data must be made accessible over http in a XML format specified by uma. First a list with links to the records is scanned. After that, all records are scanned and indexed and made available to the user. If data matches the search query, the user is offered a link to a HTML page that contains this data.

3.2.3 Discussion and critical implementation factors

Melvil™ is actually implemented. Two real life applications of Melvil are:

1. As a search solution for www.bildung.at
   A Melvil™ search for the education portal of the elearning portal of the bm:bwk (federal ministry of education) combining 60 web services on the topic of e-learning in Austria.

2. LWK – An Information System on the Topic “Precision Farming”
   The information system has the objective to make the wealth of information about “Precision farming” on the internet easily available to anyone.
   http://www.agrarelektronik.at

Melvil™ is using Topic Map as the semantic representation languages, which is quite natural for digital library related application. Based on the available information, it is difficult to see how the ontology looks like, how to define concepts and how to define the relations between concepts. Hence further investigation is needed to conclude whether Melvil™ is handling multi-ontologies, ontology mediation and ontology versioning.

3.3 MuseumFinland

Eero Hyvönen, Samppa Saarela, Kim Viljanen, and Avril Styrman
University of Helsinki, Helsinki Institute for Information Technology (HIIT)
P.O. Box 26, 00014 UNIV. OF HELSINKI, Finland
firstname.lastname@cs.helsinki.fi
http://www.cs.helsinki.fi/group/seco/

3.3.1 Introduction

MuseumFinland (MF) is a web portal based on Semantic Web technologies and developed by the Helsinki Institute for Information Technology (HIIT) at the University of Helsinki. It is an excellent example for a working, real-world Semantic Web application. Through this portal - unfortunately only available in Finnish at the moment - a user gains access to a large exhibition of cultural artifacts such as textiles, pieces of furniture, tools, etc. The exhibition combines collections from three Finnish Museums (the National Museum, the Espoo City Museum and the Lahti City Museum) and consists of a total of ca. 5000 exhibits.

3.3.2 Motiviation

A special characteristic of museum collections is their semantic richness in content and interconnected nature. Collection objects have a history and are related in many ways to our environment, to the society, and to other collection objects. This semantic network is not limited to a single collection but spans over other related collections in other museums.

For the end-user, MuseumFinland creates a seamless repository of web pages with two major services:

- A semantic view-based search engine that is based on the underlying ontology and knowledge base instead of keywords.

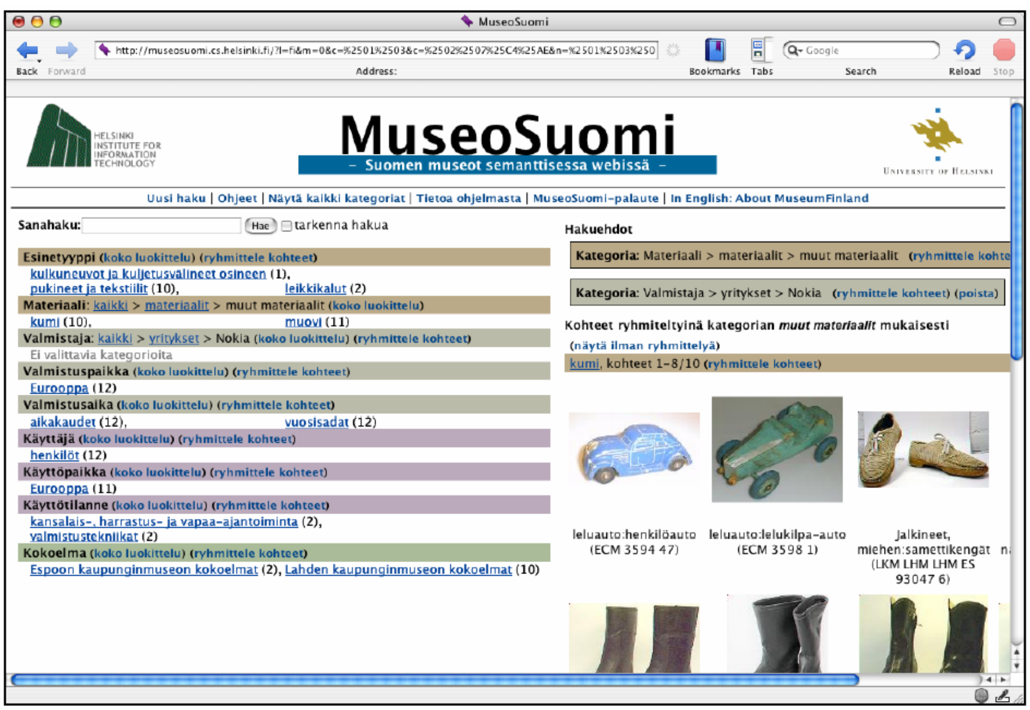
- A semantic recommendation system by which the user can find out explicit and implicit semantic associations within the global collection data, and use the associations for browsing the collections.

For the participating museums, the portal provides a channel to publish content easily together on the Semantic Web.

3.3.3 Principles and distinguishing features

The MuseumFinland demonstration contains an exhibition of some 5000 cultural artifacts, such as textiles, pieces of furniture, tools etc. Collection objects are linked with each other through 7 ontologies containing some 4500 classes and 2500 individuals. The contents come from the collections of three National Museums. These museums are situated in different cities, use three different relational database schemas, data base systems, and collection management systems.

The different museum collections, even though originating from physically distant and structurally different sources, are integrated using a number of ontologies (expressed in RDF(S)) and a corresponding knowledge base. This, combined with a logic-based query mechanism and a graphical user interface, allows the user to browse and explore the collections in a very intuitive and efficient way. Although similar at first in its visual appearance to web catalogs like Yahoo, MF's functionality goes beyond this kind of approach by offering multi-faceted, ontology-based search. Exhibits can be searched by categories like their geographical, chronological or cultural origin, type, similarity, relation to other exhibits, or a combination of any of the available categories.



The Web Interface of MuseoSuomi (MuseumFinland)[13]

3.3.4 Discussion and critical implementation factors

MuseumFinland as a semantic portal is able to publish heterogeneous museum collections on the semantic web. The RDF(S) based semantic web technologies enable the knowledge-based web exhibitions and collection interoperability in content. This portal is able to provide semantic browsing facilities that reveal to the end-user a most interesting aspect of the collection items: the explicit and implicit semantic relations that relate collection data with each other. The semantic recommendation system is implemented as a logic server based on SWI-Prolog and its HTTP server and RDF modules. There also exists a prototype implementation of MuseumFinland in Nokia Series 60 mobile handies.

This application handles more than one ontology, but the mediation of the ontologies is done manually. Versioning of the ontologies is not considered here. Interesting points for this portal are the semantic browsing and semantic recommendation system.

Museum Finland is successfully in use. There is not yet information available on end-user satisfaction.

3.4 Promoottori, a Semantic Photograph Search and Browsing System

Eero Hyvönen, Samppa Saarela, Kim Viljanen, and Avril Styrman
University of Helsinki, Helsinki Institute for Information Technology (HIIT)
P.O. Box 26, 00014 UNIV. OF HELSINKI, Finland
firstname.lastname@cs.helsinki.fi
http://www.cs.helsinki.fi/group/seco/

3.4.1 Introduction

Promoottori is used for retrieving photographs from an image repository at the Helsinki University Museum (http://www.helsinki.fi/museo/). Promoottori is a deployed kiosk application system that is currently in daily use by the visitors of the museum.

3.4.2 Motivation

The Helsinki Institute for Information Technology has implemented a semantic search engine and a browser for RDF(S) (http://www.w3.org/RDF) repositories. Its main novelty lays in the idea of combining benefits ontology-based and multi-facet search methods. Furthermore, the search system is enhanced with a knowledge based recommendation system that facilitates semantic browsing.

To test and validate the approach, this system has been applied by a creating the real-life application called Promoottori.

3.4.3 Principles and distinguishing features

Promoottori provides the end-user with two major services. Firstly, the user can find images of interest by using a **semantic multi-facet search engine**. In multi-facet search, the repository contents are annotated in terms of orthogonal classification hierarchies, such as “Persons”, “Locations”, “Time”, etc. In Promoottori, the facets are taxonomies projected from a set of six top level categories of an RDF(S) ontology. It describes the domain concepts of the photograph repository and their properties. The domain is the academic promotion ceremonies of the University of Helsinki. A query is formulated by selecting categories from the exposed facet hierarchies. This is convenient to the users because they are usually unfamiliar with the vocabulary of the domain.

After search the user can select a photograph from a hit list for a closer look. A knowledge-based **semantic recommendation system** is then available for browsing. This system exposes to the user explicit and implicit semantic associations between the photographs as hyperlinks. The labels and titles of the links explain what kind of semantic relation is in question. The linkage is created by defining a set of predicate logic rules. Promoottori's search engine is written in Java and makes use of the Jena system of the HP Labs. The semantic recommendation system is based on SWI-Prolog (http://www.swi-prolog.org) and its RDF parser.

3.4.4 Discussion and critical implementation factors

MuseumFinland and Promoottori are based on the same key technologies: semantic multi-facet search engine and semantic recommendation system. In Promoottori, there exists one ontology in RDF(S) with a set of six top level categories which serve as the facet for classification and browsing. Therefore there does not exist the ontology mediation problem. Versioning of the ontology is not considered here.

Promoottori is successfully in use. There is not yet information available on end-user satisfaction.

3.5 KIM Platform

Atanas Kiryakov, Borislav Popov, Angel Kirilov, Dimitar Manov, Damyan Ognyanoff, Miroslav Goranov
Ontotext Lab, Sirma AI EAD, 135 Tsarigradsko Shose, Sofia 1784, Bulgaria
{borislav, naso, angel, mitac, damyan, miro}@sirma.bg
Tel: +359 2 9768 303; Fax: +359 2 9768 311

3.5.1 Introduction

The KIM (Knowledge and Information Management) platform offers generation of Semantic Web data through ontology-aware annotation of natural language text, as well as indexing and retrieval of the generated data. This allows, among other things, for hyper-linking and improved visualization and navigation.

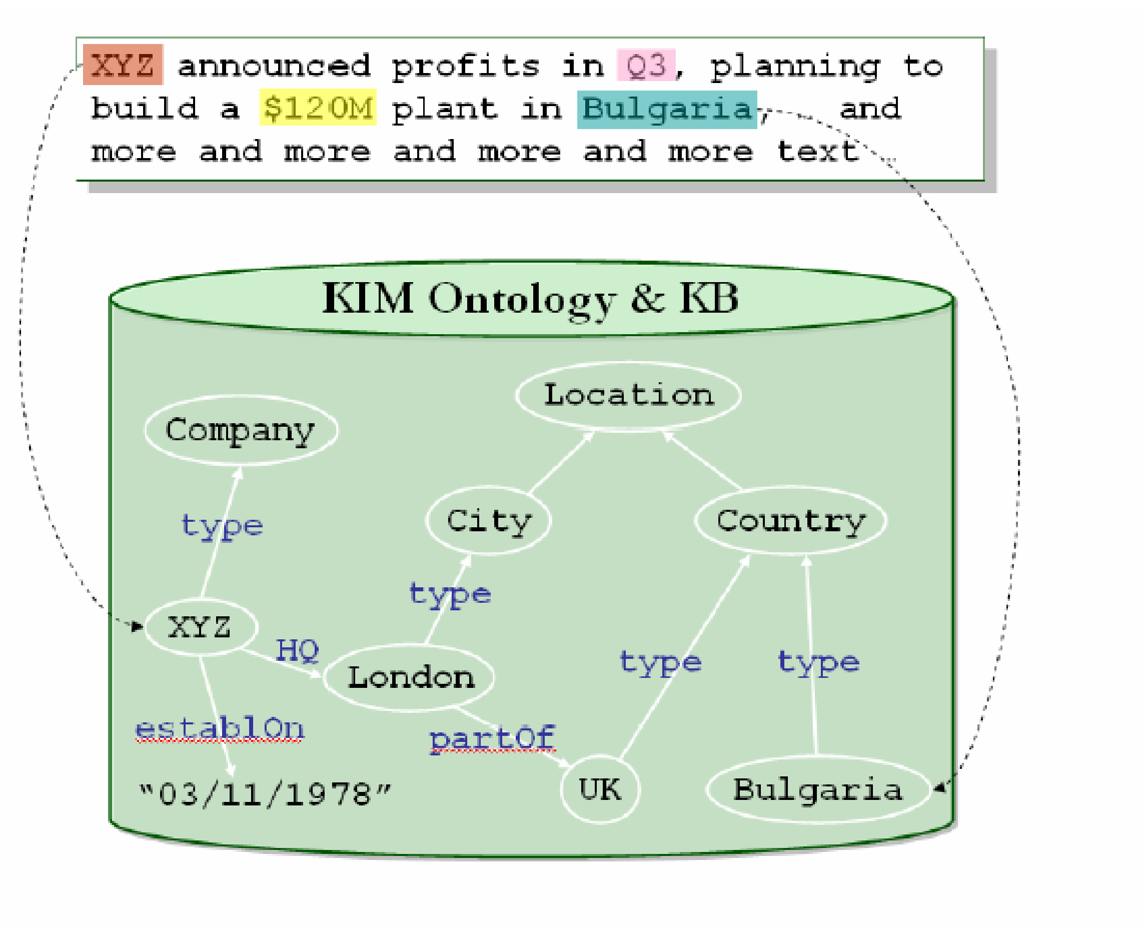
3.5.2 Principles and distinguishing features

The KIM platform provides infrastructure and services for automatic semantic annotation, indexing, and retrieval of documents. It also provides infrastructure for scaleable and customizable ontology-based information extraction (IE) as well as annotation and document management, based on GATE. In order to provide basic level of performance and allow easy bootstrapping of applications, KIM is equipped with an upper-level ontology and a massive knowledge base providing extensive coverage of entities of general importance. The later are accessed and managed using high-performance semantic repository (a tuning of Sesame handling about 4M statement). Semantically-enhance information retrieval is provided on the basis of the Lucene.

The KIM platform provides services and infrastructure for semantic annotation, indexing, and retrieval. To do this in a consistent fashion, it performs information extraction based on an ontology and a massive knowledge base.

The traditional flat named entity (NE) type sets consist of several general types (such as Organization, Person, Date, Location, Percent, Money). Although these represent the most important domain-independent NE types, still an extension is feasible. We identified an inter-domain NE type hierarchy from a corpus of general news and integrated it within the KIM Ontology (KIMO). It contains definitions of about 250 entity classes and 100 attributes and relations. The semantic descriptions of entities and relations between them are kept in a knowledge base (KB) encoded in the KIMO. The semantic annotations generated by KIM provide for each entity reference in the text (i) a link (URI) to the most specific class in the ontology and (ii) a link to the specific instance in the KB. The semantic repository is initially pre-populated with the KIM World KB, consisting of about 200 000 descriptions of entities of general importance compiled from a number of “trusted” sources. The KB is continuously enriched with new entities and relations found during the annotation process.

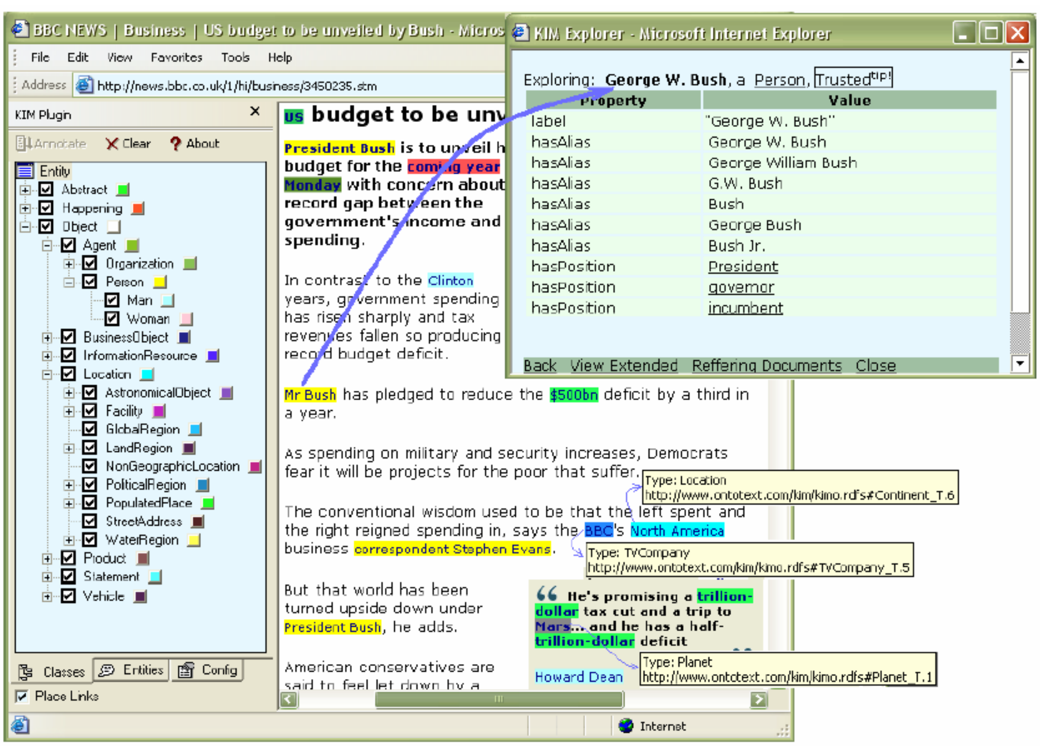
KIM provides a light-weight top-level ontology (KIMO), which consists of basic concepts such as *Company*, *Person*, *Location* or *Event*, as well as corresponding super- and sub-concepts. Currently, KIMO contains roughly 250 concepts and about 100 axioms and relations connecting these concepts. Building on this ontology is the KIM World Knowledge Base (KIM-KB), which is pre-populated with about 200.000 trusted instances. These instances represent entities such as actual companies, persons, cities, countries, etc., and have been extracted semi-automatically from a large corpus of general news texts. Through its indexing component, KIM allows to search for documents with respect to the ontology and knowledge base. It is e.g. possible to perform a query such as "Show me the documents mentioning the CEOs of companies in Norway". When new documents are processed by KIM's annotation component (in a process called "Semantic Annotation"), the entities found in these documents are either linked to existing instances in the knowledge base, or added as new instances (which are considered to be untrusted until validated). The following figure illustrates this:



KIM integrates a number of existing and well tested open source components in its architecture. GATE is used to perform automatic extraction of formal data from natural language texts (see the section on GATE). In particular, the ready-made NER components included in GATE are used. However, these have been extended to facilitate annotation with respect to KIM's ontology. For storage as well as query of the ontology and knowledge base, the Sesame RDF(S) repository is used, while an adapted version of the Lucene engine is responsible for indexing and key word based query.

3.5.3 Discussion and critical implementation factors

Currently, the KIM platform is accessible for testing through a web based UI as well as a plugin[6] for Microsoft's Internet Explorer (see picture below). While the former only operates on the existing corpus of documents, the latter can be used to annotate any web page with respect to KIMO and KIM-KB. However, since all of IM's functionality is also accessible through an Java API, any kind of custom application could be built on top of KIM (see e.g. the SWAN project in the following section).



The KIM IE-plugin at work

In addition, KIM is capable of handling multiple ontologies, ontology mediation and ontology versioning with limited automated support.

3.6 SWAN

DERI Ireland
Sheffield NLP Group
Ontotext Lab.

3.6.1 Introduction

The idea behind the Semantic Web Annotator Project (SWAN) is to integrate a number of well tested systems in a large scale, distributed environment and offer customizable annotation, information extraction and reasoning services. These resources could either be offered to third party providers for integration in their own web-based products or directly to end-users.

The SWAN project is similar to the above described KIM project in the respect that one of the aims is to provide annotation of existing web content to extract semantic web data. In fact, SWAN will incorporate KIM as one of its components. However, automated metadata extraction in SWAN will be scaled up significantly to allow industrial strength Semantic Web applications development.

Possible scenarios are:

- Quantitative media reporting. Scenario: "The media distance between your company and the subject of XML is 0.09; for IBM the value is 0.2".

- Custom knowledge services. Scenario: client specifies information need overt he web ("Throughout the next year, I'll be interested in performances by XYZ or a similar theater group in London."), provider develops custom IE and query front-end.

SWAN will also offer Knowledge Services - web services for content annotation and indexing, as well as for ontology, knowledge, and metadata access and management. These services will be offered free of charge for evaluation and research purposes. In this way, SWAN will become one of the central places for the Semantic Web community, offering a testbed and access to huge volumes of metadata, consistent ontology and further world knowledge which could be used directly or for reference purposes.

3.6.2 Principles and distinguishing features

Version 1 of SWAN will be based on a combination of three existing systems:
1. GATE, a General Architecture for Text Engineering (University of Sheffield)
2. KIM, a Knowledge and Information Management product (OntoText)

3. SECO, a focussed crawler and integration site for Semantic Web Metadata (DERI Semantic Web cluster)

GATE, a General Architecture for Text Engineering, is a development environment and middleware framework for creating, adapting and deploying HLT components, plus a collection of components for various HLT tasks. GATE is in use across the research spectrum from blue sky experimentation through technology transfer and into productisation, with users from large corporates and SMEs to academic research. The IE tools in GATE have been proven in a wide range of contexts and have participated in all the major quantitative evaluation competitions since 1995 (including MUC, TREC/QA, ACE, DUC, and TIDES/Surprise Language), so we can now estimate quite precisely the development resource required to do a particular task, and the accuracy that can be expected as a result. We have built or are in process of developing IE systems in languages including: Arabic, Bengali, Bulgarian, Chinese, English, French, German, Greek, Russian, Spanish and Swedish. The system is increasingly in use for automated adaptation of HLT using machine learning, and we have successful experiments with Waikato's WEKA system and with an HMM component from OntoText. Related work in Sheffield has been widely published in relation to the S-CREAM and MnM assisted annotation systems.

KIM, a platform for Knowledge and Information Management*,* adds entity-based indexing, ontology transducer callouts from the pattern language, and disambiguation of entities in the context of an upper-level ontology.
KIM is a software platform for:

- o Semantic annotation of text: automatic ontology population and open-domain dynamic semantic annotation of unstructured and semi-structured content for Semantic Web and KM applications.
- o Indexing and retrieval (an IE-enhanced search technology)
- o Query and exploration of formal knowledge

KIM includes:

- o KIM Ontology (KIMO) and KIM World KB
- o KIM Server with API for remote access and integration
- o Front-ends: KIM Web UI and Plug-in for Internet Explorer

KIM currently makes use of the SESAME RDFS repository.

The SECO system collects RDF files and integrates RDF data sources. The system comprises an RDF crawler to gather data, a integration component to translate the original data into a mediated schema, and an HTML user interface to browse the integrated data set. The key challenge is to process data that has been created in a decentralized manner, leading to data which is sometimes incomplete or incorrect. The scutter component collects RDF files that are dispersed across several thousand web sites by following links inside these RDF files. A considerable amount of RDF instance data has been made available in RSS (RDF Site Summary) describing news items and FOAF (Friend of a Friend) describing people, but has not been utitlized to the extent possible.

The integration component accesses the collected data set via a remote query interface over HTTP. The data is transformed into a mediated schema to facilitate displaying the collected data. We use inference rules to carry out query expansion and schema mapping. The user interface in HTML is constructed by transforming the data set in the mediated schema to XML, and from XML to XHTML which is finally displayed by any standards-compliant web browser. SECO is implemented in Java and based on HP's Jena2 toolkit, and is available under an open source license.

3.6.3 Discussion and critical implementation factors

From a technical point of view, the version of KIM used in SWAN will be based on a server cluster. The different components (annotators, semantic repository, document queue, web server, etc.) are spread over a number of servers. This architecture provides much better scalability and extensibility.

Currently, the main kinds of RDF data covered by Seco are FOAF (Friend of a Friend, an ontology for formally describing Persons) and RSS 1.0 (RDF Site Summary). In the context of SWAN, Seco will provide the means to integrate existing semantic web data with new, automatically created data originating from KIM. In one possible scenario, new Person instances for the KIM Knowledge Base could be derived from the FOAF descriptions collected by Seco. Another possibility is to have KIM annotate the RSS items in Seco's data repository and add the found entities to KIM's knowledge base.

SWAN is the project just running recently. There is no prototype available during the reporting time.

4. Semantic tools presented at the road show

4.1 *The CO-ODE Project, an Environment for OWL based on Protégé and OilEd.*

Nick Drummond, Matthew Horridge, Holger Knublauch, Mark Musen, Alan Rector, Hai Wang
Medical Informatics Group, University of Manchester, Manchester M13 9PL, UK
{ndrummond, mhorridge, rector, hwangg}@cs.man.ac.uk
Stanford Medical Informatics, Stanford University, Stanford, CA 94305-5479
{holger, museng}@smi.stanford.edu

4.1.1 Introduction

The Co-ode project presents the first version of a new editor environment for building OWL ontologies. The project integrates two of the most widely used ontology editing tools – OilEd (from University of Manchester) and Protégé (from Stanford University). The aim is to provide a user-oriented, scalable, extendable and open source ontology development environment for diverse groups of users without some pre-existing, human-created agreements outside the Web.

4.1.2 Motivation and goal

Ontologies are a pivotal concept for realizing the vision of the ”Semantic Web”. They are becoming increasingly important at all levels. Recently the new Web Ontology Language OWL has become a standard for representing ontologies, but tools for OWL are still limited. Furthermore, ontologies come in many different forms from small local ontologies for specific applications to very large ontologies serving whole communities. Ontology developers vary greatly in their experience and sophistication. Many of the most successful ontologies are built by subject matter experts with little knowledge of formal ontologies.

The CO-ODE project (*http://www.co-ode.org*) assumes that different user interfaces will be required for different users and types of ontologies. It uses Protégé plug-and-play architecture to make it easy to create multiple interfaces based on a single underlying infrastructure. It combines Protégé’s object-oriented frame view with the logic and classification power of OWL/OilEd and then adds new user interface mechanisms to address well recognized limitations of each.

Three target groups of users are envisaged: neophytes, large scale ontology developers, and those interested in the underlying theory - logicians, ontologists, and students. Currently, there are two versions of the neophytes' interface - one based on "wizards" and the other on a more conventional frame-like view. Both deliberately limit the functionality available to an easily understandable but powerful subset. By contrast, the logicians' interface aims to give access to the full range of OWL features, but within a framework which is easier to use and more object oriented than OilEd's.

4.1.3 Principles and distinguishing features

All interfaces maintain five principles:

1. Object oriented presentation bringing together as much information as possible about each class in a single view. Whereas the OilEd interface is closer to OWL's fundamentally relational paradigm with the basic class definition and axioms in separate tabs, the Prot´eg´e interfaces keeps definitions and any axioms about each class in a single view.
2. Support for extensive metadata about the editorial process and status of the entities in the ontology.
3. A clear distinction between the classification as asserted by the author and the classification as inferred by reasoning engine.
4. The use of visual cues to indicate whether classes are primitive or defined and whether restrictions are 'asserted' or 'inherited', etc.
5. Automation of common operations, e.g. making primitive classes disjoint and moving expressions between axioms and definitions.

A major aim of the project is to develop visualization and debugging plugins within the overall architecture. The first of these, *OWLViz* (see picture below) , appears highly successful at least for moderately large ontologies. Other developments include much improved expression editors, usability aids such as text completers, and facilities inherited from the new version of Protégé including drag and drop features and the potential for multi-user operation.

4.1.4 Discussion and critical implementation factors

The project is still in an early stage, and there are plans for a wide range of additional tools and facilities to support the full life cycle of ontology development and maintenance. The Protégé plug and play framework is providing a flexible means of building multiple interfaces for different users on a single underlying infrastructure. Protégé native frame structure is a natural way to represent metadata about the ontology itself. The mapping between Protégé frame structure and OWL's logical structure is far from straightforward. It is worth noting, however, that 'restrictions' are treated as classes rather than slots in the Protégé framework, and the dynamic mapping between the OWL and Protégé views involves significant transformation.

4.2. ezOWL – Visual OWL Editor.

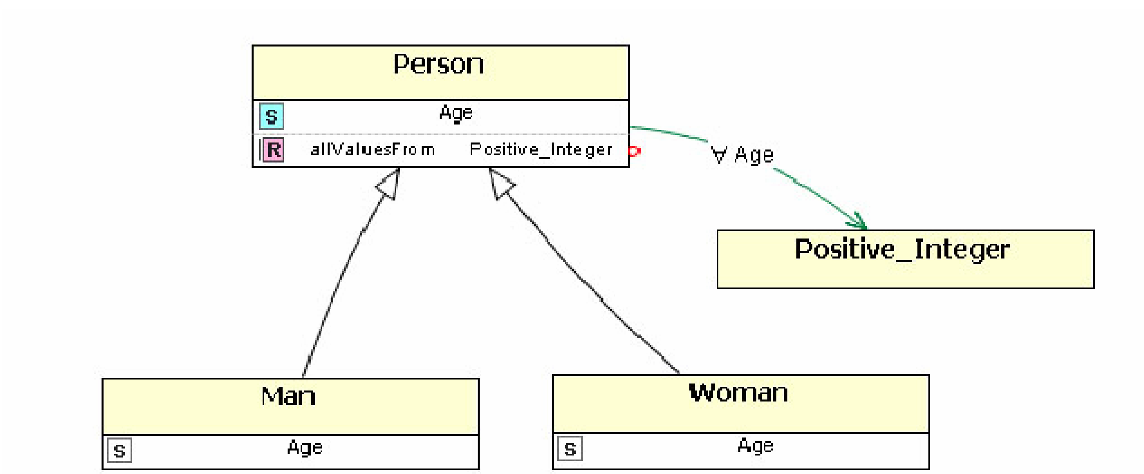
Sooyoung Oh, Moonyoung Chung and HyunKyu Cho
Department of Intelligent Services Platform Research
Electronics and Telecommunications Research Institute
{runtou, mchung, hkcho}@etri.re.kr

4.2.1 Introduction

ezOWL is a visual OWL editor that can intuitively construct an ontology based on OWL. There are two versions of ezOWL. One is a standalone version for a proprietary OWL editor and the other is a plug-in version for Protégé-2000. ezOWL standalone version supports more powerful features for authoring an OWL ontology and ezOWL plugin for Protégé-2000 was published already and it is based on Protégé-2000.

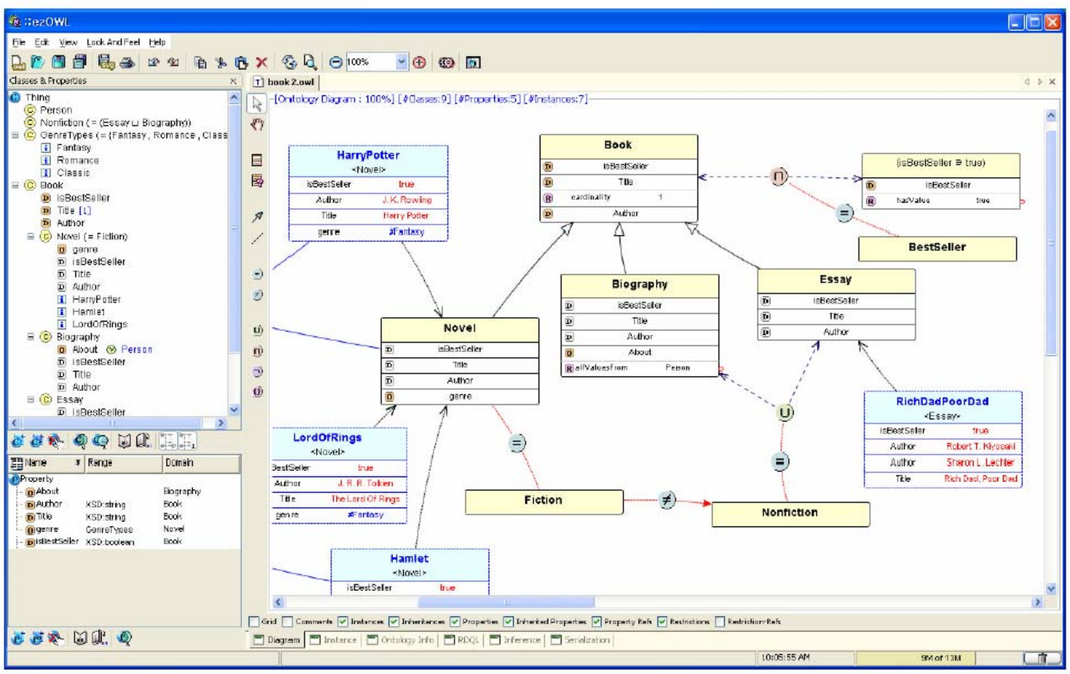
4.2.2 Principles and distinguishing features

ezOWL uses the diagram-based authoring techniques for supporting more intuitive ontology editing similar to UML. But, ezOWL dose not use UML. UML is difficult to represent an OWL ontology. ezOWL uses more simple representation method (see picture below).

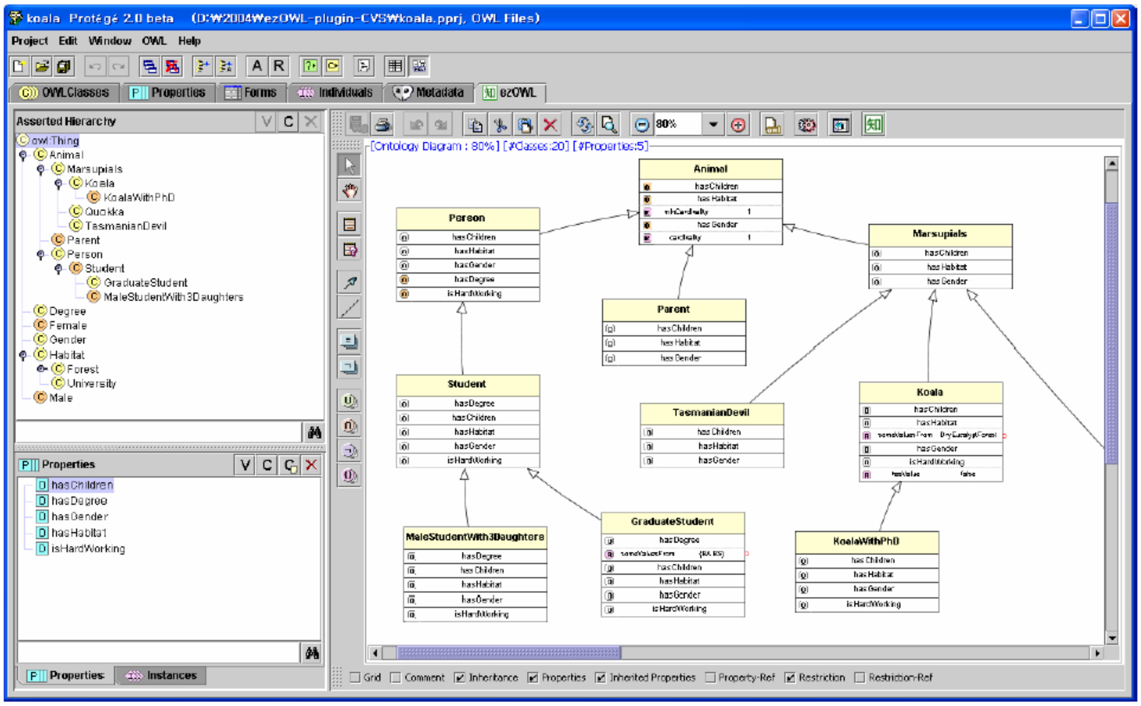


ezOWL supports classes, properties, restrictions, relations (class hierarchy, property reference, …), axioms (union, intersection, …) and instances. ezOWL Diagram View can control the level of view for accessing easily complex ontologies. In addition, ezOWL has the feature of printing a diagram and of saving a diagram to SVG format.

The ezOWL Visual OWL editor (stand alone version):



ezOWL plug-in for Protégé2000:



4.2.3 Discussion and critical implementation factors

A critical issue for ezOWL is the visual representation of a complex ontology. ezOWL Diagram View can also control the access level of the complex ontologies.

4.3 GATE

The Sheffield NLP Group.

4.3.1 Introduction

GATE (General Architecture for Text Engineering) is a free, well documented and widely used platform for the development, evaluation, integration and deployment of HLT components. While GATE does offer a number of ready-made components for Information Extraction (IE) and Named Entity Recognition (NER), its main purpose is to serve as a common interface for otherwise heterogeneous linguistic tools. Developers can either design new components from scratch or easily integrate existing tools by applying a wrapper that conforms to the CREOLE component model defined by GATE.

It has been in development over the past eight years in the Natural Language Processing Group at the University of Sheffield. The system has been used by thousands of people at hundreds of sites worldwide and was recently assessed as “outstanding” and “internationally leading” by an anonymous peer review.

GATE has been used for many language processing projects, in particular for Information Extraction in many languages including English, Bulgarian, Romanian, Swedish, Chinese, Arabic, Russian, Cebuano and a number of Indic languages. GATE has been used as the underlying platform and as a provider of Information Extraction services for a large number of research projects both in house and at other laboratories in locations around Europe, the United States, Japan, Australia, etc. The list of such projects includes SEKT, PrestoSpace, KnowledgeWeb, MUSE, MUMIS, AKT and The American National Corpus.

4.3.2 Principles and distinguishing features

In the last two decades, Information Extraction has emerged as a successful technology for extracting and managing the knowledge contained in large collections of textual data. The technology has attained increasing maturity and has partially migrated from academia into industry and commercial applications. The advent of the Internet, the emergence of the Semantic Web and the development of technologies such as Web Services provide both an opportunity and a challenge for the developers of IE systems. The opportunity is to make use of the existing expertise and resources for automatically creating semantic metadata for the next generation Internet – the Semantic Web. The challenge is posed by the sheer size of the task, by the diversity of the document types, domains, genres and languages currently present on the web.

GATE is well placed take advantage of that opportunity; the Sheffield NLP group is one of the world leading research centres for Information Extraction and was a participant in all the main evaluations in the field like MUC, ACE and TREC. GATE is also a good starting point for building a platform that addresses the new challenges facing the evolution of the Semantic Web. Development of Semantic Web related technologies is a strategic aim for the further development of the system and we have already been involved in research in this area through our participation in projects like AKT, hTechSight and Multiflora II. The GATE team is taking part in the next research step as well by being part of the consortia for the SEKT EU Integrated Project and the KnowledgeWeb Network of Excellence – two of the most important developments in this field on the European stage for the next few years.

Components in GATE can either be processing resources such as tokenizers, part-of-speech taggers or gazetteer lookups, or language resources such as documents, corpora, lexika, etc.. In principle however, any kind of resource can be integrated into the platform, as long as it conforms to the CREOLE component model.

Processing resources can be combined to so-called processing pipelines or applications. In an application, each component will be run in sequence on some language resource (most likely a corpus) and produce some kind of formal output. The output of one component is also accessible to the succeeding components, such that e.g. a part-of-speech tagger can make use of the output of a tokenizer.

Apart from serving as an integration and development platform, GATE also offers a number of other useful features, e.g. a pattern matching engine for the rule based, context sensitive JAPE language, and both document and corpus based evaluation tools, providing results with respect to the well-known precision, recall and f-measure metrics. The complete range of GATE's functionality is available through a graphical user interface (for the not so technically minded user), as well as through a well documented Java API, which allows easy integration of GATE in other applications, as can be seen in the sections on KIM, SWAN and Balto.

4.3.3 Discussion and critical implementation factors

KIM provides the NLP support for various semantic web applications mentioned before. It has been developed for a long time -even before the Semantic Web started to emerge. KIM is one of the key tool packages in NLP areas and now acts as the bridge for both communities. Gate is an open source project, and therefore can be widely adopted. But for different applications, the Gate system needs to be tuned.

4.4 OWLCHESTRA: Facilitating the Development and Publishing of Small-Scale Web Ontologies

Benjamin Nowack
appmosphere web applications, Kruppstr. 82-100, 45145 Essen, Germany
bnowack@appmosphere.com

4.4.1 Introduction

OWLCHESTRA is a browser-based Web ontology modeler with an integrated publishing system. It eases the collaborative and distributed development of OWL and RDF Schema vocabularies. The publishing tool can be used to generate both machine-readable RDF/XML serializations and human-oriented documentation. The demonstration includes an introduction to OWLCHESTRA's graphical user interface, the modeling of a multi-namespace ontology, and the system's customizable serialization and publishing features. The integrated creation of semantic Web pages can be demonstrated as well.

4.4.2 Motivation and goal

Web ontologies are the backbone of semantic Web applications. Instead of trying to model a huge central ontology, the Semantic Web is expected to be built on a large number of distributed (rather small) vocabularies, which can be combined and extended. Indeed, most of the current RDF Schema and OWL vocabularies in wider use consist of less than 75 terms1. Among other things, the success (i. e. the ability to create a large user base) of a new ontology depends on its documentation, including a description of the terms, code examples, and links to related information. As ontologies usually change over time, it is important to update both the formal serialization and the user-oriented information.

Although OWL-capable editors such as Protégé can export a browsable HTML documentation, updates have to be manually initiated. Apart from that, most of the OWL implementations (tracked by the Web Ontology Working Group during the Candidate Recommendation phase of OWL) require Java support and root privileges for installation. The demonstrated toolkit tries to address the points mentioned above. It cannot compete with Java-based APIs and tools in terms of features or performance. But it may attract those developers who prefer using scripting languages for their applications.

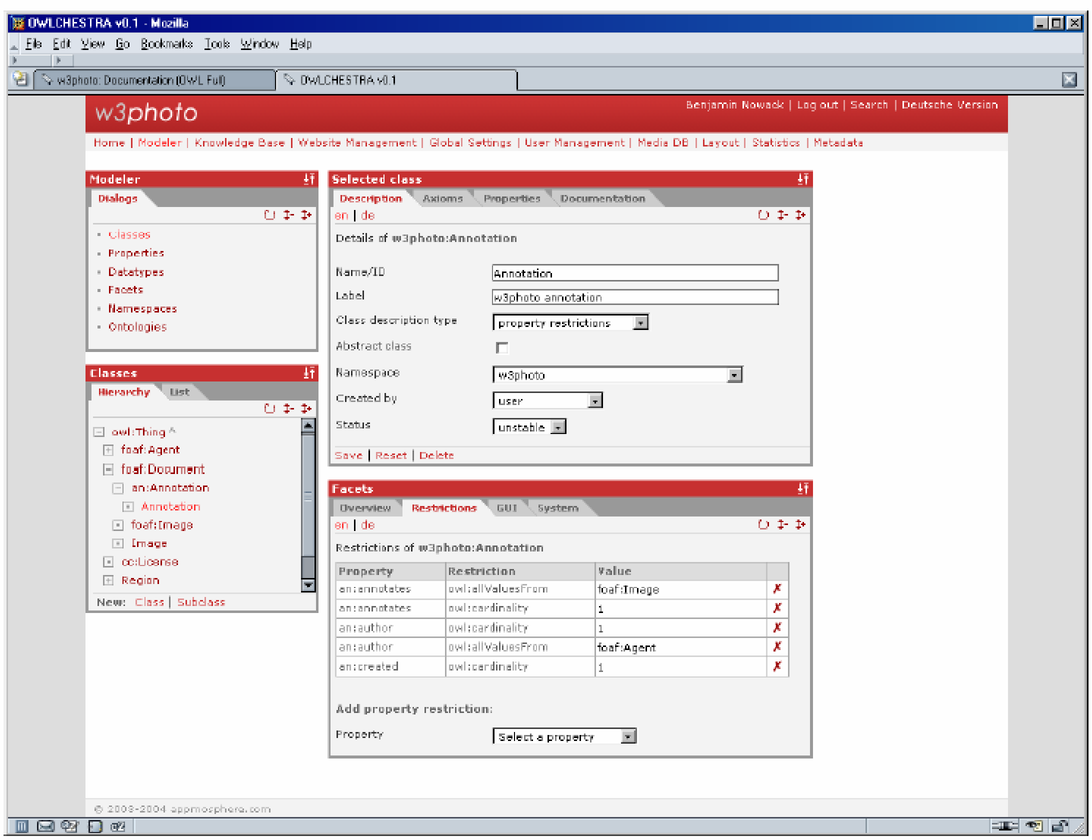
4.4.3 Principles and distinguishing features

OWLCHESTRA is based on PHP and uses a MySQL database for storing ontologies, content and application data. It can be installed on standard LAMP servers and is therefore particularly suitable for hosted Web server environments that don't support Java. It can be used for collaborative modeling, ontology prototyping, and for OWL demonstrations.

4.5.3 Discussion and critical implementation factors

OWLCHESTRA is OWL-based, browser-based web ontology editor with an integrated publishing system. It eases the collaborative and distributed development of OWL and RDF Schema vocabularies. The publishing tool can be used to generate both machine-readable RDF/XML serializations and human-oriented documentation. One of the application of OWLCHESTRA is to handle the W3C conference photos.

OWLCHESTRA is still at a very early stage, but some features can already be demonstrated. The following screenshot shows the browser-based ontology modeler.



5. Concluding Remarks

Observations

At the first International Semantic Technology Road Show, three of the nine demonstrators fall in the category ´semantic tools´, the remaining six demonstrators cover a wide range of end-user applications. In addition, when discussing with participants, there was often a reference to other semantic applications that had been built but were not chosen to demonstrate at this particular road show. This reflects what we consider a general tendency in the semantic research community; *a shift from the development of tools towards the deployment of applications.*

Semantic Technologies:
A shift from
tools to applications

This shift can be considered a natural change; the first generation of semantic technologies focused on building the foundations for semantic applications, and now the construction is moving upward. The Promoottori team formulated this natural tendency when indicating that Promoottori was built in order to test and validate their research approach in the semantic domain.

Semantic tools:
Taking the user serious

We noticed another important step forward within the category of semantic tools. *Whereas in previous years, the focus was mainly on technology, the tools now seriously address issues of user-friendliness*. Another remarkable observation is that many tools share parts of eachother, which indicates a clear beginning of standardization. An example is the Protégé editor which is being enhanced or plugged on for OWL editing (by for example ezOWL and Co-ode). Other examples are KIM, SWAN and Balto, which all incorporate the GATE platform as the infrastructure for their Human Language Technology tasks.

Semantic tools:
Towards standarization!

Critical issues for implementation in a business environment

All participants were asked what they would consider as a major critical issue if their demonstrator would be implemented in a business environment. In the category of semantic application, this question was sometimes considered odd; the application was already up and running. The development teams did however remark that (1) it had initially not been easy to convince the target group that this *was* the best solution (for example Museum Finland) and (2) that there was no detailed data available on actual use and end user satisfaction. An interesting field of study that still lays bare!

The semantic tool teams mentioned versioning, maintenance and support as a major critical issue for actual deployment in a business environment. In fact, some even indicated that versioning strategy was a critical issue for themselves. It was for example remarked that it requires a fair amount of time to keep up with the continuous new releases of Protégé.

Conclusions

The most promising observation of the Road Show was the open, inspiring atmosphere full of dialog. Every demonstrator and visitor was more that willing to share experiences, to discuss pros and cons and to think through possible future developments. For the work of our WorkPackage specifically, and KWEB in general, we would like to indicate some lessons learned.

First, we consider the shift towards the development of semantic applications rather than semantic tools an opportunity for the promotion of semantic technologies that we should embrace. In other words; *we suggest that WP 1.4 should focus more on semantic applications rather than semantic tools.*

At the same time, we noticed that whereas the semantic research community is well informed about the semantic tools that are being developed, and of the direction the development is taking, the knowledge on the semantic applications is less disseminated.

We can safely assume that if the research community is not well informed on the semantic applications that are being developed, the business community is neither. *Hence we conclude that the major critical issue for the adoption of semantic web technologies is the sheer knowledge of the existence of semantic applications and their possibilities.*

Recommendations

Promoting Semantic Technologies: Use the power of the applications

In discussing the previous observations with participants of the KWEB conference, several interesting ideas took shape. Based on these discussions we put forward the following two recommendations:

(1) A call out to KWEB partners to explore the possibility within KWEB for the development of a user driven web-based repository for semantic applications. (ensure alignment with other initiatives in this field, for example current efforts at the W3C)

(2) A recommendation to develop a semantic application identification template which enumerates central features of the technology in a simple Yes/No list. The identification template should allow for a swift and generic understanding of the applications. The attached template 1 could function as a base of development.

6. Annex: A First Categorization of Semantic Applications and Tools

6. 1 A Semantic Application Identification Template

Following a semantic application template we developed drafting this report. The major considerations taken into account are:

- The template has to be short and easy to complete;
- The entries have to exclude misunderstanding;
- The central features of the application have to be presented;
- Considerations that are important for actual adoption of industry and organizations have to be included.

Application Name:
Developed by:
Intended user group / target group:

	YES	NO
The main feature of the application is data integration		
The main feature of the application is to browse data		
The main feature of the application is to search data		
The main feature of the application is to ……………..		
The main feature of the application is to……………..		

YES NO

The application is user friendly		
Users interact with each other		
The application is a remote, server-based application (as opposed to a local desktop application)		
The application is in finished status (versus β version)		
Development of the application will be continued beyond its current status		
………….		

6. 2 A Semantic Tool Identification Template

Following the semantic tool template we developed drafting this report. The same considerations as for the semantic application were taken into account.

Tool Name:
Developed by:
Intended user group / target group:

	YES	NO
Editor		
Ontology		
Mapping		
Human Language Techn.		
Visualization		
Repository		
Reasoners		
API		
Tool located on server		
Finished status (versus β)		
Ongoing development		
…………..		
…………..		
…………..		