

Auer, Sören (Ed.); Schaffert, Sebastian (Ed.); Pellegrini, Tassilo (Ed.)

**Proceedings — Published Version**

## I-SEMANTICS '08: International conference on semantic systems

*Suggested Citation:* Auer, Sören (Ed.); Schaffert, Sebastian (Ed.); Pellegrini, Tassilo (Ed.) (2008) : I-SEMANTICS '08: International conference on semantic systems, Verlag der Technischen Universität Graz, Graz

This Version is available at:

<https://hdl.handle.net/10419/44448>

**Standard-Nutzungsbedingungen:**

Die Dokumente auf EconStor dürfen zu eigenen wissenschaftlichen Zwecken und zum Privatgebrauch gespeichert und kopiert werden.

Sie dürfen die Dokumente nicht für öffentliche oder kommerzielle Zwecke vervielfältigen, öffentlich ausstellen, öffentlich zugänglich machen, vertreiben oder anderweitig nutzen.

Sofern die Verfasser die Dokumente unter Open-Content-Lizenzen (insbesondere CC-Lizenzen) zur Verfügung gestellt haben sollten, gelten abweichend von diesen Nutzungsbedingungen die in der dort genannten Lizenz gewährten Nutzungsrechte.

**Terms of use:**

*Documents in EconStor may be saved and copied for your personal and scholarly purposes.*

*You are not to copy documents for public or commercial purposes, to exhibit the documents publicly, to make them publicly available on the internet, or to distribute or otherwise use the documents in public.*

*If the documents have been made available under an Open Content Licence (especially Creative Commons Licences), you may exercise further usage rights as specified in the indicated licence.*

**I-SEMANTICS '08**  
**International Conference on**  
**Semantic Systems**

<b>Preface</b>	1
S. Auer, S. Schaffert, T. Pellegrini	
<b>Wikis in Global Businesses</b>	4
P. Kemper	
<b>Managing Knowledge that Everybody Knows Already</b>	5
H. Lieberman	
<b>Humans and the Web of Data</b>	6
T. Heath	
<b>Commercialization of Semantic Technologies in Malaysia</b>	7
D. Lukose	

## *Scientific Track*

### *Semantic Web*

- What is the Size of the Semantic Web?** 9  
M. Hausenblas, W. Halb, Y. Raimond, T. Heath
- The Utility of Specific Relations in Ontologies for Image Archives** 17  
A. Walter, G. Nagypal
- Semantic Web Applications in- and outside the Semantic Web** 25  
C. Carstens

### *Semantic Web for Collaboration*

- Collaborative Tasks using Metadata Conglomerates – The Social Part of the Semantic Desktop** 34  
O. Grebner, H. Ebner
- Phrase Detectives - A Web-based Collaborative Annotation Game** 42  
J. Chamberlain, M. Poesio, U. Kruschwitz
- Building a Semantic Portal from the Discussion Forum of a Community of Practice** 50  
M. Bassem, K. Khelif, R. Dieng-Kuntz, H. Cherfi

### *Semantic Web for Processes and Policies*

- A Semantic Approach towards CWM-based ETL processes** 58  
A.-D. Hoang Thi , T. B. Nguyen
- A Semantic Policy Management Environment for End-Users** 67  
J. Zeiss, R. Gabner, A. V. Zhdanova, S. Bessler
- A Knowledge Base Approach for Genomics Data Analysis** 76  
L. Kefi-Khelif, M. Demarchez, M. Collard

### *Semantic Web Services*

- GRISINO – a Semantic Web services, Grid computing and Intelligent Objects integrated infrastructure** 85  
T. Bürger, I. Toma, O. Shafiq, D. Dögl, A. Gruber
- Pervasive Service Discovery: mTableaux Mobile Reasoning** 93  
L. Steller, S. Krishnaswamy
- Community Rating Service and User Buddy Supporting Advices in Community Portals** 102  
M. Vasko, U. Zdun, S. Dustdar, A. Blumauer, A. Koller, W. Praszl

*Folksonomies*

- Seeding, Weeding, Fertilizing - Different Tag Gardening Activities for Folksonomy Maintenance and Enrichment** 110  
K. Weller, I. Peters
- Quality Metrics for Tags of Broad Folksonomies** 118  
C. Van Damme, M. Hepp, T. Coenen

*Knowledge Engineering*

- Conceptual Interpretation of LOM and its Mapping to Common Sense Ontologies** 126  
M. E. Rodriguez, J. Conesa, E. García-Barriocanal, M. A. Sicilia
- Collaborative Knowledge Engineering via Semantic MediaWiki** 134  
G. Chiara, M. Rospocher, L. Serafini, B. Kump, V. Pammer, A. Faatz, A. Zinnen, J. Guss, S. Lindstaedt
- Building Ontology Networks: How to Obtain a Particular Ontology Network Life Cycle?** 142  
M. C. Suárez-Figueroa, A. Gómez-Pérez
- Managing Ontology Lifecycles in Corporate Settings** 150  
M. Luczak-Rösch, R. Heese
- Interoperability Issues, Ontology Matching and MOMA** 158  
M. Mochol



### *Short Papers*

<b>Exploiting a Company's Knowledge: The Adaptive Search Agent YASE</b>	166
A. Kohn, F. Bry, A. Manta	
<b>Integration of Business Models and Ontologies: An Approach used in LD-CAST and BREIN</b>	170
R. Woitsch, V. Hrgovcic	
<b>Semantic Tagging and Inference in Online Communities</b>	174
A. Yildirim, S. Üsküdarli	
<b>Semantics-based Translation of Domain Specific Formats in Mechatronic Engineering</b>	178
M. Plößnig, M. Holzapfel, W. Behrendt	
<b>Semantifying Requirements Engineering – The SoftWiki Approach</b>	182
S. Lohmann, P. Heim, S. Auer, S. Dietzold, T. Riechert	
<b>RDFCreator – A Typo3 Add-On for Semantic Web based E-Commerce</b>	186
H. Wahl, M. Linder, A. Mense	

## *Triplification Challenge*

<b>Introduction</b> S. Auer, S. Schaffert, T. Pellegrini	190
<b>DBTune – <a href="http://dbtune.org/">http://dbtune.org/</a></b> Y. Raimond	191
<b>Linked Movie Data Base</b> O. Hassanzadeh, M. Consens	194
<b>Semantic Web Pipes Demo</b> D. Le Phuoc	197
<b>Triplification of the Open-Source Online Shop System osCommerce</b> E. Theodorou	201
<b>Interlinking Multimedia Data</b> M. Hausenblas, W. Halb	203
<b>Integrating triplify into the Django Web Application Framework and Discover Some Math</b> M. Czygan	205
<b>Showcases of Light-weight RDF Syndication in Joomla</b> D. Le Phuoc, N. A. Rakhmawati	208
<b>Automatic Generation of a Content Management System from an OWL Ontology and RDF Import and Export</b> A. Burt, B. Joerg	211



# **I-SEMANTICS '08**

## **International Conference on Semantic Systems**

### **Preface**

This volume contains the proceedings of the I-SEMANTICS '08 which together with the I-KNOW '08 and I-MEDIA '08 are part of the conference series TRIPLE-I. TRIPLE-I is devoted to explore the fields of knowledge management, new media and semantic technologies.

I-SEMANTICS '08 offered a forum for exchange of latest scientific results in semantic systems and complements these topics with new research challenges in the area of social software, semantic content engineering, logic programming and Semantic Web technologies. The conference is in its fifth year now and has developed into an internationally visible event.

The conference attracted leading researchers and practitioners who presented their ideas to about 450 attendees. Attendees have been provided with high quality contributions reviewed by a program committee featuring renowned international experts on a broad range of knowledge management topics. This year, 25 high quality papers were accepted for inclusion in the conference proceedings of I-SEMANTICS. The program of the I-SEMANTICS '08 was structured as follows: In the main conference the contributors of long papers gave their presentations in thematically grouped sessions. Submitters of short papers had the opportunity to present their research in a poster session.

These presentations covered a broad palette on current trends and developments in semantic technologies amongst others:

- Communities supported by Semantic Technologies
- Folksonomies
- Knowledge Engineering
- Semantic Web Services
- Semantic Web for Collaboration
- Semantic Web in Industry
- Semantic Web Applications

For the first time I-SEMANTICS included a Linking Open Data Triplification Challenge which awarded three prizes to the most promising triplifications of existing Web applications, Websites and data sets. The challenge was supported by OpenLink, punk.netServices, W3C and patronized by Tim Berners-Lee. We received a number of submissions from which 8 were nominated for the final challenge. The winners

were elected by the challenge committee and announced at the I-SEMANTICS conference.

Besides, we offered an international cooperation event which served only the purpose of fostering networking among researchers and between researchers and practitioners. Also, deliberately long breaks in a well-suited venue throughout the conference and social events provided excellent opportunities for meeting people interested in semantics related topics from different disciplines and parts of the world.

We are grateful to our invited keynote speakers Henry Lieberman (MIT, USA), Peter Kemper (SHELL, Netherlands), Tom Heath (TALIS, United Kingdom) and Dickson Lukose (MIMOS, Malaysia) for sharing with our attendees their ideas about the future development of knowledge management, new media and semantic technologies. Many thanks go to all authors who submitted papers and of course to the program committee which provided careful reviews in a quick turnaround time. The contributions selected by the program committee are published in these proceedings.

We would like to thank the sponsors insiders GmbH, Punkt. netServices, Top Quadrant and Go International. Special thanks also go to Dana Kaiser who prepared the conference proceedings in time and with highest quality. We also would like to thank our staff at the Know-Center, the Semantic Web School and Salzburg NewMediaLab for their continuous efforts and motivation in organizing these two conferences.

We hope that I-SEMANTICS '08 will provide you with new inspirations for your research and with opportunities for partnerships with other research groups and industrial participants.

Sincerely yours,

Sören Auer, Sebastian Schaffert, Tassilo Pellegrini

Graz, August 2008

### **Program Committee I-SEMANTICS '08**

- Baker Tom, Kompetenzzentrum Interoperable Metadaten, Germany
- Bergman Michael K., Web Scientist, USA
- Bizer Chris, Free University Berlin, Germany
- Blumauer Andreas, Semantic Web Company, Austria
- Breslin John, DERI Galway, Ireland
- Buerger Tobias, Salzburg Research, Austria
- Cyganiak Richard, DERI Galway, Ireland
- Damjanovic, Violeta, Salzburg Research, Austria
- Davies Marcin, ftw. Forschungszentrum Telekommunikation Wien, Austria
- Dietzold Sebastian, University of Leipzig, Germany
- Erling Orri, OpenLink Software, USA
- Gams Erich, Salzburg Research, Austria
- Gray Jonathan, Open Knowledge Foundation, Great Britain
- Grimnes Gunnar AAstrand, DFKI, Germany
- Groza Tudor, DERI Galway, Ireland
- Güntner Georg, Salzburg Research, Austria
- Heath Tom, Talis, Great Britain
- Hellmann, Sebastian, Universität Leipzig, Germany
- Houben Geert-Jan, Vrije Universiteit Brussel, Netherlands
- Kopsa Jiri, Sun Microsystems, USA
- Kühne, Stefan, University of Leipzig, Germany
- Lange, Christoph, DERI Galway / Jacobs University Bremen, Ireland / Germany
- Lehmann Jens, University of Leipzig, Germany
- Loebe Frank, University of Leipzig, Germany
- Lohmann Steffen, University of Duisburg, Germany
- Lukose Dickson, MIMOS, Malaysia
- Martin, Michael, Universität Leipzig, Germany
- Mika Peter, Yahoo! Research Barcelona, Spain
- Mitterdorfer Daniel, Salzburg Research, Austria
- Mueller Claudia, University of Potsdam, Germany
- Pellegrini Tassilo, Semantic Web Company, Austria
- Riechert, Thomas, University of Leipzig, Germany
- Sack Harald, University of Potsdam, Germany
- Stocker Alexander, Know Center Graz, Austria
- Troncy Raphaël, Centre for Mathematics and Computer Science, Netherlands
- Voss Jakob, Wikimedia e.V., Germany
- Wieser Christoph, Salzburg Research, Austria
- Zhdanova Anna V., ftw. Forschungszentrum Telekommunikation Wien, Austria

## **Wikis in Global Businesses**

### **Keynote Speaker**

#### **Peter Kemper**

(SHELL International, Netherlands  
peter.kemper@shell.com)

In his talk Peter Kemper outlines the use of emerging Web 2.0 Technologies in a multi-national enterprise environment. Shell's internal Wiki - having over 40.000 users - demonstrates the success of Wiki Technology in a corporate setting. Fully integrated into Shell's information landscape, it fosters a central anchor point for technical, business and scientific knowledge in Shell. This central pool of knowledge offers new ways of information management for Shell's employees and establishes a platform for new innovative services like for example automatic linking of documents. Shell's Wiki cultivates collaboration and knowledge work across national borders - making knowledge transfer globally happening. Peter Kemper will point out the details of this 2.5 year old success story and will share his experience in deploying Web 2.0 technologies in Enterprises in this talk

### **About Peter Kemper**

Peter Kemper works since 1982 in Information Technology within Shell. After his Bachelor degree from the Rotterdam Nautical College he sailed as a Ship's Officer on both passenger ships as well as VLCC's (Very Large Crude Carriers). In 1981 he followed the internal Shell Informatics Education with BSO (now ATOS Origin). He worked in several different Shell companies (Nederlandse Aardolie Maatschappij, Pernis Refinery, Shell Nederland and Shell Exploration & Production) and his current work is within the Knowledge, Innovation & Design team of Shell International as Knowledge Management portfolio manager. Current projects are the Shell Wiki and several innovation projects to Virtual Worlds and Information Similarity Checking.

## **Managing Knowledge that Everybody Knows**

### **Keynote Speaker**

**Henry Lieberman**

(MIT, USA)

lieber@media.mit.edu)

Traditional knowledge management is focused on representing knowledge that is special in some way: unique to a person or group; technical or specialized knowledge; specific situation-dependent data about people, things or events. What everybody forgets is that that specialized knowledge builds on a base of Commonsense knowledge -- simple, shared knowledge about everyday life activities. A database might represent an airline flight with airline name, flight number, origin and destination time and place, etc. But no database represents the fact that if you travelling less than a kilometer, you can walk; if you are travelling thousands of kilometers, you probably need to fly. Why bother to represent this obvious knowledge explicitly, since everybody knows these things already? Because computers don't. If we would like to have computers be helpful to people, avoid stupid mistakes, and make reasonable default guesses about what people might want, they have to have Commonsense knowledge. I will present Open Mind Common Sense, a project to collect human Commonsense knowledge; ConceptNet, its semantic representation; and AnalogySpace, a new reasoning technique that draws plausible inferences, despite the fact that our knowledge base is incomplete, imprecise, and inconsistent.

### **About Henry Lieberman**

Henry Lieberman has been a Research Scientist at the MIT Media Laboratory since 1987. His interests are in the intersection of artificial intelligence and the human interface. He directs the Software Agents group, which is concerned with making intelligent software that provides assistance to users in interactive interfaces.



## **Humans and the Web of Data**

### **Keynote Speaker**

#### **Tom Heath**

(TALIS, United Kingdom  
tom.heath@talिस.com)

How will the availability of linked, machine-readable data change the way humans interact with the Web, and what role can existing social processes play in supporting this interaction? In contrast to the conventional Web, in which documents are designed primarily for human consumption and connected by untyped links, the Semantic Web is one in which data is published in machine-readable form and the nature of connections between related items is made explicit. The transition from a Web of documents to a Web of data lowers the barriers to integration of data from distributed sources, and paves the way for a new generation of applications that can exploit this in order to enhance the user experience. This talk will demonstrate how the Web of data has moved from vision into reality, question how applications built on this distributed data set may change our mode of interaction with the Web, and examine how the Web of data might allow existing social processes to mitigate spam and information overload.

### **About Tom Heath**

Tom Heath is a researcher in the Platform Division at Talis, a UK software company specialising in Semantic Web technologies and applications, where he is responsible for research into recommender systems and collective intelligence. Tom's work at Talis builds on his previous doctoral research into trust, recommendation and social networks in the Semantic Web, conducted at The Open University's Knowledge Media Institute. As part of that work he developed Revyu.com, a reviewing and rating site for the Web of data and winner of the 2007 Semantic Web Challenge. Tom has over 10 years development experience with Web technologies, and a first degree in Psychology.

## **Commercialization of Semantic Technologies in Malaysia**

### **Keynote Speaker**

**Dickson Lukose**

(MIMOS, Malaysia)

dickson.lukose@mimos.my)

In recent years, Semantic Technologies have been in the forefront of attention of the major governments, industry, academic and investors around the world. Much research was conducted by Artificial Intelligence researches in the 1980's in the area of Knowledge Representation and Reasoning, specifically in the area of Semantic Networks. But, only recently, we achieved standardization via the W3C initiatives, which gave the impetus for industry players and investors to look seriously into Semantic Technologies. Though the major drivers of innovation in Semantic Technologies are Semantic Web, one could see that Semantic Technologies are a somewhat broader concept than the Semantic Web. Although the Semantic Web is obviously based on Semantic Technologies, the latter include non-Web applications. The main goal of Semantic Technologies is to capture (semi-)structured and unstructured knowledge in a format that can be used by computers and humans alike. When one looks into the overall eco-system that drives semantic technologies, one could conclude that Europe is pretty much the leaser in research, while north-America is leading in the development of semantic technology tools. Even though there are numerous application developments taking place in Europe and north-America, the industry analyst predictions are that largest growth and investments will be taking place in the Asia region. Realization of this is no well reflected in the national R & D agenda of many nations in the region. Malaysia is one of the countries in the Asian region that are totally committed in preparing itself to capitalize on frontier technologies. Semantic Technology being one of the major focuses. Conducting applied R & D on Semantic Technologies and moving these technologies to local industries to take it up is a monumental challenge. Some of the major obstacles faced (certainly not an exhaustive list) include the lack of expert researches in Semantic Technology in the country, non-competitive compensation packages makes it difficult to attract best people from around the world to Malaysia, lack of research culture among the local industry players, preference of local industry to purchase western technologies rather the home grown technologies, and the lack of skilled personal within our local industry players to take on sophisticated technologies like the Semantic Technology. In this keynote address, the speaker will outline the methodology adopted by MIMOS BHD on how we overcome the above mentioned challenges, to carry our the necessary R & D in Semantic Technologies, preparing local industries to become our technology recipient, and how we help local companies to commercialize Semantic Technologies.

**About Dickson Lukose**

Dr. Dickson Lukose (PhD) is the Head of the Knowledge Technology Cluster at MIMOS BHD. Dr Lukose is also the director of the Artificial Intelligence Laboratory as well as the Centre of Excellence in Semantic Technologies. Prior to MIMOS BHD, Dr Lukose was involved with a startup company named DL Informatique Sdn. Bhd., an MSC Status Company specializing in the applications of Artificial Intelligence Technology in developing software applications in the areas of Risk Management and Knowledge Management.

## What is the Size of the Semantic Web?

**Michael Hausenblas, Wolfgang Halb**

(Institute of Information Systems & Information Management  
JOANNEUM RESEARCH, Graz, Austria  
firstname.lastname@joanneum.at)

**Yves Raimond**

(Centre for Digital Music, Queen Mary University of London  
London, United Kingdom  
yves.raimond@elec.qmul.ac.uk)

**Tom Heath**

(Talis  
Birmingham, United Kingdom  
tom.heath@talis.com)

**Abstract:** When attempting to build a scaleable Semantic Web application, one has to know about the size of the Semantic Web. In order to be able to understand the characteristics of the Semantic Web, we examined an interlinked dataset acting as a representative proxy for the Semantic Web at large. Our main finding was that regarding the size of the Semantic Web, there is more than the sheer number of triples; the number and type of links is an equally crucial measure.

**Key Words:** linked data, Semantic Web, gauging

**Category:** H.m, D.2.8

### 1 Motivation

Developments in the last twelve months demonstrate that the Semantic Web has arrived. Initiatives such as the Linking Open Data community project<sup>1</sup> are populating the Web with vast amounts of distributed yet interlinked RDF data. Anyone seeking to implement applications based on this data needs basic information about the system with which they are working. We will argue that regarding the size of the Semantic Web, there is more to find than the sheer numbers of triples currently available; we aim at answering what seems to be a rather a simple question: *What is the size of the Semantic Web?*

We review existing and related work in section 2. Section 3 introduces the linked dataset we use for our experiments. Further, in section 4 we analyse the reference dataset syntactically and semantically attempting to answer the *size* question. Finally, we conclude our findings in section 5.

---

<sup>1</sup> <http://linkeddata.org/>

## 2 Existing Work

On the Web of Documents, typically the number of users, pages or links are used to gauge its size [Broder et al. 00, Gulli and Signorini 05]. However, Web links (`@href`) are untyped, hence leaving its interpretation to the end-user [Ayers 07]. On the Semantic Web we basically deal with a directed labelled graph where a fair amount of knowledge is captured by the links between its nodes.

From semantic search engines we learn that mainly the documents and triples as such are counted. No special attention is paid to the actual interlinking, i.e. the type of the links [Esmaili and Abolhassani 06]. In the development of the semantic search engine `swoogle` [Finin et al. 05] it has been reported that “... the size of the Semantic Web is measured by the number of discovered Semantic Web Documents”. However, later, they also examined link characteristics [Ding and Finin 06]. Findings regarding the distribution of URIs over documents are well known in the literature [Tummarello et al. 07, Ding et al. 05]. Unlike other gauging approaches focusing on the schema level [Wang 06], we address the interlinking aspect of Semantic Web data represented in RDF, comparable to what Ding et. al. [Ding et al. 05] did in the FOAF-o-sphere.

## 3 Linked Datasets As A Proxy For The Semantic Web

The *reference test data set* (RTDS) we aim to use should be able to serve as a good proxy for the Semantic Web, hence it (i) must cover a range of different topics (such as people-related data, geo-spatial information, etc.), (ii) must be strongly interlinked, and (iii) must contain a sufficient number of RDF triples (we assume some millions of triples sufficient). As none of the available alternatives—such as the Lehigh University Benchmark dataset<sup>2</sup>, Semantic Wikis (such as [Völkel et al. 06]) or embedded metadata—exhibit the desired characteristics, the Linking Open Data datasets were chosen as the RTDS. We note that embedded metadata (in the form of microformats, RDFa, eRDF and GRDDL) are constituting a large part of the openly published metadata. However, the interlinking of this data is not determinable unambiguously.

The basic idea of linked data was outlined by Sir Tim Berners-Lee; in his note<sup>3</sup>, a set of rules is being provided. The Linking Open Data (LOD) project is a collaborative effort; it aims at bootstrapping the Semantic Web by publishing datasets in RDF on the Web and creating large numbers of links between these datasets [Bizer et al. 07]. As of time of writing roughly two billion triples and three million interlinks have been reported (cf. Fig. 1<sup>4</sup>, ranging from rather centralised ones to those that are very distributed. A detailed description of the datasets contained in the LOD is available in Table 1.

<sup>2</sup> <http://swat.cse.lehigh.edu/projects/lubm/>

<sup>3</sup> <http://www.w3.org/DesignIssues/LinkedData.html>

<sup>4</sup> by courtesy of Richard Cyganiak, <http://richard.cyganiak.de/2007/10/lod/>

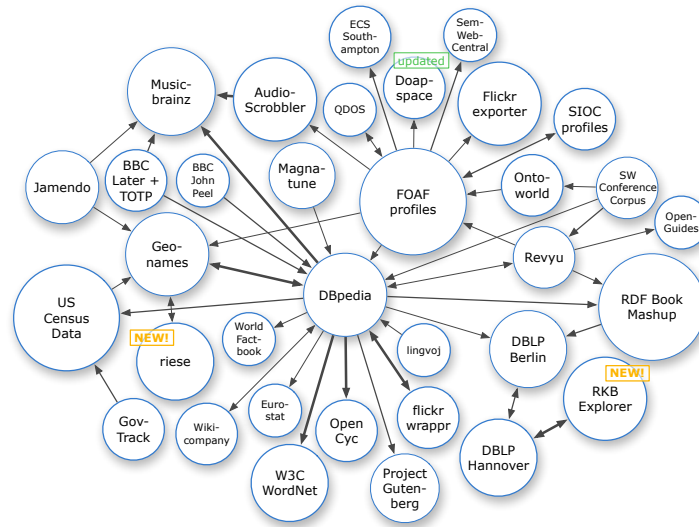


Figure 1: The Linking Open Data dataset at time of writing.

#### 4 Gauging the Semantic Web

In order to find metrics for the Semantic Web we examine its properties by inducing from the LOD dataset analysis. One possible dimension to assess the size of a system like the Semantic Web is the data dimension. Regarding data on the Semantic Web, we roughly differentiate into: (i) the **schema level** (cf. ontology directories, such as OntoSelect<sup>5</sup>), (ii) the **instance level**, i.e. a concrete occurrence of an item regarding a certain schema (see also [Hausenblas et al. 07]), and the actual **interlinking**: the connection between items; represented in URIs and interpretable via HTTP. This aspect of the data dimension will be the main topic of our investigations, below.

As stated above, the pure number of triples does not really tell much about the size of the Semantic Web. Analysing the links between resources exhibits further characteristics. The LOD dataset can roughly be partitioned into two distinct types of datasets, namely (i) **single-point-of-access datasets**, such as DBpedia or Geonames, and (ii) **distributed datasets** (e.g. the FOAF-o-sphere or SIOC-land). This distinction is significant regarding the access of the data in terms of performance and scalability.

Our initial approach aimed at loading the whole LOD dataset into a relational database (Oracle 11g Spatial). Due to technical limitations this turned

<sup>5</sup> <http://olp.dfki.de/ontoselect/>

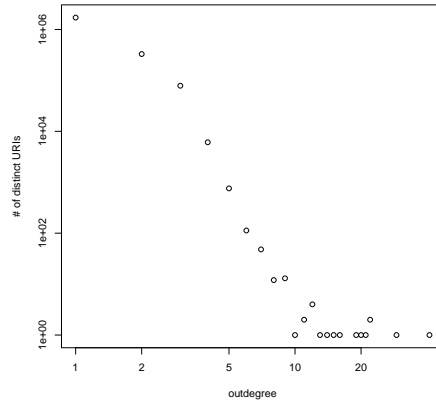
Name	Triples (millions)	Interlinks (thousands)	Dump download	SPARQL endpoint
BBC John Peel	0.27	2.1		
DBLP	28	0		yes
DBpedia	109.75	2,635	yes	yes
Eurostat	0.01	0.1		yes
flickr wrappr	2.1	2,109		
Geonames	93.9	86.5	yes	
GovTrack	1,012	19.4	yes	yes
Jamendo	0.61	4.9	yes	yes
lingvoj	0.01	1.0	yes	
Magnatune	0.27	0.2	yes	yes
Musicbrainz	50	0		
Ontoworld	0.06	0.1	yes	yes
OpenCyc	0.25	0	yes	
Open-Guides	0.01	0		
Project Gutenberg	0.01	0		yes
Revyu	0.02	0.6	yes	yes
riese	5	0.2	yes	yes
SemWebCentral	0.01	0		
SIOC	N/A	N/A		
SW Conference Corpus	0.01	0.5	yes	yes
W3C Wordnet	0.71	0	yes	
Wikicompany	?	8.4		
World Factbook	0.04	0		yes

**Table 1:** Linking Open Data dataset at a glance.

out not to be feasible—the overall time to process the data exceeded any sensible time constraints. As not all LOD datasets are available as dumps, it became obvious that additional crawling processes were necessary for the analysis. We finally arrived at a hybrid approach. The available and the self-crawled dumps together were loaded into the relational database, where the analysis took place using SQL. Additionally, we inspected the descriptions provided by the LOD dataset providers in order to identify parts of the dataset which are relevant for interlinking to other datasets. Where feasible, we also used the available SPARQL-endpoints.

#### 4.1 Single-point-of-access Datasets

It has to be noted that only a certain subset of the links actually yields desirable results in the strict sense, i.e. return RDF-based information when performing an HTTP GET operation. Taking the DBpedia dataset as an example yields that



**Figure 2:** Outgoing Links From the DBpedia dataset.

only half of the properties in this dataset are dereferenceable. Fig. 2 depicts the distribution of the dereferenceable outgoing links from the DBpedia dataset. We would expect this distribution to be modelled by a power-law distribution considering the degree of DBpedia resources (the number of resources having a given number of links to external datasets). However, Fig. 2 does not clearly suggest this, which may be due to too little data or due to the fact that links from DBpedia to other datasets are created in a supervised way, whereas scale-free networks tend to represent organic and decentralised structures. We found

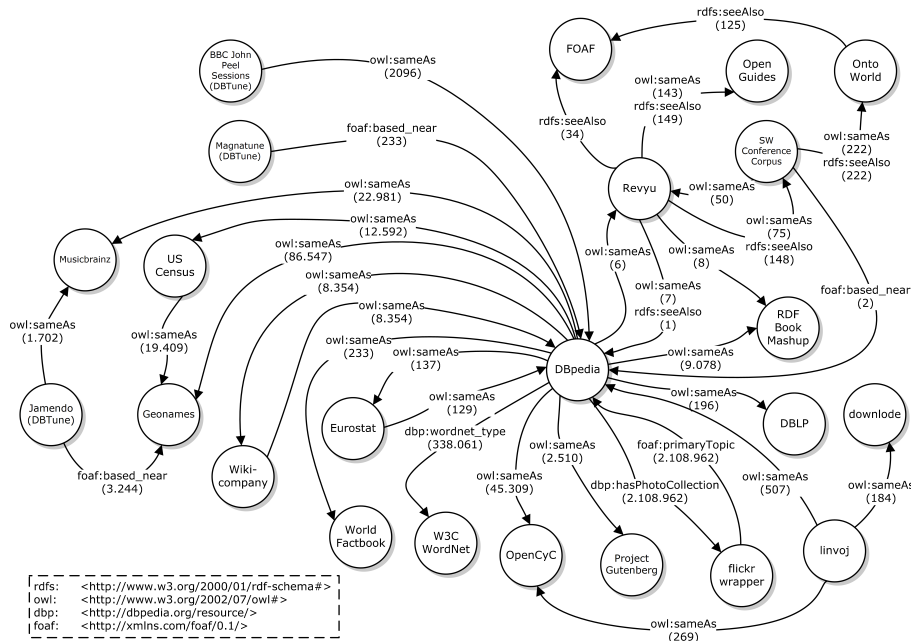
Property (Link Type)	Occurrence
<a href="http://dbpedia.org/property/hasPhotoCollection">http://dbpedia.org/property/hasPhotoCollection</a>	2.108.962
<a href="http://xmlns.com/foaf/0.1/primaryTopic">http://xmlns.com/foaf/0.1/primaryTopic</a>	2.108.962
<a href="http://dbpedia.org/property/wordnet_type">http://dbpedia.org/property/wordnet_type</a>	338.061
<a href="http://www.w3.org/2002/07/owl#sameAs">http://www.w3.org/2002/07/owl#sameAs</a>	307.645
<a href="http://xmlns.com/foaf/0.1/based_near">http://xmlns.com/foaf/0.1/based_near</a>	3.479
<a href="http://www.w3.org/2000/01/rdf-schema#seeAlso">http://www.w3.org/2000/01/rdf-schema#seeAlso</a>	679

**Table 2:** Overall Occurrence of Link Types in the LOD dataset.

only a limited number of dereferenceable links in the LOD dataset (Table 2); this distribution is biased towards the DBpedia dataset and the flickr wrapper, however. In case of the single-point-of-access datasets, we found that mainly one



or two interlinking properties are in use (Fig 3). The reason can be seen in the way these links are usually created. Based on a certain template, the interlinks (such as `owl:sameAs`) are generated automatically. As the data model of the



**Figure 3:** Single-point-of-access Partition Interlinking.

Semantic Web is a graph the question arises if the density of the overall graph can be used to make a statement regarding the system's size. The LOD dataset is a sparse directed acyclic graph; only a few number of links (compared to the overall number of nodes) exist. Introducing links is costly. While manual added, high-quality links mainly stem from user generated metadata, the template-based generated links (cheap but semantically low-level) can be added to a greater extent.

## 4.2 Distributed Datasets

In order to analyse the partition of the LOD covering the distributed dataset, such as the FOAF-o-sphere, we need to sample it. Therefore, from a single seed URI<sup>6</sup>, approximately six million RDF triples were crawled. On its way, 97410 HTTP identifiers for persons were gathered. We analysed the resulting sampled FOAF dataset, yielding the results highlighted in Table 3.

<sup>6</sup> <http://kmi.open.ac.uk/people/tom/>

To	Interlinking Property	Occurrence
FOAF	foaf:knows (direct)	132.861
FOAF	foaf:knows+rdfs:seeAlso	539.759
Geonames	foaf:based_near	7
DBLP	owl:sameAs	14
ECS Southampton	rdfs:seeAlso	21
ECS Southampton	foaf:knows	21
DBpedia	foaf:based_near	4
DBpedia	owl:sameAs	1
RDF Book Mashup	dc:creator	12
RDF Book Mashup	owl:sameAs	4
OntoWorld	pim:participant	3
Revyu	foaf:made	142
Other LOD datasets	-	0
Total inter-FOAF links	-	672.620
Total of other links	-	229

**Table 3:** Interlinking from a sampled FOAF dataset to other datasets.

Although the intra-FOAF interlinking is high (in average, a single person is linked to 7 other persons), the interlinking between FOAF and other datasets is comparably low; some  $2 * 10^{-3}$  interlinks per described person have been found. Also, the proportion of *indirect* links from a person to another (using foaf:knows and rdfs:seeAlso) is higher than *direct* links (through a single foaf:knows).

## 5 Conclusion

We have attempted to make a step towards answering the question: *What is the size of the Semantic Web?* in this paper. Based on a syntactic and semantic analysis of the LOD dataset we believe that answers can be derived for the entire Semantic Web. We have identified two different types of datasets, namely single-point-of-access datasets (such as DBpedia), and distributed datasets (e.g. the FOAF-o-sphere). At least for the single-point-of-access datasets it seems that automatic interlinking yields a high number of semantic links, however of rather shallow quality. Our finding was that not only the number of triples is relevant, but also how the datasets both internally and externally are interlinked. Based on this observation we will further research into other types of Semantic Web data and propose a metric for gauging it, based on the quality and quantity of the semantic links. We expect similar mechanisms (for example regarding automatic

interlinking) to take place on the Semantic Web. Hence, it seems likely that the Semantic Web as a whole has similar characteristics compared to our findings in the LOD datasets. Finally we return to the initial question: *What is the size of the Semantic Web?* In a nutshell, the answer is: just as the surface of a sphere is bounded but unlimited, the Semantic Web is.

## Acknowledgement

The research leading to this paper was carried out in the “Understanding Advertising” (UAd) project<sup>7</sup>, funded by the Austrian FIT-IT Programme, and was partially supported by the European Commission under contracts FP6-027122-SALERO and FP6-027026-K-SPACE.

## References

- [Broder et al. 00] Broder A., Kumar R., Maghoul F., Raghavan P., Rajagopalan S., Stata R., Tomkins A., and Wiener J. Graph structure in the Web. *Computer Networks: The International Journal of Computer and Telecommunications Networking*, 33(1-6):309–320, 2000.
- [Gulli and Signorini 05] Gulli A. and Signorini A. The Indexable Web is More than 11.5 Billion Pages. In *WWW '05: Special interest tracks and posters of the 14th international conference on World Wide Web*, pages 902–903, 2005.
- [Ayers 07] Ayers D. Evolving the Link. *IEEE Internet Computing*, 11(3):94–96, 2007.
- [Esmaili and Abolhassani 06] Esmaili K. and Abolhassani H. A Categorization Scheme for Semantic Web Search Engines. In *4th ACS/IEEE International Conference on Computer Systems and Applications (AICCSA-06)*, Sharjah, UAE, 2006.
- [Finin et al. 05] Finin T., Ding L., Pan R., Joshi A., Kolari P., Java A., and Peng Y. Swoogle: Searching for knowledge on the Semantic Web. In *AAAI 05 (intelligent systems demo)*, 2005.
- [Ding and Finin 06] Ding L. and Finin T. Characterizing the Semantic Web on the Web. In *5th International Semantic Web Conference, ISWC 2006*, pages 242–257, 2006.
- [Tummarello et al. 07] Tummarello G., Delbru R., and Oren E. Sindice.com: Weaving the Open Linked Data. In *The Semantic Web, 6th International Semantic Web Conference, 2nd Asian Semantic Web Conference, ISWC 2007 + ASWC 2007*, pages 552–565, 2007.
- [Ding et al. 05] Ding L., Zhou L., Finin T., and Joshi A. How the Semantic Web is Being Used: An Analysis of FOAF Documents. In *38th International Conference on System Sciences*, 2005.
- [Wang 06] Wang T. D. Gauging Ontologies and Schemas by Numbers. In *4th International Workshop on Evaluation of Ontologies for the Web (EON2006)*, 2006.
- [Hausenblas et al. 07] Hausenblas M., Slany W., and Ayers D. A Performance and Scalability Metric for Virtual RDF Graphs. In *3rd Workshop on Scripting for the Semantic Web (SFSW07)*, Innsbruck, Austria, 2007.
- [Völkel et al. 06] Völkel M., Krötzsch M., Vrandečić D., Haller H., and Studer R. Semantic Wikipedia. In *15th International Conference on World Wide Web, WWW 2006*, pages 585–594, 2006.
- [Bizer et al. 07] Bizer C., Heath T., Ayers D., and Raimond Y. Interlinking Open Data on the Web (Poster). In *4th European Semantic Web Conference (ESWC2007)*, pages 802–815, 2007.

<sup>7</sup> <http://www.sembase.at/index.php/UAd>

## The Utility of Specific Relations in Ontologies for Image Archives<sup>1</sup>

**Andreas Walter**

(FZI Research Center for Information Technologies Haid-und-Neu-Strae 10-14,  
76131 Karlsruhe, Germany, awalter@fzi.de)

**Gábor Nagypál**

(disy Informationssysteme GmbH, Erbprinzenstr. 4-12, Eingang B, 76133  
Karlsruhe, Germany, nagypal@disy.net)

**Abstract:** The ImageNotion methodology and tools [Walter and Nagypal (2007)] support collaborative ontology creation and semantic image annotation in one integrated web-based application. In previous evaluations, we received very positive feedback from our users about this visual methodology. Users found the approach understandable and useful. So far, the ImageNotion methodology supports for the sake of simplicity only three kinds of relations: broader, narrower and unnamed relations. We were interested, however, whether users would find it useful to have more kinds of relations, which would also make our ontology more expressive. We therefore evaluated in an online survey what users think of this issue. The evaluation was based on the publicly available online prototype of the system. We could attract more than one hundred participants. This paper analyzes the results of this survey.

**Key Words:** semantic image annotation, case study, relations, ImageNotion

**Category:** H.3.3, H.3.5, H.5.1, H.5.2

### 1 Introduction

State-of-the-art popular image archives on the Web, such as Flickr [Flickr (2007)] or Riya [Riya (2007)] still use textual annotations (tags) and full-text search. This simple and efficient approach has some problems with synonyms and homonyms, with tags in different languages, and with the lack of relations among tags. Semantic technologies solve these issues and thus may improve search results and may simplify user navigation in an image archive. Semantic technologies have the drawback, however, that they need ontologies and semantic annotations: both are complicated and resource intensive to develop with state-of-the-art methodologies and tools.

In the ImageNotion methodology and tools developed in the IMAGINATION EU project, our aim is to exploit semantic technologies in image archives. The motivation is to improve the quality of search results and to make navigation in the image archive easier. At the same time we would like to keep the methodology

---

<sup>1</sup> This work was co-funded by the European Commission within the project IMAGINATION. Also, we would like to thank all participants of our online evaluation.

and the application as simple as possible. The ultimate goal is that even ontology creation and semantic annotation should be usable for non-ontology-experts. These considerations resulted in the ImageNotion web application that supports the collaborative creation of ontologies integrated into the process of semantic annotation. The same web application also provides semantic search over images, and visual navigation among images. The visual methodology that guides the creation of the ontology, the creation of semantic image annotation and also the search process is termed ImageNotion, as well. For the sake of simplicity, the ImageNotion methodology currently supports only three different types of relations, namely broader, narrower and unnamed relations. This equals to the relations usually offered in thesauri [Brickley and Miles (2005)]. We believe, however, that one of the main difference between a full-fledged ontology and a thesaurus is the availability of rich set of non-linguistic relations. Therefore, we were interested whether there is an end-user need for more types of relations that would motivate the extension of our methodology and tools. To answer this question, we executed an online survey with more than one hundred participants. In this paper, we will present and analyze the results of the survey. We believe that our analysis is interesting for any kind of information system that employs ontologies with a richer set of relations than is available in thesauri.

The paper is structured as follows: Section 2 gives a brief overview on the ImageNotion methodology. Section 3 shows related methodologies which support collaborative ontology generation. Section 4 contains the results of our survey and Section 5 concludes.

## **2 The ImageNotion application for semantic image annotation**

The ImageNotion methodology is a visual methodology which supports collaborative, work-integrated ontology development, collaborative semantic annotation and visual semantic search. The main driving force for the methodology was that it should be usable and understandable also for non-ontology-experts, such as employees of image agencies. We give here only a very brief overview of the methodology because we have already reported on various aspects in other publications. For further details please refer to [Braun et al. (2007)], [Walter and Nagypal (2007)], [Walter and Nagypal (2008)]. The ImageNotion web application that implements the ImageNotion methodology is publicly available at [www.imagenotion.com](http://www.imagenotion.com).

### **2.1 The ImageNotion methodology: visual and collaborative ontology development**

The basis of our ontology formalism is a concept we call *imagenotion* [Walter and Nagypal (2007)]. An imagenotion (formed from the words image and notion)

graphically represents a semantic notion through an image. Furthermore, similarly to many existing ontology formalisms, it is possible to associate descriptive information with an imagenotion, such as a label text and alternative labels in different languages, date information and links to web pages to give further background information about semantic elements. In addition, it is possible to create relations among imagenotions.

The aim of the ImageNotion methodology is to guide the process of creating an ontology of imagenotions. The methodology is based on the ontology maturing process model [Braun et al. (2007)] and therefore consists of three different steps. Step 1 is the creation of new imagenotions, Step 2 is the consolidation of imagenotions in communities and Step 3 is the formalization of imagenotions by defining creation rules (such as naming conventions) and relations. Imagenotions from each maturing grade may be used for semantic image annotations. In the ImageNotion application, imagenotions are used for the semantic image annotation instead of textual tags as in traditional image archives.

## 2.2 ImageNotion in action – a concrete example

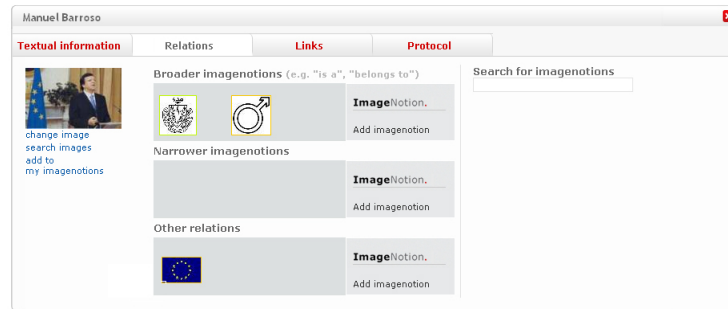
To illustrate the creation of imagenotions and relations using the ImageNotion methodology, we give an example for creating the semantic element representing “Manuel Barroso”, the current president of the European Commission.

**Figure 1:** Descriptive information in the imagenotion for Manuel Barroso

Our content provider in the IMAGINATION project are a small group of six image editors. They know each other and work collaboratively for the generation of semantic image annotations. To annotate images showing Manuel Barroso, one of them has created the imagenotion “Manuel Barroso” and selected an image showing him as representing image (see Fig. 1). In addition, she gave this

imagenotion a main label. Some other member of the group added an alternative label text, the full name of Barroso which is “José Manuel Durão Barroso”, and another member added his birthday, 1956-03-23. All in all, they created and matured the descriptive and visual information of this imagenotion.

### 2.3 Usage of ImageNotion for the creation of relations



**Figure 2:** Relations in the imagenotion for Manuel Barroso

The ImageNotion methodology currently provides three types of relations: broader, narrower and unnamed relations. Our content providers have created the broader relations to the existing imagenotions “president” (to state that Barroso is a president), “male” (to state that Barroso is male) and the unnamed relation to “EU Commission” (to state that Barroso has something to do with the EU Commission) (see Fig. 2). More specific relations such as “works for” could now help to make more precise statements and thus to improve the quality of the background knowledge. Our guiding principle for the ImageNotion methodology was, however, that we do not add any advanced feature until it is not explicitly required and useful for our users. Therefore, we evaluated, whether those more specific relations are understandable for our users, and whether they are considered as useful.

## 3 Related Work

In this section, we present related applications, which also support collaborative ontology creation and/or semantic annotation.

SOBOLEO [Zacharias and Braun (2007)] is a system for web-based collaborative engineering of ontologies and annotation of web resources. Similar to

ImageNotion, this system only supports broader, narrower and unnamed relations, too. The OntoGame system Siorpaes and Hepp (2007) implements an online game. Two players can play and collaboratively map contents in Wikipedia articles to instances and concepts. The mapping or definition of relations is currently not supported in this game, but it would be theoretically possible. Semantic Wikis [Völkel et al. (2006)] allow for the generation of relations. Such a method would be a possible way to extend the ImageNotion methodology if our survey results show that specific relations are required by our users.

## 4 Evaluation

In this section, we present the results of our evaluation. In the following tables, we show only the top ten results. E.g. for the creation of descriptive information, we counted only the ten most stated values.

### 4.1 Parts of our evaluation

Our evaluation consisted of three different parts. In the first part, we collected the main label and the alternative labels that our users would use to describe the current EU president Manuel Barroso. In the second part, we were interested, with which other semantic elements (imagenotions) our users connect Manuel Barroso, and which relation names they use. For these parts we chose Barroso because he is generally known in European countries. In addition, we assume that users would create similar relations for other famous persons such as kings, politicians, military people or celebrities. Since a lot of images are about such kinds of people in popular image archives, we think that our evaluation results are generally interesting for all kinds of image archives having images on historically relevant people. In the final part of the evaluation, we asked our users whether they find relations important for semantic search in general.

### 4.2 Setup of the evaluation

We created an online survey and sent email invitations to mailing lists of currently running EU projects (such as Theseus, Mature-IP and IMAGINATION), to German and French image agencies, professional image searchers, historical archives, universities and companies. Altogether, we reached over 1.000 recipients with our email invitations. 137 users accepted the invitation and participated in our survey. In addition, we executed the same online survey during a workshop of the IMAGINATION project in February in Rome. There, three groups of six people have participated. The groups were recruited from different communities: from Wikipedia users, from employees of French image agencies and from Italian history students. Altogether, 155 participants filled out our online survey.



Label	Percent
politician	14
Barroso	11
Jose Manuel Durao Barroso	11
European Commission	9
Manuel Barroso	7
baroso	2
portugal	5
person	5
EU commission president	5
EU president	4

**Figure 3:** Top ten: alternative label

Label	Percent
Manuel Barroso	33
Barroso	28
politician	17
EU	2
Jose Manuel Barroso	2
Jose Manuel Durao Barroso	2
Barroso EU president 2007	1
barroso, italy	1
President	1
Emanuell Barroso	1

**Figure 4:** Top ten: label text

### 4.3 Evaluation results

In imagenotions, textual information consists of a main label and of the alternative labels. For part I of our evaluation, the task of our users was to enter textual information for Manuel Barroso. Therefore, we asked them how they would search for the “current president of the EU” using tags. Table 4 shows, that the most frequently mentioned labels were two different versions of his name: “Manuel Barroso” and “Barroso”. In addition, further version of his name and his profession “politician” were entered. (Table 3). For the alternative label, most people chose “politician”. In terms of semantics, this may already be seen as specifying a semantic element. “Barroso” was the second most frequent alternative label, while on the third place we got the full name of Manuel Barroso, “Jose Manuel Durao Barroso”. I.e., the mostly used tags for searching for Manuel Barroso are his name and his profession, followed by different spellings of the name and finally semantic elements such as “EU” or “person”. This is a very motivating result for us, because it shows that people in general not only think in terms of tags but also consider semantically relevant aspects. This might motivate users to use semantic elements instead of tags to improve their search results in an image archive based on semantic technologies.

In the second part of our evaluation, we first were interested, how the evaluation participants create unnamed relations to imagenotions (i.e. semantic elements) that they see as somehow related to Manuel Barroso. Most people stated the birth place of Barroso as the most important related semantic element (see Table 5). Also, they stated a lot of unnamed relations to his profession such as “Politician”, “EU”, or “President”.

In the second part of our survey we evaluated whether users would like to create other relations than the broader, narrower and unnamed relations cur-

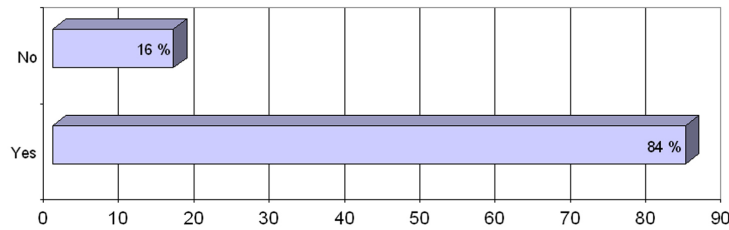
Imagenotion	Percent
Portugal	16
EU commission	14
Politician	14
President	9
EU	6
European commission	6
man	4
person	4
European Union	3
Italy	3

**Figure 5:** Top ten imagenotions

Relation	Percent
is president of	24
works for	8
has nationality	6
is born in	6
is member of	6
has position	4
is head of	4
lives in	4
has meeting with	2
has profession	2

**Figure 6:** Top ten for relations

rently supported by our system. Therefore, we asked the participants, what kind of named relations they would use for the relations they created. We merged similar suggestions such as “is president”, “president of” and “president for” together to one relation such as “is president of”. Even after this merging process different relations emerged, such as “is president of”, “works for” and “has nationality” (see Table 6).

**Figure 7:** Are relations important for the refinement of image search requests?

Finally, we asked our users whether they think that relations are important to refine semantic search requests and whether they would use them to refine search requests in an image archive. With 84 percent, most of the participants thought that relations are important for semantic image search (see Fig. 7).

## 5 Conclusions

In our online survey with more than hundred participants, we were interested, how users like and understand the idea of having relations between imagenotions

(i.e. semantic elements). In addition, we were interested, whether users request for other types of relations than normally used in thesauri (broader, narrower and unnamed relations).

Indeed, our users not only created many relations to imagenotions but they also requested and created named relations that could help to refine semantic image search requests to very powerful search requests such as “all images from persons born in Portugal who work for the EU commission”. Moreover, most participants stated that relations are important for the refinement of search requests in an image archive based on semantic technologies. Based on the results of our survey, we will create different prototypical implementations of user interfaces that could support users creating and using relations in an image archive. Then, we will evaluate which of these interfaces our users like the most.

## References

- [Braun et al., 2007] Braun, Simone ; Schmidt, Andreas ; Walter, Andreas ; Nagypal, Gabor ; Zacharias, Valentin: Ontology Maturing: a Collaborative Web 2.0 Approach to Ontology Engineering. In: *Proceedings of the Workshop on Social and Collaborative Construction of Structured Knowledge at the 16th International World Wide Web Conference (WWW 07), Banff, Canada, 2007*
- [Brickley and Miles, 2005] Brickley, Dan ; Miles, Alistair: SKOS Core Vocabulary Specification / W3C. URL <http://www.w3.org/TR/2005/WD-swbp-skos-core-spec-20051102>, November 2005. – W3C Working Draft
- [Flickr, 2007] Flickr: *Welcome to Flickr - Photo Sharing*. <http://www.flickr.com/>. 2007. – (accessed 2008-04-14)
- [Riya, 2007] Riya: *Riya - Visual search*. <http://www.riya.com/>. 2007. – (accessed 2008-04-15)
- [Siorpaes and Hepp, 2007] Siorpaes, K. ; Hepp, M.: OntoGame: Towards Overcoming the Incentive Bottleneck in Ontology Building. In: *Proceedings of the 3rd International IFIP Workshop On Semantic Web & Web Semantics (SWWS '07) co-located with OTM Federated Conferences. Springer LNCS: Vilamoura, Portugal, November 29-30, 2007, 2007*
- [Völkel et al., 2006] Völkel, Max ; Krötzsch, Markus ; Vrandečić, Denny ; Haller, Heiko ; Studer, Rudi: Semantic Wikipedia. In: *Proceedings of the 15th international conference on World Wide Web (WWW'06)*, ACM Press, 2006, pp. 585–594. – ISBN 1-59593-323-9
- [Walter and Nagypal, 2007] Walter, Andreas ; Nagypal, Gabor: ImageNotion - Methodology, Tool Support and Evaluation. In: *GADA/DOA/CoopIS/ODBASE 2007 Confederated International Conferences DOA, CoopIS and ODBASE, Proceedings, LNCS. Springer, 2007*
- [Walter and Nagypal, 2008] Walter, Andreas ; Nagypal, Gabor: EFFICIENT INTEGRATION OF SEMANTIC TECHNOLOGIES FOR PROFESSIONAL IMAGE ANNOTATION AND SEARCH. In: *Proc. of the IADIS International Conference e-Society, Portugal, 8-12. April 2008, IADIS-press, 2008*
- [Zacharias and Braun, 2007] Zacharias, Valentin ; Braun, Simone: SOBOLEO - Social Bookmarking and Lightweight Engineering of Ontologies. In: *Proceedings of the Workshop on Social and Collaborative Construction of Structured Knowledge at 16th International World Wide Web Conference (WWW2007), 2007*

# **Semantic Web Applications in- and outside the Semantic Web**

**Carola Carstens**

(German Institute for International Educational Research  
Frankfurt, Germany  
carstens@dipf.de)

**Abstract:** This article deals with the question which criteria make up a Semantic Web application. It will be shown that not all applications that make use of Semantic Web technologies actively contribute to the realization of the Semantic Web vision. In order to illustrate this, several exemplary applications are presented. They employ Semantic Web technologies mainly for goals that are rather independent from the Semantic Web vision. Nevertheless, the development of such applications may in the long run even promote the evolution of the Semantic Web, as will be explained in the course of this article.

**Keywords:** Semantic Web, Semantic Web applications, Semantic Web technologies, Semantic Web vision

**Categories:** H.2.4, H.3.4, H.3.5, H.4.0

## **1 Introduction**

In the context of Semantic Web research, new applications that are based on Semantic Web technologies are constantly being developed. Trying to get an overview of these manifold applications, the question may arise which criteria actually make up a Semantic Web application. Is every application that makes use of Semantic Web technologies automatically to be considered as a Semantic Web application? Or should the term Semantic Web application only be employed for applications that actively contribute to the realization of the Semantic Web vision?

In the remainder of this paper the usefulness of this differentiation will be illustrated by the presentation of several selected projects that all make use of Semantic Web technologies while their contribution to the Semantic Web is rather subordinate. Nevertheless, the development of such applications may in the long run even promote the evolution of the original Semantic Web goals, as will be explained in the course of this article.

The remainder of this article is structured as follows. In section 2 the Semantic Web vision as articulated by Tim Berners-Lee is described, which leads to the identification of several goals that contribute to its realization. The following section outlines how Semantic Web technologies can be used for supporting both Semantic Web goals and Semantic Web independent goals (section 3). The pursuit of these latter aims with the help of Semantic Web technologies is further illustrated by several exemplary applications. The following section discusses in how far these applications actually contribute to the Semantic Web (section 4), which leads to a final conclusion in section 5.

## 2 The Semantic Web Vision and its Goals

The Semantic Web vision was first articulated by the inventor of the current web, Tim Berners-Lee, in [Berners-Lee et al. 2001]. He describes it as “an extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation” [Berners-Lee et al. 2001]. The extension refers to semantic annotations that have to be added to information in the current web in order to create the Semantic Web. If these annotations are based on an ontologically structured vocabulary, machines that can interpret these formally defined knowledge structures will be able to aggregate and semantically integrate data from different information sources in the Semantic Web.

Moreover, new knowledge can be inferred by applying ontological inference rules to information that is annotated with ontology vocabulary. This is what Berners-Lee states in the following: “More advanced applications will use ontologies to relate the information on a page to the associated knowledge structures and inference rules” [Berners-Lee et al. 2001].

Furthermore, reasoning mechanisms may infer new relationships between data objects. This process can be considered as data enrichment, as mentioned in a more recent article about the nature of the Semantic Web: “The Semantic Web we aspire to makes substantial reuse of existing ontologies and data. It’s a linked information space in which data is being enriched and added. It lets users engage in the sort of serendipitous reuse and discovery of related information [...]” [Shadbolt et al. 2006]. This quotation also stresses the interlinking of distributed information sources with the aim of discovering source crossing relationships and reusing semantically connected information.

From these statements, the following interim goals that contribute to the Semantic Web vision can be deduced: formal knowledge representation, publication of semantically annotated data, linking of distributed information sources, reuse of existing information sources and inferencing on structured data.

Nevertheless, the Semantic Web currently seems to be evolving only slowly. This might be due to the fact that publishing Semantic Web data does not directly bring short-term benefit to the publisher. Therefore publishing semantically annotated data on the one hand still means being one step ahead of the times as long as only few Semantic Web applications exist that are able to aggregate and semantically interpret this data. On the other hand many applications that are aimed at processing Semantic Web data do not overcome their prototypical state because of the current lack of data in the Semantic Web. This seems to be a vicious circle, paralyzing the prosperity of the Semantic Web.

## 3 Use of Semantic Web Technologies for different Goals

In the context of the German Education Portal<sup>1</sup>, a specialized information service of the German Institute for International Educational Research, a thematic research on the potential use of Semantic Web technologies was conducted. In order to assess

---

<sup>[1]</sup> [http://www.fachportal-paedagogik.de/start\\_e.html](http://www.fachportal-paedagogik.de/start_e.html)

how the implementation of Semantic Web technologies can create a surplus value for specialized information services, their use in several Semantic Web technology based applications was analysed. This research did on the one hand illustrate the use of the technologies for the Semantic Web goals defined in section 2, as will be described in subsection 3.1. On the other hand, four more goals were identified that can be achieved by the use of Semantic Web technologies. While not actively contributing to the realization of the Semantic Web vision, they might nevertheless create a surplus value in the context of an information service, namely in the fields of semantic visualisation and navigation, data integration, information retrieval support and personalization. These goals will be presented in subsection 3.2, along with several illustrative example applications.

### 3.1 Semantic Web Goals

In section 2 the Semantic Web goals of formal knowledge representation, publication of semantically annotated data, linking of distributed information sources, reuse of existing sources and inferencing on structured data were identified. Their realization with the help of Semantic Web technologies is briefly described in the following.

**Formal knowledge representation:** The backbone of a Semantic Web application should be formally defined knowledge in the form of an ontology. For this purpose, the standards RDF, RDFS and OWL have been developed.

**Publication of semantically annotated data:** Applications actively support the Semantic Web if they publish data that is semantically annotated with ontologically structured vocabulary. Only if this is the case, the data can be reused by other applications and contribute to the growth of the Semantic Web. The publication may take the form of websites containing machine readable RDF data. Another method is the provision of downloadable RDF data dumps. Alternatively, access through querying endpoints is also a solution for enabling data reuse.

**Linking of distributed information sources:** One of the main aims of the Semantic Web is the semantic interlinking of distributed information sources. If data on the web is published in RDF format, every resource is described by an URI, which allows the easy reference to it by other websites or applications. This strategy has been described in detail in the linked data approach by Berners-Lee [Berners-Lee 2006] and Bizer et al [Bizer et al. 2007]. In brief, every imaginable resource, be it a document or a real world object such as a person, a place or a book, should be identified by a URI so that it is unambiguously identifiable, enabling other applications to refer to it.

**Reuse of existing information sources:** In their description of the Semantic Web vision, Shadbolt et al. propagate the reuse of already existing resources that are based on Semantic Web standards [Shadbolt et al. 2006]. The linking to already existing RDF knowledge bases such as DBpedia [Auer et al. 2007] as well as the reuse of foreign Semantic Web standard based ontologies fall into this category.

This means that an application's contribution to the Semantic Web is most effective if it does not only make its own data reusable by others, but also fosters the reuse of existing Semantic Web resources. This way the evolution of standard URIs for certain resources and standard ontologies for certain domains can be promoted,

eventually simplifying the interlinking of more and more knowledge bases in the long term.

**Inferencing on structured data:** One of the main objectives of the Semantic Web is the generation of added value through reasoning on structured data. For this purpose, explicit inference rules may be defined that deduct additional relationships between ontological objects, thus contributing to the enrichment of a knowledge base.

### 3.2 Other Goals

As already denoted, Semantic Web technologies can furthermore be used for the realization of functions that are rather independent from the Semantic Web vision, nonetheless worth considering to be implemented in an information service. In the following, these goals will be briefly described and illustrated by several projects and use cases. These exemplary applications are classified by the respective main aim they pursue by means of Semantic Web technologies.

**Semantic visualisation and navigation:** Ontological structures are well suited for the visualisation of relationships between data objects in an application. In semantic portals the whole hyperlink structure may even be generated on the basis of an underlying ontology. In order to generate different views onto a knowledge base inferencing facilities can be applied.

For example, both the Ontoframe [Jung et al. 2007] and the SEMPort [Sah et al. 2007] projects aim at creating information portals that are based on Semantic Web technologies. The SEMPort use case describes how an ontological knowledge base serves as a basis for the generation of semantic navigation structures that allow browsing the ontological relationships in the portal. In the Ontoframe project, inferencing on an ontological knowledge base is the basis for visualised knowledge analysis services on a researcher platform.

**Information retrieval support:** In contrast to data retrieval for ontological objects, information retrieval deals with the search in less structured resources, such as natural language documents. Therefore queries have to be matched to the document or indexing language. This process can well be supported by ontologies and ontology based inferencing mechanisms which enable the implementation of functions such as query expansion and query refinement as well as the identification of semantically related recommendation objects.

This is the case in the NPBib Search engine [Sack 2005] and the MultimediaN E-Culture demonstrator project [Schreiber et al. 2006] where Semantic Web technologies are mainly used for supporting information retrieval processes. The NPBibSearch engine enhances the search on a bibliographic database by generating ontology based search suggestions such as narrower and broader search terms as well as cross references to related terms. In the MultimediaN E-Culture demonstrator project, semantic clustering techniques are applied to an ontology for cultural heritage resources. This clustering is based on ontological graph traversal. As a result, the users can be presented with terms that are semantically related to their original query terms.

**Data integration:** Semantic Web technologies are often used for data integration purposes. In this case ontologies serve as a semantic layer that integrates data from

heterogeneously structured data sources. Throughout this integration process, inferencing facilities may be used, for example for identifying identical objects.

The integration may either be physical or virtual. In the latter case, data sources are not transformed, but rather mapped to an ontology based data model in RDFS or OWL. For this purpose mapping languages such as D2RQ may be used that make it possible to treat relational databases as virtual RDF graphs, thus avoiding the duplication of data in RDF format [Bizer and Seaborne 2004]. Consequently, queries to the ontological model can deliver integrated results from the legacy databases. This strategy is considered as virtual data integration. It differs from the Semantic Web goal of interlinking distributed sources in so far as in the latter case data sources have to be expressed in the RDF standard. This is not the case when legal data sources are virtually integrated via an ontology.

Physical data integration with ontologies takes place if data from legacy databases is transformed into Semantic Web standards. In many cases this means that data is duplicated and imported into an integrative ontology based system.

Semantic Web technologies are used for the integration of heterogeneous data sources in the DOPE project [Stuckenschmidt et al. 2004], the use case from Traditional Chinese Medicine [Chen et al. 2006], the Neurosciences project [Lam et al. 2006] and the MuseumFinland application [Hyvönen et al. 2005], for example. While the two latter process the integration physically, a virtual integration is realized in the DOPE project and in the use case from Traditional Chinese Medicine.

**Personalization:** With the help of Semantic Web technologies, personalized views and recommendations can easily be generated by making use of inference rules. For this purpose, user profile information is usually stored in the knowledge base along with the application's content data. Adaptivity rules can then be applied to infer user dependent recommendations or data views.

In a personalized information system, a user whose special interest lies in Semantic Web technologies may be alerted whenever an article dealing with the topics *RDF* or *OWL* comes out. The relevance of such an article can be inferred from the ontology where *RDF* and *OWL* would be classified as subclasses of the class *Semantic Web technologies*.

Ontological inference mechanisms are the basis for personalized services in the CHIP project [Aroyo et al. 2007], the Personal Publication Reader [Baumgartner et al. 2005] and the iFanzly project [Bellekens et al. 2007]. These inference based services range from personal recommendations of museum collection items in the CHIP project over personalized views onto distributed research data in the Personal Publication Reader to personalized TV programme recommendations in the iFanzly project context.

## 4 Discussion of Applications and Goals

Although the applications presented in the preceding section are ordered by the main goals they pursue with the help of Semantic Web technologies, it has to be stated that none of the applications realizes only a single goal. For this reason the following table [Tab. 1] gives a more comprehensive overview of both the Semantic Web goals and the Semantic Web independent goals, outlined in section 3, that are being realized



with Semantic Web technologies in the particular applications<sup>2</sup>. Their respective main goals are marked grey in the table.

Project	Semantic Web goals					Other goals			
	FKR	PSAD	LDIS	REIS	IOSD	SVN	IRS	DI	P
Neurosciences	X	X			X			X	
MuseumFinland	X				X	X		X	
Traditional Chinese Medicine	X				X	X	X	X	
DOPE	X			X	X	X	X	X	
SEMPort	X			X	X	X			X
Ontoframe	X				X	X	X	X	
NPBibSearch	X					X	X		
MultimediaN E-Culture demonstrator	X			X		X	X	X	
CHIP	X			X	X			X	X
Personal Publication Reader	X			X	X	X		X	X
iFanzzy	X			X	X	X	X	X	X

Table 1: Overview of applications and goals

**FKR:** Formal Knowledge Representation

**PSAD:** Publication of Semantically Annotated Data

**LDIS:** Linking of Distributed Information Sources

**REIS:** Reuse of Existing Information Sources

**IOSD:** Inferencing on Structured Data

**SVN:** Semantic Visualisation and Navigation

**IRS:** Information Retrieval Support

**DI:** Data Integration

**P:** Personalization

Although the focus of the applications presented above clearly lies on the pursuit of Semantic Web independent goals, the table shows that several Semantic Web goals are nonetheless prevalent in most applications, namely formal knowledge representation (FKR), the reuse of existing information sources (REIS) and inferencing on structured data (IOSD). This can be explained by the fact that the use of Semantic Web technologies is closely linked to the formal representation of knowledge (FKR) that is the basis of all applications listed here, each of them relying on an ontological backbone. As far as the reuse of existing information sources (REIS) is concerned, it can be stated that the reuse of carefully modeled standard ontologies is also attractive for internal use, which is a step towards the semantic interoperability that is envisioned in the Semantic Web. As already partly denoted in subsection 3.2, inferencing on structured data (IOSD) is not only applicable to the Semantic Web. By contrast, it can also be used to support each of the delineated Semantic Web independent goals, which is the case for most of the applications listed above.

Nevertheless, an active contribution to the Semantic Web requires both the publication of semantically annotated data (PSAD) and its interlinking with other sources (LDIS). With the exception of the Neurosciences project, none of the applications presented here supports these Semantic Web goals.

<sup>[2]</sup> As the main interest in this article lies in the use of Semantic Web technologies, it is not stated if one of the goals is realized with the help of traditional technologies.

With regard to the selection criteria for the presented applications, it has to be stressed that they in the first instance serve as illustrative examples for the pursuit of Semantic Web independent goals. Therefore, they cannot be considered as representative for the high amount of Semantic Web applications that is existing. Of course, there are also many applications that primarily foster Semantic Web goals, especially the publication of semantically annotated data (PSAD) and the linking of distributed information sources (LDIS). Good examples are the Semantic Media Wiki [Krötzsch et al. 2007] and the Revyu project [Heath and Motta 2007], to name just two. Nevertheless, they are not in the focus of this paper.

## 5 Conclusion

This paper has shown that Semantic Web technologies are not only useful for the realization of the Semantic Web vision, but that they can also be successfully implemented for the realization of functions that are rather independent from this vision. Nevertheless, it has to be stated that the mere use of Semantic Web technologies for formally defining knowledge and inferencing on structured data is very similar to traditional artificial intelligence methods. Apart from that, an application aimed at contributing to the Semantic Web should also realize the publication and interlinking of the Semantic Web standards based data it relies on.

Returning to the initial question if all applications that make use of Semantic Web technologies are to be considered as Semantic Web applications, we can infer that we should differentiate between (1) applications that are solely based on Semantic Web technologies and (2) those who, in addition, actively contribute to the Semantic Web vision. The latter implies publishing the application data on the Semantic Web, thereby making it available for interlinking and reuse.

Accordingly, the applications presented in this paper would fall into the first category. Nevertheless, the development of such applications that use Semantic Web technologies primarily for internal goals, may ultimately help the Semantic Web vision turn into reality. As stated in section 2, one of the main obstacles that currently still impede the growth of the Semantic Web is the uncertainty if the active contribution to this vision is worthwhile. It cannot be guaranteed that the publication of standardized data and its interlinking with other sources will directly bring benefit to the data owner. However, if developers and data owners perceive a personal advantage in making use of Semantic Web technologies for pursuing their own goals, such as the ones described in subsection 3.2, a vast amount of data will be prepared for the Semantic Web. If this is the case, the most labour intensive task for participating in the Semantic Web is already realized. Besides that, the practice of reusing already existing resources such as evolving standard ontologies and RDF data sets will simplify the potential interlinking of knowledge bases with the Semantic Web.

Finally, the use of Semantic Web technologies for internal goals may foster the technology use in general, this way creating a wide base for the envisioned semantic interlinking of more and more data sources on the Semantic Web.

## References

- [Aroyo et al. 2007] Aroyo, L., Stash, N., Wang, Y., Gorgels, P., Rutledge, L.: "CHIP Demonstrator: Semantics-driven Recommendations and Museum Tour Generation". Proc. ISWC/ASWC 2007, Busan, Korea, 879-886.
- [Auer et al. 2007] Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., Ives, Z.: "DBpedia: A Nucleus for a Web of Open Data". Proc. ISWC/ASWC 2007, Busan, Korea, 722-735.
- [Baumgartner et al. 2005] Baumgartner, R., Henze, N., Herzog, M.: "The Personal Publication Reader: Illustrating Web Data Extraction, Personalization and Reasoning for the Semantic Web". Proc. ESWC 2005, Heraklion, Greece, 515-530.
- [Bellekens et al. 2007] Bellekens, P., Aroyo, L., Houben, G., Kaptein, A., van der Sluijs, K.: "Semantics-Based Framework for Personalized Access to TV Content: The iFanzo Use Case". Proc. ISWC/ASWC 2007, Busan, Korea, 887-894.
- [Berners-Lee 2006] Berners-Lee, T.: "Linked Data", from <http://www.w3.org/DesignIssues/LinkedData.html>. Last access: 30.06.2008.
- [Berners-Lee et al. 2001] Berners-Lee, T., Hendler, J., Lassila, O.: "The Semantic Web". *Scientific American*, May 2001: 34-43.
- [Bizer et al. 2007] Bizer, C., Cyganiak, R., Heath, T.: "How to Publish Linked Data on the Web", from <http://sites.wiwi.fu-berlin.de/suhl/bizer/pub/LinkedDataTutorial/>. Last access: 30.06.2008.
- [Bizer and Seaborne 2004] Bizer, C., Seaborne, A.: "D2RQ -Treating Non-RDF Databases as Virtual RDF Graphs" (Poster). ISWC 2004, Hiroshima, Japan.
- [Chen et al. 2006] Chen, H., Wang, Y., Wang, H., Mao, Y., Tang, J., Zhou, C., Yin, A., Wu, Z.: "Towards a Semantic Web of Relational Databases: a Practical Semantic Toolkit and an In-Use Case from Traditional Chinese Medicine". Proc. ISWC 2006, Athens, USA, 750-763.
- [Heath and Motta 2007] Heath, T., Motta, E.: "Revyu.com: A Reviewing and Rating Site for the Web of Data". Proc. ISWC/ASWC 2007, Busan, Korea, 895-902.
- [Hyvönen et al. 2005] Hyvönen, E., Mäkelä, E., Salminen, M., Valo, A., Viljanen, K., Saarela, S., Junnila, M., Kettula, S.: "MUSEUMFINLAND - Finnish Museums on the Semantic Web". *Journal of Web Semantics*, 3(2): 224-241.
- [Krötzsch et al. 2007] Krötzsch, M., Vrandečić, D.; Völkel, M.; Haller, H., Studer, R.: "Semantic Wikipedia". *Journal of Web Semantics*, 5(4): 251-261.
- [Jung et al. 2007] Jung, H., Lee, M., Sung, W., Park, D.: "Semantic Web-Based Services for Supporting Voluntary Collaboration among Researchers Using an Information Dissemination Platform". *Data Science Journal*, 6: 241-249.
- [Lam et al. 2006] Lam, H., Marenco, L., Shepherd, G., Miller, P., Cheung, K.: "Using Web Ontology Language to Integrate Heterogeneous Databases in the Neurosciences". Proc. AMIA Annual Symposium, 464-468.
- [Sack 2005] Sack, H.: "NPBibSearch: An Ontology Augmented Bibliographic Search". Proc. 2nd Italian Semantic Web Workshop, Trento, Italy.
- [Sah et al. 2007] Sah, M., Hall, W., Gibbins, N., de Roure, D.: "SEMPort - A Personalized Semantic Portal". Proc. 18th ACM Conference on Hypertext and Hypermedia, Manchester, United Kingdom, 31-32.
- [Schreiber et al. 2006] Schreiber, G., Amin, A., van Assem, M., de Boer, V., Hardman, L., Hildebrand, M., Hollink, L., Huang, Z., van Kersen, J., de Niet, M., Omelayenkko, B., van Ossenbruggen, J., Siebes, R., Taekema, J., Wielemaker, J., Wielinga, B.: "MultimediaN E-Culture Demonstrator". Proc. ISWC 2006, Athens, USA, 951-958.

[Shadbolt et al. 2006] Shadbolt, N., Hall, W., Berners-Lee, T: "The Semantic Web Revisited". IEEE Intelligent Systems, 21(3): 96-101.

[Stuckenschmidt et al. 2004] Stuckenschmidt, H., van Harmelen, F., de Waard, A., Scerri, T., Bhogal, R., van Buel, J., Crowlesmith, I., Fluit, C., Kampman, A., Broekstra, J. van Mulligen, E.: "Exploring Large Document Repositories with RDF Technology: The DOPE Project". IEEE Intelligent Systems, 19(3): 34-40.

## Collaborative Tasks using Metadata Conglomerates – The Social Part of the Semantic Desktop

**Olaf Grebner**

(SAP Research, Karlsruhe, Germany  
olaf.grebner@sap.com)

**Hannes Ebner**

(Royal Institute of Technology, Stockholm, Sweden  
hebner@csc.kth.se)

**Abstract:** This paper<sup>1</sup> presents an application that enables loose and ad-hoc task collaboration for knowledge work and explicitly integrates task metadata into the collaboration process. With the increasing availability of semantic desktop technology, knowledge workers (KWers) can organize their personal information in a formalized way, including tasks. In an organization, KWers need to collaboratively access task-related information and to collaboratively work on it. In such a scenario, today's available collaboration support applications, e.g. enterprise collaboration systems like wikis, either sacrifice end-user experience or semantic richness when dealing with structured knowledge. The presented collaborative task management (TM) application circumvents this trade-off by showing how the Kasimir TM client and the Collaborilla collaboration server interact closely. The TM client supports the KWer's personal TM and incorporates collaborative tasks. It invokes the collaboration server which centrally manages the collaborative tasks. It implements elaborated methods for metadata sharing and collaborative editing of this shared metadata. We present the detailed proposal for an underlying architecture of the application, review related work and conclude this paper by pointing out future work.

**Keywords:** Task Management, Ad-hoc Collaboration, Metadata Sharing, Collaborative Metadata Editing, Semantic Desktop

**Categories:** H.1, H.4, M.3, M.4, M.5, M.6

### 1 Introduction

People performing knowledge-intensive work, i.e., knowledge workers (KWers) like, e.g., managers and researchers, have a highly dynamic working style [Davenport, 05]. Executing knowledge-intensive work tasks is characterized by a highly dynamic work approach as well as the situation that diverse outcomes satisfy the given goal.

In today's business environment, KWers increasingly work in teams to achieve organizational goals. For this, they work together on defined tasks towards a common, task-related goal, a *collaborative task*. Often, collaborative work takes place under

---

<sup>1</sup> This work is supported by the European Union (EU) IST fund (Grant FP6-027705, project NEPOMUK, <http://nepomuk.semanticdesktop.org>) and the EU eContentplus programme (project Organic.Edunet, <http://www.organic-edunet.eu>).

time pressure and is *distributed* with respect to time and place. This leads to the need for loose and ad-hoc collaboration and corresponding task management (TM).

Key to a collaborative task is that participating KWers *share a common set of information* to enable joint work and to achieve the task goal. Thereby, the KWers need to *commonly access and collaboratively work with the information*. E.g., this comprises of task information like task status, goal and description as well as attached task resources like e.g. files, contacts and emails. Each KWer can add, remove and modify the commonly available information, e.g., attach new documents, add new participants, change the status of the task or create new subtasks. Roles grant a KWer rights on a collaborative task, e.g., the task owner can change all task information.

A recent trend is the *increasing availability of structured, semantic information* in both personal and enterprise environments. Enterprise applications organize their information in a structured way and *organizational knowledge management* initiatives leverage ontologies to create a flexible information model. With the advent of *semantic desktop technologies*, e.g., on the Nepomuk Social Semantic Desktop (SSD) [Groza, 07], *personal* information is available in a structured form by using e.g. a personal information model ontology (PIMO) [Sauermann, 07]. This enables a KWer to keep and retrieve personal knowledge in a structured way.

However, the *problem* is that today's support applications can't deal with structured knowledge in the collaborative task scenario as described above. There is no widely-adopted enterprise solution enabling KWers to share and collaboratively edit structured task information. Existing solutions have severe drawbacks with regard to metadata handling. E.g. one option is to enable task collaboration by serializing task metadata into a human-readable form and putting it onto a wiki page. However, on the wiki the semantic task structure gets lost, even in a semantic wiki many KWers have a hard time applying correct syntax in the editing process. Even if a wiki page is attached to each structured task, the collaboration there doesn't include the available structured task metadata. Another option consists of serializing a task into a machine-readable form, like e.g. RDF and putting it into a shared RDF store. This preserves task semantics, but versioning problems cannot be resolved, e.g., in case of two concurrent modifications, as no handling protocol is in place. As well, every task participant needs write access to the shared RDF store, a hard prerequisite for ad-hoc tasks and in large organizations with extensive administrative procedures.

Our *proposed solution* is to apply a *wiki principle to structured content* for enabling loose and ad-hoc collaboration for TM. We present the combination of the Kasimir personal TM client application [Grebner, 08] based on the Nepomuk SSD, with the Collaborilla collaboration server (CS) [Ebner, 07 & Collaborilla, 08]. The strength of the proposed approach is that the CS enables collaborative tasks and ad-hoc collaboration while preserving the full semantics of the collaborative task information. To ensure a high usability, the KWer interacts with the CS transparently through a TM client that she uses regularly and she is familiar with, like e.g. Kasimir. The preserved semantics enable other KWers to efficiently re-use the structured task information in their semantic desktop applications, e.g., in their favorite TM client.

The *paper structure* is as follows: First, we present the two collaborative TM application parts in detail. Second, we look at the underlying architecture. Third, we review related work. Finally, we conclude with an outlook on how to generalize the proposed infrastructure for metadata sharing among semantic desktops.

## 2 Collaborative Task Management with Metadata conglomerates

In this section, we first explain the KWer's task client and then, how the collaboration server (CS) will enable collaborative task work in conjunction with the task client.

### 2.1 Collaborative Tasks embedded into the Personal TM Client

The Kasimir TM application [Grebner, 08] is the KWer's personal TM system using semantic web technologies of the Nepomuk SSD. It's designed to address the KWer's task overflow [Kirsh, 00] by efficiently supporting information management. E.g., by using application plug-ins it doesn't require KWers to copy and paste information from different application contexts to their TM tools, being a major drawback of personal information management tools [Belotti, 02]. Kasimir organizes a KWer's task lists and tasks. A KWer can assign task metadata to each task, like e.g. due date, task description, and the KWer can prioritize tasks. The KWer can attach as well SSD metadata, i.e., personal concepts of the PIMO like e.g., participating persons, projects and topics. Another key feature is the close integration of tasks with information objects on the desktop. Kasimir allows the KWer to add in the TM application references to emails and files, e.g., task-relevant documents.

*Collaborative tasks* are integrated in the Kasimir personal TM application and thereby into the KWer's TM process. A KWer manages collaborative tasks like personal tasks within the known personal TM application, as a local version of the collaborative task is checked out from the CS. This version shows the most recent state of the collaborative task with all consolidated contributions and offers additional collaboration options compared to a personal task. The KWer can create a new collaborative task, mark an existing task as collaborative or upon synchronization, a new collaborative task triggered by a task collaborator is put into the task inbox, see below.

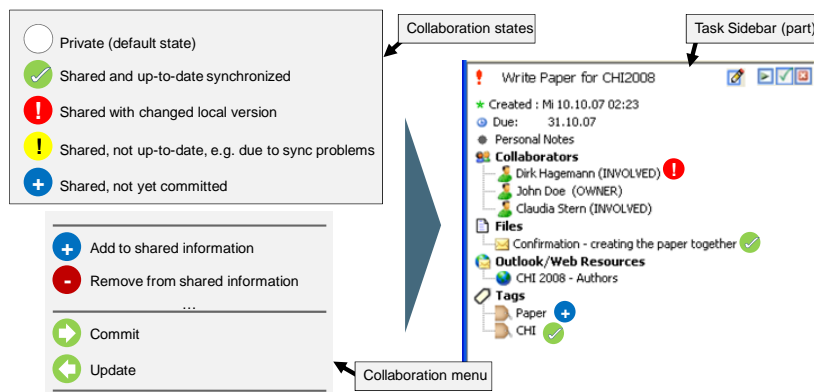


Figure 1: Collaboration status indicators & -menu & Task Sidebar (mock-up).

The KWer can add SSD metadata to the task, e.g., information like notes, emails or documents, and can prioritize this task in the context of the other personal tasks. By default, all information attached to the local version of the collaborative task is private

and it is only shared on request, i.e. by the KWer's explicit opt-in. A visual indicator shows the collaboration state for each task information object, i.e., whether it is shared and if yes along with the state of the shared information, see .

All task information is queued locally until the KWer initiates a synchronization transaction. The KWer initiates a synchronization transaction, i.e., an update of the local version by the current CS version, by selecting "Update" in the collaboration menu. The KWer can contribute to a collaborative task by committing this information to the CS by selecting "Commit", see . This way, a KWer can add or modify task information, i.e., adding a contribution to the collaborative task on the CS.

The KWer can change own contributions and commit them, the KWer's local version is the master. Contributions owned by other task collaborators can be changed as well, however the CS handles this by annotating the changes to the task collaborator's contribution and changes the local version accordingly. This procedure is transparent to the KWer, e.g., when a KWer removes a shared email from a task, the respective information object is annotated as to be deleted from the collaborative task. The KWer's next commit transaction publishes this annotation to the CS.

## 2.2 Collaboration server hosts collaborative tasks and mediates interaction

The *Collaborilla* CS caters for metadata sharing and collaborative editing of task information and thus addresses mentioned metadata sharing problems. Collaborilla implements elaborated *methods for collaboratively constructing artifacts* and building knowledge conglomerates [Ebner, 06 & Ebner, 07].

It can handle concurrent contributions, i.e., modifications. Following the wiki approach, each KWer can comment on the task metadata as well as on the task contributions of other KWers. A KWer can *modify the task without changing the original task*, including other KWer's contributions to it, as the modification is attached itself as contribution to the task. The client merges all contributions by participating KWers to present the KWer with an up-to-date task version. Thus, the KWer can modify this task conglomerate without write access to the KWer's original task contribution. Furthermore, it enables *ad-hoc collaboration* as there's *no need to grant write access to each participating KWer*. The KWer doesn't need to know in advance who will work with this task, because the contributions can be made by all participating KWers and the client filters the contributions based on the invited participants. For the KWer all these functions are available transparently in the TM client.

## 3 Architecture

In this section we show information management details of the Kasimir TM client based on the Nepomuk SSD, of the Collaborilla CS and the corresponding interaction between these components. See Figure 2 for an architecture overview.

### 3.1 Kasimir task management client

The Nepomuk SSD establishes a semantic layer on the desktop. Its core, the personal information model, is formalized using the Personal Information Model Ontology (PIMO) [Sauer mann, 07] and represents concepts like e.g. tasks, persons, projects and



possible relations among them. The model instances represent the KWer's personal information. For example, the KWer can model that the person "Claudia Stern" is related to the project "NEPOMUK" using the "works for" relation. As well, this personal information model is grounded with the desktop resources like e.g. emails or files. So-called data wrappers crawl desktop resources to create corresponding metadata items which can be referenced in the semantic layer. E.g., the Aperture data wrapper [Aperture, 05] crawls files which can be related to a task.

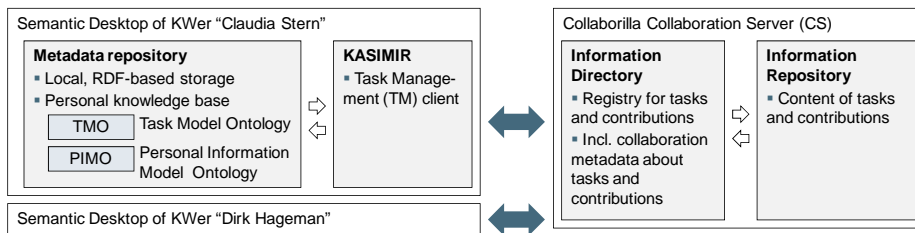


Figure 2: Architecture of Kasimir TM client and Collaborilla CS.

The Kasimir TM client handles both personal and collaborative tasks (local version) using the same task model. The representation of tasks is supported by the Nepomuk task model and the corresponding task model ontology (TMO) [Nepomuk, 06], an extension of the PIMO task concept [Sauer mann, 07]. It serves as the ontology for tasks in the presented TM prototype and uses PIMO concepts to express the relations to other concepts and information objects on the desktop.

### 3.2 Collaborilla collaboration server

The *Collaborilla CS* has two components. A central registry acts as *information directory* and keeps track of artifacts, i.e., a collaborative task, and its decentralized contributions by KWers. It provides a list of metadata elements for each task and a complementary description of each contribution in order to allow end-users to make informed decisions about what to include in their task views. Each contribution can include multiple task information elements. The information directory needs to be updated whenever someone publishes a task or a contribution to a task. The actual task metadata and task information elements are not stored in the information directory, as it only acts as resolving and referring entity to the information repository.

The central *information repository* hosts the content of the task metadata and contributions, i.e., stored in RDF files. It only contains task metadata, e.g., in case of an attached task document it contains a document reference to a shared information space. It is implemented by a remotely accessible RDF store, which can be accessed using WebDAV, and uses Subversion for storing and versioning the RDF files. Using Subversion allows as well for version control on collaborative tasks, i.e., KWers can revert to older collaborative task states in the TM client. Tasks are expressed using RDF and multiple contributions can be merged into a single contribution or task by merging their corresponding RDF graphs. This requires a conflict-free ontology (e.g. no cardinality issues). Otherwise KWers need to resolve such conflicts manually.

We focus on Collaborilla's *information directory* containing two *metadata* types:

- Registry for tasks and contributions – Correlation between the task URI and a contribution and the place where the contribution can be loaded from.
- Metadata about a published task or contribution – It helps the KWer making a decision whether to take a task contribution into account or not, e.g., for a task contribution the contributing person and time is relevant metadata.

For the *information directory*, all involved collaborative task elements have a *globally unique identifier*, a URI, which is assigned on the initial contribution of the task to the CS. This allows the task to occur in several tasks, i.e. a contribution to a task could include elements from other tasks. Contributions have no own identity separate from the task identity, as a separate identity would hinder the merging on the level of RDF graphs. A contribution can be indirectly identified by the identifier of the task and which entity, i.e. the RDF file, it is expressed in.

Some contributions have a specific status, e.g., the initial task contribution by the creating KWer will be considered as the *original* contribution distinct from all *optional* contributions. Such collaborative task dependencies trigger different TM client behavior. If a dependency is referred to as original, the referring task cannot be loaded without it, since it contains essential information, i.e., basic task information. An optional dependency is a real contribution, i.e., the task can be loaded without it. It is up to the KWer whether or not to load optional dependencies. Contributions are *not dependent on earlier contributions*, i.e., all contributions are (in principle) expressed independently of each other. However, a contribution that provides additional metadata on a task element will not be visible unless the task element itself is visible.

### 3.3 Interaction between task management client and collaboration server

Collaborilla exposes its information directory resources via REST-based web services, which the Kasimir TM client invokes. We demonstrate the proposed interaction at the example of making a task available for collaboration, i.e., creating a collaborative task from the TM client on the CS, publishing it and thus making it available for contributions. This requires a set of steps, see below. Committing and updating collaborative tasks from the TM client follow a similar scheme.

- The KWer marked a task as shared and starts the commit transaction.
- The TM client invokes the CS to publish the task information to the *information repository*. This eventually returns the URL where task information can be requested from later on.
- This location is sent together with the task's URI to the *information directory's* resolver service. It then can resolve an identifier into a real location.
- The task URI is sent together with the dependencies to the *information directory's* referrer service, which then keeps track of the original task and eventually existing contributions to it.

## 4 Related Work

For *collaborative task support*, i.e., task information sharing and collaborative work, there are state-of-the-art applications both in research as well as commercial products. These applications approach the problem from different perspectives.

*Knowledge management* applications and *collaboration support* applications provide collaborative information spaces that enable KWers to share information, like e.g. in wikis or blogs. But, this spans only unstructured information, like e.g. putting meeting notes into a wiki page that is associated to a collaborative task. *Document management* applications as part of collaboration support applications provide document sharing. Here, documents are made available to all authorized task participants, but again no collaborative work on metadata is possible. *Business Process Management (BPM)* applications have coordination and collaboration functions enabling KWers to work together on tasks in a defined sequence as described by the BPM application. However, only rudimentary functions to collaboratively edit task metadata are provided, especially in conjunction with personal information.

For *collaborative metadata management*, there are alternative approaches for CSs supporting loose collaboration. *Annotea* [Annotea, 03] is a W3C RDF standard for collaboration around web pages, using shared metadata and annotation servers. It uses constructs similar to XPointers for locating the information to be annotated. However, the Collaborilla CS works on a higher level, i.e., it does not go into the structure of a document (e.g. a task) and instead uses plain URIs. E.g., individual task contributions change the overall task only by merging the corresponding RDF graphs. This leaves a contribution valid even if the structure of the original task changes. *Semantic indices* like *Sindice* [Tummarello, 07] provide another way of keeping track of collaborative tasks and their contributions. After announcing all task-related RDF data to such an index, a search for a task's URI would return a list of RDF files mentioning it. The task could then be loaded including all contributions. However, the lack of metadata describing the contributions makes it difficult for the KWer to decide about its relevance, e.g. it's not clear e.g. in which sequence the contributions were added. Another issue is the need of indexing the RDF data before querying the index which might result in unwanted delays before updating a task. These approaches' drawbacks do not occur with a registry supporting ad-hoc collaboration like Collaborilla.

## 5 Discussion & Future Work

We presented a collaborative task management application that enables loose and ad-hoc task collaboration among KWers and that explicitly integrates metadata into this collaboration process. It brings the information sharing process in collaborative task work to the next level. KWers can collaboratively work on task information in a familiar environment while preserving the semantic structure of the task information. This enables for the collaborating KWers a better information management using their semantic desktop applications. The personal information on the semantic desktop, including tasks, provides the metadata fundament. Using the Kasimir TM client, the KWer can now share this structured knowledge with co-working KWers. The Collaborilla CS enables KWers to collaboratively work on this information. The here proposed combination of both components, i.e., leveraging the CS approach for collaborative TM, uniquely offers the KWer both recognized usability and information management efficiency. This solution's implementation is work in progress, but both contributing components Kasimir [Grebner, 08] and Collaborilla (for collaborative Context-Maps [Ebner, 07], supporting conceptual modeling and distributed discourse management) are evaluated as working well today.

In future work, the presented application and its infrastructure can be leveraged to support metadata sharing and collaborative editing between semantic desktops beyond task information. In such an electronic portfolio system, like for example already realized with Confolio [Confolio, 07], ad-hoc and non-invasive collaboration is possible for all metadata like e.g. the whole KWer's personal information on the SSD.

In the current CS version, KWers may contribute independently of each other with separate contributions to the same task. However, just as anyone is allowed to link to any page on the web (if we ignore juridical concerns), anyone will be allowed to contribute to tasks that have been published. This is a security problem for closed work groups who need to have full control over the published task conglomerates. This will be addressed with the next, planned version of the Collaborilla CS.

## References

- [Annotea, 03] Annotea, 2003, <http://www.w3.org/2001/Annotea/>.
- [Aperture, 05] Aperture, 2005, <http://aperture.sourceforge.net>.
- [Belotti, 02] Bellotti, V.; Ducheneaut, N.; Howard, M.; Smith, I.: Taskmaster: recasting email as task management. Workshop: "Redesigning Email for the 21st Century", CSCW. 2002.
- [Collaborilla, 08] Collaborilla, 2008, <http://collaborilla.conzilla.org>.
- [Confolio, 07] Confolio, 2007, <http://www.confolio.org>.
- [Davenport, 05] Davenport, T.H.: Thinking for a Living. Harvard Business School Press, Boston, MA, 2005.
- [Ebner, 06] Ebner, H.: Collaborilla - An enhancement to the Conzilla concept browser for enabling collaboration. Master's Thesis at the Department of Computer and Systems Sciences, Royal Institute of Technology (KTH), Stockholm, Sweden, 2006.
- [Ebner, 07] Ebner, H., Palmér, M., Naevé, A.: Collaborative Construction of Artifacts, Proceedings of 4th Conference on Professional Knowledge Management, Workshop Collaborative Knowledge Management (CoKM2007), Potsdam, Germany, 28-30 March 2007.
- [Grebner, 08] Grebner, O.; Ong, E. & Riss, U. V. (2008), KASIMIR - Work process embedded task management leveraging the Semantic Desktop, in Martin Bichler et al., ed., Proceedings of 'Multikonferenz Wirtschaftsinformatik' MKWI 2008, GITO-Verlag, Berlin, , pp. 715-726.
- [Groza, 07] Groza, T., Handschuh, S., Moeller, K., Grimnes, G., Sauermann, L., Minack, E., Mesnage, C., Jazayeri, M., Reif, G., Gudjonsdottir, R.: The nepomuk project - on the way to the social semantic desktop. In Proceedings of I-Semantics' 07, JUCS, (2007) 201–211.
- [Kirsh, 00] Kirsh, D.: A few thoughts on cognitive overload. *Intellectica*, 30, 19-51, 2000.
- [Nepomuk, 06] Nepomuk Project Deliverable D3.1, Task Management model, 2006, <http://nepomuk.semanticdesktop.org/xwiki/bin/view/Main1/D3-1>.
- [Sauermann, 07] Sauermann, L., van Elst, L., Dengel, A.: Pimo - a framework for representing personal information models, In Proceedings of I-Semantics' 07, JUCS (2007), 270–277.
- [Tummarello, 07] Tummarello, G., Delbru, R., Oren, E.: Sindice.com: Weaving the Open Linked Data, ISCW, 2007.

## Phrase Detectives: A Web-based Collaborative Annotation Game

**Jon Chamberlain**

(University of Essex, Colchester, UK  
jchamb@essex.ac.uk)

**Massimo Poesio**

(University of Essex, Colchester, UK and Università di Trento, Trento, Italy  
poesio@essex.ac.uk)

**Udo Kruschwitz**

(University of Essex, Colchester, UK  
udo@essex.ac.uk)

**Abstract:** Annotated corpora of the size needed for modern computational linguistics research cannot be created by small groups of hand annotators. One solution is to exploit collaborative work on the Web and one way to do this is through games like the ESP game. Applying this methodology however requires developing methods for teaching subjects the rules of the game and evaluating their contribution while maintaining the game entertainment. In addition, applying this method to linguistic annotation tasks like anaphoric annotation requires developing methods for presenting text and identifying the components of the text that need to be annotated. In this paper we present the first version of *Phrase Detectives* (<http://www.phrasedetectives.org>), to our knowledge the first game designed for collaborative linguistic annotation on the Web.

**Key Words:** Web-based games, distributed knowledge acquisition, object recognition, social networking, anaphoric annotation, user interaction, XML, Semantic Web

**Category:** H.5.2, I.2.5, I.2.6, I.2.7

### 1 Introduction

Perhaps the greatest obstacle to progress towards systems able to extract semantic information from text is the lack of semantically annotated corpora large enough to be used to train and evaluate semantic interpretation methods. Recent efforts to create resources to support large evaluation initiatives in the USA such as Automatic Context Extraction (ACE), Translingual Information Detection, Extraction and Summarization (TIDES), and GALE are beginning to change this – but just at a point when the community is beginning to realize that even the 1M word annotated corpora created in substantial efforts such as Prop-Bank [Palmer et al., 2005] and the OntoNotes initiative [Hovy et al., 2006] are likely to be too small. Unfortunately, the creation of 100M-plus corpora via hand annotation is likely to be prohibitively expensive, as already realized by the creators of

the British National Corpus [Burnard, 2000], much of whose annotation was done automatically. Such a large hand-annotation effort would be even less sensible in the case of semantic annotation tasks such as coreference or wordsense disambiguation, given on the one side the greater difficulty of agreeing on a 'neutral' theoretical framework, on the other the difficulty of achieving more than moderate agreement on semantic judgments [Poesio and Artstein, 2005, Zaenen, 2006]. For this reason, a great deal of effort is underway to develop and/or improve semi-automatic methods for creating annotated resources and/or for using the existing data, such as active learning and bootstrapping.

The primary objective of the ANAWIKI project (<http://www.anawiki.org>) is to experiment with a novel approach to the creation of large-scale annotated corpora: taking advantage of the collaboration of the Web community, both through co-operative annotation efforts using traditional annotation tools and through the use of game-like interfaces [Poesio et al., 2008]. In this paper we present our work to develop *Phrase Detectives*, a game designed to collect judgments about anaphoric annotations.

## 2 Creating Resources

### 2.1 Traditional Annotation Methodology

Large-scale annotation of low-level linguistic information (part-of-speech tags) began with the Brown Corpus, in which very low-tech and time consuming methods were used; but already for the creation of the British National Corpus (BNC), the first 100M-word linguistically annotated corpus, a faster methodology was developed consisting of preliminary annotation with automatic methods followed by partial hand-correction [Burnard, 2000]. This was made possible by the availability of fairly high-quality automatic part-of-speech taggers (CLAWS). With the development of the first medium high-quality chunkers this methodology became applicable to the case of syntactic annotation, and indeed was used for the creation of the Penn Treebank [Marcus et al., 1993] although in this case much more substantial hand-checking was required.

Medium and large-scale semantic annotation projects (coreference, wordsense) are a fairly recent innovation in Computational Linguistics. The semi-automatic annotation methodology cannot yet be used for this type of annotation, as the quality of, for instance, coreference resolvers is not yet high enough on general text. Nevertheless semantic annotation methodology has made great progress with the development, on the one end, of effective quality control methods (see for example [Hovy et al., 2006]); on the other, of sophisticated annotation tools such as Serengeti [Stührenberg et al., 2007]. These developments have made it possible to move from the small-scale semantic annotation projects of a few years ago, whose aim was to create resources of around 100K words in size,

e.g. [Poesio, 2004], to projects aiming at creating 1M words corpora. But such techniques could not be expected to be used to annotate data on the scale of the British National Corpus.

## 2.2 Creating Resources through Web Collaboration

Collective resource creation on the Web offers a different way to the solution of this problem. Wikipedia is perhaps the best example of collective resource creation, but it is not an isolated case. The willingness of Web users to volunteer on the Web extends to projects to create resources for Artificial Intelligence. One example is the Open Mind Commonsense project, a project to mine commonsense knowledge to which 14,500 participants contributed nearly 700,000 sentences [Singh, 2002]. Current efforts in attempting to acquire large-scale world knowledge from Web users include Freebase (<http://www.freebase.com/>) and True Knowledge (<http://www.trueknowledge.com/>).

A slightly different approach to the creation of commonsense knowledge has been pursued in the Semantic MediaWiki project [Krötzsch et al., 2007], an effort to develop a ‘Wikipedia way to the Semantic Web’: i.e., to make Wikipedia more useful and to support improved search of web pages via semantic annotation.

A perhaps more intriguing development is the use of interactive game-style interfaces to collect knowledge such as LEARNER [Chklovski and Gil, 2005], Phetch, Verbosity and Peekaboom [von Ahn et al., 2006]. The ESP game is perhaps the best known example of this approach, a project to label images with tags through a competitive game. 13,500 users played the game, creating 1.3M labels in 3 months [von Ahn, 2006]. If we managed to attract 15,000 volunteers, and each of them were to annotate 10 texts of 700 words, we would get a corpus of the size of the BNC.

## 2.3 Annotating Anaphoric Information

ANAWIKI builds on the proposals for marking anaphoric information allowing for ambiguity developed in ARRAU [Poesio and Artstein, 2005] and previous projects [Poesio, 2004]. The ARRAU project found that (i) using numerous annotators (up to 20 in some experiments) leads to a much more robust identification of the major interpretation alternatives (although outliers are also frequent); and (ii) the identification of alternative interpretations is much more frequently a case of implicit ambiguity (each annotator identifies only one interpretation, but these are different) than of explicit ambiguity (annotators identifying multiple interpretations). The ARRAU project also developed methods to analyze collections of such alternative interpretations and to identify outliers via clustering that will be exploited in this project. These methods for representing multiple

interpretations and for dealing with them are used as the technical foundation for an annotation tool making it possible for multiple Web volunteers to annotate semantic information in text.

### **3 Game Interface for Annotating Data**

#### **3.1 Description of the Game**

*Phrase Detectives* is a game offering a simple user interface for non-expert users to learn how to annotate text and to make annotation decisions. The goal of the game is to identify relationships between words and phrases in a short text. “Markables” are identified in the text by automatic pre-processing. There are 2 ways to annotate within the game: by selecting a markable that corefers to another highlighted markable (Annotation Mode - see Figure 1); or by validating a decision previously submitted by another user (Validation Mode - see Figure 2).

#### **3.2 Annotation Mode**

In Annotation Mode the user has to locate the closest antecedent markable of an anaphor markable highlighted in orange i.e. an earlier mention of the object. The user can move the cursor over the text and markables are revealed in a bordered box. To select it the user clicks on the bordered box and the markable becomes highlighted in blue. They can repeat this process if there is more than one antecedent markable (i.e. for plural anaphors such as “they”). They submit the annotation by clicking the “Found it!” button and are given points. The user can indicate that the highlighted markable has not been mentioned before (i.e. it is not anaphoric), or they can skip the markable and move on to the next one.

#### **3.3 Validation Mode**

In Validation Mode the user is presented with an annotation from a previous user. The anaphor markable (orange) is shown with the antecedent markable(s) (blue) that the previous user chose. The current user has to decide if they agree with this annotation. Points are given to the current user, and also to the previous user who made the original annotation. If the current user disagrees with the previous user he is shown the Annotation Mode so he can enter a new annotation.

#### **3.4 Training and Motivating Users**

Users begin the game at the training level where they are given a set of annotation tasks created from the Gold Standard. They are given feedback and



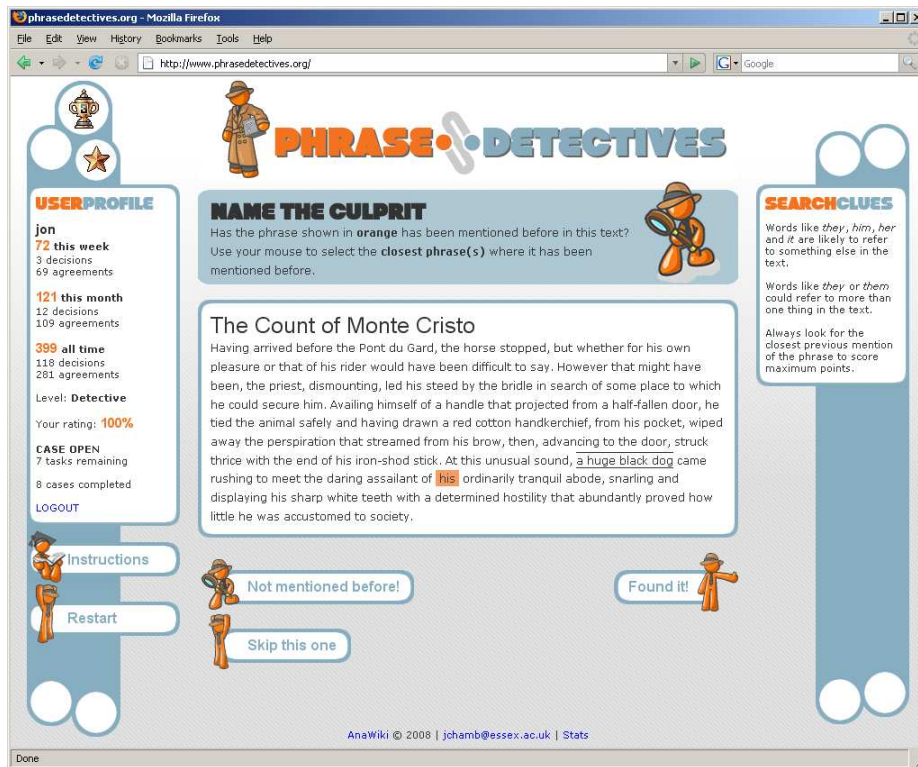


Figure 1: A screenshot of the Annotation Mode.

guidance when they select an incorrect answer and points when they select the correct answer. When the user gives enough correct answers they graduate to annotating texts that will be included in the corpus.

Occasionally, a graduated user will be covertly given a Gold Standard text to annotate. A bonus screen will be shown when the user has completed annotating the text indicating what the user selected incorrectly, with bonus points for agreeing with the Gold Standard. This is the foundation of a user rating system to judge the quality of the user's annotations.

The game is designed to motivate users to annotate the text correctly by using comparative scoring (awarding points for agreeing with the Gold Standard), and collaborative scoring (awarding points to the previous user if they are agreed with by the current user). Using leader boards and assigning levels for points has been proven to be an effective motivator, with users often using these as targets [von Ahn, 2006].

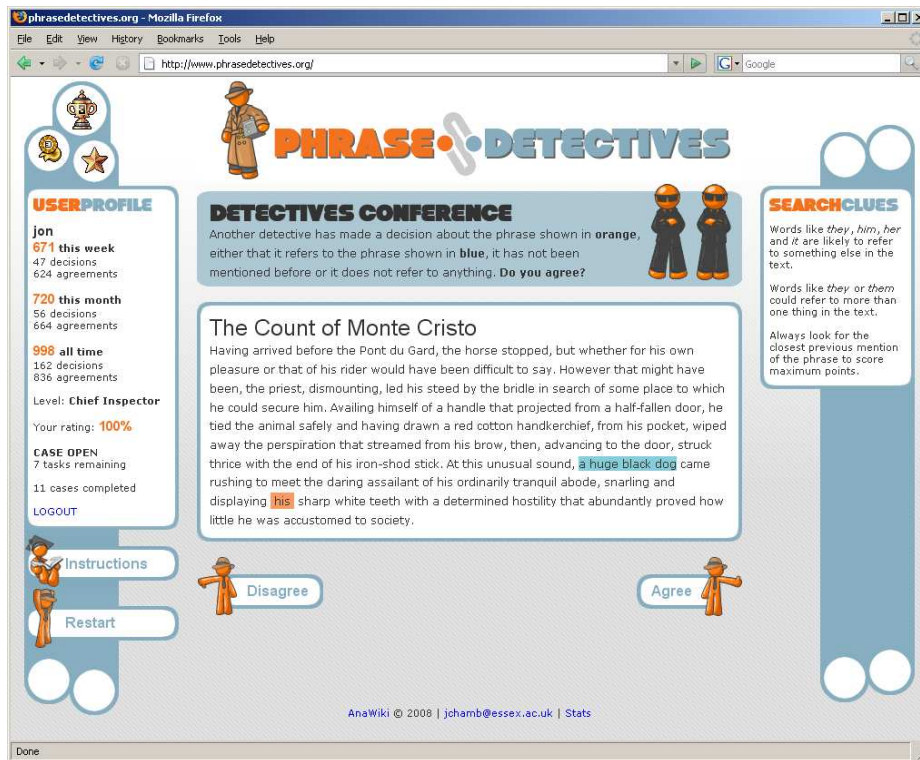


Figure 2: A screenshot of the Validation Mode.

### 3.5 Preventing Cheating and Filtering Erroneous Annotations

Several methods will be used to identify users who are cheating or who are providing poor annotations. These include checking the IP address, randomly checking annotations against known answers and keeping a blacklist of players to discard all their data [von Ahn, 2006]. Additionally we will time annotations, as this could indicate that the user either did not spend long enough reading the text or it is an automated submission. We anticipate annotation times will be different for each mode, with validation mode being approximately twice as fast as annotation mode [Chklovski and Gil, 2005].

## 4 Preliminary Study of the Game Interface

A prototype of the game interface was informally evaluated by 16 randomly selected volunteers from the University of Essex which included staff and students.

Feedback was collected in interviews after each session with the aim of getting an insight into the game tasks and the user interface.

We discovered that a training task was necessary, in addition to the instructions, to help the users understand the tasks. Most (80%) of volunteers felt that 2 example tasks would have been sufficient for training.

The reading styles of each volunteer varied considerably, with some reading the whole text, some reading backwards from the markable and others using scanning techniques to look for specific grammatical elements. They were interested in a broad range of topics, including news, travel, factual and literature.

Of the volunteers who used Facebook (67%), all said they would be motivated to play the game if it was integrated with their profile. It is our intention to use social networking sites (including Facebook, Bebo, and MySpace) to attract volunteers to the game and motivate participation by providing widgets (code segments that display the user's score and links to the game) to add to their profile pages.

A beta version of the game was released online in May 2008 to evaluate the game interface, review the systems in place, to train users and determine the quality of the annotations compared to the Gold Standard.

## 5 Corpus Selection

One of the biggest problems with current semantically annotated corpora (unlike, say, the BNC) is that they are not balanced – in fact they tend to consist almost exclusively of news articles. We plan to address this issue by including a selection of English texts from different domains and different genres. Only copyright-free texts will be included. One obvious example of texts not extensively represented in current semantically annotated corpora, yet central to the study of language, is narratives. Fortunately, a great deal of narrative text is available copyright-free, e.g., through Project Gutenberg for English and similar initiatives for other languages. Another example of texts not included in current semantically annotated corpora are encyclopaedic entries like those from Wikipedia itself. We also expect to include sample text from emails (e.g. from the Enron corpus), text from the American National Corpus and transcripts of spoken text.

The chosen texts will be stripped of all presentation formatting, HTML and links to create the raw text. This will be automatically parsed for POS tags and to extract markables consisting of noun phrases. The resulting XML file can then be inserted into the game database to be annotated.

## 6 Future Work

Our aim is to have a fully functioning game annotating a corpus of one million words by September 2008. We will be considering extending the interface to include different annotation tasks, for example marking coreference chains or Semantic Web mark-up and will present the game interface to gain feedback from the linguistic and Semantic Web community.

## Acknowledgements

ANAWIKI is funded by EPSRC grant number EP/F00575X/1. Thanks to Ron Artstein as well as the Sekimo people at the University of Bielefeld: Daniela Goecke, Maik Stührenberg, Nils Diewald and Dieter Metzger. We also want to thank all volunteers who have already contributed to the project.

## References

- [Burnard, 2000] Burnard, L. (2000). The British National Corpus Reference guide. Technical report, Oxford University Computing Services, Oxford.
- [Chklovski and Gil, 2005] Chklovski, T. and Gil, Y. (2005). Improving the design of intelligent acquisition interfaces for collecting world knowledge from web contributors. In *Proceedings of K-CAP '05*, pages 35–42.
- [Hovy et al., 2006] Hovy, E., Marcus, M., Palmer, M., Ramshaw, L., and Weischedel, R. (2006). OntoNotes: The 90% Solution. In *Proceedings of HLT-NAACL06*.
- [Kröttsch et al., 2007] Kröttsch, M., Vrandečić, D., Völkel, M., Haller, H., and Studer, R. (2007). Semantic Wikipedia. *Journal of Web Semantics*, 5:251–261.
- [Marcus et al., 1993] Marcus, M. P., Marcinkiewicz, M. A., and Santorini, B. (1993). Building a large annotated corpus of english: the Penn Treebank. *Computational Linguistics*, 19(2):313–330.
- [Palmer et al., 2005] Palmer, M., Gildea, D., and Kingsbury, P. (2005). The proposition bank: An annotated corpus of semantic roles. *Computational Linguistics*, 31(1):71–106.
- [Poesio, 2004] Poesio, M. (2004). The MATE/GNOME scheme for anaphoric annotation, revisited. In *Proceedings of SIGDIAL*.
- [Poesio and Artstein, 2005] Poesio, M. and Artstein, R. (2005). The reliability of anaphoric annotation, reconsidered: Taking ambiguity into account. In *Proceedings of the ACL Workshop on Frontiers in Corpus Annotation*, pages 76–83.
- [Poesio et al., 2008] Poesio, M., Kruschwitz, U., and Chamberlain, J. (2008). ANAWIKI: Creating anaphorically annotated resources through Web cooperation. In *Proceedings of LREC'08*, Marrakech.
- [Singh, 2002] Singh, P. (2002). The public acquisition of commonsense knowledge. In *Proceedings of the AAAI Spring Symposium on Acquiring (and Using) Linguistic (and World) Knowledge for Information Access*, Palo Alto, CA.
- [Stührenberg et al., 2007] Stührenberg, M., Goecke, D., Diewald, N., Mehler, A., and Cramer, I. (2007). Web-based annotation of anaphoric relations and lexical chains. In *Proceedings of the ACL Linguistic Annotation Workshop*, pages 140–147.
- [von Ahn, 2006] von Ahn, L. (2006). Games with a purpose. *Computer*, 39(6):92–94.
- [von Ahn et al., 2006] von Ahn, L., Liu, R., and Blum, M. (2006). Peekaboom: a game for locating objects in images. In *Proceedings of CHI '06*, pages 55–64.
- [Zaenen, 2006] Zaenen, A. (2006). Mark-up Barking Up the Wrong Tree. *Computational Linguistics*, 32(4):577–580.

## Building a Semantic Portal from the Discussion Forum of a Community of Practice

**Bassem Makni, Khaled Khelif, Rose Dieng-Kuntz, Hacène Cherfi**

(INRIA Sophia Antipolis Méditerranée, Edelweiss Team,  
2004 route des Lucioles - BP 93, 06902 Sophia Antipolis Cedex, France  
{bassem.makni, khaled.khelif, rose.dieng, hacene.cherfi}@sophia.inria.fr)

**Abstract:** In this paper, we describe SemanticFAQ system built using a method for semi-automatic creation of an ontology and of semantic annotations from a corpus of e-mails of a community of practice. Such an e-mail corpus raises several original issues for Natural Language Processing (NLP) techniques. The annotations thus generated will feed a frequently asked questions (FAQ). The ontology and the annotations constitute the basis of a semantic portal, SemanticFAQ, that offers the CoP members a semantic navigation among the e-mails.

**Keywords:** ontology, semantic annotation, semantic portal, NLP, e-mail processing

**Category:** H.3.1, H.3.3, M.0, M.7

### 1 Introduction

Textual genre of texts handled or produced by a community of practice (CoP) varies from very formal texts such as scientific articles or reports to very informal texts such as e-mails, discussion forums, etc, where the members of the community express quite freely, possibly with orthographic or grammar mistakes or with abbreviations. The informal nature of e-mails and their low linguistic quality make harder the use of Natural Language Processing (NLP) tools on such texts. However, e-mails are sometimes the main knowledge source of a CoP. This is the case of @pretic CoP, a community of teachers in Belgian secondary schools who are responsible for the management of computer labs in their schools. @pretic members mainly communicate through exchange of e-mails describing the encountered technical problems or suggesting solutions for solving such problems. In order to facilitate navigation among past e-mails and to find solutions to problems previously discussed, we propose an approach for automatic creation of semantic annotations on such e-mails, annotations based on an ontology partly created from linguistic analysis of this corpus of e-mails. SemanticFAQ portal relies on such generated annotations and on a semantic search engine for offering ontology-guided navigation through the e-mails. Section 2 presents the structure of @pretic ontology, section 3 details the process of semi-automatic creation of the *Computer-Problem* ontology from the e-mail corpus, section 4 describes semantic navigation and semantic search on the e-mails, and section 5 concludes.

## 2 @PRETIC Ontology Structure

Since a member of @pretic CoP writes or seeks e-mails about a problem related to computer components, the @pretic ontology is structured in four ontologies: (i) an ontology describing computer components, (ii) an ontology describing e-mails, (iii) an ontology describing members of the CoP, and (iv) an ontology describing computer problems. The @pretic ontology consists of the following sub-ontologies:

1. *OntoPedia*: All possible computer components on which problems may occur are not necessarily mentioned in the e-mails; so relying on a linguistic analysis of the e-mail corpus would have led to an incomplete ontology. Therefore, we preferred to reuse an existing term hierarchy (Webopedia). We developed a program that, from the term hierarchy of this online encyclopaedia, generates automatically an ontology represented in RDFS.
2. *Oemail*: it describes metadata on e-mails by defining generic concepts (e.g. E-mailMessage), more specific concepts (e.g. ReplyMessage) and semantic relationships (e.g. author, date, recipient, etc.).
3. *O'CoP*: this ontology detailed in [Tifous, 07] comprises concepts enabling to describe a CoP, its actors, their roles and competences, the resources they use, etc. We used O'CoP ontology to describe @pretic CoP members.
4. *Computer-Problem* ontology: it is the main module of @pretic ontology and it aims to provide concepts and properties enabling to describe the computer problems faced by CoP members. To initiate and enrich this ontology, we applied NLP techniques on the corpus of e-mails.

## 3 Computer-Problem Ontology Building

Due to the very low quality of the e-mail corpus, a significant cleanup phase was needed to obtain texts in quality acceptable by NLP tools.

### 3.1 Corpus Cleaning

This phase consists of five steps:

- *Preliminary cleaning*: we developed a module, based on JavaMail API, for exporting the e-mails in XML format, deleting "spams" and attachments, and restoring the links between the origin messages and their responses.
- *Filtering signatures*: as often the e-mail senders did not respect MIME standards for digital signature, our algorithm detects signatures in e-mails.
- *Language detection*: although @pretic is a French-speaking community, the e-mail corpus was trilingual (several messages were written in English and Flemish). We used the TextCat<sup>1</sup> tool for detecting the language so as to keep only the messages written in French.
- *Re-accentuation*: To solve the problem of absence of accentuation in e-mails texts, we used the REACC<sup>2</sup> tool.

<sup>1</sup> <http://odur.let.rug.nl/~vannoord/TextCat/>

<sup>2</sup> <http://rali.iro.umontreal.ca/Reacc/Reacc.fr.cgi>

- *Repetitive cleaning*: the bodies of e-mails contain a lot of noise even after cleaning: greetings, thanks, not filtered signatures, and so on. We adopted a method of semi-automatic cleaning in order to speed up the detection of noise: we browsed the extracted candidate terms to detect those that were not meaningful or that corresponded to private messages. We used them to generate new filters and we developed a tool for cleaning assistance.

### 3.2 Ontology Bootstrap and Enrichment

The extraction of candidate terms aims at extracting meaningful terms enabling to build an ontology and covering most of computer problems. For this, we used two existing term extractors: FASTR<sup>3</sup> based on syntactic approach and ACABIT<sup>4</sup> based on syntactico-statistical approach.

To bootstrap the *Computer-Problem* ontology, we considered candidate terms stemming from “initial messages”, i.e. messages that open a discussion and that are likely to raise a problem. These messages share a syntactic regularity through the terms used to express a problem. This regularity consists of the use of the word “*problème*” followed, after a few words, by the computer component or by the computer-related task concerned by this problem: e.g. “*problème de câblage*” (*wiring problem*), “*problème de mauvaise connexion*” (*bad connection problem*)... Use of such regularities allowed us to start the ontology building process by selecting and formalizing such candidate terms. We thus obtained an initial ontology which was validated by members of @pretic CoP. However, this initial ontology, albeit interesting in covering most of the encountered problems, was fairly generic and may induce an ambiguity when generating annotations. In order to enrich our ontology, we carried out a manual analysis of all candidate terms generated by the NLP tools. The following list shows examples of terms thus extracted by both tools and used for enriching ontology:

“*lenteur de connexion*” (*slow connection*), “*manque de mémoire*” (*lack of memory*), “*perte de donnée*” (*loss of data*), “*retard dans la réponse*” (*delayed response*) etc.

The study of this list of terms allowed us to:

- Detect new meaningful terms to directly enrich the ontology (“*lack of memory*”, “*slow connection*”, etc.).
- Detect synonymy relationships between some significant terms (“*insufficient memory*” and “*memory insufficiency*”, “*infected message*” and “*message infection*”, etc.). These terms resulted in synonymous terms used as labels for the same concept of the ontology.
- Determine structural regularities (i.e. syntactic patterns) for terms expressing problems (“*slow X*”, “*loss of X*”, “*difficulty in X*”, “*delay of X*”, “*lack of X*”, etc., where X is a concept in *Ontopedia* ontology).

In a second stage, we took inspiration from [Golebiowska, 01] to propose heuristic rules supporting semi-automatic enrichment of the ontology. These rules detect predefined structures in the text and enrich ontology terms by candidates that

<sup>3</sup> <http://www.limsi.fr/Individu/jacquemi/FASTR/>

<sup>4</sup> [http://www.sciences.univ-nantes.fr/info/perso/permanents/daille/acabit\\_en.html](http://www.sciences.univ-nantes.fr/info/perso/permanents/daille/acabit_en.html)

had not necessarily been detected by NLP tools. These rules are written in JAPE [Cunningham, 02] syntax and plugged in the annotation process by OntoPedia.

### 3.3 Semi-automatic Building of Hierarchical Relationships among Concepts

After the phase of detection of terms expressing problems, the *Computer-Problem* ontology had no hierarchical relationship between concepts. Therefore, we developed an algorithm for automatic attachment to the relevant generic concept of *Computer-Problem* ontology (*Hardware Problem*, *Software Problem*, etc.). For each concept in *Computer-Problem* ontology, we generate a list of neighbour concepts appearing in the same e-mail and annotated by concepts of the *Ontopedia* ontology. Then we chose a set of core concepts from *Ontopedia* ontology, that were detected in the majority of discussions. For each obtained list, we calculated the sum of the semantic distance between the concepts of this list and the core concepts. We calculated these distances using the semantic distance offered by the semantic search engine CORESE [Corby, 04]. The chosen category for a term is the one that has the smallest semantic distance.

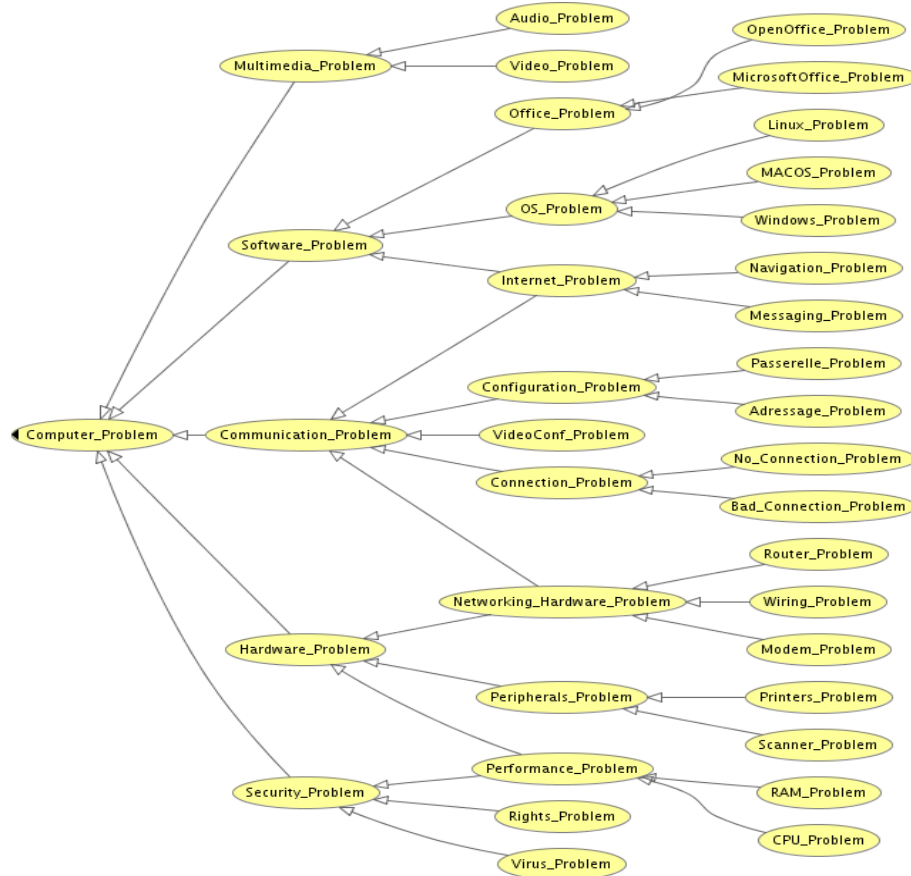


Figure 1: The Computer-Problem ontology (in English)



## 4 @PRETIC Ontology Use

The aim of @PRETIC ontology building is semantic annotation of e-mails so as to offer semantic information retrieval from e-mails.

### 4.1 Semantic Annotation of e-mails

The architecture of our annotation system which offers both meta-data and content annotation is shown in Figure 2.

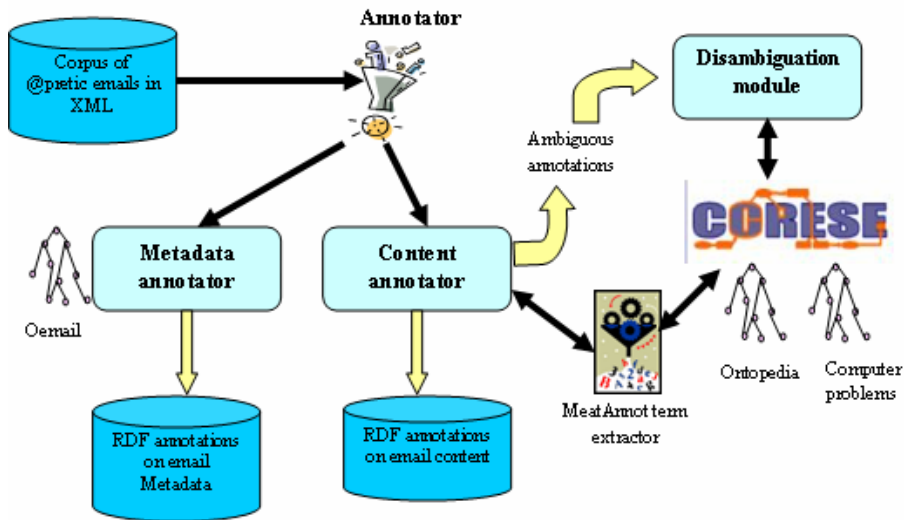


Figure 2: Annotator architecture

*Oemail* ontology describes metadata on e-mails. The metadata-based annotation process consists of two sub-processes:

1. *Mail parsing*: this process consists of acquiring the e-mail data in its raw form and of extracting headers and body in XML. On this purpose, we have developed an e-mail monitor which listens to a mail server (IMAP or POP3) and builds automatically an XML tree for each incoming e-mail. Our implementation is based on JavaMail API.
2. *Mail annotation*: this process involves mapping of XML elements detected in the previous phase with their corresponding concepts in *Oemail* and then exporting to RDF.

For the annotation of the e-mail content (i.e. body), we adapted the MeatAnnot term extractor [Khelif, 07] in order to send queries to Corese for identifying in the texts the terms corresponding to labels of concepts of the *Ontopedia* and *Computer-Problem* ontologies.

Our ontology may contain ambiguities since some terms can be attached to more than one concept: e.g. the term "*Upgrade*" can be considered as instance of "*Hardware*" and "*Software*". Therefore, we developed a disambiguation algorithm

based on CORESE semantic distance [Corby, 04]: it generates a vector of all concepts found in the message, calculates a matrix of semantic distances, and selects the concept which has the smallest semantic distance from its neighbours.

#### 4.2 Information Retrieval from e-mails through Semantic Portal

The @prectic ontology aims to guide semantic search among the frequently asked questions to solve the problems faced by the CoP members. SemanticFAQ thus offers a Web interface enabling ontology-based navigation in the description of problems and their answers. We adapted the hyperbolic navigation [Munzner, 95] originally used in the navigation of websites to ontology navigation. Hyperbolic navigation has the advantage of giving an overall view well suited to a member of the CoP that does not know the hierarchy of problems. The choice of one or several concepts is followed by a query to the semantic search engine CORESE to get the messages annotated by these problems, the answers then displayed in the form of discussion feed. In order to reduce the execution time of our Web application, the metadata are calculated and displayed when the user flies over a message.

We have designed a web portal application to encapsulate the hyperbolic navigation through @prectic ontology. The main aim of choosing the portal architecture is to enable user profile awareness and secure access to the CoP knowledge resources. The SemanticFAQ portal allows the semantic navigation through *Computer-Problem* and *Ontopedia* ontologies. We have plugged some semantic functionalities in the @prectic portal, for example the registration process checks whether the user is a member of the CoP by querying the metadata annotation through CORESE. As an example, in figure 3, the CoP member seeks for communication and hardware problems. The SemanticFAQ system queries the semantic search engine CORESE to get the discussion feeds annotated by the chosen concepts and displays the messages sorted either by subject, by author or by date.

SemanticFAQ retrieves the whole discussion feed annotated by one or several concepts through a unique query to the semantic search engine CORESE that implements SPARQL query language through graph homomorphism [Corby, 07]. SemanticFAQ algorithm comprises the following modules: (1) *Global\_Discussion\_Feed* builds all the paths guided by the "Oemail" property "ReplyTo" indicating that an e-mail is a response to another one; (2) *Query* retrieves the e-mails annotated by the concepts indicated as input; (3) *Path\_restriction*, having a set of e-mails and the Global Discussion feed, maintains the paths that cross the e-mails of the set; (3) *Paths\_to\_Tree* builds recursively a tree from a given list of paths.

The screenshot displays the 'Apretic Portal' interface. At the top, there is a navigation bar with 'Navigation hyperbolique' and 'Accueil'. The main content area is divided into two panels. The left panel, titled 'Hypergraph', shows a complex network of nodes representing concepts from a 'Problème Informatique' ontology. Nodes include 'Absence de connexion', 'Problème de configuration', 'Problème de communication', 'Problème de passerelle', 'Problème d'adressage', 'Problème de mauvaise connexion', 'Problème de bureau informatique', 'Problème Matériel', 'Problème Informatique', and 'Problème Logiciel'. A search bar at the bottom of the hypergraph shows 'Les concepts choisis: Problème de communication, Problème Matériel'. The right panel, titled 'Fil de discussion', shows a search bar and a list of email threads. One thread is selected, showing a subject 'descente d'image' and a snippet of text: 'Celle arrivée, pas une seule fois... et pourtant, le CCIM tourne...'. Below the text is a 'Mots clés' section with 'Re: [prets\_sel] descente d'image', 'Françoise Bouko', and 'Mer: Jun 04 20:03:54 CEST 2001'.

Figure 3: Hyperbolic navigation through the Computer-Problem ontology

## 5 Conclusions

In this paper, we presented an original approach for semi-automatic building of an ontology and semantic annotations from a corpus of e-mails of a CoP, and a semantic portal for this CoP. Moreover, @pretic CoP members validated the *Computer-Problem* ontology building and evaluated SemanticFAQ portal from user viewpoint.

Our approach is partially inspired by the method for ontology learning from textual comments of databases [Golebiowska, 01]. In comparison to current work on ontology learning [Aussenac-Gilles, 00] [Buitelaar, 05] or on annotation learning [Uren, 06], our work is original by its effective use of NLP techniques on the highly degraded corpus constituted by the body of e-mails. This low linguistic quality did not allow us to use relation extraction techniques as those we previously proposed in [Khelif, 07]. Our cleaning and extraction techniques are very different from mail cleaning offered in [Tang, 06] and from techniques presented in [Even, 02] where the authors extract knowledge from degraded but formal texts. Extracting information from e-mails was also offered by [Zhong, 02] and [Sakurai, 05] but their main objective is e-mail classification for spam filtering.

As further work, we will study the possibility of exploitation of the other ontologies (Human-Problem, Learning-and-Teaching), in particular so as to exploit the pedagogical messages occurring in the e-mails. An evaluation of the quality of annotation and retrieval processes will be performed.

## Acknowledgements

We thank very much the EU Commission for funding the European project IST Palette, as well as the @pretic CoP members.

## References

- [Aussenac-Gilles, 00] Aussenac-Gilles, N., Biébow, B., Szulman, S.: Corpus analysis for conceptual modelling. In EKAW'2000 Workshop "Ontologies and texts", Oct. 2000.
- [Buitelaar, 05] Buitelaar, P., Cimiano, P., Magnini, B.: *Ontology Learning from Text: Methods, Evaluation And Applications*, IOS Press, July 2005.
- [Corby, 04] Corby, O., Dieng-Kuntz, R., Faron, C.: Querying the semantic web with the Coresense engine. Proc. of the ECAI'2004, Valencia, August 2004, IOS Press, p. 705-709, 2004.
- [Corby, 07] Corby, O., Faron-Zucker, C.: Implementation of SPARQL query language based on graph homomorphism. In ICCS'2007, pages 472–475, 2007.
- [Cunningham, 02] Cunningham, H.: GATE, a General Architecture for Text Engineering. *Computers and the Humanities*, 36:223–254, 2002.
- [Even, 02] Even, F., Enguehard, C.: Extraction d'information à partir de corpus dégradés. In Actes, (TALN 2002), volume 1, pages 105–114, 2002.
- [Golebiowska, 01] Golebiowska, J., Dieng-Kuntz, R., Corby, O., Mousseau, D.: Building and exploiting ontologies for an automobile project memory, K-CAP, p. 52-59, October 2001.
- [Khelif, 07] Khelif, K., Dieng-Kuntz, R., Barbry, P.: An ontology-based approach to support text mining and information retrieval in the biological domain, JUCS, 13(12):1881-1907, 2007.
- [Munzner, 95] Munzner T., Burchard, P.: Visualizing the structure of the World Wide Web in 3D hyperbolic space. In Proc. 1st Symp. The VRML Modelling Language: p 33–38, 1995
- [Sakurai, 05] Sakurai, S., Suyama, A.: An e-mail analysis method based on text mining techniques. *Applied Soft Computing*, 6(1):62-71, 2005.
- [Tang, 06] Tang, J., Li, H., Cao, Y., Tang, Z., Liu, B., Li, J.: E-mail data cleaning. Technical Report MSR-TR-2006-16, Microsoft Research (MSR), Feb. 2006.
- [Tifous, 07] Tifous, A., El Ghali, A., Giboin, A., Dieng-Kuntz, R.: O'CoP, an Ontology Dedicated to Communities of Practice, Proc. of I-KNOW'07, Graz, Austria, Sept. 2007.
- [Uren, 06] Uren, V., Cimiano, P., Iria J., Handschuh, S., Vargas-Vera, M., Motta, E. and Ciravegna, F.: Semantic annotation for knowledge management: Requirements and a survey of the state of the art. In *Web Semantics*, 4(1): 14-28, 2006.
- [Zhong, 02] Zhong, N., Matsunaga, T., and Liu, C. 2002. A Text Mining Agents Based Architecture for Personal E-mail Filtering and Management. Proc. of the 3rd Int. Conference on Intelligent Data Engineering and Automated Learning, August 2002, p. 329-336.

## **A Semantic Approach towards CWM-based ETL Processes**

**Anh Duong Hoang Thi**

(Hue University Information Technology Center, Hue, Vietnam  
htaduong@hueuni.edu.vn)

**Binh Thanh Nguyen**

(Hue University Information Technology Center, Hue, Vietnam  
ntbinh@hueuni.edu.vn)

**Abstract:** Nowadays, on the basis of a common standard for metadata representation and interchange mechanism in data warehouse environments, Common Warehouse Metamodel (CWM) – based ETL processes still has to face significant challenges in semantically and systematically integrating heterogeneous sources to data warehouse. In this context, we focus on proposing an ontology-based ETL framework for covering schema integration as well as semantic integration. In our approach, beside the schema-based semantics in CWM-compliant metamodels, semantic interoperability in ETL processes can be improved by means of an ontology-based foundation to better representation, and management of the underlying domain semantics. Furthermore, within the scope of this paper, a set of CWM-based modelling constructs driven by ontology for the definition of metadata required for ETL processes is defined, facilitating the extraction, transformation and loading of useful data from distributed and heterogeneous sources. Thus, the role of interconnecting CWM and semantic technologies in populating data warehousing systems with quality data and providing data warehouse an integrated and reconciled view of data is highlighted.

**Keywords:** Ontology, Common Warehouse Model, Metadata Interoperability, ETL process, Data integration.

**Categories:** H.3.2, H.3.4, H.4.1, H.4.2

### **1 Introduction**

As data warehousing expands its scope increasingly in more and more areas, thus deployed in totally heterogeneous, and distributed application environments, the Extraction, Transformation and Loading disparate data sources to integrated data warehouse has become one of the fundamental issues for the success of data warehousing systems. Fully conformant to the CWM specifications, a metadata interchange standard in the data warehousing and business analysis environments proposed by Object Management Group (OMG) [OMG, 03], CWM-based ETL processes can support the model-driven transformation of raw data into strategic business information, in which object models representing metadata are constructed according to the syntactic and semantic specifications of a common metamodel. On the other hand, the semantics and interfaces of CWM make it a powerful model for the construction of a new generation of consistent, uniform ETL tools that are both dynamically configurable and highly adaptive to different environments.

Unfortunately, as other OMG's suite of standards, CWM metamodel and metadata, which is the instances of metamodel, primarily reveal adequate semantics for resolving only the schema conflicts and say little about the underlying semantics of the domain being modelled [Dragan 06]. As a result, the ETL processes based on CWM still inevitably has to face critical challenges due to the heterogeneity, inconsistencies of metadata, different business processes, data organization that need to be reconciled [Kimball, 04]. Containing knowledge about a domain in a precise and unambiguous manner, ontology has the ability to represent an adaptive knowledge representation for both the semantics of the modelling language constructs as well as the semantics of model instances, especially fulfilling the modelling requirements in representation and enforcement of the semantics of multilayered models like CWM.

In that context, the main contribution of this paper is to propose a semantic model approach in ETL processes, the fundamental part of in DWH system. In the proposed approach, a combined use of both CWM and semantics in terms of ontology – based ETL framework up to CWM will be introduced, providing well-defined integrated, semantic foundation for CWM based metadata interoperability of various aspects of the ETL process. Specifically, supported by the transformation process of CWM-based models into an ontology-based models, the semantics of CWM modelling concepts, whose syntax is defined by the metamodel, are explicitly expressed by means of ontology, especially in Description Logics [Franz, 03] providing significant advantages for ETL processes, enriched with the stabilized descriptions of a business domain [Liu, 05], e.g. allowed values for attributes of model instances.

The rest of this writing is organized as follows: section 2 introduces some approaches related to our work; after providing the insight that a combined use of CWM and semantic is a potential approach of enhanced semantic interoperability of ETL processes, in section 3, the ETL framework is presented with its specification and core components as well as its semantic mechanisms, before the concepts of the link establishment between model elements and ontology concepts is discussed in section 4, founding the semantic basis for ontology-based extraction, transformation and loading of disparate data sources to integrated data warehouse. At last, section 5 gives a summary of what have been achieved and future works.

## 2 Related works

The characters of the proposed approach rise from being rooted in several traditionally disparate research fields such as ontology engineering, model driven development, ETL processes and data integration in general.

Considering the ETL process as a key component in a data warehousing environment, significant number of works by several groups have been put into action to the modelling, design, control and execution of ETL processes. For example, Trujillo and Luján-Mora [Luján-Mora, 06] proposed a UML-based approach for modelling the ETL process so as to ease its correct design by providing a reduced, yet powerful, set of ETL mechanisms, e.g. aggregations, conversions, and filters. However, the approaches have not focused on semantic heterogeneity problems in data integration, which is one of the main objectives of this paper.

In the past few years, significant numbers of works by different groups have been put into action to move towards ontology-based data integration framework as a way

to solve ETL problems of semantic heterogeneity. The most extensive study of common models for ETL and ETL task optimization is by Simitis et al. [Simitis, 05]. Their work proposes a multi-level workflow model that can be used to express ETL tasks. ETL tasks expressed in their model can be analyzed and optimized using well-understood logical inference rules. Although the presented projects cover the complete ETL process, their metamodels do not conform to emerging standards such as the CWM, thus, making it difficult to integrate the tool into existing heterogeneous data warehousing environments.

Meanwhile, various researches have been done to bridge the gap between the metamodel-based integration and semantic technology, supporting the business model interoperability [Höffner, 07]. To the best of our knowledge, in the state-of-the-art research and practice, little related research on the argument in this paper is found. From our point of view, these proposals are oriented to the conceptual and logical design of the data warehouses development, and do not seriously taking into account important aspects such as the challenges of ETL processes with structural and semantic heterogeneity, the main focus of this research.

In our work, we present an ontology-driven approach towards CWM-based ETL processes, in which an integrated use of CWM and ontology provides the wider practitioner population to develop semantic technology in the process of extracting, transforming and loading data into data warehouse. Thus, the well-defined descriptions of schema-based and content-based semantics in ontology can facilitate the truly interoperable ETL processes up to CWM standard, so that the integration, reusability and interoperability in DWH environments can be achieved, populating data warehousing systems with reliable, timely, and accurate data.

### **3 Semantic ETL framework based on CWM standard**

This section outlines the combined use of ontology-based and the model-based approach towards semantic interoperability. Hereafter, a framework based on ontology for extracting, transforming, and loading data from heterogeneous data sources is presented, producing a conceptually and semantically unambiguous foundation of CWM-based ETL processes. Moreover, an overview of basic concepts provided by proposed framework for modelling and executing the ETL process is given, showing how a semantic and CWM-driven solution contributes to achieving the objective of a flexible warehouse ETL process.

#### **3.1 Metamodels and ontologies – Complementary strengths**

As stated in previous sections, to integrate data from disparate operational sources in ETL process, a relationship between semantically similar but autonomously designed data needs to be established. However, the data sources will be highly heterogeneous in syntax and semantic so the mapping between these sources can hardly be fully resolved by fixed metamodels or frameworks [Höffner, 07]. Moreover, in the current stage metamodels are mainly concerned with the static semantics, with syntax of models, integrity, and well-formedness constraints [Dragan 06]. In this context, the data integration, one of the challenging issues in ETL processes, depends on the form of knowledge representation facilitating conflict solving for the source schemas and

contents. Providing explicitly defined meaning of the information to be exchanged, ontologies enable tasks like logical reasoning and instance classification yielding additional benefits for semantic integration. Hence, from an abstract point of view, the two paradigms and their various technological spaces seem closely related, in which model and ontology-based approaches can be viewed as complementary solutions for addressing semantic interoperability.

In this approach, concerning the relationship between metamodels and ontologies, metamodels can be seen as provider of the syntax of a modelling language, providing valid ways to combine all available modelling constructs [OMG, 03]. Meanwhile, ontologies on the other hand basically provide the semantics and describe both the semantics of the modelling language constructs [OMG, 07] as well as the semantics of model instances. The ontological representation of the CWM-based model enables reasoning about the instance model, which enables a dependency analysis to deduce unknown or implied relationships among entities within the instance model [Liu, 05].

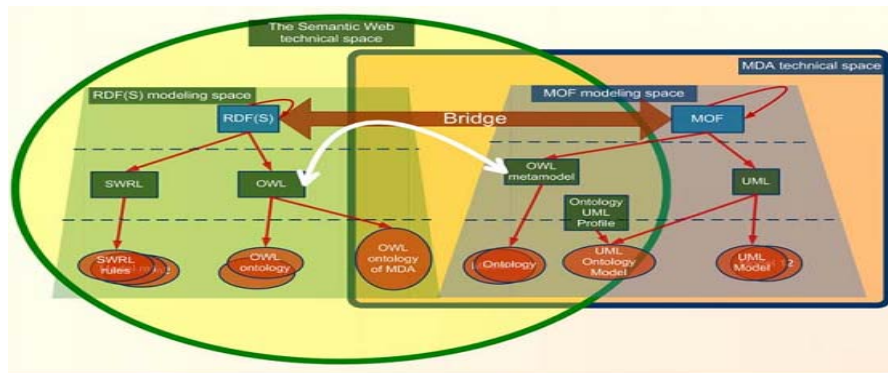


Figure 1: The Semantic Web and MDA technical spaces [Dragan 06]

Moreover, bridges could also be created between different languages used to express semantic relationships between two conceptual models. This would open the possibility to bring e.g. models into the ontology-based world in order to carry out semantic coordination using ontology-related technology. Further, the information about semantic relationships could be brought back to the MDA world, e.g. to refine relationships using transformation operators or to create formal expressions and automatically transform these into executable code. Therefore, in our view, ontologies fulfil an ETL process whereas semantic interoperability can only be achieved when both concepts, metamodels and ontologies, are used in combination.

### 3.2 Conceptual Design for CWM-based ETL framework driven by Ontology

This section will describe a framework addressing semantic integration problems by using ontology to explicitly describe the conceptual specifications of modelling and executing the data warehouse ETL process up to CWM standard. In this approach, CWM-based ETL processes can improve semantic metadata interoperability with the



defined ontology as an adaptive knowledge representation of shared vocabularies to reveal both schema-based and content-based semantics.

Using semantically enriched, domain-specific metamodels which describe a reasonable amount of semantics as well as structure, the ETL framework, as represented in Figure 2, can support three levels of a semantically coupling of a metamodel with an ontology. At the *model level*, ontologies can be used for definition of attribute domains. At the *instance level*, the ontology will be applied for evaluating data quality (e.g., detection of non-fitting attribute values which can be classified into imprecise values [Höfferer, 07]). And at the *metamodel level*, ontology top-level concepts and relations can be coupled with UML constructs (e.g., is-part-of relations or compositions), describing, at high abstraction levels, complex application domains.

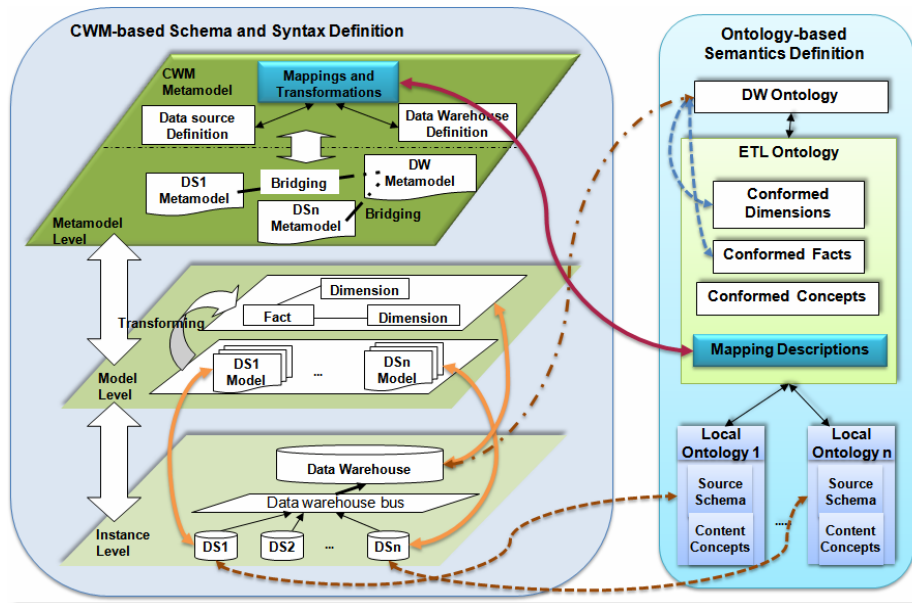


Figure 2: Conceptual CWM-based ETL framework driven by Ontology

Shown in Figure 2, the data sources represent operational systems designed to manage large amount of transaction data. By its heterogeneous, distributed, interrelated nature, each metadata source will be analyzed to define its own metamodel. Moreover, for each source, a local ontology is used as a semantic layer over the source model and data content residing in each source with specifications of concepts and their relations. On the other hand, we assume that the data warehouse also has its own metamodel and ontology to describe the structure and semantics of data warehouse components, e.g. dimensions, and facts [Nguyen, 00].

The core modules in the architecture are CWM-based metamodel and the ETL ontology, providing syntactic and semantics to the ETL processes. In the approach, every model of each sources have to be compared and transformed into DW model with respect to the description of their language. With the state of fact that existing

models to be integrated conform to different metamodels, the integration task of models is passed up in the metamodeling hierarchy and implies the integration of metamodels [Liu, 05], in which metamodel are used as a filter for a first comparison of similarity. Therefore, to guarantee the correct integration of ETL models it has to be ensured first that there is a correct mapping of the metamodels available.

In this context, the CWM-driven metamodel, playing the role of a common metamodel, provides constructs for the definition of metadata required for executing the ETL process, i.e. the description of operational data to be integrated and target warehouse data, along with the definition of mappings between operational and warehouse data. Based on CWM, the ETL process can be started by defining a CWM Transformation model for movement from a data source to a data target. Parameters of the source data, target data, and transformation logic are assigned values in the model [OMG, 03]. Source and target data parameters depend on the type of the data source (relational, XML, multidimensional etc.). Transformation logic parameters include identification of a transformation task and of data sources and data targets.

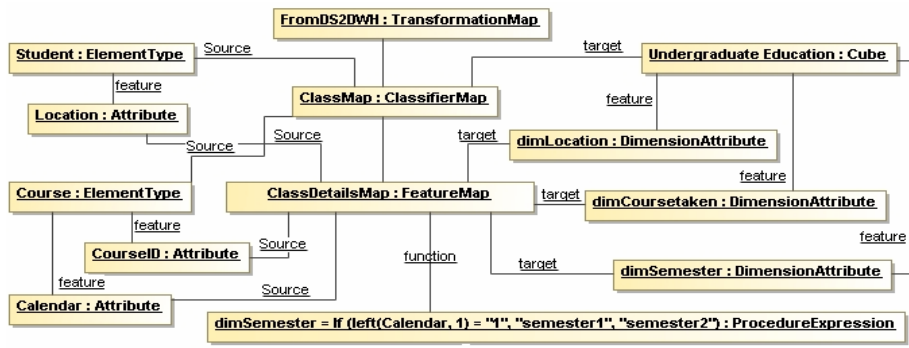


Figure 3: Example of a CWM transformation map between data source and DW

Meanwhile, a shared ETL ontology will be used, including conformed dimensions and conformed facts (equivalent to lowest levels of predefined dimensions and fact class in DW ontology) as well as conformed concepts, an extensible set of reconciled and integrated concepts defining the data content that will be loaded into data warehouse. With the well-defined ontology both in local sources and data warehouse, the ETL ontology then can define (1) the structure, syntax and vocabulary to be used when exchanging data, (2) semantic relationship between model elements, and (3) refined semantic relationships where transformation operators needed to resolve conflicts identified.

Furthermore, the mapping descriptions are also defined in ETL ontology, with a particular focus on the interrelationships between the ETL and local ontologies as well as the interpretations of the concepts in local data sources, at both the class and attribute levels. Moreover, the mapping descriptions also define abstractions of necessary operations executing transforming task, i.e. data cleaning and schema transformation. In this approach, the ETL ontology does not define these transformations explicitly; instead it explicitly identifies the class, attributes or data manipulated with a particular task. Encapsulating the transformation definitions and

associated methods, the ETL ontology supports complex rules to be defined between ontology concepts reflecting domain specific characteristics. In addition, the heterogeneity problems between data sources and warehouse can be resolved automatically [Simitsis, 05], enhancing the quality of semantic interoperability.

#### 4 Semantic and CWM-based metamodel for ETL processes

As discussed in previous sections, ETL is a common term for the warehouse load process comprising a set of data movement operations from a data source to a data target with some transforming or restructuring logic applied [OMG, 03]. In our approach, we take advantage of the combined use of ontology and a predefined CWM metamodel, supporting the modelling and execution of the ETL process at an intermediate layer between operational sources and the data warehouse without making any assumptions about the way operational and warehouse data is stored.

Based on a predefined CWM metamodel providing a set of constructs, the metamodel is used to define a metamodel instance, i.e., the description of tasks related to the ETL process. Hereafter, an ETL process is realized as an operation consisting of a number of transformation step models in sequence, e.g. the transformation combines, or reduces the data and then stores it in the format of the warehouse, etc. Furthermore, to fully support the expressive power for CWM structuring mechanisms, we focus on using Description Logics [Franz, 03], regarded as the foundation of some ontology languages, in formalizing and reasoning on CWM metadata. Taking advantage of the explicit information in the metamodel and the implicit information obtained by reasoning on the metadata which is the instance of metamodel [OMG, 03], we can translate the CWM Transformation metamodel into the Tbox and the metadata into the Abox (illustrated in figure 4). Thus, the framework can provide the formal semantics of CWM metadata by means of a logical approach.

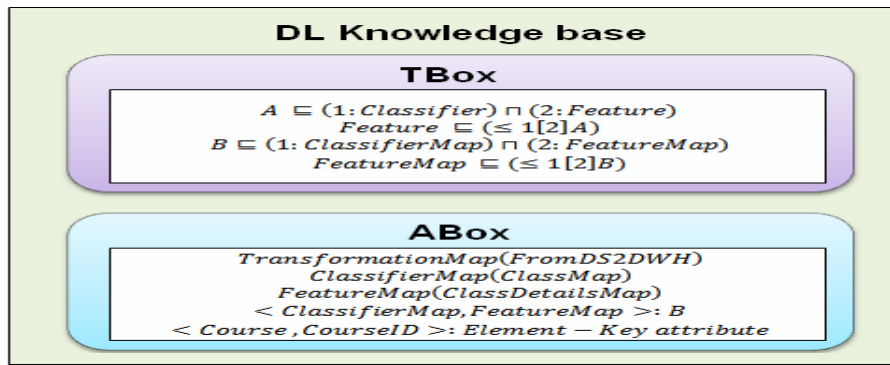


Figure 4: Formalization of CWM in terms of DL

On this conceptual basis, we apply ATL UML2OWL ([www.eclipse.org/gmt/atl](http://www.eclipse.org/gmt/atl)), implemented according to the ODM (Ontology Definition Metamodel) [OMG, 07], facilitating the conversion of CWM-based models into OWL ontology (i.e. CWM classes into OWL classes, attributes into data type property, etc.). Hereafter, the

output ontology can be imported into a tool specialized for ontology development, e.g., Protégé ([www.protege.stanford.edu](http://www.protege.stanford.edu)) where the ontologies can be further refined, enriched with additional data semantics (such as synonym and hyponyms), analysed and compared to determine their similarities and differences. Along with the local and global ontologies, the just-defined ontology, specifying the relationship between model elements and ontology concepts, provides both the semantics already presented in the UML models (i.e. classes, inheritance, aggregation, etc.) and additional semantics (i.e. other class semantics, synonym, etc.). Moreover, rules can be added to generate new facts to the resulting ontologies, and hereafter, an inference engine is employed to derive more facts about the data models. Thus, we can take advantages of the combination of CWM semantics with semantics from ontologies, ensuring semantically structured, integrated ETL process.

## 5 Conclusions

Within the scope of this paper, we focus on a combined use of ontology and a predefined metamodel based on CWM to support the ETL process, in which data integration is specified both at syntactic and semantic levels. The syntactic level deals with metamodels which define the structures and data types of models, whereas the semantic level uses ontologies describing both the semantics of the modelling constructs as well as the semantics of model instances. Thus, our approach can enhance the semantic interoperability in ETL process, improving the reliability of integration process as well as the data quality of the data warehouse.

## 6 Future Work

Clearly, a lot of work remains to be done for demonstrating and fulfilling the benefits outlined in this paper that enable semantic management and interoperability of ETL process. The main challenge is the practical application of this disciplined approach in real world cases and its further tuning to accommodate extra practical problems. For the near future, efforts are currently focused on developing this proposed approach further by defining well-defined rules, syntax and semantics for the metamodel and mapping it to the core ontologies. Thus, our approach can be exploited for developing system that support automated reasoning on CWM-based ETL process, so as to improve the reliability of data integration process and data warehouse system.

## References

- [Dragan 06] Dragan, G., Dragan, D., Vlanan, D. and Bran, S.: Model Driven Architecture and Ontology Development. Springer-Verlag New York, Inc., 2006.
- [Franz, 03] Franz, B., Diego, C., Deborah, L.M., Daniele, N. and Peter, F.P.-S. (eds.): Description logic handbook: theory, implementation, and applications. Cambridge University Press, 2003.
- [Höfferer, 07] Höfferer, P.: Achieving Business Process Model Interoperability Using Metamodels and Ontologies, In Proc. of the 15th European Conference on Information Systems (ECIS2007), Switzerland, June 7-9 2007, p. 1620-1631.

- [Liu, 05] Liu, J., He, K., Li, B., and He, F.: A Perspective of Fusing Ontology and Metamodeling Architecture in Interconnection Environment. In Proc. of the 1st Int. Conf. on Semantics, Knowledge and Grid, November, 2005. SKG. IEEE Computer Society, 6.
- [Luján-Mora, 06] Luján-Mora, S., Trujillo, J., and Song, I. 2006. A UML profile for multidimensional modeling in data warehouses. *Data Knowl. Eng.* 59, 3 Dec. 2006, 725-769.
- [Kimball, 04] Kimball, R. Caserta, J.: *The Data Warehouse ETL Toolkit: Practical Techniques for Extracting, Cleaning, Conforming and Delivering data.* John Wiley & Sons. 2004.
- [Nguyen, 00] Nguyen, T.B., Tjoa, A M., Wagner, R.R.: Conceptual Multidimensional Data Model Based on MetaCube. In Proc. of 1<sup>st</sup> Int. Conf. on Advances in Information Systems (ADVIS'2000), TURKEY, 2000. Lecture Notes in Computer Science (LNCS), Springer, 2000.
- [OMG, 03] OMG (Object Management Group). Common Warehouse Metamodel (CWM) Specification. Version 1.1, Volume 1 (No. formal/03-03-02). 2003.
- [OMG, 07] OMG (Object Management Group). Ontology Definition Metamodel (Fifth Revised Submission). Available at: <http://www.omg.org/docs/ad/06-01-01.pdf>, access: 2007-10-27.
- [Simitsis, 05] Simitsis, A., Vassiliadis, P., Sellis, T. K.: Optimizing ETL Processes in Data Warehouses, ICDE, 2005, pp. 564–575.

# **A Semantic Policy Management Environment for End-Users**

**Joachim Zeiss, Rene Gabner, Anna V. Zhdanova, Sandford Bessler**  
(ftw. Telecommunications Research Center Vienna, Austria  
{zeiss, gabner, zhdanova, bessler}@ftw.at)

**Abstract:** We describe a policy management environment (PME) for the Semantic Web and show its added value compared to existing policy-related developments. In particular, we detail a part of the PME, the policy acquisition tool that enables non-expert users to create and modify semantic policy rules. The implementation of such a policy editor has at its core a semantic reasoner operating on N3 rules and a simple web-based user interface. We describe applications in which PME is used and discuss the feasibility and advantages of ontology-based and community-driven policy management.

**Keywords:** user-generated content, policy web, semantic policies, privacy

**Categories:** M.5, H.3.5

## **1 Introduction**

Nowadays, community-driven Web services and portals such as Facebook, 43things.com, SecondLife, YouTube, LinkedIn and Orkut, a.k.a. Web 2.0 developments are at their popularity peak attracting millions of members. However, the existing Semantic portals and community sites [Davies et al. 2002, Karat et al. 2006, Zhdanova 2008], while collecting large amounts of user-generated content, are highly limited in providing adequate management and sharing control of the submitted data. In particular, the users normally cannot specify to whom the content or service is dedicated or under which circumstances and how it can be used. The inability to dynamically define and apply policies<sup>1</sup> often leads to undesired personal information disclosure, increasing amounts of electronic spam, revenue losses on licensed content and services, and inflexible information management. Due to their complexity and rather narrow scope, typical existing standalone policy solutions or platforms [XACML 2008, P3P 2008] cannot be directly employed by the end users.

A more precise knowledge representation and thus a larger degree of flexibility and adaptability in the actual policy employment can be achieved by deploying Semantic Web technologies [Berners-Lee et al. 2001, Davies et al. 2002] and community-driven ontology management [Zhdanova 2008, Zhdanova and Shvaiko 2006]. At present, Semantic Web and social software technologies are already applied in numerous community environments [Karat et al. 2006]. However, the policies and

---

<sup>1</sup> A policy is “a plan or course of action, as of a government, political party, or business, intended to influence and determine decisions, actions, and other matters” (The Free Online Dictionary).

rules on the light-weight (and often tag-based) social Web have not gained a broad usage yet, largely due to the policy acquisition bottleneck from the non-expert users.

In this paper, we argue that current Semantic social Web can gain substantial benefits from integrating community-driven policy management, i.e., enabling the community members to develop, maintain and share semantic policies.

A community-driven policy management infrastructure has to support developers and users in their efforts to create the Semantic Web policy content, i.e., designing ad-hoc policies on operating on existing ontologies. In practice, adding policy management support to applications will naturally allow more users to create, share and reuse policies on the Web, or contribute to the appearance of the “Web of Trust”, extending the Semantic Web by user-generated rules.

A number of policy construction environments [Karat et al. 2006] and policy frameworks based on mark-up policy description languages such as XACML and P3P [XACML 2008, P3P 2008] have been proposed. However, none of these systems meets all the expectations of policy management for the social Semantic Web: most of these works address only narrowly restricted specific policy management functionality and underestimate the importance of community-driven policy management and shared semantics trends.

The main contributions of the presented work are:

- Definition of a user-driven PME for open, sharable infrastructures such as for Web or mobile services,
- Semantic-based implementation of the PME,
- Identification of showcases for such environment.

The paper is organized as follows. In Section 2, we describe our approach of a policy management for the social Semantic Web. In Section 3, architecture and implementation aspects are presented. Our experience with practicing community-driven policy management use cases is described in Section 4. Section 5 concludes the paper.

## 2 Semantic Policy Management

The following paragraphs describe the basic components of our architecture. The architecture is strongly related to conventional ontology and policy management services [Bonatti et al. 2006, Davies et al. 2002], but is enriched with end-user generated policy acquisition and advanced policy communication. The basic model is that of an open system in which policy rules can be shared, adapted to individual needs and enriched with facts and instance combinations.

A **Policy Storage and Query** component is provided to efficiently store and query parts of policy data and metadata by providing indexing, searching and query facilities for ontologies. In addition to conventional policy management services [Bonatti et al. 2006, Davies et al. 2002], we propose to enrich the existing search and query components with community-generated policy information. This would improve their performance and make the search, reasoning and consistency checking features mature and more attractive to use.

As the users of the environment are generally not bound to a single community or application, they must be able to publish personal and community-related policies in a

multi-accessible way. The current focus in semantic policy storage and querying is thus maintaining distributed repositories with functionalities for aggregation, decomposition and discovery of information in simple ways.

A **Policy Editing** component is introduced for creating and maintaining policies and instance data. The front-end, a user-friendly interface, helps users to easily add and modify policy-like rules on the basis of existing imported ontology classes and properties shared among several users and communities, policies and instances. The back-end consists of a storage and query system. A Policy Editor enables sharable editing for multiple users and tight integration with semantic publishing, delivery and visualization components, allowing the involved parties to observe the evolution of policy settings. These requirements are due to the elevated degree of flexibility required by community-oriented environments as the social Semantic Web and its members to freely evolve schemata, policies and to influence community processes.

A **Policy Versioning** component is introduced to maintain different versions of policy definitions, as communities, content and relationships change over time. The user should be able to easily adapt policies to new scenarios and communities without losing previous definitions. Earlier versions can be reused for definitions of new policies. Also users could experiment with more restricting policy definitions and roll back to previous versions wherever practical. A Policy Versioning component interacts with existing versioning systems like svn [Collins-Sussman et al. 2004] to provide a versioning service to the user. Semantic metadata describes the necessary versioning information inside the policy definition itself.

A **Policy User Profile and Personalization** component is responsible for the users' access to the environment and it connects the policies with the user profiles. At a more advanced level, the component helps to share and communicate policies across the user's profiles, apply policies depending on the user profiles and recommend policies basing on the user profiles. In particular, access and trust policies can be implemented taking into consideration community and social networking information provided by the users [Golbeck et al. 2003].

Our *overall* ontology-based **policy management** approach features: *user-driven policy construction*, meaning that the system extensively assists the users to model the policies correctly (e.g., proactive suggestion of the ontology items that can be combined in a policy, consistency checking for the modelled policy solutions); *policy semantic representation and sharing across communities*, essential for the further extension for the rules layer of the Semantic Web; ontology import and *policy creation on the basis of shared ontologies*, the user is free to input any ontologies he/she likes and define policies on them.

Thus, ontology-based and community-oriented policy management is an advance over a conventional policy management. The advantages are gained by introducing an infrastructure that enables the communities to manage their policies.

### 3 Implementation

The implemented infrastructure is designed as a component for a community Semantic Web portal providing policy management facilities to the community members and managers. The infrastructure is built as a Web-based application using JSON technology [Crockford 2006, JSON 2007] and exploiting a Python version of



Euler [De Roo 2008] for manipulating ontology schemata, instance data and policies in an N3 format [Berners-Lee et al. 2007]. In

Figure 1, the architecture consists of two major blocks, the policy engine and the policy acquisition tool (PAT). Whereas the PAT server interacts with the end-user via a web front end, the policy engine is responsible for the “logical” side of the system, accomplishing integration of external and internal information, reasoning and rule production. The PAT server sends requests to the policy engine whenever the user loads a policy, selects the policy building blocks or saves a policy. The incoming request and the user context are the only input data.

In order to develop policies for a certain application (domain), we need the availability of domain-dependent and domain-independent ontologies. We need as well service support for the portals’ data and metadata, mostly, through publishing services for making human-readable the semantically-enriched data. Non-semantic data from a profile or context management system (e.g., XML structured data) is converted to a N3 format. OWL and RDF data can also be used.

The core component of the architecture is the policy engine (PE), a stateless request/response based server that deals with any kind of requests expressed in N3 [1]. The policy engine has associated a *Decision Space*, a set of files containing N3 triplets as well as rule objects, i.e., parsed N3 statements, kept in memory. The files contain persistent semantic data like ontology definitions, instance data and rules. Volatile semantic data relevant for the current policy request are added to the N3 objects in memory. The *Request Processor* is the part of the PE that extracts data from the request (out of a SIP message, a http GET/POST message or a SMS) and inserts it into the decision space. The policy engine may also extract data from a user profile, user context such as location, or policy data via an additional context interface. The *Reasoner*, the heart of PE, is a N3 rule engine that is invoked with the receipt of a request and uses all semantic data made available in the decision space as reasoning input. The reasoner is based on the python implementation of Euler (backward-chaining, enhanced with Euler path detection).

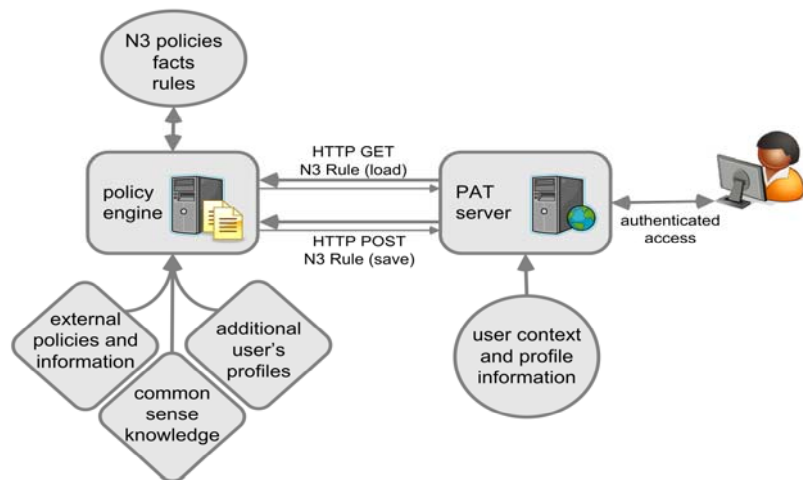


Figure 1: Policy management infrastructure

In the design of the policy acquisition tool (PAT) we have introduced certain *novel techniques*.

The user *interface is dynamically generated* from the ontologies and the instance data. The latter are provided by the end user or deduced by the policy engine based on defined business logic. In addition to ontologies, user profiles and context data are used to assist the user in editing the policies. The implemented user interface is shown in Figure 2.

At the moment PAT offers to combine data according to profile and context ontologies. All semantic information (describing policies and user rules) is encoded in N3 format. As N3 format is triplet-based, the environment's knowledge representation is lightweight and *caters to a straightforward reuse of semantic content in other formats* (e.g., in RDF(S) and OWL).

The tool consists of a web front end that presents a JavaScript enabled user interface and a (PAT) server part that contains the logic of creating and processing a rule. The PAT server queries the policy engine every time it receives an update request from the client (and converts the JSON data received from the client into a valid N3 request and vice versa). This feature enables PAT to *provide the user only with data (subjects, predicates and objects) which are valid* from the ontology perspective to construct a consistent rule. All the rules are acquired from and stored into the policy engine's decision space.

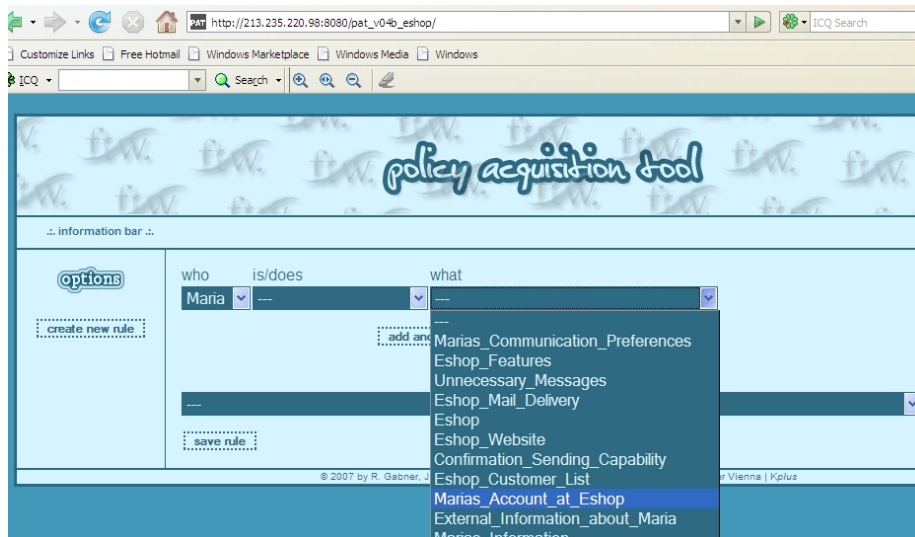


Figure 2: Graphical interface for policy acquisition

## 4 Applications

In this section, we describe applications of the ontology-based PME and our experiences with them.

**Eshop / Internet Shopping:** In this case study, the user of the PME is a manager of an internet shop (Eshop). He/she needs to model the online shop's privacy policies in a policy editor for a group of customers. In the example below, we present such a policy for a typical customer named Maria. Maria regularly shops online, likes special offers and recommendations, but wants to keep sharing of her personal profile information under control. One of the policies; that might be valid for an Eshop and should be known to Maria and its N3 representation are as follows: "*We might receive information about you from other sources and add it to our account information*".

```

Maria a :Customer.
Eshop a :Eshop.
External_Information_about_Maria a :External_Customer_Information.
Marias_Account_at_Eshop a :Eshop_Customer_Account.
{Maria :has Marias_Account_at_Eshop.
Eshop :receives External_Information_about_Maria
}> {External_Information_about_Maria :is_added_to
Marias_Account_at_Eshop}

```

**Mobile Service Marketplace:** The policy management environment is applicable for selling services at the Web or mobile markets. The end user defines the service and sets the service descriptions and as well as policies as the conditions on provisioning or selling the service. The core service descriptions and their model are predefined in an ontology for a micro-service. Only specific areas of service definition can be overwritten or provisioned by the user employing an ontology-based policy management environment.

**Policy-Driven Personal Networks:** In [Zeiss et al. 2007] two or more owners of personal networks decide to interact and form a so-called federation to access the each other data or services. In this scenario our PME is being used to control setup, usage and takedown of personal network federations. It enables the user to maintain privacy and to control the access to resources and services inside his/her network, while opening up towards other private networks to use foreign services securely. A complex task made easy by introducing semantic policies and a user-friendly policy editor.

**Context Aware Services:** In [Bessler and Zeiss 2006] the relationship between user and provider of a context aware service is discussed. The provider needs to protect his/her resources without degrading service functionality. The user in turn is interested in protecting his/her privacy but still wants to offer sufficient context data to obtain personalized service results. Our PME helps both parties to keep the balance of contradicting interests by automating the necessary negotiation between user and provider.

**User Availability:** In [Bessler and Zeiss 2007] an alternative to existing presence and availability architectures is being introduced. In this use case the idea is not to control the delivery of intimate presence or location information. In fact no such information is revealed at all. A decision on if, how and when a user wants to be contacted is being communicated instead of delivering sensitive private data. In this scenario, the PME enables a communication device to automatically decide how, when and by whom a connection can be established based on context data, user profiles, buddy lists and user defined policies.

## 5 Discussion and Conclusions

Summarizing, we see the following value being added by an ontology-based policy management compared to conventional policy practices:

1. **Spreading of policies**, freedom in policy distribution and sharing, annotation of the end users' data and services, easiness in reading other people and organizations' policies; all these are would be difficult without the semantic practices.
2. **Reduction of costs for policy construction**: existing similar policies may be available and easy to reuse elsewhere. For example, most of the internet shops have very similar polices on how to deal with the customer data and they would not need to redefine all the policies from scratch. One could also advance eGovernment visions by provisioning machine readable laws, e.g., on data protection,
3. **Reduction of the mistakes in the user-generated policy modeling** as the system's storage, query and reasoning service as well as sharing of policies within communities act as controllers for policy correctness.
4. **Better awareness of the end users about policies, rules and regulation**: With the suggested system the policies are easily retrieved and presented to the users.

The evaluation of the policy management environment is being done via the user studies, i.e., by placing the system online and letting the volunteers to build policies and/or to give a feedback via a pre-designed questionnaire. Then the users' inputs and feedback is analyzed. The sample scenarios specified in Section 5 are being offered as the evaluation scenes.

Apart from technical and usability issues, the following more socially-oriented questions should be investigated in community-driven policy modeling studies:

- How users share personal data, multiple identities, etc. Initial observations can be drawn from social networking websites (e.g., LinkedIn, Xing, etc.) where users can select whether they share specific type of information with other users;
- Specifying, accumulating and storing arbitrary policies could result in a "policy Wikipedia" provisioning commonsense knowledge rules of what users find right and appropriate, e.g., "do not drink and drive". Such community effort would also have an anthropological effect in enabling observation of which kinds of policies are shared between large communities and which policies are less popular.
- Certain policies vary by countries, cultures and time (e.g., eating any kind of foods using hands could have been acceptable in certain countries in the past, but not in the present). This adds to additional technical challenges in policy versioning, matching and comparison.

We have introduced ontology-based policy management and its benefits. In addition, we have described an implementation supporting ontology-based policy management and discussed its actual and potential applications. As a conclusion, we are convinced that the ontology-based policy management is a highly important concept for services offered in user-centered open environments, such as Web or mobile environments. Also we foresee that implementations of such ontology-based policy management infrastructure will become an essential part of end-user service-oriented environments involving policies.

## Acknowledgements

The Telecommunications Research Center Vienna (ftw.) is supported by the Austrian government and the City of Vienna within the competence center program COMET. This work is partially funded by the IST project Magnet Beyond (<http://www.ist-magnet.org>).

## References

- [Berners-Lee et al. 2007] Berners-Lee, T., Connolly, D., Kagal, L., Scharf, Y., Hendler, J., 2007. "N3Logic: A Logic for the Web", *Journal of Theory and Practice of Logic Programming (TPLP)*, Special Issue on Logic Programming and the Web, 2007.
- [Berners-Lee et al. 2001] Berners-Lee, T., Hendler, J., Lassila, O., 2001. "The Semantic Web". *Scientific American* 284(5), pp. 34-43.
- [Bessler and Zeiss 2006] Bessler, S., Zeiss, J., 2006. „Semantic modelling of policies for context-aware services”, *Wireless World Research Form (WRF17)*, November 2006.
- [Bessler and Zeiss 2007] Bessler, S., Zeiss, J., 2007. „Using Semantic Policies to Reason over User Availability”, *Proc. of 2<sup>nd</sup> Int. Workshop on Personalized Networks (Pernet07)*, 10 August 2007, IEEE Press.
- [Bonatti et al. 2006] Bonatti, P.A., Duma, C., Fuchs, N., Nejdil, W., Olmedilla, D., Peer, J., Shahmehri, N., 2006. "Semantic web policies - a discussion of requirements and research issues", In *3rd European Semantic Web Conference (ESWC)*, 11-14 June 2006, Budva, Montenegro, Springer-Verlag, LNCS 4011, pp. 712-724.
- [Collins-Sussman et al. 2004] Collins-Sussman, B., Fitzpatrick, B.W., Pilato, C.M., 2004. "Version Control with Subversion", o'Reilly 2004.
- [Crockford 2006] Crockford, D., 2006. *The application/json Media Type for JavaScript Object Notation (JSON)*, RFC 2647, July 2006.
- [Davies et al. 2002] Davies, J., Fensel, D., van Harmelen, F. (eds.), 2002. *Towards the Semantic Web: Ontology-Driven Knowledge Management*, John Wiley & Sons.
- [Golbeck et al. 2003] Golbeck, J., Parsia, B., Hendler, J., 2003. "Trust Networks on the Semantic Web", In *Proceedings of Cooperative Intelligent Agents 2003*, Helsinki, Finland.
- [JSON 2007] JSON, 2007. URL: <http://www.json.org>.
- [Karat et al. 2006] Karat, C.-M., Karat, J., Brodie, C., Feng, J., 2006. Evaluating Interfaces for Privacy Policy Rule Authoring, In *Proc. of the Conference on Human Factors in Computing Systems (CHI 2006)*, pp. 83-92.
- [Mika 2007] Mika, P., 2007. *Social Networks and the Semantic Web*. Springer Verlag, 234 p.
- [XACML 2008] OASIS eXtensible Access Control Markup Language (XACML) 2.0, 2008. URL: [http://www.oasis-open.org/committees/tc\\_home.php?wg\\_abbrev=xacml](http://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml).
- [P3P 2008] Platform for Privacy Preferences (P3P) 1.0, 2008. URL: <http://www.w3.org/P3P>.
- [De Roo 2008] De Roo, J., 2008. Euler proof mechanism, 2008. URL: <http://www.agfa.com/w3c/euler>.

[Zeiss et al. 2007] Zeiss, J., Sanchez, L., Bessler, S., 2007. „Policy driven formation of federations between personal networks”, *Proc. of Mobile & Wireless Communications Summit*, July 2007.

[Zhdanova 2008] Zhdanova, A.V., 2008. "Community-driven Ontology Construction in Social Networking Portals", *International Journal on Web Intelligence and Agent Systems*, Vol. 6, No. 1, IOS Press, pp. 93-121 (2008).

[Zhdanova and Shvaiko 2006] Zhdanova, A.V., Shvaiko, P., 2006. "Community-Driven Ontology Matching". In *Proc. of the 3rd European Semantic Web Conference (ESWC'2006)*, 11-14 June 2006, Budva, Montenegro, Springer-Verlag, LNCS 4011, pp. 34-49 (2006).

## A Knowledge Base Approach for Genomics Data Analysis

**Leila Kefi-Khelif**

(INRIA Sophia Antipolis, France  
leila.khelif@inria.sophia.fr)

**Michel Demarchez**

(IMMUNOSEARCH, Grasse, France  
mdemarchez@immunosearch.fr)

**Martine Collard**

(INRIA Sophia Antipolis, France  
University of Nice-Sophia Antipolis, France  
martine.collard@inria.sophia.fr)

**Abstract:** Recent results in genomics and proteomics and new advanced tools for gene expression data analysis with microarrays have produced so huge amounts of heterogeneous data that biologists driving comparative genomic studies face a quite complex task for integrating all relevant information. We present a new framework based on a knowledge base and semantic web techniques in order to store and semantically query a consistent repository of experimental data.

**Key Words:** Genomics, Knowledge base, Ontology, Semantic web technology

**Category:** H.2, H.4, H.3.2

### 1 Introduction

Recent results in genomics and proteomics and new advanced tools for gene expression data analysis with microarrays have led to discovering gene profiles for specific biological processes. Data on gene profiles are now available for the entire scientific community from public databases such as the Gene Express Omnibus<sup>1</sup>(GEO) or the ArrayExpress<sup>2</sup> repository. So it becomes conceivable for a biologist to take advantage of this whole set of responses in order to compare them and characterize the underlying biological mechanisms. Nevertheless, biologists that are interested in studying these data and finding novel knowledge from them face a very complex task. Navigating into huge amounts of data stored in these public repositories is such a tedious task that they lead restricted studies and make limited conclusions. Indeed, one can observe that publications dedicated to gene expression data analysis generally focus on the hundred first differentially expressed genes among thousands of a whole genome and they deeply discuss on

<sup>1</sup> <http://www.ncbi.nlm.nih.gov/geo/>.

<sup>2</sup> <http://www.ebi.ac.uk/microarray-as/aer/>.

ten of them only. In order to highlight similar and specific biological responses to a particular biological test, it seems promising to transversally analyze the largest set of related data. A meta-analysis on multiple independent microarray data sets may provide more comprehensive view for intervalating previous results and comparing to novel analyses. In the past few years some attempts were made on meta-analyses and focused on differentially expressed genes or on co-expressed genes. Moreau and al. [Moreau et al., 2003] discussed different issues for an efficient integration of microarray data. [Hong and Breitling, 2008] evaluated three statistical methods for integrating different microarray data sets and concluded that meta-analyses may be powerful but have to be led carefully. Indeed a critical point is to combine directly data sets derived from different experimental processes. Our approach for a better insight into huge amounts of independent data sets, is quite different. Our proposition is to build a kind of warehouse for storing expression data at a more synthetic level. We have designed a specific framework organized on two main tools: *a knowledge base* that structures and stores refined information on experiments and *an intelligent search engine* for easy navigation into this knowledge. The knowledge base is expected to include correlated information on experiments such as refined expression data, descriptive data on scientific publications and background knowledge of biologists. In this paper, we present the overall approach of the *AMI (Analysis Memory for Immunosearch)* project which aims at providing the scientist user with semi-automatic tools facilitating navigation and comparative analyses into a whole set of comparable experiments on a particular biological process. This work is done in collaboration with the Immunosearch organization<sup>3</sup> whose projects focus on human biological responses to chemicals. The system should allow to confront novel analyses to previous comparable results available in public repositories in order to identify reliable gene signature of biological responses to a given product. In a first stage AMI is devoted to human skin biological reactions only. Technical solutions in the AMI knowledge base and its search engine take mainly advantage of semantic web techniques such as semantic annotation languages and underlying ontologies in order to integrate heterogeneous knowledge sources, and query them in an intelligent way. The following is organised in four sections: Section 2 gives a global overview of AMI, Section 3 is devoted to the AMI knowledge base and details how its semantic annotations and ontologies are exploited, in Section 4 we demonstrate the benefit of the semantic search through examples and we conclude in Section 5.

<sup>3</sup> <http://www.immunosearch.fr>.



## 2 AMI Overview

A central point in our solution is to build the knowledge base on semantic annotations. Each relevant source of available information on a genomic experiment is represented as a set of semantic annotations. A semantic search engine relying on semantic ontological links is a powerful tool which may retrieve interesting approximate answers to a query as well as inferred knowledge deduced from logical rule annotations. The AMI knowledge base consists on three underlying ontologies and four sets of semantic annotations. As presented in Figure 1, AMI provides the biologist with three main tools: *ANNOTATER*, *ADVANCED MINER* and *SEMANTIC SEARCH*. The system takes input data describing experiments either from public repositories or from new experiments driven specifically by the system user. Semantic annotations represent different kinds of information: (i) background knowledge of biologists which has to be explicitly stated through logical facts and rules, (ii) scientific publications selected by the biologist into public microarray data repositories like GEO, ArrayExpress or PUBMED<sup>4</sup>, (iii) descriptive information on experiments (laboratory, microarray) and conditions (tissue, treatment), (iv) synthetic data obtained from numeric raw expression data by processing transformation, statistical and data mining tools. The *ANNOTATER* tool takes each kind of available information as inputs and generates semantic annotations. It produces annotations on textual sources as scientific publications by extracting them from the text. It annotates data resulting from statistical and mining operations on raw expression data provided by the *ADVANCED MINER*. Semantic annotations include expressions of the expert background knowledge that the biologist clarifies through dialogs and graphical interfaces. The *ADVANCED MINER* tool allow the users to process data transformations for further combined meta-analysis and to run statistical and data mining tasks such as differentially expressed gene analysis or co-expressed genes clustering on relevant subspaces of the data set. The *SEMANTIC SEARCH* tool is invoked to navigate into the knowledge base and retrieve experiments, conditions or genes according more or less complex criteria. This tool generates either exact answers and approximate answers extracted according similarity links in ontologies or deduced answers obtained by logic inference rules.

## 3 Knowledge base

Ontologies and annotations in AMI are expressed in RDFS and RDF<sup>5</sup> languages as recommended by the World Wide Web Consortium (W3C)<sup>6</sup>, respectively to represent light ontologies and to describe web resources using ontology-based

<sup>4</sup> <http://www.ncbi.nlm.nih.gov/PubMed/>.

<sup>5</sup> <http://www.w3.org/RDF/>.

<sup>6</sup> <http://www.w3.org/>.

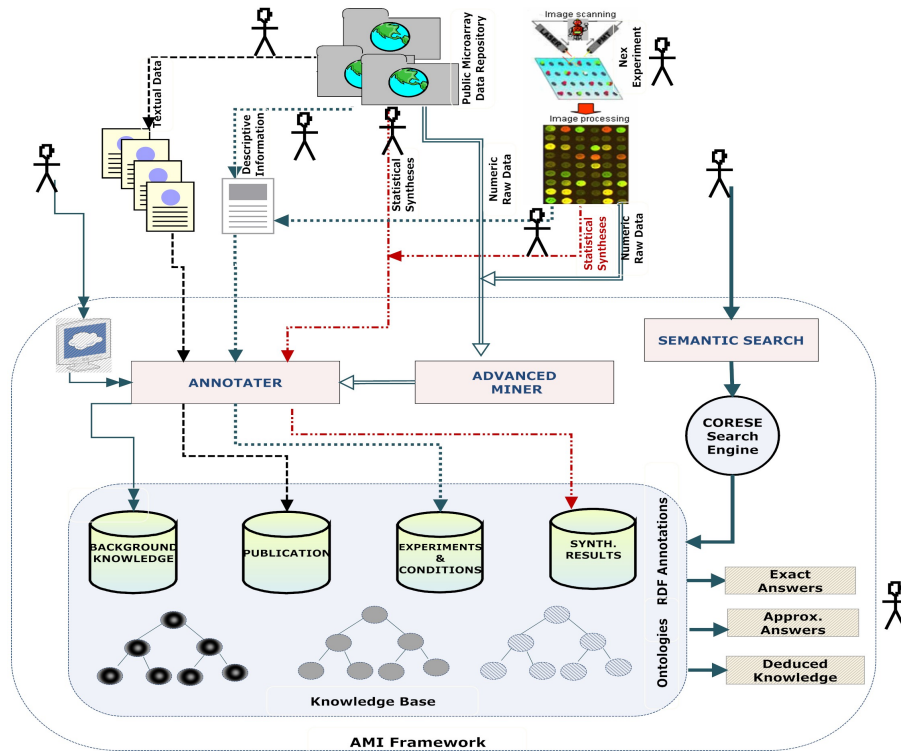


Figure 1: Global Overview of AMI framework

semantic annotations. This choice enables to use the semantic search engine CORESE [Corby et al., 2006] as explained in Section 4.

### 3.1 Ontologies

Ontologies provide an organizational framework of concepts and a system of hierarchical and associative relationships of the domain. In addition to the possibility of reuse and sharing allowed by ontologies, the formal structure coupled with hierarchies of concepts and relations between concepts offers the opportunity to draw complex inferences and reasoning. In AMI, we chose to reuse the existing ontology *MeatOnto* [Khelif et al., 2007] in order to annotate biomedical literature resources, and to develop two new ontologies: *GEOnto* for experiments and conditions, and *GMineOnto* to annotate statistical and mining results on numeric experimental data. Both ontologies will be implemented in RDF and RDFS loaded into CORESE. *MeatOnto* is based on two sub-ontologies: UMLS(Unified Medical Language System) semantic network (which integrates

the Gene Ontology<sup>7</sup>) enriched by more specific relations to describe the biomedical domain, and DocOnto to describe metadata about scientific publications and to link documents to UMLS concepts. *GEOnto* (*Gene Experiment Ontology*) is devoted to concepts related to an overall microarray expression experiment (contributors, pubmedId, keywords, general description...) and its experimental conditions (sample, treatment, subject...). While ontologies describing experiments are already available (MGED Ontology) [Stoeckert et al., 2002] and OBI (Ontology for Biomedical Investigations) [Smith et al., 2007], we choose to propose a dedicated ontology which integrates original concepts specifically relevant in our context. In fact, OBI and MGED Ontology provides models for the design of an investigation (protocols, instrumentation, material, data generated, analysis type) while GEOnto allows the description of its experimental conditions. Some of the MGED ontology concepts are included in GEOnto but they are differently structured in order to support the annotation of the experiments in our context. Some concepts in GEOnto cover general biology fields (in vivo, inductor, subject, sample, etc.) and others are specific to a particular field. In a first step, as presented in 2, we limit it to dermatology (skin, eczema, contact dermatitis, etc.) but GEOnto can be extended towards other biologic fields. To build GEOnto, we rely on (i) a corpora of experiment descriptions used to pick out candidate terms, (ii) biologists who help us to structure the concepts and validate the proposed ontology and (iii) existing ontologies (UMLS and OntoDerm<sup>8</sup>) to extract specific concepts (for example, UMLS to enrich the concept "cell of the epidermis" and OntoDerm to enrich the concept "disease of the skin"). *GMineOnto* provides concepts for the description of basic statistical analysis and more complex mining processes on expression data. Gene expression data are stored in two different modes: (i) refined gene expression value in a given condition, (ii) gene regulation (up, down or none) behaviour in a given condition compared to another condition. Figure 2 and Figure 3 give respectively fragments of GEOnto and GMineOnto ontologies.

### 3.2 Annotations

Annotations on a resource attach the most relevant descriptive information to it. In this section, we focus on the AMI approach for annotating experiments. We consider here two types of experiments: so-called "public" experiments selected by biologists from the public repositories and so-called "local" experiments led directly by the biologist. For instance, if we consider a public experiment selected from the public repository GEO, we annotate the MINiML formatted family file which is an XML document relying on the MIAME formalism [Brazma et al., 2001]. MINiML assumes only basic relations between objects:

<sup>7</sup> <http://www.geneontology.org/>.

<sup>8</sup> <http://gulfdactor.net/ontoderm/>.

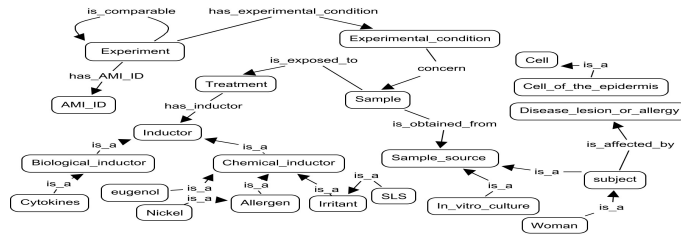


Figure 2: Fragment of GEOnto

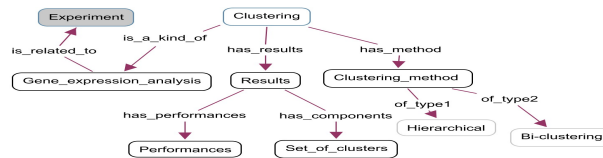


Figure 3: Fragment of GMineOnto

Platform, Sample (e.g., hybridization), and Series (experiment). The annotation process is semi-automatic. Instances of GEOnto concepts are detected in the document, some instances are directly used to generate the annotation describing the experiment (exp. contributors, pubmedID, keywords, condition titles), and others are proposed to the biologist who selects the more relevant instance for each condition (exp. time point, treatment, subject). For local experiments, the biologist has to give a structured description of the experiment and its conditions. In both cases, he uses an interactive interface. The background knowledge of biologists may be embedded into annotations on experiments too. For instance, information about *comparable* experiments can be stated by biologists solely. The RDF code below provides partly an example: The experiment annotated has the accession number GSE6281 in the GEO repository. The pubmedID 17597826 references the published article describing this experiment: "Gene expression time-course in the human skin during elicitation of allergic contact dermatitis". The experiment is declared to be comparable to experiment AMI\_2008\_125. It concerns a patch test with 5% nickel sulfate taken from a nickel allergic woman (age range 33-49). The patch test was exposed for 48h immediately followed by a skin biopsy.

```

<geo:Experiment rdf:about="GSE6281">
  <geo:has_AMI_ID>AMI_2008_41</go:has_AMI_ID>
  <geo:has_PMID>17597826</go:has_PMID>
  <geo:has_title>Gene Expression time_course in the human skin ...

```

```

</geo:has_title>
<geo:is_comparable rdf:resource="#AMI_2008_125"/> ...
<geo:has_experimental_condition rdf:resource="#GSM144432"/>...
<geo:Experimental_condition rdf:ID="GSM144432">
<geo:concern><rdf:Description rdf:about="#GSM144332_BioSample">
<geo:has_type rdf:resource="#Skin"/>
<geo:is_analysed_at rdf:resource="#0h"/>
<geo:is_obtained_from
      rdf:resource="#1_nickel-allergic_Woman_33-49"/>
<geo:is_exposed_to rdf:resource="#Nickel5_48h_patch"/>
</rdf:Description></geo:concern>...</geo:Experimental_condition>
...
<geo:Treatment rdf:ID="Nickel5_48h_patch">
<geo:has_dose>5%</geo:has_dose>
<geo:is_exposed_for>48h</geo:is_exposed_for>
<geo:has_delivery_method rdf:resource="#Patch_test"/>
</geo:Treatment>... </geo:Experiment>

```

#### 4 Semantic search

AMI SEMANTIC SEARCH tool uses the semantic search engine CORESE [Corby et al., 2006] which supports navigation and reasoning on a whole base of annotations taking into account concept and relation hierarchies defined into ontologies. In addition, CORESE allows defining logic rules which extend basic annotations. The benefit for AMI is to provide search capacities on its knowledge base built from different heterogeneous sources (publications, gene expression data analyses, domain knowledge). CORESE interprets SPARQL<sup>9</sup> queries as sets of RDF triples with variables. Let us consider the SPARQL query presented below to retrieve all experimental conditions where the sample was exposed to a nickel patch and where the genes  $IL1\beta$  and  $TNF\alpha$  are highly expressed.

```

SELECT MORE ?c    WHERE {
?c    rdf:type      geo: Experimental_condition
?c    geo:concern   ?s
?s    is_exposed_to ?treat
?treat geo:has_inductor geo:Nickel
?treat geo:has_delivery_method geo: Patch_test
?g1   rdf:type      umls:Gene_or_Genome
?g1   go:name       ?n1 filter(regex(?n1, '^ IL1 '))
?g2   rdf:type      umls:Gene_or_Genome

```

<sup>9</sup> <http://www.w3.org/TR/rdf-sparql-query/>.

```
?g2 m:name ?n2 filter(regex(?n2, '^ TNFa '))
?g1 gmo:is_highly_expressed_in ?c
?g2 gmo:is_highly_expressed_in ?c}
```

The *MORE* keyword in the query *SELECT* clause enables to ask for an approximate answer. An approximate search for *a sample exposed to Nickel* can retrieve *a sample exposed to eugenol* since *eugenol* is defined as a very closed concept in the GEOnto ontology. A similar approximate search to retrieve genes involved in the same cluster as *IL1 $\beta$*  and obtained by hierarchical clustering method on comparable experiments would produce results derived from bi-clustering method too since hierarchical clustering and bi-clustering are very close concepts in the GMineOnto ontology. CORESE rule language provides an inference mechanism to deduce new facts from declared annotations. Thus inferring rules on the annotation base reduces silence in the information retrieval (IR) phase. In AMI, rules are a good mean to reflect background knowledge. For instance the following rule: *If the sample studied in an experimental condition, is taken from a subject affected by psoriasis, then we can consider this condition as using the IL22 inductor* may be coded by the following lines:

```
IF          ?c rdf:type      geo: Experimental_condition
           ?c geo:concern  ?s
           ?s is_obtained_from ?subj
           ? subj is_affected_by geo:psoriasis
THEN       ?s geo:is_exposed_to ?t
           ?t geo:has_inductor ?geo:IL22
```

In AMI, the rule inference mechanism will provide the system with much more abilities to assist the biologist in exploring the huge information space. For instance, if the previous rule is inferred, a query asking for *all experimental conditions where this condition is using the IL22 inductor will automatically suggest extended answers with subject affected by psoriasis* avoiding a tedious manual search on well known topics closed to IL22 inductor.

## 5 Conclusions and Future Work

In this paper we have introduced the AMI designed to offer an easy-to-use and customized environment for assisting the biologist in comparative genomic studies. The main originality is to offer the ability to take advantage of most public available information about genomics experiments through automatic and semi-automatic tools. We have highlighted AMI originality relying on semantic web techniques such as ontologies, RDF annotations and semantic search engine. AMI is in its preliminary development phase which focused on the ANNOTATER tool. Further works will consist partly on solutions devoted to collect all heterogeneous data in order to drive real scale tests on the system.

## References

- [Brazma et al., 2001] Brazma, A., Hingamp, P., Quackenbush, J., Sherlock, G., Spellman, P., Stoeckert, C., Aach, J., Ansorge, W., Ball, C. A., Causton, H. C., Gaasterland, T., Glenisson, P., Holstege, F. C., Kim, I. F., Markowitz, V., Matese, J. C., Parkinson, H., Robinson, A., Sarkans, U., Schulze-Kremer, S., Stewart, J., Taylor, R., Vilo, J., and Vingron, M. (2001). Minimum information about a microarray experiment (miame)-toward standards for microarray data. *Nat Genet*, 29(4):365–71.
- [Corby et al., 2006] Corby, O., Dieng-Kuntz, R., Faron-Zucker, C., and Gandon, F. (2006). Searching the semantic web: Approximate query processing based on ontologies. *IEEE Intelligent Systems*, 21(1):20–27.
- [Hong and Breitling, 2008] Hong, F. and Breitling, R. (2008). A comparison of meta-analysis methods for detecting differentially expressed genes in microarray experiments. *Bioinformatics*, 24(3):374–382.
- [Khelif et al., 2007] Khelif, K., Dieng-Kuntz, R. ., and Barbry, B. (2007). An ontology-based approach to support text mining and information retrieval in the biological domain. *J. UCS*, 13(12):1881–1907.
- [Moreau et al., 2003] Moreau, Y., Aerts, S., Moor1, B., Strooper, B., and Dabrowski, M. (2003). Comparison and meta-analysis of microarray data: from the bench to the computer desk. *Trends Genet.*, 19:570–7.
- [Smith et al., 2007] Smith, B., Ashburner, M., Rosse, C., Bard, J., Bug, W., Ceusters, W., Goldberg, L. J., Eilbeck, K., Ireland, A., Mungall, C. J., Leontis, N., Rocca-Serra, P., Ruttenberg, A., Sansone, S. A., Scheuermann, R. H., Shah, N., Whetzel, P. L., and Lewis, S. (2007). The obo foundry: Coordinated evolution of ontologies to support biomedical data integration. *Nat Biotechnol*, 25(11):1251–5.
- [Stoeckert et al., 2002] Stoeckert, C., Causton, H., and Ball, C. (2002). Microarray databases: standards and ontologies. *Nature Genetics*, 32:469 – 473.

## **GRISINO - a Semantic Web services, Grid computing and Intelligent Objects integrated infrastructure**

**Tobias Bürger, Ioan Toma, Omair Shafiq**  
(Semantic Technology Institute - STI Innsbruck,  
University of Innsbruck, Austria  
firstname.lastname@sti2.at)

**Daniel Dögl**  
(Uma Information Technology GmbH, Vienna, Austria  
daniel.doegl@uma.at)

**Andreas Gruber**  
(Salzburg Research Forschungsgesellschaft mbH, Salzburg, Austria  
andreas.gruber@salzburgresearch.at)

**Abstract:** Existing information, knowledge and content infrastructures are currently facing challenging problems in terms of scalability, management and integration of various content and services. The latest technology trends, including Semantic Web Services, Grid computing and Intelligent Content Objects provide the technological means to address parts of the previously mentioned problems. A combination of the three technologies could provide a sound technological foundation to build scalable infrastructures that provide highly automated support in fulfilling user's goals. This paper introduces GRISINO, an integrated infrastructure for Semantic Web Services, Intelligent Content Objects and Grid computing, which may serve as a foundation for next generation distributed applications.

**Key Words:** Semantic Web Services, Grid Computing, Intelligent Content Objects, Semantic Web

**Category:** H.3.1, H.3.2, H.3.3, H.3.7

### **1 Introduction**

GRISINO<sup>1</sup> [Toma et al. 2006] investigates the use of semantic content models in semantically enhanced service oriented architectures by combining three technology strands: Semantic Web Services (SWS) [Fensel and Bussler, 2002], Knowledge Content Objects (KCO) [Behrendt et al. 2006] and Grid Computing [Foster and Kesselmann 1999]. By that, GRISINO aims at defining and realizing intelligent and dynamic business processes based on dynamic service discovery and the internal state of complex objects. The main output of the project is a test bed for experimentation with complex processes and complex objects that

---

<sup>1</sup> Austrian FIT-IT project Grid Semantics and Intelligent Objects (GRISINO), <http://www.grisino.at>



takes into account user requirements and fulfils them by dynamically integrating the three underlying technologies. For this testbed, advanced prototypes of each of the technology strands are combined:

- The Web Service Modelling Ontology (WSMO) [Roman et al. 2005], the Web Service Modelling Language (WSML) <sup>2</sup> and the Web Service Modelling Execution Environment (WSMX) <sup>3</sup> as a framework for the description and execution of SWS,
- KCOs as a model for the unit of value for content to be exchanged between services, together with its management framework, the Knowledge Content Carrier Architecture (KCCA).
- The Globus toolkit<sup>4</sup> as an existing Grid infrastructure.

In this paper we will detail the main results of the GRISINO project: its architecture (section 2) and the core parts of the architecture which realize the integration of the three technologies, i.e. a set of transformers between the protocol and description standards used (section 3 and 4). Furthermore, we provide details about the proof of concept implementation which serves to demonstrate the functionality and interoperability within the GRISINO testbed in section 5.

## 2 GRISINO Architecture

The GRISINO system architecture as shown in figure 1 provides a set of APIs and an implementation of these APIs to ease the handling and development of applications which intend to use the three technologies together:

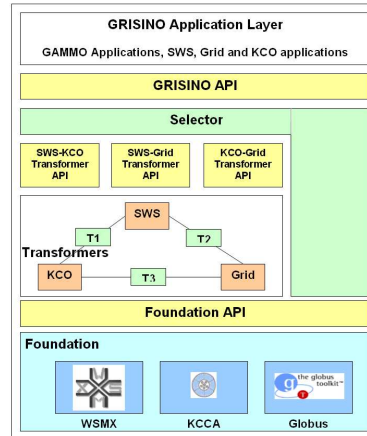
- the *GRISINO API* which gives application developers easy access to the combined functionality of the three technologies.
- the *Transformer API* including protocol transformations,
- the *Selector API* issuing calls to transformers or the Foundational API, and
- the *Foundational API*, an abstraction of the API's of the core technologies.

Most notably the GRISINO system architecture includes extensions to the core components that enable communication between the technologies. These include (i) an extension of WSMX for the interpretation of KCOs, (ii) a semantic layer for services offered by KCCA to enable their discovery and (iii) an extension of the Globus toolkit which extends Globus with a semantic layer in order to handle Grid services like other SWS. The GRISINO system architecture inte-

<sup>2</sup> <http://www.wsmo.org/wsm/>

<sup>3</sup> <http://www.wsmx.org>

<sup>4</sup> <http://www.globus.org/toolkit/>



**Figure 1:** GRISINO System Architecture

grates specific SWS and Grid solutions because of the existence of a wide variety of different and diverse approaches: We based our efforts on WSMO and WSMX as its execution platform because of being well supported by an active research community to handle SWS. Furthermore we are using the Globus Toolkit as being the most widely used Grid computing toolkit which is fully compatible with the OGSA<sup>5</sup> - and Web Service Resource Framework (WSRF) specifications<sup>6</sup>.

The SWS and GRID integration in particular includes an extension of a SWS infrastructure for modelling Grid Services and resources on the Grid in order to realize the vision of the Semantic Grid. Benefits of this integration include:

- Resources on the Grid may profit from machine reasoning services in order to increase the degree of accuracy of finding the right resources.
- The background knowledge and vocabulary of a Grid middleware component may be captured using ontologies. Metadata can be used to label Grid resources and entities with concepts, e.g. for describing a data file in terms of the application domain in which it is used.
- Rules and classification-based reasoning mechanisms could be used to generate new metadata from existing metadata, for example describing the rules for membership of a virtual organization and reasoning that a potential member's credentials are satisfactory for using the VO resources.
- Activities like Grid Service discovery or negotiation of service level agreements can be potentially enhanced using the functionalities provided by

<sup>5</sup> <http://www.globus.org/ogsa/>

<sup>6</sup> <http://www.globus.org/wsrf/>

Semantic Web Service technologies.

- Searches / discovery of SWS can be seamlessly extended to Grid Services.

Benefits of the integration of SWS and KCO/KCCA include:

- Goal-based Web service execution can be based on the various kinds of information which is modelled in so called semantic facets inside KCOs; e.g. to search for a KCO that matches a certain licensing scheme.
- Choreography of Web services can be based on facet information,
- Plans that describe how to handle content and which are modelled inside a KCO can be automatically executed by using SWS or Grid services.

The following section will provide further details about two of the three major aspects of the integration, i.e. the integration of SWS and Grid, as well as the integration of SWS and KCOs.

### 3 SWS–Grid Transformer

The main task of this transformer is the realization of the link between SWS based systems and Grid Computing systems. Our approach was to extend and refactor an existing SWS solution (WSMO/L/X) with Grid concepts in order to address Grid related requirements. The resulting modeling framework for Semantic Grid enriches the OGSA with semantics by providing a Grid Service Modeling Ontology (GSMO)<sup>7</sup> as an extended version of WSMO. GSMO has six major top level entities which were either newly added to the WSMO conceptual model, are refinements of original entities or are entities which are inherited from the WSMO model:

- *Job* represents the functionality requested, specified in terms of what has to be done, what are the resources needed. A Job is fulfilled by executing one or more Grid Services. Ontologies can be used as domain terminology to describe the relevant aspects. Job as one of the top level entities of GSMO is adapted from WSMO Goals and is taken in GSMO as its extended version.
- *Ontologies* provide the terminology used by other GSMO elements to describe the relevant aspects of a domain. This element has been inherited from the WSMO top level entity as Ontologies.
- *Grid Service* describes the computational entity providing access to physical resources that actually perform the core Grid tasks. These descriptions comprise the capabilities, interfaces and internal working of the Grid Service. All

<sup>7</sup> <http://www.gsmo.org/>

these aspects of a Web Service are described using the terminology defined by the ontologies. The Grid Service top level entity has been adapted from WSMO's Web Services as its top level entity.

- *Mediators* describe elements that overcome interoperability problems between different WSMO elements. Due to the fact that GSMO is based on WSMO, it will be used to overcome any heterogeneity issues between different GSMO elements. Mediators resolve mismatches between different used terminologies (data level); communicate mismatches between Grid services (protocol level) and on the level of combining Grid Services and Jobs (process level).
- *Resources* describe the physical resources on the Grid which can be further classified into computing resources and storage resources. These computation- and storage-resources are key elements of the underlying Grid.
- *The Virtual Organization* element describes any combination of different physical resources and Grid Services formed as virtual organizations on the Grid. This element will help in automated virtual organization formation and management.

Based on the proposed conceptual model for Semantic Grid services, a new language called GSML (Grid Service Modelling Language) was developed that inherits the syntax and semantics of the WSML language and adds a set of additional constructs reflecting the GSMO model. Last but not least an extension of the Web Service Modeling Execution Environment (WSMX), called Grid Service Modeling Execution Environment (GSMX) has been proposed. More details about the conceptual model, the language and the new execution environment are available in [Shafiq and Toma 2006].

#### 4 SWS-KCO Transformer

The main task of this transformer is the realization of the link between knowledge content based systems (resp. the KCCA system) and Semantic Web Service based systems (resp. WSMX).

The main intention of this integration is to use the information stored inside KCOs for service discovery and plan execution, e.g. to automatically negotiate or to automatically enrich content and knowledge about that content during the execution of web based workflows like e.g. a document-based business process or workflow to index and enrich documents with additional knowledge. In order to do so, WSMX needs to be able to interpret KCOs and the services offered by KCCA need to be able to communicate with the other services offered by the GRISINO system. Our approach to integrate existing KCO / KCCA technology with the SWS/Grid technologies in the GRISINO system was twofold:

1. Metadata descriptions that are contained inside KCOs are translated into WSMO descriptions in order to be useable for service discovery and ranking.
2. The KCCA system is wrapped with Web service descriptions that describe its invoke-able functionality. These descriptions are further semantically described.

At the center of this integration was the investigation of the ontology of plans [Gangemi et al. 2004] as well as the “Description and Situation” modules embedded in the OWL DL 397<sup>8</sup> version of the foundational ontology DOLCE. Similar work has been reported in [Belecheanu 2007] or [Mika et al. 2004]. In GRISINO we described all relevant concepts of DDPO which are used in a KCO for their mapping onto WSMO elements. These translations comprise for the following elements which are part of the community facet of the KCO. Plans and goals (as descriptions of the intended workflow and the desired result) are integrated into a WSMO description as participating ontology, whereas:

- Tasks and their sequencing structure are translated into ASM statements [Stärk et al. 2001] to be used within the choreography element of the WSMO description.
- Functional Roles and Parameters are used to identify the objects of the regarding service descriptions.
- The pre-condition and the post-condition property can be used to describe outputs of tasks and their inputs

The rationale of this transformer and its relations to Semantic Business Process Management are presented in more detail in [Bürger 2007].

## 5 Use Case Example

In order to demonstrate the functionality of the integration and the interoperability between the technologies in the GRISINO test bed, a semantic search application is designed that realizes a scalable, flexible and customizable search application generator that enables knowledge-based search in unstructured text. The search applications generated are customized and tailored to specific needs expressed by end users. The search applications include very specialized knowledge about a particular domain (e.g. football in the 19<sup>th</sup> century), collected from different knowledge bases and consolidated into one index to provide a single point of access.

To achieve this, a number of processing services deployed on the grid, are tied together to selectively collect, index and annotate content according to different

---

<sup>8</sup> [http://www.loa-cnr.it/ontologies/DLP\\_397.owl](http://www.loa-cnr.it/ontologies/DLP_397.owl)

knowledge bases and to generate custom search applications according to a users' input. The foundation of the users' input is his/her knowledge background or special interests. In particular the search application generator decomposes the user input (e.g. data sources of interest, specific keywords or entities considered important, etc.), into different sub goals which are used to consider different service providers for enriching the initial input. It queries these services to ask for related terms and entities, as well as authoritative information sources, such as popular websites according to the topic of interest. Using additional services, such as clustering services, the collected documents are then indexed and deployed for the use by the end user.

The goal of the scenario is amongst others to exploit as much of the GRISINO functionality as possible, e.g. to select services based on plans modelled inside KCOs or based on document types, and to parallelise indexing on the Grid. The underlying GRISINO infrastructure enables automation of the whole process of putting together the custom search application by using a number of different services from different service providers and bundling its output into a coherent application that combines knowledge and functionality from different sources. This reflects the particular and very common situation in which both knowledge found in all kinds of knowledge bases and specific skills encapsulated in special technical functionality is not found within one organization or provided by a specific technology provider, but is spread over a greater number of specialized organizations. While the benefit for the user obviously is a richer output informed by knowledge of a number of authoritative service providers, this model allows the commercial aspect of contributing specialized services as input to an open service mix by selling functionality and/or encapsulated knowledge bundled into one coherent service.

## **6 Conclusion and Future Work**

The GRISINO project brought forward the integration of three distinct technologies as detailed in this paper. Two major sub-results of GRISINO are a new approach to realize the Semantic Grid which has been the goal of the SWS - Grid transformer and the possibility to use self-descriptions of documents for dynamic SWS discovery in order to automate and execute specific tasks. Regarding the first result, we have followed a novel and previously unexplored approach. More precisely we started from a SWS system (i.e. WSMO/L/X) and we added Grid specific features, transforming the SWS system into a SWS-Grid system. Furthermore we support the integration of pure Grid systems such as Globus. The second result, namely SWS discovery in order to automate and execute specific tasks, might be applied in document processing, multimedia content adaptation or other similar scenarios. The semantic search application genera-

tor implemented as a proof-of-concept, shows the added value of the GRISINO system both for service providers as well as for end users.

The evaluation of the system will be done once the application scenario has been fully set-up. Experiences so far have shown that the complexity of the generic model for knowledge content objects as put forward in other projects could hinder the adoption of the model in practice. Therefore we decided to develop a more lightweight model with reduced complexity.

### Acknowledgements

The reported work is funded by the Austrian FIT-IT (Forschung, Innovation, Technologie - Informationstechnologie) programme under the project GRISINO.

### References

- [Behrendt et al. 2006] Behrendt, W., Arora, N., Brger, T., Westenthaler, R.: “A Management System for Distributed Knowledge and Content Objects”; Proc. of AXMEDIS '06 (2006).
- [Belecheanu 2007] Belecheanu, R. et al.: “Business Process Ontology Framework”; SUPER Deliverable 1.1 (May 2007).
- [Bürger 2007] Bürger, T.: “Putting Intelligence into Documents”; Proc. of the 1<sup>st</sup> European Workshop on Semantic Business Process Management (2007).
- [Fensel and Bussler, 2002] Fensel, D. and Bussler, C.: “The Web Service Modeling Framework WSMF”; Electronic Commerce Research and Applications 1, 2 (Apr. 1991) 127-160.
- [Foster and Kesselmann 1999] Foster, I. and Kesselman, C.: “The Grid: Blueprint for a New Computing Infrastructure”; Morgan Kaufmann (1999)
- [Gangemi et al. 2004] Gangemi, A., Borgo, S., Catenacci, C., Lehmann, J. : “Task Taxonomies for Knowledge Content”; METOKIS Deliverable D07 (2004) [http://metokis.salzburgresearch.at/files/deliverables/metokis\\\_d07\\\_task\\\_taxonomies\\\_final.pdf](http://metokis.salzburgresearch.at/files/deliverables/metokis\_d07\_task\_taxonomies\_final.pdf)
- [Mika et al. 2004] Mika, P., Oberle, D., Gangemi, A., Sabou, M.: “Foundations for Service Ontologies: Aligning OWL-S to DOLCE”; Proc. of the 13<sup>th</sup> Int. World Wide Web Conf. (WWW2004), ACM Press (2004)
- [Roman et al. 2005] Roman, D., Lausen, H. (Ed.): “Web service modeling ontology (WSMO)”; Working Draft D2v1.2, WSMO (2005) <http://www.wsmo.org/TR/d2/v1.2/>
- [Shafiq and Toma 2006] Shafiq, O. and Toma, I.: “Towards semantically enabled Grid infrastructure”; Proc. of the 2<sup>nd</sup> Austria Grid Symposium, Innsbruck, Austria, September 21-23, 2006 (2006)
- [Toma et al. 2006] Toma, I., Bürger, T., Shafiq, O., Dögl, D., Behrendt, W., Fensel, D.: “GRISINO: Combining Semantic Web Services, Intelligent Content Objects and Grid computing”; Proc. of ESscience '06 (2006).
- [Stärk et al. 2001] Stärk, R., Schmid, J., Börger, E.: “Java and the Java Virtual Machine. Definition, Verification, Validation.: Definition, Verification, Validation”; Springer, Berlin (2001)

## **Pervasive Service Discovery: mTableaux Mobile Reasoning**

**Luke Steller**

(Monash University, Melbourne, Australia  
laste4@student.monash.edu.au)

**Shonali Krishnaswamy**

(Monash University, Melbourne, Australia  
Shonali.Krishnaswamy@infotech.monash.edu.au)

**Abstract:** Given the ever increasing availability of mobile devices and web-based services which are available to them, pervasive service discovery architectures are required, which effectively manage additional challenges. These challenges include finding relevant services rapidly while facing constrained computational resources of these devices in a dynamic/changing context. The key to improving relevance of discovered services is to leverage the Semantic Web technologies. However, reasoning engines used for semantic service discovery are resource-intensive and therefore not suitable for mobile environments. This mandates the development of optimisation strategies which enable mobile reasoning on resource constrained devices. In this paper, we present an overview of our mTableaux algorithm for enabling cost-efficient and optimised semantic reasoning to support pervasive service discovery. We also provide comparative evaluations with other semantic reasoners to demonstrate the improved performance of our engine.

**Keywords:** Pervasive Services, Service Discovery, Semantic Reasoning

**Categories:** H.1.m, H.2.m

### **1 Introduction**

Studies such as [Roto, 05] have established that mobile users typically have a tolerance threshold of about 5 to 15 seconds in terms of response time, before their attention shifts elsewhere, depending on their environment. Thus, service discovery architectures that operate in mobile environments must cope with the very significant challenges of not merely finding relevant services, but being able to do so rapidly in a highly dynamic and varying context.

The limitations of syntactic, string-based matching for web service discovery coupled with the emergence of the semantic web implies that next generation web services will be matched based on semantically equivalent meaning, even when they are described differently [Broens, 04] and will include support for partial matching in the absence of an exact match.

The current focus for semantic reasoning is to rely on a high end server. This reliance on a high-end, centralised node for performing for performing semantically driven pervasive service discovery can clearly be attributed to the fact that semantic reasoners used by these architectures (including Prolog, Lisp and Jess, as well as more newly available OWL reasoners such as FaCT++ [Fact, 07], RacerPro [RaceProc, 07]



and KAON2 [Kaon, 07]) are all resource intensive. As such, they are unsuitable for deployment on small resource constrained devices, such as PDAs and mobile phones. These small devices which are typical in the context of mobile service discovery are quickly overwhelmed when the search space in terms of the size of ontologies and reasoning complexity increases. KRHyper [Kleeman, 06] implements a First Order Logic (FOL) reasoner using Tableaux without the expected performance degradation that one would expect compared to DL (a decidable subset of FOL) reasoning. KRHyper performs better than Racer, however it is still quickly overwhelmed (as ontologies/complexity grows), out of memory exceptions occur and no response is provided. Clearly, this shows that such reasoners cannot be directly ported to a mobile device in their current form.

The reality of mobile environments is a world characterised by ad-hoc an intermittent connectivity where such reliance on remote/centralised processing (and continuous interaction) may not always be possible or desirable given the need for rapid processing and dynamically changing context (e.g. a driver has gone past a parking area). Pervasive service discovery has to necessarily be under-pinned by the current context to meet the all-important criteria of relevance in constantly changing situations. The communication overhead (not to mention the infeasibility/impracticability) of constantly relaying contextual and situational changes of the user/device to a central server will lead to inevitable delays. Thus there is a clear imperative that for semantically driven pervasive service discovery to meet the very real response-time challenges of a mobile environment, the capacity to perform matching and reasoning must occur on the resource limited device itself. Therefore, there is a need for developing a pervasive services discovery architecture, which more flexibly manages the trade-off between computation time and precision of results, depending on the available resources on the device.

In this paper we define our mTableaux algorithm, which incorporates a weighted approach to reasoning and implements strategies to optimise DL reasoning tasks so that relatively large reasoning tasks of several hundred individuals and classes may function on small devices. The remainder of the paper is structured as follows. In section 2 we outline our approach to reasoning and provide several optimisations and ranking strategies. In section 3 we provide an implementation and performance evaluation and in section 4 we conclude the paper.

## 2 mTableaux for Mobile Reasoning

In this section we discuss current Tableaux semantic reasoners and present mTableaux, our algorithm for enabling Tableaux reasoning on mobile devices.

### 2.1 Semantic Reasoners

The effective employment of semantic languages requires the use of semantic reasoners such as Pellet [Pellet, 03], FaCT++ [Fact, 07], RacerPro [RacerPro,07] and KAON2 [Kaon, 07]. Most of these reasoners utilise the widely used Tableaux [Horrocks, 05] algorithm. These reasoners are shown in Figure 1, which illustrates the component parts required for OWL reasoning. Reasoners can be deployed on servers and interacted with via DL Implementation Group (DIG) interface specification

which uses XML over HTTP. Alternatively, interaction may be facilitated directly using native APIs. These require which requires RDF/XML parsing functionality to load OWL files into the resaoner. Pellet utilises either Jena or OWL-API for interaction and RDF parsing.

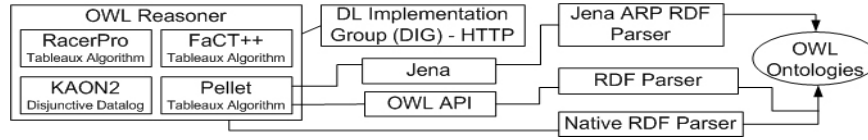


Figure 1: Semantic reasoner components.

DL Tableaux reasoners, such as Pellet, reduce all reasoning tasks to a consistency check. Tableaux is a branching algorithm, in which disjunctions form combinations of branches in the tree. If all branches contain a clash, in which a fact and its negation are both asserted, then a clash exists for all models of the knowledge base. Generally, ontologies are checked for consistency, before inference reasoning occurs. In this paper, we assume that all ontologies are consistent and do not perform an ontology-wide consistency check, when resources are low. We suggest that only those individuals representing services be checked for potential membership to only the concept representing the user request. A service individual  $I$ , matches a user request  $RQ$ , if  $I \in RQ$ .

Consider a scenario in which Bob wishes to discover a printer which matches the attributes of black and white and support for wireless connectivity. In the following, we provide a Tableaux consistency check excerpt to check the truth of  $LaserPrinter1 \in Request$ . See section 4.1 for full case study, we include the first two attributes below only, for brevity.

```
Add: ¬Request to Individual: LaserPrinter1
      ¬Request ≡ ¬PhModem ∪ ∇ hasColour.¬{Black}.
Apply Branch Element: ¬PhModem ≡ ∇ hasComm.¬(Modem ∩ ≥1phNumber).
  Add: ¬Modem ¬(≥ 1 phNumber) to Fax, Modem1, BT.
  Apply Branch Element: ¬Modem to Modem1, CLASH.
  Apply Branch Element: ¬(≥1phNumber) to
    Modem1, CLASH.
Apply Branch Element: ∇ hasColour.¬{Black}
  Add: ¬{Black} to Nominal Black, CLASH.
```

All elements of the negated Request generate a clash, so  $LaserPrinter1 \in Request$  is proven to be true.

## 2.2 mTableaux Reasoning Strategies

The work in this paper concentrates on optimisations for the Tableaux algorithm (see figure 1). We observed that DL Tableaux reasoners leave scope for optimisation to enable reasoning on small/resource constrained devices with a significant improvement to response time and avoiding situations such as “Out of Memory” errors encountered in [Kleeman, 06]. Our mTableaux algorithm involves a range of optimisation strategies such as: 1 selective application of consistency rules, 2.

skipping disjunctions, 3. establishing pathways of individuals and disjunctions which if applied would lead to potential clashes and associating weights values to these elements such that the most likely disjunctions are applied first, by 3a. ranking individuals and 3b. ranking disjunctions.

Application of consistency rules to a subset of individuals only, reduces the reasoning task. This subset can be established using the universal quantifier construct of the form  $\forall R.C = \{ \forall b.(a, b) \in R \rightarrow b \in C \}$  [Baader, 03], where R denotes a relation and C denotes a class concept. The quantifier implies that all object fillers of relation R, are of type C. Application of this rule adds role filler type C to all objects for the given role R, which can give rise to an inconsistency. Therefore, we define the subset as being limited to the original individual being checked for membership to a class, and all those individuals which branch from this individual as objects of roles specified in universal quantifiers.

Disjunctions can be skipped (not applied), according to whether they relate to the request type. A disjunction may be applied when one of its elements contains a type found in a set populated by adding the request type and all its unfolded types, where elements of conjunctions and disjunctions or role fillers of universal quantifiers are also added and unfolded. Weighted individuals and disjunctions can be established using a weighted queue. We have employed a weighted approach, so that we can leverage this in future work, to avoid “Out Of Memory” errors by providing a result to the user with a level of uncertainty, when resources become low. Disjunctions in a weighted queue, can be ranked by recursively checking each element in the disjunction for a potential clash. If a pathway to a clash is found, the weighted value of all individuals and disjunctions involved in this path are increased. Individuals can be ranked according to whether they contain disjunctions which are likely to give rise to a clash. This occurs by taking the last applied element concept C in a disjunction from individual I, which did not give rise to a clash, and attempting to find a pathway to a future clash, in the same way as for ranking disjunctions. Formal descriptions of these optimisation and ranking strategies are given in [Steller, 08].

### 3 Implementation and Performance Evaluation

In this section we provide two case studies in order to evaluate our mTableaux algorithm and implementation details.

#### 3.1 Case Study 1 – Searching for a Printer

Bob wishes to send a fax from his PDA and issues a service request for a listing of black and white, laser printers which support a wireless network protocol such as Bluetooth, WiFi or IrDA, a fax protocol and which have a dialup modem with a phone number. Equations 1-4 show Bob’s request in Description Logic (DL) [Baader, 03] form, while equation 5 presents a possible printer.

$$\text{Request} \equiv \text{PhModem} \cap \exists \text{hasColour.}\{\text{Black}\} \cap \text{hasComm.}\{\text{Fax}\} \cap \text{LaserPrinterOperational} \cap \text{WNet} \quad (1)$$

$$\text{PhModem} \equiv \exists \text{hasComm.}(\text{Modem} \cap \geq 1 \text{ phNumber}) \quad (2)$$

$$\text{LaserPrinterOperational} \equiv \text{Printer} \cap \exists \text{hasCartridge.}\{\text{Toner}\} \cap \geq 1 \text{ hasOperationalContext} \quad (3)$$

$$\text{WNet} \equiv \exists \text{hasComm.}\{\text{BT}\} \cup \exists \text{hasComm.}\{\text{WiFi}\} \cup \exists \text{hasComm.}\{\text{IrDA}\} \quad (4)$$

$$\begin{aligned} & \text{Printer}(\text{LaserPrinter1}), \text{hasColour}(\text{LaserPrinter1}, \text{Black}), \\ & \text{hasCartridge}(\text{LaserPrinter1}, \text{Toner}), \text{hasComm}(\text{LaserPrinter1}, \text{BT}), \\ & \text{hasComm}(\text{LaserPrinter1}, \text{Fax}), \text{hasOperationalContext}(\text{LaserPrinter1}, \\ & \text{Ready}), \text{Modem}(\text{Modem1}), \text{hasComm}(\text{LaserPrinter1}, \text{Modem1}), \\ & \text{phNumber}(\text{Modem1}, \text{"9903 9999"}) \end{aligned} \quad (5)$$

Equation 1 defines five attributes in the request, the first is unfolded into equation 2, specifying the printer must have a modem which has a phone number. The second attribute specifies a black and white requirement. The third attribute requires support for the fax protocol, and the fourth unfolds into equation 3, specifying a printer which has a toner cartridge and at least one operational context. The fifth unfolds into equation 5, which specified that one of the wireless protocols (Bluetooth, WiFi or IrDA) are supported. Equation 5 shows a DL fragment defining the LaserPrinter1 individual as meeting the service request. We also defined faxmachine5 and printer12 individuals which match the request and an additional 17 individuals which did not.

### 3.2 Case Study 2 – Searching for a Cinema

Bob wishes to find a movie cinema with a café attached which has a public phone and WiFi public Internet. He issues a request for a retail outlet which has at least 5 cinemas attached that each screen movies, has a section which sells coffee and tea, sells an Internet service which supports access using the WiFi protocol and sells a fixed phone service. We specified 20 individuals, in which VillageCinemas, HoysCinemas and AcmeCinemas match the request, while the remaining 17 do not.

### 3.3 Implementation

We implemented the selective consistency, rank individuals, rank disjunctions and skip disjunctions strategies defined in section 3, in the Pellet v1.5 reasoner. We selected Pellet because it is open source while the other reasoners were not, allowing us to provide a proof of concept and compare performance with and without the strategies enabled.

We implemented the two scenarios outlined in section 4.1 and 4.2. Case study 1 comprises 141 classes, 337 individuals and 126 roles. Case study 2 defines 204 classes, 241 individuals and 93 roles. Due to the resource intensive nature of XML parsing, we pre-parsed OWL XML files into text files of triples and leave optimised XML parsing for future work.

We performed an evaluation on a HP iPAQ hx2700 PDA, with Intel PXA270 624Mhz processor, 64MB RAM, running Windows Mobile 5.0 with Maysaifu Java J2SE Virtual Machine (JVM) [MySaify, 07], allocated 15MB of memory.

A type check was undertaken, comparing a positive (matching) and negative (non-matching) individual against the service request for each case study. Executing the case studies on Pellet with no optimisations resulted in the “Out Of Memory“ exception in figure 2(a), while executing the case studies with the mTableaux strategies enabled resulted in successful completion of the type check. A detailed performance analysis comparing each individual and optimisation strategy is given in [Steller, 08].

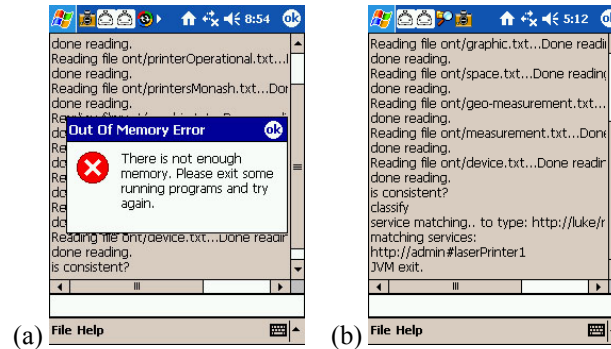


Figure 2: (a) Out of memory error (b) Correctly matched service URI returned.

### 3.4 Performance Comparison

In order to show how mTableaux compares to other commercial OWL semantic reasoners, we provide a performance comparison with RacerPro. Since RacerPro is a desktop reasoner, which cannot be deployed to mobile devices, we have undertaken a performance evaluation on a Pentium Centrino 1.82GHz computer with 2GB memory with Java 1.5 (J2SE) allocated maximum of 500MB for each experiment. All timings presented are computed as the average of 10 independent runs. Reasoners used are RacerPro 1.9.2 beta, Pellet 1.5 and Pellet 1.5 with mTableaux optimisation strategies implemented and enabled. Racer automatically performed classification of a class type hierarchy, while Pellet did not. Therefore, we provide results for Pellet with and without classification (which involved type checks only). Each of the 20 services in each case study was checked to see if it matched the request, to provide a set of matching services.

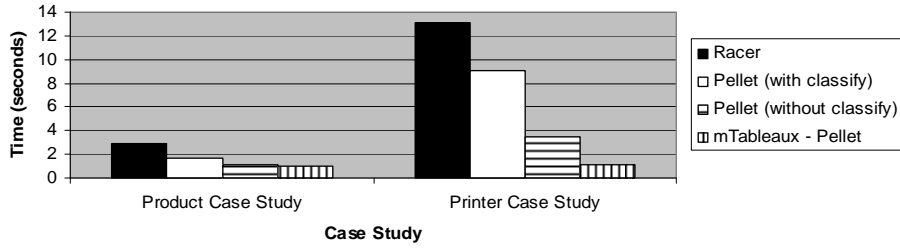


Figure 3: Processing time required to perform discovery with a service set of 20.

Figure 3 shows that Pellet outperformed RacerPro. We found that although RacerPro outperformed Pellet when only one matching individual was present in the ontology, as we increased the number to two or three, Pellet began to outperform RacerPro. The results show that mTableaux performed in less than half the time of Pellet without classify for the Printer case study, with less substantial improvements for the Product ontology. However, we observed that the number of branches applied when executing the Product case study using mTableaux was less than half that of Pellet. We conclude that when the amount of available memory available is constrained (eg 15MB) as on a small device, the difference in performance between Pellet and mTableaux is significantly amplified.

Bernstein and Klein [Bernstein, 02] suggest the metrics of recall and precision (see equation 6 and 7) to assess the quality of match with the user request, where recall measures how effectively the relevant services are discovered and precision measures how effectively irrelevant services are not discovered. Let  $x$  denote the number of relevant services returned to the user, let  $n$  denote the total number of relevant services available and  $N$  the total number of services returned.

$$\text{Recall} = x / n \quad (6)$$

$$\text{Precision} = x / N \quad (7)$$

In our tests all reasoners (Racer, Pellet and mTableaux) returned perfect recall and precision of 1, effectively discovering only the three relevant printer and product services in the printer and product case studies, respectively. This result is illustrated in figure 4 (printer and product ontologies are shown on the left and right of the figure, respectively).

```

Output - ProductScenarioMain-32MEReady (run)
Checking.. http://luke/printers#Monash#Printer14
Checking.. http://luke/printers#Monash#SinkPrinter7
Checking.. http://luke/printers#Monash#FaxMachine5
Checking.. http://luke/printers#Monash#Printer18
Checking.. http://luke/printers#Monash#Printer12
*****
Services which matched: #Request
Match: http://luke/printers#Monash#LaserPrinter1
Match: http://luke/printers#Monash#FaxMachine5
Match: http://luke/printers#Monash#Printer12
done
BUILD SUCCESSFUL
Finished building ScenarioMain-32MEReady (run).

Output - ProductScenarioMain-32MEReady (run)
Checking.. http://luke/starbucksCoffee#Request#StarBucksCoffee
Checking.. http://luke/coffeebeans#CoffeeBeansAndTeaLeaf
Checking.. http://luke/waffinbreak#WaffinBreak
Checking.. http://luke/harveynorman#HarveyNorman
Checking.. http://luke/acmecinemas#AcmeCinemas
*****
Services which matched: #Request
Match: http://luke/acmecinemas#AcmeCinemas
Match: http://luke/hoytcinemas#HoytCinemas
Match: http://luke/villagecinemas#VillageCinemas
done
BUILD SUCCESSFUL
Finished building ScenarioMain-32MEReady (run).

```

Figure 4: Output for the successful matching of three services out of 20.

## 4 Conclusion and Future Work

mTableaux was shown to improve the performance of pervasive discovery reasoning tasks in two case studies, so that they can be completed on small resource constrained devices. It was shown to out perform RacerPro without reducing the quality of results returned. The mTableaux strategies achieve this by limiting the number of consistency rules applied and by applying the most important branches first to avoid the need for full branch saturation.

We are developing a resource-aware reasoning strategy to better manage the trade-off between result correctness and resource availability. This strategy will provide a result with a level of uncertainty, rather than an “Out Of Memory” error, when resources become low, by leveraging our weighted approach. We will utilise our weighted queue to associate dynamically changing weights to all potentially matching services and to each request attribute. As such the most likely services will be checked for the most important attributes, first.

In addition we researching ways to pre-emptively weight the potential service set according to their match to contextual requirements specified in user profiles, before a request takes place (eg check only services in my current building). Context-attributes and the weighted importance of these, will be determined by explicit user preferences and implicit profiling of previous user activity.

## References

- [Baader, 03] Baader F, Calvanese D, McGuinness DL, Nardi D, Patel-Schneider PF. The Description Logic Handbook: Theory, Implementation, and Applications: Cambridge University Press 2003.
- [Bernstein, 02] Bernstein A, Klein M. Towards High-Precision Service Retrieval. International Semantic Web Conference (ISWC 2002); 2002 9 - 12 June; 2002. p. 84 – 101.
- [Broens, 04] Broens T. Context-aware, Ontology based, Semantic Service Discovery [Master]. Enschede, The Netherlands: University of Twente; 2004.
- [FaCT, 07] FaCT++. 2007 [cited 2007 May 1]; Available from: <http://owl.man.ac.uk/factplusplus/>
- [Horrocks 05] Horrocks I, Sattler U. A Tableaux Decision Procedure for SHOIQ. 19th Int Joint Conf on Artificial Intelligence (IJCAI 2005); 2005; Morgan Kaufman; 2005.

[Kaon, 07] KAON2. 2007 [cited 2007 June 21]; Available from: <http://kaon2.semanticweb.org>

[Kleemann, 06] Kleemann T. Towards Mobile Reasoning. International Workshop on Description Logics (DL2006); 2006 May 30 - June 1; Windermere, Lake District, UK; 2006.

[Mysaifu, 07] Mysaifu JVM. [cited; Available from: [http://www2s.biglobe.ne.jp/~dat/java/project/jvm/index\\_en.html](http://www2s.biglobe.ne.jp/~dat/java/project/jvm/index_en.html)

[Pellet, 03] Pellet. 2003 [cited; Available from: <http://www.mindswap.org/2003/pellet/>

[RacerPro, 07] RacerPro. 2007 [cited 2007 May 23]; Available from: <http://www.racer-systems.com>

[Roto, 05] Roto V, Oulasvirta A. Need for Non-Visual Feedback with Long Response Times in Mobile HCI. International World Wide Web Conference Committee (IW3C2); 2005 May 10 - 14; Chiba, Japan; 2005.

[Steller, 08] Steller L, Krishnaswamy S, Cuce S, Newmarch J, Loke S. A Weighted Approach for Optimised Reasoning for Pervasive Service Discovery Using Semantics and Context. 10th International Conference on Enterprise Information Systems (ICEIS); 2008 12 - 16 June; Barcelona, Spain; 2008.



## Community Rating Service and User Buddy Supporting Advices in Community Portals

Martin Vasko<sup>1</sup>, Uwe Zdun<sup>1</sup>, Schahram Dustdar<sup>1</sup>, Andreas  
Blumauer<sup>2</sup>, Andreas Koller<sup>2</sup> and Walter Praszl<sup>3</sup>

<sup>1</sup>Distributed Systems Group, Vienna University of Technology, Austria  
{m.vasko,zdun,dustdar}@infosys.tuwien.ac.at

<sup>2</sup>punkt.netServices, Vienna, Austria  
{koller,blumauer}@punkt.at

<sup>3</sup>Special Interest Magazines, Vienna, Austria  
w.praszl@simskultur.net

**Abstract:** Many community portals allow users to search for events, such as concerts, festivals or other things of interest and to rate them. Especially in the culture domain the users' impressions of events is based on many factors, such as quality, personal interests, etc. Such factors can be represented using an ontology. The ratings provided by the users of community portals are often highly biased by personal opinions, and hence not all information provided by users is useful for all other users. But it can be observed that users with similar interests provide similar opinions. This paper introduces a community rating approach based on this observation. Our concept introduces for each user a user buddy, representing the part of the community with similar opinions as those of the user. The buddy uses a community rating service as a basis to give advices to users, such as recommendations for events or help in searching the portal. Our approach gathers opinions using a domain ontology, but it is not dependent on a specific ontology.

**Key Words:** Community Rating, Ontologies, Web Services, Community Portals

**Category:** H.3, H.3.1, H.3.5

### 1 Introduction

Many community portals [Schuemmer and Lukosch 2007] for diverse domains are existing on the Web. The users of the community portals usually provide information about events or things of interest to other users in the community. In many cases the relevant information is hard to find. A simple reason for this is the mass of information provided in community portals: it is hard for the user to filter out the useful information. Unfortunately, automatically filtering the information is difficult, too, because of the diversity of opinions in online user communities.

Consider the culture domain as a typical example. In this domain, cultural events, such as concerts or festivals, are advertised and rated on community portals. Large community portals in this domain usually have many users with diverse interests. Even people going to the same kind of event often have different

preferences. Consider the example of a festival: For some users only the quality of the music counts, for others additional attractions or the quality of the camping site are as important. This paper deals with the question: If thousands of opinions for such events are provided in a community portal, how can a user retrieve the useful information, given the user's personal preferences, and how can the community portal help the user to retrieve the information?

Structuring the various factors in a user's opinion is – in the context of the Semantic Web [Berners-Lee 1999] – done using ontologies. Ontologies unify diverse understandings by introducing a central perception hierarchy for different knowledge domains. However, there is the problem that in many domains which use community portals, such as the culture domain, no well accepted standard ontology exists for the whole domain, but only partial ontologies. In addition, a generic approach should not be dependent on one domain ontology, but be open for any domain. Also, over time, the domain ontology must be adapted.

In our approach users can provide opinions about events or things of interest. Other users can provide a rating about a given opinion. We assume that users with similar interests and background provide the most useful information for a user. Hence by collecting and evaluating the user's ratings about other users' opinions, we can provide a *community rating* that shows the relevance of a user's opinion for another user. The community rating is transitive in the sense that it not only considers the ratings of a user about other users' opinions, but also the ratings of the other users about yet other users' opinions, and so on.

A central *community rating service* calculates the community ratings between the user and all other users. This way, a *user buddy* is dynamically constructed, which is an abstraction representing the user's specific view on the community. The user buddy is then used to calculate advices for the users of the community portal. Note that our approach uses a custom ontology as the basis to describe all the factors relevant for user ratings. The general approach hence is open to be used with any ontology. We use a service-oriented architecture for the community service in order to deal with heterogeneous platforms and technologies that are usually used for community portals and semantic web ontologies.

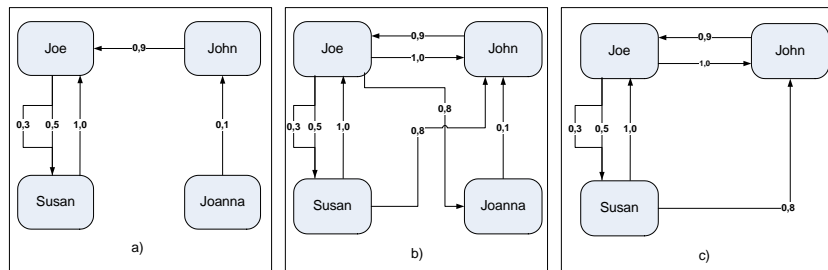
Our approach is exemplified for a community portal for cultural events [SCG 2008]. Our community portal offers an *event buddy* using the community rating service to give advice for future events. The community ratings are calculated based on ratings given for user opinions on (mostly past) events. The culture portal uses a culture ontology as a basis for all user opinions, which is maintained by the culture experts who run the portal.

This paper is organized as follows: An overview and a motivating example are provided in Section 2. The community rating service, the user buddy, and the system architecture are described in Section 3. Our approach is compared to existing approaches in Section 4. Finally Section 5 concludes the paper.

## 2 Motivating Example

In this section we give an overview of our approach from the user's perspective using a motivating example for a situation in which a community rating can be applied. This example is resolved later in this paper. Consider users provide opinions for events, attractions, or things of interest via the community portal. A user opinion consists in our approach of a preselected set of elements from an ontology. The ontology is maintained by experts from the domain, and the experts also select the elements from the ontology which can be rated by users. For instance, in our culture portal 4-8 ontology elements are selected to be rated per culture event, and in addition the user can provide free text. Users are automatically asked to provide their opinion after they visited events (i.e., if they bought a ticket via the portal), as well as future events in which they are interested. They are motivated to participate using lotteries for free tickets.

When a user views another user's (anonymous) opinion, the user is asked to give a rating on that opinion. This rating should be provided in a simple and easy-to-use fashion. The user is simply asked: "How do you rate this user opinion?" The user can answer on a scale ranging from "very helpful" to "not helpful at all", and as a result values ranging from 1.0 to 0.0 are produced. This way, incrementally a graph of ratings about other users' opinions is built up. This graph is the basis for calculating the user buddy. New users must first train their buddy – to get connected to the community graph. A number of random user opinions are shown to the user, and the user must give a rating for them.



**Figure 1:** Sample User Ratings

Three rating scenarios are illustrated in Figure 1. In first scenario (a) four users have provided ratings. For instance, user *Susan* has provided one rating on an opinion by *Joe*, and the opinion was very helpful to her (1.0). *Joe*, in turn, has provided two ratings about opinions by *Susan*, with the values 0.3 and 0.5. The complete ratings derived from Scenario (a) are:  $\langle John \xrightarrow{0.9} Joe \rangle, \langle$

$Susan \xrightarrow{1,0} Joe >, < Joe \xrightarrow{0,3} Susan >, < Joe \xrightarrow{0,5} Susan >, < Joanna \xrightarrow{0,1} John >$ . Scenario (b) shows the addition of more ratings over time. Finally, Scenario (c) illustrates the removal of a user and the consequences for the ratings. Based on such rating graphs we can derive the *community rating* to produce the user buddy that represents the user's view on the community.

### 3 Community Rating Service

In this section we present the details of the community rating service and introduce the prototype implementation details.

#### 3.1 Application Logic of the Community Rating Service

Algorithm 1 illustrates the main logic of the community rating service. The algorithm recursively calculates the community rating between two users *from* and *to* for a depth of *levels* through the graph. All users for which the user *from* has given one or more ratings are considered. If there is a direct rating between the user and *to*, the direct rating is considered with a factor *DirectRatingFactor*. Otherwise the community rating is calculated recursively, and added with a factor of 1. If no rating has been added, or if the *levels* are exceeded,  $-1$  is returned to indicate the stop of the recursion. This way all direct and transitive ratings from user *from* to *to* up to the depth *levels* are added. Finally they are weighted by the *number* of ratings that have been added. We use the function *getAverageRating()* to obtain a rating between two users, because each user might have *n* ratings for another user. Please note there are two ways to fine-tune this algorithm:

- *levels* determines the depth of the search for ratings through the graph. If the graph is only loosely populated, the number of levels can be increased to obtain better results. If the performance decreases because of the size of the graph, the number of levels can be decreased.
- *directRatingFactor* determines the importance of a user's own judgment compared to ratings made by others.

#### 3.2 Example Resolved

Figure 2 illustrates the community ratings calculated for the examples from Figure 1. The user ratings from Scenario (a) in Figure 1 result in the community ratings depicted in Scenario (a) in Figure 2. Scenarios (b) and (c) in Figure 1 illustrate the adding of ratings and the removing of users. Scenario (b) in Figure 2 illustrates the resulting community ratings.

**Algorithm 1** CommunityRating

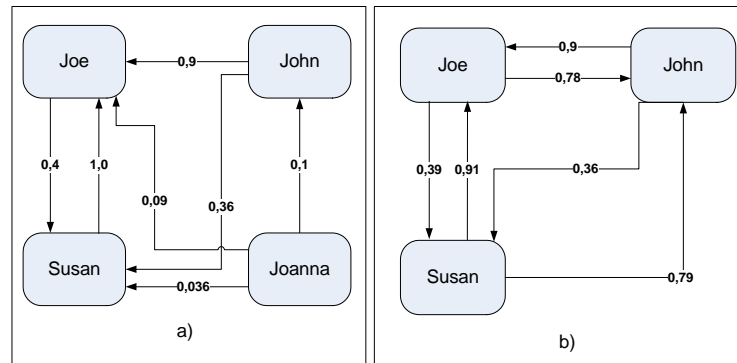
---

```

Require: from : User, to : User, levels : Integer
Ensure: directRatingFactor
number := 0
ratingSum := 0
addedRating := false
if levels == 0 then
  return -1
end if
for all u : User in getUserRatings(from) do
  if u == to then
    addedRating := true
    number += directRatingFactor
    ratingSum += directRatingFactor * getAverageRating(from, u)
  else
    communityRating := CommunityRating(u, to, levels - 1)
    if communityRating != -1 then
      addedRating := true
      number += 1
      ratingSum += 1 * getAverageRating(from, u) * communityRating
    end if
  end if
end for
if addedRating == false then
  return -1
end if
return ratingSum / number

```

---

**Figure 2:** Sample Community Ratings**3.3 User Buddy**

To use the community ratings in order to give advices to users, we developed a virtual *User Buddy* concept. The aim of this abstraction is to track individual user preferences and to aggregate interesting events, attractions, or things of interest according to the community rating. The buddy uses the community rating service to give advice to users, such as recommendations or helping users to search the portal. By actively providing user ratings about other user's opinions, users teach the buddy their personal preferences.

From a user perspective, trustworthy rating mechanisms will only be accepted

if they (1) help to improve ranking and filtering of information and (2) if they do not rely on methods which need intensive training efforts accomplished by the user. In our approach the buddy learns quickly, even when single users do not put much efforts on the buddy training since our approach relies on the overall community behavior. That is, the user can benefit from the training of the buddy that other users have performed.

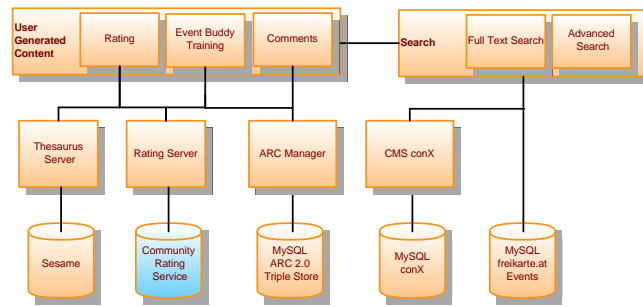
In the culture portal case study, we provide an *event buddy* to give advices for cultural events. The event buddy aggregates the community ratings for each user. When the user searches for a cultural event, hence the opinions of those users that the user has directly or indirectly given a high community rating, are considered. Three exemplary usage scenarios are:

- The event buddy can be asked to provide a list of recommended events in a time frame. This is done by calculating a list of users (number can be configured) with the highest community ratings. Then the events with a positive opinion (i.e. over a certain threshold) that take place in the specified time frame are recommended.
- The event buddy can provide opinions on a specific event sorted by their relevance for the user, based on the community ratings.
- The event buddy can tell the user the relevance of an opinion for him. Consider user *John* in Scenario (c) of Figure 1 would like to see a theater play, and the user *Susan* has given a very positive opinion on the play. Even though *John* has never rated *Susan* himself, the event buddy can give the advice that this positive opinion has to be considered with care, because *Susan* has only a low community rating, meaning that users (in this case *Joe*) that have similar opinions like *John* have given low ratings for *Susan*.

### 3.4 Service-Based Integration

Figure 3 illustrates an abstract overview of the service environment of the culture portal. The heterogeneous environment in the project motivated us to implement the algorithm as a Web Service based on Apache Axis [Axis 2008]. Additional components (Thesaurus Server, Triple Store, CMS functionality etc.) are integrated by the use of Web Services as well.

As can be seen in the previous section, the community rating service has no direct dependencies to information of the Web Portal, such as the ontologies used for rating the user opinions. The only input dependencies are users (addition, removal) and user ratings (addition, removal), and the only output dependencies are the community ratings for the user buddy. Hence, it makes sense to enable the integration of the community rating service into different system architectures which is achieved using Web Services.



**Figure 3:** System Architecture

## 4 Related Work

[Staab et al. 2000] introduce a community Web Portal using semantic approaches to structure and unify diverse information. Their motivation to use ontologies for capturing knowledge in a generic way and to explicitly specify shared concepts corresponds to the motivation for this work.

[Galizia et al. 2007] introduce a trust based methodology for Web Service selection. Their work introduces a Web Service Trust Ontology (WSTO) based on Web Service Modeling Ontology (WSMO [Fensel et al. 2006]). Their approach matches classes of Web Services with participant trust profiles.

Ranking user content is quite popular in the field of semantic computing.

An approach to rank ontologies is presented by [Tartir and Arpinar 2007]. They introduce OntoQA, a tool to evaluate and rank ontologies based on a catalog of metrics. OntoQA provides a tuneable ranking approach by allowing the user to bias the preference for certain ontologies.

[Massa and Bhattacharjee 2004] provide an experimental analysis of a community Web Portal based on a recommender system incorporating trust. The authors argue, that classical collaborative filtering approaches consider only a small portion of the user base whereas trust-aware mechanisms build on a high rate of the whole user community. [Zhdanova and Fensel 2005] identify the creation of semantic web content and a community-driven ontology management as approaches to overcome the limitations of current community Web Portals.

[Schuemmer and Lukosch 2007] describe software design patterns in groupware systems. The *Buddy List* pattern describes personalized user lists. This description matches to the group of users identified by the *community rating* algorithm. The *Expert Finder* pattern describes the problem to identify a user having expertise on a special artifact in the platform. This pattern can be mapped to the problem of identifying users in the community sharing the same preferences.

## 5 Conclusion and Future Work

The introduced *community rating service* provides an approach to relate and weigh diverse opinions of community portal users. The approach can work with arbitrary ontologies for defining the rating of opinions on events, attractions, and other things of interest, but it is not dependent on the ontology used. Our approach provides individual users with an individualized view onto the communities' opinions. As part of the Web platform a *user buddy* is introduced, which uses the community rating service to provide advices, such as recommendations or help in searching the portal, to the users. By "teaching" individual event preferences, the user is able to sharpen the accuracy of recommendations. The combination of the community rating service and the user buddy significantly improves the recommendation functionality in community portals. Future work covers the application of the algorithm on bigger communities by the integration of further cultural event platforms in the SCG prototype [SCG 2008].

### Acknowledgment

This work is supported by a BMVIT/FFG grant for the FIT-IT project "Semantic Culture Guide" [SCG 2008].

### References

- [Berners-Lee 1999] Berners-Lee, Tim. Weaving the Web, Orion Business Books, London, 1999
- [Fensel et al. 2006] Fensel, Dieter; Lausen, Holger; Polleres, Axel; Brujin, Jos de. Enabling Semantic Web Services: Web Service Modeling Ontology. Springer, 2006
- [Galizia et al. 2007] Galizia, Stefania; Gugliotta, Alessio; Domingue, John. A Trust Based Methodology for Web Service Selection. International Conference on Semantic Computing(ICSC), pp.193-200, September 2007
- [Massa and Bhattacharjee 2004] Massa, Paolo; Bhattacharjee, Bobby. Using trust in recommender systems: an experimental analysis. iTrust, pp. 221-235, Springer, 2004
- [Schuemmer and Lukosch 2007] Schuemmer, Till; Lukosch, Stephan. Patterns for computer-mediated interaction. Wiley Series in Software Design Patterns, 2007
- [Staab et al. 2000] Staab, Stefan; et al. Semantic community Web portals. Int. Journal of Computer Networking and Telecommunication, pp. 473-491, June 2000
- [Tartir and Arpinar 2007] Tartir, Samir; Arpinar, I. Budak. Ontology Evaluation and Ranking using OntoQA. Int. Conf. on Semantic Computing(ICSC), 2007
- [Weerawarana et al. 2005] S. Weerawarana, F. Curbera, F. Leymann, T. Storey, D.F. Ferguson. Web Services Platform Architecture, Prentice Hall, 2005
- [Zhdanova and Fensel 2005] Zhdanova, Anna; Fensel, Dieter. Limitations of Community Web Portals: A Classmates Case Study. Int. Conf. on Web Intelligence, 2005
- [Axis 2008] Axis2 Version 1.3, Apache Software Foundation, <http://ws.apache.org/axis2/>, Last Access: 2008
- [SCG 2008] Web Portal Prototype, Semantic Culture Guide Project, <http://www.semantic-culture-guide.net/>, Last Access: 2008
- [WSDL 2008] Web Service Description Language, W3C, <http://www.w3.org/TR/wsdl>, Last Access: 2008



## **Seeding, Weeding, Fertilizing – Different Tag Gardening Activities for Folksonomy Maintenance and Enrichment**

**Katrin Weller**

(Heinrich-Heine-University, Düsseldorf, Germany  
weller@uni-duesseldorf.de)

**Isabella Peters**

(Heinrich-Heine-University, Düsseldorf, Germany  
isabella.peters@uni-duesseldorf.de)

**Abstract:** As social tagging applications continuously gain in popularity, it becomes more and more accepted that models and tools for (re-)organizing tags are needed. Some first approaches are already practically implemented. Recently, activities to edit and organize tags have been described as “tag gardening”. We discuss different ways to subsequently revise and reedit tags and thus introduce different “gardening activities”; among them models that allow gradually adding semantic structures to folksonomies and/or that combine them with more complex forms of knowledge organization systems.

**Keywords:** Social tagging, folksonomy, tag gardening, emergent semantics, knowledge organization system, knowledge representation

**Categories:** H.3.1, H.3.3, H.3.5, L.1.3, L.1.0

### **1 Introduction**

Social tagging functionalities are by now a common feature for most social software applications (e.g. video or photo sharing platforms, social networking and social bookmarking tools). Folksonomies are used to organize various types of resources such as scientific articles, references, bookmarks, pictures, videos, audio files, blog posts, discussions, events, places, people etc. They have been greatly accepted by (Web) users as well as by a considerably large scientific community – although several shortcomings of folksonomies have been pointed out [Peters 06], [Peters & Stock 07]. These critiques are mainly based on comparisons of folksonomies with traditional methods of knowledge organization systems (KOS, like thesauri, classification systems etc.) and professional indexing techniques. Yet, the boundaries between structured KOS and folksonomies are not at all solid but rather blurred. This means, amongst others, that folksonomies can adopt some of the principle guidelines available for traditional KOS and may gradually be enriched with some elements of vocabulary control and semantics. On the other hand, folksonomies provide a useful basis for the stepwise creation of semantically richer KOS and for the refinement of existing classifications, thesauri or ontologies [Weller 07].

One of the basic questions regarding the enhanced use of folksonomies is: how to combine the dynamics of freely chosen tags with the steadiness and complexity of controlled vocabularies? It appears that a gradual refinement of folksonomy tags and

a stepwise application of additional structure to folksonomies is a promising approach. Some platforms already provide different features to actually manipulate, revise and edit folksonomy tags. Theoretical approaches for structural enhancement of folksonomies are discussed under such diverse headlines as “emergent semantics” [Zhang et al. 06], “ontology maturing” [Braun et al. 07], “semantic upgrades” or “semantic enrichments” [Angeletou et al. 07]. Lots of research in this regard currently deals with developing different algorithms to restructure folksonomies automatically. We discuss “tag gardening” as a mainly manual activity, performed by the users to manage folksonomies and gain better retrieval results, which can be supported by certain automatic processes.

## 2 Tag Gardening - Revision and Maintenance of Folksonomies

The image of “tag gardening” has been introduced in a blog post by James Governor [Governor 06]. By now it is used to describe processes of manipulating and re-engineering folksonomy tags in order to make them more productive and effective. Along the lines of this, we now specify different “gardening activities” which are relevant for the maintenance of folksonomies and their effective usage in the course of time. These activities are to some extent based on common procedures for building classical KOS, e.g. [Aitchison et al. 04].

To discuss the different gardening activities, we first have to imagine a document-collection indexed with a folksonomy. This folksonomy now becomes our *garden*, each tag being a different *plant*. Currently, most folksonomy-gardens are rather savaged: different types of plants all grow wildly. Some receive high attention, others almost none. Some are useful, others are not. – Actually, folksonomies have been criticized for being a “mess” [Tanasescu & Streibel 07]. First approaches to make them more easily accessible and navigable are for example tag clouds, computations of related tags by co-occurrence, or tag-recommendations.

In the long run, improvement of folksonomies will be needed on different levels: (a) *Document collection vs. single document level*: should the whole collection of all tags of a folksonomy be edited in total, or does one only want to change the tags of a single document. (b) *Personal vs. collaborative level*: We may distinguish tag gardening performed individually by single users for the personal tags they use within a system (personomy level<sup>1</sup>), and situations that enable the whole user community of a certain platform collectively or collaboratively to edit and maintain all tags in use (folksonomy level). (c) *Intra- and cross-platform level*: Usually, a folksonomy is defined as the collection of tags within one platform or system. Yet, for some cases the use of consistent tags across different platforms will be useful.

### 2.1 Basic Formatting

One basic problem of folksonomies is that there is no guarantee for correct spelling or consistent formatting of tags. The very first activity in tag gardening would thus be *weeding*: Tag weeding is the process of removing “bad tags”. Elimination of spam tags should be the simplest form of tag weeding, and can probably even be performed

---

[1] A personomy is defined as the tag collection of a single user [Hotho et al. 06].

automatically (to keep the image of gardening, this automatic spam removal could be characterized as using *pesticides*). As in real gardens, the identification of *weed* is not always easy. For example, one has to consider which tags may be removed from the whole folksonomy (probably even permanently) and which should only be removed from certain documents. Furthermore, due to the nature of most folksonomy tools which do not allow adding multi word concepts as tags, we end up with inconsistent makeshifts such as “semanticweb”, “semanticWeb” or “semantic\_web”. In this case, to make the folksonomy more consistent, a decision would be needed about which forms are preferred, and which should be removed as weed (of course on the document level, these tags should not be removed completely but replaced by the preferred terms – the same also holds for typing errors). In social tagging applications it seems more feasible to provide some general formatting guidelines to the tagging community in advance, than to manually re-edit tags. Alternatively, we may treat these spelling variants as synonyms (see below). Similar problems arise with the handling of different word forms, e.g. singular and plural forms or nouns and corresponding verb forms. For example, the reduction to only singular nouns may be useful for enhancing recall e.g. in publication databases (if both singular and plural forms are allowed one would miss documents tagged with “thesauri” if searching for “thesaurus”), while it would bring about loss of information in other cases, e.g. for photo databases (where one may for example explicitly want to look for a photo showing more than one cow and would therefore need the plural “cows”).

Such formatting problems can be addressed automatically with methods of Natural Language Processing (NLP) [Peters & Stock 07]. Additionally, a folksonomy based system would profit from editing functionalities which allow users to delete or edit the tags assigned to single documents and (carefully) remove certain tags from the whole system. For this purpose, some formatting guidelines may be provided to or discussed by the users.

## 2.2 Tag Popularity

Common entry points to folksonomies are tag clouds. They display most popular tags in different font sizes according to the degree of popularity. In some cases, these highly popular tags are added to too many resources to render precise and useful retrieval results. In this case, it might be necessary to explicitly *seed* new, more specific tags into the tag garden which can help to narrow down the search results. These little *seedlings* will sometimes require specific attention and care, so that they do not get lost among the bigger plants. An ‘inverse tag cloud’ could be used to highlight the very rarely used tags and provide an additional access point to the document collection.<sup>2</sup>

## 2.3 Vocabulary Control, Tag Clustering and Hierarchical Structures

After the formatting problems have been solved, the actual requests of the “vocabulary problem” [Furnas et al. 87] begin: In folksonomies (a) synonyms are not

---

[2] These aspects have also been discussed in the Workshop „Good Tags – Bad Tags. Social Tagging in der Wissensorganisation“, 21.-22. February 2008, Institut für Wissensmedien, Tübingen, Germany. A similar concept to “seedlings” was introduced as “baby tags”.

bound together (thus, someone searching a photo-portal for pictures of “bicycles” would also have to use the tags “bike” and probably even translations to other languages for comprehensive searching and higher recall); (b) homonyms are not distinguished (searching for “jaguar” will retrieve pictures of the animal as well as the car); and (c) there are no explicit relations to enable semantic navigation between search- or index-terms (e.g. a search for photos of “cats” cannot automatically be broadened to include “siamese”, “european shorthair”, “birman” etc.). This lack of vocabulary control is the price for facile usability, flexibility and representation of active and dynamic language. Yet, the additional and subsequent editing of folksonomies may be the key for allowing free tagging as well as basic control functionalities over the vocabulary in use. Folksonomy users become more and more aware of these effects – which is the basis for introducing gardening techniques to enable the user to improve their tags.

For our garden this means, that we have some plants that look alike, but are not the same (homonyms), some plants which can be found in different variations and are sometimes difficult to recognize as one *species* (synonyms) and others which are somehow related or should be combined. Thus, we have to apply some *garden design* or *landscape architecture* to turn our savage garden. We may use *labels* for the homonyms, and establish *flower beds* as well as *paths* between them and *pointers* or *sign posts* to show us the way along the synonyms, hierarchies and other semantic interrelations. We need some additional structure and direct accessibility to provide additional forms of (semantic) navigation (besides tag clouds, most popular tags and combinations of tags-user-document co-occurrences).

Within classical KOS, homonyms are often distinguished by additional specifications (e.g. “bank (finance)” vs. “bank (river)”) or unique identifiers (e.g. notations in classification systems). Synonyms can be interlinked to form a set of synonyms, sometimes “preferred terms” are chosen which have to be used exclusively to represent the whole set. Some folksonomy systems already provide functionalities to derive “clusters” and “related tags”, which mainly rely on information about co-occurrence and term frequencies. For example, the photo-sharing community Flickr provides a clustering function to distinguish homonyms<sup>3</sup>. Lots of research concentrates on automatic clustering and different clustering algorithms [Begelman et al. 06], [Schmitz 06], [Grahl et al. 07]. Methods for automatically distinguishing homonyms<sup>4</sup> in folksonomies by context information (users, documents, tags) are also being developed, [Au Yeung et al. 07]. Even automatic approaches for “converting a tag corpus into a navigable hierarchical taxonomy” [Heymann & Garcia-Molina 06] can be found.

Besides these automatic approaches, options for individual manual manipulation of tags are needed. This is particularly useful for personal tag management, where categories, taxonomies and cross-references of tags can be built and maintained for individually customized information management. Del.icio.us already offers a simple model for grouping different tags manually under different headlines.

Folksonomies typically include implicit relations between tags [Peters & Weller 08] which should be made explicit in order to obtain gradually enriched semantics. A

---

[3] One example for the term „jaguar” can be found at <http://www.flickr.com/tags/jaguar/clusters>.

[4] Sometimes also referred to as „tag ambiguity“.

first approach could be the “tagging of tags” and their interrelations as discussed by [Tanasescu & Streibel 07].

## 2.4 Interactions with other Knowledge Organization Systems

Some of the problems discussed above can also be approached by combining Folksonomies with other, more complex Knowledge Organization Systems which then act as *fertilizers*.

Behind the scenes of a folksonomy system, thesauri or ontologies may be used for query expansion and query disambiguation [Au Yeung et al. 07]. Search queries over folksonomy tags may be (automatically) enhanced with semantically related terms, derived e.g. from an ontology. For example, WordFlickr expands query terms with the help of relational structures in the WordNet Thesaurus to perform enhanced queries in the Flickr database [Kolbitsch 07]. Users submitting a query to WordFlickr may choose which types of relations (e.g. synonyms, hypernyms, hyponyms, holonyms or meronyms) should be used for expanding the query. Thus, if a user searches for “shoes” the query may be expanded with the hyponyms “slippers” and “trainers” to retrieve pictures tagged with these subtypes of shoes from the Flickr collection.

Furthermore, an ontology can be used for the tag recommendation process (which is by now based on co-occurrences). In an ontology-based approach, the nature of the suggested tags could be made explicit, which would help the user to judge its appropriateness. For example, if a user types the tag “Graz”, an ontology-based system might suggest to also use the *broader-terms* “Styria” and “Austria”; another user choosing the tag “folksonomy” might be provided with the information that a folksonomy *is used by* “social software” and can then decide whether this tag should be added [Weller 07]. Fertilizing folksonomies with existing KOS is a promising approach to enable semantic enrichment. The key factor for success will be the availability of enough appropriate structured vocabularies. Angeletou et al. are developing algorithms to automatically map folksonomy tags to ontologies which are currently available on the Web to make semantic relations between tags explicit: “we can already conclude that it is indeed possible to automate the semantic enrichment of folksonomy tag spaces by harvesting online ontologies” [Angeletou et al. 07].

## 2.5 Distinguishing Different Tag Qualities and Purposes

Another peculiarity of folksonomies is that tags can be intended to fulfil different purposes. We do not only find tags referring to the documents content, but also to its author, origin, data format etc., as well as tags which are intended for personal (e.g. “toread”) and interpersonal (e.g. “@peter”) work management [Kipp 06]. Currently, all these tags are handled indifferently in folksonomy systems. That means, in our garden we have all different plants used for different purposes wildly mixed up (e.g. economic plants mixed with ornamental plants and medical herbs) – which of course makes it hard to find exactly what we need. Thus, our garden would need some additional structuring and – most of all – labelling. We need a way to distinguish the different tag qualities and label the tags accordingly (we may even decide to have different gardens, one for agriculture next to a flower garden and a herb garden and probably even a wine yard). While the clusters and hierarchies as discussed above

focus on the meaning of tags and should represent some kind of real-world knowledge in a structured way, this additional level regards the different purposes of tags. In practice, this distinguishing of different tag qualities might be done in form of facets, categories or fields. For each document different fields may be provided for tagging according to the different tag functionalities, e.g. one for content-descriptive tags, one for formal tags (or more specific one for the author, one for the document's file type etc.), and one for organizational tags (e.g. task organization, reference to projects). Alternatively, complex naming conventions could be established to specify the purpose of non-content-descriptive tags. Certain conventions are already coming up to use specific formats for labelling different tag purposes (like the "@" in "@name"-tags which are attached to documents to be forwarded to a colleague or friend).

At this stage, we have our garden weed-free, with nicely arranged flower beds and walking paths between different areas and now with different areas for differently used plants plus little information panels providing information about what the plants can be used for.

### 3 Community and Cross-Platform Tag Maintenance

#### 3.1 Gardening Principles for Communities

Manual Tag gardening is rather difficult to be done collaboratively by a community and within a shared tag collection – particularly if the system is not explicitly *collaborative* but rather profits from the *collective* participation of a community<sup>5</sup>. There is always the danger that one user destroys the arrangements another one has made, or that someone regards certain tags as weed while others consider them pretty flowers. Thus, the manual editing and weeding of tags should rather be done on a personal level. In this way, they allow each user to enhance the performance of his own tags. On the other hand, the collection of these individual approaches can be used for computing and providing tag recommendations – both during the tagging and the information retrieval process.

For communities, the use of automatic tools may generally be the more appropriate solution. At least for small communities (e.g. single working groups) the use of shared guidelines for tagging behaviour may be useful. In the context of small groups, collaborative tag gardening tools may also be reconsidered. If specific communication channels are provided, the community may agree on a shared structure for their tagging vocabulary. This is also the key for the collaborative engineering of ontologies in the sense of emergent semantics.

#### 3.2 Personal Tag Repository

On the personomy level, we do mainly need cross-platform solutions for tag maintenance and gardening. Someone, using different Web 2.0 tools in parallel might want to use his own terminology consistently across the different platforms. A potential solution would be a personal tag repository, an individual controlled vocabulary to be used independently with the different platforms. We envision a small tool which helps a user to collect, maintain and garden his very own tagging

---

[5] For a discussion on collaborative vs. collective tagging see [Vander Wal 08].

vocabulary. The user should be able to collect all tags he has used within different folksonomy systems (ideally with additional information on how often a single tag has been used in the different systems) and should then create his own vocabulary hierarchy, synonym collections and cross-references to related terms. Probably, at a later stage, such a tool could also be used for the exchange of terminologies within communities. Each user could take a walk in other users' gardens – and probably bring home some *cuttings* or *seeds* for his own one. A rather distant vision could then be the merging of two different personomies.

#### 4 Conclusions & Future Work

This article provides an overview on activities which help to maintain and enhance folksonomies. We have discussed formatting guidelines, vocabulary control, distinguishing of different tag qualities and combinations with other KOS as major activities for improving social tagging systems. Automatic and manual approaches should be combined. In future, we expect more and more of these aspects to be integrated to existing tagging systems. Our future work comprises the integration of tagging activities into collaborative ontology engineering processes and the critical investigation of semantic relations as a means for gradually enriching folksonomies. A personal tag repository as envisioned in chapter 3 is currently under development.

#### References

- [Aitchison et al. 06] Aitchison, J., Bawden, D., Gilchrist, A.: *Thesaurus Construction and Use* (4. ed.). London: Aslib (2004).
- [Au Yeung et al. 07] Au Yeung, C. Man, Gibbins, N., Shadbolt, N.: Understanding the Semantics of Ambiguous Tags in Folksonomies. In *Proceedings of the International Workshop on Emergent Semantics and Ontology Evolution (ESOE2007) at ISWC/ASWC 2007*. Busan, South Korea (2007), 108-121.
- [Angeletou et al. 07] Angeletou, S., Sabou, M., Specia, L., Motta, E.: Bridging the Gap Between Folksonomies and the Semantic Web. An Experience Report. In *Bridging the Gap between Semantic Web and Web 2.0, SemNet 2007* (2007), 30-43.
- [Braun et al. 07] Braun, S., Schmidt, A., Walter, A., Zacharias, V.: The Ontology Maturing Approach for Collaborative and Work Integrated Ontology Development. Evaluation, Results and Future Directions. In *Proceedings of the International Workshop on Emergent Semantics and Ontology Evolution at ISWC/ASWC 2007*. Busan, South Korea, (2007), 5-18.
- [Begelman et al. 06] Begelman, G., Keller, P., Smadja, F.: Automated Tag Clustering. Improving Search and Exploration in the Tag Space. In *Collaborative Web Tagging Workshop at WWW2006*, Edinburgh, Scotland (2006).
- [Furnas et al. 87] Furnas, G.W., Landauer, T.K., Gomez, L.M., Dumais, S.T.: The Vocabulary Problem in Human-System Communication. In *Analysis and a Solution*. Communications of the ACM, 30 (1987), 964-971.

- [Governor 06] Governor, J.: On the Emergence of Professional Tag Gardeners. Blog Post, 10.01.2006, retrieved from: <http://www.redmonk.com/jgovernor/2006/01/10/on-the-emergence-of-professional-tag-gardeners/> (2006).
- [Grahl et al. 07] Grahl, M., Hotho, A., Stumme, G.: Conceptual Clustering of Social Bookmarking Sites. In Proceedings of I-KNOW '07, Graz, Austria (2007), pp. 356-364.
- [Heymann & Garcia-Molina 06] Heymann, P., Garcia-Molina, H.: Collaborative Creation of Communal Hierarchical Taxonomies in Social Tagging Systems: Info-Lab Technical Report 2006-10, Stanford University (2006). Retrieved April 09, 2008, from <http://dbpubs.stanford.edu:8090/pub/2006-10>.
- [Hotho et al. 06] Hotho, A., Jäschke, R., Schmitz, C., Stumme, G.: Information Retrieval in Folksonomies. Search and Ranking. Lecture Notes in Computer Science, 4011 (2006), 411-426.
- [Kipp 06] Kipp, M.E.I.: @toread and cool: Tagging for Time, Task and Emotion. In 17th ASIS&T SIG/CR Classification Research Workshop. Abstracts of Posters (2006), 16-17.
- [Kolbitsch 07] Kolbitsch, J.: WordFlickr. A Solution to the Vocabulary Problem in Social Tagging Systems. In Proceedings of I-MEDIA '07 and I-SEMANTICS '07, Graz, Austria, (2007), 77-84.
- [Peters 06] Isabella Peters: Against Folksonomies. Indexing Blogs and Podcasts for Corporate Knowledge Management. In Online Information 2006, Conference Proceedings, London: Learned Information Europe Ltd (2006), 93-97.
- [Peters & Stock 07] Peters, I., Stock, W.: Folksonomy and Information Retrieval. Joining Research and Practice: Social Computing and Information Science. In Proceedings of the 70th Annual Meeting of the American Society for Information Science and Technology, Vol. 44, CD-ROM (2007), 1510-1542.
- [Peters & Weller 08] Peters, I., Weller, K.: Paradigmatic and Syntagmatic Relations in Knowledge Organization Systems. In Information – Wissenschaft und Praxis 59 (2008) 2, 100-107.
- [Schmitz 06] Schmitz, P.: Inducing Ontology from Flickr Tags. In Proceedings of the Collaborative Web Tagging Workshop at WWW'06 (2006).
- [Tanasescu & Streibel 07] Tanasescu, V., Streibel, O.: Extreme Tagging: Emergent Semantics through the Tagging of Tags. In Proceedings of the International Workshop on Emergent Semantics and Ontology Evolution (ESOE2007) at ISWC/ASWC 2007. Busan, South Korea (2007), 84-85.
- [Vander Wal 08] Vander Wal, T.: Keeping up with Social Tagging. In Workshop: Good Tags – Bad Tags. Social Tagging in der Wissensorganisation, Institut für Wissensmedien, Tübingen, Germany (2008). Video Presentation, retrieved from: <http://www.e-teaching.org/community/taggingcast>.
- [Weller 07] Weller, K.: Folksonomies and Ontologies. Two New Players in Indexing and Knowledge Representation. In H. Jezzard (Ed.), Applying Web 2.0. Innovation, Impact and Implementation. Online Information 2007 Conference Proceedings, London (2007), 108-115.
- [Zhang et al. 06] Zhang, L., Wu, X., Yu, Y.: Emergent Semantics from Folksonomies: A Quantitative Study. Lecture Notes in Computer Science, 4090 (2006), 168-186.



## Quality Metrics for Tags of Broad Folksonomies

**Céline Van Damme**

(MOSI, Vrije Universiteit, Brussel, Belgium  
celine.van.damme@vub.ac.be)

**Martin Hepp**

(E-Business and Web Science Research Group, Bundeswehr University  
München, Germany  
mhepp@computer.org)

**Tanguy Coenen**

(STARLab, Vrije Universiteit, Brussel, Belgium  
tanguy.coenen@vub.ac.be)

**Abstract:** Folksonomies do not restrict its users to use a set of keywords preselected by a group of experts for interpersonal information retrieval. We assume that the quality of tag-based information retrieval and tag suggestions when tagging a resource can be ameliorated if we are able to automatically detect the tags that have an intersubjective meaning or tags that are understood and used by many members of a group. In this paper, (1) we suggest three possible tag quality measures for broad folksonomies (2) by means of an analysis of a del.icio.us dataset, we provide preliminary evidence that the suggested metrics return useful sets of intersubjectively valid tags and (3) as an additional evaluation, we asked individuals to judge the tag sets obtained through the metrics.

**Key Words:** broad folksonomies, tag quality, metrics, del.icio.us

**Category:** H.3.5, J.4.

### 1 Introduction

Folksonomies involve their user community into the creation process of categories by inclusion of their tags. The absence of a controlled vocabulary, allows the community members to produce any category or tag that enters their mind. Since users are not restricted to a controlled vocabulary, the question arises concerning the quality of the tags and the information retrieval capacities of folksonomies. [Golden and Huberman 2006] showed that users primarily tag for a personal purpose. Tags used for a private purpose can in some cases be useful for the whole community, e.g. many people could say that they have *toread* a certain book when annotating a book at Librarything. However an annotated picture with the tag *ourdog* does not imply that it is a picture of everyone's dog.

We assume that the quality of tag-based information retrieval and tag suggestions when annotating new objects can be improved if we are able to automatically judge the quality of a tag in terms of the intersubjective comprehension it

engenders. We define intersubjective comprehension as the degree that a tag is understood by many members of a group.

In order to create metrics for an automatic tag quality judgement, we have to distinguish between the two kinds of folksonomies: broad and narrow folksonomies as classified by [Vander Wal 2005]. The difference between these types lies in the number of people that tag the object. In the case of broad folksonomies, a resource is tagged by many people (e.g. web pages) whereas in the case of narrow folksonomies there are only a few persons involved, in most situations only the author or creator of the resource (e.g. pictures on Flickr).

### 1.1 Related Work

Besides [Guy and Tonkin 2006][Sen et al. 2007] [Lee et al. 2007], research on quality of tags is scarce. In [Guy and Tonkin 2006] the authors focus on how they can help the user providing good and consistent tags. They suggest giving a sort of tag education to the user to improve the quality of the tags, for instance train the user to use singular terms. The authors in [Sen et al. 2007] as well as in [Lee et al. 2007] propose to extend tags with a kind of rating mechanism. In [Lee et al. 2007] the user has to tag a resource as well as add a positive (e.g. people like) or negative context (e.g. people do not like) to each tag (e.g. people do not like war). Positive and negative contexts are respectively indicated with a plus and minus sign. Different tag rating scenarios are tested and discussed in [Sen et al. 2007]. The authors conclude with a number of design guidelines for the creators of websites that contain a tagging mechanism.

Still, asking individuals to rate tags as a quality measure is time-consuming. An interesting alternative would be to automatically detect intersubjective tags and regard intersubjectivity as an indicator of quality.

### 1.2 Contribution and Overview

In this paper, (1) we suggest three possible tag quality measures for broad folksonomies, (2) by means of an analysis of a del.icio.us dataset, we provide preliminary evidence that our suggested metrics return useful intersubjective tag sets and (3) as an additional evaluation, we asked individuals to judge the tag sets obtained by applying the metrics.

The structure of this paper is as follows: in section 2, we give an overview of the metrics. We elaborate on the implementation issues of the metrics applied to a del.icio.us dataset in section 3. In section 4, we discuss the evaluation of the metrics. We give some limitations of the research in section 5 and end with a conclusion and discussion of future research in the last section.

## 2 Metrics

In this section, we present three metrics to automatically select the intersubjective tags from a set of tags used to annotated a web resource by the principles of broad folksonomies. For each metric, we give a description, explain how it can be calculated and motivate why we propose it.

### 2.1 Metric 1: High Frequency Tags

#### 2.1.1 Description

We select tags with the highest 5 frequencies.

#### 2.1.2 Calculation

For each tagged resource, we order the tags and count the frequency of each distinct tag. We then choose the tags with the highest 5 frequencies.

#### 2.1.3 Motivation

Because many people express their thoughts about a particular resource through their selection of tags, we propose to analyze high frequency tags.

### 2.2 Metric 2: Tag Agreement

#### 2.2.1 Description

We define tag agreement for resource  $x$  as the tags that are selected by more than 50% of the users who have tagged resource  $x$ .

#### 2.2.2 Calculation

We first determine the frequency of each unique tag. Then, we calculate the number of users that have tagged each resource. The tag agreement is consequently calculated by dividing the tag frequency by the number of users that have tagged a resource, and multiply the result by 100 in order to have a percentage as a result. When all the users agree on a certain tag, this number should be equal to 100%. The closer to 0%, the less the users agree on that particular tag.

#### 2.2.3 Motivation

Decisions in various areas of human activity are often taken on the basis of a majority: more than 50% of the people have to agree on a certain proposal in order to get it accepted. Tagging in the case of broad folksonomy can be seen as a way of voting for the semantic labeling of a resource and this is why we suggest tag agreement as a second metric.

## 2.3 Metric 3: TF-IRF

### 2.3.1 Description

For each tag we calculate its Tag Frequency Inverse Resource Frequency or TF-IRF weight and select tags with the highest 5 TF-IRF scores. We derived the TF-IRF metric from Term Frequency Inverse Document Frequency or TF-IDF, a common metric in the domain of automatic indexing for finding good descriptive keywords for a document. When selecting the appropriate tags for a certain document, the TF-IDF formula takes the intra as well as inter document frequency of keywords into account. The higher the TF-IDF weight, the more valuable the keyword.

### 2.3.2 Calculation

*Corpus*: To calculate the TF-IRF weights, we need a corpus of similar resources. This can be obtained through tag clustering. By calculating the co-occurrences of tag pairs and transforming the pairs (as nodes) and their co-occurrences (as weighted edges) into a graph, we can apply the Markov Clustering (MCL) Algorithm [Van Dongen 2000]. Results in [Van Dongen 2000] show that the MCL algorithm is very good and highly performant for clustering graphs. Therefore, we choose this algorithm to build the corpus.

*Calculating TF-IRF*: In order to transform TF-IDF into TF-IRF, we have to make some adjustments to the formula. We have to exclude the textual information or documents from our formula since tagged resources are not always textual (e.g. an mp3 audio file). The only data we can analyse are tags. As a consequence, we suggest the equation below to calculate the TF-IRF weight for a certain tag annotated to a resource. The formula is based on TF-IDF (with  $t_{x,y}$  = frequency of  $tag_x$  for  $resource_y$ ,  $T_y$  = total number of tags for  $resource_y$ , corpus = sum of resources and  $R_x$  = sum of resources that have  $tag_x$ ).

$$TF - IRF(tag_{x,y}) = \frac{t_{x,y}}{T_y} * \log\left(\frac{|corpus|}{R_x}\right)$$

### 2.3.3 Motivation

In the domain of automatic indexing a lot has been written on how to select the most appropriate keywords. Research on automatic indexing dates back to the 1950's and consequently represents a large body of knowledge. We believe it is interesting to apply TF-IDF, which is one of the common techniques in this area, to broad folksonomies.

### 3 Data set

For the analysis we used the Del.icio.us dataset from [Laniado et al. 2007]. It contains more than 3.400.000 unique bookmarks from around 30.000 users retrieved in March 2007.

#### 3.1 Preparing Steps

In order to be able to compare the results from the different metrics, we had to calculate each metric on the same collection of bookmarks. Since the TF-IRF metric required the creation of a corpus or a set of related resources, in this case bookmarks, we started the analysis by making the corpus.

##### 3.1.1 Cleansing

Before we could apply the MCL algorithm to build the corpus, we had to do some data cleaning. We

- Removed all the English stop words from the tag set, since most of the high frequency tags of the dataset were in English.
- Stemmed the remaining tags by removing the end suffix. Words that have the same stem or root are considered to be referring to the same concepts (e.g. running and run have the same stem, i.e. run).
- Merged duplicate tags since a duplication of tags appeared after stemming.
- Disregarded all bookmarks that are tagged by less than 100 users. Because we only want to include bookmarks that are evaluated by a large number of users. We reduced the number of bookmarks in the collection to 3898.
- Calculated the co-occurrence of the tag pairs.

##### 3.1.2 Applying MCL Algorithm

We applied the MCL algorithm on all tag pairs and their corresponding frequencies obtained in previous step. We excluded the tag pairs with a frequency of less than 100. This means both tags have been used less than 100 times together to tag a particular resource. A lower threshold value did not result in clearly distinguishable clusters. We opted for the cluster which contained the following tags: entertainment, film and movie since these tags are common used terms. We decided to include a bookmark in the corpus if it had at least one tag with a frequency of 10 that belonged to this cluster. We opted for a number of 10 since we wanted to be sure that a link with the cluster existed. As a result, we obtained 127 bookmarks for this cluster.

URL	Metrics <sup>1</sup>
http://www.imdb.com	M1: movie film refer database entertainment M2: movie M3: database cinema movie film refer
http://www.ifilm.com	M1: video movie film entertainment fun M2: video M3: video trailer film ifilm movie
http://www.apple.com/trailers/	M1: movie trailer entertainment apple film M2: movie M3: trailer apple quicktime importediefavorites movie

**Table 1:** Tag sets obtained by applying the metrics

### 3.2 Results

We calculated the tag sets for each metric and bookmark in the cluster. Some examples of the results are included in table 1. For each metric, we ordered the tags from the left to the right based on decreasing values. We noticed a close linkage between the tag sets obtained by the first and third metric. In some cases, the high frequency tags and TF-IRF metrics only differ by the order of the tags. In the other cases, there is a close overlap between metric 1 and 3 because they often share similar tags.

When applying the tag agreement metric on the dataset we received, we noticed that the average number of tags per bookmark where agreement exists was very low. The minimum and maximum values lay between 0 and 3, and the modus and median both had a value of 1. It was therefore not possible to select 5 tags for the tag agreement metric since there were on average 0.94 tags per bookmark that correspond to the definition. There were even 26 bookmarks that did not have any tags confirming to this pattern. After excluding these 26 bookmarks, the mean increased just slightly to 1.18. There was a very weak negative correlation between the number of tags retrieved by the tag agreement metric and the number of users that have tagged the object ( $\rho = -0.17$ ). This means that an increase in the number of users that tag a certain resource will slightly decrease the number of tags that comply with the tag agreement metric.

## 4 Preliminary Evaluation

To answer the question which metric is generating the best results, we decided to set up an online survey. To conduct the survey we created a tool in PHP that

<sup>1</sup> M1 = High Frequency Tags; M2 = Tag Agreement; M3 = TF-IRF

chooses a bookmark as well as its tag sets randomly from the MySQL database. In each session 10 bookmarks had to be evaluated. There were 101 bookmarks in the database, because we excluded the 26 bookmarks that did not have any tags for the tag agreement metric.

Since our cluster of bookmarks was selected based on the requirement of sharing one of the tags (entertainment, film and movie) with a frequency of 10, we asked an international group of 20 students to participate in the online survey. We asked them to select the tag set of which they thought it did the best job at describing a specific bookmark. In case of doubt, we told them to take the order of the tags into account. Indeed, a tag placed at the beginning, was more important than one which is located more to the right.

First, we gave the students a one hour presentation on tagging and folksonomies. We also introduced them to Del.icio.us and gave a brief demonstration of the system. Then, we invited them to a computer room to participate in the survey.

Due to randomness, 75 of the 101 bookmarks were evaluated and some of them were assessed several times. In total, 173 times a bookmark was evaluated. We did not obtain the logical number of 200 (20 students doing 10 evaluations), since (1) some of the websites were down during the survey and had to be removed from the result list and (2) not all the students pursued the survey until the end. On average, the students opted in 52.6% of the cases (n=91) for the high frequency tags metric and in 41% of the cases (n=71) for the TF-IRF metric. The tag agreement scored poorly: in 6.3% (n=11) they selected this option. A possible explanation for this might be the low number of tags.

## 5 Limitations of the Research

Although we obtained first preliminary results, there are certain limitations that apply to the online survey. We did not ask the students why did they opted for a certain tag set. The students were not asked whether the chosen tag set contained all tags, too many tags or not enough tags and the number of participants was too low.

## 6 Discussion and Conclusion

In this paper, we proposed three metrics to automatically detect the intersubjective tags for a resource tagged in the context of a broad folksonomy. We applied the three metrics to a Del.icio.us dataset and through an online survey we tried to find the most appropriate metric. Preliminary results show that the High Frequency Tag metric generates the best results. However, the TF-IRF metric also produces valuable results.

In the near future, we want to set up a large scale online survey to find out if the results suggested in this small-scale setup can be reproduced in a larger scale survey. Further, we want to find out what the characteristics are of the tags that comply with the metric. For instance, how many of the tags are general or specific.(3) In a next step, we plan to extend this research to the case of narrow folksonomies.

## References

- [Guy and Tonkin 2006] Guy, M., Tonkin, E.: "Tidying up Tags?"; *D-Lib Magazine*, 12, 1 (2006)
- [Golden and Huberman 2006] Golder, S. and Huberman, B. A.: "Usage patterns of collaborative tagging systems"; *Journal of Information Science*, 32, 2(2006),198-208
- [Laniado et al. 2007] Laniado, D., Eynard, D., Colombetti, M.: "Using WordNet to turn a Folksonomy into a Hierarchy of Concepts"; *Proc. SWAP, CEUR, Bari (2007)*
- [Lee et al. 2007] Lee, S., Han, S.: "Qtag: introducing the qualitative tagging system"; *Proc. Hypertext, ACM Publishing, Manchester (2007)*, 35-36.
- [Luhn 1958] Luhn, H.P.: "The automatic creation of literature abstracts"; *IBM Journal of Research and Development*, 2 (1958), 159-165.
- [Salton et al. 1974] Salton, G., Yang, C.S., Yu, C.T.: "A Theory of Term Importance in Automatic Text Analysis"; *Journal of the American Society for Information Science*, 26, (1974), 33-44
- [Sen et al. 2007] Sen, S., Harper, F., LaPitz, A., Riedl, J.: "The Quest for Quality Tags"; *Proc. Supporting group work, ACM Publishing, Sanibel Island (2007)*, 361-370
- [Van Dongen 2000] van Dongen, S.: "Graph Clustering by Flow Simulation"; PhD thesis, University of Utrecht, (2000).
- [Vander Wal 2005] Vander Wal, T.: "Explaining and Showing Broad and Narrow Folksonomies"; (2005) [http://www.personalinfocloud.com/2005/02/explaining\\_and\\_.html](http://www.personalinfocloud.com/2005/02/explaining_and_.html)



## **Conceptual Interpretation of LOM and its Mapping to Common Sense Ontologies**

**M. Elena Rodríguez, Jordi Conesa**  
(Universitat Oberta de Catalunya, Barcelona, Spain  
mrodriguezgo@uoc.edu, jconesac@uoc.edu)

**Elena García-Barriocanal, Miguel Ángel Sicilia**  
(Universidad de Alcalá de Henares, Madrid, Spain  
elena.garciab@uah.es, msicilia@uah.es)

**Abstract:** In this paper we discuss about semantics of Learning Object Metadata (LOM) standard as result of using ontologies. We also propose improvements to LOM that deal with extensions and resolution of its semantic ambiguities. The paper also presents a mapping between LOM and a common sense ontology that promotes semantic interoperability among heterogeneous learning systems.

**Keywords:** LOM standard, learning object, common sense ontology, semantic interoperability  
**Categories:** L.1.2, L1.3, D.2.12.

### **1 Introduction**

The use of Information and Communication Technologies (ICT) in education has widened the feasibility of e-learning. Amongst others, e-learning needs to encompass both pedagogic and technological issues. Technological aspects include hardware and software architectures based on standards and specifications that make possible the interoperability, i.e., the search, access, reusing, sharing and interchange of learning resources (also known as Learning Objects (LOs)) and services between learning systems. Standards and specifications provide a common language. The more rigorous and formal the language is, the better semantic interoperability levels we obtain.

In this paper we are interested in metadata standards that allow the description of LOs. More specifically, our goal is to increase their semantic expressiveness by means of ontologies in order to improve searching capabilities and to facilitate the annotation of LOs. Achieving this objective requires answering research questions as: “Can we use ontologies to disambiguate, validate and improve metadata standard definitions?” and if so, “Can we establish mappings between ontologies and a given standard to make that standard even more generalized?” Along the paper the reader will find the answers and results associated to the previous questions.

The paper is structured as follows: Section 2 briefly describes the LOM standard, revises the LO concept and justifies the benefits to cope with LOs categorizations. This section also presents an overview of different ontology kinds and the advantages and requirements that an ontology in the LOs context has to have, justifying why we use OpenCyc as the support ontology. Section 3 describes our results of comparison

between LOM and OpenCyc, presenting a LOs categorization and its mapping to OpenCyc. Last section concludes the paper and presents the further work to be done.

## 2 Related Work

### 2.1 The IEEE LOM Standard, the LO Concept and LO Types

The LOM standard [IEEE LOM 2002] defines, grouped in categories, the metadata elements set to be used for describing LOs. All metadata elements are optional and they are structured hierarchically. We can find generic metadata elements (general category), metadata that describe the history and current state of a LO (life cycle category), data elements over specified metadata elements (meta-metadata category), metadata that describe LO technical requirements (technical category), metadata elements stating pedagogical characteristics of the LO (educational category), metadata that deal with legal issues (rights category), metadata that allow specifying relationships between LOs (relation category), metadata for adding comments about LO use experiences (annotation category) and metadata that define the LO according to a classification system (classification category).

A LO, in terms of LOM standard, is defined as any entity, digital or non-digital, that may be used for learning, education or training. This broad definition is complemented and/or restricted by some authors in order to clarify the LO meaning and boundary. For example Wiley [Wiley 2002], restricts LOs to be digital entities and emphasizes its potential to be reused in different learning experiences. On the other hand, Polsani [Polsani 2003], to the reusing requisite, adds the need to provide instructional context to LOs.

Despite the efforts, a main problem when working with LOs, is they are fairly heterogeneous with regards to several characteristics. Moreover, current specifications and implementations do not include the notion of LO type, according to different criteria, as guiding method for metadata structuring [Sicilia et al. 2004]. For example, all metadata in LOM are equally applicable to any type of LO, irrespective of its type. In fact, attending to metadata elements, we can specialize LOs with regards to several characteristics as, for example, their internal structure (1.7 *structure element*), granularity (1.8 *aggregation level*), interactivity form (5.1 *interactivity type*) or kind of learning resource (5.2 *learning resource type*). Other interesting distinction between LOs can be made from a conceptual point of view. When we design and develop LOs we can distinguish, at least, between the LO understood as creative work, and the different representations we offer from this creative work, i.e. the final digital or physical content elements (i.e. the existing LOs).

Exploiting the LO type notion inside ontologies is key for different processes that can be (semi-)automated, given that LO types determine the reasoning processes that are applicable to each kind of LO. Some examples of these processes are: 1) annotation: the distinction between kinds of LOs clarifies annotation process. Some metadata are only applicable to a specific LO type, while some other metadata can be automatically derived depending on the LO type. 2) Location: the kind of a LO can act a discriminator element in LOs searches in large repositories. 3) Composition and sequencing: the LO kind can be used for LOs combination and for specifying their

use restrictions in a learning experience. And 4) personalization: types of LOs constitute a way to model user preferences about kind of LO, interaction styles, etc.

## 2.2 Ontologies and their Role in E-learning

Ontologies may be seen as the representation of one view of the world. As more consensus exists in ontologies, more uses they have. Ontologies may be classified in several ways [Guarino 98]. Depending on the kind of knowledge that they represent, ontologies may be classified in application ontologies, which describe one or more domains in a very particular context. Obviously, this kind of ontologies cannot be reused out of the context they belong. Another more general kind of ontologies is domain and task ontologies. These ontologies deal with a given generic domain or task, respectively. They are more reusable than the previous ones, but the problem is that their reusability is limited according to the domains of interest, i.e., if we are interested in dealing with more than one domain, they may be incomplete. The third kind of ontologies is the upper-level ontologies, which represent general knowledge applicable to several domains. There is also another type of ontology, usually called common sense ontology. These ontologies include general, domain, task and application knowledge and are usually larger than the others and more reusable in general. However, to reuse them we need to overcome the usability problems of dealing with large ontologies. We plan to use ontologies with two main purposes:

1. Checking and validating LOM, as well as finding possible improvements (extensions and resolution of semantic ambiguities).
2. Establishing a correspondence between the LOM metadata and the ontology. This will improve semantic interoperability of LOs and services among learning systems ([Aroyo et. al 2006], [Sicilia 2006], [Sánchez et al. 2007]).

For the first purpose we need an ontology that describes the domain (or domains) that are dealt in LOM. For the second objective we need an ontology which provides reliable background and top level concepts. Since LOM deals not only with the educative data, but also with the different ways of tracking changes, composing objects, authorship etc. we cannot choose a single domain ontology to accomplish the first purpose. Hence, we use one of the most well known and large ontologies available: the common sense Cyc (encyclopedia) ontology [Lenat 95]. Cyc is an ontology that contains more than 2.2 million assertions (facts and rules), describing more than 250,000 terms and including 15,000 predicates.

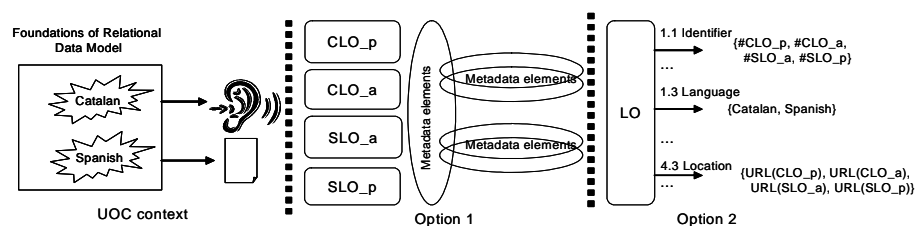


Figure 1: UOC motivating example and Los annotation options

Since Cyc is a commercial product, we decided to use OpenCyc (<http://www.cyc.com/cyc/opencyc>) for that work. OpenCyc is a public version of Cyc and contains a subset of the Cyc knowledge. Even being a subtype of Cyc, we believe that OpenCyc contains enough knowledge to deal with our validation and mapping processes.

### 3 A Conceptual Framework for LO and its Mapping to OpenCyc

#### 3.1 Motivating Example: the UOC Case

As we have stated, LOM does not explicitly consider the possibility of specialize LOs. In addition, we find very specific metadata (implementation-dependent metadata) merged with very general metadata that are independent of a concrete final implementation of a LO.

For example, the UOC, which stands for Open University of Catalonia in Catalan, is a virtual university that, amongst others, offers several degrees related with ICT knowledge field. Moreover, the UOC offers learning in two linguistic environments (Catalan and Spanish). Let consider a subject as “Introduction to Databases” where, amongst others topics, we explain the foundations of the relational data model. Imagine we have a lesson in narrative text that explains the previous topic. This lesson is available both in Catalan and Spanish languages. We also assume that, for each language, it is available in textual (pdf) and audio format (Figure 1 shows the described situation). Assuming the UOC has selected its own metadata set of interest, when we proceed to annotate LOs, we have many different options. Without loss of generality, let examine two possible options (depicted also in Figure 1).

In option 1, four metadata records for describing LOs are created, one for each available LO, according to available languages and formats. Each metadata record includes all metadata selected by the UOC. We want to point out the following issues: 1) some metadata values depend on the language of the available LO under consideration (e.g. general metadata as 1.2 *title*, 1.3 *language*, 1.5 *keywords* etc.). 2)

Option 1	Option 2
<ul style="list-style-type: none"> <li>- Too many data redundancy; therefore hard data maintenance</li> <li>- All metadata are defined at the same level</li> <li>- There is not way to know that, for example, CLO_p and SLO_p are the same content with different languages even when relationship between them is defined. The problem is that “is based on” or “is version of” LOM relationships do not have the right semantic. “Translation of” should be a more appropriate relationship type but it is undefined in LOM</li> <li>- Changes in content imply new available LOs and then new metadata records</li> <li>+ Annotation fits LO definitions (each available LO is a LO, and each available LO has its own metadata record)</li> </ul>	<ul style="list-style-type: none"> <li>- Very unclear and confusing conceptual structure</li> <li>- All metadata are defined at the same level</li> <li>- Different semantic interpretations, for example: one LO that uses two languages or two LO, each one with different languages?</li> <li>- Adhoc heuristics are needed to interrelate metadata values, i.e., to know which is the location of the Spanish versions</li> <li>- It is not clear, for example, what to do when a change in the content involves only one of the languages</li> <li>- Annotation does not fit LO definitions (except LOM LO definition in a very broad sense interpretation “[...]as any <i>entity</i>, digital or <i>non-digital</i>, that may be used for learning [...]”)</li> <li>+ No data redundancy</li> </ul>

Figure 2: Disadvantages and advantages of each annotation option

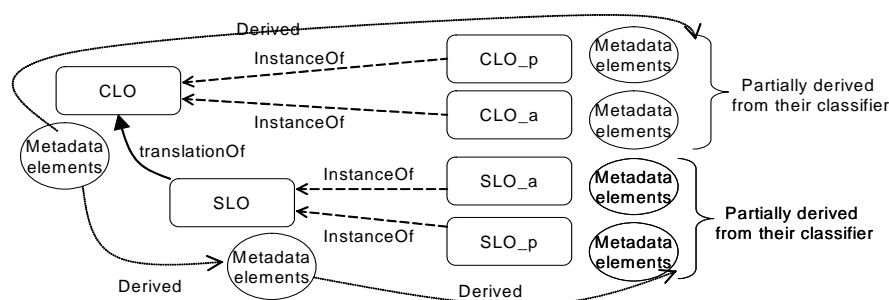


Figure 3: LO types and relationships

Some other metadata depend on the implementation format (as technical metadata: 4.1 *format*, 4.4 *requirements* etc.) of the available LO. 3) Other metadata remain invariable, independently of previous aspects (1.6 *coverage*, 1.7 *structure*, 1.8 *aggregation level* most of educational metadata etc.). And 4) relationships among available LOs (not shown in the figure) can be declared though metadata contained in the relation category.

In option 2, only one metadata record is provided for the four available LOs. This option takes advantage on the fact that some metadata can be multivalued. For instance, metadata element 4.3 *location* (technical category) stores the URL where the available LOs can be download, metadata element 1.3 *language* the human languages within the LO to communicate to the intended user etc.

Figure 2 summarizes the most relevant disadvantages (-) and advantages (+) of the presented options for LOs annotation.

### 3.2 An Interpretation of LOM

In that subsection we present our conceptual interpretation of LOM which is the result of a comparison based on the intensive study we have done between LOM and the part of OpenCyc that deals with their semantic equivalent (or almost close) concepts.

Our interpretation is driven by a LOs specialization which takes as criteria the nature, from a conceptual point of view, of LOs. At a first level, we can distinguish between the conceptual LOs (the abstract LOs consequence of a creative work) and existing LOs (the available LOs). In addition, we can identify, at least, a subtype of conceptual LOs, i.e. the conceptual LOs that we can derive as versions of the previous conceptual LO (i.e. copy conceptual LOs). In a similar way, metadata elements are filtered according to their abstraction level, distinguishing which metadata are applicable to each LO type. In addition, applicable metadata are classified in basic and derived. Only basic metadata elements will potentially receive value(s). The value(s) of derived metadata are inferred trough the appropriate relationships.

CLO represents the abstract LO conceived in Catalan (conceptual LO). Metadata elements related to the intellectual process of design and creation only need to be defined for CLO. Examples include general category metadata (e.g. 1.6 *coverage*, 1.7 *structure*, 1.8 *aggregation level*) as well as (amongst others), metadata belonging to educational and classification categories. Given that CLO is a textual LO, it is also

required to specify language dependant metadata elements (like 1.2 *title*, 1.3 *language*, 1.5 *keywords*, etc.) which take value according to Catalan language.

The translation of CLO to Spanish gives rise to SLO (copy conceptual LO, subtype of conceptual LO). Some of the SLO metadata will be derived from CLO through the “translation of” relationship type. Only language dependent metadata are considered basic.

We believe it is important to know the original language of a creative work and therefore to store its original language within the conceptual LO. The reason is that we believe that the original language, which is closely related to work cultural aspects, is a fundamental part of the conceptual LO nature.

Each conceptual LO is available in two different formats: audio and pdf. Therefore, four new LOs representing the available LOs for SLO and CLO are created: CLO\_p, CLO\_a, SLO\_p and SLO\_a (existing LO which are related with their conceptual LO by the “instance of” relationship type). CLO\_a and CLO\_p derive most of their metadata from their conceptual LO (CLO). Metadata to be considered basic are the implementation-dependent metadata as, for example, it would be the case of technical category metadata elements. The same happens to SLO\_p and SLO\_a regarding to SLO. “Instance of” and “translation of” relationship types constitute an extension of relationships proposed by LOM in the relation category; in fact, they are a specialization of the “is based on” LOM relationship.

It is important to note that not only translations trigger new copies of a conceptual LO. For example, imagine that in “Introduction to Databases” students must practice SQL over a relational DBMS as PostgreSQL. In this case, we can also differentiate among the common characteristics associated to PostgreSQL (conceptual LO), the properties associated to specific PostgreSQL versions (copy conceptual LO), and the available instances of PostgreSQL (existing LO). Oracle DMBS will correspond to a different conceptual LO.

### 3.3 Mapping to OpenCyc

This subsection presents a possible mapping of our LOs categorization to OpenCyc (see Figure 4). We also sketch some suggested mappings between LOM metadata and OpenCyc.

A mapping between a concept of OpenCyc and a concept of LOM means that all the instances of the OpenCyc concept are instances of the LOM concept and vice-versa. Obviously, the inheritance relationship of Conceptual LO and Copy Conceptual LO also exists in the corresponding concepts of OpenCyc (CW is supertype of TSW). Hence, the predicate `instantiationOfWork` defined in the context of CW is also applicable to TSW. One of the subtypes of CW, for example, is `DatabaseProgram`. This reinforces our statement that a computer program as a DBMS may be seen as a Conceptual LO. In that example, we would be able to create an instance of `DatabaseProgram` for the PostgreSQL. There are also subtypes of IBT that deals with running computer programs, such as `ComputerIBT`.

Some mappings depend on the level of abstraction of LO where are applied. For example, when dealing with the CW, the predicate `languageOriginalWrittenIn` describes its native language. In the other cases the language is not the original one and therefore it is represented by `languagePublishedIn`. The selected OpenCyc

predicate relates LOs with Language instead of Human Language. That makes possible to represent a LO that use formal (but not human) languages such as the mathematical language.

LO root concepts	Learning Object Conceptual Learning Object Copy Conceptual Learning Object Existing Learning Object	OpenCyc	Thing Conceptual Work (=CW) Textual Specific Work (=TSW) Information Bearing Thing (=IBT)
LO root relationships	Conceptual LO <-> Copy LO Conceptual LO <-> Existing LO		translationOfPCW( CW, TSW ) instantiationOfWork( CW, IBT )
Some LOM Metadata Examples	12_title 13_language 16_coverage		titleOfWork( CW, String ) { languageOriginallyWrittenIn( CW, Language ) languagePublishedIn( CW, Language ) } timeIntervalInclusiveFn { culturalRegionOfOrigin Other cultural information derived from the applicable microtheories

Figure 4: Excerpt of the mapping between LOM Standard and OpenCyc

All derived metadata of our proposal may be also inferred within OpenCyc by using rules. As an example, in the following we present the OpenCyc rule that derives the title of existing LOs from their conceptual version:

```
(#$implies
  (#$and (#$isa ?IBT #$InformationBearingThing)
    (#$isa ?CW #$ConceptualWork)
    (#$instantiationOfWork ?IBT ?CW)
    (#$titleOfWork ?CW ?T))
  (#$titleOfWork ?IBT ?T))
```

## 4 Conclusions and Future Work

Under our point of view, this work answers the research questions stated in the introduction by using OpenCyc in order to study possible improvements to LOM. As a result, we have created a conceptual framework that categorizes LOs and their metadata. In addition, new meaningful relationships between LOs have been added to LOM. Our conceptual framework provides several advantages to the current ambiguity in LOM: 1) it refines LOM and serves as a guideline to annotate LOs. 2) It avoids redundancy and facilitates data maintenance thanks to a clear separation of concerns. 3) It allows to make inferences due to its semantic expressiveness. And 4) it differentiates LO types, permitting to improve search mechanisms.

At first glance, it may look that our conceptual framework may increase the number of necessary LOs. So would imply to consider more metadata records to describe LOs. Even though, this is a relative drawback because most of the metadata are derived. In fact, we can say that the number of basic metadata is at most the same

in our interpretation versus the others. In other cases, we have less basic metadata because other interpretations may tend to repeat metadata and do not allow metadata derivation.

A possible mapping between OpenCyc and an excerpt of LOM standard also has been presented in this paper. That mapping may improve the interoperability between heterogeneous learning systems, helping to deal with some disambiguation and to establish a conceptual framework. The fact of using a very large ontology such as OpenCyc allows further inferences, such as reusing the taxonomy that this ontology has. For example, the taxonomy that may be extracted from OpenCyc that deals with Conceptual LO (CW) consists of more than 4,000 concepts.

In near future, we plan to complete the mapping between LOM and OpenCyc in all levels. That means, to rewrite the derivation rules within OpenCyc and try to see how we can reuse the taxonomies of OpenCyc in order to derive other relevant specializations of LOs, such as for example one based on the kind of learning resources (LOM 5.2 *learning resource type* metadata element).

### Acknowledgements

This work has been supported by the PERSONAL research project (TIN2006-15107-C02-01/02 (UOC/UAH)) funded by the Spanish Ministry of Education.

### References

- [Aroyo et al. 2006] Aroyo, L., Dolog, P., Houben, G.J., Kravcik, M., Naeve, A., Nilsson, M., and Wild F.: "Interoperability in Personalized Adaptive Learning"; *Educational Technology & Society*, 9, 2 (2006), 4-18.
- [Guarino 98] Guarino, N.: "Formal Ontology and Information Systems"; *Proc. FOIS'98*, IOS Press (1998), 3-15.
- [Lenat 95] Lenat, D.B.: "Cyc: A Large-Scale Investment in Knowledge Infrastructure"; *Communications of the ACM*, 38, 11 (1995), 33-38.
- [IEEE LOM 2002] LTCS WG12: "IEEE Learning Technology Standards Committee. Draft Standard for Learning Object Metadata"; Technical Report 1484.12.1, IEEE Inc, (2002).
- [Polsani 2003] Polsani, P.R.: "Use and abuse of reusable learning objects"; *Journal of Digital Information*, 3, 4 (2003).
- [Sánchez et al. 2007] Sánchez-Alonso, S., Sicilia, M. A., Pareja, M.: "Mapping IEEE LOM to WSML: an ontology of learning objects". *Proc. ITA'07*, (2007), 92-101.
- [Sicilia et al. 2004] Sicilia, M.A., García, E., Sánchez-Alonso, S., Rodríguez, M. E.: "Describing learning object types in ontological structures: towards specialized pedagogical selection". *Proc. ED-MEDIA'2004*, (2004), 2093-2097.
- [Sicilia 2006] Sicilia, M.A.: "Metadata, semantics and ontology: providing meaning to information resources"; *International Journal of Metadata, Semantics and Ontologies*, 1, 1 (2006), 83-86.
- [Wiley 2002] Wiley, D.A.: "Connecting learning objects to instructional design theory: A definition, a metaphor, and a taxonomy"; In D. A. Wiley (ed.), *The Instructional Use of Learning Objects: Online Version*, (2002).



## **Collaborative Knowledge Engineering via Semantic MediaWiki**

**Chiara Ghidini, Marco Rospocher, Luciano Serafini**

(FBK-irst. Via Sommarive 18 Povo, 38050, Trento, Italy  
ghidini@fbk.eu, rospocher@fbk.eu, serafini@fbk.eu)

**Barbara Kump, Viktoria Pammer**

(Knowledge Management Institute, TU Graz. Inffeldgasse 21a, 8010 Graz, Austria  
bkump@tugraz.at, viktoriam.pammer@tugraz.at)

**Andreas Faatz, Andreas Zinnen**

(SAP Research, SAP AG. Bleichstraße 8, 64283 Darmstadt, Germany  
andreas.faatz@sap.com, andreas.zinnen@sap.com)

**Joanna Guss**

(EADS France - Innovation Works. 12, Rue Pasteur - BP76 - 92150 Suresnes - France  
Joanna.Guss@eads.net)

**Stefanie Lindstaedt**

(Know-Center<sup>1</sup>. Inffeldgasse 21a, 8010 Graz, Austria  
slind@know-center.at)

**Abstract:** Formal modelling is a challenging and expensive task, especially for people not familiar with design techniques and formal languages. In this paper we present the modelling experience within the APOSDLE EU-project, and we describe the methodology we have developed to support an integrated modelling process of the ontologies and workflows inside APOSDLE. Our approach is based on two main pillars: (i) the usage of Semantic MediaWiki (SMW) for the collaborative development of informal models to be created, and (ii) a tight, and as much as possible automatic, integration of the SMW with tools for formal modelling, in order to reuse the informal models for formal models creation.

**Key Words:** modelling methodology, semantic web technology, APOSDLE project

**Category:** I.2.4

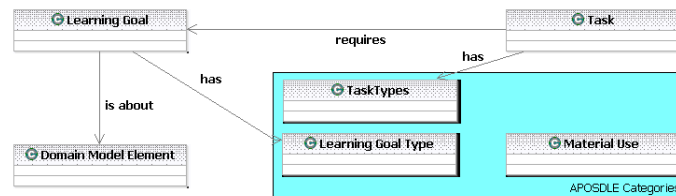
### **1 Motivation and Related Work**

The spread of Semantic Web technology, and Service Oriented Computing has the effect that many tools rely on the availability of formal models of specific domains (for instance in the form of an ontology), of business processes (for instance in the form

---

<sup>1</sup> The Know-Center is funded within the Austrian COMET Program - Competence Centers for Excellent Technologies - under the auspices of the Austrian Ministry of Transport, Innovation and Technology, the Austrian Ministry of Economics and Labor and by the State of Styria. COMET is managed by the Austrian Research Promotion Agency FFG

of a BPMN<sup>2</sup>, or YAWL<sup>3</sup> workflow) and learning goals to achieve particular competencies. Formal modelling is a challenging and expensive task, especially for people not familiar with formal languages. Thus, organisations interested in using a system are first asked to develop the formal models of their domains without having the expertise to do so. In this paper we present the experience of tackling this problem within the EU project APOSDLE<sup>4</sup>, and the methodology developed to support an integrated modelling process of ontologies and workflows by APOSDLE. APOSDLE aims at developing a software platform to support the process of *learning@work*, i.e. learning within the context of the immediate work of a user and within her current work environment. Presenting context-sensitive learning material, tailored to specific needs, the APOSDLE system needs to know not only the profile of the user, but also the context in which the user is acting: the tasks a user can perform, the learning goals required to perform the tasks, and a description of the domain of affairs (application domain) of the organisation. Most of this knowledge is contained in the APOSDLE Knowledge Base, that we illustrate in figure 1.



**Figure 1:** The APOSDLE Knowledge Base

Of the four models contained in the Knowledge Base, *the APOSDLE Categories* (used to classify tasks, learning goals and learning material) are APOSDLE-inherent, while the *task model*, *domain model*, and *learning goal model* are entirely organisation/domain dependent and need to be provided every time the system is deployed for a new domain. From the typical application environment of our system, we can assume:

1. most of the organisations won't have these formal models already available;
2. most likely, they will not even be interested in defining such models, as their main interest is in setting up a tool enhancing their work environment's productivity;
3. we need to model in an integrated way both specific static domains and learning goals, in the form of OWL ontologies, and business processes, in the form of YAWL workflows. This requires the organisation to become acquainted with more than one formal representation language.

<sup>2</sup> Business Process Modelling Language

<sup>3</sup> Yet Another Workflow Language

<sup>4</sup> Advanced Process-Oriented Self-Directed Learning Environment <http://www.aposdle.org>

4. lack of skilled knowledge engineers inside the organisation to take care of the whole modelling process, but we have to encourage, train and support knowledge engineers and domain experts from the organisation to be the main actors of the modelling phase.

We realise that while certain aspects and problems of the modelling process are specific to APOSDLE, the big problems beyond that have to do with the general problem of supporting the integrated modelling of an application domain, processes and learning goals, and to do it in a collaborative manner, without a single knowledge engineer in an organisation feeding the entire process, but on a modelling team composed of different knowledge engineers and domain experts.

Despite the number of methodologies available we have found that none was adequate to support the collaborative and integrated modelling of knowledge, processes and learning goals dependably. Carefully reinvestigating related methodologies, we found the following gaps and shifts, which were influencing the development of the APOSDLE methodology. [Uschold and King, 1995] present a methodology which is purely manual. In contrast, APOSDLE needs a modelling methodology supported by automatic or semi-automatic tools. [Gruninger and Fox, 1994] uses competency questions to define the scope of the application domain. We have followed a different approach and based the first step on our methodology on questionnaires, workshops and usage scenarios to get an exact idea of the application partners domains. Many related methodologies except METHONTOLOGY [Fernández-López et al., 1997], are focused on the creation of formal models and do not support an explicit phase where informal models are created with appropriate tools, which is a major gap we are tackling with our methodology. The description of models is usually created using word processors or paper documents. Within APOSDLE we have suggested Semantic MediaWiki as a collaborative tool for informal modeling. Finally, METHONTOLOGY parallels many activities. We felt that a methodology composed of sequential and well-defined steps was easier to follow and simpler to implement in the context of APOSDLE, where we can rely on possibly not enough skilled knowledge engineers.

Starting from ideas and techniques available from existing methodologies, we have developed a specific methodology to support the creation of the APOSDLE knowledge base. While certain aspects of this methodology are tailored to the specific requirements of APOSDLE, the novelty of our overall approach concerns the development of tools for informal modelling, and their tight integration with tools for formal modelling, which can provide a concrete support to the integrated modelling process of knowledge, processes and learning goals in a general domain. Our approach is based on two main pillars: first, the usage of Semantic MediaWiki (SMW) [Krotzsch et al., 2005] for the collaborative development of informal, but structured, natural language descriptions of the domains to be modelled, and second, a tight, and as much as possible automatic, integration of the SMW with tools for ontology and workflow modelling. This in order to re-use the informal descriptions for automatic ontology and workflow creation.

In this paper we provide an overview of the methodology we have built, abstracting as much as we can from the specific settings of APOSDLE. The paper is structured as follows: first we present a brief overview of the entire methodology (Section 2), and then we concentrate on the part concerning the informal and formal modelling of a domain ontology (Section 3). We end with some concluding remarks (Section 4).

## 2 The Entire Methodology

In this section we present a brief overview of the methodology that we propose to support the integrated modelling process of ontologies and workflows. This methodology consists of five distinct phases which cover the entire process of models creation:

**Phase 0. Scope & Boundaries.** At the beginning, questionnaires and workshops are used to identify an appropriate domain to be modelled. Once the appropriate application domain has been chosen, its scope is determined with the help of use case scenarios. The results of this phase are documented in a MediaWiki<sup>5</sup>.

**Phase 1. Knowledge Acquisition.** In this phase, knowledge about the domain to be modelled is acquired both (i) from experts of the domain, and (ii) from available digital resources relevant for the domain. Knowledge elicitation from experts is achieved using well-known and established techniques like interviews, card sorting and laddering, while knowledge extraction from digital resources is based on state of the art algorithms and tools for term extraction (e.g. see [Pammer et al., 2007]).

**Phase 2. Informal Modelling.** Starting from the knowledge acquired in Phase 1, an informal, but structured and rather complete, description of the different models which will constitute the knowledge base is created. The descriptions of the informal models are obtained by filling some predefined templates in a SMW. The use of the SMW allows to describe the elements of the different models in an informal manner using Natural Language. However, at the same time it allows to structure the descriptions in a way that they can be easily (and often automatically) translated in formal models, without forcing the members of the modelling team to become necessarily experts in the formal languages used to produce the formal models.

**Phase 3. Formal Modelling** In this phase the informal descriptions of the models are transformed in formal models in a way which is as much automatised as possible. The result of this phase is an OWL ontology describing the application domain, an OWL ontology describing the learning goals, and YAWL workflows modelling tasks.

**Phase 4. Validation & Revision.** Finally, the knowledge base created so far is evaluated and possibly revised. The methodology provides support to automatically check, via SPARQL queries, different properties both of the whole knowledge base, and of its single components. The results of these checks are evaluated and used to revise the knowledge base, if needed.

<sup>5</sup> [www.mediawiki.org](http://www.mediawiki.org)

We describe Phase 2 and Phase 3 of the integrated modelling methodology (for a more detailed description of the whole methodology see [Ghidini et al., 2007]). We also report our experience and some lesson learnt applying the methodology in APOSDLE.

### 3 Modelling a Domain Ontology

*Informal Modelling.* Starting from the (flat or already partially structured) list of elements obtained and documented during the knowledge acquisition phase in a MediaWiki, the modelling team filters them, retaining only the relevant ones. The elements considered in APOSDLE are only concepts, but the approach could be extended to consider relations and individuals as well. In order to help the modelling team deciding whether a concept is relevant or not w.r.t. a particular domain, guidelines are provided. Examples questions used during the deployment of the methodology in APOSDLE are:

1. Is this domain concept useful for retrieval?
2. Are there resources dealing with this domain concept, or is it reasonable to expect resources dealing with this domain concept in the future?
3. Does this domain concept help to differentiate between resources?
4. Does this domain concept refer to a learning goal of a hypothetical APOSDLE user?
5. Does this concept help APOSDLE to support the acquisition of a learning goal?

Once the skeleton list of elements (concepts) is ready, we provide it as input to our informal modelling tool: the SMW. The idea is to use a pre-defined template to automatically create a page for each one of the concepts of the domain ontology and let the modelling team to fill the templates, thus providing the information needed to create the ontology. The reason to use a SMW is that it allows the modelling team (composed of domain experts and knowledge engineers) to provide the descriptions about the elements of the domain model in Natural Language. Differently from using a word processor, the Natural Language descriptions inserted in a SMW can be structured according to the pre-defined templates, and with the help of semantic constructs like attributes and relations. Therefore, the informal descriptions in Natural Language contain enough structure for (semi-)automatic translation in OWL ontologies, thus allowing to reuse the informal descriptions for automatic ontology creation.

*Template for the domain concepts.* Figure 2 shows a screenshot of the template we have used for describing concepts in APOSDLE. The template is tailored to the information we needed to obtain for APOSDLE. The bottom-level part of the template concerns the name and description of the concept, and take into account the aspects of multi-linguality we had to face in APOSDLE. These elements ("English Description", "English Name", "Name", "Short Description", "Synonyms") are modelled as String attributes in the SMW. The top-level part of the template concerns the relation of the concept with other concepts in the (still informal) ontology. In order to help the modelling team we predefine in the template some typical relations such as "Is a" and "Is part of". We also provide a general "Is related to" relation and ask the modelling team

Domain concept template	
Is a	Other concept
Is part of	Other concept
Is related to	Other concept
English Description	describe/define the concept in English
English Name	insert the English name of the concept
Name	insert the name of the concept in your preferred language
Short Description	describe/define the concept in your preferred language
Synonyms	insert the synonyms of the concept

**Figure 2:** The domain concept template in the SMW

to instantiate (rename) it to specific, domain dependent, relations, when possible. All these relations are modelled as Relations also in the SMW.

The information required in the template is very basic, but allows to guide the modelling team to provide all the information that was needed by the particular application. In addition mentioning explicitly the "Is a" and "Is part of" relations prevented the modelling team to incur in one of the typical modelling mistakes of non expert designers when they use the graphical environment of ontology editors to create is-a taxonomies, that is, to actually use a mixture of is-a, is-part-of, and other specialisation relations in the creation of the taxonomy, thus making the taxonomy more similar to a directory structure than to an is-a taxonomy. The usage of the methodology in different domains could require the extension of these templates or the creation of new ones. Creating templates in the SMW is not a difficult operation. The challenging aspect here is to be able to design appropriate templates which can guide the modelling team in the process of providing appropriate descriptions of the different elements.

Another advantage of using a SMW are checks (also in automatic) to evaluate the quality of the informal descriptions produced. In addition, the domain concepts' relevance and unambiguity can be checked using the SMW. The verbal descriptions of concepts have proved to be very useful to help with this task.

*Formal Modelling.* The content of the SMW, created in the informal modelling phase, is then automatically translated into OWL to produce the formal ontology. The idea behind the translation is to transform each concept description (concept template) of the informal domain model into the corresponding OWL class. The starting point for creating automatically the OWL ontology from the informal domain model is the built-in SMW `Export pages to RDF` functionality. Using this functionality, it is possible to generate a document in OWL/RDF format containing information on the relations and attributes used in the pages selected to be exported. However, the model of knowledge used by SMW is quite different from the one we developed for the modelling methodology. In a nutshell pages of SMW are by default seen as instances, and not as classes. Therefore the straightforward application of the `Export pages to RDF` functionality produces, for each concept template, an instance of a top class `smw:Thing`, in-

stead of an OWL class. Similarly, the “is a” relation is mapped by the `Export pages to RDF` functionality to an object property named `is a`, while in our approach this relation needs to be mapped to the RDFS `subClassOf` relation. For this reason we developed a Java tool which post-processes the files obtained using the `Export pages to RDF` functionality, and generates an OWL ontology.

Once the OWL ontology is created, the modelling team corrects the hierarchy and the relations in the informal domain model. Missing or redundant concept correlations are updated in the formal model. Note that the informal model is not kept up-to-date with changes made in the formal model at this stage because reflecting the changes made into an OWL ontology back to SMW is not a trivial task. We still have to address it, partly because the interaction with the formal model should be narrowed down to checks of the hierarchy, of the granularity of concepts and relations, and of formal consistency checks and detection of weak concepts.

A similar approach is proposed by the methodology for modelling processes and learning goal. The template we proposed in APOSDLE for informal task description, asks for possible mappings between a task and required domain elements of the domain model. These mappings are then used in the formal modelling phase to create the learning goals model with an ad-hoc tool developed for APOSDLE, the TACT<sup>6</sup> tool.

*Lessons Learnt.* After using the methodology for three APOSDLE application partners, we have collected qualitative feedback from different members of the modelling teams. On the positive side, the SMW allows for sharing information among all the members of the modelling team. Furthermore, templates helped guiding the people in charge of modelling to provide complete descriptions of the elements of the models. Finally, by using the SMW as informal modelling tool, the transfer from the informal domain model to a formal domain ontology was almost effort free, and the interaction with the formal language was kept very low. Last, but not least, the entire modelling process is well documented in the SMW. On the negative side, the SMW does not provide any built-in textual or graphical overview of the overall structure that can be easily used to show the taxonomy of domain concepts. Although some visualisation tools for MediaWiki are available, they can not be easily adapted to work with our approach. In addition the SMW does not support a user-friendly revision of filled templates. This makes the refinement of the informal models, and the use of an iterative process towards modeling, quite laborious. The integration of tasks with formal modelling tools and the automatic translation into YAWL workflows was more challenging than the one for the domain model. While the SMW makes easy to represent and export declarative knowledge about concepts (by RDF triples) processes have also a procedural aspect which is difficult to reconstruct starting from the (declarative) description of the single task components. Summarized, the SMW’s coherent interface to describe the elements of the different data models proved to be quite useful for informal modelling. SMW

---

<sup>6</sup> Task And Competency Tool

also helped in producing an effective integrated development of the models. Solving the issues of visualisation and revision of the models or support for an automatic translation of the task model from the SMW to YAWL would make the tool even more user-friendly.

#### 4 Conclusions

In this paper we introduced a methodology to support the integrated modelling process of ontologies and workflows within APOSDLE. Using this methodology, three APOSDLE Application Partners and two academic partners built their own Knowledge Bases with between 43 and 144 concepts, between 99 and 304 learning goals as well as between 40 and 42 tasks in different domains (e.g. innovation management, environmental consulting and requirements analysis). Our novel approach of developing tools for informal modelling, and their tight integration with tools for formal modelling, can provide a concrete support to the modelling process of ontologies and workflows in other domains, e.g. some preliminary encouraging results have been obtained in a biomedical domain.

#### Acknowledgements

This work has been partially funded under grant 027023 in the IST work programme of the European Community.

#### References

- [Fernández-López et al., 1997] Fernández-López, M., Gómez-Pérez, A., and Juristo, N. (1997). Methontology: from ontological art towards ontological engineering. In *Proc. Symp. on Ontological Eng. of AAAI*.
- [Ghidini et al., 2007] Ghidini, C., Rospocher, M., Serafini, L., Kump, B., Pammer, V., Faatz, A., and Guss, J. (2007). Integrated modelling methodology - first version. EU APOSDLE project Deliverable D1.3.
- [Gruninger and Fox, 1994] Gruninger, M. and Fox, M. (1994). The role of competency questions in enterprise engineering.
- [Krotzsch et al., 2005] Krotzsch, M., Vrandečić, D., and Volkel, M. (2005). Wikipedia and the semantic web - the missing links. In *Proc. of Wikimania 2005 - The First International Wikimedia Conference*.
- [Pammer et al., 2007] Pammer, V., Scheir, P., and Lindstaedt, S. (2007). Two protégé plug-ins for supporting document-based ontology engineering and ontological annotation at document-level. In *10th International Protégé Conference*.
- [Uschold and King, 1995] Uschold, M. and King, M. (1995). Towards a methodology for building ontologies. In *Proc. of IJCAI95's WS on Basic Ontological Issues in Knowledge Sharing*.



# **Building Ontology Networks: How to Obtain a Particular Ontology Network Life Cycle?**

**Mari Carmen Suárez-Figueroa**

(Ontology Engineering Group, Departamento de Inteligencia Artificial  
Facultad de Informática, Universidad Politécnica de Madrid, Madrid, Spain  
mcsuarez@fi.upm.es)

**Asunción Gómez-Pérez**

(Ontology Engineering Group, Departamento de Inteligencia Artificial  
Facultad de Informática, Universidad Politécnica de Madrid, Madrid, Spain  
asun@fi.upm.es)

**Abstract:** To build an ontology, ontology developers should devise first a concrete plan for the ontology development, that is, they should establish the ontology life cycle. To do this, ontology developers should answer two key questions: a) which ontology life cycle model is the most appropriate for their ontology project? and b) which particular activities should be carried out in their ontology life cycle? In this paper we present a set of guidelines to help ontology developers and also naïve users answer such questions.

**Keywords:** Ontology engineering, Ontology development, ontology life cycle

**Categories:** I.2.4, M.2

## **1 Introduction**

The methodological support for developing ontologies and ontology networks should include the *identification and definition of the development process, life cycle models and the life cycle*.

There are many different approaches for building ontologies. Thus, an analysis of methodologies was included in [Fernández-López, 02]; a series of existing methods and methodologies for developing ontologies from scratch have been reported in [Gómez-Pérez, 03]; a set of ontology learning methods for building ontologies was included in [Gómez-Pérez, 05]; and the experience of using wikis for gaining consensus on ontology modelling during the ontology development was reported in [Hepp, 07], among other approaches.

However, existing methodologies for building ontologies have some limitations with respect to the aforementioned issues. We analyzed such issues in three well known existing methodologies: METHONTOLOGY [Gómez-Pérez, 03], On-To-Knowledge [Staab, 01] and DILIGENT [Pinto, 04]).

With regard to the *identification and definition of the development process*, from the aforementioned methodologies, only METHONTOLOGY proposes explicitly a development process that identifies a set of activities performed during ontology development.

As for *life cycle models*, the three methodologies propose a unique life cycle model: METHONTOLOGY proposes an ontology life cycle model based on evolving prototypes; On-To-Knowledge proposes an incremental and cyclic ontology life cycle model based on evolving prototypes; and DILIGENT proposes an ontology life cycle model also based on evolving prototypes. However, it is well known in Software Engineering that there is no a unique life cycle model valid for all the developments.

Additionally, the literature lacks guidelines that help ontology developers to create a particular *ontology life cycle* based on a model.

To devise the concrete plan for the ontology development, two important questions have to be answered: 1) how do ontology developers decide which life cycle model is the most appropriate for their ontology? and 2) which particular activities should be carried out in their ontology life cycle? To respond to such questions, a collection of ontology life cycle models and some guidelines are presented in this paper. Such guidelines used an activity glossary (the so-called NeOn Glossary of Activities [Suárez-Figueroa, 08]) and the collection of models.

The rest of the paper is organized as follows: section 2 presents a collection of theoretical ontology life cycle models, section 3 explains the guidelines to obtain a particular ontology life cycle, and finally, section 4 includes some conclusions.

## 2 Ontology Network Life Cycle Models

An *ontology network life cycle model* is defined as the framework, selected by each organization, on which to map the activities identified and defined in the NeOn Glossary in order to produce the ontology network life cycle [Suárez-Figueroa, 07].

Within the Software Engineering field, it is acknowledged that there is not a unique life cycle model valid for all the software development projects and that each life cycle model is appropriate for a concrete project, depending on several features. For example, sometimes it is better a simple model (like waterfall [Royce, 70]), whereas other times it is most suitable a more complex one (like spiral [Boehm, 88]).

The same occurs in the Ontology Engineering field, where neither there is a unique model valid for all the ontology development projects, since each life cycle model is appropriate for a concrete development, depending on several features. Therefore, to propose a unique life cycle model for all the ontology network developments is not very realistic. Thus, taking into account the specific features of the ontology network development, a collection of theoretical ontology network life cycle models based on the models commonly used in Software Engineering has been created and proposed in [Suárez-Figueroa, 07]. These ontology network life cycle models vary from trivial and simple models to difficult and complex ones.

The proposed collection of models includes the following ones:

- *Waterfall life cycle model*. Its main characteristic is that it represents the stages of an ontology network as sequential phases. Thus, a concrete stage must be completed before the following stage begins.

Because of the importance of knowledge resources reuse and reengineering and ontology merging, five significantly different versions of the waterfall ontology network life cycle model have been defined and proposed: (1) *five-phase waterfall*, (2) *six-phase waterfall* that extends the previous one with a new phase in which the reuse of already implemented ontological resources is considered,

- (3) *six-phase + merging phase waterfall*, (4) *seven-phase waterfall* in which the six-phase model is taken as general basis and a new phase, the reengineering one, is included after the reuse phase, and (5) *seven-phase + merging phase*.
- *Incremental life cycle model*. Its main feature is that it divides the requirements in different parts and then develops each part in a different cycle. The idea is to incrementally “produce and deliver” the network of ontologies (full developed and functional), that is, the ontology network grows in layers (in a concentric way). Figure 1.a shows how an ontology network grows using this model (the striped parts in the figure mean the developed parts).
  - *Iterative life cycle model*. Its main characteristic is that it divides all the requirements into small parts and develops the ontology network including requirements from all the parts. Figure 1.b shows how the ontology network is developed following this model (the striped parts mean the developed parts).

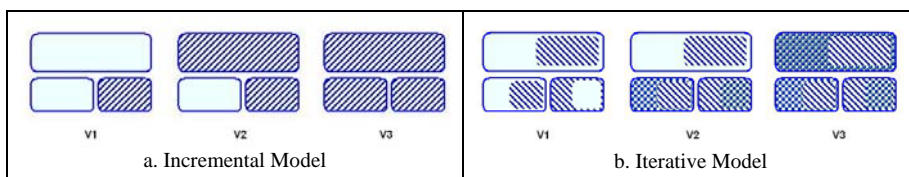


Figure 1: Schematic vision of an ontology network following (a) an incremental model and (b) an iterative model

- *Evolving prototyping life cycle model*. Its main feature is that it develops a partial product (in this case, partial ontology network) that meets the requirements best understood. The preliminary versions of the ontology network being developed (that is, the prototypes) permit the user to give feedback of unknown or unclear requirements.
- *Spiral life cycle model*. Its main feature is that it proposes a set of repetitive cycles based on waterfall and prototype models. In this model, taking into account the special characteristics of ontology networks, the space is divided into three sections: planning, risk analysis, and engineering. This division is based on the need to evaluate and assess all the outputs of all the ontology network stages, and not only after the engineering phase as it happens in software projects.

Relying on our own experience, we can briefly say that the waterfall ontology network life cycle model is the easiest model to understand, and that with this model it is also easy to schedule an ontology development. As for the incremental ontology network life cycle model, it permits to develop the ontology network having complete layers, following any type of waterfall model. Finally, the most sophisticated model is the spiral model that permits analyzing the different risks during the ontology network development.

### 3 Obtaining a Particular Ontology Network Life Cycle

The **ontology network life cycle** is defined as the project-specific sequence of activities created by mapping the activities identified in the NeOn Glossary of Activities onto a selected ontology network life cycle model [Suárez-Figueroa, 07]. The main objective of the *ontology network life cycle* is to determine when the activities identified should be carried out and through which stages the ontology network moves during its life.

Two key questions arise here: 1) how do ontology developers decide which ontology network life cycle model is the most appropriate for their ontology network? and 2) which particular activities should be carried out in their ontology network life cycle?

To help ontology developers to answer the above questions, we recommend the five steps presented in Figure 2. If they follow these steps, ontology developers will be able to answer both questions and to obtain the particular life cycle for their ontology network by mapping the selected ontology network life cycle model and the selected activities, and then ordering such activities.

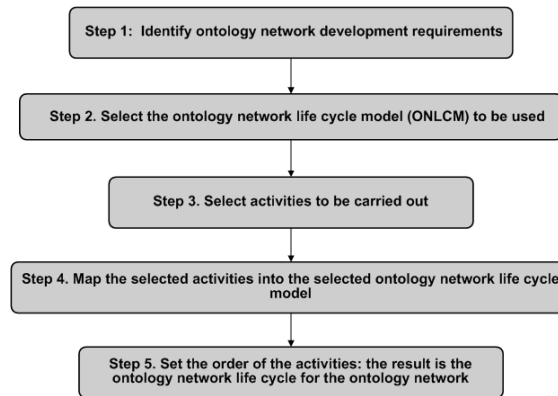


Figure 2: Steps for establishing the ontology network life cycle

**Step 1: Identify ontology network development requirements.** In this step, ontology developers identify the main needs of the ontology network development.

**Step 2: Select the ontology network life cycle model (ONLCM) to be used.** The main question here is: “which ontology network life cycle model should be chosen?”. To carry out step 2, we propose the informal decision tree shown in Figure 3, which helps to select which ontology life cycle model is the most appropriate for the ontology network being built.

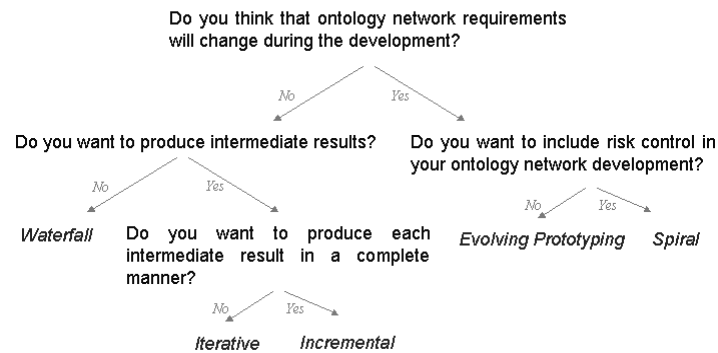


Figure 3: Decision tree for selecting the ontology network life cycle model

**Step 3: Select activities to be carried out.** Activities potentially involved in the ontology network development process are defined in the NeOn Glossary of Activities<sup>1</sup> [Suárez-Figueroa, 08]. In order to facilitate ontology developers the selection of activities from the NeOn Glossary for a concrete development, we have distinguished between required and if applicable activities.

- *Required or Mandatory activities* refer to those activities that should be carried out when developing networks of ontologies. The activities identified as “required” can be considered as core for the ontology development.
- *If Applicable or Optional activities* refer to those activities that can be carried out or not, depending on the case, when developing ontology networks.

To group the activities of the NeOn Glossary into one of the two previous categories, we made an open call and invited ontology developers participating in international projects (NeOn, KWeb, X-Media, etc.) and working in universities and companies (DERI group, OEG group, iSOCO, etc.) to participate in an on-line survey<sup>2</sup>. This survey began on July 27<sup>th</sup> 2007 and the results were collected on August 21<sup>st</sup> 2007. It was answered by thirty five people.

The table of ‘Required-If Applicable’ activities, which is shown in Table 1, has been built considering the results of this survey and our own experience on developing ontologies. The table includes all the activities identified and defined in the NeOn Glossary.

Required activities plus all others applicable to the ontology network development should be selected to be carried out during the ontology network life cycle. The result of step 3 is the table of selected activities. In this step, we propose to distinguish between two distinct kinds of ontology developers:

- *Experienced Ontology Developers.* We assume that, drawing on their own experience, ontology developers are able to select the activities to be carried out during the ontology network life cycle from the “Required-If Applicable” table. Activities identified as “required” in the “Required-If Applicable” table are

<sup>1</sup> <http://www.neon-project.org/web-content/images/Publications/neonglossaryofactivities.pdf>.

<sup>2</sup> <http://droz.dia.fi.upm.es/survey/index.jsp>

selected automatically. Ontology developers should only select those “if applicable” activities they need for their ontology network development.

- *Naïve Ontology Developers.* For those “if applicable” activities, we propose a list of “yes/no” natural language questions (some examples are shown in Table 2) to be answered by naïve ontology developers. If the response of a concrete question is positive, then the corresponding activity is selected; otherwise, the activity is not selected. As in the previous case, activities identified as “required” in the “Required-If Applicable” table are selected automatically.

<i>Required</i>		<i>If Applicable</i>	
O. Annotation	O. Configuration Management	O. Aligning	O. Forward Engineering
O. Assessment	Control	O. Customization	Ontology Learning
O. Comparison	O. Diagnosis	O. Enrichment	O. Localization
O. Conceptualization	O. Documentation	O. Extension	O. Matching
O. Elicitation	O. Feasibility Study	O. Merging	O. Partitioning
O. Environment Study	O. Formalization	O. Modification	O. Population
O. Evaluation	O. Implementation	O. Modularization	O. Pruning
O. Evolution	O. Integration	O. Module Extraction	Non Ontological Resource Reengineering
Knowledge Acquisition for Ontologies	Scheduling	O. Reengineering	O. Specialization
O. Quality Assurance	O. Search	O. Restructuring	O. Summarization
O. Repair	O. Selection	Non Ontological Resource Reuse	O. Translation
O. Reuse	O. Specification	O. Reverse Engineering	O. Update
O. Upgrade	O. Verification		
O. Validation	O. Versioning		

Table 1: Required-If Applicable Activities

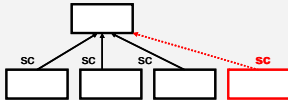
Activity	Natural Language Questions
<b>Ontology Customization</b>	Do you wish to modify the ontology network to meet specific user’s needs?
<b>Ontology Extension</b>	Do you wish to stretch, widen, broaden or expand your current ontology network by adding new concepts “in a horizontal way/direction” to widen its sphere of action? 
<b>Ontology Localization</b>	Do you wish to have your ontology network in different natural languages, as for example, in English, Spanish and/or French?
<b>Ontology Reengineering</b>	Do you wish to take an existing and implemented ontology to enhance it and implement it again?
<b>Non Ontological Resource Reuse</b>	Do you intend to use non ontological resources (such as a controlled vocabularies or data bases) in the development of your ontology?

Table 2: Examples of Proposed “Yes/No” Natural Language Questions

**Step 4: Map the selected activities into the selected ontology network life cycle model.** To carry out this mapping, ontology developers should match the selected activity outputs against the requirements of each phase or stage in the selected ONLCM. This step provides an activity map or matrix for the ontology network development.

**Step 5: Set the order of the activities: the result is the ontology network life cycle for the ontology network.** After obtaining the activity map or matrix, ontology developers should order the activities of this matrix, thus obtaining the ontology network life cycle. The order in which the activities will be performed are determined by three major factors:

- The selected ONLCM dictate an initial ordering of activities.
- Schedule constraints may require the overlapping of activities in the ONLCM and may thus impact the ordering.
- Selection and ordering of activities might be impacted by the entry and exit criteria of associated activities. The availability of output information from one activity could affect the start of another activity.

The guidelines proposed in this paper are being used and thus evaluated in the development of the ontologies in two use cases within the NeOn project [Suárez-Figueroa, 07]: invoice management and semantic nomenclature, both belonging to the pharmaceutical domain.

## 4 Conclusions

The main contribution of our paper is the set of guidelines we have created to help ontology developers obtain the concrete life cycle of an ontology network.

Our guidelines for obtaining the concrete life cycle for an ontology network are mainly created to help ontology developers to make these two decisions: (1) selecting the ontology network life cycle model that is the most appropriate for a concrete case and (2) selecting which activities, from the NeOn Glossary of Activities, should be carried out.

Thus, for the first decision, we propose some guidelines involving the collection of ontology network life cycle models presented in this paper. Such models are based on the models defined in the Software Engineering field and take into account the specific features of the ontology network development.

For the second decision, we suggest some guidelines that use the NeOn Glossary of Activities, which identifies and defines the activities potentially involved in the ontology network development. The activities in the NeOn Glossary have been divided into activities required for ontology network development and those that could or could not be applicable, depending on the concrete case, and consequently non-essential or dispensable. The proposed guidelines are founded on natural language questions for helping naïve users to select the activities they have to perform.

### Acknowledgements

This work has been supported by the NeOn project (IST-2005-027595). We are very grateful to Elena Montiel-Ponsoda for her help with the natural language questions.

## References

- [Boehm, 88] Boehm, B. W.: *A spiral model of software development and enhancement*. ACM SIGSOFT Software Engineering Notes. Vol. 11, no. 4, pp. 14-24, August 1986; reprinted in Computer, vol. 21. no. 5, pp. 61-72, May 1988.
- [Fernández-López] Fernández-López, M., Gómez-Pérez, A.: *Overview and Analysis of Methodologies for Building Ontologies*. Knowledge Engineering Review (KER). Vol. 17, Nº 2, pp 129-156. 2002.
- [Gómez-Pérez, 05] Gómez-Pérez, A., Manzano-Macho, D.: *An overview of methods and tools for ontology learning from texts*. Knowledge Engineering Review, 19:187–212, 2005.
- [Gómez-Pérez, 03] Gómez-Pérez, A., Fernández-López, M., Corcho, O.: *Ontological Engineering*. November 2003. Springer Verlag. Advanced Information and Knowledge Processing series. ISBN 1-85233-551-3.
- [Haase, 06] Haase, P., Rudolph, S., Wang, Y., Brockmans, S., Palma, R., Euzenat, J., d'Aquin, M.: *NeOn Deliverable D1.1.1 Networked Ontology Model*. November 2006. Available at: <http://www.neon-project.org/>.
- [Hepp, 07] Hepp, M., Siorpaes, K., Bachlechner, D.: *Harvesting Wiki Consensus: Using Wikipedia Entries as Vocabulary for Knowledge Management*. IEEE Internet Computing 11(5): 54-65. 2007.
- [Pinto, 04] Pinto, H. S., Tempich, C., Staab, S.: *DILIGENT: Towards a fine-grained methodology for Distributed, Loosely-controlled and evolving Engineering of ontologies*. In Ramón López de Mantaras and Lorenza Saitta, Proceedings of the 16th European Conference on Artificial Intelligence (ECAI 2004), August 22nd - 27th, pp. 393--397. IOS Press, Valencia, Spain, August 2004. ISBN: 1-58603-452-9. ISSN: 0922-6389.
- [Royce, 70] Royce, W. W.: *Managing the development of large software systems: concepts and techniques*. In Proceedings Western Electronic Show and Convention (WESCON), Los Angeles. August 25<sup>th</sup>-28<sup>th</sup> 1970.
- [Staab, 01] Staab, S., Schnurr, H.P., Studer, R., Sure, Y.: *Knowledge Processes and Ontologies*. IEEE Intelligent Systems 16(1):26–34. (2001).
- [Suárez-Figueroa, 07] Suárez-Figueroa, M. C., Aguado de Cea, G., Buil, C., Caracciolo, C., Dzbor, M., Gómez-Pérez, A., Herrero, G., Lewen, H., Montiel-Ponsoda, E., Presutti, V.: *NeOn Deliverable D5.3.1 NeOn Development Process and Ontology Life Cycle*. August 2007. Available at: <http://www.neon-project.org/>.
- [Suárez-Figueroa, 08] Suárez-Figueroa, M. C., Gómez-Pérez, A.: *Towards a Glossary of Activities in the Ontology Engineering Field*. 6th Language Resources and Evaluation Conference (LREC 2008). Marrakech (Morocco). May 2008.



## Managing Ontology Lifecycles in Corporate Settings

**Markus Luczak-Rösch and Ralf Heese**

(Freie Universität Berlin)

Institut für Informatik, AG Netzbasierte Informationssysteme

Takustr. 9, D-14195 Berlin, Germany

{luczak,heese}@inf.fu-berlin.de)

**Abstract:** Weaving the Semantic Web the research community is working on publishing publicly available data sources as RDF data on the Web. Well-known cost- and process-oriented problems of ontology engineering hinder the employment of ontologies as a flexible, scalable, and cost-effective means for integrating data in corporate contexts. We propose an innovative ontology lifecycle, examine existing tools towards the functional requirements of the lifecycle phases and propose a versioning approach supporting them integratively.

**Key Words:** knowledge management, knowledge engineering methodologies, knowledge life cycles, knowledge maintenance

**Category:** M.1, M.2, M.3

### 1 Introduction and Related Work

Within the past years the Semantic Web community has developed a comprehensive set of standards and data formats to annotate semantically all kinds of resources. Currently, a main focus lies on integrating publicly available data sources and publishing them as RDF on the Web. In contrast, many corporate IT areas are just starting to engage in Semantic Web technologies. Early adopters are in the areas of enterprise information integration, content management, life sciences and government. Applying Semantic Web technologies to corporate content is known as *Corporate Semantic Web*. To employ ontologies as a flexible, scalable, and cost-effective means for integrating data in corporate contexts, *corporate ontology engineering* has to tackle cost- and process-oriented problems [Simplerl 06].

In Section 2 we present a set of requirements characterizing corporate settings for ontology-based information systems. We use these requirements in Section 3 to conclude the need of a new lifecycle model, which we introduce afterwards. The lifecycle raises new functional requirements, which we use for a comparison of accepted tools for ontology engineering (Section 5) and a conclusion about their applicability.

Research reached a wide range of ontology engineering methodologies which mostly differ in details referring to the composition of ontology engineering and application development, the range of users interacting in ontology engineering

tasks, and the degree of lifecycle support. Some methodologies assume the users to be ontology experts only or at least to be knowledge workers with little technical experience while others also address users with no experience with ontologies at all.

*METHONTOLOGY* [Fernandez 97] transfers standards for software engineering to the task of ontology engineering and is a concept-oriented approach to build ontologies from scratch, reuse existing ontologies or re-engineer knowledge. The lifecycle model of *METHONTOLOGY* does not respect any usage-oriented aspects. The On-To-Knowledge methodology (*OTK*) [Sure 02] is less concept-oriented because it has an application-dependent focus on ontology engineering. It integrates participants which are not very familiar with ontologies in early phases of the process for identification of the use cases and competency questions. *OTK* assumes a centralized and a distributed strategy for ontology maintenance but neither presents a detailed description or evaluation of both strategies nor addresses ontology usage. The methodologies *HCOME* [Kotis 06] and *DILIGENT* [Pinto 06] address the problem of ontology engineering from the point of view that reaching an ontology consensus is highly depending on people with disparate skill level. Both methodologies assume a distributed setting. Every individual is free in adapting the central ontology consensus locally. The evolution of the consensual model is depending on these local adoptions. Thus, *HCOME* and *DILIGENT* propose a human-centered approach, but they do not provide any application-dependent point of view. Recently, the well-thought approach of agile engineering has come into focus of research in ontology engineering. In [Auer 06] *RapidOWL* is introduced as an idea of agile ontology engineering. Auer proposes a paradigm-based approach without any phase model. *RapidOWL* is designed to enable the contribution of a knowledge base by domain experts even in absence of experienced knowledge engineers. However, the view on ontology usage is limited to the rapid-feedback, which is nonspecific referring to the stakeholder who gives it and how it is given. As a recent result of the NeOn project the NeOn methodology for ontology engineering and the NeOn architecture for lifecycle support have been developed [Tran 07]. Again, both lack the agility of knowledge lifecycles referring to knowledge evolution by usage of ontologies in an enterprise.

## 2 Corporate Ontology Engineering Settings

*Corporate Semantic Web* assumes a corporate environment for the application of Semantic Web technologies. Two major goals aim at their beneficial application in the core areas *Corporate Semantic Search* and *Corporate Semantic Collaboration*. We consider ontologies, which appear highly *application-dependent* in this scenarios, to characterize a corporal competency.

As the result of personal interviews with industrial partners of the project Corporate Semantic Web, we collected key-points about applying ontology-based information systems in corporate contexts. We raise the following main requirements as the results of the interviews: (1) application-dependence of the ontologies, (2) central allowance and control of the conceptualization, (3) individual views to the knowledge domain (e.g. units or hierarchies), (4) lack of time for manual population and/or annotation, (5) unexperienced users working with the ontologies and (6) need for estimation of development costs and potential improvement. We also derived two use-cases from the interviews which proof the correctness of these requirements: (a) a **semantic trouble-ticketing system** where the terminology of internal employees and external customers have to be matched against the fixed terms of the released product and (b) a **corporate semantic intranet** application which integrates data from various conventional sources and should transfer views and restrictions of the sources into the central semantically enriched knowledge base.

Based on this set of requirements of corporate ontology engineering settings, we derive a new point of view on ontology engineering processes. The widely accepted methodologies METHONTOLOGY, OTK, HCOME, and DILIGENT regard ontology engineering loose from ontology usage. However, they agree that ontologies are undergoing lifecycles with engineering phase and usage phase, but they do not consider ontology engineering as a combination of both. From our perspective the evolution of knowledge is the basal entity of an adequate ontology lifecycle and that it is strongly depending on the usage by unexperienced people with lack of time to note, annotate, or feedback explicitly.

### 3 A Corporate Ontology Lifecycle

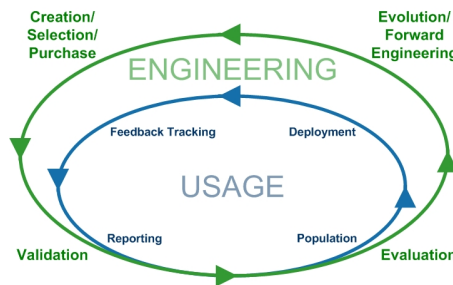
From our assumptions mentioned in Section 2, we conclude a need of a new ontology lifecycle model for ontologies in corporate contexts. The model should allow an intuitive understanding of raising costs per iteration. Because of the change in the environment complexity from Web-scale to corporate-scale, we assume that it is possible to converge ideas of agile software engineering and ontology engineering. But we think it is necessary to change the definition of what is assumed as being agile.

Recent approaches such as RapidOWL focus the agile paradigms *value*, *principle*, and *practice* as a development philosophy. That accompanies agile software engineering as it is intended in the *Agile Manifesto*<sup>1</sup>. But, again, this focus is limited to engineering tasks, while usage is factored out. It comes clear, that changing requirements over time are only one agile aspect influencing ontology prototype evolution. Another is the dynamic of knowledge referring to the

<sup>1</sup> <http://agilemanifesto.org/>

evolving dimensions of data and user activities depending on processes.

The corporate setting needs an ontology engineering model which respects these agile aspects and allows an intuitive way of estimating costs for evolution steps. Both points play a key-role for our approach towards a generic corporate ontology lifecycle which is depicted in Figure 1. Eight phases of our two-part cycle are marked, which refer either to the outer cycle as creation/selection, validation, evaluation, evolution/forward engineering or to the inner circle as population, deployment, feedback tracking, and synchronization. The outer cycle represents pure engineering, which is an expert-oriented environmental process. The inner constitutes the ontology usage, which is a human-centered concurrent process.



**Figure 1:** The Corporate Ontology Lifecycle

Starting the process at *creation/selection* means to start the knowledge acquisition and conceptualization, to re-use or re-engineer existing ontologies, or to commission a contractor to develop an ontology. The result of this phase is an ontology, which is *validated* against the objectives. At the intersection point between the engineering and the usage cycles the ontology engineers and the domain experts decide whether the ontology suites the requirements or not. If this is approved the ontology is *populated*, which means that a process for instance generation from structured, semi-structured and unstructured data runs up. The ontology is *deployed* and in use by applications. Throughout the whole *feedback tracking* phase, formal statements about users feedback and behavior are recorded. A *reporting* of this feedback log is performed at the end of the usage cycle. That means that all feedback information, which were collected until a decisive point, are analyzed respecting internal inconsistencies and their effects to the currently used ontology version. The usage cycle is left and the knowledge engineers *evaluate* the weaknesses of the current ontology with respect to the feedback log. This point may also be reached, when the validation shows that the

new ontology is inappropriate to the specification. The lifecycle closes with the *evolution/forward engineering* of the ontology by engineers and domain experts.

The innovative approach towards agile ontology engineering allows an evolution of rapidly released ontology prototypes. We expect from our model to allow an intuitive view to ontology engineering processes and facilitate a cost estimation in the run-up of cost-intensive evolution steps. We reach these improvements by a convergence of ontology engineering and ontology usage controlled by an innovative versioning approach.

#### 4 Functional Requirements of Corporate Ontology Engineering

Since we developed this corporate ontology lifecycle<sup>2</sup> it is possible to derive a set of functional requirements, which enables a holistic support for it. Thus, we examined each phase for needed tools and list the requirements as follows:

**Creation:** Access to global repositories of standard ontologies or available contractor for initial ontology development.

**Validation:** Discussion support and support for collaborative decision making for experts and non-experts.

**Population:** Tools for automatic knowledge acquisition

**Deployment:** System for supplying the appropriate ontology version to applications.

**Feedback tracking:** System for integration of lightweight extended communication platforms, e.g., forums or feedback forms and automatic recovery of user behavior into a feedback log.

**Synchronization:** System, which exports a snapshot of the log at a dedicated point of time.

**Evaluation:** Validation and reasoning tools which enable an evaluation of the log snapshot referring to the actual working ontology version.

**Evolution:** System which allows the evolution of ontologies (e.g. creation of views, coexisting branches or just new versions).

As a result of this it is necessary to examine existing tools along the new functional requirements raised. We aim at finding the appropriate tool(s), which suite the process integrative.

---

<sup>2</sup> <http://www.corporate-semantic-web.de/colm-lifecycle-methodology.html>

## 5 Comparison of Tools

In this section we give a brief overview of some accepted tools for ontology engineering tasks compared to the support of the different phases of our corporate ontology lifecycle. The desktop-applications Protégé and SWOOP as well as the web-applications OntoWiki and Ikewiki are in focus. These tools are representatives for the currently most accepted approaches to ontology engineering under the requirements of the methodologies we introduced.

**Protégé** is the most accepted tool for ontology building. Its appearance is similar to software development environments. Protégé is rich in function and language support and very scalable in cause of its extensibility. Since Protégé contains collaborative components it is possible to develop consensual ontologies in a distributed fashion using lightweight access to the process by discussion and decision making about proposed changes. This feature does not respect any roles or permissions. Versioning control is enabled on ontology level, but not on conceptual level, enriched by the annotations from the structured argumentations. Any abstraction from technical terms is missing. To sum up, Protégé is a very useful tool for engineering ontologies in a team of experts with a lack of lifecycle support in a usage-oriented architecture.

**SWOOP** is a desktop environment for ontology engineering, which is a bit straightforward at the expense of functionality. The representation of the concepts allows a web-browser-like navigation and is a bit intuitive for non-experts. A search form supports quick searches on the recently used ontology or at least all ontologies stored. Quick reasoning support is implemented in the same fashion. However, there is no abstraction from technical primitives enabled. By definition of remote repositories, it is possible to commit versions of ontologies.

Altogether, SWOOP is a tool for ontology engineering tasks for experts and well-experienced users. It has its strengths in quick and intuitive navigation in and search on ontologies but lacks functional flexibility and lifecycle support.

**OntoWiki** is a php-based wiki-like tool for viewing and editing ontologies. It is setting up on pOWL which makes use of the RAP API<sup>3</sup>. OntoWiki<sup>4</sup> allows administration of multiple ontologies (called knowledge bases) and provides in-line editing as well as view-based editing. As an abstraction from conceptual terms OntoWiki includes an alternative visualization for geodata (Google Maps) and calendars auto-generated from the semantic statements stored. However, a general abstraction from technical primitives (e.g. class,

---

<sup>3</sup> <http://www4.wiwiss.fu-berlin.de/bizer/rdfapi/>

<sup>4</sup> <http://aksw.org/Projects/OntoWiki>

subclass, SPARQL, etc.) in the user front-end is missing. Altogether, it allows only one single view for all users and does not respect any roles or permissions. Changes to the conceptualized knowledge have to be done manually. The ontology history is concept-oriented not ontology-oriented and implemented as known from wiki systems.

We subsume that OntoWiki is an ontology engineering tool and a knowledge base for experienced users with an academic background and that it does not support lifecycle management.

**Ikewiki** implements the semantic wiki-idea and focuses annotation of wiki-pages and multimedia content. It is possible to generate an alternative graph visualization for the context of each annotated page. However, Ikewiki does not support any abstraction from technical primitives for users with less experience in the field of ontologies. Restricted views referring to roles or permissions are not provided. The ontology history is concept-oriented not ontology-oriented and implemented as known from wiki systems.

We summarize about Ikewiki, that this tool addresses familiar wiki users with technical experience which do not need any control of the conceptualization and lifecycle support.

Our experience includes that there exist a strong distance between the recently accepted approaches and the needs of our ontology lifecycle. The tools either have an engineering-oriented perspective, which deals with the ontology application- and user-independent, or they reckoning the conceptualization on an application level for knowledge management without respecting unfamiliar users. The latter is emphasized if we note that the barriers of wiki-syntax for users without any technical background are underestimated. Thus, we subsume the support per phase of our model as follows:

Phase	Protégé	SWOOP	OntoWiki	Ikewiki
Creation/Selection	+	+	+	+
Validation	+	-	-	+
Population	-	-	-	-
Deployment	-	-	-	-
Feedback Tracking	-	-	-	-
Reporting	-	-	-	-
Evaluation	-	-	-	-
Evolution/Forward Engineering	+	+	+	+

Finally, we now conclude that no tool exists, which currently supports our lifecycle model. This is because available tools handle engineering tasks and ontology usage separately. Some tools work with ontologies as the central artifact on an engineering level while others support the application level only. Searching

for an adequate architecture or tool for integrative lifecycle support means to start from the perspective of the evolution by usage of knowledge. A smart versioning control is needed as the central component to enable this.

## 6 Conclusion and Outlook

In this paper we introduced our approach towards an innovative ontology lifecycle for corporate settings. From this model we derived functional requirements for an integrative tool support and compared four ontology development tools with reference to these requirements. We concluded that there is yet a lack of methodological foundations as well as tool support for the agile engineering of ontologies which is strongly needed in corporate contexts. We aim at an extension of this approach towards an innovative architecture for ontology lifecycle management in corporate contexts.

### Acknowledgement

This work has been partially supported by the "InnoProfile-Corporate Semantic Web" project funded by the German Federal Ministry of Education and Research (BMBF).

### References

- [Auer 06] Auer, S., Herre, H.: RapidOWL - An Agile Knowledge Engineering Methodology. In Irina Virbitskaite Andrei Voronkov, editeurs, Ershov Memorial Conference, volume 4378 of Lecture Notes in Computer Science, pages 424-430. Springer, 2006.
- [Abrahamsson 03] Abrahamsson, P., Warsta, J., Siponen, M. T., Ronkainen, J.: New Directions on Agile Methods: A Comparative Analysis. ICSE, pages 244-254. IEEE Computer Society, 2003.
- [Fernandez 97] Fernández-López, M., Gómez-Pérez, A., Juristo, N.: METHONTOLOGY: From Ontological Art Towards Ontological Engineering. AAAI-97 Spring Symposium on Ontological Engineering: Stanford, AAAI Press, 1997.
- [Kotis 06] Kotis, K., Vouros, A.: Human-centered ontology engineering: The HCOME methodology. *Knowl. Inf. Syst.*10(1), pages 109-131, 2006.
- [Lindvall 02] Lindvall, M. et al.: Empirical Findings in Agile Methods. Second XP Universe and First Agile Universe Conference, LNCS 2418, pages 197-207. Chicago, IL, USA, August 2002.
- [Noy 01] Noy, N. F., McGuinness, D. L.: *Ontology Development 101: A Guide to Creating Your First Ontology*. Stanford University, 2001.
- [Pinto 06] Pinto, H. S., Tempich, C., Staab, S., Sure, Y.: *Distributed Engineering of Ontologies (DILIGENT)*. Semantic Web and Peer-to-Peer, Springer, 2005.
- [Simperl 06] Simperl, E., Tempich, C.: *Ontology Engineering: A Reality Check*. OTM Conferences (1), volume 4275 of LNCS, pages 836-854. Springer, 2006.
- [Sure 02] Sure, Y., Studer, R.: *On-To-Knowledge Methodology — Expanded Version*. On-To-Knowledge deliverable, 17. Institute AIFB, University of Karlsruhe, 2002.
- [Tran 07] Tran, T. et al.: *Lifecycle-Support in Architectures for Ontology-Based Information Systems*. ISWC/ASWC, volume 4825 of LNCS, pages 508-522. Springer, 2007.



## Interoperability Issues, Ontology Matching and MOMA

Malgorzata Mochol

(Free University of Berlin, Königin-Luise-Str. 24-26, 14195 Berlin, Germany  
mochol@inf.fu-berlin.de)

**Abstract:** Thought interoperability has been gaining in importance and become an essential issue within the Semantic Web community, the main challenge of interoperability and data integration is still ontology matching. With this in mind, we wish to contribute to the enhancement of (semantic) interoperability by supporting the ontology matching issue; we propose an evaluation framework for matching approaches that contributes to the resolution of the data integration and interoperability issue by creating and maintaining awareness of the link between matchers and various ontologies.

**Key Words:** interoperability, ontology matching, MOMA Framework

**Category:** I.2.4, K.6.3

### 1 Introduction

Over the last few years, interoperability has gained in importance and become an essential issue within the Semantic Web (SW) community. The more standardized and widespread the data manipulation tools are, including a higher degree of syntactic interoperability, the easier and more attractive using the SW approach has become. Though SW technologies can support the unambiguous identification of concepts and formally describe relationships between concepts, Web developers are still faced with the problem of semantic interoperability, which stands in the way of achieving the Web's full potential. The main problem with semantic interoperability is that the cost of its establishing, due to the need for content analysis, is usually higher than what is needed to establish syntactic interoperability [Decker et al. 00]. Semantic interoperability is necessary before multiple applications can truly understand data and treat it as information; it will thus be, according to [Decker et al. 00], a *sine qua non* for the SW. To achieve semantic interoperability, systems must be capable of exchanging data in such a way that the precise meaning of the data is readily accessible, and the data itself can be translated by any system into a form that the system understands [Heflin and Hendler 00]. Hence, a central problem in (semantic) interoperability and data integration issues in the SW vision is schema and ontology matching and mapping [Cruz and Xiao 03].

Considering these problems and the current situation in SW research, we wish to contribute to the enhancement of (semantic) interoperability by providing support to the ontology matching issue. On the one hand, the number of use cases for ontology matching justifies the great importance of this topic in the

SW [Euzenat et al. 04]. On the other hand, the development and existence of tried and tested ontology matching algorithms and support tools will be one of the crucial issues that may have a significant impact on future development, for instance, the vast SW-based information management systems. Furthermore, it has also turned out that different matching algorithms are better suited for matching different sets of ontologies. Today it takes an expert to determine the best algorithm and a decision can usually be made only after experimentation, so as a result the necessary scaling and off-the-shelf use of matching algorithms are not possible. To tackle these problems we have developed an evaluation framework – Metadata-based Ontology Matching (MOMA) Framework – that helps to resolve the data integration and interoperability issue by creating and maintaining awareness of the link between matching algorithms and various ontologies. Our approach allows for a more flexible deployment of matching algorithms (depending on the particular requirements of the application to which the matchers are to be utilized) and the selection of suitable approaches performed prior to the execution of a matching algorithm.

The remain of this paper is organized as follows: In Sec. 2 we specify the main open issues within the ontology matching domain. Then, we outline a possible solution to tackle these problems by introducing the MOMA Framework (Sec. 3); we elaborate the main use cases together with the high-level architecture and sketch the evaluation results. Sec. 4 summaries the work and provides some issues for the future work.

## 2 Ontology Matching Domain

Despite of the pervasiveness of ontology matching and although the development of tools to assist in the matching process has become crucial for the success of a wide variety of information management applications [Doan et al. 04], the matching process is still largely conducted by hand, in a labor-intensive and error-prone process. There is still a number of short, middle, and long-term problems that need to be resolved in order to overcome the interpretability and heterogeneity issues and to realize the vision of a fully developed SW.

***No overarching matching:*** Many methods and tools are under development to solve specific problems in the SW however, none of these solutions can be deployed due to all the existing problems. This statement is also true in the ontology matching field, as there is no overarching matching algorithm for ontologies capable of serving all ontological sources and new approaches tackle only minor aspects of the “larger” problem in the matching domain or are mere “stop gaps” [Fürst and Trichet 05].

***“Unused” reuse:*** The ontology matching field continue to pay little notice to a strategy based on reusing existing matching, merging, and aligning

approaches. Consequently, the reuse of these semantic-based approaches have not yet been analyzed satisfactorily within the SW realm. Our experiences collected during the development of ontology-based applications [Bizer et al. 05, Garbers et al. 06, Niemann et al. 06] confirm previous findings in the literature that building such applications is still a tedious process, as a result of the lack of tested and proved support tools and that reusing of existing methods within new application contexts is currently not extensively discussed in depth. When implementing an application using a matching approach, the corresponding algorithm is typically built from scratch, and only small, marginal attempts to reuse existing methods are made.

**“Evil” diversity:** Since much time and effort have been spent on the development of new ontology alignment and matching algorithms, the collection of such algorithms is still growing. For this reason, we are all confronted with the same problem: there is an enormous amount of divergent work from different communities that claims some sort of relevance to ontology mapping, matching, alignment integration, and merging [Kalfoglou et al. 03]. Given this multiplicity, it is difficult to identify both the problem areas and the solutions. In this view, the diversity of matching approaches is a weakness rather than a strength. Part of the problem is also the lack of a comprehensive survey, a standard terminology, obscure assumptions or undisclosed technical details, and the dearth of evaluation metrics [Kalfoglou et al. 03].

**“Holes” in the approaches** Despite an impressive number of research initiatives in the matching field, current matching approaches still feature significant limitations [Shvaiko 04, Giuchiglia et al. 04, Melnik et al. 02, Madhavan 01]: current matching approaches, though containing valuable ideas and techniques, are tailored to particular types of ontologies and are confined to specific schema types [Do et al. 02]; they need to be customized for a particular application setting (like schema and data integration); they cannot be applied across various domains with the same effect; they do not perform well (or have not yet been tested) on inputs with heterogeneous (graph) structures or on large-sized inputs.

**Lack of infrastructure:** After years of extensive research and development of numerous matching approaches, it is time to deploy some of the procedures, techniques, and tools created [Zhao 07]. Thus, what is required are techniques and tools capable of handling different ontological sources [Castano et al. 04] and the requirements of the emerging applications. Furthermore, users need help in choosing an appropriate matcher or combining the most appropriate matchers for their particular use [Euzenat and Shvaiko 07].

**Additional issues:** Beside the problems mentioned, there are many other aspects of a general nature which need to be resolved. There is the question of what should be matched based upon what needs to be found. It is also important to avoid performing blind matching while knowing when to stop the process. To

this end, it is necessary to adapt the systems, i.e. adjust it, not to the data to be processed, but to the issue that needs to be resolved with the given matcher.

Though we most definitely will not be able to “solve all these problems and save the world” in our research work, we will tackle some of these issues. We have concentrated on the selection of suitable matching approaches, which, in our opinion, is one of the main issues in the ontology matching domain. By proposing a framework that supports the selection of relevant matching algorithms suitable w.r.t the given specification while taking into account the definition of the appropriate criteria for the decision making process, we address the issues of “lack of infrastructure” and “evil diversity” and, in some measure, the problems in terms of “unused” reuse and “no overarching matching”.

### 3 MOMA Framework

Due to the above mentioned issues and the fact that the existing matching algorithms cannot be optimally used in ontology matching tasks, as envisioned by the SW community, we need a strategy to remedy the weaknesses and take advantages of the particularity of the various approaches in the selection of suitable matchers; we need a matcher evaluation process, which performing prior to the execution of a matching algorithm, allows a selection of suitable algorithms. Thus, we have developed a *Metadata-based Ontology Matching (MOMA) Framework* which on the basis of dependencies between algorithms and the ontology types on which the former are able to process successfully, the capabilities of existing matching algorithms and factors that influence the matching tasks recommends the appropriate matchers for a given application.

#### 3.1 Main Use Cases

During discussions with SW-based application developers, researchers, and experts in the ontology matching domain, we noticed there were two types of users interested in the matcher application and the utilization of relevant supportive tools. Consequently, we made a conscious decision to ensure that our MOMA Framework serves both developers/computer scientists by supporting them in their implementation and research work, and the matching providers, enabling them to utilize our matching tool in different service tasks. To this end, we have classified the MOMA users into two main groups: (i) *human matcher users* (e.g. ontology engineers, SW application developers<sup>1</sup>) - the process of choosing the suitable approach can occur both manually and (semi-)automatically; (ii) *machine matcher users* (e.g. service/matching providers) - in this case, the

---

<sup>1</sup> In terms of the ontology development, which is mostly not conducted by people with a high level of expertise in the ontology matching domain, there is a need to aid them in selecting and applying ontology management tools, incl. matching algorithms.

process of choosing suitable matchers is envisioned to be performed only (semi-)automatically. Considering these use cases the objective of MOMA is to supply a tool that offers methods to support the manual and (semi-)automatic detection of suitable matchers (manual and (semi-)automatic mode, respectively).

### 3.2 High-level Architecture

The MOMA Framework (cf. Fig. 1) consists of three main components: (i) *Multilevel Characteristic for Matching approaches (MCMA)* - utilized to describe the matching algorithms, their incoming sources, and feasible output, together with application features in which the matching approach is to be applied; (ii) *Knowledge Base* that includes information (based on the MCMA structure) regarding existing matchers which may be selected for application and sources that are to be matched; it also contains some rule statements that describe the dependencies between the matching approaches and ontologies; (iii) *Selection Engine* that is responsible for the matcher selection which conducts manually or (semi-)automatically the matcher determination process. Therefore, in the following, we analyze MOMA w.r.t the manual and (semi-)automatic selection.

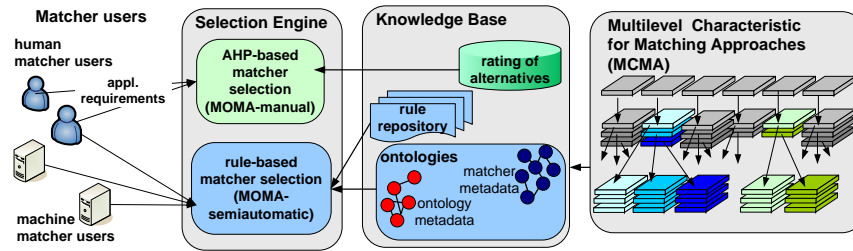


Figure 1: MOMA Framework

**Matcher characteristic:** To find suitable matching approaches for a particular application, it is important to recognize cross application needs and define a matcher characteristic that allows comparison of different approaches and the subsequent selection of suitable algorithms. For this reason, we have collected various features of matching approaches (together with input, output, costs, etc.) and targeted application, identified those that have an impact on the selection of an appropriate matching approach, and finally build a matcher characteristic - *Multilevel Characteristic for Matching Approaches (MCMA)* - that serves as the basis for the final decision regarding the suitability issue <sup>2</sup>.

**Manual approach:** To allow the manual selection of matchers and thereby serve the human matcher users, we have adopted the Analytic Hierarchy Process (AHP) [Saaty 00], which uses pairwise comparisons along with a semantic and ratio scale to assess the decision maker's preferences [Guitouni and Martel 98].

<sup>2</sup> For the detailed description of the MCMA, the reader is referred to [Mochol et al. 06]

AHP allows decision makers to model a complex problem in a hierarchical structure in order to show the relationships of the goal objectives (find suitable matchers), sub-objectives (MCMA), and alternatives (different matchers). The crucial step in the manual selection is the comparison and weighting of sub-objectives. This means that the users of the MOMA manually define the requirements of their application concerning their specification of the potential suitable matching approach by weighing the properties defined within the MCMA in the pairwise comparison. By reducing complex decisions to a series of pairwise comparisons and synthesizing the results, decision-makers arrive at the optimal decision based on a clear rationale [Saaty 99]. In our case, the users of the MOMA Framework obtain a list of matchers ordered by their suitability to the given context<sup>3</sup>.

**Semi-automatic approach:** In order to serve the machine users, we need to provide a (semi-)automatic selection process. As a possible solution, we propose a framework based on rules and defined in the form of ontologies metadata: *ontology metadata* - additional information regarding the ontologies (based on MCMA), like size or representation language, and *matcher metadata* - information regarding existing ontology matching algorithms; to determine automatically which algorithms suit the concrete inputs, explicit knowledge is needed concerning the dependencies between these algorithms and the structures on which they operate. We have formalized this knowledge in terms of dependency rule-statements - *rule repository*. The core of the MOMA Framework within an automatic mode is the *selection engine* which is responsible for the decision making process by means of rules grouped into a rule repository; for a given set of ontologies to be matched, the selection engine must decide (concerning the ontology and matcher metadata and by firing the predefined rules) which matching algorithms are applicable w.r.t the given context.

### 3.3 Evaluation

The evaluation process started with the expert-based evaluation of the MCMA, which resulted in refinement of the preliminarily defined characteristic and, in turn, in a revised MCMA, which has been used within both matcher selection approaches. The further evaluation was dedicated to the accuracy of MOMA predictions and was connected with the usage of MOMA framework in real-world situations. We conducted the evaluation on the basis of the test cases from the Ontology Alignment Evaluation Initiative (OAEI)<sup>4</sup> which aims to establish a consensus for the evaluation of alignment approaches by setting up an evaluation campaign and benchmark tests to assess the strengths and weaknesses of the alignment approaches. The application of the AHP-based MOMA Framework to

<sup>3</sup> For more details regarding AHP-based selection, the reader is referred to [Mochol et al. 06, Mochol et al. 07]

<sup>4</sup> <http://oaei.ontologymatching.org>

the OAEI case studies showed that it produces very relevant results which can serve as a direct basis for the reuse of existing matchers in new ontology-based applications (cf. [Mochol et al. 07]). The evaluation of the rule-based MOMA Framework attested to the fact that the (semi-)automatic matcher selection, which in comparison to the manual approach acts on the much less detailed information, delivers very promising results which can serve as a basic module for further examination of algorithms (cf. [Mochol and Jentzsch 08]).

## 4 Conclusions and Future Work

In this paper, we propose the MOMA Framework that takes into account the capabilities of existing matchers and suggests appropriate approaches for individual cases. Our framework contributes to data integration and interoperability by maintaining awareness of the link between matching algorithms and a wide variety of ontologies. It is the first step towards the reuse of existing ontology matching approaches that contributes to the more optimal utilization of ontology matching tasks as envisioned by the SW community, tackles the issues of matchers heterogeneity, exploits the valuable ideas embedded in current matching approaches, and supports developers by giving them recommendations regarding suitable matcher solutions. The future work will be mainly dedicated to the development of the web service-based MOMA access and the (semi-)automatic utilization of the recommended matchers in the particular application.

## Acknowledgements

This work has been partially supported by the "InnoProfile - Corporate Semantic Web" project funded by the German Federal Ministry of Education and Research (BMBF) and the BMBF Innovation Initiative for the New German Länder - Entrepreneurial Regions.

## References

- [Bizer et al. 05] Bizer, C., Heese, R., Mochol, M., Oldakowski, R., Tolksdorf, R., Eckstein, R.: "The Impact of Semantic Web Technologies on Job Recruitment Processes"; In Proc. of the 7th Int. Tagung Wirtschaftsinformatik, (2005), 1367-1383.
- [Castano et al. 04] Castano, S., Ferrara, A., Montanelli, S.: "Methods and Techniques for Ontology-based Semantic Interoperability in Networked Enterprise Contexts"; In Proc. of the 1st CAiSE INTEROP Workshop On Enterprise Modelling and Ontologies for Interoperability, (2004), 261-264.
- [Cohen 98] William W. Cohen: "Integration of Heterogeneous Databases Without Common Domains Using Queries Based on Textual Similarity"; In Proc. 17th International Conference on Management of Data (SIGMOD), (1998), 201-212.
- [Cruz and Xiao 03] Cruz, I. F., Xiao, H.: "Using a Layered Approach for Interoperability on the Semantic Web"; In Proc. of the 4th Int. Conference on Web Information Systems Engineering (WISE 2003), (2003), 221-232.

- [Doan et al. 04] Doan, A.-H., Madhavan, J., Domingos, P., Halevy, P.: "Ontology Matching: A Machine Learning Approach"; Handbook on Ontologies, Springer, (2004), 385-516.
- [Do et al. 02] Do, H. H., Melnik, S., Rahm, E.: "Comparison of Schema Matching Evaluations"; In Proc. of the 2nd International Workshop on Web Databases, vol. 2593 of Lecture notes in Computer Science, (2002), 221-237.
- [Decker et al. 00] Decker, S., Melnik, S., van Harmelen, F., Fensel, D., Klein, M. C. A., Broekstra, J., Erdmann, M., Horrocks, I.: "The Semantic Web: The Roles of XML and RDF"; IEEE Internet Computing, 4, 5, (2000), 63-74.
- [Euzenat et al. 04] Euzenat, J. et al.: "State of the art on current alignment techniques"; Technical Report Deliv. 2.2.3, Knowledge Web EU NoE, (2004).
- [Euzenat and Shvaiko 07] Euzenat, J., Shvaiko, P.: "Ontology Matching"; Springer Verlag, ISBN-3540496114, (2007).
- [Fürst and Trichet 05] Fürst, F., Trichet, F.: "Axiom-based Ontology Matching"; In Proc. of the 3rd Int. Conf. on Knowledge Capture, ACM Press, (2005), 195-196.
- [Guitouni and Martel 98] Guitouni, A., Martel, J.-M.: "Tentative guidelines to help choosing an appropriate MCDA method"; Europ. Journal of Operational Research, 109, (1998), 501-521.
- [Garbers et al. 06] Garbers, J., Niemann, M., Mochol, M.: "A personalized hotel selection engine" (Poster); In Proc. of the 3rd Europ. Semantic Web Conference, (2006).
- [Giuchiglia et al. 04] Giuchiglia, F., Shvaiko, P.: "Semantic Matching"; Knowledge Engineering Review Journal, 18, 3, (2004), 265-280.
- [Hu et al. 06] Hu, W., Cheng, G., Zheng, D., Zhong, X., Qu, Y.: "The Results of Falcon - AO in the OAEI 2006 Campaign"; In Proc. of the Int. Workshop on Ontology Matching (OM-2006) colloc. with ISWC2006, (2006), 124-133.
- [Heflin and Hendler 00] Heflin, J., Hendler, J.: "Semantic Interoperability on the Web"; In Proc. of Extreme Markup Languages 2000, (2000), 111-120.
- [Kalfoglou et al. 03] Kalfoglou Y., Schorlemmer, M.: "Ontology mapping: the state of the art"; The Knowledge Engineering Review (KER), 18, 1, (2003), 1-31.
- [Madhavan 01] Madhavan, J., Bernstein, P.A., Rahm, E.: "Generic Schema Matching with Cupid"; In Proc. of the 27th VLDB Conference, (2001), 48-58.
- [Melnik et al. 02] Melnik, S., Garcia-Molina, H., Rahm, E.: "Similarity Flooding: A Versatile Graph Matching Algorithm and Its Application to Schema Matching"; In Proc. of the 18th International Conference on Data Engineering (ICDE02), (2002).
- [Mochol and Jentzsch 08] Mochol, M., Jentzsch, A.: "Towards a rule-based matcher selection"; 6th International Conference on Knowledge Engineering and Knowledge Management Knowledge Patterns (EKAW2008), (to appear 2008).
- [Mochol et al. 06] Mochol, M., Jentzsch, A., Euzenat, J.: "Applying an Analytic Method for Matching Approach Selection"; In Proc. of the Int. Workshop on Ontology Matching (OM-2006) colloc. with the ISWC2006, (2006).
- [Mochol et al. 07] Mochol, M., Jentzsch, A., Euzenat, J.: "Towards a methodology for selection suitable matching approaches - A Case Study" (Poster); In Proc. of the 4th European Semantic Web Conference (ESWC2007), (2007).
- [Niemann et al. 06] Niemann, M., Mochol, M., Tolksdorf, R.: "Improving online hotel search - what do we need semantics for?"; In Proc. of the Semantics 2006, (2006).
- [Shvaiko 04] Shvaiko, P.: "Iterative schema-based semantic matching"; Technical Report DIT-04-020, University of Trento, (2004).
- [Saaty 99] Thomas L. Saaty: "How to Make a Decision: The Analytic Hierarchy Process"; European Journal of Operational Research, 48, 1, (1990), 9-26.
- [Saaty 00] Thomas L. Saaty: "How to Make a Decision"; Models, Methods, Concepts & Applications of the Analytic Hierarchy Process, Kluwer Int. Series, (2000), 1-25.
- [Zhao 07] Zhao, H.: "Semantic matching. Across Heterogeneous Data Sources"; Communication of the ACM, 50, 1, (2007), 45-50.



## **Exploiting a Company's Knowledge: The Adaptive Search Agent YASE**

**Alex Kohn, François Bry**

(Institute of Informatics, University of Munich, Germany  
alex.kohn@contractors.roche.com, bry@lmu.de)

**Alexander Manta**

(Roche Diagnostics GmbH, Penzberg, Germany  
alexander.manta@roche.com)

**Abstract:** This paper introduces YASE, a domain-aware search agent with learning capabilities. Initially built for the research community of Roche Penzberg, YASE proved to be superior to standard search engines in the company environment due to the introduction of some simple principles: personalized ranking based on a user's role and organizational embedding, automatic classification of documents by using domain knowledge and learning from search history. While the benefits of the learning feature need more time to be fully realized, the other two principles have proved to be surprisingly powerful.

**Keywords:** search agent / engine, metadata, personalization, information retrieval, YASE

**Categories:** H.3.3, H.4.0, L.2.2

### **1 Introduction**

All of us, regardless of our domain, field or specialty, face the same problem of information overload. In this paper we describe some promising approaches to find the relevant information in steadily growing information flows. The concepts are examined in the context of a research department of Roche Diagnostics GmbH.

In the following sub-sections we describe the hypothesis drawn from two complementary analyses performed last year of the way scientists access information. Section two introduces the search agent<sup>1</sup> YASE which incorporates original ideas of how to improve and personalize the ranking of results. In the last section we conclude the paper and show some perspectives.

#### **1.1 1<sup>st</sup> hypothesis: One single entry point is what scientists prefer**

In order to understand how Roche scientists retrieve information, two complementary in-house studies have been conducted: a survey and a log file analysis. The survey [Mühlbacher, 08] was based on personal questionnaires, targeting approx. 90 scientists from R&D. The second study was a log file analysis based on the monitoring of the usage of a subset of the information sources. During a period of one

---

<sup>1</sup> The term search *agent* is used in this context to distinguish our approach from standard search engines.

month we monitored 5 different search engines targeting approx. 400 employees from research and measured their relative usage (cp. Table 1).

Search Engine	Relative access
Google (Internet)	80,8 %
Wikipedia	8,9 %
PubMed (biomedical abstracts DB)	5,6 %
FAST (intranet search engine)	3,8 %
Google Search Appliance (in-house file search)	0,9 %

*Table 1: Usage of search engines linked from a Pharma Research homepage.*

Both analyses show that a small minority of resources are heavily used by almost all scientists, while the majority of resources are barely accessed. Interestingly, because of the familiarity with the interface and due to its search performance even in specialized data sources like PubMed, patents and Wikipedia, scientists use Google more and more as the main entry point, even for scientific information. With Google there is no need to start an extra search at e.g. Wikipedia or PubMed. This suggests that – similarly to Google for external information - one single entry-point to internal resources would dramatically increase the use of the specialized but valuable data repositories of the company.

### 1.2 2<sup>nd</sup> hypothesis: Standard search engines less used because of poor ranking

A closer look at the usage analyses shown in the Table 1 suggests that valuable sources of in-house information and knowledge (those covered by Fast 1 and Google Search Appliance) are not accessed via search but rather by navigating the folder tree.

Enterprise search engines usually use the vector-space-model for results ranking. Algorithms successful in the Internet like PageRank show bad performance because the linkage structure of the Intranet is either poor [Fagin, 03], [Xue, 03] or completely missing as is the case with most document repositories. Besides, high redundancy (many versions of the same document) and notational heterogeneity (synonyms) distort the search results. Complex queries which go beyond the simple full text search can't be carried out with standard search engines. While cross products or joins are almost impossible to compute on the Internet, this would be possible in an intranet environment as this is comparatively much smaller.

### 1.3 3<sup>rd</sup> hypothesis: Dynamically built navigational structures can compensate for the missing linkage structure

The wealth of meta data available in the company (distribution lists, departmental and project membership lists, domain related thesauri and ontologies, access lists and other meta data extracted from the file system) can be used to assess the relevance of the documents to a certain user, to cluster and classify the documents, to improve the ranking and to create ad-hoc navigational structures. The search history (tuples of search terms and clicked documents), combined with functionality for manual document annotation adds a learning dimension to the repository of metadata, with potential of continuous self-improvement. By adding adequate reasoning features an

adaptive, context-aware search agent can be built, as demonstrated by the prototype YASE.

## 2 The adaptive search agent YASE

Some of the metadata existing in a company and exploited by YASE are given in the table below:

<b>File system</b>	<b>Document attributes and structure</b>	<b>Business context of the user</b>	<b>Domain related knowledge</b>
Size, path, time (creation, last modification, last access), security (read & write permissions, owner)	Author, title, subject, company, manager, width, height, resolution, text content, links, comments, ...	Contact details (name, e-mail, phone, office), department, involved projects and groups	Controlled vocabularies (gene names, protein names, project names), databases, applications, ...

*Table 2: Sources of metadata.*

These metadata have no well-defined semantics in any RDF formalism. Some sources can be even messy, e.g. the “title” attribute (file format metadata) which can contain values like “untitled” or “slide 1”. Sources like the controlled vocabularies on the other hand can be considered clean and curated. Regardless of the source, YASE will treat all accessible data as metadata, whether it is correct or not.

Using metadata annotators of different types (statistical, machine learning or knowledge based) these sources can be used to associate appropriate attributes to documents. As an example consider the annotation of project relevance to a certain document. Project names from a controlled vocabulary are matched against the lexical substrings of the path and against the document vocabulary using a fuzzy string matching approach. In just the same manner we assign departmental relevance to documents. By using the annotator we basically put a file in several categories. At query time, facets according to the annotated categories are automatically displayed, by which a user can further browse through the data. In this way the navigational freedom to browse by project categories which otherwise are spread over several folders is enabled.

An even more powerful join is the association of document metadata with administrative user data, i.e. the user’s working context. We know the documents belonging to a project and we also know in which project a scientist is working. Hence, by joining both we know which project-related documents are relevant to a scientist. The true potential of this join is exploited when personalizing the ranking of results. After a user enters a query in YASE his administrative metadata is automatically retrieved and a temporary user profile reflecting his role in the company is created. A first ranked results list is obtained by the vector space model. In the next step the hit list is re-ranked according to the user’s profile. Documents lying closer to

the user's assumed interests are ranked higher than others. This is a key difference between YASE and search engines. Our ranking idea assumes that a user's interests are reflected by his role and context embedding. However, this assumption does not always hold. Therefore we plan to allow a user to slip into different roles during a search session.

After having released YASE as a prototype for a significant part of the research community, we did a usage analysis based on log files over a period of three months. The results show a 39% usage of YASE (much more than Fast or Google Search Appliance from Table 1). This is an indication that YASE is accepted by the users and that it has a higher value compared to the other two internal search engines. It also suggests that at least some of the applied hypotheses are valid.

### 3 Conclusion and Perspectives

We have successfully used existing metadata which isn't exploited by standard search engines. Faceted navigation over the documents has been enabled and in addition the ranking of results was improved by applying a role-based adaptation. Exploiting existing metadata was the key to the success of YASE in its first prototype version.

Even though YASE is tailored to the specific environment of a research department, we argue that the concepts behind YASE allow its use in other intranet environments as well with only minor adjustments. First of all, the described shortcomings of standard search engines prevail in many other intranet environments as well. Further, domain metadata or administrative data, such as those described earlier, are available in every company or institution.

The learning features based on the search history and the inference capabilities using domain thesauri and ontologies, though partially implemented, are still to be investigated in depth. These anonymized data can be used for various purposes: recommendations of alternative queries or additional documents (URLs), improving ranking of results, etc. The integration of "deep knowledge" extracted from company databases with published documents will reveal further potentials of the adaptive search agent.

### References

- [Baeza-Yates, 99] R. Baeza-Yates, B. Ribeiro-Neto, *Modern Information Retrieval*, ACM Press, Addison Wesley, 1999
- [Bry, 03] F. Bry, P. Kröger, *Bioinformatics Databases: State of the Art and Research Perspectives*, In Proc. 7<sup>th</sup> ADBIS, 2003
- [Fagin, 03] R. Fagin, R. Kumar, K. McCurley, J. Novak, D. Sivakumar, J. Tomlin, D. Williamson, *Searching the Workplace Web*, In Proc. 12<sup>th</sup> Int. WWW Conference, 2003
- [Mühlbacher, 08] S. Mühlbacher, University of Regensburg / Roche Diagnostics GmbH Penzberg, Dissertation, Q4 2008, forthcoming
- [Xue, 03] G. Xue, H. Zeng, Z. Chen, W. Ma, H. Zhang, C. Lu, *Implicit Link Analysis for Small Web Search*, SIGIR'03, 2003

## **Integration of Business Models and Ontologies: An Approach used in LD-CAST and BREIN**

**Robert Woitsch**

(BOC Asset Management GmbH, Vienna, Austria  
Robert.Woitsch@boc-eu.com)

**Vedran Hrgovic**

(BOC Asset Management GmbH, Vienna, Austria  
Vedran.Hrgovic@boc-eu.com)

**Abstract:** This paper introduces an approach for integrating business models and ontologies supported by the PROMOTE<sup>®</sup> knowledge management system, applied in the LD-CAST project in order to involve domain experts from the business layer into the aspects of the actual service delivery situated in the IT layer. The use case in LD-CAST is the cross-border interaction between European Chambers of Commerce and potential customers from other countries.

**Keywords:** business models, workflows, ontology, Web services, integration, bootstrapping  
**Categories:** H.4

### **1 Introduction**

In the LD-CAST project [LD-CAST, 08] (Local Development Actions Enabled by Semantic Technology) the main objective is to establish an easy to use service delivery platform for European Chambers of Commerce and their end users. The LD-CAST system has to deal with many standard issues that occur in the cross-border business interaction such as language issues, different legislative, etc. Therefore the challenge was to align different business practices and models in order to create a system for service delivery to the end users.

LD-CAST had to satisfy the requirements to create a system that allows co-existence of different business processes (e.g due to legislative issues) transformed into workflows that should be composed of services from different service providers. The second challenge was twofold. First, to allow business service providers to directly influence the generation of the workflows responsible for the actual service delivery to the end users in the IT layer of the system (by simply updating their business models), and second, to allow end users to compose the service delivery by choosing one of the possible workflow configurations. The work done on the integration, as a it was a highly intensive knowledge task, was supported by applying the PROMOTE<sup>®</sup> [Woitsch, 04] approach in order to provide the LD-CAST system with a powerful knowledge management system. The paper is structured as follows: In the second chapter the possible scenarios that would benefit from the integration of the business models and ontologies are presented. The workflow governance from LD-CAST is presented in the third chapter, where the ontology and business model

generation process is described. The fourth chapter is dedicated to the conclusion and to the future work.

## **2 Scenarios for Business Models and Ontologies Integration**

The goals achieved through Business Models and Ontologies integration is the (1) possibility to share common understanding of the knowledge available in the business models among the domain experts, (2) to enable transfer of this knowledge from the domain experts who created the models to other parties involved in the business process, (3) to allow users others than IT experts to take over governing of the workflows, (4) to allow integrity and accuracy verifications of the business models based on the ongoing bootstrapping process and (5) in the human resources area for example to enable profiling of the users based on their usage of monitored items. In the next chapter, the workflow governance in the LD-CAST project will be presented.

## **3 Business Models and Ontologies Integration in LD-CAST**

This chapter will show, based on the example from LD-CAST, how the business models and consequently the ontology has been generated, how the actual integration between them was accomplished and what results have been achieved.

### **3.1 Business Models Generation**

Generation of the business models was the initial task to be performed in order to gain a starting point for the future integration of the LD-CAST system. It was a highly complex assignment due to fact that business models created were based on the every day work performed by the different actors that had to comply with diverse laws and directives in their own countries and had to take into account some other factors (e.g. ways of doing business, customs). Business models were created using the meta model top-down approach of the ADOeGov++ [Palkovits, 03] modelling language that was customized to satisfy all requirements demanded by this task. The initial modelling was conducted using the ADONIS software produced by BOC [BOC, 08], both as a rich and as a web-based client.

Business models generated were then used, utilizing the BPEL [OASIS, 06] notation, to create abstract workflows that correspond to activities found in the business models (workflows are marked as abstract, as up to this point in time they do not have any services bound to their activities).

### **3.2 Ontology Generation**

Generation of the ontology is considered to be an extremely time and resources consuming task [Cardoso, 07]. There are many methodologies that cover the aspects of the ontology generation and maintenance [Uschold, 95], [Grüninger, 95], [Fernández, 97]. They all outline that it is important to define the purpose of the ontology, to specify the terminology, build the ontology and continue with enrichment to keep the ontology alive and provide valid documentation for the end users. The approach used in LD-CAST to build the initial ontology, in respect to the mentioned

methodologies, was self-proposing, due to fact that a repository containing business models and all available description was already accessible within the system. Models stored within the system were created using the ADOeGov++ modelling language, which was enhanced to allow extraction of the concepts available inside the business models directly into the OWL [OWL, 07] format. Created concepts were then imported into the ATHOS ontology management system [ATHOS, 08]. Next step was the enrichment of the ontology using the OPAL [D'Antonio, 07] notation, by the domain experts and knowledge engineers using the ATHOS tool.

### **3.3 Business Models and Ontologies Integration**

Although many different scenarios leaning toward integration of the business models and ontologies can be identified, in LD-CAST the most important goal was to provide possibility for the governance of the workflows

#### **3.3.1 Integration Process**

The actual process of integrating the Business Models and Ontologies was fostered by the fact that tools used to create the items in question had the feature to allow collaboration with 3<sup>rd</sup> party components and systems, therefore a mash-up of the Model Viewer (Business Models) and ATHOS Viewer (Ontology) was developed and introduced to the project. This tool was used to annotate items available in the Model Repository (and other repositories) with the concepts from the Ontology.

This approach made possible to connect all items available in the repositories of the LD-CAST system to each other, thus allowing appliance of the semantic search and discovery for particular objects and services. The integration process described in this chapter was also applied to services which service providers registered to the system to be used in the concrete workflows and thus delivered to the end users

#### **3.3.2 Workflow Governance**

The idea of providing such aspect as a workflow governance (WFG) to the business users that may not be so technical savvy to carry out such tasks on the IT level, lies in the fact that it is extremely valuable to involve business domain experts directly and to use their knowledge that may otherwise stay hidden from the IT layer.

The WFG approach applied in the LD-CAST has two goals, namely on the one hand it allows the interaction of the business domain experts with the IT layer of the system, allowing them directly to interfere with the orchestration of the activities composing the abstract workflow, and on the other hand it allows the end users to fill the abstract workflow with chosen services and execute it.

This approach is made possible by using the integration between business models (abstract workflows and services) and ontology, namely as soon as business domain experts change the business models (e.g. by changing the activity composing the business models, that is annotating it by using a different concept) this change is transferred to the abstract workflow (currently this task is performed manually due to consortium agreement) and it reflects the services published to the end users on the LD-CAST Portal.

## 4 Conclusions and Future Work

The integration approach used in LD-CAST allowed efficient involvement of actors from business into the IT layer, creating novel solution and lowering complexity reflected in the amount of knowledge needed for each user to be able to address all tasks for described scenarios. This solution has been tested involving 100 end users (public and private owned companies) from Romania, Poland, Italy and Bulgaria providing feedback to enhance the integration tasks in order to simplify the usage of the system by the end users[D7.2, 08]. Based on the results derived from LD-CAST, future work to automate the process (ontology generation, annotation and integration) and enhance the search and discovery for needed services will be conducted in the EU Project BREIN[BREIN, 08]. Second goal is to transfer this solution to other application areas such as Knowledge Management which will be tackled in the EU Project Mature[MATURE, 08] and the area of Modelling support tools (Integrity and Accuracy Verification through bootstrapping process of ontology and business models evolution)

## References

- [ATHOS, 08] ATHOS, 2008, <http://leks-pub.iasi.cnr.it/Athos/> access: 03.04.2008
- [BOC, 08] BOC Group, 2008, <http://www.boc-group.com/> access: 03.04.2008
- [BREIN, 08] BREIN Project, 2008, <http://www.eu-brein.com/> access: 03.04.2008
- [Cardoso, 07] Cardoso, J.: The Semantic Web Vision: Where Are We?, IEEE Intelligent systems 2007, pp 22-26, 2007
- [D 7.2, 08] D 7.2 Validation Report, <http://www.ldcastproject.com/> access 03.04.2008
- [D'Antonio, 07] F. D'Antonio, M. Missikoff, F. Taglino, Formalizing the OPAL eBusiness ontology design patterns with OWL. I-ESA 2007
- [Fernández, 97] Fernández, M., Gómez-Pérez, A. and Juristo, N.: METHONTOLOGY: From Ontological Art Towards Ontological Engineering. AAAI97 Spring Symposium Series, 1997
- [Grüninger, 95] Grüninger, M. and Fox, M.: Methodology for the Design and Evaluation of Ontologies. Proceedings of IJCAI95, 1995
- [LD-CAST, 08] LD-CAST Project, 2008, <http://www.ldcastproject.com/> access: 03.04.2008
- [MATURE, 08] Mature-Homepage, <http://mature-ip.eu>, access: 03.04.2008
- [OASIS, 06] OASIS, 2006, <http://docs.oasis-open.org/wsbpel/2.0/> access: 03.04.2008
- [OWL, 07] OWL Working Group, 2007, <http://www.w3.org/2007/OWL/> access: 03.04.2008
- [Palkovits, 03] Palkovits, S., Wimmer, M.: Processes in e-Government – A Holistic Framework for Modelling Electronic Public Services. In EGOV 2003, Springer Verlag, 2003.
- [Uschold, 95] Uschold, M. and King, M.: Towards a Methodology for Building Ontologies. Proceedings of IJCAI95's Workshop on Basic Ontological Issues in Knowledge Sharing, 1995.
- [Woitsch, 04] Woitsch, R., Karagiannis, D.: Process Oriented Knowledge Management: A Service Based Approach, In Proceedings of I KNOW '04, Graz, Austria



## Semantic Tagging and Inference in Online Communities

**Ahmet Yıldırım**

(Boğaziçi University, Turkey  
ahmet@ahmetyildirim.org)

**Suzan Üsküdarlı**

(Boğaziçi University, Turkey  
suzan.uskudarli@cmpe.boun.edu.tr)

**Abstract:** In this paper we present UsTag, an approach for providing user defined semantics for user generated content (UGC) and process those semantics with user defined rules. User semantics is provided with a tagging mechanism extended in order to express relationships within the content. These relationships are translated to RDF triples. RDF triples along with user defined rules enable the creation of an information space, where the content and its interpretation is provided by users.

**Key Words:** semantic web, metadata, online communities, collective knowledge, tagging, semantic tagging

**Category:** M.0, M.7

### 1 Introduction

Recent advances in Web technologies, such as wikis and weblogs (blogs) have enabled novice users to become content producers in addition to consumers. Such technologies have dramatically increased user contribution resulting in a massive amount of content ranging across all human interests. While there is no shortage of Web content, technologies for effectively finding and utilizing this content remains quite limited.

Our work focuses on enabling and utilizing user generated semantics. We introduce an extension to tagging for the purpose of providing user defined relationships between content. We, furthermore, introduce a mechanism for providing user defined rules for processing these relationships. Due to severe space limitations, we are only able to provide our work outline.

In Section 2, we present related work, in Section 3, we describe our approach, in Section 4, we give future work and conclusions related to this work.

### 2 Related Work

There are other approaches that also aim to enrich user generated content. Semantic Wikipedia enables users to embed relations in articles by extending the link syntax [Volkel et al.(2006)Volkel, Krotzsch, Vrandečić, Haller, and Studer].

These markups relate the article to another subject with user defined relations. RDF triples are generated from these relations enabling semantic searching.

SemKey [Marchetti et al.(2007)Marchetti, Tesconi, and Ronzano] enables associating tags with Web resources. Triples of <Content URI,relation URI, Tag URI>are created, where the relations of three types: hasAsTopic, hasAsKind, and myOpiononIs. These relations are considered the most useful.

Flickr [Flickr(2004)] provides an API that helps users define machine processable information for the content. This API supports Machine Tags [Flickr(2008)] that essentially enable users to add extra information with the help of the tag syntax. Flickr machine tags have a namespace, a predicate, and a value. The value is not a semantic web URI, but can be considered as a semantic web literal value. Flickr does not export machine tags as RDF. Searching a content semantically can be done using Flickr API.

MOAT [MOAT(2008)] is a framework that provides a mechanism for users to define tags and their meanings using semantic web URIs. MOAT comes with a client and a server. A moat client interacts with the server to retrieve tags and their meaning. While the user is entering the tag, if the intended meaning is not found, the user defines a URI for that tag. MOAT uses FOAF to identify people and relate tags to creators.

EntityDescriber [Connotea(2008)] lets Connotea users tag resources with the terms coming from structured knowledge systems such as Freebase or ontologies.

### 3 Our Approach: UsTag

UsTag is a User Generated Content (UGC) environment, that enables users to tag the content with its semantics and to define rules to process these semantics with an inference mechanism implemented in the system. These definitions lead to better search results, and easily finding and utilizing the content. Considering that we want average users to provide semantics, we need an easy and familiar mechanism. Users define the semantics by adding a predicate to the tag. The scenario that we envision is that someone will make a contribution, others will make corrections, additions, and define relationships by semantically tagging the content.

#### 3.1 Definitions

We use the term “Conventional Tag” to refer a tag which is only a label seen in existing tagging systems. We use the term “predicate” to refer to the type of a relationship constructed by semantic tagging. A predicate can be entered while users are tagging the content.

UsTag supports conventional tags by relating the content to the tag with the predicate “is-about”. “is-about” is the default predicate, but can be changed

while installing the system. A predicate is not required to be entered. In this case, user feels like using a conventional tagging system.

“Subject” is the tagged content, and “object” is the tag itself. A subject and an object can be related using a predicate. All subjects, objects and predicates are URI’s in the system. A subject, an object, and a predicate create a relation which is output as RDF triple.

If user desires to input semantic information about the content, he clicks on the Tag button. The user enters a tag and a predicate for the tag. This type of tag is a “semantic tag”. While tagging, tags starting with what the user is typing are suggested via an auto complete area.

A rule is used to process the inserted relationships. A rule is defined by the user via the rule definition interface for a specific predicate. A rule consist of an IF part and a THEN part. If the IF part of the rule is satisfied, then the relations defined in the THEN part are inserted into the system.

### 3.2 Semantic Search

In addition to basic text based search, UsTag supports semantic search. User can ask a query in novel-author domain such as “Find Movements that influenced authors who are influenced by Modernism”. This query is not asked in natural language, but through semantic search interface.

### 3.3 Inference

UsTag supports inference when rules are input by user. As rules are processed, new relationships appears as results and these relationships are added to the relationship repository. Predicates and tags for a content are listed below the content in an infobox. The inferred relationships are also included in the infobox like user defined relationships. Both in basic search and semantic search, inferred relationships are taken into account. We have implemented inference mechanism using Jena[HPLabs.(2003)].

With an example, we will explain the inference mechanism. We will use “subject:predicate:object” notation to represent a relationship. Suppose that we have user defined relationships: “Berlin:located-in:Germany”, “Berlin:is-a:city”, and “Germany:located-in:Europe”. If a user defines “located-in” predicate as transitive, and when we query cities in Europe, Berlin appears in results. Defining “located-in” as transitive is a primitive action. User just clicks on the transitive button in the predicate properties page to declare it as transitive. In addition, the system allows definition of complex rules. For instance, in novels-authors domain, a user can define a rule such as “If a novel is influenced by a movement, then the author of the novel is also influenced by that movement.”. Rules are not defined in natural language, but in rule definition interface.

## 4 Future Work and Conclusions

We have explained related work and UsTag. UsTag supports user defined rules and processing user defined rules with user defined relationships. These relationships emerges into the system by semantic tagging. For the future, we are planning to develop a simpler user interface for rule definition and semantic search. We are also planning to open the system for large scale user test and evaluation.

Our initial experience for UsTag is that the system achieves its goals of remaining lightweight, as no tags need to be given and common tagging behavior is supported. The use of semantic tags are optional, however when given, they nicely extend the utility of the system with better search results that lead to more comprehensive content creation. User defined rules enabled inference and introduction of new relationships that are not input by users. This paper presents the first results of this work. The early prototype has been very beneficial in getting early feedback and provides a very useful platform for experimentation. Our experience is encouraging with respect three are primary motivation of enabling easy user content creation that is machine processable, processing this content for eliciting information that is not input by user, and effective information retrieval. We are continuing work on the approach as well as the system.

### Acknowledgements

This work has been partially supported by BAP07A107 and BAP08A103 funding.

### References

- [Connotea(2008)] Connotea. Entitydescriber, 2008. URL <http://www.connotea.org/wiki/EntityDescriber>.
- [Flickr(2004)] Flickr. *Flickr - The best way to store, search, sort and share your photos*. <http://www.flickr.com>, 2004. URL <http://www.flickr.com>.
- [Flickr(2008)] Flickr. Machine tags wiki, 2008. URL <http://www.machinetags.org/wiki/>.
- [HPLabs.(2003)] HPLabs. *Jena Semantic Web Framework*. <http://jena.sourceforge.net/>, 2003. URL <http://jena.sourceforge.net/>.
- [Marchetti et al.(2007)Marchetti, Tesconi, and Ronzano] Andrea Marchetti, Maurizio Tesconi, and Francesco Ronzano. Semkey: A semantic collaborative tagging system. 2007.
- [MOAT(2008)] MOAT. Meaning of a tag, 2008. URL <http://moat-project.org/>.
- [Volkel et al.(2006)Volkel, Krotzsch, Vrandecic, Haller, and Studer] Max Volkel, Markus Krotzsch, Denny Vrandecic, Heiko Haller, and Rudi Studer. Semantic wikipedia. In *WWW '06: Proceedings of the 15th international conference on World Wide Web*, pages 585–594, New York, NY, USA, 2006. ACM Press. ISBN 1595933239. doi: 10.1145/1135777.1135863. URL <http://portal.acm.org/citation.cfm?id=1135777.1135863>.

## **Semantics-based Translation of Domain Specific Formats in Mechatronic Engineering**

**Manuela Plößnig**

(Salzburg Research Forschungsgesellschaft, Salzburg, Austria  
manuela.ploessnig@salzburgresearch.at)

**Merlin Holzapfel**

(Salzburg Research Forschungsgesellschaft, Salzburg, Austria  
merlin.holzapfel@salzburgresearch.at)

**Wernher Behrendt**

(Salzburg Research Forschungsgesellschaft, Salzburg, Austria  
wernher.behrendt@salzburgresearch.at)

**Abstract:** We present a semantics-based approach to integrating product information from multiple domains in the field of mechatronics in order to allow for more efficient cross-domain and collaborative engineering. We use the DOLCE foundational ontology as a basis for representing the mechatronic subdomains of software engineering, mechanical engineering and electronic engineering, and we build adapters based on the common ontological ground in order to translate between domain-specific data exchange formats and standards.

**Keywords:** mechatronics, ontologies, collaboration, translation

**Categories:** M.4

### **1 Ontology-based Modelling of Mechatronic Product Data**

In the ImportNET<sup>1</sup> project, we focus on cross-domain collaboration between mechanical, electronic and software engineers. An essential aspect of cross-domain collaborations is the exchange of information to clarify design issues. In order to bridge engineering domains it is necessary to integrate the data formats and underlying data models of the existing single domain applications. Hence, one of the aims in ImportNET is to provide a version consistent integrated view of all needed information (e.g. design documents) related to the overall mechatronic product in order to propagate design changes into the integrated view and into the domain views, respectively. In order to bridge the structural and semantic gap between electronic, mechanical and software design, it is no longer sufficient to only consider translations between standardized formats. Rather, it is necessary to bring together information model concepts as well as organisational and interaction concepts on different abstraction levels. The design of cross-domain engineering ontologies is considered to

---

<sup>1</sup> The ImportNET project has been part-funded by the EC in the Sixth Framework Programme under contract IST-2006-033610 (<http://www.importnet-project.org/>)

be an essential area of research for mechatronics since it attempts to bring together concepts and ideas in relation to a product or system [Bradley, 04]. One promising approach is to use knowledge models based on foundational ontologies. Foundational ontologies are formal theories which are developed independently of specific domains aimed at facilitating mutual understanding in the large. These ontologies comprise only general concepts and relations, and to be applied they need to be populated with notions specific to the domain of interest. The foundational ontology used in ImportNET for modelling the mechatronic ontologies is DOLCE, the Descriptive Ontology for Linguistic and Cognitive Engineering [Masolo et al., 03]. It allows the integration of insular and distributed knowledge models into a common shared knowledge model, e.g. in our case, the integration of existing product data standards or standardised process descriptions. The advantage is clear – the common basis of a foundational ontology offers the possibility to connect separated knowledge models.

Engineering in mechatronics is complex and knowledge intensive. Until now, development in engineering has mostly taken place separately in the domains involved, on the basis of established development methods mainly tailored for a specific domain. The paper describes the ImportNET approach towards a structured and integrated view on a mechatronic product for cross-domain engineering on the basis of a DOLCE-based knowledge model which allows to integrate well-established industry standards for data exchange. As an example we will present IDF (Intermediate Data Format), a specification designed to exchange printed circuit assembly data between mechanical design and printed circuit board (PCB) layout.

## 2 Information Exchange Using a Reference Ontology

Our main objective is to provide a way to integrate product information from the different domains that make up mechatronics. The ImportNET ontology can act as the “hub-representation” between the different data standards. This is achieved by developing so-called adapters, which allow us to integrate design documents such as CAD files into the ImportNET system and which act as bi-directional translation services between standardised file formats and the ImportNET data base, via the Reference Ontology. One of the components that are being built in ImportNET is the Intelligent Adapter Generation Tool (IAGT). The IAGT allows a specialist user to build adapters between specific file formats and the ImportNET ontology. The user provides a formal specification of the file format in question and uses a graphical interface to specify a mapping between file format and ontology. From this (formal model of format and mapping specification), an adapter is then created automatically.

A typical use case for mechatronic collaboration is the process of clarifying design issues about a PCB. The PCB used in a mechatronic product is represented in both mCAD (mechanic domain) and eCAD (electronic domain) files. Thus, if the electronic engineer changes the position of a component, it needs to be updated in mCAD because eCAD and mCAD designs must be synchronised. Currently this is done only by data exchange via STEP and IDF. The problem is that this replaces the entire design either in eCAD or mCAD. It is therefore not possible to transfer movements of single components or holes, making it impossible to efficiently perform parallel work on the PCB design in both eCAD and mCAD. The ImportNET approach allows the modification of components integrated with the synchronization

of the domain-specific designs. The design tasks are derived from a reference process for engineering changes (which is beyond of the scope of this paper) and an integrated and structured PDM view which includes and updates the eCAD and mCAD views.

### 3 Mechatronic Artefacts and Product Data Management

The focal point of a successful Product Data Management (PDM) are the information artefacts which can be of complex nature and accompany a product through the whole lifecycle. Current solutions do not provide an integrated view on the exchanged data, but allow a mapping of data structures between the involved systems. In ImportNET a structured PDM view describes the component structure of the intended product as a neutral model. The integrated product data view includes all three engineering domains. In the case of mechanical and electronic domain, ImportNET is able to extract information out of design documents (e.g. the components of a PCB described in an IDF file). Concerning the software domain ImportNET currently is able to include software-specific files (e.g. design documents, source code, libraries). The PDM view helps to verify whether all product components are available, and it stores domain-specific information as documents (e.g. 3D models created by a mCAD system) or as information entities. Hence, the structured PDM view in ImportNET integrates components including their detailed information from all three domains.

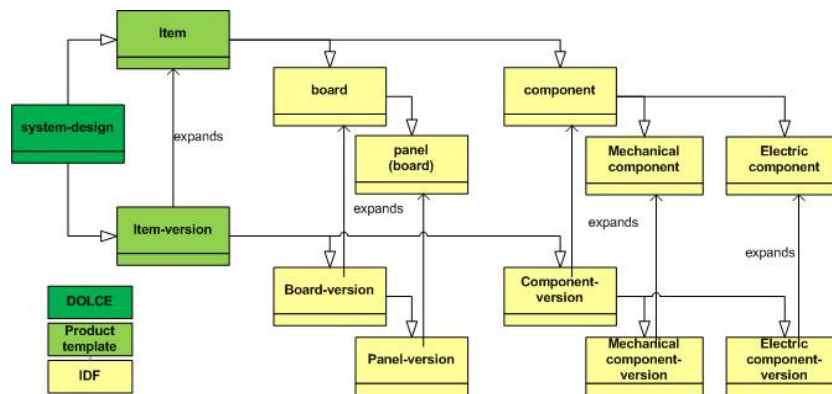


Figure 1: Item types based on IDF

The basic ontology fragment for the structured product data view defines the basic elements. A component in the ImportNET ontology is called an *item* and represents either an atomic component (called a part) or an assembly which represents the root of a subtree again including parts and/or further subassemblies. Additionally, a PDM structure also manages the location of each component in relation to its parent node. This ontology fragment represents a basic model on a top level design view. With it, it is not possible to distinguish between different kinds of components; standards like IDF allow for a much more detailed level of modelling. In order to provide a more defined and useful ontology it is necessary to extend the basic design template in much the same way as the template itself extends DOLCE.

IDF version 3.0 [IDFO-96]) is a comparably simple example for this kind of ontology expansion: a set of IDF files (Board or Panel file and Library file) defines the design of a PCB including routing and outline information as well as component placement and outline. IDF differentiates between boards, panels and components (electric and mechanical). Components can be placed on both boards and panels, while only panels can have boards placed on them. Figure 1 shows how to represent these concepts as subtypes (of *item* and *item-version*) in the basic product template. IDF provides a number of shape definitions associated with either the PCB or its components. For the board, various outline and keep-out definitions exist - they describe not only the board itself but also areas on the board that have electric (e.g. routing) or mechanical (e.g. placement) significance. This information, as the 2 ½D (outline plus height) representation given for the components and the information about drilled holes, can be incorporated into the ImportNET ontology as specific types of *shape association* which can be part of either *electric* or *mechanical design view* (modelling the idea of multiple views on one item). Electric properties finally, which can be associated with electronic components in IDF, are represented as subclasses of the concept *electronic property*, also a part of *electronic design view*.

#### 4 Conclusions and Future Work

We have shown a top-down approach for modelling mechatronic artefacts based on a structured PDM view and based on the foundational ontology DOLCE. We use a high level design template representing a generic model of a product, which can be specialised for existing industrial data standards. As a first industry standard IDF has been included in this generic ontology and an IDF adapter has been implemented in order to exchange IDF data between mCAD and eCAD systems. The next activity is to integrate concepts from the much more detailed and comprehensive STEP AP 214 [ISO-03] (Core Data for Automotive Mechanical Design Processes). Inclusion of STEP concepts will necessitate significant further extensions of the ImportNET Ontology, but will also prove the value of using foundational ontologies as a "semantic bus" in order for the models to remain modular and expandable.

#### References

- [Bradley, 04] Bradley, D.: What is Mechatronic and Why Teach It? International Journal of Engineering, Education (2004), online available (Last access: 2008-04-08): [http://findarticles.com/p/articles/mi\\_qa3792/is\\_200410/ai\\_n10298146/pg\\_1](http://findarticles.com/p/articles/mi_qa3792/is_200410/ai_n10298146/pg_1)
- [IDFO-96] Intermediate Data Format – Mechanical Data Exchange Specification for the Design and Analysis of Printed Wiring Assemblies (Version 3.0, Revision 1), Online, available at: [http://www.aertia.com/docs/priware/IDF\\_V30\\_Spec.pdf](http://www.aertia.com/docs/priware/IDF_V30_Spec.pdf) (as of 14.04.2008), 1996
- [ISO-03] ISO 10303-214, Industrial automation systems and integration — Product data representation and exchange — Part 214: Application protocol: Core data for automotive mechanical design processes, International Standard, 2nd Edition, 2003
- [Masolo et al., 03] Masolo, C., Borgo, S., Gangemi, A., Guarino, N., Oltramari, A. 2003. WonderWeb DeliverableD18 Ontology Library (final).



# **Semantifying Requirements Engineering – The SoftWiki Approach**

**Steffen Lohmann, Philipp Heim**

(University of Duisburg-Essen, Germany  
{steffen.lohmann, philipp.heim}@uni-duisburg-essen.de)

**Sören Auer, Sebastian Dietzold, Thomas Riechert**

(University of Leipzig, Germany  
{auer, dietzold, riechert}@informatik.uni-leipzig.de)

**Abstract:** This paper presents an integrated approach of basing requirements engineering on semantic technologies. First, the general approach of semantifying requirements and the underlying ontology are introduced. Second, tools that support requirements elicitation, development, analysis, and exchange on the basis of the semantic foundation are described.

**Keywords:** Semantic Web, Requirements Engineering, Distributed Software Development, Ontology-Driven, User-Oriented, Wiki, Semantic Interoperability, Linked Data

**Categories:** D.2.1, D.2.12, H.3.2, I.2.4

## **1 Motivation**

Semantic interoperability, linked data, and a shared conceptual foundation become increasingly important prerequisites in software development projects that are characterized by spatial dispersion, large numbers of stakeholders, and heterogeneous development tools. Founding distributed software development on semantic web technologies seems promising in order to serve these demands.

The *SoftWiki*<sup>1</sup> project focuses specifically on semantic collaboration with respect to requirements engineering. Potentially very large and spatially distributed groups of stakeholders, including developers, experts, managers, and average users, shall be enabled to collect, semantically enrich, classify, and aggregate software requirements. Semantic web technologies are used to support collaboration as well as interlinking and exchange of requirements data. In the following, we will present the general approach and the tools we are currently developing in this context.

## **2 Semantification of Requirements**

Within the SoftWiki approach, each requirement gets its own URI making it a unique instance on the semantic web. Then, it is linked to other resources using semantic web standards such as RDF and OWL. To ensure a shared conceptual foundation and semantic interoperability, we developed the *SoftWiki Ontology for Requirements*

---

<sup>1</sup> Research project, funded by the German Federal Ministry of Education and Research – <http://softwiki.de/>

*Engineering* (SWORE) [Riechert et al. 2007] that defines core concepts of requirement engineering and the way they are interrelated. For instance, the ontology defines frequent relation types to describe requirements interdependencies such as *details*, *conflicts*, *related to*, *depends on*, etc. The flexible SWORE design allows for easy extension. Moreover, the requirements can be linked to external resources, such as publicly available domain knowledge or company-specific policies.

We call the whole process *semantification* of requirements. It is envisioned as an evolutionary process: The requirements are successively linked to each other and to further concepts in a collaborative way, jointly by all stakeholders. Whenever a requirement is formulated, reformulated, analyzed, or exchanged, it might be semantically enriched by the respective participant. However, in order to reduce the user effort and to ease participation, stakeholders are not forced to semantify requirements.

### 3 Tool Support

We are currently developing several applications within the project that enable the elicitation, development, analysis, and exchange of semantified requirements (see Figure 1).

The central platform for semantic collaboration is based on the *OntoWiki* tool [Auer et al. 2006] that is extended to support requirements engineering according to the SWORE ontology. The effort and formal overhead for expressing or modifying requirements and relations is minimized due to the adoption of the *Wiki* paradigm [LC01]. The collaboration is supported by common wiki features such as revision control mechanisms allowing to track, review, and selectively rollback changes or a facility to discuss requirements.

The central platform is extended by decentralized participation channels. The bottom left screen in Figure 1 shows a tool that can be easily integrated into the web browsers of users. It enables the users to express requirements on basis of an already existing web application. In addition, it links the user input to application parts and the usage context. These relations can be semantified if the application or usage context is represented in an ontology and linked to the SWORE. Such context relations can be valuable for later analysis, reconstruction, and understanding of requirements.

A pre-defined topic structure supports the classification of requirements. Depending on the respective domain of the software project, the topic structure can be easily adapted or extended in the administration mode. The platform implements the *SKOS Core Vocabulary*<sup>2</sup> as representation form for the topic structure to enable semantic interoperability with other applications. In addition, stakeholders can tag requirements with freely chosen key words, resulting in an emerging tag space that represents the stakeholders' vocabulary. The tagging process is also ontologically grounded<sup>3</sup>.

According to the different ways of semantification, the system provides various access points and ways to navigate the requirements. For instance, the user can

---

<sup>2</sup> Simple Knowledge Organization System – <http://www.w3.org/2004/02/skos/>

<sup>3</sup> An ontology for tags – <http://www.holygoat.co.uk/owl/redwood/0.1/tags/>

explore the requirements by using the tree navigation and additionally narrow down the shown requirements set by choosing tags from the tag cloud. Requirements that are linked to geographical locations according to the *Basic Geo Vocabulary*<sup>4</sup> can additionally be explored on a map. Furthermore, the system provides graph visualizations that specifically support the discovery of relationships and interdependencies between requirements by highlighting instances that are semantified in a similar way.

In order to enable semantic interoperability with further tools, the requirements collection can be exported in RDF-format according to the SWORE schema or other underlying ontologies. Alternatively, the requirements can be accessed via a SPARQL endpoint. Moreover, we are currently working on an extension that enables export in RIF<sup>5</sup>-format to integrate the SoftWiki approach with established requirements and project management tools – even though this goes along with a loss of semantics.

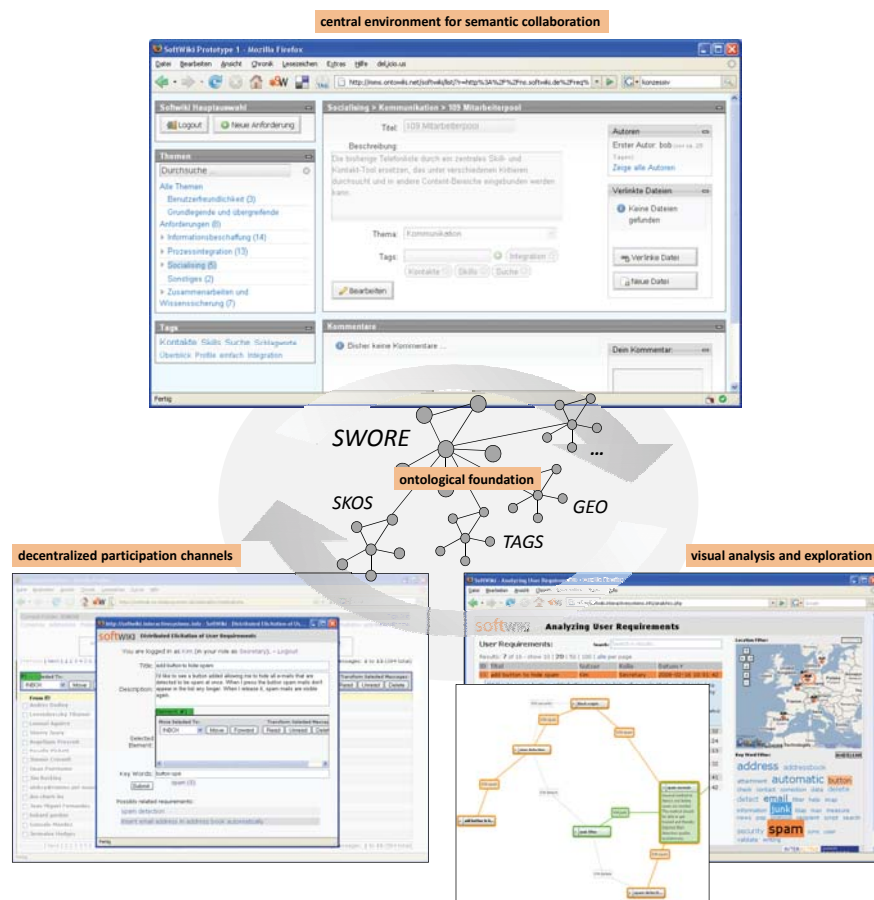


Figure 1: Semantic based tool support for requirements engineering

<sup>4</sup> Basic Geo Vocabulary – <http://www.w3.org/2003/01/geo/>

<sup>5</sup> Requirements Interchange Format – <http://www.automotive-his.de/rif/doku.php>

## **4 Conclusion and Future Work**

The SoftWiki approach of semantifying requirements engineering aims to support distributed software development with a large number of participants. First experiences with use cases of the project indicate several benefits compared to non-semantified requirements engineering, including easier detection of conflicts and dependencies or better means to exchange requirements. Our current activities include further development of the tools and the semantic foundation as well as a comprehensive testing of the approach in use cases of different application domains.

## **References**

- [Auer et al. 2006] Auer, S., Dietzold, S., Riechert, T.: “OntoWiki – A Tool for Social, Semantic Collaboration”; Proceedings of the 5th International Semantic Web Conference, LNCS 4273, Springer, Berlin / Heidelberg (2006), 736-749.
- [Leuf and Cunningham 2001] Leuf, B., Cunningham, W.: “The Wiki Way: Quick Collaboration on the Web”. Addison-Wesley Longman, Amsterdam (2001)
- [Riechert et al. 2007] Riechert, T., Lauenroth, K., Lehmann, J., Auer, S.: “Towards Semantic based Requirements Engineering”. Proceedings of the 7th International Conference on Knowledge Management (2007), 144-151.

## **RDFCreator – A Typo3 Add-On for Semantic Web based E-Commerce**

**Harald Wahl**

(University of Applied Sciences Technikum Wien, Vienna, Austria  
wahl@technikum-wien.at)

**Markus Linder**

(Smart Information Systems GmbH, Vienna, Austria  
ml@smart-infosys.com)

**Alexander Mense**

(University of Applied Sciences Technikum Wien, Vienna, Austria  
mense@technikum-wien.at)

**Abstract:** In the wide field of E-Commerce it has become well accepted to use RDF files based on ontologies for storing product data decentralized. The proposed poster demonstrates functionality and application of an add-on for the open source content management system (CMS) Typo3 called RDFCreator. The RDFCreator add-on allows to easily define RDF instances based upon any well defined ontology. For this purpose the ontology can be chosen (for example using ontology of [www.myontology.org](http://www.myontology.org)) and the user automatically gets a matching questionnaire to fill in. After completing the form a corresponding RDF file is generated and can be stored locally or on a server for further processing.

**Keywords:** RDF instantiation, ontology processing, Semantic Web Based E-Commerce

**Categories:** H.3.2, H.5.0

### **1 Introduction**

Recently, using Semantic Web Technologies in E-Commerce have become more and more popular. For example product information is stored in decentralized files, mostly RDF, based upon ontologies defined by RDFS or OWL. For example these files could represent products of companies' product catalogues available for viewing via the internet. On the other hand software agents could search for such kinds of files to navigate through the catalogues.

The exact and correct writing of a RDF files (i.e. syntactically and semantically right) based on ontologies can be a rather complex process. Especially skilled employees are needed for creation.

The RDFCreator was implemented to simplify RDF creation for employees who are not trained in reading RDFS or OWL.

## 2 Functionality of RDFCreator

The idea behind RDFCreator was to create a tool for employees in a company who are familiar with product details but who are not familiar with writing correct RDF files. In companies some employees have to edit company's web presence. These employees mostly have deep knowledge of company-internal information like product details but they mostly do not know the RDF syntax. The RDFCreator simplifies the creation of any RDF file as it could be embedded in the company's CMS.

Based upon any ontology the tool offers a simple questionnaire to fill in. Details of supposed data types are provided, data can be validated and faulty insertion can be avoided. Sometimes, not all details of the ontology are needed so the tool offers the possibility to hide specific entries. Even the order of entries is changeable in a simple way. After completing the form the corresponding RDF file is generated and can be stored for further processing.

The following chapters indicate the steps supported by the RDFCreator.

### 2.1 Step 1: OWL File Selection

The RDF creation is initialized by selecting an OWL file that contains the ontology definition. The OWL can be read from a local file or even from an URI via the internet. At the current state the RDFCreator can only access internet resources without HTTP authentication.

### 2.2 Step 2: Ontology Selection

At step 2 the OWL file is primarily parsed. The defined class elements as well as the main ontology are read and presented to the user (compare Figure 1). The user decides for a class element or even for the main ontology and provides this selection to the OWL Parsing.



Figure 1: Users can choose the main ontology or a class element

### 2.3 Step 3: OWL Parsing

The OWL parser processes the OWL file depending on selection of step 2. It searches for relevant namespaces, classes, elements and properties. Internally, data is stored in a meta-structure consisting of nested arrays.

### 2.4 Step 4: Survey Generation

Using this meta-structure the RDFCreator constructs a survey. The survey shows headings as well as supposed data types and is a kind of questionnaire. Supported data types are String, Double, Float, Boolean, Datetime, Date, Time, Hexbinary, and Base64binary. If restrictions cannot be found the default data type String is chosen.

If not all entries should appear in the resulting RDF or if a specific order of storage is needed the RDFCreator offers possibilities to hide entries or to change the order (compare Figure 2).

**1. MobileTelephone**  
hide down   
erwartete Eingabe: String

**2. Display 2 Information**  
Shows the type of information on the second display. hide up down   
erwartete Eingabe: String

**3. Universal Serial Bus (USB)**  
USB port hide up down TRUE  FALSE   
erwartete Eingabe: Wahrheitswert

**4. Manufacturer**  
Legally valid designation of the natural or judicial person which is directly responsible for the design, production, packaging and labelling of a product in respect to its being brought into circulation, whereby this person is responsible independently of whether these activities were carried out by this person or by third parties on its behalf. hide up  
down   
erwartete Eingabe: String

**5. Display1 Resolution Vertical**  
Vertical resolution of the main display. hide up down   
erwartete Eingabe: String

Figure 2: The tool offers a survey depending on the ontology selection

### 2.5 Step 5: RDF Export

Finally, after completing the form the RDFCreator transfers data into a valid RDF file. The allowed file size, set by the CMS, is the one and only restriction of the export process.

### 3 Technical Details

#### 3.1 Programming language

The tool is implemented using the programming language PHP in version 4. It is implemented from scratch and no third party libraries are used. A transfer to PHP version 5 should work fine but is not tested so far.

#### 3.2 Integration into Typo3

There are several possibilities to integrate RDFCreator into Typo3: using TypoScript, defining a plug-in or an extension. Each method has specific pros and cons. More details about integration can be found at the Typo3 web site ([www.typo3.org](http://www.typo3.org)).

### 4 Current Status and Future Work

So far, the RDFCreator is available in an Alpha Release. Basically, bug fixing is the main task to reach a stable release. A widespread usage of the tool can be guaranteed if RDF representation can be automatically visualized in a company's web site. On the other hand integration in other content management systems than Typo3 will be realized. Therefore, PHP based content management systems indicate first candidates for further integration.

### References

[Antoniou, 08] Antoniou, G., Van Harmelen, F., A Semantic Web Primer, 2<sup>nd</sup> edition. Cambridge: The MIT Press, 2008.

[Berners-Lee, 01] Berners-Lee, T., Hendler, J., Lassiala, O., The Semantic Web A new form of Web content that is meaningful to computers will unleash a revolution of new possibilities. in: Scientific American, May 2001 issue

[Fensel, 03] Fensel, D., Hendler, J., Liebermann, H., Spinning the Semantic Web, Cambridge: The MIT Press, 2003.

[Myontology, 08] Creation of ontologies, 2008, <http://www.myontology.org/>

[Typo3, 08] Open Source CMS, 2008, <http://www.typo3.org/>



## **TRIPLIFICATION CHALLENGE**

**as part of  
I-SEMANTICS '08  
International Conference on  
Semantic Systems**

For the success of the Semantic Web it is from our point of view crucial to overcome the chicken-and-egg problem of missing semantic representations on the Web and the lack of their evaluation by standard search engines. One way to tackle this problem is to reveal and expose existing structured (relational) representations, which are already backing most of the existing Web sites. The Linking Open Data Triplification Challenge aims to expedite this process by raising awareness and showcasing best practices.

The papers of the Triplification Challenge did not go through the peer-review evaluation process for scientific papers. Instead the evaluation of the contributions took place in two phases. In the first phase the conference chairs of I-SEMANTICS nominated 8 submissions out of the 15 ones with regard to the formal criteria set out in the call for submissions. In the second phase the organizing committee and the Linking Open Data (LOD) community were invited to vote about their favourite nominations. Based on these votes the Organizing Committee finally selected the winners, which are announced at the conference. The organizers would like to thank everybody for participating in the challenge and contributing to get LOD out to the Web!

### **Challenge Organization Committee Members**

- Sören Auer, Universität Leipzig
- Chris Bizer, Freie Universität Berlin
- Ivan Herman, World Wide Web Consortium
- Kingsley Idehen, OpenLink Software
- Andreas Koller, punkt.netServices, Austria
- David Peterson, BoAB interactive

## DBTune – <http://dbtune.org/>

Yves Raimond

(Centre for Digital Music, Queen Mary, University of London, United Kingdom  
[yves.raimond@elec.qmul.ac.uk](mailto:yves.raimond@elec.qmul.ac.uk))

**Abstract:** We describe the different datasets and applications hosted by our DBTune service, available at <http://dbtune.org/>. DBTune now gives access to more than 14 billion music-related RDF triples, as well as a wide range of SPARQL end-points. We also provide end-user interfaces to interact with this data.

**Key Words:** Music, DBTune, Prolog, RDF, Sparql, GNAT, Linking Data

**Category:** H.3., H.4., H.5

### 1 Introduction

We set up in early 2007 the DBTune service available at <http://dbtune.org/> in order to experiment with heterogeneous interlinked music datasets, designed using the Music Ontology [Raimond et al., 2007] and related vocabularies. To build this service, we designed a set of lightweight SWI-Prolog [Wielemaker et al., 2008] modules to convert existing data sources to linked data. We also designed an automated interlinking algorithm, allowing us to relate our datasets to other ones available on the data web. Finally, we provide a set of tools and user interfaces to exploit this data.

### 2 Publication and interlinking tools

In order to publish heterogeneous data sources on the web, we created the Prolog-2-RDF software (P2R<sup>1</sup>, available as a module for SWI-Prolog). P2R translates Prolog predicates to RDF dynamically, when SPARQL queries are issued to a particular end-point. As in other publication tools such as D2R [Bizer and Cyganiak, 2006], P2R uses a declarative mapping from Prolog predicates to a set of RDF triples. Prolog predicates can themselves wrap a variety of data sources, from relational databases to web services or spreadsheets. Once a SPARQL end-point is set up through P2R, another SWI-Prolog module (UriSpace) can be used to publish the corresponding data as linked data. P2R was used in other large-scale projects, such as RIESE, publishing European statistical data as linked data. The RIESE end-point gives access to 3 billion triples, generated on-the-fly from a set of spreadsheets<sup>2</sup>.

<sup>1</sup> <http://km-rdf.googlecode.com/files/p2r.tar.gz>

<sup>2</sup> More details about the use of P2R within this project are available at <http://tinyurl.com/6rx7fy>

For small datasets, such as an individual's FOAF file, it is possible to create links to other datasets manually. However, doing so for large datasets is impractical: we need a way to automatically detect the overlapping parts of heterogeneous datasets. We tackled this issue in [Raimond et al., 2008]. We showed that interlinking methodologies based on traditional record linkage techniques (where we try to match two resources based on a comparison between matching literal properties) perform really badly. By also encoding the respective types in this comparison, the results are slightly better. However, a test on Jamendo and Musicbrainz still gives an unacceptable 33% rate of false-positives. Intuitively, we could examine resources in the neighbourhood of the two resources we are trying to match, in order to take a more informed interlinking decision. If by any chance we are still not able to take a decision, we could examine resources in the neighbourhood of this neighbourhood, etc. We therefore designed in [Raimond et al., 2008] an algorithm matching whole RDF graphs at once, with a really low rate of false-positives. In the case of Jamendo and Musicbrainz, it drops to 3%. We developed a SWI-Prolog implementation of this algorithm<sup>3</sup>. A variant of this algorithm is implemented within the GNAT software described in § 4.

### 3 Published datasets

We used these publication and interlinking tools to make 7 datasets available as linked data. We published the Magnatune and the Jamendo Creative Commons repositories, interlinked with Geonames, DBpedia and Musicbrainz. We published the MySpace and the Last.fm social networks, interlinked with Musicbrainz. We published the BBC John Peel sessions and the BBC playcount data (artists per brand or episode), interlinked with DBpedia, Musicbrainz, and the BBC programmes catalogue. We also published our own linked data version of Musicbrainz, with links to DBpedia, Lingvoj and MySpace, using D2R Server. All the source code and the mappings running these different services are available at <http://sourceforge.net/projects/motools/>. Overall, DBTune gives access to more than 14 billion triples.

### 4 End-user tools

In order to provide an interesting user experience using these vast amounts of data, we designed two tools, GNAT and GNARQL<sup>4</sup>. GNAT finds web identifier for tracks in a personal music collection. GNARQL aggregates structured data from these entry points in the data web, and provides a SPARQL end-point

<sup>3</sup> <http://motools.svn.sourceforge.net/viewvc/motools/dbtune/lmapper/>

<sup>4</sup> <http://sourceforge.net/projects/motools/>

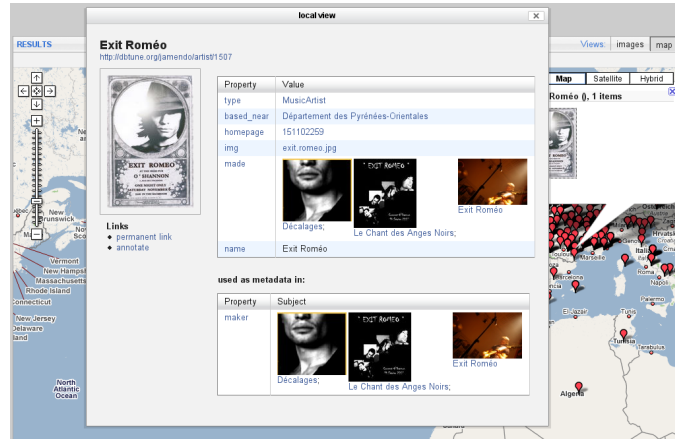


Figure 1: Slashfacet working on top of the aggregation generated by GNARQL. Here, we plot the artists in our collection on a map and select a particular location. A live demo is available at <http://dbtune.org/facet-demo/>

on top of the aggregated data. GNARQL then generates a tailored database, describing the user’s music collection. The corresponding end-point is able to answer queries such as “Create me a playlist of married hip-hop artists in my collection who featured in a particular BBC brand, ordered by crime rates in their city”. We also plugged Slashfacet [Hildebrand et al., 2006] on top of GNARQL in order to provide new ways for exploring a music collection. The example at <http://dbtune.org/facet-demo/> uses as an input a collection of Creative Commons tracks. Clicking on “MusicArtist” and on “map” plots all the artists in the collection on a map. The collection can then be browsed by selecting a particular location, as depicted in fig. 1.

## References

- [Bizer and Cyganiak, 2006] Bizer, C. and Cyganiak, R. (2006). D2R server publishing relational databases on the semantic web. In *Proceedings of the 5th International Semantic Web conference*.
- [Hildebrand et al., 2006] Hildebrand, M., van Ossenbruggen, J., and Hardman, L. (2006). *The Semantic Web - ISWC 2006*, volume 4273/2006 of *Lecture Notes in Computer Science*, chapter /facet: A Browser for Heterogeneous Semantic Web Repositories, pages 272–285. Springer Berlin / Heidelberg.
- [Raimond et al., 2007] Raimond, Y., Abdallah, S., Sandler, M., and Giasson, F. (2007). The music ontology. In *Proceedings of the International Conference on Music Information Retrieval*, pages 417–422.
- [Raimond et al., 2008] Raimond, Y., Sutton, C., and Sandler, M. (2008). Automatic interlinking of music datasets on the semantic web. In *Proceedings of the Linked Data on the Web workshop, collocated with the World-Wide-Web Conference*.
- [Wielemaker et al., 2008] Wielemaker, J., Huang, Z., and Meij, L. V. D. (2008). SWI-Prolog and the web. *Theory and Practice of Logic Programming*, 8(3):363–392.

## **Linked Movie Data Base**

**Oktie Hassanzadeh**

(University of Toronto, Toronto, Canada  
oktie@cs.toronto.edu)

**Mariano Consens**

(University of Toronto, Toronto, Canada  
consens@cs.toronto.edu)

**Abstract:** The Linked Movie Data Base (LinkedMDB) project provides a demonstration of the first open linked movie dataset connecting several major existing (and very popular) web resources about movies. The database exposed by LinkedMDB contains hundreds of thousands of RDF triples with tens of thousands of RDF links to existing web data sources (that are part of the growing Linking Open Data cloud), as well as to popular movie-related web pages (such as IMDb). LinkedMDB showcases the capabilities of a novel class of tool, Open Data Dataset Linker (ODDLinker) that facilitates the task of creating and maintaining large quantities of high quality links among existing datasets. ODDLinker employs state-of-the-art approximate join techniques for finding links between different data sources, and also generates additional RDF metadata about the quality of the links and the techniques used for deriving them.

**Keywords:** Semantic Web, linked data, movie, film database, RDF, equivalence mining

**Categories:** H.3.1, H.3.2, H.3.3, H.3.7, H.5.1

### **1 Introduction**

Movies are highly popular on the Web, and yet they are recognized as missing from the LOD datasets (listed under the “Nice to have on the Web of Data”). We developed the “Linked Movie Data Base” in order to:

- Provide a high quality source of RDF data (LinkedMDB.org) about movies. This data source appeals to a wide audience, enabling further demonstrations of the LOD capabilities.
- Demonstrate the value of a novel tool under development (ODDLinker) to facilitate high-volume and dense interlinking of RDF datasets.

In this report, we present an overview of the movie data triplification effort showcased in LinkedMDB.org (the demo website, the movie data sources and the links), and mention the tool and methodology employed in creating it.

### **2 Triplification of Movie Data**

#### **2.1 The LinkedMDB Website**

The online demo website is available at [www.linkedmdb.org](http://www.linkedmdb.org). It relies on the D2R Server to publish RDF data about movies and the links created. Our database

currently contains information about over 100,000 entities including movies, actors, movie characters, directors, producers, editors, writers, music composers and soundtracks, movie ratings and festivals from the movie data sources described in the next subsection. The database also contains an additional 50,000 links to the LOD datasets, plus over 250,000 links to movie webpages.

We expect the number of interlinks to significantly increase over the next few weeks (up-to-date statistics can be found at the LinkedMDB website). There is already a remarkably high ratio of interlinks to triples in LinkedMDB. In contrast, the current state of the LOD datasets has over two billion triples with three orders of magnitude less interlinks among them (only millions of interlinks).

## 2.2 Web Movie Data Sources

There are several sources of information about movies on the Web of documents. IMDb ([www.imdb.com](http://www.imdb.com)) is the biggest database of movies on the Web, and while it is downloadable, there are copyright restrictions that prevent re-publishing it online. FreeBase ([www.freebase.com](http://www.freebase.com)) is an open, shared database of the world's knowledge, with the "film" category having more than 38,000 movies and thousands of other data items related to movies. DBpedia ([www.DBpedia.org](http://www.DBpedia.org)) contains information about 36,000 movies with corresponding Wikipedia entries. OMDb ([www.omdb.org](http://www.omdb.org)) is another open data source of movies that contains information about 9,000 movies. Stanford Movie Database ([infolab.stanford.edu/pub/movies](http://infolab.stanford.edu/pub/movies)) is another public database of movie information. There are many other movie websites, such as RottenTomatoes.com, that do not make their data available for public use.

We currently use the FreeBase data download (under Creative Commons Attribution Licensing) as the basis for our interlinking, and information from the additional datasets can easily be added to the existing base as they become interlinked.

## 2.3 Interlinking Movies to Datasets in the LOD Cloud and Other Websites

Our database is linked to several datasets in the LOD cloud as well as to popular websites. We create owl:SameAs and rdfs:SeeAlso links to DBpedia, Geonames, YAGO, FlickrWrapp, lingvoj, and other LOD data sources. We also include foaf:page links to the IMDb, RottenTomatoes.com, FreeBase and OMDb websites. LinkedMDB interlinks add value to all the existing datasets: while there may be a large amount of overlap in the data (e.g., Wikipedia movie entries that appear in DBpedia also appear in FreeBase), the datasets evolve independently and contain data that is unique to them.

## 3 ODDLInker: a Toolset for Linking Open Data

A highlight of this project is the use of a tool under development, ODDLInker, supporting state-of-the-art approximate join and link mining techniques for interlinking data sources. ODDLInker enables an administrator to setup tens of thousands of links in a matter of hours. The tool also helps maintaining existing links and to incrementally add new links among the datasets of interest. The ODDLInker administrator can select the type of interlink to create (owl:SameAs, rdfs:SeeAlso,

foaf:page, or any other user-specified predicate) based on different criteria, such as the strength of the linkage. ODDLInker maintains a database (available as RDF) of the interlinks generated, including metadata about the linkage methodology and its quality. In the future, we plan to incorporate user feedback regarding interlinks to further enhance the quality of the links. A description of the methodology used (including assessments of the quality of the interlinks generated) can be found at the LinkedMDB website).

## **4 Summary**

LinkedMDB.org currently provides access to several hundred thousands of triples, including tens of thousands of high-quality interlinks to other LOD project data sources (such as DBpedia, YAGO, FlickrWrapp and Geonames), and over a quarter billion foaf:page links to IMDb, RottenTomatoes and FreeBase. LinkedMDB can leverage the power of ODDLInker, the novel tool developed to create its links, to quickly interlink to additional sources such as RDFBookMashup (to interlink to books related to the movies), Musicbrainz (to link to data about movie soundtracks) and can also link to Revyu.com (for movie reviews). By developing a tool for automatic linking and for tracking metadata about the quality of the links, we hope to have helped the LOD community to considerably increase the quantity and the quality of interlinks.

## **Semantic Web Pipes Demo**

**Danh Le Phuoc**

(Digital Enterprise Research Institute, NUIG, Ireland  
danh.lephuoc@deri.org)

**Abstract:** Making effective use of RDF data published online (such as RDF DBLP, DBpedia, FOAF profiles) is, in practice, all but straightforward. Data might be fragmented or incomplete so that multiple sources need to be joined, different identifiers (URIs) are usually employed for the same entities, ontologies need alignment, certain information might need to be "patched", etc. The only approach available to these problems so far has been custom programming such transformations for the specific task to be performed in a Semantic Web application. In this demo, we illustrate a paradigm for creating and reusing such transformation in an easy, visual web-based and collaborative way: Semantic Web Pipes.

**Keywords:** RDF mashup, semantic web, software pipe

### **1 Introduction**

There is an increasing amount of RDF data exposed by current Web applications such as DBLP[1], DBpedia [2], blogs, wikis, forums, etc that expose their content as e.g. SIOC[3], FOAF[4] data through SIOC exporters, Triplify[5] plugins. Furthermore, these RDF data are offered in a variety of formats, such as interlinked RDF/XML files, RDF statements embedded in HTML/XML pages, etc[6].

Unfortunately, there is no clear and established model on how to use this amount of information coming from many diverse sources. In general, such data is unlikely to be directly injected into an end application for multiple reasons. The common approach to these problems so far has been the custom programming of such transformations for the specific task to be performed in a Semantic Web application. In this demo, we present a paradigm for creating and reusing such transformations in a simple way: a Web based Software Pipeline for the Semantic Web.

This metaphor has been inspired by Yahoo Web Pipes[7], which allows to implement customized services and information streams by processing and combining Web sources (usually RSS feeds) using a cascade of simple operators. Since Web pipes are themselves HTTP retrievable data sources, they can be reused and combined to form other pipes. Also, Web pipes are "live": they are computed on demand at each HTTP invocation, thus reflect the current status of the original data sources.

Unfortunately, Yahoo Pipes are engineered to operate using fundamentally the RSS paradigm (item lists) which does not map well to the graph based data model of RDF. For this purpose we create the Semantic Web Pipes (SWP)[8], an open source application with a strong emphasis on Semantic Web data and standards. SWP offers specialized operators that can be arranged in a graphical web editor to perform the most important data aggregation and transformation tasks without requiring programming skills or sophisticated knowledge about Semantic Web formats. This enables developers as well as end users to create, share and re-use semantic mashups that are based on the most current data available on the (semantic) web.



When a pipe is invoked (by a HTTP GET request with the pipe URL), the external sources of the pipe are fetched dynamically. Therefore, the output of a pipe always reflects the data currently available on the web. Since SWP are themselves HTTP retrievable data sources, they can be reused in other pipes and combined to form pipes of increasing complexity....

## 2 Semantic Web Pipes engine and features

While it would be possible to implement pipe descriptions themselves in RDF, our current ad hoc XML language is more terse and legible. If an RDF representation will be later needed, it will be possible to obtain it via GRDDL[9]. The executable pipe XML syntaxes are stored in a database. When they are invoked, the execution engine fetches data from remote sources into an in-memory triple store, and then executes the tree of operators. Each operator has its own triple buffer where it can load data from input operators, execute the SPARQL[10] query or materialize implicit triples of RDF closures. Since each operator is implemented as a processing unit, it can be scheduled in parallel and distributed processing structure to improve scalability.

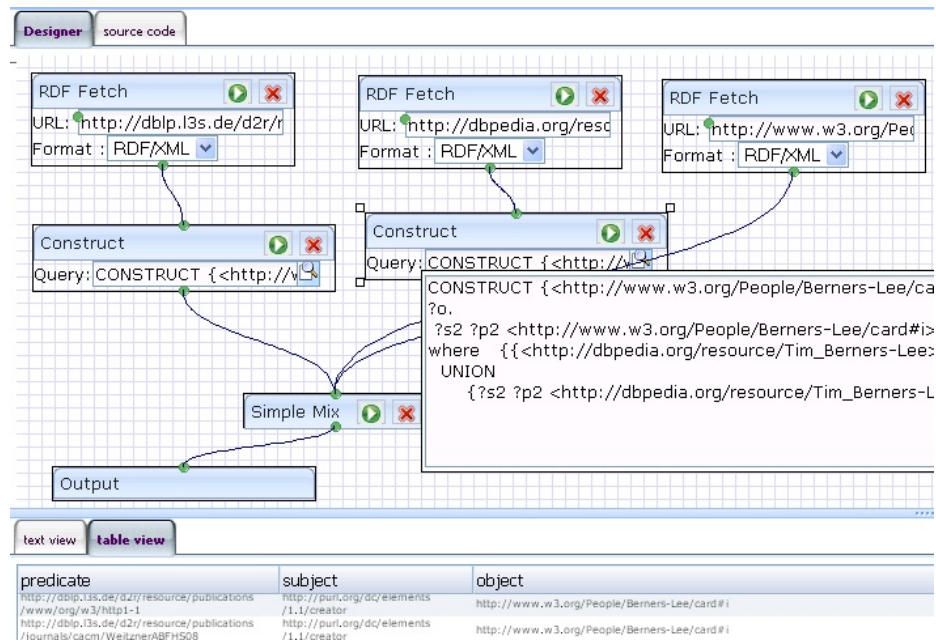


Figure 1. Semantic Web Pipes Editor

In this version, SWP is supporting 4 types of operators. The first type includes the operators for fetching data from RDF, SPARQL-XML result, XML, HTML (with embedded RDFa and microformats). The second type is for issuing SPARQL query like SELECT, CONSTRUCT. Processing operators like mixing, patching and reasoning represent another type. The last type of operators are input operators like

parameter, URL builders. All the specifications of these operators can be found at <http://pipes.deri.org/documentation.html>.

A graphical editor (figure 1) with a visual and user friendly user interface has been implemented based on the framework ZK. We are using ZK as integrating framework to control several Javascript based GUI libraries by Java which is used for server-side manipulation of RDF data. The drag and drop editor lets users view and construct their pipeline, inspecting the data at each step in the process. Furthermore, any existing pipe can be copied and integrated into the pipe that is edited by the user.

Thanks to HTTP content negotiation, humans can use each Semantic Web Pipe directly through a convenient web user interface. The format of a pipe output depends on the HTTP header sent in the request. For example, RDF-enabled software can retrieve machine-readable RDF data, while users are presented a rich graphical user interface. Therefore, along with supporting various common serialized RDF formats (RDFXML[11],N3[12],Turtle[13],etc), we also support JSON formats for pipe output which is suitable for light-weight data-level web mashup. For example, our pipe's preview interface has been created from Javascript-based Exhibit facet browser consuming Exhibit[14] JSON data directly from pipes.

### 3 Showcases

This section will give some typical showcases which were created on [pipes.deri.org](http://pipes.deri.org) as pipes by using graphical editor. Firstly, we present a simple pipe that shows how to remixing data from various sources. Data about Tim Berners-Lee is available on various sources on the Semantic Web, e.g. his FOAF file, his RDF record of the DBLP scientific publication listing service and from DBpedia. This data can not simply be merged directly as all three sources use different identifiers for Tim. Since we prefer using his self-chosen identifier from Tim's FOAF file, we will create a pipe as an aggregation of components that will convert the identifiers used in DBLP and DBpedia. This is performed by using the Construct-operator with a SPARQL query (see TBLonTheSW pipes at [8]). The whole showcase is then easily addressed by the pipe shown in Figure 1: URIs are normalized via the CONSTRUCT-operators and then joined with Tim's FOAF file.

Inspired by the Yahoo Pipes use cases of cross-search engine feed aggregation, Semantic Web Pipes should enable us to fetch data from parametric sources as well as extracting RDF statements from non-native RDF formats. For example, we want to collect facts about London that are published as RDF statements on heterogeneous sources like news, web blogs, geo-based web services, etc. These RDF statements stated in HTML-based documents as RDFa, microformats or are provided by web services calls which support RDF format outputs. On top of that, we do not know beforehand which are URLs containing such RDF data, so we have to employ semantic web search engines like Sindice to search for such URLs. Firstly, we ask for RDF URIs indexed in Sindice[15], then use the SELECT operator to filter the interested URIs. After that, we use the FOR operator to repeatedly fetch RDF data from those URIs. Furthermore, we can get geo information from Geonames [16]. Finally, we use the Simplemix operator to mix these RDF data to create a parametric

pipe (c.f. the Cityfacts pipe at [8]) which allows users to enter a city name for searching RDF data about that city.

Examples for all operators and other showcases can be found at <http://pipes.deri.org>.

## Acknowledgement

The work presented in this paper was supported by the Lion project supported by Science Foundation Ireland under grant no. SFI/02/CE1/I131.

## References

- [1] DBLP Bibliography - Home Page. <http://www.informatik.uni-trier.de/~ley/db/>
- [2]. Sören Auer, Christian Bizer, Georgi Kobilarov, Jens Lehmann, Richard Cyganiak, and Zachary Ives. Dbpedia: A nucleus for a web of open data. In 6th Int.l Semantic Web Conf., Busan, Korea, November 2007.
- [3]. J. G. Breslin, A. Harth, U. Bojars, S. Decker, "Towards Semantically-Interlinked Online Communities", Proceedings of the 2nd European Semantic Web Conference (ESWC '05), LNCS vol. 3532, pp. 500-514, Heraklion, Greece, 2005. <http://rdfs.org/sioc/spec/>
- [4]. D. Brickley and L. Miller. FOAF Vocabulary Spec., July 2005. <http://xmlns.com/foaf/0.1/>.
- [5]. Triplify <http://triplify.org/>.
- [6]. Custom Rdf Dialects <http://esw.w3.org/topic/CustomRdfDialects>.
- [7] Yahoo Pipes. <http://pipes.yahoo.com/>.
- [8]. C. Morbidoni, A. Polleres, G. Tummarello, and D. Le Phuoc. SemanticWeb Pipes. Technical Report, see <http://pipes.deri.org/>. Nov. 2007.
- [9]. D. Conolly (ed.). Gleaning Resource Descriptions from Dialects of Languages (GRDDL), July 2007. W3C Prop. Rec., <http://www.w3.org/TR/grddl/>.
- [10] E. Prud'hommeaux, A. Seaborne(eds.). SPARQL Query Language for RDF.
- [11] P. Hayes. RDF semantics, Feb.2004. W3C Rec.
- [12] T. Berners-Lee. Notation 3, since 1998. <http://www.w3.org/DesignIssues/Notation3.html>.
123. D. Beckett. Turtle - Terse RDF Triple Language, Apr. 2006. <http://www.dajobe.org/2004/01/turtle/>.
- [14] Huynh, D. F., Karger, D. R., and Miller, R. C. 2007. Exhibit: lightweight structured data publishing. In Proceedings of the 16th international Conference on World Wide Web (Banff, Alberta, Canada, May 08 - 12, 2007). WWW '07. ACM, New York, NY, 737-746.
- [15] GeoNames. <http://www.geonames.org/>.
- [16] Sindice – The Semantic Web Index. <http://www.sindice.com/>.

# **Triplification of the Open-Source Online Shop System osCommerce**

**Elias Theodorou**

(University of Leipzig, Leipzig, Germany  
mai01jy@studserv.uni-leipzig.de)

**Abstract:** This article describes the integration of osCommerce with a Triplify configuration based on an commerce vocabulary. osCommerce is the leading open source online shop system with an active user and developer community. We will briefly introduce osCommerce and how Triplify can be used with it. We also describe the used vocabularies and the resulting Triplify commerce vocabulary.

**Keywords:** Semantic Web, vocabulary, FOAF, DCMI, SIOC, SKOS, e-commerce, Triplify

**Categories:** H.2.1, H.2.2, H.2.7, H.3.2, H.3.5, H.5.1, H.5.3

## **1 Introduction**

The osCommerce community<sup>1</sup> was started in March 2000 and since then has fuelled 13.895 online shops around the world. osCommerce offers a wide range of out-of-the-box features. That means it is not requiring any additional installations, plug-ins, expansion packs or products. In a very short time, simple and without any licensing fees (Open Source) an online shop can be set up based on osCommerce.

Since Open Source software enables to freely use, study, share, participate and join an open source software community we aim at contributing to osCommerce's community by integrating Triplify into the osCommerce software. What does Triplify offer? Triplify is a small plug-in that reveals the semantic structures of web applications (such as osCommerce) by converting their database content into semantic formats. The osCommerce-Triplify integration defines a dictionary of named properties and classes using W3C's RDF technology.

## **2 osCommerce**

osCommerce allows Web shop owners to setup, run, and maintain online stores with minimum effort and with no costs, fees, or limitations involved. The big advantage of this software when compared to commercial solutions is the active community where members help each other and participate in development issues reflecting upon the current state of the project.

The osCommunity Support Forums have at the moment more than 181,000 registered members which are ready for answering questions. osCommerce is based on the PHP web scripting language, the Apache web server and the MySQL database

---

<sup>1</sup> <http://www.oscommerce.org>

server. Without restrictions or special requirements, osCommerce can be installed on any PHP (PHP $\geq$ 3) web server, on any environment that PHP and MySQL supports, which includes Linux, Solaris, BSD and Microsoft Windows environments.

After logging in the store admin (cf. Figure 1), a box of links guides to individual sections for modifying and building up a Web store.

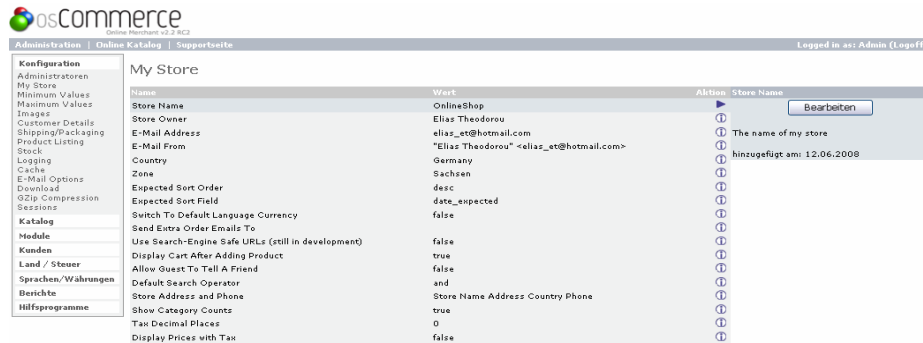


Figure 1: Administrator interface

### 3 Integration of Triplify

We have seen in the past section, how simple it is to create an online shop quickly. In order to make potential customers aware of our products and offerings we need ways to disseminate product and pricing information. The structure and semantics encoded in the relational databases behind osCommerce based online shops are not accessible to search engines. Here we employ technologies of Semantic Web. Triplify<sup>2</sup> is able to “semantify” the structured data behind osCommerce.

In order to integrate osCommerce and Triplify we created a adopted version of osCommerce, which contains a preconfigured Triplify installation. The osCommerce Triplify configuration contains SQL queries which select the information to be exposed from the relational DB backend. We also integrated a Triplify-specific installation step into the osCommerce installer. The configuration file for osCommerce can be found at <http://triplify.org/Configuration/osCommerce>, the patched osCommerce version is also downloadable from this address.

### 4 Vocabularies

Last but not least we want to describe briefly the vocabulary, which was used in the configuration of Triplify. We were reusing classes and properties from the FOAF, SIOC, SKOS, DCMI, eClassOWL and vCard vocabularies. For certain data (such as products, prices etc.) adequate, descriptive classes and properties did not exist. We defined such properties and classes in the Triplify commerce vocabulary, which is available at: <http://triplify.org/vocabulary/oscommerce>.

<sup>2</sup> <http://triplify.org>

## Interlinking Multimedia Data

**Michael Hausenblas and Wolfgang Halb**

(Institute of Information Systems & Information Management,  
JOANNEUM RESEARCH, Austria  
firstname.lastname@joanneum.at)

**Abstract:** “Catch Me If You Can” (CaMiCatzee) is a multimedia data interlinking concept demonstrator. The goal is to show how images from flickr can be interlinked with other data, such as person-related (FOAF) data, locations, and related topics. We report on CaMiCatzee’s architecture and introduce a first version of the demonstrator, available at <http://sw.joanneum.at/CaMiCatzee/>.

**Key Words:** linked data, multimedia, user contributed interlinking, FOAF, flickr

**Category:** H.5.1

### 1 Motivation

The popularity of social media sites (such as flickr) has led to an overwhelming amount of user contributed multimedia content. Although features for tagging and commenting are available, the outcome is mainly shallow metadata. On the other hand, current linked data sets<sup>1</sup> basically address textual resources. Further, the interlinking is usually done automatically, based on string matching algorithms. However, multimedia resources have been neglected so far<sup>2</sup>. When referring to multimedia resources interlinking, we do not talk about global meta-data such as the creator or a title; we rather focus on a fine-grained interlinking, for example, objects in a picture. We envision to extend the User Contributed Interlinking (UCI) [Hausenblas et al. 08a, Hausenblas et al. 08b, Halb et al. 08] to multimedia assets. Clearly, the advantage is having high-quality semantic links from a multimedia asset to other data, hence allowing to connect to the linked datasets.

### 2 CaMiCatzee

In flickr it is possible to annotate parts of a picture using so called “notes”. As the primary domain, we chose people depictions. Typically, flickr notes contain a string stating, e.g., “person X is depicted in this picture”. However, there is no straight-forward way to relate this information with other data, such as FOAF data, locations, and contextual information (conference, holiday, etc.). This is where we step in: we apply the UCI principle by harnessing the fine-grained annotation capabilities of flickr in order to let people semantically annotate pictures.

<sup>1</sup> <http://richard.cyganiak.de/2007/10/lod/>

<sup>2</sup> <http://community.linkeddata.org/MediaWiki/index.php?InterlinkingMultimedia>

In the initial release of the multimedia interlinking demonstrator “Catch Me If You Can” (CaMiCatzee) the query for depictions can be performed using a person’s FOAF document, a person’s URI or simply a name (note that in the latter two cases a semantic indexer<sup>3</sup> is used to propose matching FOAF documents). Subsequently flickr is queried for matching annotations (on the person URI extracted from the FOAF document) yielding all pictures containing the desired person. Additionally, in the “full report”, the flickr tags of a picture are evaluated and used as a base for introducing `rdfs:seeAlso` links; this overview is offered in XHTML+RDFa<sup>4</sup>, allowing consumption by both humans and machines. Fig. 1 depicts the system’s architecture, showing the CaMiCatzee server as well as the client.

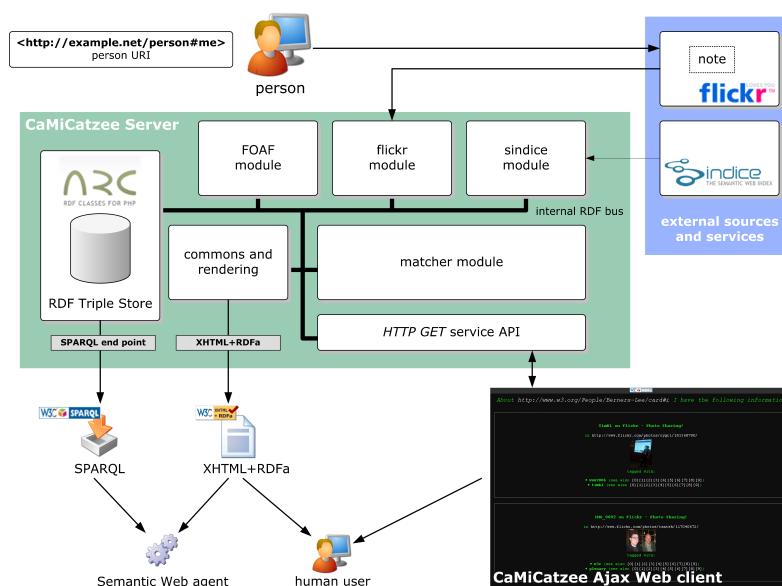


Figure 1: CaMiCatzee’s architecture.

## References

- [Halb et al. 08] W. Halb, Y. Raimond, and M. Hausenblas. Building Linked Data For Both Humans and Machines. In *WWW 2008 Workshop: Linked Data on the Web (LDOW2008)*, Beijing, China, 2008.
- [Hausenblas et al. 08a] M. Hausenblas and W. Halb. Interlinking of Resources with Semantics (Poster). *5th European Semantic Web Conference (ESWC2008)*, Tenerife, Spain, 2008.
- [Hausenblas et al. 08b] M. Hausenblas, W. Halb, and Y. Raimond. Scripting User Contributed Interlinking. In *4th Workshop on Scripting for the Semantic Web (SFSW08)*, Tenerife, Spain, 2008.

<sup>3</sup> <http://sindice.com>

<sup>4</sup> <http://www.w3.org/TR/xhtml1-rdfa-primer/>

# **Integrating *triplify* into the Django Web Application Framework and Discover Some Math**

**Martin Czygan**  
University of Leipzig  
martin.czygan@gmail.com

**Abstract:** We demonstrate a simple way to integrate parts of the original triplify script into the Django web application framework. We discuss a small demo application, which overlays the content of a popular site in the mathematics domain with some semantic structure.

**Key Words:** metadata, triplify, web application framework, python  
**Categories:** H.0, H.4

## **1 Introduction**

Django [Holovaty et al., 2005] is a popular web application framework for the python [van Rossum et al., 1991] programming language. We present and discuss a simple approach to integrate triplify [Auer, 2008] into the Django ecosystem. In the second part we describe shortly a demo application [Czygan, 2008], which takes all PlanetMath [PM] articles and adds some navigational features to them. There, triplify is used to export aspects of the data.

## **2 Django Triplify Application**

Django organizes its program logic into applications. Therefore we will discuss our triplify application, which can be used with different models. Since Django implements the complete web application stack, it is bundled with an object-relational mapper. With that we can avoid pure SQL queries.

To make more sense of the following explanations of the source code, we now give a quick overview of the target application (for a detailed explanation please see the next section): Saturn (as we might call the demo application for now) exports triples, which describe the relation between mathematical topics, such as a set and the concept of countable. In particular it relates one topic to another topic, which the application classified as dependent - in a paraphrased sense: "If you want to understand the concept of a Set you might need to know what countable means first." The concepts are stored in the database as Entries (Entry), the dependent topics are layed down in a self-referential many-to-many relationship, which are accessible via links.

To propel the discussion, we will discuss a source code snippet:

```
v = Vocab()  
t = Triples(
```



```

    '%s' % entry.get_absolute_url(),
    v.dc['requires'],
    [ '%s' % l.get_absolute_url() for l in
      entry.links.all() ]
  )

```

This snippet initializes the vocabulary and generates triples, which express the relation (or dependency) of one topic (as URL) to another topic (as URL) via the requires term of the Dublic Core vocabulary (which might be not the ideal vocabulary term). Exports for different formats such as RDF/XML, nt or n3 are available. The list of vocabularies is easily extensible.

### 3 Saturn Demo Application

The demo application (Saturn) organizes mathematical articles from the PlanetMath.org site according to the importance of an article inside the mathematical universe of the about 7000 articles of PlanetMath.org. It is a vague and in this case very discussable ranking, in particular: PageRank [Brin and Page, 1998]. The application shows all related topics on a sidebar according to its pagerank class (which we call chamber<sup>1</sup> in this application). Each entry/article with its dependencies can be exported in the form of triples. As an experimental feature we parse a PDF file (given the URL), which may point to some scientific document. We compare the used terms in the document with the 7000 terms in the applications database and list the matching terms sorted by their pagerank class (or chamber) - therefore giving a quick impression what this paper may be about and how hard it might be to understand.

Beside all this serious considerations we want to emphasize the fact that this application is first only a demo application and second that it was written with a smile, considering the severe IR-task which underlies this approach.<sup>2</sup>

### References

[Auer, 2008] Auer, S.: "Triplify"; 2008. <http://www.triplify.org/>

[Brin and Page, 1998] Brin, S., Larry Page, L.: "The Anatomy of a Large-Scale Hypertextual Web Search Engine"; Computer Science Department. Stanford University. 1998. <http://infolab.stanford.edu/~backrub/google.html>

[Czygan, 2008] Czygan, M.: "Saturn: Demo Application"; 2008. <http://pcai042.informatik.uni-leipzig.de:9103/>

[Holovaty et al., 2005] Holovaty, A. et al.: "Django: The web application for perfectionists"; 2005. <http://www.djangoproject.com/>

<sup>1</sup> The application is inspired by the 1979 Kung-Fu movie Shao Lin san shi liu fang which is known to the western world as The 36th Chamber of Shaolin . In short: to master a particular martial art style, one must master all the 36 chambers.

<sup>2</sup> which might be formulated compactly as the generation of dependency trees for certain difficulty levels in scientific publications. . .

[PM] The PlanetMath.org contributors: "PlanetMath.org"; <http://www.PlanetMath.org/>

[van Rossum et al., 1991] van Rossum, G., et al.: "Python Programming Language"; 1991.  
<http://www.python.org/>

## **Showcases of light-weight RDF syndication in Joomla**

**Danh Le Phuoc**

(Digital Enterprise Research Institute, NUIG, Ireland  
danh.lephuoc@deri.org)

**Nur Aini Rakhmawati**

(Sepuluh Nopember Institute of Technology, Surabaya, Indonesia  
iin@its.ac.id)

**Abstract:** This demo showcases light-weight RDF syndication in the Content Management System Joomla. It will show how to syndicate RDF data in small websites like CMS, blogs, etc without having to install any triple storage and employ a dedicated triple processing engine. This RDF syndication will be as easy to embed into a website as a widget as embedding Google Maps.

**Keywords:** RDF syndication, RDFizing, CMS.

### **1 Introduction**

Semantic Web is becoming more real, there is increasing number of web services, applications supporting RDF data. Moreover, there is a variety of valuable semantic data sources like those from the LinkingOpenData cloud [1]. However, it normally takes a lot of efforts to implement a web site supporting Semantic Web data. Specially, it is unlikely to take Semantic Web into account when implementing a small website such as a CMS, blog, forum due to the complexity and cost of the implementation. Hence, inspiring from RSS syndication which is very popular in any website, we implement this demo to give some showcases of RDF syndication. In order to encourage web developers to expose, integrate, syndicate RDF data, this demo will show how the light-weight exposing and integration of data with the Joomla! CMS platform works without installing any additional database and triple processing library.

In this demo, we will adopt the Triplify[2] script as a Joomla! component to expose data as RDF/JSON feed which can be crawled by Semantic Search Engine (Sindice, SWSE, SWOOGLE) or syndicated by using a scripting language. Similar to a RSS feed reader, we customize Exhibit as an RDF/JSON feed reader. This feed reader plays the role of a facet browser which can be embedded to any HTML page as a widget. This feed reader can read RDF data encoded in the Exhibit[3] JSON format from the following types of resources: Joomla! exposed RDF data, Semantic Web Pipes and Sindice's SIOC sphere search results.

## 2 Architecture

Our showcases will be implemented in the architecture of Figure 1. In this architecture, data from content publishing platforms such as Wordpress, Drupal, Joomla! can be exposed in the RDF format and then processed by search engines as Sindice[4] or remixing servers like Semantic Web Pipes[5]. Thus, such RDF data processing (i.e. querying, crawling, indexing, reasoning, remixing, etc) is done on those remote servers. On the other hand, small plugins such as Triplify or SIOC exporters will reveal semantic structures encoded in relational database and make their underlying content available in RDF or JSON.

The processed data or exposed RDF data can be reached by RDF syndication script like Javascript in JSON format. This syndication script will be embedded in the *cross-site syndication component* of Joomla!, which plays the role of an RDF feed aggregator and reader. The syndication script which has been used in this architecture is Exhibit. It can consume Exhibit JSON data from multiple and distributed resources and represented these in a facet based browser. These Exhibit JSON data sources will be loaded into a graph-based database which can be queried by simple graph pattern expressions. Because all data querying and rendering will be done in the browser, the cross-sites syndication component is only responsible for configuring data sources and output views (i.e. lenses) in order to provide appropriate syndication to users.

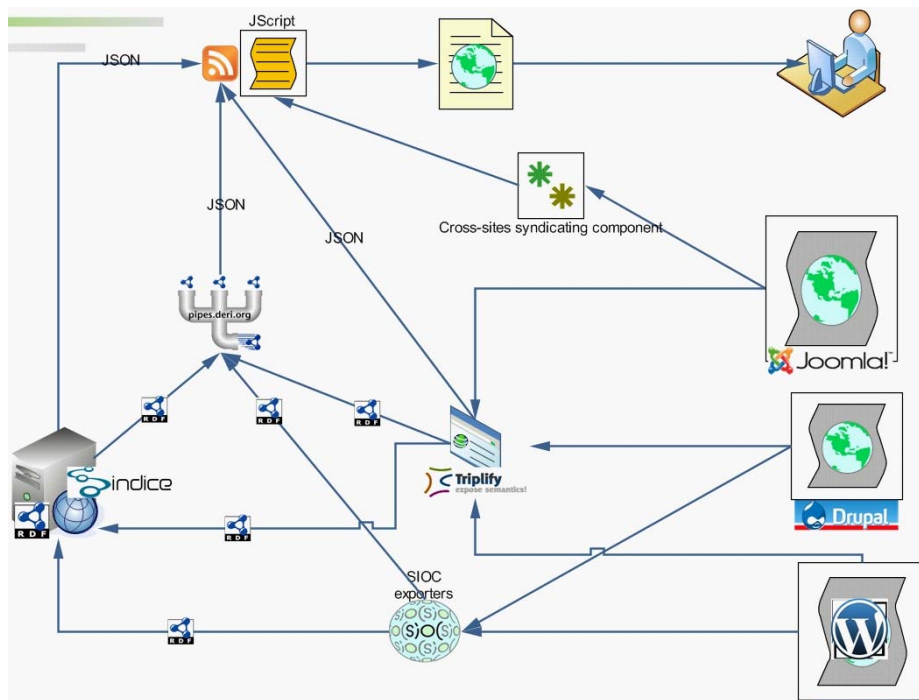


Figure 1. Architecture of light-weight RDF syndication with Joomla!

### 3 Showcases

Our showcases are hosted at <http://swm.deri.org/jsyndication/>. This website, called JSyndication, is an example Joomla! site where our Joomla! syndication component is installed. This component will expose content from this site as RDF and JSON. The exposed content can be viewed in the rendered HTML page. Similarly, we installed our component to two other sites and feed the content of these remote sites to the JSyndication (*Joomla RDF feeds*). We also added a page called “*Favourite RDF feeds*” which provides RDF data sources in Exhibit JSON format.

Users can add feeds to their profiles as personal information by creating Semantic Web Pipes remixing Foaf files and other information from other resources. For example, the author of a news article, named Tim Berner-Lee, can create some Pipes which remix his Foaf file, publications from DBpedia, DBLP, etc. and then add these Pipes’ URLs to his profile. Thus, when clicking on an author, we will see the “personal information” link which redirects us to author’s syndicated profile.

Another link we can see when clicking an author is “recent Posts”. This link will forward us to the syndication which feeds all of indexed posts of an author from Sindice. These posts were crawled by Sindice from submitted links from Web sites where the author published his posts.

### Acknowledgement

The work presented in this paper was supported by the Lion project supported by Science Foundation Ireland under grant no. SFI/02/CE1/I131.

### References

- [1] Linking Open Data .<http://esw.w3.org/topic/SweoIG/TaskForces/CommunityProjects/LinkingOpenData>
- [2]. Triplify <http://triplify.org/>.
- [3] Huynh, D. F., Karger, D. R., and Miller, R. C. 2007. Exhibit: lightweight structured data publishing. In Proceedings of the 16th international Conference on World Wide Web (Banff, Alberta, Canada, May 08 - 12, 2007). WWW '07. ACM, New York, NY, 737-746.
- [4] Sindice – The Semantic Web Index. <http://www.sindice.com/>.
- [5]. C. Morbidoni, A. Polleres, G. Tummarello, and D. Le Phuoc. SemanticWeb Pipes. Technical Report, see <http://pipes.deri.org/>. Nov. 2007.

# **Automatic Generation of a Content Management System from an OWL Ontology and RDF Import and Export**

**Alastair Burt**

(German Research Center for Artificial Intelligence, Saarbrücken, Germany  
Alastair.Burt@dfki.de)

**Brigitte Jörg**

(German Research Center for Artificial Intelligence, Saarbrücken, Germany  
Brigitte.Joerg@dfki.de)

**Abstract:** We present a system, OWL2AT (OWL to Archetypes), that will automatically generate a content management system (CMS) from an OWL-Lite ontology. The resulting CMS will automatically import and export information about the content in RDF format that corresponds exactly to the original ontology. The use of OWL in such a closely integrated manner with a CMS greatly simplifies a system's lifecycle management: it enables cooperative design in a community with OWL as the lingua franca, and ensures that data is exchanged within the community using a well understand format and semantics.

**Keywords:** RDF, OWL-Lite, ontology, content management system, automated setup, Zope, Plone, LT World, Semantic Web, Web portal

**Categories:** H.2.1, H.2.2, H.2.4, H.2.7, H.3.2, H.3.5, H.5.1, H.5.3

## **1 Introduction**

A Content Management System (CMS) has become a standard tool for setting up corporate websites and portals. There are a variety of proprietary and free software CMS systems that save development time and human resources during the initial launch of a web site and provide many tools necessary for the management of content once the site is running. This paper presents an extension for Plone (<http://plone.org/>), a popular free software CMS that is currently in use in more than 1,200 high profile sites. The approach described here is to use OWL and RDF in a novel way with a CMS to provide two powerful features not found in other systems: (1) Automatic generation of a turn-key CMS from an OWL-Lite ontology. (2) Automatic import and export of content from the CMS in an RDF format that fully corresponds to the original OWL ontology.

## **2 Related Work**

*Plone Ontology* is a way to use OWL ontologies to manage vocabularies (<http://plone.org/products/ploneontology>). In contrast, our approach makes more extensive use of OWL, employing it for schema generation. ArchGenXML

(<http://plone.org/products/archgenxml>) uses UML to specify Plone schemas but does not define an import and export format.

The ontology driven approach presented here has much in common with systems based on the model-driven architecture (MDA) concept: both address problems of lifecycle management, that is to say of deploying, integrating and managing applications, in a platform agnostic way. The MDA approach has the imprimatur of an industry standards' body [OMG 2003] and is being supported by tools for development environments. However, it lacks the specification of a formal semantics [Breu et. al. 2002], [Cranefield and Purvis 1999], which the semantic web can offer. Our approach combines elements of the life cycle management of MDA with formal semantics.

### 3 The OWL2AT System

#### 3.1 Automatic Schema Generation for a CMS from an OWL Ontology

Currently, content in Plone is typically specified in a schema via Archetypes (<http://plone.org/products/archetypes>), which operates one layer above Plone. This specification is normally written in the programming language Python. OWL2AT generates such Archetype schemas automatically from a declarative ontology in OWL-Lite.

#### 3.2 Automatic Import and Export of Content via RDF

The CMS generated with the above technique offers as a by-product a means to exchange data in a REST-based fashion. Data can be imported with an HTTP PUT call as RDF triples, and exported with an HTTP GET call as RDF triples. The semantics and format of these triples is exactly described by the original OWL ontology.

More information on the OWL2AT system can be found at the demo website: <http://www.lt-world.org/triplify/>. The code can be downloaded from <http://www.lt-world.org/triplify/code/owl2at.py>

The OWL2AT system is published under a GNU Public Licence (version 2 or later).

### 4 LT World Experience

The OWL2AT system was developed to handle a relaunch of the LT World portal; the central entry point for accessing information in the field of language technology. LT World (<http://www.lt-world.org>) is being prepared for the third generation release, to enable community-driven knowledge specification, maintenance and exchange over the web. From its beginning the portal has been tightly associated with its underlying ontology [Uszkoreit & Jörg 2003], [Jörg & Uszkoreit 2005]. The OWL2AT version of LT World is complex and still needs some finishing touches before going fully public. For this challenge, we therefore present a demonstration of the approach with a small extract of a university ontology, and refer interested users

to the LT World portal (<http://beta.lt-world.org>) to view results of the approach on a larger scale.

### Acknowledgements

The work has been funded by the German Federal Ministry of Education and Research (BMBF) within the extended Collate II project, in 2006.

### References

[Breu et al. 2002] Breu, R.; Grosu, R.; Huber, F.; Rumpe, B.; Schwerin, W.: Towards a Precise Semantics for Object-Oriented Modeling Techniques. Lecture Notes in Computer Science. Volume 1357, 1998.

[Cranefield, S.; Purvis, M.: UML as an ontology modelling language. In Proceedings of the Workshop on Intelligent Information Integration. 16<sup>th</sup>, International Joint Conference on Artificial Intelligence (IJCAI-99), 1999.

[Jörg & Uszkoreit 2005] Jörg, B.; Uszkoreit, H.: The Ontology-based Architecture of LT World, a comprehensive Web Information System for a Science and Technology Discipline. In: Leitbild Informationskompetenz: Positionen – Praxis – Perspektiven im europäischen Wissensmarkt. 27. Online Tagung der DGI. Frankfurt am Main, 23.-25.Mai, 2005.

[OMG 2003] Open Management Group: OMG Unified Modeling Language Specification. Version 1.5, March 2003. <http://www.omg.org/docs/formal/03-03-01.pdf>

[Uszkoreit & Joerg 2003] Uszkoreit, H., Jörg, B.: An Ontology-based Knowledge Portal for Language Technology. Enabler/Elsnet Workshop *International Roadmap for Language Resources*. Paris 2003.



## Author Index

### A

Auer, S., 1, 182, 190

### B

Bassem, M., 50  
Behrendt, W., 178  
Bessler, S., 67  
Blumauer, A., 102  
Bry, F., 166  
Burt, A., 211  
Bürger, T., 85

### C

Carstens, C., 25  
Chamberlain, J., 42  
Cherfi, H., 50  
Chiara, G., 134  
Coenen, T., 118  
Collard, M., 76  
Conesa, J., 126  
Consens, M., 194  
Czygan, M., 205

### D

Demarchez, M., 76  
Dieng-Kuntz, R., 50  
Dietzold, S., 182  
Dustdar, S., 102  
Dögl, D., 85

### E

Ebner, H., 34

### F

Faatz, A., 134

### G

Gabner, R., 67  
García-Barriocanal, E., 126  
Gómez-pérez, A., 142

Grebner, O., 34

Gruber, A., 85

Guss, J., 134

### H

Halb, W., 9, 203  
Hassanzadeh, O., 194  
Hausenblas, M., 9, 203  
Heath, T., 6, 9  
Heese, R., 150  
Heim, P., 182  
Hepp, M., 118  
Hoang Thi, A., 58  
Holzapfel, M., 178  
Hrgovcic, V., 170

### J

Joerg, B., 211

### K

Kefi-Khelif, L., 76  
Kemper, P., 4  
Khelif, K., 50  
Kohn, A., 166  
Koller, A., 102  
Krishnaswamy, S., 93  
Kruschwitz, U., 42  
Kump, B., 134

### L

Le Phuoc, D., 197, 208  
Lieberman, H., 5  
Linder, M., 186  
Lindstaedt, S., 134  
Lohmann, S., 182  
Luczak-Rösch, M., 150  
Lukose, D., 7

### M

Manta, A., 166  
Mense, A., 186

Mochol, M., 158

## **N**

Nagypal, G., 17

Nguyen, T., 58

## **P**

Pammer, V., 134

Pellegrini, T., 1, 190

Peters, I., 110

Plößnig, M., 178

Poesio, M., 42

Praszl, W., 102

## **R**

Raimond, Y., 9, 191

Rakhmawati, N., 208

Riechert, T., 182

Rodriguez, M., 126

Rospoche, M., 134

## **S**

Schaffert, S., 1, 190

Serafini, L., 134

Shafiq, O., 85

Sicilia, M., 126

Steller, L., 93

Suárez-Figueroa, M., 142

## **T**

Theodorou, E., 201

Toma, I., 85

## **V**

Van Damme, C., 118

Vasko, M., 102

## **W**

Wahl, H., 186

Walter, A., 17

Weller, K., 110

Woitsch, R., 170

## **Y**

Yildirim, A., 174

## **Ü**

Üsküdarlı, S., 174

## **Z**

Zdun, U., 102

Zeiss, J., 67

Zhdanova, A., 67

Zinnen, A., 134