



NeOn: Lifecycle Support for Networked Ontologies

Integrated Project (IST-2005-027595)

Priority: IST-2004-2.4.7 — “Semantic-based knowledge and content systems”

D1.4.3 Cupboard—Supporting Ontology Reuse by Combining a Semantic Web Gateway, Ontology Registry and Open Ratings Systems

Deliverable Co-ordinator: Holger Lewen

Deliverable Co-ordinating Institution: Universität Karlsruhe (TH)

Other Authors: Mathieu d’Aquin (OU), Jérôme Eurenat (INRIA), Chan Le Duc (INRIA), Raúl Palma (UPM)

In this deliverable, we present the Cupboard system for ontology publishing, sharing and reuse. This system is intended to support both ontology engineers and ontology users/practitioners. For the developers of ontologies, it offers a complete infrastructure to host their ontologies in online ontology spaces, providing mechanisms to describe, manage and effectively exploit these ontologies (through APIs). Furthermore, these ontologies are then exposed to the community, providing users with a complete, friendly environment to find, assess and reuse ontologies.

Document Identifier:	NEON/2009/D1.4.3/v1.0	Date due:	February 28, 2009
Class Deliverable:	NEON EU-IST-2005-027595	Submission date:	February 28, 2009
Project start date	March 1, 2006	Version:	v1.0
Project duration:	4 years	State:	Final
		Distribution:	Public

NeOn Consortium

This document is part of the NeOn research project funded by the IST Programme of the Commission of the European Communities by the grant number IST-2005-027595. The following partners are involved in the project:

<p>Open University (OU) – Coordinator Knowledge Media Institute – KMi Berrill Building, Walton Hall Milton Keynes, MK7 6AA United Kingdom Contact person: Martin Dzbor, Enrico Motta E-mail address: {m.dzbor, e.motta}@open.ac.uk</p>	<p>Universität Karlsruhe – TH (UKARL) Institut für Angewandte Informatik und Formale Beschreibungsverfahren – AIFB Englerstrasse 11 D-76128 Karlsruhe, Germany Contact person: Peter Haase E-mail address: pha@aifb.uni-karlsruhe.de</p>
<p>Universidad Politécnica de Madrid (UPM) Campus de Montegancedo 28660 Boadilla del Monte Spain Contact person: Asunción Gómez Pérez E-mail address: asun@fi.ump.es</p>	<p>Software AG (SAG) Umlandstrasse 12 64297 Darmstadt Germany Contact person: Walter Waterfeld E-mail address: walter.waterfeld@softwareag.com</p>
<p>Intelligent Software Components S.A. (ISOCO) Calle de Pedro de Valdivia 10 28006 Madrid Spain Contact person: Jesús Contreras E-mail address: jcontreras@isoco.com</p>	<p>Institut 'Jožef Stefan' (JSI) Jamova 39 SL-1000 Ljubljana Slovenia Contact person: Marko Grobelnik E-mail address: marko.grobelnik@ijs.si</p>
<p>Institut National de Recherche en Informatique et en Automatique (INRIA) ZIRST – 665 avenue de l'Europe Montbonnot Saint Martin 38334 Saint-Ismier, France Contact person: Jérôme Euzenat E-mail address: jerome.euzenat@inrialpes.fr</p>	<p>University of Sheffield (USFD) Dept. of Computer Science Regent Court 211 Portobello street S14DP Sheffield, United Kingdom Contact person: Hamish Cunningham E-mail address: hamish@dcs.shef.ac.uk</p>
<p>Universität Koblenz-Landau (UKO-LD) Universitätsstrasse 1 56070 Koblenz Germany Contact person: Steffen Staab E-mail address: staab@uni-koblenz.de</p>	<p>Consiglio Nazionale delle Ricerche (CNR) Institute of cognitive sciences and technologies Via S. Marino della Battaglia 44 – 00185 Roma-Lazio Italy Contact person: Aldo Gangemi E-mail address: aldo.gangemi@istc.cnr.it</p>
<p>Ontoprise GmbH. (ONTO) Amalienbadstr. 36 (Raumfabrik 29) 76227 Karlsruhe Germany Contact person: Jürgen Angele E-mail address: angele@ontoprise.de</p>	<p>Food and Agriculture Organization of the United Nations (FAO) Viale delle Terme di Caracalla 00100 Rome Italy Contact person: Marta Iglesias E-mail address: marta.iglesias@fao.org</p>
<p>Atos Origin S.A. (ATOS) Calle de Albarraçín, 25 28037 Madrid Spain Contact person: Tomás Pariente Lobo E-mail address: tomas.pariantelobo@atosorigin.com</p>	<p>Laboratorios KIN, S.A. (KIN) C/Ciudad de Granada, 123 08018 Barcelona Spain Contact person: Antonio López E-mail address: alopez@kin.es</p>

Work package participants

The following partners have taken an active part in the work leading to the elaboration of this document, even if they might not have directly contributed to the writing of this document or its parts:

- OU
- INRIA
- UPM

Change Log

Version	Date	Amended by	Changes
0.1	07-03-2008	Holger Lewen	Skeleton of Deliverable created
0.2	07-14-2008	Jérôme Euzenat	Author added
0.3	01-25-2009	Holger Lewen	Seeded document with ESWC Demo Submission
0.4	02-12-2009	Mathieu d'Aquin	Added the development plan chapter with details of the integration of Watson, and other things
0.5	02-14-2009	Raul Palma	Added content about Oyster integration
0.6	02-18-2009	Jérôme Euzenat	Added content about Alignment integration
0.7	02-19-2009	Holger Lewen	Added content about TS-ORS integration and user manual
0.8	02-20-2009	Mathieu d'Aquin	Corrections
0.9	02-23-2009	Holger Lewen	Compiled final version for internal review.
1.0	03-20-2009	Holger Lewen	Addressed comments from the internal review.

Executive Summary

In this deliverable, we present the Cupboard system for ontology publishing, sharing and reuse. This system is intended to support both ontology engineers and ontology users/practitioners. For the developers of ontologies, it offers a complete infrastructure to host their ontologies in online ontology spaces, providing mechanisms to describe, manage and effectively exploit these ontologies (through APIs). Furthermore, these ontologies are then exposed to the community, providing users with a complete, friendly environment to find, assess and reuse ontologies. We furthermore describe the current state of implementation and the plan for completing the rest. Also a user's guide is provided to guide the users through the application.

Contents

1	Introduction – What is Cupboard	7
2	The Cupboard System	8
2.1	Overview	8
2.2	Architecture	9
3	Development Plan	11
3.1	Details of the Current Implementation	11
3.1.1	The Cupboard Core	11
3.1.2	Integration of Watson	12
3.1.3	Integration of Oyster	12
3.1.4	Integration of the TS-ORS review system	13
3.2	Availability	13
3.3	Scheduled Developments	14
3.3.1	Integration of the Alignment Server (March 2009)	14
3.3.2	Integration of advanced visualization mechanism (April 2009)	14
3.3.3	Federated API and NeOn Toolkit Plugins (May 2009)	15
3.3.4	Distributed System (end 2009)	15
3.4	Release Plan	15
4	Conclusion	17
5	Appendix	18
5.1	User Manual	18
5.1.1	Installation	18
5.1.2	Getting started – Logging in	18
5.1.3	Ontology Space	19
5.1.4	Adding an ontology	19
	Bibliography	28

List of Figures

2.1	Screenshots of some components of the Cupboard Web interface.	9
2.2	Cupboard's architecture.	10
5.1	This welcome screen is currently displayed when surfing to the Cupboard website.	18
5.2	The system accepts an OpenID as login-credential.	19
5.3	The ontology space contains all ontologies added by the user. It also displays additional metadata of the ontologies.	19
5.4	Using this form, ontologies can be added to the ontology space.	20
5.5	Here the user can decide to adapt the namespace of the ontology.	20
5.6	Metadata can be entered for the ontology.	21
5.7	For certain metadata, entries can be selected from a drop-down list.	22
5.8	Here the metadata for an ontology is displayed.	22
5.9	Using this form, the Cupboard system can be queried.	23
5.10	A result set for the query for "University".	23
5.11	A result set for the query for "University" within the specified ontology.	24
5.12	After clicking on the "+", the results are extended to display additional information.	24
5.13	The user can provide star ratings and explanations for these ratings for each of the five ontological properties.	25
5.14	The reviews are presented broken down into the 5 properties. The most relevant review is displayed first. By clicking on "I trust" or "I don't trust", trust or distrust statements are stored in the databased and the web of trust is updated.	26
5.15	The reviews are presented broken down into the 5 properties. The most relevant review is displayed first, based on the general opinion of the users.	27

Chapter 1

Introduction – What is Cupboard

In the recent years, there has been an ever-increasing amount of ontologies indexed by ontology search engines such as Watson¹. Yet, even with so many ontologies now available to reuse, the tasks of finding, assessing and effectively making use of existing ontologies remain difficult. We argue that the main factor inhibiting ontology reuse relates to the lack of support for ontology practitioners to find, assess and exploit existing ontologies, and incidentally to the lack of support for ontology engineers to effectively deploy and expose their ontologies for use and reuse.

Ontology search engines provide a partial answer to this problem, as they support users in finding, selecting and, in some cases, using ontologies (see e.g., [dMS⁺08]). However, they still lack mechanisms for assessing ontologies, for providing rich ontology metadata, for providing alignments between ontologies, etc. In other terms, they only focus on providing the ontologies, without leaving ontology publishers and users any control or any possibility of interaction among themselves and with the ontologies. In addition, there have been several initiatives coming up with ontology repositories lately (see e.g., [MSN⁺08, BED04]). However, these systems are either too closed (e.g., not letting users add ontologies) or they did not draw a significant number of participants. In general, the benefits for ontology engineers of exposing ontologies through these repositories is not obvious, as there is currently no clear incentive.

Apart from the Watson ontology search engine already mentioned, the NeOn project² is developing a number of technologies that, if put together and integrated in a broader system, can provide a more comprehensive answer to the aforementioned problem. More specifically, Oyster [PH05] is a peer-to-peer ontology sharing system which relies on rich metadata for ontologies, using the OMV format [HPS⁺05]. The alignment server [Euz04] is an open system dedicated to the creation, management and evaluation of ontology alignments. In addition, a *topic specific open rating system* (TS-ORS) [LSNM06] for ontologies is being developed to allow users to review different properties of ontologies and to express trust in reviewers.

In this deliverable, we present Cupboard, an online system that relies on these components and integrates them to provide a complete and friendly environment for ontology publishing, sharing and reuse. One of the most important questions addressed in this development concerns the incentives provided for an ontology engineer to share his/her ontology and invest the time to properly add it to a repository. For many users, setting up their own server infrastructure, generating search indexes and setting up SPARQL-endpoints was quite a cumbersome task. Cupboard does not only allow users to add their ontologies in a personal space, but actually indexes them (using the Watson engine), provides mechanism to link them (using the Alignment Server), hosts them and exposes them through APIs and SPARQL, thus offering the infrastructure for engineers to deploy their ontologies with minimal efforts. Moreover, Cupboard is designed to be a community tool, helping ontology users and practitioners (including ontology developers) in finding and reusing ontologies, through the use of rich ontology metadata (thanks to Oyster and OMV) and to advanced ontology review mechanisms (using the TS-ORS system).

¹<http://watson.kmi.open.ac.uk>

²<http://neon-project.org>

Chapter 2

The Cupboard System

Cupboard is different from classical ontology repository systems as it does not provide one single space where ontologies are exposed. On the contrary, each user can create his/her own *ontology space*, containing the ontologies he/she has selected. In a sense, Cupboard can be seen more as a system to host ontology repositories, than as an ontology repository itself.

2.1 Overview

The main advantage in relying on this notion of ontology spaces is that each ontology space virtually implements a complete infrastructure for building semantic applications. Indeed, each uploaded ontology is automatically indexed using the Watson engine. Watson is designed to be an ontology search engine, but focuses on providing the necessary access mechanisms (through services, APIs, SPARQL, etc.) to enable semantic applications to dynamically exploit ontologies made available online (see [dMS⁺08]). In Cupboard, each ontology space becomes like a little Watson, making it possible for developers to easily build applications exploiting the exposed ontologies.

However, for ontology and application developers, Cupboard's ontology spaces should do more than listing ontologies. To be used jointly in applications, ontologies have to be related with each other. Using the Alignment Server [Euz04] as a core component, Cupboard allows users to populate their ontology spaces not only with ontologies, but also with alignments (mappings). Using the *Alignment Format* [Euz04], alignments can be uploaded for a given (pair of) ontology(ies). The Alignment server then offers to Cupboard the necessary features to support the management, evaluation, and even production of alignments.

Uploading an ontology into Cupboard is not only a way to build applications, but also a way to share this ontology with the community for reuse. It is therefore important to provide efficient ways for users to find, assess and select ontologies in Cupboard's ontology spaces. The system naturally inherits the powerful search mechanisms implemented in the Watson engine. In addition, information for example on the provenance of the ontology or the methodology used to build it are essential for ontology selection. Cupboard relies on the Oyster system [PH05] to provide users with the possibility to enter rich metadata about ontologies (in the OMV format [HPS⁺05]) and to manage these metadata.

Finally, one of the most important issues hampering reuse concerns the evaluation of ontologies. In Cupboard, we intend to exploit the community feature of the system, so that users can provide precise reviews on ontologies (reviewing separately different characteristics, like reusability, complexity, coverage, etc.) The TS-ORS System [LSNM06], used to handle and manage these reviews, implements a mechanism to rank ontologies according to these evaluations. However, reviews are subjective evaluations, as users might have different views on ontologies. When searching for an ontology to reuse, not only the review information should be exploited to support the user, but also the information about how much he/she would trust a particular review or a particular reviewer. Such trust information is obtained through the classical "Did you find this review useful?" question, and handled by the TS-ORS system to compute trust information used to personalize

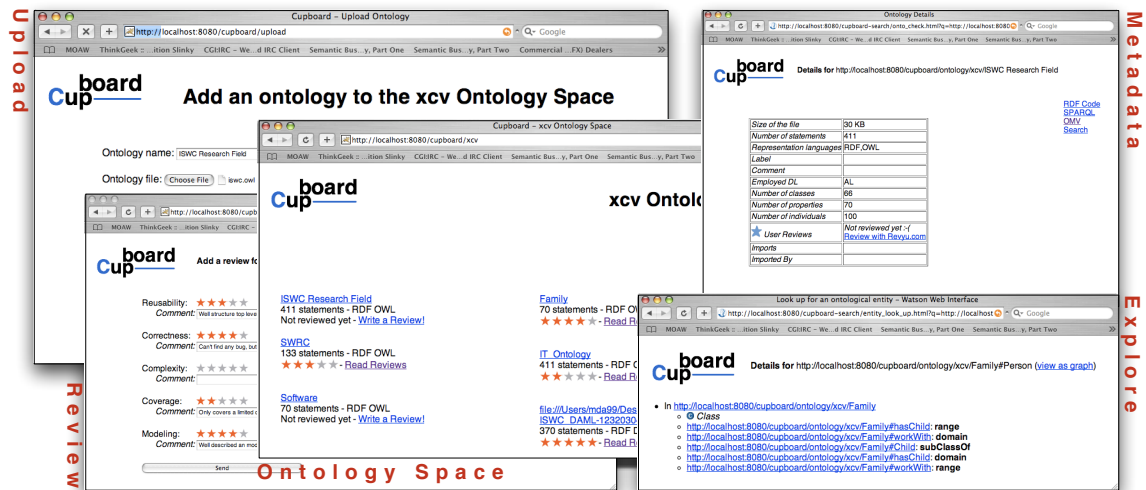


Figure 2.1: Screenshots of some components of the Cupboard Web interface.

ontology ranking.

Figure 2.1 shows different parts of the Web interface of Cupboard, in particular displaying an ontology space, uploading an ontology, inspecting and searching the content of a particular ontology, reviewing it, etc.

2.2 Architecture

While the different technologies employed and combined in the system have been described and implemented before, this is the first time that they are combined to expose a broader and more comprehensive set of functionalities to ontology engineers and users alike. The main challenges during integration were defining the interfaces between components, optimizing the performance and generating an easy-to-use Web interface. So, while Cupboard is building on existing systems, the user experience is completely different and we believe that users will find it easier to take advantage of all the available techniques.

As depicted in Figure 2.2, the central component of Cupboard is a specifically made component, called the *Cupboard core*, which is in charge of orchestrating the interaction between all the other components, external applications and users. It heavily relies on the Watson engine to provide basic functionalities such as storing the ontologies, indexing them for search, validating them and exposing them to applications. The TS-ORS open rating system, the Oyster ontology metadata sharing system and the Alignment Server each provide specific functionalities to the system, that are federated and exposed in an homogeneous way to the Cupboard Web interface and APIs, through the Cupboard core.

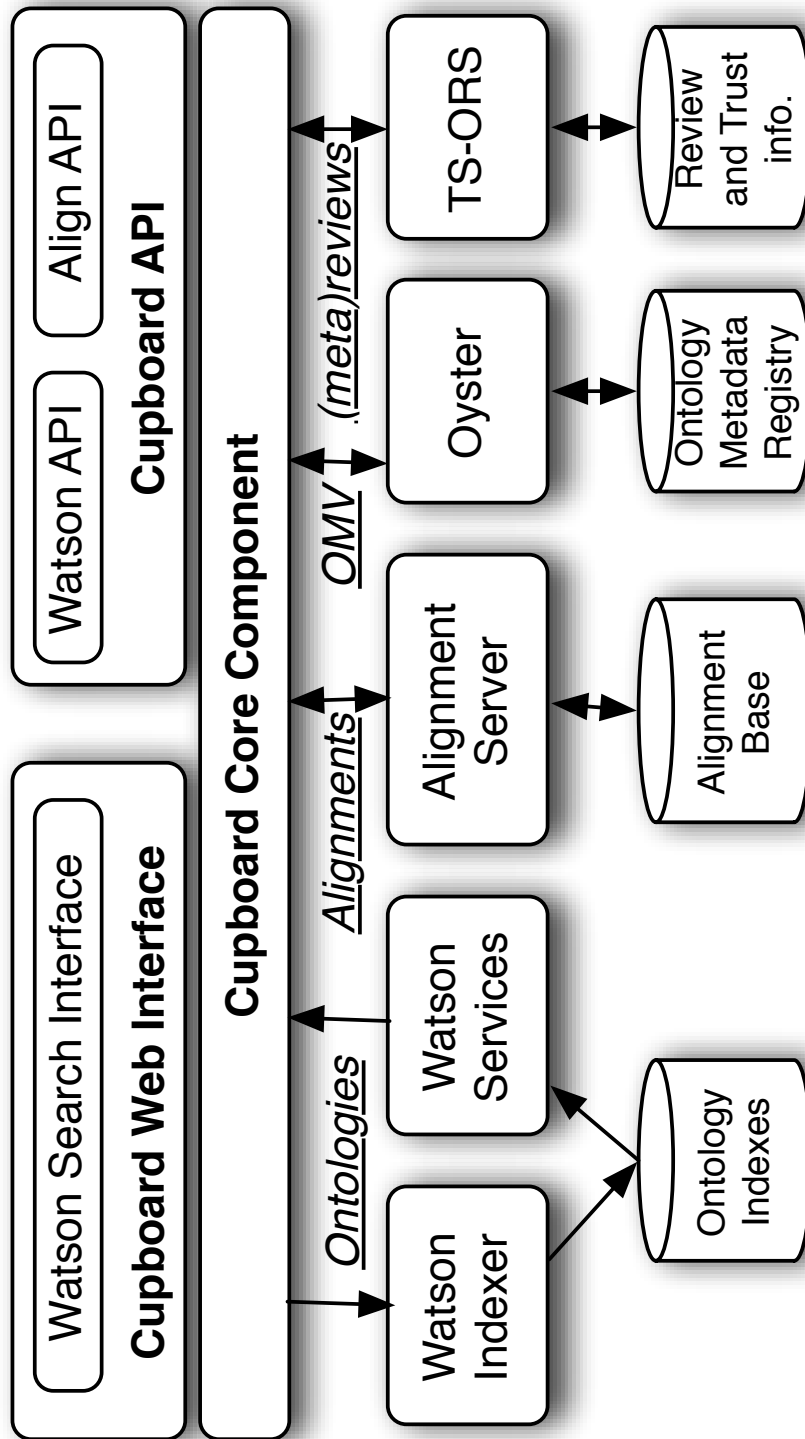


Figure 2.2: Cupboard's architecture.

Chapter 3

Development Plan

At the current stage, a running prototype of the Cupboard system has been developed, which provides the essential functionalities for sharing, reviewing and annotating ontologies with rich metadata. Testing, extending and finalizing this prototype are ongoing tasks. Below, we provide details of the current implementation, in particular on the integration of Watson, Oyster and the TS-ORS. We also indicate a concrete schedule for extending this prototype with planned features (e.g., through the integration of the Alignment server) and for more robustness, as well as the release plan for the system, towards an open tool to which any web user could register.

3.1 Details of the Current Implementation

As explain in Section 2.2, the architecture of Cupboard is centered around a ‘core’ component, to which all the other components are plugged. In the current implementation, three of the planned components have been (at least partly) integrated: Watson, Oyster and the TS-ORS system.

3.1.1 The Cupboard Core

The role of the Cupboard Core component is twofold:

- it provides the necessary interfaces to integrate the other components of the system and to federate them, in order to offer a complete set of functionalities.
- it provides the interfaces to the external world, in particular user interfaces, to access the functionalities offered by the other components in an homogeneous way.

As such, the Cupboard Core is the only part of the development of Cupboard which is entirely new and which do not rely on the reuse of previously existing components. In the following, mentions to the integration of components to Cupboard concretely relate to the implementation within the Cupboard Core of the interaction with specific components.

Concretely, the Cupboard Core is implemented as a Java Web application (i.e., a set of servlet, enhanced with Javascript based interface elements). It provides, on the one hand, a set of GUIs for the functions provided by Cupboard (see Figure 2.1) and on the other hand, the back-end operations, relying on the integrated components, to realize these functions. It is also intended to provide a complete API to build applications on top of exposed ontology spaces, but currently only realizes a very limited sub-part of the planned functions (see Section 3.3).

3.1.2 Integration of Watson

The Watson system¹ can be seen as a search engine for the Semantic Web. It supports users and applications in finding, selecting and (re)using ontologies that are available online, through advanced search and exploration mechanisms. Its role in Cupboard is to index the ontologies that are uploaded, in order to provide the same search and exploration mechanisms, not for automatically crawled ontologies, but for ontologies exposed within ontology spaces. As such, the integration is realized by deploying a dedicated Watson engine within the Cupboard server, comprising both the Watson indexer and the Watson services. Concretely, this means that Cupboard relies on its own local indexes for uploaded ontologies, which are produced, accessed and exploited through a local (i.e., on the Cupboard server) instantiation of the Watson engine.

In practice, the Watson engine is then used to provide the following functionalities:

- Retrieving all the ontologies contained in a given ontology space (one of the core functionality of Cupboard)
- Analyzing the content of ontologies to automatically extract elementary metadata (e.g., label, comment, URI, imported ontologies, etc.), which are then passed on to Oyster
- Exploring the content of ontologies, by providing navigation functions to investigate the relations between ontological entities
- Searching for semantic content exposed through Cupboard, to find ontologies or ontological entities to be reused

In addition, elements of the Watson web interface are also reused within Cupboard (e.g. the search interface), and simply 're-branded' to appear as an homogeneous part of the Cupboard GUI.

3.1.3 Integration of Oyster

Oyster is a distributed registry that exploits semantic web techniques in order to provide a solution for exchanging and re-using ontologies and related entities. As an ontology registry, it provides services for storage, cataloging, discovery, management, and retrieval of ontology (and related entities) metadata definitions. To achieve these goals, Oyster implements the proposal for metadata standard OMV² as the way to describe ontologies and related entities, supporting advanced semantic searches of the registered objects and providing services to support the management and evolution of ontologies in distributed environments.

The role of Oyster in Cupboard is the management of OMV metadata for the ontologies exposed within ontology spaces in order to facilitate their discovery by supporting metadata-based searches. Hence, the integration is realized by deploying an Oyster peer within the Cupboard server that is populated by metadata about every ontology uploaded into the system. Concretely, a local Oyster peer running in server mode is accessed by the Cupboard system via its API. Consequently, ontologies exposed within Cupboard also become available to the whole Oyster network.

In practice, it provides the following functionalities:

- Storing OMV metadata for ontologies exposed within ontology spaces
- Retrieving OMV metadata for displaying ontology details.
- Searching for ontologies using metadata-based conditions.
- Extracting metadata information from uploaded ontologies that is then complemented with Watson extracted metadata.

¹<http://watson.kmi.open.ac.uk>

²<http://omv.ontoware.org>

3.1.4 Integration of the TS-ORS review system

The topic-specific open rating system provides users the possibility to review ontologies within Cupboard. For each of the properties of the ontologies, they can provide a 5 star rating and a textual explanation for the rating. Other users can specify whether they find the reviews helpful, and thereby express trust to the reviewers. Based on the web of trust, the reviews can be ranked in a user-specific way and also overall ratings of the ontologies can be computed according to a users needs and preferences. For more technical information, see [LSNM06, SAd⁺07].

The role of TS-ORS in Cupboard is to add user-based evaluation to the ontologies. Without quality information available, it is hard for users to judge with ontology is good enough to be reused in their applications / ontologies. Because evaluating the ontologies themselves can take quite some time, user like to rely on existing evaluations. The quality information provided by the community can then be used to order reviews of the ontologies, and also the ontologies themselves based on user-specific trust and parameters.

The TS-ORS's functionality is exposed through servlets, which are called by the Cupboard core. It can be seen as a REST-like API. The results are provided as HTML, XML or JSON, based on content negotiation between the Cupboard core and the TS-ORS (based on the HTTP-request header).

In practice, the TS-ORS provides the following functionalities:

- Storing user reviews
- Exposing user reviews in a personalized (if user is identifiable) order
- Calculating overall ratings for ontologies based on their reviews and user-specific parameters and trust (if user is identifiable).
- Managing the trust and metatrust information

3.2 Availability

At the time of writing this deliverable, the Cupboard system is being tested locally by the contributors to its development. It is quickly evolving towards a more stable and robust system. A test and demonstration version is made available at the following URL

`http://kmi-web06.open.ac.uk:8081/cupboard`

It is not yet possible in this version to register new users (see the release plan Section 3.4), but a test user has been created, which can simply be used by entering the following ID

`http://example.com/openid/testuser`

in the login screen

`http://kmi-web06.open.ac.uk:8081/cupboard/login`

Once logged in, the interface automatically redirects to the ontology space owned by this test user:

`http://kmi-web06.open.ac.uk:8081/cupboard/test`

where ontologies can be added, reviewed, inspected, searched and annotated with metadata. The ontology space of another user is also available to explore:

`http://kmi-web06.open.ac.uk:8081/cupboard/test2`

3.3 Scheduled Developments

The current prototype for the Cupboard system constitutes a promising first step towards a more complete and robust system, providing users and application developers with the ability to create their own space on the semantic web, to share, expose and reuse networked ontologies in a efficient and collaborative environment. Below, we list and detail the next steps in the development of Cupboard to achieve this goal.

3.3.1 Integration of the Alignment Server (March 2009)

Alignment servers are middleware components of the semantic web infrastructure for storing and sharing alignments. They enable applications to share high quality and well referenced alignments featuring metadata annotations. Alignment servers provide access to libraries of matchers and alignments and offer retrieval, manipulation (through applying threshold or transformations), and export of alignments under the adequate format. It is accessible from web browsers, integrated development environments, web services and agents. Thanks to its plug-in architecture it can be extended by new matchers, renderers, communication drivers and directories. They can be used by semantic web applications as well as by the infrastructure itself. They can be used at two different moment in applications:

at design time through invocation by design and engineering environments: they can be integrated within development environments, where they will be loosely coupled components which may be asked for alignments and for exploiting these alignments (like the NeOn toolkit through the NeOn Alignment plug-in[DdB⁺08]).

at run time alignment servers can be invoked directly by the application.

Cupboard takes advantage of Alignment servers in order to connect ontologies together with alignment. Hence, an Alignment server will provide on demand alignments between ontologies referred to in Cupboard. These alignments remain stored in the Alignment server and Cupboard maintain the references to these alignments. This provides a convenient way to store networked ontologies within Cupboard.

The integration of Alignment servers with Cupboard results from a loose coupling between Cupboard and available remote servers. Cupboard uses the extended REST web service interface in order to obtain information from an Alignment server. Cupboard will integrate the Alignment API, which is the basis of Alignment servers, so as to manipulate easily alignments and information about alignments.

In practice, the integration will provide the following functions:

- reference to alignments from the Cupboard ontology descriptions;
- functions to "import" alignments within Cupboard;
- display of alignment metadata within Cupboard;
- display of alignment content (correspondences) within Cupboard;
- alignment rating through ORS (in a latter phase).

3.3.2 Integration of advanced visualization mechanism (April 2009)

NeOn workpackage 4 is currently developing novel tools and techniques to visualize and navigate within ontologies [Dzb09]. The idea is to rely on a summary of an ontology in the form of key concepts. The key concepts provide an general overview of the content of the ontology, allowing users to quickly make sense of what it is about and if it could be of particular interest. Based on the technique described in [PMd08] to extract key concepts from an ontology, new visualization methods are devised that not only provide a preview of an ontology, but also allow the user to expand this preview by 'zooming' in to more detailed content and to identify focus points within these detailed views. Such visualization techniques are of evident interest for the Cupboard system and will be integrated within the user interface in order to provide the user with a rapid overview of the content of ontologies exposed through an ontology space.

3.3.3 Federated API and NeOn Toolkit Plugins (May 2009)

As explained before, one of the core aspect of Cupboard is not only to provide user interfaces for sharing, finding and using ontologies, but also a complete set of services and APIs for applications to be able to automatically expose ontologies, alignments and metadata to ontologies, as well as to find, select and exploit such elements from Cupboard. In particular, one of the essential elements of this API will be based on the Watson API, which will be adapted so that developers could search and query specific ontology spaces, and build new applications directly exploiting the content of Cupboard's ontologies. However, this API will have to be extended to integrate functionalities provided by the other components of Cupboard, giving the ability to applications to share and exploit alignments, reviews and ontology metadata as well.

In addition, as part of this development plan, we intend to develop a set of NeOn toolkit plugins relying on the Cupboard federated API. The goal here is to realize a direct connection between the environment of the NeOn Toolkit and the infrastructure facilities offered by Cupboard, allowing ontology engineers to automatically share and store ontologies developed within the toolkit in Cupboard, as well as to import, exploit and reuse existing ontologies exposed in Cupboard. These plugins will rely on previous work related to the development of the Watson plugin [dSM08], the Oyster plugin³ and the Alignment plugin [DdB⁺08], integrating them in order to fully exploit all the functionalities attached to Cupboard ontology spaces.

3.3.4 Distributed System (end 2009)

Finally, in the current implementation, Cupboard is intended to be deployed in a unique server, providing a central point for sharing and finding ontologies. However, for various reasons, we intend to distribute the system as an open source package, that individual organizations can install on their own servers and use locally. This could be useful in particular when ontologies cannot be shared externally for privacy reasons.

This would lead to the deployment of several Cupboard servers that should be linked with each other and communicate, thus making Cupboard a distributed access point to shared ontologies, rather than a central one. This require additional efforts to implement communication and query routing mechanisms between Cupboard servers, which will rely on existing mechanisms implemented by the integrated components (e.g., the P2P network of Oyster or the distributed version of Watson relying on the P2P architecture of the Open-Knowledge project [GLP⁺08]).

3.4 Release Plan

As explained before, Cupboard is currently at the stage of prototype and is being tested by contributors to its development. Besides the development plan, a release plan has been devised, fixing the targets and objectives towards the final launch of Cupboard and its opening to external users for registration. This release plan also include testing the system on concrete use cases by supporting experiments and systems developed within NeOn:

February 2009 (now) Demonstration and test system, including 2 test users. Deployed on a server at the OU and accessible from outside, to allow NeOn members to try and test the system.

April 2009 Working prototype used for the reuse experiment in WP5. Small numbers of ontology spaces populated with existing ontologies and metadata. Selected users manually registered to provide reviews on these ontologies.

June 2009 Use of Cupboard for the `ontologydesignpatterns.org` portal. A couple of dedicated ontology spaces automatically populated from contributions to the portal. The Cupboard API used in the portal to provide search and reuse functionalities.

³<http://www.neon-toolkit.org/wiki/index.php/Oyster-GUI>

June 2009 Opening of Cupboard as a beta system to external users (on invitation).

September 2009 Opening of Cupboard for any user to register.

January 2010 Distribution of Cupboard as an installable open source package.

Chapter 4

Conclusion

In this document, we presented Cupboard, an online ontology publishing and sharing environment that addresses two major problems faced by the Semantic Web community: the deployment of inter-linked ontologies for exploitation in applications, often requiring heavy infrastructures, and the reuse of existing ontologies where, besides search mechanisms, rich metadata and proper assessment of the ontologies are required. To tackle these problems, we propose a complete system, based on a number of well-established technologies, allowing ontology engineers to deploy their ontologies, providing the necessary infrastructures to support their exploitation, and ontology users in reusing available knowledge, providing essential, community-based functionalities to facilitate the search, selection and exploitation of the available ontologies. To the best of our knowledge, Cupboard is the first system to put together all these functionalities to create an essential infrastructure component for Semantic Web developers and more generally, a useful, shared and open environment for the ontology community.

We have also presented a detailed overview of what is available now and what will be implemented at a later stage. Furthermore, a user manual provided as an appendix.

Chapter 5

Appendix

5.1 User Manual

We plan to also have the latest version of the user manual linked from the Cupboard website.

5.1.1 Installation

Cupboard does not have to be installed by the end user, it is accessible by Web Browser (currently tested on Firefox and Safari) by simply entering the URI. We will however, at a later point, make a complete Cupboard package available for companies wanting to run their own local version of Cupboard. This package will then come with a dedicated developer and system administrator handbook. Our local installation can be found at <http://kmi-web06.open.ac.uk:8081/cupboard>. At the moment we display a screen with information on how to become a betatester (see fig. 5.1).

5.1.2 Getting started – Logging in

You can normally log into the system using your OpenID¹ (see fig. 5.2). Once logged in, the Ontology Space of the logged-in user is displayed.

¹for more information see <http://openid.net/>



Welcome to Cupboard!
This service is currently in development. A Beta version will be available soon.
If you want to be among the firsts to possess their own *Ontology Space* on Cupboard, send an e-mail to [Mathieu d'Aquin](mailto:Mathieu.d'Aquin) with your *OpenId* and the name of your *Ontology Space*.
You will then be added to the list of *Beta Testers* of Cupboard, and have the chance to have your own space on the Semantic Web very soon!

Figure 5.1: This welcome screen is currently displayed when surfing to the Cupboard website.



OpenID:

Figure 5.2: The system accepts an OpenID as login-credential.



selahi Ontology Space

[Home](#)
[Add Ontology](#)
[Search](#)
[Logout](#)

[selahiFourth](#)
70 statements - RDF OWL
[Write a Review!](#)
★★★★☆ - [Read Reviews](#)

<http://swrc.ontoware.org/ontology-07>
1168 statements - RDF OWL
[Write a Review!](#)
★★★★★ - [Read Reviews](#)

[selahiSeventh](#)
2332 statements - RDF OWL - [Read Reviews](#)
[Write a Review!](#)

[selahiEighth](#)
32 statements - RDF OWL
[Write a Review!](#)
★★★★☆ - [Read Reviews](#)

[selahiThird](#)
411 statements - RDF OWL
[Write a Review!](#)
★★★★☆ - [Read Reviews](#)

Figure 5.3: The ontology space contains all ontologies added by the user. It also displays additional metadata of the ontologies.

5.1.3 Ontology Space

The ontology space (for an example see fig. 5.3) contains all the ontologies added by a user. If you browse your own ontology space, you can find all the ontologies you added. It is like the home screen of Cupboard. For each ontology, the title, number of statements, format and overall rating are displayed. From this screen you have several options: Add an ontology, search, write a review or read reviews. We will now explain each of these functionalities.

5.1.4 Adding an ontology

When you click on “Add Ontology” in your ontology space, you will get a dialog that allows you to upload ontologies. You can enter the name, and either specify a location on the filesystem where your ontology is located, or enter the ontology source code directly (see fig. 5.4). We will also allow adding ontologies by URI very soon. After the ontology has been added, the URI has to be adapted to be properly accessible within Cupboard. The adaptation can be done by simply clicking the button (see fig. 5.5). As a next step, metadata information is gathered by the system. By default, only one form is expanded (see fig. 5.6), but by



Add an ontology to the xcv Ontology Space

Ontology name:

Ontology file:

or ontology source code:

Figure 5.4: Using this form, ontologies can be added to the ontology space.



There is no declared default namespace for the xcvSixth ontology.
In order for this ontology to be properly accessible when published in Cupboard, it has to be changed into/set to

<http://localhost:8080/cupboard/ontology/xcv/xcvSixth>

Do you agree with this change?

Note that if you chose not to apply this change, the ontology will still be added to your Ontology Space, but the entities it contains will not be *dereferenceable*

Figure 5.5: Here the user can decide to adapt the namespace of the ontology.



Ontology Metadata Form

General Metadata

Ontology Name:

URI:

Description:

Creation Date:

Provenance Metadata

Applicability Metadata

Format Metadata

Availability Metadata

Statistics Metadata

Figure 5.6: Metadata can be entered for the ontology.

clicking on the “+” sign, also the other categories can be extended, so that metadata can be entered (see fig. 5.7). Whenever possible, we use controlled vocabularies, which then are presented as a drop-down list (see fig. 5.7).

In the ontology space, ontology metadata can be seen when clicking on the URI of the ontology (see fig. 5.8)

Search

If Cupboard, the user can access the search functionality by either clicking on “search” in the ontology space (see fig. 5.3), or search an ontology directly from the metadata view (see fig. 5.8). The search screen can be seen in fig. 5.9. The user can enter a search term, and also specify specific search options. Once the results are found, they are presented (see fig. 5.10) to the user.

It is also possible to search within an ontology (see fig. 5.11). The resultset can be extended by clicking on the “+” signs (see fig. 5.10).

Reviews and Reviewing

One of the central ideas of Cupboard is to also allow the community to make use of other users’ experience. We therefore allow for detailed reviews of the ontologies, and employ a sophisticated ranking system to provide the results in a user-specific order. When a user clicks on “Write a review”, a form is displayed (see fig. 5.13) that allows the user to rate each of the ontological properties defined in the system by selecting a 1–5 star rating. They can and should also comment on the rating, explaining why this star rating was chosen. It is important to note that not all the properties have to be reviewed, it is also possible to review only one or some properties. On the other hand, when clicking read reviews on the ontology space, the reviews for this ontologies are displayed in a personalized order (if logged in) (see fig. 5.14), or based on global trust



Ontology Metadata Form

General Metadata

Ontology Name:

URI:

Description:

Creation Date:

Provenance Metadata

Creator:

Applicability Metadata

Ontology Type:

Please select the Ontology type

Upper Level

Core

Domain

Task

Application

Ontology Language:

Ontology Syntax:

Availability Metadata

Resource Location:

Figure 5.7: For certain metadata, entries can be selected from a drop-down list.



Details for <http://localhost:8080/cupboard/ontology/selahi/selahiThird>

- [RDF Code](#)
- [SPARQL](#)
- [OMV](#)
- [Search](#)

Size of the file	30 KB
Number of statements	411
Representation languages	RDF, OWL
Label	
Comment	
Employed DL	AL
Number of classes	66
Number of properties	70
Number of individuals	100
User Reviews	Not reviewed yet :(Review with Revwu.com
Imports	
Imported By	

Figure 5.8: Here the metadata for an ontology is displayed.



Figure 5.9: Using this form, the Cupboard system can be queried.

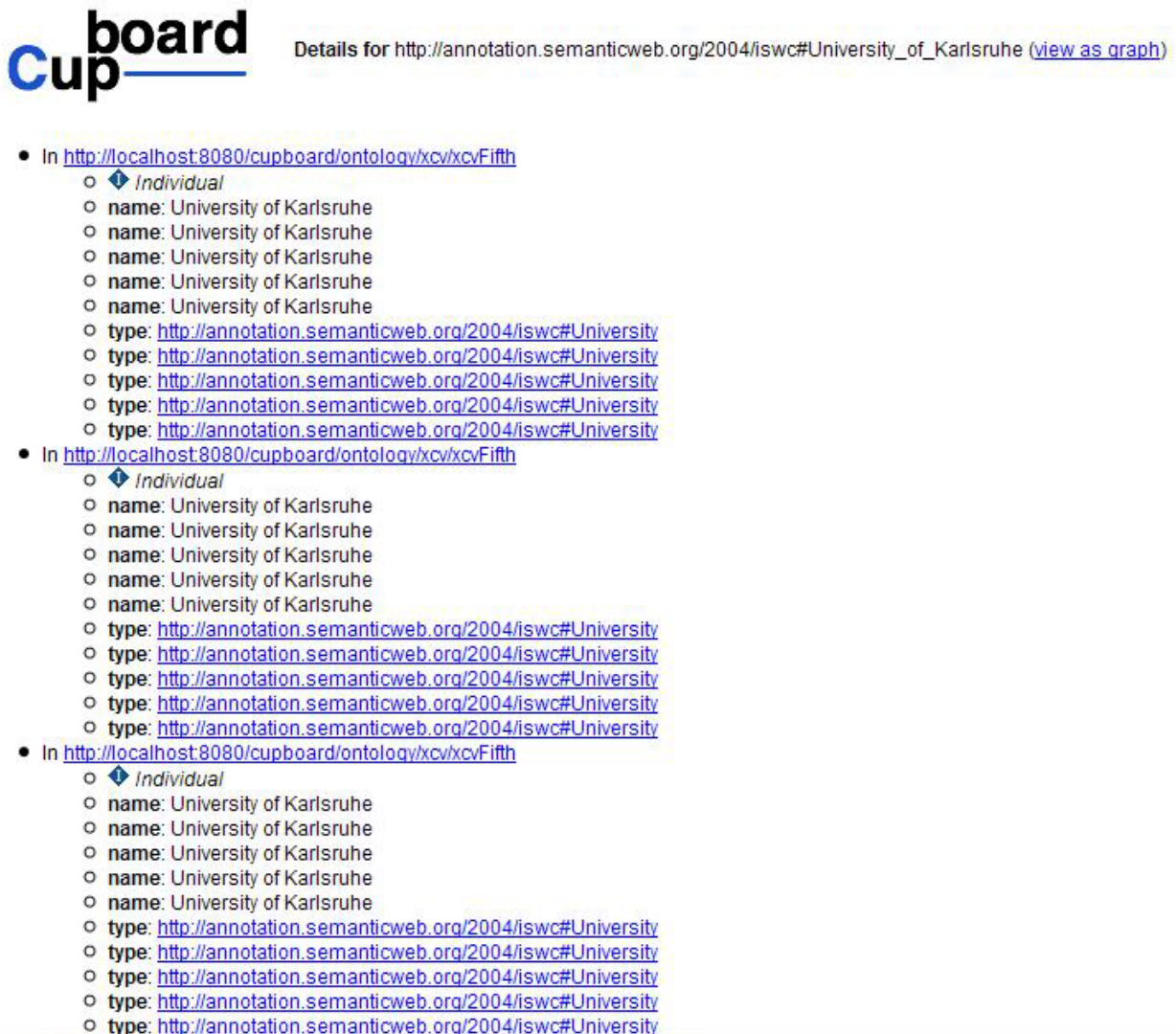


Figure 5.10: A result set for the query for “University”.

Cupboard

Ontology URI

Keywords [Search](#)

Found 10 entities

- http://annotation.semanticweb.org/2004/iswc#University_of_Karlsruhe +
- http://annotation.semanticweb.org/2004/iswc#University_of_Karlsruhe +
- http://annotation.semanticweb.org/2004/iswc#University_of_Karlsruhe +
- http://annotation.semanticweb.org/2004/iswc#University_of_Karlsruhe +
- http://annotation.semanticweb.org/2004/iswc#University_of_Karlsruhe +
- <http://annotation.semanticweb.org/2004/iswc#University> +
- <http://annotation.semanticweb.org/2004/iswc#University> +
- <http://annotation.semanticweb.org/2004/iswc#University> +
- <http://annotation.semanticweb.org/2004/iswc#University> +
- <http://annotation.semanticweb.org/2004/iswc#University> +

Figure 5.11: A result set for the query for “University” within the specified ontology.

Cupboard


Ontology URI

Keywords [Search](#)

Found 10 entities

- http://annotation.semanticweb.org/2004/iswc#University_of_Karlsruhe -
Label:
Comment: null
- http://annotation.semanticweb.org/2004/iswc#University_of_Karlsruhe -
Label:
Comment: null
- http://annotation.semanticweb.org/2004/iswc#University_of_Karlsruhe -
Label:
Comment: null
- http://annotation.semanticweb.org/2004/iswc#University_of_Karlsruhe -
Label:
Comment: null
- http://annotation.semanticweb.org/2004/iswc#University_of_Karlsruhe -
Label:
Comment: null
- <http://annotation.semanticweb.org/2004/iswc#University> +
- <http://annotation.semanticweb.org/2004/iswc#University> +
- <http://annotation.semanticweb.org/2004/iswc#University> +
Label:
Comment: null
- <http://annotation.semanticweb.org/2004/iswc#University> +
- <http://annotation.semanticweb.org/2004/iswc#University> +

Figure 5.12: After clicking on the “+”, the results are extended to display additional information.



Reusability
★ ★ ★ ★ ★
Comment:

Correctness
★ ★ ★ ★ ★
Comment:

Complexity
★ ★ ★ ★ ★
Comment:

Domain Coverage
★ ★ ★ ★ ★
Comment:

Modelling
★ ★ ★ ★ ★
Comment:

Figure 5.13: The user can provide star ratings and explanations for these ratings for each of the five ontological properties.

(what most people think is right) (see fig. 5.15). Note that when you are logged in, you can also add trust statements to the reviews, and thus change the way the reviews are ranked.



Review for xcvFirst in xcv

Review by Mathieu, DAQUIN

Property reviewed: usability

good



Review by Mathieu, DAQUIN

Property reviewed: domain coverage

good



Review by Mathieu, DAQUIN

Property reviewed: maturity

great



Review by Mathieu, DAQUIN

Property reviewed: correctness

OK



Figure 5.14: The reviews are presented broken down into the 5 properties. The most relevant review is displayed first. By clicking on “I trust” or “I don’t trust”, trust or distrust statements are stored in the databased and the web of trust is updated.



Review for xcvFirst in xcv

Reusability	
★★★★☆ (By Mathieu, DAQUIN)	good
★★★★☆ (By salman, ELAHI)	teest
★☆☆☆☆ (By Salman, NOOR)	noor
Correctness	
★☆☆☆☆ (By salman, ELAHI)	enwer
★★☆☆☆ (By Salman, NOOR)	noor
Complexity	
★★☆☆☆ (By Salman, NOOR)	noor
★★★★★ (By Mathieu, DAQUIN)	good
★★☆☆☆ (By salman, ELAHI)	teest
Domain Coverage	
★★★★★ (By salman, ELAHI)	Salman

Figure 5.15: The reviews are presented broken down into the 5 properties. The most relevant review is displayed first, based on the general opinion of the users.

Bibliography

- [BED04] P. Buitelaar, T. Eigner, and T. Declerck. OntoSelect: A Dynamic Ontology Library with Support for Ontology Selection. In *Proc. of the Demo Session at the International Semantic Web Conference, Hiroshima, Japan, 2004*.
- [DdB⁺08] Chan Le Duc, Mathieu d'Aquin, Jesus Barrasa, Jérôme David, Jérôme Euzenat, Raul Palma, Rosario Plaza, Marta Sabou, and Boris Villazón Terrazas. Matching ontologies for context: The neon alignment plug-in. NeOn Deliverable D3.3.2, 2008.
- [dMS⁺08] M. d'Aquin, E. Motta, M. Sabou, S. Angeletou, L. Gridinoc, V. Lopez, and D. Guidi. Toward a new generation of semantic web applications. *Intelligent Systems, IEEE*, 23(3):20–28, 2008.
- [dSM08] Mathieu d'Aquin, Marta Sabou, and Enrico Motta. Reusing Knowledge from the Semantic Web with the Watson Plugin. In *Proc. of the poster and demo session of the International Semantic Web Conference, ISWC, 2008*.
- [Dzb09] Martin Dzbor. Neon toolkit plugin for visualization and navigation in ontologies and ontology networks based on concept summarization and categorization. NeOn Deliverable D4.5.4, 2009.
- [Euz04] Jérôme Euzenat. An API for ontology alignment. In Sheila A. McIlraith, Dimitris Plexousakis, and Frank van Harmelen, editors, *International Semantic Web Conference*, volume 3298 of *Lecture Notes in Computer Science*, pages 698–712. Springer, 2004.
- [GLP⁺08] Davide Guidi, Vanessa Lopez, Silvio Peroni, Enrico Motta, Mathieu d'Aquin, Marta Sabou, George Anadiotis, Paolo Besana, and David Dupplaw. OpenKnowledge Question Answering System. OpenKnowledge Deliverable D8.4, 2008.
- [HPS⁺05] Jens Hartmann, Raúl Palma, York Sure, María del Carmen Suárez-Figueroa, Peter Haase, Asunción Gómez-Pérez, and Rudi Studer. Ontology metadata vocabulary and applications. In *OTM Workshops*, volume 3762 of *LNCS*, pages 906–915. Springer, 2005.
- [LSNM06] Holger Lewen, Kaustubh Supekar, Natalya F. Noy, and Mark A. Musen. Topic-Specific Trust and Open Rating Systems: An Approach for Ontology Evaluation. In *Proc. of the 4th International Workshop on Evaluation of Ontologies for the Web (EON2006) at the 15th International World Wide Web Conference (WWW), MAY 2006*.
- [MSN⁺08] M. Musen, N. Shah, N. Noy, B. Dai, M. Dorf, N. Griffith, JD Buntrock, C. Jonquet, MJ Montegut, and DL Rubin. BioPortal: Ontologies and Data Resources with the Click of a Mouse. In *Proc. of the Annual Symposium proceedings/AMIA Symposium. AMIA Symposium*, page 1223. AMIA Annu Symp Proc, 2008.
- [PH05] R. Palma and P. Haase. Oyster - sharing and re-using ontologies in a peer-to-peer community. In *Proc. of International Semantic Web Conference*, pages 1059–1062, 2005.
- [PMd08] Silvio Peroni, Enrico Motta, and Mathieu d'Aquin. Identifying key concepts in an ontology through the integration of cognitive principles with statistical and topological measures. In *Proc. of the third Asian Semantic Web Conference, ASWC, 2008*.

- [SAd⁺07] Marta Sabou, Sofia Angeletou, Mathieu d'Aquin, Jesus Barrasa, Klaas Dellschaft, Aldo Gangemi, Jos Lehmann, Holger Lewen, Diana Maynard, Dunja Mladenic, Malvina Nissim, Wim Peters, Valentina Presutti, and Boris Villazon. D2.2.1 methods for selection and integration of reusable components from formal or informal user specifications. NeOn Project Deliverable D2.2.1, The Open University, MAY 2007.