

DEMO - Design Environment for Metadata Ontologies

Jens Hartmann¹, Elena Paslaru Bontas², Raúl Palma³, and Asunción Gómez-Pérez³

¹ Institute AIFB, University of Karlsruhe, Germany

hartmann@aifb.uni-karlsruhe.de

<http://www.aifb.uni-karlsruhe.de/WBS/>

² Networked Information Systems,

Institute of Computer Science, Free University of Berlin, Germany

paslaru@inf.fu-berlin.de

³ Ontology Engineering Group, Laboratorio de Inteligencia Artificial

Facultad de Informática, Universidad Politécnica de Madrid, Spain

rpalma@delicias.dia.fi.upm.es

Abstract. Efficient knowledge sharing and reuse—a pre-requisite for the realization of the Semantic Web vision—is currently impeded by the lack of standards for documenting and annotating ontologies with metadata information. We argue that the availability of metadata is a fundamental dimension of ontology reusability. Metadata information provides a basis for ontology developers to evaluate and adapt existing Semantic Web ontologies in new application settings, and fosters the development of support tools such as ontology repositories. However, in order for the metadata information to represent real added value to ontology users, it is equally important to achieve a common agreement on the terms used to describe ontologies, and to provide an appropriate technology infrastructure in form of tools being able to create, manage and distribute this information. In this paper we present DEMO, a framework for the development and deployment of ontology metadata. Besides OMV, the proposed core vocabulary for ontology metadata, the framework comprises an inventory of methods to collaboratively extend OMV in accordance to the requirements of an emerging community of industrial and academia users, and tools for metadata management.

1 Introduction

As the Semantic Web grows, increasing numbers of private and public sector communities are developing ontologies which represent their domain(s) of interest. As ontologies are also intended to act as commonly agreed domain conceptualizations[7] it is expected that reuse will play a crucial role in the widespread dissemination of ontology-driven technologies. First, reusability is an intrinsic property of ontologies, originally defined as means for “*knowledge sharing and reuse*”[13]. Sharing and reusing existing ontologies increases their quality—as they are continuously accessed, used and revised by many people—and the quality of the applications using them—since these applications become (more) interoperable and are provided with a deeper, machine-processable understanding of the underlying domain. Second, analogously to other engineering disciplines, reusing existing ontologies—if performed in an efficient way—reduces the costs

related to ontology development, since it avoids the re-implementation of the components, which are already available on the Web and can be directly or after some additional customization integrated to a target ontology. Finally, access to available ontologies across the Web is a fundamental requirement for the dissemination of Semantic Web Services, which are envisioned to automatically use these sources in order to describe their capabilities and for interprocess communication.

While finding a solution to the problematic trade-off between usability in a specific application setting and wide-scale reusability is often considered an art more than a science, poor reusability can be significantly alleviated by resorting to several design principles, which have proved their relevance across various computer science disciplines confronted with this issue: *modularity* and *decomposition* along abstraction and functionality levels, the deployment of *standardized* tools and technologies, as well as careful *documentation* of the development process and up-to-date *component repositories*. Current research achievements in the Semantic Web field provide a feasible basis for many of these principles to be put into practice. The usage of standardized ontology representation languages in conjunction with the ubiquitous URI-based access on Semantic Web resources and the emergence of ontology management methods and tools definitely constitute a solid inventory for building more reusable ontologies. However their wide-scale reusability is still impeded by the lack of standards for documenting and annotating ontologies with metadata information, which is in the same time a pre-requisite for the realization of fully-fledged ontology repositories on the Web. At present, Semantic Web ontologies are poorly documented (at most), while additional clarifying information is spread across the Web in various forms, thus not being optimally available to potential ontology users. Further on, current repositories restrict to a minimal set of search and navigation services, usually offering a simple Web interface to the ontological resources. Their limited retrieval functionality is significantly influenced by the absence of metadata information about the administrated ontologies.

A first step towards the alleviation of this situation is the development of a feasible metadata schema for the systematic description of ontologies. Represented in a machine-processable form this schema would provide a basis for a more effective access and exchange of ontologies across the Web. However, in order for it to represent real added value to ontology users, it is equally important to achieve a common agreement on the terms used to describe ontologies, and to provide appropriate tools being able to create, manage and distribute metadata information.

Our contribution consists of the **DEMO** (**D**esign **E**nvironment for **M**etadata **O**ntologies), framework for the development and deployment of an ontology metadata vocabulary. The main part of the framework is the **OMV** model (**O**ntology **M**etadata **V**ocabulary), which is intended to capture reuse-relevant information about ontologies (i.e. ontology metadata) in a machine-understandable form. Further on, the framework is concerned with methodologies, methods and tools to support the dissemination of OMV towards a *representative* metadata vocabulary, which reflects the needs of an expanding community of ontology users w.r.t. ontology reuse. In particular, DEMO focuses on the evolution and extension of OMV ; we introduce a technical, methodological and organizational setting to allow interested parties to effectively contribute to this

initiative and foresee extensions of the core model for more specific reuse activities (such as the evaluation of ontologies, their engineering process etc).

The remainder of this paper is organized as follows: we introduce DEMO, its goals, contents and realization in Section 2. The main components of the framework are described in Sections 3 to 5. Section 3 addresses the development process of the current OMV metadata model. Further on, it accounts for the procedure currently applied to operationalize the evaluation and refinement of the OMV core on the basis of large scale agreement discussions with partners of the EU Network of Excellence KnowledgeWeb.¹ The contents of the metadata schema is detailed in Section 4, while extension modules, which focus on a particular topic or activity of the reuse process, are discussed in Section 5. Our approach is compared against related work in the area of ontology reuse and metadata in Section 6. We conclude with a brief description of the limitations of the current approach and sketch the planned future work (Section 7).

2 DEMO - A Design Environment for Metadata Ontologies

Developing and maintaining an ontology metadata vocabulary is a long-term, resource-intensive process for every involved participant, as it requires a well-defined operational structure and appropriate technological support. The current state of the OMV model is the result of an intensive collaboration among many Semantic Web experts affiliated to academia or industry. Our experiences in the development of the OMV so far (see Section 3), consolidated by general engineering guidelines recommended by the majority of ontology engineering methodologies currently available, indicated the need of an environment which provides support to systematically realize and promote ontology metadata vocabularies for the Semantic Web.

DEMO (**D**esign **E**nvironment for **M**etadata **O**ntologies) aims at providing this kind of support at *organizational*, *methodological* and *technological* level. The mission of the framework can be categorized as follows:

- Provision of an *organizational infrastructure* for the development and maintenance of a commonly agreed metadata vocabulary for ontologies. In particular this includes the facilitation of equitable participation mechanisms for the organizations involved in the DEMO activities.
- Identification and application of suitable methodologies and technologies to support the complete life cycle of the OMV Core.
- Development and maintenance of the OMV Core.
- Promotion of OMV extensions relying on the OMV Core.
- Provision of an appropriate technical infrastructure for the enumerated activities.

DEMO aims to establish and ensure an efficient engineering process and application of the proposed ontology metadata vocabulary. For usability and extendability reasons, DEMO distinguishes between the OMV Core and OMV extensions. The former provides information about the core metadata vocabulary which should be sufficient for an efficient reuse and access of ontologies in the Semantic Web (see Section 3). For

¹ <http://knowledgeweb.semanticweb.org/>

specific applications or differentiated aspects of ontologies, e.g. detailed evolution information of an ontology, it foresees the development and usage of OMV extension modules. This flexible mechanism allows on the one hand all participants to engineer a base vocabulary (i.e. the OMV Core) and on the other hand, to provide more detailed information by developing domain, task or application-specific vocabularies (i.e. OMV Extensions).

The framework is divided into several functional components in accordance to the aforementioned objectives:

- The **Engineering Component** is responsible for the development and maintenance of the OMV Core.
- The **Evolution Component** is authorized to perform changes on the OMV Core according to the requirements of the OMV users.
- The **Extensions Component** coordinates the realization of extension modules.
- The **Applications Component** is responsible for the propagation of the OMV results to new application scenarios, demonstrating the usability of the metadata standards in these applications and contributing to their evaluation in real-world settings.

From an organizational perspective, DEMO activities are driven and supervised by the **Management Board (MB)**, consisting of representatives from the **OMV Consortium**, which includes all active OMV contributors. A central organization objective of DEMO is to keep the barrier low for participants to join the OMV Consortium and to get involved in the development and recommendation process. Complementary to this distinction, DEMO foresees several **Working Groups (WG)** corresponding to the aforementioned components (WG Engineering, WG Evolution, WGs Extensions, WGs Applications), which are further organized in working group board and members.

DEMO provides the technical means required for metadata management and maintenance for the Semantic Web in form of the semantic engineering platform OntoWare² which provides a scalable, collaborative software and ontology engineering environment for the collaborating partners.

3 The Development of the OMV Core Ontology

The OMV core ontology was realized in accordance to the present research achievements in Ontology Engineering, a field in which several elaborated methodologies have already proved their applicability in real-world situations (refer to [3] for a description of and a comparison among the most relevant ones). The development process is performed in five stages, which are described in the remaining of this section.

3.1 Requirements Analysis

In this step we elaborated an inventory of requirements for the metadata model as a result of a systematic survey of the state of the art in the area of ontology reuse. Besides analytical activities, we conducted extensive literature research, which focused

² c.f. <http://ontoware.org>

on theoretical methods [16, 4, 12], but also on case studies on reusing existing ontologies [22, 18, 15], in order to identify the real-world needs of the community w.r.t. a descriptive metadata format for ontologies. Further on, the requirements analysis phase was complemented by a comparative study of existing (ontology-independent) metadata models and of tools such as ontology repositories and libraries (implicitly) making use of metadata-like information. An overview of the results of this study is given in Section 6. The main requirements identified in this process step are the following:

Accessibility: Metadata should be accessible and processable for both humans and machines. While the human-driven aspects are ensured by the usage of natural language concept names, the machine-readability requirement can be implemented by the usage of Web-compatible representation languages (such as XML or Semantic Web languages, see below).

Usability: This requirement states for the necessity of building a metadata model which 1). reflects the needs of the majority of ontology users, as reported by current case studies in ontology reuse, but in the same time 2). allows proprietary extensions and refinements in particular application scenarios. The realization of the latter is further discussed in Section 5. From a content perspective, usability can be maximized by taking into account multiple metadata types, which correspond to specific viewpoints on the ontological resources and are applied in various application tasks. Despite the broad understanding of the metadata concept and the use cases associated to each definition, several key aspects of metadata information have already established across computer science fields [14]:

- **Structural metadata** relates to statistical measures on the graph structure underlying an ontology. In particular we mention the number of specific ontological primitives (e.g. number of classes, instances). The availability of structural metadata influences the usability of an ontology in a concrete application scenario, as size and structure parameters constraint the type of tools and methods which are applied to aid the reuse process.
- **Descriptive metadata** relates to the domain modelled in the ontology in form of keywords, topic classifications, textual descriptions of the ontology contents etc. This type of metadata plays a crucial role in the selection of appropriate reuse candidates, a process which includes requirements w.r.t. the domain of the ontologies to be re-used.
- **Administrative metadata** provides information to help manage ontologies, such as when and how it was created, rights management, file format and other technical information.

Interoperability: Similarly to the ontology it describes, metadata information should be available in a form which facilitates metadata exchange among applications. While the syntactical aspects of interoperability are covered by the usage of standard representation languages (see “Accessibility”), the semantical interoperability among machines handling ontology metadata information can be ensured by means of a formal and explicit representation of the meaning of the metadata entities, i.e. by conceptualizing the metadata vocabulary itself as an ontology.

Separation between Knowledge and Implementation Levels: In accordance to the recommendations of current ontology engineering methodologies it should be

clearly distinguished between the conceptual model of an ontology and particular implementations (in various languages, syntaxes, versions etc.). The realization of this criterion is illustrated in Section 4.

3.2 Categorisation

On the basis of the aforementioned analysis we designed the core structure of the metadata model in terms of classes and properties/attributes of these classes. In order to increase the usability of the model w.r.t. its extendability we assigned the metadata entities to three usage categories (in the style of XML Schema):

- **Required:** These metadata facts are mandatory. Missing elements lead to incomplete metadata descriptions of ontologies and are handled accordingly by metadata management tools.
- **Optional:** The specification of optional metadata elements, though not mandatory, increases the reusability of the corresponding ontology.
- **Extensional:** This class of metadata elements is not represented in detail in the core model, but can be further elaborated in extension modules (see Section 5).

Further on, every metadata entity was labelled in accordance to a predefined naming schema and carefully documented.

3.3 Implementation

Due to the high accessibility and interoperability requirements, as well as the nature of the metadata, which is intended to describe Semantic Web ontologies, the conceptual model designed in the previous step was implemented in the OWL language. An implementation as XML-Schema or DTD was estimated to restrict the functionality of the ontology management tools using the metadata information (mainly in terms of retrieval capabilities) and to impede metadata exchange at semantical level. Further on, a language such as RDFS does not provide a means to distinguish between required and optional metadata properties. The implementation was performed manually by means of a common ontology editor.

3.4 Evaluation

The evaluation of the first draft of the metadata model was conducted in two parallel phases: on one hand, the content of the model was subject to human-driven evaluation w.r.t. the inventory of the included metadata elements, their meaning and labelling. On the other hand the usability of the proposed OMV was tested in several application settings.

The content-based evaluation was performed by conducting interviews with a group of experts in the area of Ontology Engineering. Considering that the people best placed to give a comprehensive assessment of the ontology metadata vocabulary are currently researchers being directly involved in theoretical or practical issues of Ontology Engineering, we organized an expert group of four academics affiliated in this community and in the EU Network of Excellence KnowledgeWeb which evaluated the model against a pre-defined set of criteria[5]:

- **Consistency:** this criterion refers to the existence of explicit or implicit contradictions in the represented ontological content.
- **Completeness:** according to [5], an ontology is complete if it (explicitly or implicitly) covers the intended domain.
- **Conciseness:** complementary to the previous feature, conciseness states for the redundancy-free representation of the application domain of an ontology and for the avoidance of useless definitions.
- **Expandability/Sensitiveness:** the criterion refers to the possibility of adding new definitions to the ontology without altering the existent content.

The aforementioned evaluation framework was extended with two dimensions: **readability**, which accounts for the usage of intuitive labels to denominate the OMV entities, and **understandability**, which mainly relates to the quality of the documentation of the metadata model.

The evaluation resulted in changes on both conceptual and implementation levels of the OMV ontology. In the following we summarize the key aspects of the evaluation process:

- **Consistency:** according to human judgement and to the automatic consistency checking no inconsistencies were found.
- **Completeness:** during the evaluation the participants identified several aspects which were missing in the initial draft of OMV . For instance, information about the representation language of an ontology (syntax, representation paradigm etc.) were found to be insufficiently covered by OMV . As a result we introduced concepts such as `RepresentationParadigm` and `OntologyRepresentationLanguage` to account for these aspects. Further on, a classification of tasks ontologies are typically designed for was included in form of a root concept `OntologyTask` and more specialized sub-concepts such as `SemanticSearch` or `SemanticAnnotation`.
- **Conciseness:** parallel to extending the ontology, the experts expressed their concerns w.r.t. a series of concepts which were too specific for a core metadata vocabulary. Most of these concepts related to particular aspects of the engineering process in which the corresponding ontology was originally created and of the evaluation of the ontology (the concepts `OntologyReviews`, `OntologyReviewer` etc.). These aspects were removed from the OMV core and transferred to the OMV extensions (see Section 5).
- **Readability:** the naming of particular metadata entities was one of the most challenging parts of the evaluation process. The experts proposed alternative names for several fundamental OMV concepts, such as those representing the conceptual model and the implementation of an ontology, respectively (see below). Finally an agreement was achieved with the result that the original names of these two metadata entities were changed to `Conceptualization` and `Implementation`, respectively. Further on, the experts indicated the poor readability of abbreviated concept labels, which were modified accordingly.
- **Understandability:** the experts evaluated the OMV model favorably. They were able to easily understand its scope, content and limitations and expressed their confidence in its usability.

In summary, the results of the expert-driven evaluation significantly contributed to the quality of the OMV ontology, confirming our expectations towards the realization of a standardized metadata schema for Semantic Web ontologies. However, the evaluation process has already pointed out the main challenge of our approach, which is related to the achievement of a common agreement in a large community of ontology users w.r.t. their requirements and perceptions about ontology metadata. This issue is addressed in the next section.

3.5 Evolution

As aforementioned the real added value of the proposed metadata vocabulary is fundamentally determined by the representativeness of its content and its dissemination across the Semantic Web community. For this purpose one of the foci of the DEMO environment is the provision of an organizational and methodological setting, which allows the OMV consortium to participate at the metadata development initiative.

For the realization of this goal we take advantage of the results already available in the Ontology Engineering community w.r.t collaboratively building ontologies in distributed environments. Based on the long-standing tradition of argumentation and conflict mediation research in Knowledge Management, Agent-based Computing or Linguistics, approaches such as [17, 11, 19] provide a deep analysis of the challenges of such engineering settings and means to operationalize it at process and technology level. They describe the organizational setting of evolving multi-site ontology development processes and propose an inventory of tools which can be used to optimize the achievement of commonly accepted agreements w.r.t. ontology modelling decisions.

For the collaborative refinement of the current OMV Core we decided to apply the DILIGENT methodology[17], because, compared to alternative approaches, it provides a more fine-grained description of the underlying process model, whose validity has been tested in several case study.

The organizational structure of the OMV consortium is fully compatible with the recommendations of the aforementioned methodology. The working group responsible for the OMV evolution (WG Evolution, see Section 2) is divided into a working group board and its members. As foreseen by the DILIGENT methodology[17],³ a first version of the shared ontology (in our case the OMV Core ontology described in this paper) was already distributed to the members of the working group, which will submit their change requests accompanied by the arguments justifying them in regular time spans to the working group board. The board analyzes the requests on the basis of their arguments, decides upon the changes and releases a new version of the OMV Core, which is analyzed by the rest of the participants. The tools supporting the evolution activity are provided by the OntoWare platform.

The evolution process aiming at realizing a representative metadata vocabulary for Semantic Web ontologies is still in its infancy. However, we are confident that, given the positive feedback received during the development of the current OMV version and the willingness of many academia and industrial institutions to get involved in this process, the proposed OMV will evolve towards a high-quality metadata standard.

³ Due to space considerations, we restrict to a minimal description of the process model in this paper.

In the following we turn to a more detailed description of the current OMV core ontology, while OMV extensions are addressed in Section 5.

4 The Ontology Metadata Vocabulary - OMV Core

As the result of the requirements distilled from comprehensive literature and case study research we designed a first version of the OMV core ontology, which is subject of a recently initiated evolution process (see Section 3). The main components of the ontology, manually implemented in OWL, are presented in the remaining of this section.

4.1 Conceptualization vs. Implementation

OMV distinguishes between the **ontology conceptualization** and an **ontology implementation** as concrete realization of an ontology in a particular representation language. This separation is based on the observation that any ontology is based on a language-independent conceptual model. The conceptualization represents the view of the engineering team upon the application domain, which then is implemented using an ontology editor and stored in a specific format. The same conceptualization might result in several implementations, with various classes, properties and axioms, depending on the concrete representation paradigm, language and syntax. Therefore we define:

- **Ontology Conceptualization:** An *Ontology Conceptualization (OC)* represents the abstract or core idea of an ontology. It describes the core properties of an ontology, independent from any implementation details.
- **Ontology Implementation:** An *Ontology Implementation (OI)* represents a specific implementation of a conceptualization. Therefore, it describes implementation-specific properties of an ontology.

The distinction between OC and OI leads to an efficient mechanism, for tracking multiple ontology versions, as well as for different representations of one knowledge model in different languages. Technically, the two are modelled as separate classes connected by means of the relation `realizes`. This means that there may be many possible ontology implementations for one conceptualization, but one ontology implementation can only implement one conceptual model.

An instance of the OI class should not be able to exist without a corresponding conceptualization. However, for practical reasons, we allow the existence of the two independently of each other. We cannot assume that every existing ontology will be annotated by its original author who might have created the underlying conceptualization.

An excerpt of the main classes and properties of the OMV core ontology are illustrated in figure 1. The ontology is available for download in several ontology formats.⁴

It should be noticed that there exist several properties defined at the classes OC and OI which are denominated with similar names, but have a different semantics. Consider, for instance, an ontology engineer developing an ontology in OWL using the RDF/XML syntax and annotating it with OMV. The values of the properties of the corresponding OC and OI individuals definitely overlap to some extent (e.g., both would have the

⁴ OMV representations are available at <http://ontoware.org/projects/omv/>

4.2 Basic Classes of OMV

The OMV focuses on the two main classes OC and OI for representing core information about ontologies. However, additional classes are required to adequately represent ontology-related information, especially in the context of the Semantic Web (Figure 1).

Common ontology engineering assumes that ontologies are created and owned by Person(s) or Organizations. We group these two classes under the generic class Party by a subclass-of relation. A Party can have several locations by referring to a Location and can *create* or *contribute* to a *OntologyConceptualization* or *OntologyImplementation*, respectively. Tools such as ontology editors can be referred to by the class *OntologyEngineeringTool*, which itself can be developedBy a Party. The different existing syntactical representations and ontology languages are representable by *OntologySyntax*, *OntologyLanguage* and *RepresentationParadigm*. OMV further contains the class *OntologyEngineeringMethodology* which makes explicit the methodology (or methodologies) used during the engineering process. Ontologies might be categorized according to various dimensions[6]. Those types are modelled as sub-classes of *OntologyType*. For commercial settings it might be relevant to propose usage licenses which can be realized by the class *LicenseModel* relating to each *OntologyConceptualization* or *OntologyImplementation*.

Exemplary, the SWRC ontology project is applying OMV to annotate their Semantic Web Research Community Ontology [21]. The core idea of the SWRC ontology is to model entities of research communities and their relationships. This conceptualization of SWRC (core idea) originated several different implementations of SWRC. As a first result, OMV increases the transparency within the engineering process and supports the dissemination due facilitating reuse and discovery of SWRC⁵.

5 The Development of OMV Extensions

The OMV Core is intended to represent a commonly agreed metadata for the Semantic Web. In contrast to that, we are aware that for specific domains, tasks or communities extensions in any direction might be required. These extensions should be compatible to the OMV Core, but in the same time fulfill the requirements of a domain, task or community-driven setting. In order to ensure the compatibility to the commonly agreed core, one of the goals of DEMO is the elaboration of a procedure to supervise and promote the creation of new OMV extension modules.

An OMV extension can be originated by members of the **OMV Working Group Extensions**. The group provides deliverables documenting their work for the public. Initially, it creates a charter, nominates a responsible chair and sets up a technical infrastructure, for example a Web site and mailing-list. Such a charter describes the initial motivation and explains specific requirements for a planned OMV extension. The character of an OMV extension is a metadata ontology itself which imports the OMV core ontology. There are no restricting modelling guidelines to be met. However DEMO provides a basic inventory of design decisions and guidelines, which are recommended to

⁵ The ontology and metadata example are available at <http://swrc.ontoware.org/>.

be applied for the extension modules[8]. The engineering team of an extension ontology might resort to the same engineering methodologies and tools as those applied for the OMV Core (see Section 3).

The first DEMO working group on OMV extensions has already been initiated. It aims at the realization of an OMV extension module on the topic Ontology Evaluation. Its results are intended to be applied in the EU Network of Excellence KnowledgeWeb as a support tool for the OOA (Ontology Outreach Advisory) initiative, whose goal is to provide consultancy, promote and outreach high-quality ontological content in key industrial sectors of the Semantic Web such as eHealth or eRecruitment.⁶ From an ontology engineering perspective, the working group elaborated a first draft of a metadata schema relying on the OMV Core, which describes various aspects of the ontology evaluation field: typical methods applied to evaluate an ontology from multiple points of view (e.g. consistency checking, validation, requirements-based evaluation, ontological evaluation) and in particular application scenarios (e.g. NLP-based evaluation methods), roles involved in the evaluation process (e.g. reviewer), tools (e.g. reasoners, validators). As in the case of the OMV Core, the evaluation module is currently subject of revisions within the members of the working group.

6 Related Work

Metadata and metadata standards have a long-tradition in a variety of computer sciences areas, such as digital libraries or data management and maintenance systems. We will briefly mention related metadata standards, including in particular those ones relevant for the Semantic Web. The **Dublin Core (DC)** metadata standard is a simple yet effective element set for describing a wide range of networked resources⁷. It includes two levels: Simple (with fifteen elements) and Qualified, including an additional element as well as a group of element refinements (or qualifiers) that adapt the semantics of the elements for resource discovery purposes. The **Reference Ontology**[1] is a domain ontology that gathers, describes and has links to existing ontologies. However its focus is to characterize ontologies from the user point of view, and provides only a list of property-value pairs for describing ontologies. The Semantic Web search engine **SWOOGLE**[2] makes use of an implicitly defined metadata schema, which covers information which can be extracted automatically from ontology implementations. Our approach includes and extends this metadata vocabulary. Ideally, future versions of SWOOGLE would also take into account the additional vocabulary defined in OMV. Further on, the issue of creating a metadata standard for ontologies is addressed by various ontology repositories initiatives. However, the majority of these repositories rely on a restricted, implicitly declared vocabulary, whose meaning is not machine-understandable. The **DAML ontology library** provides a catalog of DAML ontologies that can be browsed by different properties⁸. The **FIPA ontology service**[20] defines an agent wrapper of open knowledge base connectivity. The **SchemaWeb Directory**

⁶ <http://knowledgeweb.semanticweb.org/>

⁷ c.f. <http://dublincore.org/>

⁸ c.f. <http://www.daml.org/ontologies/>

is a repository for RDF schemas expressed in RDFS, OWL and DAML+OIL⁹. Finally we mention the ontology metadata example presented in [10] emerged within the EU Network of Excellence **Knowledge Web**. The metadata consists only of attribute-value pairs, and does not consider the distinction between conceptualizations and implementations. However, the work presented there provided a preliminary basis for the OMV ontology introduced in this paper.

7 Conclusions and Future Work

In this paper we present DEMO, a framework for the development and deployment for ontology metadata. Besides OMV, the proposed core vocabulary for ontology metadata, the framework comprises an inventory of methods to collaboratively extend OMV in accordance to the requirements of a emerging community of industrial and academia users, to develop extension modules for particular applications, user communities or aspects of the reuse process, as well as tools for metadata management.

The proposed OMV Core captures information that is similar to other metadata standards, such as Dublin Core. However, it goes beyond this general-purpose level and provides a vocabulary for capturing information about **ontologies**, represented in a well-defined machine and human interpretable language. The differences between arbitrary information sources and ontologies make the usage of metadata standards such as Dublin Core inappropriate. When talking about ontologies, there is a distinction between the conceptual representation of an application domain (the ontology at knowledge level) and its various implementations (in particular representation languages). As these two parts are characterized by different properties, the metadata about ontologies should be able to differentiate between the semantic conceptualization and its particular realization as a concrete ontology document. Besides, aspects related to application scenario, scope, purpose, or evaluation results are essential coordinates for a successful ontology reuse and should be captured by the ontology metadata schema. On the other hand, besides structural and technical information on ontologies—which can be captured automatically—there is a strong demand for representing descriptive metadata, like authorship information, categorizations or underlying methodologies. The enumerated factors indicate the need for a ontology-specific metadata vocabulary, which, though remaining compatible to information represented in generic metadata standards like Dublin Core, is customized to the particular requirements of ontology sharing and reuse.

Within the DEMO environment we initiated an activity aiming at achieving a broad scale agreement on the OMV contents and representation, which was received favorably by both academia and industry institutions and will continue in the future. As a first result of DEMO activities, the *WG Applications* developed the P2P metadata sharing tool Oyster¹⁰ and the metadata portal ONTHOLOGY¹¹. Both systems compose an interlocked metadata management framework and contribute to the development and dissemination activities [9].

⁹ c.f. <http://www.schemaweb.info>

¹⁰ c.f. <http://oyster.ontoware.org/>

¹¹ c.f. <http://www.onthology.org/>

While the basis technical infrastructure for the DEMO activities is provided by the OntoWare platform, we are experimenting with methods and heuristics to operationalize the metadata generation process and to check the quality of the existing metadata information. These tasks, though not trivial, can be automatized to a considerable extent due to the ontology-based representation of the OMV .

Acknowledgments. This proposal is based on a huge number of discussions and many helpful arguments by persons from academia and industry. Especially we would like to thank our colleagues York Sure (AIFB), M. Carmen Suárez-Figueroa (UPM), Peter Haase (AIFB), Denny Vrandečić(AIFB) and Rudi Studer (AIFB). Furthermore, we thank our partners from the EU project Knowledge Web for their present and future collaboration.

References

1. J. Arpírez, A. Gomez-Perez, A. Lozano-Tello, and H. Pinto. Reference Ontology and (ONTO)2 Agent: The Ontology Yellow Pages. *Knowledge and Information Systems*, 2:387–412, 2000.
2. L. Ding et al. Swoogle: A search and metadata engine for the semantic web. In *Proc. of the 13th ACM Conf. on Information and Knowledge Management*, pages 58–61, 2004.
3. M. Fernandez-Lopez and A. Gomez-Perez. Overview and analysis of methodologies for building ontologies. *Knowledge Engineering Review*, 17(2):129–156, 2002.
4. A. Gangemi, D. M. Pisanelli, and G. Steve. An overview of the ONIONS project: Applying ontologies to the integration of medical terminologies. *Data Knowledge Engineering*, 31(2):183–220, 1999.
5. A. Gomez-Perez. Evaluation of ontologies. *Int. Journal of Intelligent Systems*, 16(3), 2001.
6. A. Gómez-Pérez, M. Fernández-López, and O. Corcho. *Ontological Engineering*. Springer, 2003.
7. T. R. Gruber. Toward principles for the design of ontologies used for knowledge sharing. *Int. J. Hum.-Comput. Stud.*, 43(5-6):907–928, 1995.
8. J. Hartmann and R. Palma. OMV - Ontology Metadata Vocabulary for the Semantic Web, 2005. Technical Report V. 1.0, available at <http://omv.ontoware.org/>.
9. J. Hartmann, Y. Sure, R. Palma, P. Haase, M.C. Suarez-Figueroa, R. Studer, and A. Gomez-Perez. Ontology metadata vocabulary and applications. In Robert Meersman, editor, *International Conference on Ontologies, Databases and Applications of Semantics. In Workshop on Web Semantics (SWWS)*, OCT 2005.
10. KnowledgeWeb European Project. Identification of standards on metadata for ontologies (Deliverable D1.3.2 KnoweldgeWeb FP6-507482), 2004.
11. K. Kotis, G. A. Vouros, and J. Padilla Alonso. HCOME: tool-supported methodology for collaboratively devising living ontologies. In *SWDB'04: 2. Int. Workshop on Semantic Web and Databases*, 2004.
12. A. Lozano-Tello and A. Gomez-Perez. ONTOMETRIC: A Method to Choose the Appropriate Ontology. *Journal of Database Management*, 15(2), 2004.
13. R. Neches, R. E. Fikes, T. Finin, T. R. Gruber, T. Senator, and W. R. Swartout. Enabling technology for knowledge sharing. *AI Magazine*, 12(3):35–56, 1991.
14. National Information Standards Organization. Understanding metadata. NISO Press, 2004.
15. E. Paslaru Bontas, M. Mochol, and R. Tolksdorf. Case Studies on Ontology Reuse. In *Proceedings of the IKNOW05 International Conference on Knowledge Management*, 2005.

16. H. S. Pinto and J. P. Martins. A methodology for ontology integration. In *Proc. of the International Conf. on Knowledge Capture K-CAP01*, 2001.
17. H. S. Pinto, S. Staab, and C. Tempich. Diligent: Towards a fine-grained methodology for distributed, loosely-controlled and evolving engineering of ontologies. In *Proc. of the ECAI04*, pages 393–397, 2004.
18. T. Russ, A. Valente, R. MacGregor, and W. Swartout. Practical Experiences in Trading Off Ontology Usability and Reusability. In *Proc. of the Knowledge Acquisition Workshop (KAW99)*, 1999.
19. S. Buckingham Shum, E. Motta, and J. Domingue. Augmenting design deliberation with compendium: The case of collaborative ontology design. In *Proc. of the HypACoM02 Workshop: Facilitating Hypertext-Augmented Collaborative Modeling*, 2002.
20. H. Suguri et al. Implementation of FIPA Ontology Service. In *Proc. of the Workshop on Ontologies in Agent Systems, 5th Int. Conf. on Autonomous Agents Montreal, Canada*, 2001.
21. Y. Sure, S. Bloehdorn, P. Haase, J. Hartmann, and D. Oberle. The SWRC ontology - Semantic Web for Research Communities. In Carlos Bento, Amilcar Cardoso, and Gael Dias, editors, *Proceedings of the 12th Portuguese Conference on Artificial Intelligence - Progress in Artificial Intelligence (EPIA 2005)*, volume 3803 of *LNCS*, pages 218 – 231, Covilha, Portugal, DEC 2005. Springer.
22. M. Uschold, M. Healy, K. Williamson, P. Clark, and S. Woods. Ontology Reuse and Application. In *Proc. of the Int. Conf. on Formal Ontology and Information Systems FOIS98*, 1998.