The SWRC Ontology – Semantic Web for Research Communities

York Sure, Stephan Bloehdorn, Peter Haase,
Jens Hartmann, and Daniel Oberle

University of Karlsruhe, Institute AIFB, D-76128 Karlsruhe, Germany
http://www.aifb.uni-karlsruhe.de/WBS
{sure,bloehdorn,haase,hartmann,oberle}@aifb.uni-karlsruhe.de

Abstract. Representing knowledge about researchers and research communities is a prime use case for distributed, locally maintained, interlinked and highly structured information in the spirit of the Semantic Web. In this paper we describe the publicly available 'Semantic Web for Research Communities' (SWRC) ontology, in which research communities and relevant related concepts are modelled. We describe the design decisions that underlie the ontology and report on both experiences with and known usages of the SWRC Ontology. We believe that for making the Semantic Web reality the re-usage of ontologies and their continuous improvement by user communities is crucial. Our contribution aims to provide a description and usage guidelines to make the value of the SWRC explicit and to facilitate its re-use.

1 Introduction

One of the driving forces of the Semantic Web is the need of many communities to put machine-understandable data on the Web which can be shared and processed by automated tools as well as by people. Representing knowledge about researchers, research communities, their publications and activities as well as about their mutual interrelations is a prime use case for distributed, locally maintained, interlinked and highly structured information in the spirit of the Semantic Web.

The SWRC ontology – initially phrased *Semantic Web Research Community Ontology* – which we will describe in this paper, generically models key entities in a typical research community and reflects one of the earliest attempts to put this usage of Semantic Web Technologies in academia into practice.

The SWRC ontology initially grew out of the activities in the KA^2 project [1]. By then it was already used in the context of Semantic Community Web Portals [2] and has been ported to various knowledge representation languages including both Semantic Web Standards like RDF(S) or DAML+OIL and other languages like F-Logic consecutively. In the most recent versions, the SWRC ontology has been released in OWL format[1]. Since its initial versions it has been used and adapted in a number of different settings, most prominently for providing structured metadata for web portals. These include the web portal of the authors' institute AIFB and for the portals in the research projects OntoWeb and SemIPort. These and other usages of the ontology in different settings will be described later on.

In this paper, we will focus on describing the SWRC ontology and on making the design considerations explicit that have led to a particular modelling approach. We show a number of typical modelling problems and report on their solutions. While some of the issues that arose in modelling the SWRC ontology are domain specific –

[1] The SWRC ontology itself and some of its extensions are available via http://ontoware.org/projects/swrc/

sometimes also specific to the chosen modelling language, e.g. OWL– others appear to be more general and may thus serve as a handy reference for knowledge engineers. At the same time this paper aims at providing usage guidelines for the SWRC ontology.

The remainder of this paper is structured as follows. Section 2 gives an initial overview over the ontology itself and the modelled domain. Section 3 reviews the critical design considerations made and discusses alternatives. Section 4 lists a number of guidelines for users and systems working with the SWRC ontology while Section 5 reports on three prototypical usages of the ontology. Finally, Section 6 reviews related schemas and discusses their relation to the SWRC ontology. We conclude in Section 7.

2 Overview of the Ontology

The SWRC ontology generically models key entities relevant for typical research communities and the relations between them. The current version of the ontology comprises a total of 53 concepts in a taxonomy and 42 object properties, 20 of which are participating in 10 pairs of inverse object properties. All entities are enriched with additional annotation information.

SWRC comprises at total of six top level concepts, namely the Person, Publication, Event, Organization, Topic and Project concepts. Figure 1 shows a small portion of the SWRC ontology with its main top-level concepts and relations. The Person concept models any kind of human person and a large number of properties restrict their domain or range to individuals of this concept like studiesAt or author, respectively. The Person concept is specialized by a large number of – not necessarily disjoint – subconcepts, e.g. Employee, Student and the like. The Event concept is meant to model different types of events and is thus specialized by a wide range of concepts including events like Lecture or Conference. The Publication concept subsumes all different types of research publications modelled in close correspondence with the well known BibTeX publication types like Article or InProceedings. The Organization and Project concepts model more abstract concepts like the subconcepts Department or SoftwareProject, respectively. Both concepts can participate in a large number of relations like for example to the Person concept via the employs or member relations. The Topic captures arbitrary topics of interest which are arranged on the instance level via the subTopic relation.

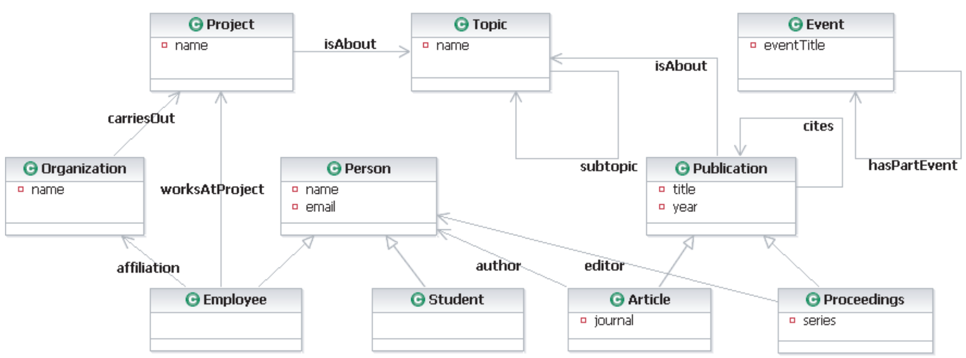

Fig. 1. Main Concepts of the SWRC Ontology

3 Design Considerations

In this section we outline and justify major design decisions that have been made in the development of the SWRC ontology. By making them explicit we facilitate the proper usage of the SWRC but also provide decision support for other modelling efforts.

3.1 Choice of Language

As of today, many different ontology representation languages embodying various paradigms exist, including description logics, frame-based and logic-programming languages etc. Apparently, some of the reasons why there are competing ontology representation languages lie in the differing requirements imposed by the respective application scenarios to which semantic technologies are being applied. Simpler languages are easier and more efficient to deal with but lack the complex modelling facilities of the richer paradigms. Some scenarios may require particular modelling facilities or even tools which are only available for certain paradigms.

For applying semantic technologies, the practitioner is faced with the question, which ontology modelling paradigms should be adopted. For performance reasons, simpler paradigms are often preferable, but equally important is interoperability between different systems and reusability in the future, i.e. compatibility with to be established standards.

When the SWRC ontology was first created, no widely accepted and standardized format for representing ontologies existed. Therefore, the SWRC was initially made available in various formats, including for example F-Logic and DAML+OIL. In 2004, the Web Ontology Language OWL [3, 4] was adopted by the W3C as the standard for representing ontologies on the web. OWL is a very expressive ontology representation language which can be described as a fragment of first-order logic. It comes in three different variants, namely OWL-Lite, OWL-DL and OWL-Full, with differences in expressiveness and reasoning complexity. As compatibility with standards was one of our major design goals, we decided to model the SWRC ontology in an OWL-compliant manner.

In particular, we decided for a subset of the OWL language, called Description Logic Programs (OWL-DLP)[2] [5, 6]. We argue that OWL-DLP provides a basic ontology modelling paradigm which meets most of the requirements above while being a flexible choice for future developments, as it is not only a proper fragment of OWL, but also of logic programming languages such as F-Logic. As for performance, OWL-DLP features polynomial data complexity and exptime combined complexity, which renders it to be far better than the more expressive languages we mentioned. Extensions made for either more general language can be adopted for the fragment in a straightforward manner. Modelling and reasoning tools available for OWL or F-Logic can naturally deal with OWL-DLP, and interoperability is guaranteed to the largest extent possible.

For modelling of the above mentioned domain we found the primitives offered by OWL-DLP sufficiently expressive. However, since some natural support for e.g. sequences of authors we had to find alternative solutions which are explained in the following subsections.

3.2 Representing Entities as Objects or Data Values

A common question that occurs frequently while modeling data is whether certain entities should be modelled as "first class objects" or dependent of some other object, e.g. in the form of an attribute.

[2] see `http://logic.aifb.uni-karlsruhe.de`.

This issue also applies to the modelling of ontologies. For example, many bibliographic models simply provide a concept to represent publications, and model all properties of publications, e.g. their authors, corresponding organizations, topic classifications etc. as attributes. While such a model has advantages because of its simplicity, it is not easily extensible and also does not make important relations between entities explicit. We have therefore chosen to introduce explicit concepts wherever possible to make the semantics of the model explicit.

3.3 Representing Sequences of Authors

The choice of OWL-DLP as the ontology languages with its foundation in first order logic provides no native support to model sequences, lists, or any data structure requiring ordering of entities in an extension. Instead, all extensions of concepts and properties have the semantics of sets. However, sometimes it is necessary to capture the ordering of elements, for example the order of authors of a publication is important.

While RDF provides a container model in its data model, which allows to describe bags, sequences and alternatives, this container model is only available in OWL-Full. An alternative would be to model a ternary relation of publication, person, and order. This ternary relation can be reduced to binary relations by introducing a new concept, e.g. Authorship. However, this significantly increases complexity and is not intuitive.

Here, our design decision was to follow the basic idea of the RDF container model, but to capture it in a way compatible with OWL-DLP. RDF uses the predefined property `<rdf:li>` to denote the membership of a container. The order of the members is defined by subproperties of `<rdf:li>`, i.e. `<rdf:_1>`, `<rdf:_2>`,... to denote the first, second member and so on. Similarly, for applications that need to express the order of authors, we use subproperties of author, i.e. author_1, author_2,... Applications not interested in the order still can use the author property, as it subsumes the newly introduced properties.

3.4 Representation of Topic Classification

A key feature of bibliographic metadata is the classification against established topic hierarchies. In a sense, topic hierarchies can be seen as light-weight ontologies by themselves that complement the SWRC ontology with domain specific models of particular research domains. One example is the ACM topic hierarchy for the Computer Science domain. Further, many other topic hierarchies for other domains exist, for which an integration with the SWRC ontology is desirable. Here, a standard way of representing topics is required. Several options for such a representation exist. Some of these options model topics as concepts in the ontology, others require a feature called meta-modelling in the ontology language, where concepts can also be used as instances (cf. [7] for a detailed comparison), leading to an OWL-Full ontology.

In alignment with the choice of our ontology language, which requires a strict separation between concepts and instances, we model topics as instances of the concept Topic. The instances of Topic are arranged hierarchically by means of the subTopic relation. The relation between publications and a topic is established via the isAbout property. To link a specific topic hierarchy with the SWRC ontology, the topic concept of the topic hierarchy specializes the ResearchTopic concept of the SWRC ontology, e.g. for the ACM topic hierarchy: $\mathsf{ACMTopic} \sqsubseteq \mathsf{Topic}$.

3.5 Modularization of OWL Ontologies

In general, it is desirable to have a modularized ontology design for mainly two reasons. First, it facilitates reuse of individual ontology modules. Second, maintenance efforts can be decreased by reuse.

In our case, modularization is of central importance, because the different uses of the SWRC ontology require minor variations in modelling. Particular settings might require the addition of Dublin Core attributes or additional concepts and associations. Thus, we factorized a common core from application-specific extensions like shown in Figure 2. The common core facilitates information integration across different applications.

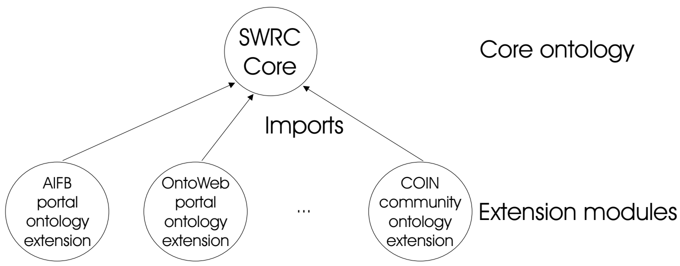

Fig. 2. SWRC modules.

Modularization is realized by the `owl:imports` statement. An `owl:imports` statement references another OWL ontology containing definitions, whose meaning is considered to be part of the meaning of the importing ontology. Each reference consists of a URI specifying from where the ontology is to be imported. Syntactically, `owl:imports` is a property with `owl:Ontology` as its domain and range.[3] In our case, the extension modules import the SWRC core module.

In Section 5 we discuss the different uses of the SWRC ontology. Most of them define their own extension module to specify application-specific idiosyncracies like the AIFB portal ontology module, for instance.

3.6 Pragmatic Versioning

Ontologies are not static entities, but evolve over time. In its history, the SWRC ontology has undergone several changes, and we expect future changes as new requirements arise. With versioning, the old ontologies are kept along with the modified one. It thus provides the ability to relate metadata with a particular version of the ontology. Unfortunately, no standard versioning scheme for ontologies exists yet. We have therefore followed a versioning scheme that has proven practicable for other web data (such as W3C recommendations) already: Identifying versions via namespaces. Here, all versions share a common base URI, appended with a suffix indicating the date of creation. The relation between the versions is established via the OWL annotation property `owl:priorVersion`, which provides a link to the immediate predecessor. The base URI itself refers to the latest version of the ontology. An application not interested in particular versions thus simply relies on the base URI.

[3] `http://www.w3.org/TR/owl-ref/`

4 Guidelines for Usage

In this section we describe guidelines for using the SWRC ontology that evolved out of our experience. First, we elaborate on how to create extensions to the SWRC in order to model the idiosyncracies of particular applications. Second, the generation of identical URIs for identical entities across different SWRC applications requires special care. Finally, we discuss the handling of duplicates.

4.1 How to create Extensions for SWRC Core

Section 3.5 already discussed the modular design of the SWRC where applications might introduce extension modules for their particular modelling requirements. In this subsection we give some guidelines on how to create extensions by means of examples.

A particular variation concerns whether to use the OWL datatype restrictions in the ontology or not. The question here is whether one should model datatypes for attributes such as "year" as `xsd:integer` or even as `xsd:datetime`. Often bibliographic metadata is very unclean, such that it may be more appropriate to simply use `xsd:string`. One example is the AIFB portal where information is coming out of a database. When entering the year for a publication, e.g., users are often sloppy and include additional blanks or non-integers. In this case, defining `year` as `xsd:integer` would yield an error in type-checking ontology editors like Protege. However, another usage of the SWRC might require `xsd:datetime` in order to enable datatype checking. The solution is to omit datatype restrictions in the SWRC and include them on demand in extending modules.

Another example where extensions come in handy are additional concepts and associations which are only required in a particular application. Community portals might extend the SWRC core by COIN which is a set of concepts and associations formalizing communities, user, their interrelationships etc. It is obvious that such information is not required in all uses of the SWRC.

Further extensions might require a more powerful representation formalism. The SWRC core is formalized in OWL-DLP whose expressiveness might not be enough for reasoning in concrete applications. Thus, we can think of extensions in OWL-DL, that import the SWRC core, but add axioms that are beyond the expressiveness of OWL-DLP.

Finally, extensions might define mappings to other ontologies, e.g. the ones mentioned in Section 6. An example would be the associations defined in Dublin Core (DC). We could specify a mapping of `dc:author` to `swrc:author` (the author of a publication) via `owl:equivalentProperty`, for instance.

4.2 URI Generation

Applications of the SWRC ontology such as semantic portals (see Section 5) and interoperability with other applications require sophisticated methods and techniques for identifying not only classes and properties of the ontology but also instances that describe (real-world) entities. Therefore we describe a common mechanism for generating Uniform Resource Identifiers (URI) for such instances.

The underlying idea is to generate *uniform* identifiers which are *identical* for the same entity in all applications. Hence, an instance generated in one application can easily be identified or re-used in another application. The process of comparing and verifying instances can be reduced to comparing and verifying URIs which makes functions such as *integration on-the-fly* and *duplicate detection* more scalable and efficient.

Technically, several mechanisms are available and well-known for standard URI generation. A widely used approach is to generate a hash code of an object and use this hash as URI whereby a change of an object would generate also a new hash.

The benefit of this approach is to ensure interoperability and re-use of instances. The drawback of this uniform identifier generation is the computing time for generating the URIs.

Practical experiences have shown limitations and practical restrictions. Our experiences with common ontology editors and dynamic HTML pages are worth mentioning. Dynamic pages with contents stemming from a database often feature an additional parameter in their URL, e.g. `http://www.aifb.uni-karlsruhe.de/Personen/viewPersonOWL?id=79`. It would be natural to adopt these URLs also for ontology instances URIs when generated dynamically. The OWL specification does not explicitly disallow this usage. However, the current state-of-the-art editor, viz. the OWL plug-in of Protege, forbids URIs with parameters due to internal constraints ("=" is internally used for cardinality and "?" for "someValuesFrom"). It is thus necessary to rewrite such URIs to `http://www.aifb.uni-karlsruhe.de/Personen/viewPersonOWL/id79.owl`, for instance.

Another problem is the lack of a default mechanism for retrieving ontologies describing an instance for given URI. When we have dynamically generated ontologies, each describing one instance, we cannot use the same URI for both the ontology and the instance. The AIFB portal, e.g., features numerous dynamically generated ontologies with inter-linkage between instances. The URIs of the described instance and its corresponding ontology cannot be equal. A crawler cannot easily follow the link and harvest required information. A solution is to establish a forward reference from the URI of the ontology instance to the URI of the ontology itself. This would allow a crawler to follow the link to the ontology. A valid solution (as used in the AIFB portal) is depicted below

```
<rdf:RDF>
 <owl:Ontology rdf:about=
   "http://www.aifb.uni-karlsruhe.de/Personen/viewPersonOWL/id79.owl">
 </owl:Ontology>
 <rdf:Description rdf:about=
   "http://www.aifb.uni-karlsruhe.de/Personen/viewPersonOWL/id79instance">
 ...
 </rdf:Description>
</rdf:RDF>
```

4.3 How to deal with Duplicates

When dealing with bibliographic metadata, but also in other common usage scenarios, one often encounters duplicate entries, i.e. entries which refer to the same bibliographic item, but which differ in their representation, e.g. by having the first names of authors abbreviated or not.

Duplicate detection is a challenging task in most information systems as well as in our applications. Quality of service and data quality are key successors of current ontology-based systems. Hence, we have come up with an intelligent method for duplicate detection, which will be presented only shortly here.

Avoiding duplicates in the first place is obviously the most efficient method for solving the problem which can be achieved by using the described URI technique above. Furthermore, dealing with existing duplicates is still a challenge. However, the smart combination of existing comparison algorithms applied in a multilayered duplicate detection system using additional domain and world knowledge has shown promising results in our experiments. Specific methods for semantic similarity functions are described in [8].

The semi-automatic system works in an iterative mode in which the user (mainly the administrator of the system) can stop the process at each iteration or continue with a more intensive search. In general, the detection starts with an light-weight comparison of *promising* attributes whereby promising attributes are identified by *statistical properties* or *semantic relevance*. Semantic relevance is given by the occurrence of facts in a domain ontology and corresponding attributes in the analyzed data. Each iteration then takes further attributes into account or uses a more cost-intensive comparison algorithm. In this manner, it is possible to identify a large set of duplicates within the first iterations.

5 Known Usages of SWRC

This section describes some of the different usages of the SWRC ontology. First, there are several portals that apply the SWRC to semantically annotate their resources. All of them follow our SEAL (SEmantic portAL) approach. Second, we introduce Bibster, a semantics-based Peer-to-Peer application, which uses the SWRC to describe bibliographic metadata. Finally, we describe a comprehensive extension of the SWRC, called COIN, that supports the modelling of communities of interest.

5.1 Semantic Portals (SEAL)

SEAL (SEmantic portAL) is a conceptual model that exploits ontologies for fulfilling the requirements set forth by typical community web sites. The requirements are the need to integrate many different information sources and the need for an adequate web site management system. The ontology provides a high level of sophistication for web information integration as well as for web site management. For further information about the general architecture please cf. [9, 10]. We have implemented several portals that conform to our SEAL approach and use the SWRC ontology correspondingly:

AIFB The portal of the Institute AIFB uses the SWRC to annotate staff, publications, projects and their corresponding interrelationships. The usage of the ontology will be explained below. `http://www.aifb.uni-karlsruhe.de`

OntoWare Ontoware is a software development community platform for Semantic Web related software projects. The SWRC ontology is extended for annotating developers and software projects. `http://www.ontoware.org`

OntoWeb The community site for the European Union founded project OntoWeb uses the SWRC to facilitate ontology-based information exchange for knowledge management and electronic commerce [10]. `http://www.ontoweb.org`

SEKT The SEKT project is a European research projects bringing together the communities of knowledge discovery, human language technology and Semantic Web. Its corresponding portal uses the SWRC much like the OntoWeb portal. `http://www.sekt-project.com/`

SemiPort The SemIPort (Semantic Methods and Tools for Information Portals) project develops innovative methods and tools for creating and maintaining semantic information portals (annotated by the SWRC ontology) for scientific communities. `http://km.aifb.uni-karlsruhe.de/semiport`

In the following we will describe the usage of the SWRC in the AIFB portal in more detail. Besides regular pages for human consumption, the portal of the Institute AIFB also provides annotated pages which contain machine processable content in form of OWL annotations (Figure 3).

The AIFB portal is based on the Zope application server where information about persons, publications or projects are retrieved from a relational database and presented to the user by XHTML. Such sites are also annotated by machine-understandable descriptions according to the SWRC ontology. Both XHTML files and annotations are interlinked by the `<link rel="meta">` tag. In addition, pages about persons, publications or projects feature an “OWL/RDF” button at the bottom to view the automatically generated information.

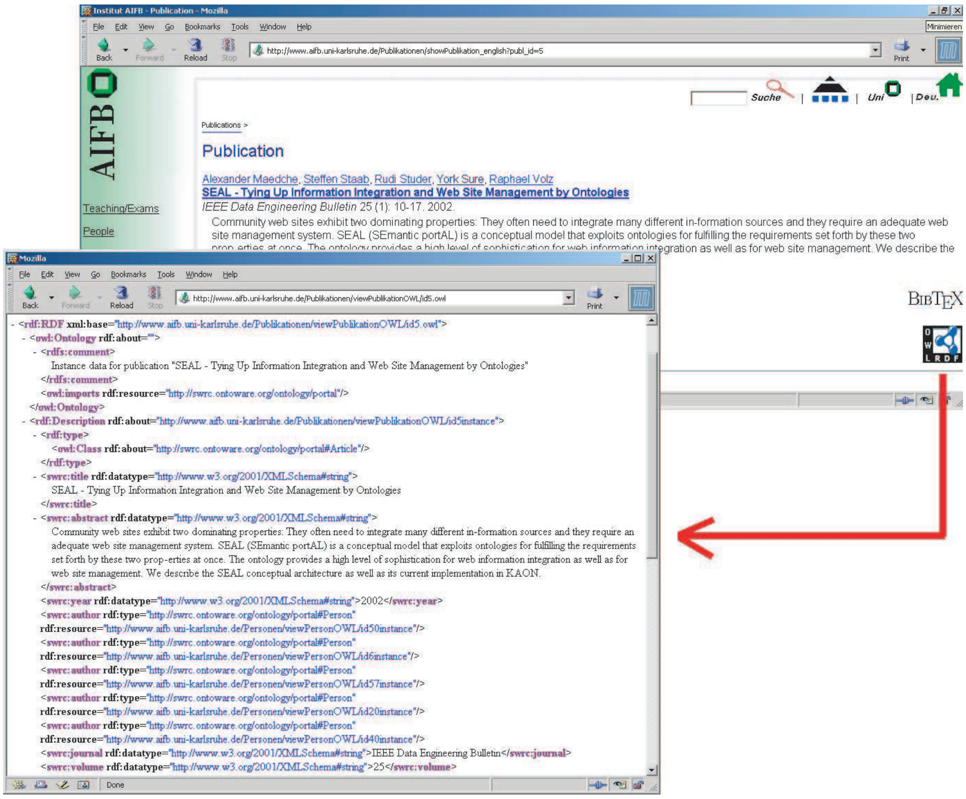

Fig. 3. Every page on Publications, Persons or Projects in the AIFB portal has its OWL annotation.

A typical annotation contains an ontological description about exactly one instance, viz. the person, publication or project regarded. However, there are also bulk annotations, e.g. for sites listing the whole staff or sites listing all publications of a person. In this case the descriptions just contain links to actual publication description (cf. also Section 4.2). We also offer a complete export of the database as 2 MB OWL file.

5.2 Bibster Application

Bibster[4] [11] is an award-winning semantics-based Peer-to-Peer application aiming at researchers who want to benefit from sharing bibliographic metadata, which they usually laboriously maintain manually in BibTeX files. Bibster enables the management of bibliographic metadata in a Peer-to-Peer fashion: it allows to import bibliographic metadata into a local knowledge repository, to share and search the knowledge in the Peer-to-Peer system.

[4] `http://bibster.semanticweb.org/`

Two ontologies are used to describe properties of bibliographic entries in Bibster, an application ontology and a domain ontology [12]. Bibster makes a rather strong commitment to the application ontology, but the domain ontology can be easily substituted to allow for the adaption to different domains. Bibster uses the SWRC ontology as application ontology while the domain ontology is used for classification of metadata entries, enabling advanced querying and browsing. It describes topic hierarchies using relations such as subTopic, relatedTopic, etc. In Bibster, we use the ACM Topic Hierarchy[5] as the default domain ontology. This topic hierarchy describes specific categories of literature for the Computer Science domain. It covers large areas of computer science, containing over 1287 topics ordered using taxonomic relations, e.g.: subTopic(Artificial Intelligence, Knowledge Representation Formalisms). Bibster relies on BibToOnto `https://sourceforge.net/bibtoonto` to automatically extract an ontology-based representation of the bibliographic metadata from plain BibTeX files. It also automatically classifies bibliographic entries according to the ACM topic hierarchy.

5.3 COIN **Community Ontology**

While the SWRC ontology itself is directed at modelling the explicit objects and relationships in the academic domain, it does not cover the implicit collaborations and interests of the researchers – or users – in general. The COIN community ontology is an extension to SWRC to support modelling of *communities of interest* [13] and aims at capturing the membership of users in communities of interest, the knowledge flow within a community and relevant community resources. Together, the SWRC+COIN ontologies provide a flexible means for modelling knowledge communities, their people, individual roles played in the communities, the status of a community in its life cycle and relevant events or publications. SWRC+COIN strictly extends SWRC through additional concepts, subconcepts and additional restrictions on OWL properties and the like.

6 Related Work

BibTeX is a metadata format for modelling bibliography entries used within the LaTeX document preparation system. LaTeX which can be used on most every operating system is a well known system for typesetting documents and is used often in the scientific and academia communities or by commercial publishers. LaTeX itself is based on TeX an initial typesetting system developed by Donald Knuth. BibTeX is on the one hand a metadata standard and on the other hand a system for managing bibliographic entries obtaining these data from bibliographic databases. The metadata format and program was developed in 1985 by Oren Patashnik and Leslie Lamport [14]. BibTeX provides metadata attributes (entry types) for nearly every kind of bibliographic entry which has its own set of attributes describing a reference. The tag-based syntax of BibTeX is at the moment the most well-known (exchange-) format for bibliography metadata, especially on the World Wide Web.

The *Dublin Core (DC)* metadata standard is an attribute-value based set for describing a wide range of resources. It was developed during 1995 and 1996 as a response to the need to improve retrieval of information resources, especially on the World Wide Web. The DC standard includes two levels: Simple and Qualified. Simple DC comprises fifteen elements; Qualified DC includes an additional element, Audience, as well as a group of element refinements (or qualifiers) that refine the semantics of the elements in ways that may be useful in resource discovery. The

[5] `http://www.acm.org/class/1998/`

semantics of DC have been established by an international, cross-disciplinary group of professionals from librarianship, computer science, text encoding, the museum community, and other related fields of scholarship and practice.

FOAF, or 'Friend of a Friend', provides a way to create machine-readable Web documents, mainly for people (their interests, relationships and activities), groups and companies. Therefore the FOAF project uses a vocabulary to provide a collection of basic terms that can be used in these Web documents and to express relations about things in the world. The initial focus of FOAF has been on the description of people. Founded by Dan Brickley and Libby Miller, FOAF is an open community-lead initiative which is tackling head-on the wider Semantic Web goal of creating a machine processable web of data. Technically, FOAF is an RDF/XML Semantic Web vocabulary. FOAF documents are most commonly used as a way of representing information about people in a way that is easily processed, merged and aggregated.

SIOC (Semantically Interlinked Online Communities)[6] describes discussion forums and posts of online community sites. SIOC is modeled as an ontology in RDFS whereby it reuses terms from DC and FOAF. The main focus of SIOC is to represent information about online communities. Hence, concepts like 'forum' and 'post' has been modelled. This technical oriented representation allows an easy application and integration of many communities and complements very well the SWRC so that we foresee possible future joint applications using SIOC and SWRC.

Closest to the SWRC are the *AKT reference ontology*[7] which is developed by the AKT project and e.g. is used in the CS AKTive Portal and the *Knowledge Web portal ontology*[8] which is developed by the Knowledge Web consortium and e.g. is used in the Knowledge Web portal. Both contain similar concepts and relationships as the SWRC ontology and serve a similar purpose. In fact, as has been shown in [15], it is fairly easy to generate mappings between the Knowledge Web portal and the SWRC (referenced in the deliverable as OntoWeb portal ontology) ontologies.

7 Conclusion

In this paper we have described the SWRC ontology – the Semantic Web Research Community Ontology, which models key entities in research communities. The SWRC ontology has its origin in early research projects in the Semantic Web Community and has been consecutively refined up to the current OWL version of the ontology which we have presented in this paper and is thus a good example for ontology reuse.

We have also presented a set of known usages for the ontology. The benefits of using, reusing and extending SWRC are obvious. On the one hand, the ontology in its current state is mature and has been agreed upon by many parties. It is therefore a good choice for a ready and established out-of-the-box ontology in the research domain, thus e.g. saving time and effort for building a comparable schema. On the other hand, using the SWRC ontology ensures inter-operability and easy data integration between different applications, for example between different portals. Future work may include optional modules for alignment with the related schemas pointed to in Section 6. To conclude, we believe that for making the Semantic Web reality the re-usage of ontologies and their continuous improvement by user communities is crucial. Our contribution aims to provide a description and usage guidelines to make the value of the SWRC explicit and to facilitate its re-use.

[6] `http://rdfs.org/sioc/`
[7] `http://www.aktors.org/publications/ontology/`
[8] `http://knowledgeweb.semanticweb.org/`

Acknowledgements: Research reported in this paper has been partially financed by EU in the IST projects OntoWeb (FP5-29243), SWAP (FP5-34103), Knowledge Web (FP6-507482) and SEKT (FP6-506826) and by the German Ministry for Education and Research (bmb+f) in the project SemiPort.

References

1. Benjamins, V.R., Fensel, D.: Community is knowledge! $(\mathrm{KA})^2$. In: Proceedings of the 11th Workshop on Knowledge Acquisition, Modeling, and Management (KAW '98), Banff, Canada. (1998)
2. Staab, S., Angele, J., Decker, S., Hotho, A., Maedche, A., Schnurr, H.P., Studer, R., Sure, Y.: Semantic community web portals. In: Proc. of Int. Conf. WWW9, Amsterdam, NL, May, 15-19, 2000, Elsevier (2000)
3. : Web ontology language (OWL). www.w3.org/2004/OWL/ (2004)
4. Antoniou, G., van Harmelen, F.: Web Ontology Language: OWL. In Staab, S., Studer, R., eds.: Handbook on Ontologies. Springer (2004) 67–92
5. Hitzler, P., Studer, R., Sure, Y.: Description logic programs: A practical choice for the modelling of ontologies. In: 1st Workshop on Formal Ontologies Meet Industry, FOMI'05, Verona, Italy, June 2005. (2005)
6. Grosof, B., Horrocks, I., Volz, R., Decker, S.: Description logic programs: Combining logic programs with description logics. In: Proc. of WWW 2003, Budapest, Hungary, May 2003, ACM (2003) 48–57
7. Noy, N.: Representing classes as property values on the semantic web. Technical report, W3C (2005) `http://www.w3.org/TR/swbp-classes-as-values/`.
8. Ehrig, M., Haase, P., Stojanovic, N., Hefke, M.: Similarity for ontologies - a comprehensive framework. In: 13th European Conference on Information Systems. (2005)
9. Maedche, A., Staab, S., Studer, R., Sure, Y., Volz, R.: SEAL – tying up information integration and web site management by ontologies. IEEE Computer Society Data Engineering Bulletin, Special Issue on Organizing and Discovering the Semantic Web **25** (2002) 10–17
10. Hartmann, J., Sure, Y.: An infrastructure for scalable, reliable semantic portals. IEEE Intelligent Systems **19** (2004) 58–65
11. Haase, P., et al.: Bibster – a semantics-based bibliographic peer-to-peer system. In: Proc. of the 3rd Int. Semantic Web Conference, Hiroshima, Japan, 2004. Volume 3298 of LNCS., Springer (2004) 122–136
12. Guarino, N.: Formal ontology in information systems. In Guarino, N., ed.: Formal Ontology in Information Systems. Proceedings of FOIS'98, Trento, Italy, June 6-8, 1998, Amsterdam, IOS Press (1998) 3–15
13. Bloehdorn, S., Haase, P., Hefke, M., Sure, Y., Tempich, C.: Intelligent community lifecycle support. In: Proceedings of the 5th International Conference on Knowledge Management (I-KNOW 05). (2005)
14. Lamport, L.: LaTeX: A Document Preparation System. Addison-Wesley (1986)
15. Suárez-Figueroa, M.C., Gómez-Pérez, A., López-Cima, A.: Portal ontology. Knowledge Web Deliverable 1.6.2, UPM (2005)