Ontological Bridge Building – Using Ontologies to Merge Spatial Datasets

Catherine Dolbear and Glen Hart

Ordnance Survey of Great Britain
Romsey Road, Southampton, SO16 4GU, United Kingdom
{Catherine.Dolbear, Glen.Hart}@ordnancesurvey.co.uk

Abstract

This paper discusses the approaches we have taken to addressing issues of ontology merging and mapping from an ontology to a database, which arise from the semantic data integration problem. We present a proposal for curtailing the extent of an ontology module referenced within another ontology, so that the entire ontology need not be imported if only a small subset is required for reuse. As the national mapping agency for Great Britain, we are particularly concerned with the integration of spatial data, so the second contribution of this paper is to describe how we incorporate a spatial relations ontology into a data ontology that describes the mapping from domain classes to database. Since the spatial relations ontology is based on the Egenhofer 9 Intersection model employed by the Oracle spatial database, spatial SQL queries can be automatically generated from the data ontology, or a SPARQL query using the domain ontology terms.

Introduction

Ordnance Survey, the national mapping agency of Great Britain, maintains a continuously revised database of the topography of Great Britain. The database contains around 500 million features representing everything from forests, roads and rivers down to individual houses, garden plots, and even pillar boxes. This data acts as a referencing framework, underpinning many other government, commercial and scientific data. In the last five years our data has been used in over 1000 British research projects [EDINA 2007] as well as by numerous overseas research institutions. One of the primary concerns of our organisation is how to integrate our data with that belonging to the end user in an efficient and accurate manner, given the technical challenges that such integration poses. The integration problem is compounded for us by the complex nature of spatial data. [Hart and Dolbear, 2007].

The GeoSemantics research team at Ordnance Survey have therefore been investigating the use of ontologies to assist this integration process. This paper provides an overview of the approach that we are taking to address the two main problems concerning semantic data integration. These are how to handle the semantic differences between domains (ontology merging), and how to handle the semantic differences between a domain-level understanding of the world and a representation in a database (ontology to database mapping to overcome the "semantic gap").

Data Integration and the Use of Ontologies

At present, the process of data integration, whether it be the physical combining of data within a database, or the dynamic integration of data drawn from diverse resources as part of a query, is, at best *ad hoc*. There are three principal problems that need to be overcome:

- Syntactic or structural differences between the data sources. Typically, this manifests itself in terms of different database schemas and transfer formats;
- Semantic differences between the domains of interests;
- Semantic differences within a domain with respect to the manner in which the domain is understood by those working within the domain and the semantics of the data as represented within the database.

In all cases there is often a lack of proper documentation about domain, data and database.

To date, most resources have been directed at overcoming the syntactic issues. Traditionally, this has relied upon conventional approaches, often involving the production of bespoke software. More recently, there have been approaches that have applied semantic web technologies [Alani et al, 2007]. Such semantic approaches typically employ an RDF schema [Brickley and Guha, 2004] description of the data structures enabling simple mappings to be established between the datasets. Most, if not all, of the syntactic differences can be overcome if the data itself is represented as RDF. However, at present, such an approach is impractical for very large datasets due to the number of triples that are generated. For our own main database, we estimate that the current 500 million features held in our own main database, and represented by 3.5 terabytes of data, could easily expand to perhaps 15-20 billion triples, occupying many petabytes of storage. Such quantities are currently well beyond the capabilities of RDF based technologies. Even when considering that typical scientific uses of our data require much smaller quantities of data, the resultant RDF inflation is still too great for existing solutions. This approach also does not really allow much expressivity in the semantic description,

making it difficult for the merger to decide which concepts from the two datasets are really related.

Our own research has included work understanding the application of RDF to our data. The majority of our effort though has been directed towards using OWL [McGuinness and van Harmelen 2004] based ontologies to provide interfaces between domains, and from domain to database. Our reasoning is based upon the assumptions that we have made about the development and implementation of semantic technologies, given the current infrastructure of the information economy.

Specifically, these assumptions are:

- In the medium term, there will be practical implementations of very large RDF triple stores with realistic query times. However, even if such implementations are able to compete with existing resources held within relational databases, it will be many years before a significant volume of data is held in this form due a natural inertia to move to new technologies, and the time and economics required to make such changes.
- Given that conventional database resources will continue to hold a significant proportion of all data well into the future, it is likely that RDF triple stores will provide the front-end solutions, and that RDF could become a universal transfer format.
- In time, ontologies for data based on RDF will prove insufficient in terms of descriptive richness.

Ontological descriptions support two end purposes: to describe a database or domain in sufficient detail to enable data to be integrated; and to support inferencing to be made over the data, producing richer information than is simply held within the database. Since ontologies authored in OWL can provide more complete descriptions of a domain, this should allow more accurate data integration (and which go beyond differences of syntax), and more sophisticated inference.

Our approach uses OWL ontologies to bridge the semantic gap that exists between domain and database, and also to bridge between domains. Thus, we use domain ontologies to describe the domains of interest, using the vocabulary of those working within those domains, and use data ontologies to map the domain ontologies to existing database implementations. We are also following with interest recent suggestions of using rules instead of data ontologies to map from database to domain [Seabourne, Steer and Williams, 2007]. Our intention is to enable conventional relational databases to be queried by transforming SPARQL [Prud'hommeaux and Seaborne, 2006] queries to SQL, with the result of the query being translated into RDF. As shown in figure 1, our research therefore covers two main areas. The first is the understanding of how to merge domain ontologies to enable a database in one domain to be queried using a foreign domain view. Our second strand of research is the querying of spatial relational databases via SPARQL queries that obtain their vocabulary from a domain ontology, and that are in turn mapped to the database using a data ontology.

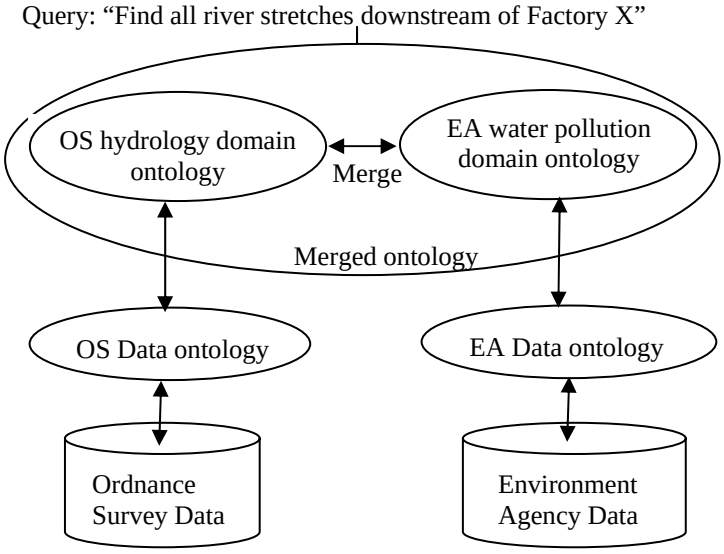


Figure 1: Example of a semantic data integration scenario

Ontology Merging

If we wish to extract data held across two or more separate databases (or other data resource), and where each of these databases is described by a domain ontology, then it will be necessary to integrate these ontologies to produce an ontology that provides a unified view. This merging is carried out in the context of performing a particular, task and the resultant ontology may be a physical, distinct and persistent ontology, or it can be a more ephemeral collection of concepts, built on the fly as part of the loading process of a reasoner. At present only the former is possible.

Merging ontologies will involve the following processes:

- Alignment, where the concepts held in the separate ontology are compared, and mappings are made between identical concepts held in each ontology. It may also involve bridging between similar, but distinct, concepts.
- Modularisation, where only those concepts that are required in the task are extracted, or viewed as distinct modules isolated from the rest of the domain ontology. This is necessary to ensure that the merged ontology does not become bloated with unnecessary concepts.

- Extension where the merged ontology is extended, in order to provide an accurate description of the task. In this situation, either completely new concepts are

added, or additional axioms are added to concepts taken from the domain ontologies.

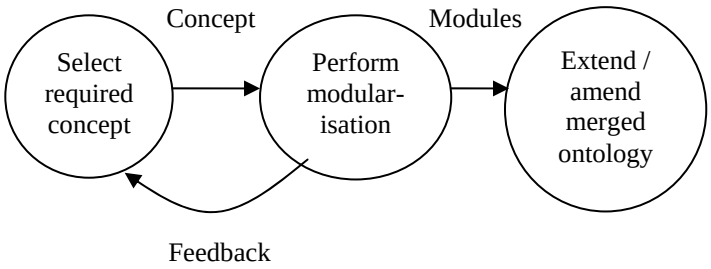


Figure 2: Ontology merging process

Much existing work on ontology merging has been concerned with the problem of assessing the similarity of concepts in different ontologies e.g. [Ehrig and Sure, 2004] and [Schwering, 2006]. Whilst acknowledging the importance of this work, we are at present more interested in the problems of mapping and modularisation. Previous approaches [Klein and Stuckenschmidt 2006] have dealt with the issue of what happens when changes in the foreign ontology affect the reasoning in the local ontology. We are more concerned in identifying those aspects of a foreign concept that are relevant to the domain of interest than in preserving the totality of that concept (and hence can accept a structured amount of change to the concept definition). For example, consider the case of reusing an ontology representing the Valuation Office (the UK government department for property taxation) coding schemes to build a topographic land use description. For a particular concept, the Valuation Office ontology will contain axioms that are clearly relevant to the new ontology, such as the geographic object's Address, and Land classification type. However, it will also include irrelevant axioms, e.g. valuation methods applied by the tax inspector. If the latter axioms are not included in the new land use ontology, the logical meaning of the concept has changed, but the topographic land use domain is not affected since it uses accesses a logical subset of the full VO definition.

Our ontologies[1] have been modelled as far as possible according to the approach suggested in [Rector, 2003], whereby primitive classes are specialised along one hierarchy only, to prevent "tangled" hierarchies and result in a "normalised" ontology made up of independent disjoint taxonomies. However, this does not solve the problem of modularisation as we have also followed the good practice guideline of not creating 'weak' primitive concepts that don't exist as nameable entities in the domain. Such weak concepts are often a result of over-using hierarchies. This means our topographic ontologies tend to have very shallow hierarchies, and are not suitable as stand-alone modules that can be separating from the main ontology.

Since OWL has been designed for its ontologies to act as web-based resources, this would suggest that OWL supports constructs necessary to enable linkages between ontologies to be made. However, OWL provides only a crude method of linking ontologies: the *import* facility. This enables entire ontologies to be imported. The only control an author then has is to add axioms relating concepts in each ontology, and for the same concepts in different ontologies to be associated through means of the *EquivalentClass* property. Thus merging can only proceed by extracting the required module from the foreign ontology, and either directly integrating its content into the merged ontology, or creating a new stand-alone module.

Work at the University of Manchester [Cuenca Grau et al, 2007] has extended the functionality of the SWOOP ontology editor [Kalyanpur et al, 2005] that, given a set of input concepts, will produce a module comprising the minimal set of concepts and axioms that satisfies the needs of the input set. This is useful when applying the ε Connections technique [Cuenca-Grau, Parsia and Sirin, 2005], which links disjoint ontologies. However, topographic ontologies we have experimented with tend to be very interconnected, and hence divide only into one large module, comprising nearly all of the ontology, and a few very small disjoint modules. Therefore our merging application does not easily lend itself to the ε Connections requirement for disjoint domains. Without modification to either OWL and (and minimally) the reasoners, the modules output by the SWOOP tool would have to exist as stand-alone ontology modules. What is needed instead is a mechanism for indicating, within the merged ontology itself, where we want the import of the foreign ontology to stop, that is, to indicate the extent of the module that we are importing. Using such modularisation tools, our research aims to demonstrate that minimal changes are required to OWL and DL reasoners, to enable modules to be referenced from within their original ontologies, rather than extracting them into stand-alone modules.

Our overall view of the ontology merging process is summarised in figure 2. The first stage, to select the required concepts, is performed by experts in the task domain, and involves identifying those concepts from the domain ontologies that are appropriate to the task. At present, this process will largely be reliant on the domain expert interpreting the ontologies and drawing on additional domain knowledge. In the future, it will be possible for this task to be assisted by tools that are based on calculated similarity measures. However, it is unlikely that it will ever be possible to completely automate such a process. Tools may be of most assistance in helping the domain expert locate and manage potential candidates for merging, particularly when dealing with very large, complex ontologies.

[1] Available at http://www.ordnancesurvey.co.uk/ontology

The selected concepts can then be used as the input in to the next process which is modularisation. Here, the use of tools, such as that in SWOOP, could completely automate the process of identifying those additional concepts and their minimal set of axioms that are required to produce internally coherent modules. Note that these modules will be *logically* consistent, but in terms of their content, a tool may not output a module that actually makes sense standing alone. Even though this process can be completely automated, we still suggest that at this stage the modules are checked by the domain expert, as the additional content may cause the domain expert to rethink earlier decisions. Once modularised, the merged ontology is able to reference these modules and may wish to extend certain descriptions and include new concepts, to describe the task more fully. We propose to test this process by combining ontologies with assistance from a tool to aid modularisation, and by repeating the process in a purely manual fashion.

One aspect of our research has been to define a minimum set of techniques that are required to merge concepts in a domain ontology into a new merged ontology. We have identified and are testing three:

- Referencing. Here the domain concepts are referenced by the new ontology. This is the simplest form of merging and is the preferred and default approach.

- Referencing with Extension. In this situation, the domain concept is referenced by the new ontology and is then extended by the addition of new axioms.

- Referencing with Restriction In this method, a domain concept is referenced, but the degree by which the concepts referenced by the new concept are interpreted, is restricted. This enables the extent of use of the domain ontology to be constrained.

The first two techniques are covered by the constructs already available in OWL. In the first instance, the class in the domain ontology is merely referenced via its URL. In the second case, a class created in the new ontology is made equivalent to the class in the domain ontology and then extended with additional axioms. Referencing with Restriction is both more complex to understand, and can only be implemented in OWL in a limited fashion. Referencing with Restriction means that if a domain ontology class is referenced with restriction in a new ontology, all the domain class's axioms are ignored when reasoning over the merged ontology. For example, suppose we have an ontology about birds and an ontology about water-bodies such as rivers, lakes and ponds. If we wanted a third ontology to describe a duck pond, we could use both the other ontologies to provide the relevant classes. However, it is enough to describe a duck pond as a pond that contains ducks. The bird ontology may contain lots of additional information about ducks that will not be required. Using Referencing with Restriction would mean that only the fact that Duck was a concept and had been taken from the bird ontology would be incorporated in to the merged ontology. This approach is similar to the approach of the SWOOP tool [Cuenca Grau et al, 2005] although it works at the cruder level of concept rather than axiom. (The SWOOP tool drops unnecessary axioms).

At present the best that can be done, in terms of using OWL 1.1, is to generate stand-alone ontology modules, rather like the SWOOP tool. By indicating that concepts are Referenced with Restriction using annotation properties within OWL, it would be possible to minimally modify reasoners to discriminate between concepts when loading an ontology. This is an area of ongoing experimentation for us.

Data Ontologies

The second element of our semantic data integration research relates to the issue of how to map from our relational, spatial database to an organisational-level understanding of the domain. We believe that existing tools designed to generate ontologies based on database schemas [for example, Crossvision 2007] are missing the point: databases are rarely good descriptions of a domain, being the result of initial design constraints, performance optimisation processes and a contingent maintenance history. Furthermore, the schema itself seldom contains a full description of the domain, as other relevant relationships are often buried in software code or in the encoding of various attributes. If databases were that orderly and easy to understand in the first place, semantics would not be needed!

Rather, legacy relational databases tend to include all sorts of terminology in the column headings or field values that require in-depth reading of the (text) specification to work out what it really means at a domain level. For example, in our OS MasterMap™ product, a "Building" is any row in the Topographic Area table where the Theme is "Buildings"; while an "Island" could be defined as any row in the Topographic Area table which has a Theme column containing the value "Land" and has a Polygon column that is spatially inside the Polygon of another feature which has a Theme of "Water". An even more complex example is that of Fields. A "Field" is any row in the TopographicArea table whose Theme column is "Land" and which has an area to perimeter ratio greater than 8. (The area and perimeter values come from geometrical calculations on the Polygon column value that holds the boundary information of the topographic object.) It's not a good idea to bury all of this detail in the query alone, or indeed in the product manual. A promise of semantic technology is to bring all of this hidden complexity out into the open.

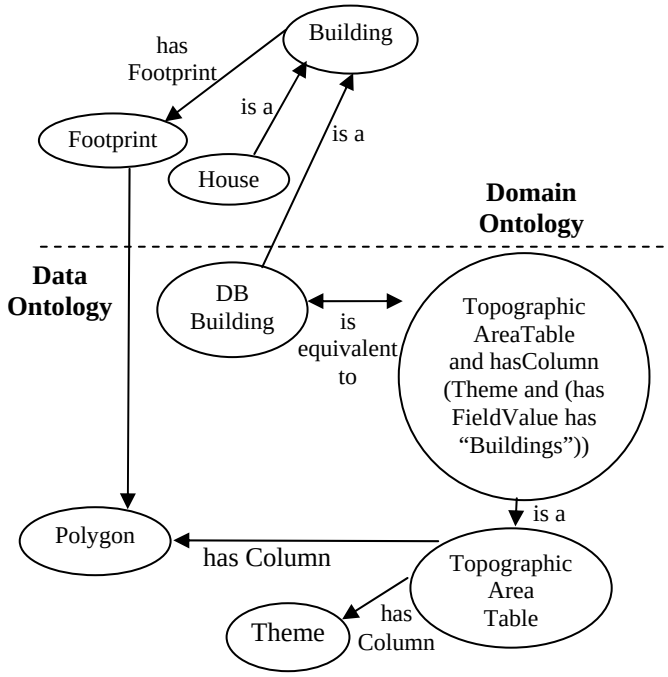
Several different methods have been proposed for exposing relational data as RDF, such as D2RQ [Bizer and Seaborne, 2004] which creates virtual RDF graphs using an N3 mapping file, or more recently, SQL-RDF [Seaborne, Steer and Williams, 2007], which uses rules to carry out the mapping from data to domain. Our approach has been to use OWL data ontologies to encode our mappings, as we need more expressivity than N3 can offer. We also want to avoid any knowledge being hidden in a database-to-ontology mapping file, or the Java code that creates the SQL from the ontology. There have been several suggestions for ontologies describing the well-known components of a database – Tables, Views, Columns and so on, in various formats such as N3 (used as an input to Schemagen within D2RQ) and OWL Full [Perez de Laborda and Conrad, 2005]. In order to limit our database ontology to OWL-DL expressivity, we modelled actual database instances as subclasses of the "Database" concept, rather than more obviously modelling them as instances, which would require the use of classes as values.

This Database ontology is imported by our data ontology, which defines the parameters of the specific database where Ordnance Survey's topographic data is stored. This includes information about how to connect to the database instance, along with details of how the data in the database can be understood in terms of the domain concepts. A simplified example of this mapping is shown in figure 3. The concept "DB Building" in the data ontology is a subclass of the domain ontology "Building" and will be instantiated by data from the database, by using information in the data ontology to construct SQL queries to retrieve the relevant data from the TopographicArea Table. The necessary and sufficient conditions for "DB Building" (i.e. the bubble which is equivalent to "DB Building") can be converted to an SQL query to retrieve data to instantiate the domain ontology's Building class, while the necessary conditions for "DB Building" (i.e. the bubble which is a subclass of "DB Building") provides the information for how to construct a SQL query to retrieve data to instantiate properties of the Building, such as the Building's Footprint in this case.

Spatial Querying

As the "Island" example shows for our applications and data, we also need to consider spatial relations, and how these map onto SQL operators. Our solution has been to author a geometric spatial relations ontology module, which covers the 9 Intersection model relations [Shariff, Egenhofer and Mark, 1998], the basis for the spatial operators used by the Oracle database. This ontology is

Figure 3: Example of how a data ontology links to a domain ontology for the class Building



then imported into the data ontology, allowing the Island class to be defined in OWL as follows:

```
DB_Island ≡ TopographicArea
and  hasColumn some (Theme
      and hasFieldValue value "Land"))
and hasColumn some (Polygon
      and isCompletelySpatiallyInside some
      (TopographicArea
            and hasColumn some (Theme
                  and hasFieldValue value "Water")
            and hasColumn some Polygon))
```

Note that it is the spatial representation, the Polygon, that is inside another Polygon, rather than the Island object itself being inside the Water. This information can then be used to automatically generate the SQL query for finding Islands:

```
SELECT ta1.FID, ta1.Theme
FROM TopographiArea ta1, TopographiArea ta2
WHERE ta1.Theme  = 'Land' AND ta2.Theme = 'Water'
AND SDO_INSIDE(ta1.Polygon, ta2.Polygon) = 'TRUE';
```

Figure 4 shows a high level system diagram of our method of combining spatial and semantic querying into a relational database. Our current research is now looking at how the vocabulary defined in the data ontology, particularly the geometric spatial relationships, can be used to create SPARQL queries using spatial elements, and how the system may also benefit from OWL reasoning at the domain level.

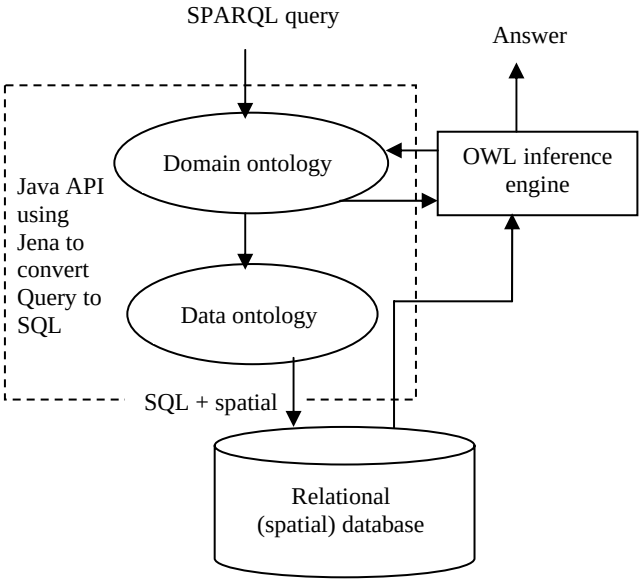


Figure 4: System to combine spatial and semantic queries into a relational database.

Conclusions

This paper has discussed some of the approaches we are taking to address the dual problems of ontology merging and domain ontology to database mapping, which together are necessary components of a semantically-enabled data integration system. We believe some extensions to the OWL standard are needed to address modularisation in the case of Referencing with Restriction, and we have suggested a way of enabling spatial queries through the use of a geometric spatial relations ontology that can be mapped on to the Oracle spatial operators. Since many scientific applications have a spatial dimension to them, addressing both these issues will be of direct relevance to the scientific information technology community.

This article has been prepared for information purposes only. It is not designed to constitute definitive advice on the topics covered and any reliance placed on the contents of this article is at the sole risk of the reader.

References

Alani, H., Dupplaw, D., Sheridan, J., O'Hara, K., Darlington, J., Shadbolt, N. and Tullo, C. (2007) Unlocking the Potential of Public Sector Information with Semantic Web Technology. In *Proceedings of The 6th International Semantic Web Conference (ISWC)*, Busan, Korea.

Bizer, C. and Seaborne, A. D2RQ –Treating Non-RDF Databases as Virtual RDF Graphs. Poster at 3rd *International Semantic Web Conference* (ISWC2004) Hiroshima, Japan, November 2004.

Brickley, D. and Guha, R.V eds *RDF Vocabulary Description Language 1.0: RDF Schema W3C Recomendation*., 2004 http://www.w3.org/TR/rdf-schema/

Crossvision: *Generating ontologies from database schemas* http://documentation.softwareag.com/crossvision/xei/ostudio/accrdbms.htm

Cuenca Grau, B., Horrocks, I. Kazakov, Y. and Sattler, U. Just the Right Amount: Extracting Modules from Ontologies, *In Proceedings of WWW-2007: the 16th International World Wide Web Conference*

Cuenca Grau, B. Parsia, B., and Sirin, E., "Combining OWL ontologies using E-Connections," *Web Semantics: Science, Services and Agents on the World Wide Web*, 4, 40-59, 2006

EDINA, 2007, http://edina.ac.uk/

Ehrig, M and Sure, Y. Ontology Mapping – An Integrated Approach, 2004, Christoph Bussler, John Davis, Dieter Fensel, Rudi Studer, *Proceedings of the First European Semantic Web Symposium*, volume 3053 of Lecture Notes in Computer Science, pp. 76-91. Springer Verlag, Heraklion, Greece 2004.

Hart G. and Dolbear C. "What's so special about spatial?" In *The Geospatial Web: How Geobrowsers, Social Software and the Web 2.0 are Shaping the Network Society* Arno Scharl, Klaus Tochtermann (Eds.) Advanced Information and Knowledge Processing Series 2007, London: Springer, ISBN 1-84628-826-6

Kalyanpur, A. Parsia, B. Sirin, E. Cuenca-Grau, B. and Hendler, J. "Swoop: A 'Web' Ontology Editing Browser", Journal of Web Semantics Vol 4(2), 2005

Klein, M. and Stuckenschmidt, H., Evolution Management for Interconnected Ontologies citeseer.ist.psu.edu /672047.html

McGuinness, D.L and van Harmelen, F. *OWL Web Ontology Language Overview* W3C Recommendation, Ed.. 10 February 2004 http://www.w3.org/TR/owl-features/

Perez de Laborda, C. and Conrad, S. "Relational.OWL – A data and schema representation format based on OWL", *2nd Asia-Pacific Conference on Conceptual Modelling*, Newcastle, Australia. Conferences in Research and Practice in Information Technology Vol 43. Seven Hartmann and Markus Stumptner Eds. 2005.

Prud'hommeaux. E. and Seaborne, A. *SPARQL Query Language for RDF* W3C Working Draft. http://www.w3.org/TR/rdf-sparql-query/ 4 October 2006

Rector, A. Modularization of domain ontologies implemented in description logics and related formalisms including OWL, *Proceedings of the international conference on Knowledge capture*, 121-128, 2003

Schwering, A: *Semantic Similarity Measurement including Spatial Relations for Semantic Information Retrieval of Geo-Spatial Data*. University of Münster, September 2006.

Seaborne, A., Steer, D. and Williams, S. SQL-RDF Andy Seaborne, *W3C RDF Access to Relational Databases workshop*, October 2007

Shariff A, R Egenhofer M J and Mark D M, Natural Language spatial relations between linear and areal objects: The topology and metric of English Language terms.(1998) *Int Journal of Geographical Information Science* 12(3): 215-246