# Position paper on Expressing Relational Data as RDF

Catherine Dolbear and John Goodwin,
Ordnance Survey of Great Britain
{Catherine.Dolbear, John.Goodwin}@ordnancesurvey.co.uk

## 1. Introduction and use case

As the national mapping agency of Britain, Ordnance Survey's position at this workshop is that of a data vendor. Unlike mapping agencies in many countries, Ordnance Survey is required to cover its operating costs through revenue generated from the sale of electronic and graphic products.

Spatial data is generally considered to be complex and difficult to integrate into users' business processes. Therefore we are looking to semantic to improve our customers' experience in two ways. Firstly, to provide the means to repurpose our data according to customers' terminology, which will in effect increase our product portfolio. Secondly, the technology can facilitate the integration of our data with customers' own, in a semantically meaningful way. This will thus increase the take up of our data; expand our markets into areas that traditionally could not afford to or didn't know how to use our data and reduce errors in data integration due to implicit assumptions as to the meaning of the data.)

We own one of the largest vector spatial databases in the world, describing half a billion "real world objects" such as buildings, road segments, fields and post boxes. Our surveyors and photogrammetrists submit around 5000 changes to the database every day, recording accuracies of around 10 cm precision on the ground. Converting our entire dataset to RDF would result in a minimum of 10 billion triples (a figure that covers only a small element of our data as it would be impractical to include geometric data due to its requirement for spatial indexing)., Even this minimal set is beyond the capabilities of current triple stores. In addition, it is not commercially or operationally viable to replace the tried and tested relational database management system, on which our entire business is based, with a triple store. Therefore we are experimenting with mapping relational data to RDF, extending D2RQ [1] with a spatial relations ontology to create SQL queries that use Oracle spatial operators. The following describes our wish list, based on our experiences so far of ontology to database mapping technology.

## 2. Our position and some preliminary requirements

### *2.1 Performance*

The bottom line is that querying has to be more efficient! While we can accept that a SPARQL query onto a virtual RDF graph can be slower than SQL direct into a RDBMS, because of the advantages of explicit semantics and easier-to-construct queries in SPARQL, it shouldn't be *that* much slower. This comment is based on two findings – firstly construction of a SPARQL query in two minutes versus a full day to correctly construct and test the equivalent SQL query. Secondly, on an experiment using D2RQ linked into a very small test data set of only 31000 Buildings, which took several seconds to return query results. We would welcome more understanding of how to optimise our use of such mapping tools, and to see them tested on large scale relational databases.

### *2.2 Consider the semantics!*

Test sets shouldn't be "toy" examples, not just because of scaling issues, but also because of the semantic complexity that gets buried in a practical database. 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 "Building"; to extract instances of Fields from the same dataset, you would have to employ an even more complex query. 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 which holds the boundary information to the topographic object.) It's not a good idea to bury all of this detail in the query alone. The promise of semantic technology is to bring all of this hidden complexity out into the open– to do this we are writing "data ontologies" to explain the mapping from the relational model to the domain ontology concepts, and any standardisation process needs to take this issue into account. This is particularly relevant when it comes to what we call "product repurposing" – varying the data ontology to produce a differently classified RDF data set from the same underlying relational data, based on the perspective and needs of a particular customer. We believe that existing tools designed to generate ontologies based on database schemas are missing the point: databases are rarely good descriptions of a domain being the result of both performance optimisation processes and contingent maintenance history. And, in any case, the schema itself will not support a full description of the domain, other relevant relationships often being buried in code or in the encoding of various attributes. If databases were that orderly and easy to understand in the first place, you wouldn't need semantics! So our requirement here is for any standardised tool/system for expressing relational data as RDF is to take into account the need for a data ontology to describe the complexity of a mapping from a relational to RDF model.

### *2.3 Standardised "database" 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 [1]) and OWL Full [2]. 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. This database ontology was then imported into the data ontology, so that the mappings from the database to domain concepts could be modelled in a consistent way. We think it would be useful, and pretty straightforward to agree on a standard database ontology, so different tools could manipulate any data ontology importing it.

### *2.4 Encoding of spatial data*

We would like to see tools that allow storage and manipulation of various different datatypes (not just strings, integers etc.). Our specific interest is in spatial data, and we have been experimenting with using GML [3] tags around spatial data embedded within RDF. It would then be useful if these data types could be manipulated and be used for efficient querying as is possible in many standard relational databases. This could potentially allow for "SPARQL" queries such as that below:

```
select ?a ?b
where
{
      ?a rdf:type Land
      ?b rdf:type FloodPlain
      ?a geom ?g1
      ?b geom ?g2
```

```
    FILTER (?g1 overlaps ?g2)
}
```

This query finds all the instances of Land and FloodPlain that have spatially overlapping geometries.

This issue may well be beyond the scope of this workshop, but we would like any standardisation process or tool development to bear it in mind, so that they are flexible enough to allow spatial extensions. (One example we've come across where this wasn't happening was where a tool vendor limited string length to 250 characters: far too short for geometry).

### *2.5 Querying needs to be easier than SQL*

One advantage of RDF over relational data is that it is generally easier to construct SPARQL queries than SQL queries. It would be useful to have a SPARQL like query language that allows for the semantics of OWL DL (and eventually OWL1.1). Such a query language has been proposed in [4]. Given the lack of recursive queries in SPARQL it would be useful (at the very least) to allow for reasoning over a subset of the OWL DL semantics based on some of the tractable fragments defined in [5].

## References


[1] 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.

[2] Perez de Laborda C. and Conrad S. "Database to Semantic Web Mapping using RDF query languages". 25th International Conference on Conceptual Modelling, Tucson Arizona, November 2006.

[3] Geography Markup Language, Open Geospatial Consortium http://www.opengeospatial.org/standards/gml

[4] Sirin, E and Parsia, B. SPARQL DL: SPARQL Query for OWL-DL

[5] OWL 1.1 Web Ontology Language Tractable Fragments http://pellet.owldl.com/papers/sirin07sparqldl.pdfhttp://www.w3.org/Submission/owl11-tractable/