European Commission

ISA Action 1.17: A Reusable INSPIRE Reference Platform (ARE3NA)

Expert contract supporting the
Study on RDF and PIDs for INSPIRE
Deliverable D.EC.3.2

RDF in INSPIRE – Open issues, tools, and implications

Linda van den Brink

Joint Research Centre

This publication is a Deliverable of Action 1.17 of the Interoperability Solutions for European Public Administrations (ISA) Programme of the European Union, A Reusable INSPIRE Reference Platform (ARE3NA), managed by the Joint Research Centre, the European Commission's in-house science service.

Contents

Chapter 1

Introduction

This document provides a brief report (Deliverable 2):

- Outlining open issues or potential obstacles to the application of the proposed common methodology to other INSPIRE Annex Themes;
- Describing potential tools to be used for the transformation of INSPIRE-related source data (in their original format and schema) as well as INSPIRE-compliant data (in GML) to the generated RDF vocabularies. This shall include both proprietary and open source software and shall indicate if any development work is needed in the software itself or its documentation to support the methodology;
- Outlining any implications that using RDF as an encoding would have for other INSPIRE components (e.g. INSPIRE registers, network services, metadata, access control, applications and geoportals, data policy and monitoring & reporting).

Chapter 2

Open issues

This section describes the open issues that must be addressed for creating INSPIRE RDF vocabularies.

2.1 General

The experience thus far shows that it is not realistic to expect that fully automated conversion of INSPIRE UML models to RDFS/OWL is possible. Before and/or after conversion, manual steps are needed to map INSPIRE terms to well-known vocabularies, to indicate whether properties are global, specific to a package or specific to a class, etc. However, this is not insurmountable. The INSPIRE vocabulary is not a small one but still, the work needed to make them available as RDFS/OWL is doable.

As to conversion of data, however, if this turns out to be impossible to do automatically, that would be a big obstacle. Thus far in this study, we have not focussed on this topic. Experience we have from another project, unrelated to INSPIRE (see 3.2 for more information), makes us hopeful. However, more study is necessary to confirm that automated conversion of INSPIRE data to RDF on a large scale is feasible.

2.2 Missing external vocabularies

The INSPIRE model imports a lot of standardized information models, such as models from the ISO 19100 series, Observations and Measurements, etc. This presents a problem when translating the INSPIRE model to an RDF vocabulary: these external models are not available as RDF vocabularies.

We concluded that it is not possible to solve this issue at the moment – we need the official RDF vocabularies in order to model INSPIRE's relationship with them, but they are not available yet.

2.3 Real world things and spatial objects

In the linked data community it is common to make a separation between 'real world things' and the information you can get about these 'real world things' over the internet. The question is if this is feasible in the context of INSPIRE. In two well-known geo-linked data resources, GeoNames and DBPedia, the separation between the two seems to be in place. In the geo-information world however, traditionally there has not been a clear separation between real world thing and representation. In the geo-information world, the spatial object 'thing' is often an entity on a map, not a thing in the real (time and space) world, although it may represent a real thing.

Further experimentation should be done to study this question.

2.4 Voidability

There is no common design pattern for this in linked data. Voidable properties present a certain amount of difficulty to RDF under the open-world assumption. Just because the value of a property may not be given does not mean that there is no value for that property that could be given elsewhere.

We propose that no conversion rule is needed for the concept of voidability; however the decision still has to be made.

2.5 ISO 19150-2

One of the results of this project is that based on the experience gained, we can formulate some issues with and comments on the draft standard ISO 19150-2 (DIS). Action must be taken to take these issues to ISO/TC 211.

2.6 CityGML ontology

An experimental CityGML ontology is available, as was mentioned in our deliverable 1. This is of interest because of the relationship the Buildings theme has with CityGML. This means a possible action to propose / support finalizing this ontology and promoting its endorsement by a standards organization.

Chapter 3

Transformation tools

3.1 UML to RDF conversion tools

For conversion of UML models to RDF/RDFS/OWL vocabularies, the most relevant candidate is ShapeChange[1]. ShapeChange is an open source Java tool that takes application schemas constructed according to ISO 19109 from a UML model and derives implementation representations. RDF is one of the supported representations.

Currently the RDF output option of ShapeChange is experimental, but a version of ShapeChange supporting RDF output conform ISO 19150-2 (current DIS), and supporting the mapping of class- and property names to well-known vocabulary terms, will soon become available.

There are other tools that can convert UML models to RDF, but they are not ISO 19109 aware and not conform ISO 19150-2. We have not experimented with other standard-available tools.

One other option is to create a custom transformation (e.g. in XSLT or SPARQL[2]) from UML exchange format XMI to RDF/XML. In our experience (in the context of another project) this is not a very complex task, but seems superfluous since ShapeChange can already do the same task.

Note that no conversion method has been found that fully automates the task. There is always a necessity of configuration / defining mappings beforehand, and/or manually checking and adapting the conversion result to get a usable ontology.

Of course another option is not to automate this at all but have an expert perform this task manually.

3.2 INSPIRE data to RDF conversion tools

For conversion of INSPIRE data to RDF there are several options.

One option we studied in another project is conversion of GML data to RDF using a generic XSLT stylesheet[3]. It is based on the fact that GML has its roots in RDF and still has a basic object-property structure that can very easily be translated to triples. Because of this, it is easy to define a transformation for translating any correctly structured (that is, conformant to the object-property triple structure) GML data to RDFS/OWL automatically. As an experiment, we implemented such a transformation using XSLT 2.0. In this Generic-GML2RDF script, well-known GML content elements such as names and descriptions are mapped to their RDF equivalent. Objects, including nested features, data types and properties are recognized based on their place in the triple structure and are transformed accordingly.

The experimental implementation has 10 templates; counting whitespace and comments it has 98 lines. This shows the simplicity of the transformation. However, there are several issues to be resolved before using this for anything other than experimentation:

- As one of the other experts (CP) pointed out, there may be problems with the method, such as how to translate hyperlinks in the GML data to resources that reside in web services. This has not yet been considered.
- The stylesheet is highly experimental and should be extensively tested with INSPIRE data.
- Support for converting GML geometries to WKT is incomplete (only gml:Point and gml:Surface geometries at the moment, these are transformed to WKT) and should be extended, either by implementing rules for transforming geometries to WKT in XSLT, or by calling an external library (e.g. ogr2ogr) for this.

[1] http://shapechange.net
[2] http://topquadrantblog.blogspot.nl/2011/02/converting-uml-models-to-owl-part-1.html
[3] For a full description see http://ijsdir.jrc.ec.europa.eu/index.php/ijsdir/article/view/351

- Minor changes may be needed to support GML 3.2 (3.1 supported at the moment).

Other options:

- The same could also be done with ETL tooling such as FME.
- Convert the data from the relational database in which it is stored to RDF. There are various tools that sit as an extra layer on top of a database to allow access of the data as linked data. They allow you to query the database using SPARQL and get the data as RDF. E.g. D2RQ[4].
- Create linked data on top of OGC web services. E.g. GetLOD[5] takes data from WFS or WCS services and serves them as linked data.

[4] http://d2rq.org/dump-rdf

[5] http://www.planetek.it/eng/products/all_products/getlod

Chapter 4

Implications of using RDF on other INSPIRE components

4.1 General remark

We foresee using RDF as an *alternative encoding* for INSPIRE vocabularies and data. Hence it has no implications on policy or legislation. Most implications are of a technical nature; they are described below.

4.2 Geoportals

4.2.1 RDF support

An implication of using RDF encodings in INSPIRE is that geoportals should be extended to offer RDF as encoding format. "Offering RDF" implies:

- Possibly, publishing RDF files as downloadables. This is probably the easiest, but also less useful option.
- A SPARQL endpoint to the data.
- Publishing the data as linked data on the web, i.e. information about each resource (each feature) should be retrievable via HTTP using its URI.

The latter involves implementing content negotiation. The server transcribes requests for resources into the requested format on the basis of ID and supplies a document – the 'useful information' – to the client with reference to the http-returncode '303-redirect'. A technical infrastructure that does this must be set up, configured and maintained.

4.2.2 JSON

In addition to RDF/XML and Turtle, it is very interesting to consider JSON as encoding for linked data; more specifically a combination of GeoJSON and JSON-LD. As explained in our deliverable 1, this is a very promising option for publishing linked geo-data in an easily consumable form.

JSON-LD (a recent W3C recommendation) is a JSON encoding for Linked Data. It lets you add meaning to the terms and values in a JSON document. This is done inside a @context object that is either referenced from or embedded inside the JSON document. Applications that are not aware of what @context is, can simply ignore it while applications that are aware, can parse the @context and gain knowledge on the semantics of the JSON data.

GeoJSON and JSON-LD can be combined and can be offered as the web-encoding for linked geospatial data. GIS server software that already offers JSON as encoding could create GeoJSON-LD as well.

4.3 Registers

INSPIRE registers must be extended to support SKOS codelists. Again, these should not (just) be offered as downloadable files, but accessible through a SPARQL endpoint and published as linked data on the web including content negotiation.

4.4 Metadata

In order for SPARQL endpoints to be discoverable, they should be listed and described just as WMS or WFS services are. A candidate format for describing SPARQL endpoint metadata is VoiD[6].

[6] http://www.w3.org/TR/void/

4.5 Access control

Linked data resources are accessed through the HTTP protocol, so access control methods already used on the web can be used in this case as well. There seem to be no extra technical implications. The W3C Linked Data Platform specification[7] briefly states this in section 8.

There could be, however, a social implication, in the sense that links that turn out to be unaccessible due to access constraints will be perceived as unuseful and this could lead to low acceptance and uptake of linked INSPIRE data on the web[8]. Openness of data thus becomes very important in a linked data environment.

4.6 Monitoring & reporting

In the yearly monitoring one of the things evaluated is the accessibility of spatial data sets through view and download services. Accessibility through SPARQL endpoints and as linked data on the web could be added to the monitoring procedure.

7 http://www.w3.org/TR/ldp/

8 Source: http://www.pilod.nl/wiki/Boek/Portele