EEA and Linked Environment Data

A status report

Søren Roug¹

1. Transforming Reportnet

When the SEIS initiative was launched in 2008 one big question for EEA was how it would affect the Reportnet system. On the surface, with its centralisation, Reportnet seemed to be the antithesis of the SEIS principles. Nevertheless, we started to analyse what would be needed to transform Reportnet into a decentralised SEIS compliant system.

It became clear that reporters making deliveries to Reportnet is only part of the pipeline. Before-hand comes a lengthy process of agreeing to what data to collect and what data format to use between countries and the requester. Deciding where to place it is just a tiny part. It doesn’t really matter whether the dataset is stored at the country or at EEA. What matters is that these datasets are made to order, i.e. the idea of multiple use is already compromised. In that respect they are more like intermediate calculations than the end result.

But what about all the data the countries have available, but they are not required to deliver to any supra-governmental body? If the countries could somehow publish it with little effort, then EEA could collect that data and not request Reportnet deliveries.

The main problem is that there is no useful format to deliver in. INSPIRE has GML and PNG formats. The only equivalents for tabular data are office formats. In our world, where we have to merge data from 30 sources and often combine it with other data, office formats such as CSV and Excel entail lots of manual labour.

If the countries start to post data in comma separated values (CSV) and Excel formats for data that would in earlier times be Reportnet deliveries, then SEIS wouldn’t be progress!

On a parallel track we also worked on how to merge the current Reportnet deliveries and require less manual labour to do it. We settled on converting XML to RDF because RDF can hold any format. RDF can then be imported into a triple store and queried with the new language SPARQL.

RDF can be seen as a replacement/improvement to CSV. Its first benefit is that it is self-describing. The second is that in contrast to relational databases (SQL), the database schema is embedded in the data, and not outside of it.

¹ European Environment Agency, Copenhagen
We have therefore chosen to recommend that when datasets are made available for the Shared Environmental Information System, then they are also made available in RDF.

If our recommendation is taken up, then there are some beneficial side-effects and the rest of the document is going to present those.

2. **Transforming the work of data analysts**

This is the situation today. You search for some data on Google, or you go to an organisation’s website and search. Since datasets don’t have much text in them you’ll most likely find yourself doing the second operation. You might even have to do it in a foreign language if you’re trying to compare data across Europe.

The figure below illustrates how you would navigate a table of contents to find what you’re looking for. The screen shot is taken from Eurostat.

![Diagram of table of contents](image)

When you have located your dataset, then you download it. A helpful website will provide the data in several formats:
The dataset is ready to be used. The problem comes when you want to do other things with the dataset than what its creators have anticipated. Like load it into a database; join it with other data. It becomes a lot of manual work to convert it to the format you need.

2.1 A new approach

Imagine a solution, where you search for datasets using a central index. When you find what you need you simply click once to load it into your database – the triple store. The data is imported and it already knows what joins can be made with other datasets. This is what RDF allows for.

Having the data available as RDF changes the world. Suddenly the conversion is not an issue any longer. And if we factor in that storage space has become cheap, we can be bold and proactively load all the data we think we’ll need some day into our working database. If we regularly check status at the source, we can also keep our copy reasonably up-to-date.

What remains is to formulate queries and create visualisations of it.

2.2 Status

How far have we come? We have deployed two such triple stores. One for Reportnet deliveries called the Content Registry and one for validated environmental datasets called the Semantic Data Service.
We have created RDF dumps of several data sets: GEMET, EUNIS, ROD, (ITIS²), NUTS, NACE³ etc. and most XML-formatted Reportnet deliveries. Last, but not least: Most of the information products on our EEA website such as reports, indicators, datasets have RDF output of metadata and in some cases the data as well. All of it is loaded into one or both of the triple stores. The triple store is set up to automatically load all newly discovered data.

Let’s see how it works.

### 2.3 Example of query on Reportnet deliveries

The first example focuses on the original need for a solution: *To merge Reportnet deliveries.* With RDF, since all data is loaded, the deliveries are automatically merged.

The Habitats Directive article 17 deliveries contain information of species of concern. We want to know: What are the assessments for the future of the European otter?

To make such a query we need to know beforehand what data was reported and have some idea about how it was structured in the delivery. There is no way around it. It is however possible to use the tool to investigate the data.

The article 17 deliveries are linked to the EUNIS database. The unique identifier for the otter is http://eunis.eea.europa.eu/species/1435. By using identifiers we avoid ambiguous results.

This is the query:

```sql
PREFIX art17: <http://rdfdata.eionet.europa.eu/article17/ontology/>
PREFIX eea: <http://rdfdata.eionet.europa.eu/eea/ontology/>

SELECT ?country ?region ?future WHERE {
  art17:hasRegionalReport ?report .
  ?report art17:conclusion_future ?future;
  art17:forCountry ?curl;
  art17:region ?bgregion.
  ?curl eea:name ?country
} ORDER BY ?country ?region
```

This query was executed on [http://cr.eionet.europa.eu/sparql](http://cr.eionet.europa.eu/sparql)

Results:

---

² ITIS is the North American species database. It’s content is published as a SQL-dump monthly. We have converted it into triples.
³ We downloaded the NUTS and NACE code lists, converted them to triples and gave them back to Eurostat to place them on their RAMON website.
<table>
<thead>
<tr>
<th>country</th>
<th>region</th>
<th>future</th>
</tr>
</thead>
<tbody>
<tr>
<td>Austria</td>
<td>Alpine</td>
<td>Inadequate (U1)</td>
</tr>
<tr>
<td>Austria</td>
<td>Continental</td>
<td>Inadequate (U1)</td>
</tr>
<tr>
<td>Belgium</td>
<td>Atlantic</td>
<td>Bad (U2)</td>
</tr>
<tr>
<td>Belgium</td>
<td>Continental</td>
<td>Bad but improving (U2+)</td>
</tr>
<tr>
<td>Czech Republic</td>
<td>Continental</td>
<td>Favourable (FV)</td>
</tr>
<tr>
<td>Czech Republic</td>
<td>Pannonian</td>
<td>Favourable (FV)</td>
</tr>
<tr>
<td>Estonia</td>
<td>Boreal</td>
<td>Favourable (FV)</td>
</tr>
</tbody>
</table>

### 2.4 Checking the quality of our data

We also have a need to organise our list of species in EUNIS. One approach is to check what other databases have. We have created links between EUNIS and ITIS. Then we have imported the ITIS database into the triple store and we can now make comparisons. The following shows the species that are considered valid names in EUNIS, but invalid in ITIS.

This query shows how SPARQL joins three records from two sources. The joins look like this:

It was executed on [http://semantic.eea.europa.eu/sparql](http://semantic.eea.europa.eu/sparql).

```sparql
PREFIX e: <http://eunis.eea.europa.eu/rdf/species-schema.rdf#>
PREFIX itis: <http://eunis.eea.europa.eu/rdf/schema.rdf#>
PREFIX dwc: <http://rs.tdwg.org/dwc/terms/>

  ?eunisurl e:validName 1;
  e:sameSynonym ?itisurl;
  e:binomialName ?eunisname;
  dwc:scientificNameAuthorship ?eunisauthor.
  ?itisurl itis:nameUsage "invalid",?usage;
  itis:completeness-name ?itisname;
  itis:hasAuthor ?auurl.
  ?auurl itis:shortAuthor ?itisauthor
}
```
### Results

<table>
<thead>
<tr>
<th>eunisname</th>
<th>eunisauthor</th>
<th>itisname</th>
<th>itisauthor</th>
<th>usage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chondrocladia alaskensis</td>
<td>Lambe, 1900</td>
<td>Chondrocladia alaskensis</td>
<td>Lambe 1895</td>
<td>invalid</td>
</tr>
<tr>
<td>Myxilla parasitica</td>
<td>(Lambe, 1900)</td>
<td>Myxilla parasitica</td>
<td>Lambe 1893</td>
<td>invalid</td>
</tr>
<tr>
<td>Hymedesmia primitiva</td>
<td>Lundbeck, 1910</td>
<td>Hymedesmia primitiva</td>
<td>Lundbeck 1910</td>
<td>invalid</td>
</tr>
<tr>
<td>Asbestopluma lycopodium</td>
<td>(Levinsen, 1886)</td>
<td>Asbestopluma lycopodium</td>
<td>Levinsen 1886</td>
<td>invalid</td>
</tr>
<tr>
<td>Esperiopsis rigida</td>
<td>Lambe, 1900</td>
<td>Esperiopsis rigida</td>
<td>Lambe 1893</td>
<td>invalid</td>
</tr>
<tr>
<td>Cordylophora lacustris</td>
<td>Allman, 1844</td>
<td>Cordylophora lacustris</td>
<td>Allman 1844</td>
<td>invalid</td>
</tr>
<tr>
<td>Tubularia crocea</td>
<td>(Agassiz, 1862)</td>
<td>Tubularia crocea</td>
<td>L Agassiz 1862</td>
<td>invalid</td>
</tr>
<tr>
<td>Tubularia larynx</td>
<td>Ellis &amp; Solander, 1786</td>
<td>Tubularia larynx</td>
<td>Ellis and Solander 1786</td>
<td>invalid</td>
</tr>
<tr>
<td>Ectopleura dumortieri</td>
<td>(Van Beneden, 1844)</td>
<td>Ectopleura dumortieri</td>
<td>van Beneden 1844</td>
<td>invalid</td>
</tr>
</tbody>
</table>

### 3. Query examples of SDMX data

SDMX is an important format since it is so widely accepted. SDMX is not Linked Data friendly. The concepts, code lists, datasets, and observations are not named with URIs or routinely exposed to browsers and other web-crawlers. There are however now tools and standards to make it into RDF.

The International Food Policy Research Institute⁴ (IFPRI) has made their Global Hunger Index available as RDF. It is originally an SDMX dataset, which they transformed according to RDF Cube specifications and placed at [http://data.ifpri.org/rdf/ghi/](http://data.ifpri.org/rdf/ghi/). We just imported the dataset as we found it into our SPARQL database.

---

⁴ [http://www.ifpri.org/](http://www.ifpri.org/)
Each observation in this file can have up to four values:

- ghi - Global Hunger Index
- supportingData-cm - Child mortality rate (%)
- supportingData-pun - Proportion of undernourished in the population (%)
- supportingData-cuw - Prevalence of underweight in children under five years (%)

Due to more data being available, the examples use supportingData-cm.

### 3.1 List of countries with the highest values of under 5 mortality rates

The first query is a simple report of mortality rates using only the data in the source. It is arguably easier to just download the spreadsheet and sort the table. We are interested in the child mortality rate per country for 2010.

```
PREFIX cube: <http://purl.org/linked-data/cube#>
PREFIX ghi: <http://data.ifpri.org/rdf/ghi/structure/>#
PREFIX gn: <http://www.geonames.org/ontology#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT ?country ?childmortality ?ghi WHERE {
    _:subject a cube:Observation ;
    ghi:refPeriod "2010-01-01"^^xsd:date ;
    ghi:refArea ?curl;
    ghi:supportingData-cm ?childmortality .
    OPTIONAL { _:subject ghi:ghi ?ghi } 
    ?curl rdfs:label ?country
} ORDER BY DESC(xsd:decimal(?ghi))
```

Results as executed on [http://semantic.eea.europa.eu/sparql](http://semantic.eea.europa.eu/sparql):

<table>
<thead>
<tr>
<th>country</th>
<th>childmortality</th>
<th>ghi</th>
</tr>
</thead>
<tbody>
<tr>
<td>Democratic Republic of the Congo</td>
<td>19.9</td>
<td>41</td>
</tr>
<tr>
<td>Burundi</td>
<td>16.8</td>
<td>38.3</td>
</tr>
<tr>
<td>Eritrea</td>
<td>5.8</td>
<td>35.7</td>
</tr>
<tr>
<td>Chad</td>
<td>20.9</td>
<td>30.9</td>
</tr>
<tr>
<td>Yemen</td>
<td>6.9</td>
<td>27.3</td>
</tr>
</tbody>
</table>
### 3.2 Adding information to the core data

Because the GHI data is linked to the GeoNames database for countries we can join with information from same. Here we join to get population size and continent to get some context.

```
PREFIX cube: <http://purl.org/linked-data/cube#>
PREFIX ghi: <http://data.ifpri.org/rdf/ghi/structure/>#
PREFIX gn: <http://www.geonames.org/ontology/>#

    _:subject a cube:Observation ;
    ghi:refPeriod "2010-01-01"^^xsd:date ;
    ghi:refArea ?curl;
    ghi:supportingData-cm ?childmortality .
    OPTIONAL { _:subject ghi:ghi ?ghi }
    ?curl gn:name ?country;
    gn:population ?pop;
    gn:parentFeature ?parent.
    ?parent gn:name ?continent
} ORDER BY DESC(xsd:decimal(?ghi))
```

Results:

<table>
<thead>
<tr>
<th>country</th>
<th>childmortality</th>
<th>ghi</th>
<th>continent</th>
<th>population</th>
</tr>
</thead>
<tbody>
<tr>
<td>Democratic Republic of the Congo</td>
<td>19.9</td>
<td>41</td>
<td>Africa</td>
<td>70.916439</td>
</tr>
<tr>
<td>Burundi</td>
<td>16.8</td>
<td>38.3</td>
<td>Africa</td>
<td>9.863117</td>
</tr>
<tr>
<td>Eritrea</td>
<td>5.8</td>
<td>35.7</td>
<td>Africa</td>
<td>5.792984</td>
</tr>
<tr>
<td>Chad</td>
<td>20.9</td>
<td>30.9</td>
<td>Africa</td>
<td>10.543464</td>
</tr>
<tr>
<td>Yemen</td>
<td>6.9</td>
<td>27.3</td>
<td>Asia</td>
<td>23.495361</td>
</tr>
<tr>
<td>Angola</td>
<td>22.0</td>
<td>27.2</td>
<td>Africa</td>
<td>13.068161</td>
</tr>
</tbody>
</table>
### 3.3 Example using Eurostat data

The LATC⁵ and LOD2⁶ research projects were set up with FP7 funding to support public administrations to use and produce Linked Data. As part of their work programme they have converted all of Eurostat’s datasets’ to RDF. We’ve imported some of their converted datasets into http://semantic.eea.europa.eu/.

The query below uses dataset ‘demo_r_pjanaggr3’, which contains ‘Population by sex and age groups on 1 January - NUTS level 3 regions.’ We need populations for NUTS level 2 and we therefore aggregate the dataset by using the NUTS vocabulary to find the parent regions.

We only want data for 2007, and both sexes. We then join the data with the ‘env_n2_wu’ dataset, which contains ‘Water use (NUTS2) - mio m³.’ We can then find the regions with the most domestic water (code W18_2_7_2) use per million inhabitants.

```sparql
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX qb: <http://purl.org/linked-data/cube#>
PREFIX e: <http://ontologycentral.com/2009/01/euros tat/ns#>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX g: <http://eurostat.linked-statistics.org/on tologies/geographic.rdf#>
PREFIX dataset: <http://eurostat.linked-statistics.org/data/>

SELECT ?nuts2
    SUM(xsd:decimal(?pop)) AS ?population
    ?wateruse
    xsd:decimal(?wateruse)*1000000/SUM(xsd:decimal(?pop)) AS ?percapita
WHERE {
  ?observation qb:dataset dataset:demo_r_pjanaggr3 ;
    e:time <http://eurostat.linked-statistics.org/dic/time#2007>;
    e:age <http://eurostat.linked-statistics.org/dic/age#TOTAL>;
    e:sex <http://eurostat.linked-statistics.org/dic/sex#F>;
    e:geo ?ugeo;
    sdmx-measure:obsValue ?pop.
}
```

---

⁵ http://latc-project.eu/
⁶ http://lod2.eu/
⁷ The datasets are available at http://eurostat.linked-statistics.org/.
Results:

<table>
<thead>
<tr>
<th>nuts2</th>
<th>population</th>
<th>wateruse</th>
<th>percapita</th>
</tr>
</thead>
<tbody>
<tr>
<td>(MK00) Poranesna jugoslovenska Republika Makedonija (provisional code)</td>
<td>1018202</td>
<td>226.4</td>
<td>222.352735508278318</td>
</tr>
<tr>
<td>(CY00) Cyprus</td>
<td>395324</td>
<td>73.9</td>
<td>186.935273345407817</td>
</tr>
<tr>
<td>(DE60) Hamburg</td>
<td>898050</td>
<td>95.06</td>
<td>105.851567284672346</td>
</tr>
<tr>
<td>(DE21) Oberbayern</td>
<td>2185884</td>
<td>225.34</td>
<td>103.088727489656359</td>
</tr>
<tr>
<td>(DE91) Braunschweig</td>
<td>837569</td>
<td>84.51</td>
<td>100.899149801389497</td>
</tr>
</tbody>
</table>

4. Conclusion

We are well on our way and we’re already reaping some benefits. We have for example added a feature to ROD to show products derived from Reportnet dataflows.

Next year we will make more datasets available as RDF. We’ll create more tools and more integration. E.g. when we have RDF output from our reference databases, we can improve the QA systems in Reportnet. The scripting language for our automatic QA can already integrate with SPARQL.