Model Driven Software Development using semantic web technology
RDF-store as repository

14. June 2013
Lars Schnell, Senior IT Architect, Statnett
CIM Users Group Spring 2013 Meeting
Agenda

1. Abstract and Statnett background
2. RDF store concept with OWL profile together with CIM/RDF exchange format
3. How to adapt CIM model/profile/version in a generic way to expose an API for subscriber applications together with OWL and RDF store?
4. Short demo
5. Questions / Wrap-up
Abstract of presentation

**Model Driven Software Development using semantic web technology**

**RDF-store as repository**

"Model Driven Software Development" project for using Resource Description Framework (RDF)- store as CIMXML repository for

- Operational Information System (OIS) - used for Outage Management
- Market Management System
- Other Statnett applications as data warehouse

The presentation will explain RDF/OWL concepts and how to process from modeling CIM Equipment profile (EQ) in Unified Modeling Language (UML and OWL) to generated API interfaces which than can used to expose CIM/RDF power network model content through RDF store.
**Statnett Vision with CIM (power network model)**

*Interoperability, within the context of Power Network delivery, is the ability of disparate and diverse applications/services to interact towards mutually beneficial and agreed common goals, involving the sharing of information in a common way.*
Statnett Vision with CIM (power network model)
Agenda

1. Abstract and Statnett background
2. RDF store concept with OWL profile together with CIM/RDF exchange format
3. How to adapt CIM model/profile/version in a generic way to expose an API for subscriber applications together with OWL and RDF store?
4. Short demo
5. Questions / Wrap-up
As every time some explanations in complex world…

- **Ontologies**
  - Ontologies provide a shared understanding of a domain
  - They provide background knowledge to systems to automatize certain tasks
  - By the process of annotation, knowledge can be linked to ontologies
    - Example: “Nicole Kidman” (Text) linked to concept Actress
    - In our ontology we also know that an actress always is female and a person

- **Semantic Web**
  - “An extension of the current web in which information is given well-defined meaning, better enabling computers and people to work in cooperation.”
  - Semantic Web is about:
    - (Web) Data Annotation and Linking/Integration over Web

- **RDF store / RDF**
  - Much like a relational database, one stores information in a triplestore and retrieves it via a query language. Unlike a relational database, a triplestore is optimized for the storage and retrieval of triples. In addition to queries, triples can usually be imported/exported using Resource Description Framework (RDF) and other formats.
    (http://en.wikipedia.org/wiki/Triplestore)
  - Resource Description Framework (RDF) is the HTML of the Semantic Web
    - Based on triples <subject, predicate, object>
RDF short recap

- Data model for expressing knowledge

- Basic building block: statement

```xml
<cim:Line rdf:ID="_9dfd4c49-028c-6964-e040-1e828c94e002">
  <cim:IdentifiedObject.name>C420ADALRINGERIKE</cim:IdentifiedObject.name>
  <cim:IdentifiedObject.aliasName>ÅdalRingerike</cim:IdentifiedObject.aliasName>
  <cim:Line.Region rdf:resource="#a653d4bf-4959-1197-e040-1e828c947b3b"/>
</cim:Line>
```

- Groups of statements form graphs
RDF short recap

- RDF Schema is a vocabulary description language
  - It allows specification of domain vocabulary and a way to structure it
  - Class, Property, subClassOf, subPropertyOf, domain, range

- Formal semantics add simple reasoning capabilities:
  - Class and property classification
  - Domain and range inference
Turtle syntax of semantic web

- **Subject / Predicate / Object**

<table>
<thead>
<tr>
<th>Subject</th>
<th>Predicate</th>
<th>Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>&lt;file/DS4 SNAG1228 CIM15 20130528.xml#_0d1f46e-028c-6063-a040-18928c9c0402&gt;</td>
<td>rdf:type</td>
<td>cim:DateTime</td>
</tr>
<tr>
<td>&lt;file/DS4 SNAG1228 CIM15 20130528.xml#_0d1f46e-028c-6063-a040-18928c9c0402&gt;</td>
<td>rdf:type</td>
<td>cim:ConnectivityNodeContainer</td>
</tr>
<tr>
<td>&lt;file/DS4 SNAG1228 CIM15 20130528.xml#_0d1f46e-028c-6063-a040-18928c9c0402&gt;</td>
<td>rdf:type</td>
<td>cim:IdentifiedObject</td>
</tr>
<tr>
<td>&lt;file/DS4 SNAG1228 CIM15 20130528.xml#_0d1f46e-028c-6063-a040-18928c9c0402&gt;</td>
<td>rdf:type</td>
<td>cim:PowerSystemResource</td>
</tr>
<tr>
<td>&lt;file/DS4 SNAG1228 CIM15 20130528.xml#_0d1f46e-028c-6063-a040-18928c9c0402&gt;</td>
<td>rdf:type</td>
<td>cim:EquipmentContainer</td>
</tr>
<tr>
<td>&lt;file/DS4 SNAG1228 CIM15 20130528.xml#_0d1f46e-028c-6063-a040-18928c9c0402&gt;</td>
<td>rdf:type</td>
<td>cim:Region</td>
</tr>
<tr>
<td>&lt;file/DS4 SNAG1228 CIM15 20130528.xml#_0d1f46e-028c-6063-a040-18928c9c0402&gt;</td>
<td>rdf:type</td>
<td>cim:IdentifiedObject.name</td>
</tr>
<tr>
<td>&lt;file/DS4 SNAG1228 CIM15 20130528.xml#_0d1f46e-028c-6063-a040-18928c9c0402&gt;</td>
<td>rdf:type</td>
<td>cim:IdentifiedObject.aliasName</td>
</tr>
<tr>
<td>&lt;file/DS4 SNAG1228 CIM15 20130528.xml#_0d1f46e-028c-6063-a040-18928c9c0402&gt;</td>
<td>rdf:type</td>
<td>cim:IdentifiedObject.mRID</td>
</tr>
</tbody>
</table>

June 13, 2013 / CIM Users Group Spring 2013 Meeting
RDF framework (SESAM/OWLIM)

- **Framework**
  - To combine RDF and OWL you can find a couple of open source and product provider in the market space
  - We using open source Sesame (OpenRDF) and OWLIM semantic repo store ([http://www.ontotext.com/owlim](http://www.ontotext.com/owlim) and [http://www.openrdf.org/](http://www.openrdf.org/))

- **Reasons why we use it**
  - In-memory implementation and very god performance as extremely fast loading of data (including inference and storage)
  - Stability of Sesame and OWLIM
  - Follow most the W3 open standards (OWL, RDF, SPARQL)
  - Support of other tools integrated with Sesame, e.g. ontology editors like Protégé, TopBraid Composer and compatible with CIMTool
  - Small infrastructure foot print and easy installation

- **Integration into application portfolio**
  - The easiest way for developers to integrate applications with SESAME/OWLIM is to use it with the Alibaba and OWLIM/Sesame framework as a set of libraries with different integration technologies as
    - HTTP/REST (.NET, Java, …)
    - API (Java)
    - DDL / ETL
RDF rammeverk (SESAM/OWLIM)

- Applications can communicate over HTTP/REST and JAVA API with a RDF/OWLIM server and update data or do queries
- RDF - The core RDF model, containing objects and interfaces for URIs, blank nodes, literals, statements
- SAIL - Storage and Inference Layer / System API for 'wrapping' storage backend
- Declarative Querying through SeRQL or SPARQL
- RIO - RDF I/O Set of parsers and writers for RDF/XML, Turtle, N3, N-Triples
- RDF API – Offers developer-friendly methods for manipulating RDF data (query, adding, removing, updating)

June 13, 2013 / CIM Users Group Spring 2013
Agenda

1. Abstract and Statnett background
2. RDF store concept with OWL profile together with CIM/RDF exchange format
3. How to adapt CIM model/profile/version in a generic way to expose an API for subscriber applications together with OWL and RDF store?
4. Short demo
5. Questions / Wrap-up
CIM information model and RDF framework

**Challenge**
- How to adapt "abstract" CIM UML model (61970, 61968, Statnett profile) and content for applications like Data Warehouse, Outage Management and Market without provide an "custom" API/framework?
  - Adapt "smart" CIM new version and profile changes with less human coding/development
  - Release an API for the different CIM/Profile versions to provide a service oriented service interface
  - Use open software standards and scalable infrastructure to adapt you business needs with shorter times of releases and reduced costs

**Solution**
- Not simple with the complexity/richness of CIM and different technologies provided by software vendors including staff which have software engineering and industry knowledge
- Let see how Statnett approached a solution … it is not magic 😊
The diagrams show the conceptual model of how from the CIM UML model (Statnett EQ profile version) machine interfaces generated without manual code implementation.

Applications can use the interfaces API to access version/content of our CIM (profile) power network model.
Agenda

1. Abstract and Statnett background
2. RDF store concept with OWL profile together with CIM/RDF exchange format
3. How to adapt CIM model/profile/version in a generic way to expose an API for subscriber applications together with OWL and RDF store?
4. Short demo
5. Questions / Wrap-up
Demo

- JAVA API
Demo

- REST API
**DEMO**

**SPARQL**

```
PREFIX rdfs:<http://www.w3.org/2000/01/rdf-schema#>
PREFIX cim:<http://iec.ch/TC57/2010/CIM-schema-cim15#>
PREFIX rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX SN:<http://www.statnett.no/CIM-schema-cim15-extension#>

SELECT ?line ?lineName ?region ?regionName
{
  ?line cim:IdentifiedObject.aliasName ?lineName .
  ?region cim:IdentifiedObject.aliasName ?regionName .
  FILTER (?regionName = "NORDLAND")
}
```

**Query Result (130)**

```
<table>
<thead>
<tr>
<th>Line</th>
<th>LineName</th>
<th>Region</th>
<th>RegionName</th>
</tr>
</thead>
<tbody>
<tr>
<td>[file://Cimen_Statsnet_20100223.6.4.0.12] A52A0F3F50625E4A244642981C6C38FA</td>
<td>716 Linbro:02.20.01</td>
<td>NORDLAND</td>
<td></td>
</tr>
<tr>
<td>[file://Cimen_Statsnet_20100223.6.4.0.12] A52A0F3F50625E4A244642981C6C38FA</td>
<td>716 Linbro:02.20.01</td>
<td>NORDLAND</td>
<td></td>
</tr>
<tr>
<td>[file://Cimen_Statsnet_20100223.6.4.0.12] A52A0F3F50625E4A244642981C6C38FA</td>
<td>716 Linbro:02.20.01</td>
<td>NORDLAND</td>
<td></td>
</tr>
<tr>
<td>[file://Cimen_Statsnet_20100223.6.4.0.12] A52A0F3F50625E4A244642981C6C38FA</td>
<td>716 Linbro:02.20.01</td>
<td>NORDLAND</td>
<td></td>
</tr>
</tbody>
</table>
```
Agenda

1. Abstract and Statnett background
2. RDF store concept with OWL profile together with CIM/RDF exchange format
3. How to adapt CIM model/profile/version in a generic way to expose an API for subscriber applications together with OWL and RDF store?
4. Short demo
5. Questions / Wrap-up