Modeling and Querying Ontologies

Part 1 : RDF and RDFS Modeling

In this first part of the tutorial. We will use the RDF and RDFS languages to convert a simple text into an RDF graph. The idea here is to see how RDF and RDFS enable knowledge representation and manipulation.

"Wolfgang Amadeus Mozart (27 January 1756 – 5 December 1791), known as Mozart, was a prolific and influential composer of the classical era.

Mozart was Born in Salzburg and death at Vienna, the capital of Austria. He is the son of Léopold Mozart and Constance Weber's husband. At 17, Mozart was engaged as a musician at the Salzburg court. He composed more than 600 works. Among his best known symphonies, we can find the symphony no. 41 (Jupiter), symphony no. 31 (Paris) and symphony no. 38 (Prague). Jupiter symphonie is composed of four parts :

1. *Allegro Vivace*
2. *Andante Cantabile*
3. *Menuetto*
4. *Molto Allegro*

1. From this text, extract manually the entities (person, city, …), the relations (son of, capital of, …) and properties (born at, …)
2. Create a file that describes your entities, relations and properties using the rdf language and validate it using the W3C RDF Validator
3. Using RDFS, create a simple ontology that models the classes and properties of this text and domain
4. Optional question : Explore the DBpedia Spotlight tool to automatically extract the entities from the text and attach them to

Notes :

- Always identify your resources with a unique URI : Namespace + Local identifier
- Once you have defined your classes, relations and properties, search in the linked open vocabularies (LOV) website for vocabularies that can be used to replace the ones that you have created

Part 2 : Jena API

Jean is a free and open source Java framework for building Semantic Web and Linked Data applications. It is composed of different APIs that implement the concept of the Semantic Web Standard. These APIs interact together to enable the user process RDF Graphs.

We will use Jena in this tutorial as the main framework to build our semantic web application. To begin, please visit https://jena.apache.org/tutorials/index.html and take the RDF core API tutorial. You can fin tutorials in other languages like French Simplified Chinese.

1. Create a Java project and import the Jena library
2. Declare a new Class and declare all the entities and relations of your graph
3. List all the statements of your graph using a statement iterator
4. One you have created your model, write it to a file using a Jena model writer

Part 3 : SPARQL Querying

In the two first parts of this tutorial. We have leveraged Jena API capabilities to model and process RDF Graph (Jena Model). In this part, our objective is to learn how to query RDF Graphs using the SPAQL querying language.

We will use the generic SPARQL Endpoint available at this address : http://sparql.org/sparql.html. The RDF document at www.w3.org/2001/sw/SW-FAQ-feed.rdf, corresponds to an RSS feed for generating the page www.w3.org/RDF/FAQ . The RSS feed is declared as an instance of the class rss:channel. It contains articles, which are instances of the class rss:item.

Let's begin with a simple query :

```
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rss: <http://purl.org/rss/1.0/>
PREFIX swq: <http://www.w3.org/2001/sw/SW-FAQ#>
SELECT DISTINCT ?varClass
FROM <http://www.w3.org/2001/sw/SW-FAQ-feed.rdf>
WHERE { ?s rdf:type ?varClass }
```

Using SPARQL, write the queries to answer the following questions :

1. What is the title (rss:title) of the RSS feed (rss:channel) ?

2. What are the titles of the published items (rss:item) in the RSS feed?

3. Give the first ten items of the RSS feed ?

4. What is the penultimate item of the RSS feed (chronological order) ?

5. Is there any items published at the same date ?

6. What items have been published on 2007-04-12??

(Suggestion : date format "20070412T00:00+00:00"^^xsd:string)

7. Give the list of authors (without repetition) items in this RSS feed ?

8. Does the RSS feed use the "title" property defined by the Dublin Core namespace ? (dublincore.org/)

9. Does the RSS feed use the "subject" property defined by the Dublin Core namespace ?

10. List the publication dates of articles in the RSS feed.