Technologies du Web sémantique

Semester work 1 - Enriched Trails

The general goal of this project is to produce an enriched description of a GPS track that represents a walking trail. For this you will use three data sources:

- a set of GPX tracks, stored on the course moodle site
- Open Street Map: a set of geographic entities covering the whole earth
- DBpedia: an RDF graph derived from Wikipedia and Wikidata

You must create an application that for a given a GPX track produces a mini tourist information guide (in the form of an HTML page) about the surroundings of the track. In OSM you will find nodes that represent the villages, hamlets, mountain passes, peaks, amenities (restaurant, water point, . . .). In dbpedia there are additional information about some points of interest (cities, villages, monuments, moutains)

To do

1. Create an RDF graph that represent the GPX tracks and the information from OSM that are relevant for these tracks.

- define the RDFS or OWL schema of your graph
- write simple scripts to transform the GPX and OSM data (in XML) into RDF triples
- upload the schema and data into a triple store

2. Write a program that queries your graph and the dbpedia graph to produce the tourist information guide for each GPX track.

3. Write a short report that describes your schema, the mapping from the data sources to your graph, and what your application is doing.

Technical information

GPX Data

A GPX file describes tracks (lists of trak points) and waypoints (specific points along a track)

```
<trk>
  <name>Pointe des Jottis à partir du Châtelet</name>
  <trkseg>
   <trkpt lat="46.216693" lon="6.510028">
    <ele>1052</ele>
    <time>2019-07-18T06:41:49+02:00</time>
   </trkpt>
```

```
  <trkpt lat="46.216804" lon="6.510076">
   <ele>1053</ele>
   <time>2019-10-21T12:37:51+02:00</time>
  </trkpt>
   ...
   ...
 </trkseg>
</trk>
```

The GPS Exchange Format (GPX) is described in this wikipedia article : https://en.wikipedia.org/wiki/GPS_Exchange_Format

Open Street Map

Open Street Map data can be loaded with the following URL scheme:

https://api.openstreetmap.org/api/0.6/map?bbox=longitude-min,latitude-min,longitued-max,latitude-max

for example

https://api.openstreetmap.org/api/0.6/map?bbox=11.54,48.14,11.543,48.145

More information at https://wiki.openstreetmap.org/wiki/Downloading_data .

An OSM file describes geographic entities either as nodes

```
 <node id="369314798" visible="true" version="2" changeset="29685197"
    timestamp="2015-03-23T18:38:56Z" user="tartiflex" uid="507417"
    lat="46.2523028" lon="6.5775254">
  <tag k="ele" v="1650"/>
  <tag k="name" v="Pointe des Riandets"/>
  <tag k="natural" v="peak"/>
  <tag k="source" v="cadastre-dgi-fr"/>
 </node>
```

or as "ways"

```
<way id="93493533" visible="true" version="4" changeset="71444385"
timestamp="2019-06-20T13:21:01Z" user="Yugo74" uid="1887927">
  <nd ref="1083904565"/>
  <nd ref="1083904609"/>
  <nd ref="5777821791"/>
  <nd ref="1083904616"/>
  <nd ref="1083904577"/>
  <nd ref="1083904584"/>
  <nd ref="1083904561"/>
  <nd ref="1083904565"/>
  <tag k="addr:city" v="Boëge"/>
```

```
<tag k="addr:housenumber" v="227"/>
<tag k="addr:postcode" v="74420"/>
<tag k="addr:street" v="Rue de la Vallée Verte"/>
<tag k="brand" v="Spar"/>
<tag k="brand:wikidata" v="Q610492"/>
<tag k="brand:wikipedia" v="en:Spar (retailer)"/>
<tag k="building" v="supermarket"/>
<tag k="building:levels" v="1"/>
<tag k="name" v="Spar"/>
<tag k="official_name" v="Spar Boëge"/>
<tag k="operator" v="Spar"/>
<tag k="phone" v="+33 4 50 39 10 49"/>
<tag k="shop" v="supermarket"/>
<tag k="source" v="cadastre-dgi-fr "/>
<tag k="website" v="https://magasins.spar.fr/fr/s/spar-boege"/>
</way>
```

The complete description is in https://wiki.openstreetmap.org/wiki/OSM_XML and https://wiki.openstreetmap.org/wiki/Elements .

DBpedia

The dbpedia graph use the vocabulary defined in the DBpedia ontology. This ontology can be browsed from http://dbpedia.org/ontology/Place (Place is probably the most important class for our application)

To access a SPARQL endpoint, in this case dbpedia, from a python program use the *sparqlwrapper* library. Everything is explained in https://github.com/RDFLib/sparqlwrapper , with an example.

To interactively query dbpedia go to http://dbpedia.org/sparql