Master MIASHS
UFR SHS, Université Grenoble Alpes

Manuel Atencia Arcas and Jérôme David
2018-2019

UE Web Sémantique

Project – A semantic web album application

As part of the evaluation of the course, you will have to develop a semantic web album application using the semantic web technologies that we have seen in the course (RDF, SPARQL, RDF Schema and OWL) and the Apache Jena Java framework. This project will be in conjunction with the Java EE course. In what follows, we focus on the Semantic Web part of the project.

The album application will allow users to share their albums with other users, and to search for pictures both in their albums and other users' albums, according to some search criteria. In order to enhance this search, users will annotate their pictures before. The Semantic Web part of this project mainly concerns the tagging and search of the application. Below we elaborate more on these functionalities.

Tagging

The album application first will enable users to tag the pictures of an album of them. This tagging will be based on the following four Ws questions:

- Who is in the picture?
- What is in the picture?
- Where was the picture taken?
- When was the picture taken?

The user is supposed to answer each of these questions. For example, to the question *Who is in the picture?* the user could answer *Manuel Atencia*, or *Manuel Atencia* and *Jérôme David*; to the question *What is in the picture?* he could answer the *Eiffel Tower* or *a mountain*; to the question *Where was the picture taken?* he could answer *Paris* or *Grenoble*; and to the question *When was the picture taken?* he could answer *12/5/2014* or *summer 2015*.

In addition to these questions, other informations could be enquired: the title of the picture, the author of the picture, the type of the picture (if it is a family picture, a birthday party picture…)…

Searching for pictures

Once the tagging is done, the application will enable users to search for pictures in their albums and other users' albums, according to some search critera. These criteria should help users to find, for example,

- All pictures depicting a particular person, for instance, Manuel Atencia.
- All pictures depicting some people together, for example, Manuel Atencia and Jérôme David.
- All pictures with people in them.
- All pictures with no people in them.
- All pictures of Manuel Atencia's friends.
- All pictures of house-warming parties (*crémaillères*), farewell parties (*pots de départ*)…
- All house-warming party pictures depicting Manuel Atencia.

- All pictures within a particular period of time (from Feb. 2014 to Apr. 2014, in summer, on Christmas, on holidays…).
- All pictures in a particular place (Grenoble, the region of Isère, France…).
- All pictures of an object (the Eiffel Tower, a mountain, a beach…).
- All pictures of a particular animal (a cow, a dog…)
- A user's selfies.
- …

Use of semantic web technologies

You will have to use semantic web technologies for developping your album application.

You will have to describe in RDF the information provided by users while tagging the pictures of an album. For this, you will have to create URIs for the "things" involved in these pictures, to identify properties and their values, whether they are objects or literals (see Figure 1).

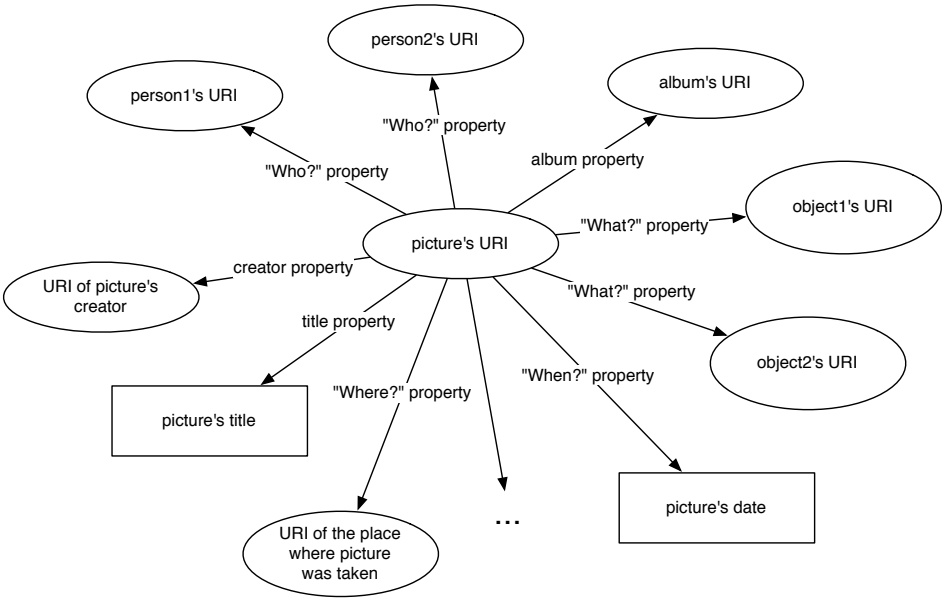

Figure 1. An RDF description of a picture.

Also, you will use RDF-S and OWL to create an ontology for your application, including classes and properties, and distinguished instances (for example, the class Party, the property isBrotherOf, the instance jeromeDavid). This ontology will have to include meta-information such as domain and range statements, property characteristics, class disjointness, class membership... We will assume that this ontology does not change. The ways to create new ontological entites are out of the scope of this project.

Finally, you will use SPARQL to implement the picture search engine. This may require to ask external SPARQL endpoints (e.g. DBpedia) and query your data using OWL reasoning (e.g. to identify a picture as a party picture even if it was tagged by the user as a farewell party picture only).

In order to develop your application you will have to use the Apache Jena Java framework together with the Java EE platform.

Rendu du projet

Ce projet devra être réalisé en trinôme. Chaque groupe devra envoyer par courrier électronique à manuel.atencia-arcas@univ-grenoble-alpes.fr et jerome.david@univ-grenoble-alpes.fr dès que possible les informations suivantes : nom, prénom et adresse électronique du chacun des membres du groupe.

Vous devrez nous rendre un dossier contenant

- le code de votre application,
- une guide de déploiement de l'application,
- un fichier turtle avec un échantillon des données RDF utilisées par l'application,
- le fichier de votre ontologie en format turtle,
- une liste avec des exemples de requêtes SPARQL et les résultats d'exécuter ces requêtes vers l'échantillon des données et votre ontologie.

Vous devrez faire une présentation de votre projet. Cette présentation se déroulera en salle machine avec une démonstration de votre application. Il est possible, d'y associer une présentation de type power point si vous le jugez utile.

La date de rendu du projet est le 17 janvier 2019 (23h59) et la date des soutenances le 21 janvier 2019. L'ordre des soutenances vous sera communiqué prochainement.

Evaluation

The evaluation of the project will be based on the following criteria:

- The modelization of your data: choice of URIs, reuse of existing vocabularies (like FOAF and Dublin Core), links to external datasets.
- The expressivity of your ontology: use of different OWL constructors, class expressions, domain and range property statements, property characteristics (transitivity, symmetry…)…
- The consistency of your ontology.
- The use of OWL reasoning to complete the results of SPARQL queries.
- The complexity of your SPARQL queries: filters, optionals, union, negation, federated queries…
- The functionalities of your application.
- The quality of your code.

Indicative scale :

- RDF modelization (2pts)
- RDF-S/OWL ontology (6pts)
- SPARQL queries (6pts)
- Jena code (6pts)

Remark : You may conceive SPARQL queries and ontology axioms that may not be used by your application.