

Practical Session 4 – RDF Schema

Today we are going to look more closely at RDF Schema, check how it is used in some of the RDF datasets we have seen, and experience the use of inference through blazegraph.

T1: Using the SPARQL endpoint of DBpedia (<https://dbpedia.org/sparql>) write a query to get all the classes that are the types of http://dbpedia.org/resource/University_of_Lorraine.

T2: Now write a query to get all the classes of which those classes are subclasses.

T3: Adapt the previous query to get the same super classes of classes of University of Lorraine, but without the ones that were already listed as direct classes (T1). From the result, do you think it is possible that inference has been enabled on the DBpedia SPARQL endpoint?

T4: Get, in one query, all the domains and ranges defined for all the properties used with http://dbpedia.org/resource/University_of_Lorraine as subject. Note that it is possible for a property to have a domain and no range, and inversely.

T5: Based on this, check if inference might be enabled (i.e. if rule `rdfs2` might have been applied).

The file at https://mdaquin.github.io/d/cheese_voc.ttl contains a simple vocabulary for the cheese dataset. Open it and check what it says. Does it intuitively make sense?

T6: Bring back the installation of Blazegraph you used last time. The cheese dataset should still be there. Re-add it if not. Upload the file https://mdaquin.github.io/d/cheese_voc.ttl into that same graph/namespace as the cheese dataset. Generalizing from the query used above, write a query that tries to find examples showing that the rule `rdfs2` was not applied.

T7: Create a new namespace in blazegraph with inference enabled. Upload both <https://mdaquin.github.io/d/cheese.ttl> and https://mdaquin.github.io/d/cheese_voc.ttl into it. Write queries to check that inferences have indeed been applied.

T8 (bonus): Write an insert query that corresponds to the application of both rules `rdfs2` and `rdfs3`.

Project

For the project, you are asked to build a knowledge graph of children stories. The idea is that your knowledge graph could be used by someone researching a particular theme, plot, type of character, or other aspects of children stories to find relevant ones, to compare stories with each other on those aspects, or to analyze trends in the way stories have evolved over time and cultures. As the lectures and practical sessions go, we will learn more about how that could be done and how we could use it.

At the end of all the practical sessions, you will have to submit:

1. The RDF code of the knowledge graph.
2. A short report briefly describing the steps you have gone through, the choices you have made, any SPARQL query you have used, and a description of any code you might have written.

So, keep notes of what you do and find!

The following task is to be started at the end of this practical session and completed before the next one.

You should have, previous times, found useful things that you could reuse, integrate and take inspiration from in the SPARQL endpoints listed below and potentially others.

Check the schema of those things, i.e. what classes (and subclasses) they rely on, how the properties are defined, etc. What can you reuse from those in the schema for your own knowledge graph? Can you update your SPARQL construct queries from last time so they also include the schema elements you want to reuse?

Draw a first, informal version of what your schema will look like as a diagram.

- BNF: <https://data.bnf.fr/sparql/>
- DBpedia: <https://dbpedia.org/sparql>
- Wikidata: <https://query.wikidata.org/>
- Europeana: <https://sparql.europeana.eu/>
- National Library of Finland: <https://data.nationallibrary.fi/bib/sparql>
- Linked Movie Database: <https://triplydb.com/Triply/linkedmdb/sparql/linkedmdb>