Lab 5 – 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).

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

T7: Write an insert query that corresponds to the application of both rules rdfs2 and rdfs3.

T8: Create a new namespace in blazegraph with inference enabled. Upload both the `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, checking each entailment rule seen in the lecture separately.