Data Integration Using OWL and Rules

(Petr Křemen,Miroslav Blaško) & Michal Med

December 10, 2020

Ex. 1 — Download the ZIP archive from the web site and unpack data1.ttl, data2.ttl and ontology.ttl.

Ex. 2 — Take a look at each of the files in Protege.

Ex. 3 — Create a new OWL ontology in Protege, import all three ontologies in it and save it along with the other as integration.ttl.

Ex. 4 — Align the classes and Object Properties of data1.ttl and data2.ttl with ontology.ttl. For example, you might want to say that `d1:parent` is a subclass of (or equivalent class of) `o:parent`, or that `d1:is-child-of` is a subproperty of `inverse(o:has-child)`. Try to be as precise as possible.

Ex. 5 — Define characteristics (transitivity, functionality, etc.) of the object properties.

Ex. 6 — Define a SWRL rule that infers `o:has-mother` property assertions using the `o:woman` class and `o:has-parent` property.

Ex. 7 — Define a SWRL rule that infers `o:has-step-mother` property assertions out of the existing data (e.g. `d1:wenceslas-iv` has three step mothers (we neglect that they need not have lived during Wenceslas' life).

Ex. 8 — Define the class `mother-with-at-least-two-children`. Which instances belong to it?

Ex. 9 — Open the SPARQL Query Tab (You will need to have SNAP Plugin installed) and construct a query that retrieves all pairs of step siblings (with the same logic as the SWRL rule in ontology.ttl).

Ex. 10 — Take the resulting artifact and upload it into GraphDB. Compare the inferences to those in Protege.

1 References

1. https://www.w3.org/Submission/SWRL