

Practical Session 2

Objective:

1. Define an RDF graph
2. Define an OWL ontology in Protege
3. Infer facts using a reasoner
4. Define SWRL rules

Exercise 1:

Define an RDF graph based on the following description of the QBOAirbase collection defined for representing and storing air pollution data in European countries, the text is an extracted from reference¹

"In QBOAirbase an observation maps to a measurement ... that is, the aggregation of a set of measurements for a single air pollutant in an annual time span. QBOAirbase includes measurements for a list of 15 pollutants.... This is the minimal list of pollutants that a country must measure according to EU regulations... An observation is characterized by its coordinates in the year, station, and sensor dimensions... The station dimension contains three levels: station, city, and country. ...The sensor dimension...consists of two levels: sensor and component."

Exercise 2:

1. Download the LUBM ontology: <http://swat.cse.lehigh.edu/onto/univ-bench.owl>
2. Open the LUBM ontology in Protégé editor.
3. Identify the non canonical class definitions (class expressions)
4. Analyze the definition of Student class
5. Add the following individuals: "Student1" (of type *Person*) is following the course "Semantic_Technologies" (of type *teaching course*)
6. Start Hermit reasoner and list the instances of the class Student, then stop the reasoner.
7. Complete the ontology model to:
 - Link a "unofficial publication" to a resulting "article" using the property "resulted_in". Define individuals "Article_0" and "Article_1" to test this link.
 - Make the classes School, Institute and University disjoint (add the disjointness relationship in Institute class only)
 - Add an individual "Institute1" to the Institute class
 - In the tab "individuals" (under "Entities" tab), add the classes University and School as types for the individual "Institute1"
 - Run the reasoner
 - Correct the identified inconsistencies.

¹ Galárraga, Luis, Kim Ahlstrøm, and Katja Hose. "QBOAirbase: the European air quality database as an RDF cube." *The 16th International Semantic Web Conference*. CEUR Workshop Proceedings, 2017.

- Add an object property "corresponding_author" that must be unique (Functional)
- Add an individual "Researcher1" of type "university research assistant".
- Define these two links : "Article1" having two corresponding authors "Student1" and "Researcher1" (using the link "corresponding_author") and then run the reasoner
- Click on "Student1" then "Researcher1" individual (under "Individuals by class" tab: Description) and identify their types, what can you conclude?
- Make the corresponding corrections

- What is the inverse property of the relationship "has a degree from" ?
- Create an individual "University_USTHB" of type University.
- Link "Student1" to "University_USTHB" using the relationship "has a degree from".
- Run the reasoner and identify the inferred property assertions in "University_USTHB".

- Create an object property "has_editor" and the class "Scientific_Editor".
- Add a class "Proceeding" as a subclass of the class "book"
- Define the class "Proceeding" as equivalent to a book that have some scientific editors. Use the class expression: *book and (has_editor some scientific_editor)*.
- Test this equivalence by creating "Proceeding1" of type Book having Springer as a scientific editor.
- Define the new subsumption relationship identified.

- Add the following definition to the class "article": "An article is a research report produced by a team of researchers". Use the annotation "rdfs:isDefinedBy"

Exercise 3:

Add the following classes and properties:

- Class "University_Researcher" as a subclass of the class Employee.
- Class "Developer" as a subclass of Person
- Class "Master_Course" as a subclass of "teaching course"
- The Object property "is_author", "is_teaching"

Display the menu SWRLTab by clicking on Windows → Tabs → SWRL Tab

Add the following rules:

- A Faculty member authoring a Publication is considered as a University_Researcher
SWRL rule: *Faculty(?x) ^ is_author(?x, ?y) ^ Article(?y) -> University_Researcher(?x)*
- A student authoring a Publication is considered as a 'university research assistant'
- A person authoring a 'software program' is a Developer
- A person who teaches a Master_Course is a Researcher