RDF(S)

Petr Křemen

October 11, 2018

1 Background

This seminar will be devoted to the RDF(S) model. Please refer to lecture 2 for details on RDF(S).

To ensure data you put into your repository will be dereferenceable, everyone will use IRIs of the form `http://onto.fel.cvut.cz/ontologies/REPOSITORYNAME/WHATEVERYOUWANT`, where

`REPOSITORYNAME` is the name of the repository in GraphDB.

`WHATEVERYOUWANT` is a local identifier, according to your local identification scheme.

For example, in a repository named `testrepo`, we would like to create a new reference to a person John Doe. We decide to represent the IRI as `http://onto.fel.cvut.cz/ontologies/testrepo/person/doe-john` and the class person as `http://onto.fel.cvut.cz/ontologies/testrepo/person`. Note, that this is not the only option and it is a matter of design decision, how a IRI is constructed, e.g.

generic identification scheme creates unified IRIs for all individuals, another for all IRIs, e.g. `http://onto.fel.cvut.cz/ontologies/testrepo/object-1`,

class-prefixed identification scheme creates unified IRIs for all individuals of a particular class, e.g. `http://onto.fel.cvut.cz/ontologies/testrepo/person-1`,

class-related identification scheme creates unified IRIs for all individuals of a particular class, e.g. `http://onto.fel.cvut.cz/ontologies/testrepo/person/1`.

2 Excercises

Ex. 1 — Open a Turtle editor at `http://onto.fel.cvut.cz/turtle-editor` and explore the default turtle document. Take a look at its graphical view as well. Delete german labels from all resources, producing a valid turtle document.

Answer (Ex. 1) — Remove the triples with `@de` language tag (in turtle shorthand syntax).

Ex. 2 — Consider the RDF graph G in Figure 1.

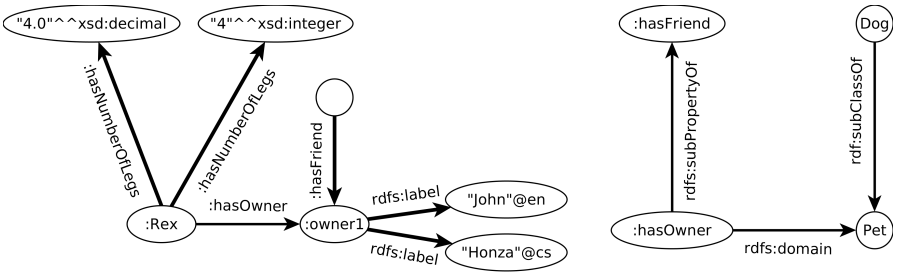

Figure 1: An example RDF graph

1. decide, whether G is ground,
2. decide, whether G is lean, if no, simplify it, so that it becomes lean.
3. rewrite the graph into the Turtle syntax
4. which triples are entailed by G under simple entailment,
5. which triples are entailed by G under RDF-{xsd:decimal} entailment,
6. which triples are entailed by G under RDFS-{xsd:decimal,xsd:integer} entailment
7. write a statement describing that the information about number of legs of Rex was provided by a person with IRI `:Tom`.

Answer (Ex. 2) — The answers follow:

1. no (there is a blank node)
2. yes (none of its instances is its proper subgraph)
3.
```
@prefix : <http://onto.fel.cvut.cz/ontologies/2017-osw/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
:Rex  :hasNumberOfLegs 4.0, 4 ;
      :hasOwner :owner1 .
[] :hasFriend :owner1 .
:owner1 rdfs:label "John"@en,"Honza"@cs .
:hasOwner   rdfs:subPropertyOf :hasFriend ;
            rdfs:domain :Pet .
:Dog rdfs:subClassOf :Pet .
```
4. many statements that are generalizations of the RDF graph subgraphs, e.g.
```
[]  :hasNumberOfLegs [] .
```
5. additionally to the previous ones also e.g.

```
:hasNumberOfLegs a rdf:Property.
```

6. additionally to the previous ones also e.g.

```
:Rex :hasFriend :owner1.
:Rex a :Pet.
```

7.
```
[ rdf:subject :Rex ;
  rdf:predicate :hasNumberOfLegs ;
  rdf:object 4 ] dc:creator :Tom .
```

Ex. 3 — Create an RDF document in Turtle syntax, representing the following knowledge. Define your own IRIs for named resources:

- John is a husband of Mary.
- Mary and George have the same mother (who is unknown).
- George is 180 cm tall.

Answer (Ex. 3) — The following graph is an example. Note, that the representation of complex data values (values+units) does not use any shared vocabulary and thus is not much reusable.

```
@prefix : <http://onto.fel.cvut.cz/ontologies/2017-osw/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
:mary :has-husband :john .
[] :is-mother-of :mary, :george .
:george :has-height [
   :value 180 ;
   :unit :centimeter
] .
```

Ex. 4 — Create a schema document to the previous example, formalizing the knowledge about people – namely classes `Person`, `Man`, `Woman`, and properties `date − of − birth`, `has − husband`, `is − relative − of`, `has − mother`, `has − father`. Try to express as much knowledge about these classes/properties, as possible, using RDF Schema 1.1 constructs.

Answer (Ex. 4) —

```
@prefix : <http://onto.fel.cvut.cz/ontologies/2017-osw/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
:john a :Person .
:mary a :Person .
:george a :Person .
```

```
:Person a rdfs:Class .
:Man a rdfs:Class .
:Woman a rdfs:Class .
:Man rdfs:subClassOf :Person .
:Woman rdfs:subClassOf :Person .
:date-of-birth rdfs:domain :Person ;
               rdfs:range xsd:date .
:is-relative-of rdfs:domain :person ;
                rdfs:range :person.
:has-husband rdfs:domain :woman ;
             rdfs:range :man ;
             rdfs:subPropertyOf :is-relative-of .
:has-father rdfs:domain :person ;
            rdfs:range :man ;
            rdfs:subPropertyOf :is-relative-of .
:has-mother rdfs:domain :person ;
            rdfs:range :woman ;
            rdfs:subPropertyOf :is-relative-of .
```

Ex. 5 — Using a text editor, create an RDF document (in Turtle) with your public RDF profile (i.e basic data, your interests, etc.). Use FOAF vocabulary (`http://xmlns.com/foaf/spec/`), where possible.

Answer (Ex. 5) — See e.g.
`http://onto.fel.cvut.cz/ontologies/kbss/people/petr-kremen`

3 Relevant References

- RDF Validator – `http://www.w3.org/RDF/Validator/`
- Any23 (transformation between RDF formats) – `http://any23.org/`