Foundations of Semantic Web Technologies

Tutorial 7

Sebastian Rudolph

SS 2014

Exercise 7.1. Assume the default graph to contain the triples from Table 1. Give the answer for the following query under the RDFS Entailment Regime and explain for every solution why it belongs to the answer:

```
PREFIX  rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX  ex: <http://example.org/>
SELECT  ?aut ?pub
WHERE   { ?pub rdf:type ex:Publication .
          ?pub ex:authors ?seq .
          ?seq ?ind ?aut }
```

Exercise 7.2. Probably you have noticed that the answer in Exercise 7.1 had two solutions where the binding for `?aut` was not an author's name. How can the query be changed to prevent that?

Exercise 7.3. Again, assume the default graph to consist of the triples in Table 1. Consider the following query:

```
PREFIX  rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX  w3c:  <http://www.w3.org/TR/>
SELECT  ?type
WHERE   { w3c:sparql11-entailment rdf:type ?type }
```

What solutions does the query have under the RDFS Semantics and why?

Exercise 7.4. Assuming the default graph consisting of the triples from Table 1, is the triple `w3c:sparql11-entailment ex:authors _:x` a consequence of the graph under RDFS semantics? Consequently, what is the answer to the following query?

```
ASK  { w3c:sparql11-entailment ex:authors _:x }
```

Tabelle 1: RDF Graph for Exercise 7.1 to 7.4

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>.
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>.
@prefix ex: <http://example.org/>.
@prefix w3c: <http://www.w3.org/TR/>.
```

(1)	w3c:sparql11-entailment	rdf:type	ex:WorkingDraft .
(2)	w3c:sparql11-entailment	ex:authors	_:l1.
(3)	_:l1	rdf:type	rdf:Seq.
(4)	_:l1	rdf:_1	"Birte Glimm".
(5)	_:l1	rdf:_2	"Chimezie Ogbuji".
(6)	w3c:rdf-sparql-query	rdf:type	ex:Recommendation .
(7)	w3c:rdf-sparql-query	ex:writtenBy	_:l2 .
(8)	_:l2	rdf:type	rdf:Seq.
(9)	_:l2	rdf:_1	"Eric Prud'hommeaux".
(10)	_:l2	rdf:_2	"Andy Seaborne".
(11)	ex:WorkingDraft	rdfs:subClassOf	ex:Publication .
(12)	ex:Recommendation	rdfs:subClassOf	ex:Publication .
(13)	ex:writtenBy	rdfs:subPropertyOf	ex:authors .

Exercise 7.5. [optional] A typical task for OWL reasoners is to compute the class hierarchy. Given an Ontology O, this means that all pairs (C, D) are computed where C and D are atomic classes and C is a direct subclass of D, that is, $O \models C \sqsubseteq D$ holds and there is no class E different from C and D such that $O \models C \sqsubseteq E$ and $O \models E \sqsubseteq D$ would hold. Can a SPARQL query be used to obtain a class hierarchy as a result?