

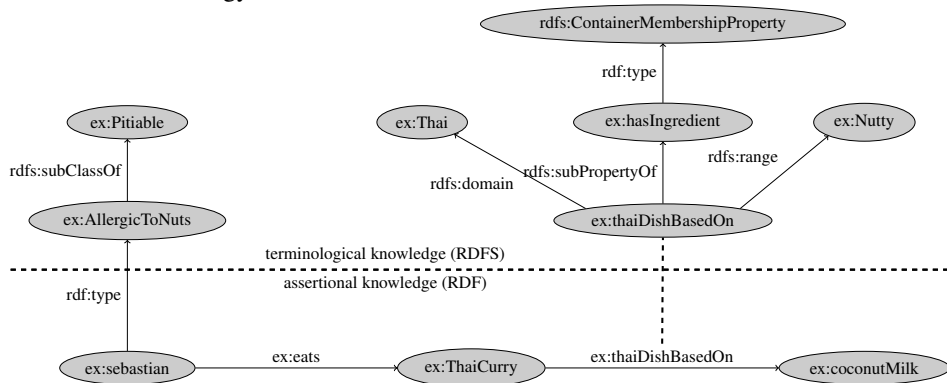
Foundations of Semantic Web Technologies

Tutorial 3

Dörthe Arndt

WS 2022/23

Exercise 3.1. Describe an RDFS interpretation that is a model of the example ontology from Exercise 1.4 from Tutorial 2. For reference, find here the RDFS graph representation of this ontology:



Exercise 3.2. For the ontology from Exercise 2.1, find

- a triple that is simply entailed,
- a triple that is RDF-entailed but not simply entailed,
- a triple that is RDFS-entailed but not RDF-entailed.

Exercise 3.3. The empty graph does not contain any triples (i.e. it corresponds to the empty set). Give derivations showing that the empty graph RDFS-entails the following triples:

- `rdfs:Resource rdfs:type rdfs:Class .`
- `rdfs:Class rdfs:type rdfs:Class .`
- `rdfs:Literal rdfs:type rdfs:Class .`
- `rdf:XMLLiteral rdfs:type rdfs:Class .`
- `rdfs:Datatype rdfs:type rdfs:Class .`

- (f) `rdf:Seq` `rdf:type` `rdfs:Class` .
- (g) `rdf:Bag` `rdf:type` `rdfs:Class` .
- (h) `rdf:Alt` `rdf:type` `rdfs:Class` .
- (i) `rdfs:Container` `rdf:type` `rdfs:Class` .
- (j) `rdf>List` `rdf:type` `rdfs:Class` .
- (k) `rdfs:ContainerMembershipProperty` `rdf:type` `rdfs:Class` .
- (l) `rdf:Property` `rdf:type` `rdfs:Class` .
- (m) `rdf:Statement` `rdf:type` `rdfs:Class` .
- (n) `rdfs:domain` `rdf:type` `rdf:Property` .
- (o) `rdfs:range` `rdf:type` `rdf:Property` .
- (p) `rdfs:subPropertyOf` `rdf:type` `rdf:Property` .
- (q) `rdfs:subClassOf` `rdf:type` `rdf:Property` .
- (r) `rdfs:member` `rdf:type` `rdf:Property` .
- (s) `rdfs:seeAlso` `rdf:type` `rdf:Property` .
- (t) `rdfs:isDefinedBy` `rdf:type` `rdf:Property` .
- (u) `rdfs:comment` `rdf:type` `rdf:Property` .
- (v) `rdfs:label` `rdf:type` `rdf:Property` .

Exercise 3.4. Note that the theorem linking the RDFS entailment with the presented deduction calculus just guarantees soundness of the latter. When the calculus was provided in the RDF semantics specification, it was also considered complete, but a bit later that turned out not to be the case.

As an example of the calculus' incompleteness, consider the following set of triples:

```

ex:isHappilyMarriedTo  rdfs:subPropertyOf  _:bnode .
_:bnode                rdfs:domain        ex:Person .
ex:markus              ex:isHappilyMarriedTo  ex:anja .

```

It is not hard to show that the triple

```

ex:markus  rdf:type  ex:Person .

```

is a semantic consequence of the above. However, it cannot be derived by means of the given deduction calculus. Make a suggestion how the calculus could be "repaired".

Exercise 3.5. Consider the following RDFS graph:

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

ex:Platypus ex:name "Ornitorrinco"@es-CL , "Platypus"@en ;
            ex:speciesName "Ornithorhynchus anatinus"@la ;
            a ex:NearThreatenedSpecies ; ex:genus ex:Ornithorhynchus .

ex:Knuckles a ex:TachyglossusAculeatus .

ex:TachyglossusAculeatus ex:name "Equidna"@es-CL , "Short-beaked Echidna"@en ;
                        ex:speciesName "Tachyglossus aculeatus"@la ;
                        a ex:OviparousSpecies ; ex:genus ex:Tachyglossus .

ex:ZaglossusBruijnii ex:name "Western long-beaked echidna"@en ;
                    a ex:CriticallyEndangeredSpecies ; ex:genus ex:Zaglossus .

ex:Ornithorhynchus ex:name "Ornithorhynchus"@la ; ex:family ex:Ornithorhynchidae .
ex:Tachyglossus ex:name "Tachyglossus"@la ; ex:family ex:Tachyglossidae .
ex:Zaglossus ex:name "Zaglossus"@la ; ex:family ex:Tachyglossidae .
ex:Ornithorhynchidae ex:name "Ornithorhynchidae"@la ; ex:order ex:Monotremata .
ex:Tachyglossidae ex:name "Tachyglossidae"@la ; ex:order ex:Monotremata .
```

Under the link <http://ppr.cs.dal.ca:3002/n3/editor/s/2U7uuLiQ> this graphs is preloaded together with some of the rules of the RDFS-calculus we had in class¹. You can execute the rules by clicking on the "execute"-button on the left side (with option eye).²

(a) Add one RDFS (sub-property) triple to infer:

```
ex:Platypus ex:name "Ornithorhynchus anatinus"@la .
ex:TachyglossusAculeatus ex:name "Tachyglossus aculeatus"@la.
```

(b) Add one RDFS (sub-class) triple to infer:

```
ex:ZaglossusBruijnii a ex:Species .
```

(c) Add one RDFS (domain) triple to infer:

```
ex:Platypus a ex:Species .
ex:TachyglossusAculeatus a ex:Species .
```

(d) Add three RDFS (range) triples to infer:

```
ex:Ornithorhynchus a ex:Genus .
ex:Tachyglossus a ex:Genus .
```

¹We only use a selected rule set to keep the derivations simple.

²If you did not add any triples the execution should not provide any results.

ex:Zaglossus a ex:Genus .
ex:Ornithorhynchidae a ex:Family .
ex:Tachyglossidae a ex:Family .
ex:Monotremata a ex:Order .

(e) Add three RDFS triples to infer:

ex:Platypus ex:parentTaxon ex:Ornithorhynchus .
ex:TachyglossusAculeatus ex:parentTaxon ex:Tachyglossus .
ex:ZaglossusBruijnii ex:parentTaxon ex:Zaglossus .
ex:Ornithorhynchus ex:parentTaxon ex:Ornithorhynchidae .
ex:Zaglossus ex:parentTaxon ex:Tachyglossidae .
ex:Tachyglossus ex:parentTaxon ex:Tachyglossidae .
ex:Ornithorhynchidae ex:parentTaxon ex:Monotremata .
ex:Tachyglossidae ex:parentTaxon ex:Monotremata .