



RDF Validation - RDF(s)/OWL/SHACL

8th tutorial

Ontologies and Semantic Web

Michal Med

michal.med@fel.cvut.cz

RDF(S) reasoning

- Create new repository in GraphDB with Ruleset OWL-Max
- Insert following snippet (or file john-mary.ttl found in CW):

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
```

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

```
@prefix : <https://onto.fel.cvut.cz/ontologies/medmicha/> .
```

```
:John :hasWife :Sue .
```

```
:John a :Man .
```

```
:hasWife rdfs:domain :MarriedMan ;
```

```
    rdfs:range :MarriedWoman ;
```

```
    rdfs:subPropertyOf :hasRelative .
```

RDF(S) reasoning

- Check if John is a :MarriedMan using ASK statement

```
ASK { :John a :MarriedMan }
```

YES

RDF(S) entailment

Check what new information about John and Sue were inferred. Which were got by RDFS entailment? Check https://cw.fel.cvut.cz/wiki/_media/courses/b4m36osw/lecture-02-rdf-s.pdf

	subject	predicate	object	context
1	:John	:hasRelative	:Sue	http://www.ontotext.com/implicit
2	:John	:hasWife	:Sue	http://www.ontotext.com/explicit
3	:John	rdf:type	:Man	http://www.ontotext.com/explicit
4	:John	rdf:type	:MarriedMan	http://www.ontotext.com/implicit

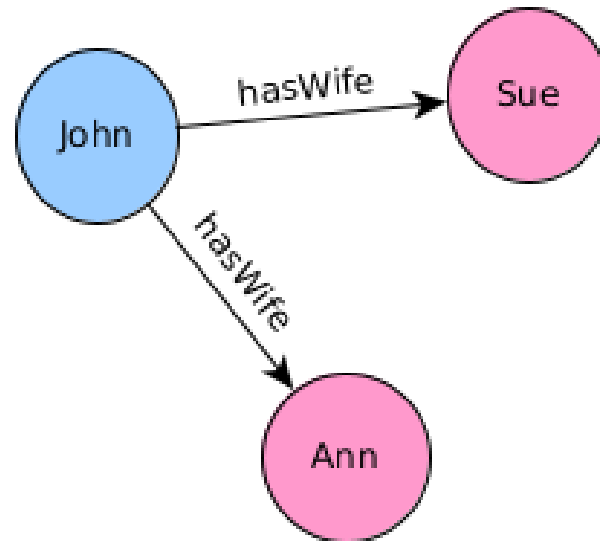
	subject	predicate	object	context
1	:Sue	rdf:type	:MarriedWoman	http://www.ontotext.com/implicit



How to make this in OWL?

Validation problem

- How to ensure that John has ONLY ONE wife?



Create validation rules in OWL

- Download osw8.owl and open it in protégé.
- Create new class "Camping in a tent", requiring usage of exactly one tent for overnight stay.
`using_a_roof exactly 1 tent`
- Create two instances of a tent and two instances of a "Camping in a tent". Make one instance using both tents and other instance no tent. Run reasoner.

```
using_a_roof tent_1
```

```
using_a_roof tent_2
```

Problems with validation through OWL

Screenshot of the Protege interface showing the class **táboření_pod_stanem** selected in the left pane. The right pane displays the class description and property assertions. The description is: **táboření_pod_stanem**. The property assertions section shows two assertions for the property **'probíhá s použitím přístřešku'**: one for **Husky_Boyard_4** and one for **Jurek_ALP_2.5**. The bottom status bar indicates "Git: master" and "Reasoner active Show Inferences".

Screenshot of the Protege interface showing the class **Husky_Boyard_4** selected in the left pane. The right pane displays the class description and property assertions. The description is: **Husky_Boyard_4**. The property assertions section shows one assertion for the property **Stan** for the individual **Jurek_ALP_2.5**. The bottom status bar indicates "Git: master".

What happened?

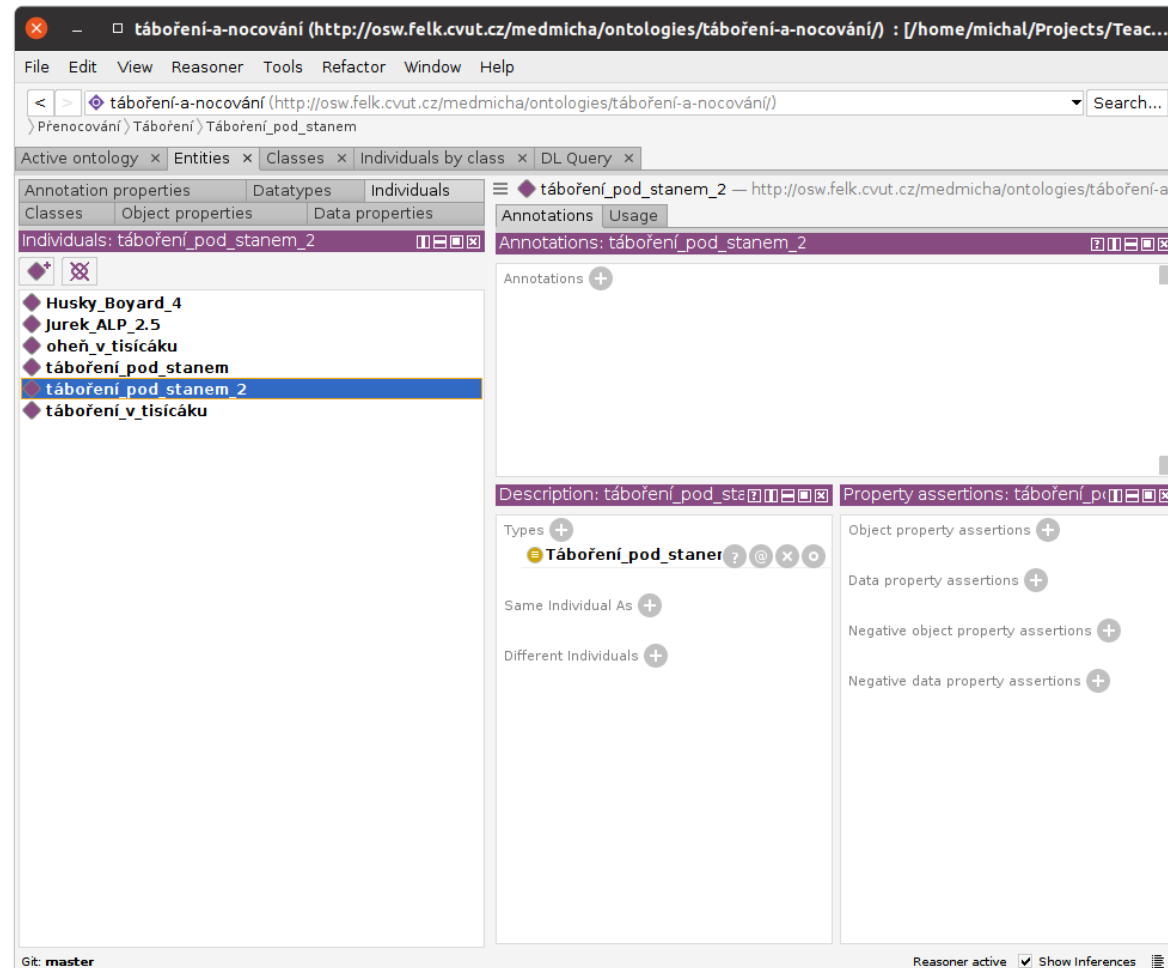
- Reasoner sees the rule that

'probíhá s použitím přístřešku' **exactly** 1 Stan

but sees two instances of "Tent". Based on the rule it expects the tents must be one tent using two various resources. Reasoner creates owl:sameAs relation between the tents.

Try to state that both tents are different individuals - there will be an error in reasoner.

Problems with validation through OWL



Why is there no error?

- Reasoner sees the rule that

'probíhá s použitím přístřešku' **exactly** 1 Stan

and uses OWA to expect, that given this rule, there must be exactly one tent used, otherwise it could not be "Camping in a tent", so tableau creates anonymous instance of Tent and assigns it to this particular camping.



How to check if data are valid?

Use SHACL

What is SHACL

- Shapes Constraint Language
- W3C recommendation
- <https://www.w3.org/TR/shacl/>
- <http://shacl.org/playground/>

Playground

- Try to validate this snippet:

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix : <http://onto.fel.cvut.cz/ontologies/medmicha/>.
```

```
:John :hasWife :Sue .
:John :hasWife :Ann .
:John a :Man .

:hasWife rdfs:domain :MarriedMan ;
         rdfs:range :MarriedWoman ;
         rdfs:subPropertyOf :hasRelative .
```

Why is it valid?

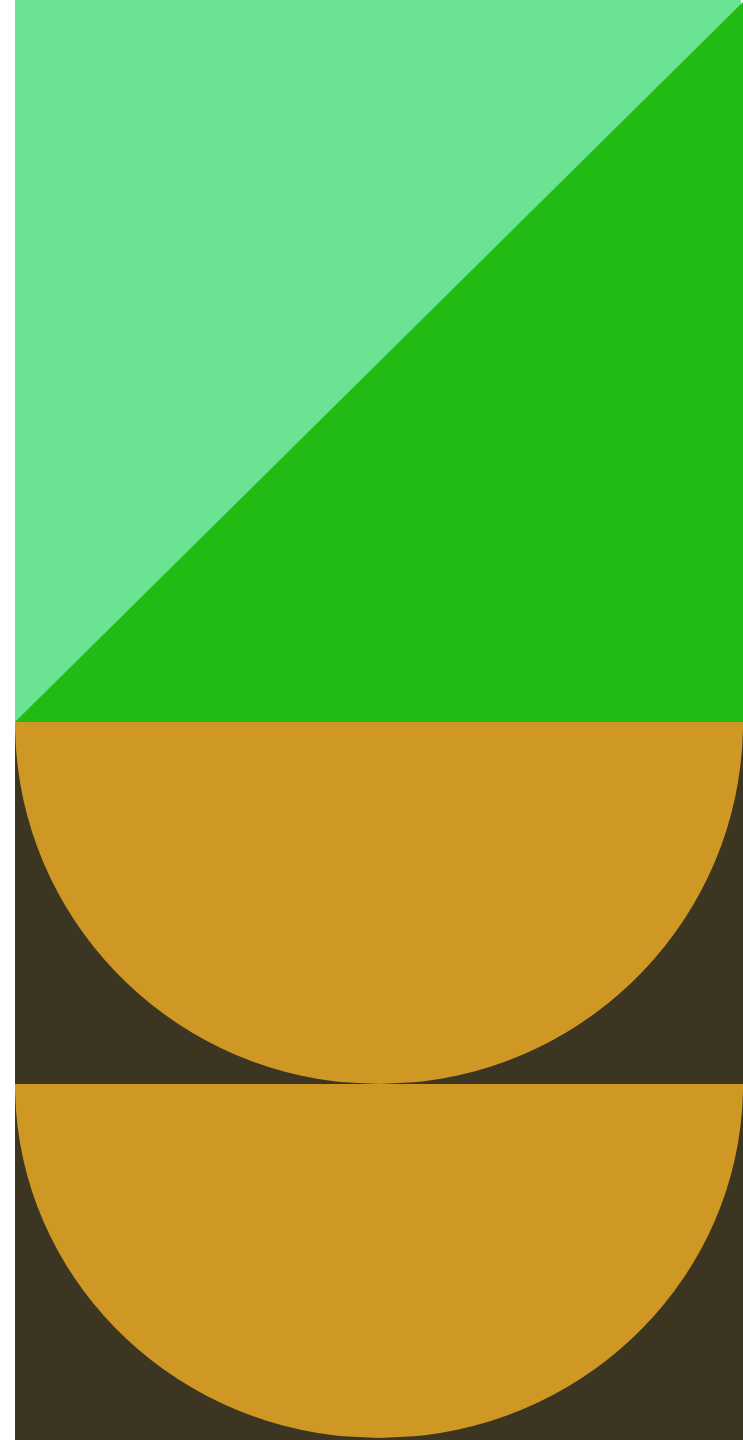
Playground

- Open <http://shacl.org/playground/> and insert following shape:

```
:MarriedManShape a sh:NodeShape ;  
  sh:targetClass :MarriedMan ;  
  sh:property [  
    sh:path :hasWife ;  
    sh:minCount 1 ;  
    sh:maxCount 1 ;  
  ] .
```

SHACL in protégé

- Install SHACL plugin
- Restart protégé
- Window -> Tabs -> SHACL editor



SHACL in protégé

- Write a SHACL shape rule to check that every "Camping in a tent" has exactly one used "Tent".
- Check what happens for the two instances of "Camping in a tent" created before.

Data validation in GraphDB

- GraphDB allows validation of data using SHACL under following conditions:

Create a repository with *Supports SHACL validation* option on

Import shapes as data into the repository

SHACL rules must be imported into named graph `http://rdf4j.org/schema/rdf4j#SHACLShapeGraph`

Other imported data re tested during import and ARE NOT imported if not valid.

Details and list of supported SHACL features:

<https://graphdb.ontotext.com/documentation/9.3/standard/shacl-validation.html>

SHACL in protégé

```
@prefix sh:    <http://www.w3.org/ns/shacl#> .
```

```
@prefix ex:    <http://www.example.org/> .
```

```
@prefix :      <http://osw.felk.cvut.cz/medmicha/ontologies/táboření-a-nocování/> .
```

```
ex:CampingShape
```

```
  a sh:NodeShape ;
```

```
  sh:targetClass :Stanování ;
```

```
  sh:property [
```

```
    sh:path :probíhá_s_použitím_přístřešku ;
```

```
    sh:maxCount 1 ;
```

```
    sh:minCount 1 ;
```

```
  ] .
```