

RDF Validation - RDF(s)/OWL/SHACL

8th tutorial

Ontologies and Semantic Web

Michal Med

michal.med@fel.cvut.cz

RDF(S) reasoning

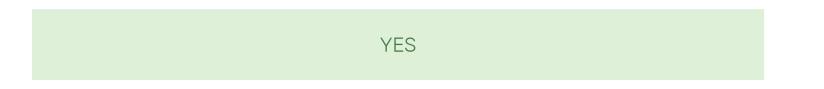
- Create new repository in GraphDB with Ruleset OWL-Max
- Insert following snippet:

@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/__username__/>.
: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}



RDF(S) entailment

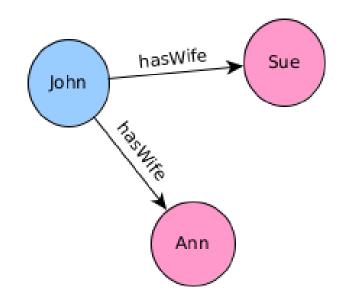
Check what new information about John and Sue were inferred. Which were got by RDFS entailment? Check <u>https://cw.fel.cvut.cz/wiki/_media/courses/b4m36osw/lecture-02-rdf-s.pdf</u>

	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.c		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	

Validation problem

• How to ensure that John has ONLY ONE wife?





How to make this in OWL?

Create validation rules in OWL

- Download osw8.owl and open it in protége.
- Create new class "Camping in a tent", requiring usage of exactly one tent for overnight stay.
- 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.

Problems with validation through OWL

lp	
cha/ontologies/táboření-a-nocování/)	▼ Search
s × DL Query ×	
táboření_pod_stanem — http://osw.felk Annotations Usage	.cvut.cz/medmicha/ontologies/táboření-a-no
Annotations: táboření_pod_stanem	
Annotations 🕂	
Description: tábořen(_pod_state []] = 💌 Types 🕂 Táboření_pod_staner ? @ X O Same Individual As 🕂 Different Individuals 🕂	Property assertions: táboření_p([] =] Object property assertions + "'probíhá s použitím ? @ × O přístřešku' Husky Boyard_4 "'probíhá s použitím ? @ × O přístřešku' jurek_ALP_2.5 Data property assertions + Negative object property assertions +
	Negative data property assertions
	 táboření_pod_stanem http://osw.felk Annotations Usage Annotations: táboření_pod_stanem Annotations + Description: táboření_pod_state[]] = IS Types + Táboření_pod_staner ? (a) × (o) Same Individual As +

táboření-a-nocování (http://osw.felk.cvut.cz/med	dmicha/ontologies/táboření-a-nocování/)	▼ Search
>Prenocování > Táboření > Táboření pod_stanem		
Active ontology × Entities × Classes × Individuals by a	lass × DL Query ×	
Annotation properties Datatypes Individuals	📄 🚍 🔶 Husky_Boyard_4 — Husky_Boyar	rd:4 — http://osw.felk.cvut.cz/medmicha/ontolo
Classes Object properties Data properties	Annotations Usage	
Individuals: Husky_Boyard_4	Annotations: Husky_Boyard_4	208
★* X	Annotations 🕂	
 Husky_Boyard_4 Jurek_ALP_2.5 		
• oheň_v_tisícáku		
táboření_pod_stanem		
 táboření_pod_stanem_2 táboření_v tisícáku 		
	Description: Husky_Boyard_4	Property assertions: Husky_Boy 🔳 🗖
	Types 🕂	Object property assertions (+)
	• Stan ? @ X	
		Data property assertions 🕂
	Same Individual As +	
	Jurek_ALP_2.5	Negative object property assertions
	Different Individuals +	Negative data property assertio

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

File Edit View Reasoner Tools Refactor Windo	•	
A state of the second	medmicha/ontologies/táboření-a-nocování/)	▼ Search
}Přenocování } Táboření } Táboření_pod_stanem		
Active ontology × Entities × Classes × Individuals		
Annotation properties Datatypes Individuals		elk.cvut.cz/medmicha/ontologies/táboření-a
Classes Object properties Data properties ndividuals: táboření pod stanem 2 III -	Annotations Usage	
	■☑ Annotations: táboření_pod_stanem_2	
●* <u>×</u>	Annotations 🛨	
Husky_Boyard_4 Jurek ALP 2.5		
oheň_v_tisícáku		
táboření_pod_stanem	_	
táboření_pod_stanem_2 táboření v tisícáku		
	Description: táboření_pod_sta 🛛 🗖 🗖 🗖	Property assertions; táboření p(m=n
	Types (+)	Object property assertions +
	Táboření pod staner ? @ x o	object property assertions
		Data property assertions 🕂
	Same Individual As 😛	
		Negative object property assertions 🕂
	Different Individuals 🛨	
		Negative data property assertions 🛨

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 at exactly one tent used, otherwise it could not be "Camping in a tent", so tableau creates anonymous instance of Tent and assignes it to this particular camping.



How to check if data are valid?

Use SHACL

What is SHACL

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

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 ;

].

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 ;

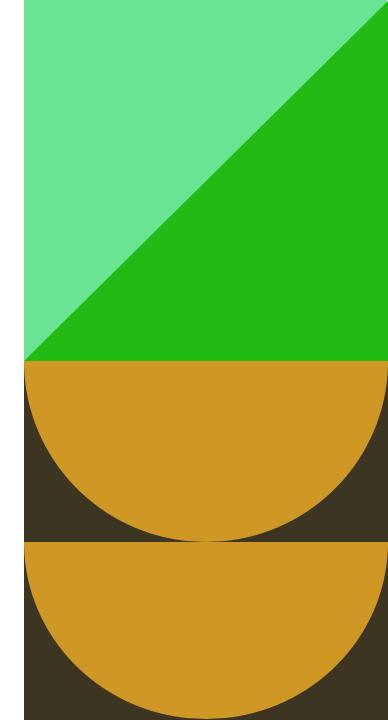
rdfs:range :MarriedWoman ;

rdfs:subPropertyOf :hasRelative .

Why is it valid?

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.

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 :Táboření_pod_stanem ;
sh:property [
sh:path :probíhá_s_použitím_přístřešku ;
```

sh:maxCount 1 ;

sh:minCount 1 ;

].



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