

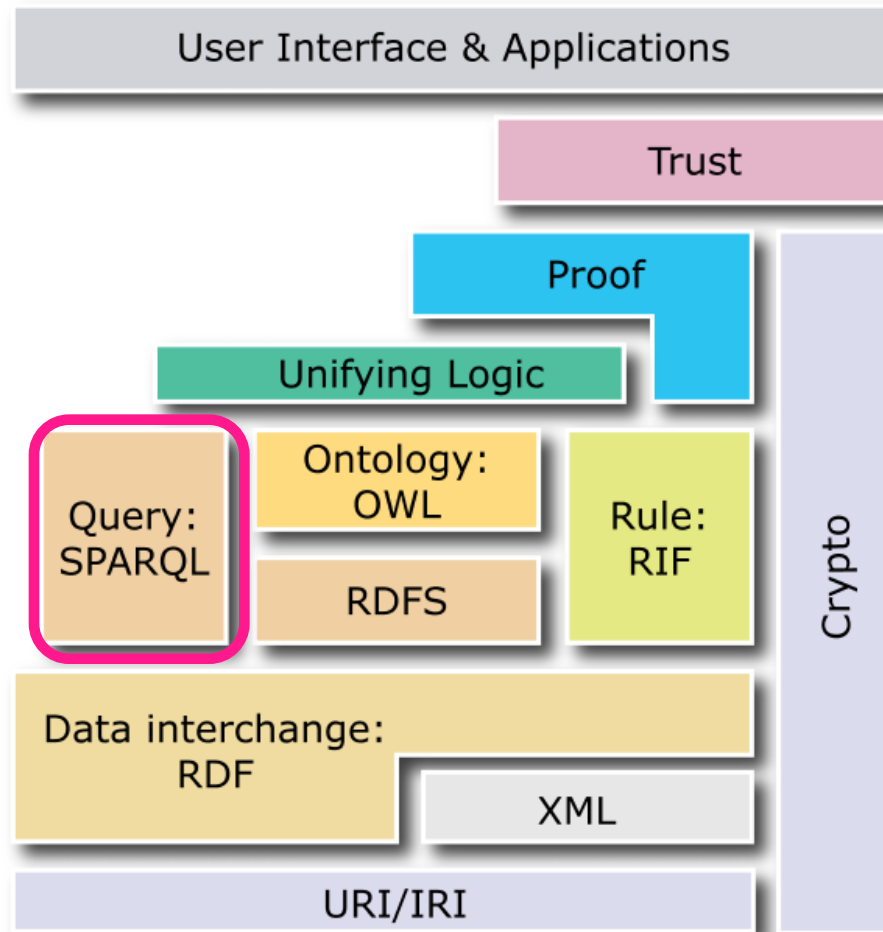
SPARQL

NIKOLA MILIKIĆ

EMAIL: nikola.milikic@fon.bg.ac.rs

URL: <http://nikola.milikic.info>

Semantic Web Layer Cake



Šta je SPARQL?

- SPARQL - SPARQL Protocol and RDF Query Language
- Omogućava izvršenje upita nad RDF bazom podataka (ili triple store)
- SPARQL $\approx$ SQL

Šta je SPARQL?

Omogućava:

- Izvlačenje vrednosti iz strukturiranih i polu-strukturiranih podataka
- Istraživanje strukture podataka
- Spajanja podataka koji dolaze iz različitih baza, a uz pomoć jednog, prostog upita
- Transformaciju RDF podataka iz jednog vokabulara u drugi

SPARQL u akciji

SPARQL upit definiše obrazac (šablon) rezultujućih tripleta koji se zove **graf patern**

Varijable graf paterna

SPARQL varijable počinju sa **?** ili **\$** i mogu predstavljati bilo koji deo tripleta.

Graf paterni su kao RDF tripleti s tim što neki delovi tripleta mogu biti zamenjeni varijablom.

RDF triplet:

ex:John	schema:name	“John” .
---------	-------------	----------

Graph pattern:

?person	schema:name	“John” .
---------	-------------	----------

Vrste SPARQL upita

SELECT

ASK

CONSTRUCT

DESCRIBE

Struktura upita

definisanje prefiksa

PREFIX foo: <http://example.com/resources/>

...

ako se izvršava nad eksternim izvorima

FROM ...

vrednosti koje će biti rezultat upita

SELECT ...

graf patern

WHERE {

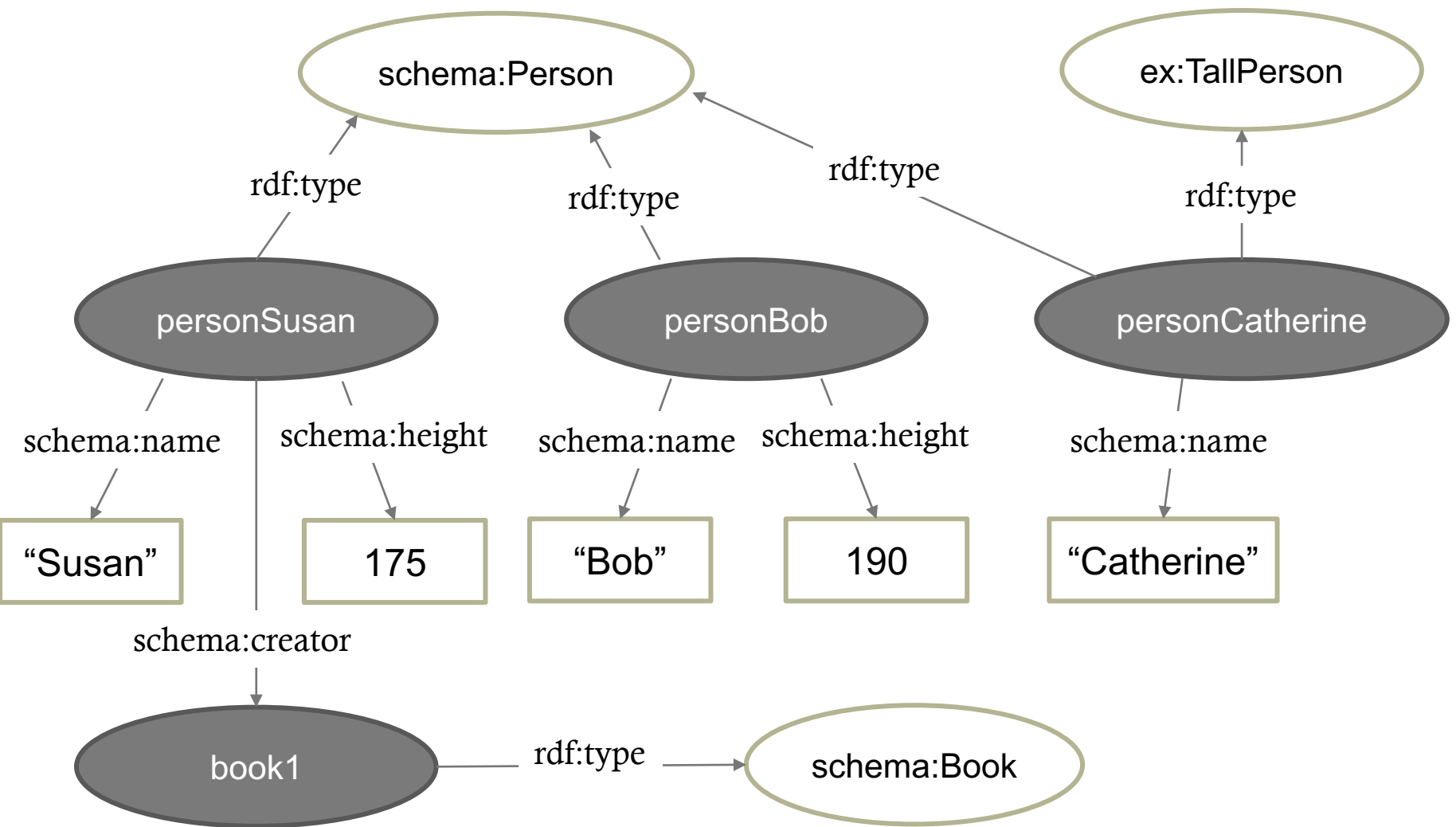
...

}

modifikatori upita

ORDER BY ...

Primer – Graf 1



Primer – Graf 1 – tekstualni oblik

personSusan	rdf:type	schema:Person .
personSusan	schema:name	“Susan” .
personSusan	schema:height	175 .
personSusan	schema:autor	book1 .
personBob	rdf:type	schema:Person .
personBob	schema:name	“Bob” .
personBob	schema:height	190.
personCatherine	rdf:type	schema:Person .
personCatherine	rdf:type	ex:TallPerson .
personCatherine	schema:name	“Catherine” .
book1	rdf:type	schema:Book .

Pretpostavke

U primerima ćemo koristiti sledeće prefikse:

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

PREFIX schema: <http://schema.org/>

PREFIX ex: <http://example.com/schema#>

Q1: Vрати sve osobe

```
SELECT ?person
```

```
WHERE {
```

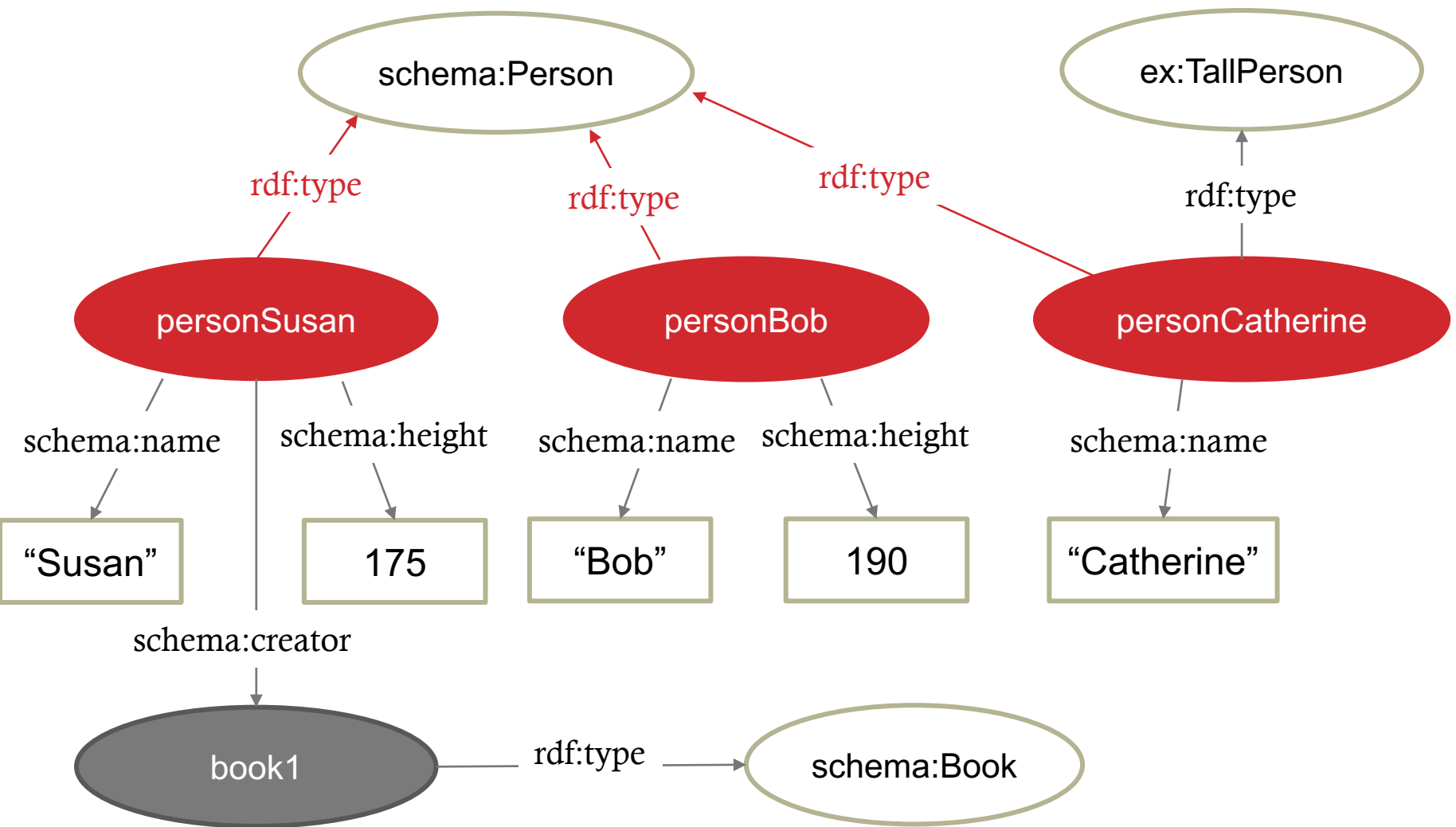
```
    ?person
```

```
    rdf:type
```

```
    schema:Person .
```

```
}
```

Q1 – Graf



Rezultat upita Q1 nad podacima Graf 1

person
personSusan
personBob
personCatherine

Q2: osobe i njihova imena

```
SELECT ?person ?name
```

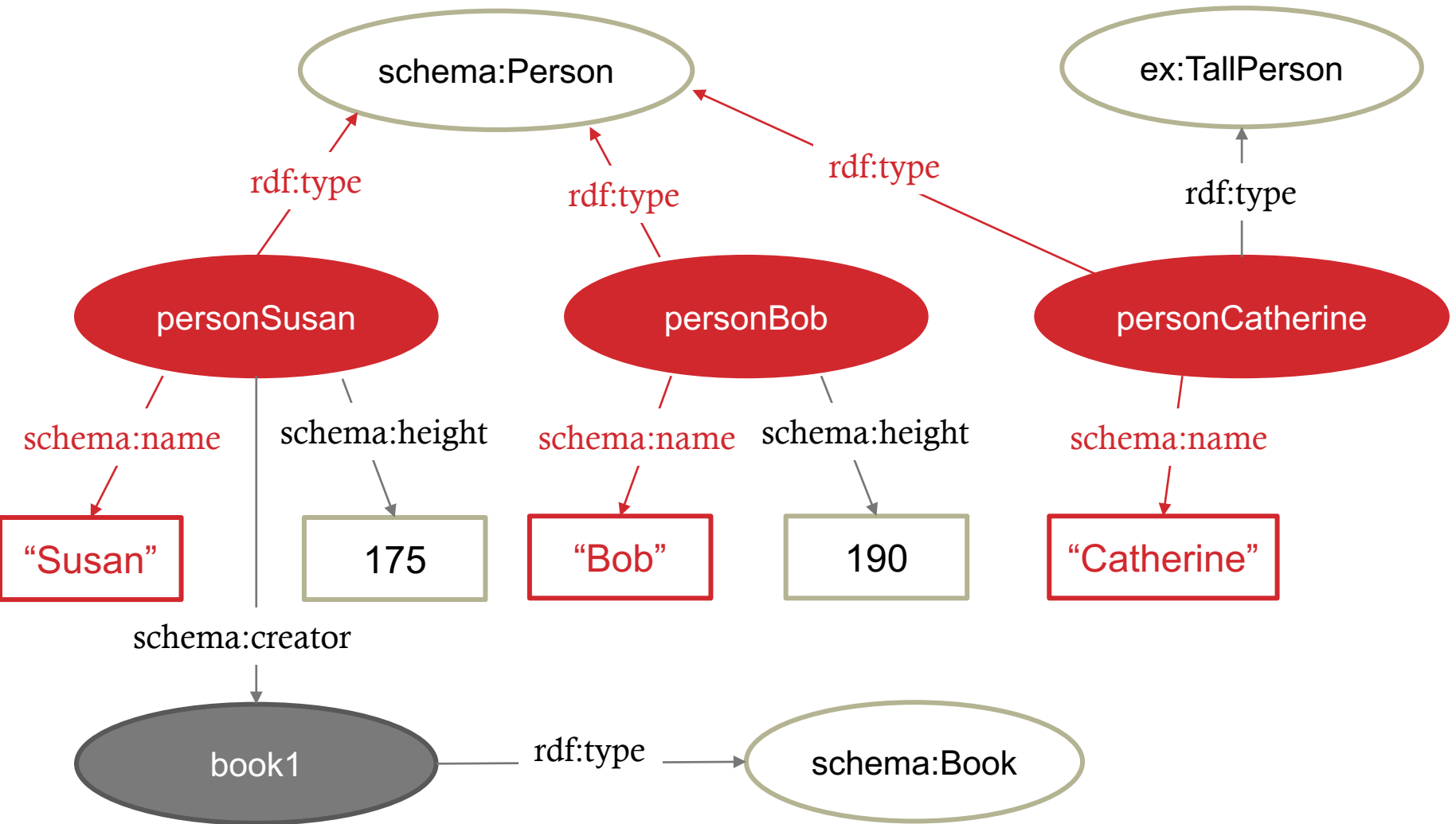
```
WHERE {
```

```
    ?person      rdf:type      schema:Person .
```

```
    ?person      schema:name    ?name .
```

```
}
```

Q2 – Graf



Rezultat upita Q1 nad podacima Graf 1

person	name
personSusan	"Susan"
personBob	"Bob"
personCatherine	"Catherine"

FILTER

- Definisanje ograničenja vrednosti u graf paternu (npr. numerička ograničenja $X > 180$)
- Mogu se koristiti boolean operatori, kao i regularni izrazi (regex)

Q2: Osobe više od 180 cm

```
SELECT ?person ?name
```

```
WHERE {
```

```
    ?person      rdf:type          schema:Person .
```

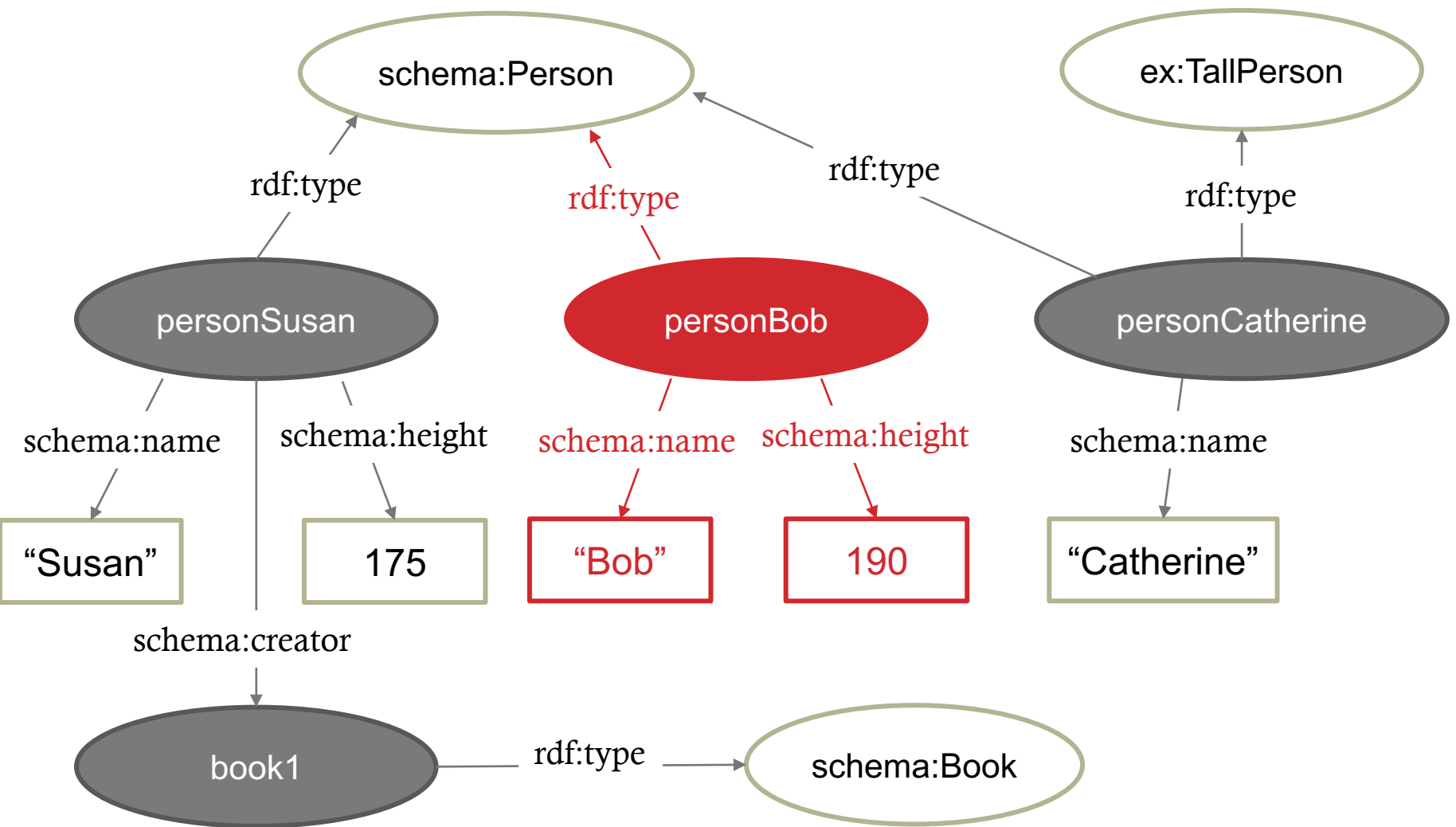
```
    ?person      schema:name       ?name .
```

```
    ?person      schema:height     ?height .
```

```
    FILTER ( ?height > 180 )
```

```
}
```

Primer – Graf 1



Rezultat upita Q2 nad podacima Graf 1

person	name
personBob	"Bob"

OPTIONAL

- Deo graf paterna se može definisati kao opcioni koristeći ključnu reč OPTIONAL.

Q3: Vratiti visinu ako postoji podatak o njoj

```
SELECT ?person ?name ?height
```

```
WHERE {
```

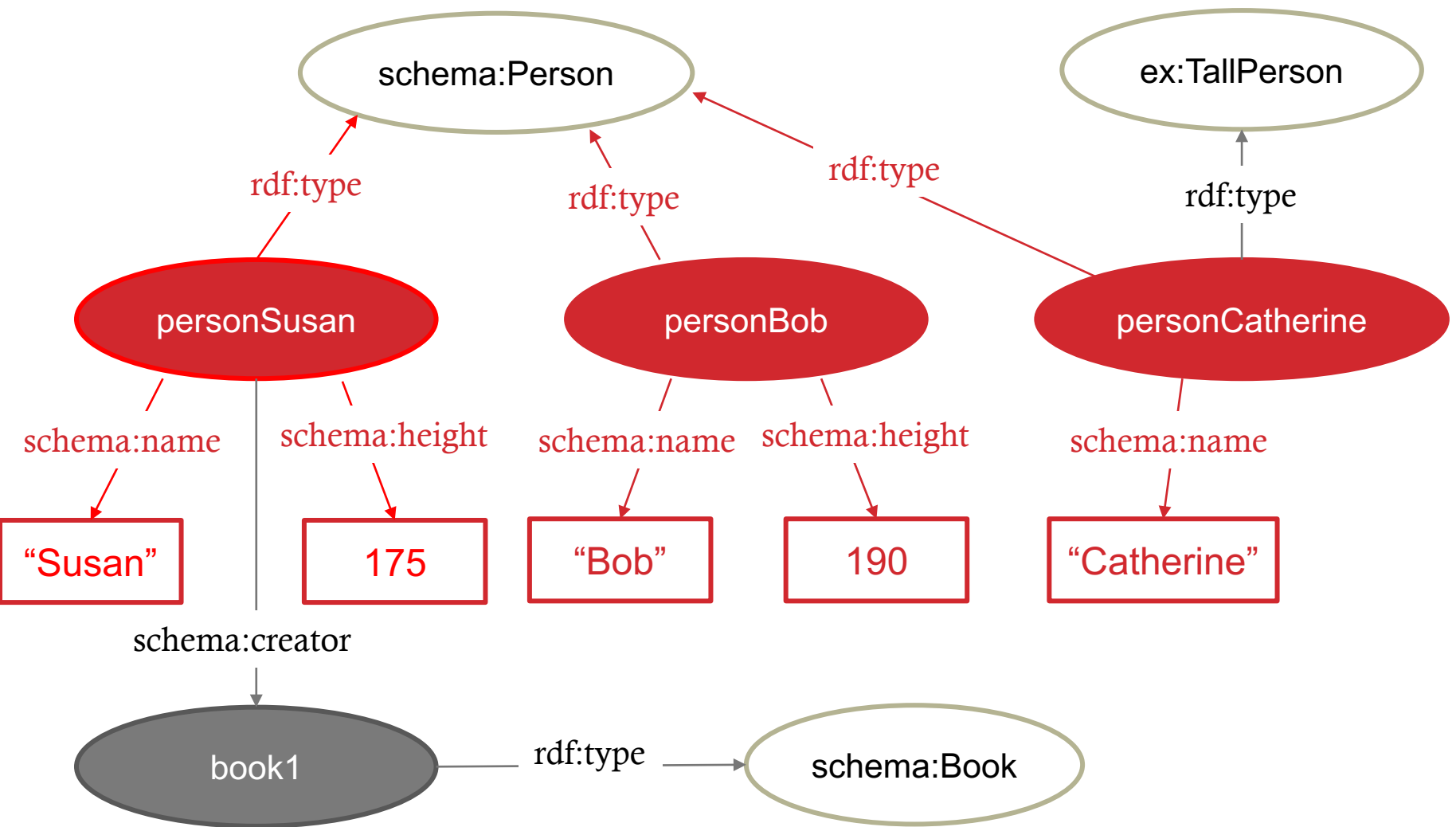
```
    ?person      rdf:type      schema:Person .
```

```
    ?person      schema:name    ?name .
```

```
    OPTIONAL { ?person      schema:height    ?height }
```

```
}
```

Primer – Graf 1



Rezultat upita Q3 nad podacima Graf 1

person	name	height
personSusan	"Susan"	175
personBob	"Bob"	190
personCatherine	"Catherine"	

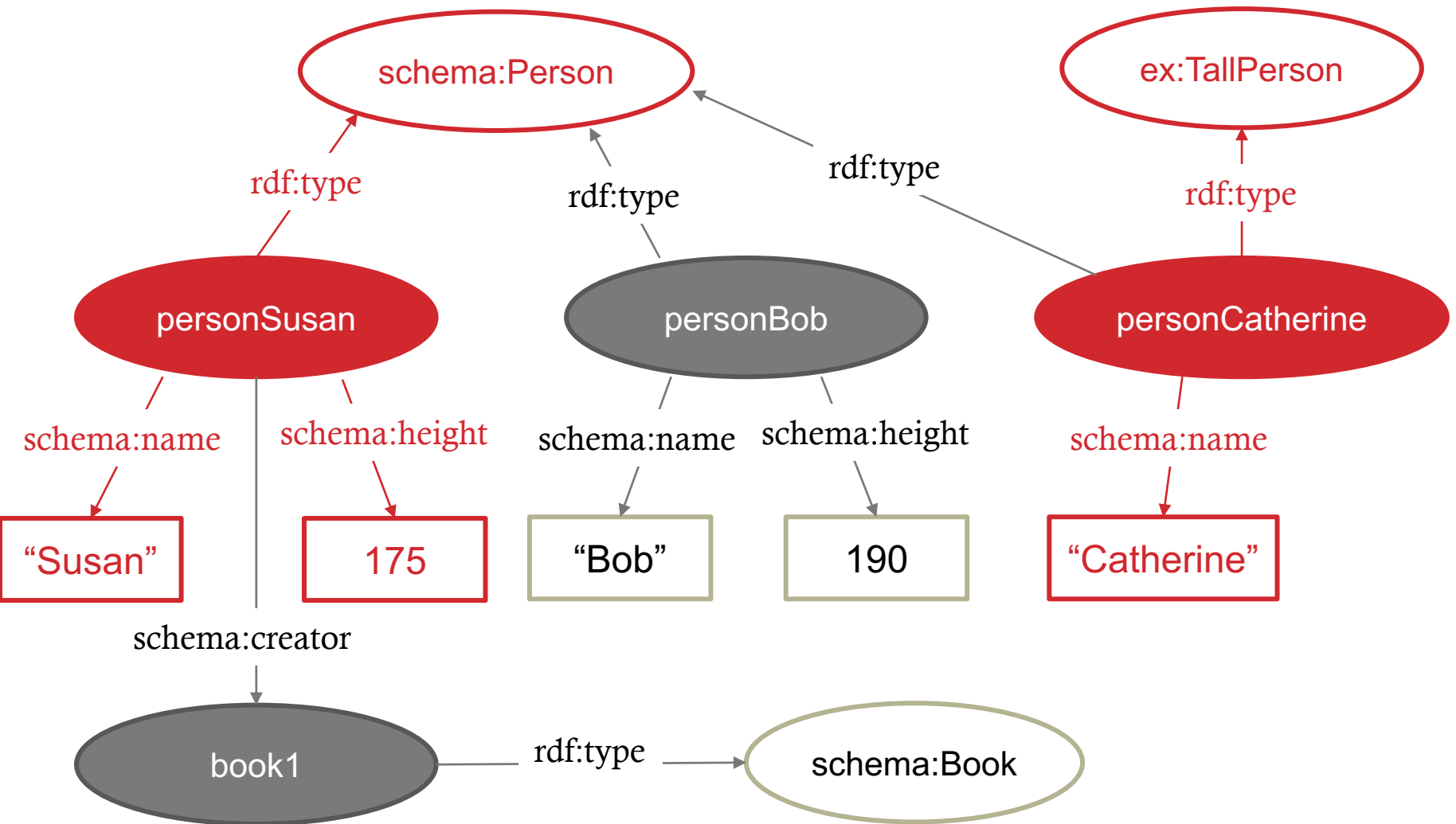
UNION

- Definiše alternative u graf paternu. Rezultat upita su rešenja svih ispunjenih alternativa.

Q4: Visoke osebe definisane eksplicitno ili implicitno

```
SELECT ?person ?name
WHERE {
    ?person      rdf:type      schema:Person .
    ?person      schema:name    ?name .
    {
        {
            ?person      rdf:type      ex:TallPerson
        }
        UNION
        {
            ?person      schema:height ?height .
            FILTER ( ?height > 180 )
        }
    }
}
```

Primer – Graf 1



Rezultat upita Q4 nad podacima Graf 1

person	name
personBob	"Bob"
personCatherine	"Catherine"

Sortiranje i paginacija

ORDER BY za sortiranje

LIMIT broj rezultata

OFFSET pomeraj prvog rezultata

Q5: Rezultati od pozicije 21 do 30 sortirani po imenu

```
SELECT ?person ?name
```

```
WHERE {
```

```
    ?person      rdf:type      schema:Person .
```

```
    ?person      schema:name    ?name .
```

```
}
```

```
ORDER BY ?name
```

```
LIMIT 10
```

```
OFFSET 20
```

BOUND

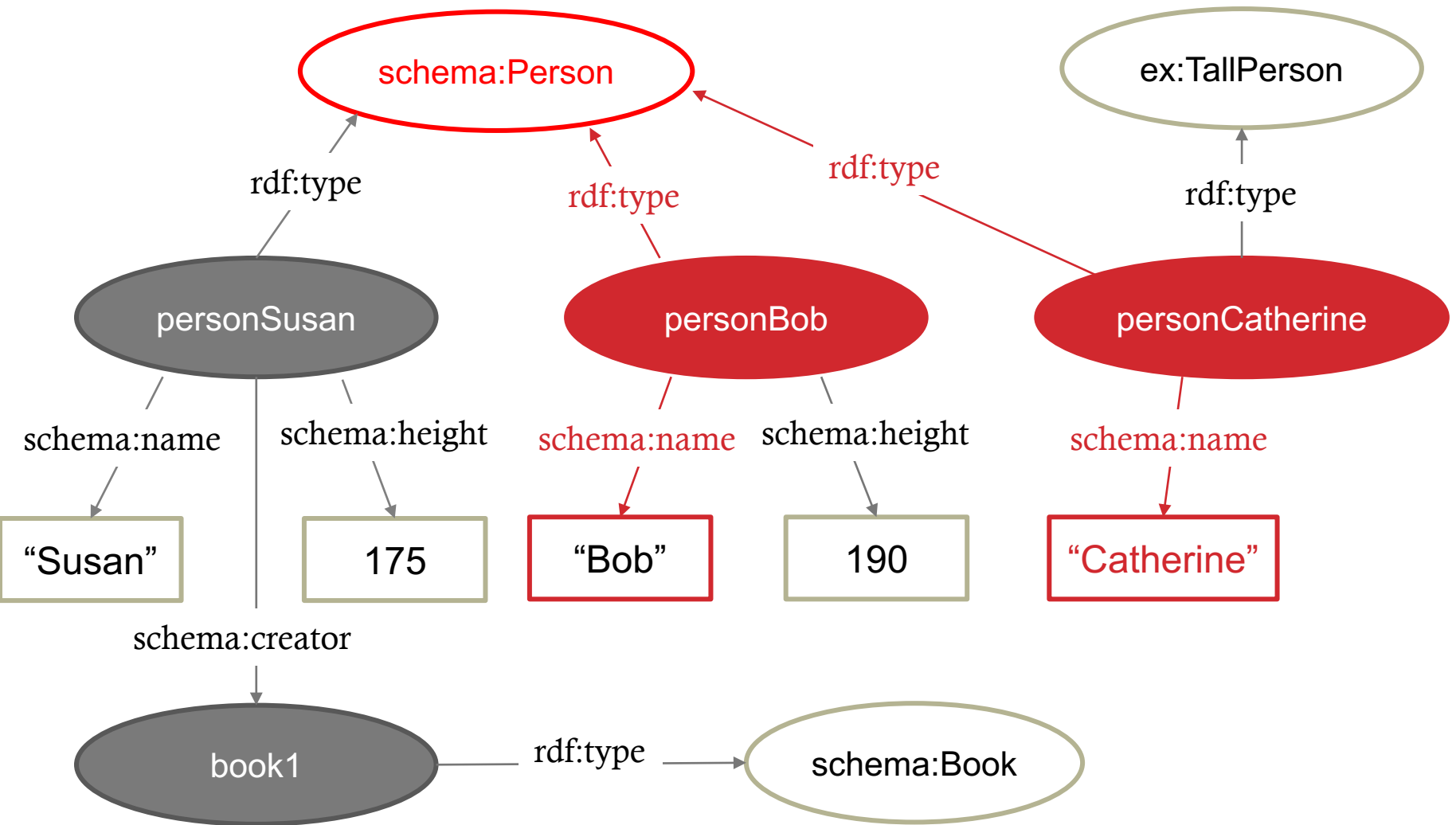
- Proverava da li varijabla ima vrednost ili ne. Može se koristiti i u negaciji.

Q6: osobe koje nisu kreatori nijednog dela

```
SELECT ?person ?name
WHERE {
    ?person      rdf:type      schema:Person .
    ?person      schema:name    ?name .

    OPTIONAL ( ?person schema:creator      ?x )
    FILTER ( !bound(?x) )
}
```

Primer – Graf 1



Rezultat upita Q6 nad podacima Graf 1

person	name
personBob	"Bob"
personCatherine	"Catherine"

ASK

- Koristi se za proveru da li neki upit ima rezultat ili ne
- Vraća true ili false. Ne dobijaju se nikakvi podaci o mogućim rezultatima upita, već samo da li postoji rezultat ili ne

Q7: Da li postoje osobe više od 180 cm?

ASK

{

 ?person schema:height ?height .

FILTER (?height > 180)

}

Rezultat upita Q7 nad podacima Graf 1

true

CONSTRUCT

- Kreira novi RDF graf na osnovu rezultata upita
- CONSTRUCT upit za RDF graf pruža slične funkcionalnosti kao XSLT za XML podatke

Q8: Vraća graf sa instancama klase TallPerson za osobe više od 180 cm

CONSTRUCT

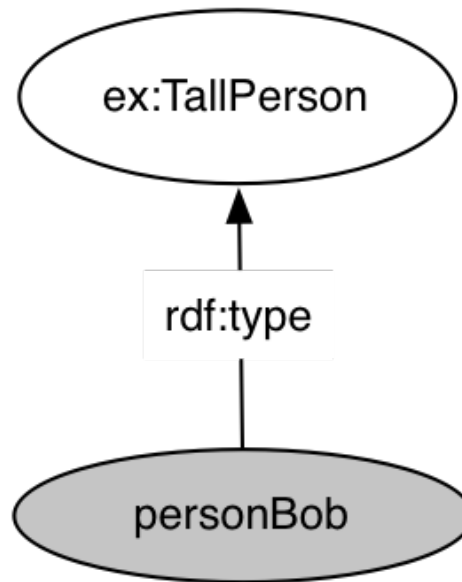
```
{  
    ?person rdf:type ex:TallPerson .  
}
```

WHERE

```
{  
    ?person      rdf:type      schema:Person .  
    ?person      schema:height  ?height .  
    FILTER (?height > 180)  
}
```

Rezultat upita Q8 nad podacima Graf 1

personBob rdf:type ex:TallPerson .



DESCRIBE

- Vraća graf koji sadrži sve relacije (propertije) resursa koji su opisani graf paternom

Q9: vratiti sve informacije o osobi koja se zove “Susan”

DESCRIBE ?person

WHERE {

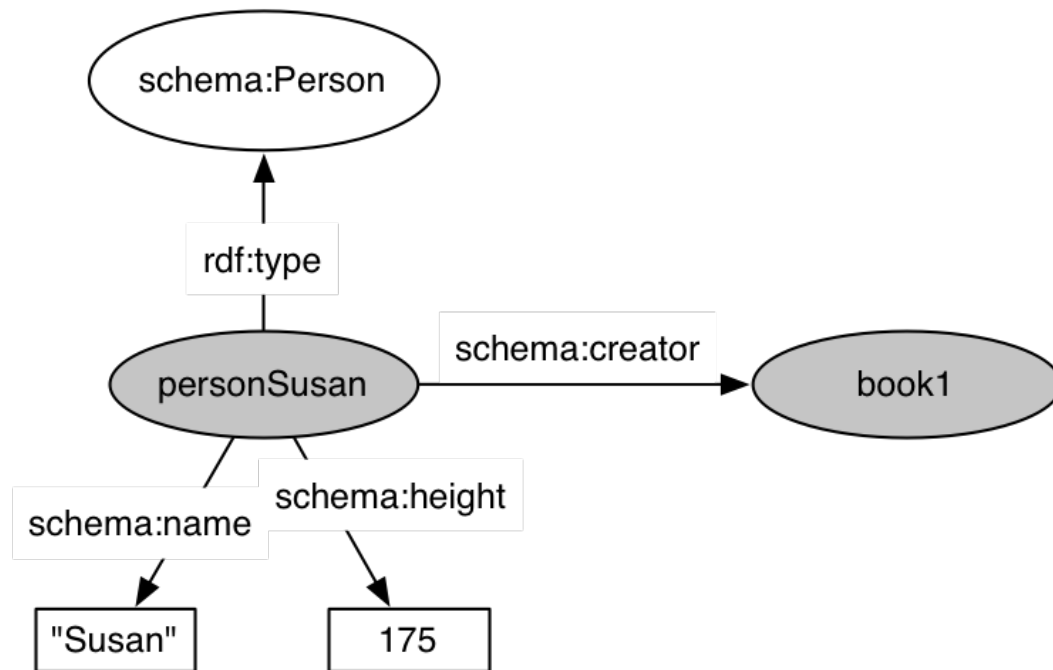
 ?person rdf:type schema:Person .

 ?person schema:name “Susan” .

}

Rezultat upita Q9 nad podacima Graf 1

personSusan	rdf:type	schema:Person .
personSusan	schema:name	"Susan" .
personSusan	schema:height	175 .
personSusan	schema:author	book1 .



Korisni linkovi

“SPARQL By Example”, Lee Feigenbaum, Eric Prud'hommeaux,
<http://www.cambridgesemantics.com/semantic-university/sparql-by-example>

“Bee Node: A FOAF Tale”, Leigh Dodds,
<http://www.ldodds.com/blog/2008/01/bee-node-a-foaf-tale/>

SPARQL Query Language for RDF = specification
<http://www.w3.org/TR/rdf-sparql-query/>

Zahvalnica

Deo sadržaja preuzet sa:

- “SPARQL in a nutshell”, Fabien Gandon,
http://www.slideshare.net/fabien_gandon/sparql-in-a-nutshell

(Anonimni) upitnik za vaše kritike,
komentare, predloge:

<http://goo.gl/cqdp3I>

Pitanja?

Nikola Milikić

Email: nikola.milikic@fon.bg.ac.rs

URL: <http://nikola.milikic.info>