



**KOIO2.**

Ljudi. Znanje. Rezultati.

# **Primjena ontologija u Oracle informacijskim sustavima**

Vedran Vrbanić  
Martina Kocet



## Sadržaj

- Motivacija i prikaz problema
- Tehnologije semantičke pohrane podataka
- Što je ontologija
- Uporaba u praksi
- Performanse
- Zaključak



## Motivacija i prikaz problema

- Izazovi sustavima za upravljanje bazama podataka
  - Velika količina podataka
  - Raznolikost formata zapisanih podataka
  - Loša kvaliteta podataka
  - Otežana prilagodba izmjenama poslovnih procesa
  - Poslovna pravila “skrivena” unutar PL/SQL koda
  - Nefleksibilni i netransparentni izvještajni sustavi
- Korak prema rješenju – sustav “svjestan” semantike podataka



# DB shema vs. ontologija

- DB shema
  - Konceptualna – entiteti, veze
  - Logička – tablice, kolone (upravljanje podacima)
  - Fizička – fizička konfiguracija na mediju za pohranu
- Ontologija
  - Specifikacija konceptualizacije
  - Definira koncepte i veze između njih, što predstavlja strukturu poslovne domene
- Razlika :
  - DB shema – struktura baze podataka
  - Ontologija – struktura određene domene; **s ciljem razmjene znanja**



# Semantičke tehnologije



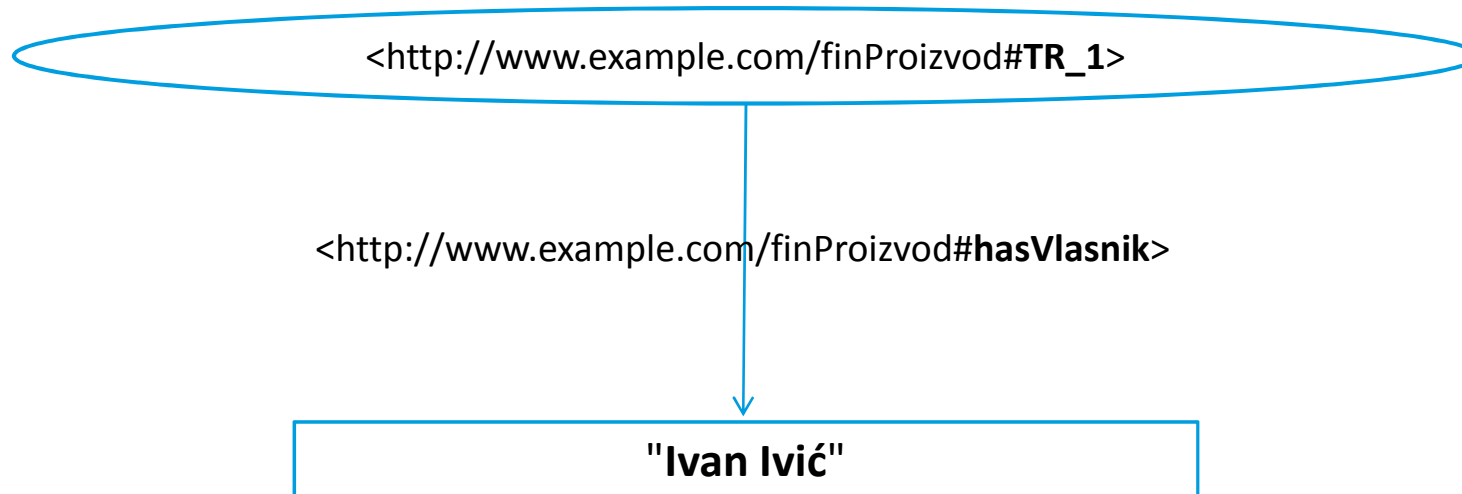
# Resource Description Framework (RDF),

- Skup W3C specifikacija za pohranu podataka u obliku trojki (prostih rečenica):
  - Subjekt – identifikator resursa
  - Predikat – označava relaciju između subjekta i objekta
  - Objekt – vrijednost dodijeljena subjektu
- S, P i O definirani su URI-em (*Uniform Resource Identifier*)
- Trojke čine usmjereni graf: *Vlasnik tekućeg računa TR\_1 je Ivan Ivić*



## RDF graf

- Usmjereni RDF graf (trojka)





# RDF Schema i OWL

- RDFS je nadogradnja RDF-a
- Definira dozvoljeni vokabular RDF modela
  - *rdfs:Class*
  - *rdfs:subClassOf*
  - *rdfs:domain, rdfs:range*
- Web Ontology Language (OWL)
  - Kompleksniji vokabular
  - Tranzitivni, inverzni, jedinstveni predikati
  - Ograničenja kardinalnosti
  - Međusobna isključivost klasa



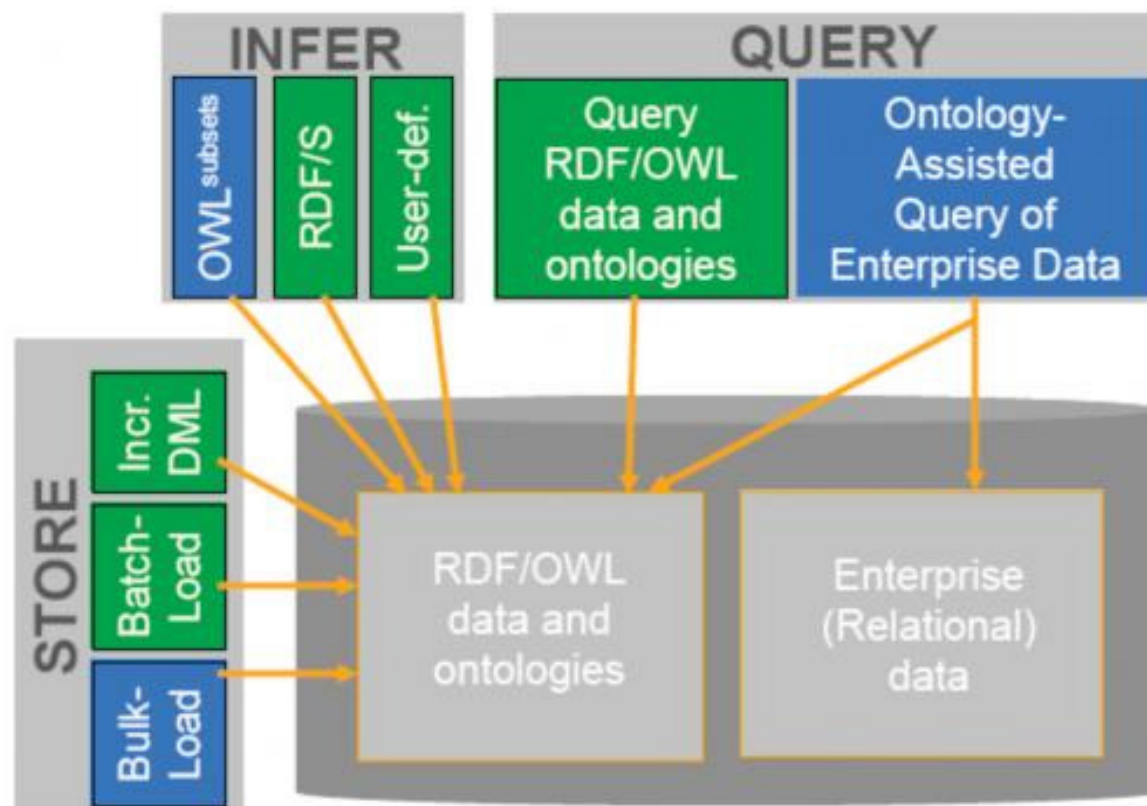


## Rezoniranje

- Razrješavanje postojećih pravila
- Kreiranje novih pravila temeljem:
  - Hijerarhijskih odnosa
  - Korisničkih pravila
- Povezivanje podataka i zaključivanje iz odnosa između koncepata (SQL - isključivo sintaksno povezivanje)



# Semantičke mogućnosti Oracle baze





# Što je ontologija?

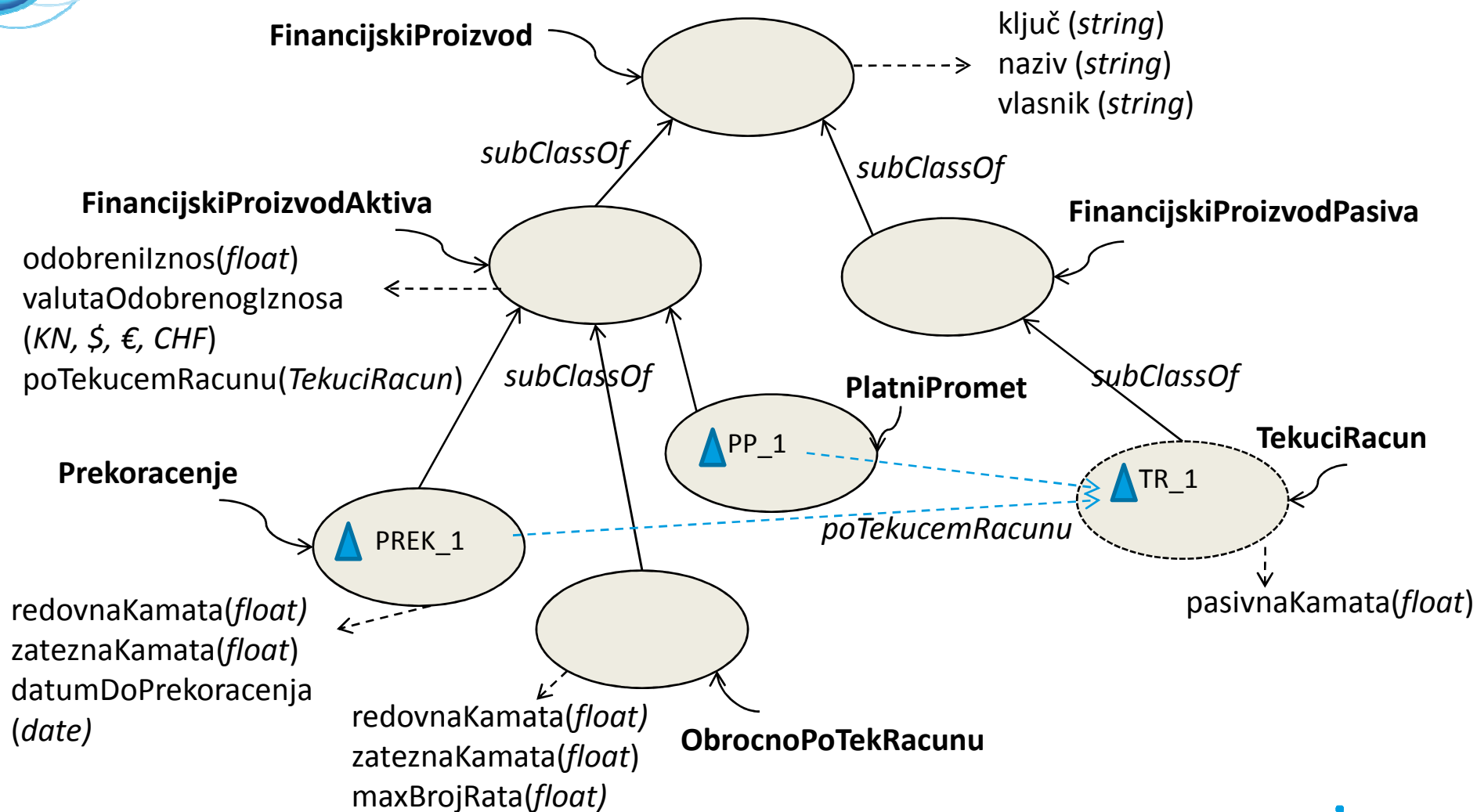
- Formalna, eksplicitna specifikacija koncepata iz određene domene
- Koncepti definirani u obliku pravila zapisanih u trojkama koje čine usmjereni graf
- Uređen vokabular – temelj računalu za raspoznavanje vrijednosti u različitim kontekstima
- Sastoji se od:
  - Klasa
  - Atributa klasa
  - Odnosa između klasa
  - Instanci



# Uporaba u praksi



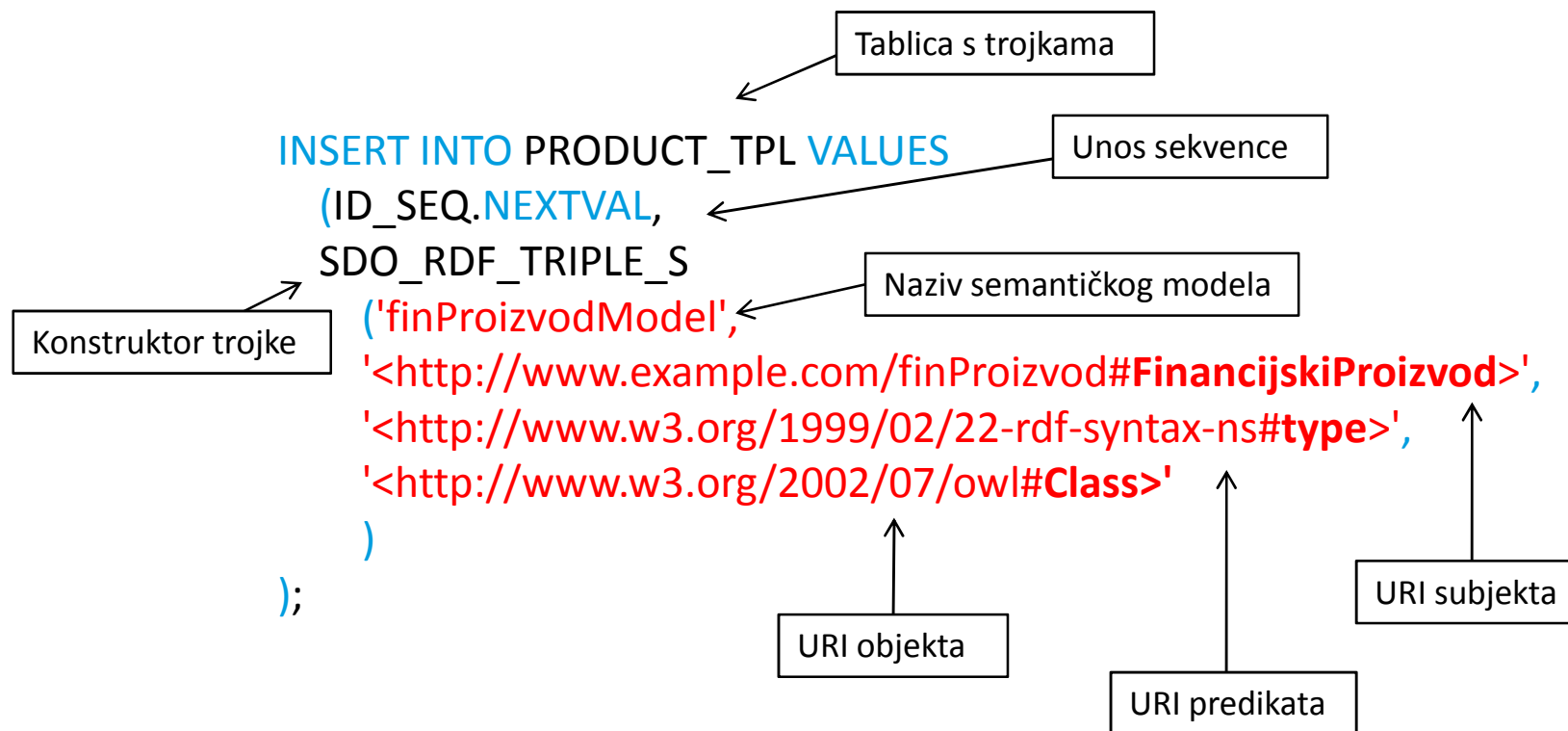
# Opis poslovnog primjera





# Kodiranje modela

- Definicija klase





# Kodiranje modela

- Definicija hijerarhije

```
INSERT INTO PRODUCT_TPL VALUES
(ID_SEQ.NEXTVAL,
SDO_RDF_TRIPLE_S
('finProizvodModel',
'<http://www.example.com/finProizvod#TekuciRacun>',
'<http://www.w3.org/2000/01/rdf-schema#subClassOf>',
'<http://www.example.com/finProizvod#FinancijskiProizvodPasiva>'
)
);
```



# Kodiranje modela

- Kreiranje instanci i pripadne veze

| SUBJEKT        | PREDIKAT                | OBJEKT               |
|----------------|-------------------------|----------------------|
| example:TR_1   | rdf:type                | example:TekuciRacun  |
| example:PREK_1 | rdf:type                | example:Prekoracenje |
| example:PREK_1 | example:poTekucemRacunu | example:TR_1         |

- example: *http://www.example.com/finProizvod#*
- rdf: *http://www.w3.org/1999/02/22-rdf-syntax-ns#*



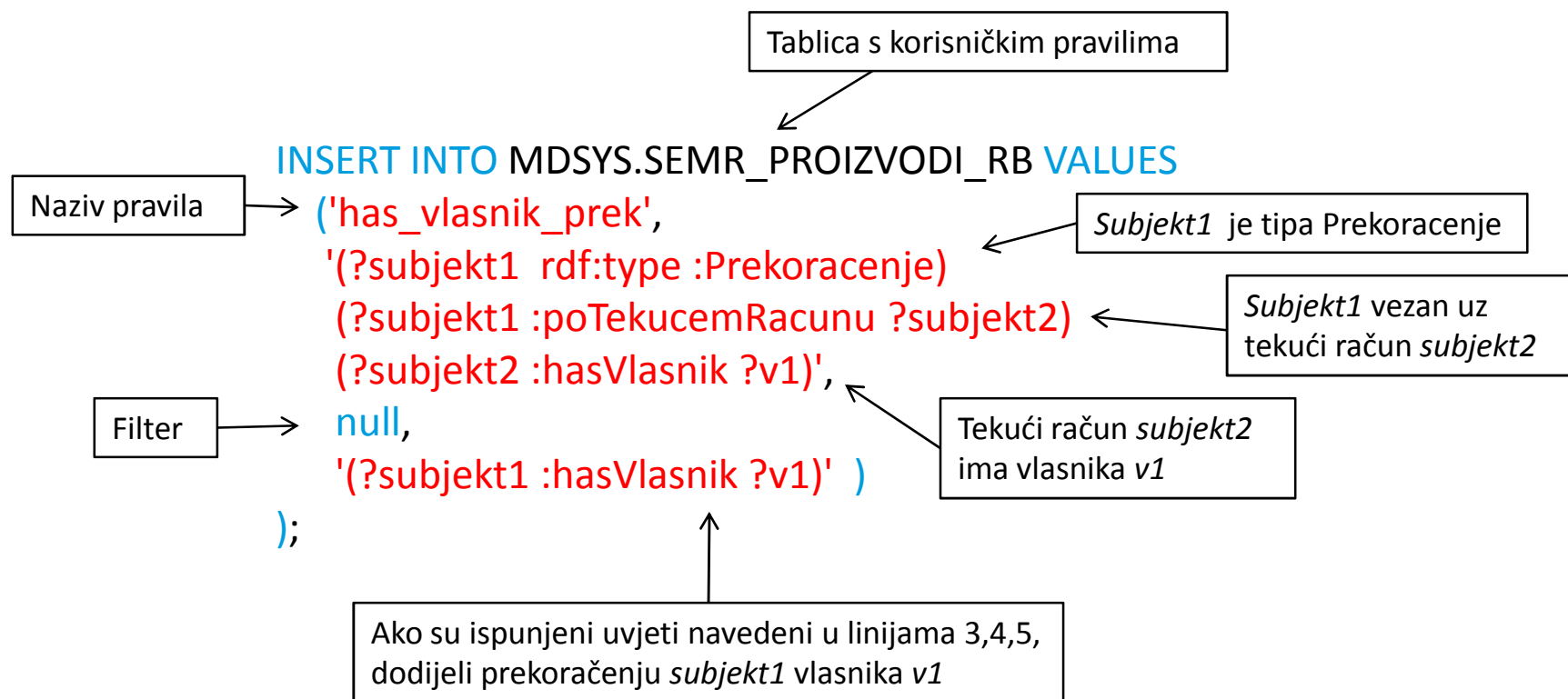


## Korisnički definirana pravila

- Kreiranje specifičnih poslovnih pravila
- Korištenje SPARQL-a (*Simple Protocol and RDF Query Language*)
  - Jezik za pretragu podataka u RDF grafu
  - Pandan SQL-u za pretragu podataka u relacijskim bazama
- Primjer pravila:
  - Ako je prekoračenje (bez eksplicitno navedenog vlasnika) povezano s tekućim računom, prekoračenju dodijeliti istog vlasnika



# Definicija korisničkog pravila





# SPARQL upit i rezoniranje

```
SELECT sub, pred, obj  
FROM TABLE (SEM_MATCH (  
  '{ ?subjekt ?predikat ?objekt .  
    filter(?sub = :PREK_1) }',  
  SEM_MODELS('finProizvodModel'),  
  SEM_RULEBASES('owlprime', 'proizvodi_rb'),  
  SEM_ALIASES(SEM_ALIAS(null, 'http://www.example.com/finProizvod#'),  
    SEM_ALIAS('owl', 'http://www.w3.org/2002/07/owl#')),  
  null)) ;
```

Oracle tablična funkcija za  
pretraživanje semantičkog  
rezpozitorija

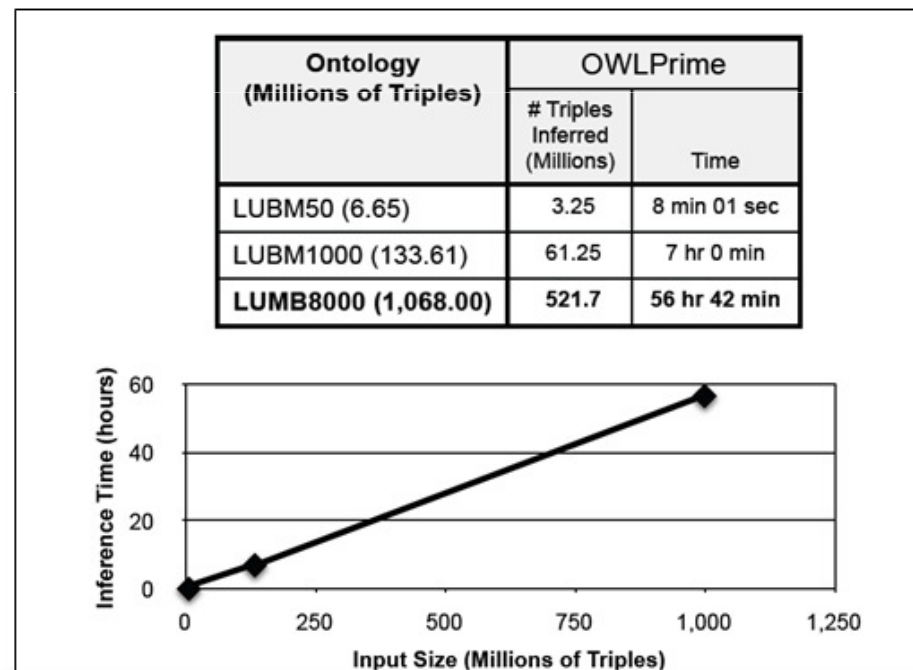
Pravila kod rezoniranja – owl i  
korisnički definirana

| SUBJEKT        | PREDIKAT                | OBJEKT                            |
|----------------|-------------------------|-----------------------------------|
| example:PREK_1 | rdf:type                | example:FinancijskiProizvod       |
| example:PREK_1 | rdf:type                | example:FinancijskiProizvodAktiva |
| example:PREK_1 | rdf:type                | example:Prekoracenje              |
| example:PREK_1 | example:poTekucemRacunu | TR_1                              |
| example:PREK_1 | example:hasVlasnik      | Ivan Ivić                         |



## Performanse

- Vrijeme rezoniranja ovisno o broju trojki i kompleksnosti pravila (broju novih trojki nastalih rezoniranjem)





## Zaključak

- Uređen vokabular koncepata – dobra podloga za opis poslovne domene:
- Prednosti
  - Ponovna iskoristivost znanja određene domene
  - Ontologije su objašnjive same po sebi (nema potrebe za upitima u rječnik podataka)
  - Olakšavaju razumijevanje podataka
  - Fleksibilna pohrana podataka u obliku trojki
- Nedostaci
  - Ne postoje generalne upute za razvoj ontologija
  - Teško predvidljiv utjecaj izmjena na rezoniranje
  - Dugotrajan proces učenja
  - Potencijalno negativan utjecaj na performanse



**Hvala!**

[www.koios.hr](http://www.koios.hr)