

# SPARQL - pokračovanie

## Spracované podľa prednášok

- <http://www.inf.tu-dresden.de/content/institutes/ki/cl/study/summer14/fswt/slides/FSWT2014-L06-SPARQL-Syntax.pdf>
- <http://www.inf.tu-dresden.de/content/institutes/ki/cl/study/summer14/fswt/slides/FSWT2014-L07-SPARQL-Semantics.pdf>

## a dokumentov W3C

- <http://www.w3.org/TR/sparql11-query/>
- <http://www.w3.org/TR/rdf11-datasets/>

## Dalšie zdroje

- <http://www.cambridgesemantics.com/semantic-university/learn-sparql>
- <https://www.cambridgesemantics.com/blog/semantic-university/learn-sparql/sparql-by-example/>

# Typy dotazov - Query forms

<http://www.w3.org/TR/rdf-sparql-query/#QueryForms>

- **SELECT** – vracia množinu (alebo jej čas) riešení vo formate tabulky, xml alebo json. Riešenie (binding) je funkcia, ktorá mapuje (binds) premenné na RDF-termíny.
- **ASK** – vracia logickú hodnotu (true, false). Odpovedá na otázku, či existuje riešenie pre BGP. T.j. či z dotazovanej datovej sady (databázy znalostí) vyplýva dôsledok, ktorý má štruktúru BGP.
- **CONSTRUCT** – návratová hodnota je nový RDF-graf. Nájde riešenie (rovnako ako SELECT) a hodnoty premenných použije na vytvorenie nového RDF-grafu podľa šablony zadanej klientom v dotaze.
- **DESCRIBE** – návratová hodnota je nový RDF-graf. Nájde riešenie (rovnako ako SELECT) a hodnoty premenných použije na vytvorenie nového RDF-grafu. Na rozdiel od CONSTRUCT štruktúru grafu nedefinuje klient ale je definovaná servisom. Zostrojený graf má poskytovať informácie o nájdených objektoch (resursoch)

# Formáty výstupu

<http://www.w3.org/TR/2013/REC-sparql11-query-20130321/#selectproject>

SELECT může výsledky vrátit' okrem tabulky aj vo formátoch **xml**, **json** a **csv/tsv**:

## xml

```
<?xml version="1.0"?>
<sparql xmlns="http://www.w3.org/2005/sparql-results#">
  <head>
    <variable name="autor"/>
    <variable name="titul"/>
  </head>
  <results>
    <result>
      <binding name="autor">
        <uri>http://eg.org/Orwell</uri>
      </binding>
      <binding name="titul">
        <literal>Zvieracia farma</literal>
      </binding>
    </result>
    <result> a.t.d'
```

## tsv:

```
?autor ?titul
<http://eg.org/Orwell> "Zvieracia farma"
<http://eg.org/Lindgren> "Pipi dlhá pančucha"
<http://eg.org/Tolkien> "Pán prsteňa"
<http://eg.org/Tolkien> "Hobit"
```

# Rozšírenia v SPARQL 1.1

- **SPARQL QUERY 1.1:** <http://www.w3.org/TR/sparql11-query/>

Rozšírenia dotazovacieho jazyka o mnoho ďalších konštrukcií analogických ako v SQL

- **SPARQL Update:** <http://www.w3.org/TR/sparql11-update/>

Jazyk pre modifikáciu RDF-grafov (adding, removing triples)

- **Datasets**

Dotazy nad znalostnou bázou pozostávajúcou z viacerých grafov.

- **Federation extension:** <http://www.w3.org/TR/sparql11-federated-query/>

Rozšírenie jazyka pre distribuované dotazy.

- **Service Description:** <http://www.w3.org/TR/sparql11-service-description/>

Špecifikácia mechanizmov pre zisťovanie konfigurácie služby, slovníka a dátových sád, ktoré poskytuje SPARQL-endpoint.

- **Graph Store HTTP Protokol:** <http://www.w3.org/TR/sparql11-http-rdf-update/>

Definuje HTTP (RESTful) servis pre manažment RDF-databázy (RDF Graph Store).

- **Entailment Regimes:** <http://www.w3.org/TR/sparql11-entailment/>

Špecifikácia definujúca sémantiku SPARQL pre rôzne dedukčné režimy.

# SPARQL QUERY 1.1

## Ďalšie možnosti dotazovovacieho jazyka

- Aggregates: **GROUP BY, HAVING**
  - Aggregate functions: **COUNT, MAX, MIN, SUM, AVG, SAMPLE,...**
- Subqueries
- Negations: **NOT EXISTS, MINUS**
- Property path
- Projected Expressions – priradenie výrazov premenným: **BIND, AS**

### **Pozri:**

- <https://www.w3.org/TR/2013/REC-sparql11-query-20130321/> - štandard
- <https://www.cambridgesemantics.com/blog/semantic-university/learn-sparql/sparql-by-example/> - tutorial Cambridg University
- <http://vocab.getty.edu/queries> - množstvo príkladov

**Pozn.** *Ilustračné príklady na cvičeniach...*

# SPARQL Update

<http://www.w3.org/TR/sparql11-update>

<http://www.w3.org/TR/2013/REC-sparql11-update-20130321/#insertData>

## **INSERT DATA** Vloženie trojic do default grafu

```
PREFIX : <http://rzz.edu/>
INSERT DATA {
  :Lindgren a :Writer .
  _:a1 :title "Pipi";
        :author :Lindgren.
}
```

## **DELETE DATA** Vymazanie konkrétnej trojice

```
PREFIX : <http://rzz.edu/>
DELETE DATA {
  :Lindgren a :Writer .
}
```

Vymazanie celého datasetu

**CLEAR ALL**

# Datové sady s viacerými grafmi

*Pripomeňme si, že endpoint umožňuje dotazovať údaje (znalostnú bázu) uložené v datase.*

*Dataset obsahuje jeden primárny (default) rdf-graf ale môže mať aj ďalšie pomenované grafy.*

## Ukážka - default graf

```
PREFIX ex: <http://eg.org/>
INSERT DATA {
  _:a ex:author ex:Tolkien;
      ex:title "Hobit".
  _:b ex:author ex:Tolkien ;
      ex:title "Pán prsteňa" ;
      ex:price "10.50" .
  _:c ex:author ex:Lindgren ;
      ex:title "Pipi dlhá pančucha" ;
      ex:price "17".
  _:d ex:author ex:Orwell ;
      ex:title "Zvieracia farma" ;
      ex:price "9".
}
```

# SPARQL Update – pomenovaných grafov

```
{PREFIX ex: <http://eg.org/>
INSERT DATA GRAPH ex:de {
  _:b ex:author ex:Lindgren ;
    ex:title "Pipi Langstrumpf";
    ex:price "12" .
}}
```

```
WITH <g1> DELETE { a b c } INSERT { x y z } WHERE { ... }
```

Vymazanie všetkých trojíc z pomenovaného grafu

```
PREFIX ex: <http://eg.org/>
WITH ex:de
DELETE { ?person ?property ?value }
WHERE { ?person ?property ?value }
```

Vymazanie default , pomenovaného grafu alebo celého datasetu

```
CLEAR ( SILENT )? ( GRAPH IRIref | DEFAULT | NAMED | ALL )
```

Import grafu

```
LOAD ( SILENT )? IRIref\_from ( INTO GRAPH IRIref\_to )?
```



# Ukážka - pomenované grafy

Pridáme **graf ex:de** so zoznamom nemeckých kníh:

```
PREFIX ex: <http://eg.org/>
```

```
INSERT DATA
```

```
{
```

```
GRAPH ex:de {
```

```
_:b ex:author ex:Tolkien ;
```

```
  ex:title "Herr der Ringe" ;
```

```
  ex:price "9.50".
```

```
_:c ex:author ex:Lindgren ;
```

```
  ex:title "Pipi Langstrumpf";
```

```
  ex:price "12" .
```

```
}}
```

a **graf ex:gauth** s celými menami autorov:

```
PREFIX ex: <http://eg.org/>
```

```
INSERT DATA
```

```
{
```

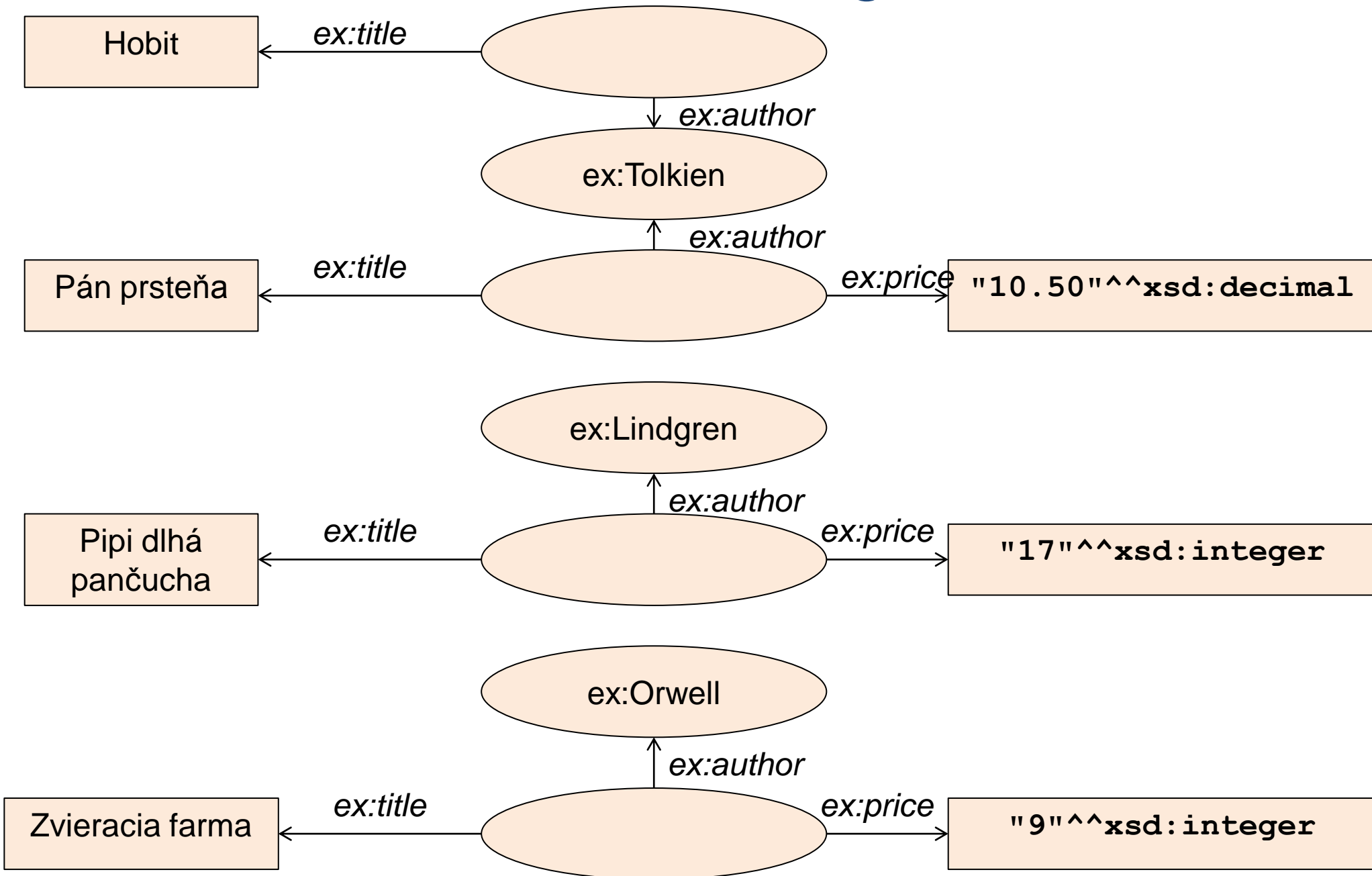
```
GRAPH ex:gauth {
```

```
ex:Tolkien ex:name "John R. R. Tolkien".
```

```
ex:Lindgren ex:name "Astrid Lindgren".
```

```
}}
```

# Príklad – RDFgraf



# GRAPH - dotaz na pomenovaný graf

*Všetky dotazy, ktoré sme si doposiaľ ukazovali dotazovali údaje z default grafu.*

## Dotaz:

```
PREFIX ex: <http://eg.org/>
SELECT ?autor ?nazov
WHERE {
  GRAPH ex:de { ?x ex:author ?autor.
                 ?x ex:title ?nazov. }
}
```

## Odpoved'

*Výsledky len z menovaného grafu **ex:de***

```
-----
| autor          | nazov          |
=====
| ex:Lindgren   | "Pipi Langstrumpf" |
| ex:Tolkien    | "Herr der Ringe"  |
-----
```

# Dotaz na všechny pomenované grafy

## Dotaz:

```
PREFIX ex: <http://eg.org/>
SELECT *
WHERE {
  GRAPH ?g {?s ?p ?o}
}
```

## Odpoved'

Vrátí všechny trojice zo všech pomenovaných grafov

s	p	o	g
ex:Lindgren	ex:name	"Astrid Lindgren"	ex:gauth
ex:Tolkien	ex:name	"John R. R. Tolkien"	ex:gauth
_:b0	ex:price	"9.50"	ex:de
_:b0	ex:title	"Herr der Ringe"	ex:de
_:b0	ex:author	ex:Tolkien	ex:de
_:b1	ex:price	"12"	ex:de
_:b1	ex:title	"Pipi Langstrumpf"	ex:de
_:b1	ex:author	ex:Lindgren	ex:de

# Kombinácia dotazu z rôznych grafov

Dotaz: **vypísať mena autorov namiesto ich IRI**

```
PREFIX ex: <http://eg.org/>
SELECT ?autor ?titul
WHERE {
  ?x ex:author ?a.
  ?x ex:title ?titul.
  GRAPH ex:gauth {?a ex:name ?autor}
}
```

Odpoved'

**Premenné (?a) sú zdieľané**

autor	titul
"Astrid Lindgren"	"Pipi dlhá pančucha"
"John R. R. Tolkien"	"Pán prsteňa"
"John R. R. Tolkien"	"Hobit"

# Prázdne uzly z rôznych grafov

**Dotaz:** skombinujeme nemecké tituly a slovenské ceny

```
PREFIX ex: <http://eg.org/>
SELECT ?Title ?cena
WHERE {
  ?k ex:price ?cena
  GRAPH ex:de {?k ex:title ?Title}
}
```

**Odpoved':** žiadne výsledky

Poznámka. problém však nie je v tom, že by nebolo možné predávať nemecké knihy za slovenská ceny.

Problém je v tom, že **prázdne uzly medzi grafmi nie sú zdieľané** - takto sa správa FUSEKI server.

Iné endpointy však môžu zvoliť iný prístup pri interpretácii prázdnych uzlov.

**Poznámka:**

Štandard (zatiaľ) nedefinuje mechanizmy, ktorými je možné dosiahnuť rôzne interpretácie dátových sád.

Interprácia teda závisí od konkrétnej implementácie SPARQL-endpontu.

Diskusia je na <http://www.w3.org/TR/rdf11-datasets/#declaring>

# Sémantika dátovej sady

To či grafy dátovej sady maju byť spájané zlúčením alebo zjednotením záleží na interpretácie dátovej sady.

Aký je vzťah medzi jednotlivými grafmi navzájom a ku celému datasetu?

Do úvahy prichádzajú **rôzne interpretácie**:

<http://www.w3.org/TR/rdf11-datasets/#sec-introduction>

- trojice z pomenovaných grafov sú zároveň súčasťou default grafu. (logické rozčlenenie znalostí)
- rôzne pomenované grafy predstavujú rôzne kontexty. (Např. stav znalostnej bázy k určitému dátumu.) Default graf môže (ale nemusí ) predstavovať globálny kontext platný pre pomenované grafy resp. predstavovať metadata pre pomenované grafy.
- pomenované grafy sú „hypotetické grafy“ predstavujúce hypotézy, ktorých dôsledkom sú fakty reprezentované default grafom. (Hypotézy si môžu navzájom aj odporovať)
- jednotlivé grafy vôbec nesúvisia. (Každý reprezentuje samostatnú bázu znalostí.)

# Federation queries

Klauzula service:

```
SERVICE endpoint-IRI { BGP }
```

presmeruje vyhodnotenie BGP na zadaný endpoint

**V dbpedii zistíme mená autorov kníh z našej lokálnej databázy**

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

```
PREFIX dbo: <http://dbpedia.org/ontology/>
```

```
PREFIX ex: <http://eg.org/>
```

```
SELECT ?kniha (STR(?a) AS ?autor) ?narodnost
```

```
WHERE {
```

```
  ?x ex:title ?kniha;
```

```
  ex:author ?y .
```

```
SERVICE <http://dbpedia.org/sparql> {
```

```
  ?y rdfs:label ?a.
```

```
  ?y dbo:nationality ?narodnost .
```

```
  FILTER ( LANG(?a) = "en")
```

```
}
```

```
}
```



# Service Description

**Service description** – je mechanizmus pre zisťovanie konfigurácie služby, slovníka a dátových sád, ktoré poskytuje SPARQL-endpoint.

**Príklad** : Použite SPARQLer klienta <http://sparql.org/sparql.html> na dotaz na dbpedia sparql service:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX sd: <http://www.w3.org/ns/sparql-service-description#>
PREFIX sdf: <http://www.w3.org/ns/formats/>
SELECT * FROM <http://dbpedia.org/sparql/>
WHERE {?s ?p ?o}
```

## Odpoved'

s	p	o
<http://dbpedia.org/sparql>	rdf:type	sd:Service
<http://dbpedia.org/sparql>	sd:resultFormat	sdf:Turtle
<http://dbpedia.org/sparql>	sd:resultFormat	sdf:SPARQL_Results_XML
<http://dbpedia.org/sparql>	sd:url	<http://dbpedia.org/sparql>
<http://dbpedia.org/sparql>	sd:resultFormat	sdf:SPARQL_Results_CSV
<http://dbpedia.org/sparql>	sd:endpoint	<http://dbpedia.org/sparql>
<http://dbpedia.org/sparql>	sd:feature	sd:UnionDefaultGraph
<http://dbpedia.org/sparql>	sd:supportedLanguage	sd:SPARQL10Query
<http://dbpedia.org/sparql>	sd:resultFormat	sdf:N-Triples
<http://dbpedia.org/sparql>	sd:resultFormat	sdf:N3
<http://dbpedia.org/sparql>	sd:resultFormat	sdf:RDFa
<http://dbpedia.org/sparql>	sd:resultFormat	sdf:RDF_XML
<http://dbpedia.org/sparql>	sd:resultFormat	sdf:SPARQL_Results_JSON
<http://dbpedia.org/sparql>	sd:feature	sd:DereferencesURIs