



# Linked Data Engineering

## Lecture 4: Querying RDF with SPARQL

### 4.6 SPARQL is more than a Query Language



Leibniz Institute for Information Infrastructure

Prof. Dr. Harald Sack  
FIZ Karlsruhe

Leibniz Institute for Information Infrastructure  
Karlsruhe Institute of Technology

**Autumn 2016**

# SPARQL - A Query Language for RDF

- **SPARQL Protocol and RDF Query Language** is
  - a **Query Language** for RDF graph traversal  
(*SPARQL Query Language Specification*)
  - a **Protocol Layer**, to use SPARQL via http  
(*SPARQL Protocol for RDF Specification*)
  - an **XML Output Format Specification** for SPARQL queries  
(*SPARQL Query XML Results Format*)

# SPARQL Result Format

- SPARQL results are given as well formed and valid XML documents

```
<?xml version="1.0"?>
<sparql xmlns="http://www.w3.org/2005/sparql-results#">
  ...
</sparql>
```

- In a **<head>** element all variables of the SPARQL query are listed

```
<head>
  <variable name="x"/>
  <variable name="hpage"/>
  <variable name="name"/>
  <variable name="mbox"/>
  <variable name="blurb"/>
</head>
```

# SPARQL Result Format

- For each SPARQL Query result exists a `<result>` element

```
<?xml version="1.0"?>
<sparql xmlns="http://www.w3.org/2005/sparql-results#">
  <head>
    <variable name="x"/>
    ...
  </head>
  <results>
    <result>
      <binding name="x"> ... </binding>
      <binding name="hpage"> ... </binding>
    </result>

    <result> ... </result>
    ...
  </results>
</sparql>
```

*single SPARQL query result*

# SPARQL Result Format

- Within a `<binding>` element a `<head>` variable is bound to a result

```

<result>
  <binding name="x">
    <bnode>r2</bnode>
  </binding>
  <binding name="hpage">
    <uri>http://work.example.org/bob/</uri>
  </binding>
  <binding name="name">
    <literal xml:lang="en">Bob</literal>
  </binding>
  <binding name="age">
    <literal datatype="http://www.w3.org/2001/XMLSchema#integer">
      30
    </literal>
  </binding>
  <binding name="mbox">
    <uri>mailto:bob@work.example.org</uri>
  </binding>
</result>

```

*variable bound to result*

# SPARQL - A Query Language for RDF

- **SPARQL Protocol and RDF Query Language** is
  - a **Query Language** for RDF graph traversal  
(*SPARQL Query Language Specification*)
  - a **Protocol Layer**, to use SPARQL via http  
(*SPARQL Protocol for RDF Specification*)
  - an **XML Output Format Specification** for SPARQL queries  
(*SPARQL Query XML Results Format*)

# SPARQL Protocol

- Method to query/respond of SPARQL queries via http
- A SPARQL URI consists out of 3 parts:
  - (1) URL of a SPARQL endpoint (e.g. <http://example.org/sparql>)
  - (2) RDF Graph(s) to be queried  
(optional, part of the query string,  
e.g. [named-graph-uri=http://example.org/testrdf.rdf](#))
  - (3) SPARQL query  
(part of the query string, e.g. [query=SELECT...](#))

```
http://example.org/sparql?named-graph-uri=http%3A%2F%2Fexample.org%2Ftestrdf&
query=SELECT+%3Freview_graph+WHERE+%7B%0D%0A++GRAPH+%3Freview_graph+%7B%0D%0A+++++%3Freview+rev%3Arating+10+.%0D%0A++%7D%0D%0A%7D
```



# SPARQL Protocol - Example

- Simple SPARQL query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbo: <http://dbpedia.org/ontology/>
SELECT ?author ?work
WHERE {
    ?author rdf:type dbo:Writer ;
           dbo:notableWork ?work .
} LIMIT 100
```

- HTTP Trace of the SPARQL query

```
GET
http://dbpedia.org/sparql?default-graph-uri=http%3A%2F%2Fdbpedia.org&query=PREFIX+rdf%3A+%3Ch
ttp%3A%2F%2Fwww.w3.org%2F1999%2F02%2F22-rdf-syntax-ns%23%3E%0D%0APREFIX+dbo%3A+%3C
http%3A%2F%2Fdbpedia.org%2Fontology%2F%3E%0D%0ASELECT+%3Fauthor++%3Fwork%0D%0AWH
ERE+%7B%0D%0A++++++%3Fauthor+rdf%3A+type+dbo%3AWriter+%3B%0D%0A++++++dbo
%3AnotableWork+%3Fwork+.%0D%0A%7D+LIMIT+100%0D%0A
Host: dbpedia.org
User-agent: Mozilla/5.0 ...
Accept:text/html,application/xhtml+xml,application/xml
```



# SPARQL Protocol - Example

- HTTP Trace of the SPARQL response

```
HTTP/1.1 200 OK
Date: Tue, 18 Aug 2015 09:55:07 GMT
Content-Type: application/sparql-results+xml; charset=UTF-8
Content-Length: 21055
Connection: keep-alive
Server: Virtuoso/07.20.3214 (Linux) x86_64-redhat-linux-gnu VDB
X-SPARQL-default-graph: http://dbpedia.org
...
<sparql xmlns="http://www.w3.org/2005/sparql-results#" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xsi:schemaLocation="http://www.w3.org/2001/sw/DataAccess/rf1/result2.xsd">
  <head>
    <variable name="author"/>
    <variable name="work"/>
  </head>
  <results distinct="false" ordered="true">
    <result>
      <binding name="author"><uri>http://dbpedia.org/resource/Ding_Ling</uri></binding>
      <binding name="work"><uri>http://dbpedia.org/resource/Miss_Sophia&#39;s_Diary</uri></binding>
    </result>
    ...
  </results>
</sparql>
```

# SPARQL is not only a Query Language

- **SPARQL Protocol and RDF Query Language** is
  - a **Query Language** for RDF graph traversal  
(*SPARQL Query Language Specification*)
  - a **Protocol Layer**, to use SPARQL via http  
(*SPARQL Protocol for RDF Specification*)
  - an **XML Output Format Specification** for SPARQL queries  
(*SPARQL Query XML Results Format*)

# SPARQL is not only a Query Language

- In addition to `SELECT` queries SPARQL allows:
- **ASK**
  - Check whether there is at least one result
  - Result: true or false
  - Result is delivered as XML or JSON

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

ASK
FROM <http://dbpedia.org/>
WHERE {
    ?author rdf:type dbo:Writer .
    ?author dbo:notableWork ?work .
}
```

- *Example: Is there an author with a notable work?*

# SPARQL is not only a Query Language

- In addition to `SELECT` queries SPARQL allows:
- **DESCRIBE**
  - Result: an RDF graph with data about resources
  - Result is RDF/XML or Turtle

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

DESCRIBE ?author ?work
FROM <http://dbpedia.org/>
WHERE {
    ?author rdf:type dbo:Writer .
    ?author dbo:notableWork ?work .
} LIMIT 10
```

# SPARQL is not only a Query Language

- In addition to `SELECT` queries SPARQL allows:
- **CONSTRUCT**
  - Result: an RDF graph constructed from a template
  - Template: graph pattern with variables from the query pattern
  - Result is RDF/XML or Turtle

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

CONSTRUCT { ?author <http://example.org/hasWritten> ?work .}
FROM <http://dbpedia.org/>
WHERE {
    ?author rdf:type dbo:Writer .
    ?author dbo:notableWork ?work .
} LIMIT 10
```



