



# SPARQL

**Andy Seaborn**

© 2006 Hewlett-Packard Development Company, L.P.  
The information contained herein is subject to change without notice.



## SPARQL



1. Query Language
  2. Protocol
    - HTTP binding
    - SOAP binding
  3. XML Results Format
    - Easy to transform (XSLT, XQuery)
- Status: Later stages of standardisation
    - Design finished, getting implementation feedback

# SPARQL

- Basic Graph Pattern Matching
  - Building block for data access and extensibility
- Algebra: combining graph patterns
  - Building block for data access and extensibility
  - Filters for restricting values
- Solution Modifiers
  - ORDER BY, LIMIT/OFFSET, DISTINCT, REDUCED
- Result forms
  - SELECT, CONSTRUCT, DESCRIBE, ASK

# It's Turtles all the way down

- Turtle: An RDF serialization
  - The RDF part of N3
  - Commonly used in examples (and tutorials and papers)
  - SPARQL uses Turtle+variables as triple pattern syntax

```
@prefix person: <http://example/person/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

person:A foaf:name "Alice" .
person:A foaf:mbox <mailto:alice@example.net> .

person:B foaf:name "Bob" .
```



## SPARQL : Triple Pattern

```
@prefix person: <http://example/person/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

person:A foaf:name "Alice" .
person:A foaf:mbox <mailto:alice@example.net> .
person:B foaf:name "Bob" .
```

```
PREFIX person: <http://example/person/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT ?name
WHERE
{ ?x foaf:name ?name }
```

```
-----
| name |
=====
| "Bob" |
| "Alice" |
-----
```

5



## SPARQL : Basic Graph Pattern

```
@prefix person: <http://example/person/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

person:A foaf:name "Alice" .
person:A foaf:mbox <mailto:alice@example.net> .
person:B foaf:name "Bob" .
```

```
PREFIX person: <http://example/person/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT ?name
WHERE
{ ?person foaf:mbox <mailto:alice@example.net> .
  ?person foaf:name ?name . }
```

```
-----
| name |
=====
| "Alice" |
-----
```

6



## SPARQL : FILTER

```
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix stock: <http://example.org/stock#> .
@prefix inv: <http://example.org/inventory#> .

stock:book1 dc:title "SPARQL Query Language Tutorial" .
stock:book1 inv:price 10 .
stock:book1 inv:quantity 3 .

stock:book2 dc:title "SPARQL Query Language (2nd ed)" .
stock:book2 inv:price 20 ; inv:quantity 5 .

stock:book3 dc:title "Moving from SQL to SPARQL" .
stock:book3 inv:price 5 ; inv:quantity 0 .

stock:book4 dc:title "Applying XQuery" .
stock:book4 inv:price 20 ; inv:quantity 8 .
```

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX stock: <http://example.org/stock#>
PREFIX inv: <http://example.org/inventory#>

SELECT ?book ?title
WHERE {
  ?book dc:title ?title .
  ?book inv:price ?price . FILTER ( ?price < 15 )
  ?book inv:quantity ?num . FILTER ( ?num > 0 ) }
```

```
-----
| book          | title                                     |
=====
| stock:book1  | "SPARQL Query Language Tutorial"        |
-----
```



## SPARQL : OPTIONAL

```
@prefix person: <http://example/person/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

person :a foaf:name "Alice" .
person :a foaf:nick "A-online" .

person:b foaf:name "Bob" .
```

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ?nick
{ ?x foaf:name ?name .
  OPTIONAL { ?x foaf:nick ?nick }
}
```

```
-----
| name          | nick                                     |
=====
| "Alice"      | "A-online"                              |
| "Bob"        |                                           |
-----
```

## SPARQL : UNION

```
@prefix book: <http://example/book/> .
@prefix dc10: <http://purl.org/dc/elements/1.0/> .
@prefix dc11: <http://purl.org/dc/elements/1.1/> .

book:a dc10:title "SPARQL Query Language Tutorial" .
book:b dc11:title "SPARQL Query Language (2nd ed)" .
book:c dc10:title "SPARQL" .
book:c dc11:title "SPARQL" .
```

```
PREFIX dc10: <http://purl.org/dc/elements/1.0/>
PREFIX dc11: <http://purl.org/dc/elements/1.1/>

SELECT DISTINCT ?title
{
  { ?book dc10:title ?title } UNION { ?book dc11:title ?title }
}
```

```
-----|
| title |
|-----|
| "SPARQL Query Language Tutorial" |
| "SPARQL" |
| "SPARQL Query Language (2nd ed)" |
|-----|
```

9

## Solution Modifiers

- After matching, the set of solutions is turned into a sequence then:
  - ORDER BY
  - Project
  - DISTINCT, REDUCED
  - OFFSET
  - LIMIT

10

## Result Sets

```
<sparql xmlns="http://www.w3.org/2005/sparql-results#">
  <head>
    <variable name="name"/>
    <variable name="mbox"/>
  </head>

  <results ordered="false" distinct="false">
    <result>
      <binding name="name"><literal>Johnny Lee Outlaw</literal></binding>
      <binding name="mbox"><uri>mailto:jlow@example.com</uri></binding>
    </result>

    <result>
      <binding name="mbox"><uri>mailto:peter@example.org</uri></binding>
    </result>
  </results>
</sparql>
```

name	mbox
"Johnny Lee Outlaw"	<mailto:jlow@example.com>
	<mailto:peter@example.org>

## Inference

- An RDF graph may be backed by inference
  - OWL, RDFS, application, rules

```
:x rdf:type :C .
:C rdfs:subClassOf :D .
```

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT ?type
WHERE
{
  ?x rdf:type ?type .
}
```

type
:C
:D

# CONSTRUCT

```
@prefix person: <http://example/person/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .

person:a foaf:name "Alice" .
person:a foaf:mbox <mailto:alice@example.net> .
person:b foaf:name "Bob" .
```

```
PREFIX person: <http://example/person/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX vcard: <http://www.w3.org/2001/vcard-rdf/3.0#>

CONSTRUCT { ?person vcard:FN ?name }
WHERE
  {?person foaf:name ?name . }
```

```
@prefix person: <http://example/person/> .
@prefix vcard: <http://www.w3.org/2001/vcard-rdf/3.0#>

person:a vcard:FN "Alice" .
person:b vcard:FN "Bob" .
```

# SPARQL : RDF Dataset

- RDF Dataset – collection of graphs
  - One, unnamed default graph ;
  - Zero or more named graphs
- Access with the GRAPH keyword

```
SELECT . . .
FROM <contact.ttl>
FROM NAMED <aliceFoaf.ttl>
FROM NAMED <bobFoaf.ttl>
WHERE { . . . }
```

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?graph ?name
WHERE
{
  ?alice foaf:name "Alice" .
  ?alice foaf:mbox ?mbox .
  GRAPH ?graph
  {
    ?x foaf:mbox ?mbox .
    ?x foaf:knows ?person .
    ?person foaf:name ?name .
  }
}
```