SPARQL Query Language

Basics

Mark S. Fox PhD FAAAI FIEEE FEIC LEL
Distinguished Professor of Urban Systems Engineering
Professor of Industrial Engineering and Computer Science
Associate Director (Research), School of Cities
University of Toronto

Outline (Slides based on F. Freitas, Cin/UFPE, Brazil)

1. **Basic queries**
2. Constraints
3. Aggregation
4. Graphs
5. Query Forms
6. SPARQL in Fuseki

Readings

- Prud'hommeaux, E., and Seaborne, A., (2008), "SPARQL Query Language for RDF", http://www.w3.org/TR/rdf-sparql-query/
- Feigenbaum, L., (2009), "SPARQL by Example: A Tutorial", https://www.w3.org/2009/Talks/0615-qbe/
- SPARQL Validator: http://sparql.org/query-validator.html

SPARQL

- **SPARQL** is a **recursive** acronym standing for
  - **S**PARQL **P**rotocol **a**nd **R**DF **Q**uery **L**anguage
- It provides facilities to:
  - extract information in the form of IRIs, blank nodes, plain and typed literals.
  - extract RDF subgraphs.
  - construct new RDF graphs based on information in the queried graphs

Components of a Query

Query

```
@prefix foaf: <http://xmlns.com/foaf/0.1/>
@prefix rel:  <http://purl.org/vocab/relationship/>
@prefix uoft: <http://ontology.eil.utoronto.ca/MIE1501/foaf_example#>
```

```
SELECT ? x ?gname
FROM <http://endpoint.domainname:port>
WHERE   { ?x  foaf:givenName  ?gname }
ORDER BY ?gname
```

Prologue:

- Prefix definitions

Result form specification:

- SELECT, DESCRIBE, CONSTRUCT, or ASK
- SELECT: - Variable list or asterisk (“*”) character for all - DISTINCT for disjoint results

Dataset specification:

- Specify the dataset to be queried
- Use FROM and FROM NAMED clauses (each with a URI)

Query Pattern:

- WHERE clause specifies the graph pattern to be matched

Solution modifiers:

Modify the result set, but not the single results

- ORDER BY, LIMIT, or OFFSET

Basic Query: Select

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rel:  <http://purl.org/vocab/relationship/> .
@prefix uoft: <http://ontology.eil.utoronto.ca/MIE1501/
        foaf_example#> .

uoft:js rdfs:type foaf:Person .
uoft:js foaf:givenName "Joe" .
uoft:js foaf:familyName "Smith" .
uoft:js foaf:age 20 .
uoft:js rel:friendOf uoft:fj.

uoft:fj rdfs:type foaf:Person ;
      foaf:givenName "Fred" ;
      foaf:familyName "Jones" ;
      foaf:age 20 ;
      rel:worksWith uoft:js .

uoft:mj rdfs:type foaf:Person ;
      foaf:givenName "Mary" ;
      rel:spouseOf uoft:fj ;
      rel:friendOf [ rel:employedBy uoft:js ] .
```

Query

```
@prefix foaf: <http://xmlns.com/foaf/0.1/>
@prefix rel:  <http://purl.org/vocab/relationship/>
@prefix uoft: <http://ontology.eil.utoronto.ca/MIE1501/foaf_example#>
SELECT ? x ?gname
WHERE   { ?x  foaf:givenName  ?gname }
```

Basic Query: Select

	Subject	Property	Value
(1)	uoft:js	foaf:givenName	"Joe"
	uoft:js	foaf:familyName	"Smith"
	uoft:js	foaf:age	20
	uoft:js	rel:friendOf	uoft:fj
(2)	uoft:fj	foaf:givenName	"Fred"
	uoft:fj	foaf:familyName	"Jones"
	uoft:fj	foaf:age	22
	uoft:fj	rel:worksWith	uoft:js
(3)	uoft:mj	foaf:givenName	"Mary"
	uoft:mj	rel:spouseOf	uoft:fj
	uoft:mj	rel:freindOf	_:a
	_:a	rel:employedBy	uoft:js

Query

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

@prefix rel: <http://purl.org/vocab/relationship/>

@prefix uoft: <http://ontology.eil.utoronto.ca/MIE1501/foaf_example#>

SELECT ? x ?gname

WHERE { ?x foaf:givenName ?gname }

Result

x	gname
uoft:js	Joe
uoft:fj	Fred
uoft:mj	Mary

Compound Select

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rel:  <http://purl.org/vocab/relationship/> .
@prefix uoft: <http://ontology.eil.utoronto.ca/MIE1501/
        foaf_example#> .

uoft:js rdfs:type foaf:Person .
uoft:js foaf:givenName "Joe" .
uoft:js foaf:familyName "Smith" .
uoft:js foaf:age 20 .
uoft:js rel:friendOf uoft:fj.

uoft:fj rdfs:type foaf:Person ;
      foaf:givenName "Fred" ;
      foaf:familyName "Jones" ;
      foaf:age 20 ;
      rel:worksWith uoft:js .

uoft:mj rdfs:type foaf:Person ;
      foaf:givenName "Mary" ;
      rel:spouseOf uoft:fj ;
      rel:friendOf [ rel:employedBy uoft:js ] .
```

Query

```
@prefix foaf: <http://xmlns.com/foaf/0.1/>
@prefix rel:  <http://purl.org/vocab/relationship/>
@prefix uoft: <http://ontology.eil.utoronto.ca/MIE1501/foaf_example#>
SELECT  ?gname ?fname
WHERE   { ?x  foaf:givenName  ?gname ;
              foaf:familyName ?fname }
```

Compound Select

	Subject	Property	Value
(1)	uoft:js	foaf:givenName	"Joe"
(2)	uoft:js	foaf:familyName	"Smith"
	uoft:js	foaf:age	20
	uoft:js	rel:friendOf	uoft:fj
(3)	uoft:fj	foaf:givenName	"Fred"
(4)	uoft:fj	foaf:familyName	"Jones"
	uoft:fj	foaf:age	22
	uoft:fj	rel:worksWith	uoft:js
(5)	uoft:mj	foaf:givenName	"Mary"
	uoft:mj	rel:spouseOf	uoft:fj
	uoft:mj	rel:freindOf	_:a
	_:a	rel:employedBy	uoft:js

Query

```
@prefix foaf: <http://xmlns.com/foaf/0.1/>
@prefix rel:  <http://purl.org/vocab/relationship/>
@prefix uoft: <http://ontology.eil.utoronto.ca/
                                MIE1501/foaf_example#>
SELECT  ?gname ?fname
WHERE   { ?x  foaf:givenName  ?gname ;
              foaf:familyName ?fname }
```

Result

gname	fname
Joe	Smith
Fred	Jones

What happened to Mary?

Linked/Join Select

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rel:  <http://purl.org/vocab/relationship/> .
@prefix uoft: <http://ontology.eil.utoronto.ca/MIE1501/
        foaf_example#> .

uoft:js rdfs:type foaf:Person .
uoft:js foaf:givenName "Joe" .
uoft:js foaf:familyName "Smith" .
uoft:js foaf:age 20 .
uoft:js rel:friendOf uoft:fj.

uoft:fj rdfs:type foaf:Person ;
      foaf:givenName "Fred" ;
      foaf:familyName "Jones" ;
      foaf:age 20 ;
      rel:worksWith uoft:js .

uoft:mj rdfs:type foaf:Person ;
      foaf:givenName "Mary" ;
      rel:spouseOf uoft:fj ;
      rel:friendOf [ rel:employedBy uoft:js ] .
```

Query

```
@prefix foaf: <http://xmlns.com/foaf/0.1/>
@prefix rel:  <http://purl.org/vocab/relationship/>
@prefix uoft: <http://ontology.eil.utoronto.ca/
                        MIE1501/foaf_example#>
SELECT  ?gname1 ?gname2
WHERE   { ?x rel:friendOf ?y ;
              foaf:givenName ?gname1 .
           ?y  foaf:givenName  ?gname2 }
```

Linked/Join Select

	Subject	Property	Value
(2)	uoft:js	foaf:givenName	"Joe"
	uoft:js	foaf:familyName	"Smith"
	uoft:js	foaf:age	20
(1)	uoft:js	rel:friendOf	uoft:fj
(3)	uoft:fj	foaf:givenName	"Fred"
	uoft:fj	foaf:familyName	"Jones"
	uoft:fj	foaf:age	22
	uoft:fj	rel:worksWith	uoft:js
	uoft:mj	foaf:givenName	"Mary"
	uoft:mj	rel:spouseOf	uoft:fj
	uoft:mj	rel:freindOf	_:a
	_:a	rel:employedBy	uoft:js

Query

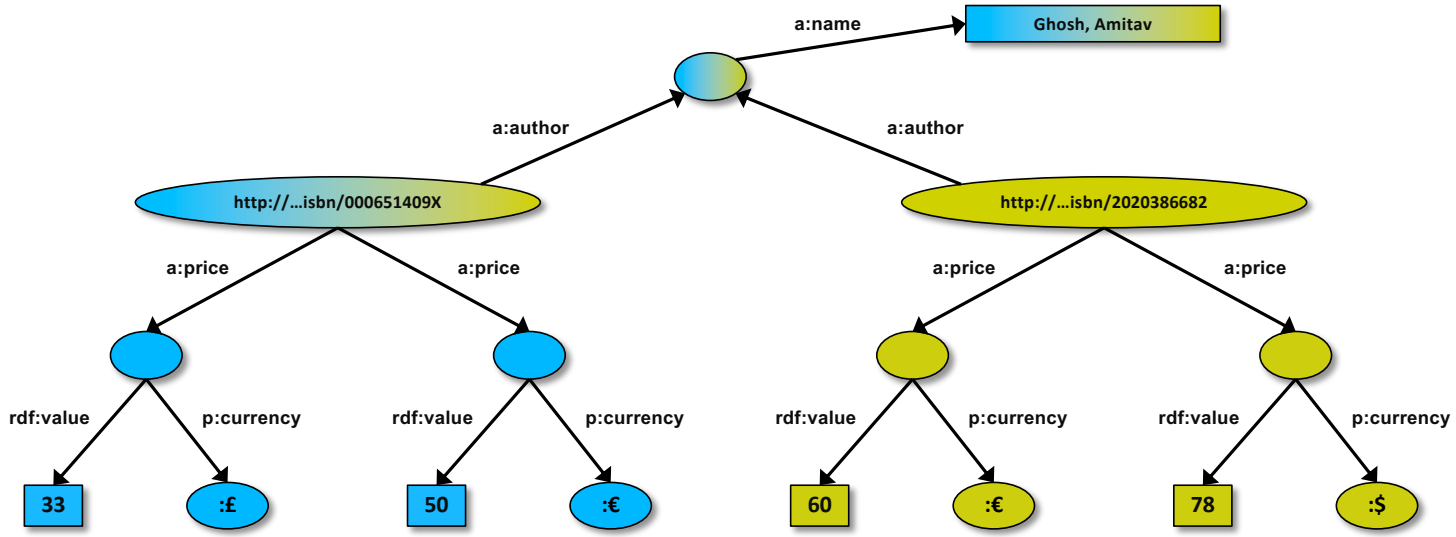
```
@prefix foaf: <http://xmlns.com/foaf/0.1/>
@prefix rel:  <http://purl.org/vocab/relationship/>
@prefix uoft: <http://ontology.eil.utoronto.ca/
                                MIE1501/foaf_example#>
SELECT  ?gname1 ?gname2
WHERE   { ?x rel:friendOf ?y ;
                foaf:givenName ?gname1 .
          ?y  foaf:givenName  ?gname2 }
```

Result

gname1	gname2
Joe	Fred

Book Example: Graph Perspective

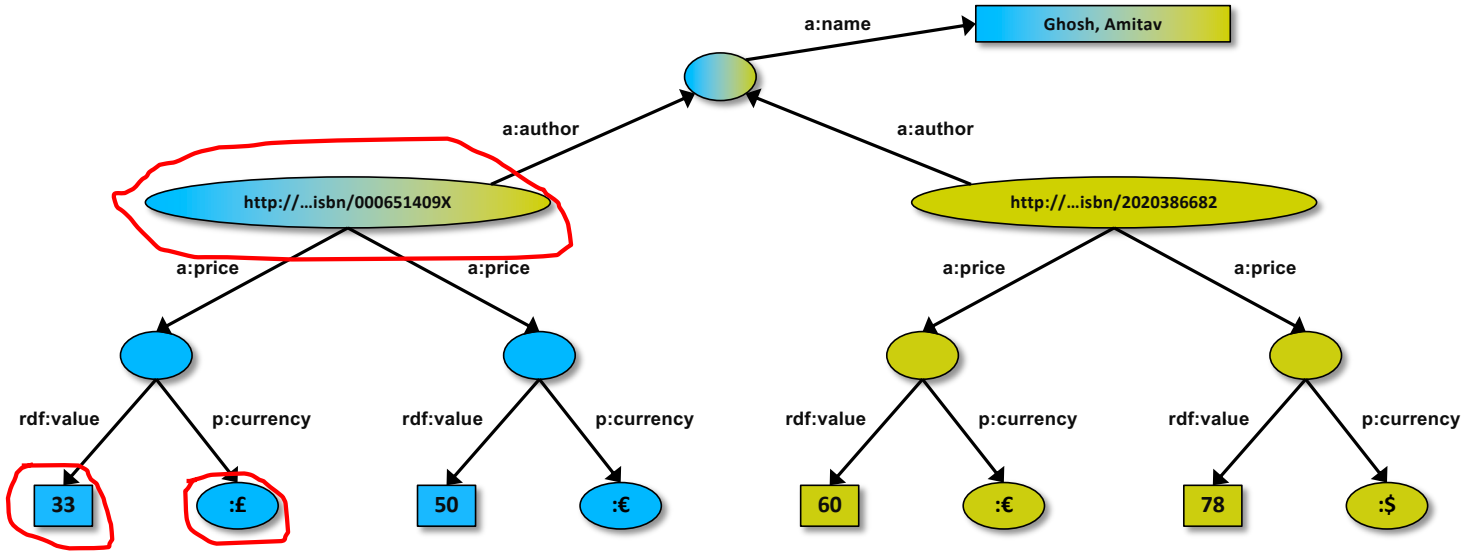
```
SELECT ?isbn ?price ?currency # note: not ?x!
WHERE {?isbn a:price ?x. ?x rdf:value ?price. ?x p:currency ?currency }
```



Graph Perspective

```
SELECT ?isbn ?price ?currency # note: not ?x!
WHERE {?isbn a:price ?x. ?x rdf:value ?price. ?x p:currency ?currency }
```

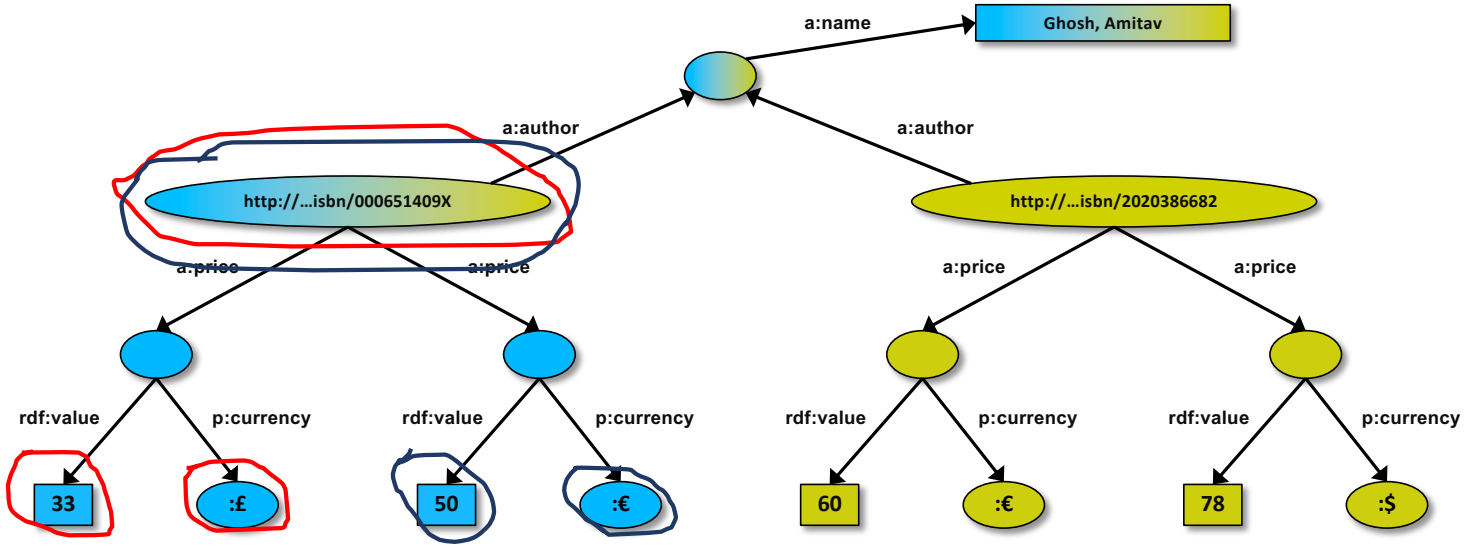
Returns: [<…409X>,33,:£]



Graph Perspective

```
SELECT ?isbn ?price ?currency # note: not ?x!
WHERE {?isbn a:price ?x. ?x rdf:value ?price. ?x p:currency ?currency }
```

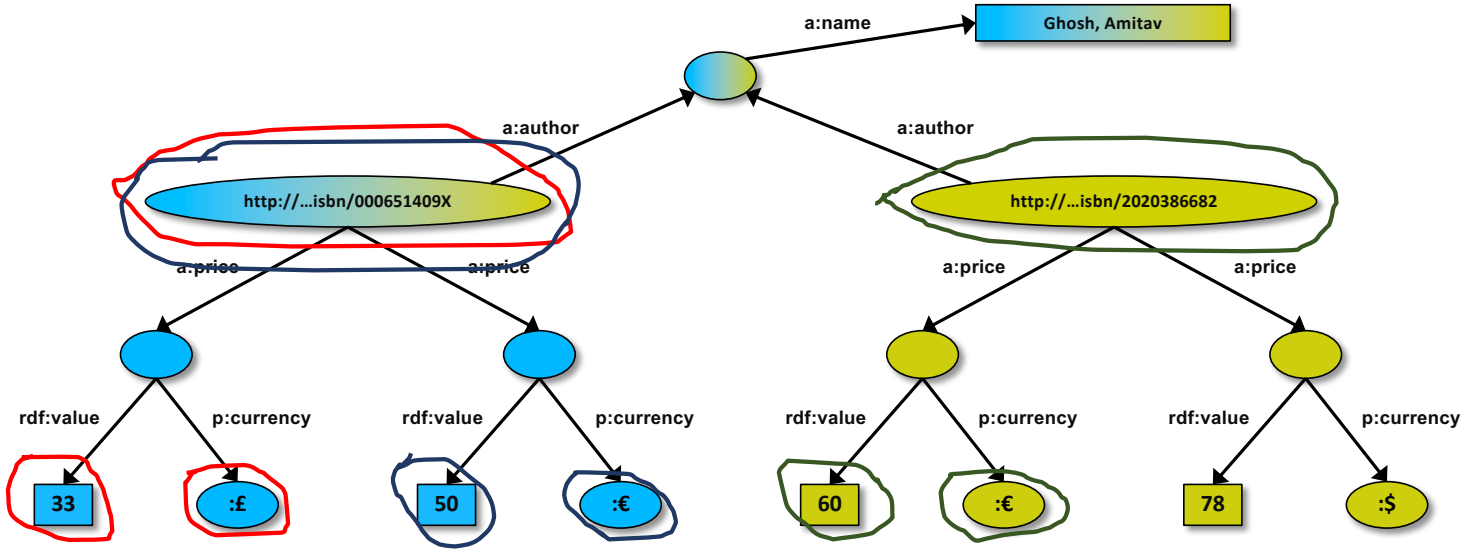
Returns: [<…409X>,33,:£], [<…409X>,50,:€]



Graph Perspective

```
SELECT ?isbn ?price ?currency # note: not ?x!
WHERE {?isbn a:price ?x. ?x rdf:value ?price. ?x p:currency ?currency }
```

Returns: [<…409X>,33,:£], [<…409X>,50,:€],
[<…6682>,60,:€]



Graph Perspective

```
SELECT ?isbn ?price ?currency # note: not ?x!
WHERE {?isbn a:price ?x. ?x rdf:value ?price. ?x p:currency ?currency }
```

Returns: [<…409X>,33,:£], [<…409X>,50,:€],
[<…6682>,60,:€], [<…6682>,78,:$]