Web Services and Web Data

XJCO3011

Session 17 – Querying Linked Data with SPARQL

SPARQL (SPARQL Protocol and RDF Query Language)

- SPARQL is a recursive acronym for *SPARQL Protocol and RDF Query Lang*uage
- SPARQL is to RDF data as SQL is to relational databases. SPARQL is the query language for structured data on the Web, i.e., data accessible in RDF formats or representable as such.
- With SPARQL, we can query the Web of Data as if it were a database—a big, highly distributed database on the internet. SPARQL can query local files containing RDF data, or RDF files accessible on the Web. It is also able to query multiple data sources at once and thus dynamically build a large, virtual RDF graph from those multiple data sources.
- Because many people are already familiar with SQL, SPARQL was designed to look and act as much like SQL as possible, even though the traditional relational data model differs significantly from the graph data model of RDF.
- Like SQL, SPARQL is based on a widely implemented standard, but various vendors have extended the language.

Querying Flat RDF Files with SPARQL

- The select query in the following listing looks for people that the owner of a FOAF document knows.
- This query will return some number of people and their URLs.
- Anyone listed in the FOAF file with an rdfs:seeAlso URL and a foaf:name will be returned.

Listing 5.2 SPARQL query to find FOAF friends

```
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix foaf: <http://xmlns.com/foaf/0.1/>

select ?name ?url
where {
  ?person rdfs:seeAlso ?url ;
          foaf:name ?name .
}
```

Namespace declarations

Names information to return in the results

Defines patterns to match and filters to perform on the data

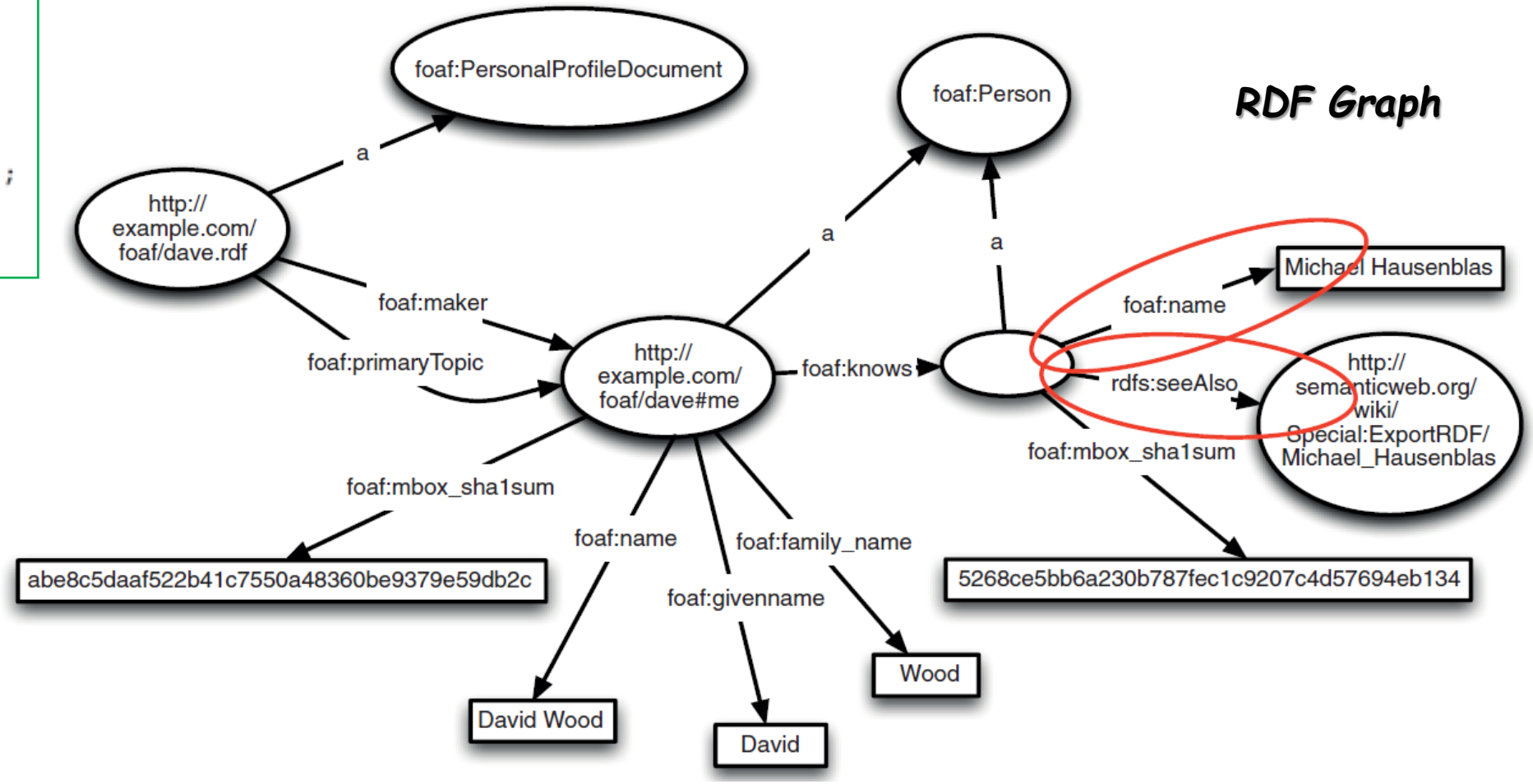
Triple patterns used to match RDF statements

Select Query in Action

Query

```
select ?name ?url
where {
  ?person rdfs:seeAlso ?url ;
          foaf:name ?name .
}
```

RDF Graph



Result

name	url
"Michael Hausenblas"	<http://semanticweb.org/wiki/Special:ExportRDF/Michael_Hausenblas>

Querying Multiple RDF Files

- Unlike SQL, SPARQL isn't limited to querying a single data source. You can use SPARQL to query multiple files, web resources, databases, or a combination thereof.
- For example, we can the personal information in a FOAF profile can be extended with address information (which can be represented in RDF via the vCard vocabulary).
- This shows that RDF files may be combined, just like any other RDF graphs. Graphs of information combine well (unlike tables and trees). The magic is in the reuse of identifiers. Both files refer to the same URI identifying a person.

Listing 5.6 A SPARQL query that combines FOAF and vCard data

```
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix vcard: <http://www.w3.org/2006/vcard/ns#>

SELECT ?name ?city ?state
where {
  ?person foaf:name ?name ;
          vcard:adr ?address .
  ?address vcard:locality ?city ;
          vcard:region ?state .
}
```

The SELECT clause, showing three variable bindings to be returned in the results

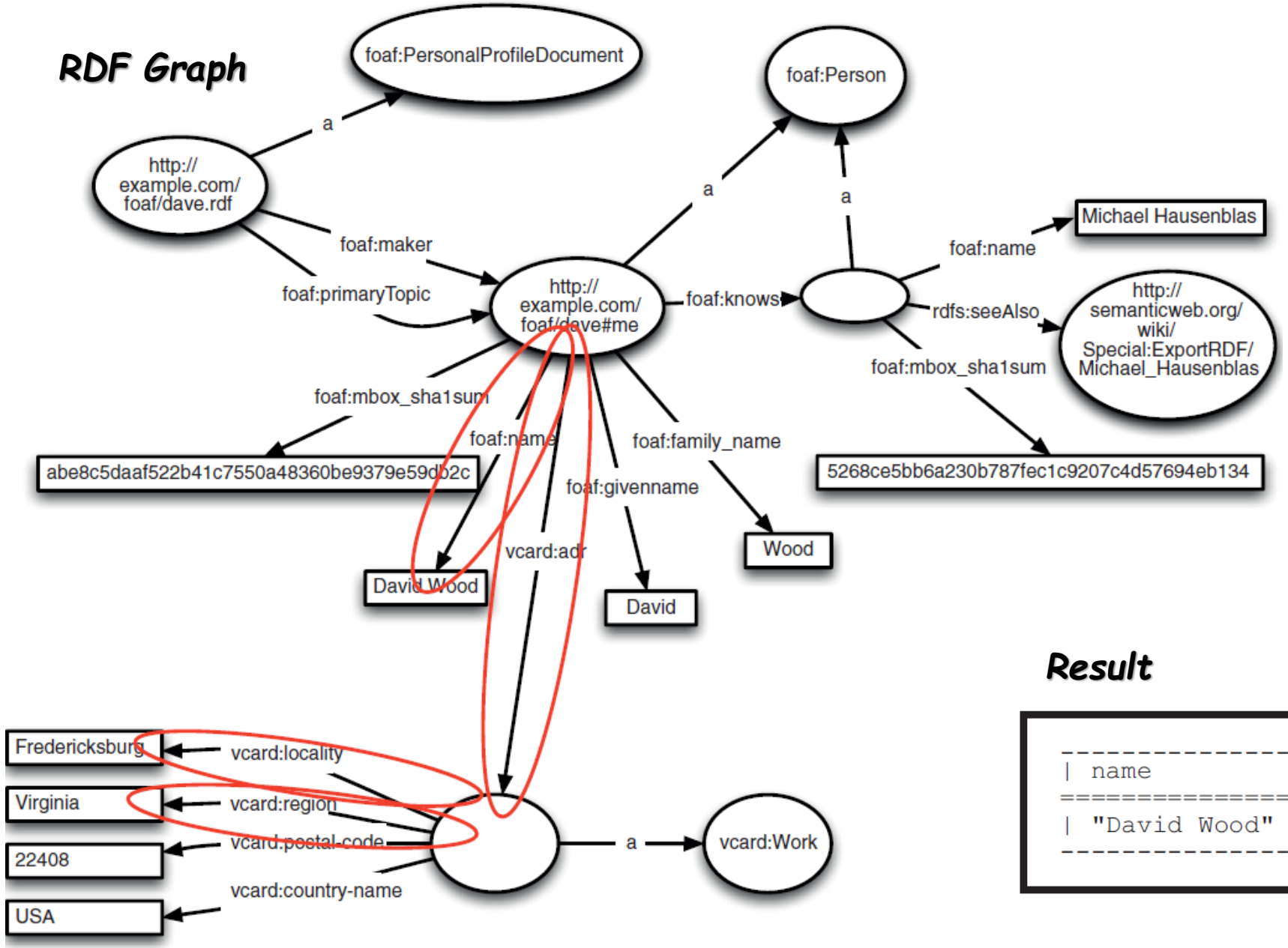
A triple pattern to find the blank node representing an address

Triple patterns mapping the address to a city and state

NOTE One of the primary assumptions of Linked Data is that two people using the same identifier are talking about the same thing. Reusing identifiers for resources allows data to be combined.

Querying Multiple RDF Files in Action

RDF Graph



Result

name	city	state
"David Wood"	"Fredericksburg"	"Virginia"

SQL vs SPARQL

Developers used to SQL might note that variable names in SPARQL's `SELECT` clause don't name variables to query from the database; they determine which variables used in the `WHERE` clause's triple patterns get returned in the output. That's confusing for some new users, but it makes sense once you wrap your mind around the concept of matching triple patterns against an RDF graph.

Listing 5.6 A SPARQL query that combines FOAF and vCard data

```
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix vcard: <http://www.w3.org/2006/vcard/ns#>

SELECT ?name ?city ?state
where {
  ?person foaf:name ?name ;
          vcard:adr ?address .
  ?address vcard:locality ?city ;
          vcard:region ?state .
}
```

The SELECT clause, showing three variable bindings to be returned in the results

A triple pattern to find the blank node representing an address

Triple patterns mapping the address to a city and state

Example of Querying Multiple RDF Files

```
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix pos: <http://www.w3.org/2003/01/geo/wgs84_pos#>

select ?name ?latitude ?longitude
from <http://3roundstones.com/dave/me.rdf>
from <http://semanticweb.org/wiki/Special:ExportRDF/Michael_Hausenblas>
where {
  ?person foaf:name ?name ;
          foaf:based_near ?near .
  ?near pos:lat ?latitude ;
        pos:long ?longitude .
}
LIMIT 10
```

Namespace prefixes.

Requesting three fields be retrieved.

Results will be retrieved from two sources.[1]

Criteria described in the form of a triple pattern.

Items preceded by ? represent variables in the results.

Only the first 10 results will be returned.

Querying an RDF file on the Web

- This can be done by adding a FROM clause after the SELECT clause, e.g.

  FROM http://3roundstones.com/dave/me.rdf

- One of the things that makes structured data on the Web interesting is that it's distributed, unlike a relational database where the data exists in a single system.

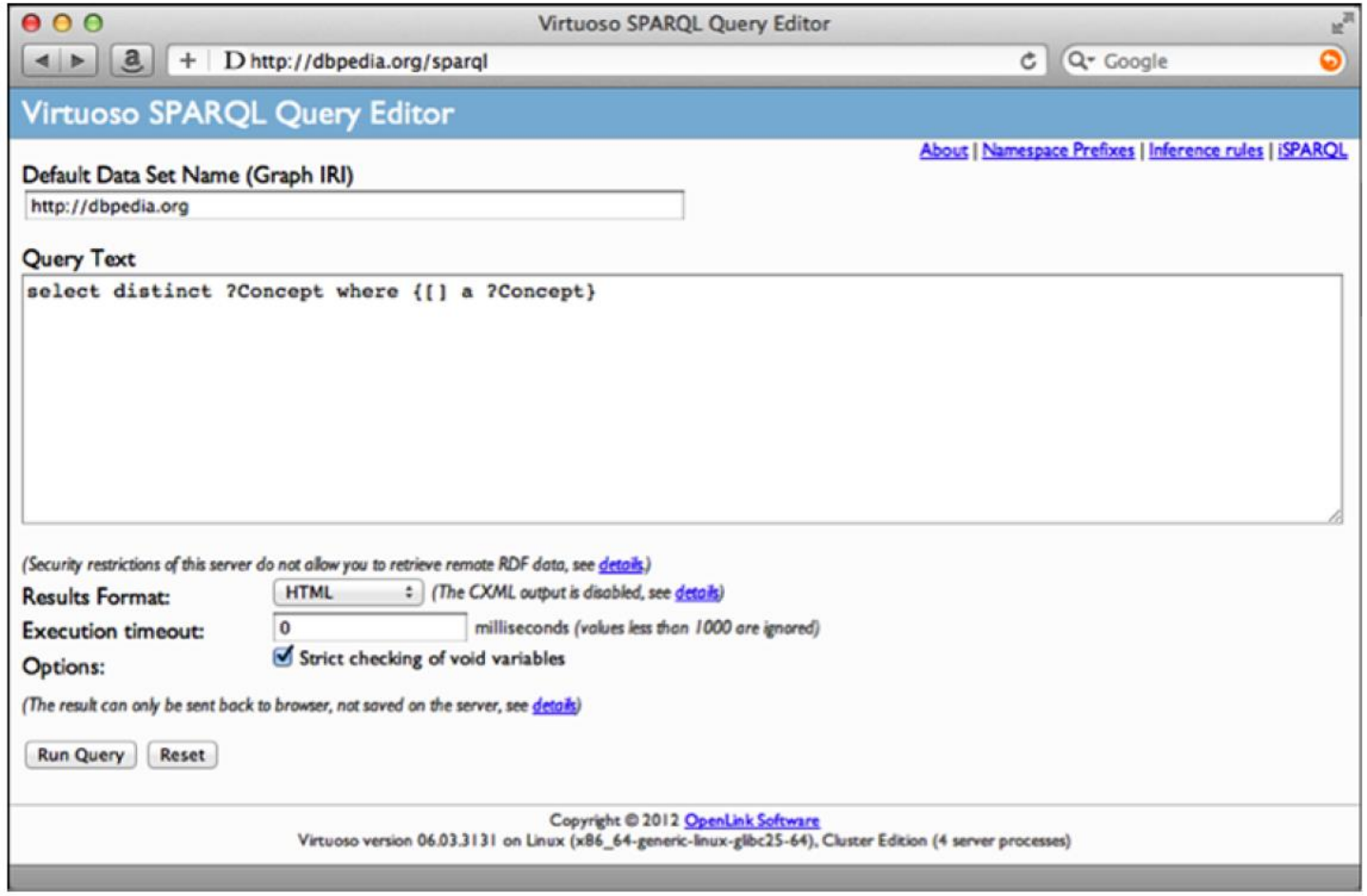
- SPARQL allows you to have multiple FROM clauses in a single query.

- You can add multiple FROM clauses with different URLs to a query

Querying SPARQL Endpoints

- Linked Data sites on the Web often expose a SPARQL endpoint.
- A SPARQL endpoint is a web-accessible query service that accepts the SPARQL query language.
- An HTTP GET on a SPARQL endpoint generally returns an HTML query form.

NOTE As is true in Turtle, the syntactical convenience `a`, when used as a property, is the same as saying `rdf:type`, which is used to say that an RDF resource is an instance of a particular RDF class. The term `[]` is a blank node and will therefore match any subject.



SPARQL Endpoints

- The growing convention used by datasets on the Linked Open Data cloud is exposing SPARQL endpoints on the path /sparql. You can generally determine whether a given Linked Data site has a SPARQL endpoint by constructing a URL like http://{hostname here}/sparql. This is just a convention, but it's a useful one. Of course, you can put a SPARQL endpoint on any URL.
- DBpedia's default query gives a hint to new users on how they can discover what information the service holds. You can rewrite DBpedia's default query with more whitespace to make it more readable, as shown in the following listing.

Listing 5.8 Query the `rdf:types` a server holds

```
select DISTINCT ?Concept
WHERE {
   [] a ?Concept
}
```

The DISTINCT keyword ensures that duplicate results are filtered out; only unique matching results are returned.

That's it Folks

Further Reading

Chapter 5: Linked Data Structured Data on the Web, David Wood et al.