

SPARQL Queries and SPARQL Technology

Bob DuCharme
Charlottesville Semantic Web Meetup

October 20, 2011





Introductions

- Presentation and all its URLs:
<http://snee.com/semwebmeetup/2011-10-20>
- Me: SGML and XML at Moody's, LexisNexis, Innodata Isogen, TopQuadrant
- Weblog: <http://www.snee.com/bobdc.blog>
- Twitter: @bobdc



Learning SPARQL: The Book

learningsparql.com
@learningsparql





Outline

- SPARQL and the Semantic Web
- Simple queries of local data and remote data
- SPARQL 1.1
- SPARQL tools, SPARQL's use in applications
- Questions



What is SPARQL?

- **SPARQL Protocol and RDF Query Language**
- “Trying to use the Semantic Web without SPARQL is like trying to use a relational database without SQL” – Tim Berners-Lee



The Semantic Web

A set of standards and best practices for sharing data and the semantics of that data over the web for use by applications.



The Semantic Web

A set of standards and best practices for sharing data and the semantics of that data over the web for use by applications.

- RDF
- SPARQL



The Semantic Web

A set of standards and **best practices for sharing data** and the semantics of that data **over the web for use by applications.**

- Linked Data



- Resource Description Framework
- Store data about anything, but especially metadata about resources
- Stored where?
- Very easily aggregated



An RDF “statement”: the triple

- (Subject, predicate, object)
- “index.html has the title ‘My Home Page’.”
- Easily stores (resource ID, propertyName, propertyValue) assertions



Triples

```
# rdf1.nt: sample RDF file in n-triples format.
```

```
<http://www.snee.com/bob/index.html>
```

```
<http://purl.org/dc/elements/1.1/title>
```

```
"My Home Page".
```

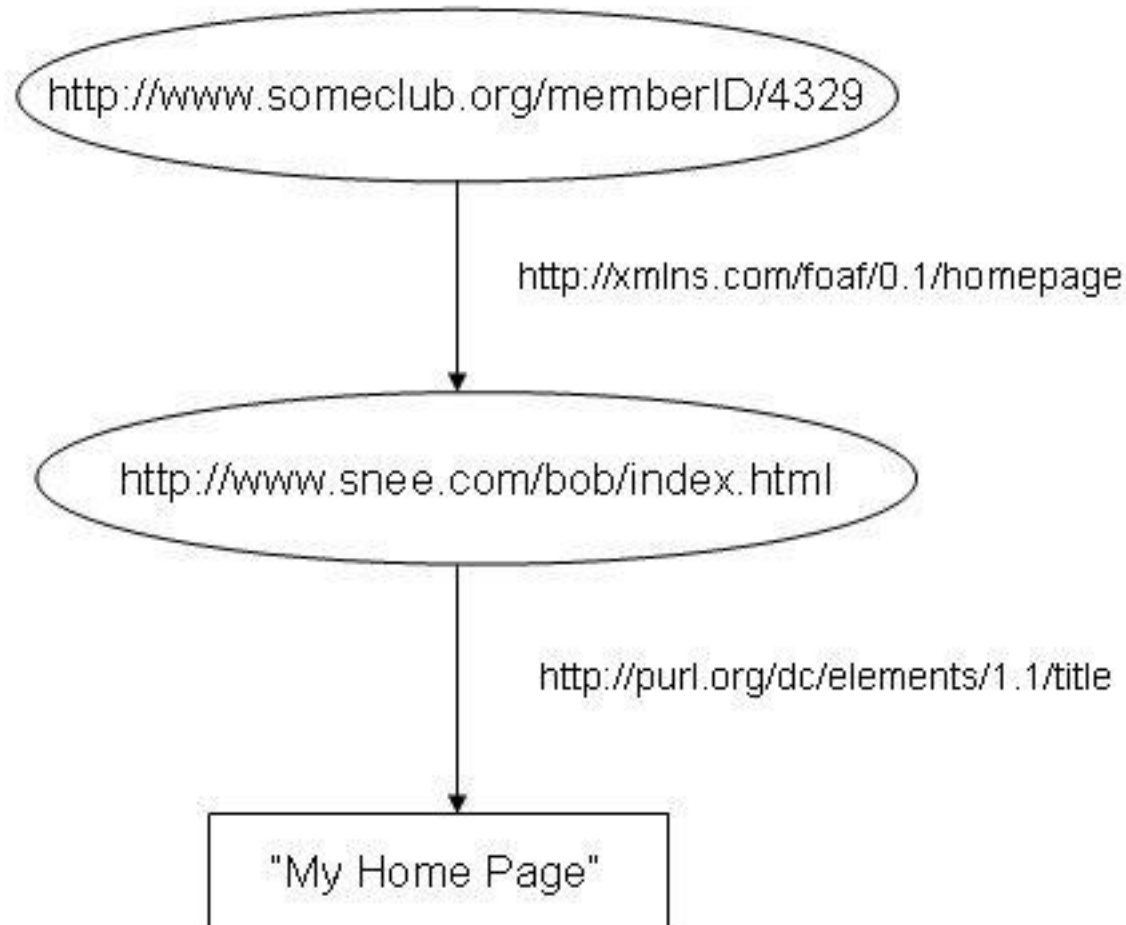
```
<http://www.someclub.org/memberID/4329>
```

```
<http://xmlns.com/foaf/0.1/homepage>
```

```
<http://www.snee.com/bob/index.html>.
```



Linking triples into a “graph”





SPARQL specs

- 1.0 became W3C spec January 2008:
 - SPARQL Query Language for RDF
 - SPARQL Protocol for RDF
 - SPARQL Query Results XML Format
- 1.1 still in Working Draft status, adds:
 - UPDATE
 - Several smaller documents that are being gradually rolled into the ones above



Data to query

```
# filename: ex2.ttl
@prefix ab: <http://learningsparql.com/ns/demo#> .

ab:richard ab:homeTel "(229) 276-5135" .
ab:richard ab:email "richard49@hotmail.com" .

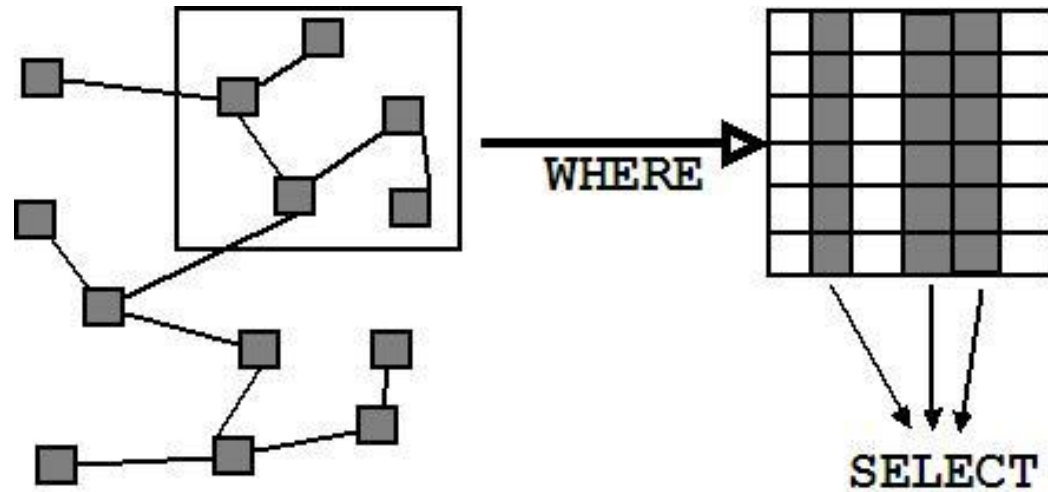
ab:cindy ab:homeTel "(245) 646-5488" .
ab:cindy ab:email "cindym@gmail.com" .

ab:craig ab:homeTel "(194) 966-1505" .
ab:craig ab:email "craigellis@yahoo.com" .
ab:craig ab:email "c.ellis@usairwaysgroup.com" .
```

Our first query

```
# filename: ex2b.rq
PREFIX ab: <http://learningsparql.com/ns/demo#>

SELECT ?craigEmail
WHERE {
  ab:craig ab:email ?craigEmail .
}
```





Running it with ARQ

```
arq --data ex2a.ttl --query ex2b.rq
```

Result:

```
-----  
| craigEmail |  
=====
```

"c.ellis@usairwaysgroup.com"
"craigellis@yahoo.com"

```
-----
```




More realistic data

```
@prefix ab: <http://learningsparql.com/ns/demo#> .
```

```
ab:i0432 ab:firstName "Richard" .
```

```
ab:i0432 ab:lastName "Mutt" .
```

```
ab:i0432 ab:homeTel "(229) 276-5135" .
```

```
ab:i0432 ab:email "richard49@hotmail.com" .
```

```
ab:i9771 ab:firstName "Cindy" .
```

```
ab:i9771 ab:lastName "Marshall" .
```

```
ab:i9771 ab:homeTel "(245) 646-5488" .
```

```
ab:i9771 ab:email "cindym@gmail.com" .
```

```
ab:i8301 ab:firstName "Craig" .
```

```
ab:i8301 ab:lastName "Ellis" .
```

```
ab:i8301 ab:email "craigellis@yahoo.com" .
```

```
ab:i8301 ab:email "c.ellis@usairwaysgroup.com" .
```



Finding Craig's email address

```
PREFIX ab: <http://learningsparql.com/ns/demo#>
```

```
SELECT ?craigEmail
```

```
WHERE {
```

```
    ?person ab:firstName "Craig" .
```

```
    ?person ab:email ?craigEmail .
```

```
}
```

Answer:

```
-----  
| craigEmail |  
=====
```

"c.ellis@usairwaysgroup.com"
"craigellis@yahoo.com"

```
-----
```



Finding Craig Ellis's email address

Assume ab: prefix is declared in remaining examples

```
SELECT ?craigEmail
WHERE {
    ?person ab:firstName "Craig" .
    ?person ab:lastName "Ellis" .
    ?person ab:email ?craigEmail .
}
```

Or...

```
SELECT ?craigEmail
WHERE {
    ?person ab:firstName "Craig" ;
            ab:lastName "Ellis" ;
            ab:email ?craigEmail .
}
```



Who just called me?

```
SELECT ?person
WHERE {
  ?person ab:homeTel "(229) 276-5135" .
}
```

Answer:

```
-----
| person      |
=====
| ab:i0432   |
-----
```



Yeah, but what's their name?

```
SELECT ?first ?last
WHERE {
    ?person ab:homeTel "(229) 276-5135" .
    ?person ab:firstName ?first .
    ?person ab:lastName ?last .
}
```

Answer:

first	last
"Richard"	"Mutt"



Other options

- FILTER: results that meet a certain condition
- UNION: specify two or more sets of data to pull at once
- ORDER BY ?variableName: sort output
- LIMIT
- DISTINCT: don't show duplicates
- Data types
- Functions



Creating triples with SPARQL (part 1)

```
PREFIX m: <http://www.example.com/ns/mytestvocab#>

SELECT ?child ?dad
WHERE {
    ?child m:parent ?dad .
    ?dad   m:gender m:male .
}
```




Creating triples with SPARQL (part 2)

```
PREFIX m: <http://www.example.com/ns/mytestvocab#>

CONSTRUCT { ?child m:father ?dad }
WHERE {
    ?child m:parent ?dad .
    ?dad   m:gender m:male .
}
```

Creating new data with CONSTRUCT

```
PREFIX ab: <http://learningsparql.com/demo#>
```

```
PREFIX v: <http://www.w3.org/2001/vcard-rdf/3.0#>
```

```
CONSTRUCT {  
    ?s v:given-name ?firstName ;  
        v:family-name ?lastName ;  
        v:homeTel ?homeTel .  
}  
  
WHERE {  
    ?s ab:firstName ?firstName ;  
        ab:lastName ?lastName .  
    OPTIONAL {  
        ?s ab:homeTel ?homeTel.  
    }  
}
```



SPARQL 1.1

- Still at least a few months to go
- Supported by ARQ and TopBraid
- Adding on to what we've seen so far:
 - Variable assignment
 - Aggregates and functions
 - Better negation
 - Subqueries and federated queries
- New: UPDATE



Variable assignment

```
SELECT ?tip
WHERE {
    ab:meal ab:bill ?bill .
    BIND (( ?bill * .2) AS ?tip ) .
}
```



Aggregates

```
PREFIX : <http://books.example/>
SELECT (SUM(?lprice) AS ?totalPrice)
WHERE {
    ?org :affiliates ?auth .
    ?auth :writesBook ?book .
    ?book :price ?lprice .
}
GROUP BY ?org
HAVING (SUM(?lprice) > 10)
```



Functions

- “SPARQL provides a subset of the functions and operators defined by XQuery [Operator Mapping](#). XQuery 1.0 section [2.2.3 Expression Processing](#) describes the invocation of XPath functions. ”
- More functions: sum, count, avg, min, max, regex, isURI, isNumeric, casting...
- More supported by ARQ: contains, starts-with, lower-case... see <http://jena.sourceforge.net/ARQ/library-function.html>



Negation in SPARQL 1.0

```
SELECT ?first ?last
WHERE {
    ?s ab:firstName ?first ;
       ab:lastName ?last .
    OPTIONAL {
        ?s ab:homeTel ?homeTel .
    }
    FILTER(!bound(?homeTel))
}
```



Negation in SPARQL 1.1

```
SELECT ?first ?last
WHERE {
    ?s ab:firstName ?first ;
       ab:lastName ?last .
MINUS {
    ?s ab:homeTel ?homeTel .
}
}
```




Subqueries and federation

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dbpo: <http://dbpedia.org/ontology/>
PREFIX imdb: <http://data.linkedmdb.org/resource/movie/>
PREFIX dcterms: <http://purl.org/dc/terms/>

SELECT ?birthDate ?spouseName ?movieTitle ?movieDate {
  { SERVICE <http://dbpedia.org/sparql> {
    ?actor rdfs:label "Arnold Schwarzenegger"@en ;
          dbpo:birthDate ?birthDate ;
          dbpo:spouse ?spouseURI .
    ?spouseURI rdfs:label ?spouseName .
    FILTER ( lang(?spouseName) = "en" )
  }
}
{ SERVICE <http://data.linkedmdb.org/sparql> {
  ?actorName imdb:actor_name "Arnold Schwarzenegger".
  ?movie imdb:actor ?actorName ;
         dcterms:title ?movieTitle ;
         dcterms:date ?movieDate .
}
}
}
```



SPARQL UPDATE

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

```
PREFIX ab: <http://learningsparql.com/ns/addressbook#>
```

```
DELETE
```

```
{ ?s ab:email ?o }
```

```
INSERT
```

```
{ ?s foaf:mbox ?o }
```

```
WHERE
```

```
{ ?s ab:email ?o }
```

- <http://esw.w3.org/SparqlImplementations>
- ARQ: command line tools, Jena Java library
- Libraries for other languages: Java, JavaScript, Python, Perl, PHP...
- Built into:
 - websites (SPARQL endpoints)
 - triplestores
 - TopBraid Composer
- SPARQL engine web service...



SPARQL web service



www.sparql.org/sparql.html



SPARQLer - General purpose processor

General SPARQL query : input query, set any options and press "Get Results"

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX air: <http://www.megginson.com/exp/ns/airports#>
SELECT ?personName ?airportCode WHERE {
  ?person foaf:name ?personName ;
          foaf:nearestAirport ?airport .
  ?airport air:iata ?airportCode .
}
```

Target graph URI (or use FROM in the query)

Output XML: with XSLT style sheet (leave blank for none):

or JSON output:

or text output:

or CSV output:

or TSV output:

Force the accept header to text/plain regardless



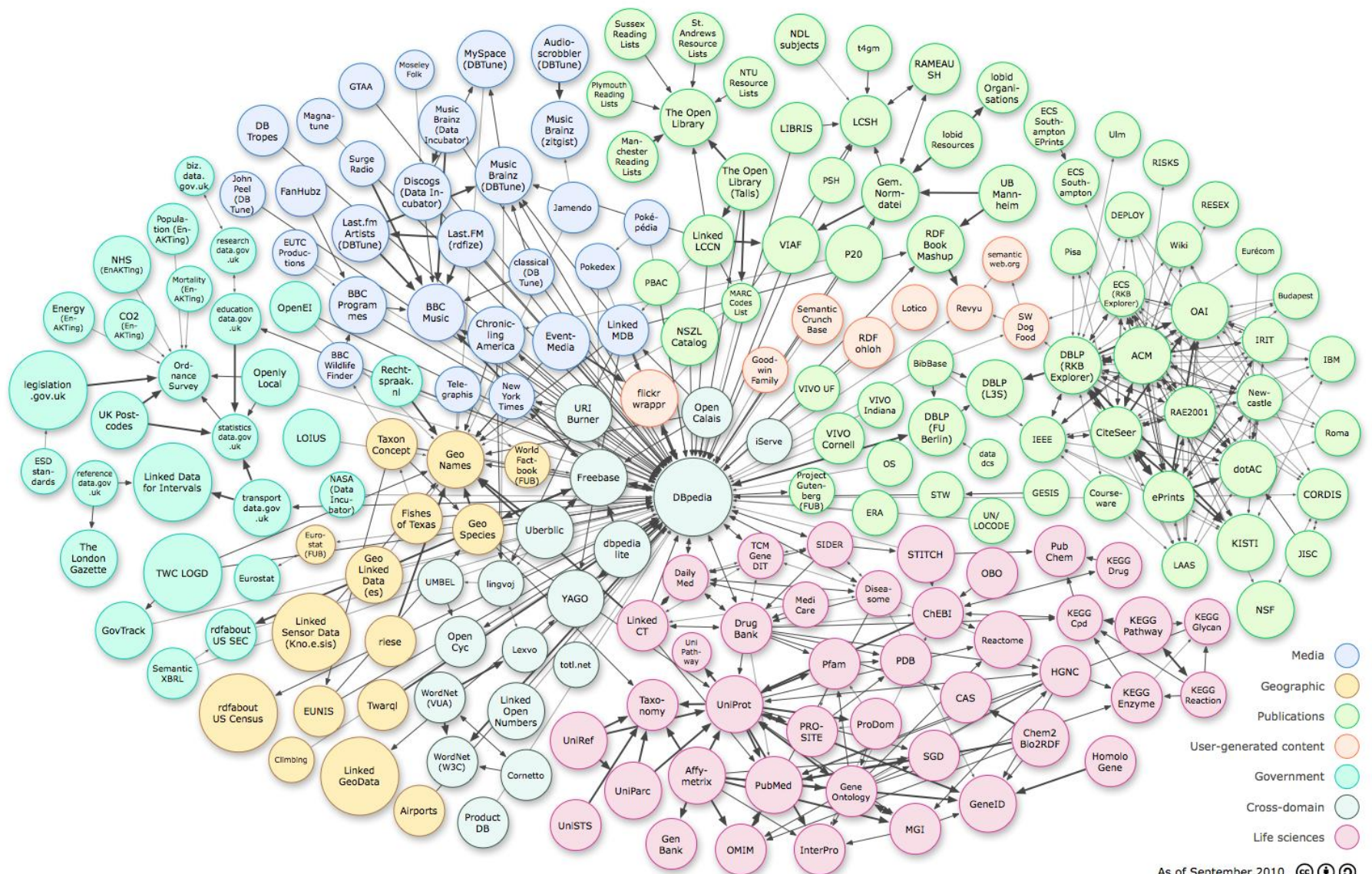
sparql.org query result

← → ↻ 🏠 www.sparql.org/sparql?query=PREFIX+foaf:+<http://xmlns.com/foaf/0.1/>%0D%0APREFIX+air:+<http://www.megginson.com/exp/ ☆ 🔍

SPARQLer Query Results

personName	airportCode
"Bob DuCharme"	"CHO"

Linked data cloud, color-coded





Is SPARQL Difficult?

File Edit View History Bookmarks Tools Help

http://www.cnn.com/2008/TECH/12/17/db.semanticweb/

Google

Making sense of the 'semantic Web' ...

digitalbiz IN ASSOCIATION WITH KONICA MINOLTA

updated 10:01 a.m. EST, Thu December 18, 2008

Making sense of the 'semantic Web'


STORY HIGHLIGHTS

- The "semantic web" could make the Internet more interactive and rewarding
- Things on the Web will be described in languages that computers can understand
- An experimental kiosk lets users see the potential of this Web 3.0 application
- Widespread adoption of the semantic Web is still a long way off

[Next Article in Technology »](#)

By Steve Mollman
For CNN

(CNN) -- The "semantic Web" does not sound like it's fun and easy to use, but it could make surfing Web 3.0 a more rewarding and interactive experience. Some believe it could even lead to a new form of artificial intelligence.




COMET
Spotlets pop up on a 'semantic Web' kiosk run by a pair of German researchers.

The idea behind the semantic Web, very broadly, is that things on the Internet will be described with descriptor languages so that computers can "understand" what they are.

An object might be marked as a car part or a person, for instance. If objects were thus identified, an enormous network of linked data would emerge and machines, with their vast processing speeds, could suggest surprising and useful links that the human mind could never come up with, posing the possibility of a new sort of artificial intelligence.

The semantic Web is considered a key part of the upcoming "Web 3.0." It's starting to occur here and there, but widespread adoption is still a long way off.



THIS WAY >

THE FUTURE IS HERE
Hot technology creates a whole new world.

A pair of German researchers have created an experimental kiosk that lets you easily use semantic Web capabilities -- even if you have no idea what they are. All that is needed is an iPhone and a finger with which to drag icons around on the kiosk's touch screen.

Find: [Next](#) [Previous](#) [Highlight all](#) Match case

Done



Is SPARQL Difficult?

“Consider, for instance, SPARQL, a query language. To find, say, music artists associated with the producer Timbaland, you'd have to type a long piece of convoluted code that most of us wouldn't bother to do.”



Timbaland, the Resource

- Wikipedia page:
[http://**en.wikipedia.org/wiki**/Timbaland](http://en.wikipedia.org/wiki/Timbaland)
- Dbpedia URI for Timbaland as a resource to query about:
[http://**dbpedia.org/resource**/Timbaland](http://dbpedia.org/resource/Timbaland)



This query...

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

SELECT DISTINCT ?artist
WHERE {
    ?album d:producer
        <http://dbpedia.org/resource/Timbaland> .
    ?album d:musicalArtist ?artist .
}
```



entered here...

Virtuoso SPARQL Query Form - Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://dbpedia.org/sparql

Virtuoso SPARQL Query Form

OpenLink Virtuoso SPARQL Query

This query page is designed to help you test OpenLink Virtuoso SPARQL protocol endpoint. Consult the [Virtuoso Wiki page](#) describing the service or the [Online Virtuoso Documentation](#) section [RDF Database and SPARQL](#).

There is also a rich Web based user interface with sample queries. You can access it at: [/isparql](#).

Query

Default Graph URI

Use only local data (including data retrieved before), but do not retrieve more

Query text

```
PREFIX d: <http://dbpedia.org/property/>

SELECT DISTINCT ?artist WHERE {
  ?album d:artist ?artist.
  ?album d:producer <http://dbpedia.org/resource/Timbaland>.
}
```

Display Results As: Rigorous check of the query

Execution timeout, in milliseconds, values less than 1000 are ignored

Find: Match case

Done



and there they are.

Mozilla Firefox

File Edit View History Bookmarks Tools Help

http://dbpedia.org/sparql?default-graph-uri=http%3A%20... Google

http://dbpedia.o...bug=on&timeout=

artist
Various Artists
http://dbpedia.org/resource/MC_Solaar
http://dbpedia.org/resource/Nicole_Wray
http://dbpedia.org/resource/Big_Boi
http://dbpedia.org/resource/Missy_%22Misdemeanor%22_Elliott
http://dbpedia.org/resource/Ashlee_Simpson-Wentz
http://dbpedia.org/resource/Madonna_%28entertainer%29
http://dbpedia.org/resource/OneRepublic
http://dbpedia.org/resource/Justin_Timberlake
http://dbpedia.org/resource/Missy_Elliott
http://dbpedia.org/resource/Ludacris
http://dbpedia.org/resource/Lil_Kim
http://dbpedia.org/resource/Aaliyah
http://dbpedia.org/resource/Brandy_%28entertainer%29
http://dbpedia.org/resource/Da_Brat
http://dbpedia.org/resource/Snoop_Dogg
http://dbpedia.org/resource/Jay-Z
http://dbpedia.org/resource/Pussycat_Dolls
http://dbpedia.org/resource/The_Pussycat_Dolls
http://dbpedia.org/resource/Xzibit

Find: Next Previous Highlight all Match case

Done



On the other hand...

Some JavaScript from a View Source of that same CNN page:

```
if(cnnWinExtraRegExp.test(cnnWinExtra)){var cnnOmniExtra =
cnnWinExtraRegExp.split(cnnWinExtra);cnnWinLoc = cnnWinLoc +
cnnOmniExtra[0];} else {cnnWinLoc = cnnWinLoc +
cnnWinExtra;}} if (typeof(cnnPageName) != "undefined")
{s.pageName = cnnPageName;s.eVar1 = cnnPageName;} else
{s.pageName = cnnWinLoc;s.eVar1 = cnnWinLoc;} if
(typeof(cnnSectionName) != "undefined")
{s.channel=cnnSectionName;s.eVar2=cnnSectionName;} else
{s.channel="Nonlabeled";s.eVar2="Nonlabeled";} if
(typeof(cnnSubSectionName) != "undefined")
{s.server=cnnSubSectionName;s.eVar3=cnnSubSectionName;} else
{s.server="";s.eVar3="";} if (typeof(cnnSectionFront) !=
"undefined") {s.prop1=cnnSectionFront;} if
(typeof(cnnContentType) != "undefined")
{s.prop4=cnnContentType;s.prop6=s.pageName;}
```



The Semantic Web

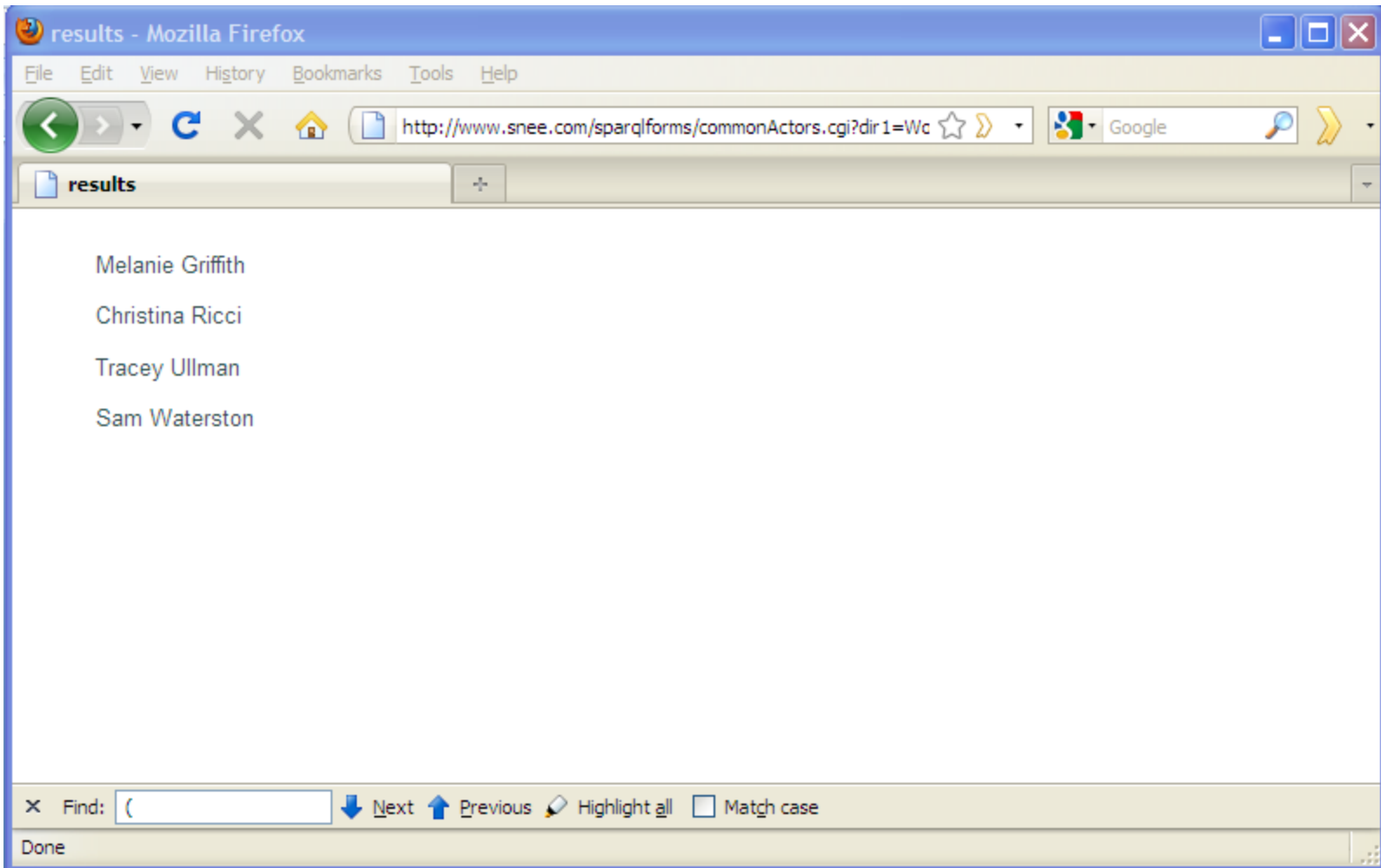
A set of standards and best practices for sharing data and the semantics of that data **over the web for use by applications.**



Form-based SPARQL app

The screenshot shows a Mozilla Firefox browser window with the title "Find common actors between two directors - Mozilla Firefox". The address bar contains the URL "http://www.snee.com/sparqlforms/commonActors.html". The page content includes the heading "Find common actors between two directors" and a paragraph: "Enter the 'official' name of each director (check [IMDB](#) if you're not sure) and click 'Search' to list actors who have appeared in movies by both directors." Below this text are two text input fields containing "Woody Allen" and "John Waters", and a "search" button. The browser's status bar at the bottom shows "Done" and search options like "Find: (", "Next", "Previous", "Highlight all", and "Match case".

Form-based SPARQL app: results



The screenshot shows a Mozilla Firefox browser window with the title "results - Mozilla Firefox". The address bar contains the URL `http://www.snee.com/sparqlforms/commonActors.cgi?dir1=Wc`. The page content displays a list of actor names:

- Melanie Griffith
- Christina Ricci
- Tracey Ullman
- Sam Waterston

At the bottom of the browser window, there is a search bar with the text "Find: (" and navigation buttons for "Next", "Previous", "Highlight all", and "Match case". The status bar at the very bottom shows "Done".



commonactors.cgi main() part 1

```
def main():  
    form = cgi.FieldStorage()  
    dir1name = form.getvalue('dir1')  
    dir2name = form.getvalue('dir2')  
  
    sparql = SPARQLWrapper("http://data.linkedmdb.org/sparql")  
    queryString = ""
```

```
PREFIX m: <http://data.linkedmdb.org/resource/movie/>
```

```
SELECT DISTINCT ?actorName WHERE {
```

```
    ?dir1      m:director_name "DIR1-NAME".
```

```
    ?dir2      m:director_name "DIR2-NAME".
```

```
    ?dir1film m:director ?dir1;
```

```
        m:actor ?actor.
```

```
    ?dir2film m:director ?dir2;
```

```
        m:actor ?actor.
```

```
    ?actor    m:actor_name ?actorName.
```

```
}
```

```
"""
```



commonactors.cgi main() part 2

```
queryString = queryString.replace("DIR1-NAME",dir1name)  
queryString = queryString.replace("DIR2-NAME",dir2name)
```

```
sparql.setQuery(queryString)  
sparql.setReturnFormat(JSON)
```

```
try:  
    ret = sparql.query()  
    results = ret.convert()  
    requestGood = True  
except Exception, e:  
    results = str(e)  
    requestGood = False
```



commonactors.cgi main() part 3

```
print """Content-type: text/html

<html>
  <head>
    <title>results</title>
    <link href="simple.css" type="text/css" rel="stylesheet" />
  </head>
  <body>
"""
```

```

if requestGood == False:
    print "<h1>Problem communicating with the server</h1>"
    print "<p>" + results + "</p>"
elif (len(results["results"]["bindings"]) == 0):
    print "<p>No results found.</p>"

else:
    for result in results["results"]["bindings"]:
        print "<p>" + result["actorName"]["value"] + "</p>"

print "</body></html>"
```



SPARQL Query Results XML Format

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX air:
<http://www.megginson.com/exp/ns/airports#>

SELECT ?personName ?airportCode
WHERE {
    ?person foaf:name ?personName ;
            foaf:nearestAirport ?airport .
    ?airport air:iata ?airportCode .
}
```



sparql.org query result

← → ↻ 🏠 www.sparql.org/sparql?query=PREFIX+foaf:+<http://xmlns.com/foaf/0.1/>%0D%0APREFIX+air:+<http://www.megginson.com/exp/ ☆ 🔍

SPARQLer Query Results

personName	airportCode
"Bob DuCharme"	"CHO"



SPARQL Query Results XML Format

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

```
<head>
```

```
<variable name="personName"/>
```

```
<variable name="airportCode"/>
```

```
</head>
```

```
<results>
```

```
<result>
```

```
<binding name="personName">
```

```
<literal>Bob DuCharme</literal>
```

```
</binding>
```

```
<binding name="airportCode">
```

```
<literal>CHO</literal>
```

```
</binding>
```

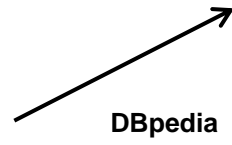
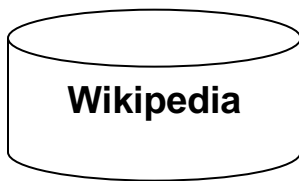
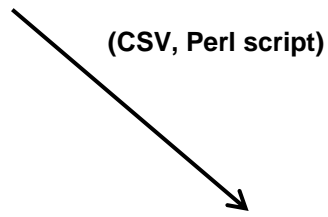
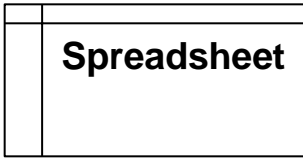
```
</result>
```

```
</results>
```

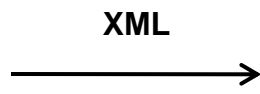
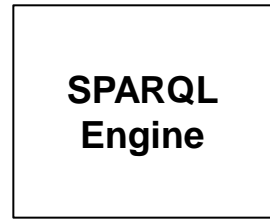
```
</sparql>
```

Three basic steps

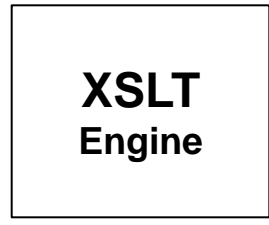
1.



2.



3.



Analyst recommendation spreadsheet

	A	B	C	D	E	F
1	analyst	Ticker Symbol	Wikipedia ID	recommendation	date-time	description
2	Nick Perkins	GOOG	Google	SELL	2009-12-14T13:36:00	Google has had an interesting quarter. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent sed lectus augue. Suspendisse nisl nisl, pulvinar eu luctus non, sodales non magna. Sed in metus arcu, sit amet ornare nunc. Duis fermentum, nibh quis fermentum sagittis, mi eros
3	Liz Ford	VOD	Vodafone	BUY	2009-12-15T18:24:00	Vodafone has had an interesting quarter. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent sed lectus augue. Suspendisse nisl nisl, pulvinar eu luctus non, sodales non magna. Sed in metus arcu, sit amet ornare nunc. Duis fermentum, nibh quis fermentum sagittis, mi eros
4	Betty Bailey	SNE	Sony	HOLD	2009-12-16T17:21:00	Sony has had an interesting quarter. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent sed lectus augue. Suspendisse nisl nisl, pulvinar eu luctus non, sodales non magna. Sed in metus arcu, sit amet ornare nunc. Duis fermentum, nibh quis fermentum sagittis, mi eros
5	Marilyn Walker	HMC	Honda	BUY	2009-12-16T10:13:00	Honda has had an interesting quarter. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent sed lectus augue. Suspendisse nisl nisl, pulvinar eu luctus non, sodales non magna. Sed in metus arcu, sit amet ornare nunc. Duis fermentum, nibh quis fermentum sagittis, mi eros
6	Linda Morales	IBM	IBM	SELL	2009-12-14T13:36:00	IBM has had an interesting quarter. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent sed lectus augue. Suspendisse nisl nisl, pulvinar eu luctus non, sodales non magna. Sed in metus arcu, sit amet ornare nunc. Duis fermentum, nibh quis fermentum sagittis, mi eros
	Matt Moore	TWX	Time_Warner	SELL	2009-12-16T14:24:00	Time Warner has had an interesting quarter. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent sed lectus augue. Suspendisse nisl nisl, pulvinar eu luctus non, sodales non magna. Sed in metus arcu, sit

Sheet1 Sheet2 Sheet3

Ready 100%

Generated report

Analyst Recommendations: Report

Time Warner TWX \$28.16 at 2010-01-15T16:00:00

2008: Revenue 46980000000 Net Income -13400000000

Time Warner Inc. is the world's largest media and entertainment conglomerate, headquartered in the Time Warner Center in New York City. Formerly three separate companies (and owns the assets of a fourth, Turner Broadcasting System, Inc. , acquired by a pre-AOL merger TW in 1996): Warner Communications, Inc. and Time Inc. before the Time-Warner merger in 1990 and America Online, Inc. before its purchase of Time Warner in 2001 has created the current Time Warner, with major operations in film, television, publishing, Internet service and telecommunications. Among its subsidiaries are AOL, New Line Cinema, Time Inc. , HBO, Turner Broadcasting System, The CW Television Network, TheWB.com, Warner Bros. Entertainment, Kids' WB, The CW4Kids, Cartoon Network, Hanna-Barbera, Ruby-Spears Productions, Adult Swim, CNN, DC Comics, and Warner Bros. Games. Terra Firma Capital Partners is likely to buy a 10% Stake in TW, since it would buy all the remaining rights and stakes of Time Warner in Warner Music Group and merge it with EMI.

Analyst Matt Moore	Recommendation SELL	Date 2009-12-16T14:24:00
---------------------------	----------------------------	---------------------------------

Time Warner has had an interesting quarter. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent sed lectus augue. Suspendisse nisi nisi, pulvinar eu luctus non, sodales non magna. Sed in metus arcu, sit amet ornare nunc. Duis fermentum, nibh quis fermentum sagittis, mi eros porttitor magna, sed dictum tortor quam ut lectus. Praesent eu est augue.



IBM \$131.78 at 2010-01-15T16:00:00

2008: Revenue 103630000000 Net Income 12334000000

International Business Machines Corporation, abbreviated IBM, is a multinational computer technology and IT consulting corporation headquartered in Armonk, New York, United States. The company is one of the few information technology companies with a continuous history dating back to the 19th century. IBM manufactures and sells computer hardware and software (with a focus on the latter), and offers infrastructure services, hosting services, and consulting services in areas ranging from mainframe computers to nanotechnology. It has been nicknamed "Big Blue" for its official corporate color. IBM has been well known through most of its recent history as the world's largest computer company and systems integrator. With over 388,000 employees worldwide, IBM is the largest and most profitable information technology employer in the world. IBM holds more patents than any other U.S. based technology company and has eight research laboratories worldwide. The company has scientists, engineers, consultants, and sales professionals in over 170 countries. IBM employees have earned five Nobel Prizes, four Turing Awards, five National Medals of Technology, and five National Medals of Science. As a chip maker, IBM has been among the Worldwide Top 20 Semiconductor Sales Leaders in past years.

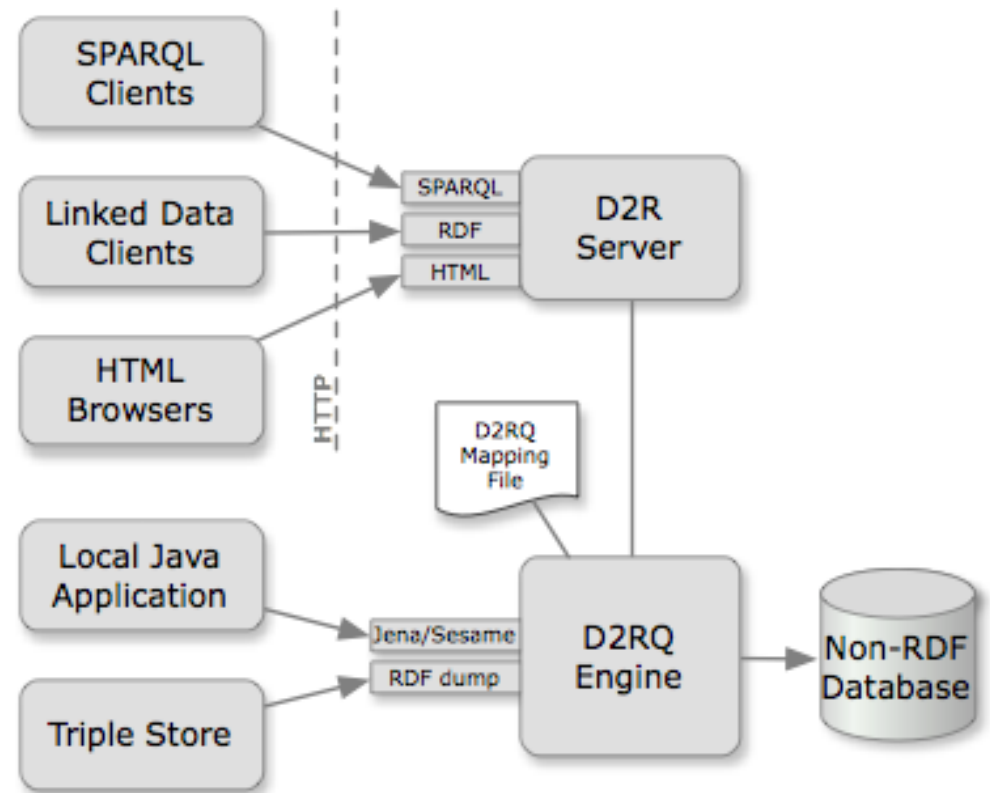
Analyst Linda Morales	Recommendation SELL	Date 2009-12-14T13:36:00
------------------------------	----------------------------	---------------------------------

IBM has had an interesting quarter. Lorem ipsum dolor sit amet, consectetur adipiscing elit. Praesent sed lectus augue. Suspendisse nisi nisi, pulvinar eu luctus non, sodales non magna. Sed in metus arcu, sit amet ornare nunc. Duis fermentum, nibh quis fermentum sagittis, mi eros porttitor magna, sed dictum tortor quam ut lectus. Praesent eu est augue.

HONDA HMC \$36.90 at 2010-01-15T16:03:00
The Power of Dreams

Querying relational data with SPARQL

- Open source
- Query multiple databases at once
- Used by Linked Movie Database
- Great for linked data behind the firewall



- W3C developing standardized version of D2RQ language



SPARQL Rules

- “When condition X is true, generate these new triples”
- “When condition Y is true, alert me—it shouldn’t be true”
- Developed at TopQuadrant, now a W3C submission



UN AGROVOC Thesaurus

- SKOS: RDF/OWL-based W3C standard for taxonomies and thesaurii
- SKOS spec lists six rules not implemented by ontology
- e.g. same term can't be preferred and alternative term in the same language
- Expressed with SPARQL Rules
- Violated by AGROVOC over 1600 times
- e.g. Slovak word for “Buds” (“púèiky”)



In conclusion...

When you know some SPARQL, you can do more with:

- RDFS
- OWL
- triplestores
- SPARQL endpoints
- Data aggregation applications
- Linked Data
- Relational data, spreadsheets
- Data quality rules



1. Jumping Right In: Some Data and Some Queries
2. The Semantic Web, RDF, and Linked Data (and SPARQL)
3. SPARQL Queries: A Deeper Dive
4. Copying, Creating, and Converting Data (and Finding Bad Data)
5. Datatypes and Functions
6. Updating Data with SPARQL
7. Building Applications with SPARQL: A Brief Tour

Glossary

Index

