

Tema 4. Tecnologías de la Web3.0

3ª Parte: SPARQL

J. Javier Samper Zapater
José Fco. García Calderaro
Departamento de Informática

VNIVERSITAT  ID VALÈNCIA

Breve Introducción a SPARQL

Introducción

- SPARQL es bastante similar a SQL. Por ejemplo:

```
SELECT ?persona ?nombre WHERE {  
  ?persona <http://www.w3.org/1999/02/22-rdf-syntax-ns#type>  
  <http://xmlns.com/foaf/0.1/Person> .  
  ?persona <http://xmlns.com/foaf/0.1/name> ?nombre .  
}
```

- Como se puede ver, las variables comienzan con "?" y después del WHERE comienzan una lista de patrones de grafos (una lista de sujeto, predicado y objeto que finaliza con un punto) que los posibles resultados deben cumplir. Así, la consulta anterior busca cualquier entidad que sea de tipo "*Persona*" y que además posea un nombre.

Introducción

- Esta consulta se puede optimizar de varias maneras. Primero, como las URIs son generalmente muy largas, es posible definir un prefijo usando una forma abreviada de éstas. El prefijo se define al comienzo y puede ser usado en el resto de la consulta:
- PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

```
SELECT ?persona ?nombre WHERE {  
  ?persona rdf:type foaf:Person .  
  ?persona foaf:name ?nombre .  
}
```

- Así, el SPARQL endpoint traducirá foaf:Person a http://xmlns.com/foaf/0.1/Person, que es la URI original. Esto hace más fácil leer los patrones.

Introducción

- Sin embargo, podemos hacer dos pequeñas optimizaciones más: Al ser tan común, el predicado `rdf:type` también puede ser descrito simplemente con una "a". Además, como en este caso el sujeto se repite mucho, es posible "reusar" el primer sujeto, indicando que el próximo patrón de grafo utiliza el mismo. Para esto, al final del primer patrón usamos un punto y coma en vez de un punto. Nuestra consulta queda así:
- PREFIX foaf: <<http://xmlns.com/foaf/0.1/>>
PREFIX rdf: <<http://www.w3.org/1999/02/22-rdf-syntax-ns#>>

```
SELECT ?persona ?nombre WHERE {  
  ?persona a foaf:Person ;  
           foaf:name ?nombre .  
}
```

Introducción

- SPARQL permite acceder a una gran cantidad de información disponible en la Web, ya sea en [DBpedia](#), el [New York Times](#), o el gobierno de [El Reino Unido](#). Es importante recordar que en el caso de DBpedia, la información es tomada de Wikipedia y extraída lo mejor posible; sin embargo, siempre es posible que exista algún error en los datos, algo que es más difícil que ocurra en los datos de un periódico o un gobierno.
- Hay varias formas de publicar (y por ende usar) Linked Data, pero quizás la más fácil por ahora sea utilizar **SPARQL endpoints**, los cuales son servicios web que permiten hacer consultas a repositorios semánticos utilizando el lenguaje de consulta SPARQL.

Ejemplo de uso

- Supongamos que necesitamos crear una aplicación sobre poetas chilenos. Por ejemplo, el identificador en DBpedia de Pablo Neruda es http://dbpedia.org/resource/Pablo_Neruda. Si ponemos esa dirección en el navegador, seremos redirigidos a un documento HTML que describe información acerca de Pablo Neruda. Entre los atributos que posee Pablo Neruda está la categoría http://dbpedia.org/resource/Category:Chilean_poets. Así, podemos crear una consulta en base a esta categoría para obtener todos los poetas chilenos.
- PREFIX dcterms: <<http://purl.org/dc/terms/>>
PREFIX rdfs: <<http://www.w3.org/2000/01/rdf-schema#>>
SELECT ?poeta ?nombrePoeta WHERE {
 ?poeta dcterms:subject <http://dbpedia.org/resource/Category:Chilean_poets>;
 rdfs:label ?nombrePoeta .
FILTER (LANG(?nombrePoeta) = "es")
}
- Esta consulta retorna todos los poetas que están asociados a la categoría http://dbpedia.org/resource/Category:Chilean_poets y que tengan un nombre, descrito basado utilizando [rdfs:label](#). Un detalle importante. Por simplicidad, filtramos los resultados para que los nombres que aparecen sean sólo en español (por eso la instrucción FILTER (LANG(?nombrePoeta) = "es")).

Ejemplo de uso: Resultado

Ejecutamos la consulta en el endpoint de DBpedia (<http://dbpedia.org/sparql>) y obtenemos el siguiente resultado:

Poeta

nombrePoeta

http://dbpedia.org/resource/Nicanor_Parra

"Nicanor Parra"@es

http://dbpedia.org/resource/V%C3%ADctor_Jara

"V́ctor Jara"@es

http://dbpedia.org/resource/Eduardo_Parra_Pizarro

"Eduardo Parra Pizarro"@es

http://dbpedia.org/resource/Alberto_Baeza_Flores

"Alberto Baeza Flores"@es

http://dbpedia.org/resource/Juan_Guzm%C3%A1n_Cruchaga

"Juan Guzmán Cruchaga"@es

http://dbpedia.org/resource/Sergio_Badilla_Castillo

"Sergio Badilla Castillo"@es

http://dbpedia.org/resource/V%C3%ADctor_Domingo_Silva

"V́ctor Domingo Silva"@es

....

....

Ejemplo de uso

Ahora podemos agregar más restricciones. Por ejemplo, queremos saber las fechas de nacimiento y muerte de todos los poetas chilenos. Es importante recalcar que si algún poeta no tiene alguna de esas fechas, no aparecerá en los resultados. Así, definimos la siguiente consulta:

```
PREFIX dcterms: <http://purl.org/dc/terms/>
```

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

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

```
SELECT ?poeta ?nombrePoeta ?fechaNacimiento ?fechaFallecimiento
```

```
WHERE{
```

```
  ?poeta dcterms:subject
```

```
  <http://dbpedia.org/resource/Category:Chilean_poets>;
```

```
    rdfs:label ?nombrePoeta ;
```

```
    dbp:birthDate ?fechaNacimiento ;
```

```
    dbp:deathDate ?fechaFallecimiento .
```

```
FILTER (LANG(?nombrePoeta) = "es")
```

```
}
```

Por comodidad definimos un nuevo prefijo, al que llamamos dbp y que será un shortcut de <http://dbpedia.org/ontology/>.

Ejemplo de uso: Resultado

poeta	nombrePoeta	fechaNacimiento	fechaFallecimiento
http://dbpedia.org/resource/V%C3%ADctor_Jara	"Víctor Jara"@es	1932-09-28	1973-09-16
http://dbpedia.org/resource/V%C3%ADctor_Domingo_Silva	"Víctor Domingo Silva"@es	1882-05-12	1960-08-20
http://dbpedia.org/resource/Pablo_Neruda	"Pablo Neruda"@es	1904-07-12	1973-09-23
http://dbpedia.org/resource/Vicente_Huidobro	"Vicente Huidobro"@es	1893-01-10	1948-01-02
http://dbpedia.org/resource/Andr%C3%A9s_Bello	"Andrés Bello"@es	1781-11-29	1865-10-15
http://dbpedia.org/resource/Roberto_Bola%C3%B1o	"Roberto Bolaño"@es	1953-04-28	2003-07-15
http://dbpedia.org/resource/Gabriela_Mistral	"Gabriela Mistral"@es	1889-04-07	1957-01-10
http://dbpedia.org/resource/Daniel_de_la_Vega_%28poet%29	"Daniel de la Vega"@es	1892-06-30	1971-07-29
http://dbpedia.org/resource/Herm%C3%B3genes_Irisarri	"Hermógenes Irisarri"@es	1819-04-19	1886-07-22
http://dbpedia.org/resource/Carlos_Pezoa_V%C3%A9liz	"Carlos Pezoa Véliz"@es	1879-07-21	1908-04-21
http://dbpedia.org/resource/Augusto_d%27Halmar	"Augusto d'Halmar"@es	1882-04-23	1950-01-27
http://dbpedia.org/resource/Mar%C3%ADa_de_la_Cruz	"María de la Cruz Toledo"@es	1912-09-18	1995-09-01
http://dbpedia.org/resource/Juvencio_Valle	"Juvencio Valle"@es	1900-11-06	1999-02-12

Otras consultas

Podemos crear otras consultas que nos entreguen distintos tipos de información, dependiendo de lo que necesitemos. Por ejemplo, si buscamos quiénes han influenciado a los poetas chilenos, basta con buscar de la siguiente forma:

```
PREFIX dcterms: <http://purl.org/dc/terms/>
```

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

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

```
SELECT ?poeta ?influencia WHERE{  
  ?poeta dcterms:subject  
<http://dbpedia.org/resource/Category:Chilean_poets>.  
  ?influencia dbp:influenced ?poeta.  
}
```

Otras consultas: Resultado

poeta

http://dbpedia.org/resource/Nicanor_Parra

http://dbpedia.org/resource/Pablo_Neruda

http://dbpedia.org/resource/Vicente_Huidobro

http://dbpedia.org/resource/Roberto_Bola%C3%B1o

http://dbpedia.org/resource/Roberto_Bola%C3%B1o

http://dbpedia.org/resource/Roberto_Bola%C3%B1o

http://dbpedia.org/resource/Gonzalo_Rojas

http://dbpedia.org/resource/Gabriela_Mistral

influencia

http://dbpedia.org/resource/Carlos_Pezoa_V%C3%A9liz

http://dbpedia.org/resource/Rabindranath_Tagore

http://dbpedia.org/resource/Tristan_Tzara

http://dbpedia.org/resource/Jorge_Luis_Borges

http://dbpedia.org/resource/Philip_K._Dick

http://dbpedia.org/resource/Julio_Cort%C3%A1zar

http://dbpedia.org/resource/Carlos_Pezoa_V%C3%A9liz

http://dbpedia.org/resource/Rabindranath_Tagore

Ejercicios

- Hacer uso de las referencias al final de esta presentación para una detallada comprensión de la sintaxis SPARQL.
- Realizar estos dos ejercicios:
 - 1) Obras de Roberto Bolaño y sus títulos en inglés.
 - 2) Poetas chilenos que no hayan muerto y que hayan nacido antes de 1950:

Referencias

- SPARQL Lenguaje de consulta para RDF. Recomendación del W3C de 15 de enero de 2008: <http://skos.um.es/TR/rdf-sparql-query/>
- DBpedia español: Ejemplos de consultas SPARQL: <http://es.dbpedia.org/Wiki.jsp?page=Ejemplos%20de%20consultas%20SPARQL>
- SPARQL Overview: <http://www.w3.org/2009/Talks/0615-qbe/Overview.html>
- RDF y SPARQL "Aprendiendo a nadar en el diluvio de datos" (II). Miguel A. Martínez Prieto & Javier D. Fernández. Departamento de Informática, Universidad de Valladolid
- SPARQL. Jose Emilio Labra Gayo. Departamento de Informática. Universidad de Oviedo