

Consultas SparQL para el filtrado de información mediante Twinkle sobre una estructura RDF extraída de la Wikidata.

BRYAN ASMAT, DANILO BLAS, VICTOR CHAVEZ, WILLIAM GUADALUPE

Universidad Nacional de Ingeniería

Lima, Perú

basmatf@uni.pe, iblass@uni.pe, vchavezb@uni.pe, wguadalupeq@uni.pe

Resumen—En el presente trabajo se implementaron consultas de tipos SparQL que permiten hacer la búsqueda en distintas fuentes de datos como estructuras RDF, la creación de esta estructura, así como un resumen de los pasos que realizamos para la ejecución del proyecto.

Para iniciar el proyecto se usó la base de conocimiento de la Wikidata realizando consultas online <https://query.wikidata.org/> con lo cual seguiremos los siguientes pasos para que se entienda como se ha creado la estructura RDF. Accedemos al enlace mencionado anteriormente. Podemos usar las consultas con extensión .rq que están almacenadas en el repositorio de este proyecto: Si usamos el contenido del archivo `onlineQuerySELECT.rq`

copiaremos la información y la usaremos en el servicio de consultas de la Wikidata nos devuelve la tabla que hemos construido mediante las consultas que hemos hecho. Si usamos el contenido del archivo `onlineQueryCONSTRUCT.rq` copiaremos la información y la usaremos en el servicio de consultas de la Wikidata nos devuelve como serían las tripletas en una nueva estructura RDF que vamos a manejar. Descargamos el archivo TSV detallado que se genera al usar

el archivo `onlineQueryCONSTRUCT.rq` y lo modificamos para que tenga el formato Turtle a usar en el presente proyecto. El archivo RDF en formato turtle una vez generado y se encuentra en el siguiente en el directorio RDF con el siguiente nombre `RDE.ttl`. Hasta ahora los pasos que

hemos hecho es para extraer una estructura RDF nueva de Wikidata realizando consultas SparQL para obtener datos de investigaciones y trabajos científicos, digamos obras, que se basen sobre el tema de Ciencias de la Computación. De estos datos hemos extraído las siguientes características:

Título, Autor, Tema, Idioma, Revista, Fecha de Publicación, URL.

I. INTRODUCCIÓN

La información en la web se puede representar bajo un formato de datos llamado RDF. El consorcio World Wide Web (W3C) propuso una especificación llamada SparQL (lenguaje de consultas) para manipular grafos compuestos por tripletas RDF en la web y en repositorios de datos, así como extraer datos mediante consultas a relaciones desconocidas y realizar búsquedas con combinaciones complejas de base de datos.

La estructura de los resultados de una consulta SPARQL es una tabla donde cada fila es un resultado y cada columna se corresponde con una de las variables de la consulta.

Una consulta SparQL consta de tres partes:

- Coincidencia de patrones: optional, union, nesting, filtering.
- Modificadores de solución: projection, distinct, order, limit, offset.
- Parte de salida: construcción de nuevas tripletas.

Las búsquedas se realizan a través de un SparQL Endpoint, que es la interfaz de entrada y salida de estas bases de datos y que permitirá nuestras consultas. Dicho de otra forma, los Endpoints de SparQL no son más que una dirección web (URL) que acepta peticiones de búsqueda SparQL y devuelve resultados, las formas de consulta SparQL son select, construct, ask y describe.

¿Cuáles son los desafíos de diseño en SparQL?

SparQL debe tener en cuenta las características distintas de RDF:

- Debería poder extraer información de grafos RDF interconectados.
- Debe ser coherente con la semántica de mundo abierto de RDF.
- Debe ofrecer la posibilidad de agregar información opcional si está presente.
- Debería ser capaz de interpretar correctamente grafos RDF con un vocabulario con semántica predefinida.
- Debería ofrecer algunas funcionalidades para navegar en un grafo RDF.

II. ¿QUE ES LA WEB SEMÁNTICA?

La Web Semántica vendría a ser una extensión de la Web actual dotada de significado, esto es, un espacio

donde la información tendría un significado bien definido, de manera que pudiera ser interpretada tanto por agentes humanos como por agentes computarizados.

La web Semántica ha sido impulsada, entre otros, por el creador de la web, Tim Berners Lee en el 2000 para que la información sea reunida de forma que los buscadores puedan "comprender", más allá de limitarse a colocar los documentos en una lista

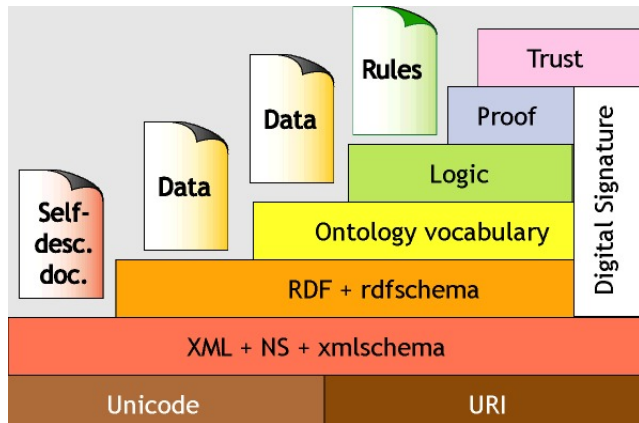


Figura 1. Web semántica [4]

III. URLs, URIs, IRIs

III-A. URLs

El LRU es una cadena de caracteres con la que se asigna una dirección única a cada uno de los recursos de información disponibles en Internet. Existe un URL único para cada página de cada uno de los documentos de la WWW

III-B. URIs

Un identificador de recursos uniforme es una cadena de caracteres que identifica los recursos de una red de forma unívoca. La diferencia respecto a un localizador de recursos uniforme (URL) es que estos últimos hacen referencia a recursos que, de forma general, pueden variar en el tiempo.

Normalmente estos recursos son accesibles en una red o sistema. Los URI pueden ser localizador de recursos uniforme (URL),(URN)

III-C. IRIs

Es un estándar de protocolo de Internet que se basa en el protocolo Identificador uniforme de recursos (URI) al expandir en gran medida el conjunto de caracteres permitidos.

IV. ESTRUCTURA RDF

RDF es un método general para descomponer cualquier tipo de conocimiento en trozos pequeños, con algunas reglas acerca de la semántica o significado, de esas piezas.

El punto es tener un método tan simple que puede expresar cualquier hecho, y a la vez tan estructurada que las aplicaciones informáticas pueden hacer cosas útiles con él. [5]

RDF representa la información en una declaración triple -también llamada tripleta- que sigue la estructura sujeto-predicado-objeto [5]

Ejemplo

<La ronda de noche><fue creada por><Rembrandt>

En este ejemplo <La ronda de noche> y el objeto <Rembrandt> se pueden considerar como dos nodos en un grafo donde el predicado <fue creada por> define una arista o la relación entre estos . [5]

V. BUSCANDO RDF CON SPARQL

SPARQL nos permite traducir datos en grafo, intencionalmente enlazados, en datos normalizados en formato tabular, esto es, distribuidos en filas y columnas, que se pueden abrir en formato como .csv .json .ttl ,etc o también se pueden ver en programas de visualización

Resulta útil pensar las consultas SPARQL como un conjunto de oraciones con espacios en blanco. La base de datos tomará esta consulta y encontrará cada conjunto de oraciones que encaje correctamente en estos espacios en blanco, devolviéndonos los valores coincidentes como una tabla.

VI. CONSULTAS SPARQL

SPARQL, acrónimo de SPARQL Protocol and RDF Query Language, es un lenguaje estandarizado para hacer consultas en estructuras RDF. Es una tecnología clave en el desarrollo de la web semántica que se constituyó como recomendación oficial del W3C el 15 de enero de 2008, siendo actualizado a la versión 1.1 en 2013 [1]. En un principio SPARQL únicamente incorpora funciones para la recuperación sentencias RDF. Sin embargo, algunas propuestas también incluyen operaciones para el mantenimiento (creación, modificación y borrado) de datos.

VI-A. Especificaciones

SPARQL 1.0:

- SPARQL como lenguaje de consultas para RDF, cubre la sintaxis de las consultas (QL).
- SPARQL como protocolo para RDF especifica cómo un programa debe pasar consultas SPARQL a un servicio de procesamiento de consultas SPARQL y cómo este servicio las retorna.
- Formato XML de resultados de consultas SPARQL, describe un formato simple XML para que los procesadores de consultas lo utilicen al devolver resultados.

Sparql 1.1:

- La especificación de consulta federada describe como una simple consulta puede traer data de multiples fuentes, esto hace más fácil construir aplicaciones que toman ventaja de entornos distribuidos.
- La especificación de actualización es la diferencia clave entre SPARQL 1.0 y 1.1 porque lleva a SPARQL de ser un lenguaje de consulta a algo que puede agregar datos a un conjunto de datos, reemplazarlos y eliminarlos también.
- La especificación de Descripción del servicio describe cómo un programa cliente puede preguntarle a un motor SPARQL exactamente qué características admite.
- La especificación del formato JSON de los resultados de la consulta describe un equivalente JSON del formato XML de los resultados de la consulta.
- La especificación de los formatos CSV y TSV de los resultados de la consulta describe los equivalentes de valores separados por comas y tabuladores del formato XML de los resultados de la consulta.
- La especificación del Protocolo HTTP de Graph Store amplía el Protocolo SPARQL con una API similar a REST para la comunicación entre un cliente y un procesador SPARQL sobre gráficos o conjuntos de tripletas. [4]

VI-B. Ejemplos de Consultas

Veamos algunas consultas que son comunes y así poder entenderlas.

Ejemplo

```
SELECT ?subject ?predicate ?object
WHERE {
    ?subject ?predicate ?object .
}
```

Listing 1. SPARQL query

En esta consulta, seleccionamos todas las variables del patrón para poder representarlas con un *carácter. Además, las variables se pueden llamar de cualquier manera.

Obviamente, para obtener consultas mas útiles, se deberá corregir las variables . Cambiando el predicado podemos obtener veinte entidades con sus etiquetas:

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
SELECT *
WHERE {
    ?obra wdt:titulo ?titulo .
}
LIMIT 10
```

Listing 2. SPARQL query

Si se desea revisar mas ejemplos de consultas revisar el siguiente enlace [7]

VI-C. Estructura RDF a utilizar

```
3@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
4@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
5@prefix wd: <http://www.wikidata.org/entity-/> .
6@prefix wdt: <http://www.wikidata.org/prop/direct/> .
7
8wd:Q15710145    rdf:type    wdt:obra .
9wd:Q15710145    wdt:tema    "ciencias de la computación"@es .
10wd:Q15710145   wdt:titulo   "International Journal of High Performance Computing Applications"@en .
11wd:Q15710145   wdt:idioma   "inglés"@es .
12wd:Q15749681   rdf:type    wdt:obra .
13wd:Q15749681   wdt:tema    "ciencias de la computación"@es .
14wd:Q15749681   wdt:titulo   "CIT"@en .
15wd:Q15749681   wdt:idioma   "inglés"@es .
16wd:Q15716568   rdf:type    wdt:obra .
17wd:Q15716568   wdt:tema    "ciencias de la computación"@es .
18wd:Q15716568   wdt:titulo   "Cognitive Computation"@en .
19wd:Q15716568   wdt:idioma   "inglés"@es .
20wd:Q15757245   rdf:type    wdt:obra .
21wd:Q15757245   wdt:tema    "ciencias de la computación"@es .
22wd:Q15757245   wdt:titulo   "Journal of Intelligent Systems"@en .
23wd:Q15757245   wdt:idioma   "indonesio"@es .
24wd:Q15761913   rdf:type    wdt:obra .
25wd:Q15761913   wdt:tema    "ciencias de la computación"@es .
26wd:Q15761913   wdt:titulo   "International journal of modelling & simulation"@en .
27wd:Q15761913   wdt:idioma   "inglés"@es .
28wd:Q15761913   wdt:titulo   "International Journal of Modelling and Simulation"@en .
29wd:Q15753694   rdf:type    wdt:obra .
30wd:Q15753694   wdt:tema    "ciencias de la computación"@es .
31wd:Q15753694   wdt:titulo   "AJIS, Australian journal of information systems"@en .
32wd:Q15753694   wdt:idioma   "inglés"@es .
33wd:Q15753694   wdt:titulo   "Australasian Journal of Information Systems"@en .
34wd:Q15749681   wdt:titulo   "Journal of Computing and Information Technology"@en .
35wd:Q15757508   rdf:type    wdt:obra .
36wd:Q15757508   wdt:tema    "ciencias de la computación"@es .
37wd:Q15757508   wdt:titulo   "Ethics and Information Technology"@en .
```

Figura 2. 30 tripletas de la estructura RDF utilizada (ver archivo RDF.ttl)

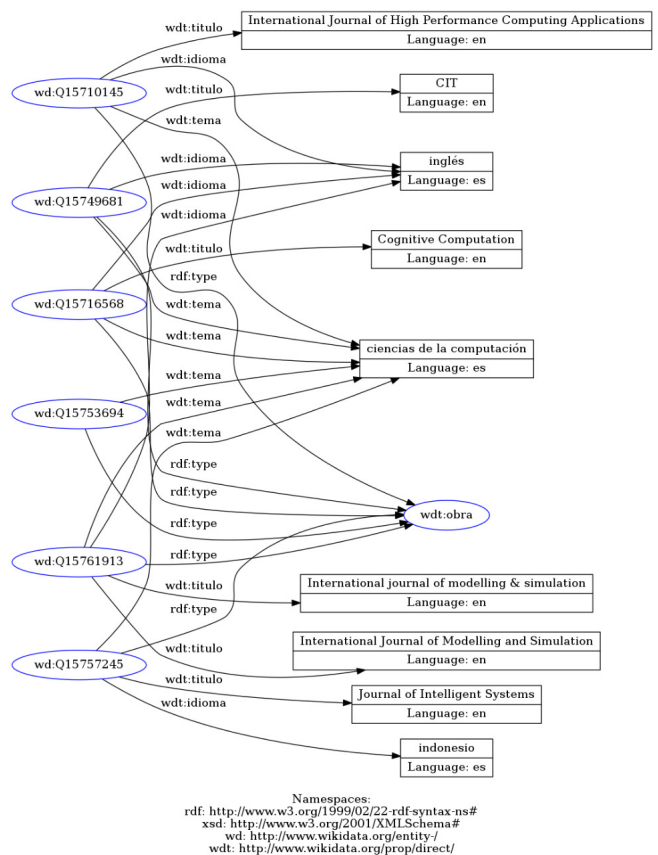


Figura 3. Grafo generado con 30 tripletas de la estructura RDF utilizada [2]

Como podemos apreciar en la figura anterior es muestra la data ya lista para usarse y pasare a explicar algunas líneas y sea mas fácil para los usuarios que vienen a ser abreviaturas de prefijos para WikiData que es donde hemos bajado la información

- **wd:** Una entidad de WikiData como se puede apreciar en el URL <<http://www.wikidata.org/entity/>>
- **wdt:** Propiedades de una entidad de WikiData <<http://www.wikidata.org/prop/direct/>>
- **@en:** Nos dice en que idioma esta en este caso *English* y así sucesivamente si en caso fuera **@es** el idioma español
- **type:** Es una instancia de una clase

VII. CONCLUSIONES

- SPARQL es un lenguaje bastante flexible e intuitivo de aprender , dando nos la facilidad de poder realizar consultas complejas.
- Después extracción de datos en WikiData, estos deben ser limpiados y dar formato *turtle* para así poder realizar consultas en Twinkle.
- Es mejor trabajar con los servicios de consultas en línea por la mayor capacidad de data que tienen y por la mayor funcionalidad que tienen para realizar consultas.

REFERENCIAS

- [1] <https://www.w3.org/TR/rdf-sparql-query/>.
- [2] Página web para generar grafos a partir de una estructura RDF: <https://www.ldf.fi/service/rdf-grapher>.
- [3] Jimmy Lei Ba, Jamie Ryan Kiros, Geoffrey E.Hinton. Layer Normalization (2016). Disponible en <https://arxiv.org/abs/1607.06450>.
- [4] Learning SPARQL Querying and Updating with SPARQL 1.1
- [5] Uso de SPARQL para acceder a datos abiertos enlazados <https://programminghistorian.org/es/lecciones/retirada/sparql-datos-abiertos-enlazados>
- [6] SPARQL query <https://medium.com/wallscope/constructing-sparql-queries-ca63b8b9ac02>
- [7] Consultas SparQL Github <https://github.com/wguadalupeq/practicaSparQL/tree/main/RDF>