

SPARQL

Mikel Egaña Aranguren

mikel-egana-aranguren.github.io

mikel.egana@ehu.eus

erriak ta zabalak 2020

Universidad
del País Vasco Euskal Herriko
Unibertsitatea

BILBOKO
INGENIARITZA
ESKOLA
ESCUELA
DE INGENIERÍA
DE BILBAO

SPARQL

<https://github.com/mikel-egana-aranguren/ABD>



SPARQL

SPARQL: SPARQL Protocol and RDF Query Language

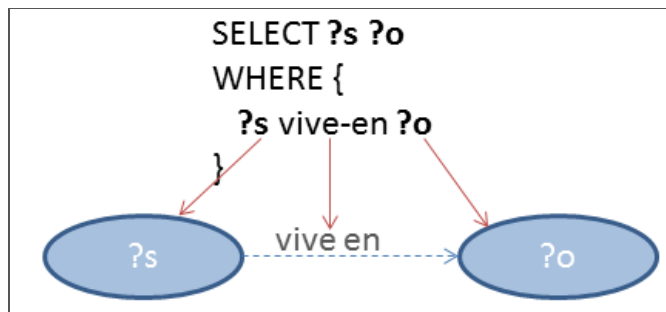
Estándar [oficial del W3C](#) para:

- Hacer consultas sobre grafos RDF
- Ejecutar esas consultas sobre HTTP contra un SPARQL endpoint (Incluyendo consultas federadas)

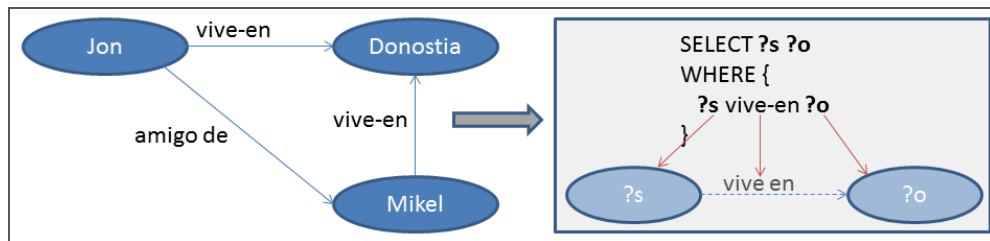
SPARQL

```
SELECT ?s ?o  
  
WHERE {  
  
    ?s vive_en ?o  
  
}
```

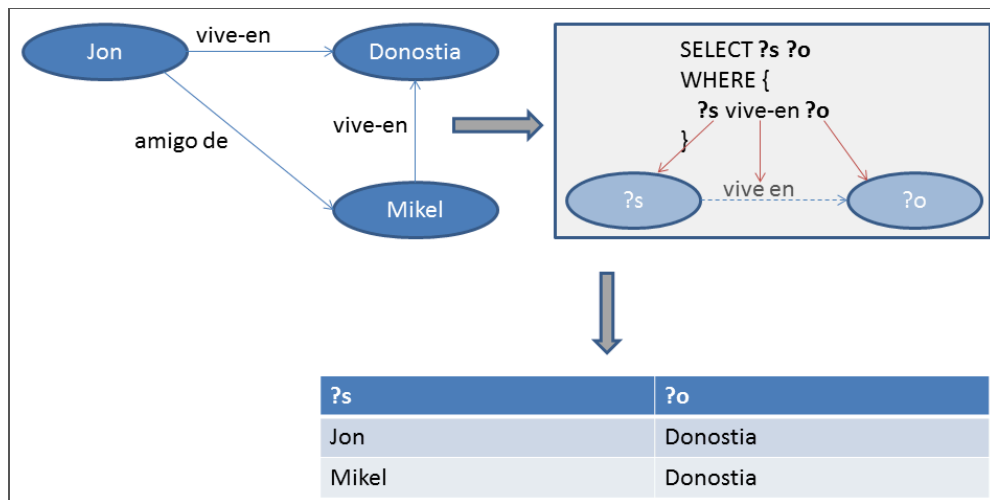
SPARQL



SPARQL



SPARQL



Estructura de la consulta

(Subir museoak.rdf a GraphDB) ([1.rq](#))

```
# Prefixes
PREFIX gip_prop: <http://gipuzkoa.eus/prop/>
PREFIX gip: <http://gipuzkoa.eus/resource/>

# Variables que queremos recibir
SELECT ?sujeto ?clase

# Patrón del grafo que queremos extraer del grafo mayor
WHERE {
  ?sujeto gip_prop:bizilekua gip:donostia .
  ?sujeto rdf:type ?clase
}
```


Estructura de la consulta

([2.rq](#))

```
# Prefixes
PREFIX gip_prop: <http://gipuzkoa.eus/prop/>
PREFIX gip: <http://gipuzkoa.eus/resource/>

# Queremos recibir todas las variables
SELECT *

# Patrón del grafo que queremos extraer del grafo mayor
WHERE {
  ?sujeto gip_prop:bizilekua gip:donostia .
  ?sujeto rdf:type ?clase
}
```

Optional

([3.rq](#))

```
PREFIX gip_prop: <http://gipuzkoa.eus/prop/>
PREFIX gip: <http://gipuzkoa.eus/resource/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT ?langile ?bizilekua

WHERE {
  # Tiene que ser una persona
  ?langile rdf:type foaf:person .

  # Tiene que vivir en algun sitio
  ?langile gip_prop:bizilekua ?bizilekua
}
```

Optional

([4.rq](#))

```
PREFIX gip_prop: <http://gipuzkoa.eus/prop/>
PREFIX gip: <http://gipuzkoa.eus/resource/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT ?langile ?bizilekua

WHERE {
  # Tiene que ser una persona
  ?langile rdf:type foaf:person .
  # Puede vivir en un sitio o no
  OPTIONAL {
    ?langile gip_prop:bizilekua ?bizilekua
  }
}
```

Union

([5.rq](#))

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

SELECT ?entitatea ?izena

WHERE {
  {?entitatea rdfs:label ?izena }
  UNION
  {?entitatea foaf:name ?izena }
}
```

Ordenar resultados

([6.rq](#))

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

SELECT ?persona ?izena

WHERE {
  ?persona foaf:name ?izena
}

ORDER BY DESC (?izena) # Puede ser DESC o ASC
```

Limitar resultados

([7.rq](#))

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

SELECT ?persona ?izena

WHERE {
    ?persona foaf:name ?izena
}

ORDER BY DESC (?izena)
LIMIT 3
```

Paginación

([8.rq](#))

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

SELECT ?persona ?izena

WHERE {
  ?persona foaf:name ?izena
}

ORDER BY DESC (?izena)
LIMIT 3
OFFSET 3
```

Paginación

([9.rq](#))

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

SELECT ?persona ?izena

WHERE {
  ?persona foaf:name ?izena
}

ORDER BY DESC (?izena)
LIMIT 3
OFFSET 6
```


Filtrar resultados

([10.rq](#))

```
PREFIX gip_prop: <http://gipuzkoa.eus/prop/>

SELECT ?museoa ?langile_kop

WHERE {
    ?museoa gip_prop:kopurua ?langile_kop .
    FILTER (?langile_kop > "800"^^xsd:int)
}
```

Filtrar resultados

([11.rq](#))

```
SELECT ?langile ?izena  
  
WHERE {  
    ?langile foaf:name ?izena .  
    FILTER regex(?izena, 'Mi.*')  
}
```

Filtrar resultados

Lógica: !, &&, ||

Calculos: +, -, *, /

Comparaciones: =, !=, >, <

Filtrar resultados

Tests SPARQL: isURI, isBlank, isLiteral, bound

Acceder a datos: str, lang, datatype

Más: sameTerm, langMatches, regex, ...

Evitar duplicados

([12.rq](#))

```
SELECT ?lantokia
WHERE {
  ?person rdf:type foaf:person .
  ?person <http://vocab.data.gov/def/drm#worksFor> ?lantokia
}
```

```
SELECT DISTINCT ?lantokia
WHERE {
  ?person rdf:type foaf:person .
  ?person <http://vocab.data.gov/def/drm#worksFor> ?lantokia
}
```

Ask

([13.rq](#))

```
PREFIX gov:<http://vocab.data.gov/def/drm#>  
PREFIX gip:<http://gipuzkoa.eus/resource/>
```

ASK

```
WHERE {  
  ?person gov:worksFor gip:gugenheim  
}
```

Describe

DESCRIBE <<http://gipuzkoa.eus/resource/mikel-aranguren>>

CONSTRUCT

([14.rq](#))

```
PREFIX gip_prop:<http://gipuzkoa.eus/prop/>
PREFIX gip:<http://gipuzkoa.eus/resource/>
PREFIX gov:<http://vocab.data.gov/def/drm#>

CONSTRUCT {
  ?langile rdf:type gip:gugenheim_langilea
}
WHERE {
  ?langile gov:worksFor gip:gugenheim
}
```


Delete data

```
DELETE DATA {  
    rdf:type foaf:person  
}
```

```
DESCRIBE <http://gipuzkoa.eus/resource/aitor-labajo>
```

Delete

```
DELETE {?person rdf:type foaf:person}
WHERE {?person foaf:name ?name}
SELECT * WHERE {
    ?person rdf:type foaf:person
}
```

INSERT DATA

```
PREFIX gip:<http://gipuzkoa.eus/resource/>
```

```
INSERT DATA {
```

```
    gip:aitor-labajo rdf:type gip:hiritar
```

```
}
```

```
DESCRIBE <http://gipuzkoa.eus/resource/aitor-labajo>
```

INSERT

([15.rq](#))

```
PREFIX gip:<http://gipuzkoa.eus/resource/>

INSERT {
  gip:jon-alfaro rdf:type ?type .
}
WHERE {
  gip:aitor-labajo rdf:type ?type .
}
```

DESCRIBE <http://gipuzkoa.eus/resource/jon-alfaro>

Consultas federadas

([16.rq](#))

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX dbpedia_ont: <http://dbpedia.org/ontology/>
PREFIX gip: <http://gipuzkoa.eus/resource/>

SELECT ?poblacion_bilbo
WHERE {
  gip:bilbo owl:sameAs ?town
  SERVICE <http://dbpedia.org/sparql> {
    ?town dbpedia_ont:populationTotal ?poblacion_bilbo
  }
}
```

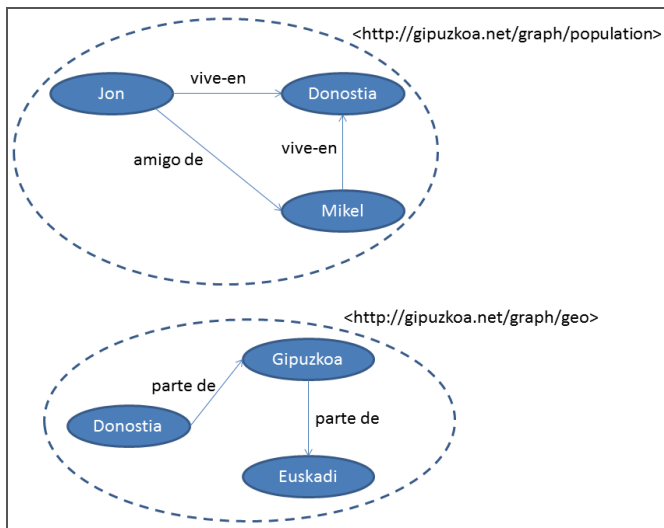
GRAFOS

Grafo: conjunto de triples

El conjunto entero se identificada con una URI (diferente de la de los datos)

Todas las Triple Stores tienen un Default Graph

GRAFOS

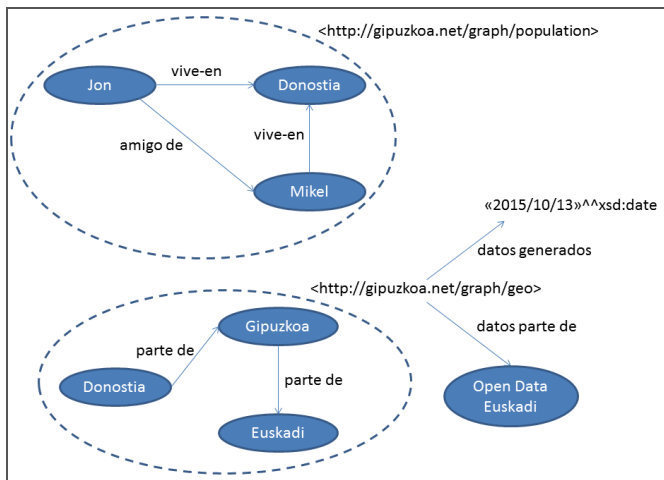


GRAFOS

Los grafos son muy utiles para añadir datos sobre los datos (metadatos): ej.
procedencia, autoria, fecha de generación

(entre otras cosas)

GRAFOS



GRAFOS

([17.rq](#))

```
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX ns: <http://example.org/ns#>

INSERT DATA{
  GRAPH <http://example/bookStore> {
    <http://example/book1> ns:price 42
  }
}
```

GRAFOS

```
SELECT *  
FROM <http://example/bookStore>  
WHERE {?s ?p ?o}
```

```
SELECT *  
FROM <http://example/bookStore>  
WHERE {?s ?p ?o}
```