Assignment 4 – group 12

By Freek van Tienen, Hylke Visser and Herman Banken

# Application

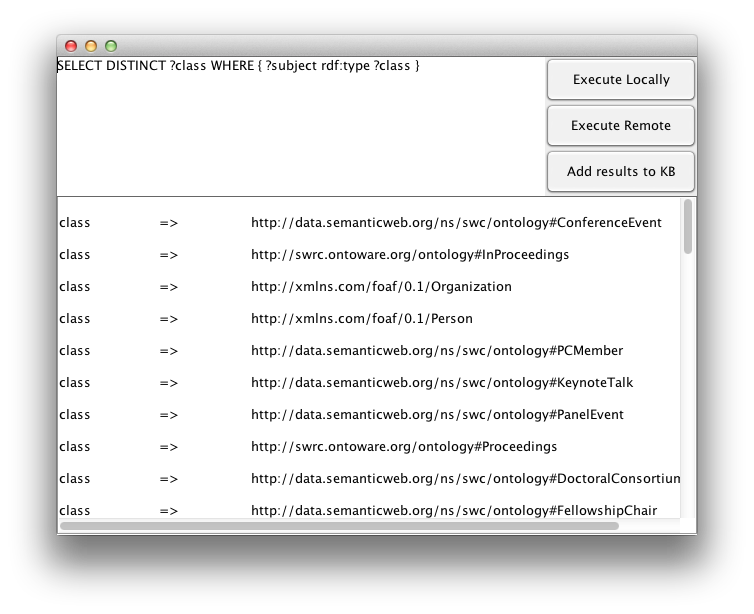


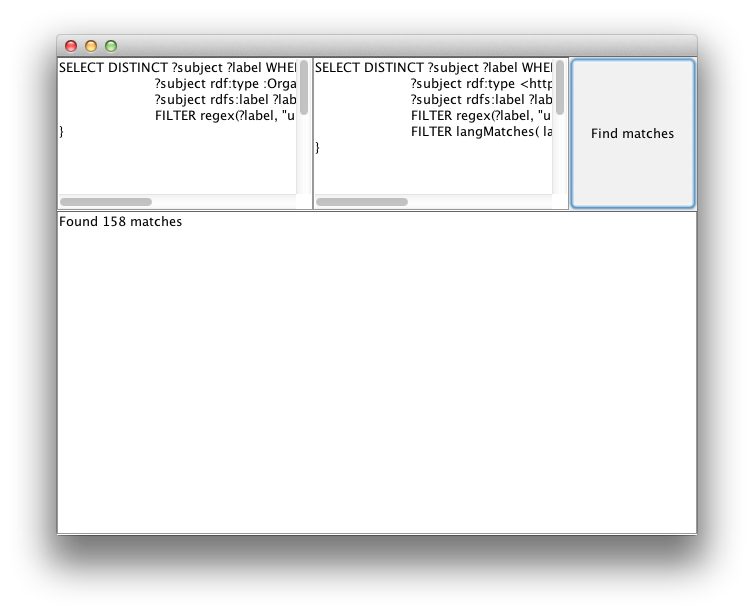
Figure 1: GUI for querying locally and remote

Figure 2: Linking tool. This tool System.out.println the links found when executing the remote and local query's after each other.

# Query’s

We have made a construct and a select part for each query. The construct query needs to be executed on the remote connection with DBPedia and before the select part and its response needs to be added to our knowledge base. We can do this with our tool.

## Query 1 : Select all persons in the dogfood dataset that are affiliated with U.S. Universities that have more than 100 000 students

### Remote query:

CONSTRUCT { ?university ?predicate ?object }

WHERE {

?university rdf:type <http://dbpedia.org/ontology/EducationalInstitution>.

?university <http://dbpedia.org/ontology/country> <http://dbpedia.org/resource/United\_States>.

?university <http://dbpedia.org/property/students> ?students.

FILTER ( ?students > 100000 ).

?university ?predicate ?object.

}

### Local query:

SELECT ?person

WHERE {

?university rdf:type <http://dbpedia.org/ontology/EducationalInstitution>.

?university <http://dbpedia.org/ontology/country> <http://dbpedia.org/resource/United\_States>.

?university <http://dbpedia.org/property/students> ?students.

FILTER ( ?students > 100000 ).

?localuniversity skos:exactMatch ?university .

?person swrc:affiliation ?localuniversity

}

### Result

person => <http://data.semanticweb.org/person/ying-ding>

### Description

This query shows all the persons that have spoken at the swc conferences and are affiliated with a university in the United States having more than 100000 students.

## Query 2: Select all persons who are affliated universities that are located in Indiana.

### Remote query:

CONSTRUCT { ?university ?predicate ?object }

WHERE {

?university rdf:type <http://dbpedia.org/ontology/University>.

?university <http://dbpedia.org/property/city> ?city.

?city rdf:type <http:/ /dbpedia.org/class/yago/CitiesInIndiana>.

?city <http://dbpedia.org/property/populationTotal> ?population.

FILTER ( ?population > 100000).

?university ?predicate ?object

}

### Local query:

SELECT ?person ?city

WHERE {

?person swrc:affiliation ?localuniversity.

?localuniversity skos:exactMatch ?university .

?university <http://dbpedia.org/property/city> ?city

}

### Result

person => http://data.semanticweb.org/person/ying-ding

city => http://dbpedia.org/resource/Bloomington,\_Indiana

### Description

This query shows all the persons that have spoken at the swc conferences and are affiliated within Indiana. Only one person, Ying Ding does so.

## Query 3: Select all persons who work in different countrys than the are based.

### Remote:

CONSTRUCT { ?university <http://dbpedia.org/property/country> ?country }

WHERE {

?university rdf:type <http://dbpedia.org/ontology/University>.

?university <http://dbpedia.org/property/country> ?country.

?university rdfs:label ?label.

FILTER regex(?label, "university" , "i")

}

### Local:

SELECT DISTINCT ?person

WHERE {

?person swrc:affiliation ?localuniversity.

?localuniversity skos:exactMatch ?university.

?person foaf:based\_near ?home.

?university <http://dbpedia.org/property/country> ?work.

FILTER( ?work != ?home).

}

### Results (some of them):

person => http://data.semanticweb.org/person/andreas-harth

person => http://data.semanticweb.org/person/eyal-oren

person => http://data.semanticweb.org/person/renaud-delbru

person => http://data.semanticweb.org/person/stefan-decker

person => http://data.semanticweb.org/person/uwe-assmann

person => http://data.semanticweb.org/person/rafael-penaloza

person => http://data.semanticweb.org/person/anni-yasmin-turhan

person => http://data.semanticweb.org/person/gwenn-englebienne

person => <http://data.semanticweb.org/person/jan-wielemaker>

### Description

Query 3 will first get the country’s the universities are located in. Then it will compare those locations to the based\_near property of the persons. One problem arised and it’s that dbpedia uses names like “Republic of Ireland” and swc uses “Ireland”. We did not find a property that corresponded so we let it this way, for now. A lot of people still lived in other places than they worked. Like Andreas Harth who works in Ireland but lives in Germany. Eyal Oren who works at the same place but lives in the Netherlands.

## Query 4: Select all movies that where put in theathers during swc 2010

CONSTRUCT { ?movie <http://dbpedia.org/ontology/releaseDate> ?date }

WHERE {

?movie rdf:type <http://dbpedia.org/ontology/Film>.

?movie <http://dbpedia.org/ontology/releaseDate> ?date.

FILTER (?date > "2009-01-01"^^xsd:date && ?date < "2010-01-01"^^xsd:date)

}

prefix xsd:<http://www.w3.org/2001/XMLSchema#>

SELECT ?ds ?de

WHERE {

?movie <http://dbpedia.org/ontology/releaseDate> ?date.

?conf ical:dtend ?end.

?conf ical:dtstart ?start.

?start ical:date ?ds.

?end ical:date ?de.

FILTER (?conf = "http://data.semanticweb.org/conference/www/2010").

FILTER (xsd:dateTime(?date) > xsd:dateTime(?ds) && xsd:dateTime(?date) < xsd:dateTime(?de) )

}

### Description

We had some problem converting dates so this query doesn’t actually run. But when executing parts of it you can see it will work when we get the conversion right. The xsd:dateTime is a function that should convert the 2 different date formats to the same dateTime format, but it didn’t work, yet.

## Query 5: Select all persons who work at universities that are located in Germany.

CONSTRUCT { ?university ?predicate ?object }

WHERE {

?university rdf:type <http://dbpedia.org/ontology/University>.

?university <http://dbpedia.org/property/country> <http://dbpedia.org/page/Germany>.

?university ?predicate ?object

}

SELECT ?person

WHERE {

?person swrc:affiliation ?localuniversity.

?localuniversity skos:exactMatch ?university .

?university <http://dbpedia.org/property/country> <http://dbpedia.org/page/Germany>.

}

-----------------------------------------------------------------------------

## Query 6: Select all universities that are older than 1900.

CONSTRUCT { ?university <http://dbpedia.org/property/established> ?year }

WHERE {

?university rdf:type <http://dbpedia.org/ontology/University>.

?university <http://dbpedia.org/property/established> ?year.

FILTER( ?year < 1800 )

}

SELECT ?university

WHERE {

?localuniversity skos:exactMatch ?university .

?university <http://dbpedia.org/property/established> ?year.

FILTER( ?year < 1900 )

}

### Result

university => http://dbpedia.org/resource/California\_State\_University

university => http://dbpedia.org/resource/University\_of\_North\_Carolina

university => http://dbpedia.org/resource/University\_of\_Notre\_Dame

university => http://dbpedia.org/resource/University\_of\_Evansville

university => http://dbpedia.org/resource/Indiana\_University\_School\_of\_Law\_%E2%80%93\_Indianapolis

university => http://dbpedia.org/resource/Butler\_University

university => http://dbpedia.org/resource/Marian\_University\_%28Indiana%29

university => http://dbpedia.org/resource/Indiana\_University\_School\_of\_Dentistry

university => http://dbpedia.org/resource/University\_of\_Saint\_Francis\_%28Indiana%29

### Description

This query returns all universities older than 1900. First we liked to show each person working at one of these universities but none of the people from swc works at one of the universities with a established attribute. So we took that out and look at the universities only now.