FREME call 24 April 2015, on e-Link

Time: 12 p.m. CEST
http://www.timeanddate.com/worldclock/fixedtime.html?iso=20150424T1000

Attendance:

Milan Dojchinovski	INFA
Jan Nehring	DFKI
Pieter Heyvaert	iMinds
Gerald Haesendonck	iMinds
Phil Ritchie	Vistatec
Marta Borriello	Vistatec

Regrets: Felix

Dial-in information:

Dial in via internet: https://global.gotomeeting.com/join/910806877

Dial in via phone:
- call +49 (o) 692 5736 7208
- access code: 910-806-877

Meeting-ID: 910-806-877

Agenda:
- data sets to be used in e-Link - concrete next implementation steps wrt to making data sets available via FREME. background presentation from Milan

Milan: With datasets do you want for start?
Phil: Use datasets from datahub.io
Jan: Check the table in Google Docs:
https://docs.google.com/spreadsheets/d/1a9xD31U4ZPbGZqCJQ0bHC_n1PyyIf2VAorIHjLCsWds/edit#gid=0
Milan: DBpedia and LinkedGeoData are required by three BCs so we should start with these. Everyone agrees.

Jan: They are a good start because they have a sparql endpoint and are therefore easy to use.

- e-Link design
  - just a sparql call for the first round?
  - conversion functionality, see e.g. tools linked here
  - sparql templating, see Phil's example
- deriving from the above: concrete implementation tasks for the 1st version of e-Link

Phil: I hope I can use e-Link as extended search. Two queries: Submit plaintext to e-Entity first. Then use these against a templated query.
We want to make recursive queries about the entities retrieved by e-Link.
Jan: We can construct one SPARQL query that fetches all these entities with one query.
Phil: e-Entity performs better on the whole document or on single sentences?
Milan: The context is important. So we need the whole document. Regarding e-Link the text / context is not important.
Phil: response times of queries against e-Link/e-entity
Pieter: LDF? performance should be better than SPARQL endpoints (under high load)
Jan: whether we query one endpoint or multiple endpoints, also how we define the endpoints - as part of the templates or the user defines these endpoints?
Phil: sparql endpoints defined as part of the templates - is better
Pieter: on federated queries

Discussion about query templates:

We discussed two ways of defining query templates:

1. Use "simple templates" like Felix suggested in
https://github.com/freme-project/technical-discussion/issues/8
Pro: Easy to understand
Pro: Easy to implement
Contra: This is our own proprietary solution and is not compatible to NIF
Contra: To use this with pipelines we would need to define filters / extractors that processes output of e-Entity and feeds this into e-Link.

A POST request to e-Link "simple templates" can look like this:

template with id=events_around_city: All events around ….(city)
POST /e-link/templates/events_around_city?distance=100
POST variables: var1=...&var2=...

2. Use smart template that are compatible to NIF.

E.g. the template has variables like @@@entity of class City@@@. e-Link then iterates through all NIF data submitted to it and extracts all entities of class city. Then it executes the query template once for each entity. This solution also requires variables as in the “simple templates” to define things like: “Give me all events around a city within a distance of 100 miles” (the 100 miles part)

A POST request to e-Link ”smart templates” can look like this:

template with id=events_around_city: All events around ….(city)

POST /e-link/templates/events_around_city/
POST variables: NIF in RDF
Output: will be also NIF with the enrichments

Pro: This is compatible to NIF
Pro: This is compatible to pipelines
Pro: This is easy to use because user does not have to process the NIF coming from e-Entity.
Contra: There can be ambiguities. E.g. if the template contains two entites of class city then it is hard to put the entities in the right place. We could add a new variable in NIF to overcome this issue, like the last line in this example:

```
// Detected entity mention linked with its DBpedia representation and ontology class.
<http://example.org/document/1#char=14,20>
   a nif:String , nif:RFC5147String , nif:Word, nif:Phrase ;
   nif:referenceContext <http://example.org/document/1#char=0,3680> ;
   nif:anchorOf "Xiamen"""^^xsd:string ;
   nif:beginIndex "14"^^xsd:nonNegativeInteger ;
   nif:endIndex "20"^^xsd:nonNegativeInteger ;
   itsrdf:taIdentRef  <http://dbpedia.org/resource/Xiamen> .
   itsrdf:taClassRef  <http://dbpedia.org/ontology/City> ;
   e-link:id “variable1” ;
```

Contra: We need to define our own syntax how to define the “selectors” (like @@@entity of class city@@@)

ACTION: INFAI to start working on the first e-Link prototype.