



Universität Freiburg
Institut für Informatik
Prof. Dr. G. Lausen
Alexander Schätzle
Martin Przyjaciel-Zablocki

Georges-Köhler Allee, Geb. 51
D-79110 Freiburg
lausen@informatik.uni-freiburg.de
schaetzle@informatik.uni-freiburg.de
zablocki@informatik.uni-freiburg.de

Data Models and Query Languages Summerterm 2013

7. Exercise Sheet: SPARQL 1.1 & nSPARQL

Discussion: 09.07.2013

Course evaluation: We ask all participants of the course to take part in the survey. The system is open till Friday, July 19th 2013, at the latest. You can access it at:
http://www.informatik.uni-freiburg.de/~welte/lehrevaluation/ss2013/webseite_links_en.html

Exercise 1 (SPARQL 1.1: Aggregations, Subqueries, Explicit Negation)

Consider the RDF document that models a social graph in a music domain:

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix lb:   <http://example.org/lastfm/> .
lb:bob      foaf:knows      lb:user2, lb:user3, lb:user4 ;
            foaf:age       25 ;
            lb:listenedTo  lb:track1, lb:track2, lb:track2 ;
            lb:topArtist   lb:artist1, lb:artist2 .
lb:user2    foaf:knows      lb:user5, lb:user6 ;
            foaf:age       40 ;
            lb:listenedTo  lb:track1, lb:track2, lb:track3, lb:track3 ;
            lb:topArtist   lb:artist2, lb:artist4 .
lb:user3    foaf:knows      lb:user5, lb:user6 ;
            foaf:age       19 ;
            lb:listenedTo  lb:track2, lb:track3, lb:track3, lb:track4 ;
            lb:topArtist   lb:artist2, lb:artist3 .
lb:user4    lb:listenedTo  lb:track2, lb:track3, lb:track4 ;
            foaf:age       61 ;
            lb:topArtist   lb:artist3, lb:artist4, lb:artist5 .
lb:user5    foaf:knows      lb:user7 ;
            foaf:age       23 ;
            lb:topArtist   lb:artist1, lb:artist3 .
```

Formulate the following requests as SPARQL 1.1 queries. Evaluate them on the given RDF document and write down the final result.

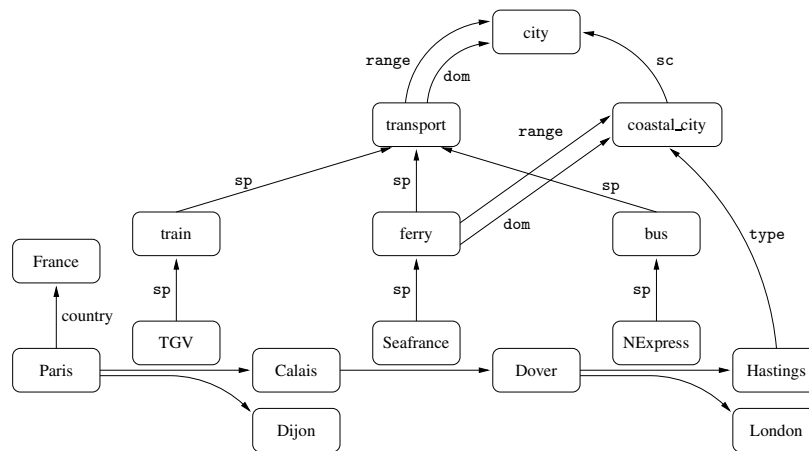
- Find all people that are connected to user bob via an arbitrary Friend-of-a-Friend path. Use SPARQL 1.1 Property Paths to express paths of arbitrary length.
- Compute the average age of all people.

- c) What are the top-2 track recommendations for Bob based on the listening history of his friends? A recommendations can be derived from the tracks, which were listened by Bobs friends. The ranking should consider how often a track was played and exclude already known tracks. This query should exploit negation and aggregation in SPARQL 1.1.
- d) Who are the most popular artists with an average fan age above 30 years? The popularity of an artist can be estimated by counting the number of users that refer to him as top artist. The query should use a subquery to determine those artists with an average fan age above 30 years.

Verify your results using the ARQ engine.

Exercise 2 (SPARQL Property Paths, nSPARQL)

Consider the RDF Graph [1]:



Evaluate the following nSPARQL expressions on the given RDF graph and write down the final result. Furthermore, formulate the nSPARQL expressions as SPARQL property path queries if possible or explain why they are not expressible.

- a) $P_1 = (?x, (next :: TGV \mid next :: Seafrance)^+, Dover) \text{ AND } (?x, next :: country, ?y)$
- b) $P_2 = (?x, (next :: TGV \mid next :: Seafrance)^+, Dover) \text{ OPT } (?x, next :: country, ?y)$
- c) $P_3 = (?x, (next :: Seafrance \mid next :: NExpress)^+ / self :: [next :: NExpress / self :: London] / (next :: Seafrance \mid next :: NExpress)^+, ?y)$
- d) $P_4 = (?x, next :: [(next :: sp)^* / self :: transport], ?y)$
- e) $P_5 = (?x, (next :: [(next :: sp)^* / self :: transport])^+, ?y)$
- f) $P_6 = (?x, (trans(transport) / self :: [trans(type) / self :: coastal_city])^+, ?y)$

References:

1. Jorge Prez, Marcelo Arenas, Claudio Gutierrez: nSPARQL: A Navigational Language for RDF, 7th International Semantic Web Conference, 2008.