SEWELIS: Reconciling Expressive Querying and Exploratory Search

Sébastien Ferré
Team LIS, Data and Knowledge Management, Irisa

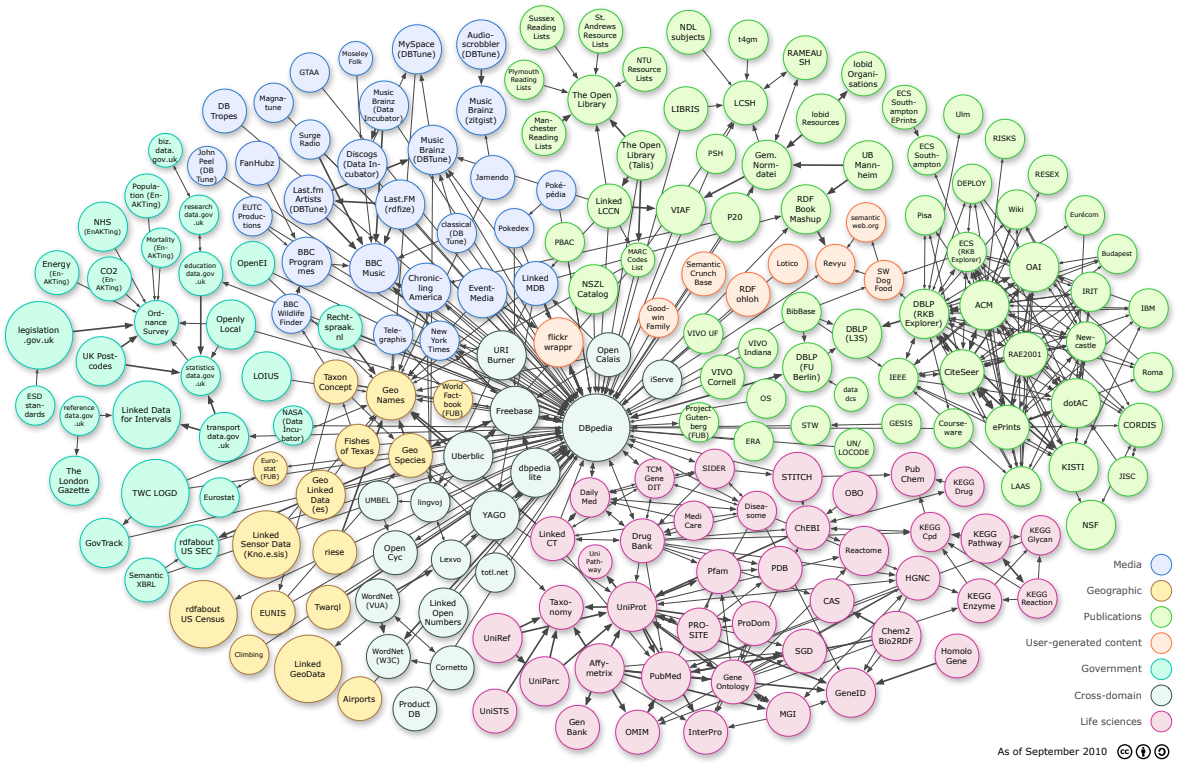
Atelier CrEDIBLE, 15-17 octobre 2012, Nice

INSTITUT DE RECHERCHE EN INFORMATIQUE ET SYSTEMES ALÉATOIRES

UMR IRISA LIS

The Web of Data

- How to search and explore RDF graphs ?
- How to fill the gap between end users and formal languages ?



Linking Open Data cloud diagram, by Richard Cyganiak and Anja Jentzsch. http://lod-cloud.net/

Query-based approaches

- Query languages: e.g., SPARQL
  - very expressive but difficult to use
  - no guiding, possibly empty results
- NLP interfaces: e.g., NLP-Reduce
  - easier to use but less precise (ambiguities)
  - no guiding, possibly empty results
- Guided query composition: e.g., Ginseng (Controlled English), Semantic Crystal (graphical)
  - ensures correct syntax and vocabulary, but does not avoid empty results
  - no feedback: no result before the query is complete
  - no way to navigate from one query to another (refinements)

Navigation-based approaches

- Graph navigation: e.g.: Disco, Tabulator, Semantic wikis
  - RDF triples as labelled hyperlinks
  - only one resource at a time
- Faceted Search: e.g., Ontogator, BrowseRDF, SlashFacet
  - guided navigation to selections of resources
  - limited expressiveness compared to SPARQL

Limits of set-based faceted search

Why faceted search has a limited expresiveness ?

- because set-based: $S_{t+1} = f(S_t)$
  - S_t: selection at step t (a set of items)
  - f: set-based operations with atomic selections (R_i) and relations (p_i)
  - operations: intersection, union, difference, relation crossing
- lack of flexibility: fixed ordering of navigation steps
- lack of expressiveness: unreachable selections
  - unions of complex selections: $(R_1 \cap R_2) \cup (R_3 \cap R_4)$
  - intersection of crossings from complex selections: $p_1(., R_1 \cap R_2) \cap p_2(., R_3 \cap R_4)$

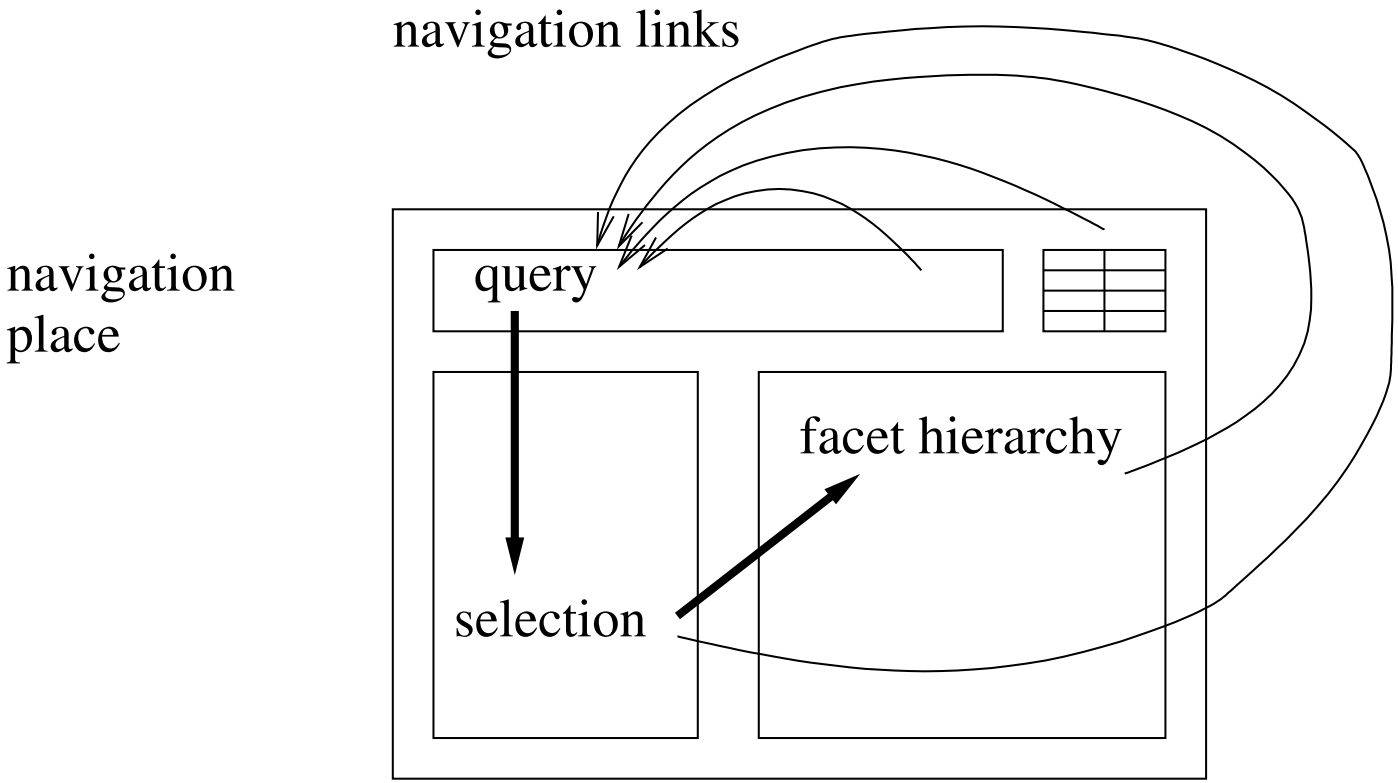
Query-based Faceted Search

Reconciling querying and navigation:

- query-based navigation: $q_{t+1} = f(q_t)$
  - q_t: query at step t with a distinguished subquery (focus)
  - $S_t = items(q_t)$: set of answers of the query
  - f: query transformations with atomic queries (f_i)
  - operations: conjunction, disjunction, negation, existential restrictions
- several navigation paths to a same query
- reaching the unreachable selections
  - $(f_1$ and $f_2)$ or $\underline{f_3}$ $\xrightarrow{\text{and } f_4}$ $(f_1$ and $f_2)$ or $(f_3$ and $\underline{f_4})$
  - $p_1 : (f_1$ and $f_2)$ and $p_2 : \underline{f_3}$ $\xrightarrow{\text{and } f_4}$ $p_1 : (f_1$ and $f_2)$ and $p_2 : (f_3$ and $\underline{f_4})$

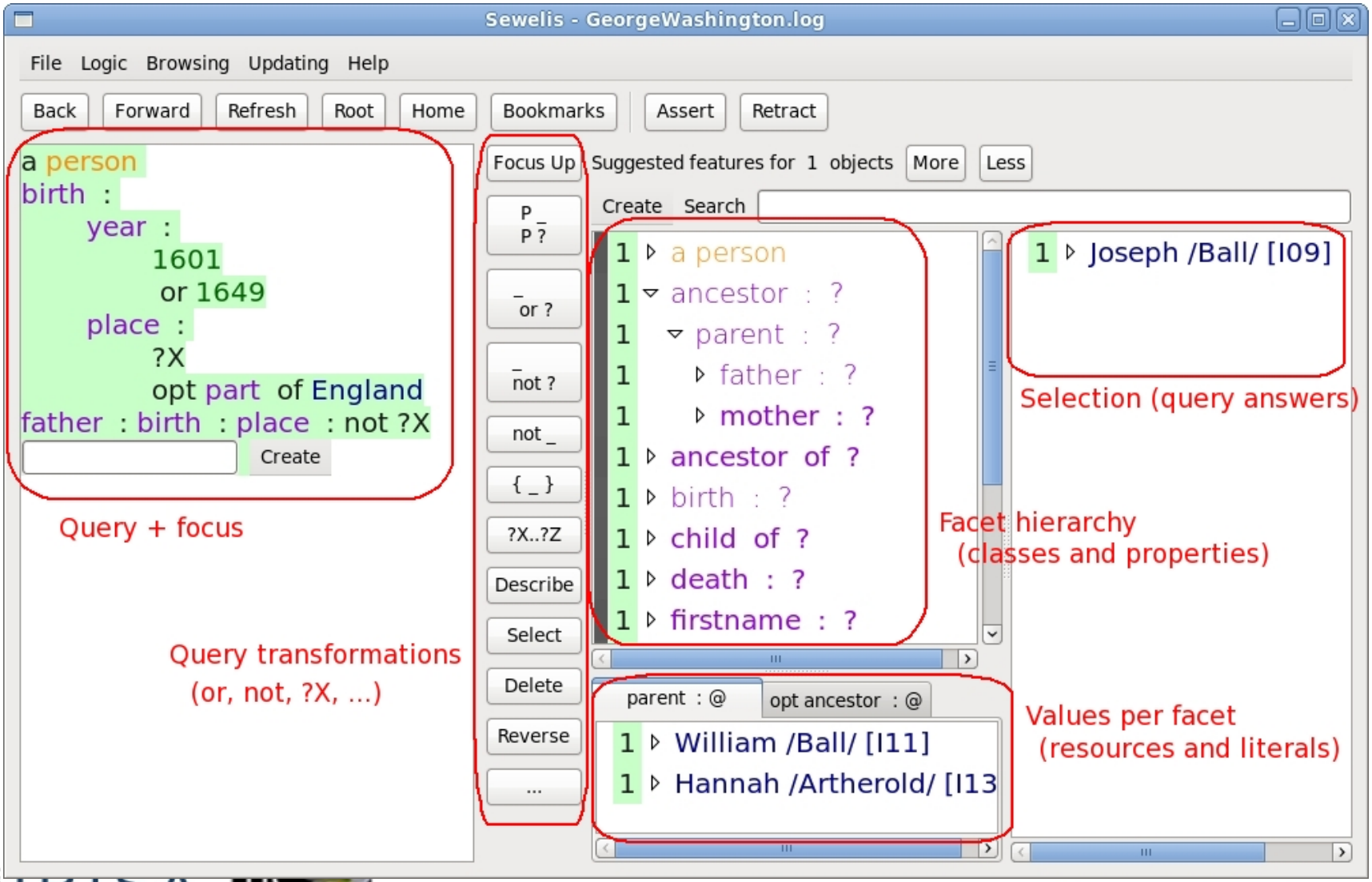
Query-based Guided Navigation

A schema for the navigation graph and the user interface.



Navigation links are query transformations

User Interface: a Screenshot of Sewelis



LISQL: the Sewelis Query Language

The LISQL syntax reflects query transformations

- *q* `and` *q* / *q* `or` *q* / `not` *q* / *p* `:` *q* / *p* `of` *q* + `?x`
- `a person and birth : (year :(1601 or 1649) and place :(?X and part of England)) and father : birth : place : not ?X`
  - *which person was born in 1601 or 1649 at some place X in England, and has a father born at a place that is not X*
- same query with focus on `?X and in England`
  - *at which place (X) in England, a person was born in 1601 or 1649, and the father of this person was not born*
- equivalent SPARQL query (7 variables)

```
SELECT ?x WHERE { ?p a person.  ?p birth ?b.
?b year ?y FILTER (?y=1601 || ?y=1649).  ?b
place ?x.  ?x in England.  ?p father ?f.  ?f
birth ?fb.  ?fb place ?fl FILTER ?fl != ?x }
```


LISQL: the Sewelis Query Language

The LISQL syntax reflects query transformations

- q `and` q / q `or` q / `not` q / p `:` q / p `of` q + `?x`
- `a person and birth : (year :(1601 or 1649) and place :(?X and part of England)) and father : birth : place : not ?X`
  - *which person was born in 1601 or 1649 at some place X in England, and has a father born at a place that is not X*
- same query with focus on `?X and in England`
  - *at which place (X) in England, a person was born in 1601 or 1649, and the father of this person was not born*
- equivalent SPARQL query (7 variables)

```
SELECT ?x WHERE { ?p a person.  ?p birth ?b.
?b year ?y FILTER (?y=1601 || ?y=1649).  ?b
place ?x.  ?x in England.  ?p father ?f.  ?f
birth ?fb.  ?fb place ?fl FILTER ?fl != ?x }
```

LISQL: the Sewelis Query Language

The LISQL syntax reflects query transformations

- *q* `and` *q* / *q* `or` *q* / `not` *q* / *p* `:` *q* / *p* `of` *q* + `?`*x*
- `a person and birth : (year :(1601 or 1649) and place :(?X and part of England)) and father : birth : place : not ?X`
  - *which person was born in 1601 or 1649 at some place X in England, and has a father born at a place that is not X*
- same query with focus on `?X and in England`
  - *at which place (X) in England, a person was born in 1601 or 1649, and the father of this person was not born*
- equivalent SPARQL query (7 variables)

```
SELECT ?x WHERE { ?p a person.  ?p birth ?b.
?b year ?y FILTER (?y=1601 || ?y=1649).  ?b
place ?x.  ?x in England.  ?p father ?f.  ?f
birth ?fb.  ?fb place ?fl FILTER ?fl != ?x }
```

The Facet Hierarchy

- used as a dynamic index of the selection *items*(q)
- atomic queries f: variables in q, classes, properties
- restricted to relevant elements
  - *items*(q[and f]) $\neq \emptyset$
- organized according to class/property hierarchies:
  - `?X`
  - `a person`
    - `a man`
    - `a woman`
  - `parent : ?`
    - `father : ?`
    - `mother : ?`
  - `parent of ?`
  - ...

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

A Navigation Scenario

1. ?
2. a person
3. a person and birth : year : ?
4. a person and birth : year : 1601
5. a person and birth : year : (1601 or ?)
6. a person and birth : year : (1601 or 1649)
7. a person and birth : (year : (1601 or 1649) and place : ?)
8. a person and birth : (year : (1601 or 1649) and place : ?X)
9. a person and birth : (year : (1601 or 1649) and place : (?X and part of England))
10. a person and birth : (. . .) and father : birth : place : ?
11. a person and birth : (. . .) and father : birth : place : not ?
12. a person and birth : (. . .) and father : birth : place : not ?X

Theoretical Results

- safeness: no navigation path leads to a dead-end
  - except for some focus changes through negation and disjunction
- completeness: there is a navigation path for every LISQL query
  - guaranteed if it has no unsafe focus change
- efficiency: equivalent to set-based faceted search
  - + query answering when focus change
  - + query answering for each variable in q_t (most often 0)

User Evaluation

- 20 students (from IFSIC and INSA Rennes)
- dataset: genealogy of George Washington (70 persons)
- 18 questions of increasing difficulty
  - property chains, negation, disjunction, variables
  - the number of navigation steps ranges from 0 to 12
- results
  - all answered correctly to $\geq$11/18 questions
  - 8/20 answered correcty to $\geq$16/18 questions
  - the average time spent on the test is 40min ([21,58]min)
  - for each category of question, $\geq$18/20 answered correctly to at least one question of the category
  - for most categories, success rate and response time improve on 2nd and 3rd queries

Some Questions of the Study

1. Which man was born in 1659 ?
   `a man and birth : year : 1659`
2. Which man is married with a woman born in 1708 ?
   `a man and married with (a woman and birth : year : 1708)`
3. Which women have for mother Jane Butler or Mary Ball ?
   `a woman and mother : (Jane or Mary)`
4. How many women have a mother whose death's place is not Warner Hall ? `a woman and mother : death : place : not Warner Hall`
5. Who died in the same area where they were born ?
   `a person and death : place : part of ?X and birth : place : part of ?X`

Demo

- dataset from DBpedia (imported as RDF)
  - a selection of films (120), people (396), and countries (37)
- Exploration
  1. *films directed by Tim Burton and starring Johnny Depp and Helena Bonham Carter* (standard faceted search)
  2. *films released in 2000-2010 whose director was born in an english-speaking country* (property path, `or`)
  3. *films related to France... or not* (general `or`, `not`)
  4. *people born in the US, and director of a film starring Johnny Depp and released after 2000* (property tree)
  5. *people being both a director and an actor, in the same film* (equality/property cycle)
  6. *films from some country, whose director was born in another country* (inequality)
- Edition
  7. *adding the film “Charlie and the chocolate factory”*

Conclusion

We have shown that Query-based Faceted Search

- can be used on RDF graphs
- with an expressive SPARQL-like query language
- where users can entirely rely on navigation
- without ever falling in dead-ends
- after a short training stage

Current Work

- UTILIS: Guided Creation and Update of RDFS Data [PhD Alice Hermann]
  - part of SEWELIS, same UI for querying and updating
  - query $\rightarrow$ description, answers $\rightarrow$ similar objects
  - similarity based on query/description relaxation
- PEW: Possible World Explorer for OWL Ontologies Understanding and Enrichment [with Sebastian Rudolph]
  - adaptation of SEWELIS UI on top of OWL API/HermiT
  - query $\rightarrow$ class expression, answers $\rightarrow$ "possible worlds"
  - works on ontologies without instances
  - applied to discover and eradicate unwanted "possible worlds", by adding an axiom that is the negation of the class expression
  - a pizza that has no topping $\Rightarrow$ every pizza has a topping

Current Work

- UTILIS: Guided Creation and Update of RDFS Data [PhD Alice Hermann]
  - part of SEWELIS, same UI for querying and updating
  - query $\rightarrow$ description, answers $\rightarrow$ similar objects
  - similarity based on query/description relaxation
- PEW: Possible World Explorer for OWL Ontologies Understanding and Enrichment [with Sebastian Rudolph]
  - adaptation of SEWELIS UI on top of OWL API/HermiT
  - query $\rightarrow$ class expression, answers $\rightarrow$ "possible worlds"
  - works on ontologies without instances
  - applied to discover and eradicate unwanted "possible worlds", by adding an axiom that is the negation of the class expression
  - a pizza that has no topping $\Rightarrow$ every pizza has a topping

Future Work

- Guided composition of workflows - application to bioinformatics [PhD Mouhamadou Ba, with GenOuest platform]
  - description of (bioinformatic) tasks: inputs, outputs, etc.
  - description of workflows as combinations of tasks
  - guidance based on existing tasks and workflows
- Scaling of SEWELIS navigation on top of SPARQL endpoints [MSc Joris Guyonvach]
  - rely on a SPARQL endpoint to compute answers and suggestions
  - analyse trade-offs between expressiveness, accuracy of suggestions, and efficiency
  - dataset preprocessing vs on-demand computation

Thank you for your attention!

Questions ?

SEWELIS at
`http://www.irisa,fr/LIS/softwares/sewelis/`

What is a Semantic dataset

A semantic dataset is a RDF graph:

- nodes are resources
  - URIs: Universal Resource Identifiers
    - can denote anything: objects, people, places, classes, properties, datatypes
  - literals: concrete values such as strings, dates, etc.
  - blank nodes: anonymous entities
- edges are triples (subject, predicate, object)
  - subject: a URI
  - predicate: a property URI (a resource itself)
  - object: a URI or a literal

Example of a RDF Graph

Some data about Georges Washington, including part of the schema, and meta-schema.