Consuming multiple sources of Linked Data: Challenges & Experiences

Ian Millard, Hugh Glaser, Manuel Salvadores, Nigel Shadbolt

8th November 2010
But where are all the apps?

• Continued growth in the quantity of Linked Open Data
  – Particularly government & public sector info

• But has Linked Data had any impact on Joe Public?

• What about the promises of data aggregation & interoperability?

• It is still hard to use Linked Data in real applications
  – especially when using multiple datasets
Challenge 1: Co-reference

• Lots of data in the 'cloud'

• Lots of duplication

• Relatively few links
  – the last, often overlooked step?

• However there are a variety of tools and frameworks which are now beginning to address these issues
The Web of Data has many equivalent URLs. This service helps you to find co-references between different data sets. Enter a known URI, or use Sindice to search first.

Search results from Sindice, with co-references applied...


Show 44 more

48.
Challenge 2: heterogeneity of vocabularies

• As the cloud has grown, so to have the number of emerging vocabularies used to model the structure of that data

• Starting to see some convergence
  – but how many ways to describe a book, journal article or a place?

• Automated ontology alignment / mapping has been a research topic for many years
  – but on-the-fly translation services are not readily available to easily facilitate data interoperation
Challenge 3: Discovery of resources

• Finding data in LOD Cloud is hard
  – Index of the Cloud?
  – Search engines?

• Even if we have a known triple pattern, there can be issues of asymmetry
Challenge 3: Discovery of resources

• Finding data in LOD Cloud is hard
  – Index of the Cloud?
  – Search engines?

• Even if we have a known triple pattern, there can be issues of asymmetry

\[ \text{foaf:knows} \]
Challenge 3: Discovery of resources

• Finding data in LOD Cloud is hard
  – Index of the Cloud?
  – Search engines?

• Even if we have a known triple pattern, there can be issues of asymmetry
Challenge 3: Discovery of resources

- voiD documents describe datasets

- Effort to collect sets of descriptions into a repository or 'voiD store'

- Enables many useful discovery services

- CKAN

- Back-link services, search engines
Challenge 4: Using multiple datasets

• Example – find coordinate location of users
  
  51.508056  -0.124722
Challenge 4: Using multiple datasets

- Example – find coordinate location of users

```
SELECT ?lat ?lng WHERE {
  <joe> eg:lives_in ?place .
  ?place geo:lat ?lat .
  ?place geo:long ?lng
}
```

```
lives in

<london>

51.508056  -0.124722
```
Challenge 4: Using multiple datasets

• Example – find location of users with foaf profiles

```
foaf:based_near
<london>
```

data.semanticweb.org

```
51.508056  -0.124722
```

dbpedia.org
Related Work: SemWeb Client Library

• URI resolution based approach to answering queries across the Web of Data

• Given one or more bound predicates in a query, the required URIs are resolved and cached into a local store before the query is then executed

  + can answer almost any query, incl multiple datasets

  – performance can be very slow, can incur large amounts of redundant data retrieval and processing
Related Work: DARQ

- Distributed SPARQL query engine
- Accesses known endpoints directly, breaking down query, executing part-by-part, handling result joins
  
  + simple queries can sometimes be executed efficiently
  
  - requires detailed statistical information about each predicate for every endpoint to be compiled before queries can be made
  
  - round-robin approach where repositories share common predicates does not scale well
RKB Explorer: Overview

• Application with simple user interface to help researchers highlight and discover new relationships in the field of Resilient Systems and Dependable Computing

• Many data sources, one of the first applications to try and fully embrace a distributed data model – each held in a separate LOD/SPARQL store, each with a CRS

• Hybrid query approach utilising combination of SPARQL, co-reference expansion, and URI resolution
RKB Explorer: Query Heuristic

- All SPARQL queries fed through a middleware layer which employs very simple heuristic for best effort results
  - If all bound subjects and objects originate from a single known dataset with available SPARQL endpoint, execute against endpoint directly
  - Else resolve all bound URIs into local cache repository then execute query over that endpoint
- Originally used manual configuration, can now use voiD store to discover appropriate datasets/endpoints
RKB Explorer: CoP Engine

- “Community of Practice” usually refers to group of related people, often with similar interests

- RKB Explorer computes associated groups of resources of a particular type related to a specific input resource, e.g., find papers related to this person

- Pairwise source_type/target_type configuration files, akin to rules specifying the important features relating instances of those two types of resource

- Each “rule” is expressed in at most two query stages, combined with sameAs expansion
RKB Explorer: CoP Query Example

• Find other papers related to a given article, based upon commonality of author(s)

doCOP(

    "<$targetURI> eg:hasAuthor ?intermediate" ,
    "?result eg:hasAuthor <$intermediate>" ,
    1
)


$target$
$target$
CoP Engine: Summary

- Not solved generic distributed query problem yet!
- Two-phase execution with sameAs expansion of intermediate results allows a degree of execution over multiple sources
  - Need to bear limitations in mind with authoring
- Careful summation of results (again, co-reference issues)
- Mostly simple SPARQL queries, executed efficiently against appropriate endpoint(s)
CoP Engine: Future work

- Would like to relax constraint of two-phase approach to enable arbitrary queries to be processed
  - Then faced with similar problems to DARQ
  - Work on rdfstats, and next version of voiD introducing better statistical information
  - Heuristic metrics based on evaluating commonly occurring predicates over typical datasets
- Already extensive low-level caching; further investigation
- May benefit by threading CoP engine execution
Conclusions

- Exciting growth in Linked Open Data
  - Government, PSI, Life sciences
- However still number of hurdles wrt ease of use
  - Coreference, vocabularies, discovery, query
- Summarised how RKB Explorer addresses these
  - CRS, mapping, voiD store, hybrid CoP engine
- Still important work to be done in enabling applications to easily use full potential of the Web of Data
Thanks. Any questions?

http://sameAs.org

http://rkbexplorer.com

http://schooloscope.com

This work has been supported with finance and time by many projects, organisations and people over the years, most recently through the EnAKTing project