Querying over Federated SPARQL Endpoints -A State of the Art Survey(1)

(DERI Technical Report 2013-06-07)

- http://www.deri.ie/sites/default/files/publications/1306.1723v1.pdf

Table I: The Existing Frameworks Support SPARQL 1.1 Federation Extension.

Framework	Platform	SERVICE	BINDINGS	VALUES
ARQ	Jena	✓	✗	✓
SPARQL-FED	Virtuoso	✓	✗	✓
Sesame	Sesame	✓	✓	✓
SPARQL-DQP	OGSA-DAI and OGSA-DQP	✓	✓	✗

(2)

Table II: The Existing Frameworks Supports Federation over SPARQL Endpoints without reformulating query to SPARQL 1.1.

Framework	Catalogue	Platform	Source Selection	Cache	Query Execution	Source Tracking	GUI
DARQ	Service Description	Jena	Statistic of Predicate	✓	Bind Join or Nested Loop Join	Static	✗
ADERIS	Predicate List during setup phase	✗	Predicate List	✗	Nested Loop Join	Static	✓
FedX	✗	Sesame	ASK	✓	Bind Join parallelization	Dynamic	✓
Splendid	VoID	Sesame	Statistic + ASK	✗	Bind Join or Hash Join	Static	✗
GDS	Service Description	Jena	Statistic of Predicate	✓	Bind Join or Semi Join	Dynamic	✗
Avalanche	Search Engine	Avalanche	Statistic of predicates and ontologies	✓	Bind join	Dynamic	✗
Distributed SPARQL	✗	Sesame	✗	✗	Bind join	✗	✗

(3)

Table III: The Existing Frameworks Supports Federation over SPARQL Endpoints, Reformulate query to SPARQL 1.1.

Framework	Catalogue	Platform	Source Selection	Cache	Query Execution	Source Tracking	GUI
SemWIQ	RdfStats+VoID	Jena	Statistic + Service	✓	Bind Join	Dynamic	✓
Anapsid	Predicate List and Endpoint status	Anapsid	Predicate List	✗	Symmetric Hash Join and XJoin	Dynamic	✓
WoDQA	VoID Stores	Jena	List of predicates and ontologies	✗	✗	Dynamic	✓

DARQ(1)

- Last update: 2006 http://darq.sourceforge.net/
- License: as per Jena
- Source code is available through SVN

  svn checkout svn://svn.code.sf.net/p/darq/code/ darq-code
- **Limitations and known issues**
  - DARQ is in an early stage of development. No extensive testing has been performed.
  - Since DARQ uses predicates to decide where to send triple patterns, no ?s ?p ?o queries are supported - **predicates must be bound**
  - Joins using **Blank nodes** are not supported. If a join operation finds a blank node an exception will be thrown.
  - DESCRIBE is not supported.
  - GRAPH is not supported.
  - Query optimization is not very performant and will not support many endpoint or triples. #endpoints, #triples/endpoint < 7 are a good choice for the moment.

(2)

- Local configuration: The list of SPARQL endpoint ; service description
- Service description is local configured, and it is difficult to find an endpoint with service description that can be publicly open accessed

FedX(1)

FedX 3.0

- http://www.fluidops.com/fedx/ ISWC 2011
- Licensed under the GNU Affero General Public License (AGPL) for use in open source applications. Offer alternative license terms upon request for proprietary, closed source applications, and other commercial applications
- Source code available

(2)

- 1. Simple configuration

  The list of SPARQL endpoint , query file and optional JVM, binding block size configuration
- 2 The problems mainly comes from the changing SPARQL address and endpoint server interface

SPLENDID(1)

- ISWC 2011
  http://code.google.com/p/rdffederator/
- **Code licence**

  GNU Lesser GPL
- **Source available**

  svn checkout http://rdffederator.googlecode.com/svn/trunk/ splendid

  including a voiD statistics generator

(2)

- Automatic source selection and query optimization is based on statistical information provided by *VOID* descriptions.
- For triple patterns with unbound predicates: assign all data sources as there is no further information available from the VOID descriptions
- For triple patterns with bound variables which are not covered in the VOID statistics:

  send a *SPARQL ASK* query including the triple pattern to all pre-selected data sources and remove sources which fail the test

(3)

```
PREFIX owl:     <http://www.w3.org/2002/07/owl#>
PREFIX dbpedia:  http://dbpedia.org/resource/
SELECT ?predicate ?object WHERE {
  { dbpedia:Barack_Obama ?predicate ?object }
 UNION
  { ?subject owl:sameAs dbpedia:Barack_Obama .
    ?subject ?predicate ?object }
}
```

1. Without the limitation of "Predicate bound"
2. Configuration:

   SPARQL endpoints and VOID (local)
3. VOID generator. It took a long time. 53 minutes to deal with 10 million Allie data

ADERIS(1)

- ADERIS: An Adaptive Query Processor for Joining Federated SPARQL Endpoints

http://rd.springer.com/chapter/10.1007%2F978-3-642-25106-1_28

- A Hybrid Approach to Linked Data Query Processing with Time constraints

http://events.linkeddata.org/ldow2013/papers/ldow2013-paper-07.pdf

- Make use of both link traversal-based and distributed query processing-based approaches
- Time constraints, the tradeoff with result completeness

(2)

- On the Move to Meaningful Internet Systems: OTM 2011
  http://code.google.com/p/sparql-aderis/
- **Code licence** Apache License 2.0
- SPARQL-ADERIS is a research prototype developed at the National Institute of Advanced Science and Technology (AIST) Japan.
  http://code.google.com/p/sparql-aderis/downloads/list
- http://aderis.linkedopendata.net/index.html

It is unkonwn that the result is from distributed processing or deference URI
Required to specify SPARQL endpoint
Unknown how the SPARQL endpoint list influence the result