Collaborative development of cross-database Bio2RDF queries

Peter Ansell
Microsoft Queensland University of Technology
eResearch Centre

Linked Data

1) Use URIs as names for things
2) Use HTTP URIs so that people can look up those names.
3) When someone looks up a URI, provide useful information, using the standards (RDF, SPARQL)
4) Include links to other URIs, so that they can discover more things.

http://www.w3.org/DesignIssues/LinkedData.html
Linked Data querying

- **Strategy 1 (Naive):**
  - Retrieve resources
  - Retrieve linked resources
  - Cache them locally and perform queries

- **Strategy 2 (Search engine):**
  - Retrieve resources
  - Retrieve directly linked resources
  - Query a semantic search engine for related resources
  - Cache them locally and perform queries
Linked Data querying

- Strategy 3 (Distributed query):
  - Mix SPARQL endpoint queries with URI based resolution to avoid having a large local cache
  - Normalise results from each site to form final query result
- Strategy 4 (Federated query):
  - Find SPARQL endpoints for a single SPARQL query
  - Attempt to join SPARQL results across the network using statistics to minimise the resources transferred
  - Requires single URI for any particular resource across all endpoints

Bio2RDF URI principles

- Normalise URI's down to their basic information
  - Namespace
    - Publicly created to represent a section of knowledge
    - May have more than one provider and URI structure, but needs to have unique private identifiers across all providers
  - Identifier
    - Privately created to represent a resource within a namespace
    - Minimal identifying string for a unique resource in a namespace
Bio2RDF distributed queries

- Assign Namespaces to providers
  - SPARQL endpoints may have one or more graphs related to a namespace
  - URI based resolution may have a common prefix that corresponds to a namespace, even if the URI's on the endpoint are not normalised Bio2RDF URI's
- Query across relevant providers given a users query
- Aggregate all results into a single RDF document and return to the user

Customised queries

- A given input string may match more than one query definition and more than one namespace
  - For example, label/go:0000345 attempts to get any available labels for the GO term 0000345 (cytosolic DNA-directed RNA polymerase complex)
  - It will match two query definitions, but only one namespace
  - Different sites need different structured queries and results may come from different namespaces
- New query definitions are inserted alongside current queries
  - Users can choose which query definitions are relevant to them
- The same query definition may be used for both SPARQL and HTTP GET URL resolutions
User attribution

- Users who create queries are attributed using provenance records that are separated from the query, as the provenance information does not need to be distributed along with the query definitions or namespaces for the query to function
- Users are authenticated using an OpenID URL, but this doesn't need to be the only authentication mechanism as long as user contributions are uniquely referenced

Query definitions

- Query definitions are created based on parameters that can be derived from user input
- Example: http://bio2rdf.org/label/go:0000345
  - Protocol and host taken off, as it is not currently utilised within the distributed query model
  - label/go:0000345 matches the regular expression label/([^w-]+):(.+) on both query definitions http://bio2rdf.org/query:labelsearch and http://bio2rdf.org/query:labelsearchforgo
Workflow

Resolved URI: http://bio2rdf.org/label/go:0000345

Host name: http://bio2rdf.org/
Query: label/go:0000345

Regular expression: label/([w-]+):(.+)

http://bio2rdf.org/query:labelsearch
http://bio2rdf.org/query:labelsearchfogo

Query definitions

- http://bio2rdf.org/query:labelsearchfogo
  - Searches for a predicate, http://bio2rdf.org/ns/go:name, that is used by databases hosting the go namespace to indicate a possible label
  - The first regular expression matching group ([w-]+) for this query is defined to contain a namespace "prefix" that may match exactly with a prefix on one or more NamespaceEntry definitions
  - This query is limited to the namespace identified by http://bio2rdf.org/ns:go
Query definitions

- http://bio2rdf.org/query:labelsearch
  - Searches for common generic label properties such as rdfs:label (RDF Schema) and dc:title (Dublin Core)
  - Not limited to the "go" namespace in general, but is localised to endpoints which support the "go" namespace for the example "label/go:0000345"
  - Can be used on endpoints where http://bio2rdf.org/query:labelsearchforgo is also used

Editor demo

- Web based configuration editor provides an easy way to create basic queries, namespace entries, and provider configurations that can then be included into software by exporting the RDF definition of the item
- Configurations created by other users are also accessible and attributed to the relevant users
- http://qut.bio2rdf.org/secured/configurationeditor/