BIOINFORMATICS ORIGINAL PAPER

Vol. 29 no. 13 2013, pages 1663–1670
doi:10.1093/bioinformatics/btt192

Databases and ontologies

Advance Access publication April 25, 2013

Sharing and executing linked data queries in a collaborative environment

María Jesús García Godoy, Esteban López-Camacho, Ismael Navas-Delgado and José F. Aldana-Montes*

Lenguajes y Ciencias de la Computación, Universidad de Málaga, Bulevar Louis Pasteur 35, Málaga, Spain

Associate Editor: Martin Bishop

ABSTRACT

Motivation: Life Sciences have emerged as a key domain in the Linked Data community because of the diversity of data semantics and formats available through a great variety of databases and web technologies. Thus, it has been used as the perfect domain for applications in the web of data. Unfortunately, bioinformaticians are not exploiting the full potential of this already available technology, and experts in Life Sciences have real problems to discover, understand and devise how to take advantage of these interlinked (integrated) data.

Results: In this article, we present Bioqueries, a wiki-based portal that is aimed at community building around biological Linked Data. This tool has been designed to aid bioinformaticians in developing SPARQL queries to access biological databases exposed as Linked Data, and also to help biologists gain a deeper insight into the potential use of this technology. This public space offers several services and a collaborative infrastructure to stimulate the consumption of biological Linked Data and, therefore, contribute to implementing the benefits of the web of data in this domain. Bioqueries currently contains 215 query entries grouped by database and theme, 230 registered users and 44 end points that contain biological Resource Description Framework information.

Availability: The Bioqueries portal is freely accessible at http://bioqueries.uma.es.

Contact: jfam@lcc.uma.es

Supplementary information: Supplementary data are available at *Bioinformatics* online.

Received on January 18, 2013; revised on April 18, 2013; accepted on April 19, 2013

1 INTRODUCTION

The amount of information on the World Wide Web has increased during the past few years. The way that we share and access this information has been especially altered by the lowering of the barriers to access and publish documents and data, and applications have difficulties in reusing this information because of data and format heterogeneity and the lack of formal descriptions of their meaning (semantics).

In an effort to address the problems derived from the heterogeneity and the lack of semantics, Linked Data has emerged. Linked Data is commonly defined as a set of best practices with the objective of connecting, sharing and exposing data and knowledge. The underlying technology, supported by an active community, encourages the application of standards: (i) the use of HTTP URIs, (ii) the SPARQL query language and (iii) Resource Description Framework (RDF) and Web Ontology Language (OWL) for data modelling of data and representation.

This technology has encouraged the publication of interlinked datasets, which in turn has caused a huge increase in the Linked Data cloud in the past few years. Datasets in 2007 contained >2 billion RDF triples and 2 million RDF links. In September 2011, this amount had reached ~31 billion RDF triples, which were interlinked by means of ~504 million RDF links (latest update). The Linked Data community has produced many tools to take advantage of these data, such as RDF editors (e.g. Vapour), browsers (e.g. Tabulator), search engines and semantic web search machines (e.g. Watson with millions of triples indexed). However, despite the rapid growth of data and the availability of developed tools, there are some problems in the development of applications closer to end-users, specifically in the Life Sciences domain where there has been a rapid growth in the amount of biological, biochemical and medical data in recent years. For this reason, we have developed a web tool called Bioqueries that uses Linked Data technology in the Life Sciences domain, and which has been designed for biological and bioinformatic end-users.

In the Life Sciences domain, available Linked Data tools are not usually targeted at biologists because they do not take into account end-user usability. To address this, there have been some attempts to improve the application of this technology to the biological domain. Bio2RDF is a semantic web application designed to solve the integration problem in the bioinformatic area (Belleau *et al.*, 2008). Bio2RDF integrates data from the most popular databases, such as Uniprot, OMIM, Kegg, Reactome and so forth. This mash-up system offers a graphical interface in which users obtain the information in RDF keyword-based searches, and it also provides SPARQL end points.

Linked Life Data (LLD) (Momtchev *et al.*, 2009) is another integration project that stores billions of biomedical RDF statements in the Life Sciences and health care domains. The LLD group implemented RDF representation of PubMed, UMLS, Entrez-Gene and Open Biological and Biomedical Ontologies (OBO) Foundry data sources. Furthermore, it provides three ways to access the data: a web interface to introduce SPARQL queries, a browser to explore the information and a graphical visualization tool. This platform is similar to Bio2rdf project and offers the same features.

*To whom correspondence should be addressed.

The Uniprot consortium has also made a contribution and has published their knowledge base as Linked Data (Redaschi and UniProt Consortium, 2009). It offers several end points to be queried: the Bio2rdf end point and the end point provided by the Uniprot website.

The Linking Open Drug Data (LODD) (Samwald *et al*., 2011) was built around the need to integrate large amounts of biomedical data from different data sources because of the development of new therapies for diseases. This project stores 8.4 million triples and 388 000 links to external sources and is an important framework that interconnects different data from Linked Clinical Trials dataset, DrugBank, TCMGeneDIT and Diseaome.

In the approaches cited (Bio2rdf, LLD, Uniprot and LODD), the main goal is to publish helpful end points to provide direct access to the data. However, they are unconnected islands of knowledge that are at best, linked. Bioqueries addresses the problem that users have in locating and using such places of information. Bioqueries provides a central repository of queries that access distributed end points. Therefore, Bioqueries is not concerned with the publication of data as such but rather SPARQL queries for retrieving such data.

In the biological domain, biologists are conscious of the importance of interactive wikis in solving the problem of large-scale data management, even with the implicit inconsistencies in published and commonly accepted data in the literature (Salzberg, 2007). The idea of using wikis in Life Sciences was born with the objective of the visualization of intracellular biosignalling pathways to promote biocuration and interactivity between biological users. The biological wikis are based on the Wikipedia idea where users can edit online with the intention of helping biologists handle large amounts of information on genes, proteins, transcripts and metabolites. The 1000 Genomes programme for humans (Durbin *et al*., 2010) is a wiki, which contains unquantifiable information on protein structure and function, biomolecular interactions, metabolic and biosignalling pathways that need to be integrated. Wikiomics is another wiki for bioinformaticians, which contains information on 'omic' concepts and tools (Waldrop, 2008). EcoliWiki is a social community interested in the *Escherichia coli* organism (McIntosh *et al*., 2011). Wikipathways is a wiki platform created for pathway model curation (Kelder *et al*., 2012). Both Wikigenes (Hoffmann, 2008) and Gene Wiki (Huss *et al*., 2008, 2010) are wikis for the field of genetics: Wikigenes contains a set of articles edited to increase the accuracy of the large amount of data produced by sequencing techniques stored in GenBank database (Benson *et al*., 2009). Gene Wiki is a specialized section of Wikipedia that focuses on re-organizing, extending and completing the human gene articles (Mons *et al*., 2008).

Despite the existence of SPARQLbin (http://www.sparqlbin.com/) to share SPARQL queries and the collaborative projects in biological areas, there is no biological social community that contributes to the consumption of Linked Data by sharing biological SPARQL queries. Hence, we aim to encourage the sharing of users' experience trying to take advantage of Linked Data in the biological domain. Bioqueries aims to start the process towards a greater understanding of Life Sciences Linked Data sources by means of online social networks. We have used social networks because these have been a key component in the development of great discoveries in Life Sciences. Bioqueries opens up a way to build-up communities around a shared interest in certain biological domains to take advantage of public Linked Data. This is achieved by sharing a virtual space in a wiki-based portal for the design and execution of (federated and non-federated) SPARQL queries that are executed and documented using natural language descriptions. Contrary to all the aforementioned biological wikis that use Wikipedia software, Bioqueries is implemented in Drupal taking advantages of what this technology offers.

The main contributions of this work are (i) a community-based approach to share SPARQL queries as a means of sharing the knowledge in Life Sciences Linked Data repositories; (ii) a set of queries provided as a seed for building the community formed by biological and bioinformatic users; and (iii) a set of software modules to provide a wiki system with SPARQL evaluation, which opens up new possibilities for creating distributed queries to answer more complex queries that can not be answered by a single data source.

This article is structured as follows: Section 2 describes two use cases. Section 3 describes the methods applied to implement the system. Section 4 shows the website navigation, the Bioqueries content and a System Usability Scale (SUS) score usability test. Section 5 describes the advantages and limitations that Bioqueries presents in its current form and finally, Section 6 summarizes the key points presented in this article.

2 USE CASES

The use cases presented in this section are based on the Drugbank end point, which provides information about drugs and drug targets. For the first use case, we have selected the following query:

```
PREFIX drugbank: <http://bio2rdf.org/drugbank_vocabulary:>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dc: <http://purl.org/dc/terms/>
SELECT distinct ?label ?description ?pharmacology
  ?route_of_elimination ?enzyme_target ?absorption ?indication
  ?toxicity ?half_life ?label_target
WHERE {
  ?s rdf:type drugbank:Drug.
  OPTIONAL {?s rdfs:label ?label.}
  OPTIONAL {?s dc:description ?description.}
  OPTIONAL {?s drugbank:pharmacology ?pharmacology.}
  OPTIONAL {
    ?s drugbank:route-of-elimination ?route_of_elimination.}
  OPTIONAL {?s drugbank:enzyme ?enzyme_target.}
  OPTIONAL {?s drugbank:absorption ?absorption.}
  OPTIONAL {?s drugbank:indication ?indication.}
  OPTIONAL {?s drugbank:toxicity ?toxicity.}
  OPTIONAL {?s drugbank:half-life ?half_life.}
  OPTIONAL {?s drugbank:target ?target.}
  OPTIONAL {?target rdfs:label ?label_target.}
  FILTER REGEX(?label, ''Bivalirudin'', ''i'')
}
LIMIT 12
```

This SPARQL query retrieves Drugbank information relevant for an introduced drug on its pharmacology, route of elimination, target enzyme, absorption, indication, toxicity and half-life. The information retrieved could be useful to those researchers who need to compare similarities between drugs, to link drugs and their drug targets and also to obtain useful information to perform *in silico* drug target discoveries.

The second use case aims to demonstrate the use of federated queries (retrieving and integrating information from several end points). For example, the following federated query retrieves information on pharmacology, indications, toxicity and drug targets from DrugBank and resources, which link to GeneBank information on gene length and GeneBank resources for that gene related with the drug introduced. The information retrieved could be useful, for those researchers whose studies are related with pharmacogenomics, for obtaining gene information relative to the drug and understanding how possible variations in those genes contribute to the development of new alternative drugs:

```
PREFIX bio2rdf: <http://bio2rdf.org/bio2rdf#>
PREFIX drugbank: <http://bio2rdf.org/drugbank_vocabulary:>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT distinct ?label ?description ?pharmacology
  ?route_of_elimination ?enzyme_target ?absorption ?indication
  ?toxicity ?label_target ?name ?url ?length
WHERE {
{SERVICE <http://s4.semanticscience.org:16006/sparql>{
  ?s rdf:type drugbank:Drug.
  ?s rdfs:label ?label.
  OPTIONAL {
    ?s drugbank:pharmacology ?pharmacology.}
  OPTIONAL {
    ?s drugbank:xref ?xref.}
  OPTIONAL {
    ?s drugbank:route-of-elimination ?route_of_elimination.}
  OPTIONAL {
    ?s drugbank:enzyme ?enzyme_target.}
  OPTIONAL {
    ?s drugbank:absorption ?absorption.}
  OPTIONAL {
    ?s drugbank:indication ?indication.}
  OPTIONAL {
    ?s drugbank:toxicity ?toxicity.}
  OPTIONAL {
    ?s drugbank:target ?target.}
  OPTIONAL {
    ?target rdfs:label ?label_target.}
  FILTER REGEX(?label, ''Dornase Alfa'', ''i'')}}
  OPTIONAL {
    {SERVICE <http://s4.semanticscience.org:16008/sparql>{
      ?xref rdfs:label ?name.
      ?xref bio2rdf:url ?url.
      ?xref bio2rdf:length ?length.}}
  }}
LIMIT 12
```

3 METHODS

The Bioqueries portal provides the following features: (i) end point and categorized SPARQL query registry (with search capabilities); (ii) social annotations of queries; (iii) query execution; (iv) relationship navigation between RDF objects to ease the SPARQL query construction and understanding; (v) navigation. In this section, we explain the implementation of the portal and describe its main features.

Bioqueries has been implemented using Drupal 7 (http://www.drupal.org/). Different Drupal modules were used to add wiki-based features to the web application, such as the creation of different user profiles (administrator and common user), access permissions, content modification, different content types (public, private, own and non-owned content), query classification characteristics (categorization by hierarchy and labels) and a content search engine (Fig. 1 shows a general scheme of the Bioqueries architecture).

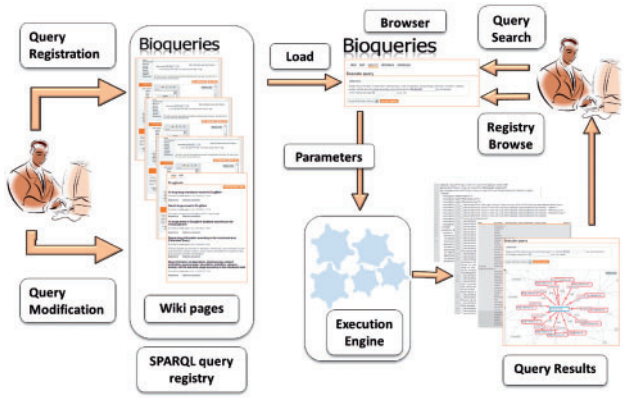

Fig. 1. Bioqueries system architecture. The Bioqueries application has been designed for two profiles with backgrounds in biology and bioinformatics. Federated and non-federated queries can be added, edited and executed. The mechanism for the query execution is different for federated and non-federated queries. Federated queries are split into different queries, and the retrieved results are combined in a single result. Details on query executions will be given in this section

3.1 End point registry

Bioqueries has an end point registry for providing users with a set of tested end points. Users can suggest new end points for registration, indicating the end point name, a brief description and the URL of end points. The administrator of the system is alerted about the new added end points and checks them before their publication in the portal. However, in some cases, the availability of these end points is temporally limited because of server maintenance, server failure, network problems and so forth. This situation is beyond the control of administrators of Bioqueries; therefore, it includes end point status information; hence, when this issue arises, users do not try and run queries in this situation. When a user clicks on the database information or opens a query, Bioqueries sends a SPARQL pre-defined query to the end point. If Bioqueries receives an answer, the end point is annotated with a green point in the user's interface. The query execution is only active if the end point(s) of a query is/are available at that moment. To the contrary, if Bioqueries does not receive an answer, the end point status is given in red, and the user is not able to execute the query.

In the first use case, Drugbank end point is used. Thus, this end point has to be previously added in Bioqueries. The end point URL is given as http://s4.semanticscience.org:16006/sparql. The end point name (Drugbank) is also included.

3.2 RDF structure exploration

We analysed the structure of biological RDF information provided by different end points to initiate a wiki with an initial set of queries to share between users and to allow them to execute pre-designed queries. Therefore, we manually analysed the structure of the RDF information from Life Sciences repositories in the Linked Data cloud. The objective of this analysis was to design an initial set of high quality of SPARQL queries, which are presented in Section 4.

For exploring the RDF structure of any biological end point, a general procedure based on a set of SPARQL queries was followed in most cases to construct the seed of the queries:

- The first query is sent to the end point to obtain the classes stored in the end point:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT distinct ?o
```

```
WHERE {
  ?s rdf:type ?o.
}
```

In the first use case, which uses Drugbank as the end point, there are 14 well-defined classes that correspond to dosage, drug, drug–drug interaction, drug–target interaction, drug–enzyme interaction, drug–transporter interaction, form, mixture, patent, pharmaceutical, route, source, target and unit.

- Once the classes are known, the second step consists of discovering the instances (resources), which are included in a class.

  If the drug class has been selected for the class to be explored, all instances classified inside this class are retrieved by this query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT ?s
WHERE {
  ?s rdf:type <http://bio2rdf.org/drugbank_vocabulary:Drug>.
}
```

- The third step then obtains the predicates and objects related to the subject (an instance of the class specified) using the following query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT ?p ?o
WHERE {
  <http://bio2rdf.org/drugbank:DB00006> ?p ?o.
}
```

We select (subject) an instance to determine its predicates. In this example, the instance 'http://bio2rdf.org/drugbank:DB00006' references 30 different predicates. In the use case, the following predicates were selected to build the query:

- 'http://www.w3.org/2000/01/rdf-schema#label',
- 'http://purl.org/dc/terms/description',
- 'http://bio2rdf.org/drugbank_vocabulary:pharmacology',
- 'http://bio2rdf.org/drugbank_vocabulary:route-of-elimination',
- 'http://bio2rdf.org/drugbank_vocabulary:enzyme',
- 'http://bio2rdf.org/drugbank_vocabulary:absorption',
- 'http://bio2rdf.org/drugbank_vocabulary:indication',
- 'http://bio2rdf.org/drugbank_vocabulary:toxicity',
- 'http://bio2rdf.org/drugbank_vocabulary:half-life',
- 'http://bio2rdf.org/drugbank_vocabulary:target'.

- If an object is a resource of the end point, the third step can be repeated and expressed in the same query. To the contrary, if the object is an external resource, a federated query can be created and then, we can obtain the predicates and objects related with that resource (subject). The instances related to the previous predicates can be analysed through an object property. For example, the predicate for the enzyme target is a predicate, the object of which is the Drugbank resource: 'http://bio2rdf.org/drugbank_target:1757'. The objects related to the rest of the predicates are literal expressions. For example, the object of the 'label' predicate is 'Bivalirudin [drugbank:DB00006]'. As we have mentioned, it is possible to explore the objects (internal resources) to extend the results retrieved from DrugBank end point. For example, 'http://bio2rdf.org/drugbank_target:1757' can be analysed through exploring the predicates and objects. Furthermore, a federated query can be created with the Genebank resource, which references new predicates and objects, which contain additional genetic information on drug targets. The DrugBank federated query mentioned in the second use case is an example of this and is included in Bioqueries.

At this point, we wish to point out that although the exploration of the data structure of each end point may be a hard task in some cases (as in the previous example), it allows the construction of a seed of SPARQL queries. These queries can be considered as the starting point for the creation of new queries as mentioned in the Bioqueries content. Additionally, some end points, such as LLD, Wikipathways, GWAS central and Bioportal publish query examples that can be parameterized to be added to Bioqueries.

3.3 SPARQL query registry

The registry of SPARQL query is a registry of structured wiki pages (only registered users can add new queries). To explain the query registry more clearly, we use the first use case (http://bioqueries.uma.es/query/drug-information-according-introduced-term). These wiki pages (registered SPARQL queries) are divided into the following sections (Supplementary Fig. S1):

- Query title, which is a natural language description of the query used for its identification. Thus, each query can be referenced as 'http://bioqueries.uma.es/query/title', enabling queries to be shared in external tools (email, social networks, publications and so forth). In the use case under discussion, the title is 'drug information according to the introduced term'; therefore, the query URL is created as 'http://bioqueries.uma.es/query/drug-information-according-introduced-term'.
- Documentation is a more extended description used to show the meaning of the query in natural language. The documentation was completed in the use case as follows: 'this query displays drug information on description, pharmacology, route of elimination, enzyme target, absorption, indication, category, toxicity, half-life and other targets from Drugbank'.
- SPARQL query, which itself is defined using an extension of SPARQL 1.1 in which any part of the query or variable can be parameterized. Parameters are preceded by '??'. For example, the given registered Drugbank query presents the drug variable and the limit of parameterized results:

```
PREFIX drugbank: <http://bio2rdf.org/drugbank_vocabulary:>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dc: <http://purl.org/dc/terms/>
SELECT distinct ?label ?description ?pharmacology
  ?route_of_elimination ?enzyme_target ?absorption ?indication
  ?toxicity ?half_life ?label_target
WHERE {
  ?s rdf:type drugbank:Drug.
  OPTIONAL {?s rdfs:label ?label.}
  OPTIONAL {?s dc:description ?description.}
  OPTIONAL {?s drugbank:pharmacology ?pharmacology.}
  OPTIONAL {
    ?s drugbank:route-of-elimination ?route_of_elimination.}
  OPTIONAL {?s drugbank:enzyme ?enzyme_target.}
  OPTIONAL {?s drugbank:absorption ?absorption.}
  OPTIONAL {?s drugbank:indication ?indication.}
  OPTIONAL {?s drugbank:toxicity ?toxicity.}
  OPTIONAL {?s drugbank:half-life ?half_life.}
  OPTIONAL {?s drugbank:target ?target.}
  OPTIONAL {?target rdfs:label ?label_target.}
  FILTER REGEX(?label, ''??term'', ''i'')
}
LIMIT ??number
```

- Bioqueries provides an option that checks the URIs in the query code and advises users to replace them with the suggested namespaces. This option can be freely taken by users when a query is created or even edited. In the analysed use case, the use of the

prefix 'drugbank:' instead of 'http://bio2rdf.org/drugbank_vocabulary:' is suggested to shorten the URIs.

- The statement section includes a description, the parameters of which are between '$'. This statement is shown to the end-user when preparing for execution, converting the parameters in text boxes to fill in these parameters. In the example query, 'term' and 'number' parameters are used to specify a drug name and a limit of results in the query. The statement associated with the first use case is 'display drug information on description, pharmacology, route of elimination, enzyme target, absorption, indication, category, toxicity, half-life and other target according to the introduced term $term$ (e.g. Bivalirudin). Limit of results per page $number$ (e.g. 100)'.
- A tag for selecting the thematic classification (e.g. biological process, gene expression, metabolism and so forth). The 'chemical compounds' tag is used in the commented query.
- A list of the end points for querying (for single end points, only an end point can be selected). Bioqueries also provides support for federated queries. In the first use case, Drugbank end point is selected.

Users have permission to further modify their own query by introducing new queries, and they are also able to share the query entries and their results in other social networks. Queries are by default private to the owners, but they can be shared by promoting them to public. Bioqueries curators are notified of these publications and then check the query to assign it a validated annotation.

To make the SPARQL query construction easier, Relfinder software (Heim *et al.*, 2009) is used to visualize relationships between two or more RDF nodes from a selected repository and to find new unknown relationships. This feature enables users to dynamically explore the graph and show how the information is related. After seeing what relationships exist between terms in the end point, the user is, therefore, able to write a SPARQL query more easily. For example, in the previous Drugbank query, the relationships (drug–drug interaction) between concepts, such as Bivalirudin (drugbank:DB00006) and *Ginkgo biloba* (drugbank:DB01381) can be analysed by introducing both concepts in the Relfinder tool as Figure 2 shows.

Furthermore, we have added an option for each query, which enables users to clone it. In this way, this option makes the registering of new queries easier by editing the existing queries (only by modifying the query structure) to design new queries.

3.4 Execution of SPARQL queries

The Bioqueries interface includes the execution of queries directly from the wiki pages by introducing values for the query parameters. This module uses Java servlets that use the Apache Jena framework, which enables the execution of SPARQL queries in each corresponding end point. This set of servlets acts as a bridge, between the Bioqueries website and the different biological end points. The query response obtained by the service is shown in the wiki interface as an HTML table, downloaded as RDF, N-triples files or in a graph. For example, in the simple query given, users introduce a drug term, such as *Bivalirudin*, *Lepidurin*, *Dornase Alfa*, *Rivaroxaban*, *Denileukin diftitox* and a limit number for results per page to execute a Drugbank query. The results of the query for *Bivalirudin* are displayed in the previously mentioned formats (Supplementary Figs S2–S4 show the results of the query displayed in each format).

3.5 Federated queries

ARQ is a query engine for Jena that supports the SPARQL RDF Query language and provides a basic query access to multiple distributed end points (McBride, 2002). The new keyword SERVICE has been added to provide the ability to make one or several SPARQL protocol calls within a query and not just send the whole query to one individual end point. The query processing is divided into four stages: (i) the parsing of the input query, (ii) the query planning that splits query into sub-queries, using each set of sub-patterns to be sent to a named SPARQL service end point, (iii) the query execution that sends the sub-queries to data sources and (iv) the joint of the different query responses. Therefore, according to these query federated processing features, we selected and integrated the ARQ engine into the Bioqueries Drupal application to provide a possibility of executing of federated SPARQL queries.

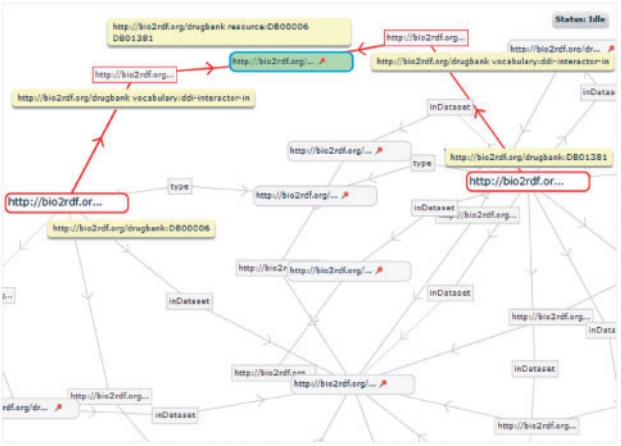

Fig. 2. This screen shot shows the relationship between two drugs, such as Bivalirudin and *Ginkgo biloba,* on Relfinder interface. The network highlighted shows the relationship between these two drugs and the types of concepts (drug–drug interaction). The highlighted node is the type of drug–drug interactions related to these two drugs

When executing the query, it is split into various sub-queries that are sent to each participating end point. Only conditions that participate in an end point are sent to each end point. In the second use case, the description, pharmacology, route of elimination, enzyme target, absorption, indication, category, toxicity, half-life and other targets are retrieved for an introduced drug from Drugbank. This process is one of the sub-queries. The other sub-query uses the genetic information retrieved in the first sub-query result to get the URL, name and gene length from the Genebank end point. The different results are combined when all the sub-queries are executed, and then they are shown to the user. The entire process is done automatically using the ARQ query engine.

We have explained how to register a non-federated query of the first use case. For the second use case (http://bioqueries.uma.es/fedquery/display-drug-information-according-introduced-term-federated-query), the federated query registration requires the addition of a minimum of two end points. These are expressed using the form $end point-number$ where the end point number refers to end points to be interrogated. The federated query retrieves information from Drugbank and Genebank databases; therefore, the $end point-1$ and $end point-2$ correspond to Drugbank and GenBank, respectively:

```
PREFIX bio2rdf: <http://bio2rdf.org/bio2rdf#>
PREFIX drugbank: <http://bio2rdf.org/drugbank_vocabulary:>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT distinct ?label ?description ?pharmacology
  ?route_of_elimination ?enzyme_target ?absorption ?indication
  ?toxicity ?label_target ?name ?url ?length
WHERE {
  {SERVICE <$endpoint-1$> {
    ?s rdf:type drugbank:Drug.
```

```
    ?s rdfs:label ?label.
    OPTIONAL {
      ?s drugbank:pharmacology ?pharmacology.}
    OPTIONAL {
      ?s drugbank:xref ?xref.}
    OPTIONAL {
      ?s drugbank:route-of-elimination ?route_of_elimination.}
    OPTIONAL {
      ?s drugbank:enzyme ?enzyme_target.}
    OPTIONAL {
      ?s drugbank:absorption ?absorption.}
    OPTIONAL {
      ?s drugbank:indication ?indication.}
    OPTIONAL {
      ?s drugbank:toxicity ?toxicity.}
    OPTIONAL {
      ?s drugbank:target ?target.}
    OPTIONAL {
      ?target rdfs:label ?label_target.}
    FILTER REGEX(?label, ''??term'', ''i'')}}
    OPTIONAL {
      {SERVICE <$endpoint-2$> {
        ?xref rdfs:label ?name.
        ?xref bio2rdf:url ?url.
        ?xref bio2rdf:length ?length.}}
}}
LIMIT ??number
```

4 RESULTS

This section is divided into four sub-sections: the first sub-section describes the website navigation, the second sub-section describes Bioqueries content, the third sub-section shows a system usage analysis and the last section concludes with a system usability assessment.

4.1 Website navigation

Unregistered users have permissions to run registered queries and also to test the software functionalities before deciding whether to become a Bioqueries community member. Thereafter, once users have registered in the system, they have full permissions to use registered queries. Users can access and locate queries using the query explorer to browse through them. The query explorer offers the possibility of organizing queries in a list according to their databases and themes, updates and post dates, the query title, the order (ascendant or descendent), the number of queries per page and a combination of all these elements. To help the query searching on the website, a classification is applied to filter queries by categories (annotations, biological process, chemical compounds, diseases, enzymology, gene expression, genetic, metabolism, proteomic, scientific texts and taxonomy) and databases. Furthermore, Bioqueries provides a search engine that allows further complex searches to be carried out using the advanced search option by applying specifications, such as keywords and type of information to find. For example, users can look up contents related to a term (of the uses cases presented) (e.g. *pharmacology* or *enzyme target*) and type of information (e.g. *query*) obtaining a list of results with their specifications.

4.2 Bioqueries content

The RDF information analysis previously described in Section 3 enabled us to start the wiki with a set of 112 SPARQL queries with the objective of creating a critical number of queries to release in the social community. This seed of initial queries could then be used to build more queries using known relationships between biological concepts. The wiki started with an optimized set of SPARQL queries to biological end points offered by Bio2RDF, Wikipathways and Pathway Commons (Cerami *et al.*, 2011) (the last one is an end point published in collaboration with Pathway Commons maintainers). The set of developed queries was classified manually into the following categories (the current number of queries for each one in brackets): proteomic (23), scientific texts (7), diseases (12), chemical compounds (13), biological process (11), metabolism (38), genetic (29), enzymology (18), gene expression (18), annotations (15) and taxonomy (3). The initial number of seed queries included federated and non-federated queries.

4.3 Bioqueries usage analysis

As we have suggested, the success of Bioqueries depends on the feedback on the use of website pages and biological and bioinformatic user contributions. In an ideal scenario, a query retrieves information that is useful for a biological user, and then he/she contributes with comments to improve and divulgate the query and the retrieved information at the same time as a bioinformatics user registers more queries using SPARQL structure of stored queries. This feedback enables the construction of an active community whose contributions draw more contributors, increasing the usage and the utility of Bioqueries. Therefore, this sub-section presents a usage analysis of Bioqueries over a long period, and also a study of the number of times that the databases are invoked.

Usage was analysed for 1 year between the September 1, 2011 and September 1, 2012 and the preceding 6 months up to February 28, 2013 (Supplementary Fig. S5). In total, the number of registered users is 230 and 215 query entries (~5.6% of queries were included by users other than us). The queries were viewed >13 368 times (an average of 798 times per month were author views, which are not included). Supplementary Table S1 shows the top-viewed queries, the database and the number of views registered. As Bioqueries suddenly experienced a rapid growth in registered queries and users, we started to track the executions per query from the first 2 weeks in December 2012. Table 1 shows the top 10 most interrogated end points and the executed queries in this period. Fraunhofer SCAI UIMA-HPC is the most executed, this is a small triple store that contains information about chemistry in European Patent Office (EPO) patents and was included on the website by the data providers. Uniprot, Kegg and Drugbank end points are (in that order) the most used end points after the Fraunhofer SCAI. Furthermore, UMLS (Bodenreider, 2004) and pharmgkb (Hewett *et al.*, 2002) are end points that have been recently added to the website and have experimented an increasing number of executed queries.

4.4 Bioqueries usability assessment

A variety of questionnaires are used and reported in the literature for assessing the usability of website applications, including QUIS, SUS, CSUQ and Microsoft's Product Reaction Cards. Tullis and Stetson (2004) conducted a study to determinate the

Table 1. The top 10 most executed end points

Pos.	End point name	Existing queries	Number of executions
1	Fraunhofer SCAI UIMA-HPC	8	236
2	Uniprot and Uniprot beta	10	39
3	Drugbank	5	31
4	Bioportal	16	29
5	Kegg	19	28
6	Wikipathways	20	22
7	Pathwcommons	16	19
8	UMLS	4	16
9	OMIM	11	14
10	pharmgkb	7	12

effectiveness of these standardized usability questionnaires. The results of this study demonstrated that SUS (Brooke, 1996) addresses different aspects of users' reactions to a web application as a whole rather than asking users to assess specific features and yields the most reliable results with a minimum of participants. These conclusions encouraged us to select the SUS score to provide a global view of subjective assessments of Bioqueries usability.

The SUS score gives us an idea of the usability grade of systems, which should cover aspects such as effectiveness (the ability of users to complete the tasks), efficiency (the level of resources consumed to carry out the tasks) and the users' satisfaction (a user's reaction derived from the system use). Also, the usability of any system or tool must be viewed in the context that is going to be used. Therefore, we selected a group of 12 participants, 5 of them with biological profiles and 7 of them with bioinformatic profiles. These participants are representative of users who will make use of Bioqueries functionalities in a collaborative context (see technical report in Supplementary Data for more details).

The SUS score consists of 10 items with odd-numbered items worded positively and even-numbered items worded negatively. These items were distributed to the 12 participants with two different profiles as five-point scales numbered from 1 to 5. Each item's score contribution ranges from 0 to 4. For positively worded items, the score contribution is the scale position minus 1. For negatively worded items, it is 5 minus on the scale position. The SUS score is calculated by multiplying the sum of item scores by 2.5, giving a range from 1 to 100. The results of the SUS score for each item are shown in Figure 3. The mean SUS score of two user profile groups was 78.3 (>100). The mean scores obtained from biologist and bioinformatician groups were 78.3 and 79.6, respectively. The highest SUS score obtained in each group was 92.5, and the lowest scores were 67.5 and 72.5 for the biological and the bioinformatic groups (Fig. 3).

5 DISCUSSION

In a social space like Bioqueries, which has a large number of participants in a collaborative space, it is important to share different points of views to enrich overall knowledge.

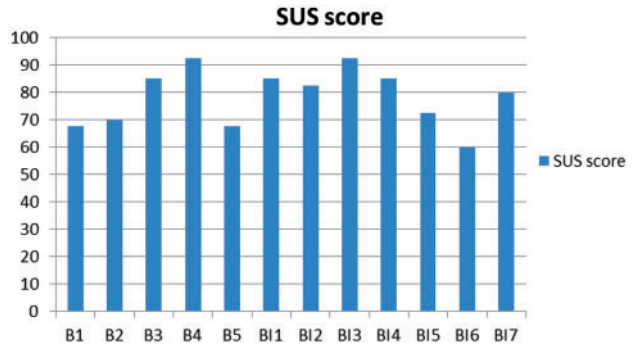

Fig. 3. The graphic shows the SUS score mean obtained by each user

Bioqueries currently has 230 registered users and 215 query entries. Also, we have shown the evolution for 1 year and a 6 month period before February 28, 2013 of the number of registered users against the number of query entries. Supplementary Figure S5 shows a registered user increase in the first months of Bioqueries because of the publicity given. The second increase corresponds to the recent addition of a new federation engine (which can be considered as the first federation system integrated in an end-user Linked Data application in Life Sciences).

Information on the top 20 most visited queries is provided by Supplementary Table S1. This table indicates that the Kegg, MGI and Reactome, top three queries, are the most visited. This information, derived from user's interests, could direct Bioqueries to solve problems in specific biological areas, such as metabolism and genetics. However, despite that these are the most viewed, they are not the most evaluated. Table 1 indicates Fraunhofer SCAI UIMA-HPC and Uniprot end points are the most evaluated, followed by Kegg end point. These statistics lead us to think that there could be a clear tendency to use registered queries by end point owners to design new queries to test non-published end points.

Furthermore, the SUS score obtained from biologist and bioinformatician groups were 78.3 and 79.6, respectively (Supplementary Data). These SUS scores are >60, which means that Bioqueries is easy to use for these two groups. According to these results, we can not ignore the fact that the introduction of SPARQL queries *ab initio* is a hard task for biologists even with the features provided by the system. However, Bioqueries does not limit any biologist who has an interest in and a minimal knowledge of SPARQL query language for introducing new queries.

6 CONCLUSION

In this work, our main goal has been the construction and dissemination of a tool to create an active community of biologists built around the use of Biological Linked Data. The objective of Bioqueries is to bring researchers in the Life Sciences domain closer to Linked Data. Therefore, we have developed a social tool to design, document and execute, in different formats, non-federated and federated queries.

In this biological community, participants use, learn and share opinions and biological information by using Linked Data technology. This system is freely available for users to introduce new

SPARQL queries and to execute queries and use the services provided by the Bioqueries portal.

The Bioqueries community is growing. The registration of new queries has improved the usability of the website; hence, we are considering integrating more features. As for future work, we propose a query recommender, which enables users to select a query from a list of queries according to the user profile and activity.

Funding: This work was supported by the Spanish Ministry of Education and Science [TIN2011-25840]; and the Innovation Science and Enterprise Ministry of the regional govenment of the Junta de Andalucía [P11-TIC-7529].

Conflict of Interest: none declared.

REFERENCES

Belleau,F. *et al.* (2008) Bio2RDF: towards a mashup to build bioinformatics knowledge systems. *J. Biomed. Inform.*, **41**, 706–716.

Benson,D.A. *et al.* (2009) GenBank. *Nucleic Acids Res.*, **37**, 26–31.

Bodenreider,O. (2004) The unified medical language system (UMLS): integrating biomedical terminology. *Nucleic Acids Res.*, **32**, D267–D270.

Brooke,J. (1996) SUS: a quick and dirty usability scale. In: Jordan,P.W. *et al.* (ed.) *Usability Evaluation in Industry*. Taylor and Francis, London, UK.

Cerami,E.G. *et al.* (2011) Pathway commons, a web resource for biological pathway data. *Nucleic Acids Res.*, **39**, 685–690.

Durbin,R.M. *et al.* (2010) A map of human genome variation from population-scale sequencing. *Nature*, **467**, 1061–1073.

Heim,P. *et al.* (2009) RelFinder: revealing relationships in RDF knowledge bases. In: *Proceedings of the 4th International Conference on Semantic and Media Technologies (SAMT)*. Graz, Austria. Lenture Notes in Computer Science, Springer, Vol. 5887, pp. 182–187.

Hewett,M. *et al.* (2002) Pharmgkb: the pharmacogenetics knowledge base. *Nucleic Acids Res.*, **30**, 163–165.

Hoffmann,R. (2008) A wiki for the life sciences where authorship matters. *Nat. Genet.*, **40**, 1047–1051.

Huss,J.W., III *et al.* (2008) A gene wiki for community annotation of gene function. *PLoS Biol.*, **6**, e175.

Huss,J.W., III *et al.* (2010) The gene wiki: community intelligence applied to human gene annotation. *Nucleic Acids Res.*, **38**, D633–D639.

Kelder,T. *et al.* (2012) Wikipathways: building research communities on biological pathways. *Nucleic Acids Res.*, **40**, 1301–1307.

McBride,B. (2002) Jena: a semantic Web toolkit. *IEEE Internet Comput.*, **6**, 55–59.

McIntosh,B. *et al.* (2011) Ecoliwiki: a wiki-based community resource for *Escherichia coli*. *Nucleic Acids Res.*, **40**, 1270–1277.

Momtchev,V. *et al.* (2009) Expanding the pathway and interaction knowledge in linked life data. In: *International Semantic Web Challenge*.

Mons,B. *et al.* (2008) Calling on a million minds for community annotation in WikiProteins. *Genome Biol.*, **9**, R89.

Redaschi,N. and UniProt Consortium. (2009) UniProt in RDF: tackling data integration and distributed annotation with the semantic web. *Nat. Precedings*, [Epub ahead of print, doi:/10.1038/npre.2009.3193.1, April 28, 2009].

Salzberg,S. (2007) Genome re-annotation: a wiki solution? *Genome Biol.*, **8**, 102.

Samwald,M. *et al.* (2011) Linked open drug data for pharmaceutical research and development. *J. Cheminform.*, **3**, 19.

Tullis,T.S. and Stetson,J.N. (2004) A comparison of questionnaires for assessing website usability. In: *Proceedings of the Usability Professionals Association (UPA) 2004 Conference*, pp. 7–11.

Waldrop,M. (2008) Big Data: wikiomics. *Nature*, **455**, 22–25.