Extending Semantic provenance into the Web of Data

Jun Zhao[1], Satya S. Sahoo[2], Paolo Missier[3], Amit Sheth[2], Carole Goble[3]

[1] Department of Zoology, University of Oxford, UK, [2] Kno.e.sis Center, Wright State University, USA,
[3] School of Computer Science, University of Manchester, UK,
[1]jun.zhao@zoo.ox.ac.uk, [2]{sahoo.2,amit.sheth}@wright.edu, [3]{pmissier,carole}@cs.man.ac.uk,

Abstract

The importance of tracking and querying the provenance of experimental data for scientific applications is only now beginning to emerge, as a number of provenance management systems reach maturity. In addition to provenance, domain-specific semantic annotations to data products and the Web of Data are playing an increasingly important role as contextual metadata that can be used to assist with the interpretation of experimental data. In this article we use an example workflow, and a simple classification of user questions on the workflow's data products, to explore the combination of these three strands of contextual metadata through a semantic data model and infrastructure, and their potential to support enhanced semantic provenance applications.

Index Terms

Primary classification: H.3.4 Systems and Software (Semantic Web); J.3 Life and Medical Sciences (Biology and genetics)

Additional classification: D.2.1 Requirements/Specifications; D.2.12 Interoperability (Data mapping)

General Terms: Design, Software

Introduction

The increasing use of computing resources is transforming the way scientific research is carried out, and in the process it is creating a vast amount of scientific data, especially in the life sciences domain. The challenge now facing both life sciences and computer science researchers is not in data generation, but rather, in making sure that any member of a scientific community has the means to correctly interpret automatically generated information, possibly a long time after it has been produced. This involves complementing observational or experimental data with various types of annotations, as well as with other contextual metadata. In this article we focus specifically on provenance metadata, which describes the way data has been produced, and on semantic annotations, whereby domain-specific terms from some agreed-upon collection of vocabularies are used to clarify the meaning of the data. Furthermore, we restrict our attention to workflow provenance, that is, the provenance of data products that are obtained through a (generally automated) computational process consisting of an orchestration of individual tasks.

The recently emerging Linked Open Data (LOD)[1] cloud, i.e. the Web of Data, provides a third kind of contextual metadata. The LOD initiative promotes the publication of data in machine-accessible format and the linking amongst heterogeneous data items. It leads to large-scale publication of interlinked data items, including scientific datasets such as UniProt, KEGG, Reactome, Drug Bank, and NCBI Entrez Gene. These datasets form a vast graph that can be seamlessly explored and navigated thanks to its uniform representation using the RDF data model.

Past research provides anecdotal evidence of how each of these three context elements, taken independently, can be used effectively. Workflow provenance is useful to answer user queries regarding data products computed by different workflow systems; semantic, domain-specific annotations find their applications mainly in the area of information interpretation and integration; and the Web of Data exposes a vast amount of scientific data in structured format that can be searched using the standard RDF query language SPARQL[2].

[1] http://www.linkeddata.org
[2] http://www.w3.org/TR/rdf-sparql-query/

These three strands have so far largely been pursued independently, however. In this article we discuss their combination into a single, integrated metadata architecture and its potential for answering complex user questions. The primary goal of such model and architecture is to eliminate the need for users to be aware of any provenance metadata infrastructure altogether, by leveraging semantic annotations and LOD mappings to translate high-level user questions into, ultimately, queries on provenance logs.

Specifically, the article offers two main contributions. In the first part, we classify provenance-related questions into three groups, according to the type of contextual metadata required to answer them, namely: (i) simple provenance traces with no domain-specific semantics, (ii) *semantic provenance* [1] (i.e. traces enhanced with domain-specific semantic annotations), and (iii) *Linked Data-aware provenance traces* where the data identifiers mapped to identifiers for data that is published elsewhere in the Web of Data. We use a bioinformatics workflow as a concrete scenario for data production, and show multiple examples of each of the three classes of questions that are related to such scenario.

In the second part of the article we first give a brief overview of our Janus semantic and Linked Data-aware provenance infrastructure, that operates on provenance metadata produced by the Taverna workflow system [2] [3]. A prototype of this infrastructure is discussed in detail in recent prior work [4]. Then, we present the use of a set of query operators for querying provenance information represented using the RDF model. Our approach illustrates the following aspects:

(a) The use of semantic provenance to answer domain-specific user questions;
(b) The use of provenance query operators to implement those questions;
(c) The use of semantics to expose provenance collected during a workflow execution as part the LOD cloud, and thereby answer LOD-aware provenance queries not supported earlier in scientific workflows.

Example workflow for bioinformatics

Our scenario is based on a Taverna bioinformatics workflow, due to Dr. Paul Fisher at the University of Manchester and available from the myExperiment repository[3]. The workflow finds metabolic pathways associated with a user-provided Quantitative Trait Loci (QTL), a region in a chromosome that is known to contain genes responsible for some phenotype under study (reaction to some infection, for example). Scientists may use information about these pathways to narrow down on candidate genes that may be responsible for the phenotype. The workflow, shown as a sketch in Figure 1, starts by retrieving all genes known to the Ensembl database in a given QTL, using the Biomart service. It then maps Ensembl gene identifiers to those used by UniProt and NCBI, so that it can use these gene names to search for associated biological pathways from the KEGG database.

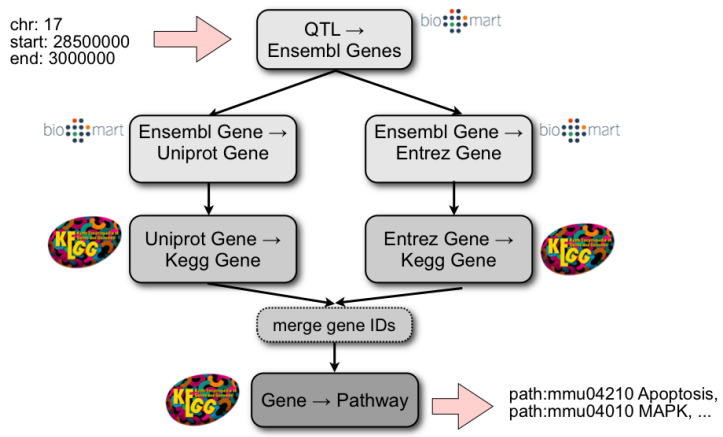

Figure 1. Sketch of the example workflow for mapping QTL genes to metabolic pathways

[3] http://www.myexperiment.org/workflows/931

On a typical QTL region with ~150k base pairs, one execution of this workflow finds about 50 Ensembl genes. These could correspond to about 30 genes in the UniProt database, and 60 in the NCBI Entrez Gene database. Each gene may be involved in a number of pathways, for example the mouse genes *Mapk13* (*mmu:26415*) and *Cdkn1a* (*mmu:12575*) participate to 13 and 9 pathways, respectively.

Provenance data and query models

In this example, the provenance trace for a run holds the direct data dependencies between inputs and outputs of each workflow task, and by extension, it allows indirect dependencies to be computed by transitive closure. The types of data and task dependencies that are captured vary between workflow systems. The Janus provenance model for Taverna is a directed acyclic graph that includes relations of the form *d1→derives_from→d2*, where *d1*, *d2* are data items, *p1→preceded_by→p2* where *p1*, *p2* are workflow tasks, and more (the complete Janus ontology has been described in [4] and is publicly available[4]). Thus, in our example, the graph contains, among many others, the triples

path:mmu04010→derives_from→mmu:26415

and

path:mmu04012→derives_from→mmu: 12575

denoting the dependencies of a pathway from a gene, each identified by the native ID used by the processors. The relation specifically denotes that, for example, *path:mmu04010* is the result of a computation (or data retrieval operation) that involves *mmu:26415*. We refer to this graph as *domain-agnostic*, as it contains no indication as to which data domain (UniProt genes or KEGG pathways, for example) the values belong. This is in contrast to the domain-aware, semantic graphs, introduced later.

Although interoperability models for provenance have been proposed, for example the OPM (Open Provenance Model) [5], most workflow management systems store their provenance using proprietary data models and few of them support queries over those graphs. Several query languages for provenance have been proposed in the recent past. These simple, dedicated languages have the advantage, over a generic graph-matching query model, that query processing can be optimized for the specific types of patterns that correspond to common user questions. A typical query involves computing a partial or complete derivation of an output data product from the workflow inputs, through the intermediate values produced by the processors along the workflow graph. Such a query can be expressed as the transitive closure over provenance relations, for example *derives_from*, which involves traversing the provenance graph. Examples of these query models are proposed by Taverna [3] as well as QLP (Query Language for Provenance) [6]. The latter is currently used mainly in the Kepler system, where users can inspect a provenance graph and interactively formulate a query using a visual provenance browser [7].

User questions on workflow data products

The dependency graph provides the basis upon which a variety of user questions regarding relationships amongst datasets in the workflow can be addressed. The first type of questions refers directly to nodes in the provenance graph. A prototypical example of such query is the following:

Q0: *Find all intermediate and initial input values that contribute to the computation of a certain output value.*

Regardless of the specific query language, expressing Q0 requires users to indicate values of interest using their natural identifier, with no semantic clue as to their meaning, for example *mmu:26415*, an ID that usually depends on the specific database and may not be familiar to the user. Therefore, users are expected to have a complete understanding of how to interpret each of the nodes in the provenance graph, a requirement that third party users, other than the workflow designers themselves, are unlikely to meet. A further problem is that

[4] http://purl.org/net/taverna/janus#

semantically-rich concepts like the *involvement* of a gene in a metabolic pathway (i.e., which genes are activated when a metabolic process in the cell follows a specific pathway) are not naturally expressed in terms of paths in a graph; rather, we would like the provenance query system to be capable of mapping a high-level question regarding which genes are involved in which pathways to a lower-level, graph traversal-type query, without requiring users to be aware of this level of complexity.

These considerations lead to a second type of questions, which involve the use of domain-specific concepts, but can still be rewritten into queries on the provenance graph. Two examples are the following:

Q1. *Find all those genes within the input QTL that are involved in a given KEGG pathway*.

The rationale for this question is that a single pathway may involve only some, but not all, of the genes within the QTL region used as input to the workflow (pathway *path:mmu04010*, for example, involves only two of the 58 genes in the input region).

Q2. *Find all the pathways in the result, which depend on UniProt genes.*

Here the user is interested in those KEGG pathways in which UniProt-specific genes, as opposed to NCBI Entrez genes, are involved.

Answering these questions requires associating data values in the provenance graph with the semantic concepts mentioned in the question. When these are available, the question can indeed be rewritten as a query consisting of (i) filtering values by their semantic type, and (ii) traversing the graph from the output to the input values identified in (i). Note that we can similarly annotate the tasks that appear in the workflow, to support questions that predicate on those annotations, for example:

Q3: *Find all genes whose derivation includes a Biomart database search.*

We can further extend these questions to include conditions on the values, which are not directly available as annotations on the provenance graph, but that may instead be available when the graph is "joined" with other, external data graphs, in a sense defined precisely below. This is our third class of questions. Some examples are as follows:

Q4: *Find all the genes that are known to perform a certain biological function, which are also associated with immune-related pathways.*

Q5: *Retrieve protein information for genes identified in the QTL. Use protein information to corroborate phenotype information inferred from pathways in the result set.*

Q6: *Group the pathways in result set according to diseases or related phenotypic traits*.

Q7: *List relevant PubMed publications for the pathways listed in the result set.*

In all of these examples, not all the conditions can be evaluated using the provenance graph alone, as the graph carries no information about proteins or biological function of a gene (**Q4**, **Q5**), no disease or phenotype information (**Q6**), and no connection between pathways and PubMed literature (**Q7**).

Our approach to answering questions **Q4**-**Q7** makes use of the resources already published in the Web of Data, i.e., by following the Linked Data conventions for data publication on the Web. Using this approach we could implement **Q4** as a hybrid query that (i) retrieves the biological functions of all the genes that appear in the graph using a Linked Data query, and (ii) for those genes that satisfy the condition, finds all paths to the corresponding pathways using a domain-level provenance query. Similarly, question **Q5** involves using Linked Data queries to retrieve proteins associated to genes. In general, we see that these queries essentially require that one (or more) provenance graphs be "joined" with relevant fragments of the Web of Data, when they are available. Such joins come in the form of mappings between the data identifiers found in the provenance graphs,

and those published by various Linked Data sets. In the following, we use the Bio2RDF[5] Linked Data source as an example of how such mappings can be established. The Janus infrastructure for semantic provenance, described in the next section, helps answer questions such as **Q4-Q7** above by making use of those mappings whenever they are available.

Note, incidentally, that the questions in each of these three classes can be extended to operate on multiple provenance graphs, each representing one workflow run. For example:

Q1b: *find the pathways that depend on genes found in a given set of QTL regions*.

Answering such questions requires the provenance query system to be capable of traversing the multiple graphs as one single graph and executing the provenance filtering and traverse functions on this composite RDF graph. The graph composition model involves merging nodes (i.e., RDF resources) that are determined to represent the same entity. As a universal data identification scheme is not available in practice, we need to provide a mapping amongst Janus identifiers from different graphs. Linked Data-aware provenance graphs manage to create data links between Janus RDF provenance and the Web of Data, and the external LOD data identifiers assist us to establish links between the Janus RDF graphs.

Overview of the Janus Semantic Provenance Framework

The Janus Semantic Provenance Framework is based upon an RDF realization of the provenance model briefly described earlier. The framework supports domain-aware provenance queries (Q1-Q7) by means of the following three key components: a semantic conceptual model (an ontology), an annotation model, and a query model.

• The *Janus S+* ontology, based on the domain-agnostic Janus provenance model cited earlier, is an extension to the Provenir Upper Ontology [8].The ontology accommodates a combination of domain-agnostic relations, e.g. *derives_from* as seen earlier, and domain-level concepts drawn from existing domain ontologies, for example those hosted at National Center for biomedical Ontologies (NCBO). The ontology provides the schema to which a provenance trace, i.e., an RDF graph, conforms.

• The *semantic provenance collector* is responsible for enhancing domain-agnostic provenance traces with domain semantics. It associates semantic types to the data processed by the workflow at two levels: 1) types expressed by experiment designers and service curators, for example "*path:mmu04010* is of type *biopax:pathway*", where *biopax:pathway* is an annotation to a service message type; and 2) types annotations that are curated by domain experts and published in existing public databases. The former are automatically carried through to the provenance graph from the corresponding annotated workflow specification, as described below, while for the latter, we specifically consider annotated Linked Data values that matches the values found in the trace. As a proof of concept, we have mapped Janus values to identifiers from the Bio2RDF project, the nucleus of the life science-related Linked Data. For example, the KEGG identifier used earlier corresponds to the Bio2RDF entry for MAPK signaling pathway, i.e., http://bio2rdf.org/path:mmu04010.

• The *semantic provenance query infrastructure* relies on a set of provenance query operators to support the provenance queries including those that involve computing the complete derivations by transitive closure over the semantically annotated Janus S+ provenance descriptions. In addition, the query infrastructure proposes to support hybrid queries that use the Web of Data. This makes published Janus provenance graphs an integral part of the Web of Data: the two aspects of information become accessible as one data space for the semantic query module.

The Janus S+ Ontology

Bridging the gap between the domain-agnostic provenance traces and the users' domain-oriented view of provenance requires a provenance data model that is capable of incorporating domain semantics as part of the

[5] http://www.bio2rdf.org

provenance descriptions. The Janus provenance from Taverna is used to track domain-neutral provenance traces during workflow executions. To include domain semantics in these traces, we need to add new concepts to the Janus ontology. As a proof of concept, we consider two initial types of domain semantics: 1) the type of data processed by the workflow, and 2) the source providing the data. We reuse concepts like *protein*, *gene*, *pathway* from existing domain ontologies to describe the types of data, and we provide the upper class *janus:domain_entity* to accommodate all these specific data types. Similarly, we describe data sources by defining instances of the common concept *NCI:Data_Sources*, for example *janus:uniprot*.

This reuse approach makes a large number of domain concepts from different ontologies available to us, at the same time requires us to provide semantic interoperability amongst these concepts. For this, we map the Janus S+ Ontology to the provenance upper ontology, Provenir [8], which has been successfully used to supply domain-specific concepts to provenance information to support several bioinformatics applications. As shown in Figure 2, the core Janus concepts are specializations of Provenir concepts. In this way, we not only provide a unified, higher level of the various ontologies but also reuse some of Provenir properties to capture relationships between Janus concepts not previously available. For example, the relationship between *janus:processor_spec* and *janus:port* is now being defined by the object property *provenir:has_parameter*.

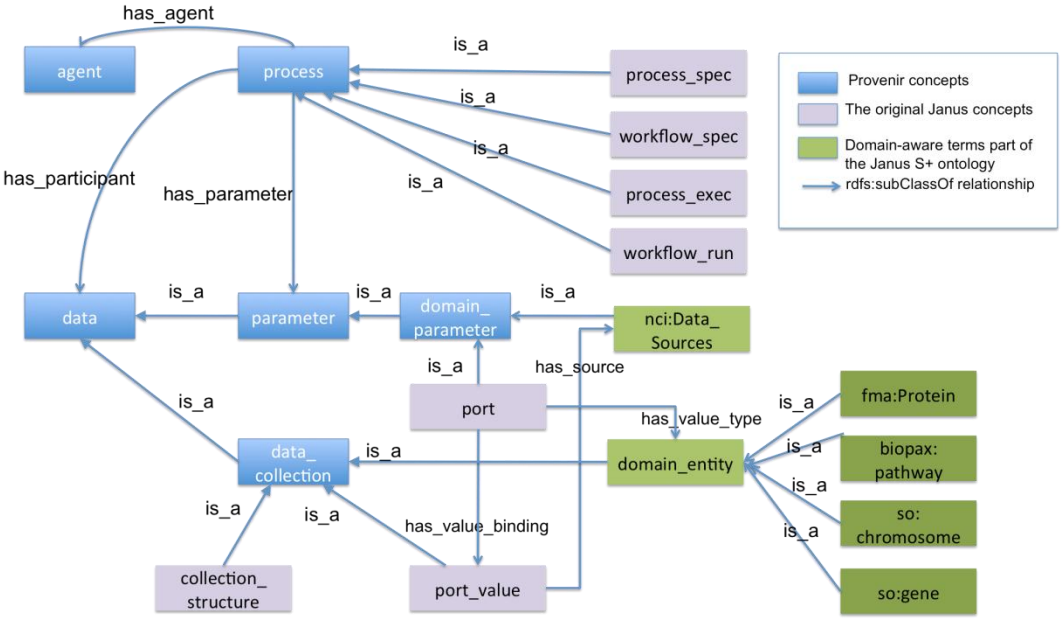

Figure 2. The conceptual model for the Janus S+ provenance ontology. For clarity many actual object and data properties from the Janus and Provenir ontologies are omitted in the diagram.

The Semantic Provenance Collector

This component defines the annotation model for Janus provenance graphs as a two-step pipeline, see Figure 3, resulting in a "Web of Science" that enhances the baseline domain-agnostic provenance graph in order to support the two groups of domain-aware provenance queries.

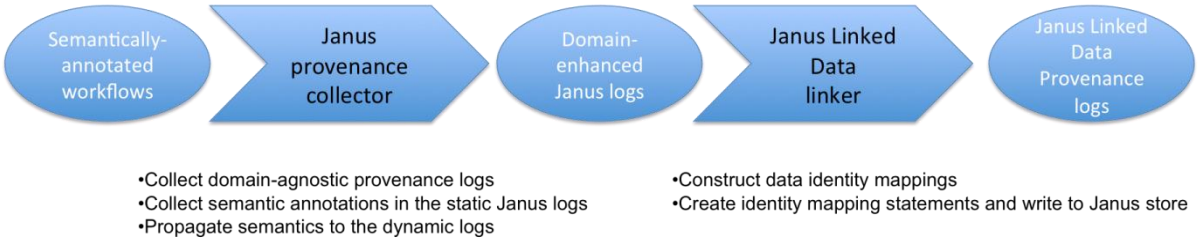

Figure 3. Two-step pipeline for making domain semantics explicit in workflow provenance graphs.

Step 1: annotating Janus provenance with domain semantics

The Janus framework aims to add semantic information as easily as possible. The domain-neutral provenance traces are automatically generated by the Janus provenance capture component during workflow runs as RDF graphs. A trace consists of a *static* segment, which describes the structure of the workflow being executed, and a *dynamic* segment, which contains the values that flow through the workflow processors. With the goal of annotating the values, we exploit any annotation that may be available on the workflow specification itself. We can view a Taverna workflow as a composition of Web Service operations invocations, where a processor's port corresponds to an operation parameter. Suppose that an annotation describing the domain-specific type of the parameter is available, for instance concept "*biopax:pathway*" is associated to the output port *results* of a KEGG database lookup operation. Such annotation carries through to the corresponding processor's port, resulting in an annotation on the static fragment of the provenance trace. Furthermore, assuming that the ports receive and send values of the correct types, we can also propagate the annotation to the dynamic portion, namely we can assign the same concept "*biopax:pathway*" to any value, say *path:mmu04010*, that flowed through the port during execution. Thus, whenever the services are annotated, either by the providers or by members of the consumers' community, we can propagate some of those annotations to the provenance graph (manual annotation of the workflow processors is required when the services are not annotated).

Step 2: linking Janus provenance with the Web of Data

Drawing on third-party domain knowledge about data is one of the key motivations behind life science data integration research. Janus offers a lightweight approach, building upon our existing provenance framework and adding "just enough knowledge" to answer the most common user questions. To do this, we take advantage of the rich content published using Linked Data principles. Connecting RDF Janus provenance graphs to the Web of Data requires (i) publishing the RDF graphs onto the Web, and (ii) creating appropriate links to the open Web of Data (WoD).

The latter step involves mapping the RDF resources that represent data values in the provenance graphs, to corresponding WoD resources. This is a knowledge-rich and thus costly operation on a large scale. Following our "just enough" principle, and to maximize the benefit of this activity, we have chosen to map Janus identifiers to resources that sit at the centre of an existing, richly interconnected WoD for the Life Sciences, namely the Bio2RDF dataset. This fulfills our immediate need to access the wide range of life science-related knowledge to assist our data interpretation queries (Q4-Q7).

String labels, providing names or descriptions about data resources, are often used to create data links [9]. However, biological entities are notorious for their ambiguous names and diverse synonyms. Janus uses both labels and semantic types information about data for creating the mapping, all of which are captured in the Janus-compliant RDF graphs. The second part of the annotation pipeline includes: accessing the Janus provenance store using SPARQL queries; retrieving data entities that are semantically annotated by the first part of the pipeline, and constructing their mapping Bio2RDF data identities using a set of rules. The resulting identity mappings are then stored in the Janus RDF store, together with the provenance traces and their semantic annotations. This store is then made accessible on the Web and compliant with the Linked Data standard using existing Linked Data publication tools, such as Pubby[6].

Semantic Provenance Query Infrastructure

The Janus query infrastructure involves the use of dedicated provenance query operators defined as part of the Provenir ontology-based semantic provenance project [8] and the use of the Linked Data query engine SQUIN[7]. The operators allow to (i) retrieve the provenance of a data entity, (ii) retrieve data entities that satisfy constraints on the provenance graph, and (iii) compare and merge multiple provenance graphs. The operators enable users to query provenance without having to manually compose a query pattern. For example, in case of a RDF-based Janus provenance implementation, given an input value the denotational semantics of the query

[6] http://www4.wiwiss.fu-berlin.de/pubby/
[7] http://squin.sourceforge.net/

operators are mapped to the corresponding SPARQL query pattern. We now briefly describe how two of the provenance query operators answer some of the example queries introduced earlier in the paper.

1. ***provenance()*** - This operator supports the category of provenance queries exemplified by **Q0** and **Q1** and involves a two-phase closure operation on the provenance graph. The operator takes as input a given data entity and returns the complete provenance information associated with it. In case of **Q1**, the input value is a KEGG pathway identifier such as *path:mmu04010* and the output is the set of genes that are found in the provenance graph for the input pathway.
   The *provenance()* operator executes in two steps, namely Initialization and Recursion. For **Q1**, during initialization, all computational tasks linked to the input value are retrieved including the *Gene→Pathway* task (Figure 1). During recursion, entities associated with the computational tasks are retrieved, e.g. the names of the genes involved in *path:mmu04010*.
2. ***provenance_context()*** – This operator supports questions such as **Q2** and **Q3**. It returns data entities that satisfy a set of constraints expressed in terms of provenance values (forming the input to the query operator). In case of **Q2**, the input is set of genes that have been retrieved from *UniProt* by the *EnsemblGene→UniProt Gene* task, and the output is set of pathways derived from these genes (thereby excluding the genes derived from *Entrez gene*).

We now turn to provenance questions like **Q4-Q7**. Using a Linked Data query engine like SQUIN, we can query distributed RDF datasets as if they were co-located. Our hybrid approach incorporates the provenance query operators described above along with SQUIN: 1) we use SQUIN to search for data entities that satisfy certain properties (such as genes related to given functions) and domain-specific information about data entities (i.e. protein functions, pathway-related diseases, and related publications), and then 2) we use the operators to query the provenance graph. In case of **Q4**, SQUIN retrieves the type information associated with the pathways output by the *Gene→Pathway* task (Figure 1) to identify a subset of pathways related to the immune system. Then the *provenance()* operator retrieves the set of genes associated with this subset. Finally, SQUIN retrieves the gene function annotations associated with genes returned by *provenance()*.

Conclusion

We have explored the synergy between semantic provenance and Linked Open Data, in the context of the Life Sciences. As provenance capture and query capabilities become commonplace for *in silico* experiments, and the Web of Data begins to incorporate large fragments of public Life Science databases, this convergence may prove critical to the success of both. Starting from a simple classification of increasingly complex user questions, we have described a prototype model and implementation for Linked-data aware semantic provenance infrastructure, called Janus. The model helps us to express the domain semantics required for supporting the example user questions. To assess the viability of our approach, we are currently extending our model to describe further Life Science domains. The domain semantic annotations enable us to interlink Taverna provenance RDF graphs with Bio2RDF dataset and make them as part of the Web of Data. Our future work is to implement the mappings between workflow data identifiers and extend our supports for annotations on the workflows.

Acknowledgements

This work is funded by NIH RO1 Grant# 1R01HL087795-01A1 and by EPSRC Grant # EP/G049327/1.

References

1. Sahoo, S.S., Sheth, A., Henson, C., *Semantic Provenance for eScience: Managing the Deluge of Scientific Data*. IEEE Internet Computing, 2008. **12**(4): p. 46-54.

2. Hull, D., Wolstencroft, K., Stevens, R., Goble, C., Pocock, M.R., Li, P., and Oinn, T., *Taverna: a tool for building and running workflows of services.* Nucleic Acids Research, 2006(34(Web Server issue)): p. W729-W732.
3. Missier, P., Paton, N., Belhajjame, K. *Fine-grained and efficient lineage querying of collection-based workflow provenance*. in *EDBT*. 2010. Lausanne, Switzerland.
4. Missier, P., Sahoo, S.S., Zhao, J., Goble, C., Sheth, A., *Janus: from Workflows to Semantic Provenance and Linked Open Data*, in *IPAW 2010*. 2010: Troy, NY.
5. *Open Provenance Model*. Available from: http://twiki.ipaw.info/bin/view/Challenge/OPM.
6. Anand, M.K., Bowers, S., McPhillips, T.M., Ludäscher, B. *Exploring Scientific Workflow Provenance Using Hybrid Queries over Nested Data and Lineage Graphs.* in *SSDBM 2009*. 2009. New Orleans, Louisiana USA.
7. Anand, M.K., Bowers, S., Ludäscher, B. *Provenance browser: Displaying and querying scientific workflow provenance graphs*. in *ICDE*. 2010. Long Island CA.
8. Sahoo, S.S., Barga, R.S., Sheth, A.P., Thirunarayan, K., Hitzler, P., *PrOM: A Semantic Web Framework for Provenance Management in Science*. 2009, Kno.e.sis Center, Wright State University.
9. Bizer, C., Volz, J., Kobilarov, J., Gaedke, M. *Silk - A Link Discovery Framework for the Web of Data*. in *WWW*. 2010. Madrid, Spain.