The Semantic Web: Service discovery and provenance in ^{my}Grid

Phillip Lord, Pinar Alper, Chris Wroe, Robert Stevens,
Carole Goble, Jun Zhao, Duncan Hull and Mark Greenwood
Department of Computer Science
University of Manchester
Oxford Road
Manchester
M13 9PL UK

September 14, 2004

1 Introduction

The vision of the Semantic Web (SW) is one "in which information is given well-defined meaning, better enabling computers and people to work in cooperation." (Berners-Lee et al., 2001). It emphasises the decentralised and autonomous nature of the data over which it operates, as well as the complexity of this data. At first sight this seems to fit extremely well with the requirements of the life sciences. The nature of the area makes extreme complexity the rule rather than the exception. Moreover, the history of the subject has ensured that most resources are decentralised and autonomous.

On the face of it then, SW technologies offer a good technological solution for some of the difficulties of the Life Sciences, while the Life Sciences offer a perfect use-case for Semantic Web. Here we discuss two applications that have used semantic web technology and discuss the strengths and weaknesses of this technology, as well as its implications.

2 Applications

The ^{my}Grid project is part of the UK e-Science program. The major aim of ^{my}Grid has been to develop middleware and user facing applications which enable the biologist to develop and execute *in silico* experiments. Traditionally, this has involved the biologist or, more frequently, the expert bioinformatician writing bespoke Perl scripts. These have generally made heavy use of screen scraping technology, which are difficult to code, difficult to share, fragile, and unsuited for large amounts of data. The ^{my}Grid project has used a number of specific use cases from biology (Williams-Beuren Syndrome (Stevens et al., 2004) and Graves Disease (Li et al., 2004)) as seed toward the development of middleware which attempts to avoid these problems.

To enable the use of distributed services ^{my}Grid has developed a Web Services based architecture, which includes a large number of biological services through the SOAPLAB framework (Senger et al., 2003), a workflow development environment (Oinn et al., 2004), and a workflow enactment engine (Addis et al., 2003). To facilitate communication, ^{my}Grid components adhere to a common information model (Sharman et al., 2004), whose role is to provide shared data abstractions that underpin important service interactions and so promote synergy between myGrid components. The information model captures the e-science process, including the collection of services, workflows, data, experiments, people, projects, types, provenance and annotation.

Around these core "Web" technologies, we have investigated the use of SW technologies. All objects in the model are identified by URN-based Life Science Identifiers (LSIDs) and all objects can be annotated with one or more concepts drawn from an OWL ontology. Thus we are well positioned to build a "Semantic Web" of bioinformatics in silico experiments. More specifically we have applied SW technologies to two problems: provenance and service/workflow discovery. Next we describe these two problems and the solutions which we have developed to them. Both of these applications use

RDF data models, both use an explicit link to an ontology of domain knowledge, and both involve search and retrieval. The intention of richly characterising them both with extensible metadata is that they are intended to attract multiple descriptions from different viewpoints held by different stakeholders, and both are expected to be interpreted and used in unanticipated ways by others unknown to their authors or providers. They are described in more detail elsewhere (Zhao et al., 2004; Lord et al., 2004).

2.1 Provenance

Typically in a wet lab environment, biologists record large quantities of information about the materials, methods and goals of the experiments that they perform. This information serves as the *provenance* of their experiments and, later, their experimental results. This information is of importance for many different stakeholders: For scientists validity checking existing results, or updating old results; for supervisors to summarise information about progress and to aggregate data from a number of researchers in a lab; and for members of other external research groups to check results in detail, regulatory authorities to ensure accuracy, or legal departments for pursing IPR claims.

Within bioinformatics much of this data has been generated and stored by the expert curator, often as free text (such as the PubMed citations within UniProt) or loosely structured (such as the Evidence Codes within GO). ^{my}Grid, however, because it offers standardised facilities for accessing the data has enabled the automatic gathering of this provenance data, in the form of *provenance logs*. Each piece of data used in an *in silico* experiment, the tools used to analyse this data, and the associations between other data needs can be recorded.

The basic architecture for gathering this provenance is shown in Figure 1. Two main forms of provenance are gathered: intermediate data, and metadata.

Intermediate data is of interest because the source databases change frequently and unless this data is stored locally it can be impossible to infer from the final results where a particular piece of information came from. Large parts of bioinformatics is flat-file based: data is both produced and consumed by services in a variety of complex flat-file, "human-readable" formats. Thus we are required to store metadata that relate these files and the files themselves.

The metadata covers a variety of different kinds of knowledge, of which the most interesting in this context are: **Data Derivation Provenance** builds a graph of data objects in a workflow run, including intermediate and final results. **Domain Provenance** stores domain specific links; so, a nucleotide sequence output from a BLAST search will be linked as a *similar sequence* to the input sequence

The data results arising from a workflow run is either stored in the ^{my}Grid Information Repository (mIR) data store (implemented as a relational database) or in a suitable specialist data store. The provenance metadata is stored using RDF in the mIR metadata store (using Jena). This technology was chosen to represent the model because: i) It provides a more flexible, graph based model, as opposed to an XML tree; ii) It provides an explicit identification system (URI's) for resources which allow metadata to be merged from several sources; iii) It provides an well-defined association with an ontology; iv) From a practical point of view, there are several mature, open-source, repositories are available for use.

While this split between data and metadata is both technically appealing and necessary, it requires that some common mechanism exists to related between the two kinds of data. For this, ^{my}Grid has used LSID's. We could have used URL's or applied other additional semantics to a URI, but LSID's provided us with a well-defined mechanism for resolving identifiers into data and metadata. The use of LSID's is attractive because of the efforts to standardise the specification through OMG[1] which has resulted in both freely available infrastructure support and promising increasing uptake within the domain. Finally, LSID's provide an explicit social commitment to the maintenance of immutable and permanent data: an LSID should always resolve to the same physical bytes of data, which is clearly an explicit requirement for storing of provenance data.

LSIDs provide a convenient access mechanism to the provenance of an object. Using the LSID metadata protocol, an object can serve the RDF triples that present its origin, which is a useful mechanism when objects are shared between applications or exported.

As well as suitable technologies for storing this

[1] Object Management Group, http://www.omg.org

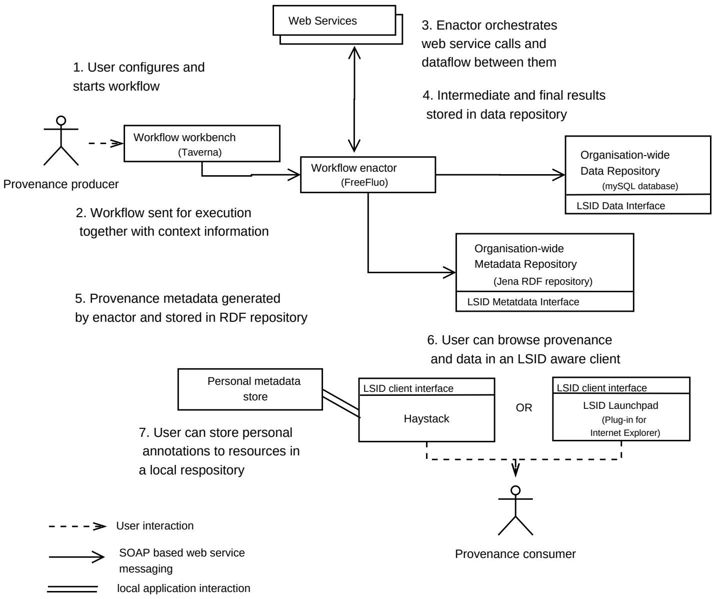
Web Services
3. Enactor orchestrates web service calls and dataflow between them
1. User configures and starts workflow
4. Intermediate and final results stored in data repository
Workflow workbench (Taverna)
Workflow enactor (FreeFluo)
Organisation-wide Data Repository (mySQL database)
LSID Data Interface
Provenance producer
2. Workflow sent for execution together with context information
Organisation-wide Metadata Repository (Jena RDF repository)
LSID Metatdata Interface
5. Provenance metadata generated by enactor and stored in RDF repository
6. User can browse provenance and data in an LSID aware client
Personal metadata store
LSID client interface
Haystack
OR
LSID client interface
LSID Launchpad (Plug-in for Internet Explorer)
7. User can store personal annotations to resources in a local respository
User interaction
SOAP based web service messaging
local application interaction
Provenance consumer

Figure 1: An architecture for provenance

data, we also need to present it back to the user. To date we have used the Haystack browser. As well as natively understanding the LSID protocol, it provides us with convenient facilities for filtering the RDF graph which is generated. This is essential as a complex, highly-connected, RDF graph quickly becomes impossible to display and interact with. However, the visual complexity is daunting, suggesting multiple view mechanisms over RDF to be a necessity.

The real benefit of using RDF should come when we integrate and aggregate across the provenance of different workflow runs, and across different experiments. We should also be able to assert new claims over data results by grounding these against the provenance statements of workflows as the provenance record of a workflow is the "proof" (cf the Semantic Web language layer model) of its outcome. Testing these hypotheses is the current focus of our work.

2.2 Service Discovery

Along with other projects such as Bio-Moby, there has been a focus within ^{my}Grid, on applying SW techniques to service discovery. Unlike many SW Services Approaches (Lara et al., 2003), ^{my}Grid has focused on providing *user-oriented* service discovery, as opposed to fully automatic discovery and composition. Within bioinformatics the user community are expert, knowledgeable, and opinionated, and may invest large amounts of time and money in further experiments based on early results. We wish to support biologists' activities (via decision support applications), rather than replace them.

Hence we have built a service discovery framework, called Feta, that supports a simplified, user-oriented data model for representing service descriptions. The model embodies not only basic service descriptions (eg. service inputs/outputs) but also descriptions of bioinformatics influenced characteristics of a service (eg. service task, service algorithm, service resource).

Besides being user-oriented the data model is built with workflows in mind: descriptions of different types of software entities, as well as "plain" web services, can be generated, stored and queried. Feta provides the users with the necessary decision support while they are building workflows.

The basic components of our architecture, shown in Figure 2, are, however, shared with other SW Services approaches. i) A domain ontology is used to provide a common vocabulary for describing services. We are investigating ways in which this domain ontology can also be used to represent the Domain Provenance described in Section 2.1. ii) A user tool, called PeDRo, is used to generate service descriptions. iii) The Feta engine loads and searches these descriptions. iv) Finally, an extension to Taverna provides a user interface to the Feta engine.

The SW technologies employed within the context of service discovery in ^{my}Grid can be summarized as follows. The domain ontology has been generated using OWL[2] (Wroe et al., 2003), which has been particularly selected for its formal semantics and rich expressivity. The ontology development process has been supported by the use of a DL reasoner (FaCT, or RACER). Initially the reasoner was used during service discovery to enable exploitation of expressivity of OWL (Lord et al., 2003). However, this added expressivity was poorly used and came at the expense of practicality. In our current implementation, Feta, we have developed an RDF based data model designed to support user-oriented querying and discovery. We prereason over the OWL ontology at development time in to a fully classified hierarchy. We use this "materialised" view of the ontology to annotate entries in the registry at the time of their publication, and use the RDFS entailment facilities within Jena, which enables specialisation and generalisation of queries, to answer queries at the time of their discovery.

Future work on Feta will focus on extending the descriptions of services with non-functional aspects and querying over them. Additionally, we wish to both extend and simplify the ontology, so that we can present alternative views of the ontology to different user communities. Finally, we are investigating techniques for providing automated service composition over a sub-set of services, which we call "Shim Services", which are experimentally neutral but required within workflows, such as format transformations or filters.

3 Experiences

Our experiences with the use of SW technologies within ^{my}Grid lead us to a number of conclusions.

Evolution not Revolution: Our application of SW to provenance and service discovery will not radically alter the way biology is performed. They do seem to support our specific applications reasonably well, although currently our applications are at a prototype stage.

Decision Support not Decision Making: In the short term, biologists will wish to monitor the results of SW technologies closely, until they fully trust it. With limited exceptions, we need to aid the users decision process, not replace it.

Tool Use not Tool Generation: Getting knowledge from users and presenting it to them is hard. There is a severe lack of user facing tools at the moment. This includes tools for the developer, the bioinformatician, and the biologist. Within ^{my}Grid, we have generated, or customised many tools ourselves (for editing ontologies, for maintaining and versioning ontologies, for generating annotation, for viewing). If SW is to be used widely within bioinformatics this barrier to entry must be lowered.

Users vs Machines: Both service/workflow discovery and provenance management have highlighted the conflicting requirements of these two communities. One the one hand comprehensive models captured when publishing experimental components seem desirable; on the other hand they are too complicated to be comprehensible to users. This suggests that view and filtering mechanisms over RDF graphs is crucial.

There are also a number of areas where we are less certain. **Scalability** has proven to be an issue for both of these applications, although is particularly true with provenance data which will potentially be produced in huge quantities. Both of these applications use **mixed models**. Provenance data is stored partly in RDF, and partly in a RDBMS, while Feta makes use of both XML (for generating and storing service descriptions) and RDF (for querying). We have partitioned the data in a pragmatic rather than principled manner[3]. Finally, whilst **aggregation** promises to enable common querying over data coming from a variety of different sources, we have yet to demonstrate its utility with large scale "real world" examples.

References

M. Addis, J. Ferris, M. Greenwood, D. Marvin, P. Li, T. Oinn, and A. Wipat. Experiences with escience workflow specification and enactment in bioinformat-

[2] We initially used DAML+OIL

[3] Mostly because we are unsure what the principles are!

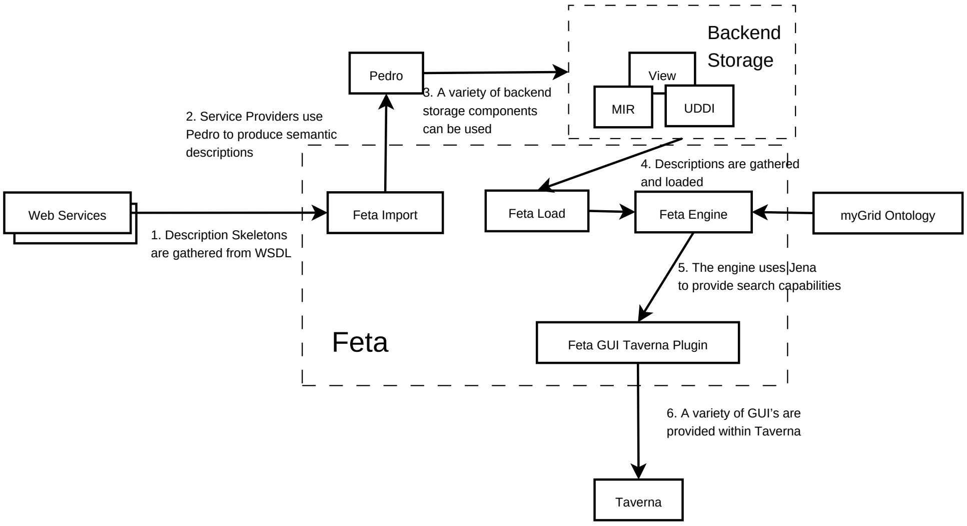
Backend
Storage
View
MIR
UDDI
Pedro
3. A variety of backend storage components can be used
2. Service Providers use Pedro to produce semantic descriptions
4. Descriptions are gathered and loaded
Web Services
Feta Import
Feta Load
Feta Engine
myGrid Ontology
1. Description Skeletons are gathered from WSDL
5. The engine uses Jena to provide search capabilities
Feta
Feta GUI Taverna Plugin
6. A variety of GUI's are provided within Taverna
Taverna

Figure 2: The Feta service discovery architecture

ics. In *Proceedings of UK e-Science All Hands Meeting*, pages 459–467, 2003.

T. Berners-Lee, J. Hendler, and O. Lassila. The semantic web. *Scientific American*, May 2001.

R. Lara, H. Lausen, S. Arroyo, J. de Bruijn, and D. Fensel. Semantic web services: description requirements and current technologies. In *Proceedings of the International Workshop on Electronic Commerce, Agents, and Semantic Web Services. (ICEC 2003)*, 2003.

P. Li, K. Hayward, C. Jennings, K. Owen, T. Oinn, R. Stevens, S. Pearce, and A. Wipat. Association of variations in nfkbie with graves' disease using classical and mygrid methodologies. In *Proceedings of UK e-Science All Hands Meeting*, 2004.

P. Lord, S. Bechhofer, M. D. Wilkinson, G. Schiltz, D. Gessler, D. Hull, C. Goble, and L. Stein. Applying semantic web services to bioinformatics: Experiences gained, lessons learnt. In *International Semantic Web Conference*, 2004. Accepted For Publication.

P. Lord, C. Wroe, R. Stevens, C. Goble, S. Miles, L. Moreau, K. Decker, T. Payne, and J. Papay. Semantic and Personalised Service Discovery. In W. K. Cheung and Y. Ye, editors, *WI/IAT 2003 Workshop on Knowledge Grid and Grid Intelligence*, pages 100–107, Halifax, Canada, Oct. 2003. ISBN 0-9734039-0-X.

T. Oinn, M. Addis, J. Ferris, D. Marvin, M. Greenwood, T. Carver, A. Wipat, and P. Li. Taverna: A tool for the composition and enactment of bioinformatics workflows. *Bioinformatics*, 2004. Accepted for publication.

M. Senger, P. Rice, and T. Oinn. Soaplab – a unified sesame door to analysis tools. In *Proc UK e-Science programme All Hands Conference*, 2003.

N. Sharman, N. Alpdemir, J. Ferris, M. Greenwood, P. Li, and C. Wroe. The myGrid Information Model. In S. J. Cox, editor, *Proce UK e-Science programme All Hands Meeting*, pages 287–293. EPSRC, September 2004.

R. Stevens, H. Tipney, C. Wroe, T. Oinn, M. Senger, P. Lord, C. Goble, A. Brass, and M. Tassabehji. Exploring Williams Beuren Syndrome Using $^{\mathrm{my}}$Grid. In *Bioinformatics*, volume 20, pages i303–310, 2004. Intelligent Systems for Molecular Biology (ISMB) 2004.

C. Wroe, R. Stevens, C. Goble, A. Roberts, and M. Greenwood. A Suite of DAML+OIL Ontologies to Describe Bioinformatics Web Services and Data. *The International Journal of Cooperative Information Systems*, 12(2):597–624, 2003.

J. Zhao, C. Wroe, C. Goble, R. Stevens, D. Quan, and M. Greenwood. Using Semantic Web Technologies for Representing e-Science Provenance . In *3rd International Semantic Web Conference (ISWC2004)*, 2004. To appear.