The Semantics of Genomic Mapping

Robert Stevens,[1]
Andrew Gibson,[1]
Matthew Pocock,[2]
[1] University of Manchester, Manchester, UK;
[2]University of Newcastle, Newcastle Upon Tyne, UK
robert.stevens@manchester.ac.uk

September 15, 2006

1 Introduction

This abstract gives an overview of the ComparaGRID project,[1] which uses Semantic Web technology to facilitate comparative genomics studies, especially using non-model organisms where complete sequence information and the means to obtain it might be lacking. Comparative genomics in model organisms has many features in common with any integration scenario in bioinformatics. Many organisms have its own databases, with their own schema, their own representation for data and their own vocabulary for describing those data, both at schema and value level. The autonomy and distribution inherent in bioinformatics makes this situation of semantic and syntactic heterogeneity somewhat inevitable.

The typical scenario for ComparaGRID is where a biologist has a region of interest in organism *x*, for example a QTL, for which there is little information. In organism *y*, however, the biologist knows there is a corresponding QTL or gene from *x*'s QTL around which there is a lot more information about the presence and location of genes. This information can be in a variety of forms, from standard detailed annotation of sequence, to a variety of maps such as cytogenetic maps, linkage maps, etc. The general idea in ComparaGRID is to be able to exploit these data on presence and ordering of genes—the synteny—to facilitate analysis of the under-characterised region in an organism.

In many respects this is an example of the classic data and schema reconciliation problem that are all too common in bioinformatics. Semantic Web technologies offer a means of addressing these perennial problems by attacking the semantic issues at root. We can use Semantic Web technologies such as OWL-DL to describe the entities and ideas of a domain and the relationships between those entities and ideas. The way the data representing those entities and ideas in the bioinformatics databases can be represented in terms of that ontology as objects that are instances of the classes in the

[1] http://www.comparagrid.org

ontology. Thus we can query the bioinformatics resources that hold the “instances” described in the ontology of the domain.

2 The ComparaGRID Ontology

The challenge for the ontology at the heart of ComparaGRID has been that not only does it need to act as a controlled vocabulary for discrete data sources, but also that it has to mediate biological knowledge that has become adapted to fit different models and research patterns. One research pattern is that of more traditional genetics in which knowledge of genetic information is modelled as maps of chromosomes with relative distances between areas of interest (aka markers). The comparative genomics begins when the gene content or gene order of a chromosome (or region of a chromosome) between two organisms can be used to identify areas that should be studied more closely. Another mode of research emerges from the post-genomic era, where sequence representations of chromosomes (or regions of chromosomes) can be used to accurately predict the functional content of that region or chromosome through direct comparison of sequences with other representations from other organisms.

The scope of the ontology is to cover the domain of comparative genomics, which is based strongly on both genetic models and genomic representations described above. The divergent knowledge from these two domains needs to be integrated however, so the anchor through which the knowledge can be traversed from one to the other is the description of the physical biological entities that the former fields model and represent respectively. Also, the ontology needs to be able to treat much of the knowledge of comparative genomics as the outcomes of experimental techniques, so we include the ability to render the performance and outcomes of experiments as well as subsequent assertions that some things are non-physically related (e.g. orthology).

The extent of the scope of the ontology may seem largely unnecessary to those in the domain, but we insist that this verboseness is necessary. A biologist handling biological data (usually) has the understanding to handle the data in the appropriate manner. A computer does not. Making the ontological distinctions above makes sure that only the appropriate actions are performed on the data in the client databases. For example, knowing that a DNA sequence is only a representation of a molecule in the cell as opposed to the actual thing in the cell means that we are forced to be explicit about what is a facet of the representation and what is a property of the physical thing.

3 The ComparaGRID Application

A user facing application called *PussyCat* provides access to a ComparaGRID knowledge base—a combination of an ontology (T-box) and instances of classes in that ontology (A-box). Fluxion is the ComparaGRID data-integration architecture, designed to allow sources with incompatible representation, syntax and semantics to be combined into such a single virtual knowledge base. Underlying data resources provide their data as OWL individuals that populate classes in the ontology. To support distribution, an implementation of Fluxion is exposed through a Web-service interface.

A typical deployment of Fluxion will have a three-teir architecture. The first tier, *the data publisher*, will expose raw data as OWL statements. There are currently data publisher back-ends for several data storage paradigms, including RDBMS/SQL (via JDBC), BioMart and tab-delimited text files. These statements are then exposed through the Web-service interface, and portions of the OWL A-box are dynamically populated in response to incoming queries.

The middle *translation* tier contains rules that map between concepts in the database-schema OWL and a domain ontology modelled in OWL.

The top *integration* tier is responsible for taking complex queries and sending on relevant portions of them to multiple translation tier services. It then collects the multiple results and performs any reasoning necessary to join these datasets back up again. End-user applications will typically interact directly with an integration service.

4 Discussion

This layered architecture allows a progressive reconciliation of diverse and distributed data in order to answer queries. The ontology both describes the domain so that underlying data can be mapped to a common form and makes distinctions in modelling those data such that semantically sensible, consistent mappings become possible. The separation into layers allows different publishers, translators and integrators to be combined, making the Fluxion architecture generic. Semantic Web recommendations offer suitable technologies for such an approach.

Acknowledgements: Funding for this work was through BBSRC grant BBS/B/17156.