Solutions for data integration in functional genomics: a critical assessment and case study


Submitted: 2nd July 2008; Received (in revised form): 8th September 2008

Abstract

The torrent of data emerging from the application of new technologies to functional genomics and systems biology can no longer be contained within the traditional modes of data sharing and publication with the consequence that data is being deposited in, distributed across and disseminated through an increasing number of databases. The resulting fragmentation poses serious problems for the model organism community which increasingly rely on data mining and computational approaches that require gathering of data from a range of sources. In the light of these problems, the European Commission has funded a coordination action, CASIMIR (coordination and sustainability of international mouse informatics resources), with a remit to assess the technical and social aspects of database interoperability that currently prevent the full realization of the potential of data integration in mouse functional genomics. In this article, we assess the current problems with interoperability, with particular reference to mouse functional genomics, and critically review the technologies that can be deployed to overcome them. We describe a typical use-case where an investigator wishes to gather data on variation, genomic context and metabolic pathway involvement for genes discovered in a genome-wide screen. We go on to develop an automated approach involving an in silico experimental workflow tool, Taverna, using web services, BioMart and MOLGENIS technologies for data retrieval. Finally, we focus on the current impediments to adopting such an approach in a wider context, and strategies to overcome them.

Keywords: data integration; interoperability; web services; BioMart; MOLGENIS; Taverna

INTRODUCTION

In the post-genomic era, the key challenges for the discovery of gene function and the development of mouse models for human disease relate to detailed trait analysis and the phenotyping of simple and compound mutants [1, 2]. One aspect of this

*These authors contributed equally to this work.

Corresponding author. Damian Smedley, European Bioinformatics Institute, Wellcome Trust Genome Campus, Hinxton, Cambridge, CB10 1SD, UK. Tel: +44-1223-494451; Fax: +44-1223-494468; E-mail: damian@ebi.ac.uk

The authors are bioinformaticians working together as part of the CASIMIR project, which aims to develop policy and principles to underpin the development of data integration in the biosciences.

Damian Smedley and Glenn Proctor are bioinformaticians working at the European Bioinformatics Institute, Wellcome Trust Genome Campus for the Protein and Nucleotide Database Group.

Morris Swertz is currently developing the MOLGENIS system at the Groningen Bioinformatics Centre at the University of Groningen.

Katy Wolstencroft is part of the myGrid group at the University of Manchester, developing Taverna amongst other applications.

Michael Zouberakis is part of the Database Developing Team at the Alexander Fleming Biomedical Sciences Research Center in Athens.

Jonathan Bard's research in the Computational Biology Research Group, John Radcliffe Hospital, Oxford is focussed on the development of biomedical ontologies, especially those for anatomy.

John Hancock heads the Bioinformatics group at the MRC Harwell centre for mouse genetics.

Paul Schofield's team at the Department of Physiology, Development and Neuroscience, University of Cambridge develops the Pathbase mouse mutant pathology database as well as other research activities.

© The Author 2008. Published by Oxford University Press. For Permissions, please email: journals.permissions@oxfordjournals.org
development, substantially overlooked at the outset, was that with the explosion in high-throughput phenotyping technologies storage, retrieval and analysis of data would be almost as great a technical challenge as their generation. The ability to undertake data mining and computational analysis on a wide variety of data sources and data types has the potential to add enormous power to experimental analysis. However, data and computational resources are spread around the world in databases funded and managed by diverse institutions with their own interests and remits. The challenge therefore is to facilitate integrated use of all these resources to maximize the power of both computational and experimental approaches to the discovery of gene function. In the mouse biology community, the wide range of experimental data being collected compound this problem [3]. Mouse data spans genotype to phenotype, with systematic endeavours like the mouse knockout programmes [4–6], the collections of -omics data and high-throughput analysis datasets [7] generating enormous amounts of information.

In order to appreciate and analyse the problems involved in database interoperability CASIMIR (coordination and sustainability of international mouse informatics resources) [8] was funded by the European Commission under Framework Programme 6. CASIMIR is a coordination action that focuses on the integration of the informatics infrastructure that underpins our current efforts in functional genomics and understanding the biology of human disease through generation and validation of mouse models. The following critical discussion is the result of community consultation and expert review of the problems and potential solutions to the issues of interoperability, with specific reference to mouse functional genomics.

A typical data-integration problem is the gathering of data from various sources to find evidence for linkage between particular phenotypes and genes. This could well involve gathering data on phenotype differences and allelic variants between strains, genotypes (gene-related data for those in particular genomic locations) and pathways in which these genes may be involved. The current difficulty is that biologists need to be aware of where these different resources are, and what they can offer. They also need to learn how to use the different user interfaces, copy and paste data and identifiers from website to website, convert them to make them match, and merge all these partial results into an Excel or Word document by hand. This difficult and error-prone process does not scale up to the hundreds or thousand of genes typically found in whole-genome experiments.

Mouse-centric resources, like many biological databases, present problems for integrated querying for a number of reasons. First, there is the huge amount and variety of data that, by necessity, leads to heterogeneity in the way the data are represented in the different resources. Second, the fact that all the resources are autonomous means their designs and/or schema can change independently of each other. Finally, each source will have its own querying capability: some will have just a web interface whilst others will have sophisticated programmatic interfaces.

This article presents our initial findings along with a proof of principle demonstration arising from the first CASIMIR co-ordination meetings held in Corfu and Rome in 2007 [9]. We describe a combination of technologies that can solve this interoperability problem and present a pilot using web services [10], BioMart [11, 12], MOLGENIS [13, 14] and Taverna [15, 16].

INTEGRATION STRATEGIES
There are generally two strategies for integrated querying of biological resources [17]. The first is a centralized architecture where all the data are stored locally in a tailor-made data warehouse. Almost by definition, this approach tends to be data-model driven and suited to a particular domain of expertise. The strategy requires considerable programming effort to bring in new data and keep the existing data up-to-date and hence is fairly inflexible. It is, however, a good approach where the usage pattern is well defined and unlikely to evolve. The Mouse Genome Database (MGD) with its set of integrated resources [18, 19] is a good example of such an approach for the mouse community.

The second approach is more flexible and loosely coupled, where distributed resources are combined dynamically. In this approach, the underlying data sources remain autonomous but they cooperate to share some functionality. No domain-specific abstractions are used, instead the generic features of data are modelled and some kind of query-based logic is used for their application programming interface (API) abstractions. This loosely coupled approach can be used in a local or distributed context although the
latter is often preferred as the domain expertise at each centre can be used to configure how and what data are presented to the user.

The second approach is gaining in popularity because of its flexibility. As researchers may not have the expertise or storage facilities to build local data warehouses, or may not have access permission to download complete (and frequently updated) local copies, the first approach becomes unsuitable. In this article, we will concentrate on the second approach of a loosely coupled, client-centred integration and discuss technologies for making distributed data sources interoperable. Stein [20] defined what is needed to create such a dynamic ‘bioinformatics nation’. Data sources need to provide commonly accepted data formats and access methods, and a directory service that allows bioinformaticians or automated scripts to find them. ‘Web services’ is an example technology for building such common access methods (using SOAP), formats (using XML) and directories of services (using WSDL).

Unfortunately, the creation of web service access requires additional implementation work, which often does not appear to directly benefit local users or database owners. BioMart and MOLGENIS are technologies that provide a solution for this problem using two different approaches. BioMart provides a service-enabled, query-optimized warehouse into which to import data, whilst MOLGENIS uses the approach of creating software wrappers around existing databases enabling automated data access from R, Java and web services. Once data sources have been made interoperable, some sort of client is then required to make the integrated querying possible. Hand-crafted web services clients can be written and BioMart already allows integrated querying of two remote BioMart datasets at once. However for more sophisticated querying over a wider variety of data sources, a workflow engine such as Taverna is the best solution. Figure 1 summarizes this strategy.

TECHNOLOGIES FOR INTEGRATING RESOURCES

In this section, we will describe in more detail some of the useful technologies for database interoperability before going on to demonstrate how a combined approach results in a flexible system for mouse data integration and querying.

Web services

The first step towards integration is to enable computational interaction between the data bearing machines on a ‘technical’ level. This involves the bridging of heterogeneity between computer
platforms, programming languages and modes of interaction of these machines, e.g. Linux or Windows, MySQL or ORACLE, Python or Java, database queries or procedural calls. Web services are often used as a ‘common language’ for such a task being the de facto standard for integrating heterogeneous, autonomous and distributed resources.

The World Wide Web Consortium (W3C) defines web services as ‘a software system designed to support interoperable machine-to-machine interaction over a network’ [10]. While the official definition may seem fairly abstract, in practice the web service architecture is clearly described in a mature suite of protocols. At the core is the SOAP protocol that is based on the simple idea of sending XML shaped messages over computer networks, most notably using the Internet protocols HTTP and HTTPS. How these XML messages should be structured and interpreted is well defined. This ensures that messages assembled by the sender can be parsed by the target. To enable this interplay, data machines provide clear contracts on their particular web services using the Web Service Description Language (WSDL) to list available services and technical details, such as their input and output parameters.

In this architecture, a web service interface (for example ‘findSNPs’) can be developed for a relational database containing Single Nucleotide Polymorphism (SNP) data, with its WSDL describing a string input parameter ‘chromosome’, numeric input parameters ‘start’ and ‘end’ and a string output parameter ‘snplist’. Bioinformaticians can then benefit by using this service in their own client programs or standard workflow programs like Taverna, to combine this data with data from other web server machines. Such clients can in turn be embedded in web-based applications and so provide a human-legible interface. Because the data layer underneath the web service API remains hidden, data mining becomes unburdened and access restriction to unpublished or redundant data can be more easily implemented. In particular, relational database management system (RDBMS) conflicts between databases are eliminated, e.g. ORACLE- and MySQL-managed resources can be queried from the same client.

Web services, by definition, provide a cross-platform, cross-language and cross-RDBMS interoperable interface for resources. Their development, deployment and absorption do however require significant programming skills. Moreover, web services may face performance issues when handling extremely large biological data sets (e.g. microarray data), partly because of bloating of the XML responses and partly due to bandwidth usage for the increased processing requirements. We now consider some core resources that facilitate cross-database data mining.

**BioMart query optimized data management system**

BioMart is an open-source, data management system that can act as a central access point to provide interoperability between databases of interest. Data providers can produce BioMart installations for combining the querying of in-house data with public data in other BioMart databases. If two datasets share common identifiers, or even mappings to a common genome assembly, these can be used to link them together in queries, so allowing distributed federation across many resources. As these queries are constructed through the same user interface, the need for users to learn multiple systems can be substantially reduced.

BioMart is used to access many of the large biological datasets in the public domain, such as dbSNP, Ensembl genomics and PRIDE proteomics data. More recently, it has been used to provide mouse-specific resources as part of EuroPhenome [21, 22], KOMP/EUCOMM high-throughput gene trapping resources [23] and the EurExpress in situ expression resource [24]. Our CASIMIR demonstration BioMart server [25] (Figure 2) contains access to various mouse resources produced by CASIMIR consortia members (Ensembl gene, variation and comparative genomic data, Vega manually annotated genes, EuroPhenome, Eurexpress, high-throughput gene trapping data from the KOMP/EUCOMM) and a prototype BioMart developed for the EMMA mouse strain resource.

For data providers, BioMart simplifies the creation and maintenance of advanced query interfaces over relational databases. If data are already in a relational database, the MartBuilder tool will automatically transform the data into the BioMart schema. Also, the BioMart software comes with a range of query interfaces and administration tools, including an ‘out of the box’ website that can be installed, configured and customized according to local requirements. As such, it is particularly suited for providing the ‘data mining’ like searches of complex biological data.
The BioMart Perl API can access BioMart databases through direct database access or via BioMart’s own version of web services (Mart Services). Programmatic access for bioinformaticians is easily achievable through both these methods, in fact, the BioMart website will auto-generate the relevant Perl scripts or XML queries.

BioMart also offers the advantage of being integrated into some external software packages, which offer their own data analysis and database interoperability solutions. The BiomaRt package forms part of the popular BioConductor [26] suite of R-coded tools. This makes it possible to feed the results of a BioMart query into the powerful statistical and analytical tools that R and BioConductor provide. Taverna, from the myGrid project [15, 16, 27], is described more fully below but BioMart queries can form part of the workflows generated by this application. Galaxy [28] uses BioMart as part of its framework to integrate genomic sequences, alignments and functional annotation and again allows the results to be fed into computational tools, such as those from EMBOSS [29]. The Generic Model Organism Database (GMOD) project is a suite of software for creating and maintaining a model organism database that includes scripts that convert the data available in the GBrowse genome visualization tool into a BioMart schema. Finally, any BioMart server can be easily configured to act as a distributed annotation system (DAS) server so that any DAS client, such as the Ensembl genome browser can display the data stored in the BioMart.

**MOLGENIS customizable database software**

MOLGENIS is an open-source software package for rapidly building database software, including a data storage back-end and graphical and programmatic front-ends to easily interact with the data [30].
Originally MOLGENIS was used to design and generate database software for new research projects [14], but now it is increasingly used to generate software to access existing databases. For this case study, part of the existing Mouse Phenome Database (MPD) [31, 32] was made accessible through programmatic interfaces by using MOLGENIS.

Figure 3 shows the generated MOLGENIS graphical user interface (GUI) that biologists can use to store, navigate and find biological data and the links to four APIs that programmers can use to batch import data and/or integrate with their other processing tools in Java, R, simple HTTP and most importantly, web services. The highlighted screenshot shows selection of these MPD web services from within the Taverna workbench.

The Java API was used to move the downloaded text files into the database. This step can be skipped when generating a MOLGENIS instance for an existing MySQL database. Additionally, the web services interface was used from within Taverna to retrieve data from the generated database.

Next to the obvious benefits of providing graphical and programmatic interfaces, the use of generative techniques has further advantages [30]. A concise description in DSL can help to bridge between different resource providers. A few changes in a single DSL file are much easier to manage than many related changes that could be spread throughout software code. Moreover, when a generator is improved with a new feature then each of the generated variants also gets this new feature ‘for free’ when the software is regenerated (such as recently in MOLGENIS with the introduction of the web services API).

### Taverna workbench

Taverna [15, 16] is an environment for the design and execution of workflows. It is open source and
has been developed as part of myGrid [27], a toolkit for the support of in silico science. In addition to the workflow environment, the myGrid toolkit provides services to support workflow design and reuse, such as semantic service discovery [33] and a workflow repository [34], as well as services to support metadata management and tracking of experiment provenance [35].

The Taverna Workbench enables interconnection and interoperation between local and remote analysis tools and databases of varying types. Taverna can combine web services, BioMart queries, R statistical analyses [36] and BioMoby services [37], to name a few. In the bioinformatics domain, there are over 3500 different services available to Taverna, providing a flexible and extensible platform for bioinformatics research.

Connecting to distributed data sources eliminates the necessity for downloading and maintaining local copies of data. If bioinformatics resources are updated frequently (i.e. every day or every week) this reduces the overhead of keeping local data integrations current. It also allows integration over new data dynamically as there is no lag between the publication of the new data and its accessibility in Taverna.

Combining distributed (and heterogeneous) services is a complex procedure. Taverna can help to manage the heterogeneity of data but it cannot eliminate it. The workflow model is a record of the integration protocol. Exactly how data sources have been linked can be traced through the workflow and any formatting of the data is transparent through the use of shim services [38], which manipulate the format of data without changing the scientific content. The shims can be the existing Beanshell or local Java services provided by Taverna or users can write their own. Consequently, the workflow is a representation of the experimental method that can be reviewed and verified by other scientists as well as easily reused or repurposed.

The generation of workflows, in Taverna or in other workflow systems is still not a trivial process. In order to combine web services into workflows, the user has to understand which services they want to use and what the inputs and outputs represent. Semantically described web services, such as those offered by BioMoby [39], offer a solution to these problems. The SeaHawk BioMoby client [40] reduces these challenges by allowing ‘programming by example’ where biologists can browse and

Figure 4: Partial MOLGENIS configuration for the MPD. This DSL is used to concisely describe what particular data types are involved and how these should be shown on screen in the final software produced by the MOLGENIS generator.
perform analyses as they normally would by clicking, cutting and pasting and then the whole process is saved as a Taverna workflow. One drawback of this approach, however, is that the user is then restricted to using only BioMoby services. Other work to semantically describe any kind of web service is underway in the BioCatalogue project, as discussed later on.

Another potential problem with Taverna, along with any client using web services, is dealing with large data volumes in a workflow which can cause the client to hang. Currently, there is a large data plugin available in Taverna to enable more high-throughput data processing and Taverna 2, the new generation of the workbench, further addresses the issues of scalability by providing facilities to pass data by reference instead of value and providing data streaming, amongst other things.

myExperiment [34], a recent development from myGrid addresses the issue of workflow reuse. myExperiment is a social networking portal for sharing workflow experiments and running workflows over the internet. myExperiment has two types of users:

- Bioinformaticians—designers and builders of workflows who want to share their expertise, experience and workflow files
- Biologists—people who do not wish to design their own workflows, but who wish to run existing workflows of complex analyses over their data.

Workflows in myExperiment can be public, or shared by a restricted number of people in a group. In this way, myExperiment can be used as an internal community resource or as a resource for publicizing the outputs from a particular project.

CASE STUDY

In the Introduction section, we mentioned a typical use-case from the mouse biology community of finding evidence for linkage between phenotypes and genotypes. As an example, a researcher may have knocked out the same gene X in two different background strains, C57BL/6N and 129/Sv and found that each strain gave different phenotypes. Wanting to investigate the underlying mechanisms behind this background-dependent response, s/he then performed an Affymetrix microarray experiment to identify genes up- and down-regulated in the knockout mice of both strains. Working on the assumption that knocking out gene X may have affected a whole metabolic pathway, s/he is interested in establishing whether any of these genes with altered expression are part of a common pathway and if different pathways are involved in the two different background strains.

S/he may also want to know the genomic locations of these genes to see if any co-regulated genes are clustered. For the candidates that appear interesting from these two analyses, s/he may want to look at the SNPs located within these genes and in particular whether the strain-specific changes in gene expression profile, in response to the same mutation, are associated with different surrounding haplotype blocks in the two different strains, a possibility that might explain the different responses.

To perform the analysis required, the researcher would need to collect variation, genomic context and pathway data. Having worked in mouse genetics for some years s/he would probably have a good knowledge of the resources containing mouse data and may have chosen to retrieve the variation data from MPD, the genomic context data from Ensembl [41, 42] and pathway data from KEGG [43, 44]. For a small number of data items, the data could be retrieved by simply accessing the web pages and copying and pasting the results into a spreadsheet, but this solution is not scalable. The initial microarray experiment is likely to have identified hundreds of potential candidate genes involved in producing the phenotype of the mouse. In order to get the most value from the available data resources, they need to be combined so that they are accessible not only to skilled bioinformaticians, but also to lab biologists who rely on this data to inform future experiments. We can envision a situation where bioinformaticians can: (i) easily automate the gathering and integration of this data and (ii) provide their biologists with a tailored experiment compendium, i.e. a customized piece of software that allows the biologists to explore the relevant experimental and public data. Here, we present a solution that enables this kind of approach, for the example, described above. The strategy we have used is summarized in Figure 5 and the workflow and some sample results shown in Figure 6.

The workflow is written using the Taverna Workbench, and incorporates the functionalities provided by web services (KEGG), BioMart queries and visualization (ENSEMBL), and MOLGENIS ‘wrapped’ databases (MPD), most of which were
already available to Taverna. KEGG, for example, provides access to its pathway resources via web services while Ensembl provides access through BioMart. MOLGENIS was needed to auto-generate web services for MPD because these resources currently only provide access to their data via GUI and as file downloads. Taverna can combine any resources that are available as web services as well as any public BioMart service.

The workflow uses BioMart at Ensembl to first recover some Ensembl Gene IDs and their corresponding EMBL IDs for a list of known MGI symbols. Then, for each Ensembl gene, KEGG is used to recover pathway IDs and HTML links to marked-up pathways, and MPD to retrieve SNPs per gene and the allelic variation per mouse strain (Figure 6).

The workflow itself is a model of the experimental method. The figure shows example results for Scm1 and Camk2b, but this workflow is available for any researcher to download and run with their own set of genes. Others can reuse it without change to run the same analysis again on different data, or they can modify it for their own purposes. The work of combining these data sources, however, only occurred once. Sharing the experimental method with others is simply a case of sharing the workflow file. This workflow can be viewed and downloaded from myExperiment (http://www.myexperiment.org/workflows/126) and can be run from within Taverna with the File->'Open workflow location' option using http://www.myexperiment.org/workflows/126/download as the URL.

A biologist with no programming experience will be able to run their analysis using this approach, but any modification of the existing workflow will probably require some informatics support. Future development of Taverna-embedded web resources would be desirable so that such a researcher could perform the analysis using a more familiar web browser environment, e.g. like the aforementioned SeaHawk tool. It should be stressed that the solution presented here is a solution to be implemented by bioinformaticians for the use of biologists and not by the biologists themselves. Considerable informatics expertise is needed to deploy the required services if not already available and to design the workflow experiments.

**Figure 5:** Three scenarios were explored to adapt autonomous, heterogeneous data to have a workflow ready web service interface. (A) Generate an adapter that directly connects to data in source system that translates its local interface to a standard interface. Here cooperation of the data provider is needed which is often not possible. (B) Copy data to a mirror server whilst keeping its original structure and generate the adapter on the data copy. This scenario is functionally equivalent to (A) but now no cooperation of provider is needed, instead regular updating of data is required when the data source changes. (C) Transform the data into a standardized, query optimized, data 'mart'. This can either be done by the data provider, as for the example here, or by a third party that copies data to a mirror server and transforms it to a mart. Regular updating of both data and possibly the transformation configuration is required when the data source changes.
DISCUSSION

Centralized versus distributed querying

It has generally been thought that the number and diversity of relevant databases precludes any sensible strategy for mouse functional genomics based on a centralized data warehouse, as it is unlikely that any single team of mouse biologists or bioinformaticians would have the data access rights, skills and resources to create such a warehouse for all the relevant data. Whether or not this view is correct is probably not important as it is becoming clear that a far better approach to data access and analysis is to use a virtual warehouse approach where databases are accessed remotely through APIs. This article has shown that such a distributed strategy can be achieved using methods of integration based on data structure transformation and ‘smart clients’. We have demonstrated in our use-case that using a combination of technologies, such as web services, BioMart, MOLGENIS and Taverna provide an efficient and powerful way to solve current database interoperability problems for the mouse community. The advantage of a loosely coupled approach like this is that many more resources and technologies can also be combined if required, for example DAS [45], or BioMoby [39]. For a defined task, an alternative distributed approach is to access each resource directly through its web interface as demonstrated by the GeneSeeker system [46]. For each database a plug-in has to be written to reformat the query into the correct URL and parse the results. This has the advantage that any web resource can potentially be added. However, addition of new resources and maintenance as the original website changes will obviously involve a lot more local programming effort than connecting your client to a new generic web service or BioMart.

Figure 6: Taverna workflow for the case study. The left hand panel is a graphical representation of the workflow. Initially this looks complicated but the only components involved in data retrieval are the two BioMart queries at the top and the five Web Service requests. All other components are data processing steps to produce the final merged results. The right-hand side shows these results in a clockwise direction from top-left: (i) KEGG pathway IDs mapped to each gene, (ii) SNP data per gene including internal MPD SNP ID, dbSNP ID, observed variation and SNP location in the genome and the gene, (iii) variation observed in mouse strains for each of the MPD SNP IDs and (iv) an example KEGG web page from one of the links recovered by the web service.
There are however two types of barrier to this sort of approach becoming widespread. First, and in spite of the maturity of the technologies required, the integration modes described above require levels of bioinformatics skills beyond those of most bench scientists. In essence, informatics skills are currently required to build and modify the workflows but not to run them. While the sharing of workflows, for example through ‘social networking’ sites, such as myExperiment, will of course assist with dissemination of ‘off the shelf’ tools for frequently used protocols, a key barrier to effective database interoperability is the funding and provision of the necessary local informatics expertise or training of experimental scientists. This resourcing and infrastructure issue needs to be recognized by funding agencies.

A second cause of concern is the dissemination and uptake of standards for data structure, data and service descriptions (semantics) and programmatic access. Although sophisticated tools like MOLGENIS can deal with heterogeneity of syntax, semantics and structure (and are therefore, to a large degree ‘future proofed’), conformity with agreed standards, together with publication by databases of technical details required for their programmatic access will facilitate the future development of tools based on web services and ‘smart clients’. One aim of the CASIMIR consortium is to promote standardization and agreement on standard data formats and semantics within the mouse biology community. We might expect that the momentum given to software development by standardization will produce a generation of much more user-friendly, easy-to-integrate and flexible software.

The BioMoby project addresses this issue through facilitating the creation of semantic web services with formats and conventions defined through MOBY. The advantage of the MOBY world is that searching for appropriate services and chaining procedures requires minimal user intervention. For example, mapping of the output of one service onto the input of the next can be automated. However, the price of automation is absolute adherence to MOBY service, namespace and object ontologies [47], while web services have to be built to the BioMoby specifications. For mouse resources, there are already collections of web services and other types of service that do not meet these standards so that a BioMoby approach for all community resources might not be practical. There are however other initiatives, such as the Feta Semantic Discovery system [48] from myGrid, that focus on providing descriptions for existing resources of multiple types, and a combination of approaches may be helpful.

The integration approach advocated in this article depends on adherence to domain-specific semantic and syntactic standards by service and data providers (predominantly a sociological challenge), but can provide service and data consumers with the facility to build the very precise tools needed to address sophisticated bioinformatics queries. It does however have to be said that the approach places the burden of work on the user and their informatics support rather than on a central resource. While this inevitably requires users to have at least some bioinformatics input into designing and implementing their tools, the benefits in the potential sustainability, stability and flexibility of the distributed system is we believe much greater than that can ever be provided by any centralized resource whose tools can never be user-specific.

As things stand, the proposed solution for our case study remains the only practical one for high-throughput, generic, distributed analysis. Each of the required resources uses different technologies for programmatic access and, short of writing your analysis application or scripts, Taverna is the only way of combining them into a single experimental workflow. If in the future all the resources became collected in a single data-warehouse or all had the same service interface deployed then a solution involving a single website or client may be possible. However, this is unlikely to happen for sociological and financial reasons.

The immediate future

There is a clear need to improve the interoperability of current mouse data resources. A feasible step forward for the mouse biology community would be to develop a Web Service Registry providing information on mouse resources and, most importantly, web services details (WSDL links and semantic descriptions). Similar initiatives are already underway: myGrid has, for example, been developing a registry of semantic service descriptions, which will be extended to a community resource in the new BioCatalogue project with the European Bioinformatics Institute. BioMoby also has a large number of semantically described services. CASIMIR is currently developing such a registry specifically for the mouse and it is becoming clear that a respectable
number of resources have already deployed web services or are in the process of doing so.

The CASIMIR consortium will continue to promote these approaches to database interoperability and help the individual mouse centres to implement the required technologies. Such an approach will put the mouse biology community in a good position to utilize all the data starting to emerge from projects, such as the high-throughput gene-knockout and phenotyping pipelines. We hope this review of the technologies and the case study will convince the mouse community of the benefit of database interoperability and demonstrate how to implement it on their own data sources.

**Key Points**
- Integrated querying of mouse resources (and biological databases in general) is hard to automate.
- A distributed approach using technologies, such as web services, BioMart and MOLGENIS can overcome such interoperability problems.
- Workflow management systems, such as Taverna, enable the chaining together of queries over distributed databases and data analysis tools.
- Many mouse resources do not provide programmatic access to their data. To improve integration and interoperability for large-scale queries, this is a crucial step and requires further deployment of web services, BioMart, MOLGENIS wrappers or similar technologies for all the most used mouse centric resources.

**FUNDING**

Framework Programme 6 of the European Commission (LSHG-CT-2006-037811 to CASIMIR).

**References**


