6

Middleware for Distributed Data Management

Alvaro A. A. Fernandes

Abstract

The widespread availability of data resources, and of the infrastructure with which to access them, creates many research opportunities. However, such access is seldom unimpeded. There are impediments stemming from conflicts in the infrastructure itself, in the representations used, and in interpretation. Among the many possible responses to these conflicts, one that is particularly promising from an end-user viewpoint relies on middleware for distributed data management. This chapter characterises the kinds of problem that can inhibit data sharing and analysis, describes how these problems are being addressed and how far the state of the art is from definitive solutions. The goal of the chapter is to introduce and motivate the notion of Web- and Grid-based middleware solutions for reconciling conflicts and removing impediments to effective and efficient access to and use of autonomous, heterogeneous, distributed data resources.

INTRODUCTION

The context that serves as background for the issues discussed in this chapter is that of the emergence of the distributed computing platforms known as the Web and the Grid. While the Web is well known, the notion of a Grid is less well understood, in large measure because it is far more technical. According to Foster (2002), 'a Grid is a system that: *(1) coordinates resources that are not subject to centralized control, (2) using standard, open, general-purpose protocols and interfaces (3) to deliver non-trivial qualities of service*.' These platforms are notable for many reasons, among which is their unprecedented reach, in the case of the former, and the promise of gains of scale in computational capability, in the case of the latter. These developments have resulted in a drive towards making data of all kinds available to users of such platforms. While the research potential latent in such technological developments and in the explosion in data availability it has fostered is uncontested, it has become clearer over time that effort is needed for datasets exposed through these platforms to be truly useful.

One lesson learned is that data is seldom truly useful unless it can be integrated with other data. The issues arising from the need to integrate heterogeneous datasets that have

[17:55 27/2/2008 5123-Fielding-Ch06.tex] Paper: a4 Job No: 5123 Fielding: Online Research Methods (Handbook) Page: 99 97–115

been autonomously generated and exposed over distributed computing platforms are central to ongoing attempts to derive from such platforms the infrastructure for *in silico* science.[1]

If the goal is sharing, one precondition is integration. If the goal is integration, one precondition is Grid-, or Web-, enabling. By *enabling* is meant not simply making a dataset visible and accessible to end users, but exposing it in such a way as to make it visible and accessible to software tools. Once a dataset is machine-readable using standard protocols and interfaces, the problem of integrating datasets becomes a tractable one. Once heterogeneous, autonomous distributed datasets can be integrated through principled, tool-driven processes, the potential for significant advances in evidence-based research is significantly increased.

This chapter considers the problems that inhibit sharing and analysis, provides a high-level explanation of the emerging solutions and discusses how far they still are from the vision projected by platforms such as the Web and the Grid. Note that the aim of this chapter is neither to review nor to survey the research underpinning the processes and tools alluded to. The aim is to identify, describe and highlight the challenges involved in integrating heterogeneous, autonomous distributed datasets using a class of software tools known as distributed data management middleware. For this reason, there are few references to the scholarly literature. Those that are provided can be used as entry points for deeper exploration.

Because the focus of this chapter is on generic software products that can, in principle, be used to access a great variety of existing datasets if the rather technical conditions described below are met, no dataset and no data-provision organisation is discussed in detail. Cole et al.'s chapter (this volume) describes in detail exemplar datasets from a variety of data sources. Likewise, no concrete example of the direct benefits for a social scientist is given in this chapter. The chapter by Crouchley and Allen (this volume) does so.

The chapter is structured into the following sections. The first section addresses the issue of data integration, explores some of the challenges involved and explains why overcoming them is so important. The second section considers in more detail what is meant by *enabling*, in phrases such as *Grid-enabling* when applied to datasets, what is involved in the process and what benefits accrue from it. The third section looks into the steps needed for effectively sharing data and maximising the benefits thereof. The final section looks forward to recent developments in Web-based computing that may positively and significantly impact the ease with which distributed data can be used to underpin novel studies, new tools and greatly empowering end-user applications.

DATA INTEGRATION

This section attempts to provide high-level answers for the following questions. What is meant by *data integration*? Why does the need for data integration arise? Why is data integration so important? Why is data integration hard from a technological viewpoint? How does one go about integrating data?

What is data integration?

Data integration (Foster and Grossman, 2003) is the process which enables the linking of different datasets together, thereby enabling tools (and not just end users) to interact with them as if they were a single, unified and homogenous resource.

Consider Figure 6.1. It depicts a scenario where different researchers, in different locations, autonomously (i.e., without cooperation and coordination) have generated datasets on three different but related subjects, while another researcher, in yet another location, is interested in making use of the three datasets, ideally in an unimpeded but still coherent manner.

The central, general question addressed in this chapter is the following. How can one

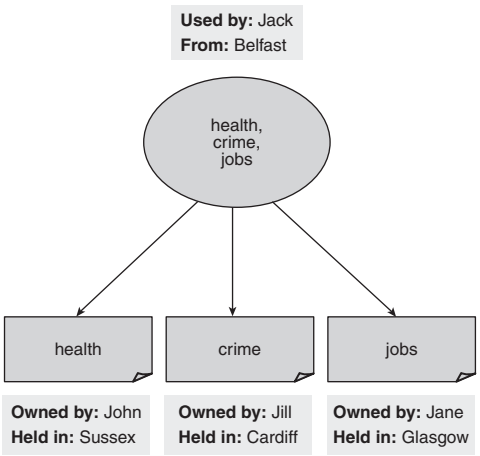

Figure 6.1 Heterogeneous, autonomous, distributed datasets

build tools that enable an integrated view of heterogeneous, autonomous, distributed datasets? One instance of this general question is, in terms of Figure 6.1, how can Jack, in Belfast, make use of the health, crime and jobs datasets held by John, Jill and Jane, in Sussex, Cardiff and Glasgow, respectively?

The goal of data integration is to make it possible for users to access distributed, heterogeneous and autonomously administered datasets more easily. One of the main goals of middleware for distributed information management (Antonioletti et al., 2005) is to facilitate the data-integration process.

Why does the need for data integration arise?

In most areas of commercial and scientific activity, more often than not, datasets are acquired and stored under an (often implicit) assumption that they will either be used individually, or only by the people (or organisations) that have collected them or that now maintain them.

However, if users need access to datasets that have different formats, or have been acquired by others, or that are stored under different administrative controls, then they may face a range of impediments that act like barriers to integration and sharing.

Overcoming these barriers calls for a number of responses. Some example responses that are needed include the following. Datasets need to be described in sufficient detail for other users to be able to understand their meaning. An agreement must be reached on standards for description (i.e., for the provision of metadata) and for the adoption of compatible computing infrastructures. Organisations that hold datasets need to agree on common policies for sharing and access.

Why is data integration so important?

In many sciences, a significant component of cutting-edge research is, or is becoming, evidence-based. In response to this, in the UK, the Economic and Social Research Council (ESRC) – with support from the Joint Information Systems Committee (JISC) – has invested substantially in establishing a distributed social science data infrastructure, providing researchers with access to a range of key datasets spanning many disciplines and research themes. In addition, many key social science data resources, such as Neighbourhood Statistics, are becoming available from outside the UK academic community.

Although there is unevenness, other national social science communities in Europe have also made initiatives in this field. The lead on the exploitation of new data-sharing technologies undoubtedly lies with the natural sciences, where grid-enabling has been motivated by the requirements of very large international scientific collaborations such as are found in particle physics, astrophysics and genome mapping. Nevertheless there is now roughly a decade's experience with the social science application of grid technologies. As well as European initiatives, the US National Science Foundation has initiated an infrastructure capacity-building priority to facilitate 'cyber-research.' The various institutional initiatives share an orientation to encouraging collaboration between dispersed research teams – the 'collaboratory' idea – in order to maximise the exploitation of expensively collected and curated datasets.

While this raises issues about the effect of technology on the working practices of social scientists, there have been early gains from the kind of interoperability of datasets profiled in the Cole et al., chapter (this volume).

Access to datasets such as these plays an increasingly important role in providing the evidence base for research, but researchers (as well as tool developers) face a number of impediments to the realisation of the full potential of these datasets. Although these impediments are quite wide-ranging in kind (e.g. administrative, legal, cultural, social or economic, amongst others), a number of these are specifically technical in nature and the remainder of this chapter concentrates on them.

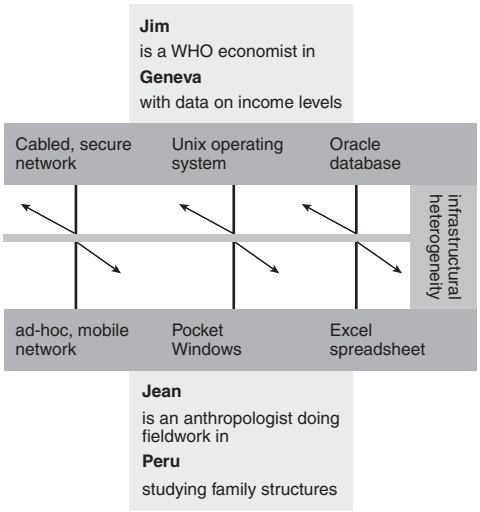

Figure 6.2 Infrastructural heterogeneity

Why is data integration technically difficult?

The main reason why data integration is technically difficult is that it has to contend with the problems and conflicts arising from the heterogeneity, distribution and autonomy that is typical of existing dataset provision in some sciences. Thus, while in physics, chemistry (Kim, 2006) and biology (Furukawa, 2004), for example, most dataset provision is highly automated in all stages of the production cycle, this is less the case in the economic and social sciences and the humanities.

For example, there are many different formats in which data can be encoded and different types of database software. Also, data may be held on many different computers which belong to different organisations (and hence, many administrative domains, each with their own mechanisms for user authorisation and authentication). In this chapter, three kinds of heterogeneity are discussed, as follows.[2]

The most basic kind of heterogeneity is *infrastructural heterogeneity*. It is so called because it is not application-specific. It stems from differences in computing technologies and software environments (e.g. different networks, operating systems, storage devices and database software, data access protocols, etc.). Infrastructural heterogeneities prevent sharing (and hence integration) even if the parties use the same applications and agree on their interpretation of the data.

One example is the kind of problem that results from someone's preferences for a Unix-based environment or a Windows-based one. Such a preference can have consequences such as a Web browser that runs in the latter but not in the former, etc.

Consider Figure 6.2. The results generated by Jean may not be accessible to Jim, or to the tools he uses, because the infrastructures used (Pocket Windows and Unix, respectively) stand in the way.

Infrastructural heterogeneity is best solved by so-called *middleware (software)* in a mediator role, viz., one in which the software component has the purpose of reconciling differences and allowing unimpeded interaction (Wiederhold, 1992). In terms of Figure 6.2, behind the scenes, the mediator software would take responsibility for ensuring that the different infrastructures do not make Jean's data inaccessible to Jim, or vice versa. The challenge here is to remove barriers to interoperation between the systems preferred by a given user and the systems used to make the data resource available.

Another, technically more challenging kind of heterogeneity is referred to as *syntactic*

heterogeneity, as it stems from the choice of different languages to describe data, or to query, analyse and update them. It is so called because the same application type (e.g. databases) may nevertheless use different languages, commands, conventions, etc. Syntactic heterogeneities prevent sharing (and hence integration) even if there are no infrastructural barriers.

One example of how it arises would be when a user wishes to load data from a commercial database system into a statistical package.

Consider Figure 6.3. Even though both Joan and Javier use relational database systems, the results generated by Joan may not be accessible to Javier, or to the tools he uses, because the languages used (different dialects of SQL, in this case) stand in the way.

Again, syntactic heterogeneity is best solved by middleware. Behind the scenes, the middleware (Antonioletti et al., 2006) takes responsibility for translating requests in the SQL dialect used by Joan into ones in the SQL dialect that the database used by Javier understands. The challenge is to remove barriers to the interchange of data between two systems that can, otherwise, interoperate in the computing environments in which they are deployed.

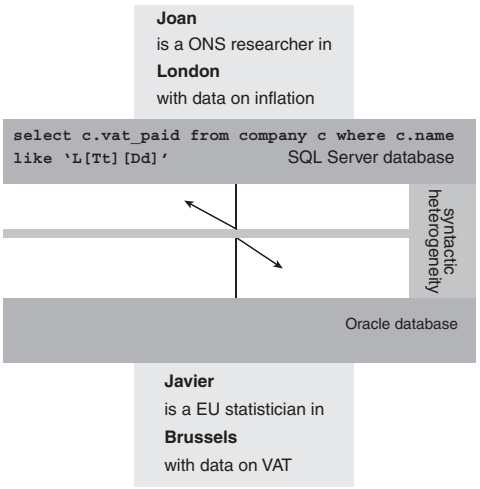

Figure 6.3 Syntactic heterogeneity

The third, particularly important and by far the most technically challenging, kind of heterogeneity is referred to as *semantic* (or *schematic*) *heterogeneity* (Kim et al., 1995). It stems from the fact that data are acquired and collected by different research communities, for different purposes and under different theoretical assumptions. It is so called because it arises from differences in interpretation between the owners of the data.

There are many forms of semantic heterogeneity conflict, e.g. different names for the same data type, same names for different data types, etc. While these have been categorised and while solutions have been devised for most of them, the complete resolution of conflicts arising from semantic heterogeneity is, undoubtedly, the central issue in data integration. It is particularly intractable because it requires the involvement, and then cooperation, within and across research communities. The impediments are non-trivial, leading to protracted, costly efforts distributed over organisational boundaries and different locations. However, the added value, and hence the benefits ensuing, are expected to be significant in the case of biomedical informatics and other scientific fields (see, e.g. Cimino and Zhu, 2006) and many are now exploring potential returns for social science.

Consider Figure 6.4. Juan stores values for the retail price index (RPI) in the field named INFLATION, whereas Joachim stores the consumer price index (CPI) in the field

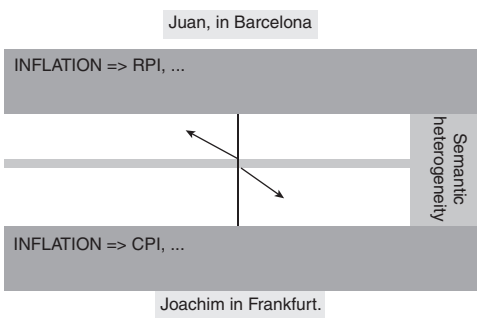

Figure 6.4 Semantic heterogeneity

with that name. This exemplifies the kind of semantic heterogeneity referred to as *attribute-name heterogeneity*, i.e., the same name is used for different concepts.

To resolve the conflict and open the way for integration and sharing, Juan, for example, would have to define what is formally known as a *mapping* into Joachim's data, viz., a rule-like assertion that states, essentially, '*What I call INFLATION, Joachim calls RPI, and what I call CPI, he calls INFLATION*.' These kinds of mapping can then be used by the middleware to translate properly Juan's requests for Joachim's data, and vice versa. Clearly, if each user were to do this, the number of mappings would proliferate.

For such reasons, it is better to have mappings that are general enough (e.g. it may be that the difference is not really between Juan and Joachim, but rather between the governmental authorities in Spain and Germany, i.e., it is a difference between financial authorities regarding which index to take as a measure of inflation). The challenge is, therefore to get different research communities to agree on mappings between their interpretations of data.

Figure 6.5 illustrates, with respect to Figure 6.1, the case in which Jack is impeded by different barriers in his wish to use data owned by John, Jill and Jane.

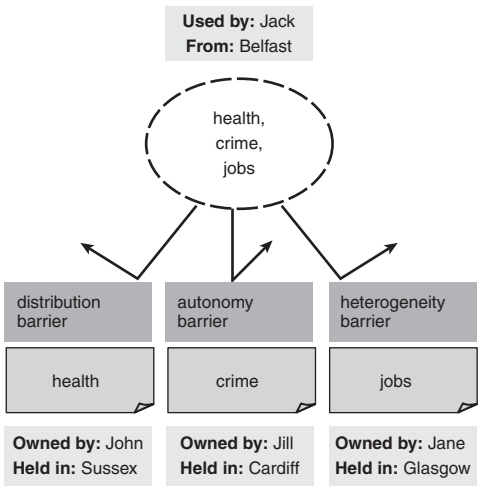

Figure 6.5 Barriers to integration

Having looked at challenges arising from heterogeneity, consideration is now given to the challenges arising from distribution and autonomy barriers. By 'barrier' in this context is meant some occurring circumstance that impedes interaction and that can be identified as arising from the fact that resources are remote (in the case of distribution) or are not under the user's administrative control (in the case of autonomy), or are heterogeneous with respect to the means the user has to gain access to and use the resource, or any combination thereof.

A data resource is said to be *distributed* if it is a composite of many distinct parts residing in different locations. Different locations often imply different owners, who are autonomous to grant and qualify access rights. The right to access does not imply the capability to access: heterogeneity barriers may prevent access nonetheless.

Data integration is about removing such barriers whenever accessing data (e.g. John's, Jill's and Jane's) requires the existence of some mechanism to project (for Jack's benefit) an aggregated, global, single view of a data resource, even as it is made of many (geographically scattered, and hence referred to as local) parts.

Distributed data resources are more likely than not to give rise to infrastructural and syntactic heterogeneity conflicts, and often comprise autonomous component parts. In contrast, autonomous data resources are very likely to give rise to semantic heterogeneity conflicts.

In the present context, a data resource is said to be *autonomous*, from the point of view of a potential user, if someone other than that user has independent control over the resource and, in particular, if as a result the data resource can change in form or content, or can migrate to different environments etc. with little or no consultation, agreement or previous notice.

The challenges arising from distribution and autonomy centre on the need to make it easier (e.g. less costly) for potential users to use a distributed, autonomous data resource. For example, with respect to distribution,

a user would wish to be protected from the inherent instability and unpredictability of communication networks. With respect to autonomy, a user would wish to be protected from the risk that, because a data resource has been changed by its owner, one's use of it will be greatly disrupted, e.g. will incur significant associated adaptation costs.

How does data integration work?

Data integration technology usually takes the form of middleware, i.e., a set of software components that are deployed together with existing software systems in order to provide a set of generic services between those systems. In particular, data-integration middleware is deployed along with a collection of distributed, heterogeneous, autonomous datasets to provide users and systems with a view of those resources that removes many (though perhaps not always all) impediments to their use. From the viewpoint of users and systems, the result is the functional equivalent of a single, unified and homogeneous dataset that is insulated from the more disruptive side-effects of autonomy.

Figure 6.6 illustrates the typical software architecture for data integration founded on middleware components. Middleware for distributed data access and integration is deployed between an application and the data it needs (inflation and VAT in the case of Figure 6.6). It uses two main kinds of component: a *mediator* and several *wrappers*.

There is typically one mediator, although there may be more than one in order to separate concerns, apportion responsibility and hence provide a better solution to what is in practice a multifaceted problem. It relies on wrappers, of which there are typically many, one per source in the limit case, for two main tasks:

1 *Resolving infrastructural heterogeneity conflicts* (e.g. to convert between communication protocols, to negotiate access and authorisation to connect to autonomous datasets, etc.).
2 *Resolving syntactic heterogeneity conflicts* (e.g. to relieve the mediator from the need to convert to and from the different languages supported by different datasets to store, access and update data).

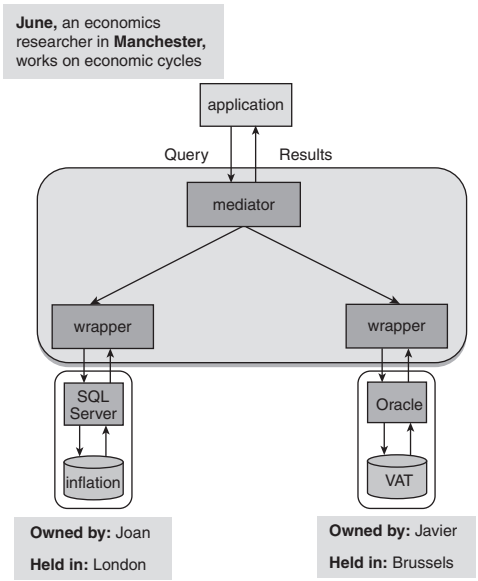

Figure 6.6 Software architecture for mediator-based data integration

This use of wrappers frees mediators to focus on two main tasks:

1 *Resolving semantic heterogeneity conflicts* by enforcing user-defined, resource-specific conversion rules that map between the conceptualisation and interpretation used by whoever owns the dataset to the conceptualisation and interpretation to be applied to the integrated dataset.
2 *Allowing users to interact with the resources* as if they constituted a single, unified dataset. This typically involves invoking a great deal of complex functionality (e.g. managing network connections, translating between different conceptions and interpretations, etc.) as transparently as possible from the viewpoint of the user.

Wrappers are components with which datasets can be Grid- or Web-enabled, i.e., by resolving certain kinds of heterogeneity they allow the development of tools (of which mediators are one example) that open the way for integration and other value-adding processes by end users.

ENABLING DATASETS FOR INTEGRATION

This section considers in more detail what is meant by *enabling*, in phrases such as *Grid-enabling*, when applied to datasets, what is involved in the process and what benefits accrue from it. It attempts to provide high-level answers to the following questions. What does Grid-enabling[3] a data resource mean? Why is Grid-enabling a data resource an important step? What is involved in Grid-enabling a data resource? How can Grid-enabled data resources be used?

The Grid is a kind of distributed computing infrastructure which is being developed by a global consortium (http://www.ogf.org/) of public and commercial sector organisations as a complement to the Web.

In contrast to the latter, it is being built with a view to supporting computationally demanding applications, among many other differences one could cite. Grid-enabling a dataset means to adapt that dataset so as to make it accessible, programmatically, over the Grid.

It is important to stress the emphasis on making datasets accessible by programs, rather than, as is commonly the case with the Web, by people only. One goal of programmes such the UK e-Science and US Cyber-infrastructure initiatives is to create the conditions for many scientific investigations to be cast as a computational process, when it is deemed beneficial or necessary to do so (as is the case, for example, in disciplines such as physics and biology, among many others).

Thus, Grid-enabling a dataset creates new opportunities for its use. In particular, and most importantly, it creates opportunities for tools, existing or yet to be conceived, to gain access and make the most of the diverse data resources required by wide-ranging, evidence-based scientific investigations.

For example, Grid-enabling a data set is an initial, necessary step towards enabling users to integrate that dataset with others. It also makes it possible to analyse the dataset using techniques that require the kind of computational power that it is only feasible to access using the Grid, as well as to standardise the procedures and mechanisms used to access and update the dataset, thereby increasing the likelihood that others will be able to share it. Even though this list is not exhaustive, all of the above clearly represent significant steps towards making the most of, and adding previously unforeseen value to, the many, scattered, isolated datasets that a researcher might wish to use if only the barriers to using them effectively and efficiently were significantly lowered.

In practical terms, Grid-enabling a data resource is often a technical task. It involves (in terms of software architecture) placing it behind wrapper middleware for some Grid fabric.[4] Once a data resource has been wrapped in this way, it is Grid-enabled, and the resulting benefit is that a great many impediments to its use or its sharing are thereby removed.

Consider Figure 6.7. It depicts an instantiation of Figure 6.6 with specific distributed data-management middleware, viz., OGSA-DAI (Antonioletti et al., 2006) and OGSA-DQP (Alpdemir et al., 2003).

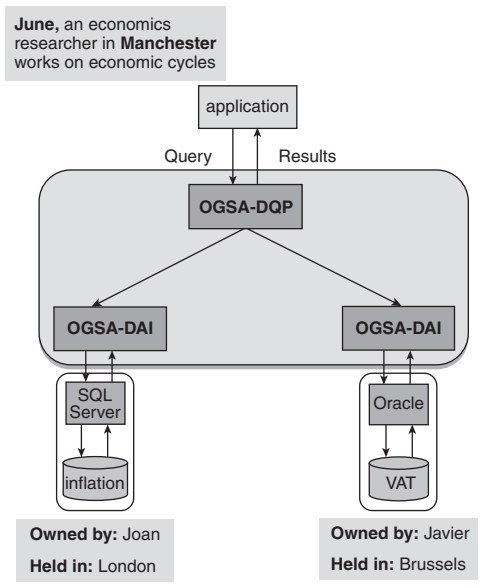

Figure 6.7 OGSA-DQP and OGSA-DAI as instances of grid data-management middleware

OGSA-DAI is a distributed data-management middleware product for wrapping (and hence Grid-enabling) data resources. As such, it removes impediments to access arising from problems of heterogeneity and distribution of relational and XML data resources. OGSA-DQP is a distributed data-management middleware product for mediation over OGSA-DAI-wrapped data resources. It allows for the querying of distributed data resources as if they formed a single virtual database.

The notion of 'wrapping,' albeit a technical one, captures a simple intuition, viz., that if two complex artefacts are dissimilar in inessential ways, then it may be possible to hide the inessential elements and expose only those elements on which they concur, i.e., which makes them seem more similar than, in their full complexity, they are known to be. A wrapper is a software product that plays this role in data-integration scenarios: by hiding the differences and letting the similarities through, it enables software artefacts to be viewed as instances of a common type, rather than as completely independent products. This role is crucial to enable mediation to take place.

Mediation is a software-based process by means of which it is possible to superimpose a common, global view on resources that retain their local independence. Albeit a relatively simple one, translation is a form of mediation, in that it allows concepts in one language to be cast in terms of another.

Note that, like any middleware product, both OGSA-DAI and OGSA-DQP are tools for tool builders, rather than for end users, although they can be used in this fashion too. OGSA-DAI comes with components that enable querying, transforming and delivering data in different ways. Thus, rather than focusing on offering user interfaces, it provides particularly useful application-program interfaces, with which to build tools that add value to the data stemming from the resource front-ended by OGSA-DAI.

Indeed, OGSA-DQP is one such tool and it, in turn, can be used by tool builders. This layering of functionality, in which one tool builds upon other tools, is good software-engineering practice, insofar as it provides for more cohesive components that, through disciplined composition, can achieve better cost–benefit ratios over the entire product life cycle than a single, monolithic product.

The purpose of these specific middleware products is to facilitate the building of tools for distributed data access and integration over Grid and Web fabrics that resolve heterogeneity conflicts, benefit from location transparency and contend more effectively with the undesirable consequences of autonomy.

One class of impediment that is removed is related to transparency of location. Once a data resource is Grid-enabled, its availability can be easily advertised in registries where advanced Grid middleware will know to find them and learn of their specific usage conditions for both access and update, as the case may be. As shown in Figure 6.7, the deployment of middleware components allows June's application to submit queries about inflation and VAT as though Joan's and Javier's data were hers and held in Manchester.

Another class of impediment that is removed by Grid-enabling a data resource is related to infrastructural and syntactic heterogeneity conflicts. As an example of removal of an infrastructural conflict, consider the fact that, once wrapped, the data resource becomes visible to systems and applications that make use of middleware such as the Globus Toolkit (Foster, 2005). For most intents and purposes, the data resource can be seen as a service, and most infrastructural barriers are either easier to overcome or cease to exist altogether.

As an example of removal of a syntactic conflict, consider the fact that, for any of the storage and access technologies supported by the wrapper middleware (e.g. relational and semi-structured databases, in the case of OGSA-DAI), any user or application can interact with a wrapped data resource using one single syntactic framework, viz., the one defined by OGSA-DAI. For most intents and purposes, an OGSA-DAI-enabled

data resource is like any other OGSA-DAI-enabled data resource in terms of how a user or an application interacts with it. Other Grid middleware can then take advantage of this fact.

Grid-enabling middleware for data access and integration, such as OGSA-DAI, often comes equipped with one particular class of component, referred to as clients, as already mentioned. A *client* is a component that one can use to interact with another component, referred to as a *server*. For example, a Website is front-ended by a Web server, and a Web browser is the kind of client used to interact with Web servers, in this case, amongst other things, to fetch and render Web pages, to submit information via forms, or to download and upload files.

In order to access an OGSA-DAI-wrapped data resource, one can use the client component that is bundled with the OGSA-DAI software distribution. This client provides means both for humans to interact with the data resource directly (e.g. for browsing it) and for higher-level, domain-specific applications (such as a Web portal, or a specialist data-analysis tool) to be built that interact with the data resource (e.g. for fetching, searching, querying, uploading, downloading, updating data) programmatically, i.e., by program-to-program interactions, using standardised interfaces and protocols.

Another way of accessing Grid-enabled data resources is to use other Grid middleware that act as mediators over OGSA-DAI-wrapped data resources. One example of this kind of mediator middleware is OGSA-DQP, which comes with its own client for users to interact with virtual databases that, by and large, they are free to set up and use 'on the fly.' OGSA-DQP mediates the process by which a user can ask a query over distributed, heterogeneous, autonomous data resources that have been wrapped by OGSA-DAI.

Once a collection of data resources have been Grid-enabled using OGSA-DAI, OGSA-DQP provides the user with the means to access and query them as a composite data resource. To a significant extent,[5] it is as if all the data resources involved were locally held without the associated costs and with not only no loss in quality, but with many potential benefits.

In other words, by using Grid-enabling data middleware like OGSA-DAI and OGSA-DQP, users can benefit from the unimpeded use of distributed, heterogeneous, autonomous data resources, by and large as if they were the user's own locally held ones. Of course, the costs of engaging in this task are far from negligible, since the technology is not, at the time of writing, as transparent or easy to use as might be wished. However, it is progressing towards becoming more transparent and easier to use (as scrutinising the release notes in http://www.ogsadai.org.uk/ will show). Moreover, it is arguable that the cost of Grid-enabling data resources is best construed as an opportunity cost, i.e., as an investment cost, given the value-adding potential of doing so – in which case the question to consider is what benefits will fail to ensue if one chooses not to incur the costs associated with the Grid-enabling of resources.

DATA SHARING

This section attempts to provide high-level answers for the following questions. Why are commenting and annotating especially important in the e-Sciences? How do vocabularies, thesauri and ontologies fit in this context? Is there a relationship between the use of such techniques for annotation and the Semantic Web?

Why are commenting and annotating especially important in the e-sciences?

Grid-enabling a data resource is a fundamental step in making it easier for other users and applications to use it, but is only the first such step (Goble et al., 2003). The opportunities for using a data resource and the benefits from doing so are very significantly increased if, besides being Grid-enabled, the data resource is purposely described as to its content, and

if such content has, e.g., its provenance and authoritativeness, for example, judiciously established.

The great advances that these activities have made possible in enabling the production of new biological knowledge by means of *in silico* experiments that leverage data resources available in areas such as genomics, proteomics and metabolomics (Stevens et al., 2004a, 2004b; Buetow, 2005) amply demonstrate the value of describing content purposely, and judiciously establishing its provenance and authoritativeness, as further steps to Grid-enabling a dataset. Current social science initiatives explore similar gains.

By *purposely describing* a dataset is meant enriching it with information that, besides providing context for human users, enables software tools to use the dataset in more effective and efficient ways, thereby greatly increasing the probability of making the best possible use of the dataset. The paradigmatic example of purposeful descriptions is the use of formal knowledge-representation techniques, as discussed below.

By *judiciously establishing* provenance[6] and authoritativeness of a dataset is meant the principled preservation, propagation and fusing of original judgements as datasets undergo a succession of transformations by Grid and Web processes. Thus, one should be concerned that the provenance and authoritativeness judgments associated with a dataset that lies at the inception of a potentially long chain of manipulation processes is properly propagated to intermediate results and up to the last product in the chain.

This is not just crucial to auditing processes such as apportioning credit, or reconstructing a long, complexly structured chain of reasoning. It is also fundamental as a source of examples of best practice. If provenance trails are available in a form that can be used by software tools, then it is possible to envision that examples of best practice in e-science can be discovered, reused and repurposed in automated fashion.

When it is possible to formally ascertain that a dataset has been properly curated, or that it was derived under conditions that assign it a certain judgement as to its authoritativeness, then its value to users and tools is thereby significantly larger. This is because the implications that can be drawn from that dataset carry much more weight as pieces of evidence.

The difference that curation[7] processes make is analogous to the difference between the definition of the term 'conspicuous' that one might overhear in a pub conversation and that provided by an authoritative dictionary that has annotated the definition of the word with its etymology and its history of usage across time and space.

The difference that provenance records make is to enable, if one so wishes, the satisfaction of one's sceptical stance through scrutiny and reconstruction of the record trail of a derivation chain. For example, one can make up one's own mind as to whether the normalisation, cleaning and weighting methods used to obtain the derived item have, or have not, sacrificed features of interest to the investigative context in hand.

Data curation and provenance are, therefore, crucial not only in fostering use but also in promoting proper use and maximising the benefits that accrue from that use.

As mentioned above, the process of purposely describing content is founded on knowledge-representation techniques (Brachman and Levesque, 2004). There are many such techniques, but they all have in common the need for pairing with the data specific comments and annotations, the collection of which is referred to as metadata, i.e., data about data. Provenance is one important kind of metadata, as are the comments and annotations by experts that result from the process of curation.

In the social sciences, a substantial move towards bringing the Grid vision to practical reality is the work that has been done to agree and adopt common data standards, such as the XML-based Data Documentation Initiative ('DDI'; see http://www.icpsr.umich.edu/DDI). As an initiative of the social science data-archiving community, the drive for common markup of data documents

preceded the field's engagement with grid computing. In a grid computing context the provision of metadata standards has not only facilitated data sharing, but provided ways of addressing changes in datasets arising from corrections and annotation. Corrections and other changes are a major concern, as changes of data can change statistical results. Standards like the DDI explicitly incorporate ways of handling such changes.

The myGrid UK e-science project (Stevens et al., 2003) has devoted a great deal of attention to capturing, recording and linking provenance data with a researcher's modes of work in fields like bioinformatics. As to curation, bioinformatics, again, has promoted a culture of excellence in annotation and curation, giving rise to data resources in the field of molecular biology that are credited with notable recent successes (e.g. the UniProt/Swiss-Prot Protein Knowledgebase (Wu, 2006), and the Protein Data Bank (Berman et al., 2000).

Most people think of data resources as being conceptualised and interpreted by humans, who then make decisions on whether to use the resource, how to combine it with other resources, which tools to use in order to analyse it, and so on. Like most decision-making activities, the processes above are very knowledge-intensive, and much experience and expertise has to be held by the person making those decisions.

The reason why commenting and annotating are especially important in e-Research is because one of the central goals of the latter is to delegate to software systems as many as possible of the tasks involved in making intelligent use of the comments and annotations, thereby relieving human users of this need and leaving them with more time to concentrate on substantive, domain-specific research questions.

For this goal of automating decision making to be possible, the comments and annotations have to be standardised into using formal knowledge-representation techniques, on the basis of which decision-making software can then be developed and deployed.

How do vocabularies, thesauri and ontologies fit in this context?

Of course, to a limited extent, data resources have always been described. One expects to find them associated with contextual information (e.g. scope in space and time), with an explanation of collection procedures (e.g. sampling methods) and with keys to the codes used, amongst other information.

Moreover, software systems have always relied on explicit description of the kinds of data in a resource (e.g. whether some feature or attribute is numeric or nominal, and so on) in order to enforce some correctness conditions when they are operated upon (e.g. that only numbers may be added, or that only strings may be concatenated).

Both the above kinds of descriptive information are used in decision making, but the latter kind is particularly important in understanding how vocabularies, thesauri and ontologies fit in this context.

The description of an attribute is one of the most basic, and limited, kinds of knowledge representation that underpin automated decision making by software systems.

Controlled vocabularies, thesauri and ontologies are also explicitly formalised knowledge and are likewise used by decision making software, but they express more facts about the data and therefore open the way for more sophisticated decisions than can be done by associating an attribute with a data type.

Figure 6.8 depicts relationships between controlled vocabularies, thesauri and ontologies. Controlled vocabularies, thesauri and ontologies are kinds of knowledge base of, respectively, increasing expressiveness, i.e., the formal language required to write down a controlled vocabulary is less expressive (i.e. is only capable of underpinning less powerful and intricate decision-making processes) than the one required to write down a thesaurus, and this latter language is, in turn, less expressive than the one required to write down an ontology. Although the terms are often used interchangeably, one approach to clarifying

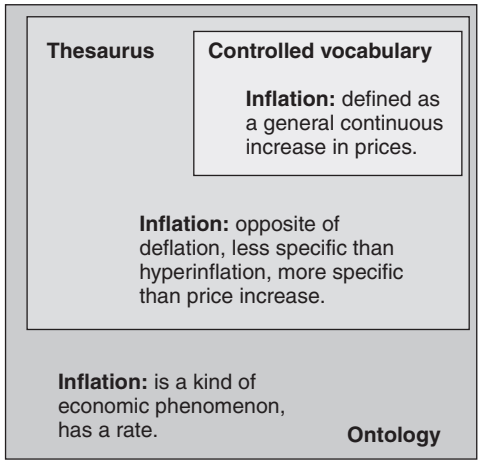

Figure 6.8 Controlled vocabularies, thesauri and ontologies

the purpose of distinguishing them is, roughly, as follow.[8]

A *controlled vocabulary* is a kind of knowledge base that holds definitions of terms. The assumption is, therefore, that such definitions demarcate usage boundaries and hence define the interpretation context more tightly than would otherwise be the case. A *thesaurus* is a more expressive kind of knowledge base than a controlled vocabulary insofar as it holds, in addition to definitions, assertions of lexical-level semantic relationships (e.g. synonymy) between terms. An *ontology* is one of the most expressive kinds of knowledge base. An ontology holds semantic relationships (e.g. taxonomic and mereologic) between terms that allow powerful modes of reasoning to be done by mediators.

In all cases, formalisation is the ultimate aim, insofar as the greatest benefits accrue from being able to have tools that make decisions in the light of such formalised knowledge representations. In all cases too, formalisation is best preceded by standardisation at the level of stakeholders.

Thus, to be truly useful, an ontology must be a consensus outcome of discussions among the stakeholders. This is a social, collective endeavour and, as such, not easy to carry through. But the pay-off can be huge: the more expressive the representation, the more, and the more complex, the actions that a mediator can hope to automate. The kind of decision-making software that is envisaged in e-Research operates upon comments and annotation in formal, knowledge-representation languages. A set of sentences in a knowledge-representation language is referred to as a knowledge base. One impressive demonstration of the power of ontologies to describe a domain of enquiry is the collective effort known as the Open Biomedical Ontologies project.

In the social sciences, in particular, it is often argued that aiming for such community-level consensus goes against the grain of the disciplines. In other words, that the kind of formalisation of concepts which knowledge-representation techniques assume stifles the very progress of the discipline. However, even if it is not possible to construct a consensual conceptualisation, it remains the case that the process of formalising alternative conceptualisations, however many are proposed, allows their objective assessment in terms of whether one subsumes the other, what consequences are derivable in one approach and not in any other, etc. Such assessments can be derived formally by using the automated reasoning tools for which ontologies represent one (indeed, a major) kind of input.

It must be stressed that the middleware products for data integration that constitute the topic of this chapter simply assume that agreements on semantics do exist. The role of the middleware products described does not go beyond providing sound and principled techniques for enacting those agreements once they have been externally negotiated by the stakeholders. Thus, it must be stressed that while this chapter paints an optimistic picture that enacting whatever semantic agreements that have been achieved is feasible, it does not follow that such agreements will be easy to achieve. At most, the existence of an enactment infrastructure could be seen to act as a spur for the agreements to be pursued, since achieving such agreements would more clearly and directly lead to significant benefits (again, see e.g. Cimino and Zhu, 2006).

Is there a relationship between this and the Semantic Web?

In a broader context, Semantic Web (Antoniou and van Harmelen, 2004), as well as Semantic Grid (Goble et al., 2005) is a term used to signify the use of formal knowledge-representation techniques to comment and annotate Web (as well as Grid) resources. The purpose of this is to make possible the widespread deployment of powerful decision-making software based on formal reasoning techniques.

Some examples of advanced functionalities that, it is hoped, the Semantic Web will make possible include personalised search – that is, computational agents that will act as proxies for people when performing complex tasks (such as arranging a holiday), with the degree of sophistication and appropriateness of performance that a human personal assistant would be expected to display.

The fundamental contrast between the Web with which we are currently familiar and the Semantic Web is that in the latter, resources are commented and annotated explicitly to form expressive knowledge bases. This makes it possible for automated decision making to take into account semantic (e.g. contextual) information as stipulated by the comments and annotations in knowledge bases that accompany and complement data resources (whose contents, by way of contrast, are merely the result of syntactic choices).

Ongoing activity in the Semantic Web and Semantic Grid research communities has made progress on many fronts, including the definition of languages of increasing expressiveness for commenting and annotating resources (e.g. XML, RDF, and OWL; Antoniou and van Harmelen, 2004).

A FORWARD LOOK

So far, this chapter has covered the motivations, challenges, solutions and benefits relating to the deployment of middleware components in fabrics such as the Grid and the Web with a view towards contributing to public infrastructure for *in silico* science.

This view dominates the thinking behind national and international initiatives for the future of scientific research that stands to benefit from widespread availability of data and the possibility of very large computational power from harnessing distributed resources. This might be called e-Science in the large. It is centred on research programmes, vast in scope and with spans measured in years. The major goal is collaboration and sharing, as required by large-scale initiatives that gravitate towards the notion of grand challenges in computing research.[9] The relationship with public support is justified, therefore, by the assumption that solutions would constitute strategic gateways to the future competitiveness of nations and/or substantial advances against enduring social problems.

While there is little dispute that sciences such as physics, chemistry and biology have much to benefit from this kind of support (hence their current status as the primary beneficiaries of the kind of middleware for data management that this chapter focuses on), in the social sciences and the humanities there may be significant benefit in a complementary view.

This view, which by way of contrast one might call e-Science in the small, is researcher-centric. In this case, the impetus is towards gathering and aggregating with minimal effort, since the driver is the development of a personal perspective on some research question. For this type of *in silico* science, low entry-cost is crucial. It must be possible for a researcher to gather data from different sources and to aggregate with minimal effort using lightweight interfaces and tools. Rather than the deep (and hence technically challenging) semantic integration discussed above, in e-Science in the small, collation, correlation and juxtaposition of data from various distributed resources may suffice, since the act of interpretation is anyway performed by the researcher that brought the data together.

Recent technical developments have led some to refer to some initiatives in the Web as comprising the Web 2.0 (O'Reilly, 2005), i.e., an evolution not only of the original Web, through to a Web of services (referred

to as the Web 1.5) to what some call (rather misleadingly) the '*Programmable Web*.'

There is no clear consensus as yet of what these notions designate, but the foundation technologies for the so-called Web 2.0, and the applications that are most often cited as paradigmatic of the advances on which it builds often involve some data integration. In this respect, they could be seen as complementary to the middleware solutions described above and as being an enabling technology for e-Science in the small, just as those solutions are enablers for e-Science in the large.

In this respect, the notion of an *information mash-up*, i.e., a quickly put-together collation and correlation of information that is exposed to users in a form that they perceive as sufficiently seamless, is proving very interesting. There are many interesting examples of mash-ups (Hof, 2005; *Mashup@Wikipedia*) already in use, but for the sake of illustration one could consider collating information from sites where ads are placed (thereby providing a window on consumption patterns and economic status), with information from sites that publish league tables on school and hospitals (thereby providing a window on the presumed quality of public services) with information from sites that provide digital maps (thereby associating the previous information with a specific geography).

In terms of the issues and challenges discussed in this chapter, the contrast to be drawn is not that such a global view could not be achieved with tools for e-Science in the large, but rather that, if the investigation context is that of an individual researcher, this could be achieved by a mash-up, i.e., by using lightweight tools that collate, correlate and juxtapose, thereby stopping short of claiming that an integrated view has been derived, and much less that that view is consensual in any way. For further discussion of mashups, see Hardey and Burrows (this volume).

CONCLUDING REMARKS

Undoubtedly, the widespread availability of fabrics such as the Grid and the Web will continue to transform the research landscape, particularly in respect of the availability of data and knowledge resources. The importance of good tool support for the generation of integrated views over distributed, heterogeneous, autonomous datasets is only set to grow as the trend towards more tool-driven, automated decision making intensifies to cope with the inherent complexity that results from the ubiquitous nature of the fabric and its indiscriminate reach.

There are initial solutions to some of the problems arising and their scope is expanding. More recently, and complementary to the initial efforts, novel ways of supporting not only collaborative programmes, but individual ones as well, are beginning to gain a foothold in the set of tools one can use to address data-integration questions and to enhance the benefits of data availability.

This is not to belittle the practical obstacles that need to be overcome. Grid technology is still complex to use at present. It is to be hoped that, as potential benefits convert to actual benefits, adoption levels will rise and, over time, the complexity will increasingly recede to lie behind tools that are usable by social scientists. At the moment, Grid-enabling datasets still requires costly and scarce human expertise. This chapter has highlighted the fact that progress is being continuously made towards reducing such costs and complexity, but this should not be understood as implying that, at present, barriers are not significant.

While it is still too early to say whether there will be closer integration between the tools that encapsulate best practice in e-Science in the large and the tools that are beginning to underpin the idea of e-Science in the small, there is the robust hope that there will.

NOTES

1 Biology is probably the science that has made the most progress in this direction. Every year, since 1996 the Nucleic Acids Research journal (http://nar.oxfordjournals.org/) has devoted its first issue of the year to a free-access special issue reviewing the state-of-the-art data resources in tools for *in silico* biological science. While the picture in the social

sciences is far from being comparable, Cole et al. (this volume) provides a glimpse of progress being made.

2 Several hypothetical scenarios are used from this point on to illustrate the technical issues involved. It is crucial for the reader to bear in mind that no claim is being made that such technical issues have indeed been solved for any of the examples used. All the examples should be taken as illustrative only.

3 Throughout this chapter, reference is made to the Grid. This is because the best examples of middleware for distributed data management were developed by the UK e-Science research initiative which, at its inception, was Grid-centric. Over time, there has been a greater, and quicker, convergence of the Grid and the Web than was anticipated and most Grid middleware is now also usable in the Web.

4 By *Grid fabric* is meant the services and protocols made available by lower-level Grid middleware such as the Globus Toolkit. As already mentioned, most Grid middleware can now be used over the Web, and not just over the Grid. Both OGSA-DAI and OGSA-DQP, which are discussed in this section, fall into this category.

5 In practice, of course, the illusion of transparency is never perfect because in extremely complex environments, such as wide-area networks, many qualities (e.g. latency, reliability, availability) cannot be guaranteed to emerge at sufficiently demanding levels of expected compliance.

6 By 'provenance' in this context is meant a trace of all the processes that were used to create a particular data value. Each step in the trace may include a time stamp, a claim of responsibility for the generation, and other metadata. For example, an average salary might be annotated with a provenance record that stipulates where the original values were obtained, when the average was calculated, which method was used, and who (or which step in the overall experimental procedure) was responsible.

7 By 'curation' in this context is meant the adding, by experts, of pertinent metadata (e.g. about authoritativeness) which enables more informed uses of the data.

8 Interested readers may find more information in Staab and Studer (2004).

9 For a UK perspective on the vision that has inspired the notion of grand challenges in computing research, see http://www.ukcrc.org.uk/grand_challenges/.

ACKNOWLEDGEMENTS

A great portion of the material in this chapter appeared first as a tutorial in the Web site of the National Centre for e-Social Science, funded by the UK ESRC. The author is grateful to Rob Procter, Titto Assini and Laura Bond for providing comments and detail that resulted in improvements in the text of that tutorial. That material has been revised and extended. Thanks are due too to the reviewers, whose comments helped free the text from many errors and imperfections, and who contributed to the enrichment of the text. Any errors are the author's responsibility.

REFERENCES

Alonso, G., Casti, F., Kuno, H. and Machiraju, V. (2003) *Web Services: Concepts, Architectures and Applications*. Dordrecht: Springer Verlag.

Alpdemir, M., Nedim, Mukherjee, Arijit, Paton, Norman, W., Watson, Paul, Fernandes, Alvaro, A.A., Gounaris, Anastasios, Smith, Jim (2003) 'Service-based distributed querying on the grid'. First International Conference on Service-Oriented Computing, Trento, Italy, December 15–18. Springer-Verlag LNCS 2910, pp. 467–482.

Antonioletti, Mario, Atkinson, Malcolm, P., Baxter, Rob, Borley, Andrew, Chue Hong, Neil, P., Collins, Brian, Hardman, Neil, Hume, Alastair, C., Knox, Alan, Jackson, Mike, Krause, Amrey, Laws, Simon, Magowan, James, Paton, Norman, W., Pearson, Dave, Sugden, Tom, Watson, Paul, Westhead, Martin (2005) 'The design and implementation of grid database services in OGSA-DAI', *Concurrency – Practice and Experience*, 17 (2–4): 357–376.

Antonioletti, Mario, Krause, Amy, Paton, Norman, W., Eisenberg, Andrew, Laws, Simon, Malaika, Susan, Melton, Jim, Pearson, Dave (2006) 'The WS-DAI family of specifications for web service data access and integration', *SIGMOD Record*, 35 (1): 48–55.

Antoniou, Grigoris and Harmelen, Frank van (2004) *A Semantic Web Primer*. The MIT Press.

Berman, H.M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T.N., Weissig, H., Shindyalov, I.N. and Bourne, P.E. (2000) 'The protein data bank', *Nucleic Acids Research*, 28: 235–242.

Brachman, Ronald, J. and Levesque, Hector J. (2004) *Knowledge, Representation and Reasoning*. Morgan Kaufmann.

Buetow, Kenneth, H. (2005) 'Cyberinfrastructure: Empowering a "Third Way"', *Biomedical Research, Science*, 308 (5723): 821–824.

Cimino, J.J., and Zhu, X. (2006) 'The Practical Impact of Ontologies on Biomedical Informatics'. In *IMIA Yearbook of Medical Informatics*.

Clery, Daniel (2006) 'Can Grid computing help us work together?', *Science*, 313 (5786): 433–434.

Foster, Ian, T. (2002) 'What is the Grid? A three-point checklist'. *GRIDToday*, 20 July. At: http://www-fp.mcs.anl.gov/~foster/Articles/WhatIsTheGrid.pdf

Foster, Ian, T. (2005) Globus Toolkit Version 4: Software for service-oriented systems. IFIP International Conference on Network and Parallel Computing. Springer-Verlag LNCS 3779. pp. 2–13.

Foster, Ian, T. and Grossman, Robert, L. (2003) 'Data integration in a bandwidth-rich world', *Commun. ACM*, 46 (11): 50–57.

Foster, Ian, T. and Kesselman, C. (eds) (2003) *The Grid: blueprint for a new computing infrastructure* (2dn edn). New York: Morgan Kaufmann.

Furukawa, Koichi (ed.) (2004) *New Generation Computing*, 22 (2). Special Issue on grid systems for life sciences.

Goble, Carole, A., De Roure, David, Shadbolt, Nigel, R. and Fernandes, Alvaro, A.A. (2003) 'Enhancing services and applications with knowledge and semantics'. In I. Foster and C. Kesselman (eds) *The Grid: Blueprint for a New Computing Infrastructure* (2nd. edn). Morgan Kaufmann. Ch. 23, pp. 431–458.

Goble, Carole, A., Kesselman, Carl and Sure, York (2005) 'Semantic Grid: The Convergence of Technologies', 3–8 July, Internationales Begegnungs- und Forschungszentrum für Informatik (IBFI). Schloss Dagstuhl, Germany.

Hof, Robert, D. (2005) 'Mix, match, and mutate'. *Business Week*, 25 July.

Kim, Sangtae (ed.) (2006) 'Cyberinfrastructure: Enabling the chemical sciences', *Journal of Chemical Information and Modeling*, 46 (3): (special issue).

Kim, Won, Choi, Injun, Gala, Sunit, K. and Scheeve, Mark (1995) 'On resolving schematic heterogeneity in multidatabase systems'. In *Modern Database Systems: The Object Model, Interoperability, and Beyond*. ACM Press and Addison-Wesley. pp. 521–550.

O'Reilly, Tim (2005) 'What Is Web 2.0?' At: http://www.oreillynet.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html

Staab, S. and R. Studer (eds) (2004) *Handbook on Ontologies*. Springer Verlag.

Stevens, Robinson, Alan, J. and Globe, Carole, A. (2003) 'myGrid: Personalised bioinformatics on the information grid', *ISMB (Supplement of Bioinformatics)*, 302–304.

Stevens, Robert, D., Wroe, Chris, Lord, Phillip, W. and Goble, Carole, A. (2004a) 'Ontologies in bioinformatics'. In Steffen Staab and Rudi Studer (eds) *Handbook on Ontologies*. Springer. pp. 635–658

Stevens, Robert, D., Tipney, Hannah, J., Wroe, Chris, Oinn, Thomas, M., Senger, Martin, Lord, Phillip, W., Goble, Carole, A., Brass, Andy and Tassabehji, M. (2004b) 'Exploring Williams–Beuren syndrome using myGrid. ISMB/ECCB', *Supplement of Bioinformatics*, 303–310.

Wiederhold, Gio (1992) 'Mediators in the architecture of future information systems', *IEEE Computer*, 25 (3): 38–49.

Wu, Cathy, H., Apweiler, Rolf, Bairoch, Amos, Natale, Darren, A., Barker, Winona, C., Boeckmann, Brigitte, Ferro, Serenella, Gasteiger, Elisabeth, Huang, Hongzhan, Lopez, Rodrigo, Magrane, Michele, Martin, Maria, J., Mazumder, Raja, O'Donovan, Claire, Redaschi, Nicole and Suzek, Baris (2006) 'The Universal Protein Resource (UniProt): an expanding universe of protein information', *Nucleic Acids Research*, 34: D187–D191.

FURTHER READING

Since all the topics covered in this chapter remain the focus of much research-oriented activity in computer science, the related literature is unavoidably technical. The references discussed in this paragraph are meant to allow interested social scientists to broaden and deepen their understanding, but they were not written for a social science audience and would, therefore, require some tenacity on the part of the reader in order to grasp the conceptual frameworks that underpin them. References mentioned here are listed in the main reference list below. The classical reference for the Grid, construed as a vision for a technical infrastructure that harnesses software, hardware and network resources with a view to enabling large-scale cooperation and collaboration in virtual organisations, is Foster and Kesselman (2003). The seminal paper on mediation as a means to contend with different forms of heterogeneity is Wiederhold (1992). A detailed survey of the service-oriented approach to web applications is Alonso et al. (2003). The best references for data access and integration in service-oriented grids are Alpdemir et al. (2003); Antonioletti et al. (2005); Antonioletti et al. (2006). A comprehensive treatment of the computing view on ontologies is Staab and Studer (2004).