1

Data mining meets grid computing: Time to dance?

Alberto Sánchez, Jesús Montes, Werner Dubitzky, Julio J. Valdés, María S. Pérez and **Pedro de Miguel**

ABSTRACT

A *grand challenge problem* (Wah, 1993) refers to a computing problem that cannot be solved in a reasonable amount of time with conventional computers. While grand challenge problems can be found in many domains, science applications are typically at the forefront of these large-scale computing problems. Fundamental scientific problems currently being explored generate increasingly complex data, require more realistic simulations of the processes under study and demand greater and more intricate visualizations of the results. These problems often require numerous complex calculations and collaboration among people with multiple disciplines and geographic locations. Examples of scientific grand challenge problems include multi-scale environmental modelling and ecosystem simulations, biomedical imaging and biomechanics, nuclear power and weapons simulations, fluid dynamics and fundamental computational science (use of computation to attain scientific knowledge) (Butler, 1999; Gomes and Selman, 2005).

Many grand challenge problems involve the analysis of very large volumes of data. *Data mining* (also known as *knowledge discovery in databases*) (Frawley, Piatetsky-Shapiro and Matheus, 1992) is a well stablished field of computer science concerned with the automated search of large volumes of data for patterns that can be considered knowledge about the data. Data mining is often described as deriving knowledge from the input data. Applying data mining to grand challenge problems brings its own computational challenges. One way to address these computational challenges is *grid computing* (Kesselman and Foster, 1998). 'Grid' refers to persistent computing environments that enable software applications to integrate processors, storage, networks, instruments, applications and other resources that are managed by diverse organizations in widespread locations.

This chapter describes how both paradigms – data mining and grid computing – can benefit from each other: data mining techniques can be efficiently deployed in a grid environment and operational grids can be mined for patterns that may help to optimize the effectiveness and efficiency of the grid computing infrastructure. The chapter will also briefly outline the chapters of this volume.

Data Mining Techniques in Grid Computing Environments Edited by Werner Dubitzky

1.1 Introduction

Recent developments have seen an unprecedented growth of data and information in a wide range of knowledge sectors (Wright, 2007). The term *information explosion* describes the rapidly increasing amount of published information and its effects on society. It has been estimated that the amount of new information produced in the world increases by 30 per cent each year. The Population Reference Bureau [1] estimates that 800 MB of recorded information are produced per person each year (assuming a world population of 6.3 billion). Many organizations, companies and scientific centres produce and store large amounts of complex data and information. Examples include climate and astronomy data, economic and financial transactions and data from many scientific disciplines. To justify their existence and maximize their use, these data need to be stored and analysed. The larger and the more complex these data, the more time consuming and costly is their storage and analysis.

Data mining has been developed to address the information needs in modern knowledge sectors. *Data mining refers to the non-trivial process of identifying valid, novel, potentially useful and understandable patterns in large volumes of data* (Fayyad, Piatetsky-Shapiro and Smyth, 1996; Frawley, Piatetsky-Shapiro and Matheus, 1992). Because of the information explosion phenomenon, data mining has become one of the most important areas of research and development in computer science.

Data mining is a complex process. The main dimensions of complexity include the following (Stankovski *et al.*, 2004).

- *Data mining tasks.* There are many non-trivial tasks involved in the data mining process: these include data pre-processing, rule or model induction, model validation and result presentation.
- *Data volume.* Many modern data mining applications are faced with growing volumes (in bytes) of data to be analysed. Some of the larger data sets comprise millions of entries and require gigabytes or terabytes of storage.
- *Data complexity.* This dimension has two aspects. First, the phenomena analysed in complex application scenarios are captured by increasingly *complex data structures and types*, including natural language text, images, time series, multi-relational and object data types. Second, data are increasingly located in *geographically distributed* data placements and cannot be gathered centrally for technological (e.g. large data volumes), data privacy (Clifton *et al.*, 2002), security, legal or other reasons.

To address the issues outlined above, the data mining process is in need of reformulation. This leads to the concept of *distributed data mining*, and in particular to grid-based data mining or – in analogy to a data grid or a computational grid – to the concept of a *data mining grid*. A data mining grid seeks a trade-off between data centralization and distributed processing of data so as to maximize effectiveness and efficiency of the entire process (Kargupta, Kamath and Chan, 2000; Talia, 2006). A data mining grid should provide a means to exploit available hardware resources (primary/secondary memory, processors) in order to handle the data volumes and processing requirements of modern data mining applications. Furthermore, it should support data placement, scheduling and resource management (Sánchez *et al.*, 2004).

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

Grid computing has emerged from distributed computing and parallel processing technologies. The so-called *grid* is a distributed computing infrastructure facilitating the coordinated sharing of computing resources within organizations and across geographically dispersed sites. The main advantages of sharing resources using a grid include (a) pooling of heterogeneous computing resources across administrative domains and dispersed locations, (b) ability to run large-scale applications that outstrip the capacity of local resources, (c) improved utilization of resources and (d) collaborative applications (Kesselman and Foster, 1998).

Essentially, a grid-enabled data mining environment consists of a decentralized high-performance computing platform where data mining tasks and algorithms can be applied on distributed data. Grid-based data mining would allow (a) the distribution of compute-intensive data analysis among a large number of geographically scattered resources, (b) the development of algorithms and new techniques such that the data would be processed where they are stored, thus avoiding transmission and data ownership/privacy issues, and (c) the investigation and potential solution of data mining problems beyond the scope of current techniques (Stankovski *et al.*, 2008).

While grid technology has the potential to address some of the issues of modern data mining applications, the complexity of the grid computing environments themselves gives rise to various issues that need to be tackled. Amongst other things, the heterogeneous and geographically distributed nature of grid resources and the involvement of multiple administrative domains with their local policies make coordinated resource sharing difficult. Ironically, (distributed) data mining technology could offer possible solutions to some of the problems encountered in complex grid computing environments. The basic idea is that the operational data that is generated in grid computing environments (e.g. log files) could be mined to help improve the overall performance and reliability of the grid, e.g. by identifying misconfigured machines.

Hence, there is the potential that both paradigms – grid computing and data mining – could look forward to a future of fruitful, mutually beneficial cooperation.

1.2 Data mining

As already mentioned, data mining refers to the process of extracting useful, non-trivial knowledge from data (Witten and Frank, 2000). The extracted knowledge is typically used in business applications, for example fraud detection in financial businesses or analysis of purchasing behaviour in retail scenarios. In recent years data mining has found its way into many scientific and engineering disciplines (Grossman *et al.*, 2001). As a result the complexity of data mining applications has grown extensively (see Subsection 1.2.1). To address the arising computational requirements distributed and grid computing has been investigated and the notion of a data mining grid has emerged. While the marriage of grid computing and data mining has seen many success stories, many challenges still remain – see Subsection 1.2.2. In the following subsections we briefly hint at some of the current data mining issues and scenarios.

1.2.1 Complex data mining problems

The complexity of modern data mining problems is challenging researchers and developers. The sheer scale of these problems requires new computing architectures, as conventional systems can no longer cope. Typical large-scale data mining applications are found in areas

such as molecular biology, molecular design, process optimization, weather forecast, climate change prediction, astronomy, fluid dynamics, physics, earth science and so on. For instance, in high-energy physics, CERN's (European Organization for Nuclear Research) Large Hadron Collider[2] is expected to produce data in the range of 15 petabytes/s generated from the smashing of subatomic particles. These data need to be analysed in four experiments with the aim of discovering new fundamental particles, specifically the Higgs boson or God particle[3].

Another current complex data mining application is found in weather modelling. Here, the task is to discover a model that accurately describes the weather behaviour according to several parameters. The climate*prediction*.net[4] project is the largest experiment to produce forecasts of the climate in the 21st century. The project aims to understand how sensitive weather models are to both small changes and to factors such as carbon dioxide and the sulphur cycle. To discover the relevant information, the model needs to be executed thousands of times.

Sometimes the challenge is not the availability of sheer compute power or massive memory, but the intrinsic geographic distribution of the data. The mining of medical databases is such an application scenario. The challenge in these applications is to mine data located in distributed, heterogeneous databases while adhering to varying security and privacy constraints imposed on the local data sources (Stankovski *et al.*, 2007).

Other examples of complex data mining challenges include large-scale data mining problems in the life sciences, including disease modelling, pathway and gene expression analysis, literature mining, biodiversity analysis and so on (Hirschman *et al.*, 2002; Dubitzky, Granzow and Berrar, 2006; Edwards, Lane and Nielsen, 2000).

1.2.2 Data mining challenges

If data mining tasks, applications and algorithms are to be distributed, some data mining challenges are derived from current distributed processing problems. Nevertheless, data mining has certain special characteristics – such as input data format, pressing steps and tasks – which should be taken into account.

In recent years, two lines of research and development have featured prominently the evolution of data mining in distributed computing environments.

- *Development of parallel or high-performance algorithms, theoretical models and data mining techniques.* Distributed data mining algorithms must support the complete data mining process (pre-processing, data mining and post-processing) in a similar way as their centralized versions do. This means that all data mining tasks, including data cleaning, attribute discretization, concept generalization and so on, should be performed in a parallel way. Several distributed algorithms have been developed according to their centralized versions. For instance, some parallel algorithms have been developed for association rules (Agrawal and Shafer, 1996; Ashrafi, Taniar and Smith, 2004), classification rules (Zaki, Ho and Agrawal, 1999; Cho and Wüthrich, 2002) or clustering algorithms (Kargupta *et al.*, 2001; Rajasekaran, 2005).

[2]http://lhc.web.cern.ch/lhc/

[3]The Higgs boson is the key particle to understanding why matter has mass.

[4]www.climateprediction.net/

- *Design of new data mining systems and architectures to deal with the efficient use of computing resources.* Although some effort has been made towards the development of efficient distributed data mining algorithms, the environmental aspects, such as task scheduling and resource management, are critical aspects to the success of distributed data mining. Therefore, the deployment of data mining applications within high-performance and distributed computing infrastructures becomes a challenge for future developments in the field of data mining. This volume is intended to cover this dimension.

Furthermore, current data mining problems require more development in several areas including data placement, data discovery and storage, resource management and so on, because of the following.

- The high complexity (data size and structure, cooperation) of many data mining applications requires the use of data from multiple databases and may involve multiple organizations and geographically distributed locations. Typically, these data cannot be integrated into a single, centralized database data warehouse due to technical, privacy, legal and other constraints.
- The different institutions have maintained their own (local) data sources using their preferred data models and technical infrastructures.
- The possible geographically dispersed data distribution implies fault tolerance and other issues. Also, data and data model (metadata) updates may introduce replication and data integrity and consistency problems.
- The huge volume of analysed data and the existing difference between computing and I/O access times require new alternatives to avoid the I/O system becoming a bottleneck in the data mining processes.

Common data mining deployment infrastructures, such as clusters, do not normally meet these requirements. Hence, there is a need to develop new infrastructures and architectures that could address these requirements. Such systems should provide (see, for example, Stankovski *et al.,* 2008) the following.

- *Access control, security policies and agreements between institutions to access data.* This ensures seamless data access and sharing among different organizations and thus will support the interoperation needed to solve complex data mining problems effectively and efficiently.
- *Data filtering, data replication and use of local data sets.* These features enhance the efficiency of the deployment of data mining applications. Data distribution and replication need to be handled in a coherent fashion to ensure data consistency and integrity.
- *Data publication, index and update mechanisms.* These characteristics are extremely important to ensure the effective and efficient location of relevant data in large-scale distributed environments required to store the large number of data to be analysed.
- *Data mining planning and scheduling based on the existing storage resources.* This is needed to ensure effective and efficient use of the computing resources within a distributed computing environment.

In addition to the brief outline described above, we highlight some of the key contemporary data mining challenges as identified by (Yang and Wu, 2006). We highlight those that we feel are of particular relevance to ongoing research and development that seeks to combine grid computing and data mining.

(a) Mining complex knowledge from *complex data.*

(b) *Distributed data mining* and mining multi-agent data.

(c) *Scaling up for high-dimensional* and high-speed *data* streams.

(d) *Mining in a network setting.*

(e) *Security, privacy and data integrity.*

(f) Data mining for biological and environmental problems.

(g) Dealing with non-static, unbalanced and cost-sensitive data.

(h) Developing a unifying theory of data mining.

(i) Mining sequence data and time series data.

(j) Problems related to the data mining process.

1.3 Grid computing

Scientific, engineering and other applications and especially grand challenge applications are becoming ever more demanding in terms of their computing requirements. Increasingly, the requirements can no longer be met by single organizations. A cost-effective modern technology that could address the computing bottleneck is grid technology (Kesselman and Foster, 1998).

In the past 25 years the idea of sharing computing resources to obtain the maximum benefit/cost ratio has changed the way we think about computing problems. Expensive and difficult-to-scale supercomputers are being complemented and sometimes replaced by affordable distributed computing solutions.

Cluster computing was the first alternative to multiprocessors, aimed at obtaining a better cost/performance ratio. A cluster can be defined as a set of dedicated and independent machines, connected by means of an internal network, and managed by a system that takes advantage of the existence of several computational elements. A cluster is expected to provide high performance, high availability, load balancing and scalability.

Although cluster computing is an affordable way to solve complex problems, it does not allow us to connect different administration domains. Furthermore, it is not based on open standards, which makes applications less portable. Finally, current grand challenge applications have reached a level of complexity that even cluster environments may not be able to address adequately.

A second alternative to address grand challenge applications is called Internet computing. Its objective is to take advantage of not only internal computational resources (such as the nodes of a cluster) but also those general purpose systems interconnected by a *wide area network* (*WAN*). A WAN is a computer network that covers a broad area, i.e. any network whose communications links cross-metropolitan, regional or national boundaries. The largest and best-known example of a WAN is the Internet. This allows calculations and data analysis to

be performed in a highly distributed way by linking geographically widely dispersed resources. In most cases this technology is developed using free computational resources from people or institutions that voluntarily join the system to help scientific research.

A popular example of an Internet-enabled distributed computing solutions is the SETI@home[5] project (University of California, 2007). The goal of the project is the search for extra-terrestrial intelligence through the analysis of radio signals from outer space. It uses Internet-connected computers and a freely available software that that analyses narrow-bandwidth signals from radio telescope data. To participate in this large-scale computing exercise, users download the software and install it on their local systems. Chunks of data are sent to the local computer for processing and the results are sent to a distributor node. The program uses part of the computer's CPU power, disk space and network bandwidth and the user can control how much of the computer resources are used by SETI@Home, and when they can be used.

Similar '@home' projects have been organized in other disciplines under the *Berkeley Open Infrastructure for Network Computing* (*BOINC*)[6] initiative. In biology and medicine, Rosetta@home[7] uses Internet computing to find causes of major human diseases such as the *acquired immunodeficiency syndrome* (*AIDS*), malaria, cancer or Alzheimer's. Malariacontrol.net[8] is another project that adopts an Internet computing approach. Its objective is to develop simulation models of the transmission dynamics (epidemiology) and health effects of malaria.

In spite of its usefulness, Internet computing presents some disadvantages, mainly because most resources are made available by a community of voluntary users. This limits the use of the resources and the reliability of the infrastructure to solve problems in which security is a key factor. Even so, by harnessing immense computing power these projects have made a first step towards distributed computing architectures capable of addressing complex data mining problems in diverse application areas.

One of the most recent incarnations of large-scale distributed computing technologies is *grid computing* (Kesselman and Foster, 1998). The aim of grid computing is to provide an affordable approach to large-scale computing problems. The term grid can be defined as a set of computational resources interconnected through a WAN, aimed at performing highly demanding computational tasks such as grand challenge applications. A grid makes it possible to securely and reliably take advantage of widely dispersed computational resources across several organizations and administrative domains. An administrative domain is a collection of hosts and routers, and the interconnecting network(s), managed by a single administrative authority, i.e. a company, institute or other organization. The geographically dispersed resources that are aggregated within a grid could be viewed as a virtual supercomputer. Therefore, it has no centralized control, as each system still belongs to and is controlled by its original resource provider. The grid automates access to computational resources, assuring security restrictions and reliability.

Ian Foster defines the main characteristics of a grid as follows (Foster, 2002).

- *Decentralized control.* Within a grid, the control of resources is decentralized, enabling different administration policies and local management systems.

[5] http://setiathome.berkeley.edu/

[6] http://boinc.berkeley.edu/

[7] http://boinc.bakerlab.org/rosetta/

[8] http://www.malariacontrol.net/

- *Open technology.* A grid should use of open protocols and standards.
- *High quality of service.* A grid provides high quality of service in terms of performance, availability and security.

Grid solutions are specifically designed to be adaptable and scalable and may involve a large number of machines. Unlike many cluster and Internet computing solutions, a grid should be able to cope with unexpected failures or loss of resources. Commonly used systems (such as clusters) can only grow up to a certain point without significant performance losses. Because of the expandable set of systems that can be attached and adapted, grids can provide theoretical unlimited computational power.

Other advantages of a grid infrastructure can be summarized as follows.

- Overcoming of bottlenecks faced by many large-scale applications.
- Decentralized administration that allows independent administrative domains (such as corporative networks) to join and contribute to the system without losing administrative control.
- Integration of heterogeneous resources and systems. This is achieved through the use of open protocols and standard interconnections and collaboration between diverse computational resources.
- A grid system is able to adapt to unexpected failures or loss of resources.
- A grid environment never becomes obsolete as it may easily assimilate new resources (perhaps as a replacement for older resources) and be adapted to provide new features.
- Provide an attractive cost/performance ratio making high-performance computing affordable.

Current grids are designed so as to serve a certain purpose or community. Typical grid configurations (or types of grid) include the following.

- *Computing (or computational) grid.* This type of grid is designed to provide as much computing power as possible. This kind of environment usually provides services for submitting, monitoring and managing jobs and related tools. Typically, in a computational grid most machines are high-performance servers. Sometimes two types of computational grid are distinguished: distributed computing grids and high-throughput grids (Krauter, Buyya and Maheswaran, 2002).
- *Data grid.* A data grid stores and provides reliable access to data across multiple organizations. It manages the physical data storage, data access policies and security issues of the stored data. The physical location of the data is normally transparent to the user.
- *Service grid.* A service grid (Krauter, Buyya and Maheswaran, 2002) provides services that are not covered by a single machine. It connects users and applications into collaborative workgroups and enables real-time interaction between users and applications via a virtual workspace. Service grids include on-demand, collaborative and multimedia grid systems.

While grid technology has been used in productive setting for a while, current research still needs to address various issues. Some of these are discussed below.

1.3.1 Grid computing challenges

Although grid computing allows the creation of comprehensive computing environments capable of addressing the requirements of grand challenge applications, such environments can often be very complex. Complexity arises from the heterogeneity of the underlying software and hardware resources, decentralized control, mechanisms to deal with faults and resource losses, grid middleware such as resource broker, security and privacy mechanisms, local policies and usage patterns of the resources and so on. These complexities need to be addressed in order to fully exploit the grid's features for large-scale (data mining) applications.

One way of supporting the management of a grid is to monitor and analyse all information of an operating grid. This may involve information on system performance and operation metrics such as throughput, network bandwidth and response times, but also other aspects such as service availability or the quality of job–resource assignment[9]. Because of their complexity, distributed creation and real-time aspects, analyzing and interpreting the 'signals' generated within an operating grid environment can become a very complex data analytical task. Data mining technology is turning out to be the methodology of choice to address this task, i.e. to *mine grid data*.

1.4 Data mining grid – mining grid data

From the overview on data mining and grid technology, we see two interesting developments, the concept of a *data mining grid* and *mining grid data*. A data mining grid could be viewed as a grid that is specifically designed to facilitate demanding data mining applications. In addition, grid computing environments may motivate a new form of data mining, *mining grid data*, which is geared towards supporting the efficient operation of a grid by facilitating the analysis of data generated as a by-product of running a grid. These two aspects are now briefly discussed.

1.4.1 Data mining grid: a grid facilitating large-scale data mining

A *data mining application* is defined as the use of data mining technology to perform data analysis tasks within a particular application domain. Basic elements of a data mining application are the *data* to be mined, the data mining *algorithm*(s) and methods used to mine the data, and a *user* who specifies and controls the data mining process. A data mining process may consist of several data mining algorithms, each addressing a particular data mining task, such as feature selection, clustering or visualization. A given data mining algorithm may have different software implementations. Likewise, the data to be mined may be available in different implementations, for instance as a database in a database management system, a file in a particular file format or a data stream.

A data mining grid is a system whose main function is to facilitate the sharing and use of *data*, *data mining programs* (implemented algorithms), *processing units* and *storage devices*

[9] A grid job could be anything that needs a grid resource, e.g. a request for bandwidth or disk space, an application or a set of application programs.

in order to improve existing, and enable novel, data mining applications (see Subsection 1.2.2). Such a system should take into account the unique constraints and requirements of data mining applications with respect to the data management and data mining software tools, and the users of these tools (Stankovski *et al.*, 2008). These high-level goals lead to a natural breakdown of some basic requirements for a data mining grid. We distinguish *user*, *application* and *system requirements*. The user requirements are dictated by the need of end users to define and execute data mining tasks, and by developers and administrators who need to evolve and maintain the system. Application program and system requirements are driven by technical factors such as resource type and location, software and hardware architectures, system interfaces, standards and so on. Below we briefly summarize what these requirements may be.

Ultimately, a data mining grid system facilitating advanced data mining applications is operated by a human user – an end user wanting to solve a particular data mining task or a system developer or administrator tasked with maintaining or further developing the data mining grid. Some of the main requirements such users may have include the following.

- *Effectiveness and efficiency.* A data mining grid should facilitate more effective (solution quality) and/or more efficient (higher throughput, which relates to speed-up) solutions than conventional environments.
- *Novel use/application.* A data mining grid should facilitate novel data mining applications currently not possible with conventional environments.
- *Scalability.* A data mining grid should facilitate the seamless adding of grid resources to accommodate increasing numbers of users and growing application demands without performance loss.
- *Scope.* A data mining grid should support data mining applications from different application domains and should allow the execution of all kinds of data mining task (pre-processing, analysis, post-processing, visualization etc.)
- *Ease of use.* A data mining grid should hide grid details from users who do not want to concern themselves with such details, but be flexible enough to facilitate deep, grid-level control to those users wish to operate on this level. Furthermore, mechanisms should be provided by a data mining grid that allow users to search for grid-wide located data mining applications and data sources. Finally, a data mining grid should provide tools that help users to define complex data mining processes.
- *Monitoring and steering.* A data mining grid should provide tools that allow users to monitor and steer (e.g. abort, provide new input, change parameters) data mining applications running on the grid.
- *Extensibility, maintenance and integration.* Developers should be able to port existing data mining applications to the data mining with little or no modification to the original data mining application program. System developers should be able to extend the features of the core data mining grid system without major modifications to the main system components. It should be easy to integrate new data mining applications and core system components with other technology (networks, Web services, grid components, user interfaces etc).

To meet the user requirements presented above, a data mining grid should meet additional technical requirements relating to *data mining application* software (data, programs) and the

underlying data mining *grid system* components. Some basic requirements of this kind are as follows.

- *Resource sharing and interoperation.* A data mining grid should facilitate the seamless interoperation and sharing of important data mining resources and components, in particular, data mining application programs (implemented algorithms), data (different standard data file formats, database managements systems, other data-centric systems and tools), storage devices and processing units.
- *Data mining applications.* A data mining grid should accommodate a wide range of data mining application programs (algorithms) and should provide mechanisms that take into account the requirements, constraints and user-defined settings associated with these applications.
- *Resource management.* A data mining grid system should facilitate resource management to match available grid resources to job requests (resource broker), schedule the execution of the jobs on matched resources (scheduler) and manage and monitor the execution of jobs (job execution and monitoring). In particular, a data mining grid resource manager should facilitate data-oriented scheduling and parameter sweep applications, and take into account the type of data mining task, technique and method or algorithm (implementation) in its management policies.

1.4.2 Mining grid data: analysing grid systems with data mining techniques

Grid technology provides high availability of resources and services, making it possible to deal with new and more complex problems. But it is also known that a grid is a very heterogeneous and decentralized environment. It presents different kinds of security policy, data and computing characteristic, system administration procedure and so on. Given these complexities, the management of a grid, any grid not just a data mining grid, becomes a very important aspect in running and maintaining grid systems. Grid management is the key to providing high reliability and quality of service. The complexities of grid computing environments make it almost impossible to have a complete understanding of the entire grid. Therefore, a new approach is needed. Such an approach should pool, analyse and interpret all relevant information that could be obtained from a grid. The insights provided should then be used to support resource management and system administration. Data mining has proved to be a remarkably powerful tool, facilitating the analysis and interpretation of large volumes of complex data. Hence, given the complexities involved in operating and maintaining grid environments efficiently and the ability of data mining to analyse and interpret large volumes of data, it is evident that 'mining grid data' could be a solution to improving the performance, operation and maintenance of grid computing environments.

Nowadays, most management techniques consider the grid as a set of independent, complex systems, building together a huge pool of computational resources. Therefore, the administration procedures are subjected to a specific analysis of each computer system, organizational units, etc. Finally, the decision making is based on a detailed knowledge of each of the elements that make up a grid. However, if we consider how more commonly used systems (such as regular desktop computers or small clusters) are managed, it is easy to realize that resource administration is very often based on more general parameters such as CPU or memory usage, not directly related to the specific architectural characteristics, although it is affected by them. This can be considered as an abstraction method that allows administrators to generalize

and apply their knowledge to different systems. This abstraction is possible thanks to a set of underlying procedures, present in almost every modern computer.

Nevertheless, in complex systems such as a grid, this level of abstraction is not enough. The heterogeneous and distributed nature of grids implies a new kind of architectural complexity. Data mining techniques can contribute to observe and analyse the environment as a single system, offering a new abstraction layer that reduces grid observation to a set of representative generic parameters. This approach represents a new perspective for management, allowing consideration of aspects regarding the whole system activity, instead of each subsystem's behaviour.

The complexity of this formulation makes it hard to face grid understanding directly as a single problem. It is desirable to focus on a limited set of aspects, trying to analyse and improve them first. This can provide insight on how to deal with the abstraction of grid complexity, which can be extended to more complete scenarios. The great variety of elements that can be found in the grid offers a wide range of information to process. Data from multiple sources can be gathered and analysed using data mining techniques to learn new useful information about different grid features. The nature of the information obtained determines what kind of knowledge is going to be obtained.

Standard monitoring parameters such as CPU or memory usage of the different grid resources can provide insight on a grid's computational behaviour. A better knowledge of the grid variability makes it possible to improve the environment performance and reliability. A deep internal analysis of the grid can reveal weak points and other architectural issues.

From a different point of view, user behaviour can be analysed, focusing on access patterns, service request, the nature of these requests etc. This would make it possible to refine the environment features and capabilities, trying to effectively fit user needs and requirements.

The grid's dynamic evolution can also be analysed. Understanding the grid's present and past behaviour allows us to establish procedures to predict its evolution. This would help the grid management system to anticipate future situations and optimize its operation.

1.5 Conclusions

With the advance of computer and information technology, increasingly complex and resource-demanding applications have become possible. As a result, even larger-scale problems are envisaged and in many areas so-called *grand challenge problems* (Wah, 1993) are being tackled. These problems put an even greater demand on the underlying computing resources. A growing class of applications that need large-scale resources is modern data mining applications in science, engineering and other areas (Grossman *et al.*, 2001). Grid technology (Kesselman and Foster, 1998) is an answer to the increasing demand for affordable large-scale computing resources.

The emergence of grid technology and the increasingly complex nature of data mining applications have led to a new synergy of *data mining* and *grid*. On one hand, the concept of a *data mining grid* is in the process of becoming a reality. A data mining grid facilitates novel data mining applications and provides a comprehensive solution for affordable high-performance resources satisfying the needs of large-scale data mining problems. On the other hand, *mining grid data* is emerging as a new class of data mining application. Mining grid data could be understood as a methodology that could help to address the complex issues involved in running and maintaining large grid computing environments. The dichotomy of these

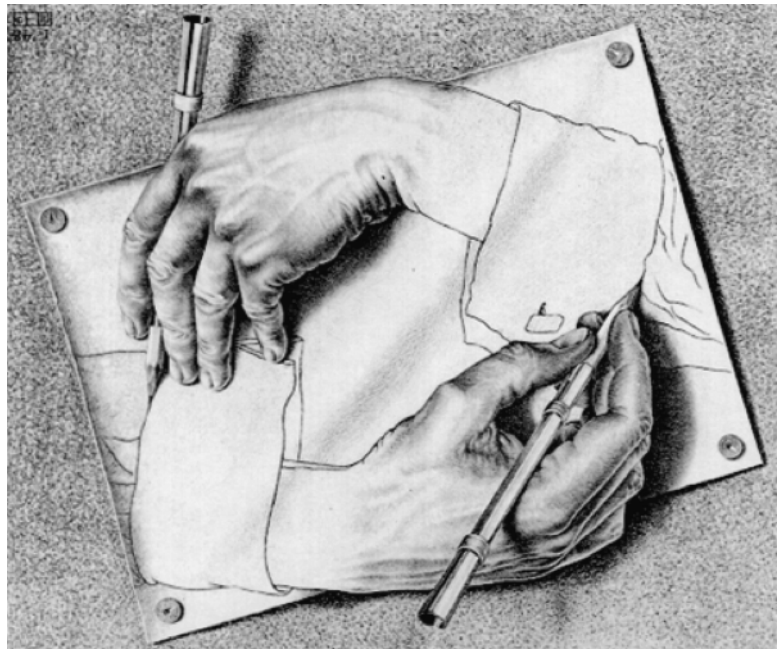

Figure 1.1 Analogy symbolizing the new synergy between a data mining grid and the mining of grid data. '*M. C. Escher* : *The Graphic Work*' (with permission from Benedikt-Taschen Publishers)

concepts – a data mining grid and mining grid data – is the subject of this volume and is beautifully illustrated in Figure 1.1. The two paradigms should go hand in hand and benefit from each other – a data mining grid can efficiently deploy large-scale data mining applications and data mining techniques can be used to understand and reduce the complexity of grid computing environments.

However, both areas are relatively new and demand further research and development. This volume is intended to be a contribution to this quest. What seems clear, though, is that the two areas are looking forward to a great future. The time has come to face the music and dance!

1.6 Summary of chapters in this volume

Chapter 1 is entitled 'Data mining meets grid computing: time to dance?'. The title indicates that there is a great synergy afoot, a synergy between data mining and grid technology. The chapter describes how the two paradigms – data mining and grid computing – can benefit from each other: data mining techniques can be efficiently deployed in a grid environment and operational grids can be mined for patterns that may help to optimize the effectiveness and efficiency of the grid computing infrastructure.

Chapter 2 is entitled 'Data analysis services in the knowledge grid'. It describes a grid-based architecture supporting distributed knowledge discovery called Knowledge Grid. It discusses how the Knowledge Grid framework has been developed as a collection of grid services and how it can be used to develop distributed data analysis tasks and knowledge discovery processes exploiting the service-oriented architecture model.

Chapter 3 is entitled 'GridMiner: an advanced support for e-science analytics'. It describes the architecture of the GridMiner system, which is based on the Cross Industry Standard Process for Data Mining. GridMiner provides a robust and reliable high-performance data mining and OLAP environment, and the system highlights the importance of grid-enabled applications in terms of e-science and detailed analysis of very large scientific data sets.

Chapter 4 is entitled 'ADaM services: scientific data mining in the service-oriented architecture paradigm'. The ADaM system was originally developed in the early 1990s with the goal of mining large scientific data sets for geophysical phenomena detection and feature extraction. The chapter describes the ADaM system and illustrates its features and functions on the basis of two applications of ADaM services within a SOA context.

Chapter 5 is entitled 'Mining for misconfigured machines in grid systems'. This chapter describes the Grid Monitoring System (GMS) – a system that adopts a distributed data mining approach to detection of misconfigured grid machines.

Chapter 6 is entitled 'FAEHIM: federated analysis environment for heterogeneous intelligent mining'. It describes the FAEHIM toolkit, which makes use of Web services composition, with the widely deployed Triana workflow environment. Most of the Web services are derived from the Weka data mining library of algorithms.

Chapter 7 is entitled 'Scalable and privacy-preserving distributed data analysis over a service-oriented platform'. It reviews a recently proposed scalable and privacy-preserving distributed data analysis approach. The approach computes abstractions of distributed data, which are then used for mining global data patterns. The chapter also describes a service-oriented realization of the approach for data clustering and explains in detail how the analysis process is deployed in a BPEL platform for execution.

Chapter 8 is entitled 'Building and using analytical workflows in Discovery Net'. It describes the experience of the authors in designing the Discovery Net platform and maps out the evolution paths for a workflow language, and its architecture, that address the requirements of different scientific domains.

Chapter 9 is entitled 'Building workflows that traverse the bioinformatics data landscape'. It describes how the myGrid supports the management of the scientific process in terms of *in silico* experimentation in bioinformatics. The approach is illustrated through an example from the study of trypanosomiasis resistance in the mouse model. Novel biological results obtained from traversing the 'bioinformatics landscape' are presented.

Chapter 10 is entitled 'Specification of Distributed data mining workflows with DataMiningGrid'. This chapter gives an evaluation of the benefits of grid-based technology from a data miner's perspective. It is focused on the DataMiningGrid, a standard-based and extensible environment for grid-enabling data mining applications.

Chapter 11 is entitled 'Anteater: service-oriented data mining'. It describes SOA-based data mining platform Anteater, which relies on Anthill, a runtime system for irregular, data intensive, iterative distributed applications, to achieve high performance. Anteater is operational and being used by the Brazilian Government to analyse government expenditure, public health and public safety policies.

Chapter 12 is entitled 'DMGA: a generic brokering-based data mining grid architecture'. It describes DMGA (Data Mining Grid Architecture), a generic brokering-based architecture for deploying data mining services in a grid. This approach presents two different composition models: horizontal composition (offering workflow capabilities) and vertical composition (increasing performance of inherently parallel data mining services). This scheme is especially significant to those services accessing a large volume of data, which can be distributed through diverse locations.

Chapter 13 is entitled 'Grid-based data mining with the environmental scenario search engine (ESSE)'. The natural environment includes elements from multiple domains such as space, terrestrial weather, oceans and terrain. The environmental modelling community has begun to develop several archives of continuous environmental representations. These archives contain a complete view of the Earth system parameters on a regular grid for a considerable period of time. This chapter describes the ESSE for data grids, which provides uniform access to heterogeneous distributed environmental data archives and allows the use of human linguistic terms while querying the data. A set of related software tools leverages the ESSE capabilities to integrate and explore environmental data in a new and seamless way.

Chapter 14 is entitled 'Data pre-processing using OGSA-DAI'. It explores the Open Grid Services Architecture – Data Access and Integration (OGSADAI) software, which is a uniform framework for providing data services to support the data mining process. It is shown how the OGSA-DAI activity framework already provides powerful functionality to support data mining, and that this can be readily extended to provide new operations for specific data mining applications. This functionality is demonstrated by two application scenarios and compares OGSA-DAI with other available data handling solutions.

References

Agrawal, R. and Shafer, J. C. (1996), 'Parallel mining of association rules', *IEEE Transactions on Knowledge and Data Engineering* **8** (6), 962–969.

Ashrafi, M. Z., Taniar, D. and Smith, K. (2004), 'ODAM: An optimized distributed association rule mining algorithm', *IEEE Distributed Systems Online* **5** (3), 2–18.

Butler, D. (1999), 'Computing 2010: from black holes to biology', *Nature* C67–C70.

Cho, V. and Wüthrich, B. (2002), 'Distributed mining of classification rules', *Knowledge and Information Systems* **4** (1), 1–30.

Clifton, C., Kantarcioglu, M., Vaidya, J., Lin, X. and Zhu, M. Y. (2002), 'Tools for privacy preserving distributed data mining', *SIGKDD Explorer Newsletter* **4** (2), 28–34.

Dubitzky, W., Granzow, M. and Berrar, D. P. (2006), *Fundamentals of Data Mining in Genomics and Proteomics*, Springer, Secaucus, NJ.

Edwards, J., Lane, M. and Nielsen, E. (2000), 'Interoperability of biodiversity databases: biodiversity information on every desktop', *Science* **289** (5488), 2312–2314.

Fayyad, U., Piatetsky-Shapiro, G. and Smyth, P. (1996), From data mining to knowledge discovery, *in* U. Fayyaad et al., ed., 'Advances in Knowledge Discovery and Data Mining', AAAI Press, pp. 1–34.

Foster, I. (2002), 'What is the Grid? A three point checklist', *Grid Today*.

Frawley, W., Piatetsky-Shapiro, G. and Matheus, C. (1992), 'Knowledge discovery in databases: An overview', *AI Magazine* 213–228.

Gomes, C. P. and Selman, B. (2005), 'Computational science: Can get satisfaction', *Nature* **435**, 751–752.

Grossman, R. L., Kamath, C., Kumar, V. and Namburu, R. R., eds (2001), *Data Mining for Scientific and Engineering Applications*, Kluwer.

Hirschman, L., Park, J. C., Tsujii, J., Wong, L. and Wu, C. H. (2002), 'Accomplishments and challenges in literature data mining for biology', *Bioinformatics* **18** (12), 1553–1561.

Kargupta, H., Huang, W., Sivakumar, K. and Johnson, E. (2001), 'Distributed clustering using collective principal component analysis', *Knowledge and Information Systems Journal* **3**, 422–448.

Kargupta, H., Kamath, C. and Chan, P. (2000), Distributed and parallel data mining: Emergence, growth, and future directions, *in* 'Advances in Distributed and Parallel Knowledge Discovery', AAAI/MIT Press, pp. 409–416.

Kesselman, C. and Foster, I. (1998), *The Grid: Blueprint for a New Computing Infrastructure*, Kaufmann.

Krauter, K., Buyya, R. and Maheswaran, M. (2002), 'A taxonomy and survey of grid resource management systems for distributed computing', *Software – Practice and Experience* **32**, 135–164.

Rajasekaran, S. (2005), 'Efficient parallel hierarchical clustering algorithms', *IEEE Transactions on Parallel and Distributed Systems* **16** (6), 497–502.

Sánchez, A., Peña, J. M., Pérez, M. S., Robles, V. and Herrero, P. (2004), Improving distributed data mining techniques by means of a grid infrastructure, *in* R. Meersman, Z. Tari and A. Corsaro, eds, 'OTM Workshops', Vol. 3292 of *Lecture Notes in Computer Science*, Springer, pp. 111–122.

Stankovski, V., May, M., Franke, J., Schuster, A., McCourt, D. and Dubitzky, W. (2004), A service-centric perspective for data mining in complex problem solving environments, *in* H. R. Arabnia and J. Ni, eds, 'Proceedings of International Conference on Parallel and Distributed Processing Techniques and Applications', Vol. 2, pp. 780–787.

Stankovski, V., Swain, M., Kravtsov, V., Niessen, T., Wegener, D., Kindermann, J. and Dubitzky, W. (2008), 'Grid-enabling data mining applications with DataMiningGrid: An architectural perspective', *Future Generation Computer Systems* **24**, 259–279.

Stankovski, V., Swain, M., Stimec, M. and Mis, N. F. (2007), Analyzing distributed medical databases on DataMiningGrid, *in* T. Jarm, P. Kramar and A. Zupanic, eds, '11th Mediterranean Conference on Medical and Biomedical Engineering and Computing', Springer, Berlin, pp. 166–169.

Talia, D. (2006), Grid-based distributed data mining systems, algorithms and services, *in* 'HPDM 2006: 9th International Workshop on High Performance and Distributed Mining', Bethesda, MD.

University of California (2007), 'SETI@Home. The Search for ExtraTerrestrial Inteligence (SETI)', http://setiathome.ssl.berkeley.edu

Wah, B. (1993), 'Report on workshop on high performance computing and communications for grand challenge applications: computer vision, speech and natural language processing, and artificial intelligence', *IEEE Transactions on Knowledge and Data Engineering* **5** (1), 138–154.

Witten, I. and Frank, E. (2000), *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*, Kaufmann.

Wright, A. (2007), *Glut: Mastering Information Through the Ages*, Henry, Washington, D.C.

Yang, Q. and Wu, X. (2006), '10 challenging problems in data mining research', *International Journal of Information Technology and Decision Making* **5**, 597–604.

Zaki, M. J., Ho, C. T. and Agrawal, R. (1999), Parallel classification for data mining on shared-memory multiprocessors, *in* 'Proceedings International Conference on Data Engineering'.