European Research Network on Foundations, Software Infrastructures and Applications
for large scale distributed, GRID and Peer-to-Peer Technologies

A Network of Excellence funded by the European Commission

Knowledge and Data Management in Grids: Notes on the State of the Art

A. Arenas
{A.E.Arenas}@rl.ac.uk
STFC Rutherford Appleton Laboratory, UK
A. Bilas, J. Luna, M. Marazakis
{bilas,jluna,maraz}@ics.forth.gr
FORTH-ICS, Greece
C. Comito, D.Talia
{ccomito, talia}@deis.unical.it
DEIS, University of Calabria, Italy
M.D. Dikaiakos
{mdd}@cs.ucy.ac.cy
University of Cyprus, Cyprus
A. Gounaris
{gounaris}@cs.man.ac.uk
University of Manchester, UK
P. Massonet, S. Naqvi
{philippe.massonet, syed.naqvi}@cetic.be
CETIC, Belgium
J. Smith, P. Watson
{jim.smith,paul.watson}@ncl.ac.uk
School of Computing Science, Newcastle University, UK
F. Stagni
{stagni}@cnaf.infn.it
INFN, Italy

CoreGRID White Paper
Number WHP-0002
May 2, 2008

Institute on Knowledge and Data Management

CoreGRID - Network of Excellence
URL: http://www.coregrid.net

CoreGRID is a Network of Excellence funded by the European Commission under the Sixth Framework Programme

Project no. FP6-004265

Knowledge and Data Management in Grids: Notes on the State of the Art

A. Arenas
{A.E.Arenas}@rl.ac.uk
STFC Rutherford Appleton Laboratory, UK
A. Bilas, J. Luna, M. Marazakis
{bilas,jluna,maraz}@ics.forth.gr
FORTH-ICS, Greece
C. Comito, D.Talia
{ccomito, talia}@deis.unical.it
DEIS, University of Calabria, Italy
M.D. Dikaiakos
{mdd}@cs.ucy.ac.cy
University of Cyprus, Cyprus
A. Gounaris
{gounaris}@cs.man.ac.uk
University of Manchester, UK
P. Massonet, S. Naqvi
{philippe.massonet, syed.naqvi}@cetic.be
CETIC, Belgium
J. Smith, P. Watson
{jim.smith,paul.watson}@ncl.ac.uk
School of Computing Science, Newcastle University, UK
F. Stagni
{stagni}@cnaf.infn.it
INFN, Italy

CoreGRID WHP-0002

May 2, 2008

Abstract

Knowledge and data management is a key topic in Grid computing. Today and in the near future, data, information, and knowledge are critical elements in the application of Grids in several sectors of our society. As Grids become pervasive in human activities, the management of data and the derivation and manipulation of knowledge play an increasingly significant role in enabling high-level applications in wired and wireless settings where Grids are used as enabling platforms, engines and tools for complex information systems.

Our view is that the Grid should be effectively exploited for deploying data-driven and knowledge-based services and applications. To support this class of applications, tools and services for data and knowledge management are vital, so that, in the coming years the Grid will be used as a platform for implementing and deploying geographically dispersed data intensive applications, distributed knowledge discovery systems and knowledge management platforms. This white paper is provided by the Institute on Knowledge and Data Management (KDM) of the CoreGRID NoE. The work originates from work done by the Institute members and its aim is to present the current state of the art with respect to Grid data and knowledge management and provide the context of the related research activities, focusing on the areas of Grid storage, information processing and data mining.

This research work is carried out under the FP6 Network of Excellence CoreGRID funded by the European Commission (Contract IST-2002-004265).

1 Introduction

Knowledge and data management is a key topic in Grid computing. As mentioned in the second NGG expert group report, ”The Grids environment must behave intelligently using knowledge-based techniques, including semantic-rich operation and interoperation” and later ”there is a need for semantically-rich knowledge-based services in both Grids Foundations Middleware and Grids Services Middleware both to improve the functionality but also to support applications on the Grids surface which require semantic and/or knowledge-based support. This need is also apparent in Grids scheduling, management, durability and dynamic reconfiguration.” The convergence of the evolving vision stated above with the service-oriented vision of significant European industry stakeholders in NESSI (Networked European Software and Services Initiative) naturally led the NGG expert group to ”define the scientific and technological requirements necessary to evolve Grids towards the wider and more ambitious vision of Service Oriented Knowledge Utilities (SOKU)” in their third and most recent report [1]. From those statements, we can figure out that, today and in the near future, data, information, and knowledge are critical elements in the application of Grids in several sectors of our society. As Grids become pervasive in human activities, the management of data and the derivation and manipulation of knowledge play an increasingly significant role in enabling high-level applications in wired and wireless settings where Grids are used as enabling platforms, engines and tools for complex information systems.

Our view is that the Grid should be effectively exploited for deploying data-driven and knowledge-based applications. To support this class of applications, tools and services for data and knowledge management are vital, so that, in the coming years the Grid will be used as a platform for implementing and deploying geographically dispersed data intensive applications, distributed knowledge discovery systems and knowledge management platforms. The aim of this white paper, which gathers contributions from the partners of the Institute on Knowledge and Data Management (KDM) of the CoreGRID NoE, is to present the current state of the art with respect to Grid data and knowledge management and provide the context of the related research activities, focusing on the areas of Grid storage, information processing and data mining.

The overall objective of the KDM Institute of the CoreGRID project is to further advance the integration of data management and knowledge discovery with Grid technologies for providing knowledge-based Grid services and systems commonly known as the Semantic Grid through a strong coordination of European researchers active in those areas. The Institute provides a collaborative network for European research teams working on: distributed storage management on Grids; knowledge techniques and tools for supporting data intensive applications; and the integration of data and computational Grids with information and knowledge Grids. The main motivation for the work performed in the KDM Institute is the pressing need to store, manage, and access increasing amounts of digital information offered to the end users as a utility functionality with established performance and dependability. The increasing dependence of our society on information and knowledge requires building infrastructures for storing and analyzing large amounts of online data and information. Moreover, the complexity of managing stored information requires new techniques for automating the management process and handling semantic metadata. Finally, new applications require high level abstractions for accessing storage (e.g. content-based search and resource discovery), as opposed to existing block- and file-level methods, querying distributed data sources and extracting knowledge from them.

The paper is organized as follows. The next section discusses the main research issues in three key areas in knowledge and data management in Grids: distributed storage management, information and knowledge management and distributed data mining and knowledge discovery.Then, section 3 reviews research activities on those areas and discusses current results. Finally, section 4 concludes the paper.

2 Overview of current approaches

This section discusses three main areas in knowledge and data management in Grids: distributed storage management, information and knowledge management and distributed data mining and knowledge discovery.

2.1 Distributed Storage Management

Existing storage systems are usually custom-built and custom-tuned to offer both scalability and good performance at high cost. Such systems are usually extended with information and data lifecycle management tools that strive to manage data continuously and based on their use. For instance, hierarchical storage management systems, move data between first and second (or third) tier storage, in an effort to optimize capacity, performance, and cost. HPSS

(High Performance Storage System) is an example of a Hierarchical Storage Management (HSM) system targeting capacity scalability and I/O performance in cluster environments. However, most HSM systems today usually target datacenter-type storage applications rather than long-distance, multi-admin-domain integration. As such, issues related to security, failures, interoperability, diverse workloads are only dealt with to a limited extend. Information Lifecycle Management (ILM) tools aim at applying management policies to data objects at all stages of a data object's life. This approach includes automatic assignment of media and devices to data objects based on the data type, data source as well as data object and media usage, performance and failure statistics. ILM solutions are currently in early stages. ILM-related products available on the market today target mainly system ownership cost, system performance optimization, and support for compliance of management operations with legal regulations, e.g. appropriate data retention and archival policies. However, they lack automatic mechanisms for data classification (and rely on manual, effort-intensive methods). Moreover, there is limited support for dealing with cross admin-domain issues, e.g. access control and secure data access.

Although such (commercial and open) tools are continuously being extended, they target existing storage systems and architectures, and mainly high-end storage systems. The work in this task focuses on how future storage architectures can both rely on commodity components as well as embed in the architecture mechanisms that will assist with data managements, rather than providing management as an add-on feature of the system. In some sense, existing solutions provide (and bind) mechanisms and policies in 3rd-party solutions, whereas research aims at separating mechanisms from policies and making mechanisms part of the system itself.

Building large installations from low-cost commodity components would make data Grids less expensive and enable them to closely track the latest technological advancements of inexpensive mass-produced storage devices. Recent advancements in commodity interconnection network technologies and the continuing reduction in disk prices present new opportunities to address these issues. Various projects [46] currently strive to address the following issues:

- Build scalable storage systems that can hold petabytes of storage in a cost-effective manner
- Make the storage infrastructure location-independent and client-agnostic in an efficient manner.
- Provide benchmarking methodologies.

Building large-scale distributed data storage systems faces the problem of storage resource virtualization incompatibility. The incompatibility results from the different levels of abstraction in the resources virtualization and the lack of a single, common framework for describing the storage services offered by the virtualized resources. Such a unified method of describing the services would make it possible to interface the different types of storage services in an effective way and would offer the starting point for building large-scale storage infrastructures. Many initiatives develop techniques for virtualizing the resources. For instance, initiatives such as Lustre, GPFS, Frangipani and Petal etc. provide a virtual volume abstraction by distributing blocks to many storage nodes. Projects, such as Storage Resource Broker aim at the virtualization of files, file repositories and similar structures. P2P systems such as Gnutella, Kazaa etc. use yet another level of abstraction of the virtualized data storage resources, providing access to files on the basis of metadata e.g. the song title or artist name. Unfortunately, these approaches remain incompatible and different scientific data Grids remain isolated. While the methodologies for building the local, metropolitan, country-wide or continent-wide storage installations are quite well-developed, connecting them with other installations remains in the area of pioneer works.

In the Grid community, there is a tendency to provide disk pool managers capable of serving large amounts of disk space distributed over several servers. Recently high performance disk storage solutions using parallel file systems are becoming increasingly important to provide reliability and high speed I/O operations needed by High Energy Physics (HEP) analysis farms. Today, Storage Area Network solutions are commonly deployed at LHC centers and parallel file systems such as GPFS and Lustre allow for high-speed native POSIX I/O operations in parallel fashion.

The need to provide grid applications with consistent and efficient wide-area access to heterogeneous storage resources drives to use the Storage Resource Manager (SRM) interface as the common standard for storage management. As of today SRM implementations exist for storage managers such as Castor, d-Cache and LCG DPM. However, such solutions manage the entire storage space allocated to them and force applications to use custom access protocols such as rfio and d-cap, sometimes penalizing performance and requiring changes in the application. New solutions, such as StoRM, designed to work over native parallel file systems or with standard POSIX file system, providing space reservation capabilities and allowing use of native high performing POSIX I/O calls for file access begin to appear.

Within the HEP community, jobs are often data intensive, and in such cases it may be necessary to move the job to where the data sites are in order to achieve better efficiency. In some applications, the volume of data is so large (in the order of hundreds of gigabytes to terabytes) that partial replication of the data is performed ahead of time to sites where the computation is expected to take place. Before moving jobs, that generate large output files, to Grid computing systems with available CPU capacity, a Grid scheduler should check for availability of the required space and then allocate it. Therefore, the possibility to manage space via the Grid becomes essential for running data-intensive applications on the Grid. Data movement and replication are also important functions to guarantee optimized access to files. To accomplish these needs, the data management services are restricted to the handling of files and one of the first needs is to identify them through some mechanism - a logical name or ID - that in turn can be used to locate and access the data, sometimes in multiple copies at several locations. In the Grid community, file replica catalogues are currently under research and the present implementation of LCG, Globus and EGEE replica catalogues are intended as a fundamental building block in Grid systems. It addresses the common need to keep track of multiple physical copies of a single logical file by maintaining a mapping from logical file names to physical locations.

Presently each VO has a Replica Catalogue that keeps a list of all the available files in the Grid. A key feature of Grid environments is sharing computing and storage resources among multiple Virtual Organizations. This process needs a comprehensive security policy and accounting framework. Concerning Storage, there are several security issues: data integrity, confidentiality, authentication, non-repudiation and authorization are only examples. Secure access to a physical file in grid context involves many steps. First, the user has to own a valid identity defined within a virtual organization. He has to be authorized to query the catalogs in order to retrieve logical file names (LFNs) and Storage URLs (SURLs). Authorization policies are metadata bound with SURLs, they live within catalogs affiliated with the VO domain. The LHC File Catalog (LFC) is one example, but other catalogs can also be used. Enforcing of permission on a physical file completes the entire security access scenario. Several security aspects belong to the virtual organization domain while others belongs to the site administrative domain. User identity definition, secure catalogs access and service authorization policies definition belong to the VO domain. Subject authentication, service authorization, and enforcing of access authorization policies on physical file belong to site domain. Every aspect must be taken into account at different levels and virtual organizations to build an effective secure context.

Another issue is that the commodity components (inexpensive disks connected to PC machines), which seem to be the future of data Grids, rarely collaborate with the high-end storage components (like specialized disk matrices, tape libraries, hierarchical systems etc.) in order to realize common or complementary services. In some cases, high-end devices supply the services for the low-end ones, but it is uncommon to have both kinds/classes of devices available to end-users. Finally, the virtualization of the resources in Grids often loses the interesting features of these resources. The virtualization is needed to include the resources into Grids, but it often hides the interesting features of the resources like high-performance which are exploitable only using the native resource interface.

In summary, there are no common methods for describing storage services realized at various levels of abstraction by different elements of the storage infrastructure, and there is no common approach to the problem of virtualizing the storage resources without losing their specific, interesting features. Without a means to effectively share storage resources and services, different data Grids remain incompatible, isolated, and ineffective. For this reason, current research is aiming at understanding requirements and characteristics of different workloads [11].

2.2 Information and Knowledge Management

This section discusses key issues in providing information needed to manage Grid resources and applications such as metadata models, semantic representation, and design of designing knowledge-oriented Grid services.

2.2.1 Information Modeling

The need to have common, platform-independent, standards for representing Grid-related information has been recognized and is currently the subject of a number of projects and working groups. These efforts have been triggered primarily by the need to enable the interoperability between large, heterogeneous infrastructures and by the emergence of Open Grid Services. One of the earliest efforts in that direction comes from the DataTAG, iVDGL, Globus, and the DataGrid projects, which collaborated to agree upon a uniform description of Grid resources. This effort resulted in the Grid Laboratory Uniform Environment (GLUE) schema, which comprises a set of information specifications for Grid resources that are expected to be discoverable and subject to monitoring [6]. GLUE represents an ontology that captures key aspects of the Grid architecture adopted by large Grid infrastructures deployed by projects such as DataGrid, CrossGrid, the Large Hadron Collider Computing Grid (LCG), and EGEE. The GLUE ontology distinguishes two classes of entities: system resources and services that give access to system resources.

Going beyond the standardization of resources and services, a number of recent efforts are trying to devise common information representations for the structure and the status of jobs running on Grids. For example, the Job Submission Description Language Workgroup of the OGF (JSDL-WG) develops the specification of the Job Submission Description Language, an XML Schema for describing computational batch jobs and their required execution environments. Another effort, led by the CIM Grid Schema Workgroup of the OGF, seeks to standardize the information that could be published by Grid schedulers about the characteristics and status of Grid jobs submitted for execution. This workgroup has adopted the Common Information Schema (CIM) of the Distributed Management Task Force's (DTMF); based on CIM v.2.8, the CIM workgroup of OGF has proposed a Job Submission Interface Model (JSIM) to describe the structure and attributes of batch jobs that run on Grid infrastructures. Finally, the need to provide basic Grid-job accounting and resource usage information in a common format is addressed by the Usage Record (UR-WG) and the Resource Usage Service (RUS-WG) workgroups of the OGF. These workgroups have started working towards the proposal of XML schemas that will describe accounting information in a general, platform-independent, way. Apart from ensuring platform independency, a challenge encountered by all the afore mentioned modeling efforts is to balance the trade-off between non-functional requirements, such as completeness, and usability and maintainability.

2.2.2 Semantic Modeling

Because of the lack of a global schema for Grid information, several researchers are investigating the application of semantic Web technologies as an alternative for bridging the gap that exists between infrastructures with incompatible information schemas. The challenges that need to be addressed in order to bridge this gap and the steps that need to be taken in order to help users retrieve information about Grid infrastructures were examined in [17].

One of the earlier efforts in these directions came from the Grid Interoperability Project (GRIP); GRIP introduces two ontologies representing the structure and attributes of UNICORE and GLUE resources, respectively. These ontologies are described in XML and fed into a tool that supports the semi-automatic association between the two ontologies; this association is used for the mapping of resource requests to hardware resources that belong to Globus and UNICORE infrastructures [9]. A similar approach for the development of an ontology-based resource matchmaker was proposed by Tangmunarunkit, Decker and Kesselman in [57]; their system comprised a matchmaker, which consisted of three components:

- an ontology component, which represents the domain model and the vocabulary for expressing resource advertisements and resource requests;
- a domain background knowledge component containing rules that express axioms, which cannot be expressed with an ontology language;
- a set of matchmaking rules, which define the matching constraints between requests and resources and are expressed in a rule-based language. An ontology editor is used for the development of three domain ontologies for resources, requests, and applicable policies; these ontologies are described with the RDF-Schema specification of W3C. Matchmaking is conducted with the help of a deductive database [57].

The definition of an upper-level domain ontology for Grid-infrastructure-related information was presented in [64] and [65]. In this paper, the authors proposed the Core Grid Ontology (CGO) to define fundamental Grid-specific concepts, and the relationships between them. The Core Grid Ontology was designed and developed based on a general model of Grid infrastructures, and described in the Web Ontology Language (OWL).

2.2.3 Data Modeling

Although the initial emphasis in Grid computing was on file-based data storage, the importance of structured data management to typical Grid applications is becoming widely recognized. In the follow are listed some proposals that have introduced data access and integration facilities for Grid-enabled database services. The European Data Grid project has developed the Spritfire infrastructure that allows a client to query a relational database over GSI-enabled HTTP(S) using an XML based protocol to formulate the query and its result [25]. The Storage Request Broker (SRB) [45] is a system that provides uniform access to datasets stored as file systems. Since the primary focus of the SRB is on file-based data, rather than on the structured data held in databases, it offers some limited capabilities for accessing data held in databases. Datasets can be held as LOBs (Large OBjects) in databases and in this case they are treated as files stored in the database, or an SQL query can be registered as an object with the SRB. The SRB provides location transparency using a metadata catalogue that allows access to datasets by logical names, or other metadata attributes.

The OGF Data Area working groups developed a suite of standards or adopted preexisting ones from different contexts, for the different aspects of data virtualization, management and integration in the GRID:

- Most early work on data grids focused principally on the provision of infrastructures for managing and providing efficient access to file-based data [30]. The emphasis was therefore not on supporting structured data collections. However, the need to provide effective metadata catalogues for file archives gave rise to the use of database technology within data grids (e.g., [16], [30]), and subsequently to the development of generic data access services, such as Spitfire [8] and OGSA-DAI [2]. This activity on metadata management, combined with the fact that an increasing number of the applications that use the Grid make extensive use of database technology, has increased awareness of the need to integrate database access interfaces with Grid middleware.

- The OGF DAIS-WG (Data Access and Integration Services Working Group) developed a service-based interface for accessing and integrating on the GRID data available in pre-existing relational and XML databases in a consistent manner.

- The OREP-WG (OGSA Replication Services Working Group) explored data replication technologies for the GRID, focusing on large files, and on replicating generic files and databases. The OREP specification regards a Replica Location Service (RLS). The RLS maintains a map of where data and replicas are stored and the availability of more copies of a given data can allow both to implement quality of service, and different level of access performance. If copies are updated, the problem of propagating updates between them has to be faced.

- An important issue of data replication is data movement: how data is moved between GRID nodes to implement replicas. The GRIDFTP-WG working group works on a reliable file transfer protocol built on the top of the usual FTP (File Transfer Protocol).Equally important is the work undergone by the GFS-WG (GRID File Systems Working Group) that is working on standard service interface(s) and architecture of a GRID file system. It is obvious that if work conducted by these last groups drives implementation of native mechanisms for efficient data movement, effective storing of files, and transparency with respect to data location, the previous services could be more easily implemented and made available. Complementarily, the GSM-WG (Grid Storage Management WG) works on the definition of standard Storage Resource Manager (SRM) functionality with a view to resolving interoperability issues between storage systems.

- The DFDL-WG (Data Format and Description Language Working Group) is working on the management of files and data streams, i.e., how describe and annotate a set of data that may constitute a stream o that their format, structure, and metadata can be exposed. Annotating or labeling data allows adding information to original data to better explain the used encoding, the meaning of data, etc. The DFDL (Data Format and Description Language) is based on XML. Although DFDL is well suited for labeling and describing scientific data it could also be used to describe relational data sources (e.g., the layout and content of a database) or a query result set (e.g., described as a labeled data stream).

- The INFOD-WG (Info Dissemination WG) works on interfaces for the description of information publishers, consumers and additional components using community-specific vocabularies. This WG also deals with extensions to the publish-subscribe paradigm paying more attention to non-functional properties to accommodate changes in the condition or state of publishers and subscribers.

- Several other OGF working groups deal with issues such as minimal service interfaces for byte streams, discovering data transport protocols available and designing a comprehensive OGSA data management architecture.

- In parallel to the development of data access and integration standards, OGSA-DAI provides high-level service-based data management utilities consisting of data access components that provide access to both relational and XML databases.In summary, OGSA-DAI web services provide a uniform way of querying, updating, transforming and delivering data that is exposed onto Grids. The current release is based upon earlier DAIS-WG specifications.

2.2.4 Grid-enabled database query systems

Structured and unstructured query systems are key components in data Grids. Polar* [47] and OGSA-DQP [5] are among the first fully-fledged generic Grid-enabled query processors, a distinguishing feature being the support for declarative (non-procedural) access to distributed data in a GRID setting.

Polar* was the first proposal to use distributed query processing in a Grid setting. Polar* evolved into the publicly available OGSA-DQP tool [5]. Polar* differs from OGSA-DQP in that it is not service-based; in Polar*, Grid middleware is accessed using a Grid-enabled version of MPI [20]. The absence of the service-based context in Polar* means that connection to external databases and computational services is much less seamless than in the OGSA setting.

OGSA-DQP is a service-based distributed query processor exposed as an OGSA Grid Service, aiming to process queries over distributed data sources obtained from multiple services on the Grid, including GDSs provided by OGSA-DAI. OGSA-DQP extends OGSA-DAI with two new services:

- a Grid Distributed Query Service (GDQS) that compiles, partitions and schedules distributed query execution plans over multiple execution nodes in the Grid.
- a Grid Query Evaluation Service (GQES). Each GQES instance is in charge of a partition of the query execution plan assigned to it by a GDQS.

Besides Polar* and OGSA-DQP, currently there is an increasingly growing interest in Grid databases. For example, SkyQuery [33] applies DQP over Grid-connected databases that contain astronomical data. The execution approach that it follows has similarities with OGSA-DQP/ Polar*, e.g., calls to WSs are regarded as typed UDFs. The main differences is that OGSA-DQP (i) supports partitioned parallelism, (ii) can employ Grid machines that may not hold data in order to evaluate parts of the query plan, and (iii) is generic with respect to the underlying databases supported and not tailored to a specific scientific scenario. In [23, 51, 52], adaptive extensions to OGSA-DQP are demonstrated, with a view to adaptively employ resources and balance workload. Apart from performance issues, OGSA-DQP considers additional non-functional requirements, such as fault-tolerance, whereas it relies on generic Grid standards for security issues.

GridDB-lite [34] is a project motivated by data-intensive scientific applications on the Grid, built upon DataCutter [10], in which the users express their retrieval tasks as SQL-like queries. However, the query is not evaluated using database technologies. Overall, GridDB-lite is benefited from the declarative manner of expressing potentially complex tasks in query processing, but develops its own execution mechanisms, thus not exploiting the full potential of a DQP system. Another project that supports database table interfaces for data processed in a workflow is GridDB [28]. As in GridDB-Lite, this feature enables declarative task expression. However, GridDB takes one step further, and employs techniques devised for adaptive query processors to prioritize partial results. Generic interfaces to Grid databases have been developed in two European projects, OGSA-DAI and European Datagrid's Spitfire [8].

2.3 Data Mining and Knowledge Discovery

The Grid can be effectively exploited for deploying data-analysis and knowledge discovery applications. It is a well-suited infrastructure for managing very large data sources and providing high-level mechanisms for extracting valuable knowledge from them. Large and distributed digital data repositories are available in several application areas from science and engineering to commerce and industry. Examples of large and distributed datasets available today include gene and protein databases, network access and intrusion data, drug features and effects data repositories, astronomy data files, and data about Web usage, content, and structure.

To analyze those data, advanced tools and services for knowledge discovery are vital. Today research teams are devising implementations of knowledge Grids in terms of the OGSA model. According to this approach, knowledge Grid services are exposed as a persistent service, using the OGSA conventions and mechanisms. Through this approach, in the near future the Grid can be used as a pervasive infrastructure for implementing and deploying geographically distributed knowledge discovery and knowledge management platforms and applications.

2.3.1 Design of Grid-aware distributed tools for data mining and knowledge discovery

Currently large amounts of data are continuously collected in distributed sites, and data mining (DM) emerged as a new discipline able to furnish tools and techniques to support knowledge extraction and decision making. This knowledge extraction process is both computationally intensive, and collaborative and distributed in nature. Thus, in the last years many distributed data mining (DDM) algorithms have been proposed [27]. DDM algorithms, when employed to devise Grid services and tools, must deal not only with distributed sources of huge datasets and multiple compute nodes, but also with distributed user community and privacy concerns. A further emerging challenge regards the updating of mined knowledge when the databases are also dynamic and evolving. For example, consider a warehouse continuously updated by streams of information [63, 31]. Grid-aware DDM services and tools must be adaptive with respect to data and platform features [36], capable of deciding whether a tightly-coupled [37] or a loosely-coupled approximate solution [49], [48] must be adopted, and optimize the use of resources [38, 41].

The opportunity of utilizing Grid-based data mining systems, algorithms and applications is interesting to users wanting to analyze data distributed across geographically dispersed heterogeneous hosts. Grid-based data mining would allow corporate companies to distribute compute-intensive data analysis among a large number of remote resources. At the same time, it can lead to new algorithms and techniques that would allow organizations to mine data where it is stored. This is in contrast to the practice of having to select data and transfer it into a centralized site for mining. Unfortunately, the number of high-level instruments to support the knowledge discovery and management in distributed environments is limited This is particularly true in Grid-based knowledge discovery, although some research and development projects and activities in this area are going to be activated mainly in Europe and USA, such as the KNOWLEDGE GRID, the Discovery Net, the AdAM, the WEKA4WS tool [59], and the DataMiningGrid citeStan08 projects. In particular, the KNOWLEDGE GRID [14] provides a middleware for knowledge discovery services targeted to a wide range of high-performance distributed applications. The system prototype under development will provide OGSA-based services for knowledge discovery in Grids on-top of WSRF; a Globus 4 based version is on going. A similar effort is done in the DataMiningGrid project [18], which is an EU STREP project that is developing tools and services for deploying data mining applications on the Grid. During the two years of activity, meetings and exchange of ideas between those two projects have been organized. Moreover, recently researchers form the CoreGRID and DataMiningGrid project are contributing to a book on Grid-based data mining systems and applications that will appear in the first months of 2008.

3 Research activities

3.1 Distributed Storage Management

3.1.1 Storage backend Gridification

Automated storage management for networked storage is a challenging problem. Recently there have been works published that try to involve intelligent approaches to tackle with some of these problems. Polus [60] aimed at mapping high level QoS goals to low level storage actions by introducing learning and reasoning capabilities. The system starts with a basic knowledge of a system administrator expressed as 'rules of thumb' and it can establish quantitative relationships between actions taken and their effect to performance by monitoring and learning. To eliminate performance flaws, the system finds an appropriate set of actions by backward reasoning in the generated knowledge base. Ergastulum [4] is aimed at supporting the configuration of storage systems. It essentially helps with reducing the search complexity of possible design decisions by utilizing a best-fit bin packaging heuristics with randomization and backtracking. It takes into consideration workload characteristics, device specifications, performance models and constraints and provides a near-optimal solution in practically acceptable time. The work introduced in [7] assists in selecting the right data-redundancy scheme for disk arrays. It is a derivative of Ergastulum and in this framework four methods were explored and evaluated for this specific problem: rule-based tagger, model-based tagger, partially and fully-adaptive solvers. While the fully-adaptive solver performs the best in most cases, it is significantly more complex than the other ones.

Given the issues in managing scalable storage backends, the idea of pervasive GRID storage services is a challenging problem. This is caused by the lack of standardized, well-understood description methods for data-related services, "detailed" enough to present the functionality of a given system to the external world. Interconnecting storage systems on one hand and offering high quality storage services to clients on the other is challenging. In fact, it is difficult to bring together the ease of using services and the possibility of exploiting the storage infrastructure while not losing their advanced features, such as predictability, data access optimization techniques, guaranteed level of service, security, confidentiality and safety of data. For instance, if one decides to provide easy access to a storage service, it may reduce user awareness of the systems capabilities and operation. The Storage Resource Broker (SRB) [3] from the San Diego Supercomputer Center has been deployed broadly during the last few years to integrate distributed heterogeneous data storage servers. It combines relational database technology with custom-built middleware software to offer unified view of disparate data sources. Even though SRB can access successfully a variety of data sources including ftp servers, database systems, file systems, and Web servers, its integration with legacy applications manifests several performance problems which makes its use inflexible. Distributed storage management involves many problems that are hard (or impossible) to formalize, involve multidimensional optimization, exponential search or ambiguous decisions. Even if there are explicit algorithms for certain problems, they quite often belong to the NP-hard class. For years many of these issues were entirely up to a system administrator who tried to give an approximate (heuristic) solution based on rules of thumb, intuition, and experience.

3.1.2 Storage System Benchmarking

Benchmarking is an essential tool in evaluating new techniques. In order to test a wide variety of features and be more realistic, benchmark programs have become more complex and benchmarking became more expensive and tedious. As a consequence, evaluating new solutions is becoming a very time consuming task. Therefore, there is an increasing interest in techniques that focus on keeping the benchmarking results accurate while reducing their execution time. Such techniques are usually based on statistical methods. Two important examples of such work in this area are: benchmark application sub-setting techniques using Principal Component Analysis (PCA) [62] or Cluster analysis [61], and execution phase analysis in order to reduce the execution time of each application [39, 50].

One of the open issues in GRID data management is to study and propose new storage solutions for the GRID platform. Benchmarking will be used in order to evaluate these solutions. As the objective for the experiments is to test the I/O system, the size of the datasets must be large, resulting in large execution times. As such, a promising approach is to apply application sub-setting and execution phase analysis to appropriate Grid benchmarks. Previously, these techniques have only been applied to the scientific SPEC benchmark suite. Finally, as the metrics and the criteria used for I/O benchmarking are significantly different from the ones used for architecture benchmarking, we expect that application of previously developed techniques to new setups is challenging.

3.2 Information and Knowledge Management

3.2.1 Semantic Grid and Ontologies

The Semantic Grid is an initiative to systematically expose semantically rich information associated with resources to build more intelligent Grid services. It is an extension of the current Grid in which information and services are given well defined and explicitly represented meaning, better enabling computers and people to work in cooperation [19], in line with the recently emerged SOKU vision.

The need for adding meaning to the Grid is motivated by that Semantic Grids not only share computational and data resources, but also explicitly share and process metadata and knowledge. Consequently, technologies that are used in the Semantic Web, such as ontology, annotation and negotiation processes, have been applied to provide a semantic metadata about Grid resources, services, data sets, etc. In [22] it is argued that Grid commuting is concerning flexible, secure, coordinated resource sharing among dynamic collections of individuals, institutions, which is closely related to the Semantic Web vision. They also induced the Semantic Web technologies into Grid computing. Significant research projects in the area of Semantic Grid are carried out by the Knowledge Grid Research Group (www.knowledgegrid.net), the OntoGrid project (www.ontogrid.net), the K-WfGrid project (www.kwfgrid.eu), and InteliGrid project (www.inteligrid.com).

One of the assumptions underlying this line of research is to use technologies of the ”the Semantic Web and Software Agent” not only to supply a general semantic-based computational network infrastructure, but also a rich, seamless collection of intelligent, knowledge-based services for enabling the management and sharing of complex resources and reasoning systems. Let us briefly address some of the contributions of those works.

- First, because of the lack of a global schema for Grid information, several researchers are investigating the application of semantic Web technologies, as an alternative for bridging the gap that exists between infrastructures with incompatible information schemas. In GRid Interoperability Project (GRIP), ontologies are introduced for interpreting/translating the information about Grid resources between two Grid information systems. One is UNICORE system where information about Grid resources is represented in NJS and IDB; another is Data Grid where information is described using GLUE schema.

- Secondly, the semantic metadata modeling is used in distributed resources allocation. In [57], the authors proposed an ontology-based resource matchmaker. The system comprises a matchmaker, which consists of three components: (i) an ontologies' component, which represents the domain model and vocabulary for expressing resource advertisements and resource requests; (ii) a domain background knowledge component containing rules that express axioms, which can not be expressed with an ontology language; (iii) a set of matchmaking rules, which define the matching constraints between requests and resources and are expressed in a rule-based language. Matchmaking is conducted with the help of a deductive database.

- Thirdly, the discovery and matching of bioinformatics workflows deployed on the Grid is the goal of the MyGrid project. MyGrid provides mechanisms for the search and discovery of pre-existing workflows based on their functionality, on the kind and format of their input/output format. Two key modules in the myGrid system are registry and the semantic find component. MyGrid registry is designed to accept and store workflow descriptions, in according to the UDDI specification. Furthermore, it supports the annotation of stored workflows with conceptual metadata. MyGrid's semantic find component is responsible for executing RDF queries upon the conceptual metadata attached to the workflow descriptions stored in myGrid registry. Also, the Grid services are attached a Service Data Element (SDE), in which the services are described in RDF metadata. To locate a suitable grid service is to query the RDF metadata [32]. Finally, some applications in bioinformatics and chemistry also used semantic way to represent the metadata of the data, which later query this kind of data [26].

- Moreover, Semantic Web technologies have been proposed as a platform for the discovery of information about software and services deployed on the Grid. An early approach comes from the ICENI project in UK, which focuses on the semantic matching between Grid services and service requests in an autonomic computing context, even when requests and resources are syntactically incompatible. To this end, the ICENI project proposed the concept of metadata space. This is an environment distinguished from the space of Grid services and resource requests. The metadata space hosts Grid-related semantic metadata, published and discovered through standard protocols. Published metadata can be classified into: (1) implementation metadata, extracted from semantic annotations of Grid services; (2) requirements metadata, which describe the semantics of resource requirements and are extracted from semantic annotations attached to resource requests; and (3) ontologies describing the inferences that can take place during matchmaking. Semantic annotations are described in the Web Ontology Language (OWL) and are attached manually to the programming interfaces of Grid-service implementation codes. The operation of the metadata space is supported by meta-services providing semantic matching and service adaptation capabilities. Service adaptation refers to the automatic adaptation of a Grid service's output to the requirements of a semantically matched but syntactically incompatible resource request. The ICENI approach is demonstrated in the case of a very simple adaptation scenario.

- It is worth pointing out that this new direction of theoretical research was accompanied by a corresponding shift in the implementation of knowledge representation systems based on very expressive DL languages. A number of Grid-related ontologies have been published by researchers for this purpose; for instance, the Earth System Grid (ESG) ontology, the Virtual Organization ontology (VOO), and the WSRF ontology (WSRF-O), which are described in Web Ontology Language (OWL) [40]. The ESG Ontology was primarily developed for adding metadata to datasets found in the Earth System Grid (ESG) portal; it is a mixed bag of concepts and includes housekeeping information as well as earth science specific information. The Virtual Organization ontology (VOO) is developed for Grid Virtual Organization management; it defines what a virtual organization is, especially but not exclusively in the context of Grid computing. VOO classes are mainly about policies and goals of a VO. The WSRF ontology defines the concepts of the Web Services Resource Framework (WSRF) and properties among them. The classes and properties of WSRF-O are referring to stateful resources and their associated Web services. The ontology especially concerns the life time and global states of WS-Resources, the coordination of Web services.

The main problem for building a general domain-ontology for Grids is that there are various proposed Grid architectures and several actual Grid infrastructures, which are comprised of thousands of entities, services, components, and applications. It is thus very difficult, if at all feasible, to develop a complete ontology that will include all aspects of Grids meeting the maintainability and usability requirements. Furthermore, different sub-domains, such as resource discovery and job scheduling, normally have different views of or interests about a Grid entity and its properties. This makes the definition of Grid entities and the relationships between them very hard. To tackle these issues, the

Core Grid Ontology (CGO) that defines fundamental specific concepts, vocabularies and relationships. One of the key goals of this work was to make this Core Grid Ontology general enough and easily extensible to be used by different architectures or middleware, so that the CGO can provide a common basis for representing Grid knowledge about resources, middleware, services, applications, and users. The Core Grid Ontology is designed to represent knowledge of Grid systems. Therefore, it should be open and extensible as there are thousands of entities, services, components, and applications of different Grid architectures and implementations. To cope with the openness and extensibility requirements, OWL was adopted to describe the concepts and classes in the Core Grid Ontology. The effort to develop an upper-level domain ontology for Grid-infrastructure-related information was presented in [64], [65]. The proposed Core Grid Ontology (CGO) focused on defining fundamental Grid-specific concepts, and the relationships between them. The Core Grid Ontology was designed and developed based on a general model of Grid infrastructures, and described in OWL. A later study showed that the expression of Grid-related metadata in OWL could be used to establish knowledge bases with Grid-related information, and that SPARQL could easily support the expression of complex user queries upon such knowledge bases, something that was not possible with state-of-the-art Grid information services [66].

3.2.2 Data Integration

Data integration is a key issue for exploiting the availability of large, heterogeneous, distributed and highly dynamic data volumes on Grids. However, there is a lack of Grid middleware schema-integration services. To the best of our knowledge, there are only few works designed to provide schema-integration in Grids [13, 12]. Based on the these reasons, a decentralized service-based data integration architecture for Grid databases referred to as Grid Data Integration System (GDIS) has been developed [15]. The proposal was motivated by the need to integrate different data models on distributed databases taking into account that in the Grid, a centralized structure for coordinating all the nodes may not be efficient because it becomes a bottleneck when a large number of nodes are involved and, most of all, it does not benefit from the dynamic and distributed nature of Grid resources.

The GDIS infrastructure exploits the middleware provided by OGSA-DQP, OGSA-DAI, and Globus Toolkit with a focus on scalability. On top of these building blocks, schema-integration services have been designed by introducing new portTypes that extend both OGSA-DAI and OGSA-DQP ones. More precisely, the GDIS system offers a decentralized wrapper/mediator-based approach to integrate data sources: it adopts a decentralized mediator approach to handle semantic heterogeneity over data sources such as to effectively exploit the available Grid resources and their dynamic allocation, whereas syntactic heterogeneity is hidden behind OGSA-DAI wrappers. In the GDIS systems when a query is posed using a nodes schema, answers should come from anywhere in the system thanks to the semantic connections among data sources. A prototype combining OGSA-DQP and GDIS is described in [21].

Among the few works designed to provide schema-integration in Grids, the most notable ones are Hyper [13] and GDMS [12]. Both systems are based on the same approach used in GDIS: building data integration services by extending the reference implementation of OGSA-DAI. The Grid Data Mediation Service (GDMS) uses a wrapper/mediator approach based on a global schema. GDMS presents heterogeneous, distributed data sources as one logical virtual data source in the form of an OGSA-DAI service. This work is essentially different from GDIS as it uses a global schema. For its part, Hyper is a framework that integrates relational data in P2P systems built on Grid infrastructures. As in other P2P integration systems, the integration is achieved without using any hierarchical structure for establishing mappings among the autonomous peers. In that framework, the authors use a simple relational language for expressing both the schemas and the mappings. By comparison, GDIS integration model follows as Hyper an approach not based on a hierarchical structure, however differently from Hyper it focuses on XML data sources and is based on schema-mappings that associate paths in different schemas.

3.3 Data Mining and Knowledge Discovery

3.3.1 Semantic Web Technologies

Semantic Web technologies have been proposed as a platform for the discovery of information about software and services deployed on the Grid. An early approach comes from the ICENI project in UK, which focuses on the semantic matching between Grid services and service requests in an autonomic computing context, even when requests and resources are syntactically incompatible [24]. To achieve this goal, the ICENI project proposed the concept of a metadata space. This is an environment distinguished from the space of Grid services and resource requests. The metadata space hosts Grid-related semantic metadata, published and discovered through standard protocols. The operation of the metadata space is supported by meta-services providing semantic matching and service adaptation capabilities. Service adaptation refers to the automatic adaptation of a Grid service's output to the requirements of a semantically matched but syntactically incompatible resource request. The ICENI approach was demonstrated in the case of a very simple adaptation scenario [24].

3.3.2 Workflows

Knowledge discovery procedures typically require the creation and management of complex, dynamic, multi-step work-flows. At each step, data from various sources can be moved, filtered, integrated and fed into a data mining tool. Examining the output results, the analyst chooses which other data sets and mining components can be integrated in the workflow or how to iterate the process to get a knowledge model. Workflows are mapped on a Grid by assigning abstract computing nodes to Grid hosts and exploiting communication facilities to ensure information/data exchange among the workflow stages. One of the aims is to consider the problem of Grid services description (semantic representation) that can be addressed through an ontology-based approach. Focus must be put on non-functional aspects of services and investigation of the usage of ontology instances (individuals) in order to enable inferences and matchmaking. One more goal, is to define Inferential Monitoring and Management algorithms that are the bases for designing Grid Agents system.

Currently, several Grid-middleware components collect, store, and publish collections of information that can be useful to Grid systems and users. These collections include:

- Information that describes the capabilities, the operation, the status, the pricing, and the usage of hardware resources available on the Grid.
- Metadata about services deployed on the Grid, such as descriptions of functionality and interface, guidelines for invocation, and policies of use.
- Metadata regarding data and software repositories deployed on the Grid, describing their organization, contents, semantics, and relevant policies of access and use.
- Job management information regarding jobs deployed on Grids: their composition in terms of software or service components, their mapping to hardware and networking resources, their cost, etc.

Information on the capability and status of Grids is typically collected and maintained by a variety of Grid-middleware sub-systems or Grid-application components, which are characterized as Grid information and/or monitoring services, although the boundaries between these two categories are not clearly defined. Grid-related information is also collected and maintained by other components: job management information is typically maintained by resource brokers, workflow engines, logging servers, etc; information about data repositories can be found in data-Grid services, such as replica catalogues, virtual file systems, and application-specific data archives.

Notably, different information sources employ diverse data models, formats, and encodings for the representation and storage of information. Some sources make their data available to third-parties (i.e., to other services, administrators or end-users) by providing support for binding, discovery, and lookup through a variety of protocols and query models. Because of the lack of a standard model or a common schema for organizing and representing information, it is difficult to establish the interoperation between different Grid platforms. Moreover, the lack of common information models and standards makes it practically impossible to achieve the automated retrieval of resources, services, software, and data, and the orchestration thereof into Grid work-flows that lead to the solution of complex problems.

A proposal for searching Grid software components has been presented in [42, 43], where the authors propose to exploit published information regarding the composition graphs (workflows) of Grid applications to rank software components. The idea is rather simple: the more a component is referred to by other applications, the more important it is considered. This concept is very close to the well-known PageRank measure used by Google to rank the pages it stores.

The discovery and matching of bioinformatics workflows deployed on the Grid is the goal of the myGrid project, which provides mechanisms for the search and discovery of pre-existing workflows based on their functionality (”task-oriented” or ”construction-time” discovery), on the kind and format of their input data (”data-driven” discovery), or on the type and format of their output data (”result-driven” discovery). To make workflows discoverable, myGrid introduces the workflow executive summary, a workflow-specific collection of metadata represented in an XML Schema.

Metadata belonging to the workflow executive summary include: (i) mandatory descriptions of the workflow's definition (e.g. its URI address, its script, its invocation interface, the types of its input and output data); (ii) optional syntactic descriptions about the format encoding of the workflow's input and output data, and (iii) optional conceptual descriptions of the workflow's characteristics. Workflow executive summary information is encoded in RDF with additional pointers to semantic descriptions described in OWL.

Two key modules in the myGrid system architecture are the registry and the semantic find component. myGrid's registry is designed to accept and store workflow descriptions, in accordance to the UDDI specification. Furthermore, it supports the annotation of stored workflows with conceptual metadata [32]. MyGrid's semantic find component is responsible for executing OWL queries upon the conceptual metadata attached to the workflow descriptions stored in myGrid's registry. Each time the semantic-find component receives notifications about metadata newly added to the registry, it updates accordingly an index with metadata descriptions. This index is used for fast replies to semantic queries. Alternatively, it can invoke a description-logic reasoner to answer semantic queries.

Also the DataMiningGrid project is working in this research area by developing Grid-enabled data mining data interfaces and services, Grid-enabled data mining workflow management tools to provide a workflow editor that facilitates the composition, execution and management of data-mining workflows in Grid computing environments.

It is natural to seek to adapt execution of a workflow in a dynamic distributed environment. A number of systems offer support for such adaptation at one or more mapping levels, for instance replanning or switching between alternate services or providers in order to minimize response time, or take account of resource failure; see [29] for a generic approach. Typically, the focus is on meeting the functional and non-functional requirements of the requestor, e.g. as expressed in an SLA. Other work seeks to address the arbitration of shared resources by such distributed applications through market mechanisms. In order to effectively contribute to, and take advantage of such an environment, it is necessary for a site to first be able to manage the concurrent workloads it accepts to best advantage. To this end, work introduced in [53, 54] investigates issues in the support required for a composite service data center. This is an obvious extension to a data center in which dynamic hosting of applications is supported not only by dynamically mapping them onto a pool of physical resources, but also by dynamically composing them from a pool of software components.

3.3.3 Grid Services for Knowledge Discovery

This research area addresses the problem of definition and composition of Grid services for implementing distributed knowledge discovery and data mining services on OGSA-based Grids. In systems such as Discovery Net and the KNOWLEDGE GRID, services for searching Grid resources, composing software and data elements, and manage the execution of the resulting data mining application on a Grid are developed. Data mining Grid services are key elements for practitioners who need to develop knowledge discovery applications that use large and remotely dispersed data sets and/or high-performance computers to get results in reasonable times and improve their organization competitiveness.

On the basis of previous experiences done in this research area, it is necessary to design and implement Gridbased distributed data mining services that leveraging the OGSA and WSRF standards will provide a distributed data mining open service middleware by which users can design higher level distributed data mining services that cover the main steps of the KDD process and offer typical distributed data mining patterns such as collective learning, ensemble learning, metalearning, and other concurrent models for composing data mining applications. Through WSRF is possible to define basic services for supporting distributed data mining tasks in Grids. Those services can address all the aspects that must be considered in data mining and in knowledge discovery processes from data selection and transport to data analysis, knowledge models representation and visualization. To do this it is necessary to define services corresponding to

- single steps that compose a KDD process such as preprocessing, filtering, and visualization;
- single data mining tasks such as classification, clustering, and rule discovery;
- distributed data mining patterns such as collective learning, parallel classification and meta-learning models;
- data mining applications including all or some of the previous tasks expressed through a multi-step scientific workflows.

This collection of data mining services can constitute an Open Service Framework for Grid-based Data Mining. This framework can support developers in designing distributed KDD processes as a composition of single services that are available over a Grid. These research topics are related to other emerging research areas, such as mobile data mining and ubiquitous knowledge discovery. Grid services features and functionality should be investigated for potential integration in such innovative scenario [58].

3.4 Data and Storage Security

3.4.1 Low-level protection: securing the Storage Elements

Storage systems have become an essential piece for the Data Grid, thus making it imperative to establish an integral and standardized security solution able to avoid common attacks on the data and metadata being managed. Grid security research has focused in specific high-level services (for example GSI and VOMS) instead of providing a systemic view able to encompass even block-level security features. Work-groups like CoreGRID's Trust and Security [67] have begun to investigate the challenges related with providing security at the Grid Storage Level. Current research into WP2 has applied to a typical use case for the Data Grid (designed by encompassing data creation, replication, discovery and retrieval) an extended framework to analyze from a systemic point of view the security guarantees provided by both, underlying infrastructure technologies and storage technologies commonly used in Data Grid sites, focusing in typical attacks that can be mounted on the data and meta-data [68]. By using the extended framework it has been possible to identify the security gaps and even redundant security features that may affect the proposed Data Grid scenario. The identified elements are being used as requirements for the security mechanisms currently proposed by FORTH/UCY's joint security research group [69], which provide and enforce the following security policies:

- Confidentiality: the Storage Elements should not have access neither to clear-text data or encryption material, even if an attacker is able to compromise some of them. This assumption is based on the vulnerabilities identified with untrusted storage sites.
- Integrity: the proposed mechanisms should enable relying parties (Grid Users) to detect unauthorized modifications to the data being retrieved from the Data Grid. Measures versus attackers destroying data on the wire should be also considered.
- Availability: untrusted storage sites give no guarantees about their availability. Destruction of data at rest is also feasible to occur if we consider that adversaries may be in full control of the storage element.
- Performance: despite it is not directly related with security, there must be a clear balance between both aspects so all the involved parties do not notice degradation on their available resources beyond the storage space they are providing to the Data Grid.

The above requirements have been used to design a cryptographic protocol that achieves its security goals based in two widely used mechanisms. The first is symmetric cryptography, which not only provides high-performance encryption (contrary to using public key cryptosystems), but also a self-contained integrity checking (using the file's hash as the encryption key) and protection versus replay attacks (using a nonce). A second mechanism, the Information Dispersal Algorithm, provides high availability and assurance for the Data Grid by means of data fragmentation. In a fragmentation scheme [44], a file f is split into n fragments, all of these are signed and distributed to n Storage Elements, one fragment per Element. The user then can reconstruct f by accessing m arbitrarily chosen fragments ($m <= n$). The fragmentation mechanism can also be used for storing long-lived data with high assurance: f is fragmented in just a few pieces (small m) and dispersed to a large number of nodes (large n) (a replication-like approach). When fragmentation is used along with encryption, data security is enhanced: An adversary colluded with the Storage Elements has to compromise and retrieve m fragments of file f, and then has to break the encryption system K. Despite of the advantages provided by encryption and fragmentation, state of the art security protocols still consider that all of the Storage Elements are homogeneous in the sense that they have the same probability of being compromised (which means they have the same vulnerabilities, hardware and even software!). Obviously this is not true in real Data Grid, because the storage nodes will most likely have different hardware and/or software features, management policies, etc. The hypothesis of a new research line into WP2 is that if subsets of Storage Elements with analogous features are clearly identified and their security level quantified, then any user of the Data Grid may be able to request storage space only into those nodes fulfilling some minimum guarantees or Quality of Security (QoSec) level. Moreover, it is worth to notice that the evaluation and further use of this QoSec into the proposed mechanism for the Data Grid would enhance the data assurance already provided by the encryption and fragmentation, but without affecting the overall performance [70].

3.4.2 Security Requirements Elaborations for Grid Data Management Systems (GDMS)

Grid data management systems [71] offer a common view of storage resources distributed over several administrative domains. The storage resources may not only be disks, but also higher-level abstractions such as files, or even file systems or databases. The scale of the stored data and its distribution over different administrative domains across geopolitical frontiers exacerbate the overall security state of the entire systems as the presence of weakest link in the security chain may simply remains invisible due to the complexity and large size of the GDMS. It is impossible to extensively identify the security requirements of such systems by making simple observations. This situation obliges the security designers to rely on formal analysis techniques [72] to precisely identify the complete set of security requirements. Typical security requirements of GDMS cover the access restriction to authorized users, the protection against transmission in an insecure environment, the availability of files.

A formal analysis of security requirements for semantic grid services to explore how these requirements can be used to derive security policy is performed under the auspices of CoreGRID WP2 [35]. This activity takes a requirements engineering approach for the elaboration of such security requirements. Requirements Engineering (RE) is concerned with the identification of goals to be achieved by the envisioned system, the refinement of such goals and their operationalization into specifications of services and constraints, and the assignment of responsibilities for the resulting requirements to agents such as humans, devices and software. Goal-oriented RE refers to the use of goals for requirements elicitation, elaboration, organization, specification, analysis negotiation, documentation and evolution [73]. More specifically the KAOS goal-oriented approach [74] is considered and applied to the security analysis of GDMS. KAOS supports a number of key features for the development of high quality security model such as: a security specific analysis (using anti-goals, modeling the attacker rationales), the ability to formalize critical parts with tool support for deep consistency and completeness checks [75], and a dynamic framework for runtime requirements enforcement [76]. This work demonstrates the benefits of such an early approach to close the gap between declarative security requirements and operational policies. It is achieved through mapping the requirements model onto high-level abstract policy by developing policy templates [77].

Security policies define the types of security measures that are used and what scope those measures have but not how those measures are designed or implemented. System security policies are derived from security requirements that specify the risks and threats that must be countered. These policies are system-specific and reflect the threat environment and the security problems assumed by system designers.

3.4.3 Security Framework for Data Curation

The growing size and scope of data generated by complex systems and the need to make this data available on a global scale led the evolution of data curation concept. Provision of adequate security to the data is as important as the preservation itself because this data is by and large generated as a result of expensive experimentations and the regeneration of data involves enormous costs. Moreover, some kind of data is irreplaceable, due to its very peculiar nature (e.g. astronomical data), and is required for analyses in the future.

A security framework for data curation should be able to preserve the data in its entirety. The preservation not only deals with the integrity issues but also the data management over time and technological change. The security framework for data curation should also assure the resilience of data by maintaining sufficient data backups. The resilience of data is crucial for the disaster recovery plan and therefore it is the sum and substance of business continuity plan (BCP). Based on a rigorous risk analysis, BCP is a comprehensive roadmap for assuring minimum mission-critical functions during a disaster and restoration of full mission-critical functions after the disaster.

4 Conclusions

Developing knowledge-based applications that exploit Grid features to achieve high performance and high availability is an area of active research in Europe, USA, and Asia, and by several research teams. At the same time, industry is very active in exploring and adopting possible solutions. However, there is still a need for a common model and framework that can integrate research results. This global scenario must coexist with the provisioning of a variety of models, architectures, prototypes, and services that are increasingly developed by the Grid community and offer different technological solutions to current problems faced by Grid applications.

The approach we use and advocate is a vertical approach that encompasses all layers involved with knowledge management in Grids: storage management at the systems-level, information and knowledge management, and knowledge discovery. We believe that future research activities should focus on core technologies needed for implementing the SOKU model, including knowledge management, data mining and knowledge discovery, and data management, also taking into account their security and trust requirements.

References

[1] NGG3 Expert Group. Future for European Grids: Grids and Service Oriented Knowledge Utilities. Bruxelles, Belgium, January 2006.

[2] M. Antonioletti, M. Atkinson, R. Baxter, A. Borley, N.P. Chue Hong, B. Collins, N. Hardman, A.C. Hulme, A. Knox, M. Jackson, A. Krause, S. Laws, J. Magowan, N.W. Paton, D. Pearson, T. Sugden, P. Watson, and M. Westhead. The design and implementation of database services in OGSA-DAI. Concurrency and Computation: Practice and Experience, 17:357-376, 2005.

[3] S. V. Anastasiadis, S. Gadde, J. S. Chase. Scale and Performance in Semantic Storage Management of Data Grids, International Journal on Digital Libraries. 2005.

[4] E. Anderson, M. Kallahalla, S. Spence, R. Swaminathan, and Q. Wang. Ergastulum: quickly finding near-optimal storage system designs, submitted to ACM Transactions on Computer Systems, HP Laboratories SSP technical report HPL-SSP-2001-05 (2002).

[5] M. N. Alpdemir, A. Mukherjee, N.W. Paton, P.Watson, A. A. Fernandes, A. Gounaris, and J. Smith. Service-based distributed querying on the grid. In the Proceedings of the First International Conference on Service Oriented Computing, pages 467-482. Springer, 15-18 December 2003.

[6] S. Andreozzi, M. Sgaravatto and C. Vistoli. Sharing a conceptual model of Grid resources and services. In Proceedings of the 2003 Conference for Computing in High Energy and Nuclear Physics, March 2003.

[7] E. Anderson, R. Swaminathan, A. C. Veitch, G. A. Alvarez, John Wilkes: Selecting RAID Levels for Disk Arrays Proceedings of the FAST ’02 Conference on File and Storage Technologies, January 28-30, 2002, Monterey, California, USA. USENIX 2002.

[8] W. H. Bell, D. Bosio, W. Hoschek, P. Kunszt, G. McCance, and M. Silander. Project Spitfire - Towards Grid Web Service Databases. In Global Grid Forum 5, 2002.

[9] J. Brooke, D. Fellows, K. Garwood and C. Coble. Semantic matching of Grid Resource Descriptions. In M.D. Dikaiakos, editor, Grid Computing. Second European Across Grids Conference, AxGrids 2004, Nicosia, Cyprus, January 2004, Revised Papers, volume 3165 of Lecture Notes in Computer Science, pages 240-249. Springer, 2004.

[10] M. Beynon, R. Ferreira, T. Kurc, A. Sussman, and J. Saltz. Datacutter: Middleware for filtering very large scientific datasets on archival storage systems. In IEEE Symposium on Mass Storage Systems, pages 119-134, 2000.

[11] M. Brzezniak, N., M. Flouris, R. Lachaize, and A. Bilas. An Analysis of GRID Storage Element Architectures: High-end Fiber-Channel vs. Emerging Cluster-based Networked Storage. In Proceedings of the CoreGRID Workshop on Grid Middleware. Dresden, Germany. Held with the International Supercomputing Conference (ISC07), June 2007.

[12] A. Bpw, M. Ta. Novel mediator architectures for grid information systems. Journal for Future Generation Computer Systems - Grid Computing: Theory, Methods and Applications. 21:107-114, 2005.

[13] D. Calvanese, G.D. Giacomo, M. Lenzerini, R. Rosati, G. Vetere. Hyper: A framework for peer-to-peer data integration on grids. In: Proc. of the Int. Conference on Semantics of a Networked World: Semantics for Grid Databases (ICSNW 2004) Volume 3226 of Lecture Notes in Computer Science, 144-157, 2004.

[14] M. Cannataro, D. Talia, The Knowledge Grid, Communications of the ACM, vol. 46, no. 1, pp. 89-93, 2003.

[15] C. Comito, D. Talia, GDIS: A service-based architecture for data integration on grids. In: Proceedings of the Workshop on Grid Computing and Its Application to Data Analysis 88-98(2004)

[16] P. Dinda and B. Plale. A Unified Relational Approach to Grid Information Services. Technical Report GWD-GIS-012-1, Global Grid Forum, 2001.

[17] M. D. Dikaiakos, Y. Ioannidis, R. Sakellariou. ”Search Engines for the Grid: A Research Agenda.” In Grid Computing. First European AcrossGrids Conference, Santiago de Compostela, Spain, February 2003, Revised Papers, Lecture Notes in Computer Science series, vol. 2970, pages 49-58, vol. 2970, Springer, 2004

[18] http://www.dataminninggrid.org.

[19] D. De Roure, N. R. Jennings, and N. R. Shadbolt, ”The Semantic Grid: A future e-Science infrastructure,” in Grid Computing - Making the Global Infrastructure a Reality, F. Berman, G. Fox, and A. J. G. Hey, Eds.: John Wiley and Sons Ltd., 2003, pp. 437-470.

[20] I. Foster and N. Karonis. A Grid-Enabled MPI: Message Passing in Heterogeneous Distributed Computing Systems. In Proc. Supercomputing. IEEE Press, 1998.

[21] A. Gounaris, C. Comito, R. Sakellariou, D. Talia: ”A Service-Oriented System to Support Data Integration on Data Grids”. Seventh IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2007), 627-635.

[22] C. Goble and D. De Roure, ”The Semantic Web and Grid Computing,” in Real World Semantic Web Applications, vol. 92, Frontiers in Artificial Intelligence and Applications, V. Kashyap and L. Shklar, Eds.: IOS Press, 2002.

[23] A. Gounaris, J. Smith, N. Paton, R. Sakellariou, A. Fernandes, P. Watson: ”Practical Adaptation to Changing Resources in Grid Query Processing”. Proc. of the 22nd IEEE International Conference on Data Engineering (ICDE), 2006.

[24] J. Hau, W. Leee and S. Newhouse. Autonomic Service Adaptation in ICENI using Ontological Annotation. In Proceedings of the 4th International Workshop on Grid Computing, pages 10-17. IEEE Computer Society, April 2003.

[25] Hoschek W., McCance G., Grid Enabled Relational Database Middleware. In Global Grid Forum, 2001. Frascati, Italy.

[26] G. Hughes, H. Mills, D. D. Roure, J. G. Frey, L. Moreau, m. schraefel, G. Smith, and E. Zaluska, ”The Semantic Smart Laboratory: A system for supporting the chemical eScientist,” Org. Biomol. Chem., 2004.

[27] H. Kargupta, P. Chan (Eds.). Advances in Distributed and Parallel Knowledge Discovery. AAAI Press/The MIT Press, 2000.

[28] D. Liu, M. Franklin, and D. Parekh. GridDB: a relational interface for the grid. In Proceedings of the 2003 ACM SIGMOD International Conference on Management of Data 2003, pages 660-660. ACM Press, 2003.

[29] K. Lee, R. Sakellariou, N. W. Paton and A. A. A. Fernandes, ”Workflow Adaptation as an Autonomic Computing Problem”, In WORKS 2007, HPDC, Monterey Bay, California, USA, 2007.

[30] R. W. Moore, C. Baru, R. Marciano, A. Rajasekar, and M. Wan. DataIntensive Computing. In I. Foster and C. Kesselman, editors, The Grid: Blueprint for a New Computing Infrastructure, chapter 5, pages 105-129. Morgan Kaufmann, 1999.

[31] F. Manola, E. Miller, eds., RDF Primer, W3C Recommendation, 2002.

[32] S. Miles, J. Papay, K. Decker, and L. Moreau. Towards a Protocol for the Attachement of Semantic Descriptions to Grid Services. In M. D. Dikaiakos, editor, Grid Computing. Second European Across Grids Conference, AxGrids 2004, Nicosia, Cyprus, January 2004, Revised Papers, volume 3165 of Lecture Notes in Computer Science, pages 240-249, Springer, 2004.

[33] T. Malik, A. Szalay, T. and A. Thakar. SkyQuery: AWeb Service Approach to Federate Databases. In Proc. CIDR, 2003.

[34] S. Narayanan, U. Catalyrek, , T. Kurc, X. Zhang, and. Saltz. Applying database support for large scale data driven science in distributed environemnts. In Proc. of the 4thWorkshop on Grid Computing, Grid’03, 2003.

[35] S. Naqvi, O. Poitou, P. Massonet, A.E. Arenas. Security Requirements Analysis for Large-Scale Distributed File Systems. In Proc. of the 1st CoreGRID Workshop on Grid Middleware, 2006.

[36] S. Orlando, P. Palmerini, R. Perego, F. Silvestri. ”Adaptive and Resource-Aware Mining of Frequent Sets”. IEEE ICDM, pp. 338-345, 2002.

[37] S. Orlando, P. Palmerini, R. Perego, F. Silvestri. ”An Efficient Parallel and Distributed Algorithm for Counting Frequent Sets”. Int. Conf. VECPAR 2002, 2002. (Appeared as Selected Paper in LNCS 2565, Springer, pp. 197-204).

[38] S. Orlando, P. Palmerini, R. Perego, F. Silvestri. ”Scheduling High Performance Data Mining Tasks on a Data Grid Environment”. Euro-Par 2002, LNCS 2400, Springer, pp. 375-384, 2002.

[39] E. Perelman, et al. Using SimPoint for Accurate and Efficient Simulation, Proceedings of the ACM SIGMETRICS the International Conference on Measurement and Modeling of Computer Systems, June 2003.

[40] H. Patel-Schneider, I. Horrocks, eds., OWL Web Ontology Language Semantics and Abstract Syntax, W3C Recommendation 10 Feb 2004.

[41] P. Palmerini, S. Orlando, R. Perego. ”On Statistical Properties of Transactional Datasets”. ACM - SAC 2004, Special track on Data Mining, pp. 515-519, 2004.

[42] D. Puppin, F. Silvestri, S. Orlando, D. Laforenza. ”Toward a Search Engine for Software Components”. Proc. of the 2004 IEEE/WIC/ACM Int. Conf. on Web Intelligence (WI 2004). Bejing, China. September 20-24, 2004, pp. 495-498.

[43] D. Puppin, F. Silvestri, D. Laforenza, S. Orlando. ”Toward GRIDLE: A Way to Build Grid Applications Searching Through an Ecosystem of Components”. In Jose C. Cunha and Omer F. Rana eds., Grid Computing: Software Environments and Tools, Springer Verlag, 2006, ISBN: 1-85233-998-52004.

[44] M. O. Rabin. ”Efficient dispersal of information for security, load balancing, and fault tolerance”. Journal of ACM, 36(2):335348, 1989.

[45] A. Rajasekar, M. Wan, R. Moore, MySRB & SRB - Components of a Data Grid. in 11th International Symposium on High Performance Distributed Computing (HPDC-11), Edinburgh, 2002.

[46] S. Rhea, P. Eaton, D. Geels, H. Weatherspoon, B. Zhao, J. Kubiatowicz. Pond: The OceanStore Prototype. Usenix FAST Conference, San Francisco, CA, March 2003.

[47] J. Smith, A. Gounaris, P. Watson, N. W. Paton, A. A. A. Fernandes, and R. Sakellariou.Distributed Query Processing on the Grid. International Journal of High Performance Computing Applications, Volume 17, Issue 4, pp 353 - 368

[48] C. Silvestri, S. Orlando. ”Approximate Mining of Frequent Patterns on Streams”. Int. Journal of Intelligent Data Analysis, Volume 11, Number 1, pp.49-73, 2007, IOS Press, ISSN 1088-467X.

[49] C. Silvestri, S. Orlando. ”Distributed Approximate Mining of Frequent Patterns”. ACM-SAC 2005, Special track on Data Mining, 2005.

[50] T. Sherwood, E. Perelman, G. Hamerly, S. Sair, and B. Calder, Discovering and Exploiting Program Phases, IEEE Micro: Micro’s Top Picks from Computer Architecture Conferences, December 2003.

[51] J. Smith and P. Watson, Fault Tolerance in Distributed Query Processing, Proc. IDEAS’05, Montreal, Canada, 2005.

[52] J. Smith and P. Watson, ”Failure Recovery Alternatives In Grid Based Distributed Query Processing: A Case Study”, in ”Knowledge and Data Management in Grids”, eds D. Talia, A. Bilas and M. D. Dikaiakos, CoreGRID series number 3, Springer, 2007, isbn: 0-387-37830-8.

[53] J.Smith and P.Watson, ”Experiments Towards Adaptation of Concurrent Workflows”, Proc. ECOWS 2007, Halle, Germany, 2007.

[54] J.Smith and P.Watson, ”Wrapper Based Adaptive Management of Concurrent Workflow Executions”, Poster at: GRID 2007, Austin, Texas, USA, 2007.

[55] V.Stankovski, M. Swain, V. Kravtsov, T. Niessen, D. Wegener, J. Kindermann and W. Dubitzky ”Grid-enabling data mining applications with DataMiningGrid: An architectural perspective” Future Generation Computer Systems, 24(4), pp. 259-279, April 2008.

[56] P. Trancoso, C. Adamou, and H. Vandierendonck, ”Reducing TPC-H Benchamarking Time”, in Proc. of the 10th Panhellenic Conference on Informatics (PCI 2005), Springer LNCS, pp. 641-650, November 2005.

[57] H. Tangmunarunkit, S. Decker, and C. Kesselman, ”Ontology-based Resource Matching,” presented at 2nd International Semantic Web Conference (ISWC2003), Florida, USA, 2003.

[58] D. Talia, P. Trunfio, ”Mobile Data Mining on Small Devices Through Web Services”. In: Mobile Intelligence: When Computational Intelligence Meets Mobile Paradigm, L. Yang, A. Waluyo, J. Ma, L. Tan, B. Srinivasan (Editors), John Wiley & Sons, 2007.

[59] D. Talia, P. Trunfio, O. Verta, ”Weka4WS: a WSRF-enabled Weka Toolkit for Distributed Data Mining on Grids”. Proc. of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases (PKDD 2005), Porto, Portugal, October 2005, LNAI vol. 3721, pp. 309-320, Springer-Verlag, 2005.

[60] S. Uttamchandani, K. Voruganti, S. Srinivasan, J. Palmer, D. Pease: Polus: Growing Storage QoS Management Beyond a ”Four-year Old Kid”. FAST ’04 Conference on File and Storage Technologies, March 31 - April 2, 2004, Grand Hyatt Hotel, San Francisco, California, USA. USENIX 2004.

[61] H. Vandierendonck, K. De Bosschere, Experiments with Subsetting Benchmark Suites. Proceedings of the Seventh Annual IEEE International Workshop on Workload Characterization. 2004.

[62] H. Vandierendonck, K. De Bosschere, Many Benchmarks Stress the Same Bottlenecks. Workshop on Computer Architecture Evaluation Using Commercial Workloads. 2004.

[63] O. Wolfson. ”Moving Objects Information Management: The Database Challenge”. 5th Workshop on Next Generation Information Technologies and Systems (NGITS’2002), Israel, 2002.

[64] W. Xing, M. D. Dikaiakos, and R. Sakellariou ”A Core Grid Ontology for the Semantic Grid.”. In Proceedings of the 6th IEEE International Symposium on Cluster Computing and the Grid (CCGrid 2006), Singapore, May 2006, pages 178-184, IEEE Computer Society.

[65] W. Xing, M. D. Dikaiakos, R. Sakellariou, Design and Development of a Core Grid Ontology. 3rd Semantic Grid Workshop, GGF16, Athens, Greece, February, 2006.

[66] W. Xing, M. D. Dikaiakos, R. Sakellariou, Towards a Grid Information Knowledge Base, Proceedings of the CoreGRID Workshop on Grid Middleware in Conjunction with Euro-Par Conference, LNCS 4375, Dresden, Germany, August 2006.

[67] A. Arenas et. al. ”CoreGRID Deliverable: D.IA.16 Update of the Survey Material on Trust and Security”. September, 2007.

[68] J. Luna et. al. ”An Analysis of Security Services in Grid Storage Systems”. In Proceedings of the CoreGRID Workshop on Grid Middleware 2007. Dresden, Germany. June, 2007.

[69] J. Luna, M. Flouris, M. Marazakis, A. Bilas, M. Dikaiakos, H. Gjermundrod, T.Kyprianou. ”A Data-centric Security Analysis of ICGrid”. Proceedings of the CoreGRID Integrated Research in Grid Computing, pp. 165-176, 2008.

[70] J. Luna et. al. ”Providing Security to the Desktop Data Grid”. Submitted to the PCGRID’08 Workshop. October, 2007.

[71] A. Jagatheesan, R. Moore, P. Watson, N. Paton, Grid Data Management Systems & Services, 29th International Conference on Very Large Databases (VLDB 2003), September 11, Berlin, Germany, 2003.

[72] C. Heitmeyer, Applying 'Practical' Formal Methods to the Specification and Analysis of Security Properties, in Proc. Information Assurance in Computer Networks (MMM-ACNS 2001), LCNS 2052, Springer-Verlag, St. Petersburg, Russia, May 21-23, 2001.

[73] A. van Lamsweerde, E. Letier, Handling Obstacles in Goal-Oriented Requirements Engineering IEEE Transactions on Software Engineering, Special Issue on Exception Handling, Vol. 26 No. 10, 978-1005, 2000.

[74] A. Dardenne, A. van Lamsweerde, and S. Fickas, Goal-Directed Requirements Acquisition, Science of Computer Programming Vol. 20, North Holland, pp. 3-50, 1993.

[75] C. Ponsard, P. Massonet, J.F. Molderez, A. Rifaut, A. van Lamsweerde, T. Van Hung, Early Verification and Validation of Mission Critical Systems, in Journal of Formal Methods in System Design (Springer), vol 30, nr 3, June 2007.

[76] M.S. Feather, S. Fickas, A. van Lamsweerde, C. Ponsard, Reconciling System Requirements and Runtime Behaviour Proceedings de IWSSD'98 - 9th International Workshop on Software Specification and Design, IEEE, Isobe, Japan, April 1998.

[77] S. Naqvi, C. Ponsard, P. Massonet, A. Arenas, 'Security Requirements Elaborations for Grid Data Management Systems', International Journal of System of Systems Engineering (IJSSE), 2008.