

Teaching Material Related to the Book's Content

Table of contents

1 Chapter 1 - An Introduction to the Grid.....	2
2 Chapter 2 - OGSA and WSRF.....	2
3 Chapter 3 - The Semantic Grid and Autonomic Computing.....	3
4 Chapter 4 - Grid Security.....	3
5 Chapter 5 - Grid Monitoring.....	4
6 Chapter 6 - Job Management and User Interaction.....	5
7 Chapter 7 - Workflow Management for the Grid.....	5
8 Chapter 8 - Grid Portals.....	5
9 Chapter 9 - Applications.....	6

1. Chapter 1 - An Introduction to the Grid

The Grid concepts and technologies are all very new, first expressed by Foster and Kesselman in 1998 [1]. Before this, efforts to orchestrate wide-area distributed resources were known as metacomputing [2]. Even so, whichever date we use to identify when efforts in this area started, compared to general distributed computing, the Grid is a very new discipline and its exact focus and the core components that make up its infrastructure are still being investigated and still being determined. Generally it can be said that the Grid has evolved from a carefully configured infrastructure that supported a limited number of grand challenge applications executing on high performance hardware between a number of US national centres [3], to what we are aiming at today, which can be seen as a seamless and dynamic virtual environment. In this book we take a step-by-step approach to describe the middleware components that make up this virtual environment which is now called the Grid.

1. Foster and Carl Kesselman (eds), *The Grid: Blueprint for a New Computing Infrastructure*, 1st edition, Morgan Kaufmann Publishers, San Francisco, USA (1 November 1998), ISBN: 1558604758.
2. L. Smarr and C. Catlett, *Metacomputing*, *Communication of the ACM*, 35, 1992, pp. 44-52, ISSN: 0001-0782.
3. D. De Roure, M.A. Baker, N. Jennings and N. Shadbolt, *The Evolution of the Grid*, in *Grid Computing: Making the Global Infrastructure a Reality*, Fran Berman, Anthony J.G. Hey and Geoffrey Fox (eds), pp. 65-100, John Wiley and Sons, Chichester, England (8 April 2003), ISBN: 0470853190.

2. Chapter 2 - OGSA and WSRF

The Grid couples disparate and distributed heterogeneous software and hardware resources to provide a uniform computing environment for scientists and engineers to solve data and computation-intensive problems. Because of the heterogeneity of the Grid, the Global Grid Forum (GGF) [1] has been organized as a working body for designing standards for the Grid.

Globus [2] toolkit 2 (GT2) and earlier versions have been widely used for building pre-OGSA oriented Grid systems. However, Grid systems based on Globus at this stage are heterogeneous in nature because these Grid systems are developed with heterogeneous protocols, which make it hard for them to interoperate. With the parallel development of GT2, Web services [3], as promoted by IBM, Microsoft, Sun Microsystems and many other Information Technology (IT) players, are emerging as a promising computing platform for building distributed business related applications in a heterogeneous environment.

At the GGF4 meeting in February 2002, the Globus team and IBM proposed a first OGSA specification [4] to merge the efforts of Globus and Web services. OGSA was proposed as the architecture for building the next generation of service-oriented Grid systems in a standard way. A working group in GGF has also been organized, called OGSA-WG [5], to work on the design of the OGSA specification. This was an important step and represented a significant milestone in the evolution of the Grid. OGSA is based on Web services, which use standard protocols such as XML and HTTP for building service-oriented distributed systems. OGSA introduces the concept of Grid services, which are Web services with some extensions to meet the specific need of the Grid.

OGSA defines various aspects related to Grid services, e.g. what kind of features a Grid service should have and the life cycle management of Grid services. However, OGSA merely defines what interfaces are needed, but does not specify how these interfaces should be implemented. Another working group in GGF has been organized, called OGSi-WG [6], to work on OGSi, a technical specification for the

implementation of Grid services as proposed in the OGSA specification in the context of Web services. Based on the OGSi technical specification, Globus toolkit version 3 (GT3) has been implemented and released as a toolkit for building OGSA compliant service-oriented Grid systems.

1. GGF, <http://www.ggf.org>
2. Globus, <http://www.globus.org>
3. Web services, <http://www.w3.org/2002/ws/>
4. Foster, I., Kesselman, C., Nick, J. and Tuecke, S. (June 2002). The Physiology of the Grid: An Open Grid Services Architecture for Distributed Systems Integration, <http://www.globus.org/research/papers/ogsa.pdf>
5. OGSA-WG, <http://www.Gridforum.org/ogsa-wg>
6. OGSi-WG, <http://www.Gridforum.org/ogsi-wg/>

3. Chapter 3 - The Semantic Grid and Autonomic Computing

The concept of the Semantic Grid [1] is evolved through the concurrent development of the Semantic Web and the Grid. The Semantic Web can be defined as "an extension of the current Web in which information is given well-defined meaning, better enabling computers and people to work in cooperation" [2]. The aim of the Semantic Web is to augment unstructured Web content so that it may be machine-interpretable information to improve the potential capabilities of Web applications. The aim of the Semantic Grid is to explore the use of Semantic Web technologies to enrich the Grid with semantics. It is the application of Semantic Web technologies to the Grid. Metadata and ontologies play a critical role in the development of the Semantic Web. Metadata can be viewed as data that is used to describe data. Data can be annotated with metadata to specify its origin or its history. In the Semantic Grid, for example, Grid services can be annotated with metadata associated with an ontology for automatic service discovery. An ontology is a specification of a conceptualization [3].

The Grid is complex in nature because it tries to couple distributed and heterogeneous resources such as data, computers, operating systems, database systems, applications and special devices, which may run across multiple virtual organizations to provide a uniform platform for technical computing. The complexity of managing a large computing system, such as the Grid, has led researchers to consider management techniques that are based on strategies that have evolved in biological systems to deal with complexity, heterogeneity and uncertainty. The approach is referred to autonomic computing [4]. An autonomic computing system is one that has the capabilities of being self-healing, selfconfiguring, self-optimizing and self-protecting.

1. The Semantic Grid, <http://www.semanticgrid.org>.
2. Lee, T.B., Hendler, J. and Lassila, O. (2001). The Semantic Web. *Scientific American*, 284(5): 34-43.
3. Gruber, T.R. (1993). A Translation Approach to Portable Ontology Specifications. *Knowledge Acquisition*, 5(2): 199-220. Academic Press.
4. Kephart, J.O. and Chess, D.M. (2003). The Vision of Autonomic Computing. *IEEE Computer*, 36(1): 41-50.

4. Chapter 4 - Grid Security

In general, IT security is concerned with ensuring that critical information and the associated infrastructures are not compromised or put at risk by external agents. Here, the external agent might be anyone that is not authorized to access the aforementioned critical information or infrastructure. The critical infrastructure we are referring to is that which supports banking and financial institutions, information and communication systems, energy, transportation and other vital human services. The

Grid is increasingly being taken up and used by all sectors of business, industry, academia and the government as the middleware infrastructure of choice. This means that Grid security is a vital aspect of its overall architecture if it is to be used for critical infrastructures.

A number of observations have been made on critical infrastructures [1]. It is clear that in today's world they are highly interdependent, both physically and in their reliance on national information infrastructure. Most critical infrastructures are largely owned by the private sector, where there tends to be a reluctance to invest in long-term and high-risk security-related technologies. Ongoing changes to business patterns are reducing the level of tolerance to errors in these infrastructures. However, there is insufficient awareness of critical infrastructure issues. The growth of IT and the Internet can therefore have major implications for the economic and the military security of the world.

IT infrastructures are changing at a staggering rate. Their scale and complexity are becoming ever greater in scope and functional sophistication. Boundaries between computer systems are becoming indistinct; increasingly every device is networked, so the infrastructure is becoming a heterogeneous sea of components with a blurred human/device boundary. There is continuous and incremental development and deployment; systems evolve by adding new features and greater functionality at an unremitting pace. These systems are becoming capable of dynamic self-configuration and adaptation, where systems respond to changing circumstances and conditions of their environment [2]. Increasingly there are multiple innovative types of networked architectures and strategies for sharing resources. This obviously leaves gaps for a multiplicity of fault types and openings for malicious faults, as well as attacks from internal and external parties.

The actors that may want to compromise critical information or infrastructures are many and varied. They include those that pose national security threats, such as information warriors or agents involved in national intelligence. Alternatively the actors could be terrorists involved in industrial espionage or organized crime and who pose a shared threat to a country. Or the threats could just be local and come from institutional or recreational hackers intent on thrill, challenge or prestige.

1. Critical Infrastructure Information Security Act, Bob, Bennett, http://bennett.senate.gov/bennettinthesenate/speeches/2001Sep25_Crit_Infrast_Inf_Sec.htm
2. eLiza Project, <http://www.ibm.com/servers/autonomic/>.

5. Chapter 5 - Grid Monitoring

A Grid environment is potentially a complex globally distributed system that involves large sets of diverse, geographically distributed components used for a number of applications. The components discussed here include all the software and hardware services and resources needed by applications.

The diversity of these components and their large number of users render them vulnerable to faults, failure and excessive loads. Suitable mechanisms are needed to monitor the components, and their use, hopefully detecting conditions that may lead to bottlenecks, faults or failures. Grid monitoring is a critical facet for providing a robust, reliable and efficient environment.

The goal of Grid monitoring is to measure and publish the state of resources at a particular point in time. To be effective, monitoring must be "end-to-end", meaning that all components in an environment must be monitored. This includes software (e.g., applications, services, processes and operating systems), host hardware (e.g., CPUs, disks, memory and sensors) and networks (e.g., routers, switches, bandwidth and latency). Monitoring data is needed to understand performance, identify problems and to tune a system for better overall performance. Fault detection and recovery mechanisms need the monitoring data to help determine if parts of an environment are not functioning correctly, and whether to restart a component or redirect service requests elsewhere. A service that can forecast performance might use monitoring data as input for a prediction model, which could in turn be

used by a scheduler to determine which components to use.

6. Chapter 6 - Job Management and User Interaction

The Grid is emerging as a new paradigm for solving problems in science, engineering, industry and commerce. Increasing numbers of applications are utilizing the Grid infrastructure to meet their computational, storage and other needs. A single site can simply no longer meet all the resource needs of today's demanding applications, and using distributed resources can bring many benefits to application users. The deployment of Grid systems involves the efficient management of heterogeneous, geographically distributed and dynamically available resources. However, the effectiveness of a Grid environment is largely dependent on the effectiveness and efficiency of its schedulers, which act as localized resource brokers.

Grid scheduling is defined as the process of mapping Grid jobs to resources over multiple administrative domains. A Grid job can be split into many small tasks. The scheduler has the responsibility of selecting resources and scheduling jobs in such a way that the user and application requirements are met, in terms of overall execution time (throughput) and cost of the resources utilized.

7. Chapter 7 - Workflow Management for the Grid

As we have discussed in Chapter 2, OGSA is becoming the de facto standard for building service-oriented Grid systems. OGSA defines Grid services as Web services with additional features and attributes. A Web service itself is a software component with a specific WSDL interface that completely describes the service and how to interact with it. Information about a particular Web service can be published in a registry, such as UDDI. A client interacts with the registry to search and discover the services available. SOAP is a protocol for message exchanging between a client and a service. Apart from that, an important feature of Web services is service composition in which a compound service can be composed from other services.

The main goal of OGSA is to make compliant Grid services interoperable. Grid services can be used in the following two ways: independent pre-OGSA Grid services and interdependent OGSA compliant Grid services.

8. Chapter 8 - Grid Portals

The Grid couples geographically dispersed and distributed heterogeneous resources to provide various services to users. We can consider two main types of Grid users: system developers and end users. System developers are those who build Grid systems using middleware packages such as Globus [1], UNICORE [2] or Condor [3]. The end users are the scientists and engineers who use the Grid to solve their domain-specific problems perhaps via a portal. A Grid portal is a Web-based gateway that provides seamless access to a variety of backend resources. In general, a Grid portal provides end users with a customized view of software and hardware resources specific to their particular problem domain. It also provides a single point of access to Grid-based resources that they have been authorized to use. This will allow scientists or engineers to focus on their problem area by making the Grid a transparent extension of their desktop computing environment. Grid portals currently in use include XCAT Science Portal [4], Mississippi Computational Web Portal [5], NPACI Hotpage [6], JiPANG [7], The DSG Portal [8], Gateway [9], Grappa [10], and ASC Grid Portal [11].

In this chapter, we will study Grid portals; the technologies they employ and the mechanisms that they use. So far, Grid portal development can be broadly classified into two generations. First generation Grid portals are tightly coupled with Grid middleware such as Globus, mainly Globus toolkit version 2.x (GT2) written in C. The second-generation of Grid portals are those that are starting to emerge and make use of technologies such as portlets to provide more customizable solutions.

1. Globus <http://www.globus.org>
2. UNICORE, <http://www.unicore.de>
3. Condor, <http://www.cs.wisc.edu/condor/>
4. Krishnan, S., Bramley, R., Gannon, D., Govindaraju, M., Indurkar, R., Slominski, A., Temko, B., Alameda, E., Alkire, R., Drews, T. and Webb, E. 2001. The XCAT Science Portal. Proceedings of Super Computing 2001 (SC'01), Denver, Colorado, USA. CS Press
5. Haupt, T., Bangalore, P. and Henley, G. 2001. A Computational Web Portal for the Distributed Marine Environment Forecast System. Proceedings of the 9th International Conference on High-Performance Computing and Networking (HPCN), June 2001, Amsterdam, Netherlands. Lecture Notes in Computer Science, Springer-Verlag
6. The Hotpage Portal, <https://hotpage.npaci.edu/>
7. Suzumura, T., Matsuoka, S. and Nakada, H. 2001. A Jini-based Computing Portal System. Proceedings of Super Computing 2001 (SC'01), Denver, Colorado, USA. CS Press
8. The DSG Portal, <https://portals.dsg.port.ac.uk/>
9. Haupt, T., Akarsu, E., Fox, G. and Youn, C. 2000. The Gateway System: Uniform Web based Access to Remote Resources. Concurrency - Practice and Experience, 12(8): 629-642
10. Grappa, <http://iutatlas.physics.indiana.edu/grappa/>
11. Allen, G., Daues, G., Foster, I., Laszewski, G., Novotny, J., Russell, M., Seidel, E. and Shalf, J. 2001. The Astrophysics Simulation Collaboratory Portal: A Science Portal Enabling Community Software Development. Proceedings of the 10th IEEE International Symposium on High Performance Distributed Computing 2001 (HPDC'01), San Francisco, California, USA. CS Press

9. Chapter 9 - Applications

In the previous chapters, we have discussed and explored core Grid technologies, such as security, OGSA/WSRF, portals, monitoring, resource management and scheduling, and workflow. We have also reviewed some projects related to each area of these core technologies. Basically the projects reviewed in the previous chapters are focused on the Grid infrastructure, not applications. In this chapter, we present some representative Grid applications that have applied or are applying the core technologies discussed earlier and describe their make-up and how they are being used to solve real-life problems.