Web-based access to the grid using the Grid Resource Broker Portal

Giovanni Aloisio, Massimo Cafaro

ISUFI High Performance Computing Center

Department of Innovation Engineering

University of Lecce, Italy

{giovanni.aloisio, massimo.cafaro}@unile.it

Abstract This paper describes the Grid Resource Broker (GRB) portal, a web gateway to computational grids in use at the University of Lecce. The portal allows trusted users seamless access to computational resources and grid services, providing a friendly computing environment that takes advantage of the underlying Globus Toolkit middleware, enhancing its basic services and capabilities.

Keywords: computational grids, web portals.

1. Introduction

Portal is a relatively recent internet neologism that usually refers to a web site providing specific contents and related services to a well known group of people having common interests, working in the same field etc. Often, portals are also equipped with powerful

search engines to ease information retrieval. A grid portal target is the scientific community, and its scope is grid computing.

Grid computing [1] ties together heterogeneous computing/software resources, storage and hardware instrumentation like sensors, electronic microscopes etc. This is not meant to be a random collection of resources geographically spread around the world and across several institutions, but a coordinated set of resources working in tight cooperation to address cutting-edge computational science problems.

While portals and grid portals share a number of ideas and concepts, they also differ in that a grid portal backend is a new complex environment, the grid. Portals do not need to manage thousands of computing resources at once to provide their services, but a grid portal may have to. A number of challenging problems arise in the grid setting, and middleware capable of dealing with them is under development.

Currently, the Globus Toolkit [2] is the *de facto* standard for grid computing because of its wide acceptance and deployment worldwide, even though several alternatives do exist, like Legion [3] and Unicore [4].

Globus is a layered architecture that addresses grid security [5], remote access and control providing support for PKI single sign-on authentication/authorization, an information rich environment [6] based on the LDAP protocol and a standardized interface to heterogeneous computing resources [7]. However, the complexity of this middleware hinders the majority of scientists (that are not computer scientists) from doing their useful work.

The Grid Resource Broker is designed to bridge the gap between those scientists scared by the complexity of the Globus Toolkit and the grid. However, GRB is much more

than a user friendly web GUI. We leverage existing functionalities in the Globus Toolkit, providing enhanced Globus services.

The paper is organized as follows. Section 2 presents an overview of the services currently provided by GRB. In Section 3 we discuss GRB architecture and its relationship with the underlying grid middleware, the Globus Toolkit. Details related to GRB implementation are given in Section 4. We review GRB services from a grid perspective in Section 5 and provide details about the project status and current work in progress in section 6. Finally, we conclude the paper in Section 7.

2. GRB Services

GRB services can be classified according to the following main categories:

- User's profile management
- Information services
- Job submission
- Job tracking
- High Performance File Transfers

The following subsections provide a detailed description of GRB services belonging to each of the aforementioned categories.

2.1 User's profile management

We do firmly believe that among the most important features of a grid portal there should be the possibility to allow the users to create and handle their personal grids on the fly, that is, the computational resources managed by the portal on behalf of a user should not constitute a static, predefined set. Instead, a user should be able to add or remove dynamically resources to her grid.

GRB services related to user's profile management fulfil this requirement. Profiles store information related to the computational resources where the users have access to. Currently, the information stored include for each machine its hostname, the user's shell pathname and the node-hour cost.

The hostname is required in order to contact Globus daemons listening on the remote machine, the shell pathname allows us to take advantage of the user's environment to submit jobs with greater flexibility (e.g., no need to provide full pathnames to executables). Finally, the node-hour cost is taken into account when brokering is requested by the users to find a suitable collection of computational resources.

We plan to extend user's profiles with other useful information that could be exploited to access remote machines easily, for brokering purposes or to notify users about grid events. GRB provides services to add, remove or edit the information related to a specified computational resource and to view the contents of a user's profile that represents the user's own grid at a specific point in time.

2.2 Information services

Computational grids exhibit heterogeneity at different levels, ranging from hardware / software to security and access policies. To cope with heterogeneity, an information rich

environment is strongly required. Information services are also crucial for event based grid programming, because grid enabled applications can only adapt to dynamic changes in the grid with timely access to required information.

GRB leverages existing Globus information services (called Metadata Directory Service – MDS) and provides white pages and yellow pages lookups. A Grid Resource Information Service (GRIS) is available on each computing resource where the Globus Toolkit is installed. Both static and dynamic information strictly related to that resource can be published and retrieved easily using the LDAP protocol.

Moreover, GRIS servers belonging to the same institution or to a virtual organization can register their contents hierarchically to a Grid Index Information Service (GIIS). A single query to the GIIS server can then be used to find machines that match some user's specified criteria like amount of memory, cpu speed, operating system etc.

GRB allows the users to query arbitrary GRIS and GIIS servers. In the near future, we will also allow authorized users to publish information to these LDAP server. This feature can be exploited by advanced grid enabled applications to retrieve dynamically information provided on the fly by others information providers.

2.3 Job submission

GRB is a general purpose grid portal. To be as general as possible, we provide several job submission services, recognizing different user's needs. A very simple tool allows the users to submit interactive jobs. By interactive we mean that the job output is sent immediately back from the remote computing resource to the user's client browser.

We provide automatic staging of both the executable and input file(s) if the executable and / or the input is stored on different machines. This service provides a convenient way to do simple things like browsing a file's contents, listing a user's directory etc.

Users can also take advantage of our batch submission tool for standard production tasks. Our batch service can provide automatic job scheduling and takes care of executable, input and output staging. Scheduling is based on user directives and also takes into account cost factors.

A new, improved Metadata Directory Service (MDS 2) is going to be released from the Globus team, so that GRB scheduling algorithms will be designed and developed to exploit the information available. The new MDS will contain considerably more information than the current one, moreover some fixes will provide correct information related to parallel machines queues (currently sometimes this information is wrong).

We also allow execution of graphical applications, automatically redirecting the remote X-Window display. This feature allows the users to steer an application, and requires only setting the appropriate permission on the machine where the client browser runs before submitting the job.

GRB can be used for High Throughput Computing. We developed a simple parameter sweep tool that provides a convenient means to submit to a collection of computing resources the same executable using each time a different input file.

Staging of executable, input and output is automated, and the user can select the machine pool or let the system chose automatically a suitable ensemble of resources matching her criteria. We are also going to develop interfaces to the Condor-G system [8] that can submit parameter sweep jobs to machines running Globus.

2.4 Job tracking

Batch jobs submitted using Globus can be tracked easily. A batch job submission returns a unique job identifier using an URL based scheme that can be used to enquiry about the job status. If the job is submitted to a remote batch scheduler it is in the *pending* state while sitting in the queue waiting to be executed. The job is *active* when actually executing and may become *suspended* due to pre-emption mechanisms. In case of normal completion the job status is *done*, otherwise the job is *failed*.

GRB also tracks file transfers, so that executable / input / output staging are monitored too. The system restarts automatically in case of failure (a predefined number of times) file transfers and job submission, providing a certain degree of fault tolerance. Users are allowed to check a specified job or all of the jobs previously submitted, including parameter sweep jobs.

2.5 High Performance File Transfers

GRB leverage a new protocol, called GridFTP [9], that has been devised for high throughput, reliable and secure data transfers. The protocol extends the well known FTP protocol and provides support for

- GSS support for authentication;
- Parallel data transfer;
- Thirdy-party transfer;
- Partial file transfer;

- File Transfer restart;
- Automatic negotiation of TCP buffer sizes.

Globus provides two different libraries that can be used to develop GridFTP based file transfers, a control library and a client library. We decided to develop our own GridFTP client library because the Globus one currently lacks some features like recursive put / get and thirdy-party transfer. Our library is also easier to use and do not require any knowledge of GridFTP protocol.

GRB provides support for transferring a single file or a directory using multiple parallel streams. Thus, users can move their data using a secure and efficient file transfer mechanism that results in reduced communication time if compared with standard FTP or Globus GASS [10].

There are no restrictions on what systems/sites could be served by GRB, because of the way user profiles are handled. As a matter of fact, GRB can be accessed regardless of system/geographical location and Grid Resources can be added/removed dynamically.

3 GRB architecture

GRB builds on the Globus Toolkit. We decided to utilize Globus as grid middleware because it is currently accepted as a *de facto* standard and has gained wide acceptance worldwide. Indeed, it is deployed at several academic institutions and organizations, and the Globus team provide support for both developers and users. GRB services are mainly based on the following Globus services:

1. Grid Resource Information Service (GRIS);
2. Grid Index Information Service (GIIS);
3. Globus Resource Allocation and Management (GRAM);
4. Grid Security Infrastructure (GSI);
5. GridFTP (GSI-FTP).

Authentication and authorization exploit the MyProxy [11][12] package. Users create restricted credentials and store them as a proxy on our Myproxy server machine. GRB retrieves the proxy using the user's supplied one time pass phrase and grant or deny access to the system checking the distinguished name of the proxy's issuer.

The user's pass phrase is sent using HTTPS to avoid network sniffing, and cookies are used to establish and maintain session information. The user's proxy is then used for single sign-on to the computational grid.

GRB is a three-tier system, as depicted in Figure 1. The first tier includes the user's client machine and a MyProxy server, the second tier is the web server hosting the GRB itself, and the third tier is the user's computational grid.

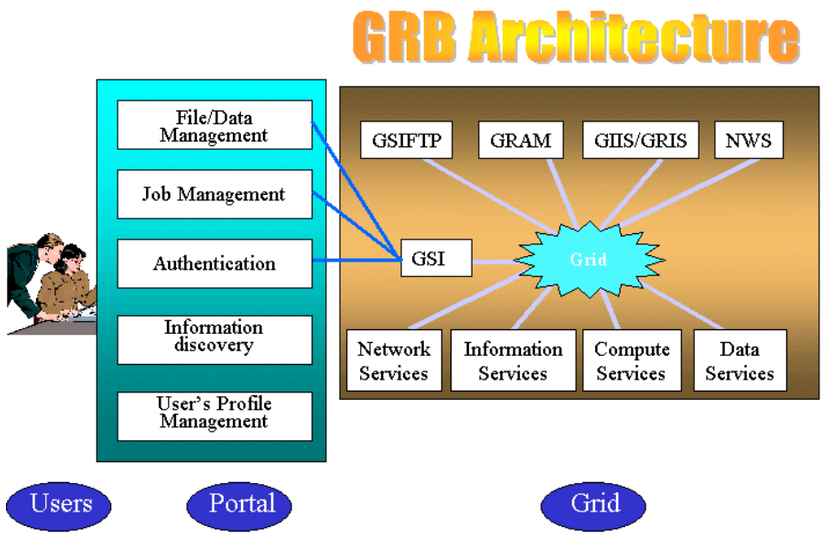

Figure 1. Grid Resource Broker architecture

4 GRB implementation

Currently, GRB is implemented using the following commodity technologies and software:

- Globus Toolkit
- MyProxy
- Apache web server
- SSL
- Openldap
- C language and UNIX system calls
- HTML
- CGI
- HTTPS
- COOKIES
- EXPECT

- LDAP

The authors are not aware of others C based grid portals. Commodity Grid Kits (CoG kits) [13] providing Perl, Python and Java bindings to Globus APIs are available and a number of portals were built with them. However, using the C language we get better performance and interface cleanly with the Globus Toolkit using its native APIs.

GRB services are implemented as CGIs and a scheduler application written using two libraries we have developed. One of the library allows developers to take advantage of the new GridFTP protocols, hiding Globus details and providing enhanced high performance file transfers. Another library provides support for proxy management, job submission / tracking, and for querying information servers [14].

5 Grid services

In this section we review the following GRB services from a grid perspective:

- Security;
- Information Servers;
- Scheduling;
- Data management.

Security in the grid environment is achieved because we exploit HTTPS and MyProxy for user authentication / authorization, and the Globus GSI infrastructure at all levels, ranging from job submission to job checking and file transfers. Currently, only the

access to information servers is made binding anonymously to GIIS / GRIS servers. The next release of MDS, now in alpha stage, will provide support for GSI access.

GRB allows the users to search for information in arbitrary GIIS and GRIS servers. Moreover, GRB can also exploit the information found (matching user's criteria) to schedule a job submission. We believe that access to information services is crucial in grid environments.

GRB allows interactive, batch and parameter sweep job submission. Interactive jobs do not require scheduling. Batch jobs can be scheduled automatically on behalf of the user. Automatic scheduling of jobs is based on user's criteria related to computational resources (like amount of memory needed for the computation, cpu speed, cpu load etc) and on the node-hour cost. Parameter sweep scheduling currently exploits a round-robin scheduling strategy.

A File transfer service is provided using GSI-FTP. GRB allows the users to transfer single files or a directory using multiple parallel streams.

6 Project status

All of the features described in this paper are currently available to selected people, mainly for testing purposes. Indeed, GRB is built using alpha stage Globus software, that is, software not yet officially released for production.

The Globus team is kindly releasing to selected institutions their latest software, so that computational grids with GSI-FTP servers deployed on computational resources are – as of this writing – not much popular. As soon as the Globus team will release its latest official version of the Globus Toolkit and GSI-FTP software, GRB will switch again to

production level. GRB includes some features deemed essential for a grid portal. However, other functionalities are missing. Work in progress is going to provide:

- Support for complex, dataflow tasks;
- Support for customized user's environment and session recording;
- Logging portal activities;
- Integration with NWS [15];
- Integration with Globus Replica Catalog [16];
- XML schemas for GRB information;
- Discovery service;
- Services descriptions using UDDI [17];
- Interfaces descriptions using WSDL [18];
- Interoperability among portals;
- Better HCI.

Support for dataflow tasks is required by some users, and completes the portal submission capabilities. The ability to customize the user's environment and to access past sessions simplifies and makes easy the interaction with the system. Beside logging and recording individual user activities, we also plan to log global activities for internal usage, maintaining statistics, and so on.

Integration with NWS will allow new scheduling algorithms that can take into account network latency and bandwidth among peers, while the Globus Replica Catalog will provide faster access to replicated data, using the closest cached copy.

The Grid Computing Environments Working Group established by the Global Grid Forum [19], is working to provide grid portals interoperability. One way to achieve this is

the use of XML schemas and the availability of a *discovery service* where portals can register their services. Interfaces can be descripted using WSDL, while services themselves are best described using UDDI. Both are XML dialects. Finally, we plan to improve HCI, making GRB even more easier to use.

7 Conclusions

The paper has presented an overview of the Grid Resource Broker, an advanced grid portal that provides enhanced Globus services. We have discussed its three-tier architecture, the implementation and services available, highlighting salient features. GRB is a web gateway to the grid that leverages the Globus Toolkit, however the users do not need to learn how to use Globus, nor they are required to rewrite their legacy applications.

GRB provides scientists with a comfortable access to the grid and can be thought of as a very general grid tool with a friendly GUI. Moreover, C libraries for developers will be released to simplify grid enabled application development, and additional services will extend current capabilities. GRB has been successfully demonstrated several times, at SuperComputing 2000 in Dallas, at the NPACI All Hands Meeting 2001 in San Diego, at the Grid developer Workshop in Berkeley and at the first EuroGlobus held in Lecce, Italy.

References

1. Foster I, Kesselman C. *The Grid. Blueprint for a new computing infrastructure*. Morgan Kaufmann: San Francisco 1999.

2. Foster I., Kesselman C. 1997. Globus: A Metacomputing Infrastructure Toolkit. *Intl J. Supercomputer Applications*, **11** (2):115-128.

3. Grimshaw A.S., Wulf W.A., French J.C., Weaver A.C, Reynolds P.F. 1997. The Legion Vision of a Worldwide Virtual Computer. *CACM* 40

4. Almond J., Snelling D. 1999. UNICORE: uniform access to supercomputing as an element of electronic commerce. *FGCS* **15** (5-6):539-548.

5. Foster I., Kesselman C., Tsudik G., Tuecke S., 1998. A Security Architecture for Computational Grids. *Proc. 5th ACM Conference on Computer and Communications Security Conference*, pp. 83-92.

6. Czajkowski K., Fitzgerald S., Foster I., Kesselman C. 2001. Grid Information Services for Distributed Resource Sharing. *Procedings of the Tenth IEEE International Symposium on High-Performance Distributed Computing (HPDC-10), IEEE Press.*

7. Foster I., Kesselman C., Lee C., Lindell R., Nahrstedt K., Roy A. 1999. A Distributed Resource Management Architecture that Supports Advance Reservations and Co-Allocation. *Intl Workshop on Quality of Service.*.

8. Frey J., Tannenbaum T., Foster I., Livny M., Tuecke S. 2001. Condor-G: A Computation Management Agent for Multi-Institutional Grids. *Proceedings of the Tenth IEEE Symposium on High Performance Distributed Computing (HPDC10)* IEEE Press.

9. GridFTP white paper, http://www.globus.org/datagrid/deliverables/C2WPdraft3.pdf

10. Bester J., Foster I., Kesselman C., Tedesco J., Tuecke S. 1999. GASS: A Data Movement and Access Service for Wide Area Computing Systems. *Sixth Workshop on I/O in Parallel and Distributed Systems.*

11. Novotny J., Tuecke S., Welch V. 2001. An Online Credential Repository for the Grid: MyProxy. *Proceedings of the Tenth International Symposium on High Performance Distributed Computing (HPDC-10)*, IEEE Press.

12. MyProxy package, http://dast.nlanr.net/Projects/MyProxy/

13. Laszewski G., Foster I:, Gawor J., Smith W., Tuecke S. 2000. CoG Kits: A Bridge between Commodity Distributed Computing and High-Performance Grids. *ACM 2000 Java Grande Conference.*

14. Aloisio G., Cafaro M., Blasi E., Depaolis L., Epicoco I. 2001. The GRB Library: Grid Programming with Globus in C. *Proceedings of HPCN Europe 2001, Amsterdam, Netherlands, Lecture Notes in Computer Science, Springer-Verlag*, 2110: 133-140.

15. Wolski R., Spring N.T:, Hayes J 1999. The Network Weather Service: A Distributed Resource Performance Forecasting Service for Metacomputing. *Journal of Future Generation Computing Systems, North Hollond.*

16. The Globus Replica Catalog, http://www.globus.org/datagrid/replica-catalog.html

17. Universal description, Discovery and Integration, http://www.uddi.org

18. Web Service Definition Language, http://www.w3.org/TR/wsdl

19. Global Grid Forum, http://www.gridforum.org