D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas
9-19-03

Grid Portals: *A Scientist's Access Point for Grid Services (DRAFT 1)*

GGF DOCUMENT SUBMISSION CHECKLIST (include as front page of submission)	
	COMPLETED (X) - Date
1. Author name(s), institution(s), and contact information	X 9-19-03
2. Date (original and, where applicable, latest revision date)	X 9-19-03
3. Title, table of contents, clearly numbered sections	X 9-19-03
4. Security Considerations section	X 9-19-03
5. GGF Copyright statement inserted (See below)	X 9-19-03
6. GGF Intellectual Property statement inserted. (See below) **NOTE that authors should read the statement.**	X 9-19-03
7. Document format - The GGF document format to be used for both GWD's and GFD's is available in MSWord, RTF, and PDF formats. (note that font type is not part of the requirement, however authors should avoid font sizes smaller than 10pt).	X 9-19-03

D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas
9-19-03

Grid Portals: *A Scientist's Access Point for Grid Services (DRAFT 1)*

Status of This Memo

This memo provides information to the Grid community interested in portal access to Grid systems. It does not define any standards or technical recommendations. Distribution is unlimited.

Copyright Notice

Abstract

This document describes an architecture for web portals designed to support access to Grid Services. It draws upon best practices derived from the collective experience of many projects, a GGF Workshop on Grid Portals, commercial standards, and the proposed Open Grid Service Architecture. This draft is presented for discussion and revision at the GGF9 GCE-RG session.

Contents

1. Introduction

A Grid Portal is problem solving environment that allows scientists to program, access and execute distributed "Grid" applications from a conventional Web Browser and other desktop tools. In one class of Grid portals, which we will call science portals, scientific domain knowledge and tools are presented to the user in terms of the application science and not in terms of complex distributed computing protocols. The goal is to allow the scientist to focus completely on the science problem at hand by making the Grid a transparent extension of the their desktop computing environment. Another class of Grid portals, which we call user portals, provides the user with direct views of Grid resources and services. These portals require the user to have a greater understanding of Grid technology than application-oriented science portals, but they often form the foundation of these more advanced tools. In "early days" of grid portal work, the efforts were exploratory. Many excellent portals for specific disciplines were built and they attracted

D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas

substantial user communities. They proved that the science portal concept was valid. Unfortunately, these systems were stovepipes that were not easily ported to new environments or applications. There was no architecture upon which a new portal could be built and there was no component model that allowed a portal to expand and add new capabilities. In addition, Grid technology has undergone a rapid evolution. In its current form, the Grid architecture is being defined using Web Services technology and the Open Grid Services Architecture is rapidly taking shape.

In response to these challenges and changes, the Grid Computing Environment Research group of GGF has organized a workshop on portal architectures at GGF7 and a second workshop was organized by the UK Edinburgh e-Science Center in July of 2003. In this note we summarize key findings of these two workshops. The primary outcome of this work is an architecture and framework for building reliable, extensible Grid Science Portals.

2. Background

A Grid portal is a user's point of access to a Grid system [3,5,6,9]. It provides an environment where the user can access Grid resources and services, execute and monitor Grid applications, and collaborate with other users. A number of Grid portals exist or are currently under construction including the NEESgrid Portal [14], the Alliance Portal [22], the GENIUS Portal for the European Data Grid [19], and the IeSE (Integrated e-Science Environment) [20]. Also, several large collaborative efforts are underway to build portal construction toolkits based on OGSA Grid services, including the GridSphere project [23] in Germany and the NSF NMI Open Grid Computing Environment project in the US. A workshop on March 06, 2003 in Tokyo, Japan at GGF7 was held where many of these groups came together for the first time.

At the workshop Roberto Barbera and Andrea Rodolico described GENIUS and the Engine Frame architecture. Geoffrey Fox presented an overview of the basic architecture of many portals and described the online knowledge center. He described the basic three-tier architecture and the concept of "user facing web services" which is described in this document in greater detail below. Tomasz Haupt described an OGSA inspired architecture of a Grid Portal with implications for Enterprise Computational Services as implemented in the Mississippi Web Portal. Michael Russel presented the first look at the GridLab GridSphere architecture which has just been released. Charles Severence presented the University of Michigan Chef portal and the application of this technology to the NEESGrid. He also described some initial reactions to the proposed Web Services for Remote Portlets standard. Satoshi Shirasuna described application integration and data management issues for portals and Toyotaro Suzumura described a process for the automatic generation of user interfaces in the Gridspeed project. Mary Thomas summarized the general area of portal services and the work at the University of Texas.

D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas

In July of 2003, a second workshop was held at the e-Science Center in Edinburgh which brought many of the same researchers back together and also introduced many more of the European community Grid portal projects. In the course of these meetings and the intervening research period, a standard architecture for Grid portals has emerged. In the paragraphs that follow we will outline this architecture and present research directions that are implied by its design. We begin with a simple requirements analysis for Grid portals.

2.1 What do users want from a Grid portal?

The primary requirements for a Grid portal system from a user' perspective involves access to Grid services. These include

- **Security services.** Users will log onto a portal using a web browser and they will authenticate by means of a user-id and password. While better technologies exist for authentication, this one is what user want and trust. More secure systems, such as smart cards are possible, but unlikely to be deployed anytime soon. Once the user is authenticated to the portal server, it is the job of the portal server to act as the user's proxy in most grid interactions. Consequently, the portal server must obtain a proxy certificate that it can use on behalf of the user. The standard approach is to have the user submit a proxy to a "MyProxy" server. The portal server then retrieves the proxy from the MyProxy server and holds it for the duration of the user's session. This process of making the user manage a key pair and submit proxy certificates to a MyProxy server is extremely unpopular with users. The general opinion is that a separate trusted service should manage credentials and key pairs for the user. We expect that such a service will be proposed soon. A second area of security is the management of group identity. Grids are often organized around the concept of a Virtual Organization which consists of a group of collaborating individuals and institutions. A VO may own or control resources and it may be temporary in existence. Services, such as the Community Authorization Service, will be trusted repositories of VO identity. When these services are available portal users will need tools to access and manage their VO identity. For example, depending upon which VO a user belongs to, different tools and resources may be visible.
- **Remote File management.** Tools to access to file metadata directories and remote file archives is a central requirement for the portal. Simple tools for Grid FTP are essential, but may files are likely to be managed by a virtual data system, where data is cataloged, curated and staged by back-end grid services. The portal user will need access to these systems and tools, such as Chimera [26] will need to be included.
- **Remote Job management.** The ability to submit jobs to the Grid for execution and monitoring is a classic requirement for portals. Users with allocation on specific resources want to be able to see job queues on those resources and consult scheduling assistants. They need to be able to keep track of job execution and understand when things fail by reading logs. The most ambitious Grid applications are those whose execution is defined by complex workflows. It is not the job of the portal to execute,

D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas

or even manage these, and many workflow languages and systems will be proposed and used. However, as workflow programming and execution systems are deployed as services, the portal must be able to provide access to them.

- **Access to information services.** Access to directories and index tools is an essential role of the portal. Each user should have a private, persistent store of references to important information they have stored on the Grid. For example, a searchable index of personal annotations about grid-based experiments and metadata references should be always available to the user. Messaging and notification is central to managing Grid applications and it should be possible to subscribe to notification services and have these notifications stored and accessible to the user through some sort of searchable index.
- **Application interfaces.** The key to science portals is being able to hide Grid details behind useful application interfaces. The user needs to be able to launch, configure and control remote applications in the same way he or she uses desktop applications. The portal must provide a way to easily integrate new application interfaces and expose them to the user and hide the details of Grid middleware.
- **Access to collaboration.** For any VO, it must be possible to share resources. This includes the ability to use real-time collaboration tools as well as asynchronous collaboration. Shared video, Access Grid controls and display, and shared power point is an essential component.

3. Portal Architecture: A Portlet Approach to Grid Services

The Grid architecture is being defined in terms of a collection of remote Grid services. Within the portal working groups a style has emerged for interacting with Grid and Web services as back-ends to the portal. The portal architecture that has developed is based on the idea that the portal server is a container for "user facing" Grid service clients which are designed according to the portlet model. A portlet is a server component that controls a small, user-configured window in a "pane" on the user's web browser. There is now a standard specification for portlets [21] and the portlet model is supported by most of the major vendors of portal software including IBM, SUN, BEA and Oracle. It is also a standard supported by the Apache open source software foundation in the Jetspeed project, the Michigan Chef server and the GridLab Gridsphere effort.

In these systems, the portal server provides the user with a set of tabbed frames which each contain one or more portlets. A portlet is a portal server component that provides a basic service rendered in a user-configurable window in a portal pane (Figure 1). The user is free to configure his or her portal frames in any way they wish. The advantage of this architecture is that it is a flexible component-based approach to building portals. By supplying the user with a basic pallet of portlets that provide the front-end, interactive part of Grid services and applications, the user can decide how to organize the environment to suit them best. Portlet containers like Chef provide the user with a host of day-to-day productivity tools such as personal or group calendars, news-readers, message boards and information services.

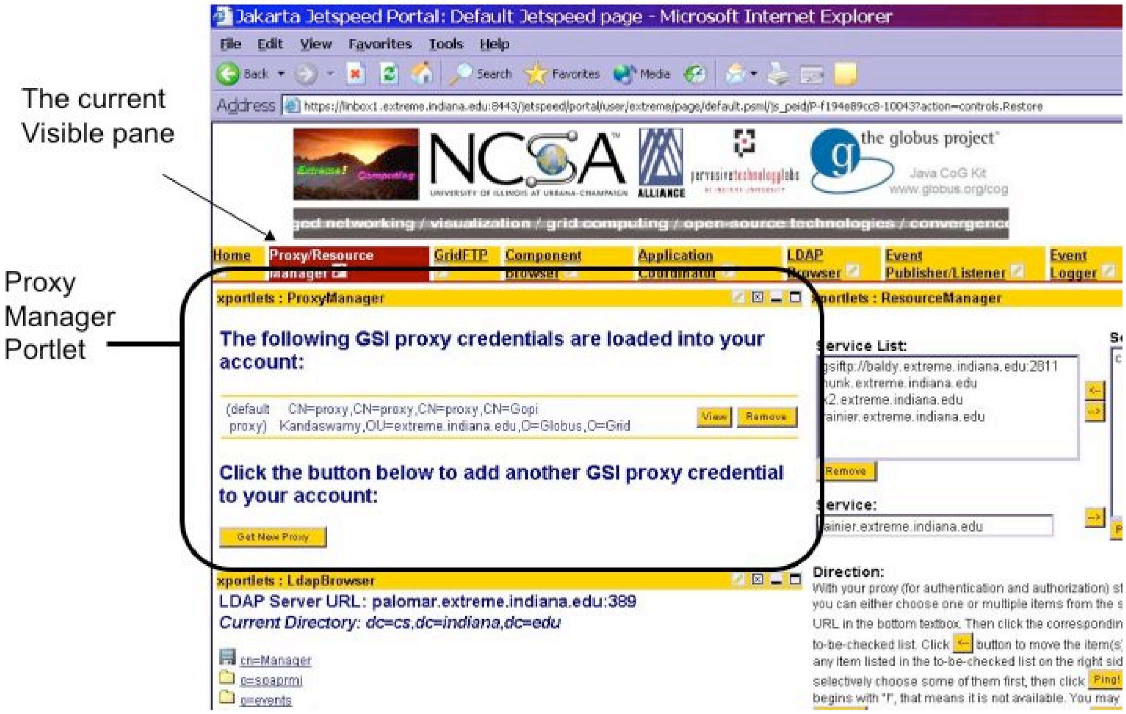


Figure 1. A Proxy Manger portlet is configured into a frame that groups together two other portlets under the heading Proxy Resource Manager.

The advantage of this architecture for Grid Portals is that it provides a natural way to incorporate “user-facing” Grid services into the portal environment. That is, if a Grid service has been designed to be used through an interactive client, that client can be written as a portlet and provided to the user in the portal. When the user logs into the portal, it should bring up his or her current “Grid Context”. By this we mean the users persistent directory of objects, annotations, references and notification logs. It should also provide the set of tools (portlets) he or she uses to access remote services, configured into groups of tabbed panes and organized the way he or she last left them (Figure 2.). Access to the users VO group identity within a collaboration gives instant a access to group news and communications.

There are many advantage of portlet architecture. Each Grid service can be associated with a unique portlet. It is very easy to add new services and many different Groups can contributed portlets which can be plugged into a portal. Currently Indiana, Argonne, Michigan, NCSA and Texas have been able to exchange porlets and install them in different portlet containers without much trouble. Now that there is an official portlet standard this will become even easier. Finally, each user can select and configure the portlets he/she wishes to use and this selection becomes part of the user's persistent context.

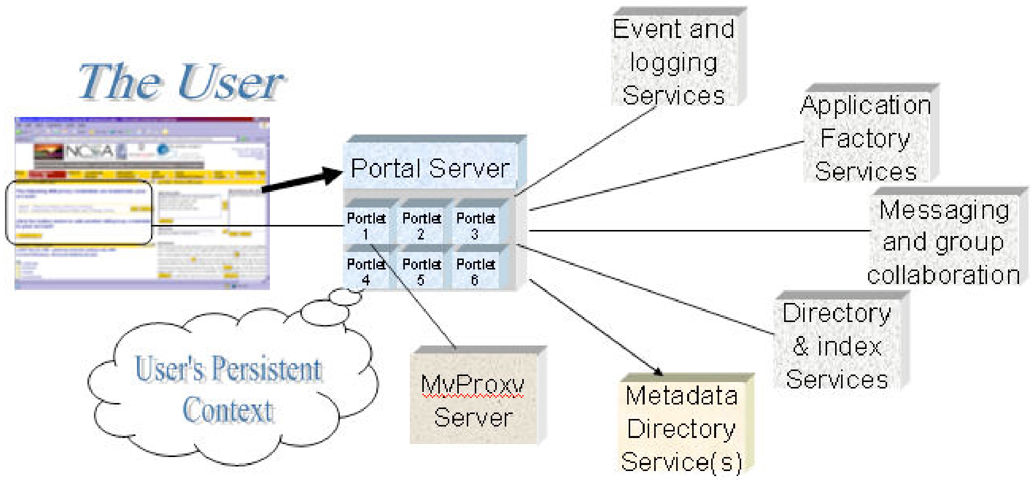
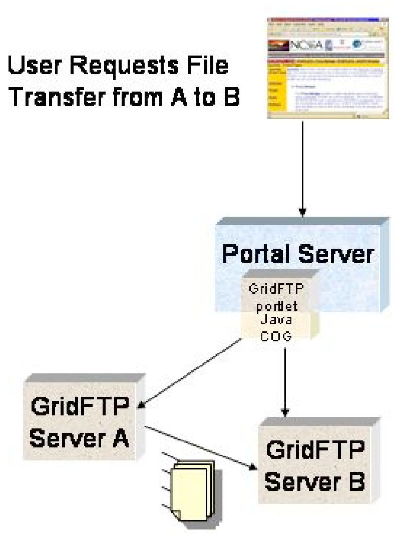


Figure 2. The user's view of the portal as a configurable collection of remote service interfaces.

The OGCE consortium currently has portlets for

- Management of user proxy certificates
- Remote file Management via Grid FTP
- News/Message systems and Grid Event/Logging service
- Basic remote job launch and monitoring.
- Access to OGSA services such as an OGSA registry browser.
- Access to directory services such as LDAP.
- Specialized Application Factory access
- Access to Metadata Index tools



For example, the Grid FTP portlet allows the user to manage remote file spaces using a proxy certificate stored in the users session context by the MyProxy portlet. The portlet provides the user with a directory view of the two remote file spaces and it is easy to transfer files from one space to the other (without moving the file through the portal server) and to download or upload files.
This is accomplished using the Argonne Java COG kit which is loaded into the portlet container and provides each portlet with access to many standard Grid utilities

D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas

such as access to the MyProxy server and the Globus gram job submission utilities.

There is, unfortunately a hidden weakness to this one-Grid service – one portlet approach. Adding new portlets to a portlet container often requires adding new libraries (Jar files). If each portlet comes with its own implementation of the way it access common services such as user proxy certificates or xml parsers, conflicts will result. The OGCE has taken the following approach to solving this problem. The portal container will be equipped with a standard library of basic portal services. As shown in Figure 4, this would supplement the COG Kit and provide a standard API for common tasks like, accessing Grid proxies, resolving Grid Service Handles to Grid Service Locators, accessing VO membership information, and providing secure mechanisms to invoke remote Grid Services.

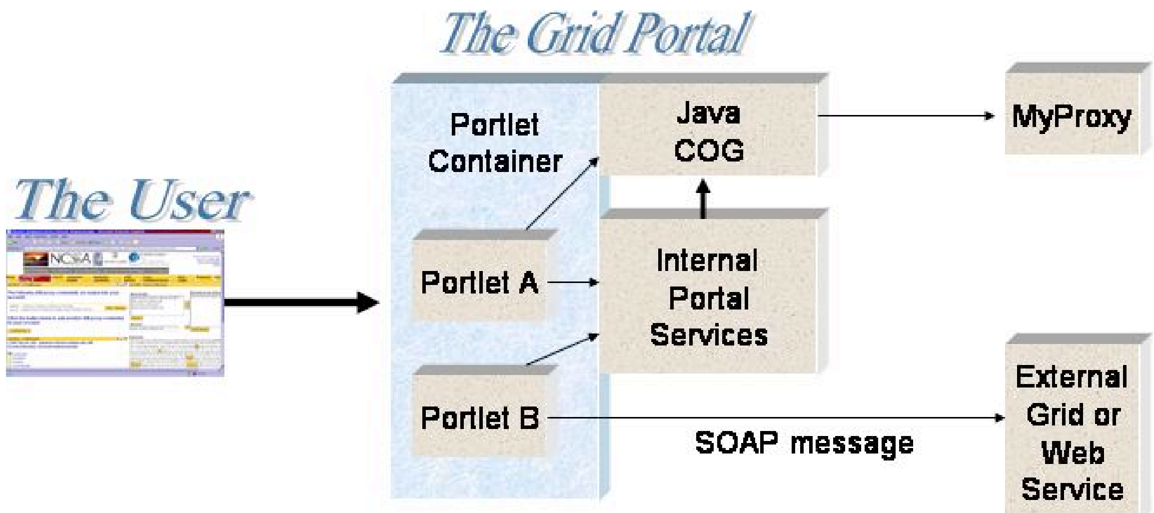


Figure 4. OGCE Grid Portal Architecture. A layer of shared services are provided to portlets to allow them to access important resources in the most elementary manner.

4. Grid Portals and Web and Grid Services

An important role for the portal is to provide the user interface to remote Grid services. As defined by the Open Grid Service Infrastructure (OGSI) [18], a Grid service is a web service that also adheres to some simple behavior patterns and implements the GridService Port Type and other ports it needs. OGSI also provides standard way to name, discover and interrogate a Grid Service. For the portal this makes it very easy to "view" a Grid service to learn about its lifetime, other ports it implements and some basic state information. For example, there are several Grid Services Browser portlets that use the default Grid Service Port to query the "service data elements" provided by the service (Figure 5). These service data elements are simple XML records that can be easily displayed for the user.

D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas

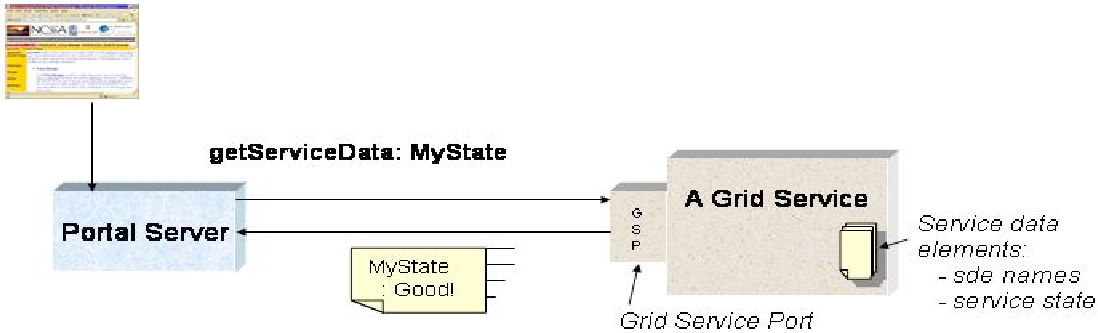


Figure 5. Interrogating a Grid Service by the findServiceData method.

There is a major weakness of the portal architecture described so far. The concept of a single specialized portlet for each remote Grid or web service is not scalable. The user should be able to discover Grid services and dynamically load an interface to interact with it, without having to install a new service client portlet into the server. There are two approaches to solving this problem. One approach is based on the fact that every web service is described by a WSDL document, which completely defines the message types it receives and the message types that constitute responses. Portal tools like Xidra/OntoBrew [1] can, on-the-fly translate a WSDL document into a set of interface web forms to present to the user. When the user fills out the forms the values are automatically encoded into a SOAP request to the service that is dispatched and the reply presented to the user. A second approach is based on the new Web Services for Remote Portlets (WSRP) [17]. In WSRP, the service provides markup information to the portal server which composes this into a portlet that can be used to get user responses and then call the remote service. Work is needed to see how WSRP can be use in the context of OGSI.

Grid services are identified by a Grid Service Handle, which is a URI for locating the service. Grid Service Handles are resolved into Grid Service References which is, most often, an annotated WSDL document for the service. A general philosophy of OGSI is that there may be millions of Grid services. For example, each document in a document repository may be a Grid Services. In this case the GSH represents the document and this can be resolved into a GSR that provides a handle to a service that can fetch and or render the document for the portal. It is extremely important for the portal to be able to adapt to such a rich service environment in a graceful manner.

5. Events and messaging

(this section to be added.)

D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas
9-19-03

6. Portlet Technology - Detailed Discussion

(this section will describe detailed portlet technology issues including Chef, GridSphere, JSR-168, WAP and WSRP.)

7. Security Considerations

This report is about Grid Portals that are accessed by user to access Grid resources. There are two levels of security that must be considered here. At the top level is the access by the user through a web browser to the portal server. At this level we use best practice HTTPS security. The portal server is itself a web server and must use industry-standard security to protect its integrity. At the second level we have the security involved with providing the portal server access to remote Grid services. In this case we will follow all recommendations and standards coming from the OGSA Security working group. GCE will not attempt to define any new security protocols. For additional security concerns see section 2.1 of this report.

8. Glossary

(to be finished)

Portal.

Grid Portal.

Scientific Grid Portal.

Administrative Grid Portal

Portlet

Author Information

Dennis Gannon
Department of Computer Science
Indiana University
gannon@cs.indiana.edu

Geoffrey Fox
Pervasive Technology Labs
Indiana University
gcf@indiana.edu

Marlon Pierce
Pervasive Technology Labs
Indiana University
marpierc@indianaa.edu

Beth Plale
Department of Computer Science
Indiana University
plale@cs.indiana.edu

Jay Alameda
NCSA
University of Illinois
jalameda@ncsa.uiuc.edu

Gregor von Laszewski
MCS
Argonne National Labs
gregor@mcs.anl.gov

D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas

Charles Severance
Media Union
University of Michigan
csev@umich.edu

Mary Thomas
TACC
University of Texas – Austin
mthomas@austin.tacc.utexas.edu

Joseph Hardin
Media Union
University of Michigan
hardin@umich.edu

John R Boisseau
TACC
University of Texas – Austin
boisseau@austin.tacc.utexas.edu

Intellectual Property Statement

Full Copyright Notice

D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas

References

[1] Chipara, O., Slominski, A., Xydra – An automatic form generator for Web Services, see: http://www.extreme.indiana.edu/xgws/xydra/

[2] Cubera, F. and Co-Authors, "Business Process Execution Language for Web Services, Version 1.0," http://www-106.ibm.com/developerworks/library/ws-bpel/

[3] *The Grid: Blueprint for a New Computing Infrastructure*, Ian Foster and Carl Kesselman (Eds.), Morgan-Kaufman, 1998.

[4] Foster, I., Kesselman, K., Nick, J., Tuecke, S., "The Physiology of the Grid. An Open Grid Services Architecture for Distributed Systems Integration", in *Grid Computing: Making the Global Infrastructure a Reality*, Fox, G. Ed. Wiley, 2003.

[6] Fox, G., Berman, F., and Hey, T., *Grid Computing: Making the Global Infrastructure a Reality*, Fox, G. Ed. Wiley, 2003.

[5] Fox, G., Pierce, M., Gannon, D., Thomas, M., An Overview of Grid Computing Environments, Global Grid Forum GFD-I.9. http://www.gridforum.org/documents/GFD, 2003.

[7] Gannon, D., Chiu, K., Govindaraju, M., Slominski, A., "A Brief Introduction to the Open Grid Services Infrastructure", to appear Journal of Computing and Informatics, Special Issue on Grid Computing. 2003.

[8] Gannon, D. and Co-Authors, "Programming the Grid: Distributed Software Components, P2P and Grid Web Services for Scientific Applications," Journal of Cluster Computing, 5(3): 325-336 (2002)

[9] Krishnan, S. and Co-Authors, "The XCAT Science Portal", Proceedings SC2001, Nov. 2001, Denver.

[10] Johnston, "Implementing Production Grids", in *Grid Computing: Making the Global Infrastructure a Reality*, Fox, G. Ed. Wiley, 2003.

[11] The Particle Physics Data Grid, http://www.ppdg.net/

[12] The Grid Physics Network, http://www.griphyn.org/

[13] Fox, G., Gannon, D., "Computational Grids", IEEE Computer Science Eng. Vol. 3, No. 4, , pp. 74-77, 2001

[14] Network for Earthquake Engineering Simulation Grid (NEESgrid), http://www.neesgrid.org

[15] Global Grid Forum, http://www.ggf.org

[16] Open Grid Service Architecture Data Access and Integration, http://www.ogsa-dai.org.uk

[17] Kropp, A., Leue, C., Thompson, R., Web Services for Remote Portlets (WSRP), OASIS http://www.oasis-open.org

[18] Tuecke, S., et. al., "The Open Grid Service Infrastructure (OGSI)", http://www.gridforum.org/ogsi-wg

[19] Barbera, R., "The GENIUS Grid Portal," Computing in High Energy and Nuclear Physics, 24-28 March 2003, La Jolla, California.

[20] Allan, R**.,** Integrated e-Science Environment for CLRC, http://esc.dl.ac.uk/IeSE

[21] JSR168 Portlet Specification. http://jcp.org/aboutJava/communityprocess/review/jsr168/

[22] NCSA Alliance Scientific Portal Project, http://www.extreme.indiana.edu/alliance

D. Gannon, G. Fox, M. Pierce, B. Plale, Indiana University
G. von Laszewski, Argonne National Labs
C. Severance, J. Hardin, University of Michigan
J. Alameda, NCSA, Univ. of Illinois
M. Thomas, J. Boisseau, TACC, University of Texas

[23] GridLab, The GridSphere Portal http://www.gridsphere.org/gridsphere/gridsphere
[24] Govindaraju, M., et. al., "Merging the CCA Component Model with the OGSI Framework", Proceedings 3rd International Symposium on Cluster Computing and the Grid, pp. 182-199, May, 2003, Tokyo.
[25] Vanderbilt, Peter, "Re: Grid-SOA and service state", GGF OGSI-WG
[26] GriPhyn Project, "Chimera Virtual Data System," http://www.griphyn.org/chimera