Grid Portal Development for Sensing Data Retrieval and Processing

Diego Arias, Mariana Mendoza, Fernando Cintron, Kennie Cruz, and Wilson Rivera

Parallel and Distributed Computing Laboratory
University of Puerto Rico at Mayaguez
P.O.Box 9042, Mayaguez, Puerto Rico 00681, USA

Abstract

This paper presents our experiences developing grid portals for radar and sensor based applications. Underlying these gateways there are existing grid technologies such as Globus Toolkit 4.0.1 and Gridsphere. The grid portals provide secure and transparent access to applications dealing with data acquired from network of radar and sensors deployed in Puerto Rico, while implementing useful functionalities for data management and analysis.

1 Introduction

Grid computing [1] involves coordination, storage and networking of resources across dynamic and geographically dispersed organizations in a transparent way for users. The Open Grid Services Architecture (OGSA) [2], based upon standard Internet protocols such as SOAP (Simple Object Access Protocol) and WSDL (Web Services Description Language), is becoming a standard platform for grid services development. Operational grids based on these technologies are feasible now, and a large number of grid prototypes are already in place (e.g. Grid Physics Network (GridPhyN) and Teragrid among many others).

Although applications can be built using basic grid services, this low-level activity requires detailed knowledge of protocols and component interactions. In contrast, grid portals hide this complexity via easy-to-use interfaces, creating gateways to computing resources. An effective grid portal provides tools for user authentication and authorization, application deployment, configuration and application execution, and management of distributed data sets.

The Open Grid Computing Environments (OGCE) portal software is the most widely used toolkit for building reusable portal components that can be integrated in a common portal container system. The OGCE portal toolkit includes X.509 Grid security services, remote file and job management, information and collaboration services and application interfaces. The OGCE portal toolkit is based on the notion of a “portlet,” a portal server component that controls a user-configurable panel. A portal server supports a set of web browser frames, each containing one or more portlets that provide a service. This portlet component model allows one to construct portals merely by instantiating a portal server with a domain specific set of portlets, complemented by domain-independent portlets for collaboration and discussion. Using the toolkit, one wraps each grid service with a portlet interface, creating a “mix and match” palette of portlets for portal creation and customization. Recently, there have been significant advances in grid portal technologies and development of scientific grid interfaces [3, 4].

This paper presents our experiences developing grid portals for radar and sensor based applications. The organization of the paper is as follows. Section 2 discusses briefly the grid testbed infrastructure deployed at the University of Puerto Rico to investigate issues related to grid computing. Section 3 and section 4 describe the applications and the grid portal developments. Section 5 discusses related work.

2 Grid Test-bed Infrastructure

The PDCLab Grid Testbed, deployed at the University of Puerto Rico-Mayaguez, is an experimental grid designed to address research issues, such as the effective integration of sensor

and radar networks into grid infrastructures. The PDClab grid test-bed components run CentOS 4.2 and the Globus Toolkit 4.0.1. The Globus Toolkit includes, among other components, services, such as a security infrastructure (GSI), data transport service (GridFTP), execution services (GRAM), and Information services (MDS).

The Grid Security Infrastructure is used by the Globus Toolkit for authentication and secure communication. GSI is implemented using public key encryption, X.509 certificates, and the secure sockets layer (SSL) communication protocol and incorporates single sign-on and delegation.

The Monitoring and Discovery Service (MDS) is used to discover, publish and access both static and dynamic information from different resources in a computational grid. MDS uses the Lightweight Directory Access Protocol (LDAP) to access such information on the different grid components and provides a unified view of the disparate grid resources.

The Globus Resource Allocation Manager (GRAM) is used for allocation and management of resources on the computational grid using a Resource Specification Language (RSL) to request resources. GRAM also updates the MDS with information as to the availability of grid resources. The GRAM API can be used to submit a job, query the status of a job, and cancel a job. A GRAM service runs on each resource that is part of the grid and that is responsible for interfacing with the local site resource management system (e.g. OpenPBS, Condor).

GridFTP is a secure, high-performance and robust data transfer mechanism used to access remote data. In addition to GridFTP, Globus provides Globus Replica Catalog to maintain a catalog of dataset replicas so that, instead of duplicating large datasets, only necessary pieces of the datasets are stored on local hosts. The Globus Replica Management software provides the replica management capabilities for data grid by integrating the replica catalog and GridFTP.

The computational resources available on the grid testbed (see Figure 1) include an IBM xSeries Linux cluster with 64 nodes, dual-processor at 1.2GHz, 53GB of memory and 1TB of storage; Eight (8) IA-64 Itanium servers, dual processor at 900 MHz, each with 8GB of memory and 140GB of SCSI Ultra 320 storage; Two (2) IA-32 Pentium IV servers, dual processor at 3 GHz, each with 1GB of memory and 120GB of ATA-100 storage; One (1) IA-32 Pentium III server, dual processor at 1.2 GHz with 2GB of memory and 140Gb of SCSI Ultra 160 storage; One (1) IA-32 Xeon server, dual processor at 2.8 GHz, L2 Cache 1MB with 1GB of memory and one 230 GB RAID of storage (STB Server); and two (2) PowerVault storage with 8TB.

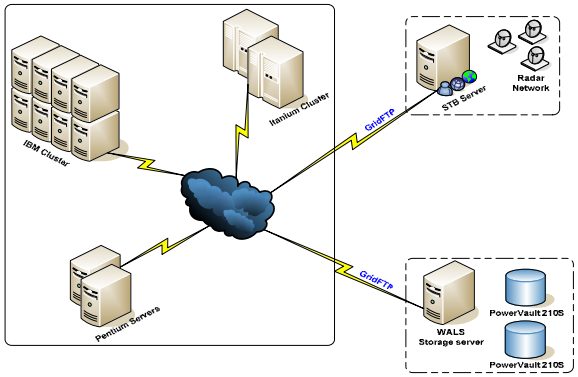

Figure 1: Grid-service based Infrastructure

3 The Student Test-Bed (STB) Grid Portal

The CASA[1] project is an NSF Engineering Research Center investigating the design and implementation of a dense network of low-power meteorological radars whose goal is to collaboratively and adaptively sense the lowest few kilometers of the earth's atmosphere. We have deployed a grid-service based tool to access and manipulate radar data from a radar network. The access to this infrastructure is provided via a grid portal interface. The developed grid portlets provide a presentation layer for the manipulation of both, processed data and raw-data from radar, and for the services to end-users. Additionally, the visualization of weather information is implemented also via portlets.

[1] http://casa.umass.edu/

The portal presentation layer and core portlets, included in the basic installation are made possible using Gridsphere. Figure 2 shows the customized STB portal. Gridsphere provides portlets for managing user accounts inside the portal framework. This set of portlets is integrated in the STB portal design to improve controlled access to certain resources and services which will be explained further on.

Figure 2: STB Grid portal interface

Users can access raw-data from radars through the grid portal. Files containing the raw-data are stored using the NetCDF[2] format. The data management portlets allows end-users to download the data in such a way that they can obtain an exact copy of a file or a set of files. To avoid the server overload, raw data requests are restricted to registered users only. Raw-data does not provide comprehensible information; it requires additional tools for extraction and processing. As a result, this feature is designed for advanced users (students, teachers and researchers) who have the adequate software and previous knowledge of data from radars. These kinds of users are able to request an account from the portal administrator.

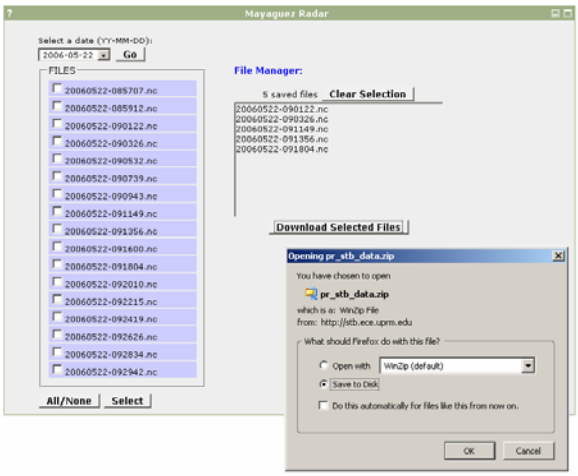

Figure 3: Data management portlet

Figure 3 shows the data management portlets. Once the user has been logged into the portal, the raw data request portlets are made available. The initial portlet shows a single selection form that permits the selection of the date of interest and then all available data is listed. Then, the data set selected can be downloaded as a compressed file. The grid portal provides current rainfall estimates over the western area of Puerto Rico through reflectivity displays. This information is unrestricted and is available for anyone who accesses the portal. Figure 4 shows how the base reflectivity information corresponding to a sweep is plotted over the Puerto Rico west coast area.

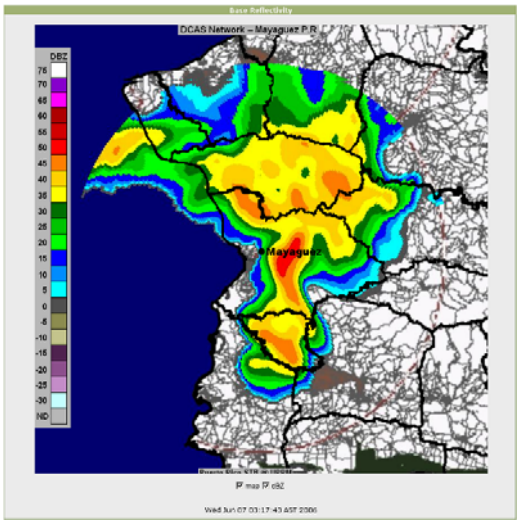

Figure 4: Base Reflectivity portlet.

[2] http://www.unidata.ucar.edu/software/netcdf/

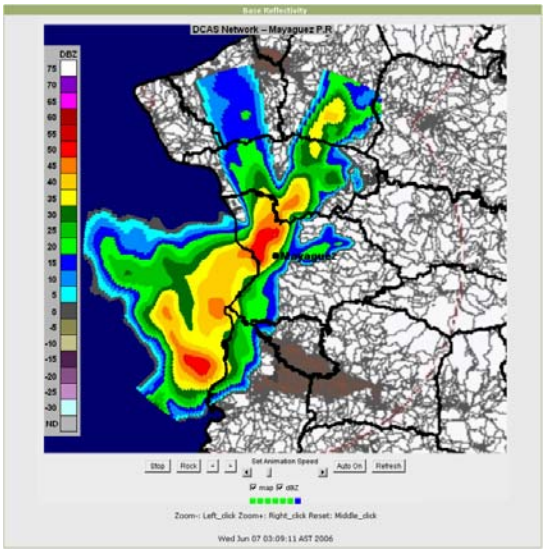

Figure 5: Base Reflectivity animation portlet

Figure 5 shows a portlet used to display a set of base reflectivity over the Puerto Rico west coast area. This portlet performs the animation of the data set and includes loop controls and zooming. The base reflectivity loop is useful in facilitating the tracking of meteorological phenomena.

NetCDF data is written as binary files, thus, it can not be read by users as plain text, and specialized software is required for its interpretation. There are several libraries, plugging, programs and a variety of tools to manipulate NetCDF files. However installation, configuration and usage of these tools can be very complex for inexperienced users. Additionally, due to format flexibility, the structure of the files varies, depending on the implementation procedure. To perform a specific task, one or more software tools are needed. For example, there is not available software to generate the reflectivity plots from the radar raw-data. A Java class was developed using more basic classes and libraries for NetCDF manipulation. Additionally, a similar class was developed to convert NetCDF to ASCII.

To facilitate the manipulation of raw-data from DCAS network nodes, two very useful services were implemented. These services allow end-users the execution of processes over the raw-data available in the storage system. Thus, users can upload its data sets from a local machine to the server, and process them. The available processes are:

- NCtoJPG: Rainfall rate plots are available in the Grid portal; but older plots are not maintained in to safe storage. Using the grid portal, users can send out from date data sets to the grid, and then receive the corresponding reflectivity plots.
- NCtoASCII: Through use of the grid portal, users can convert tNetCDF files to text files. This tool eliminates the utilization of extra software for data manipulation.

Services for end-users involve executing a process over a single file or a set of files. For instance, a set of NetCDF files can be uploaded with a NCtoJPG request. Data is processed and the output files are made available for downloading, using the grid portal. This entire procedure is transparent to users. The server may process each file and then reply to the output files; nevertheless, the server could be receiving data from the radar network or replying to other user requests at the same time. In order to avoid a crash due to an overload of simultaneous tasks, remote job execution is introduced. The server can submit a simple job or a multi-job to the grid testbed, instead the routine performance of simple local jobs only. Job submission is supported by Globus through GRAM. Additionally, PBS (Portable Batch System) is used as a job scheduler. Figure 6 shows the job submission functionality.

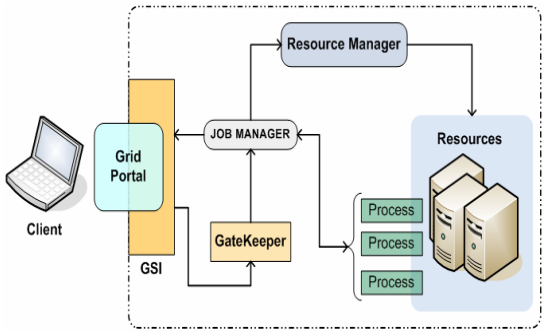

Figure 6: Job submission architecture

As shown in Figure 7, an important issue to point out when submitting multiple jobs is that CPU consumption is very quite high (≈ 97%) when a job is executed in a local server, and is close to 1%, when executing on the STB grid infrastructure.

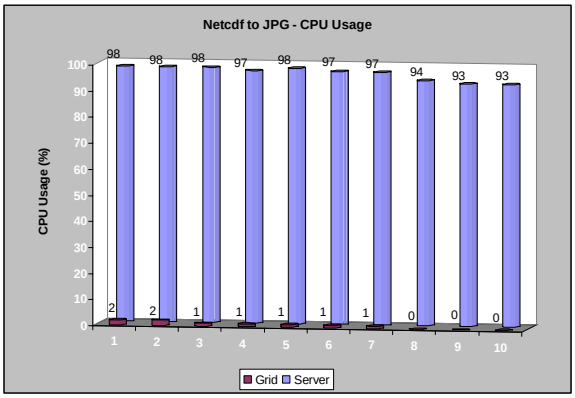

Figure 7: Percentage of CPU usage

4 The WALSAIP Grid Portal

The NSF WALSAIP[3] project is developing a new conceptual framework for the automated processing of information arriving from physical sensors in a generalized wide-area, large-scale distributed network infrastructure. The project is focusing on water-related ecological and environmental applications, and it is addressing issues such as scalability, modularity, signal representation, data coherence, data integration, distributed query processing, scheduling, computer performance, network performance, and usability. A distributed sensor network testbed is being developed at the Puerto Rico's Jobos Bay Natural Estuarine Research Reserve (JBNERR)[4]. The reserve has more than 2800 acres is located on the southern coast of Puerto Rico, between the municipalities of Guayama and Salinas. It is administered by the National Oceanic and Atmospheric Administration and it is managed locally by the Department of Natural and Environmental Resources.

One of the components of this project is developing a grid-based tool to define workflow composition of signal processing operators as an application service. This tool allows the composition of operators that may be geographically distributed and provided by diverse administrative domains. Again underlying this tool there are existing grid technologies such as Globus Toolkit 4.0 and Gridsphere. The design of the methodology for composing distributed signal operators follows two major requirements. Firstly, it is desirable to optimize resource management according to the complexity of the operators to be processed. Secondly, the composition of distributed resources requires metadata distribution and management mechanisms.

The Grid Portal Interface provides transparent and secure access to end-users. This portal allows end users to define signal processing workflows by using drag and drop functionalities. GridFTP is used to improve data transport from the data server (WALSAIP Server) to the Grid Portal server (PDC Server). Signal processing operators are deployed as grid services. This grid services may be geographically distributed and provided by different administrative domains. Figure 8 depicts the components of the application. Figure 9 illustrates the grid portal interface.

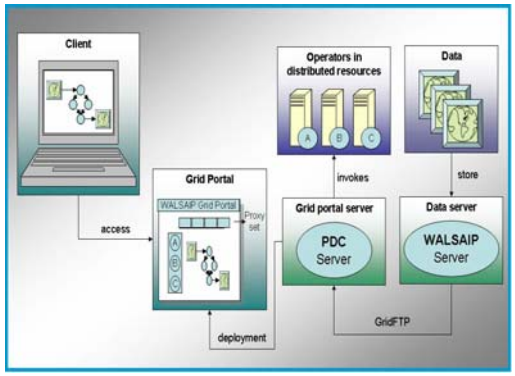

Figure 8: Signal processing application services over a grid environment

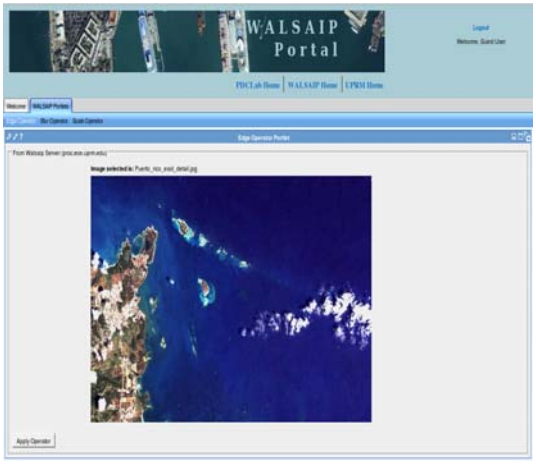

Figure 9: WALSAIP grid portal

[3] http://walsaip.uprm.edu/
[4] http://nerrs.noaa.gov/JobosBay/

5 Related Work

The Linked Environments for Atmospheric Discovery (LEAD) project [5] proposes an information technology framework for assimilating, forecasting, managing, analyzing, mining and visualizing a broad array of meteorological data and model output independent of format and physical location. LEAD is currently led by nine institutions. The LEAD system is dynamically adaptable in terms of time, space, forecasting and processing. The LEAD infrastructure includes technologies and tools, such as, Globus toolkit, Unidata's Local Manager (LDM), Open-Source Project for a Network Data Access Protocol (OpenDap) and the OGSA Data Access and Integration (OGSA-DAI) service. The LEAD portal is based on OGCE.

Majithia et. al. [6] proposed Triana, a framework that allow users graphically create complex service compositions based on BPEL4WS (Business Process Execution Language for Web Services). It also allows users to easily carry out "what-if" analysis by altering existing workflows. Using this framework it is possible to execute the composed graph service on a Grid network.

Gao et. al. [7] developed a service composition architecture that optimizes the aggregate bandwidth utilization within operator networks. A general service composition is proposed to model the loosely coupled interaction among service components as well as the estimated traffic that flows among them.

Glatard et. al. [8] discussed how build complex applications by reusing and assembling scientific codes on a production grid infrastructure. The authors stated two paradigms for executing application code on a grid. A task based approach, associated to global computing, characterized by its efficiency, and the service approach, developed in meta computing and the Internet communities, characterized by its flexibility.

References

1. Foster and C. Kesselman (1998), "The grid: blueprint for a future computing infrastructure" M*organ Kaufmann Publishers*
2. Foster, C. Kesselman, J. Nick, and S. Tuecke (2002), "The physiology of the Grid: An open Grid services architecture for distributed systems integration, *Technical report, Open Grid Service Infrastructure WG, Global Grid Forum*.
3. D. Gannon, G. Fox, M. Pierce, B. Plale, G. von Laszewski, C. Severance, J. Hardin, J. Alameda, M. Thomas, J. Boisseau, Grid Portals: A Scientist's Access Point for Grid Services, GGF Community Practice document, working draft 1, September 2003.
4. Gregor von Laszewski, Jarek Gawor, Sriram Krishnan, and Keith Jackson. Grid Computing: Making the Global Infrastructure a Reality, chapter Commodity Grid Kits - Middleware for Building Grid Computing Environments, pages 639–656. Communications Networking and Distributed Systems, Wiley, 2003.
5. K.K. Droegemeier, D.Gannon, D. Reed B. Plale, J. Alameda, T. Baltzer, K. Brewster, R. Clark, B. Domenico, S. Graves, E. Joseph, D. Murray, R. Ramachandran, M. Ramamurthy, L. Ramakrishnan, J. A. Rushing, D. Weber, R. Wilhelmson, A. Wilson, M. Xue, and S.Yalda, Service-oriented environments for dynamically interacting with mesoscale weather. C*omputing in Science &Engineering*, 7(6):12{29, Nov.-Dec. 2005.
6. S. Majithia, M. Shields, I. Taylor, I. Wang. "Triana: A Graphical Web Service Composition and Execution Toolkit." IEEE International Conference on Web Services (ICWS'2004), San Diego, California, USA, 2004.
7. X. Gao, R. Jain, Z. Ramzan, U. Kozat. "Resource optimization for Web Service Composition." In Proceedings of IEEE SCC2005, 2005.
8. T. Glatard, J. Montagnat, X. Pennec. Efficient services composition for Grid-enabled Data-intensive Applications." In Proceedings of the IEEE International Symposium on High Performance Distributed Computing (HPDC'06), Paris, France, 2006.