Monitoring the Earth System Grid with MDS4

Ann Chervenak[1], Jennifer M. Schopf[2,3,4], Laura Pearlman[1], Mei-Hui Su[1], Shishir Bharathi[1], Luca Cinquini[5], Mike D'Arcy[1], Neill Miller[2,3], David Bernholdt[6]

[1]Information Sciences Institute, University of Southern California, [2] Mathematics and Computer Science Division, Argonne National Laboratory, [3] Department of Computer Science, The University of Chicago, [4] UK National eScience Centre, University of Edinburgh, [5] National Center for Atmospheric Research, [6] Oak Ridge National Laboratory

annc@isi.edu, jms@mcs.anl.gov, laura@isi.edu, mei@isi.edu, shishir@isi.edu, luca@ucar.edu, neillm@mcs.anl.gov, bernholdtde@ornl.gov

Abstract

In production Grids for scientific applications, service and resource failures must be detected and addressed quickly. In this paper, we describe the monitoring infrastructure used by the Earth System Grid (ESG) project, a scientific collaboration that supports global climate research. ESG uses the Globus Toolkit Monitoring and Discovery System (MDS4) to monitor its resources. We describe how the MDS4 Index Service collects information about ESG resources and how the MDS4 Trigger Service checks specified failure conditions and notifies system administrators when failures occur. We present monitoring statistics for May 2006 and describe our experiences using MDS4 to monitor ESG resources over the last two years.

1. Introduction

As Grids for scientific applications become larger and more complex, the management of these environments becomes increasingly difficult. Commonly, these scientific Grids consist of a large number of heterogeneous components deployed across multiple administrative domains, including storage systems, compute clusters, Web portals, and services for data transfer, metadata management, and replica management. Monitoring these components to determine their current state and detect failures is essential to the smooth operation of Grid environments and to user satisfaction.

Monitoring systems collect, aggregate, and sometimes act upon data describing system state. This information can help users make resource selection decisions and help administrators detect problems. Monitoring systems can typically be queried and, in many cases, can take actions based on events. Grids present additional challenges for monitoring systems because of the frequency with which resources are added and removed and because of the distributed nature of the responsibility for administering resources in a Grid.

In this paper, we describe the monitoring infrastructure used by the Earth System Grid (ESG) project [1], a scientific collaboration that supports global climate research. The major goals of ESG are to provide infrastructure to publish key climate datasets and to allow scientists throughout the world to acquire these datasets, thereby increasing scientific productivity. The ESG infrastructure spans seven sites and includes components and services that support climate dataset publication and access.

To monitor this infrastructure, we use the Globus Toolkit Version 4 (GT4) Monitoring and Discovery System (MDS4) [2]. The Globus Toolkit [3] provides middleware to support secure resource sharing among participants in a Grid. MDS4 defines and implements mechanisms for service and resource discovery and monitoring in distributed environments. MDS4 is distinguished from other monitoring systems by its extensive use of interfaces and behaviors defined in the WS-Resource Framework [4] and WS-Notification specifications and by its deep integration into components of the Globus Toolkit.

The remainder of this paper is organized as follows. We describe the Earth System Grid and the monitoring needs of ESG users and system administrators. Next, we describe MDS4 and explain how we use it to monitor ESG resources. We describe specific examples of how monitoring has helped ESG to detect a variety of problems over the last two years and describe ESG monitoring statistics over a period of one month.

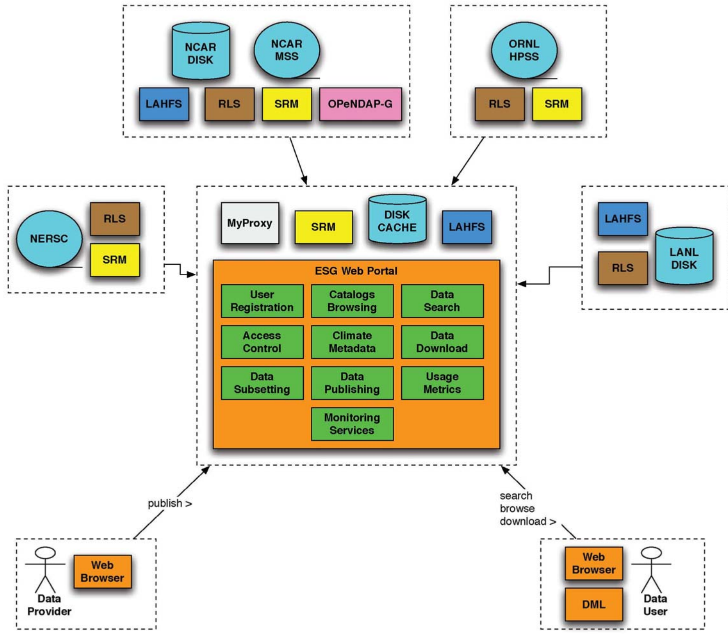

Figure 1. Currently deployed ESG components and services.

2. The Earth System Grid

The Earth System Grid (ESG) [1] supports the next generation of climate modeling research by providing the infrastructure and services that allow climate scientists to publish and access key data sets generated from climate simulation models. Important datasets that are provided by ESG to the climate community include simulations generated using the Community Climate System Model (CCSM) [5] and the Parallel Climate Model (PCM) [6]. These datasets are accessed by scientists throughout the world.

ESG has become an important community resource for climate scientists. There are currently two ESG Web portals, deployed at NCAR and LLNL. In 2005, users of the ESG Web portal at NCAR issued 37,285 requests to download 10.25 terabytes of data. Use of this portal has steadily increased, both in terms of the amount of data downloaded and the number of registered users of the system. By the fourth quarter of 2005, users downloaded approximately two terabytes of data per month. This portal registered 1881 users in 2005 and is currently adding more than 150 users per month.

The ESG portal at LLNL serves over 27 Terabytes of CCSM data sets for the Intergovernmental Panel on Climate Change [7] stored in 458,500 files. This portal has over 700 registered users. A total of 103 Terabytes of data have been downloaded from this portal at an average rate of 300 GBytes per day.

The ESG infrastructure is composed of resources at seven sites: Argonne National Laboratory (ANL), Lawrence Berkeley National Laboratory (LBNL), Lawrence Livermore National Laboratory (LLNL), Los Alamos National Laboratory (LANL), the National Center for Atmospheric Research (NCAR), Oak Ridge National Laboratory (ORNL), and USC Information Sciences Institute (ISI). Figure 1 shows the resources and services that are deployed in the ESG, including: the Web portal deployed at NCAR that provides data access to scientists; the HTTP data servers deployed at NCAR and LANL for that are used to send data to users' desktops; hierarchical mass storage systems at NCAR, LBNL and ORNL; the OPeNDAP system [8] that is used to filter and subset datasets to reduce the amount of data that must be transferred; the Storage Resource Manager (SRM) [9] that provides management of storage systems and coordinates multiple-file transfers; the GridFTP data transfer service [10, 11]; and metadata [12, 13] and replica management catalogs [14, 15].

The climate community has come to depend on the ESG infrastructure as a critical resource. Thus, any failure of ESG components or services can severely disrupt the work of many scientists. To detect failures quickly and minimize infrastructure downtime, we have spent significant effort to improve the ESG monitoring infrastructure, which is based on the Monitoring and Discovery System from the Globus Toolkit.

3. The Monitoring and Discovery System (MDS4)

Monitoring systems for Grids have many applications: listing service and resource data for resource selection, displaying information for system administrator checks, validating software installations, warning users or administrators about errors, and even assisting in debugging problems. Within the Globus Toolkit, we have developed the Monitoring and Discovery System (MDS4) to aggregate data, respond to queries, and perform event-driven actions; the services in MDS4 can be used in each of the use cases listed above.

MDS4, like much of the current Globus Toolkit, is based on the Web Services Resource Framework (WSRF) [4] standards, which provide standard interfaces and mechanisms such as the ability to associate state with services, lifetime management of services, and the ability to subscribe to state information associated with a service and be notified when the state changes. MDS4 implements a standard Web services interface to a variety of local monitoring tools and other information sources, providing a “protocol hourglass,” depicted in Figure 2, that defines standard protocols for information access and delivery and standard schemas for information representation. Below the neck of the hourglass, MDS4 interfaces to different local information sources, translating their diverse schemas into appropriate XML schema (based on standards such as the GLUE schema [16] whenever possible).

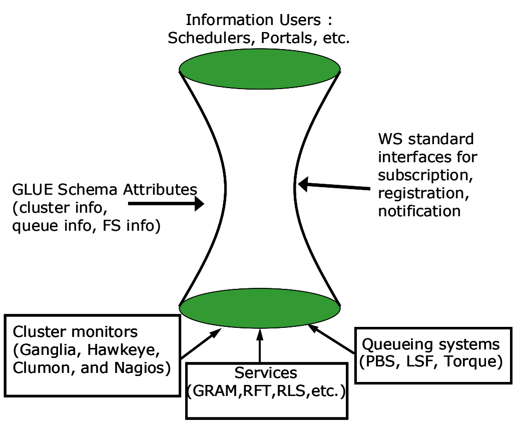

Figure 2: The MDS4 hourglass provides a uniform query, subscription, and notification interface to a variety of information sources, Web services, and other monitoring tools.

Above the neck of the hourglass, various tools and applications can be constructed to take advantage of the uniform Web services query, subscription, and notification interfaces to the information source that are defined by the WS Resource Framework (WSRF) and WS-Notification families of specifications. These interfaces are used to access state (called *resource properties)* that is associated with a Web service.

3.1. MDS4 Services

MDS4 currently includes two higher-level services: the *Index service*, which collects and publishes aggregated information about Grid resources, and the *Trigger service*, which collects resource information from the Index Service and performs actions when certain conditions are met.

An MDS4 Index service collects information about Grid resources and makes this information available as resource properties. It stores not only the location from which a piece of data is available, but also a cached version of the data, and it maintains that cached copy via lifetime management mechanisms.

The MDS4 Trigger service periodically queries the Index Service to obtain resource information and compares that data against a set of conditions. When a condition is met, the Trigger service performs an action, such as emailing system administrators to notify them that a monitored component is down or that disk space on a server has reached a threshold.

The Index and Trigger service implementations are both built on the Globus *aggregator framework*, a software framework for building services that collect and aggregate XML-formatted data supplied by information providers. Services built on this framework are sometimes called aggregator services. Such services share common mechanisms to:

• collect information using a pluggable backend interface known as the *aggregator source* interface. The MDS4 distribution includes aggregator sources that acquire resource information from remote services using standard WSRF polling or notification/ subscription or by executing external programs;

• use a common configuration mechanism to register sources indicating what data to get, and where. Remote administrators can add configuration information, for example, to tell the Index or Trigger service to collect information about additional resources, or to tell the Trigger service to perform a specific action when a particular condition occurs;

• use a relaxed consistency model so that published information is renewed at a administrator-controllable frequency; and

• self-clean the aggregator service by associating each registered information source with a lifetime; if a registration expires without being refreshed, it and its associated data are removed from the service. Outdated entries are removed automatically when they cease to renew their registrations.

3.2. MDS4 Information Providers

The data that an MDS4 aggregator source publishes into an aggregator service is obtained from an external component called an *information provider*. The information provider can be a WSRF-compliant service from which data is obtained via query or notification mechanisms, or an executable program that obtains data via some domain-specific mechanism. MDS4 includes information providers for resources including the Nagios [17], Hawkeye [18] and Ganglia [19] monitoring systems; a variety of queuing systems, including PBS-variants, LSF, and Condor; as well as standard Globus services for job submission, file transfer, and replica location.

In addition to these information providers, any executable can be made into an information provider that produces a valid XML document, which is then used by the Index or trigger service. This type of provider is often implemented as a script that performs certain actions to determine the status of a remote resource. For example, when monitoring a GridFTP service, a script might attempt to perform a small data transfer and check the result of that operation. For Storage Resource Managers (SRMs) and their associated mass storage systems, the information provider can run an SRM client program that provides status information about the SRM and the storage system. In addition, a file-scraping information provider is available.

3.3. User Interface

MDS4 also includes a user interface tool called WebMDS that uses standard resource property requests to query state information and to format and display it. In addition, GT4 command-line clients as well as Java, C, and Python APIs implement resource property query operations that can be used to query an Index service directly, when required.

4. Monitoring the Earth System Grid

The services that are monitored in the ESG infrastructure are listed in Table 1. These services include GridFTP data transfer services, the OPeNDAP service that filters requested information to reduce the amount of data transferred, the ESG Web portal, two HTTP servers for data access, Replica Location Service catalogs at five sites, Storage Resource Managers at four sites, and three hierarchical mass storage systems. The Index and Trigger services for ESG run on the machine at NCAR that also hosts the ESG Web portal and an HTTP data service, as shown in Figure 3.

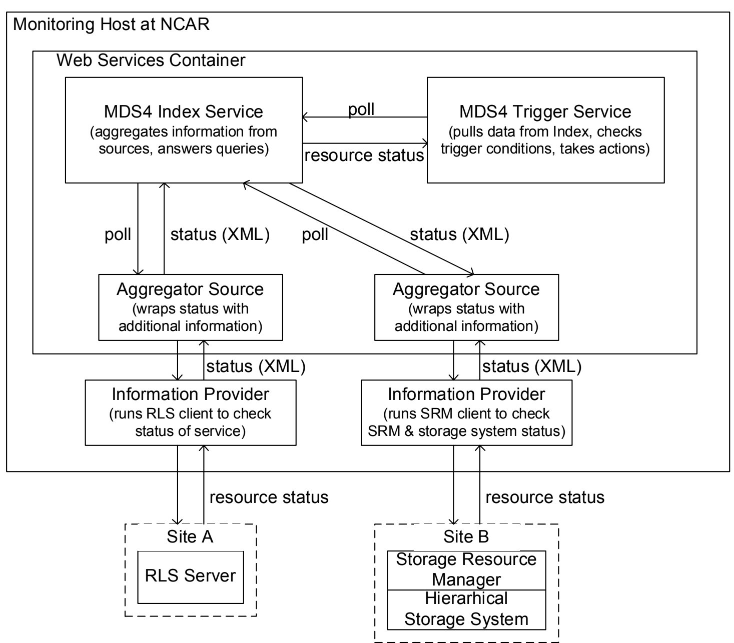

Figure 3. Components of the MDS4 deployment for ESG.

Table 1. Services monitored in ESG.

Service Being Monitored	ESG Location
GridFTP server	NCAR
OPeNDAP server	NCAR
Web Portal	NCAR
HTTP Dataserver	LANL, NCAR
RLS servers	LANL , LBNL, NCAR, ORNL
Storage Resource Managers	LBNL, NCAR, ORNL
Hierarchical Mass Storage Systems	LBNL, NCAR, ORNL

Figure 4. The ESG portal queries the Index service and displays resource status. Smiling faces indicate functioning services.

Monitored data are collected in the project Index service, which executes one or more information providers at a configured frequency. In our deployment, this is once every ten minutes. The resource information collected by the MDS4 Index Service is queried by the ESG Web portal. This information is then used to generate an overall picture of the state of ESG resources that is displayed on the Web portal page, as shown in Figure 4.

The information in the Index service is also polled periodically by the Trigger service, as shown in Figure 3. In our deployment, we poll this data every five minutes. Based on the current resource status, the Trigger service determines whether specified trigger rules and conditions are satisfied and, if so, performs the specified action for each trigger. In our current deployment, we require that a condition is valid for at least ten minutes before a trigger action is executed. This guarantees that the condition has been monitored twice at the Index service. Thus, we avoid sending failure messages based on transient conditions in the network or services. In addition, our Trigger service is configured so that subsequent execution of a particular trigger action will occur at most once per day. This prevents users from being overwhelmed by repeated failure messages from the same resource. Table 2 summarizes the configuration parameters that are used for the ESG deployment of the MDS4 Trigger service.

Table 2. Trigger service configurations.

Monitored Parameter	Value	Explanation
Poll Interval	5 Min	Poll for changes in monitored conditions to determine whether to trigger actions
Minimum Firing Interval	1 Day	Minimum interval between successive firings of a trigger.
Minimum Match Time	10 Min	Monitored conditions do not trigger actions unless they are valid for at least this value to avoid sending messages for transient failures.

Currently, the only action taken by the Trigger service is to send email to system administrators when services fail. Using these email notifications, system failures can be detected and corrected by system administrators, ideally before they affect the larger ESG user community. In the future, we plan to include richer recovery operations as trigger actions, such as automatic restart of failed services.

ESG monitoring operations impose relatively low load on the ESG Index and Trigger services. In our deployment, the Index service collects information from resources every ten minutes, and the Trigger Service pulls information from the Index service every five minutes. These services easily support these rates of update and query operations. More frequent monitoring or trigger operations would impose additional load on these services. Performance of the MDS4 Index Service under heavier loads, for example, when also performing resource discovery operations, is reported by Schopf, et al. [2].

For the month of May 2006, ESG's deployment of MDS4 generated 47 failure messages that were sent to ESG system administrators. These are summarized in Table 3. The majority of these failure messages were caused by downtime throughout the month of services at LANL due to certificate expiration and service configuration problems. (During this period, staff were not available to address these issues.) Additional messages from this site were suppressed by disabling the site's triggers during a two-week period.

Table 3. ESG monitoring messages for May 2006.

Total error messages for May 2006	47
Messages related to certificate and configuration problems at LANL	38
Failure messages due to brief interruption in network service at ORNL on 5/13	2
HTTP data server failure at NCAR 5/17	1
RLS failure at LLNL 5/22	1
Simultaneous error messages for SRM services at NCAR, ORNL, LBNL on 5/23	3
RLS failure at ORNL 5/24	1
RLS failure at LBNL 5/31	1

The remaining error messages indicate short-term service failures. Two failure messages were generated due to a network outage at ORNL on May 13th. Three messages were generated by unrelated failures of RLS servers. Three error messages for Storage Resource Managers (SRMs) at different sites were generated on May 23rd. Since it was unlikely that all three of these services would fail simultaneously, we investigated these messages and found an error in our monitoring logic, which has since been corrected.

5. Experience Monitoring ESG

In the last two years, ESG has found the system-wide monitoring infrastructure provided by MDS4 to be especially useful for the following use cases.

Overview of current system state for users and system administrators. As shown in Figure 4, the ESG portal displays an overall picture of the current status of the ESG infrastructure. This gives users and administrators an understanding at a glance of which resources and services are currently available.

Failure notification. Failure messages provided by the Trigger service have helped system administrators to identify and quickly address failed components and services. Before the monitoring system was deployed, services would fail and might not be detected until a user tried to access an ESG dataset. Earlier monitoring efforts were ad hoc, with different ESG sites devoting varying levels of effort to detecting and repairing failures. The MDS4 deployment has enabled a unified interface and notification system across ESG resources.

One enhancement made to MDS4 based on our experience was to include additional information about the location and type of failed service in the subject line of trigger notification email messages to allow message recipients to filter these messages and quickly identify which services need attention.

Validation of new deployments. On several occasions since we started monitoring the ESG, we have made major changes to the Grid infrastructure, such as modification of service configurations or deployment of a new version of a component. In these cases, we sometimes encounter a series of failure messages for particular classes of components over a period of days or weeks. For example, we experienced a pattern of failure messages for RLS servers that corresponded to a configuration problem related to updates among the services. In another case, we experienced a series of SRM failure messages relating to a new feature that had unexpected behavior. Monitoring messages helped to identify problems with these newly deployed or reconfigured services. Conversely, the absence of these failure messages can in part validate a new configuration or deployment.

Failure deduction. The monitoring system can be used to deduce the reason for complex failures.

For example, we used MDS4 to gain insights into why the ESG portal crashed occasionally due to a lack of available file descriptors. By using the monitoring infrastructure to check file descriptor usage, we were able to detect how many file descriptors had been opened by the different services running on the portal and to eliminate some suspected sources of problems. We were also able to detect a sudden spike in file descriptor usage to help debug the problem.

In another example, on May 23rd in Table 3, failure messages indicated that SRMs at three different locations had failed simultaneously. Since the chance of such simultaneous independent failures is remote, we investigated and found a problem with a query expression in our monitoring software.

The monitoring infrastructure can detect failures that are not directly monitored. A single failure examined in isolation may not accurately reflect the state of the system or the actual cause of a failure. With system-wide monitoring data like that available through MDS4, a pattern of failure messages that occur close together in time can be used to deduce a problem at a different level of the system.

Certificate Problems: In another case, all the ESG services at the LANL site reported failures simultaneously. The problem was caused by the expiration of the host certificate for the ESG node at that site. Downtime resulted while the problem was diagnosed and while administrators requested and installed a new host certificate. To avoid this downtime in the future, we have implemented additional information providers and triggers that check the expiration date of host certificates on services where this information can be queried, such as the RLS. The Trigger Service checks these conditions periodically (once a day in our deployment) and informs system administrators when certificate expiration is imminent.

Scheduled Downtime: When a particular site has scheduled downtime for site maintenance, it is not necessary to send failure messages to system administrators regarding components and services at that site. We have developed a simple mechanism that disables particular triggers for the specified downtime period. The monitoring infrastructure still collects information about service state during this period, but failure conditions do not trigger actions by the Trigger Service.

6. Related Work

The MDS4 approach is similar to others used in the area of Grid monitoring systems. Its primary advantages are the use of standard interfaces, the flexibility of the information provider infrastructure, the robust high-level services, and notification of failures. It interacts with other tools to provide a standard interface to a wide variety of monitoring data.

Other Grid monitoring systems include R-GMA [20], MonALISA [21], and BDII [22].

R-GMA [20] is a monitoring framework that has strong ties to the EGEE (Enabling Grids for eScience Project) project [23]. Within R-GMA, all published monitoring data appears as if it were resident in a single, large, relational database.

MonALISA [21], or Monitoring Agents using a Large Integrated Services Architecture, is an agent-based monitoring framework for distributed systems. Agents cooperate on information gathering, processing, and analysis.

The Berkeley Database Information Index (BDII) [22] was originally developed by NIKHEF as part of the DataGrid project, and it has since been re-engineered by members of the Large Hadron Collider Grid (LCG) and EGEE projects at CERN. BDII uses two LDAP servers to collect information from Grid Index Information Services (GIISs) at Grid sites.

MDS2 [24] is an earlier implementation of the Globus Toolkit monitoring system based on LDAP. MDS2 deploys information providers on local resources, which report to a local Grid Resource Information Service (GRIS). GRISs report to higher level Grid Index Information Services (GIIS).

MDS4 has information providers that allow it to aggregate monitoring information from other tools that monitor the state of clusters or pools of nodes, such as Nagios [17], Hawkeye [18], and Ganglia [19].

Nagios [17] is a tool that periodically checks the status of hosts and services via a monitoring daemon and sends notifications to system administrators via email, instant messaging, or other methods when services fail. Nagios has a Web browser that provides current status information, logs and reports.

Hawkeye [18] is a monitoring system developed as part of the Condor distributed computing project [18]. Hawkeye configures Condor to periodically run scripts or other programs to monitor the state of resources. Hawkeye provides programs that monitor the amount of free disk space, available memory, network status, open file descriptors, process state, uptime, loads, etc. for Condor nodes. Output from these programs is added to the classified advertisements or *ClassAds* that describe the state of Condor resources. ClassAds can be queried to provide status of the distributed system.

Ganglia is a “scalable distributed monitoring system for high-performance computing systems such as clusters and Grids” [19]. The Ganglia architecture is hierarchical and is designed to support low overheads per monitored node and high concurrency. Ganglia is used to monitor thousands of clusters around the world.

Finally, Inca [25], part of the TeraGrid project, is primarily used for software stack validation and site certification. A central manager controls a set of information providers (called reporters) that are run at regular intervals, collecting data in an archive (called the depot). Inca does not gather cluster or queuing data for resource selection data at this time, nor does it provide failure warnings.

7. Summary and Future Work

For the past two years we have successfully used MDS4 to monitor the services across the seven ESG sites. As the climate community has come to depend on the ESG infrastructure as a critical resource, it has become increasingly important to monitor ESG components and services to detect failures quickly and minimize infrastructure downtime, An Index service provides up-to-date status information for ESG resources. The ESG portal queries this resource information and presents system-wide view of the Grid infrastructure. Finally, the MDS4 Trigger service polls the aggregated monitoring information to check the status of specified triggers and sends email messages to system administrators when service failures are detected.

The ESG monitoring infrastructure has been very effective in identifying failures quickly and reducing the periods when the Grid infrastructure is unavailable to users in the climate community. Over time, we plan to add to the richness of functionality provided by the ESG monitoring system. In the short term, we plan to implement a management interface that allows us to automate the specification of downtime and the ability to disable specific triggers. We will also extend the functionality for monitoring the expiration of host certificates.

As part of the planned longer-term development of MDS4, an Archive Service is being designed and implemented. This service will maintain historical information about monitored services. We plan to collect and mine long-term data about failures in ESG services to understand patterns and trends and to use this information to increase the robustness of the ESG.

Acknowledgments

This work was supported in part by the U.S. Department of Energy under the Scientific Discovery Through Advanced Computation (SciDAC) Program Grant DE-FC02-01ER25453. This work was also supported in part by the Mathematical, Information, and Computational Sciences Division subprogram of the Office of Advanced Scientific Computing Research, Office of Science, U.S. Department of Energy, under contract W-31-109-Eng-38. Additional support was provided by NSF NMI Award SCI-0438372. Some of the work was performed at the Oak Ridge National Laboratory, which is managed by UT-Battelle, LLC under Contract No. DE-AC05-00OR22725.

References

[1] D. Bernholdt, et al,, The Earth System Grid: Supporting the Next Generation of Climate Modeling Research, *Proceedings of the IEEE*, vol. 93, pp. 485- 495, 2005.
[2] J. M. Schopf, et al., Monitoring and Discovery in a Web Services Framework: Functionality and Performance of Globus Toolkit MDS4, in *Argonne National Laboratory Technical Report #ANL/MCS-P1315-0106*, 2006.
[3] Globus Alliance, The Globus Toolkit, http://www.globus.org/toolkit/, 2006.
[4] K. Czajkowski, et al., The WS-Resource Framework Version 1.0, http://www-106.ibm.com/developerworks/library/ws-resource/ws-wsrf.pdf, 2004.
[5] Community Climate System Model, http://www.cgd.ucar.edu/csm/, 2006.
[6] Parallel Climate Model, http://www.cgd.ucar.edu/pcm/, 2004.
[7] Intergovernmental Panel on Climate Change, http://www.ipcc.ch/, 2006.
[8] OPeNDAP Inc., OPeNDAP: Open-source Project for a Network Data Access Protocol, http://opendap.org/, 2005.
[9] A. Shoshani, Alex Sim, Junmin Gu, Storage Resource Managers: Middleware Components for Grid Storage, in Nineteenth IEEE Symposium on Mass Storage Systems (MSS '02), 2002.
[10] Globus Alliance, GridFTP, http://www.globus.org/toolkit/data/gridftp/, 2006.
[11] W. Allcock, et al., The Globus Striped GridFTP Framework and Server, in SC05 Conference, Seattle, WA, 2005.
[12] G. Singh, et al., A Metadata Catalog Service for Data Intensive Applications, in SC2003, 2003.
[13] E. Deelman, et al., Grid-Based Metadata Services, in 16th International Conference on Scientific and Statistical Database Management, 2004.
[14] Globus Alliance, The Replica Location Service, http://www.globus.org/toolkit/data/rls/, 2006.
[15] A. L. Chervenak, et al., Performance and Scalability of a Replica Location Service, in Thirteenth IEEE Int'l Symposium High Performance Distributed Computing (HPDC-13), Honolulu, HI, 2004.
[16] Grid Laboratory Uniform Environment (GLUE), http://infnforge.cnaf.infn.it/glueinfomodel/, 2006.
[17] Nagios, http://www.nagios.org/ 2006.
[18] Condor Project, Condor: High Throughput Computing, http://www.cs.wisc.edu/condor, 2003.
[19] M. L. Massie, et al., The Ganglia Distributed Monitoring System: Design, Implementation, and Experience, *Parallel Computing*, vol. 30, 2004.
[20] A. Cooke, et al., R-GMA: An Information Integration System for Grid Monitoring, in Proceedings of the 11th International Conference on Cooperative Information Systems, 2003.
[21] MonALISA: MONitoring Agents using a Large Integrated Services Architecture, http://monalisa.cacr.caltech.edu/monalisa.htm, 2006.
[22] Berkeley Database Information Index (BDII), http://lfield.home.cern.ch/lfield/bdii/documentation.html, 2006.
[23] EGEE, Enabling Grids for eScience Project, http://public.eu-egee.org/ 2006.
[24] X. Zhang, Jennifer M. Schopf, Performance Analysis of the Globus Toolkit Monitoring and Discovery Service, MDS2, in Proceedings of the International Workshop on Middleware Performance (MP 2004), 2004.
[25] S. Smallen, et al., The Inca Test Harness and Reporting Framework, in Proceedings of SC04 Conference, 2004.