Parallel Processing Letters

THE OPERATIONS DASHBOARD: A COLLABORATIVE ENVIRONMENT FOR MONITORING VIRTUAL ORGANIZATION-SPECIFIC COMPUTE ELEMENT OPERATIONAL STATUS

CATHERINE L. RUBY[†,‡], MARK L. GREEN[†], RUSS MILLER[†,‡]

[†]*Center for Computational Research, University at Buffalo*
[‡]*Department of Computer Science and Engineering, University at Buffalo*
Buffalo, NY 14260-1800, USA

Received May 24, 2006
Revised August 24, 2006
Communicated by S. Sahni

ABSTRACT

Grid computing integrates heterogeneous, geographically distributed, Internet-ready resources that are administered under multiple domains. A key challenge in grid computing is to provide a high quality of service to users in a transparent fashion, hiding issues that include ownership, administration, and geographic location of a wide variety of resources that provide compute cycles, data storage, rendering cycles, imaging devices, and sensors, to name a few. Ensuring the functionality of a wide variety of resources under multiple administrative policies requires tools for discovering, repairing, and publishing information on the services offered by individual sites within a given grid. In this paper, we present the ACDC Operations Dashboard, an interactive, collaborative environment for collecting, addressing, and publishing operational service information for resources across a computational grid.

Keywords: Grid computing, grid monitoring tools, heterogeneous grid.

1. Introduction

Advances in parallel computing have led to the development of *grid computing*, a model targeted at providing a ubiquitous system of multiple distributed and heterogeneous compute resources under distinct administrative domains to an end user. This infrastructure, typically applied to a wide range of computationally-intensive problems in a variety of disciplines, takes advantage of existing Internet-ready and geographically-dispersed devices and networks to provide a seamless infrastructure to facilitate computation and collaboration [1, 2].

The heterogeneous nature of a computational grid, particularly in the presence of multiple platforms, architectures, and administrative policies, introduces key challenges in providing seamless service to grid users. When grid components fail to provide essential operational services, it is imperative that grid administrators can troubleshoot the problems in a timely fashion. Note that individual sites may provide only a subset of the services offered to a user. In addition, some of the systems on a grid may be down, either for maintenance or due to an unforeseen problem. Regardless of the issue, it is

important that grid users and administrators have access to tools for discovering, diagnosing, repairing, and publishing critical information concerning the status of the grid.

To address these issues, the *Advanced Computational Data Center (ACDC) Operations Dashboard* [3] has been developed at the Center for Computational Research (CCR) as a part of its *ACDC Grid Monitoring Infrastructure*. The ACDC Operations Dashboard is a tool that allows one to view and evaluate operational service information for compute resources in a grid. Developed in conjunction with the ACDC Grid Dashboard [4], the Operations Dashboard tests the operational status of compute resources over the Virtual Organizations (VOs) [5] they support, and provides an interactive and collaborative online interface that organizes status information for grid users and site administrators.

In this paper, we present the ACDC Operations Dashboard, including (*i*) its operational component and (*ii*) its front-end collaborative interface for addressing service issues. In Section 2, we discuss the ACDC Grid Monitoring Infrastructure and the Grid Dashboard. In Section 3, we present the underlying service tests implemented to provide accurate site status information on remote computational resources. In Section 4, we show how operational tests provide VO-specific service information for individual compute resources. In Section 5, we present the Operations Dashboard online interface, and in Section 6, we show how a suite of Action Items provide a variety of interactive tools for addressing issues across grid initiatives and collaborating with site administrators. We discuss related work by other research initiatives in the area of grid monitoring in Section 7. We present our areas of continuing development and further remarks in Section 8.

2. ACDC Grid Monitoring

The Center for Computational Research has developed an extensive grid, as well as monitoring infrastructure for tracking the overall health and quality of service of grids and their individual compute elements [6, 7]. The CCR Grid Monitoring software is used to monitor resources from the Open Science Grid [8], the Open Science Grid Integration Test Bed [9], TeraGrid [10], and the ACDC Grid.

The ACDC Grid Monitoring Infrastructure executes a series of scripts designed to collect environmental information and status from the compute resources of the monitored grid systems. These scripts, run continually every few minutes on four 1.6GHz Intel® Xeon® processors, store this information in a multi-gigabyte MySQL [11] database.

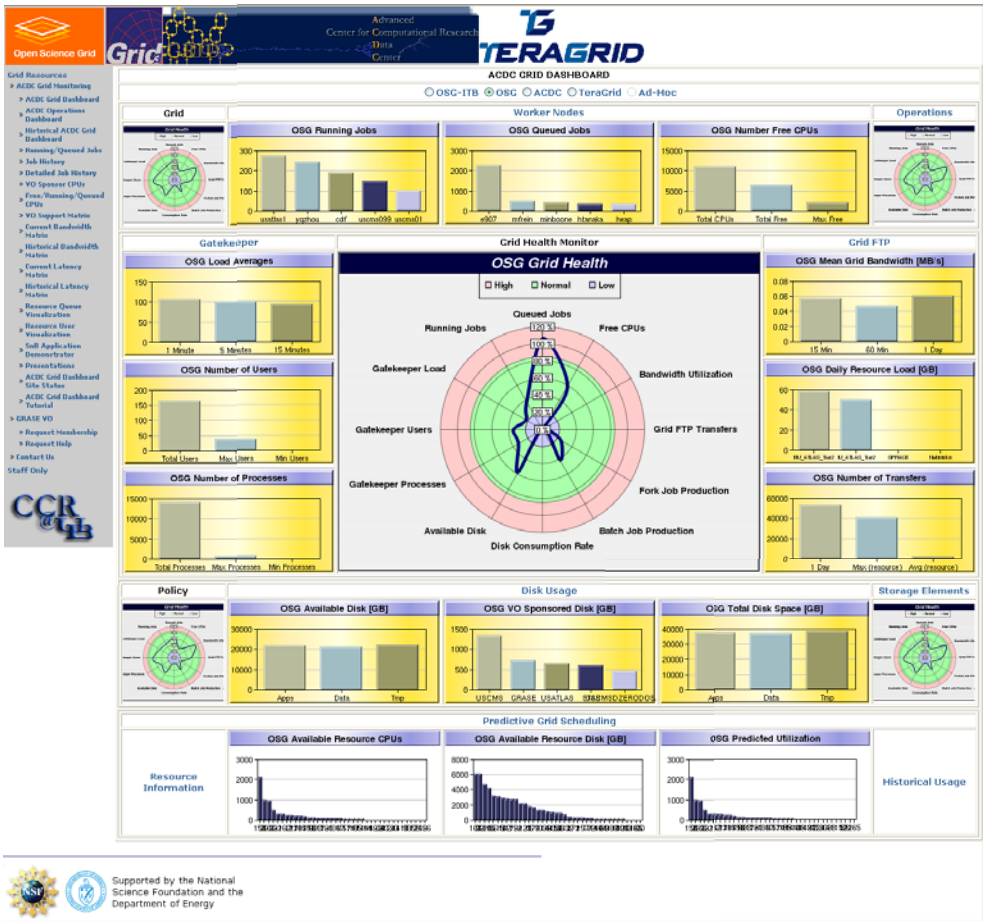


Fig. 1. ACDC Grid Dashboard.

The ACDC Grid Monitoring Infrastructure also provides a front-end visualization suite that showcases the data collected by the monitoring scripts. Served by an Apache HTTP Server [12] and written in the PHP hypertext preprocessor scripting language [13] and JavaScript [14], users are given access to hundreds of gigabytes of current and historical information through a series of dynamic charts, constructed on-the-fly from the MySQL database.

Beginning with the ACDC Grid Dashboard page, as shown in Fig. 1, users are presented with an overview of up-to-date statistics covering an entire grid. This information includes the number of jobs currently running and queued on sites across the grid, the amount of data being added or removed from all grid machines, and a substantial amount of additional information collected from the monitored sites.

An overall representation of the health of a grid is presented by a set of statistics that are aggregated in a radial plot in the center of the main page. By clicking on a region of the ACDC Grid Dashboard or on one of the links down the side of the page, users have the ability to drill down to view hundreds of dynamic, interactive charts of current and historical statistics for individual compute elements or VOs.

3. Site Status Verification

The operational status of a compute resource is determined by its ability to perform a variety of services for users of the computational grid. These services, ranging from simple connectivity and authentication to the ability to perform GridFTP [15] data transfers to and from a site, comprise the quality of service to expect from a compute element in a grid initiative.

The operational status presented by the ACDC Operations Dashboard is constituted by a series of short, incremental *Site Functional Tests*. These tests, implemented in Perl [16] and based on the site verification script by Dr. Craig Prescott of the University of Florida [17], visit many aspects of machine functionality and help administrators properly assess the operational status of compute elements. Tests run commands on remote resources using the Globus Toolkit [18], and verify a host's configuration in compliance with the standards defined by a grid initiative.

Tests are performed sequentially that verify ever more complex services available from the system under consideration. A suite of Site Functional Tests is run every few

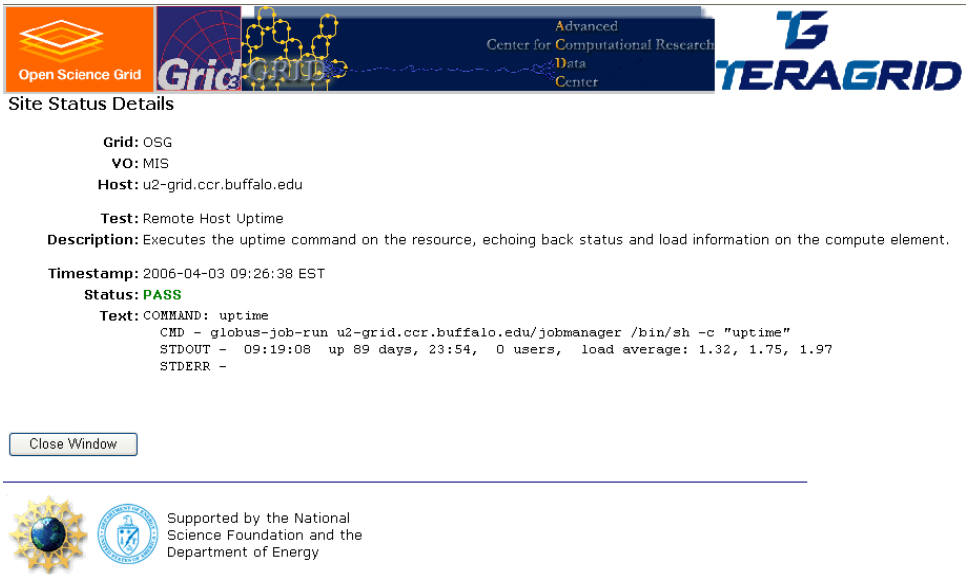


Fig. 2. Sample Site Functional Test results.

hours on a compute element using the ACDC Grid Monitoring Infrastructure. Site Functional Tests are sequenced with cascading dependencies, where if certain tests fail early in the sequence, tests that depend on it will be skipped. Though tests are typically run in this sequence to assess site operational status, they are designed to be completely modular, and with few exceptions can be run independently from one another at any time.

Site Functional Tests return results in one of five categories, namely, (*i*) no data, (*ii*) pass, (*iii*) error, (*iv*) failure, or (*v*) untested. A verification routine may have no data for a given compute element if this test has never been performed. Tests can pass or fail depending on the status of the compute element. An error status is returned if the routine cannot be executed by the ACDC Grid Monitoring Infrastructure or if the results returned are in an unexpected format. Tests are marked as untested if they are skipped because a supporting Site Functional Test failed previously. Test results are collected and stored in the MySQL database schema used by the ACDC Grid Monitoring Infrastructure.

The first four of the verification tests are labeled as *Critical Tests.* These ensure the most basic services on a compute element, namely, (*i*) that a connection to one of its ports can be established, (*ii*) that it is running a gatekeeper, (*iii*) that a grid user is able to authenticate, and (*iv*) that the fork job manager is able to execute a simple echo command. The failure of at least one of these critical tests indicates that the site is likely to be out of service for grid users and their computational jobs. Site administrators may opt to be notified when a site passes or fails the Critical Tests via the Action Items discussed in Section 6.

Fig. 2 contains sample test results from running an uptime verification routine on host u2-grid.ccr.buffalo.edu. This status module is intended to verify that Globus commands can be run on this remote site and provides a snapshot of the load on the compute resource. More detailed current and historical load information for compute elements is collected by the ACDC Grid Monitoring Infrastructure and can be viewed using the ACDC Grid Dashboard and its associated visualization tools, as discussed in Section 2.

4. VO-Specific Site Status

Grid users from many different Virtual Organizations rely on the operational status of grid compute elements in order to run their jobs. Configuration files on remote hosts contain information regarding the VOs that an individual site supports. It can be the case that due to configuration errors or other compute element-specific problems, a remote resource will appear to be operational for some VOs and not for others, making error isolation and diagnosis on a compute element a challenging task.

In order to overcome this issue and provide reliable and realistic operations information for all grid users and administrators, the ACDC Grid Monitoring Infrastructure runs Site Functional Tests for a number of VOs on each of the grids monitored. Proxies for each VO are used to verify services for each compute element supporting it, and individual VO-specific results are recorded for the Operations

Dashboard. The Operations Dashboard currently supports VOs that include Monitoring and Information Services (MIS) [19], Grid Resources for Advanced Science and Engineering (GRASE) [20], Fermi National Accelerator Laboratory (Fermilab) [21], Collider Detector at Fermilab (CDF) [22], Genome Analysis and Database Update (GADU) [23], Grid Laboratory of Wisconsin (GLOW) [24], U.S. ATLAS (USATLAS) [25], fMRI [26], International Virtual Data Grid Laboratory (iVDGL) [27], and nanoHUB [28].

Since a grid is typically used by multiple VOs, it is important to be able to monitor information specific to a VO on a given grid. For example, the MIS VO requires, among other things, the verification of the existence of the *Advanced Computational Data Center Resource Monitoring and Information Gathering Infrastructure* installation file, whereas other VOs do not use this and need not test for it. Similarly, the GRASE VO requires the verification of installed directories and storage space on individual compute elements that are not required by other virtual organizations.

To address this issue, the ACDC Grid Monitoring Infrastructure provides Site Functional Tests tailored to the individual services offered specific to a single VO. When site verification routines are executed for a compute element and a VO, these individual modules are also run and recorded for the Operations Dashboard.

5. The Operations Dashboard

The ACDC Operations Dashboard provides an interactive online interface, publishing the information collected for the Site Functional Tests. Updated by the ACDC Grid Monitoring Infrastructure and served along with the ACDC Grid Dashboard, the Operations Dashboard is comprised of several components providing detailed status information for over 150 compute elements.

The principle component of the Operations Dashboard is the *Site Resource – Service Matrix*. This graphical representation is dynamically generated when the page is loaded. It organizes operational status information into a matrix structure. Compute elements make up the rows of the matrix, and Site Functional Test results make up the columns. Row labels on the right side of the matrix indicate the last time the suite of Site Functional Tests was run for each compute element and VO. By choosing a grid and VO from the top of the page, a Site Resource – Service Matrix is generated on-the-fly from the MySQL database, containing all compute resources in the grid initiative which support the selected VO, and giving an overview of the quality of service that can be expected on grid sites for the virtual organization. Fig. 3 presents a screenshot of the Operations Dashboard, for the grid initiative OSG and MIS VO.

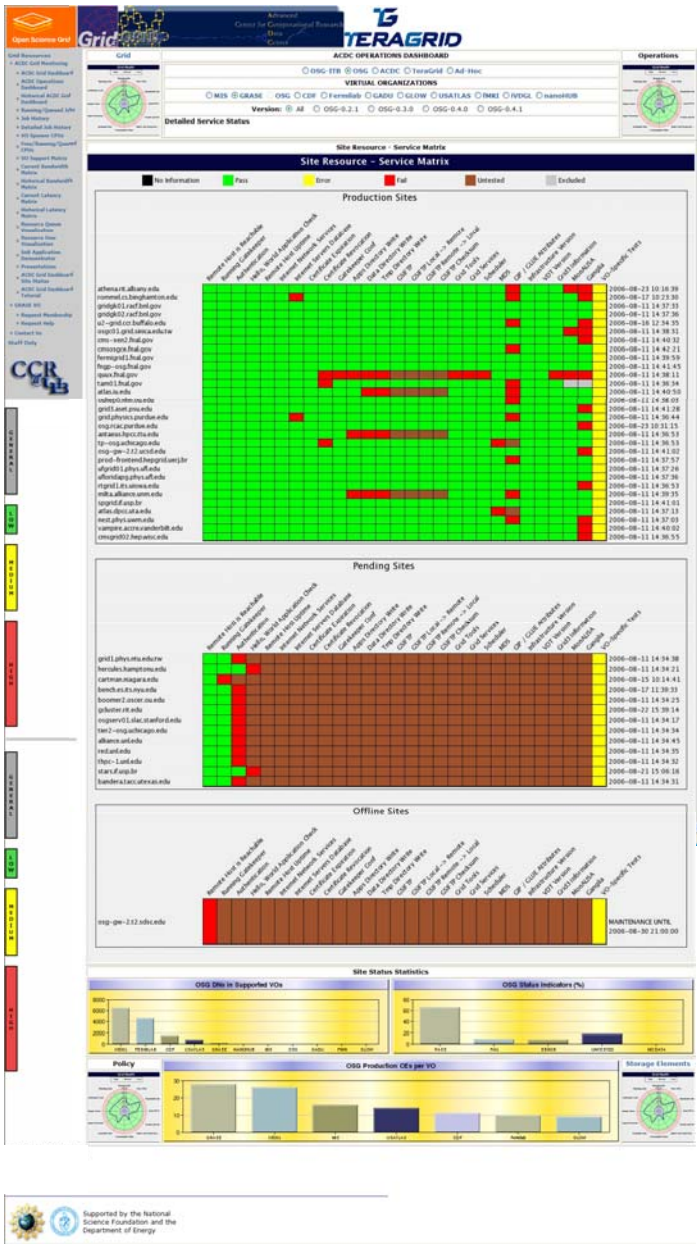
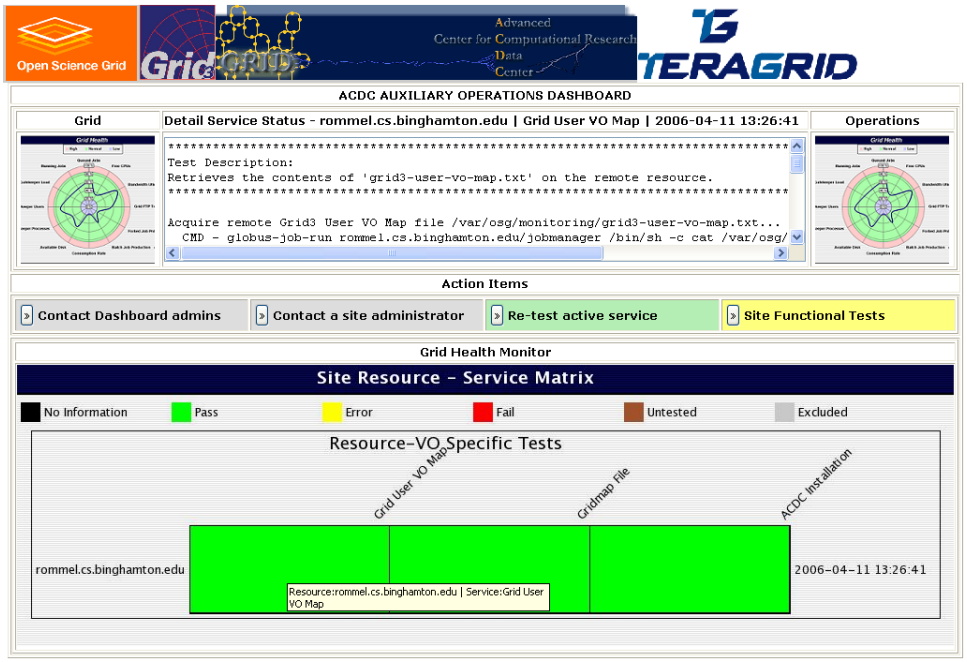


Fig. 3. The ACDC Operations Dashboard.

Each cell of the Site Resource – Service Matrix represents the latest results for the corresponding compute element and service test. The color of the cell indicates the test results for a quick at-a-glance summary of operational status for all compute elements. These color-coded indicators correspond to the five result categories for a Site Functional Test, as discussed in Section 3, and include (*i*) black, for no data available, (*ii*) green, for passing, (*iii*) yellow, for an internal monitoring system error, (*iv*) red, for failure, and (*v*) brown, if the test was skipped due to a previous failure. A sixth color indicator, grey, is used to designate a test that has been excluded for the compute element and selected VO.

In addition, each cell of the matrix is clickable for interactive access to more detailed operational information. By selecting a cell of the matrix, a new window is populated with the full text output of the Site Functional Test for the compute element and VO selected, such that specific errors can more easily be addressed on individual compute resources.

Compute elements in the Site Resource – Service Matrix are divided among three categories, namely, (*i*) production, (*ii*) pending, and (*iii*) offline. Sites that have passed the



Supported by the National Science Foundation and the Department of Energy

Fig. 4. The Auxiliary Operations Dashboard.

first four Critical Tests are considered to be "production" resources, as they are likely to be accessible and usable by users of the selected VO. If a site has failed at least one of the Critical Tests, it is labeled "pending" for the selected VO as a flag to grid administrators that problems on the compute element must be addressed. Sites are automatically assigned to production or pending status based on their performance in the Critical Tests when the Site Functional Tests are run. Site administrators also have the option of manually setting their sites to offline for a defined period of maintenance by using one of the Action Items, discussed in more detail in Section 6.

The VO-specific Site Functional Tests, verifying information relevant to certain VOs, are published using the Auxiliary Operations Dashboard, as shown in Fig. 4. This utility, accessible by clicking on the "VO-Specific Tests" column within the Site Resource – Service Matrix, displays a similar Site Resource – Service Matrix for the selected compute element and the VO-specific Site Functional Tests for the chosen VO.

6. Action Items and Collaboration

The clickable interface provided through the main ACDC Operations Dashboard is further enhanced by the availability of *Action Items*, a series of tools designed to provide greater user control and facilitate collaboration to address service issues and ensure the quality of service of compute elements across a grid initiative. Available through sliding menus along the side of the ACDC Operations Dashboard and organized in multi-tiered SSL authentication based on grid user certificates, Action Items provide a wide variety of capabilities for grid users and administrators.

Action Items are divided among four categories, namely, (*i*) general, (*ii*) low, (*iii*) medium, and (*iv*) high, according to the significance to the task with respect to the Operations Dashboard. For example, viewing historical status information or sending e-mail queries to dashboard administrators are lighter tasks and available to all users of the Operations Dashboard under the general category. Updating the VO proxies used for executing Site Functional Tests, on the other hand, is restricted to a small subset of users with high privileges, identified by the certificates in their browsers.

In addition, some Action Items are further restricted to ensure that only a site's administrators may update information for a compute element. For example, users with medium access are permitted to update compute element contact information, identify other users as site administrators, and request automatic notifications for their own site only, even though it is designated as a medium level Action Item. Site administrators are identified by their browser certificates, and are mapped to a group of approved users for each compute element within our MySQL records.

There are several Action Items in place that allow users to publish information to the Operations Dashboard for a compute element. If a remote host is required to be down for maintenance or repair, site administrators may use the Action Items to set a compute element to offline status for a specified period of time. For the duration of the offline maintenance period, automatic site verifications will not be run and all users of the

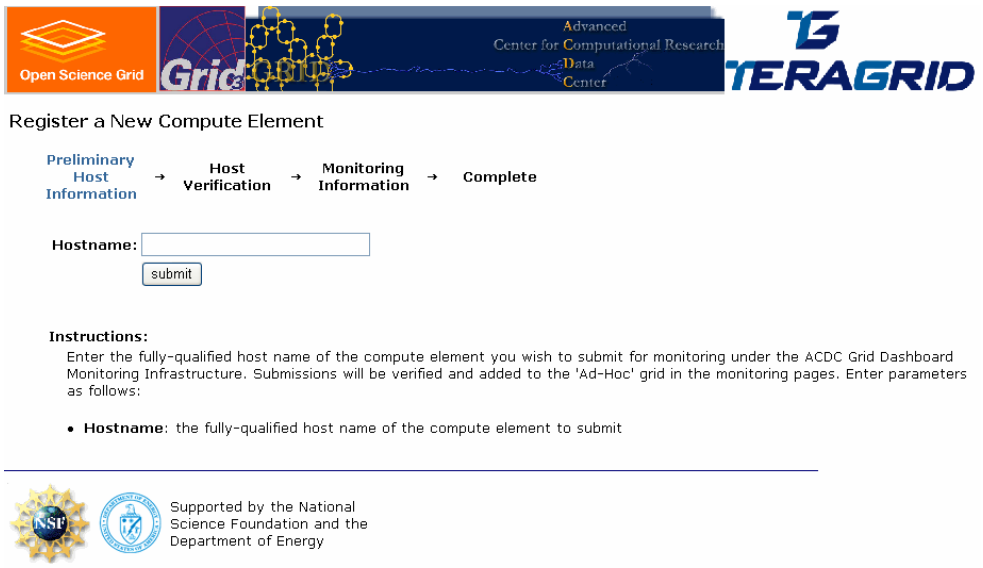


Fig. 5. Registering a new compute element

dashboard will see the site listed as being offline. Site administrators may also exclude specific Site Functional Tests from the status update routines for the Operations Dashboard, indicating both to the routines and to grid users that the service is not available on the compute resource. As the Site Resource – Service Matrix is generated on-the-fly from the latest information in the MySQL database, any changes made through the Action Items are visible immediately to all Operations Dashboard users.

In addition, users may initiate Site Functional Tests on compute resources displayed within the Operations Dashboard, updating service statistics to the latest information. For example, if an administrator repairs a service on a compute element for a VO, they may run the corresponding Site Functional Test immediately to notify all Operations Dashboard users that the service is operational once again.

New sites may also be submitted for automatic Operations Dashboard site verification using Action Items. Fig. 5 illustrates the beginning of the procedure for adding a new remote host. Once the host is verified to be a valid site and the appropriate administrative information is given, the site is added and will be rolled into the automatic site verification process and visible by grid users under the virtual organizations supported by the remote host.

Fig. 6. E-mailing a site administrator.

The interactive capabilities provided by the Action Items also present collaborative tools for troubleshooting problems and ensuring essential services for grid users. For example, grid users may alert site administrators to service errors and discuss potential solutions by contacting administrators directly via e-mail. As shown in Fig. 6, this Action Item provides contact information for administrators of the compute element, the full text of the latest Site Functional Test output for the service selected, and the ability to comment on the issue, facilitating collaboration between administrators and users in order to address problems and resolve them in a timely fashion.

7. Related Work

The work done in developing the ACDC Grid Monitoring Infrastructure is part of a widespread initiative in the distributed computing community to develop monitoring tools to manage distributed systems. These initiatives vary widely, from the method in which information is discovered and collected, to the interface provided to present information to grid users.

The process of discovering information within a monitoring service is achieved in a variety of ways. NetLogger [29] utilizes Unix monitoring tools for host and network sensors as well as manual invocations to its API from running applications, generating a log of individual events on the remote site. Other initiatives, such as the Grid Resource Monitoring (GridRM) [30] project, monitor sites by providing a single means of accessing the data collected by common grid monitoring tools, such as Ganglia [31] and the Network Weather Service [32]. Implementations and APIs of monitoring initiatives range from C [33], C++ [34], and Java [35], which are common to many monitoring tools, to languages like Perl, used by NetLogger, MDS2 [36], and in the implementation of the underlying testing infrastructure for the Operations Dashboard [37].

Collecting monitored data from sites is another challenge in a distributed, heterogeneous grid environment. The Operations Dashboard stores data locally from the results of established socket connections and Globus commands to remote sites. The Mercury project under GridLab [38] follows a different approach, installing monitors on each local site that sends results directly to a central monitor, serving requests from the monitoring service and other information consumers. Whereas some monitoring systems, such as Ganglia, use broadcast models to collect monitored information from remote sites, other systems such as MonALISA (Monitoring Agents using a Large Integrated Services Architecture) [39] store information on monitored resources and make it available for visualization through its GUI front-end [37].

Monitoring development projects have also yielded a variety of user interfaces for publishing status information. GridCat [40], created under iVDGL, displays status information as a geographic map of physical resource locations, with color-coded dots symbolizing the status of worldwide grid machines. GridICE [41] presents monitored results both in a web-based interface with text and graphics, as well as an XML representation over HTTP. Like the Operations Dashboard, GridICE presents monitored data classified by grid operational center as well as by VO, in a web interface implemented in PHP [37, 41]. The Network Weather Service provides a graphical interface that aggregates data to provide both current network statistics and a forecasting model of expected performance on the monitored network [37].

Though many monitoring initiatives provide detailed and accurate site status through a variety of interfaces, the ACDC Operations Dashboard capitalizes on the concept of grid user collaboration. Other monitoring initiatives, such as MapCenter [42], provide users with the capability of directly pinging resources or querying their MDS servers. The OCM-G (OMIS Compliant Monitor) [43] provides services such as retrieving events, on-

the-fly performance enhancements of applications, and the activation of responses to detected events [37]. However, the interactive nature of the Operations Dashboard interface provides users with the tools for not only consuming status information but using it to alert administrators to potential problems, troubleshoot and retest resources for instant results, and refresh current machine configuration information to present the most up-to-date snapshot of resource functionality. In addition, administrators may publicly mark Site Functional Tests as unavailable on their resources, alerting grid users to the services to be expected from compute elements they may plan to use.

The distributed nature of grid resources and the existence of multiple machine platforms and administrative domains necessitate the availability of tools for interacting with monitored data and facilitating collaboration between site users and administrators. The interactive, collaborative focus of the Site Resource - Service Matrix and Action Items is the hallmark of the Operations Dashboard and the ACDC Grid Monitoring Infrastructure.

8. Further Remarks

The ACDC Operations Dashboard testing infrastructure and interface described in this paper provides a lightweight, interactive, and collaborative environment for serving accurate operational status information. Modular Site Functional Tests run on, or tailored to, individual virtual organizations create a repository of VO-specific service information across multiple grid initiatives, providing finer control in dispensing status information for compute elements to grid users and administrators. Integrating flexible site functional tests with an interactive online interface and a variety of Action Items provides a powerful tool for publishing and addressing issues across a grid initiative, and ensuring the quality of service offered by available compute resources.

New site functional tests and Action Items are continually being developed to improve the quality of information provided and further extend the interactive capabilities of the interface. In addition, plans have been made to extend the work presented here to an ACDC Storage Elements Dashboard, designed to provide quality of service information for dedicated storage devices running a Storage Resource Manager (SRM) [44] for file and storage space management.

Acknowledgments

We would like to thank Steven M. Gallo and Jonathan J. Bednasz for their contributions to this work. The development of the Operations Dashboard is supported by NSF grant ACI-0204918 and the Center for Computational Research, SUNY-Buffalo.

10. References

[1] The Grid: Blueprint for a New Computing Infrastructure. Second ed, ed. I. Foster and C. Kesselman. 2004: Morgan Kaufman.

[2] Foster, I., The Grid: A new infrastructure for 21st century science, in Grid Computing - Making the Global Infrastructure a Reality, F. Berman, G.C. Fox, and A.J.G. Hey, Editors. 2002, John Wiley and Sons Ltd.: West Sussex. p. 50-63.

[3] ACDC Operations Dashboard. [webpage] 2006 [cited 2006 April 19]; Available from: http://osg.ccr.buffalo.edu/operations-dashboard.php.

[4] ACDC Grid Dashboard. [webpage] 2006 [cited 2006 April 19]; Available from: http://osg.ccr.buffalo.edu/.

[5] Foster, I., C. Kesselman, and S. Tuecke, The Anatomy of the Grid - Enabling Scalable Virtual Organizations. International Journal of Supercomputer Applications, 2001. 15(3): p. 200-222.

[6] Grid Computing at CCR. [webpage] 2004 November 3, 2004 [cited 2006 April 6]; Available from: http://www.ccr.buffalo.edu/grid/.

[7] ACDC Grid Computing Services. [webpage] 2006 April 1, 2006 [cited 2006 April 6, 2006]; Available from: http://grid.ccr.buffalo.edu.

[8] Open Science Grid. [webpage] 2006 April 13, 2006 [cited 2006 April 13]; Available from: http://www.opensciencegrid.org.

[9] OSG Integration Testbed. [webpage] 2005 March 10, 2005 [cited 2006 May 10]; Available from: http://www.ivdgl.org/osg-int/.

[10] TeraGrid. [webpage] 2006 April 13, 2006 [cited 2006 April 13]; Available from: http://www.teragrid.org.

[11] MySQL AB: The world's most popular open source database. [webpage] 2006 April 6, 2006 [cited 2006 April 6]; Available from: http://www.mysql.com/.

[12] Apache HTTP Server. [webpage] 2006 2006 [cited 2006 April 6]; Available from: http://www.apache.org/.

[13] PHP: Hypertext Preprocessor. [webpage] 2006 April 6, 2006 [cited 2006 April 6]; Available from: http://www.php.net.

[14] JavaScript.com (TM) - The Definitive JavaScript Resource. [webpage] 2006 2006 [cited 2006 April 6]; Available from: http://www.javascript.com/.

[15] Allcock, B., et al., Data Management and Transfer in High-Performance Computational Grid Environments. Parallel Computing, 2002. 28(5): p. 749-771.

[16] perl.com - The Source for Perl. [webpage] 2006 [cited 2006 April 28]; Available from: http://www.perl.com.

[17] Prescott, C., Site Verify. 2005: http://osg-docdb.opensciencegrid.org/cgi-bin/ShowDocument?docid=83.

[18] Foster, I. and C. Kesselman, Globus: A Metacomputing Infrastructure Toolkit. International Journal of Supercomputer Applications, 1997. 11(2): p. 115-128.

[19] MIS VO Privilege Page. [webpage] [cited 2006 May 11]; Available from: http://www.ivdgl.org/~ivdgl/mis-vo-privilege.html.

[20] Grid Resources for Advanced Science and Engineering (GRASE). [webpage] [cited 2006 May 11]; Available from: http://osg.ccr.buffalo.edu/grase/.

[21] Fermilab. [webpage] [cited 2006 May 11]; Available from: http://www.fnal.gov/.

[22] The Collider Detector at Fermilab. [webpage] [cited 2006 May 11]; Available from: http://www-cdf.fnal.gov.
[23] GADU VO at Open Science Grid. webpage [cited 2006 May 11]; Available from: http://compbio.mcs.anl.gov/gaduvo/gaduvo.cgi.
[24] GLOW :: Grid Laboratory of Wisconsin. [webpage] January 23, 2003 [cited 2006 May 11]; Available from: http://www.cs.wisc.edu/condor/glow/.
[25] U.S. Atlas. [webpage] [cited 2006 May 11]; Available from: http://www.usatlas.bnl.gov/.
[26] fMRI Data Center: High Performance Compute Center. [webpage] February 18, 2004 [cited 2006 May 11]; Available from: http://www.fmridc.org/f/fmridc/index.html.
[27] International Virtual Data Grid Laboratory. [webpage] [cited 2006 May 11]; Available from: http://www.ivdgl.org/.
[28] nanoHUB - Online Simulations and More. [webpage] [cited 2006 May 11]; Available from: http://www.nanohub.org.
[29] Tierney, B., et al. The NetLogger Methodology for High Performance Distributed Systems Performance Analysis. in IEEE High Performance Distributed Computing Conference. 1998.
[30] Baker, M.A. and G. Smith. GridRM: A Resource Monitoring System for the Grid. in Grid2002, 3rd International Workshop. 2002. Baltimore, MD: Springer.
[31] Massie, M.L., B.N. Chun, and D.E. Culler, The Ganglia Distributed Monitoring System: Design, Implementation, and Experience. Parallel Computing, 2004. 30(7): p. 817-840.
[32] Wolski, R., N. Spring, and J. Hayes, The Network Weather Service: A Distributed Resource Performance Forecasting Service for Metacomputing. Journal of Future Generation Computing Systems, 1999. 15(5-6): p. 757-768.
[33] C Programming Language. [webpage] 2006 [cited 2006 August 22]; Available from: http://en.wikipedia.org/wiki/C_programming_language.
[34] C++. [webpage] 2006 [cited 2006 August 22]; Available from: http://en.wikipedia.org/wiki/C++.
[35] Java Technology - The Source for Java Developers. [webpage] 2006 [cited 2006 August 22]; Available from: http://java.sun.com/.
[36] GT Information Sciences: Monitoring & Discovery System (MDS). [webpage] 2006 [cited 2006 August 22]; Available from: http://www.globus.org/toolkit/mds/.
[37] Zanikolas, S. and R. Sakellariou, A Taxonomy of Grid Monitoring Systems. Future Generation Computer Systems, 2005. 21(1): p. 163-188.
[38] GridLab: A Grid Application Toolkit and Testbed. [webpage] 2005 [cited 2006 August 22]; Available from: http://www.gridlab.org/WorkPackages/wp-11/.
[39] Newman, H.B., et al., MonALISA: A Distributed Monitoring Service Architecture, in Conference for Computing in High Energy and Nuclear Physics. 2003: La Jola, California.
[40] GridCat Development. [webpage] 2006 [cited 2006 August 22]; Available from: http://www.ivdgl.org/gridcat/home/index.html.
[41] Andreozzi, S., et al., GridICE: a Monitoring Service for Grid Systems. Future Generation Computer Systems, 2005. 21(4): p. 559-571.

[42] Bonnassieux, F., et al. MapCenter: An Open Grid Status Visualization Tool. in Parallel and Distributed Computing Conference. 2002. Louisville, Kentucky.
[43] Balis, B., et al. Monitoring Grid Applications with Grid-enabled OMIS Monitor. in First European Across Grids Conference. 2003. Santiago de Compostela, Spain: Springer.
[44] Shoshani, A., A. Sim, and J. Gu, Storage Resource Managers: Essential Components for the Grid, in Grid Resource Management: State of the Art and Future Trends, J. Nabrzyski, J.M. Schopf, and J. Weglarz, Editors. 2003, Kluwer Publishing. p. 329-347.