The Astrophysics Simulation Collaboratory Portal: a Framework for Effective Distributed Research

Ruxandra Bondarescu [a,b], Gabrielle Allen [c], Gregory Daues [a], Ian Kelley [c], Michael Russell [c], Edward Seidel [c,a,b], John Shalf [d], Malcolm Tobias [e]

[a] *National Center for Supercomputing Applications, Champaign, IL USA*

[b] *University of Illinois at Urbana-Champaign, Department of Physics and Astronomy, Champaign, IL USA*

[c] *Max Planck Institute for Gravitational Physics, Albert Einstein Institute, Potsdam, Germany*

[d] *Lawrence Berkeley National Laboratory, Berkeley, CA USA*

[e] *Washington University, St. Louis, MO USA*

Abstract

We describe the Astrophysics Simulation Collaboratory (ASC) portal, a collaborative environment in which distributed projects can perform research. The ASC project seeks to provide a web-based problem solving framework for the astrophysics community to harness computational grids. To facilitate collaboration amongst distributed researchers within a virtual organization, the ASC Portal supplies specialized tools for the management of large scale numerical simulations and the resources on which they are performed. The ASC Virtual Organization uses the Cactus framework for studying numerical relativity and astrophysics phenomena. We describe the architecture of the ASC Portal and present its components with emphasis on elements related to the Cactus framework.

Key words: Grid Computing, Globus, Cactus, Astrophysics Simulation Collaboratory Portal, GridLab

Email addresses: ruxandra@ncsa.uiuc.edu (Ruxandra Bondarescu), gabrielle.allen@aei.mpg.de (Gabrielle Allen), daues@ncsa.uiuc.edu (Gregory Daues), ikelley@aei.mpg.de (Ian Kelley), michael.russell@aei.mpg.de (Michael Russell), edward.seidel@aei.mpg.de (Edward Seidel), jshalf@lbl.gov (John Shalf), mtobias@wugrav.wustl.edu (Malcolm Tobias).

1 Introduction

In a variety of scientific disciplines, complex research problems are being investigated by geographically distributed collaborations. Computational grids are emerging as a distributed computing infrastructure that will allow these collaborations to seamlessly access computational resources and perform large scale numerical simulations. The Cactus framework (`www.cactuscode.org`) and Computational Toolkit (1), a scientific code developed at the Albert Einstein Institute and used throughout the world, is a step towards providing a common computational environment in the astrophysics community. Cactus was initially designed for solving the equations governing Einstein's Theory of General Relativity, but is presently used in various scientific disciplines such as bioinformatics and atmospheric sciences. Cactus enables research into a large variety of physical problems and is portable across all computer architectures. It has emerged as an important tool for groups of researchers performing large scale numerical simulations. While numerous scientific groups are benefiting from the features of Cactus, effective use of this code on diverse and distributed computational resources would benefit from a connection between the scientists, their simulations and emerging grid services (2). Such a connection is provided by the simulation management framework of the Astrophysics Simulation Collaboratory (ASC) portal (3).

The ASC Portal is intended to deliver a collaborative simulation management framework for generic applications, with the development driven by a particular community of astrophysicists, numerical relativists, and computational science researchers that use and develop their research codes with Cactus. This community makes up a virtual organization (VO) (4), denoted as the ASC Virtual Organization (ASC VO). End user involvement from this VO has been a crucial factor in providing truely useful and appropriate working environment through the ASC Portal, both in terms of core functionalities and also in the design of the user interface. The ASC Portal codebase serves as a solid foundation for extensions to the next generation portlet and services frameworks, such as that being developed by the Portal Group of the GridLab project (5).

An important recent aspect of the ASC development is a collaboration between the ASC and the new European GridLab project (`www.gridlab.org`). The ASC project involves the Washington University Gravity Group, the University of Illinois at Urbana-Champaign/National Center for Supercomputing Applications, the University of Chicago/Argonne National Laboratory, Rutgers University, and the Albert Einstein Institute in Potsdam, Germany. The GridLab project is a European funded collaboration involving eight countries in Europe and three institutes in the US along with industrial partners Compaq and Sun. The GridLab and ASC projects both involve components aimed

at solving the common problem of managing simulations on various supercomputers across the world; the web portal technology is a key component of the solution to this problem. The GridLab project is extending the ASC technology to a wider community of researchers (6).

The ASC Portal has prototyped and tested many different grid and collaborative capabilities (7; 8), and this paper primarily focuses on those aspects concerned with simulation management which are now being used by the researchers of the ASC VO. Section 2 describes the basic architecture of the ASC Portal, Section 3 describes how computational resources are managed and accessed, Section 4 illustrates how simulations are staged and monitored, Section 5 describes the ASC Portals role at SC2002 and Section 6 outlines future directions and plans following from this work.

2 Portal Architecture

The ASC Portal is an instance of an emerging class of Web Portals, termed Grid Portals, that utilize grid technologies. The grid infrastructure with which the portal interacts is constructed with Globus (9) middleware. In this setting, the portal application typically comprises several distinct layers consisting of multiple clients, an application server, and a collection of application services maintained on the portal host. The application server is a central, continually active web application that handles user requests from web-based clients. The ASC application server is written in Java and employs Java Servlets and Java Server Pages technology to deliver content in the form of DHTML pages and Java applets to web browser clients. Furthermore, the ASC Portal can also interact with clients such as cellular telephones and other handheld devices.

Deployment of the application server as a web application is accomplished within the open source Tomcat servlet container of the Apache Jakarta Project. To reduce the amount of data the application server must keep in memory, the portal uses a MySQL relational database system for persistent storage of information on users, accounts, resources, code configurations, simulations, *etc.* The application services on the portal host connect to the Grid through the Java Commodity Grid (CoG) Toolkit (10). The Java CoG interfaces with services on remote resources through standard Grid protocols, which are a part of the Globus Toolkit. Jobs are submitted to Globus gatekeepers through the Globus Resource and Management (GRAM) protocol. Mutual authentication between hosts is accomplished via the Grid Security Infrastructure (GSI), and file transfers to GSI-enabled FTP servers are achieved using GridFTP.

The infrastructure on which grid portals operate continues to evolve rapidly. Core Globus components are being reformulated as independent Grid services.

The ASC Portal presently interacts with higher level services such as a resource broker which facilitates the running of simulation codes on a working grid of machines. The incorporation of additional higher level services will facilitate data management, and enable users to execute elaborate work flows through the portal interface.

Various elements of the ASC Portal are being migrated to a Portlet framework, called GridSphere (`www.gridsphere.org`), currently being developed by the GridLab project. In this framework, individual components become independent modules that are easily incorporated into other projects. The portlet framework can group portlets into independent web applications simplifying development within large collaborations. Furthermore, the portlet interface brings enhanced customization, giving each user a personalized display. The migration of ASC components into the GridLab portlet framework will allow many different and diverse communities to benefit from this technology.

3 Resource Management

The portal currently stores information about computational resources and Globus services running on remote resources within its own persistence layer. To allow current portal users to work with machines with minimal deployment of Grid infrastructure, this information is entered by portal administrators through portal web interfaces. The ASC Portal does include tools for querying information about remote resources with MDS. However, since much of the information that MDS provides is essentially static, we found it useful to cache this information within the ASC Portal database. This provides the additional benefit of enabling developers to more easily relate information about other entities to Grid resources. For example, we can very easily test and record the results of whether a user's Grid credentials authenticate to the Grid services running on the machines made accessible via the portal. This information can be used to personalize a user's view of the Grid.

Certainly, our target user communities are very interested in receiving dynamic information about machines, especially in regards to queue availability for job submission. Currently, it is our experience that the Grid infrastructure technologies that support information publishing and retrieval mechanisms are either not often properly configured for production use or otherwise not yet available on the resources of particular interest to our user communities. In fact, this is partly what lead to the construction of the GridLab Project with which the ASC remains affiliated. GridLab contains several work packages that try to insure that a proper infrastructure is in place for making production Grid computing possible.

For secure access to web portal sessions and computational resources, the ASC Portal makes use of digital Grid certificates and proxies. These certificates allow a researcher, after an initial authentication, to move from one Grid resource to the next without re-establishing identity, i.e., they provide single sign-on access. Grid certificates are obtained from trusted Certificate Authority such as the NCSA Alliance or the NPACI partnership. Users first establish their identity to the Certificate Authority, whereupon the organization issues certificates and private keys. From the certificate/key pair the researcher can generate proxy credentials of a chosen finite lifetime that enable the single sign-on functionality. These credentials are stored in a repository server.

The ASC Portal utilizes a special online credential repository, the MyProxy (11) repository, to securely access proxy credentials. Portal users store their proxy credentials onto designated MyProxy servers, using MyProxy client software external to the ASC Portal. This task must be performed occasionally, e.g., once every seven days, giving a balance between security (the credentials are transient, with a finite lifetime) and convenience. The user logs onto the ASC Portal using the login name and password that protects their stored proxy credentials. The portal retrieves the credentials from the MyProxy server, enabling Grid resources freely and securely to be accessed without remembering or retyping different usernames and passwords. The instances of the proxy credentials that are obtained for work within the portal are usually short-lived, lasting for the duration of a typical user session. Advanced credential management will monitor the status of user proxies and perform proxy renewals when necessary. This feature will be extremely useful in the case of submissions to batch systems that do not start running within the proxy lifetime. Without an active proxy, the job would be unable to perform grid based operations. The single sign-on access allows researchers to work on a geographically distributed Grid of machines in the same manner that they would work on a single local machine.

3.2 Job Staging

Need a short, 1 paragraph section here that covers job staging, including (1) GRAM interface and use for submitting jobs, and (2) mentioning that distributed jobs, using multiple resources, can even be successfully submitted via the portal. This will add a small amount to the paper, but figures can be resized a bit to compensate.

The various platforms encountered in a grid of machines often require different resource management techniques. In a production environment it becomes practical for users to group resources to which they can authenticate. One can use ASC Portal interfaces to define different working groups of machines. For example, a scientist can have a group of machines on which a certain simulation code is known to run well. Within each given portal session, a particular working group may then be selected. The resource groups provide clean working interfaces, avoiding a large list of possible resources that may be unavailable or unusable to the researcher and would clutter numerous interfaces. This feature also simplifies the compatibility problem from resources belonging to different virtual organizations that accept different grid certificates.

4 Simulation Management

Performing large scale numerical simulations with advanced application codes on grid resources is complicated by various factors. Computational grids are typically heterogeneous, comprised of supercomputers of various architectures and platforms that have different installed software and present different environments to users. Furthermore, the resources are maintained by diverse virtual organizations. The collaborative group work space provided by the ASC Portal intends to facilitate simulation management in a complex grid environment. For effective collaboration, ASC members need a uniform simulation management framework containing the core elements: (1) source code management, (2) simulation tracking, (3) an executable repository.

4.1 Source Code Management

One of the key features of the Cactus framework is its modular structure. Cactus has a central core (called the "Flesh") into which independent application and computational infrastructure modules (called "Thorns") can be interfaced cleanly. The design provides the end user with the flexibility to choose the most appropriate set of thorns to perform their particular application. This feature is important for large, geographically distributed collaborations of physicists and computational scientists with the various members contributing independent elements of the overall simulation code. As improved or additional thorns become available they can be incorporated and leveraged immediately. A production simulation code can consist of many different thorns from diverse sources, and while this greatly enhances the capabilities of the simulation

code, it creates a challenge in the area of source code management. The numerous elements of the production simulation code must be gathered from various source code repositories. The management of these elaborate source code configurations becomes a problem for both individual researchers and collaborators within a research group. The ASC Portal provides useful tools and a working environment that facilitate source code management.

An important piece of the portal's source code management environment is listings the contents of Concurrent Versioning System (CVS) source code repositories which contain Cactus thorns Portal administrators may easily enter the listings of group shared repositories, and individual users may enter the contents of their personal repositories, making them shareable if they so choose. Portal users can perform multiple CVS checkouts to construct a source code configuration on a grid resource. The contents of the code configurations are recorded by the portal, enabling users to easily move an entire working setup from one machine to another. Members of a collaboration can share and distribute their code configurations amongst each other, saving valuable time and preventing duplication errors. The configurations of standard test problems of a research group may also be stored in the portal. By maintaining the source code work history of a collaboration, the portal provides better organization, extending research capabilities.

Future enhancements to the source code handling apparatus will address a variety of issues that occur when developing scientific codes to be run on a Grid. In a typical scenario, development starts by editing a code configuration on a base machine (e.g., a personal laptop). Once the local development is complete, the developer must verify that the code configuration is portable to the various architectures and operating systems that make up the Grid of machines. The portal will provide the service of automatically checking that the code configuration still compiles, executes, and passes basic unit tests successfully on a heterogeneous Grid of machines. The portal will also provide a mechanism for users to share code under development via patches.This functionality will (1) speed the rate at which updates to a configuration can be made and (2) improve the stability and portability of code configurations.

4.2 Simulation Tracking

Large scale simulations are very expensive from a computational viewpoint. The high cost of numerical simulations imposes a limit on the number of large runs a research group can expect to perform. The results obtained are very valuable and are often studied by many different scientists within the community. A large scale numerical simulation submitted to a remote machine may start only after an hour, a day, or a week. When the simulation starts, the

only available information may be that the job is active, and this information is not enough to judge the performance or the quality of results of the simulation. Under these circumstances, following the progress of the simulation is vital, but usually inconvenient or impossible.

The ASC Portal enables effective monitoring of numerical simulations by listening to messages from the application itself. Scientists who login to the ASC Portal see a complete list of their own simulations and those of other members of their VO (Figure 1). The portal maintains a group workspace indexed by project name. The history of the VO's simulations on a given project are collected and displayed to all authorized scientists. The ASC Portal displays simulation information such as the machine where the simulation is running, the name of the scientist running the simulation, the date of the initial report and the location of the data being created. Periodically, the Cactus simulation updates the portal records by providing the current iteration number and an estimate of the time to completion. Portal users can provide their Email and mobile telephone number, and can choose to receive status messages and updates about their simulations.

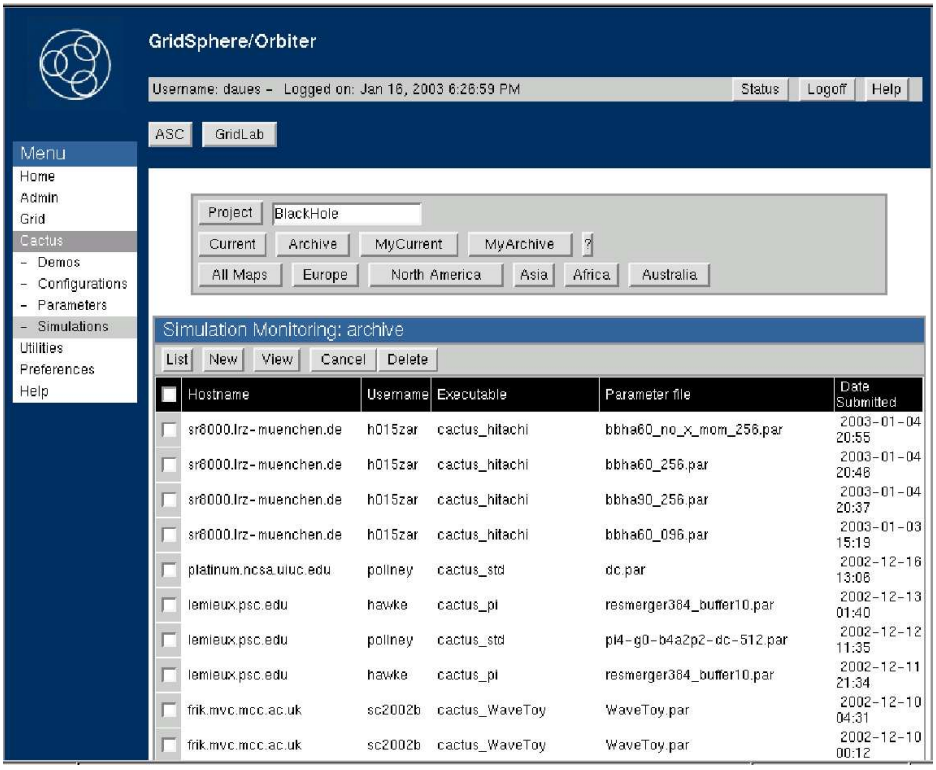

Fig. 1. The simulations list of the ASC Portal is organized by project, in this case the "Black Hole" project is shown. Interactive maps showing the physical location of simulations can be launched. All authorized ASC group members may view the list and track the progress of the simulations. The host on which the simulation is performed, the user who launched the simulation, the name of the binary executable, the parameter file used, and the date when the simulation started are displayed.

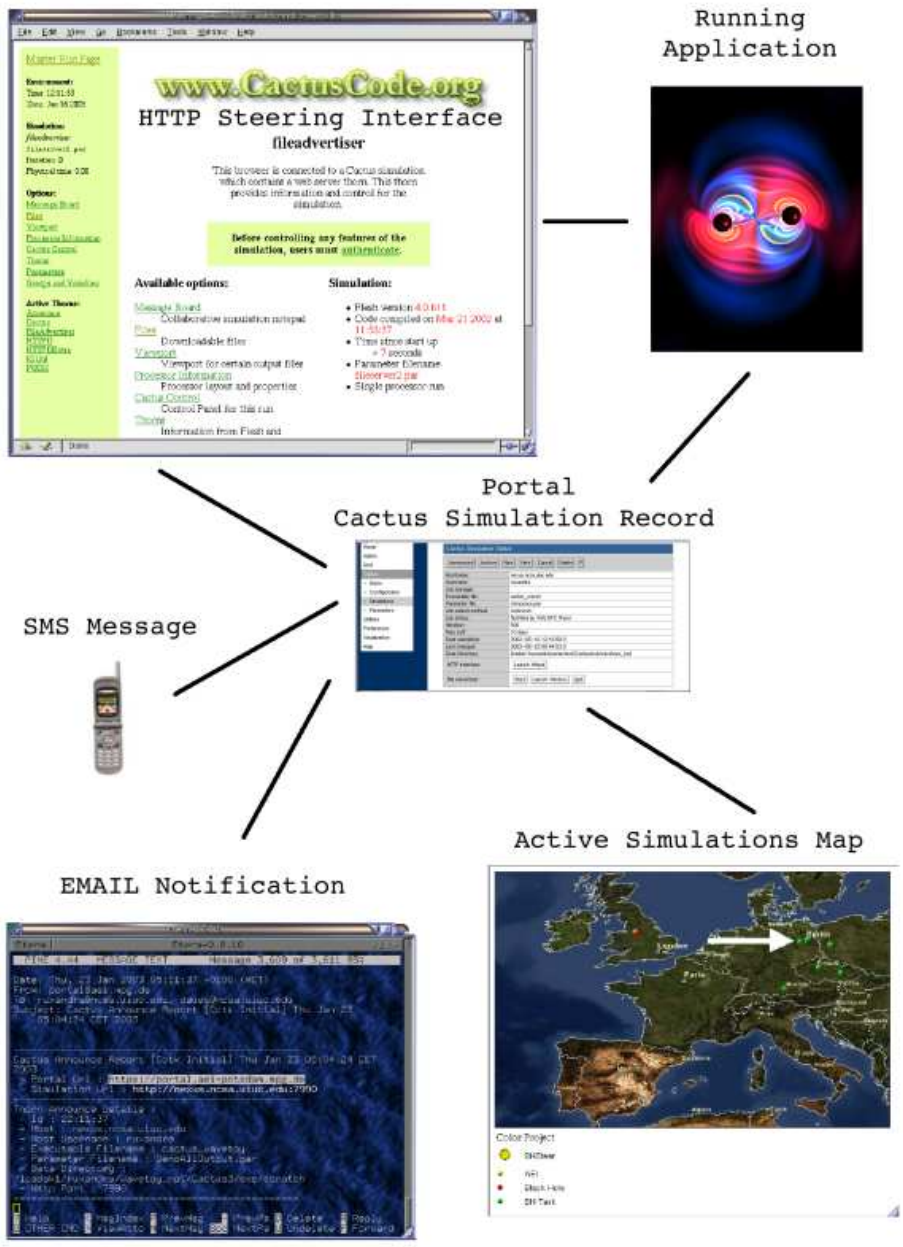

Fig. 2. Diagram displaying the current range of interactions between the ASC Portal and a running Cactus simulation. The application announces to the portal, sending information such as the hostname, the username, the name of the executable, and the location of the data being created. It also sends a URL for contacting the Cactus simulation via an HTTP steering interface, displayed in the upper left corner of the diagram. The ASC Portal notifies group members that the simulation has started by sending emails and/or messages to cellphones. Users may view the location of active simulations on an interactive map.

Cactus simulations can be monitored and interacted with in further detail. Any active simulation which includes the toolkit thorn `HTTPD` is provided with its own webserver (Figure 2). This webserver can then be connected to, from any remote browser, and provides a full description of the simulation such as a complete list of initial parameters, the name and locations of the output files, *etc.* It also provides a steering interface from which authorized users can change parameters in real time or pause and stop the run. The data generated by the simulation can be analyzed using local client or remote visualization tools. Figure 3 displays several visualization tools launchable from the ASC Portal.

The present simulation tracking framework has proved to be extremely useful for the users in the ASC VO. Currently, the messaging is a one-way communication from the application code to the portal host. Extensions to this framework will allow two-way communication so that users can control their applications through mobile devices such as cellular phones and PDAs. The communication between the Cactus code and the portal is performed via XML-RPC. The present XMLRPC server will evolve into an independent grid service that receives messages from and sends messages to any application. The XML based communication will use GSI security and the SOAP protocol, providing secure messaging between the portal and emerging grid services on a wide variety of computing platforms and architectures.

4.3 Executable Repository

An additional goal of the ASC is to enable not only experienced researchers to work with Cactus and the ASC Portal, but also students and scientists who are either casually interested in or just beginning to learn astrophysics and general relativity. Portal users can simply choose a general problem and a target machine. The portal automatically selects the necessary files from a repository, stages them to the remote machine, and starts the simulation. As the simulation is running, the generated output is displayed and standard visualization tools can be launched to view the results. By using the repository, ASC members avoid the repetition of trivial and tedious work such as the gathering and compilation of commonly used source code configurations.

While the repository contents are currently added by portal administrators, in the future portal users will be able to upload working binary executables from their production codes. The general framework will allow the addition of executables from any scientific application. The repository contents will be organized by projects, and made available to authorized groups of users. The production repository is an effective and organized way of sharing valuable compiled codes within a virtual organization.

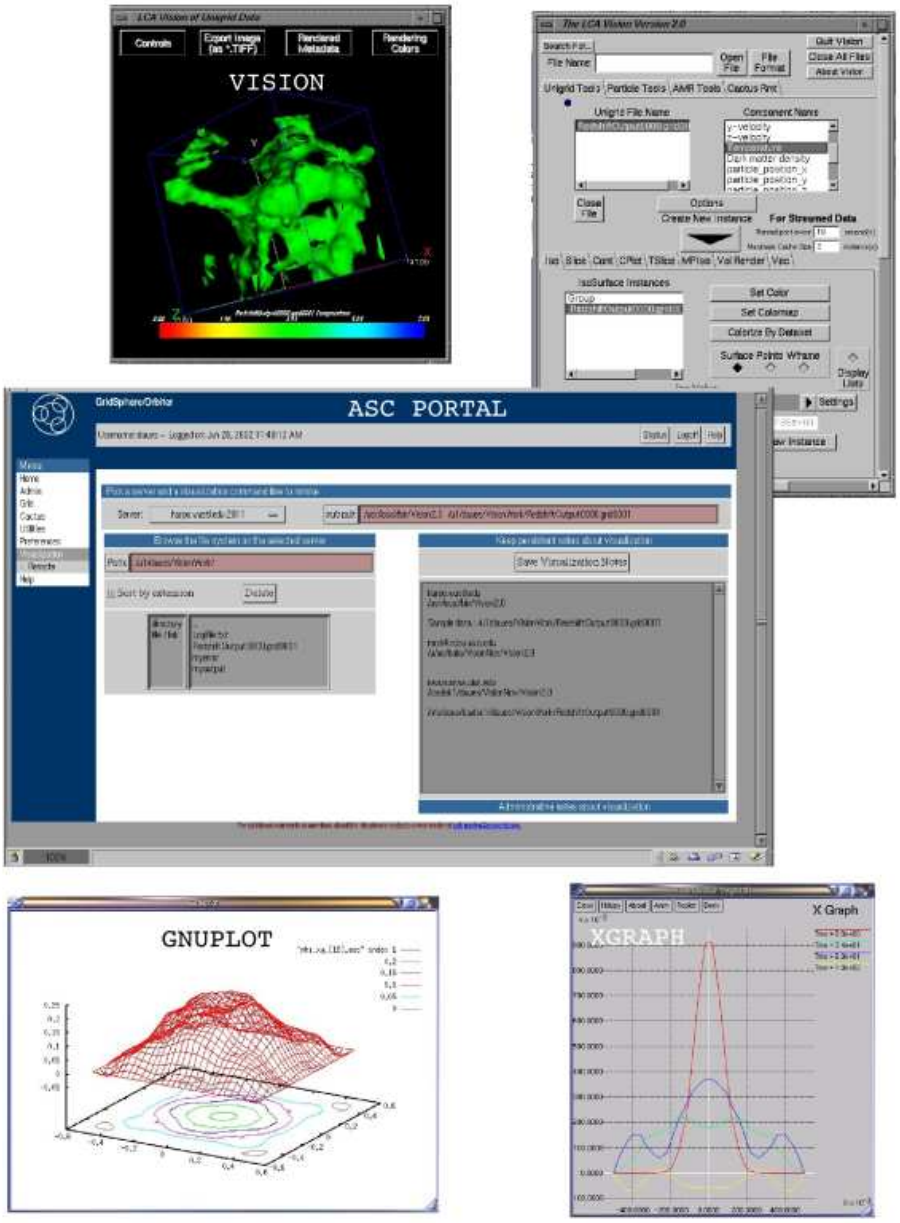

Fig. 3. Visualization tools launched by the ASC Portal. The visualization tools cover a wide range of data formats, with xgraph/ygraph displaying one-dimensional data, gnuplot displaying two-dimensional data, and Vision rendering three-dimensional data. The development of the Vision software package written by Brad Wind and Galina Wind was supported by the ASC project.

5 Example: SC2002 Demonstrations

The features of the ASC Portal described in this paper were shown in a number of demonstrations at the Supercomputing 2002 High Performance Networking and Computing Conference (SC2002). The portal could access and use many

different resources, including the GridLab, ASC and Global Grid Testbed Collaboration (GGTC) testbeds. The GGTC testbed (12) consisted of 69 different machines in 14 different countries and 5 continents. These machines were wildly diverse in size and platform, and were connected a variety of institutions and virtual organizations. The testbed contained supercomputers with thousands of processors, small Linux clusters, and even a Sony PlayStation in Manchester, England.

Each of the GGTC machines had common grid (Globus) and application (Cactus) infrastructure installed and deployed. From the ASC Portal a Cactus application could be staged and run on any machine, with the application announcing to the portal's simulation list. The portal also showed the status of a new generic task farming infrastructure developed by the GridLab project using their Grid Application Toolkit. One application running in a task farming mode was motivated by the ASC VO users: a parameter survey of binary black hole coalescence parameters. Black hole simulations are very demanding computationally, typically requiring hundreds of processors. Furthermore, black hole simulations require time consuming and tedious parameter tuning. These simulations can be expedited by task farming test problems across distributed machines, the results of which automatically steer parameters in much larger production runs. This application demonstrated how Grid computing can simplify, economise and speed-up the use of resources. The Global Grid Testbed Collaboration was awarded two HPC awards at SC2002 for its application use of this large testbed.

6 Future Directions

Grid computing promises many benefits for research communities such as the ASC VO who make heavy use of a bewildering variety of supercomputing resources, everyday developing and running complex largescale applications which create terabytes of data which must then be analysed, visualized and archived with the results somehow disseminated. This paper has described some initial advantages which are implemented in the ASC Portal: simplifying the use of resources; tracking and monitoring simulations; and coordinating and enabling virtual organisations.

This view is, however, just the beginning. Viewing distributed, heterogeneous computational resources, coupled with grid infrastructure, as a single virtual machine, leads the way to new intelligent grid-aware applications and unprecedented coordination and interaction of communities with their colleagues, data and simulations. Grid-aware applications will be able to migrate between supercomputers as more appropriate resources are needed, deciding when and if to distribute independent tasks to additional resources to save time and/or

money. Increased accuracy for largescale physics simulations will be available by coupling together supercomputers, with communication patterns and computational loads automatically adapting to the available bandwidth. Such dynamic and self configuring applications will require extensive tracking and logging, such as is beginning to be provided by the ASC Portal as they move between machines, leaving behind data in various locations.

The end users of Grid technology will be given new opportunities both for cataloguing and understanding their computational work. Right now the ASC Portal collects basic information about the VO's simulations in a persistent database. This interface could very easily be expanded to collect additional information from simulations, including:

- Building a repository of benchmark results for standard simulation modules which could then be used for selecting appropriate resource.
- Collecting accounting information, such as the number of processors and CPU hours used by the members of the VO on their different resources, to help with budgeting and resource selection.
- Building a database of physics parameters and results. For the ASC VO such a database might tabulate the type of initial data with the main computation method used for evolution and the error at a certain point of time.
- Archiving the location and content of input and output data files for a simulation.

Many such capabilities will in the future be provided by Grid services (13). For example, the GridLab project is developing an OGSA compliant set of data management services. However in the short term the ASC Portal technologies can be used to flexibly and quickly design, prototype, and test following the requirements of the ASC VO users.

The ASC Portal has implemented basic notification services, informing users by email or SMS of the status of their simulations. There are many potential uses and extensions which can be added. The Cactus team is planning to implement a generic notification thorn, which will provide application developed with the ability to request user notification for significant events, including information about the urgency of the notification. Such notifications could signal the appearance of a black hole horizon, of an unexpected physical feature which should be analysed, or could suggest that a user might want to interact with the simulation to change a physics parameter. The portal can also interact with the machines, monitoring for example disk usage to warn users when a disk is becoming full and could crash a simulation, or advise users that a queue is empty and could be immediately exploited. The GridLab project contains a workpackage for mobile computing, which will take such concepts further again, implementing interfaces to their portal on mobile devices such as telephones and PDAs.

As described in Section 2, the ASC Portal is now being migrated to the GridSphere portal framework developed in the GridLab project. This will more easily allow the present and planned simulation management capabilities described in this paper to be used by projects outside of the ASC VO.

7 Conclusion

The ASC Portal attempts to solve the problem of simulation code management on the wide variety of computer systems that comprise computational grids. The portal keeps a history of the work of a research group, and enhances collaboration within a virtual organization. The ASC/GridLab SC2002 demonstrations showed (1) a large heterogeneous computational grid could be constructed and maintained in collaboration with resource administrators and (2) applications are not restricted to run on a single resource of a particular size and type, but may be run on the computational grid itself. In the future, the portal will start and manage these applications through interactions with grid services such as resource brokers and advanced job schedulers.

Acknowledgements

First, we would like to thank the physics users of the ASC VO, in particular Peter Diener, Frank Herrmann, Denis Pollney and Jian Tao. We gratefully acknowledge the help of the Cactus team, particularly Tom Goodale, Gerd Lanfermann, and Thomas Radke, and our ASC and GridLab colleagues Thomas Dramlitsch, Ian Foster, Sai Iyer, Gregor von Laszewski, Jason Novotny, Miroslav Rudek, Wai-Mo Suen and Oliver Wehrens. The ASC project is funded by the NSF (KDI 9979985) with computational support from US NRAC grant 93S025. Important contributions to this work have been made by the EU GridLab (IST-2001-32133) and Astrophysics Network (HPRN-CT-2000-00137) projects, and the German DFN Verein GriKSL project (TK 602 - AN 200). We thank the staff at NCSA, LRZ, AHPCC, SDSC, PSC and RZG who have provided the necessary software installations and support at the different sites used by the ASC VO. We also acknowledge generous support from Microsoft.

References

[1] T. Goodale, G. Allen, G. Lanfermann, J. Massó, T. Radke, E. Seidel, J. Shalf, The Cactus framework and toolkit: Design and applications,

in: Vector and Parallel Processing - VECPAR'2002, 5th International Conference, Lecture Notes in Computer Science, Springer, Berlin, 2003.

[2] G. Allen, E. Seidel, J. Shalf, Scientific computing on the grid, Byte Spring (2002) 24–32.

[3] Astrophysics Simulation Collaboratory (ASC) Home Page `http://www.ascportal.org`.

[4] I. Foster, C. Kesselman, S. Tuecke, The anatomy of the grid: Enabling scalable virtual organizations, Intl J. Supercomputer Applications 15 (3), `http://www.globus.org/research/papers/anatomy.pdf`.

[5] GridLab project: Portal Work Package: `http://www.gridlab.org/WorkPackages/wp-4/index.htm`.

[6] I. Kelley, J. Novotny, M. Russell, O. Wehrens, Asc portal evaluation, Tech. Rep. D4.1, GridLab, `http://www.gridlab.org/Resources/Deliverables/D4.1.pdf` (2002).

[7] G. V. Laszewski, M. Russell, I. Foster, J. Shalf, G. Allen, G. Daues, J. Novotny, E. Seidel, Community Software Development with the Astrophysics Simulation Collaboratory, Concurrency and Computation: Practice and Experience Http://www.cactuscode.org/Papers/CCE_2001.pdf.

[8] M. Russell, G. Allen, G. Daues, I. Foster, E. Seidel, J. Novotny, J. Shalf, G. von Laszewski, The astrophysics simulation collaboratory: A science portal enabling community software development, Cluster Computing 5 (3), `http:www.cactuscode.org/Papers/CCJ_2001.ps.gz`.

[9] I. Foster, C. Kesselman, Globus: A Metacomputing Infrastructure Toolkit, International Journal of Supercomputing Applications 11 (2) (1997) 115–128.

[10] Commodity Grid Kits: `http://www.globus.org/cog`.

[11] J. Novotny, S. Tuecke, V. Welch, An online credential repository for the grid: Myproxy, in: Proceedings of Tenth IEEE International Symposium on High Performance Distributed Computing, HPDC-10, San Francisco, IEEE Press, 2001, `http://www.globus.org/research/papers/myproxy.pdf`.

[12] The Global Grid Testbed Collaboration: `http://scb.ics.muni.cz/static/SC2002`.

[13] I. Foster, C. Kesselman, J. Nick, S. Tuecke, The physiology of the grid: An open grid services architecture for distributed systems integration, Open Grid Service Infrastructure WG, Global Grid Forum June.