The Problem Solving Environments of TeraGrid, Science Gateways, and the Intersection of the Two

Jim Basney[1], Stuart Martin[2], JP Navarro[2], Marlon Pierce[3], Tom Scavo[1], Leif Strand[4], Tom Uram[2,5], Nancy Wilkins-Diehr[6], Wenjun Wu[2], Choonhan Youn[6]

[1]*National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign*
[2]*Argonne National Laboratory*
[3]*Indiana University*
[4]*California Institute of Technology*
[5]*University of Chicago*
[6]*San Diego Supercomputer Center, University of California at San Diego*

jbasney@ncsa.uiuc.edu, smartin@mcs.anl.gov, navarro@mcs.anl.gov, mpierce@indiana.edu, tscavo@ncsa.uiuc.edu, leif@geodynamics.org, turam@mcs.anl.gov, wilkinsn@sdsc.edu, wwjag@mcs.anl.gov, cyoun@sdsc.edu

Abstract

Problem solving environments (PSEs) are increasingly important for scientific discovery. Today's most challenging problems often require multi-disciplinary teams, the ability to analyze very large amounts of data, and the need to rely on infrastructure built by others rather than reinventing solutions for each science team. The TeraGrid Science Gateways program recognizes these challenges and works with science teams to harness high-end resources that significantly extend a PSE's functionality.

1. Introduction

Initiated in 2004, the TeraGrid Science Gateways program began with the realization that scientists were increasingly developing their own interfaces to exploding amounts of digital information. The National Center for Biotechnology Information (NCBI) Blast server had been providing genome analysis capabilities since the 1980s [1]. The Protein Data Bank had been providing access to curated data collections with visualization and analysis capabilities since 1989 [2]. Today there are many such portals or gateways providing capabilities to scientists. The goal of the TeraGrid Science Gateway program is to allow gateway developers to easily incorporate powerful compute, data and visualization resources into existing community-designed interfaces. Some scientists may not even realize that they are using the TeraGrid, they will just appreciate the tremendous increase in what they are able to accomplish scientifically.

The TeraGrid provides its own problem solving environment (PSE) used by nearly 30 gateways spanning many disciplines. Section 2 of this paper will describe the common TeraGrid gateway infrastructure while sections 3–6 highlight several individual science gateways. Highlights will include a brief description of the PSE provided by each gateway for their own research communities, as well as describe the science successes they hope to achieve through the use of high-end resources provided by the TeraGrid.

2. TeraGrid's Common Infrastructure

At their core, problem solving environments (PSEs) provided by science gateways will result in the deployment of one or more applications to solve, compute, search, estimate, or locate something. Often these applications can be run on resources local to the PSE. However in some cases the ability to scale to a large number of users, improve turn around time or to address more computationally or data-intensive problems, requires more data resources, faster compute resources, or both.

TeraGrid provides a set of services and capabilities to enable a wide range of applications to use its compute and data resources effectively. By using a set of client APIs, a PSE or gateway can interact securely with the remote TeraGrid services to: discover where services are deployed, discover details about the resources, move data to and from the resources, and submit computational jobs to the various local resource managers for remote execution.

Designed to take advantage of TeraGrid's (and other grid's) compute and data resources, gateways have the flexibility to use either local or remote resources. The user interface can evolve without regard for the underlying execution method.

The following sections will describe the services and capabilities of TeraGrid's common infrastructure that are often used by gateways.

2.1. Secure TeraGrid Access

Security is an important and essential aspect of TeraGrid's common infrastructure. TeraGrid security is based on *Grid Security Infrastructure* (GSI) [3] as implemented in Globus Toolkit [4]. Under GSI, users authenticate themselves by presenting trusted X.509 certificates to relying parties, either X.509 end entity certificates [5] or X.509 proxy certificates [6]. A distinguishing feature of GSI is its reliance on X.509 proxy certificates.

TeraGrid leverages GSI throughout its common grid middleware infrastructure. A user authenticates to a TeraGrid resource provider using a GSI proxy credential. The proxy is either created using a command-line tool like grid-proxy-init or retrieved from a MyProxy server using myproxy-logon. In either case, the user presents the proxy certificate to the resource provider and proves possession of the corresponding private key. The resource provider uses the distinguished name (DN) in the proxy certificate as a basis for access control, that is, the resource consults an access control list of DNs, called a *gridmap*. Authorization based on gridmap files is an important characteristic of TeraGrid's common security infrastructure.

GSI-OpenSSH and MyProxy form the core of the TeraGrid's single sign-on solution, offering users a means to access any TeraGrid resource with a single authentication. Likewise the GridShib software is a key component of the security architecture supporting TeraGrid science gateways, allowing TeraGrid to scale to orders of magnitude more users than would otherwise be possible with traditional HPC user management schemes while providing comparable security. We describe these security technologies in the following sections more fully.

2.1.1. GSI-OpenSSH. GSI-OpenSSH (http://grid.ncsa.uiuc.edu/ssh/) is an enhanced version of OpenSSH (http://www.openssh.org/) that adds support for GSI (i.e., X.509 proxy certificates [6] and delegation [7]), thereby providing a single sign-on login and file transfer service for grids. GSI support in GSI-OpenSSH is transparent to users, giving them the look and feel of vanilla OpenSSH to which they are accustomed. Java clients, embedded in web pages as applets, are also available, giving users the ability to access grid resources using a standard web browser. GSI-OpenSSH is used on a large scale by over 50 sites worldwide, including TeraGrid.

A user can use GSI-OpenSSH to access any compute resource throughout TeraGrid without having to maintain SSH public keys at each host. Some gateways have successfully used GSI-OpenSSH as their only common infrastructure component. However, using only GSI-OpenSSH requires gateway developers to spend more time integrating with the unique aspects of each compute resource, in particular, the various local resource managers.

2.1.2. MyProxy. For an individual TeraGrid user, GSI-OpenSSH works well, but for most gateways, it is not sufficient. Many gateways are web portals and their users are remote, that is, they do not login directly to the TeraGrid and generate proxies. These gateways require a more convenient method of storing and retrieving individual user credentials. Some gateways use MyProxy for this purpose.

MyProxy (http://myproxy.ncsa.uiuc.edu/) [8] [9] is open source software for managing X.509 credentials (i.e., certificates and private keys). MyProxy combines a credential repository with an online certificate authority to allow users to securely obtain grid credentials when and where needed for access to secure grid services. MyProxy is a mature software product in use by more than 200 grid sites worldwide, including TeraGrid.

MyProxy is a critical component of TeraGrid's security infrastructure. A central TeraGrid MyProxy server is maintained for users and gateways alike. In the rare event that the MyProxy service is unavailable, however, users and gateways may be unable to obtain needed credentials. This is an ongoing concern as TeraGrid becomes increasingly reliant on MyProxy. Consequently, a new peer-to-peer active replication capability for MyProxy is being developed to provide the enhanced reliability and performance required by next generation TeraGrid.

2.1.3. Community Account Model. Many gateway users do not require a personal account on each compute resource nor do they need their own grid credential. A user credential could be generated by the gateway for each user and used on their behalf, but significant configuration is required at the resource provider to authorize each user. Clearly this is not a scalable solution for tens of thousands of users. A simpler and more efficient method is needed. For this reason, the use of so-called community credentials has been adopted by gateways and resource providers [10]. All gateways featured in this paper make use of TeraGrid's community accounts. A community credential is a single X.509 credential used for an entire community rather than an individual user. The gateway insures that the credential is kept secure and uses it to access TeraGrid's grid services on behalf of its users. This simplifies operations for both the gateway and the resource provider.

Operationally, a gateway authenticates as itself to the resource provider using a proxy certificate issued and signed by the community credential. On the server side, the resource authorizes and maps the gateway request to a single community account shared by all users.

A science gateway based on the community account model is relatively easy to implement but unfortunately there are significant drawbacks [10] [11]. In particular, all requests from the science gateway look the same to the resource provider, so there is no opportunity for fine-grained access control since the end user (let alone the user's attributes) is unknown to the resource provider.

2.1.4. GridShib. Although the community account model is a more scalable security solution, it hides the identity of the gateway user from the TeraGrid. To address this issue, TeraGrid is deploying GridShib (http://gridshib.globus.org/), a suite of software components designed to federate campus and grid identity management infrastructure [12]. GridShib accomplishes this by providing interoperability between the Shibboleth system and the Globus Toolkit grid computing middleware [13]. In this sense, GridShib extends the reach of Shibboleth by allowing campus users to seamlessly access computational grids.

At the TeraGrid Authentication, Authorization and Account Management Workshop held at Argonne National Laboratory in August 2006, the TeraGrid leadership proposed a testbed to evaluate identity federation and attribute-based authorization [14]. The goal of the testbed was to enable scalability to a larger number of TeraGrid users than would be possible with existing infrastructure. As a result of that effort, an enhanced community account model [11] was proposed. The model incorporates GridShib SAML Tools at the gateway and GridShib for Globus Toolkit at the resource provider. With these two components installed, the gateway passes security information that the resource can use for fine-grained access control, auditing and incident response.

Specifically, attributes can be included in the proxy certificate issued by the gateway. These attributes uniquely identify the end user and therefore the user's identity is known to the grid service and can be used to authorize the request. Additionally, GridShib can provide auditing information that associates the user with the requested action. GridShib has been prototyped with GRAM4 to enable TeraGrid accounting for tracking gateway users to compute jobs. In the future, attributes uniquely identifying gateway users will be routinely included in community proxies used to launch jobs. This will fulfill a requirement by the sponsor agency to count individual users of TeraGrid resources.

2.2. Remote Job Execution

Executing remote applications on TeraGrid compute resources is a primary goal of gateways. For this purpose, TeraGrid deploys both GRAM2 [15] and GRAM4 [16] services. GRAM uses GSI for authentication and authorization, so any GRAM service deployed on TeraGrid compute resources can be accessed by a gateway with an individual or community X.509 user proxy. GRAM4 is a newer Web Services version of its GRAM2 predecessor. Both GRAM2 and GRAM4 submit, monitor, and cancel "jobs" on Grid computing resources. A GRAM "job" may include file staging before and after the execution of the user's application.

GRAM provides a level of abstraction on top of the resource provider's computing environment. The same GRAM job specification can be used to submit jobs to different compute resources that are managed by different local resource managers (e.g., PBS, Condor, SGE, LSF, etc.). Many gateways use either GRAM2 or GRAM4 for remote job execution, but the trend is towards GRAM4.

TeraGrid also provides SoftEnv, an environment management tool that uses "keys" to help users maintain a common environment across TeraGrid's compute resources. Users and gateway developers may also define their own keys, providing further control over the execution environment. GRAM has been extended to support SoftEnv.

2.3. Remote File Transfer

Data transfer is also an important component of gateway interactions with the TeraGrid and is therefore a part of TeraGrid's PSE for gateways. GridFTP [17] is a high-performance, secure, reliable data transfer protocol optimized for high-bandwidth wide-area networks. GridFTP uses GSI for authentication and authorization, so any GridFTP service deployed on TeraGrid compute resources can be accessed by a gateway using an individual or community X.509 user proxy. GridFTP has many features such as parallel transfers and striping that can significantly outperform other file transfer methods including scp. Many gateways use GridFTP for remote file transfer: 1) from the user's local file system to the gateway's own file system (and vice versa) and 2) from the gateway's file system to and from TeraGrid's remote file systems.

GridFTP is "reliable" insofar as a file transfer can be restarted from where it was left off (or interrupted). However, it does not queue or throttle file transfer requests, and has capacity limitations. The Reliable File Transfer (RFT) service [18] addresses these limitations and orchestrates the file transfers to be done between the two GridFTP servers. Because many gateways serve diverse communities that do not coordinate their use of the gateway, an automated way of handling unanticipated loads will likely be part of future gateway development plans.

2.4 Information Discovery

In order to remotely execute jobs, move files to and from the TeraGrid, and login to TeraGrid resources to build and deploy applications, gateway developers need to discover where job submission, file-transfer, and login services are available. Similarly developers need to discover whether the required software packages (such as compilers, libraries, and other tools) are available and how to access them individually.

The TeraGrid's Integrated Information Services are a network of web services responsible for aggregating the availability of TeraGrid capability kits, software, and services across all the infrastructure providers. Job submission, file-transfer, and login are three examples of TeraGrid capability kits.

Gateways and other developers access information services content manually through TeraGrid documentation and the User Portal, and can programmatically access the information by issuing queries directly to information services web interfaces.

2.4.1. Dynamic Status Discovery. Besides the fairly static discovery of capabilities, software, and services, the TeraGrid also leverages integrated information services to publish information about the relative scheduling load of TeraGrid resources, and the current contents of batch queues.

The scheduling load can be a useful data point when deciding which of various resources is most likely to complete a particular job sooner. The contents of the batch queues are useful if gateways and other problem solving environments want to see a full listing of jobs and job status without having to login to all the batch systems being used. The scheduling load and batch queue contents of all TeraGrid compute resources can be accessed from a single page on the TeraGrid user portal.

2.4.2 Testbeds. The above introduction to information discovery described how gateways and other users can discover where production capabilities, software, and services are available.

The TeraGrid's capability registry itself, however, has the ability to index capabilities that may still be in a development or a testing status. The TeraGrid is just starting to leverage this functionality, which will enable the same mechanisms used to discover production capabilities to be used to discover what is available in testbed status. Since the information about production versus development or testing capabilities is in a common information services registry, gateways and other PSEs that discover availability information will be able to switch between using production and testbed resources by simply altering the resource status of information service discovery queries.

2.4.3 Gateway/PSE Registration. Moving beyond the ability to discover information about development, testing, or production capabilities, the TeraGrid's information services framework is flexible enough to allow the extended community of service providers to register their own offerings. When leveraged, this functionality will allow gateways and PSEs to advertise themselves and their offerings into the TeraGrid-wide integrated information services index. This will effectively open up the TeraGrid's infrastructure discovery functionality to the extended community of service providers.

3. The Social Informatics Data (SID) Grid

The Social Informatics Data Grid (SIDGrid) [19] is a new cyberinfrastructure designed to transform how social and behavioral scientists collect and annotate data, collaborate and share data, and analyze and mine large data repositories. The grand challenge in social and behavioral science is how to model human behavior as a dynamic, multi-causal system that occurs over multiple time scales. It is essential for behavioral scientists to store multiple measures of neural, cognitive, and social behaviors of humans into a common database so that they can access and analyze these measures at multiple levels simultaneously in a collaborative way.

The SIDGrid enables researchers to capture multimodal behavior in real-time at multiple levels simultaneously, and then to store and analyze different data types (e.g., voice, video, images, text, numerical) in a distributed multimedia data warehouse that employs web and grid services to support data storage, access, exploration, annotation, integration, analysis, and mining of individual and combined data sets. Such a data warehouse provides transparent access to distributed, aligned, and annotated social informatics data.

SIDGrid uses TeraGrid resources for computationally-intensive tasks such as media transcoding algorithms for pitch analysis of audio tracks and fMRI image analysis. For example, speech, gesture, facial expression, and physiological measurements attending an event or interaction can be captured and automatically stored both locally and remotely in the SIDGrid. Once stored in raw form in the SIDGrid, these data streams can then be transformed into formats that are compatible with software tools for annotation, coding, integration, and analysis.

Analysis tasks for behavioral science research usually involve a large amount of multimedia data and require tremendous computational resources provided by Grid-enabled problem solving environments like TeraGrid. We have deployed the SIDGrid computationally intensive services as part of a science gateway interfacing to TeraGrid. Users are able to access the TeraGrid resources via the SIDGrid Portal. This problem solving environment greatly extends the class of automated analysis experiments that can be conducted by behavioral scientists and enables them to explore the feasibility of near-realtime analysis of experimental data.

Acting upon requests from social and behavioral researchers, SIDGrid can construct computing workflows, submit large numbers of computing jobs to the TeraGrid, monitor the progress of workflows and deposit final results directly into the SIDGrid data warehouse. A powerful workflow engine called the Virtual Data System (VDS) [20] parses the workflow descriptions generated by SIDGrid and orchestrates the entire execution procedure. Several Grid services are utilized by VDS to perform massive data transfers and job submissions. GridFTP services are used for high-performance data movements of large SIDGrid datasets among data resources in the staging steps of SIDGrid workflows. Grid compute services (GRAM) are used for the submission and monitoring of jobs to TeraGrid clusters.

Currently we are focused on integrating data from three broad and complementary areas of social and behavioral research: multimodal communication in humans and machines, neurobiology of social behavior in human and animals, and cognitive and social neuroscience. We have imported data from the CHILDES [21] and TalkBank [22] datasets that are widely used by this community into the SIDGrid environment. Dozens of computational applications related to fMRI image analysis, discourse and dialogue study as well as gesture tracking have been integrated into the SIDGrid science gateway. Active users of the SIDGrid system include a human neuroscience group and linguistic research groups from both the University of Chicago and the University of Nottingham, UK.

4. The Geosciences Network (GEON)

Within scientific communities, portals or problem solving environments (PSEs) become highly desired as research becomes increasingly interdisciplinary and includes access to non-traditional data sets and tools. GEON addresses these needs specifically for geoscience communities. GEON is providing several key features ranging from user logins, data access, computational simulations, personal work spaces and analyses environments. This type of end-to-end scientific problem solving environment is a critical need for geoscientists. The current technology enables us to build user-friendly environments that require very simple hardware and software from the user side to be able to access all that GEON offers.

GEON (http://www.geongrid.org/) is a large-scale collaborative cyberinfrastructure project involving Information Technology and Earth Science researchers from multiple institutions. The focus is on building data-sharing frameworks, developing tools and services, and identifying best practices with the objective of dramatically advancing geoscience research and education. GEON has adopted a service-oriented approach and a portlet-based approach—both of which are applicable to other science grid projects as well—that has led to the development of a number of reusable portal services [23]. User interfaces provide access to services via *portlets* implemented within a portal framework using a standard portlet API (JSR 168). These pluggable portal components process user requests and generate dynamic content, and can be deployed on remote sites. Using a standard software stack and API, GEON's partner sites are able to develop and deploy software that is compatible with the rest of the system and can be reused by others in the network in a seamless fashion.

GEON also needs to support education and research activities for a field whose computational and visualization needs are increasing rapidly and span a broad range. One example application is SYNSEIS (SYNthetic SEISmogram generation tool), a portal-based distributed system designed to access seismic waveform data and simulate seismic records using 2D and 3D models [24]. SYNSEIS provides access to a computational environment via a portlet-based mechanism, allowing authenticated and authorized users to access high-end computational resources, including TeraGrid clusters that are minimizing the requirements needed to conduct advanced calculations for simulating seismic waveforms of either earthquakes or explosions at regional distances (< 1000 km).

The ability to speedily construct earth models, access observed earthquake recordings and simulate them to understand the subsurface structure and characteristics of seismic wave propagation is a task conducted by seismologists in many parts of the world. Current practices are extremely slow and only a few select scientists have access to high performance computing and the required support systems. With the GEON portal, geoscientists are now in a position to expand the practice of simulating earthquakes and learn more about the structure of the Earth in many parts of the world in efficient ways.

For secure access, GEON employs the Grid Security Infrastructure (GSI). Account management services are provided by the Grid Account Management Architecture (GAMA) [25]. GAMA provides a central GSI certificate management system where "grid" users are approved and credentials are created for them based on global policies. The portlet connects to the job submission Web service, which checks the user identity and authorization, and creates the proxy credential. GridFTP and GRAM are used to transfer data and submit jobs.

Users are able to monitor jobs and access results via MyWorkspace. By using the portal, they are now able to analyze results much more quickly than before. Since the E3D scientific application code has a complex parameter creation process, the GEON portal minimizes the difficulty of building a set of parameterized input files and provides users with a nice variety of user-centric and dynamic environments on the web for the simulation.

To support classroom use, we have developed a class account management system that instructors can use to easily create group accounts. Our goal is to combine Web 2.0 concepts with conventional cyberinfrastructure to create virtual scientific and education communities. The *myProjects* collaboration tools available in the GEON portal can be used, along with tagging, to allow students to review and vote on submitted contents, including parameter settings and job outputs, and to support group discussions among the class. Sharing of such information may potentially help students avoid the unnecessary and expensive execution of computer coding, and may provide them with a more effective way of sharing and testing possible solutions. As the use of collaboration tools and cyberinfrastructure matures, tools such as myProjects will have the potential for significant impact on education and research [26].

5. QuakeSim

The QuakeSim Project [27] is a NASA-funded project to enable the study of earthquakes using distributed computational tools and databases. QuakeSim applications are used to study both the spatial deformations and underlying fault models associated with earthquakes as well as geo-temporal data such as GPS time series.

As a Web-based gateway, QuakeSim provides a browser-based frontend (Figure 1) as a problem solving environment. Users interact with remote databases, create and manage jobs, and interact with map interfaces in setting up problems. These interfaces hide the details of the (typically complicated) job submission process. Users may also archive projects for later editing and resubmission.

The QuakeSim gateway is built around a service-oriented architecture. We have developed Web services for QuakeSim applications (Disloc, Simplex, RDAHMM, GeoFEST, etc; see [28]) that manage the execution of these applications. We have developed portlets for the QuakeSim application services and also reused selected portlets from the Open Grid Computing Environments project [29]. The QuakeSim Web service collection also includes services for accessing fault databases and generating keyhole markup language (KML) descriptions of output files. We base our job management core on an Apache Ant engine (for simple submissions) and Condor-G (for Grid submissions) but extend this to provide specific and stateless interfaces to the computational applications.

Applications in QuakeSim range from very small applications such as Disloc (useful for rapidly calculating surface displacements associated with earthquakes) to very scalable parallel applications such as GeoFEST [30] that can be used in very detailed studies of deformations associated with fault motion. We also must support naturally parallel data mining applications (such as RDAHMM) on hundreds of GPS stations. For these latter two situations, we use the NSF TeraGrid computing facilities. We still provide standard Web service interfaces for these applications, but use Condor-G and BirdBath as our job manager core. Condor-G in turn uses GRAM2 for its job submission and data staging. We have spun this work off into the Swarm project [31].

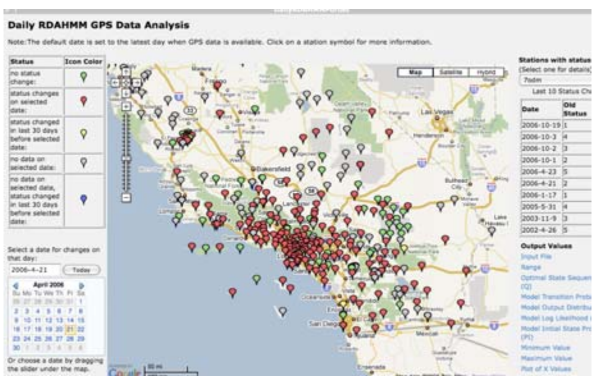

Figure 1. Screenshot of the QuakeSim portal's GPS analysis portlet, showing the detected network reset in Southern California on April 21, 2006. GPS stations are indicated by pushpins on the Google map interface. Red pushpins indicate that the station changed state on the indicated day.

Data services in the portal include fault models from the QuakeTables fault database [32], GPS archival data from Geophysical Resources Web Service [33], and real-time data from the California Real-Time Network [34]. For real-time GPS, we use message-oriented publish-subscribe software (NaradaBrokering [35]) to manage and filter the data streams.

The QuakeSim user community includes both NASA researchers and students. The Disloc and Simplex applications have been used by Prof. Gerry Simila (California State University, Northridge). All current usage is through the QuakeSim community credential, but it is possible for users to submit jobs with their personal credentials.

The QuakeSim project has been designed around a loosely coupled collection of services and user interface components in order to minimize inter-dependencies that plague enterprise middleware development [28]. Our portlets, for example, are written with Java Server Faces and use the Apache Portlet Bridge, so they do not depend on the portlet API. This will allow us to convert our applications to (for example) Open Social compatible Google gadgets. Likewise, we use simple and stateless Web services with application specific Web service definition files rather than application factories. This will allow us to deploy services on a range of computing infrastructures (Grids today, clouds in the near future) without being closely tied to any particular middleware that is not under our control.

The data storage requirements for QuakeSim are currently modest but may grow rapidly in the future. Large GeoFEST studies may generate 10s of gigabytes of data, and the storage of all real-time GPS data requires approximately 300 GB per year. However, NASA is in the planning stages of the DESDynI satellite mission (http://desdyni.jpl.nasa.gov/), which will collect over 500 GB of InSAR data per day. DESDynI will be used to detect ground motions such as those associated with earthquakes and landslides, and InSAR data can be directly ingested by several of the QuakeSim applications.

6. Computational Infrastructure for Geodynamics (CIG)

The Computational Infrastructure for Geodynamics (CIG) is "a membership-governed organization that supports and promotes Earth science by developing and maintaining software for computational geophysics and related fields." Through the development of a science gateway, CIG will allow geophysicists to run simulations on both local resources and the tremendous computational resources of the TeraGrid [36].

The CIG Seismology Web Portal (https://crust.geodynamics.org/portals/seismo/) enables researchers to request synthetic seismograms for any given earthquake, selecting from an assortment of 3D and 1D earth models. Seismograms allow scientists to understand the ground motion associated with any given earthquake. Simulations of the synthetic seismograms are performed on the Texas Advanced Computing Center's Lonestar system, which is part of the TeraGrid. Upon completion of a simulation, the user receives a notification e-mail directing them back to the web portal where they can download the resulting seismograms in ASCII and Seismic Analysis Code (SAC) format. Visualizations include "beachball" graphics depicting the earthquake's source mechanism, and maps showing the locations of the earthquake and the seismic stations. By providing this capability through the gateway, CIG allows researchers to quickly receive the results of computationally-intensive simulations and concentrate on the scientific aspects of the output rather than on the details of running the analysis on a supercomputer.

The portal runs 3D simulations using SPECFEM3D GLOBE, which simulates global and regional (continental-scale) seismic wave propagation using the spectral element method. A typical SPECFEM simulation runs from two to three hours, using 150 to 216 processors. The portal's 1D simulations are performed by the serial Mineos code, which uses normal mode summation for its simulation algorithm.

To simulate an earthquake, the portal needs source information in Harvard CMT (Centroid Moment Tensor) format. To obtain this input data, the user can search for events in the database provided by the Global CMT Project (http://www.globalcmt.org/). The database is integrated into CIG's web portal, allowing the user to select an earthquake by simply pointing and clicking. Alternatively, the user may upload custom CMT data to the portal. The portal provides a default set of seismic stations, but the user may also upload a custom set.

6.1. Technical Operation

The web site is written in Python, and built upon the Django web framework (http://www.djangoproject.com/). Data persistence is achieved using an SQLite database (http://www.sqlite.org/). The site runs on top of Apache, which provides secure HTTPS connections.

The web site is passive, that is, it does not initiate connections to carry out tasks. Instead, the portal is powered by a separate daemon script that works in the background. The daemon constantly polls the web site looking for work to do. When a simulation request is posted, the daemon springs into action. First, it downloads input files for the simulation from the web site. Next, it connects to the TeraGrid cluster using GSI-OpenSSH and MyProxy credentials where it schedules and monitors an LSF batch job. Finally, it transfers simulation output files to a web server. Throughout this process, the daemon posts status updates to the web site using HTTP POST, so that the user can monitor the progress of their request.

The daemon script and other backend helper scripts are also written in Python, but using the Pyre framework (http://www.cacr.caltech.edu/projects/pyre/). The "beachball" and map visualizations are generated on-the-fly by the web server using GMT (http://gmt.soest.hawaii.edu/). The images are cached by the server to optimize performance.

6.2. Future Directions

Currently, the portal requires the user to manually enter simulation parameters using web forms. Future enhancements may allow the user to easily specify a "parameter exploration," automatically iterating through a range of parameters and launching a simulation for each generated parameter set.

Members of the CIG seismology community have also expressed an interest in adding the ability to upload custom earth models to the portal. Currently, custom 1D models may be uploaded, but 3D models are limited to a predefined set that is hardwired into the SPECFEM3D code. Uploading 3D models to the portal poses an interesting technical challenge, as SPECFEM3D models consist of not only data, but also code to interpret that data and project it onto the mesh at runtime, in parallel.

7. Conclusions

The technical requirements of PSEs, which are becoming increasingly important to scientific discovery, emphasize the need for seamless access to high-end compute and data resources such as those offered by the TeraGrid. Often, functionality needed by one PSE will also be needed by others. A robust, flexible and scalable infrastructure provides a foundation for all PSEs to extend the capabilities and enhance the scientific tools offered to researchers.

Finally, PSEs themselves must be treated as sustainable infrastructure. Researchers will not truly rely on PSEs for their work unless they have confidence that the PSE will remain operational for the long term and provide reliable services.

8. References

[1] Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990). "Basic local alignment search tool". J Mol Biol **215** (3): 403–410. http://www-math.mit.edu/~lippert/18.417/papers/altschuletal1990.pdf

[2] Berman, H., Bourne, P., Westbrook, J. The Protein Data Bank: A Case Study in Management of Community Data, *Current Proteomic*s, 2004, 1, 49-57.

[3] Welch, V. Grid Security Infrastructure Message Specification. Open Grid Forum, 24 February 2006, Document ID GFD-I.078. http://www.ogf.org/documents/GFD.78.pdf

[4] Welch, V. (ed.) Globus Toolkit Version 4 Grid Security Infrastructure: A Standards Perspective. http://www.globus.org/toolkit/docs/4.0/security/GT4-GSI-Overview.pdf

[5] Cooper, D., S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk. Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile. IETF RFC 5280 (Standards Track), May 2008. http://www.ietf.org/rfc/rfc5280.txt

[6] Tuecke, S., V. Welch, D. Engert, L. Pearlman, and M. Thompson. Internet X.509 Public Key Infrastructure (PKI) Proxy Certificate Profile. IETF RFC 3820 (Standards Track), June 2004. http://www.ietf.org/rfc/rfc3820.txt

[7] Welch, V., I. Foster, C. Kesselman, O. Mulmo, L. Pearlman, S. Tuecke, J. Gawor, S. Meder, and F. Siebenlist. X.509 proxy certificates for dynamic delegation. In Proceedings of the 3rd Annual PKI R&D Workshop, April 2004.

[8] Novotny, J., S. Tuecke, and V. Welch. An Online Credential Repository for the Grid: MyProxy. Proceedings of the Tenth International Symposium on High Performance Distributed Computing (HPDC-10), IEEE Press, August 2001, pages 104-111.

[9] Basney, J., M. Humphrey, and V. Welch. The MyProxy Online Credential Repository. Software: Practice and Experience, Volume 35, Issue 9, July 2005, pages 801-816.

[10] Welch, V., J. Barlow, J. Basney, D. Marcusiu, and N. Wilkins-Diehr. A AAAA model to support science gateways with community accounts. Concurrency and Computation: Practice and Experience, Volume 19, Issue 6, March 2007.

[11] Scavo, T. and V. Welch. A Grid Authorization Model for Science Gateways. Concurrency and Computation: Practice and Experience (to appear), 2008.

[12] Welch, V., T. Barton, K. Keahey and F. Siebenlist. Attributes, Anonymity, and Access: Shibboleth and Globus Integration to Facilitate Grid Collaboration. 4th Annual PKI R&D Workshop, 2005.

[13] Barton, T., J. Basney, T. Freeman, T. Scavo, F. Siebenlist, V. Welch, R. Ananthakrishnan, B. Baker, M. Goode and K. Keahey. Identity Federation and Attribute-based Authorization through the Globus Toolkit, Shibboleth, GridShib, and MyProxy. 5th Annual PKI R&D Workshop, 2006.

[14] Welch, V., I. Foster, T. Scavo, F. Siebenlist, C. Catlett, J. Gemmill and D. Skow. Scaling TeraGrid Access: A Testbed for Identity Management and Attribute-based Authorization. TeraGrid 2007. http://grid.ncsa.uiuc.edu/papers/welch-tg07-Idm-final.pdf

[15] A Resource Management Architecture for Metacomputing Systems. K. Czajkowski, I. Foster, N. Karonis, C. Kesselman, S. Martin, W. Smith, S. Tuecke. Proc. IPPS/SPDP '98 Workshop on Job Scheduling Strategies for Parallel Processing, pg. 62-82, 1998.

[16] M. Feller, I. Foster, and S. Martin. "GT4 GRAM: A Functionality and Performance Study", TeraGrid Conference 2007.

[17] W. Allcock (editor), GridFTP: Protocol Extensions to FTP for the Grid. GFD-20, April 2003. http://www.ggf.org/documents/GFD.20.pdf

[18] Reliable Data Transport: A Critical Service for the Grid. W.E. Allcock, I. Foster, R. Madduri. Building Service Based Grids Workshop, Global Grid Forum 11, June 2004.

[19] Social Informatics Data Grid, Bennett Bertenthal1, Robert Grossman, David Hanley, et al, E-Social Science 2007 Conference, October 7-9, 2007, Ann Arbor, Michigan, US. http://ess.si.umich.edu/papers/paper184.pdf

[20] Virtual Data System, http://www.ci.uchicago.edu/wiki/bin/view/VDS/VDSWeb/WebMain

[21] CHILDES, http://childes.psy.cmu.edu/

[22] TalkBank, http://talkbank.org/

[23] C. Youn, C. Baru, K. Bhatia, S. Chandra, K. Lin, A. Memon, G. Memon, D. Seber, GEONGrid portal: design and implementations, *Concurrency and Computation:*

Practice and Experience, Vol. 19, Issue 12, pp 1597-1607 (2007), DOI: 10.1002/cpe.1129.

[24] C. Youn, T. Kaiser, C. Santini and D. Seber. Design and Implementation of Services for a Synthetic Seismogram Calculation Tool on the Grid. *ICCS 2005: 5th International Conference*, Atlanta, GA, USA, May 22-25, 2005, Proceedings, Part 1, LNCS 3514, pp. 469-476, 2005.

[25] Kurt Mueller, Karan Bhatia, Sandeep Chandra. GAMA: Grid Account Management Architecture. *IEEE e-Science* 2005, Melbourne, Australia, Dec 2005.

[26] C. Youn, C. Baru, N. Wilkins-Diehr, Scientific Application Portal Development for Research and Education in Cyberinfrastructure, Geoinformatics 2008 Conference, GeoForschungsZentrum Potsdam, Germany, June 11-13 2008.

[27] Donnellan, A., J. Rundle, G. Fox, D. McLeod, L. Grant, T. Tullis, M. Pierce, J. Parker, G. Lyzenga, R. Granat, M. Glasscoe, QuakeSim and the Solid Earth Research Virtual Observatory, *PAGEOPH*, 163, 1–17, 2006.

[28] Pierce, M. E., G.C. Fox, G. Aydin, Z. Qi, A. Donnellan, J. Parker and R. Granat QuakeSim: Web Services, Portals, and Infrastructure for Geophysics December 19 2007, published in 2008 IEEE Aerospace Conference March 1-8 2008, Big Sky MT.

[29] Jay Alameda, Marcus Christie, Geoffrey Fox, Joe Futrelle, Dennis Gannon, Mihael Hategan, Gopi Kandaswamy, Gregor von Laszewski, Mehmet A. Nacar, Marlon E. Pierce, Eric Roberts, Charles Severance, Mary Thomas: The Open Grid Computing Environments collaboration: portlets and services for science gateways. Concurrency and Computation: Practice and Experience 19(6): 921-942 (2007)

[30] J. Parker, G. Lyzenga, C. Norton, C. Zuffada, M. Glasscoe, J. Lou, A. Donnellan,. "Geophysical Finite Element Simulation tool (GeoFEST): algorithms and validation for quasistatic regional faulted crust problems", *PAGeoph* 165, 497-521, 2003.

[31] Pallickara, S. L., and Marlon Pierce, "SWARM: Scheduling Large-scale Jobs over the Loosely-Coupled HPC Clusters." Submitted to e-Science 2008.

[32] Grant L.B., A. Donnellan, D. McLeod, M. Pierce, G.C. Fox, A.Y. Chen, M.M. Gould[1], S.S. Sung, P.B. Rundle, A Web Service Based Universal Approach to Heterogeneous Fault Databases, *Computing in Science and Engineering Special Issue on Multi-Physics Modeling*, 51–57, July/August 2005.

[33] The Geophysical Resources Web Service: http://reason.scign.org/scignDataPortal/grwsSummary.jsp

[34] Aydin, Galip, Zhigang Qi, Marlon E. Pierce, Geoffrey C. Fox, Yehuda Bock Architecture, Performance, and Scalability of a Real-Time Global Positioning System Data Grid 17 January 2007, Special issue on Computational Challenges in Geosciences in PEPI (Physics of the Earth and Planetary Interiors) 163 (2007) 347-359

[35] Shrideep Pallickara, Geoffrey Fox: NaradaBrokering: A Distributed Middleware Framework and Architecture for Enabling Durable Peer-to-Peer Grids. Middleware 2003: 41-61

[36] Gurnis, Michael, gurnis@caltech.edu, "Science Gateway project description". Email to Nancy Wilkins-Diehr 5 May 2008.