

# A CyberGIS Gateway Approach to Interoperable Access to the National Science Foundation TeraGrid and the Open Science Grid

Anand Padmanabhan<sup>1</sup>, Shaowen Wang<sup>1</sup>, John-Paul Navarro<sup>2</sup>

<sup>1</sup>Department of Geography and National Center for Supercomputing Applications, University of Illinois at Urbana-Champaign

<sup>2</sup>Argonne National Laboratory, University of Chicago

{apadmana, shaowen}@illinois.edu, navarro@mcs.anl.gov

## ABSTRACT

The vision of creating a “virtual supercomputing” environment to solve large-scale scientific problems has largely been facilitated by the development and deployment of Grid middleware. However, with the deployment of multiple disconnected Grid environments, we are now faced with the problem of interoperable access to resources from multiple environments to meet the requirements of scientific applications. Within the U.S. cyberinfrastructure environments, two key elements: both the National Science Foundation TeraGrid and the Open Science Grid (OSG) provide varied but important capabilities and resources needed by diverse computational communities. Hence, it is critical to understand how these communities can benefit from bridging these different environments and utilize them when needed. In this paper we present a novel approach to interoperable access to both OSG and TeraGrid to users through the CyberGIS Gateway – an online geographic information system. In particular, five key interoperability themes are addressed: authentication and authorization, information services, data management, and computation management and auditing. We take a scientific application use-case (viewshed analysis) on the CyberGIS Gateway to demonstrate how to exploit resources on both OSG and TeraGrid.

## Categories and Subject Descriptors

**D.2.12 [Interoperability]:** Distributed objects

## General Terms

Documentation, Design, Standardization.

## Keywords

CyberGIS, Distributed Computing, Science Gateway, Cyberinfrastructure Interoperability

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

*TeraGrid '11*, July 18-21, 2011, Salt Lake City, Utah, USA.

Copyright 2011 ACM 978-1-4503-0888-5/11/07...\$10.00

## 1. INTRODUCTION

Computational Grids were envisaged as a geographically distributed heterogeneous environment built with resources aggregated from multiple administrative domains to create a collaborative and highly useable “virtual supercomputing” environment to tackle computationally intensive applications [1]. The fruition of this vision has largely been facilitated with the development and deployment of Grid middleware (e.g., Condor, Globus [2], Unicore, gLite) software. However, with the existence of many national, international, and domain specific Grids, there are multiple “virtual supercomputing” resources that users have to navigate to find available resources for their scientific applications. If users are to leverage all available resources, it is imperative to connect these Grid environments, which requires the interoperability of different Grids to provide seamless interfaces across Grid environments for application and user access. Failing to achieve such interoperability would limit the potential for realizing the vision of cyberinfrastructure.

In the U.S. both the OSG and TeraGrid offer production services to computational scientists and form the backbone of the national cyberinfrastructure [3-6]. However, the nature of computational resources offered are different; with TeraGrid primarily focusing on capability oriented High Performance Computing (HPC) systems contrasted by the capacity oriented High Throughput Computing (HTC) [7] resources provided by OSG. Though many end-users are satisfied with the resources provided within one of these infrastructures, nevertheless, a number of scientific communities (e.g., geospatial science and technology) have applications that could benefit from resources provided by both Grids. Hence, bridging OSG and TeraGrid to interoperate as a seamless Grid through a science gateway [32] can enhance the power of cyberinfrastructure and scientific research experiences.

This paper presents our experience of building an interoperable CyberGIS Gateway ([8, 9]), deployed with the GISolve Toolkit [10, 11], which affords the geospatial science and technology community users a unified interface to exploit the capabilities offered by both OSG and TeraGrid. Achieving a seamless interoperation from the Gateway is however non-trivial and involves developing adaptors and connectors to bridge existing technologies deployed on OSG and TeraGrid. We detail the techniques that were employed to enable the use of OSG and TeraGrid through the common Gateway interface, with a specific focus placed on the following areas: authentication and authorization, information services, data management, job

submission and execution, and auditing. There are indubitably operational and management differences between OSG and Tergid; e.g. resource provisioning; however, this paper will only focus on technical differences that are bridged by the Gateway in order to facilitate seamless use of cyberinfrastructure resources. In particular, we will take a scientific application use-case: a visibility analysis method that effectively leverages both types of cyberinfrastructure environments.

## 2. BACKGROUND

In this section we will highlight the current research into Grid interoperability and introduce the CyberGIS Gateway to motivate the need for interoperability between OSG and TeraGrid for the GIScience application community.

### 2.1 Grid Interoperability Research

There has been considerable amount of work on bridging multiple Grids. The efforts can be generally classified into a) standardization efforts (e.g. interoperability effort led by Grid Interoperability Now (GIN) Community Group within Open Grid Forum (OGF)) that aim to facilitate a truly global Grid through community standards; and b) interoperation efforts that specifically focus on bridging on two or more current production Grids. In addition to developing standards towards the long-term goal of a unified Grid interface, the GIN initiative also focused on promoting interoperation within a short timescale and has contributors from various Grids from Europe (e.g. European Grid Infrastructure (EGI)), North America (e.g. OSG, TeraGrid), Asia Pacific (e.g. Pacific Rim Application and Grid Middleware Assembly (PRAGMA)) and Australia (e.g. Australian Partnership for Advanced Computing (APAC)).

The standardization work is motivated by the well-known problem of incompatible interfaces on various Grid and middleware service stack faced by both users and application developers. The OGF has been the de-facto organization to lead the standardization effort on the Grid. Open Grid Services Architecture (OGSA) [12], was one such initial effort aiming to use web-services concepts and technologies to standardize capabilities encompassing resource management, data, security, execution management, and information. Some of the standards developed by OGF working groups (e.g. Job Submission Description Language (JSDL)) are supported by a number of widely used middleware stack [13]. Long-term solutions to facilitate Grid interoperability typically includes recommendations that are based on open standards. These recommendations were made for each of the different areas (e.g. Authentication, Job Management)

In addition to standardization efforts, many of the production Grids have embarked interoperability efforts with their peer Grids. GIN produced a report detailing the activities and recommending production solution to enable short-term interoperation between Grids [13]. These are generally adapters, tweaks or workarounds that are designed to provide an engineering solution to the problem. The middleware and infrastructure interoperation effort between these Grids tend to be pair wise and focused on a particular area. For example Flechl et al [14] detail the status of interoperation of information services on production Grids. Some of the information services interoperate quite well (e.g. OSG and EGI) and are deployed in production, while other (e.g. TeraGrid and EGI) were still missing important pieces of information. These efforts are generally driven by individual application use-cases and [14] suggest even if we continue with interoperation in

this fashion, each additional effort will bring us closer to the goal of a unified Grid.

It is important to note that in spite of all standardization and interoperability effort, we haven't achieved full interoperability because of continued use of non-standard interfaces, differences in deployment, configuration, environment and the use of different versions and implementations of standards. Hence we believe that even though having a unified standard based solution for the whole problem would be ideal, it remains imperative that we develop short-term application-focused solution to Grid interoperation. Though Reidel et al [13] distinguish between interoperability and interoperation to differentiate between standards-based and engineering-focused solution, in this paper we use these terms interchangeably.

### 2.2 CyberGIS Gateway - A Brief Introduction

The purpose of CyberGIS Gateway is to provide a web-based and cyberinfrastructure-enhanced high-performance, distributed, and collaborative problem-solving environment for multi-disciplinary researchers to perform large-scale spatial analysis and modeling, and help non-technical users directly access Grid capabilities ([8][15]). Furthermore the quality and quantity of geospatial data has increased manifold over the past few years, and there are no signs this will slow down in the foreseeable future. Hence, the Gateway has to deal with collection, management and processing of increasingly large- and multi-scale geographically referenced datasets that have become extremely important for scientific discoveries and to support decision-making in a number of fields (e.g. ecology, public health, environmental engineering, and disaster response). The Gateway bridges cyberinfrastructure and GIScience to resolve compute- and data-intensive spatial analysis and modeling applications and provides sustainable computational power to facilitate collaborative geospatial problem-solving through advanced analysis techniques. The Gateway is designed to be 1) scalable: in number of OSG and TeraGrid computational resources accessible to solve complex GIS problems and in number of GIScience community users supported; 2) useable: provides user-friendly capabilities that can be shared and reused by a large number of GIScience community users to enable effective community collaboration; 3) interoperable: individual capabilities are designed to interoperate with each other and with OSG and TeraGrid capabilities; and 4) sustainable: have consistent interfaces and APIs to allow for advancements in GIS analysis techniques and Grid technologies.

The Gateway architecture is service-oriented and each spatial analysis is treated as a service that is composed of several component services with user interface support, distributed resource and data management, and security enforcement. The Gateway provides a high-performance, distributed, and collaborative GIS to integrate cyberinfrastructure-based spatial analysis capabilities for shared access by a large number of users. The Gateway affords us with a unique GIS problem solving environment, to study the challenges of effectively utilizing, managing and interoperation of a hybrid cyberinfrastructure environment (local cluster, TeraGrid, OSG), and evaluate the solutions developed.

On CyberGIS, hundreds of registered users can simultaneously perform spatial analyses including, for example, a parallel surface interpolation analysis [16, 17], a spatial statistical analysis using Grid computing [18], a viewshed analysis [19], and a high-performance and collaborative method for Bayesian

spatiotemporal geostatistical modeling based on a Markov chain Monte Carlo approach [20]. The daily throughput of the Gateway usage can reach several tens of thousands normalized CPU hours. The use of HPC resources from TeraGrid and HTC resources from the OSG is not only critical to support the multitudes of Gateway users simultaneously conducting their analysis but also the different varieties of GIS analysis some of which might not be feasibly run using desktop GIS. For example the interpolation analysis can be performed within a minute by many users simultaneously through the Gateway, while each such analysis would require several hours, or may even be infeasible using desktop GIS (e.g., the ArcGIS Spatial Analyst) because of the data size and the limitation of desktop computing power.

### 3. TERAGRID-OSG INTEROPERATION

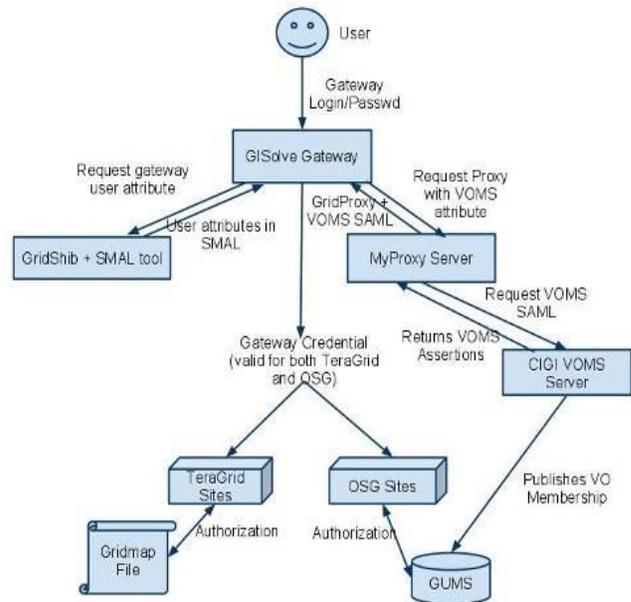
In this section we outline the different interoperation issues we encountered and how we addressed them within the Gateway. In particular we have identified five key areas where we encountered substantial differences between the Grids that needed to be bridged. For each of these areas we present techniques and adaptors developed for the Gateway to provide a seamless coupling of OSG and TeraGrid resources from the Gateway user’s perspective. Table 1 highlights the adaptors that bridge each of these areas.

Area	TeraGrid	OSG	TG-OSG Bridge in Gateway
Authentication and Authorization	Grid proxy + Gridshib SAML	Grid proxy + VOMS SAML extensions	MyProxy + VOMS + Gridshib SAML tool
Information Services	TeraGrid IIS + Inca	OSG BDII +RSV	Glue XML & LDAP schema + REST API + wrapper scripts
Data management	GridFTP	GridFTP + SRMs	Globus + OSG client tools + wrapper scripts + DB table
Job Submission and Execution	GRAM2 + WS-GRAM+ GRAM5 + Community account user environment	GRAM2+ VO user account + worker node environment	Globus client tools + Condor-G + wrapper scripts for scheduling, submission and tracking, execution and result handling of user jobs
Accounting and Audit	GRAM auditing DB + TeraGrid Central DB	Gratia DB	REST API + GratiaDB query + wrapper scripts

**Table 1: Gateway adaptors bridging OSG and TeraGrid**

### 3.1 Authentication and Authorization

Both OSG and TeraGrid use X.509 public key infrastructure; where a well-known third party (i.e. Certificate Authority (CA)) is relied on to establish the trust fabric and to establish user identity. The Gateway uses a certificate issued by The National Center for Supercomputing Applications (NCSA) CA. Since the NCSA CA is accredited by the International Grid Trust Federation (IGTF) it is included in the trust fabric of OSG and TeraGrid sites. Hence we are able to use the same certificate on both OSG and TeraGrid, to authenticate requests originating from the Gateway.



**Figure 1: Authorization in Gateway to support both OSG and TeraGrid**

However authorization, on OSG and TeraGrid is different. On TeraGrid authorization is done by using the gridmap files maintained by each site and through distributed account creation processes. All of the TeraGrid sites the Gateway is authorized to use create a community account and add the Distinguished Name (DN) of Gateway’s community certificate in their local gridmap file. Authorization on the OSG is typically done using Grid User Management System (GUMS), though the use of gridmap files by sites are also supported. The trust fabric on the OSG consists of Virtual Organizations (VOs) acting as an intermediary between users/Gateway and sites. So instead of establishing a trust relationship with individual users OSG sites select VOs they wish to trust and support. Based on this list, the authorization systems periodically query the Virtual Organization Management System (VOMS) that are maintained by the VOs, to get a list of VO user (i.e. users the VO trusts) along with the authorities/roles. Hence, in order to use the OSG, we registered a VO named CyberInfrastructure and Geospatial Information Laboratory (CIGI) with OSG, and installed and registered a VOMS server for the CIGI VO. The Gateway communities’ certificate’s DN is

listed as a user in the CIGI VOMS. Figure 1 illustrates the various components and the interactions of the authorization system.

In order to accomplish Grid tasks (e.g., data transfer, job submission and monitoring, querying TeraGrid information services) a grid-proxy is needed, which the Gateway generates on demand. The Gateway portal checks proxy store (in portal memory) for a valid proxy, and if a valid proxy not available, a new proxy is created with default lifetime (one week). Generating a simple grid-proxy is however not sufficient, because some OSG sites (e.g. Fermilab) require the proxy to have VO attribute assertions in the proxy, which essentially are Security Assertion Markup Language (SAML) attributes provided by VOMS, while TeraGrid sites on the other hand require Gateway user attributes (e.g. user id, email, timestamp of authentication, remote IP) that are able to individually identify users. The presence of VOMS extension allows for the local mapping system (e.g. GUMS) to correctly map the DN to a local Unix user account even when the same user certificate is part of multiple VOs. The Gateway uses the TeraGrid MyProxy [21] server in order to generate a VOMS proxy (i.e. a X.509 grid proxy with the VOMS SAML attribute set). The Gateway then uses the Gridshib SAML tool [22] provided for the gateways by TeraGrid to add attributes that individually identify the Gateway user. This process is outlined in figure 1. The extended Gateway credential, so generated, can be used on both OSG and TeraGrid. The SAML extensions are independent, and hence the OSG sites are able to safely ignore the Gateway user attributes while the TeraGrid sites are able to intercept and interpret them. In a similar vein the VOMS SAML extensions are ignored by TeraGrid and verified by OSG sites.

To summarize, in order to authorize on OSG the Gateway had to register a VO, set up VOMS server to provide the list of VO users to OSG sites, while in order to meet TeraGrid's mandate that Gateway's provide information to uniquely identify users we developed a wrapper that uses the API's provided Gridshib SAML tool to add SAML attributes. Both Grids use the grid-proxy obtained from the MyProxy server.

## 3.2 Information Services

In order to identify suitable resources for any particular application and/or user we need a system that provides information about available systems and a uniform mechanism to query them. The information services system is designed to do just that. For example scheduling jobs requires a wide variety of information such as the endpoint of service, the current load on a system, memory per node, number of processors and the expected wait time for a queued job. The OSG maintains a top level Berkeley Database Information Index (BDII); consisting of two or more LDAP [23] databases; that is populated periodically by the sites using Generic Information Provider (GIP) [24]. The information published follows the GLUE 1.3 schema [25]. TeraGrid, on the other hand provides Integrated Information Services (IIS) [26] APIs to facilitate information access. In the Gateway we use the REST API provided by IIS to get both capabilities and information published using GLUE 2.0 and other custom XML schemas. A module collects and combines the data provided by information services of both Grids and provides them through a uniform API to the other Gateway modules. The information provided by this module enables the data transfer module to determine endpoints for initiating data transfer. Similarly the job scheduling module is able to associate this information with the application characteristics and is able to determine which TeraGrid/OSG site the direct the users jobs to.

As TeraGrid information service is being enhanced to publish Gateway user-related job/data statistics, Gateway information module is also evolving to provide TeraGrid usage information to end users and for Gateway administration.

Additionally both OSG and TeraGrid provide service monitoring information that gives the status of the resources from the user's perspective. On TeraGrid, Inca provides a user level monitoring of Grid sites that is used to determine the availability of a particular service. A similar grid monitoring service called Resource Service Validator (RSV) is available on the OSG. Though these monitoring systems are not currently used in our Gateway, we plan to incorporate them in the future.

For the purposes of information needed by our Gateway, both OSG and TeraGrid provide the same information, though this information is published in different format. Hence the information service system includes modules that are able to read different schemas (e.g. GLUE 2.0 XML, GLUE 1.x LDAP, GLUE 2.0 LDAP) and then provides a common interface/API by which other Gateway system can access this Grid information.

## 3.3 Data Management

In order to support a wide-variety of applications many of which process high-resolution multi-scale spatio-temporal datasets the Gateway has to handle large datasets and their movements between the Gateway and the Grids. In particular the Gateway needs to manage data replication, synchronization, transfers, and archival and the data management system is designed to provide this support. The Gateway maintains a local data repository in addition to storage resources available on OSG and TeraGrid. A database is maintained to keep track of the location and versions various data items. When a dataset is needed by an application this database is consulted to find the list of Grid sites where that particular data item is available. To optimize resource usage and reduce data transfer loads, the Gateway first look at targeted site to see if data exists. Our strategy also recognizes and handles the data purging policies at sites. Only if the most recent version of the data is not available at the site where we plan to execute the job is a transfer initiated through a data management module.

The transfer itself is initiated using a Globus client tools (e.g. a wrapper around globus-url-copy). The local Gateway's data repository is exposed for data transfer using a gridftp server, while OSG and TeraGrid sites also provide either gridftp and/or srm (Storage Resource Manager) [27] endpoints. To communicate with srm endpoints the Gateway uses client tools provided by the OSG (e.g. srmcp). The data transfer module determines the source endpoint using the Gateway database and the target endpoint with the aid of information services system and initiates the transfer. Third party transfers between sites on OSG and TeraGrid are also supported. Once the transfer is initiated, it keeps track of its progress and updates the database when transfer has been successfully completed. If the job execution module is waiting on data transfer to execute a user experiment, a notification is also send to it. If the data transfer fails, the module retries a few times to complete the transfer. If still unsuccessful a failure is recorded in the database and any waiting user executions are sent back to be rescheduled by the scheduling module. In the future we hope to make use of Globus Online [28] in order to simplify data management routines especially by exploiting the error handling features to enhance reliability of data transfers.

For long-term storage and archival of data, the Gateway uses both local storage and the Gateway's TeraGrid allocations. Datasets

that are deemed important by Gateway users get flagged for long-term storage. These dataset are then transferred to storage location by data transfer module and appropriate entries are made in the database.

The current search capability on the Gateway is rudimentary (based on file and experiment names) and one of the areas targeted for future improvement. We plan to incorporate provenance metadata information to be able to do a semantic search on data from both the traditional attribute oriented search as well as spatio-temporal search.

Apart from the need to strategically manage large number of datasets and coordinate storage between local, OSG and TeraGrid site, there is no inherent interoperability issues with the underlying data transfer protocols. GridFTP is widely supported on both infrastructure and even storage elements based on SRM typically use gridftp endpoints on the backend to accomplish data transfer. Of course the SRM interface provides support for additional functionality (e.g. space reservation); however at the present time these are not used by the Gateway.

### 3.4 Job Submission and Execution

Job submission and execution system is a critical component of the Gateway. A suite of modules have been built in the Gateway to support execution of user jobs both OSG and TeraGrid including job scheduling, job submission and tracking, job execution and result handling.

The Gateway supports a number of classes of applications e.g. stochastic, parametric study, workflows, and applications that use high performance computing techniques and hardware such as message passing interface (MPI), openMP, and GPGPUs (General Purpose Graphics Processing Units). Some of these high performance computing applications are suited to run on TeraGrid resources, while other that involve Monte-Carlo simulations and parametric studies are more suited for high-throughput resources provided in OSG. Some applications on the Gateway (e.g. viewshed analysis (see case study)) have multiple implementations (an embarrassingly parallel version suitable for OSG and a GPGPU version for TeraGrid) available that could run on either Grid.

#### 3.4.1 Job Scheduling

When a Gateway user submits an experiment, the user interface notifies the job submission and execution system with a message. The job scheduling module will get the characteristics of the application being submitted from the database. This will determine the range of computing resources that could be used for executing this experiment. The module will then query the information services system to obtain status of the various OSG and TeraGrid resources that were deemed suitable. It will then merge this information with the availability of datasets needed for the experiments. Since many of the application use spatial data where spatial distribution of the dataset has potentially a large impact on the computational resources needed, the scheduler employs computational intensity maps [29] to quantify resource requirements. The scheduler will then collectively evaluate those constraints and decide where to schedule the job. The scheduler used by the Gateway employs a min-max fairness approach [18]. Additionally, we also have an experimental distributed resource broker [30] that is part of spatial middleware that is able to match application runs to Grid resources using computational intensity.

If the dataset needed for an application is not available at a site chosen to execute the job the scheduling system contacts the data management system

#### 3.4.2 Job Submission and Tracking

Grid Resource Allocation and Management (GRAM) service is used as a common interface for remote job submission and control on both OSG and TeraGrid. However GRAM2, WS-GRAM (Web Services-GRAM), and GRAM5 are all being used by TeraGrid sites, while OSG sites primarily use GRAM2 service on production compute elements (CEs). On the Gateway, under the covers is a Globus client tools wrapper (e.g. globus-job-submit) used to submit jobs to the TeraGrid, however submitting jobs to OSG employs Condor-G [31] due to community standards and recommendations. A wrapper script prepares the configuration files, input parameters, and job submission files to enable execution on the Grid. Though it is possible to use Condor-G for submitting jobs to both OSG and TeraGrid, however the requirement to track individual users will necessitate dynamic creation of Condor submit files. So even though it is currently possible to implement, this will require tweaking with the job submission system, and due to historical reasons this has not been implemented. However, there is a plan to implement this change in the future.

Due to the differences in the job submission techniques for OSG and TeraGrid, monitoring job status is also performed independently. The job tracker module uses Globus and Condor client tools (e.g. globus-job-status, condor\_q) to determine the status of user submissions. The job tracker runs periodically to monitor the status of all non-completed user submissions and updates a database. The Gateway periodically reads this database and updates the user interface.

Additionally, for long running jobs it is essential to keep the proxy current to ensure that jobs can continue running. For jobs submitted to the TeraGrid the Gateway uses the Globus proxy renewal tool; however this mechanism is unavailable for jobs submitted to WS-GRAM and hence the Gateway uses a longer term proxy for jobs submitted to WS-GRAM sites. For the jobs submitted to OSG only the local Condor-G proxy needs to be renewed before it expires, and Condor-G takes care of pushing the proxies out to the sites that are running Gateway jobs.

#### 3.4.3 Job Execution

The job execution environment in OSG and TeraGrid is significantly different. In fact even the process of installing and compiling the application are substantially dissimilar. On the TeraGrid, the principal investigator (PI) or a designated Gateway admin will have to login through ssh to the community account, and compile the application and set specific environment needed using the SoftEnv or Modules system in the home directory of community account. The user job submitted will have to run this preinstalled executable. The Gateway admin will have to install the Gateway applications on all the TeraGrid sites the community plans to use for a particular application. On the OSG sites applications do not need to be preinstalled, though a pre-installation on a shared application directory is permitted, and the job payload can be included as a part of the job execution process. In fact if the application code is not found in the shared application directory, the job wrapper script for experiments running on the OSG sites will be able to download either the source code or binary executable from a trusted Gateway machine.

Further on OSG and TeraGrid the directories from which the job gets executed (i.e. working directory), temporary directories for intermediate data, input/output data directory, home directory, etc, are different. The main problem with these is that the software, hardware, deployment, configuration, and environment differences can potentially have significant impact on application performance. For example some sites have SSD for local drives (e.g. Trestles at San Diego Supercomputer Center on TeraGrid), which should be exploited if the application happens to be I/O bound, while some OSG sites by policy require that local disks and not shared network drives be used to write intermediary data. Hence the Gateway has developed its own standard that can be appropriately controlled for different Grid sites and provides wrapper scripts to automatically copy data from network to local disks when appropriate. The job execution scripts on the Gateway define environment variables like \$CYBERGIS\_HOME, \$CYBERGIS\_TMP, \$CYBERGIS\_APP, \$CYBERGIS\_DATA, \$CYBERGIS\_CWD, signifying location of home directory, temporary directory on worker nodes, application directory location, data location, and current working directory respectively, to facilitate a seamless environment for Gateway application developers. For a Gateway job running on an OSG site, the job wrapper script will set these variables will respectively be set to the VO's home directory, \$OSG\_WN\_TMP, \$OSG\_APP/\$vo/app, \$OSG\_DATA/\$vo/data, and the location set batch system. The \$OSG\_\* variables are defined independently by each of the OSG sites. On TeraGrid these variables are set to the target site's code deployment directory (usually in home directory or community software area (e.g., on the Steele cluster)), scratch file system, and permanent data storage (project area). By providing a job wrapper scripts and defining \$CYBERGIS\_\* environment variables the Gateway is able to execute applications on both the Grid infrastructures.

Through the approach based on environment variables is an elegant method for achieving seamless integration between Grids within the Gateway, there are some legacy applications that cannot be changed to conform to the new approach. For such applications we write an application specific job wrapper for both OSG and TeraGrid, which sets all the parameters, input data files, and configuration files as required by the application. This approach is more challenging and time consuming to implement, however it facilitates integration of legacy code within the Gateway without reimplementing. The use of the common environment variables enables the Gateway is able to support different configurations of worker node storage environment (e.g. use of local disk v/s a network file system) and improves portability and masks the environment differences between Grid sites.

#### 3.4.4 Result Handling

After the job has been successfully executed on the Grid, the Gateway has to collect the standard output/error and the results directory (indicated by environment variable \$CYBERGIS\_RESULT) and provide it to the user. Standard output and error files are staged back by the client tool (both Globus and Condor client) following the execution of the job. On TeraGrid the working directory of the job is in a scratch space and outputs are stored there while on OSG the jobs are expected to restrict their heavy I/O operations to \$OSG\_WN\_TMP, which usually points to a local disk location. These local files on OSG are expected to be transferred to shared data directory (\$OSG\_DATA), by the job wrapper upon completion of the job

otherwise they will not be accessible, while the scratch data on TeraGrid remains accessible for a well-defined period of time (typically a fortnight). The job wrapper submitted by the Gateway to OSG and TeraGrid, is designed to handle these differences and provides transparent use of both Grid resources Gateway users. These datasets are then transferred to stable long-term storage.

### 3.5 Accounting/Audit

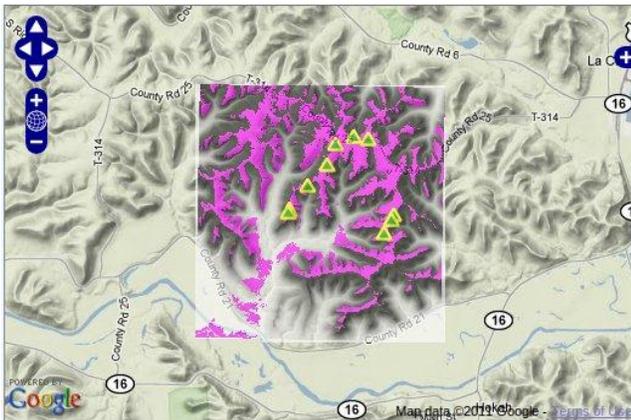
Keeping track of the overall usage of resource as well as those by its individual user is crucial for the Gateway. On TeraGrid, the SAML attribute that identifies a unique user is used to record the information into the TeraGrid's GRAM audit and central database and the Gateway uses the REST API to query this information. The OSG, on the other hand records accounting information into its Gratia accounting system based solely on the DN and the SAML VOMS attribute. Since it does not collect the Gateway user specific information, the OSG's Gratia database provides only an overall usage of resources by the Gateway. Hence for jobs submitted to OSG the Gateway itself keeps record of individual user activity. However, since this tracking is done at a Gateway level, the information will be less accurate than the wall clock time if it were directly reported by the batch system, and may only be for estimation purposes. However, the use of resources provided by OSG sites, by the Gateway is accurately obtained by querying the Gratia database.

## 4. CASE STUDY

In this section we will consider a specific application running in the Gateway, a concrete scientific use-case that uses resources from both OSG and TeraGrid. The specific example we consider is viewshed analysis [19], a well-known GIS analysis. A viewshed of a specific location essentially identifies the area that is visible from that particular location (figure 2). The analysis takes in a Digital Elevation Model (DEM) that identifies the elevation of each cell, as input and determines the visibility to and from a particular cell. The straightforward algorithm for calculating the viewshed involves shooting a ray from the source to the target cell and determining the line of sight using azimuth or viewing angle.

Common applications of viewshed analysis include landscape planning, placements of communication towers, observation stations and military assets. For some application domains increased accuracy requirements will require analyzing high-resolution datasets, while for others the terrain being analyzed is extremely large or both. Hence, viewshed analysis conducted through the Gateway tends to be both compute- and data-intensive. There are two typical use cases of this experiment on the Gateway: 1) calculate viewshed of a small number of cells (typically up to 10) on high-resolution large-scale terrain; and 2) determine the visibility from all the cells on a number of medium-scale terrains. The first use-case typically deals with an interactive exploratory analysis being conducted the Gateway user, where results are expected reasonably quickly. The use of HPC resources provided by TeraGrid is a perfect candidate for this use-case. On the Gateway we have an implementation of viewshed analysis which exploits the general purpose graphics processing units (GPGPUs) available on TeraGrid. This implementation is specifically designed to optimize data and memory transfer and is suitable for calculating viewshed for a small number of cells. For the second use-case viewshed for every single cell needs to be calculated, which for a large number of terrain dataset can essentially boil down to a HTC problem, with tens of thousands of

independent viewshed calculations. The resources provided by OSG are well-suited for tackling this problem.



**Figure 2: Viewshed analysis of 10 view points on DEM data of an area in La Crosse, WI. Computation is done on the Lincoln GPU cluster at NCSA.**

In order to submit an instance of viewshed analysis application, a Gateway user first logs in to the Gateway with his username and passphrase. S/he then locates the appropriate application and submits the job by specifying the input datasets and the application parameters through the user interface. Once the application is submitted the Gateway contacts the authentication and authorization system to obtain a Gateway credential. The user job is then sent to the job scheduling module, which consults the database and the information services system to determine what sites may be suitable for this job. The data management system ensures that the input DEM files are transferred to selected sites using gridftp. For HPC jobs the job scheduling module directs it to one GPGPU cluster on TeraGrid (e.g. Lincoln at NCSA), while for an HTC job the job submission system splits the single user job into a number (one for every cell in the DEM) of sub-jobs. These sub-jobs are individually submitted to different OSG sites through Condor-G. If the application executable is not deployed at a chosen OSG site, a wrapper script is run (through the fork job manager) that will download the source code and build it on the VO's shared application location (\$OSG\_APP/\$vo).

If any of the sub-jobs in the workflow fails, it is resubmitted, and the user job is marked as complete only after all the sub-jobs have been successfully completed. Upon completion of each sub-job the partial results are moved into the VO's shared data directory (\$OSG\_APP/\$vo). These partial results are transferred back to the Gateway by the data management module and then merged as part of the post-processing step either executed locally on the Grid. The results of the analysis are stored in a Postgis database table (Postgres+GIS extensions) and the visualization is aided by Geoserver, an open source mapping server, and OpenLayers, an open source client side web mapping API. The Gateway thus facilitates the user to interactively visualize the viewshed results.

## 5. CONCLUDING DISCUSSION

Providing a unified interface to cyberinfrastructure is imperative to enable users to leverage all available resources in Grid environments. In this paper we presented our novel approach to provide seamless interoperable access to both of the U.S. national cyberinfrastructure environment, the OSG and TeraGrid, thereby affording interoperable access to varied capabilities and resources

from multiple environments to meet the requirements of scientific applications, through the CyberGIS Gateway. The approach described can be adopted and the lessons learned can benefit diverse computational communities attempting to bridge different environments. By providing access to compute resources from both OSG and TeraGrid simulations are run on the most appropriate resource and turnaround time for users is decreased and the user experiences in enhanced.

We highlighted techniques utilized to bridge the OSG and TeraGrid in five key areas: authentication and authorization, information services, data management, job submission and execution and accounting. Finally by employing viewshed analysis as a scientific use-case we demonstrate how the CyberGIS Gateway is capable of utilizing the resources of both the OSG and TeraGrid to resolve computational challenges.

## 6. ACKNOWLEDGMENTS

This material is based in part on work supported by the National Science Foundation Grant OCI-\*1047916\*. This research was done in part using resources provided by the Open Science Grid, which is supported by the National Science Foundation and the U.S. Department of Energy's Office of Science. This research was supported in part by the National Science Foundation through TeraGrid resources. We will specifically like to acknowledge the assistance of Yan Liu, Eric Shook, and Yanli Zhao for their help with CyberGIS development. Finally, we would also like to thank Nancy Wilkins-Diehr, Ruth Pordes and Yan Liu for their useful feedback on the draft.

## 7. REFERENCES

- [1] I. Foster, C. Kesselman and S. Tuecke, "The anatomy of the grid: Enabling scalable virtual organizations," *International Journal of High Performance Computing Applications*, vol. 15, pp. 200-222, 2001.
- [2] Globus, "The Globus Project," <http://globus.org>, 2011.
- [3] TeraGrid, "The NSF TeraGrid Project," <http://teragrid.org>, 2011.
- [4] Catlett, C. and et al, "TeraGrid: Analysis of organization, system architecture, and middleware enabling new types of applications," in *Advances in Parallel Computing*, Lucio Grandinetti ed. Amsterdam, 2008.
- [5] R. Pordes and e. al., "The open science grid," *Journal of Physics: Conference Series*, vol. 78, pp. 012057, 2007.
- [6] OSG, "The Open Science Grid Project," <http://opensciencegrid.org>, 2011.
- [7] M. Livny, R. Raman, T. Tannenbaum and J. Basney, "Mechanisms for High Throughput Computing," 1997.
- [8] S. Wang, "A CyberGIS Framework for the Synthesis of Cyberinfrastructure, GIS, and Spatial Analysis," *Annals of the Association of American Geographers*, vol. 100, pp. 1-23, 2010.
- [9] CyberGIS, "The CyberGIS project," <http://cybergis.org>, 2011.
- [10] S. Wang, M. P. Armstrong, J. Ni and Y. Liu, "GISolve: A grid-based problem solving environment for computationally intensive geographic information analysis," in *Proceedings of the 14th International Symposium on High Performance Distributed Computing (HPDC-14) – Challenges of Large Applications in Distributed Environments (CLADE) Workshop*, IEEE Press, 2005, pp. 3-12.
- [11] GISolve, "The GISolve Toolkit," <http://gisolve.org>, 2011.

- [12] K. Karasavvas, M. Antonioletti, M. Atkinson, N. Hong, T. Sugden, A. Hume, M. Jackson, A. Krause and C. Palansuriya, "Introduction to OGSA-DAI Services," *Lecture Notes in Computer Science*, pp. 1-12, 2005.
- [13] M. Riedel, et. al., "Interoperation of world-wide production e-Science infrastructures," *Concurrency and Computation: Practice and Experience*, vol. 21, pp. 961-990, 2009.
- [14] M. Flechl and L. Field, "Grid interoperability: joining grid information systems," *Journal of Physics: Conference Series*, vol. 119, pp. 062030, 2008.
- [15] S. Wang and Y. Liu, "TeraGrid GIScience Gateway: Bridging cyberinfrastructure and GIScience," *International Journal of Geographical Information Science*, vol. 23, pp. 631-656, 2009, 2009.
- [16] S. Wang and M. P. Armstrong, "A quadtree approach to domain decomposition for spatial interpolation in Grid computing environments," *Parallel Computing*, vol. 29, pp. 1481-1504, 2003.
- [17] S. Wang, "Spatial interpolation," in *Encyclopedia of Geography* Anonymous SAGE Publications, Inc, 2010, .
- [18] S. Wang, M. K. Cowles and Marc P. Armstrong, "Grid computing of spatial statistics: using the TeraGrid for Gi\*(d) analysis," *Concurrency and Computation: Practice and Experience*, vol. 20, pp. 1697-1720, 2008.
- [19] P. F. Fisher, "An exploration of probable viewsheds in landscape planning," *Environment and Planning B: Planning and Design*, vol. 22, pp. 527-546, 1995.
- [20] J. Yan, M. K. Cowles, S. Wang and M. P. Armstrong, "Parallelizing MCMC for bayesian spatiotemporal geostatistical models," *Statistics and Computing*, vol. 17, pp. 323-335, 2007.
- [21] J. Basney, M. Humphrey and V. Welch, "The MyProxy online credential repository," *Software: Practice and Experience*, vol. 35, pp. 801-816, 2005.
- [22] Gridshib SAML Tool, "Science gateway credential with attributes," [http://www.teragridforum.org/mediawiki/index.php?title=Science\\_Gateway\\_Credential\\_with\\_Attributes](http://www.teragridforum.org/mediawiki/index.php?title=Science_Gateway_Credential_with_Attributes), 2011.
- [23] T. Howes and M. Smith, *LDAP: Programming Directory-Enabled Applications with Lightweight Directory Access Protocol*. Indianapolis, IN, USA: Macmillan Publishing Co., Inc, 1997.
- [24] GIP, "Generic Information Provider," <http://vdt.cs.wisc.edu/components/gip.html>, 2011.
- [25] S. Androozzi, et al., "Towards GLUE 2: evolution of the computing element information model," *J. Phys.: Conf.* 2008
- [26] L. Liming, et al., "TeraGrid's integrated information service," in *Proceedings of the 5th Grid Computing Environments Workshop*, Portland, Oregon, 2009, pp. 8:1-8:10.
- [27] A. Shoshani, A. Sim and J. Gu, "Storage Resource Managers: Middleware Components for Grid Storage," 2002.
- [28] Globus, "Globus Online," <http://globusonline.org>, 2011.
- [29] S. Wang, "Formalizing computational intensity of spatial analysis," in *Proceedings of the 5th International Conference on Geographic Information Science*, Park City, Utah, USA, 2008, pp. 184-187.
- [30] A. Padmanabhan and S. Wang, "A distributed resource broker for spatial middleware using adaptive space-filling curve," in *Proceedings of the ACM SIGSPATIAL International Workshop on High Performance and Distributed Geographic Information Systems*, San Jose, California, 2010, pp. 27-30.
- [31] J. Frey, T. Tannenbaum, M. Livny, I. Foster and S. Tuecke, "Condor-G: A Computation Management Agent for Multi-Institutional Grids," *Cluster Computing*, vol. 5, pp. 237-246, 2002.
- [32] N. Wilkins-Diehr, D. Gannon, G. Klimeck, S. Oster and S. Pamidighantam, "TeraGrid Science Gateways and Their Impact on Science," *Computer*, v.41 n.11, p.32-41, November 2008.