FERMILAB-CONF-07-451-CD

The Open Science Grid Status and Architecture

The Open Science Grid Executive Board on behalf of the OSG Consortium:

Ruth Pordes, Don Petravick: Fermi National Accelerator Laboratory

Bill Kramer, Doug Olson: Lawrence Berkeley National Laboratory

Miron Livny, Alain Roy: University of Wisconsin, Madison

Paul Avery: University of Florida

Kent Blackburn: California Institute of Technology

Torre Wenaus: Brookhaven National Laboratory

Frank Würthwein: University of California, San Diego

Ian Foster, Rob Gardner, Mike Wilde: University of Chicago

Alan Blatecky, John McGee: Renaissance Computing Institute

Rob Quick: Indiana University

Corresponding Author: ruth@fnal.gov[1]

Abstract. The Open Science Grid (OSG) provides a distributed facility where the Consortium members provide guaranteed and opportunistic access to shared computing and storage resources. The OSG project[1] is funded by the National Science Foundation and the Department of Energy Scientific Discovery through Advanced Computing program. The OSG project provides specific activities for the operation and evolution of the common infrastructure. The US ATLAS and US CMS collaborations contribute to and depend on OSG as the US infrastructure contributing to the World Wide LHC Computing Grid on which the LHC experiments distribute and analyze their data. Other stakeholders include the STAR RHIC experiment, the Laser Interferometer Gravitational-Wave Observatory (LIGO), the Dark Energy Survey (DES) and several Fermilab Tevatron experiments– CDF, D0, MiniBoone etc. The OSG implementation architecture brings a pragmatic approach to enabling vertically integrated community specific distributed systems over a common horizontal set of shared resources and services. More information can be found at the OSG web site: www.opensciencegrid.org.

1. The Open Science Grid

The Open Science Grid Consortium[2] is a collaboration of research groups, software providers, and processing and storage facility organizations working together to maintain and evolve a significantly sized shared distributed infrastructure supporting a broad range of scientific research. The OSG Consortium includes scientific communities from single researcher to several-thousand person collaborations, software development projects providing the technologies and middleware, and DOE

[1] Fermilab is supported by the U.S. Department of Energy under contract No. DE-AC02-76CH03000.

laboratory and many (DOE, NSF and other) university facilities providing networking, processing and storage resources. The owners of the resources control their access with policies that support use by other OSG members and with an expectation that 10-20% are on average available to other than the owners. The OSG facility provides the distributed data handling and analysis infrastructure in the United States for the Large Hadron Collider (LHC) and is used by LIGO, the nuclear physics STAR experiment, Dark Energy Survey (DES) simulations, Sloan Digital Sky Survey object analysis, as well as several Fermilab experiments such as CDF, D0 and MiniBoone. These applications drive the scale and schedule of the OSG to the multi-Petabyte data and Teraflop processing levels. The members of the Consortium contribute the resources available to the OSG.

The OSG currently supports biology, nanotechnology, weather research and mathematics research groups to make effective use of and/or contribute resources to the infrastructure. In addition, we are working with other non-physics groups to adapt their applications for production running. This has resulted in increased use of the resources by communities who do not own them and in an increase in the sites and resources available. The majority of the science communities participating in the Consortium are global in nature. The OSG is also working on gateways to other distributed infrastructures at the campus and regional level as well as internationally. The OSG does not develop software and negotiates with external technology projects[3] for development and support of capabilities needed by its stakeholders.

In summary, the members of the OSG are working towards the vision that our work will "transform compute and data intensive science through a national cyber-infrastructure that includes from the smallest to the largest organizations" and that this vision includes bridging local, national, & international cyber-infrastructures, and engaging communities, facilities and software providers to work together towards the common goals.

2. OSG Today

There are more than sixty active computational sites on the OSG providing access to about thirty-thousand cores. There are fifteen managed storage sites on the OSG providing a total several petabytes in disk caches. Job throughput is currently more than seventy-five thousand jobs a day, using more than two hundred and fifty thousand CPUhours. The number of unique user identities on the infrastructure is of the order of several hundred. About twenty of the sites are part of the US LHC distributed data handling and analysis systems (Brookhaven and Fermilab Tier-1, University Tier-2s). These sites are supporting ongoing data distribution tests of up to a terabyte a day.

Four sites are owned by LIGO and are being used for transitioning analysis codes from the existing LIGO data grid to full production on the common infrastructure. Four sites are owned by STAR and are being used to bring STAR data distribution, simulation and production codes to work effectively across more than the current sites. The Tevatron experiments have already benefited the delivery of their physics results by using the OSG: CDF runs monte-carlo simulations opportunistically across six OSG sites and analysis across four OSG sites locally at Fermilab; and over the past six months D0 used more than 2 million cpuhours and transferred about 100 Terabytes of date, processing over 50% of a 500 million event dataset, using opportunistically available resources on the OSG, in time for the summer conferences in 2007.

The Open Science Grid (OSG) engagement activity works closely with new user communities over periods of several months to bring them to production running. These activities include: providing an understanding of how to use the distributed infrastructure; adapting applications to run effectively on OSG sites; engaging in the deployment of community owned distributed infrastructures; working with the OSG Facility to ensure the needs of the new community are met; providing common tools and services in support of the new communities; and working directly with and in support of the new end users with the goal to have them transition to be full contributing members of the OSG. In the nine months since the start of the OSG project engagement activities have resulted in the following use of OSG[4]:

- Adaptation and production running opportunistically using more than a hundred thousand CPUhours of the Rosetta[5] application from the Kuhlman Laboratory in North Carolina across more than thirteen OSG sites which has resulted in structure predictions for more than ten proteins. We have so far tested the robustness of the system to the submission of up to about three thousand jobs simultaneously and seamlessly support the users when they make periodic submissions. (see Figure 1)

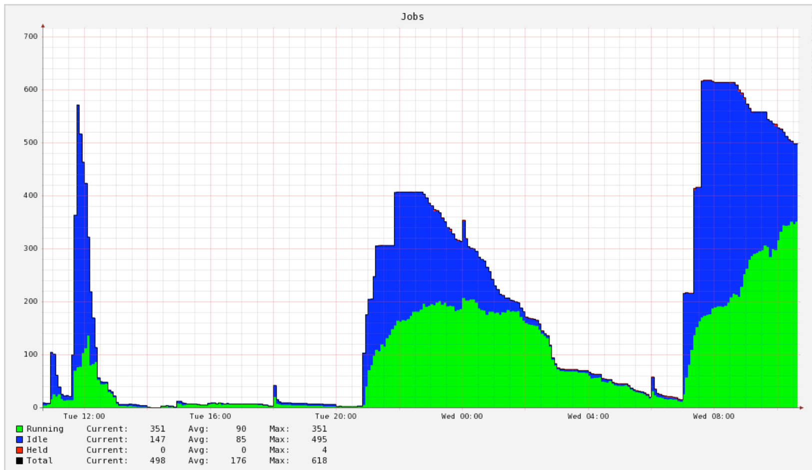

Figure 1: Periodic Runs of Rosetta on OSG

- Production runs of the Weather Research and Forecast (WRF)[6] application using more than one hundred and fifty thousand CPUhours on the NERSC OSG site at Lawrence Berkeley National Laboratory and running across three other OSG sites. We are addressing issues of different MPI versions and configurations.
- The nanoWire application from the nanoHUB [7] project running across sites on the OSG and TeraGrid.
- Production running using more than thirty thousand CPU hours of the CHARMM [8] molecular dynamic simulation to the problem of water penetration in staphylococcal nuclease using the ATLAS workload management system, PANDA [9], and opportunistically available resources across more than ten OSG sites.

There are several problems we have run into with the current users of such a heterogeneous multi-site facility, in particular:

- *Ensuring that users can run across many sites.* Each site is autonomously managed and has a configuration tailored to the local needs. When a user first starts to use a site we often find uniqueness in the specific libraries, environment variables, shared file systems that need to be "tweaked". We extend our test programs each time we run into these problems – and we are gradually seeing improvement.
- *End-to-end communication of and response to downtimes, failures and errors:* These problems can range from a site saying it is going down for a few days for maintenance and the user not realising this until several thousand jobs have been submitted and failed to the "black hole", to a completely generic error code from some layer of the middleware that could indicate a problem in any one (or multiple) of a large set of software layers and services.

The OSG trains new entrants to distributed computing through hands-on workshops, online tutorials and integrating and publishing training material and courseware. These schools build on the successful iVDGL summer grid workshops that were held annually for the past four years. The OSG training schools continue to be very well received. In the first year of the project three training schools [10] have held in Chicago, Texas and the University of Nebraska, Lincoln with more than a hundred and fifty students in all. We have also provided teaching materials as well as instructors to international schools in Argentina and a workshop in Taiwan.

3. Implementation Architecture

The OSG architecture [11] defines interfaces between resources and applications and the common infrastructure. OSG provides an integrated and tested reference set of software the Virtual Data Toolkit (VDT) [12] for the resource site administrators and the user communities to use to interface to and use the OSG distributed facility.

The OSG implementation architecture is cognizant that any resource may be supporting use through multiple interfaces—local submission and access, from the OSG, and using other similar infrastructures such as Campus Grids or other large common infrastructures such as TeraGrid [13] in the US or the Enabling Grids for EScience (EGEE) infrastructure in Europe. The implementation architecture also supports user groups using multiple infrastructures simultaneously. The user communities may have a deep set of sometimes complex software and services that are specific to their applications and operate across the infrastructures. These are additional drivers to the model that sites retain local control and management of their own resources and environments, and that the user communities have control and management over their global systems. Site and user group management processes are implemented to support sharing and opportunistic use of the resources accessible to the OSG, in addition to support for prior agreements on levels of service and resource use between resource owners and users,.

The OSG provides common services across the distributed facility: monitoring, validation and information about the full infrastructure; tracking of any and all problems and ensuring they are resolved; the Virtual Data Toolkit software packaging and support; integration and testing facilities; security; troubleshooting of the end-to-end system; and support for existing and new user communities. The OSG also provides effort to bring new services and software into the facility and to collaborate with the external projects, as well as documentation and training of site and user group administrators and users. OSG documents the interfaces supported. It allows for any implementation of these interfaces to be used to access the services behind them.

Service Interface	**Implementation**
Job Execution: single client (Condor-G [14]) multiplexing to multiple servers	Web service and non-web service GRAM, Condor-glide-ins.
Data Transfer: GridFTP [15]	Three implementations of OGF standard
Local Data Access	Environment variables on the site let users know where to put and leave files.
Storage Management: SRM [16]	Two implementations of common community standard
VO Management	Single implementation based on EGEE [17] VOMS interface.
Accounting	Single implementation based on OGF standard usage record.

Table 1: OSG Service Interfaces

Issues we are facing with the current architecture are related to:

- The immaturity of the services and components to defend themselves against overload and unintentional misuse by the application software.
- The impossibility of testing all configurations in such a heterogeneous environment before putting new software into production.
- The lack of software to support dynamic sharing of storage on a distributed set of resources.

- The lack of the infrastructure to support (iteratively) sub-groups within a community in a transparent and equivalent manner to the groups themselves. This is necessary to allow management of the resources within a community (where one of these communities is the OSG itself).

4. The Production Facility

Use of the OSG facility has been increasing over the past six months, with up to 15 science groups using multiple sites, and an ongoing usage of about ten thousand batch slots dc. US ATLAS and US CMS have made significant contributions to the global experiment job processing and data distribution challenges being done in preparation for data taking next year.

The recent D0 reprocessing, an unexpected request for several months use of up to two thousand batch slots, was a very instructive experience in learning how we, as an organization as well as an operating infrastructure, can handle significant, unanticipated, requests for the use of resources it has made accessible. The ramp up time to ensure all sites worked for the application was several months. Unique problems were experienced at each site as D0s reprocessing application was commissioned – especially in the storage and use of data. Additional problems and overheads were experienced scaling the whole system up to the throughput needed. Root cause analysis and troubleshooting of the quite different hardware configurations at the sites was a challenge; and during the steady state operations phase we found inefficiencies in our processes for notification and responding to site downtimes and problems.

The reprocessing was a success. It proved that sites on OSG can and do provide substantial sharing of their resources and user communities can rely on significant effective throughput from resources they do not own. It also showed that sustained teamwork between the application and infrastructure groups is essential throughout. And in addition, it showed that we in SG need to start preparing now for the expected future where the resources on OSG are oversubscribed and prioritization across the infrastructure is needed.

5. A Common, Integrated Software Stack

OSG software releases consist of the collection of software integrated and distributed as the Virtual Data Toolkit (VDT) (see Figure 2) with a thin layer of additional OSG specific configuration scripts. Modules in the VDT are included at the request of the stakeholders. The Condor and Globus software provide the base technologies. VDT includes about thirty additional modules, including components from other computer science groups, EGEE, DOE Laboratory facilities (Brookhaven, Fermilab and LBNL), and the application communities themselves, as well as standard open source software components such as Apache and Tomcat. OSG also supports the VDT for external projects. – for example the EGEE and Australian distributed computing infrastructures. Additionally, in support of interoperability across their infrastructures, the OSG and TeraGrid software stacks include aligned versions of the Condor and Globus software.

The VDT provides a reference software stack for use by OSG sites. Once the software is installed a site supports remote job submission, shared storage at a site, data transfer between sites, has services to manage priorities and access between user groups, and can participate in the OSG monitoring, services validation and accounting services. OSG supports the use of the reference software, providing front line support in some components and communication with the software developer groups is other. The VDT also provides client libraries and tools for the applications to use to access OSG resources and (See Table 2)

Compute Element	Locus of the OSG site services for accounting, validation, information publishing, job submission, etc.
Processing Worker Node	Includes client tools for data access.
Storage Element	Implementations of space management and Grid interfaces
VO management	Registration and user role and access management

Client	Client interfaces to the data, job, information and VO services.

Table 2: Integrated Collections of VDT Modules

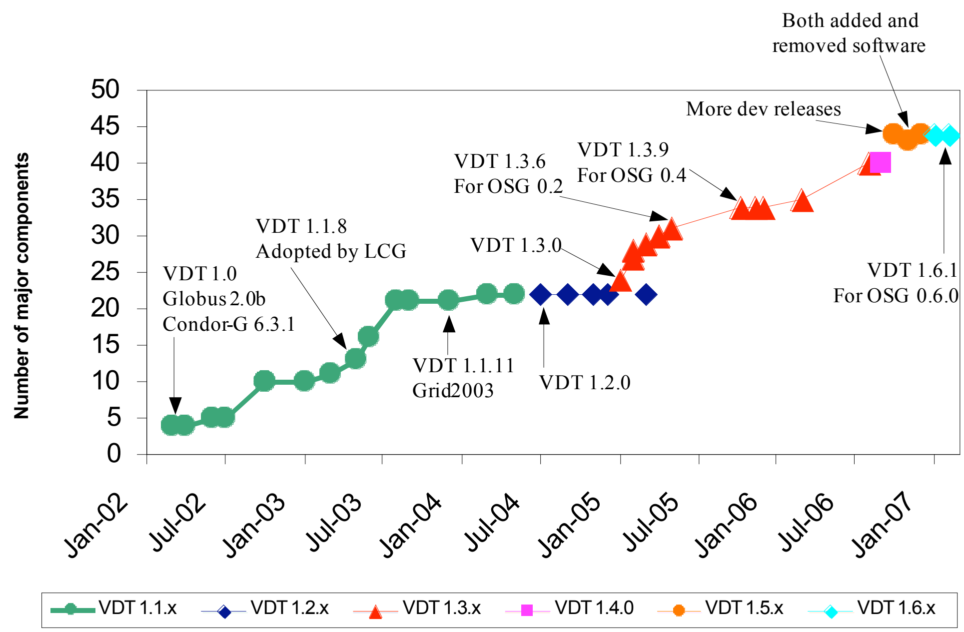

Figure 2: Timeline of VDT Releases

Before a new release of the VDT, all components are built up to ten Linux platforms (as well as some limited builds on AIX and MacOSX) using the NSF Middleware Initiative build and test facility [18] and, by re-using builds, the VDT tested on up to fifteen platforms. The validity of the integrated release is checked—for example many components use Apache web server and we check that usage is compatible—and validated on a small number of well-controlled sites (the Validation TestBed, VTB). It is then tested on a many (~15) site infrastructure, the Integration Testbed (ITB). The work of the integration activity is to check not only the individual services and software components but also that the system and applications work at scale.

User communities typically have application specific software and services layered above the OSG software stack. Most install their software using Grid methods and some do so dynamically when they are submitting their jobs. The OSG does not constrain the application level software and services except if they have negative security or performance impact on the operation of the OSG itself or other users. The integration activity is responsible for testing the complete matrix of horizontally integrated OSG software and vertically integrated user group software.

The OSG provides tools for incremental installation of new releases and patches, verification of installed configurations, and for functional testing of the sites. We are currently developing an extended system of functional tests which will enable site administrators to more easily monitor and manage their own installations, as well as alert them when problems occur.

Security and robustness of the software as deployed is a significant concern and an ongoing area of activity on the production infrastructure. We are developing tools to audit the use of and probe the software interfaces to determine overload and "denial of service" conditions. Addressing some of the following issues are part of our activities for the coming year:

- Quick and efficient patches of the OSG stack and redeployment in response to security notifications.

- Balancing the amount and effort spent on testing with the need to get new services “to the user” quickly. The stability of the resulting infrastructure becomes at risk since testing invariably does not cover the full set of usage patterns.
- Adding functionality (driven by the user communities) balanced with the need to make the software stack minimal and low impact.
- Integration of diverse software components from multiple software suppliers with different levels of development maturity and different release cycles.

6. The Security Architecture

The OSG Security architecture includes operations, management and technical aspects. The OSG policies, risk analysis and security plans [19] have become models for other projects. We have exercised the security notification and response process more than ten times in the past year. Notification is followed by a prompt analysis of the problem and assigning high priority to patches and fixes, with the timeframe for the fixes being from a few hours to a few weeks. The VDT, Condor, Globus, and EGEE software teams communicate security risks as soon as they are identified and work together with us on patches and solutions. The collaborative nature of the OSG means that communication is natural and happening all the time and security is part of the day-to-day normal processes. OSG has mechanisms to deny user’s access to sites and resources. The grouping of users into Virtual Organization (VO) communities or groups gives us a small number of well-identified responsible managers who control user entry to the infrastructure. This leads us to a model of trust between resource owners, user groups and the OSG, with delegated trust between the group managers and the end users.

The OSG security infrastructure is based on: X509 user, host and service certificates gained through one of the International Grid Trust Federation accredited Certificate Authorities; user identity proxy certificates obtained through the VO Management Service (EGEE VOMS) which provides checking of the user as part of a VO; management of extended certificate attributes to assign “roles” to a particular access by a user; flexible definition of mapping of user certificates to accounts and ACLs (access control lists, [20]) on a site; and policy (including blacklist) enforcement points at the site (processing and storage) services themselves. Several significant issues remain with this infrastructure: Not all sites yet implement these authorization components which restricts the options available for VOs to manage their policies;There are gaps in the infrastructure which prevent smooth communication of user group policies to all sites; The creation and maintenance of accounts is a manual process for the site administration; The validation and trust infrastructure between services and software components is lacking; Certificates with multiple attributes are not handled consistently.

7. Job Management and Execution

OSG sites present interfaces allowing remotely submitted jobs to be accepted, queued and executed locally with priority and policies of execution are affected both by the VO and the site itself. User group policies are defined through roles given to the user through the VOMS service. Site policies and priorities are defined through mapping the user and their roles to specific accounts used to submit the job to the batch queue. OSG supports the Condor-G job submission client which interfaces to either the pre-web service or web services GRAM Globus interface at the executing site. Job managers at the backend of the GRAM gatekeeper support job execution by local Condor, LSF, PBS, or SGE batch systems.

Stand-alone Condor match-making, the generalized D0 Resource Selection Service (ReSS) [21], and the EGEE resource broker (RB) provide alternate ways for users to automatically select which site jobs are dispatched to based on user criteria. Both RESS and the RB depend on the OSG site information services which present information about the resources using the Glue Schema [22] attributes and providers and optionally converting the information to Condor Classads.

The largest science communities on OSG use “pilot job” mechanisms to increase the VOs ability to manage the prioritization of jobs across their organizations in a manner transparent to the site, and thus increase the overall throughput. Pilot jobs are submitted through the normal Condor-G mechanism. When the VO pilot job starts execution, using a standard batch slot, it interacts with its partner "VO job scheduler" to download the application executable to be run. The VO job scheduler controls the scheduling of jobs between the users in the VO and schedules jobs to run only on those resources that are immediately ready to execute them. These user jobs execute under the identity of the pilot job submitter—which can break the policy that sites must be able to identify the end user of their resources. OSG has integrated an Apache suexec [23] derivative, the EGEE glexec module, that enables the pilot to run jobs under the identity of the originating user.

8. Data Transport, Storage and Access

Many of the OSG physics user communities have large file based data transport and application level high data I/O needs. The data transport, access, and storage implementations on OSG take account of these needs. OSG relies on GridFTP protocol for the raw transport of the data – using Globus GridFTP in all cases except where interfaces to storage management systems (rather than file systems) dictate individual implementations. The community has been heavily involved in the early testing of new versions of Globus GridFTP as well as defining needed changes in the GridFTP protocol.

OSG supports the SRM interface to storage resources to enable management of space and data transfers to prevent unexpected errors due to running out of space, to prevent overload of the GridFTP services, and to provide capabilities for pre-staging, pinning and retention of the data files. OSG currently provides reference implementations of two storage systems the LBNL Disk Resource Manager (Bestman) and dCache [24]. In addition, because functionalities to support space reservation and sharing are not yet available through grid interfaces, OSG defines a set of environmental variables that a site must implement and a VO can rely on to point them to available space, space shared between all nodes on a compute cluster, and for the use of high-performance I/O disk caches.

The major challenges for data storage and access are usable (low effort and stable) installation, configuration and operation; supporting a full range of disk storage resource sizes from several to hundreds of TeraBytes; how to provide guaranteed and opportunistic sharing between and within VOs; the on-time delivery of hundred terabyte data sets, and providing applications with high throughput access to data at a local or remote site.

9. Gateways to other Facilities and Grids

We are seeing a rapid growth in the interest and deployment of shared computational infrastructures at the local and regional level. We are also seeing a rapid growth in research communities' needing to move data and jobs between heterogeneous facilities and build integrated community computational systems across high performance computing (HPC) facilities and more traditional computing clusters. OSG can provide interfaces to these HPC facilities to support these use cases. For example, the NERSC facility is becoming a site on the OSG providing MPI based resources for OSG users.

The OSG includes software and support for campus and regional organizations to form their own locally shared facilities and distributed infrastructures which then gateway to the OSG. Three examples participating in OSG are the Fermilab Facility Grid (FermiGrid) [25], the Grid Laboratory of Wisconsin (GLOW) [26] and the Purdue University campus-wide infrastructure. Each local organization contributes to and manages the gateway between the local facility and the OSG.

The OSG also federates with other large infrastructures—notably the TeraGrid and EGEE—by providing gateways between them and OSG, and supporting groups to submit jobs across and move data between them. For example, the OSG collects information from the resources and publishes them in the format needed by the EGEE. The CMS Resource Broker job dispatcher then submits jobs transparently across EGEE and OSG resources.

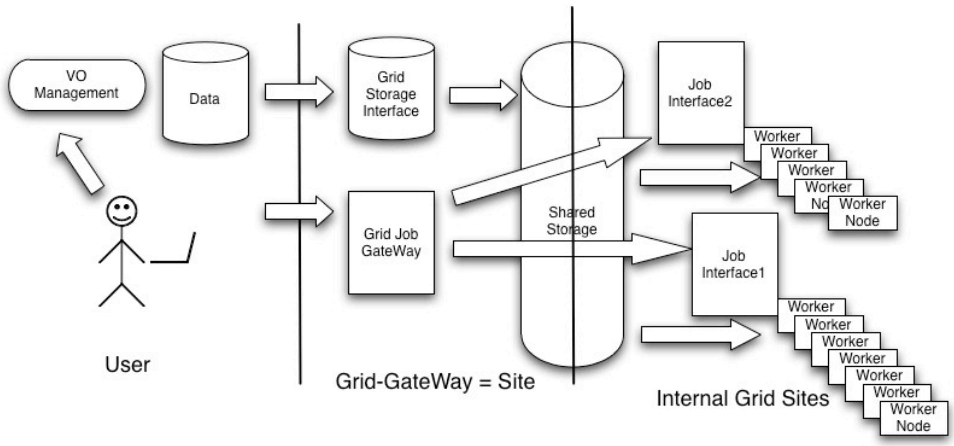

Figure 3: Accessing the FermiGrid Campus Grid from OSG

10. The Future

The turn-on of the LHC and the LIGO and STAR upgrades will result in a ten-fold increase in the data and job throughput required on the OSG over the next three years. These will challenge the scalability, performance and robustness of the infrastructure. In our experience they will also prove a challenge to the functionality and software provided as new needs of the collaborations emerge. In addition, our challenge will be to succeed in our commitments to our balance to bring new (non-physics) communities to production use at least every six months, provide access to new high performance and local computing, and be welcoming to participants and contributors from university, laboratory, national and international groups.

We are aware of the challenges and complexity of how to measure success. Success is not just the number of jobs executed and the amount of data transferred and stored. Success has to be measured by the impact on scientific productivity and maturity of computation as a cornerstone (with experimentation and simulation) of the research portfolio. We are gradually putting in place measurements of many parameters to help us determine the impact, but we do not understand yet how to translate and analyze this information to quantify value and benefit

Acknowledgments

This work was supported by the Office of Science, U.S. Department of Energy, SciDAC program under Contract DE-FC02-06ER41436 and the National Science Foundation Cooperative Agreement, PHY-0621704.

References

[1] OSG Project institutions: Boston University, Brookhaven National Laboratory, Caltech, Cornell University, Columbia University, Fermilab, Indiana University, ISI, Lawrence Berkeley National Laboratory, Renaissance Computing Institute, SLAC, University of California San Diego, University of Chicago, University of Florida, University of Iowa, University of Wisconsin Madison

[2] OSG web site www.opensciencegrid.org

[3] OSG External Projects: Community Driven Improvement of Globus Software (CDIGS), Center for Enabling Distributed Petascale Science (CEDPS), Condor, dCache, Data Intensive Science University Network (DISUN), DRM, Energy Sciences Network (ESnet), Internet2, Laser Interferometer Gravitational-Wave Observatory (LIGO) Physics and the Information Frontier, Fermilab Grid accounting and security projects, UltraLight/advanced networks, US representation of the WorldWide LHC Computing Grid Collaboration.

[4] Engagement Report 2007

http://osg-docdb.opensciencegrid.org/0006/000632/001/OSGEngagementReport200706-V2.pdf
[5] Rosetta web site: https://wl.rosettacommons.org/
Kuhlman Laboratory at the University of North Carolina, Chapel Hill: http://www.unc.edu/kuhlmanpg/
[6] WRF at RENCI web site: www.renci.org/disaster/hydromet.php
[7] NanoHUB web site: http://www.nanohub.org/
[8] CHARMM Chemistry at Harvard Macromolecular Dynamics: http://www.charmm.org/
[9] ATLAS PANDA software: https://twiki.cern.ch/twiki/bin/view/Atlas/Panda
[10] OSG grid schools: https://twiki.grid.iu.edu/twiki/bin/view/Education/WebHome#Upcoming_Grid_Schools
[11] OSG Blueprint Document: http://osg-docdb.opensciencegrid.org/0000/000018/005/OSG-Blueprintv0.10.pdf
[12] The Virtual Data Toolkit web site: http://vdt.cs.wisc.edu
[13] TeraGrid web site: www.teragrid.org
[14] Condor ™ project web site: www.cs.wisc.edu/condor
[15] Globus Tookit™ web site: www.globus.org/toolkit
[16] Storage Resource Management Collaboration: http://sdm.lbl.gov/indexproj.php?ProjectID=SRM
[17] Enabling Grids for EsciencE (EGEE) web site: http://public.eu-egee.org/
[18] Metronome (formerly NMI build and test): http://nmi.cs.wisc.edu/
[19] OSG Security plan: http://osg-docdb.opensciencegrid.org/0003/000389/017/OSGSecurityPlan.doc
[20] VO Services project web site: www.fnal.gov/docs/products/voprivilege/
[21] Resource Selection Service: https://twiki.grid.iu.edu/twiki/bin/view/ResourceSelection/WebHome
[22] Glue Scheme Open Grid Forum Working Group: http://forge.gridforum.org/sf/projects/glue-wg
[23] Apache suexec: http://httpd.apache.org/docs/1.3/suexec.html
[24] dCache Consortium web site: www.dcache.org
[25] FermiGrid web site: http://fermigrid.fnal.gov
[26] Grid Laboratory of Wisconsin web site: www.cs.wisc.edu/condor/glow