1 The Open Science Grid

The Open Science Grid consortium (OSG)[1] is a collaboration of scientific, research and educational communities to build, operate, use and evolve a *shared national high throughput computational facility* based on common concepts, technologies, and processes. The OSG provides an *open collaborative environment for communities of scientists and researchers* to work together on both common and user specific distributed computing problems and solutions. The OSG collaboration includes broad multi-disciplinary representation of scientists and researchers, IT providers, software developers, educators and computing administrators. The OSG partners with peer organizations in the US and abroad to provide more effective solutions for science.

The OSG project is jointly funded by the Department Of Energy SciDAC-2 program and the National Science Foundation for an initial five-year program of work. The project staff maintain the distributed computational facility, provide support for the facility's users, software and services, and manage the interfaces to external contributors and peer grid infrastructures. The OSG users are the communities doing science and research as well as those contributing hardware, software and effort. The user communities drive the capabilities and evolution of the OSG. The multi-agency sponsorship of the OSG provides a unique opportunity for participation at all scales - from individual research PIs and small university campus groups to the thousand researcher global scientific collaborations and large DOE laboratory facilities – with today more than 1000 users having accessed the infrastructure.

There is active use of the OSG by groups from molecular dynamics, protein structure prediction, biology, climate, text mining, and computer science. However, the user communities with the most challenging needs are the large physics collaborations in the United States. The OSG provides the computing infrastructure in the United States for the Large Hadron Collider (LHC) ATLAS and CMS experiments. Other major users are the Laser Interferometer Gravitational Wave Observatory (LIGO)[3], the Tevatron experiments (D0 and CDF)[4] and the STAR Relativistic Heavy Ion Experiment[5]. This diverse mix of (currently) more than thirty user communities and applications is ensuring the evolution of a generic national cyber-infrastructure currently including more than sixty sites.

The OSG provides software to meet the evolving needs of the users. It makes integrated tested software releases based on the OSG's Virtual Data Toolkit (VDT [6]) to enable access to and use of the ensemble of processors and storage. The OSG supports these common technologies for both OSG and other projects. We also train new users in their adoption and use.

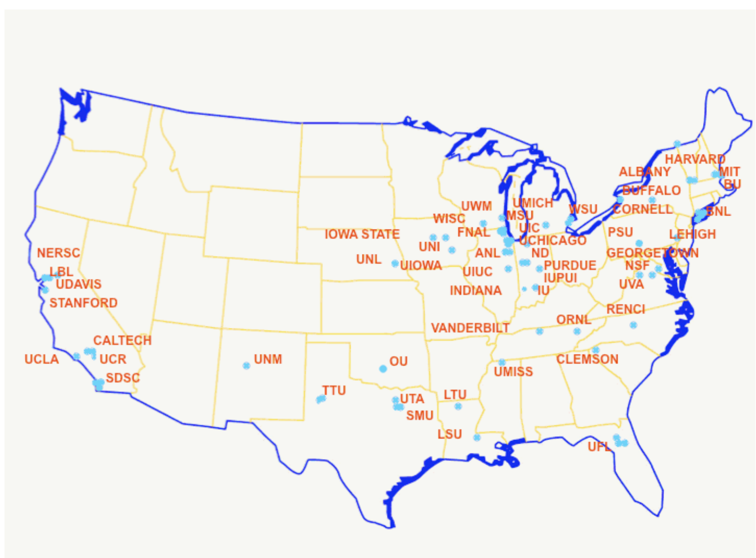

Figure 1: Map of the OSG Sites in the United States

1.1 Purpose of the OSG

The goal of the Open Science Grid is to make collaborative scientific research more effective and widespread, *stimulate new and transformational approaches to computationally based scientific discovery, and build intellectual capital* for future scientific research relying on distributed cyber-infrastructures. The scope of the OSG is *to operate, maintain and evolve an effective secure high-throughput computational infrastructure and engage existing and new communities* to benefit from its use. This heterogeneous, national *facility is defined as the set of operational services, software, and processes* enabling the contributed resources to act as a coherent distributed system in support of the users.

The OSG distributes and supports integrated software suites to provide the services and functionality needed to use and run the facility. The OSG extends the capabilities and capacities of the facility, enables and interfaces to campus and regional cyber-infrastructures, partners with other national and international grids, and collaborates with software developers. The OSG actively engages additional scientific domains and communities. The OSG has a close partnership with the Embedded Immersive Engagement For Cyber-Infrastructure (EIE4CI) NSF CI-Team project[7] to provide additional effort and attention to this very important goal.

The OSG is *not responsible for acquisition and hardware of computer resources* (compute clusters and storage elements); these are maintained by the owners. The OSG does *not develop the software technologies* (neither the grid middleware nor scientific applications); these are acquired from external software development groups. However, the OSG supports integrated software releases. The OSG also works closely with software developers to ensure the current and future needs of the user communities will be met. Through what we term “extensions” activities the OSG contributes to particular projects and, through these mechanisms as well as the testing and use of the software, stimulates new and sometimes transformational technologies and methods. And the OSG does not archive the scientific data; this remains the responsibility of the user communities.

The *OSG does not provide a single complete solution that fits all communities*. Many of the larger user communities have significant middleware and processes that augment those provided by the OSG. Such *communities support and use their own community grid, layered over the common platform* provided by the OSG. The OSG has active collaborations with community projects such as Data Intensive Science University Network[8] and the LIGO Data Grid[9] to leverage the strengths and activities of the different organizations and to help harmonize the resulting systems. On the other hand, in the researchers in smaller communities have little or no time available even to learn how to use the technologies and processes. For such communities the OSG must *ensure a low overhead for use of the common platform*. The OSG thus provides “ready-made” end-user services and software and embedded help for adapting and developing user applications to run on the OSG facility and running any local distributed infrastructures.

1.2 Characteristics of the OSG

In this section we briefly describe six of the main characteristics of the OSG spanning the sociological, conceptual, technical and programmatic aspects.

1.2.1 A Cross-Cutting Collaboration

The main characteristic of the OSG is an *energetic, committed, and sustained collaboration* across the scientists and researchers, computing resource administrators, software providers and OSG staff. A key component is the close collaboration, including the leadership, between the domain scientists in the user communities and computer scientists in the field of distributed and grid computing. This collaboration is proving invaluable in providing a solid computer science foundation to this unique laboratory where the techniques and technologies being developed for collaborative science at all scales are tested, proven, made effective and evolved.

The time and effort needed to maintain this collaborative organization and manage the work being delivered from more than fifteen institutions is significant and receives ongoing attention. It initially took over a year to completely define the governance of the consortium. As the organization matures we revisit the details about every two years.

1.2.2 The Virtual Organization and Community Structure

The OSG methods and processes are based on the *organization, management and use by community groups or Virtual Organizations (VOs).* VOs range from dynamic, ad-hoc collections for a specific short term purpose to long lived stable collaborations with well defined governances. Following the patterns established by the OSG principles, VOs can contain other VOs, sub-VOs. VOs can interface with each other and share resources. VOs can have common services, common organizational policies and methods, and common members (scientists involved in multiple communities). Many communities, deploy, manage and use their own community based distributed systems layered over and dependent on the core OSG platform. In such cases the community's VO registered with the OSG enables members of the user community to use additional OSG resources and services outside of their more parochial system. This also enables university, regional, research and scientific communities with their own grid infrastructures to integrate with and/or rely on some or all of the OSG facility[10].

The OSG itself is a VO with people, resources and services towards a common purpose and governance as described in more detail in this chapter. The OSG provides a grid infrastructure and services which can be shared (in whole or in part) by other communities. The OSG VO includes other VOs for more specific OSG activities such as education and engagement. The OSG provides policies to access the resources contributed to it.

We do not want to belabor or freeze the patterns enabled by the VO concept. The difference between a VO and an O (organization), for example, is small. In our eco-system the only difference is formal responsibility for an individual person or piece of hardware (purchase, installation etc). The concept provides a framework for the development of policies, procedures and technology needed by and between collaborative scientific communities. Its aim is to augment and not preempt the scientific community authorities, responsibilities and programs.

1.2.3 Provision of a Common Shared Distributed Facility and Services

As described above, the OSG provides access to and sharing of the set of autonomous processing and storage resources through operations of a coherent facility. The OSG provides *common, shared services* including monitoring, accounting, security, problem reporting and tracking, towards the goal of operating a robust, effective system[11].

Additionally the OSG provides a *common, shared integration and validation facility*[12] and associated processes to provide functional, performance and full-system testing of new releases of software, services and applications.

The value and characteristics of the facility are that: it provides a stable, documented and supported reference platform on which end-users can run their applications on any and all sites (collections of resources under a single administrative control) in a uniform manner; it provides a proven, sustained platform to which new communities and end-users can adapt and run their applications; it provides a focus for collaboration on many technical and procedural aspects of distributed systems, where many of the needs are in common across multiple user communities.

1.2.4 Provisioning Common Software

As mentioned above, the OSG project *packages, releases, documents and supports a well defined set of software* to enable the interfaces to and use of the contributed resources. This software, including, but not limited to, the VDT, provides the technologies used by OSG as well as other equivalent infrastructures, such as the US TeraGrid and the European Enabling Grids for EScience (EGEE. Each project, including the OSG, augments the VDT with specific configuration scripts and utilities for its own environment and users. The OSG provides software repositories from which the packages can be downloaded, installed and configured on processing, storage, VO management or user client computers.

The VDT currently includes more than forty independent components (see Table 1) from nearly as many software development groups. The modules span generic open source toolkits to those needed and provided by the user communities themselves. There is significant work to organize and integrate these independently developed modules into coherent, harmonized sets for end-user computing, data processing and storage services, and operations and management tools. The VDT is released for many variants of Linux and in client mode for MacOSX and AIX. Processes to build, test, and release the software ensure managed evolution and extension of the capabilities offered.

As an example, many modules use Apache and/or Tomcat, but invariably different versions. These must be accommodated into a single release without bloating the memory footprint, causing negative interaction between the patterns of usage, and enabling the local administrator to use existing installations of the packages whenever possible. As another example, the application software on each of the job and data submission, retrieval and execution sites is very dependent on the specific needs of each science community. This is supported by deciding on and providing a software sub-set common for all communities together with community specific sub-sets that can be compatibly added on and used without disruption to others.

The VDT includes the following software (roughly categorized):

- *Core Grid Infrastructure Software*: Condor[13] and the Globus Toolkit[14].
- *Information Services:* including information providers based on the GLUE specification[15], LDAP repositories, Gratia accounting[16], monitoring and resource validation scripts[17], and resource selection and matching services based on Condor ClassAds[18]
- *Build and testing tools:* the Metronome build and test infrastructure[[18]], regression tests etc.
- *Storage Service Implementations:* BeStMan[20] from LBNL, dCache[21] and XRootd[22].
- *Security tools and infrastructure:* X509 certificate management, VO management services based on X509 extended attributes, authorization and grid to local account mapping tools[23]
- *Client Tools:* utilities for accessing OSG services; libraries to read/write data from/to grid-accessible storage

- sites; workflow tools.
- *Support Software:* Utilities used by many software packages, including Apache, Tomcat, Berkeley DB, MySQL, OpenLDAP, PHP, Squid web caching and miscellaneous VDT tools to help administrators, support staff and users.

Table 1: Contents of the Open Science Grid Virtual Data Toolkit

In the middle of 2008 the OSG released the first major version of the software (V1.0). This release marked a level in the stability and robustness of the software. It marked an increase in level of maintaining a fully functioning production infrastructure during changes in the software. It marked a change in the approach to the release processes and policies. Previously, new releases of the software invariably required a reinstall of the complete set of modules, and a full replica of the release was provided for integration testing and validation. From this point on, new releases will allow incremental testing and upgrades and will support rollback techniques.

1.2.5 Harmonizing Campus, (Inter)National and Community Grids

An important characteristic of the OSG is to provide *transparent user access to their own community's distributed system when it spans multiple, federated (sometimes globally distributed) grids* (see Figure 2).

To this end the OSG works to bridge its infrastructure and services with other grids – from campus, state and national grids, to international and worldwide community infrastructures. Such bridges enable the submission of OSG jobs to other grids, give the ability for OSG sites to accept jobs from other grids, and for the transport and management of data across grid boundaries. This integration and interoperability is crucial for our main stakeholders. For example, to LHC scientists the OSG is "merely" the US part of the larger worldwide WLCG and, as such, should be transparent to them. OSG must ensure interoperation with EGEE and the Nordic National Grid Infrastructure (NorduGrid) in the face of independently evolving software and processes and while supporting a different broader set of user communities.

1.2.6 A Set of Underlying Principles.

The final characteristic of the OSG is the *set of underlying principles that define the concepts and practices* of all its activities[24]. For any OSG activity the principles are applied to the implementation concepts and design, and are measured against the practices and procedures. This contributes towards a coherent, consistent technical path through a very diverse set of developments.

As an example, the principle of "self-protection" is applied during software development and acquisition. Attention to a graceful defense in situations of overload and invalid access helps the developers deliver a more robust and fault tolerant service. Agreed, it sometimes takes many iterations to have the need understood and then implemented. This is even more reason to have a set of foundational concepts as a stable reference.

Principles of the Open Science Grid

- **Phased deployment** with clear operations model: The OSG infrastructure must always include a phased deployment, with the phase in production having a clear operations model adequate to the provision of production-quality service.
- **Policy as the pinnacle**: Policy should be the main determinant of effective utilization of the resources. This implies that without governing policy there would be full utilization of the resources.
- **Symmetry and recursion:** We will follow the principles of symmetry and recursion in any concepts, architectures, designs and implementations developed for the OSG.

- **Minimum impact:** Services should work toward minimizing their impact on the hosting resource, while fulfilling their functions. (Any tradeoff between benefit and impact will constraint their design).
- **Self-protection:** Services are expected to protect themselves from malicious input and inappropriate use.
- **Local rules come first**: All services should support the ability to function and operate in the local environment when disconnected from the OSG environment. This implies the local environment has control over its local namespace.
- **Supplementary services:** OSG will provide baseline services and a reference implementation. Use of other services will be allowed. VOs that require services beyond the baseline set should not encounter unnecessary deployment barriers for the same
- **Incremental shifts:** The OSG infrastructure must be built incrementally. The roadmap must allow for technology shifts and changes.
- **Middle-man:** Users are not required to interact directly with resource providers. Users and (programmable) consumers will interact with the infrastructure and services.
- **Inclusive participation:** The requirements for participating in the OSG infrastructure should promote inclusive participation both horizontally (across a wide variety of scientific disciplines) and vertically (from small organizations like high schools to large ones like National Laboratories).

Best Practice

- The OSG architecture is Virtual Organization based. Most services are instantiated within the context of a VO. The OSG baseline services and reference implementation can support operations within and shared across multiple VOs.
- Services may be shared across multiple VOs. It is the responsibility of the Service and Resource Providers to manage the interacting policies and resources.
- Resource providers should provide the same interface to local use of the resource as they do to use by the distributed services.
- Every service will maintain state sufficient to explain expected errors. There shall be methods to extract this state. There shall be a method to determine whether or not the service is up and useable, rather than in a compromised or failed state.
- The OSG infrastructure will support development and execution of (user) applications in a local context, without an active connection to the distributed services.
- The infrastructure will support multiple versions of services and environments, and also support incremental upgrades.
- The OSG infrastructure should have minimal impact on a Site. Services that must run with superuser privileges will be minimized.
- System reliability and recovery from failure should guarantee that user's exposure to infrastructure failure is minimal.
- Resource provider service policies should, by default, support access to the resource. The principle 'services should protect themselves' thus implies that services should additionally have the ability to instantaneously deny access when deemed necessary.
- Allocation and Use of a Resources and Services are treated separately.
- Services manage state and ensure their state is accurate and consistent.

Table 2: Principles of the Open Science Grid

1.3 Patterns and Usage Modes of the OSG

1.3.1 Using the Facility

In this usage pattern, OSG offers a *data center service relationship to its users as customers.* The value offered is the standing operations, support, and organizational services which a (new or existing) user community can depend on and use with little overhead. The modes of use cover "guaranteed" (where the resources are owned by the user community), "agreed upon expectations" (where there has been negotiation between the user and resource owner communities on the expected level of throughput and support) and "opportunistic" (where the users make use of available resources based on the standard policies of the owners as members in the OSG Consortium).

1.3.2 Using the Engagement VO

In this mode, the OSG *engagement community enables use of the OSG through the reuse of expertise of the OSG staff and existing users.* Such usage includes: easing a community's adaptation to and adoption of OSG technologies and use of the infrastructure; facilitating additional (self-run) campus, regional and partner cyber-infrastructures that expand the total size and capabilities of the resources available; communicating the availability, and encouraging the reuse, of tools and services adopted or developed for or by existing users; and a better understanding of the deeper community which helps the OSG provide more effective services and software.

Once a new community is running in production it is encouraged to become self-sustaining and a registered community member of the OSG consortium.

1.3.3 Community Collaboration

In this usage pattern, the use of and benefit from the OSG is through the *collaborative activities of the consortium members, who contribute to as well as receive benefit* from such collaboration. The benefits include: driving the program of work and priorities of the OSG; access to the historical knowledge of the OSG operations teams; an expanded combined group to plan and execute development activities; and access to an energetic collective of experts - a wider community that cares about the successful and effective outcomes.

1.3.4 Software Development and Testing

While the OSG does not develop software per se, the facility is used as a *platform for developing and testing distributed system technologies.* The software developers collaborate as members of (and through the extensions activity sometimes receive contributions from) the OSG. The OSG understands the user community needs as they change. The OSG inputs to and tracks the developments of the software providers. The OSG provides a standing large-scale infrastructure and measurement techniques for performance and scalability testing and hardening of the software developed.

1.3.5 Use of Multiple Grids

In this usage pattern, the use of *OSG appears to the users as transparent and symmetric with other grids.* The OSG works with the user and grid communities to ensure uniform and transparent interfaces to the application layers and users to support the submission of jobs, the transport and storage of data, and the monitoring, tracking and management of the usage.

As explained above, OSG is but one of many intersecting, overlapping and interacting grid infrastructures spanning from the local to the global sphere. *Services in or cooperating between the infrastructures dispatch and retrieve jobs, data and information transparently* between them to make the total ensemble more effective and efficient.

1.4 Common Usage Modes on the OSG

In this section we separately address common application and resource usage modes.

1.4.1 Application Usage Modes

Applications running on the OSG infrastructure span data simulation and analysis of small (CPU days) to large (CPU centuries) scale scientific application runs. The *OSG facility architecture has special utility for high-throughput computing applications.* The characteristics are large ensembles of loosely coupled parallel applications for which the overhead in placing the application and data on a remote resource is a fraction of the overall processing time. Also, added value is available to computations (loosely coupled and able to run on heterogeneous sites) that can take advantage of opportunistic resources. In summary, OSG is particularly effective for:

- High throughput, pleasantly parallel applications.
- Job runs of between one hour and several days.
- Jobs that can be check-pointed.
- Explicit management of large scale data movement and storage.
- Ensembles that can effectively run across a large number of resources.

Table 4 below summarizes the types and characteristics of applications running on the OSG[27]. Any particular application may have of one or multiple such characteristics, of simulation, production processing, complex workflow, real time response and small-scale parallelism.

	Examples	Job and Data Characteristics
Simulation	• Physics Monte Carlo event simulation. • Protein structure determination.	• CPU-intensive • Large number of independent jobs. • Large run sequences. • Small input data sets; large output data sets.
Production Processing	• Processing of physics raw event data. • Earth observation data processing	• Significant amount of input and output data from remote sources • Reuse of some files by all jobs • Long sequences of similar jobs passing through data sets.
Complex Workflow	• Physics analysis. • Text mining.	• Use of VO specific higher-level services. • Dependencies between tasks and need for good error reporting and response from all layers.
Real Time Response	• Testing, validating applications. • Grid operations and monitoring.	• Short runs with small amounts of data. • Semi-guaranteed response times.
Small-scale Parallelism	• Protein analysis • Weather forecasting. • Molecular dynamics.	• Allocation of multiple CPUs simultaneously • Use of MPI libraries

Table 3: Types of Application Running on the Open Science Grid

1.4.2 Resource Usage Modes

Usage of the computational resources through the OSG is one of three modes: *Guaranteed through ownership by the user's community; Agreed upon through policies between the resource owner and the user's community; Or opportunistic use through resource sharing.*

When communities make their resources accessible through the OSG, they define the policies of their use. Resource owners must ensure that their owner user community has guaranteed use of these resources even while they are shared. Resource owners retain control of their resources including prioritization of use, which communities and users to support, and policies of access.

As members of the OSG consortium, resource owners are encouraged to provide access to available resources (typically of the order of 10% or more) to other communities for specific computational goals as well as dynamic use of currently available cycles.

Opportunistic use is a hallmark of the OSG facility. It provides a low overhead mechanism for users to increase throughput using already provisioned resources. It allows resource owners to automatically enable use of available cycles and storage by other OSG members. It supports the principle of inclusion of members who have no resources of their own but contribute value in other areas.

1.5 How OSG supports the Usage Patterns and Users

1.5.1 Operating the Facility

The *OSG facility provides a set of services and activities in support for the use of the production infrastructure and the resources accessible through it.* The list of services was given in Table 4. Some of these services are defined as “critical” to the use of the infrastructure by one or more of the user communities. For example, the US LHC relies on the publishing of information about OSG resources to the World Wide LHC Computing Grid. The availability of such services is measured, with the target availability being agreed to with the users. Critical services, e.g. the information publisher above, are being made highly available.

The facility also runs resource validation, operations monitoring, and accounting services are used to identify problems in availability and success rates of the resources, show trends and anomalies, and allow tracking of use with respect to the agreements.

Operations makes OSG software releases available through central repositories. Collections are available for resource administrators, VO managers, and user modules for remote job execution and submission sites. Operations also provides a centralized ticketing system and tracking database and resolves issues and requests from any member or user of the OSG.

Function	Type of Service
Monitoring and Validation of the Resources	Run by the Sites and checked centrally by OSG operations; other tests run at the operations center itself.
Operational Support and Problem tracking	OSG Trouble ticket system with automated dispatch to support groups, weekly, monthly, annual tracking.
Software caches	Virtual Data Toolkit and Open Science Grid releases. Two versions of the OSG release are supported at any time; Many production and development releases of the VDT are supported concurrently due to the breadth of the customer base.
OSG Virtual Organization’s Management	Virtual Organization Management Services, for Education ,Engagement, OSG and others delegated from our members.
Security control test and evaluation	Documentation and tracking of security controls and their assessments according to the security plan
Policy documents	Signed, captured and tracked policies such as Acceptable Use, Trust relationships, Registration etc.
Information Services	Local site information publishing and central OSG information collections services.
Accounting	Local site information publishing and collection, as well as central OSG collection and reporting
Gateways to/from other Infrastructures	Technology service to support model of federated grids
Information publication	Websites and databases for publishing monitoring, accounting, registration and other information. Cache of supported Certificate Authorities. Certificate Registration Authority
Collaboration tools	Mail lists, Twiki site, FAQs

Table 4: Operational Services Offered by the OSG

1.5.2 Embedded Engagement

The OSG engagement effort provides *embedded help to new users and communities* who come to use the common infrastructure or to provide access to their resources and local distributed systems.. We help adapt applications and provide user tools and services. OSG staff provide additional software and services that select the optimal resources for the users to run on at the current time, install their application software, transfer and retrieve the needed data, and provide user level monitoring to track job completion and diagnostics solve any problems encountered. The OSG is able to reuse and extend these practices and software components for each community and thus build up more complete generally usable solutions.

The overhead of the social change in migrating from the use of a dedicated local cluster to relying on the "anonymous" wide area distributed facility should not be underestimated. In many cases this is a significant change in culture, which only slowly leads to new paradigms of research and computation.

1.5.3 Supporting Collaboration and Collaboratories

Nearly all activities within the OSG combine effort from the project itself together with contributions from the user and member communities. OSG activity leads run regular open meetings for operations, security, integration, site coordination, VO support, education and communication. OSG provides many community mail lists, maintains and documents on a collaborative wiki.

The OSG incubates open collaborative research through reaching out to similar organizations and sharing experiences, software and processes. The OSG leadership engages pedagogically in workshops, community internal meetings, and one-on-one discussions with our parallel and peer organizations.

The OSG holds a series of regular (in general quarterly) meetings between the management of the project and each major user community. These meetings cover both the current issues and challenges as well as the needs and plans for the medium term future. Published action item list provide a useful basis for tracking the issues and ensuring the attention needed to address them.

1.5.4 Software Development and Testing

The OSG supports the usage pattern for software development and testing by developing procedures for *gathering requirements for, prioritizing and delivering software releases to meet the needs of the user communities.* The last two years experiences have led the OSG to include a software tools effort to oversee and help in the development or acquisition of needed operational and security tools. These tools, following the OSG principles, are supported on local sites, as well as on the OSG and other community grids and infrastructures.

The OSG collaborates directly with the DOE Center for Enabling Petascale Distributed Science DOE SciDAC project, Condor and Globus/CDIGS, dCache, Bestman, EGEE gLite, DOE lab computing divisions, and with the application software development groups of the scientific communities (including LIGO, DZero, US ATLAS, US CMS) themselves. As part of its work to contribute to new technologies and making them robust and scalable, the OSG continues to augment key external development activities.. An example is security, monitoring, modularization and performance enhancements for the US ATLAS PANDA[25] technologies which helped its current adoption by worldwide ATLAS,

As mentioned above, the OSG integration testbed and the production infrastructure itself provide for the functional, performance and scalability testing of new software and services. As an example, test usage on the OSG has identified, and together with the CDIGS team solved, several issues with Globus WS Gram over the past eighteen months that affect its usability at the scales and performance needed by the large science users on OSG. Also, CMS has done system testing of the new just in time job scheduling capabilities (based on new Condor technologies) over the production OSG facility. They have demonstrated scaling to more than forty thousand, and simultaneous submission of more than ten thousands jobs.

1.5.5 Using OSG Together with Other Grids

OSG helps integrate and support the use of multiple infrastructures as needed by its members. The OSG acquires and supplies multiplexing software and services to hide the differences in the infrastructure, as well as bridges and gateways to transform and translate information and control to the interfaces and schema of the differing services. OSG strives to understand whether and how the policies and constraints of the other infrastructures affect what the OSG can provide to the user community as a whole. The OSG activities pay strong attention to ongoing communication and follow up of issues and problems across multiple complex distributed organizations. Examples include:

- Software that translates the resource service validation information collected by OSG scripts to that required by the US LHC agreements to report to the WLCG.
- A bridge at the OSG Grid Operations Center which publishes user community selected information about the configuration and availability of OSG resources to the ATLAS and CMS collaboration applications and users.
- Gateways between the legacy DZero SamGrid[26] community grid, the OSG and the EGEE through a set of “submission forwarding nodes” and services.

Support for this usage pattern includes *multiple methods for the federation and interfacing of OSG with peer grid infrastructures*: the Grid Laboratory of Wisconsin (GLOW) provides services to route local jobs transparently to the OSG when additional resources are needed (and available); the Fermilab Campus Grid provides a gateway to the NCSA TeraGrid resources for jobs that can be executed there based on previously negotiated allocations and policies between the OSG and TeraGrid; the EGEE and the OSG engagement VO provide a bridge for the submission of the Wisdom VO application jobs to OSG resources to transparently increase the total throughput; the Clemson Campus Grid provides mapping of jobs submitted to the OSG to the internal Windows based campus-wide cluster; the Purdue University Condor gateways enable the sharing of the campus computing farms between local access, TeraGrid allocations and use through the OSG.

1.5.6 Support of Application Usage Modes

The OSG software provides remote job scheduling, resource selection, data movement and access software. Once deployed the services present standard Condor-G, Globus Gram and GridFTP, Storage Resource Management (SRM), security (for X509 VOMS extended attribute certificates), accounting (OGF usage records) and information (Glue V1.3, ClassAds) interfaces at the boundary between the distributed infrastructure and the resource. Additional software can be used by the site owners and users to define and apply authorization, access and prioritization policies for use of the resources.

Particular aspects of support for the different applications types is show in Table 5.

	Support	Challenges
Simulation and Modelling	• Batch-system services and prioritization policies. • Small amount of data storage and management needed for results.	• Ensuring full usage of dynamically available resources – wherever they are located.
Production Processing	• Job and workload management tools. • Data placement and access management tools.	• Automation of conditional workflows, retries etc in response to a wide variety of errors. • Common tools for the efficient placement and co-location of data and jobs. • Support for VO defined policies applied effectively across the autonomous, heterogeneous resources.
Complex Workflow	• Tools for managing the workflow itself. • Pre-placement of application tools and databases at remote sites. • Tools for error reporting, response and tracking.	
Real Time Response Applications	• Prioritization services to allow immediate or minimum latency execution of jobs.	• Support for checkpointing and restart of other applications. • Dynamic nature of available set of resources precludes deterministic response times.
Small-scale Parallelism	• Local support for MPI • OSG support for publishing necessary information of site specific configurations and software versions.	• Automated use across multiple MPI site configurations and implementations.

Table 5: Support for Application Types

1.5.7 Support of Resource Usage Modes

Many communities have well defined production cycles and thus computational needs. When these exceed the resources owned by the community the *OSG provides a common and low overhead framework for brokering agreements between resource owners and the user community in need*. When these needs are large, the OSG council, the representatives of the larger stakeholders in and contributors to the OSG, provides the forum for agreement and timeline of contributions. The in-place operational services of the OSG provide mechanisms for implementation and tracking of such agreements.

The OSG provides resource information and matchmaking software[] for automated selection of remote sites on which to execute jobs. Users embed interfaces to this information and/or do manual selection of sites. Such selections are configured to match the processing and storage needs and timelines of the applications

1.6 Who Uses the OSG

1.6.1 Science Communities

The current average daily use of OSG is more than 20,000 CPU days per day. The average increased about 25% during 2008. The physics communities account for about 85% of the usage. The use of OSG by a typical high energy physics community , CMS, is well described in another section of this book. During 2008, the typical level of opportunistic usage on OSG has been more than 25%. The majority of the non-physics usage is opportunistic. We summarize the science communities using the OSG in Table 6 below.

Type	User
Simulation	• *High Energy Physics:* ATLAS, CMS, DZero, CDF generate simulated

and Modelling	events. While the individual use fluctuates, the average for each community is about 3,000 CPU days per day. In particular, the majority of the use by DZero in opportunistic. Recently, through such use of storage at ATLAS and CMS sites, the DZero CPU efficiency has increased from less than fifty to more than eighty percent[29]. • *Chemical Engineering:* Simulations at the University of Buffalo to calculate the virial coefficients of water and other compounds[30]. • *Weather Research Forecasting:* The WRF modeling application modelling volumes of space at a fine resolution of around four kilometers[31]. • *Text Mining (engagement user):* The School of Information and Library Science at the University of North Carolina ran the analysis of text using the Claim Jumping techniques[32]. • *Protein structure determination (engagement user):* Proof of a new molecular modeling software package RAPTOR to predict protein structure[33]. • *Molecular Dynamics:* Protein simulations to determine how much water exists inside proteins and whether these water molecules can influence the proteins[34]. • *Coastal Modeling:* Monte Carlo simulated storm tracks (fifty thousand) used to seed very large ADCIRC MPI runs. This has enabled much greater exploration of sensitivities to storm track selections for flood plain mapping simulations[35]. • *Genetics:* Running convergent Haplotype Association Tagging to map human mutations with disease model parameters. Through these runs newly identified loci that contain mutations have been discovered including 2 for schizophrenia, 2 for breast cancer, and 3 for Parkinsons disease[36]. • *Mathematics:* Runs for graph isomorphism and classification of incidence structures. The applications computationally detect new objects/examples and determine their reason for being thereby gaining insight. [37].
Production Processing	• *High Energy Physics:* ATLAS, CMS reconstruction of event date. OSG contributing >30% of worldwide collaboration throughput (see other section in this book). • *Gravitational Wave Physics:* LIGO Einstein@HOME searches for gravitational waves from continuous sources[38]. • *Genetics Predictions:* Running superlink[39], a user was able to ramp up quickly to 36,000 cpu hours/day across more than 20 sites on the OSG.
Complex Workflow	• *High Energy Physics:* ATLAS and CMS simulations require many steps managed by workflow systems developed by the communities themselves. • *Gravitational Wave Physics:* LIGO Inspiral Analysis science analysis under test.
Real Time Response	• Testing of scalability, robustness and performance of new workload management systems. • Educational demonstrations.
Small-scale Parallelism	• Weather Research Forecasting (WRF) at the NERSC facility; • Tests for the CHARMM application at NERSC and Purdue.

Table 6: Science Communities Using the OSG

1.6.2 Educators and Students

OSG education and engagement activities work to help and teach new scientists – and, in the case of education, specifically young new scientists – to adapt and run data and compute intensive applications on the existing distributed infrastructure.

OSG grid schools, lasting one to several days in duration, have combined lectures and hands-on laboratories to teach the fundamentals of grid technologies. To date, they have enabled more than three hundred students to actually run jobs across and transport data between specific OSG sites that advertise support for the "education community". Selected students also attend the longer, residential school, the International Summer School on Grid Computing, which since 2007 is co-sponsored by the OSG. Students receive a grounding in computer science fundamentals of distributed computing and then work in teams as proto-typical communities in a competition to develop working integrated scientific applications, running across a set of (locally) distributed computers. In additional, several faculty now rely on OSG materials for their grid computing courses.

1.7 Challenges for the OSG

OSG is a large and complicated organization - that's simply what it takes! We have set ourselves the challenge to not only provide a world class nationally distributed facility but also to valuably transform the scientists approach to computational facilities as an integral part of their research toolkits.

The challenges the Open Science Grid faces include:

- *Maintaining the highest standards of data center service and operations* during times of evolution and expansion. The existing, and even more the new, user communities expect a system which is robust against failure, defended against misuse, with availability approaching that of their local data centers. Operations and operational tools, alarms, tracking and response mechanisms, must all remain top priorities.
- Meeting the planned (and anticipating the un-planned) capacity and capability needs of the current user communities. The LHC runs and the LIGO and STAR upgrades will result in *a three- to ten-fold increase in the data and job throughput* required on the community distributed infrastructures in the US over the next three years.
- *Security and trust in the service of open science and protection* of resources and services. The size and complexity of the systems and the size and sophistication of the potential attack community continue to grow. The challenges in scale of risk and effort to maintain timely and effective response to vulnerabilities and incidents are unknown.
- *Managing and accommodating heterogeneity.* The community and OSG infrastructures include facilities that scale from small university department clusters to large leadership class high performance computing facilities. The user communities scale from individual PIs and students to very large collaborations. We anticipate increased need for and support of ad-hoc groups as the analysis of the LHC ramps up and the number of different user communities grows.

- Developing and measuring *an agreed upon sustainable economic model for growth, which takes account of the bartering and brokering approach that is the OSG hallmark.*

We recognize inadequacie*s and challenges in many areas. Below we cover only* a few specifics of sustaining and scaling the facility, operational security, software evolution, resource sharing, metrics, and sustaining collaboration.

1.7.1 Sustaining and Scaling the Facility

The OSG facility expands continuously due to the integration of new sites, the installation of new resources, the joining of new member communities together with the new partnerships and collaborations. Some of the specific technical challenges we are facing are:

- The immaturity of the services and components to defend themselves against overload and misuse by application software and users.
- Reliability of software and configuration testing in such a heterogeneous environment before putting new software into production.
- Lack in capabilities in software tools to monitor and report on the services, resources and infrastructure.
- The need for attention to high availability, fault tolerance and removal of single points of failure.
- The incompleteness of end-to-end reporting, interpretation and response to errors and failures with many different sources and interactions.

The OSG continues to strive to address these issues at all levels and with all parties.

1.7.2 Operational Security

Security is integrated into every activity in the OSG. The small amount of dedicated effort is augmented up to "all hands to the wheel" as needed to respond to reported concerns, vulnerabilities and incidents. The challenges include:

- Evolving and scaling a secure and simple security model and set of technologies.
- Dependencies on external organizations for key parts of the security infrastructure, both certificate generation and software.
- Defining and communicating policies and processes with appropriate scope, responsibilities and authorities.
- Monitoring the infrastructure, resources and users for unexpected behavior.
- Understanding and responding to each threat and incident in detail. Coordinating incident response across widely diverse and spread user communities, resource owners, and peer grids.
- Ensuring security is integrated into all software and specific security components are built, deployed and tested.

1.7.3 Software Evolution

The software needed by OSG stakeholders are supplied either by external software providers or the community naturally evolves in functionality and configuration. This can be evolutionary – small extensions and changes, patches to fix security and other bugs, - or revolutionary with the provision of whole new capabilities, methods and technologies. The OSG has to provide a stable, managed process for bringing these new versions of software into the production environment, while maintaining an operating system.

There is also the question of stability of the software supply-chain, given OSG's dependence on external organizations and providers for contributions of the software itself.
The following are some of the challenges:

- Efficient and fast patches of the OSG software and installation of updates in response to security notifications.
- Balancing the amount and effort spent on testing with the need for timely delivery of the software to the user communities. The stability of the resulting infrastructure becomes at risk since testing invariably does not cover the full set of usage patterns.
- Prioritizing additional functionality requested by the user communities with the need to make the software stack have minimal footprint, low impact, and be simple to install, configure and use.
- Integrating diverse software components from multiple software suppliers with different levels of development maturity, different release cycles, accommodating the political realities of the communities involved.

1.7.4 Resource Sharing

As the scale of the use of OSG continues to ramp up, new boundaries are discovered in the performance of the underlying services. Additionally, the deployment of shared storage with support for the needed I/O rates across such a large number of sites is immature. Some key challenges are currently:

- Common storage and data management that span the full range from a TeraByte to tens of Petabytes in size.
- Dynamic allocation and sharing technologies and methods for data storage and access, covering guaranteed, agreed upon and opportunistic usage modes.
- Adequate end-to-end support for management of available and/or over-subscribed resources - including processing, storage and networks.
- Support for sub- and ad-hoc VOs and groups within a VO.

1.7.5 Metrics.

Success is not just the number of jobs executed and the amount of data transferred and stored. Success is defined to the extent OSG is meeting its purpose and goals. Success is measured by the impact on scientific productivity and maturity of computation as a cornerstone, together with experimentation and simulation of the research portfolio. Specific challenges are definition and measurement of the:

- Openness of the OSG. This includes the *diversity of members and user, the inclusiveness of our principles and approaches,* and the effectiveness of our training and outreach.
- Scientific impact of the OSG. The OSG is a *tool for scientists to do science across many different domains*. Measuring the impact of a tool solely by the number of papers and citations of papers that use the tool, is both simplistic and difficult. We thus augment it with stories of scientific output and innovation, both planned and unanticipated,

- Quality of the capabilities provided by the OSG cyber-infrastructures. This includes the *usability, efficiencies, and support processes.*

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.

1.7.6 Sustaining Collaboration

To succeed in our vision and purpose we must sustain a broad and diverse collaboration, especially between the computer science communities providing the technological innovation and direction, and the domain scientific communities providing the needs and usage.

This requires carefully managing the overheads that occur to reach consensus and agreement within multi-organizational collaborations. It requires commitment to deliverables and milestones by contributors in matrixed activities with no direct line-management oversight. It requires science communities to believe that there is sufficient benefit and value to be gained by giving up on “going it alone”.

Additionally, the funding cycles are constrained. They result in projects of only a few years duration. This poses challenges in sustaining the trust and collaboration necessary to successfully peer and serve the typical large scientific collaboration with longer than decade life-times.

And for OSG there is our goal to federate with the NSF national leadership facility, the TeraGrid and to maintain interoperation with our European peers as they change their centralized model to one of more than ten cooperating National Grid Infrastructures.

1.8 References

[1] Pordes, R., et al. 2008. New science on the Open Science Grid . Journal of Physics: Conference Series 125:012070.
[2] Bird, I., et al. 2005. Deploying the LHC computing grid - the LCG service challenges. Local to Global Data Interoperability - Challenges and Technologies. June 20-24, 2005.
[3] "LIGO and the Detection of Gravitational Waves" Physics Today, October 1999
[4] Top quark physics at the Tevatron. CDF D0 Collaborations. Presented at 27th International Conference on Physics in Collision, Annecy, France, 26-29 Jun 2007. Published in Acta Phys.Polon.Supp.1:237-244,2008.
[5] An overview of results from the solenoidal tracker at RHIC experiment. By STAR Collaboration 2008. 7pp. Published in J.Phys.G35:044001,2008.
[6] The Virtual Data Toolkit web site http://vdt.cs.wisc.edu/
[7] Embedded Immersive Engagement For Cyber-Infrastructure (EIE4CI) http://nsf.gov/awardsearch/showAward.do?AwardNumber=0753335
[8] Data Intensive Science University Network (DISUN) web site www.disun.org.
[9] LIGO Data Grid (LDG) https://www.lsc-group.phys.uwm.edu/lscdatagrid/overview.htm
[10] List of OSG Virtual Organizations http://www.opensciencegrid.org/VO_List
[11] OSG operations services http://www.grid.iu.edu/systems/
[12] OSG integration activity https://twiki.grid.iu.edu/bin/view/Integration/WebHome
[13] Frey, J., T. Tannenbaum, I. Foster, M. Livny, S. Tuecke. 2002. Condor-G: A computation management agent for multi-institutional grids, *Cluster Computing* 5:237-246.
[14] Foster et al, The Globus Toolkit www.globus.org
[15] GIPs https://twiki.grid.iu.edu/bin/view/ReleaseDocumentation/GenericInformationProviders

[16] Gratia, a resource accounting system for OSG, http://indico.cern.ch/contributionDisplay.py?contribId=118&sessionId=26&confId=3580
[17] OSG resource and service validation (RSV) http://rsv.grid.iu.edu/documentation
[18] The OSG Engagement Matchmaker. http://osgmm.svn.sourceforge.net/viewvc/osgmm/
[19] The NMI Build & Test Laboratory: Andrew Pavlo, et al. University of Wisconsin-Madison, LISA 2006
[20] Bestman http://datagrid.lbl.gov/bestman/
[21] Rehn, J., P. Fuhrmann, et al. 2006. dCache, the Upgrade, *Proceedings of International Conference on Computing in High Energy and Nuclear Physics (CHEP 2006)*.
[22] A. Hanushevsky et al. Real-time data access monitoring in distributed, multi-petabyte systems (SLAC-PUB-13108)
[23] Altunay, M., D. Olson. 2008. Open Science Grid security activities, http://osg-docdb.opensciencegrid.org/cgi-bin/ShowDocument?docid=749 (April, 2008).
[24] OSG Blueprint https://osg-docdb.opensciencegrid.org:440/cgi-bin/RetrieveFile?docid=18&version=5&filename=OSG-Blueprintv0.10.pdf
[25] The PanDA Production and Distributed Analysis System, https://twiki.cern.ch/twiki/bin/view/Atlas/PanDA, April 2008.
[26] D0 SAMGrid http://projects.fnal.gov/samgrid/documents/design.html
[27] Loomis, C. Characteristics of Grid Applications, EGEE '06, http://indico.cern.ch/conferenceTimeTable.py?confId=1504
[28] OSG site matchmaker, http://osgmm.svn.sourceforge.net/viewvc/osgmm/
[29] OSG research highlight http://www.opensciencegrid.org/DZero_Opportunistic_Storage
[30] Further studies in the series described at http://pubs.acs.org/cgi-bin/abstract.cgi/jpcbfk/2007/111/i39/abs/jp0710685.html
[31] Etherton and Brieger, 2008: Probabilistic QPF from a WRF Ensemble (Weather and Forecasting - in Review)
[32] Blake, C. Text mining on the OSG, http://www.isgtw.org/?pid=1001114
[33] Jinbo X. http://ttic.uchicago.edu/~jinbo/
[34] Damjanovi, A., et al. Open Science Grid study of the coupling between conformation and water content in the interior of a protein. Journal of Physical Chemistry, B, August 2008
[35] B.Blanton - RENCI http://www.sura.org/programs/docs/UNCADCIRC.pdf
[36] K.Wilhelmsen - UNC-CH
[37] A.Betten - Colorado State University
[38] LIGO Einstein@HOME http://einstein.phys.uwm.edu/
[39] Silberstein, M. http://cbl-link02.cs.technion.ac.il/superlinkattechnion/download_all.php