Open Science Grid

OSG Blueprint

March 3, 2011

OSG-#18

Table of Contents

1. Introduction

1.1. What Is the OSG?

The Open Science Grid (OSG) is a partnership for shared distributed infrastructure within the US. Since its inception, it has built and operated a production infrastructure and become an internationally-recognized key element of the US national cyberinfrastructure for distributed high-throughput computing (DHTC). We are driven by a commitment to providing leading DHTC services to scientists, researchers, educators, and students and to advance the state of the art of DHTC.

There are several facets to the OSG – indeed, the acronym "OSG" has assumed many meanings in the last five years. It is helpful to outline these different aspects up-front:

- **Consortium**: The partnership between scientific communities, institutions, and organizations for promoting shared cyberinfrastructure (CI) in the US through distributed high throughput computing. The consortium consists of the communities, executed by a core project, and is led by a council[1].

- **Core Project**: The project currently funded by the NSF and the DOE to contribute services to the vision of the OSG Consortium. There are many projects involved with the OSG Consortium; these are OSG **Satellites**, as discussed in Section 2.

- **Council**: The council that governs the OSG Consortium, including the core project. The OSG Council includes the PI, executive team of the core project, user communities, software development projects, and external projects.

- **Fabric of Services**: The services provided by the OSG Core Project. These are operational and system engineering services, software infrastructure services, and consulting services. These services are separable – communities are expected to select a subset of the services according to their needs.

- **Production Grid**: A shared CI formed using the OSG Fabric of Services. The Production Grid serves as a common fabric for several large scientific organizations to execute their science and to share their resources; it is currently the largest grid using the OSG Fabric of Services.

- **Campus Infrastructure**: Another instance of shared infrastructure using OSG services. The Campus Infrastructure is being formed by linking several campus grids together.

1.2. Goals of the Blueprint Document

A key to our success is a common set of evolving principles that guide our decisions. These principals shield us from the ever changing technological and 'buzzword' landscape. The "OSG Blueprint" provides guidance on how the different elements of the OSG function and our approach to building it. The blueprint is a living document, and we emphasize four purposes:

- **State the principles**: One theme of the document is that the OSG is a principles-driven organization. We attempt to highlight the principles of the Consortium, and those used in the construction of the Production Grid.

[1] The management plan for the OSG can be found here: http://osg-docdb.opensciencegrid.org/cgi-bin/RetrieveFile?docid=314&extension=pdf

- **Explain the key terminology, concepts, and actors**: As a large, distributed partnership, the OSG Consortium has a complex social structure. We outline the key communities in this partnership, how they interact, and the conceptual framework and terminology used.
- **Capture the current set of services**: The OSG Fabric of Services is large and varied. This document attempts to present an organized outline of these services.
- **Document the DHTC architecture of the Production Grid and Campus Infrastructure:** The Production Grid utilizes the OSG's services as a part of the day-to-day infrastructure of scientific organizations. It provides a common core and is used in a variety of ways by different communities. The Campus Infrastructure focuses on bringing DHTC to organizations, then bridging them to other campuses. We document the core functionalities, and provide examples of how they are used.

The Blueprint is a "living document." It will be updated throughout the lifetime of the OSG Consortium. There are two to three Blueprint meetings every year to discuss architectural issues; as a part of these meetings, we spend two hours reading through and updating the document based upon OSG evolution from the previous months or pre-existing omissions/errors.

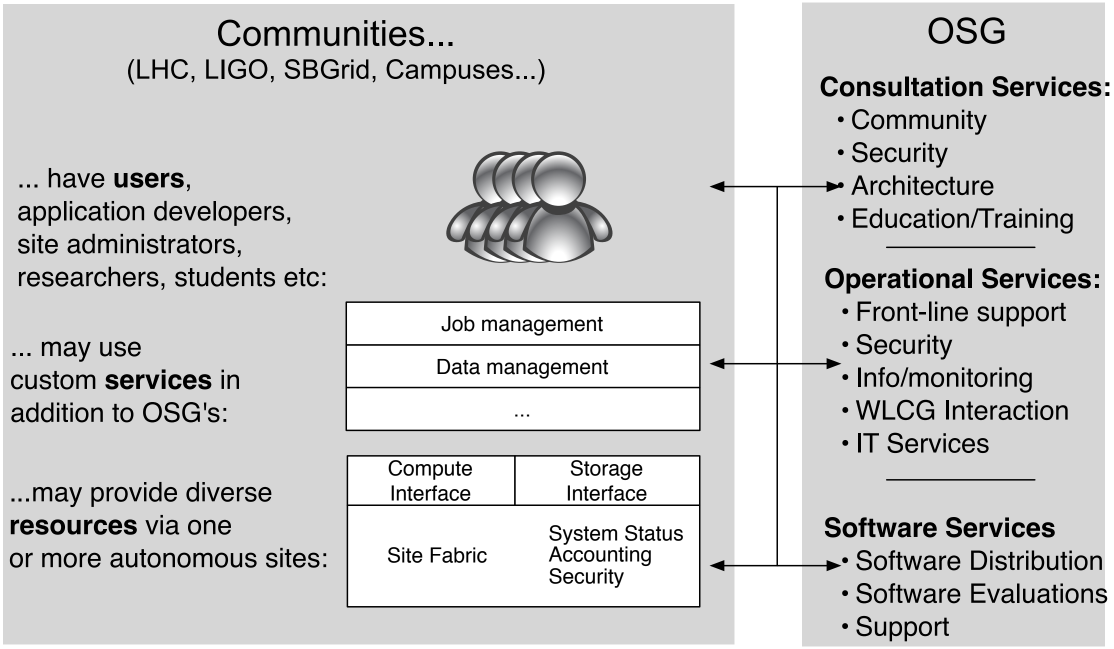
1.3. Overview

The OSG revolves around DHTC, a form of computing defined to be the shared utilization of autonomous resources where all elements are optimized to maximize computational throughput. While this type of computing can be approached from several directions, we believe our principles of *autonomy*, *diversity*, *dependability*, and *mutual trust* allow the communities interact to achieve the best possible throughput.

We do not support only a single implementation of DHTC services. We distinguish between *principles*, *best practices*, and *requirements*. We have best practices to provide known "off the shelf" solutions for a wide range of computational problems, but try to have a minimal number of requirements to allow and support other approaches within the OSG and facilitate technological evolution. By allowing or encouraging diversity of implementations, we are able to positively impact and benefit from the creativity of the widest range of communities. These principles are discussed in Section 2.

Each community we interact with has a unique set of needs. Not every community will want to use all of the fabric of services. Some may only consume software infrastructure services and others may only need consulting services. When using the Production Grid there are only a few of the many services available that are "required", others are used according to the specific needs of the community.

OSG's Community-Focused Architecture



There is a sharing of software, operational services, and knowledge between the communities and OSG in each of these areas.

The OSG provides a fabric of services. The DHTC services provided are broadly grouped as *consulting services*, *software services*, and *operational and system engineering services*. The communities OSG partners with may interact with any combination of the groups (or subsets of a group). In fact, we envision no community will ever use 100% of each. The OSG Fabric of Services is discussed in Section 4.

One characteristic worth emphasizing in our operational and software services is "production quality". Services are designed to be scalable and robust when offered to external communities; they are meant to be used for producing science and research output when released. Our production grid operates at the edge of DHTC scalability, providing "production quality" services is not a trivial exercise. The user communities find value in having the OSG to bridge the gap between research technologies and production software and operational service.

The OSG is partly defined by the communities that utilize and contribute to it. We form partnerships with our user communities in order to help them achieve their goals; we feel the partnership model helps us achieve our goal of autonomy. Communities are complex societies involving scientists, community-oriented computing specialists, and resources. The OSG aims to foster and facilitate, not "stand in the way" of the community. The community organization of the OSG is explored further in Section 3.

The Production Grid is a reference implementation of the DHTC principles and OSG fabric of services. It is an extensive community, consisting of about 10-20 VOs and a hundred different resources. As of February 2011, it currently averages about half a million jobs, a million CPU hours, and half a petabyte of data movement every day. The specific principles used for implementation, a description of its architecture, and requirements for participation are covered in Section 5.

Another shared CI implementation using the OSG Fabric of Services is the Campus Grid. The OSG provides a solution for a single campus to build a DHTC infrastructure; if desired, the campus grids can be linked together or interfaced with the Production Grid. This approach is covered in Section 6.

2. DHTC Principles

The OSG is a national cyberinfrastructure for the advancement of DHTC computing. When our communities are able to successfully take advantage of DHTC concepts to produce science, the OSG succeeds. By following our principles, we believe we are most successful. The core principles of the OSG are *autonomy*, *mutual trust*, *dependability*, and *diversity*. These principles build upon the experience of the precursors to the OSG (GLOW, PPDG, iVDGL, Trillium).

Distributed high throughput computing on the OSG is typically achieved by breaking a task (involving "computing" or "data movement" or both) into smaller jobs that can mostly be accomplished independently. While there may be some amount of high-level ordering in the set of jobs, throughput is achieved by executing many of the jobs in parallel. In order to achieve higher levels of throughput, one must maximize the average number of active resources over a long period of time. Autonomy, mutual trust, and diversity allow a community to maximize the possible number of resources it can access. Dependability increases the average number of these possible resources one can access over a length of time.

Autonomy: The OSG functions as a unifying fabric for autonomous users and resource provides. It provides the community and technology for disparate users and resources to work together. It enables multi-domain resource sharing and resource pooling. Without the abstractions maintained by the OSG ensuring autonomy, the wider community would fragment into internally-focused relationships between sites and their owners, decreasing the overall "throughput" portion of HTC and have to pay the "overhead" of maintaining their own infrastructure, rather than sharing the cost. The OSG does not own resources nor sites, nor does it dictate site policies. It is rather the glue that underpins complex relationships of scientists, researchers, educators, students, and system administrators.

Mutual Trust. The distributed nature of DHTC requires users and resources to establish a level of mutual trust. The OSG provides a valuable venue to establish this trust across institutional boundaries. The OSG enables trust relationships at many levels extending beyond the PKI technology used for authorization. Relationships include software provided, site, and users; all are necessary for DHTC.

Dependability. Dependability and graceful degradation are key principles for DHTC, as it faces special challenges due to the multiple administrative domains involved. In order to provide a plausible common set of software, system administrators must feel the software is robust and run without intervention.

Even with individually dependable pieces, the large-scale, distributed focus implies some portion will always be broken. Hence, the system as a whole must be tolerant of faults and continue to function. Total outages are unacceptable if throughput is to be maximized over long periods of time.

The user must be able to depend on the system to perform at a high level of throughput regardless of when the jobs are submitted. The user workflows must also be able to recover and restart from failures at multiple levels, and have a plan for failure of each component they interact with.

Dependability incorporates a holistic view of the system; it is not limited to only hardware, but how it is operated. A dependable system must have documented stable interfaces, well-announced planned downtimes, and a communication mechanism between involved parties.

Diversity. DHTC aims to maximize the amount of work accomplished, so a DHTC community needs flexibility to accept many types of compute and storage resources. We cannot be selective of resources or users we include. Flexibility in resource requirements has a high price; resource diversity sacrifices simplicity for end-users. Including new platforms may require significant investment by the fabric.

Resource diversity versus user friendliness is an ongoing balancing act. We attempt to manage this by providing a minimal set of uniform interfaces, and advertising differences to end-users. The same principle applies to DHTC user job portability; each runtime requirement the user adds decreases the possible resources used. To be successful, users must prioritize the resources based upon the cost (reliability of individual resource or likelihood of preemption) and benefit (contribution to the total number of compute hours).

2.1. Principles, Best Practices and Requirements

The OSG principles guide and influence the fundamental aspects of the methods, architecture, and implementation. On top of these, we have additional **requirements** for participation in the Production Grid and provide consulting for **best practices** for utilizing our DHTC operational and software services. As each Campus Grid is built independently of the OSG, there are no requirements centrally imposed unless it is linked into the Production Grid.

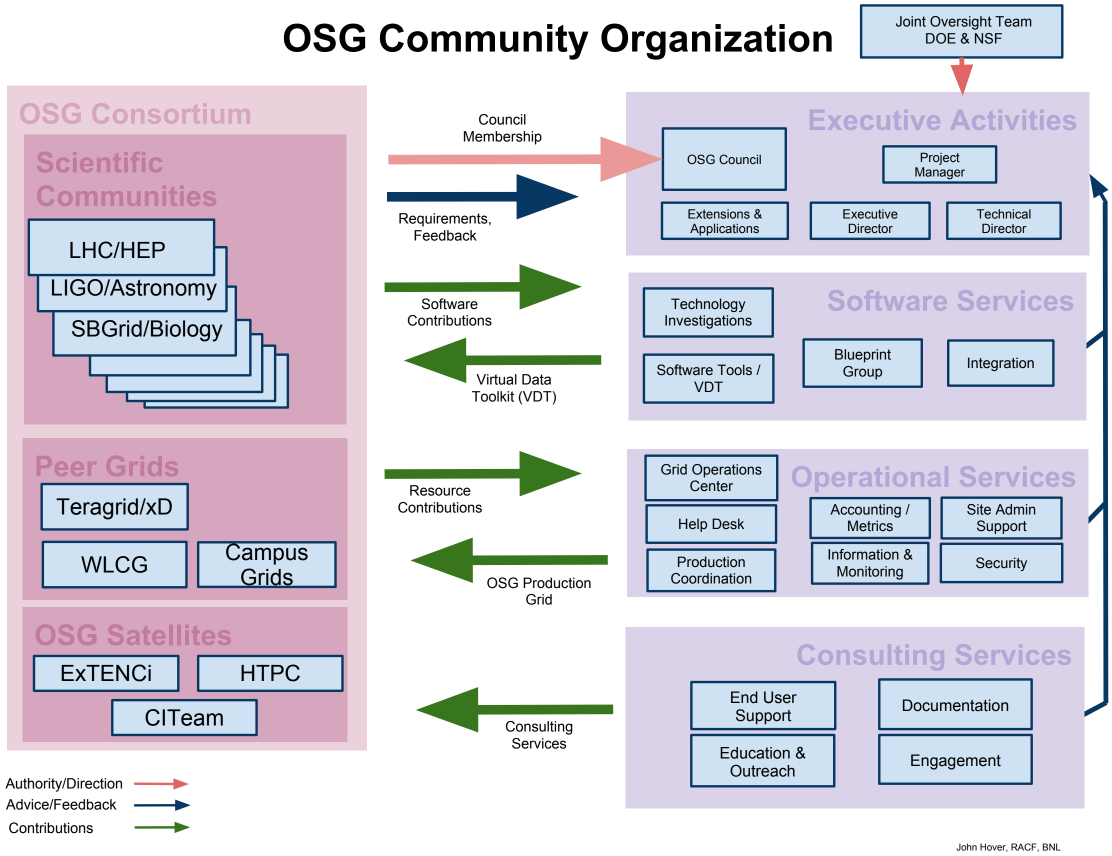
Best Practices are guidelines to be adhered to, as much as is possible, in practice. They are guided by the availability and use of existing components and technologies. The OSG provides a reference implementation of all its software and operational services for the Production Grid; this reference implementation utilizes available best practices. The OSG user documentation and education/training attempt to guide new communities along the best practices.

Requirements are formal statements that provide goals and constraints on the designs and implementations; effectively, they are limited to participation in the production grid. The goal is to have a minimal set of requirements for participation; the OSG attempts to carefully balance the set of requirements needed for the *mutual trust* principle and the freedom implied by *diversity* and *autonomy*.

The current set of best practices and requirements are covered in “Implementation of the Production Grid”, Section 5.

2.2. OSG and Community Organization

The OSG management plan[2] gives more information about the organization and management of the OSG. This section covers the key players of the OSG, both internal and external.



As the OSG forms complex partnerships with its communities, it is important to understand the basic building blocks of these relationships. We find this is an effective means to enable communities to accomplish DHTC-based science. The user communities we work with can be broadly grouped into science-based communities or regional organizations. The science-based communities are formed by a single large experiment (common in high-energy physics) or a set of researchers in a common field. The regional communities tend to focus around a single campus or US state, and may encompass a diverse set science. The non-user communities include software providers, the WLCG, and Satellite projects.

The user communities the OSG interacts with are organized as "virtual organizations," or VOs, which have a common goal. A VO is a dynamic collection of users, resources, and services. For science communities, the commonality is a science or research goal; regional VOs have a common goal of sharing resources or a common organization or institution. The OSG tries to interact at the VO level, not the level of individual users or

[2] http://osg-docdb.opensciencegrid.org/cgi-bin/RetrieveFile?docid=314&extension=pdf

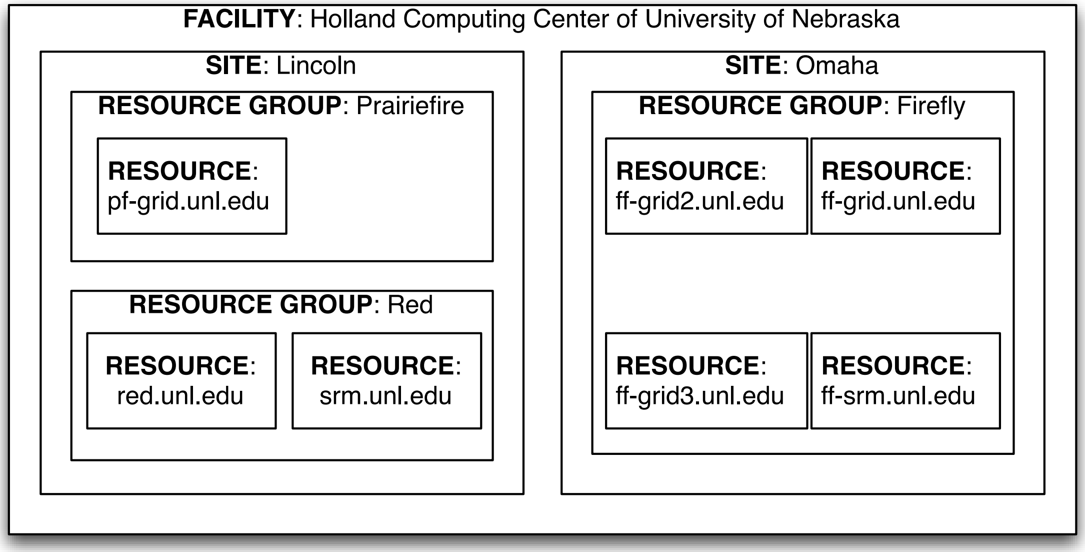
resources. By interacting primarily with the VOs, the OSG is able to better scale its limited resources and more effectively train and educate.

All computational resources are owned by one or more VO. The OSG provides a common cyberinfrastructure platform for allowing VOs to share their resources internally, and, if desired, externally through the OSG Production Grid. VO's have a range of internal services (some very thorough, others are simple), supported to enable their end-users to complete their science or research. The VO's services are referred to as community cyberinfrastructure. OSG-provided software infrastructure and services are often a key middleware component of the shared CI; depending on the community, the OSG may provide significant or relatively small limited capabilities.

The OSG is organized into several functional areas, referred to throughout the document; see the management plan for a description of each area. The services offered by each area are further explained in Section 4.

A resource in the OSG typically provides a Linux clusters and/or large-scale storage systems. The OSG has a hierarchy of resources, shown in the image below. A *resource* refers to an endpoint on the Internet that provides one or more *services* (OSG CE and SRM are the most common services). A logical grouping of resources is called a *resource group*; a resource group is often a cluster and its attached storage. All resource groups under a coherent set of administrative policies form a *site*. Finally, all sites in a given physical organization form a *facility*.

The relationships of this nomenclature and a non-trivial example of the Holland Computing Center facility is illustrated below. The Holland Computing Center is the University of Nebraska's high-performance computing center. There are two OSG sites run by different sets of administrators (Lincoln and Omaha). The Lincoln site has two clusters, Prairiefire and Red, which are registered as separate resource groups on the OSG Production Grid. Red has two endpoints registered as resources in the OSG: red.unl.edu (running the OSG CE service) and srm.unl.edu (running the SRM service).


FACILITY: Holland Computing Center of University of Nebraska
SITE: Lincoln
RESOURCE GROUP: Prairiefire
RESOURCE: pf-grid.unl.edu
RESOURCE GROUP: Red
RESOURCE: red.unl.edu
RESOURCE: srm.unl.edu
SITE: Omaha
RESOURCE GROUP: Firefly
RESOURCE: ff-grid2.unl.edu
RESOURCE: ff-grid.unl.edu
RESOURCE: ff-grid3.unl.edu
RESOURCE: ff-srm.unl.edu

Another element of the ecosystem is the OSG Satellite projects. Projects are classified as satellites if they have a significant inter-dependency with the OSG and expect to collaborate closely with OSG[3]. Often satellites deliver new technologies or VOs into the OSG and have managed “touch points” with the OSG to make sure their work adds constructively to the whole.

The OSG Fabric of Services depends on many pieces of external software (as a rule, the core project develops no new software unless absolutely necessary). As certain software is integral to its functionality, we have a close relationship with external software projects. For a few projects, the OSG assigns liaisons to assure requirements and issues are clearly communicated and tracked.

The ALICE, ATLAS, and CMS core stakeholders are both members of the Worldwide LHC Computing Grid (WLCG). They delegate the fulfillment of several WLCG requirements (interoperability, accounting, monitoring, participation in information services) to the OSG. On behalf of these VOs, the OSG has a significant interoperability effort, primarily working the European Grid Initiative (EGI). The WLCG requirements require significant coordination with external projects, and affect every OSG area.

3. Fabric of Services

The OSG breaks its services into three broad groupings: *consulting services*, *operational and system engineering services*, and *software services*. This section provides a high-level overview of the services provided. These services can be combined in various ways to form cyberinfrastructure; implementations of CI include the Production Grid (Section 5), Campus Grids (Section 6), the OSG Integration Testbed, and the OSG Overlay service.

3.1. Software Services

The goal of the OSG software services is to provide and support a dependable, trusted, easy to deploy and low cost of ownership software infrastructure for DHTC. The OSG’s *software infrastructure* encompasses the process of evaluating, building, testing, configuring, documenting and packaging the OSG’s set of software into a coherent suite. We use this term (as opposed to just “software”) to emphasize two aspects of our work: (1) We integrate existing tools from external software providers into a coherent whole and (2) we do as little new software development as possible.

Our primary service in this area is **software distribution**; the distribution is called the Virtual Data Toolkit (VDT) for historical reasons. The scope of the OSG tools includes everything between “site fabric” and “user applications”. The primary users of our software infrastructure fall into two groups: the resources administrators and the application developers that support their communities’ scientific software.

A vital service for our software distribution is **configuration management**. The VDT software distribution includes complex pieces of software that must interact with the local site services. The VDT team works with stakeholders to identify their needs, understands the software components and interactions, and communicates with the external software

[3] https://twiki.grid.iu.edu/bin/view/Management/SatelliteProjects

providers as needed. This allows the team to formulate the desired configuration; when possible, the VDT strives to provide "out-of-the-box" configured software (the best software configuration service is the one which needs no input). As this is not always possible, the VDT provides configuration utilities for each component; these are all integrated into one master configuration file, the "config.ini".

Testing and integration are another essential OSG software service. The OSG frequently runs a minimal-scope, "smoke-test" for each software build. This frequent service gives both packagers and external software providers overnight feedback of whether the new software is minimally functional. The OSG also runs VDT release candidates on a distributed testbed. This provides site testing of the candidate on as complex site environments as possible.

Finally, OSG provides **software documentation**. The DHTC software used by the OSG is sometimes maturing or research-quality software: documentation is often deficient. The OSG writes new documentation when needed, and additional documentation covering the software stack as a whole.

3.2. Operational and System Engineering Services

For VOs participating in the OSG Production Grid, we provide a variety of DHTC operational and system engineering (OSE) services run by the OSG Operations group. The group is centered at the Grid Operations Center (GOC) in Indiana, but like the OSG, is not limited to one physical facility. Other members of the group are located at UCSD and Fermilab.

One category of OSE services available on the production grid is **front-line support**. These services manage the direct communication methods for OSG users (administrators, developers, peer-CIs, end-users), typified by ticketing systems. As the OSG is a nexus for many communities and cyberinfrastructures, our frontline support services may source help requests, route requests to or from other support centers (such as regional VOs, GGUS, the central WLCG ticketing infrastructure), and solve issues.

The security team provides **a 24/7 incident response service**. The OSG coordinates the response actions across the production grid sites, VOs, and peer grids as necessary. For parties affected by an incident, we provide guidance and support via basic forensics analysis, recovery and re-installment processes. In order to measure readiness and recovery abilities, we organize incident drills and measure service and site recovery and response times.

I**nformation and monitoring services** are a necessary part of any production environment. The OSG resource and contact registry that serves as the authoritative information source about participants in the OSG. Contacts are needed, for example, for security incidents and other issues. These services also aggregate information about the current status and usage of the production grid from the sites and VOs. This information may be used immediately or recorded for historical usage accounting. This data is aggregated and re-distributed based on the appropriate policies, and is presented in several machine-readable formats as well as visual formats.

The USLHC communities rely on the OSG to provide a few special **WLCG services** to tie in the OSG's information and monitoring with the rest of WLCG. The OSG performs the aggregation, filtering, transformation, and upload of information and usage accounting from the OSG to the corresponding WLCG services for USLHC sites.

The OSG provides a software distribution that a VO may utilize to construct its DHTC infrastructure. This is a difficult task for many communities; in response, OSG has deployed some upper-layer **shared VO services**. These are services that may be deployed by a single VO's workflow, but can be delegated to and run effectively by OSG Operations for many VOs. The currently deployed VO services provide pilot job submission capabilities and VO user databases; in the future, we may also provide submit hosts for VOs.

Finally, the operations team runs the **routine IT services** needed for a modern, complex organization (web site, wiki, document repository). Communities too small to run their own, such as smaller OSG Satellites, utilize these services for their needs.

3.3. Consulting Services

The OSG provides intellectual or consulting services to several communities. These services differ from software and OSE services as a successful consultation may lack a tangible deliverable (such as software or a running daemon on a host). This does not imply a lower value of these services. Rather, these intellectual services are often about communicating OSG principles, which tends toward being more enduring than the latest technology innovation. Consulting is available for all user communities – science/research, campus, education and training – as well as software developers and peer organizations.

Often, the DHTC principles and breadth of the OSG services can be overwhelming to communities; it becomes difficult to gain a 'big picture' of the current OSG status. The OSG offers several **community services** aimed at assisting communities in better utilizing the OSG. This includes bringing new VO applications or new sites onto the OSG (the OSG User Support area) and making sure VO's applications remain efficient on the OSG production grid on a week-to-week basis (the OSG Production area). The production area also assists in the coordination between multiple campus grids and bringing DHTC to campuses. Other community services include providing specialty services needed by multiple stakeholders. An example is the WLCG requirements for the USLHC VOs: the OSG will translate new requirements from the WLCG into capabilities provided by the OSG. By adhering to our principles, we firmly believe this is not only more cost-effective than having each VO implement the requirements separately, but results in superior solutions.

In addition to the above services, a community may also ask the OSG to take a more active part in **architecture design** of their DHTC services. The OSG best practices mentioned in Section 2.1 are a key part of architectural advice we offer; assistance in the design of campus grids is another key part. Based on our expertise in DHTC, the OSG provides investigation and fair evaluation of software for VOs. The breadth of our expertise is especially useful for smaller VOs. The OSG's partnerships with the communities allow information and ideas to flow to and from the OSG; this information flow will ideally improve both OSG and the VO.

Even with the OSE and software security services provided by the OSG, security across a heterogeneous national or worldwide DHTC infrastructure is a difficult task for communities. The intellectual/consulting **security services** provided by the security area help the community to establish mutual trust relationships that bridge the differences between their local security models and thus preserve the integrity of their local security policies. We assist the communities in establishing trust relationship with the peer grids. The OSG is becoming more active in improving identity management by augmenting the current X509-based model for VOs that feel the X509-based model is not sufficiently user-friendly.

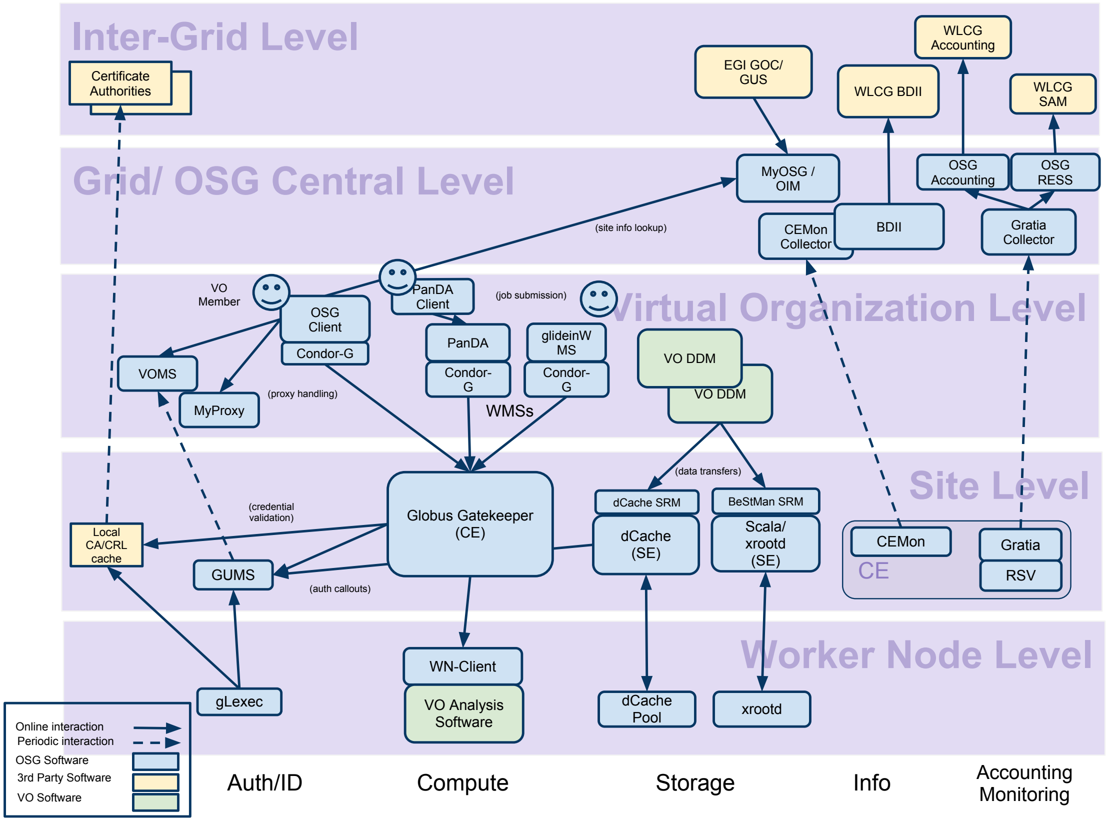
While the OSG provides software documentation, the over-arching best practices and principles are more effectively learned through **education and training courses**. We offer two annual schools, a weeklong school for students and researchers to learn about DHTC, and one for new and potential resource administrators. These are in-person training event with an emphasis on practical, hands-on experience. These schools are taught by staff (often drawn from the OSG areas) who actively use the technologies being discussed. After the school, the OSG runs a mentoring program to stay in contact with trainees, be a first point-of-contact for questions and issues that arise, and will encourage their trainees to increase their participation in the DHTC community.

4. Implementation of the Production Grid

This section aims to cover the technical details of the current Production Grid implementation. Implementation details change as the technology evolves.

We break the Production Grid functionality into five (overlapping) components: authorization infrastructure, computing, storage, information, and workflow management services. The schematic below outlines the four components, showing their key actors (both software and organizations), and how they interact at various layers.

Production Grid Functional Schematic



John Hover, RACF, BNL

In addition to the DHTC principles outlined in Section 2, we have followed additional principles in the implementation of the Production Grid:

- OSG does a minimal amount of software development. The OSG may do a significant amount of integration, but always prefers reusing software from external projects.
- Services should work toward minimizing their impact on the hosting resource, while fulfilling their functions. Any tradeoff between benefit and impact will constrain their design.
- Services are expected to protect themselves from malicious input overwhelming the hosting hardware and inappropriate use.
- All services should support the ability to function and operate in the local environment when disconnected from the OSG environment.
- While the OSG will provide baseline services and a reference implementation. Use of other services will be allowed and supported.
- The infrastructure will be built incrementally. The technology roadmap must allow for future shifts and changes.

- Users are not required to interact directly with resource providers. Users and consumers will interact with the infrastructure and VO services.

- The requirements for participating in the OSG Production Grid should promote inclusive participation both horizontally (across a wide variety of scientific disciplines) and vertically (from small universities to large ones national laboratories).

- VOs requiring VO-specific services at sites shall not encounter unnecessary deployment barriers at sites they own. However, VOs cannot require sites they do not own to run their services.

- Documentation for relevant target audiences is an essential part of any service and implementation.

- Documentation should be reviewed as appropriate for training and education.

- Services may be shared across multiple VOs. It is the responsibility of the administrative site to manage the interacting policies and resources.

- Resource providers should strive to provide the same interface to local resources as remote resources.

- 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 in a useable or failed state. The OSG will maintain an external downtime listing for services not expected to be useable.

- The infrastructure will support development and execution of 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 infrastructure should have minimal impact on a Site. Services will be run with minimal privileges on the host, especially avoiding the use of Unix user "root".

- 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. As services should also protect themselves, they should have the ability to quickly deny access when necessary.

- Allocation and use of a resource or service are treated separately.

- Services manage state and ensure their state is accurate and consistent.

We note that not all of these goals are achievable by the present implementation of the OSG Production Grid. Several are long-term goals that inform our discussions with external software providers on future improvements to their software.

4.1. Authentication and Authorization Infrastructure

The OSG has a multi-layered authentication and authorization infrastructure, designed to express the complex trust relationships, across not only the OSG but also peer grid based infrastructures – especially those in Europe that are part of the WLCG.

The grid authentication model is based upon PKI. Users obtain an X509 certificate from one of a set of trusted certificate authorities (CA). The CAs are responsible for vetting user identities, and users are responsible for keeping the private portion of the certificate secret. The authentication is based on the assumption that the holder of the private key is the person denoted by the X509 certificate.

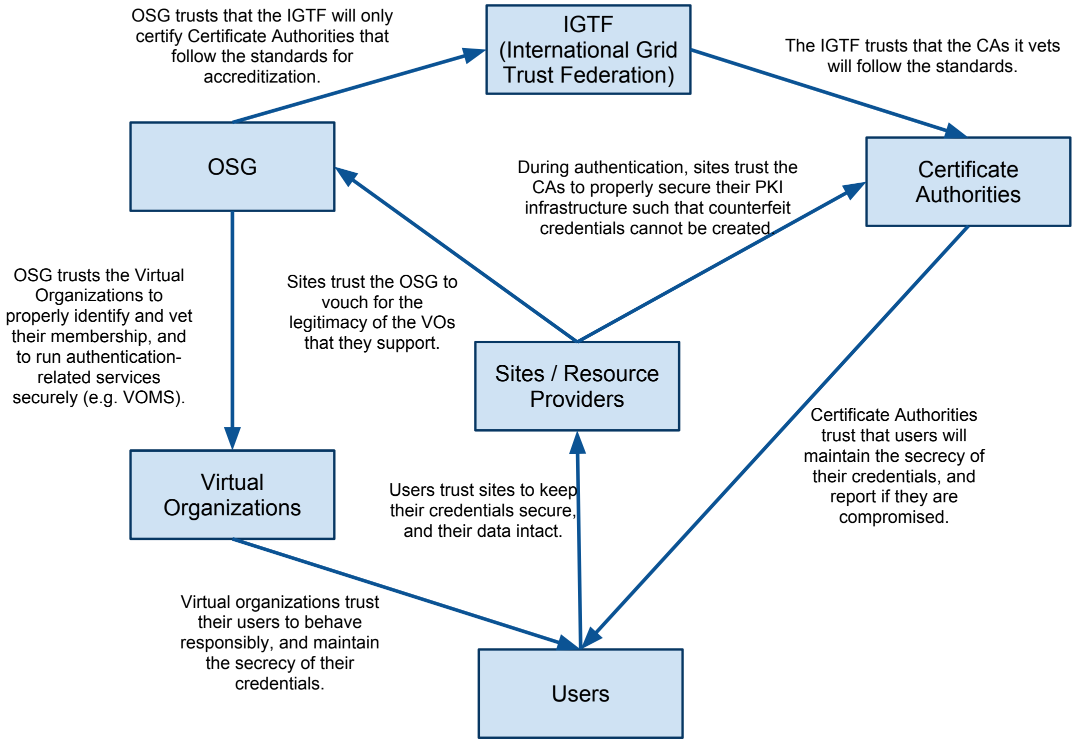
The X509 certificate can be used to create a "proxy" certificate, which is a short-lived (12 hours is the average) certificate based upon the user certificate. It's assumed the holder of a valid proxy certificate is conferred some or all of the privileges of the original user certificate. This proxy can specify additional attributes. One common attribute is whether further sub-proxies can be made from the original proxy. Another attribute is called a "VOMS extension", and it indicates membership in a VO, as well as the groups or roles the user has within the VO. Because of its limited lifetime, the proxy can be sent along with a grid job if the job needs to act with others on behalf of the user.

Authorization in the OSG Production Grid is based upon VO membership. Depending on the site's policy, an authenticated user can map is mapped: a single account for the entire VO; a shared account based on the user's group membership or role within the VO; or a pool account, allowing the user to have a unique Unix account at the site. Thus, the authorizations and privileges of the Unix user account are precisely those given to the remote user.

Under this scheme, the same user may be mapped to different accounts depending on the primary VOMS extension chosen for the interaction with the infrastructure. This allows, for a single user, different policies depending on the work to be performed. The VO controls the VOMS extensions accessible to a user. The privileges actually granted are controlled by the resource; different sites may implement different policies, although most VOs require their own sites to implement them uniformly.

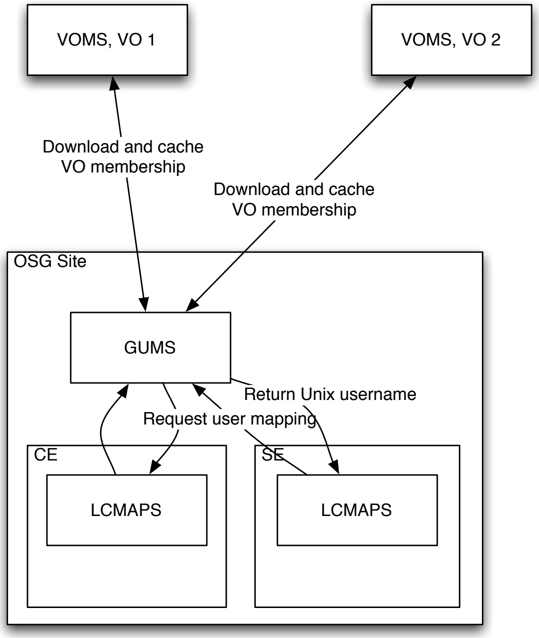
The trust relationships in the OSG Production Grid are given below.

OSG Trust Relationships


OSG trusts that the IGTF will only certify Certificate Authorities that follow the standards for accreditization.
IGTF (International Grid Trust Federation)
The IGTF trusts that the CAs it vets will follow the standards.
OSG
During authentication, sites trust the CAs to properly secure their PKI infrastructure such that counterfeit credentials cannot be created.
Certificate Authorities
OSG trusts the Virtual Organizations to properly identify and vet their membership, and to run authentication-related services securely (e.g. VOMS).
Sites trust the OSG to vouch for the legitimacy of the VOs that they support.
Sites / Resource Providers
Certificate Authorities trust that users will maintain the secrecy of their credentials, and report if they are compromised.
Virtual Organizations
Users trust sites to keep their credentials secure, and their data intact.
Virtual organizations trust their users to behave responsibly, and maintain the secrecy of their credentials.
Users

John Hover, RACF, BNL

To enable these authorization policies, each VO maintains a VOMS server. The VOMS software provides a web-services based interface that exposes the VO's membership and group structure. The information from each locally-supported VO is cached on-site in a database using software named GUMS. GUMS responds to local web services requests using a protocol called XACML. The site's services requiring authentication will send the X509 certificate's DN and VO attributes and will receive either a Unix username or an "authorization denied" message. By caching the VO membership information on-site, the GUMS server can hand out authorizations in a scalable fashion, even if the upstream VOMS server is offline. Note the authorization infrastructure can exist independently from the OSG. To lower the costs of deployment however, OSG Operations runs a number of VOMS servers on behalf of small VOs and OSG Software maintains a GUMS template configuration for new sites. These technical details are illustrated below.



4.2. Computing Infrastructure

The DHTC style is typified by taking a computational task requiring a large amount of computing time and breaking it apart into many smaller interdependent jobs. What constitutes as “large” varies between sciences, but starts at thousands of computational hours. On the high end, some results are always improved by additional throughput and are measured by the total amount of CPU time consumed over the course of a year. Better throughput is achieved by having as few interdependencies as possible, allowing many of the jobs to be run in parallel.

The OSG provides the OSG Compute Element (OSG CE) software stack. The OSG CE contains the services necessary for external entities to submit jobs to the local batch system using the OSG authorization infrastructure. To be “grid accessible”, a cluster would need one or more OSG CE endpoints. The services currently deployed include:

- **Gatekeeper software**: Allows secure invocation of batch system commands by remote clients. The gatekeeper takes a command from the remote client, performs a callout to the authorization client, translates the abstract command into a command for the local batch system, and then executes it on the remote client’s behalf. This provides an abstraction layer, ideally providing the user with a homogeneous view of the heterogeneous batch systems. Currently, the OSG CE utilizes Globus GRAM for the gatekeeper and will soon alternately provide EMI’s CREAM software.

- **Authorization clients**: The authorization infrastructure covered in Section 5.1 includes clients that run on the OSG CE. The CE calls out to the authorization client plugin for the gatekeeper and transfer services. The authorization service receives a summary of the client credential and either returns a Unix username (conferring the rights of that Unix user to the operation) or an authorization denied.

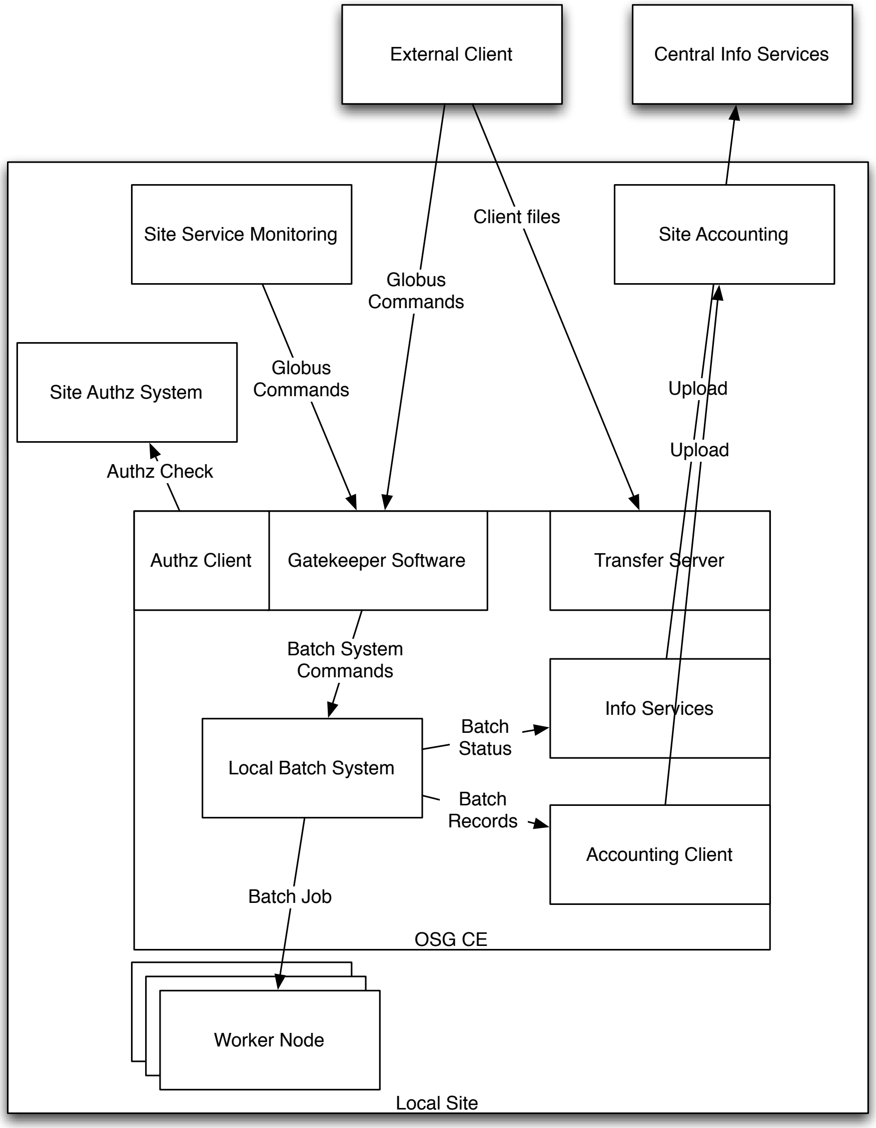
- **Transfer services**: Allows clients to transfer files back and forth to the OSG CE. Use is discouraged in favor of using dedicated storage services, but is sometimes used for small user job sandboxes.

- **Usage accounting**: Translates the batch system accounting for finished jobs into a standardized record format, the JobUsageRecord. These records are then uploaded into the Gratia accounting collector. Records are eventually sent to the OSG collector, but are preferably first aggregated at the site collector, if one exists. Usage accounting is covered in Section 5.4.

- **Information services**: Translates the CE's state into the GLUE 1.3 LDIF schema and an OSG-custom format based on Condor ClassAds. The CE's state includes information from the CE configuration files, batch system status, and any attached SE. This information is uploaded to the OSG central information services. Information services are covered in Section 5.4.

- **Service monitoring**: The service monitoring tests the gatekeeper software and job environment for minimal functionality. Most of the tests mimic very simple Globus-based jobs. Although this is installed on the OSG CE, it can be run on an external host and monitor multiple CEs.

The diagram below demonstrates how these pieces interact on the OSG CE host.



4.3. Storage Infrastructure

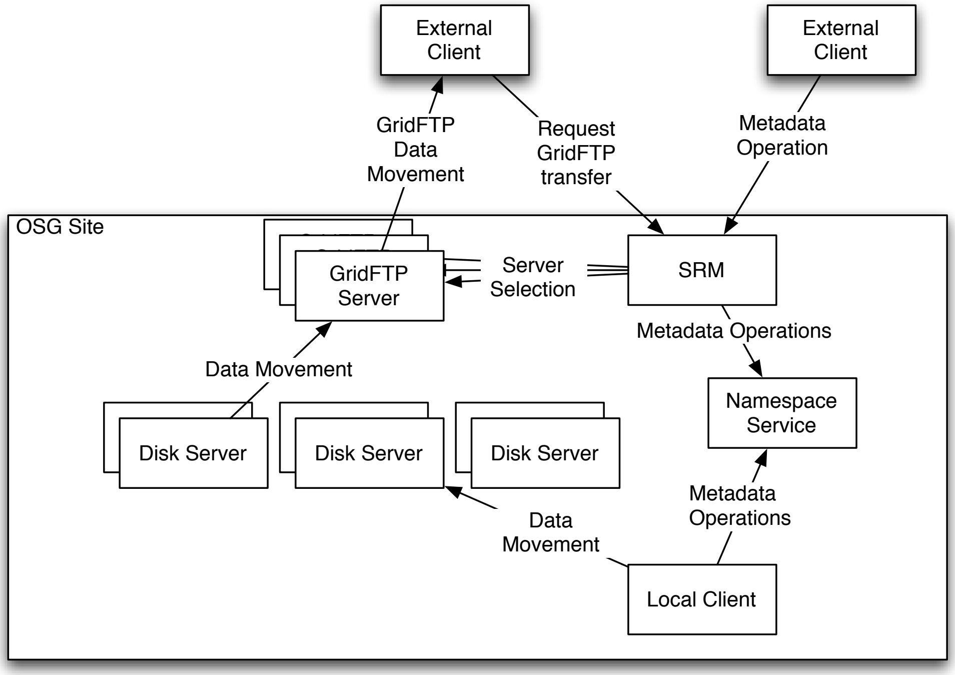
The OSG storage infrastructure provides small disk caches to large-scale (10s of TB to 10s of PB), robust storage systems, external storage management, and data transfer software. The state of the services has led OSG to provide packaging, configuration, testing, and support for the storage system implementations; lower-level work than for the computing infrastructure (we provide only minimal support for site batch systems).

The storage system allows one to take multiple disk servers and present them as a single, unified system. The OSG provides packaging and support for the Hadoop Distributed File System, dCache, and Xrootd storage systems. Other systems, such as Lustre, Panasas, and GPFS, may also be found in the OSG. OSG specifically does not limit the supported filesystems; implementations exist for the necessary external services that can be interfaced with any file system. Each system may present a unique interface to the local users. Users of dCache and Xrootd primarily utilize the dCap and Xrootd protocols via a custom APIs provided by a Linux shared library. The other systems also have custom internal protocols, but integrate at the Linux kernel level that provides users with a POSIX-like interface.

External users access the storage systems remotely using the Storage Resource Management (SRM) protocol, a web services-based protocol secured by a Globus GSI transport layer. The SRM endpoint performs authorization, mapping the remote user to a username for the storage system. The SRM operations are put into three groups:

- **Metadata operations**: Operations on the storage namespace; the equivalents to the venerable Unix *ls*, *mv*, *rm*.
- **Storage Management**: Storage systems are typically organized into several partitions. SRM allows for these partitions to be exposed as static space reservations (configured by the sysadmin) or re-partitioned into dynamic space reservations, reserved by the external user.
- **File transfer management**: Users can request SRM give them a URL the client can then use to transfer a file. This allows for load-balancing transfer servers using protocols such as GridFTP. Some SRM endpoints can even perform the transfer on behalf of the client, but this isn't uniformly implemented.

Storage systems allow for sites to serve data locally and remotely in a scalable and robust manner. For external users, the OSG has standardized around SRM for management and GridFTP for transfers. We have not standardized around a local access method or systems for coordinating large-scale transfers. The diagram below illustrates the generic components of an OSG Storage System.



The OSG storage systems require files to be copied to the local site prior to the data being usable, a “push” model. For the LHC VOs, we support deploying HTTP proxies using Squid and allowing a “pull” model. This is limited to smaller files and unauthenticated access. For authenticated, large-scale data movement using the “pull” model, we are investigating the Xrootd software.

4.4. *Information Services*

We divide the information types in the OSG into three categories: site topology, service status, and service state. These three categories are defined below:

- **Site topology**: This includes the relationships between sites and services, resource endpoints for services, and site contact information. This allows VO consumers to know the resources that officially are in the OSG Production Grid, and how to contact them.

- **Service status**: This is both the “effective status” of the service (functioning / not functioning) and the “desired status” (functioning / downtime). This informs end-users whether or not the service should be used.

- **Service state**: The overview of the activity or usage of a particular endpoint. May include size of the underlying resource (terabytes of storage or worker node cores), the resources free, and the per-VO breakdown of the utilized resource.

Some information is produced by multiple services and may be viewed by users in different formats from different hosts. While all the information sources should be consistent, this is not guaranteed to be the case in practice. For each piece of information, the OSG considers just one of the information services as authoritative in the case of conflicts.

4.4.1. Information Sources

The systems covered in this subsection are entry points for information in the OSG Production Grid.

Site topology information is kept by the **OSG Information Management** (OIM) database. This database is run centrally at GOC in Indiana. It records official contact information, the OSG facility/site/resource group/resource hierarchy, any WLCG-related information, and the site downtime info. When OIM information is needed for other services, MySQL DB replication is used to send the entire database to other hosts; due to security reasons, this database is only replicated to other machines at the GOC.

The **Generic Information Provider** (GIP) runs on the OSG CE and produces a description of the local site (including the computing and storage resources and services) using the GLUE schema in LDIF bindings (a simple text-based structured language). The GIP-produced data is transformed to a ClassAd-based format. Both the LDIF and ClassAd data is then uploaded to external aggregators using the CEMon software.

The **Resource and Service Validation** (RSV) software runs a set of common tests against OSG services to determine their functionality. Tests labeled “critical” by the OSG are used to determine whether the service is considered functional for the production grid. RSV is deployed with every CE, and we also encourage sites to host a single instance site-wide. RSV builds a local webpage and uses the Gratia transport (see below) to send the results to a Gratia collector at the GOC. Records from USLHC sites are uploaded from GOC to the corresponding WLCG system.

The batch system on each computing resource produces accounting records of jobs run; each batch system implementation (Condor, PBS, SGE, LSF) has its own format. The **Gratia** software has probes that convert the batch system records into the JobUsageRecord format. The new records are stored on disk, batched, and periodically uploaded to a remote collector. This push-architecture allows for records to be cached indefinitely on the local site if the collector is unreachable. Collectors can forward to other collectors, filtering the information as necessary, to form a tree. Large sites are recommended to run their own collector to aggregate records prior to sending them to the OSG. The collector stores the records into a local MySQL database and summarizes them into a more compact format suitable for querying. Gratia accounting has been extended to cover transfers performed by storage systems and the historical state (i.e., number of jobs running at a given point in time) of the batch systems. Gratia summary data is nightly uploaded to the WLCG for USLHC sites.

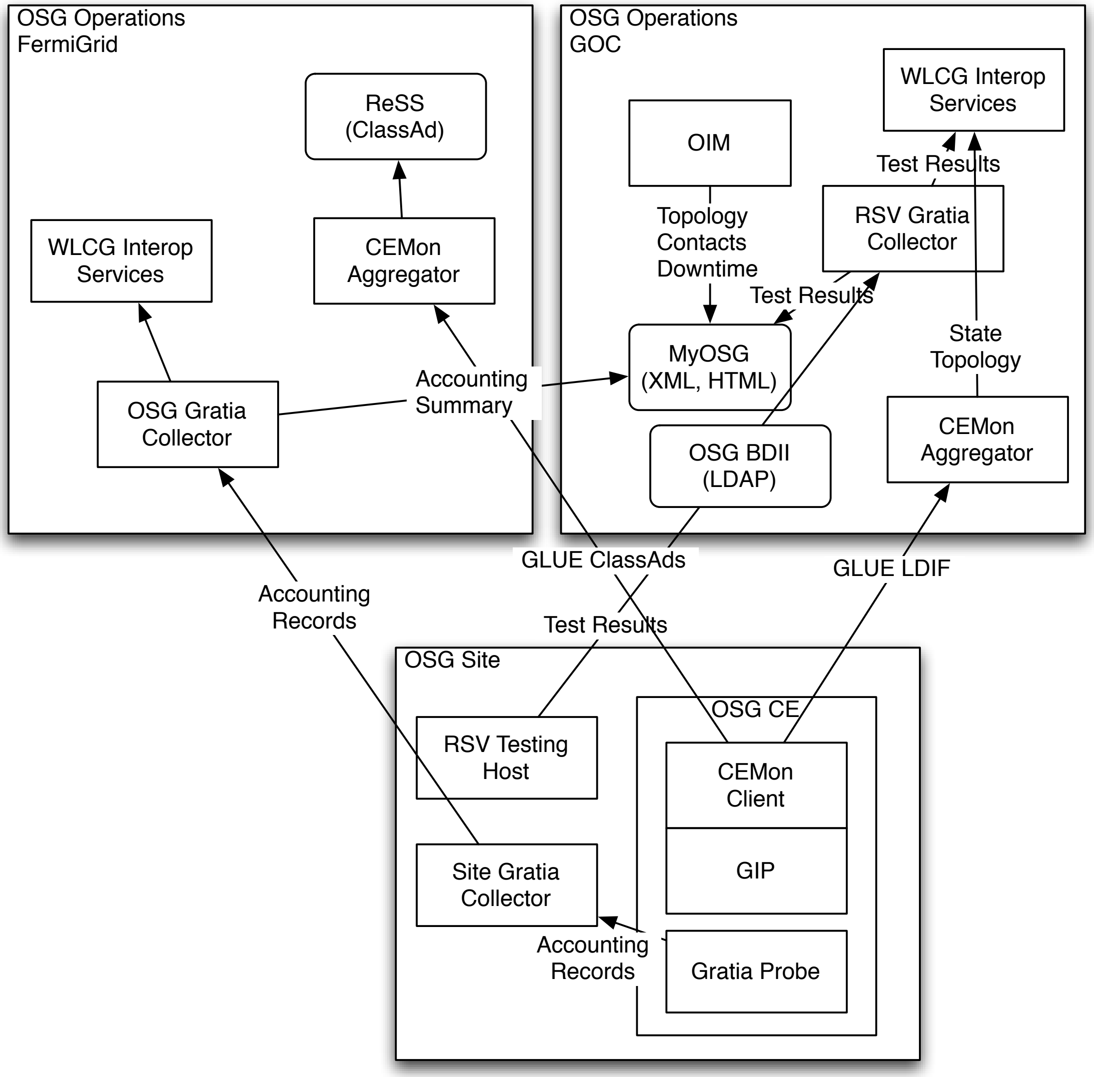
4.4.2. Information Sinks

The systems covered in this section are machine or human readable interfaces that export information outside the OSG grid.

The most heavily used machine-readable interface is the **BDII**. This provides an LDAP-based description of site topology and service state produced by the GIP. This information is primarily used to identify a set of usable service endpoints and provides state information useful in ranking the endpoints. This is used in both transfer and job submission workflows. The BDII service is used throughout the WLCG, and provides a level of interoperability at the information-service level between EGI and OSG. The LDAP data for USLHC sites is copied into a separate BDII instance that acts as a part of the WLCG information system.

The **Resource Selection Service** (ReSS) is a Condor collector daemon that presents a queryable interface to the GLUE information generated by the GIP. While querying Condor ClassAds is less well-known than LDAP or XML, it is common activity in this field and Condor has high-quality clients that VOs can utilize.

The **MyOSG** web application provides an XML and HTML view of the GIP data, Gratia accounting records, OIM topology, and RSV results. It is meant to provide a high-level overview of the OSG. While the XML interface is machine-readable, this service is primarily used by humans to view their site's performance.



4.5. Workflow Management Systems

One recent addition to the OSG Fabric of Services is the centralized support for Workflow Management Systems (WMS) for managing large-scale computational workflows. Traditionally, each VO would take the basic OSG submission service and build a WMS on top of it. As several dominant patterns emerged, we found the OSG could provide a WMS service and save significant duplicate effort.

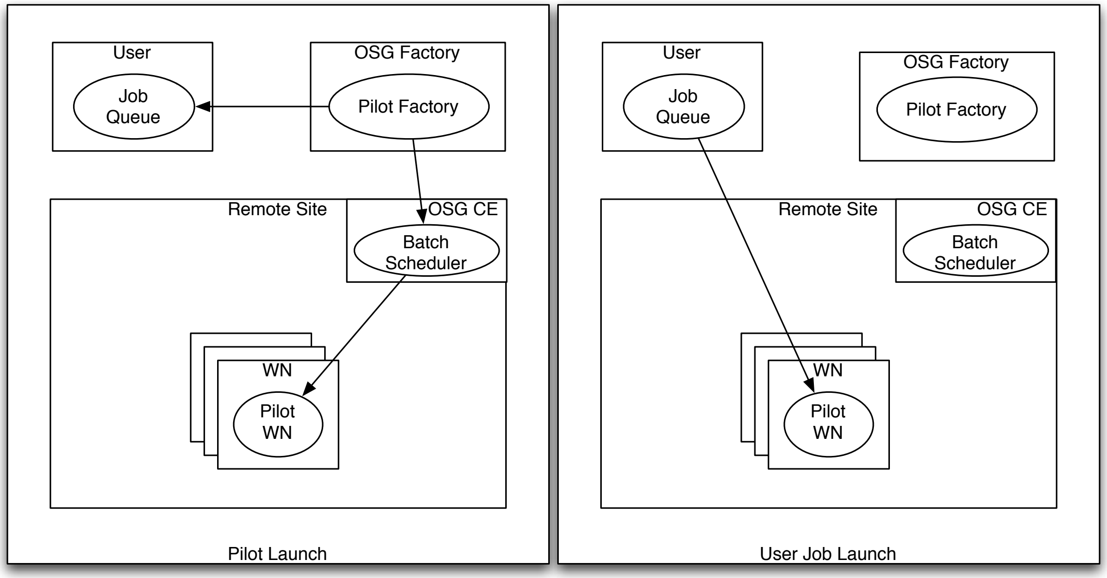
The best-practice is a pilot-based job submission system[4][5]; the most popular WMS systems are glideinWMS and PanDA. The pilot-based systems submit a "pilot job" to the remote site that, when run, verifies the runtime environment and downloads and executes the user "payload job" that contains the actual job. See the below diagram. The

[4] https://twiki.grid.iu.edu/bin/view/Documentation/JobSubmissionComparison

[5] http://osg-docdb.opensciencegrid.org/cgi-bin/ShowDocument?docid=93

payload job submit node and the pilot job submit node are typically separated; the pilot job submit node is referred to as the "pilot factory". The pilot factory is run centrally by grid experts, and can support many user submit nodes, even ones from different VOs. Thus, three key advantages of this infrastructure are:

- The user only interacts with batch slots that are already verified.
- Grid errors propagate back to the centralized grid experts, not the end-users.
- Job priority between users can be controlled by the submitting VO, not the remote site; this provides a more deterministic queue time for users, as they can't accidentally submit jobs to a site with a long or effectively infinite queue time.



4.6. Requirements of the Production Grid

In order to encourage as much usage and resources as possible, the OSG attempts to keep the requirements to a minimum. This section documents those requirements.

1. All resources must be registered. Registration implies name, administrative contact, security contact, and activation. In order for a resource in a new resource group to be activated, the administrative contact must request this at an OSG Operations meeting. New resources in existing resource groups can be activated via a support ticket.

2. Users, resources and service providers must accept the OSG Acceptable Use Policy. Services that receive delegated user credentials additionally agree to be honest stewards.

3. A User must be a member of at least one VO. OSG runs a catch-all VO for users that do not have one currently.

4. A service must be offered to at least one VO. Sites are encourage to allow the Ops VO to allow OSG Operations to assist in debugging.

5. The minimal requirements for participating in the OSG infrastructure are: the ability to advertise services in the common infrastructure; to accept use of one or more resource by applications running on the infrastructure; to abide by the security requirements; and to interact with the OSG services as needed for successful participation.

6. A minimal requirement on a Site is to provide some resources and an OSG service.

7. VOs and Sites will need to cooperate in order to permit the tracing of each transaction to a responsible user

8. Policy of a site takes precedence over the policy of a VO; both have to abide by the OSG AUP. In situations where site policy is in conflict with the owner VO, resolution happens outside of the scope of OSG.

5. Implementation of the Campus Grid

The OSG Campus Grid is a new set of distributed services being developed by the OSG. The Production Grid has been clearly motivated for several years by the needs of large communities like the LHC; the Campus Grid focuses on bringing DHTC onto the campus. Once well-established on the campus, we provide a clear path for bridging between campuses or to the Production Grid.

The current Campus Grid effort has been underway for about a year.

DHTC's advantages at the national level also apply at the campus level; however, the technology used at the national level does not necessarily translate cleanly to the campus level. The OSG's campus grid effort has focused on simplifying two aspects of the Production Grid: the PKI authorization infrastructure and the heterogeneity of the end-user submission interfaces. We do this by relying heavily on the Condor DHTC technology.

5.1. Authorization

The X509-based authorization infrastructure in Section 5.1 provides a highly secure, decentralized authorization model at the worldwide scale. It is designed to meet the needs of all the participating sites, including DOE labs. At some universities, such high level of security is not only unnecessary; it duplicates the credentials the user already maintains locally. Such duplication is a common user complaint about the Production Grid.

For campuses, we focus on allowing more heterogeneity in the selection of the authorization and authentication methods, but limit to those utilized by Condor. The campus authentication model is based on the local security in place on the campus. For inter-campus collaboration agreements are made between the campuses on the security and policies to be accepted. A non-exhaustive list of methods follows:

- **Username / password**: Many campuses still utilize the traditional username/passwords; users login to a submit host with a given Unix username using a password.
- **IP Whitelisting**: IP-based whitelisting of submit hosts are often utilized with username/password. Remote resources assume the submit host are secure (perhaps via agreement or because both are part of the same administrative domain) and map the user accordingly. As this is a lower-level of security, some resources will map the user to a low-privilege account such as Unix user "nobody".
- **Kerberos**: A common computer authentication protocol, popular for its ability for integrating with the campus's Windows Active Directory services.
- **X509 and VOMS-based**: The same authorization infrastructure on the Production Grid is available for the Campus Grid. Some campuses may use it, and it is an eventual requirement for overflowing jobs to the Production Grid.

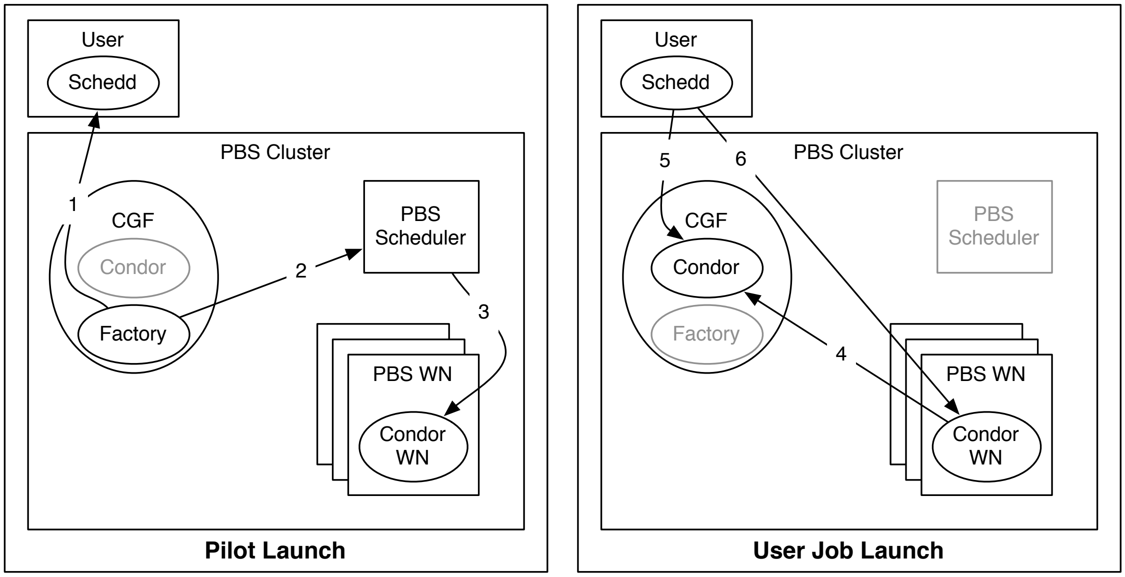
As Condor implements the authorization and authentication, a more in-depth discussion can be found in its manual[6].

5.2. Computing Infrastructure

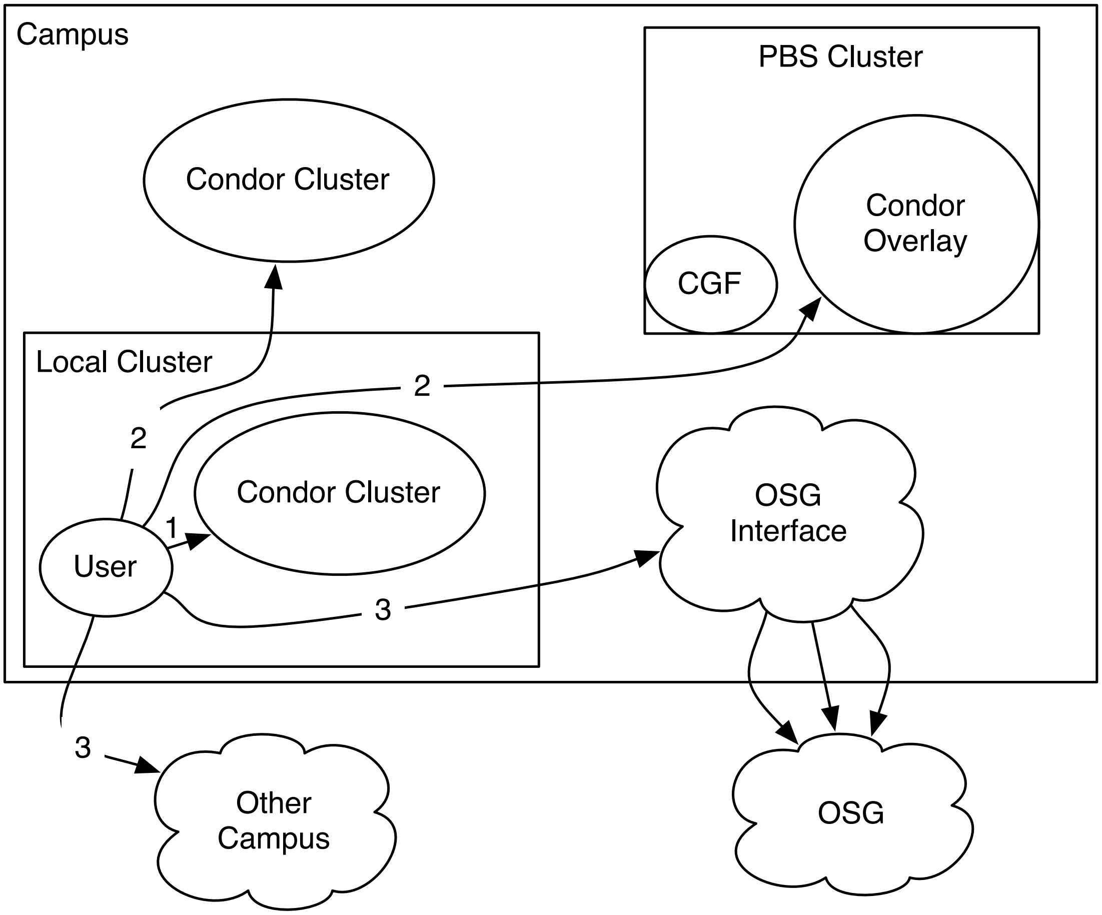
The Campus Grid computing infrastructure focuses on building a Condor external interface to each cluster (regardless of whether Condor is the batch system on the cluster). Once this interface is provided, any Condor-based submit host can submit to remote pools via a mechanism called "flocking". This allows the user to utilize the same "vanilla" Condor job on both his or her local machine and the remote campus cluster. This provides a uniformity of user experience necessary to "sell" the user on utilizing the grid.

To expose a Condor interface on a non-Condor batch system, we have built the Campus Grid Factory (CGF). The CGF is installed on each participating batch system and runs the Condor "central services" with no worker nodes. The Campus Grid Factory runs a simple process to detect when a remote user could use more resources, and submits a pilot job to the local scheduler. The pilot job is started by the non-Condor batch system and, in turn, starts a Condor WN that joins the CGF pool. This way, the CGF builds a "virtual resource" or an "overlay pool" of Condor worker nodes for the remote user. A user is then able to launch a job as if it were a normal Condor pool. See the below diagram.

[6] http://www.cs.wisc.edu/condor/manual/latest/3_6Security.html



The CGF allows the campus to join together all of its local resources in a coherent manner. Users can then start submitting to an widening circle of resources; they start with the local one they are most familiar with; then run on the friendly campus resources; then finally, bridge off-campus (to other similar campus infrastructures or the Production Grid). This allows the user to incrementally increase the complexity while getting the greatest payoff. When the user runs into issues, this expansion model should also provide the best assistance; the user will interact with local administrative help before off-campus. This method of expansion is diagrammed below.



5.3. *Information and Accounting Services*

Information and Accounting services are relatively rudimentary on the campus grid. The Condor system provides voluminous data about resource state; however, each resource one submits to must be configured by hand. For submit hosts that reach many resources, it is not uncommon to have a hand-maintained list of 10-15 endpoints.

Accounting is performed using the same Gratia software as the Production Grid. One missing capability identified is the ability to record "flocked" jobs run on the local resource. We believe this will be remedied in the future. As shared resources on the local level have varied local ownership, we believe accounting should answer "How many hours did I run on remote resources?" and "How many hours did my local resource give to remote users?". The latter is currently missing.

6. Terms and Definitions

The basic terms are defined within the scope of the Open Science Grid. An attempt has been made to define a useful set of simple definitions upon which the end-to-end infrastructure can be built. Definitions that follow dictionary definitions and standard usage are not repeated here.

- **User** – A person who makes a request of the Open Science Grid infrastructure.

- **Resource** - A resource is any physical or virtual entity of limited availability[7]. In the OSG, all resources are represented by a unique DNS endpoint.
- **Resource Owner** – has permanent specific control, rights and responsibilities for a Resource associated with ownership.
- **Agent** – A software component in OSG that operates on behalf of a User or Resource Owner or another Agent.
- **Consumer** – A User or Agent who makes use of an available Resource or Agent or Service.
- **Provider** – Makes a Resource or Agent or Service available for access and use.
- **Ownership** – A state of having absolute or well-defined partial rights and responsibilities for a Resource depending on the type of control. OSG considers two such types: actual Ownership and Ownership by virtue of a Contract/Lease. A Lessee is a limited Owner of the Resource for the duration of the Contract/Lease.
- **Service** – A method for accessing a Resource or Agent.
- **Resource Group** – A named collection of resources for administrative purposes.
- **Administrative Domain** – One or more resource groups run by under a single set of policies, often indicating the resources are run by a single team.
- **Site** - A collection of resource groups under a single administrative domain.
- **Facility** – A collection of administrative sites that are a part of the same organization.
- **Virtual Organization** – A dynamic collection of Users, Resources and Services for sharing of Resources (Globus definition). A VO is party to contracts between Resource Providers & VOs which govern resource usage & policies. A subVO is a sub-set of the Users and Services within a VO which operates under the contracts of the parent
- **Virtual Site** is a set of sites that agree to use the same policies in order to act as an administrative unit. Sites and Facilities negotiate a common administrative context to form a "virtual” site or facility.
- **Policy** – A statement of well-defined requirements, conditions or preferences put forth by a Provider and/or Consumer that is utilized to formulate decisions leading to actions and/or operations within the infrastructure.
- **Delegation** – An entrustment of decision-making authority during transfer of request for work or offer of resources from a User or Agent to another Agent or Provider, or vice versa. The latter is provided with a well-defined scope of responsibility and privilege at each such layer of transfer of request or offer.
- **Documentation –** is qualified by the target audience. This includes the User, Consumer, the Software and Resource Providers, as well as Internal for staff maintainers, supporters and new entrants, The target readers of technical documentation are Developer, Documenter, Scientist (end-User), Student, System Administrator, VO Manager.
- **Security** – Control of and reaction to intentional unacceptable use of any part of the infrastructure.
- **Grid** – A named set of Services, Providers, Resources, and Policies, overlapping and/or including other Grids operating as a coherent infrastructure in support to

[7] http://en.wikipedia.org/wiki/Resource

the contracting Virtual Organizations. Providers may delegate their contracts with the participating VOs to the Grid administration.

- **Community (Cyber-)Infrastructure** - A set of services and software that has been established by a community to meet the needs of its members. The management of the distributed infrastructure is the responsibility of the community, and the resources are all, or nearly all, owned by the VO and members.
- **Cloud** – A set of Services, Providers, Resources and Policies providing a single point of access for all the computing needs of consumers. The resources are not necessarily owned by the consumer, but may be leased or otherwise "accessed.
- **Campus Grid** – A grid operated within the context of a single facility (such as a university of a national lab).
- **"Identity" Federation** - A set of one or more Organizations and a set of zero or more Certificate Authorities that are Trusted. A Federation provides information about Individuals and the Organizations (e.g. to a CA).

Authors:

Brian Bockelman	John Hover	Miron Livny

Revisions: (is style "meta")

Version number	Date (use function: Insert -> Date and Time, format as: 9/5/03) Do NOT check "update automatically".	Initials of author making revisions	Summary of changes
2.0	3/3/11	BB, JH, ML	Reset of the Blueprint document.