GRATIA: NEW DEVELOPMENTS IN GRID ACCOUNTING

Fermilab

Tanya Levshina

Open Science Grid

Outline

- Gratia Overview
- Ongoing work
  - Campus Grid Accounting
  - Cloud Accounting
  - Integration with other Grids
    - APEL/SSM Integration Status
    - XSEDE
- Future work

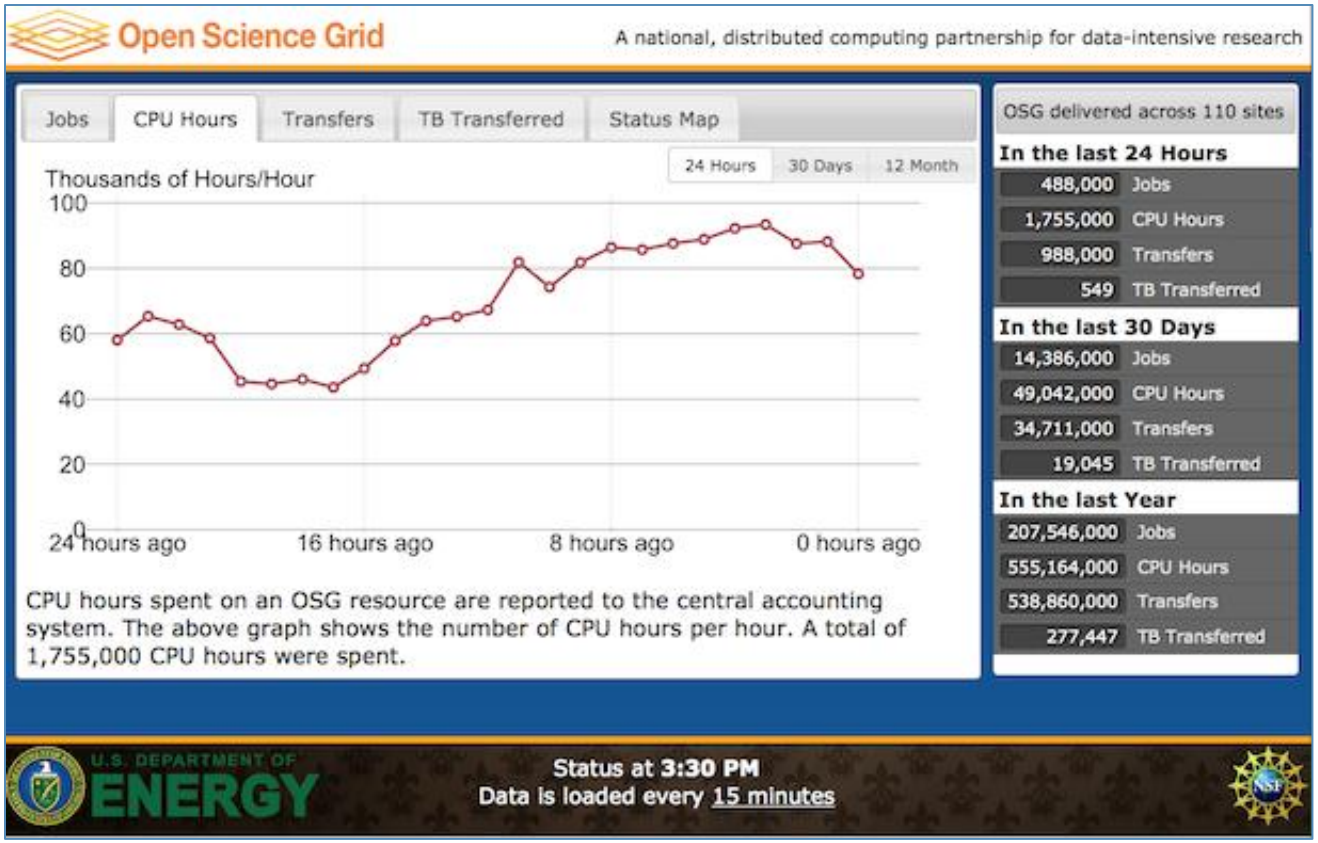
Contributors

- Brian Bockelman
- Ashu Guru
- Parag Mhashilkar
- Mats Rynge
- John Weigand
- Derek Weitzel

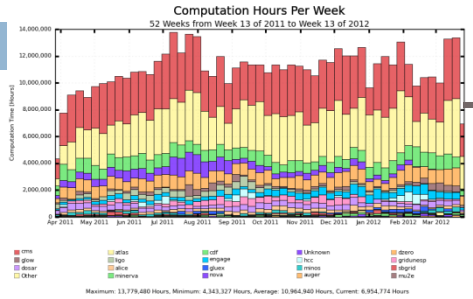
Gratia Overview

OSG Display

Displays high-level summary of the activities across the OSG



Gratia Architecture



OSG Gratia Collectors/Reporters

collector

reporter

mySQL

mySQL

OSG Site A

Gratia Service Host

collector

reporter

mySQL

CE A1

condor probe

transfer probe

CE A2

pbs probe

transfer probe

WN

WN

WN

psacct probe

glexec probe

SE A

storage probe

transfer probe

OSG Site C

OSG Site B

CE B

pbs probe

transfer probe

SE B

storage probe

transfer probe

Gratia Overview & Statistics

- Information is generated by various probes and sent to Gratia collectors via Gratia API. Collects information about:
  - Batch and glide-in jobs (condor, lsf, pbs)
  - Linux process accounting
  - Various Metrics (RSV probes)
  - File transfers
  - Storage Usage
- Supports multiple collectors and allows hierarchical forwarding between collectors. Allows data filtering and replication
- The maximum rate observed for processing is 200,000 records per hour in production.
- Number of jobs processed (not including duplicates)
  - gratia job records 634,714,713
  - gratia transfers 1,239,014,562

OSG Storage Accounting

- A dedicated collector (collects transfer accounting data as well)
- In 2009 two new accounting entities have been added:
  - Storage Element
  - Storage Element Record
- StorageElement is used to describe static information and storage topology (name and type of storage, storage area parent, etc)
- StorageElementRecord is used to store dynamic information: space measurement
- The design is based on OGF Usage Record standard
- Storage gratia probes are currently developed for:
  - dCache
  - HDFS
  - Xrootd

Gratia Probes

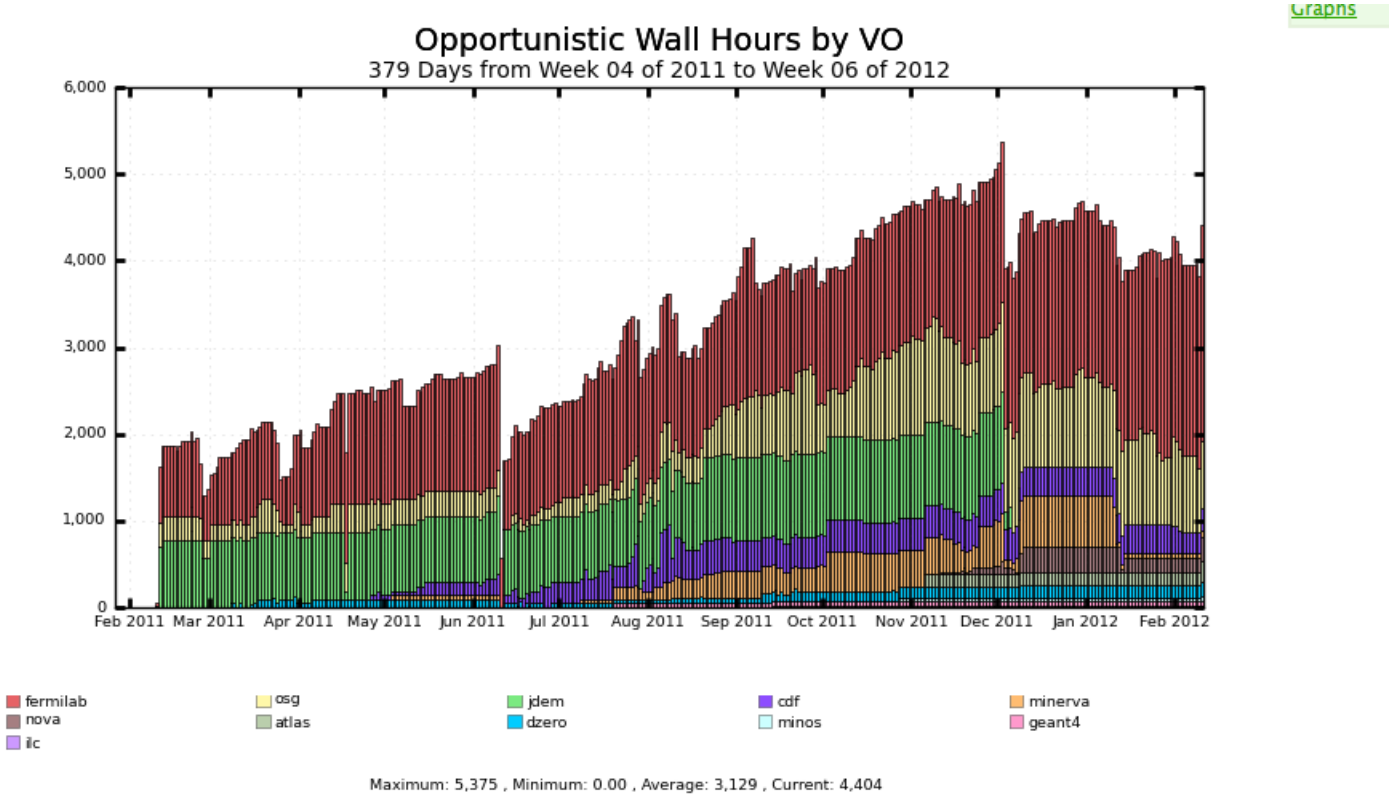
Campus Grid Accounting

- A Campus Grid(CG) offers a resource sharing mechanism that allows campus users to submit jobs to multiple computational resources. It is not limited to resources on campus but provides means to use resource from other campuses or the OSG.
- The primary concern for Campus Grids accounting is an accurate recording of usage of the OSG resources by CG users. We need to account for jobs that have "flocked" to OSG resources from CG submitters to GlideinWMS frontend as well as record local usage. A local campus users usually don't use a certificate.
- Several modifications were made to gratia probe common libs , so probes now have information on how to map a local user properly if the relevant information is specified in the probe configuration.
- Ongoing discussions:
  - Do we need to create any new reports specific to CG?
  - Do we need to register submitter nodes in OIM ?
- Derek Weitzel and Ashu Guru are working on solution for these issues:
  - https://indico.fnal.gov/materialDisplay.py?contribId=48&sessionId=5&materialId=slides&confId=5109

Cloud Accounting

- Fermi Cloud project lead by Keith Chadwick and Steve Timm
- We are running Open Nebula 2.0 (production) and 3.2 (testing)
- All information about VMs owner, physical machine, states, start and stop times, memory and cpu usage could be acquired from interfaces provided by Open Nebula.
- OGF Usage Record structure is currently sufficient to represent VM accounting information
- No changes have been required to the core Gratia Service at least during the initial phase to store the records
- Open Nebula Gratia AccountingProbe (developed by Tanya Levshina and Parag Mhashilkar)
  - Runs on a ONE management node
  - Collects data via ONE API
  - Creates standard Gratia Usage Records
  - Reports to a dedicated Gratia collector

OpenNebula Accounting Report



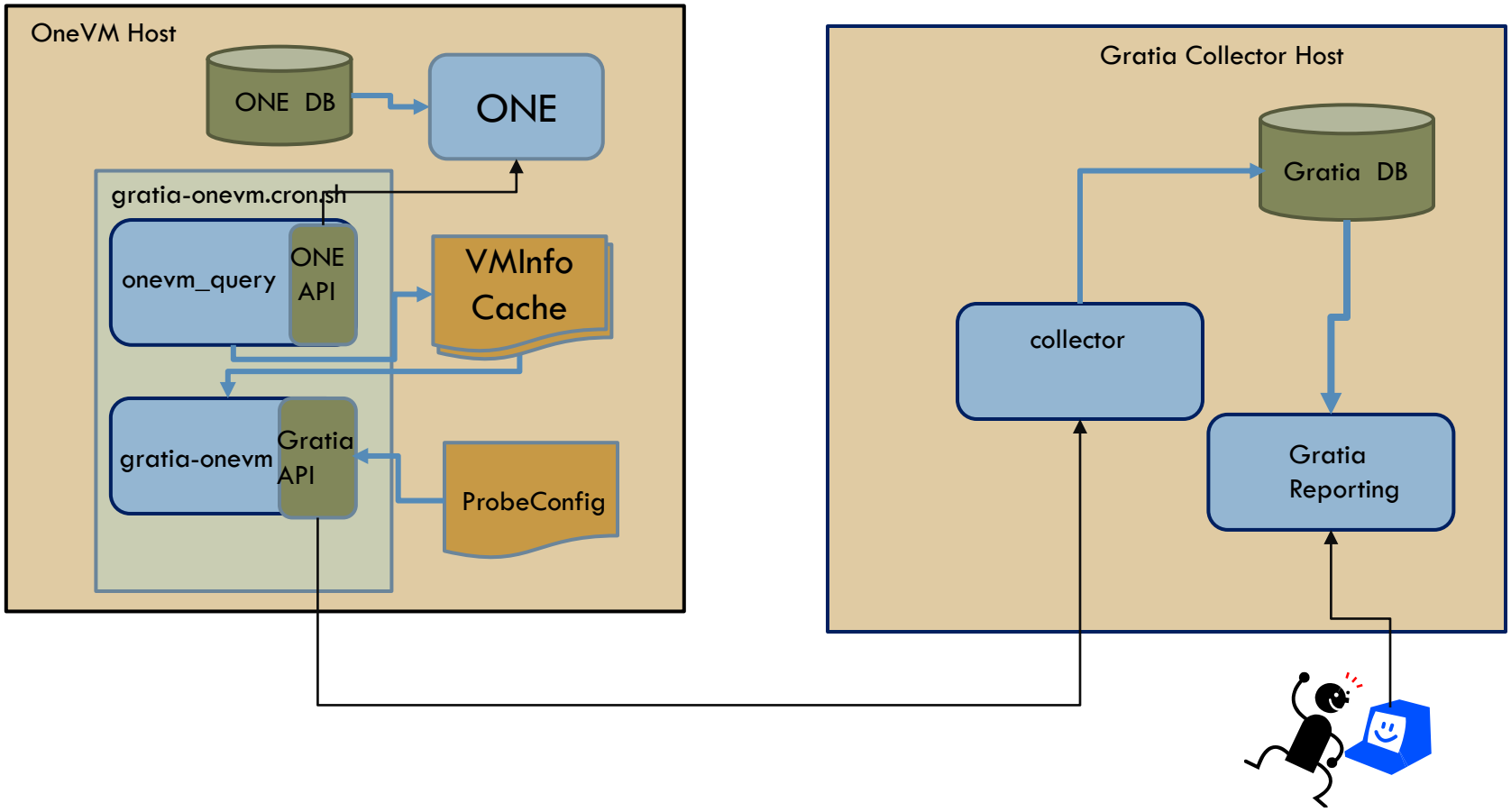
Open Nebula API

- Considerations:
  - It was tempting to use command line tools to get all the information, but as soon as VM is shutdown or deleted there is no way to get historical information
  - The historical and current VM information is accessible through ONE API
    - Filtering is limited. Currently, API doesn't provide time based filtering.
- Conclusion: Need to implement a tool to identify changes to the VMs in a given time frame.

Probe Implementation

- Option 1: Extend the ONE APIs to talk to the database and report back corresponding VMs. For the time being we don't have effort to do that. We are collaborating with ONE developers to make filtering API available (as for other activities, e.g. Authentication). Hopefully we can use this option in the nearest future.
- Option 2: Use existing APIs to extract VMs info, let Gratia VM probe go through all the data and discard info that has been already reported
  - Cons - Scalability: as the size of VM pool increases, querying all the VMs to get required info is expensive.
- Option 3: Option 2 with caching
  - Every time a query executed it caches relevant information
    - VMs that underwent a valid state change
    - Last VM Id known for the given time frame.
  - For subsequent queries, use info from the cache.

Cloud Accounting Architecture

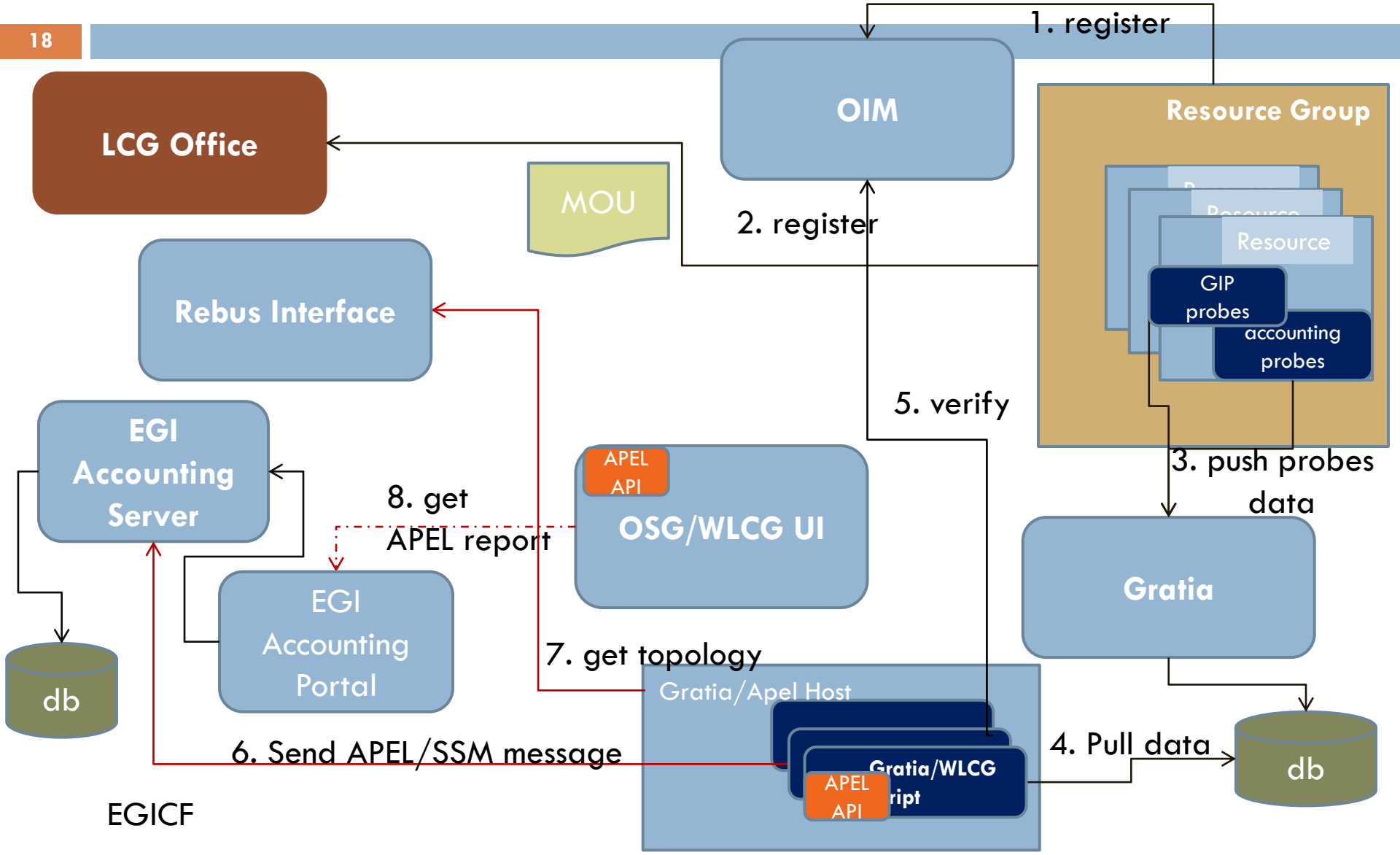


Accounting & MultiGrids Infrastructure

Gratia/APEL & SSM Status

- Development has started in October, 2011. The work is done by John Weigand.
- Daily tests are running since December, 2011.
- We are now testing SSM 0.10-1 software (rpm and tarball version).
- We have completed tests against the “production” server.
- It appears that there is no way for us to verify that data have actually propagated into the system and are accurate. The ability to retrieve data from portal critical to verifying if the interface worked, we also have other software that depends on that.

Gratia/APEL & SSM Workflow



EGICF

Reporting of International VOs

- Many international VOs use resources across the EGI and OSG infrastructure.
- We have a request from some of them (ILC, D0, SBGrid/WeNMR) to investigate if we can share the accounting information between OSG and EGI.
- Publishing of international VO OSG data to APEL/EGI.
  - In December a test was made to verify if ILC VO data would propagate to the EGI Accounting portal. This test was successful and accounting info appeared in the OSG view only. This was a one-time test and the data was subsequently removed.
  - Work will be required in the interface software itself and in MyOSG/OIM to determine how to identify those VOs that are to be reported.
  - Visibility to the data in EGI, beyond the OSG view of the data, is not within the OSG scope of work.

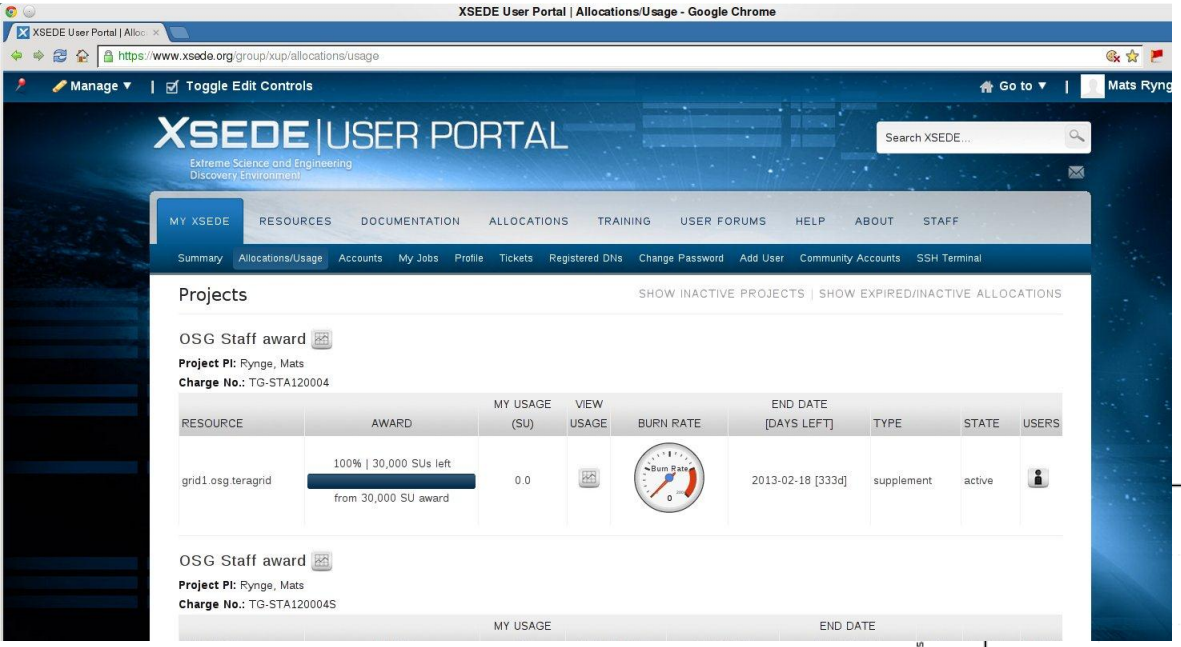
OSG as an XSEDE Service Provider

- OSG will be an official XSEDE Service Provider starting April, 2012.
- OSG Staff (Mats Rynge and others) have contributed to this effort:
  - https://twiki.grid.iu.edu/bin/view/VirtualOrganizations/OSGasXsedeSp
- OSG provides a Virtual Cluster that shields XSEDE users from the necessity to be aware of the OSG infrastructure. Jobs submitted into this Virtual Cluster will be executed on machines at several remote physical clusters.
- In order to successfully submit a job, a user should specify, among other attributes, a valid Project ID in a condor job description file.
- The condor gratia probe running on the submission node reports usage records to Gratia. It includes Project ID into the record. From querying Gratia JobUsageRecord table:

CN	Project ID	uname	Host	ProbeName
/CN=yzheng	*TG-STA120004S*	*yzheng*	*red-d8n6.red.hcc.unl.edu*	*condor:osg-xsede.grid.iu.edu*
/CN=rynge	*TG-STA120004S*	*rynge*	*node245.red.hcc.unl.edu*	*condor:osg-xsede.grid.iu.edu*

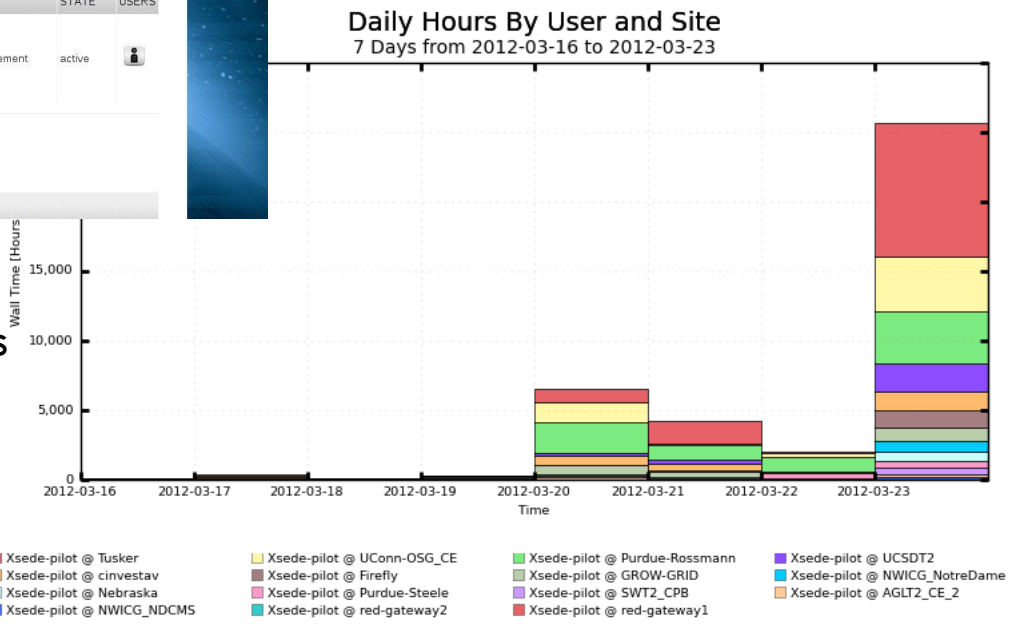
- XSEDE tracks resource usage in its own account management system (AMIE). The script, that is running on the OSG-XSEDE submits information from Gratia collector, summarizes this information by grouping it by Project ID, and pushes this information to XSEDE accounting system using AMIE API.
- In the nearest future we are planning to introduce the Project ID field in Gratia DataSummary table, pull relevant information from there, and push it to AMIE.

Integration with XSEDE Accounting



Summary of allocation usage for XSEDE users running on the OSG resources

Gratia summary bargraph of cpu hours used by XSEDE jobs running on the OSG resources



Future Work

Network Accounting

- Goal: Provide network accounting for batch systems
  - Design and prototype have been done by Brian Bockelman:
    - http://osg-docdb.opensciencegrid.org/cgi-bin/ShowDocument?docid=1083
  - Currently only Condor batch system on Linux is considered due to widespread use in OSG
  - The implementation requires Condor 7.7 & RHEL6 to gather network information
  - The proposed solution was implemented by combining network namespaces, network pipe devices and ipatbles/netfilters network accounting
- Minor modifications are needed for gratia probe
- Gratia DB Schema already allows to store network information. The modifications are required to include jobs network usage information in producing summary data.

Summary

- We are adapting the infrastructure to meet the new technological challenges of campus infrastructure, cloud computing, and interoperability.
- New probe development is in progress:
  - cloud accounting (Open Nebula)
- Modifications of the old probes are ongoing:
  - To make condor probe flexible so it could report information about campus flocked jobs
  - To assign a particular science field (project name) to a job
  - To include network utilization in job usage record
- Integration with other Grids is important (APEL/SSM, XSEDE). Good communication is crucial for integration success!