PUBLIC STORAGE FOR THE OPEN SCIENCE GRID

Tanya Levshina
Ashu Guru

Fermilab is Operated by the Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the United States Department of Energy.

Outline

- Motivation for the OSG Public Storage
- Use Cases
- Requirements
- iRODS Overview
- Public Storage Architecture
- Status
- Issues
- Summary

Motivation for the OSG Public Storage

Goals:

- Manage opportunistic storage provided by OSG sites.
- Help small Virtual Organizations with grid jobs data handling.

Problems:

- The common tools for automatic management of allocated storage do not exist.
- Small VOs have difficulties finding appropriate storage, verifying its availability, and monitoring its utilization.
- The involvement of a Production Manager, site administrators, and VO support personnel is required to allocate or rescind storage space.

Are we reinventing the wheel?

- The LHC Experiments are using the Storage Elements of Tier-1, Tier-2 and even Tier-3 sites to pre-stage and upload TBs of data.
- They maintain a file catalog to keep track of data location.
- There are multiple custom solutions used by the experiments in order to achieve these goals (FTS, PhEDEx, AliELN, SAM-Grid, AAA).
- Unfortunately, these services
  - Either are tightly coupled with the experiments' internal workflow
  - Or require significant investment in hardware, and software support and maintenance (e.g. Oracle).
  - Usually deal only with input data.
- There are other means to manage data:
  - HTCondor data transfer mechanism could be used to ship data with a job and return output files to a submission node. It is problematic to use this mechanism for "big" files (> 1 GB) simultaneous transfer to/from a submission node.
  - SQUID service is effective when jobs are reusing the same files such as calibration and configuration data, SQUID doesn't help when each job requires a unique set of files.

Use Cases

- SNOWMASS (Simulate hundreds of millions of high-energy proton-proton collisions, which mimic the collisions expected at future hadron colliders).
  - Need to pre-stage big files (3 – 15 GB) to selected SEs.
  - Need to download these files on a worker node during job execution.
- EIC (Electron Ion Collider at BNL: Modeling the performance and optimizing the design)
  - Pattern A: Pre-stage files (1 GB) to OSG_DATA and copy files from $OSG_DATA to a worker node during job execution.
  - Pattern B: Pre-stage a file to “SRM” SEs then copy file to all worker nodes.
- DetectorDesign (Medical Imaging, University of New Mexico: Investigating how different simulated SPECT system geometries can affect reconstructed images)
  - Upload output files to a local/remote storage from a worker node.
  - Download all the files from various SEs to user’s laptop.

Requirements

We have been looking for a storage service that:

- Enables small non-LHC VOs with "large" data requirements to use OSG sites with less effort.
- Allows the OSG Production manager to administer public storage allocation across all the participating sites.
- Imposes minimal burden on participating sites.

We don't want to start from scratch:

- The service should be integrated with existing OSG middleware. It should use available Storage Elements.
- Has a strong community support.

We are currently exploring the feasibility of integrating the Integrated Rule-Oriented Data System (iRODS) with the OSG SEs for providing the OSG Public Storage.

iRODS (I)

- The iRODS was developed by the Data Intensive Cyber Environments research group and its collaborators.
- iRODS implements a policy-based data management framework.
  - Handles various objects (resources, collections and files) .Each object has a set of properties (metadata) associated with it.
  - Allows to define polices (set of rules). Rules trigger a chain of actions (micro-services) that may include recovery and notification.
  - Provides means to set quota limit and enforce quota management.

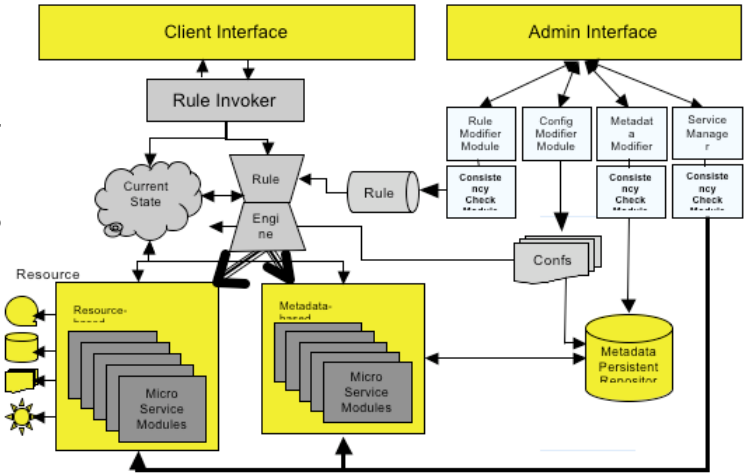


Figure 5. iRODS Architecture Components

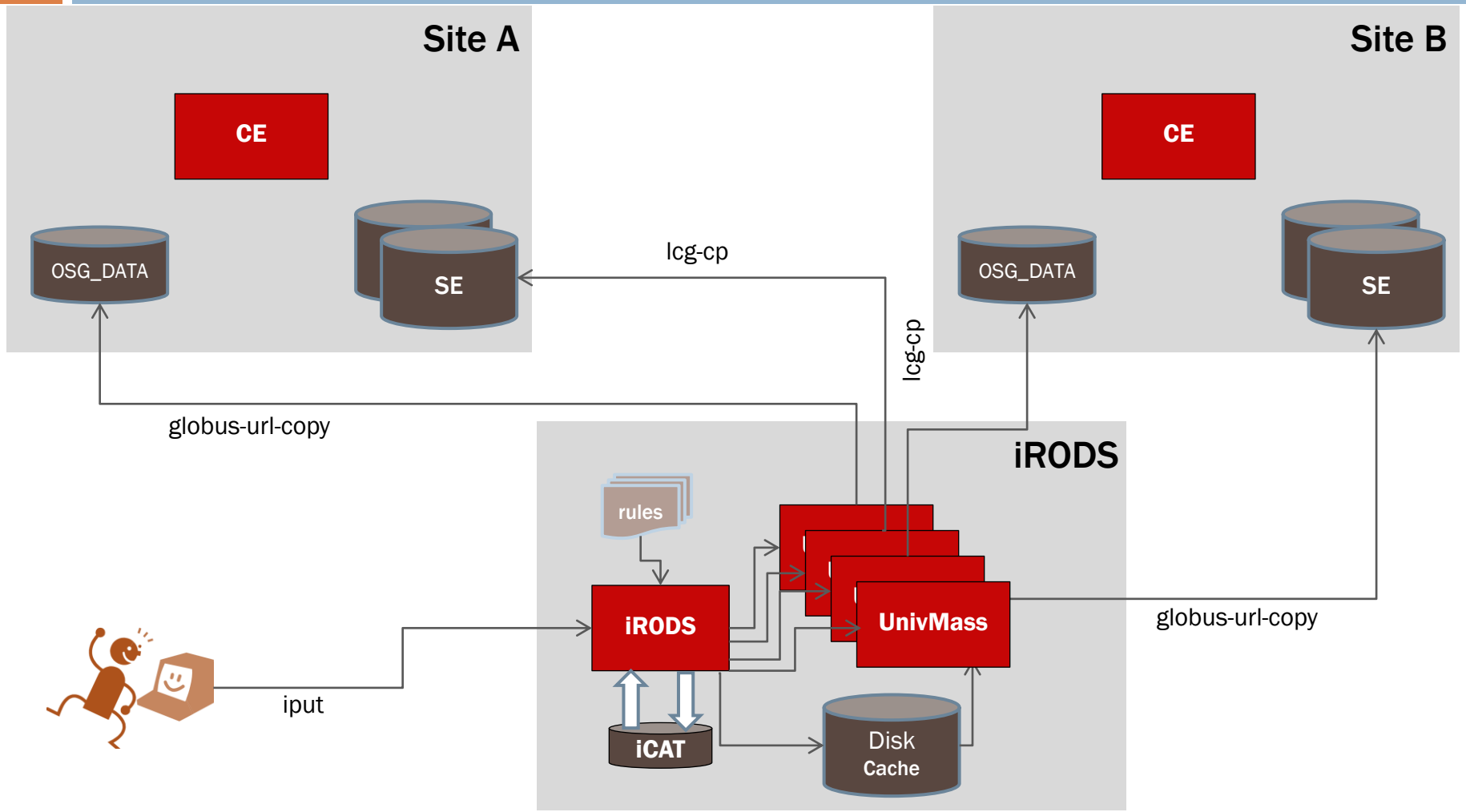
https://www.irods.org/index.php/File:Irods-intro5.png

iRODS (II)

- iRODS performs transfers by
  - using implementation specific protocol to access POSIX compliant resources
  - using an external driver to Mass Storage. The driver should implement "put" and "get" methods to transfer entire files. File transfer is performed in two steps (disk cache is needed)
- The Metadata Catalog (iCAT) stores complete state information about the system in a database. iCAT contains information about resources, resource usage, quotas and users. It also serves as a metadata catalog for users' data collections.
- Widely used by various scientific communities (Biology, Environment , Physical Sciences, Geosciences, etc.)

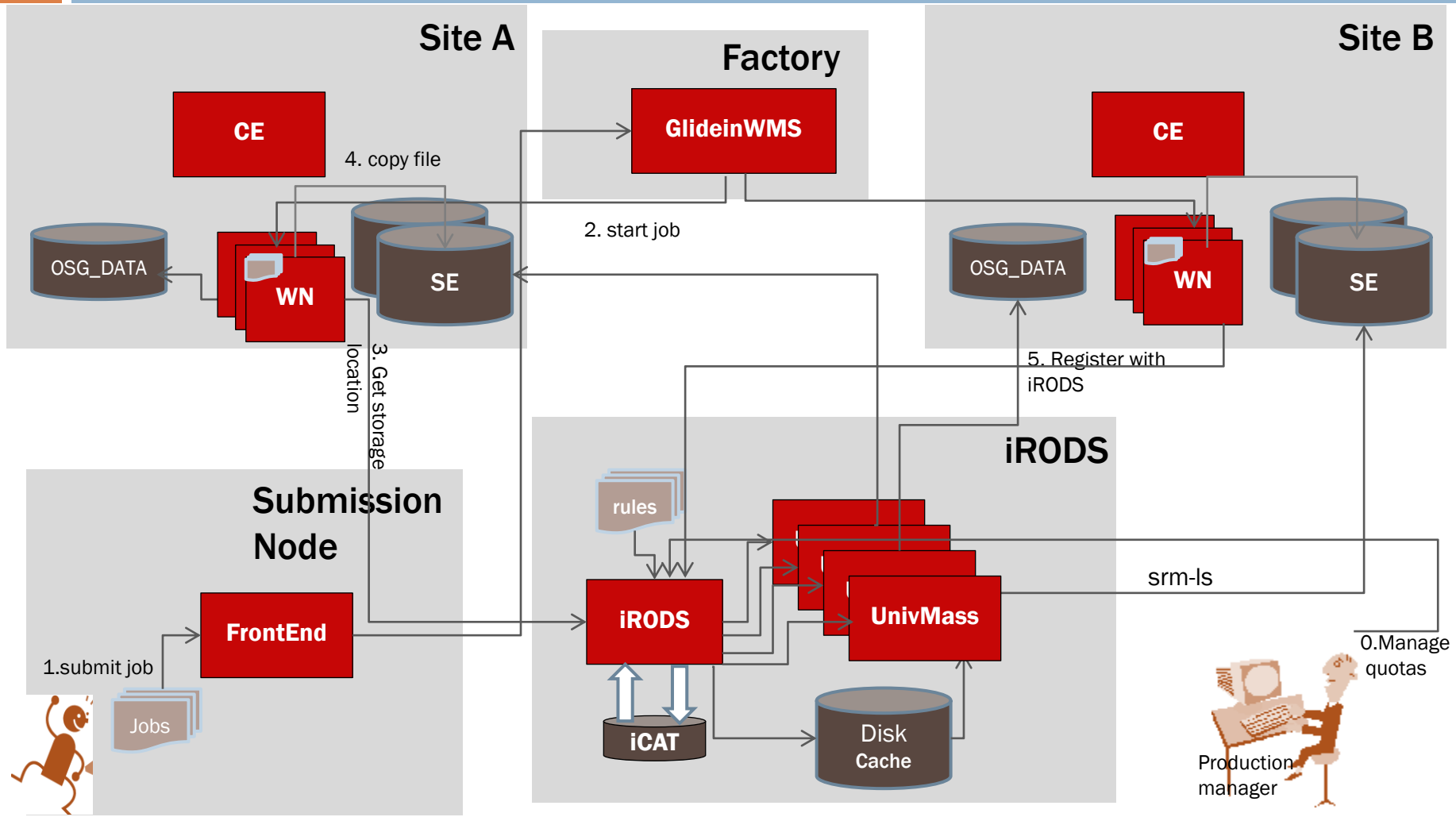
OSG/iRODS Integration

(pre-staging data)



OSG/iRODS Integration

(running grid job)



iRODS integration pros and cons

- Advantages:
  - Allows a user to pre-stage data to OSG_DATA and SRM SEs via iRODS without dealing with sites, gathering scattered information about site resources, worrying about storage location and end path.
  - Provides a global namespace that has information about files location, size, etc.
  - Manages quota per VO/resource.
  - Doesn't impose any burden on the sites
- Disadvantages:
  - File pre-staging/download happens in two hops.
  - One cannot utilize iRODS features fully because of the architecture we are using:
    - We need to write and maintain custom scripts
    - Cannot achieve same performance

Current Setup

- Currently we have defined two resource groups (osgGridFtpGroup with 10 resources and osgSRMGroup with 23). We have two VOs registered with iRODS (osg and hcc) and 14 “users”.
- “VO” is defined as iRODS users’ group
- A resource has the following attributes:
  - Quota and Usage per VO/resource
  - Status (up/down)
  - Meta data: Supported VO, Storage SURL, End path per VO
- X509 authentication is used between a user and iRODS , as well as between iRODS and SEs. Multiple certificates can be associated with the same user. A user could be a member of one or more groups.

User Level Data Management

- pre-stage a file to a specific SE: iput –R Nebraska my_file
- pre-stage a file to some resource group: iput –R osgSrmGroup my_file
- download a file from SE: iget my_file
- delete a file from SE: irm my_file
- replicate a file from one SE to all other available SEs within the same SE group:
  irepl-osg –R osgSrmGroup my_file
- list detailed file information: ils –l file_name
- In order to use the OSG Public Storage for grid jobs a user should:
  - define the following attribute in his/her condor job description file :***+UsesiRODS=True***
  - add to the job executable

$IRODS_PLUGIN_DIR/icp idrodse://irodsuser@irods.fnal.gov:1247?/osg/home/username/<input_file> <input_file>

$IRODS_PLUGIN_DIR/icp <output_file> idrodse://irodsuser@irods.fnal.gov:1247?/osg/home/username/<output_file>

Production Level Data Management

- Enable quota enforcement rule.
- Custom clean up rule that is triggered by quota limit change
- A Production Manager monitors the current space utilization by using custom script:

```
ilsresc-osg -g osg -G osgSrmGroup -q
resourcegroup: osgSrmGroup
Group: osg Resource: Firefly
Quota: 204800 MB (214748364800 bytes(s)) Used:21866 MB (22928189106 byte(s))
Group: osg Resource: CIT_CMS_T2
Quota: 204800 MB (214748364800 bytes(s)) Used:5423 MB (5686886432 byte(s))
Group: osg Resource: FNAL_FERMIGRID
Quota: 102400 MB (107374182400 bytes(s)) Used:5423 MB (5686886432 byte(s))
Group: osg Resource: Nebraska
Quota: 412121 MB (432141068068 bytes(s)) Used:21234 MB (22265609700 byte(s))
Group: osg Resource: UTA_SWT2
Quota: 204800 MB (214748364800 bytes(s)) Used:72561 MB (76086652476 byte(s))
```

- A Production Manager needs to be involved in iRODS operation ONLY when:
  - New VO has requested access to iRODS.
  - New resource is added.
  - A quota must be modified .
  - Membership is changed.

Current Status

- Deployed on a VM at Fermilab (2 Cores, 4GB memory, 2TB disk)
- Demonstrated the feasibility of managing public storage at the OSG sites with iRODS.
  - A Production Manager can manage resource allocations at remote sites between various VOs.
  - No action is required from the sites after initial allocation of resources.
  - A user can upload and download files from a user laptop or a worker node using iRODS commands and in-house developed scripts.
  - A user can pre-stage data to the registered resources (SEs).

Issues

- iRODS doesn't provide any kind of throttling.
- Currently we are collecting information about resources by using *iquest* command. The machine starts swapping and eventually gets out of memory as soon as the number of simultaneous requests reaches 50. Looking at replacement of *iquest* query as well as upgrading hardware.
- We also have to implement a throttling daemon that controls the number of simultaneously executed SRM commands initiated on iRODS machine via Universal Mass Storage plugin to remote SEs.

Summary

- The OSG still doesn't have a generic approach for public storage. A pressing need to provide a data handling solution for small VOs is mounting.
- Integration with iRODS seems to provide a feasible solution for accessing and managing public storage at the OSG sites.
- The iRODS scalability problems need to be addressed before we can move to production deployment and offer it as a common solution for the OSG small VOs.

References and Contacts

- iRODS Home Page
  https://www.irods.org/index.php/IRODS:Data_Grids,_Digital_Libraries,_Persistent_Archives,_and_Real-time_Data_Systems
- OSG iRODS Docs and Tutorial:
  https://twiki.grid.iu.edu/bin/view/VirtualOrganizations/IRODSOSG
- iRODS-Chat google group:
  https://groups.google.com/forum/?fromgroups#!forum/iROD-Chat
- Contacts:
  - tlevshin@fnal.gov

Acknowledgements

Many thanks to Gabriele Garzoglio, Chander Sehgal, and Brian Bockelman for the productive discussions related to the design and implementation of the OSG Public Storage.

Fermilab is Operated by the Fermi Research Alliance, LLC under Contract No. DE-AC02-07CH11359 with the United States Department of Energy.