System Administration Best Practices

Open Science Grid

Site Administrators Meeting

July 31, 2007

Steven Timm

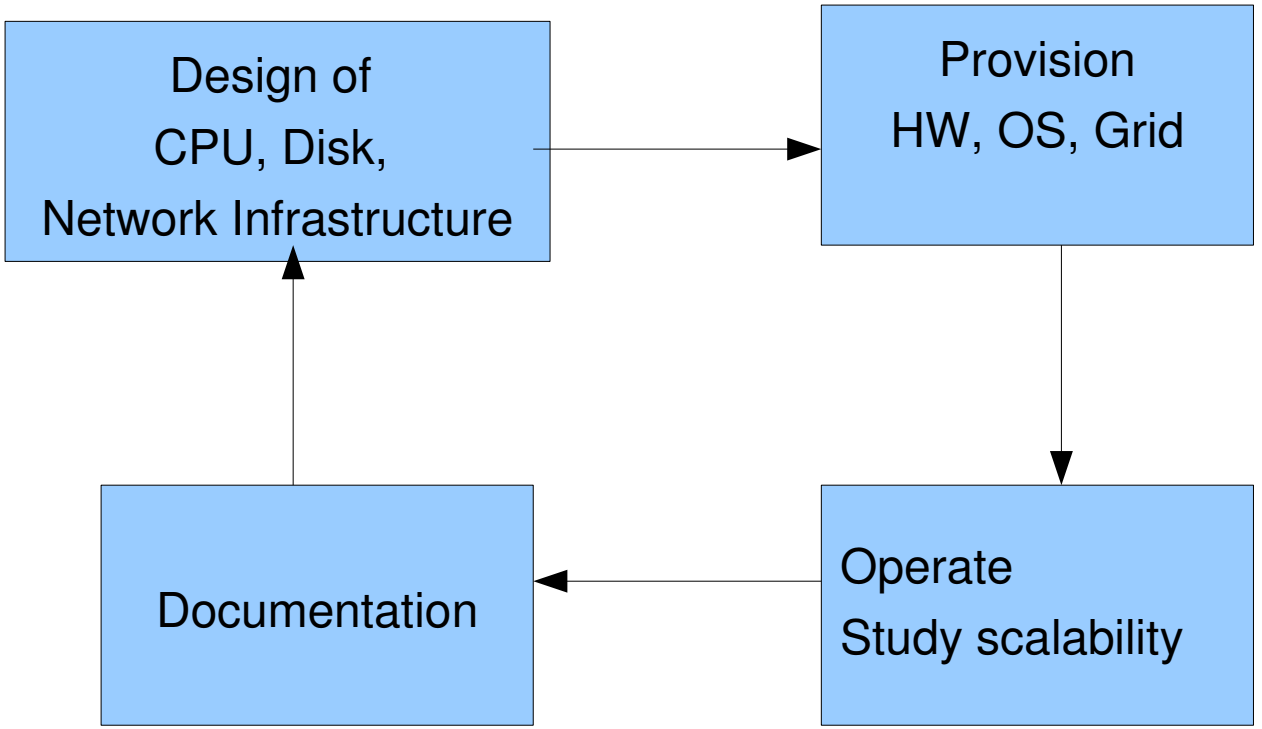
Fermilab Computing Division

FermiGrid Services Group

Introduction

- The grid as sold to site-admins:
  - Wouldn't have footprint on compute nodes
  - Was lightweight
  - Was already working and scalable

What's Involved In System Administration



Hardware and Software Scalability Experience

- Fermilab's first production OSG sites started in April 2005.
- FermiGrid now has a site job gateway and seven production gatekeepers
- http://fermigrid.fnal.gov/cgi-bin/main.cgi
- Most FermiGrid clusters are sending nearly all jobs through the grid/gt2 OSG interface—much more load than typical OSG site.
- Fermilab also has site scanner which scans all ports with nmap on regular basis—has found a couple of Denial of services vulnerabilities already.
- Scalability is dominated by the corner case—submit client going crazy, NFS glitch, etc, multi-thousands of very short jobs, fast forks pulling big tarballs, etc.

Typical current gatekeeper/head node (fcdfosg2)

- Vintage 2005, dual Xeon 3.2 Ghz cpu's, hyperthreaded
- 6GB RAM
- 3ware 9500 controller, RAID 5 array, 1TB disk
- One node runs condor collector, negotiator, schedd, globus gatekeeper, web services, cemon, and YP services.
- Biggest one has 400 nodes, 2000 Job slots
- Load often goes over 60. We are at the limit of single-node combination gatekeeper and batch system master. Have to either upgrade or split functionality.
- Load is dominated by system cpu—which means IO/network dominated, can't just buy faster cpu and hope to fix it.

Gatekeepers we budgeted for in FY2008

- Dual dual-core Intel Xeon 5150 CPU's
- 12 GB RAM (anticipate running 3 Xen instances on each).
- 6 SAS (serial attached SCSI) disk in RAID 10 configuration
- Moving towards a high availability architecture
  - Two condor collectors active at all times
  - Active/passive gatekeeper/WS/CEMon setup with shared FS
  - Active/passive condor schedd
- 2 of these machines in hand for our site gateway now.

Ways to improve scalability without $10k per node

- Split condor collector/negotiator off to a different node. Simple worker node will work.
- Put in extra condor_schedd to manage fork jobs
- Add condor_quill with postgres database to offload work from the schedd
- Add more RAM and swap

Shared File Systems

- Many VO's, both Fermilab and otherwise, find /grid/app and /grid/data areas useful.
- These areas plus the home areas of our 2500 grid-based uid/gid's are stored on Bluearc NAS appliance. (14 TB of RAID-6 SATA HDS raid behind it).
- Without this appliance we couldn't do what we are doing. There is no Linux NFS server up to this load. (40000 IO ops/sec in our cluster.)
- (a user did 200 simultaneous chmod -R from 200 nodes across a directory that contained 300K files and didn't crash it)
- If you don't have an enterprise-class NFS server consider jobmanager-condor-nfslite, now in the VDT.
- Be careful of permissions
  - Fermilab recommendations, don't share home areas between grid machines and non-grid machines
  - Don't let non-grid machines have execute access to arbitrary grid areas.

Provisioning—a Comparison

- WWW, early 1990's
  - Download tcp winsock from BBS
  - Buy CD of browser from netscape.com or build it from source
  - Build apache httpd from source
  - Consult /etc/hosts which contains all hosts on Internet
- OSG, 2007
  - Download pacman
  - Install and configure vdt
  - Install extra bleeding-edge packages and bug workarounds
  - Pacman -update--OOPS
  - Delete and start over
  - Find a site in VORS and run

Goals for Provisioning

- For large clusters, grid software should go on with OS reinstall.
- Software should just be there in the OS
  - Especially the worker node client and the user client
  - Use standard update mechanisms (up2date,yum)
  - Get software into the Linux distros wherever possible
  - **This means RPMS**. Promised Aug. 2006. Coming when?
- RPMS are only a win if they can install and upgrade clean.

Provisioning Tasks

- Open Science Grid Computing Element
- Batch System (Covered in previous talks)
- Open Science Grid Worker Node Client
- Open Science Grid Client
- Authentication/Authorization
- Extra grid stuff

OSG Computing Element

- Installing the VDT just the beginning
- Make sure GIP is right (network, cpu, memory)
- Make sure advertisement to BDII/ReSS works OK
- Hack jobmanagers as necessary for scratch dir's, queue assignments, condor requirements.
- Keep up to the pacman updates (CA Certs, etc)

The OSG Worker Node Client

- Goal—Reduce NFS dependency of OSG stack
- Essential worker node software including CA certificates
- If all worker nodes in OSG ran fetch-crl cron, woul d kill CA's.
- Current best practice—run fetch-crl on one node and NFS mount it—Ouch!
- Worker node client now bigger than initial CE stack and many OS installs
- In ITB 0.7.0/OSG 0.8.0, gLexec is available—suid executable
  - Can't do NFS—worker node client really has to be local
  - New Cert distribution is in the works to send certs and crl's from head node to workers.

The OSG User Client

- Already we have had three security alerts against this.
- What is OSG plan to make sure all Clients are upgraded?
- Many ways to configure Condor-G wrong, or insecurely.
- On Web—most security holes in the browser, on Grid, many security holes in the client. Need a grid-wide model.
- FermiGrid is encouraging users to log into central machine and use our client. Main interactive cluster “FNALU” will soon have the client too.
- A few centralized submit points have helped us greatly with support of our own VO users, when we can see client submit queue as well as CE queue.
- We will add under-covers feature to our Client to send uid, condor clusterid via globusrsl to all sites.

Authentication/Authorization

- On FermiGrid we have central authentication (GUMS, SAZ) but distributed authorization.
- NIS used to distribute 2500+ grid users uid/gid's
- All have /sbin/nologin shell
- Growing number of non-recycled pool accounts becoming a scaling problem
- Batch system right now uses unix uid/gid for priority factors and job slot quotas.
- In progress of getting host cert for each worker node to run GSI-authenticated volume.

The extra grid stuff

- FermiGrid is a beta site for Condor, Gratia, gLexec, CEMon, ReSS, SAZ, and other projects to come.
- “yum” update mechanism from Scientific Linux (and now also available in Fedora Core, RedHat 5) used to provision the extra rpm-based utilities.
- We also install Condor from rpms, not the VDT. VDT doesn't track condor upgrades that come out between OSG releases. RPMS more work but let us upgrade when we need to without messing up our VDT.

OSG Policy That Could Help

- Force individual, dynamic OSG_WN_TMP for each job.
  - Users can't be trusted to clean up after themselves
  - All batch systems have auto-cleanup utilities and dynamically created work dirs which OSG has to do work to defeat.
- Policy on how big files transferred into home area can be
- Have local site differences be advertisable
  - Initial directory supported
  - NFS-lite
  - Jobmanager time limits

Operations Helper Scripts

- Various sites have come up with their own scripts
  - Clean up $OSG_WN_TMP on worker nodes
  - Dynamically set $OSG_WN_TMP at run time
  - Clean up $HOME/.globus , gram_scratch areas
  - Kill old and stale globus-job-manager processes
  - Distribute certs among the worker nodes
  - Multi-hundred host cert collections
- It has recently been asked on osg-int if these can be organized and collected. Good idea.

Best Practices Documentation

- https://twiki.grid.iu.edu/twiki/bin/view/Integration/ITB_0_7/SiteFabricBestPractices
- This was created at 2006 Consortium meeting and has been updated since then
- One parallel session then, now a whole meeting
- Still some blank spots in this table,
- We need volunteers to update it during upcoming Integration cycle.
- There is also space in this document for site and Fabric developments
- Also look in Pacman section of Integration Twiki—there is a lot more stuff about pacman surgery in there than there used to be.