Deploying Galaxy on a Shared-Node Cluster University of Alabama at Birmingham (UAB)

2012-11-17

Curtis Hendrickson, John Osborne (CCTS)
Shantanu Pavgi & John-Paul Robinson (Research Computing)

CCTS

UAB THE UNIVERSITY OF ALABAMA AT BIRMINGHAM

UAB Research Computing

Outline

- Motivation
- Organization
- Timeline
- Infrastructure
- Deployment
- Customizations
- Successes
- Wish List

Motivations

- NGS
  - Researchers requesting help with NGS analysis
    - extra mural sequencing centers not following through
  - UAB Sequencing Core (HiSeq2000, GAIIx, MySeq, etc.)
- Genomics Workbench
  - Replace aging GCG (Accelrys SeqWeb) installation
  - Workflow engine for our informatics consulting group
  - Delivery of analysis and results by informatics group
  - Self-service by researchers
  - Easy access to campus cluster for non-bio applications.

Organization & Model

- Collaboration

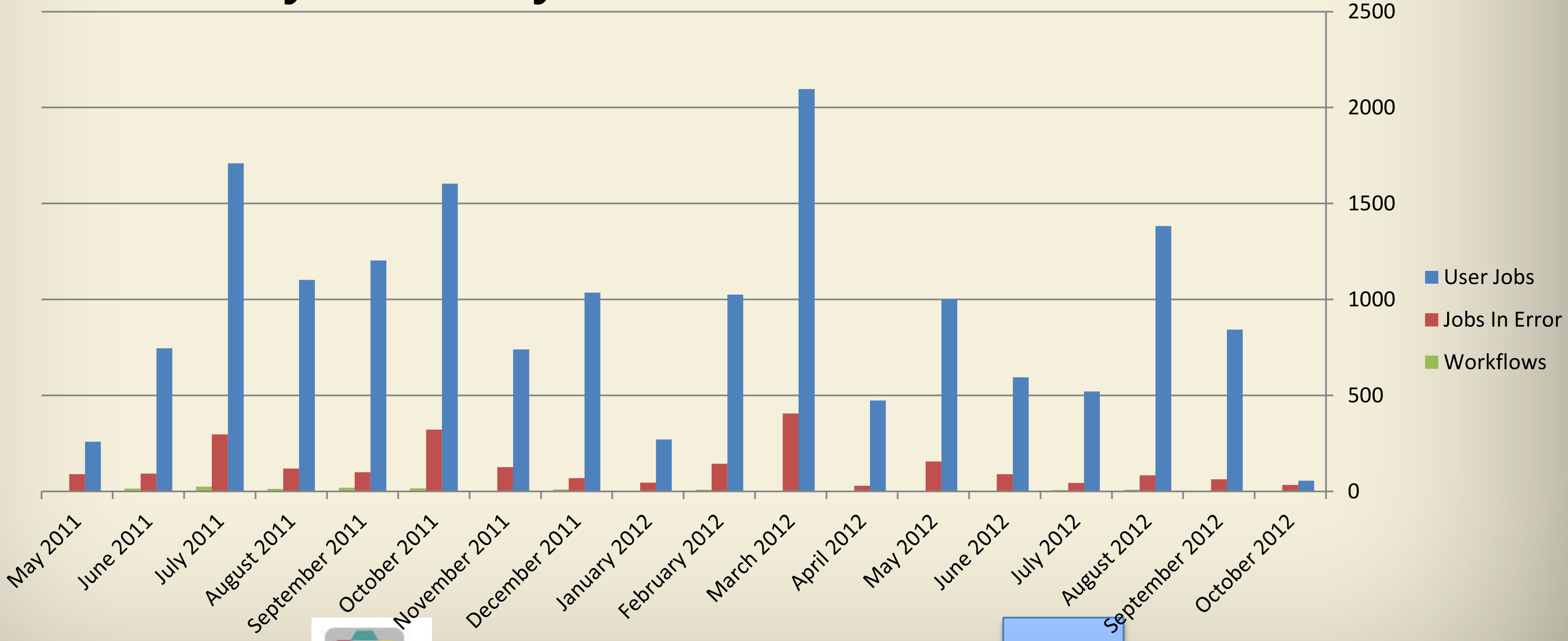
Group	FTEs
CCTS Informatics	3 Research Assoc. (%)
Sequencing Core	1 Research Faculty (%)
Research Computing	30% admin/analyst

- Model
  - Free to any university account (BlazerID)
  - no disk or CPU limits

UAB
Research
Computing

Timeline

- 2010 November – start reading papers
- 2011 Feb – Galaxy on a single VM
- 2011 May – Galaxy on the Cluster

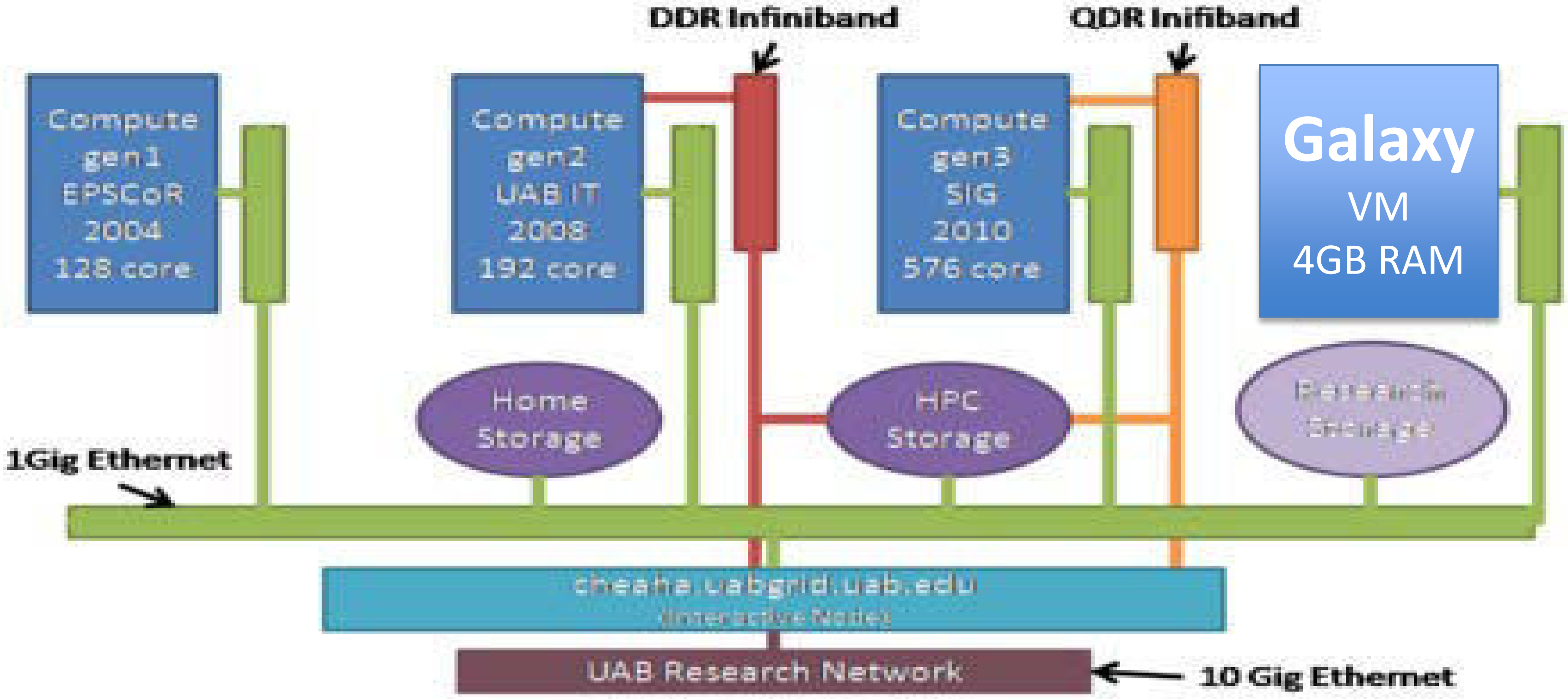


CCTS

UAB THE UNIVERSITY OF ALABAMA AT BIRMINGHAM

UAB Research Computing

Infrastructure Diagram



CCTS

THE UNIVERSITY OF ALABAMA AT BIRMINGHAM

UAB
Research
Computing

Deployment

- Git powered version control
- Change control: manual, git-hooks, fabric, Hudson etc…
- Open for collaboration on this front to come up with a good workflow

Version control

- UAB customizations are maintained in git
- Mercurial (hg) used only for pulling changes from galaxy-dist in a git branch (upstream-tracker)
- Upstream-tracker branch is merged with UAB ‘develop’ branch
- Planning to use hg-git tool in future

Customizations

- Added apache auth exceptions for published things (initial hack: now documented by galaxy folks)
- Removed mail domain from login string to match file system directories for data libraries and FTP import
- Showed dataset's file system path (initial hack: before expose dataset path was introduced in Galaxy)
- Configured FTP upload using SCP and ACLs

FTP setup

- Custom FTP upload using SCP and ACLs since the beginning (May 2011)
- FTP upload directories created automatically (within 30 mins of first login) if the user has a cluster account
- Galaxy login's email address is stripped out to return only username and it's matched with directory name
- Users need to ensure that galaxy can read-write their files (we take care of it using default ACLs, but permissions can be funny at times)

Successes

- 105 registered users
  - Though only 15 users over 100 Gb
- Used by the Informatics Group
  - Conduct our own analyses
  - Provide results, with protocols to researchers
  - Gave workshops, talks and classes on campus
  - Uptake by several labs to do their own work

Disappointments

- Informatics : Often easier to work outside Galaxy
  - Missing output files
  - Missing parameters
  - Older versions of tools
  - Missing tools
  - Hard to work inside and outside of galaxy
  - Green Screen of Death/Corruption during transfer
- Storage fabric crash October ’12!
  - Just because you told users the data is not backed up, does **not** mean they heard you.
  - Re-building reference data

Issues – DRMAA/shared cluster

- Cluster runs SGE, single queue
  - 12 core/48G ram node can be split between several jobs
  - All jobs expected to be “responsible”
- Per-tool hard-coded DRMAA
  - DRMAA string in universe_wsgi.ini
  - But thread/memory allocation hard-coded into .py files
  - Usually hard-coded to use 100% of memory
  - Usually hard-coded for some fixed thread count
- Interested in trying new parameterization
  - Only useful if tool wrapper is coded for that!

Issues – garbage in, garbage out

- Per-tool hard-coded DRMAA
- Green Screen of Death
- File upload and download woes
- De-compress failures
  - Run on head node (clarify: not cluster head node, but Galaxy app node)
  - No checksum/md5 checking – leads to garbage in/garbage out w/o any alert!
- SAM indexing running on the head node (clarify: not cluster head node, but Galaxy app node)

CCTS THE UNIVERSITY OF ALABAMA AT BIRMINGHAM UAB Research Computing

File uploads/downloads

- File upload method does more than upload – e.g during FTP upload of non-binary data file (sam), Galaxy processes new line characters. So data staged and imported in Galaxy won’t be same.
- How about upload doing only the file upload and then having a separate tool for file sanitization/conversion??
- Need uniform/consistent file download methods for all datatypes or at least documentation for each datatype. Existing approach varies according to file/datatype, so it’s difficult to debug or customize

CCTS THE UNIVERSITY OF ALABAMA AT BIRMINGHAM UAB Research Computing

Status and Successes

- Power-users going to command-line
  - Want latest versions of program
  - Want missing output files/options
  - DRMAA nightmares
  - Inter-step queue times
  - GATK
  - Velvet/Abyss

Wish List

- Universal job resource request interface
- Reference dataset handling automation
- More docs and capabilities in tool definition file logic
- Auto-download for NCBI blast & taxonomy (prototyped)
- Automation/simplification of indexes
- Index user-provided genomes
- Have many versions of a program available at once.