Software and Release

Tim Cartwright · Tim Theisen

Open Science Grid

Mission

- Integrate, test, and distribute DHTC software
  - To connect local sites to OSG
  - To give researchers access to OSG resources
- Goal: Minimize effort needed to manage the software that underlies DHTC (esp. OSG)
- Why is this hard?
  - Some contributed software is hard to build right
  - Component integration is fragile across updates
  - Configuration balance: local control vs. usability
  - Size and complexity of stack ⇒ constant updates

Open Science Grid

Main Activities

- Software Team
  - Build and package software components
  - Integrate components
  - Maintain some software tools
  - Automate testing of integrated software
- Release Team
  - Test integrated software installations
  - Release production-ready software
- Support software installations
- Maintain site admin documentation

Open Science Grid

Example: HTCondor

- OSG adds 8 patches, e.g.:
  - Patch start/stop script to get OSG security values
  - Ensure proxies are ≥1024 bits (contributed back)
- Integrated with other packages, e.g.:
  - Globus GRAM gatekeeper as batch system
  - GlideinWMS pilot jobs and central manager
- Automated tests include:
  - "Regular" HTCondor job
  - HTCondor-G job -> GRAM -> HTCondor backend
- We contributed unified source RPM to CHTC

Open Science Grid

Stakeholders

- Owners and administrators of OSG sites
  - ATLAS, CMS sites; national labs; other institutions
- Scientists, researchers, and other end users
  - Directly; indirectly via site, campus, portal, etc.

- OSG User Support and Operations
- OSG Security (software security & updates)
- OSG Network Monitoring (perfSONAR)

Open Science Grid

Stakeholder Benefits

- Build and test software so they don't have to
  - Without OSG software stack, we estimate a very competent site admin could build and configure one type of system in a few days to a week; one site usually has 3–5 different types of systems
  - Testing integration adds further time, complexity
  - Smaller sites might not exist, for lack of time
- Ultimately, lower administrative costs and greater local flexibility should result in more resources available to end users

Software/Release Teams

Name	Institution	Role	FTE
Tim Cartwright	UW–Madison	Software & Education Manager	1.0
Tim Theisen	UW–Madison	Release Manager	0.5
Brian Lin	UW–Madison	Software, Release	1.0
Carl Edquist	UW–Madison	Software	1.0
Igor Sfiligoi	UCSD	Software (Characterization)	0.2
Mátyás Selmeci	UW–Madison	Software	1.0
Edgar Fajardo	UCSD	Software	1.0
Suchandra Thapa	Chicago	Release	0.5
Xin Zhao	BNL	Release	0.4

6.6 FTE Total: 4.5 Software, 1.9 Release, 0.2 Education

Pending: 1.0 FTE Software at Nebraska

Open Science Grid

OSG Software Stack

- Builds on OS, supports user applications
- Packages
  - 161: start of Year 1
  - 233: current in OSG 3.1
  - 199: current in OSG 3.2
- Lines of code
  - 14,600,000 LOC in all source packages
  - 50,000 LOC under direct OSG control
- EPEL: use ~15 packages, rebuild ~65

Software Releases

	Q1	Q2	Q3	Q4
Year 1	5	3	3	4
Year 2	4	4 / 1	4 / 5	—

- Now on a predictable monthly schedule
- Extra releases for security or critical updates
  - Jun 2013: CA certificates (5 days)
  - Dec 2013: React to OS changes (9 days)
  - Feb 2014: Critical OSG 3.2 update (3 days)
- Tickets closed: 423 last year, 365+ this year

Open Science Grid

Software Support

- Provide support for installations
  - Via Grid Operations Center (GOC) – 15–20/month
  - Direct email
  - Online documentation

    https://twiki.opensciencegrid.org/bin/view/Documentation/Release3/
- How we help
  - Advise on installation and configuration
  - Debug software and integration failures
  - Improve packages, patch software, report bugs
  - Improve documentation and send email updates

Open Science Grid

Major Transitions

- OSG 3.1: Completed RPM transition
  All platforms and software; eliminated critical bugs; most sites converted
- OSG 3.2: Support for multiple release series
  New series is for large technology changes; can remove unused software
- Client installations anywhere by anyone
  Not an RPM strength, but needed by site stakeholders
- SHA-2: Support for better security algorithm
  Thoroughly tested all software; updated several components; ready early
- Java: Move from Oracle JDK 6 to OpenJDK 7
  Affected major components & supporting software; needed for security

Open Science Grid

Automated Testing

- Write and maintain automated tests in VMs
  - Install integrated software packages from OSG
  - Configure components as a site admin would
  - Test basic functionality, esp. across components
- Testing improvements
  - Added >100 new test cases; e.g., Gratia, GUMS, RSV
  - Improved test expressiveness, reliability, reports
- Switched to opportunistic VM resources
  - **Old:** 36 tests overnight – **Now:** 324 in a few hours
  - Good coverage of pre-release & release scenarios

Open Science Grid

Testing Benefits

- We find defects before users
  - Java packaging changes in OS
  - Java change to RSA key bit-depth
  - OpenSSL packaging change
  - BeStMan failures
- Increased confidence in releases
- Better manual testing: Focus on test cases that are new, exploratory (e.g., new error messages), or are hard to capture in code

Open Science Grid

Recent Findings

- As the OSG software, sites, administrators, and users have matured, the pace of large changes has slowed; yet there remains fairly constant pressure to update software, fix issue, and stay on top of security concerns.
- Separation of Software and Release improves focus of both teams and yields a more stable and predictable release process.
- Software issues are often also Operations and Security issues; frequent and open interaction helps us all meet stakeholder needs.

Open Science Grid

Upcoming Challenges I

- Keep providing timely updates, fixes, support

  Steady and evolving backlog of ~10 support tickets, ~150 work tickets

- Support RHEL 7 and deprecate RHEL 5

  New OS brings new technologies, challenges; old OS can block progress

- Donate common packages

  Make OSG "less special" by making basic software more widely available

Open Science Grid

Upcoming Challenges II

- Transition to major new technologies

  Example: Support migration to HTCondor CE, for all batch systems

- Improve documentation

  Update existing docs, add new ones, reorganize, cross-link better, etc.

- Expand testing coverage and expressiveness

  Tie together supported use cases, documentation, tests

Open Science Grid

A Notable Risk

- Software team inherits abandoned software
  - Original developers are gone, unfunded, busy, …
  - BeStMan, Gratia probes (some), GUMS, RSV, OSG Display, OSG Info Services, OSG PKI Tools
- Once inherited, very hard to find new owner
- Some inherited software is very complex; Software team provides only critical bug fixes, not routine updates
- Solutions?