

OSG Planning Retreat 2016

# **Software**

Tim Cartwright

10 May 2016

# Staffing

---

UW—Madison	Tim Cartwright	0.75	+ 0.15 OSG User School
	Mat Selmecci	1.00	
	Carl Edquist	1.00	
	Brian Lin	0.50	+ 0.5 Release
UCSD	Edgar Fajardo	0.50	+ 0.5 Investigations
	Jeff Dost	0.50	0.9 total OSG
	Marty Kandes	0.25	1.0 total OSG
UNL	Derek Weitzel	0.25	+ 0.75 Investigations
	<b>TOTAL</b>	<b>4.75</b>	+ UCSD

---

# Reviewing Year 4

# Year 4 Planned Work I

- **OSG 3.3**

- Shipped OSG 3.3.0 in August
- Added EL 7
- Dropped many stale packages
- No other huge differences

- **Enterprise Linux (EL) 7**

- Added support for most packages in 3.3.0
- Java changes are painful
- HDFS and BeStMan are done (just)
- Still working on GUMS, not on VOMS Admin

# Year 4 Planned Work II

- **HTCondor-CE**
  - Steady but low effort (~0.25 FTE?)
    - Better LSF, PBS, and SGE support
    - Bug fixes (incl. related software like LCMAPS)
    - Integrated CE-View and HTCondor-CE-Bosco
  - Consolidated usage info to track OSG CE hosts
- **StashCache** Finished our small role long ago
- **AutoPyFactory** Shipped
- **PanDA** No activity from ATLAS

# Year 4 Planned Work III

- **Routine updates**
  - Plenty, but seemed slower than past years
  - Tracking upstream changes well
- **Documentation**
  - Removed some old, useless documents (~10%)
  - Updated some commonly used documents
- **Support** as usual, plus ~5 CE contacts per week
- **Tests** Better reports, baby steps on capabilities
- **Internal** Moved sources (but not RPMs) to GitHub

# Year 4 Bonus Work

- Mostly small-scale tasks, e.g.:
  - GUMS - security update in admin web interface
  - VOMS Admin - restored AUP signing feature
  - APEL reporting updates
  - RSV - JSON consumer
- Tightened software processes a bit, which seemed to reduce little mistakes in the release process

# Planning Year 5

# Routine Work

- Routine updates to software packages
- Bug fixes to software and packaging
- Routine documentation updates
- Support, especially new CE support
- Scalability testing
  
- Important part of what we do
- No major obstacles
- Expect roughly equal effort in all years

# GRAM Sunset

- **Today!** OSG-CE installs do not include GRAM
- **Nov 2016** End OSG support for GRAM CEs
- Milestones to schedule
  1. OSG factories stop sending pilots to GRAM sites
  2. Drop GRAM packages from OSG software stack
  3. Remove GRAM code from OSG software

# HTCondor-CE

- Continue with long tail of improvements and fixes
- Monitoring changes as discussed yesterday (refactor CE-View, new metrics, forward by default)
- Blahp (CE-batch glue) unification, OSG/HTCondor
- Site support, especially for new or migrating sites
- Documentation tweaks

# SELinux

- *Security-enhanced Linux* – limits access to system resources (mostly files and the network) by requiring explicit access control policies
- Until now, OSG policy has been to disable SELinux
- But in EL 7, SELinux is enabled by default, and sites are asking for built-in support
- In Year 5, we plan to support SELinux for OSG SW
- Will get help from community (e.g., Oklahoma)
- May need help from upstream developers

# OSG 3.4

- Probably in first half of 2017
- Changes:
  - Drop Globus GRAM packages
  - Update to HDFS 3.x (for ECC, yet to be shipped)
  - Drop GIP and OSG Info Services
  - Drop BeStMan ... but how do we get there?
  - Drop other stale packages

# Miscellaneous Plans

- **OSG 3.2 / EL 5** Support ends in August
- **GRÅCC** Support as needed
- **Documentation**
  - Continue updating content and aligning tests
  - Deal with fallout from doc project (Rob Q.)
- **Testing**
  - Continue working on expressiveness – aiming for multiple configure-test-cleanup cycles
  - Investigate multi-host tests on CHTC

# Concerns

# Top Concerns

- **Ongoing planning within Technology area**
  - Hard to schedule quarterly reviews of plans
  - But it helps to get bigger projects done on time
- **Big software lifecycle transitions**
  - Prototype-to-production – getting better!
  - Prod.-to-EOL – how to make vision a reality
- **Java software**
  - Modernizing dependencies helps, but is slow
  - Removing unsupported Java SW is best!