

How should the VDT evolve for OSG 0.8.0? (Summer 2007)

Below is a list of possible goals that the VDT team brainstormed. A few are in progress. What do you think we should do?

At the March 2007 OSG All-Hands Consortium Meeting, we are gathering input. Decisions about what must be done will not be made here, but your input will go to the OSG Executive Team, along with input from all the stakeholders.

Tasks that are in progress

- Hire new VDT Team member
- Support VDT 1.6.1 for OSG 0.6.0
- Support AIX (Requested by NERSC)
- Add VOMRS to VDT 1.6.1 (as unsupported package)
- Fix Pacman/VDT updating problems
- Add Condor NFS Lite jobmanager
- Support TeraGrid
- Support EGEE

Tasks that could be done (not started)

These are in no particular order. Very rough time estimates are provided, just to give a feeling for how much work a task is. I don't think we can do all of this by June.

Privilege Separation (1 week work?)

Allow installation as non-root, then convert to root with well-understood process
(Requested by NERSC)

Support Mac OS X (2 weeks work?)

Probably start with Client, move to CE later

Add Globus' Virtual Workspaces (a couple days of work)

This would be to support edge services.

Support RPMS (debs?) as installation method (A few weeks of work)

This would be in addition to Pacman, not instead of. The goal is to integrate well with existing methods of installation, and methods of administering systems

Support ROCKS 4 (A few days to a couple of weeks, depending on scope)

This could be done minimally (i.e., it would work with Pacman) or as a ROCKS roll.

Support new OSes

RedHat Enterprise Linux 5, Fedora Core 6, 7, 8, Debian 4...

Upgrade to new versions of software

Keep up with software upgrades. Upgrade anything we need to. Squid, MySQL, Globus, Condor, and a few dozen more.

Daily VDT Email (A few days work, depending on scope)

Send email to sys admin about possible problems in VDT (expired host certificates, bad CRLs, services down, ...) There are infinite possibilities, but doing a few things would help nip many problems in the bud.

New GIPs/GLUE (A couple hours of work)

When new Generic Information Providers are released, update to them. Move to GLUE 1.3?

Better CRL updating (A week of work?)

The current CA CRL updating process is fraught with problems. It doesn't scale well when done per worker node. If it fails, it will be 24 hours before it tries again, which might prevent users from logging in due to an expired CRL.

Update CA Certs automatically (A week of work?)

The CA certificates are often not updated in a timely fashion by sysadmins, and this can cause problems for submitters: authentication may fail, preventing valid accesses. If they are updated automatically, we avoid a failure case. But we need to be very careful to do it securely.

Nicer VDT web site (A week of work?)

The current VDT web site seems to be hard to navigate, and it's not very pretty. Should we spend time improving it?

Update Globus

Update Globus to 4.0.4, 4.0.5, or even 4.2 (when released)

Preserve configuration for more parts of the VDT (A week of work?)

When doing a fresh installation, the current Pacman installation can copy configuration from the old installation. This is nifty, but it works for a subset of the VDT. We should add VOMRS and Gratia, and perhaps other bits as well (GridFTP? GSI OpenSS? Others?)

Support Sun Grid Engine for web-services GRAM (A couple days of work)

Currently the VDT only supports GRAM 2 for SGE

Switch from MySQL to MySQL Max

Some folks have asked for this.

Upgrade MonaLisa to the latest version (A couple hours of work)

We regularly update MonaLisa.

Add glxec**Add Grid Exerciser to the VDT (A day of work)**

Would people like the Grid Exerciser in the VDT, so VOs can use it to do their own verification of sites?

Add glite client tools for interoperability (A day to a week, depending on scope)

These are for the worker node, and would help CMS and ATLAS. Question: do we re-use glite builds to keep it simple, or do we build it ourselves? If we don't rebuild it, it won't work on as many platforms, but rebuilding it may be a huge amount of work.

Add dCache to the VDT as a Pacman package (A week or two of work?)

dCache is in the VDT as RPMs. It's a separate install. For now, that works pretty well. Should it also be in the VDT as a Pacman package?

Update to new DRM (Java version) (A few days of work?)

I understand a new version of DRM will be released soon, we'll probably want to update to it.

More tests (Days or weeks, depending on scope)

The test coverage in the VDT nightly test suite is not very broad. The more coverage we have in the VDT nightly test suite, the better our releases will be. Of course, adding new tests takes time, and input from component developers.

Better test infrastructure (1-4 weeks, depend on scope)

Improve our testing framework. Our testing framework is showing strain and really needs improvement. It's going to cause us problems soon. We can do minor changes and see some improvement, or we can do much deeper (and harder) changes. Should we switch to the NMI Build & Test environment for doing tests? This might take one to four weeks, depending on the scope of the work.

Improve VDT infrastructure (1 week, or so)

Improve our VDT infrastructure: we administer our machines ourselves with a goofy system. We want to move to something standard (cfengine?) and maybe pass off some of the administration to our local system administrators (nonVDT folks). A week?

Fix built in rpath in Globus (1 day)

There is a minor problem in our Globus builds that the rpath (roughly speaking, a built-in path to look for shared libraries) is hardcoded to point at /home/condor/<something> . This causes problems, particularly for folks using automounted NFS.

Logging (A week of work?)

Improve logging for security purposes. Figure out how long to retain logs, how to tell people to use them when there is a security incident, and maybe provide simple tools to mine the logs. I want to be ready for future security incidents!

Support SELinux (Unknown about of work)

Allow the VDT to install when SELinux is enabled. Lots of people don't want to turn it off, it breaks the VDT, and we don't know how to best fix it.

Ports (A few days?)

Provide people with better information about ports in the VDT: What ports are used? How can they configure iptables or tcpwrappers to allow them? Can we detect some cases when a firewall blocks access? I'm thinking of creating a file during installation which documents the ports in use for that particular installation, along with iptables configuration they can set for those ports. Sort of like the existing post-install/README file, but specific to ports.

Enable external collaboration (Perhaps a week?)

Enable external collaboration: Right now, people that want to help with the VDT have to have an account on our system. Can we distribute our development? Perhaps we switch from CVS to subversion, allow external access to the repository, and change our package creation mechanisms to allow it to work elsewhere.

Look for potential security problems that can be found with security scans

Security scans are known to cause problems with some software in the VDT, sometimes crashing components. Don has proposed the VDT team looks for such problems and works with component developers to encourage fixes.

SRMv2.2 (with dCache)

Get new dCache and SRM v2.2.

Change Nothing!

Perhaps you think the VDT is perfect, and want to limit upgrades for anything that isn't a security problem. Leave well enough alone.