



## Globus Distribution for Testbed 1

Andrew McNab, University of Manchester

[m.cnab@hepm.cam.ac.uk](mailto:m.cnab@hepm.cam.ac.uk)



## Summary

- ◆ Scope of this talk
- ◆ Modular Globus packaging => RPM packaging
- ◆ Structure of packages
- ◆ Practical differences 1.1.3 vs 2.0
- ◆ Security files in Globus 2.0
- ◆ LDAP based services
- ◆ GASS : introduction of GridFTP
- ◆ Developers' view
- ◆ The Bottom Line



## Scope of this talk

- ◆ Concentrate on the RPM based distribution of Globus 2.0alpha which we've built for Testbed 1.
  - This is currently only built for RedHat Linux 6.2 - although most of the comments about Globus 2.0 are general.
  - We're very interested in helping people to port the distribution to other Unix platforms.
- ◆ Concentrate on features of Globus which are used by EDG middleware products for Testbed 1.
- ◆ But avoid discussion of internals of Globus 2.0 except where they are relevant to planning / installing / configuring a Testbed site.
- ◆ (See [www.globus.org](http://www.globus.org) and the Globus mailing lists for this kind of detail.)



## Packaging / Modularisation

- ◆ Globus Project has always aimed to produce modular code.
- ◆ However, up to 1.1.3 this has not been reflected in packaging.
- ◆ With 2.0, they now aim to allow modules to be built and installed independently of unrelated modules.
- ◆ To ensure dependencies are satisfied, they now provide metadata for each Globus package which specifies a filelist, and lists of compile, link and runtime / shared library dependencies.
- ◆ In the Testbed 1 distribution, we've used this Globus metadata to automatically generate RPM SPEC files, with the original build and install dependency structures.
- ◆ This will also allow us to produce custom subsets of packages.



## Structure of packages

- ◆ Globus divide their packages into hdr (~ h), dev (~ a), rtl (~ so), pgm (executables) and doc (text) subpackages.
- ◆ We make 1 RPM for each subpackage with names like:
  - `globus_ssl_utils-gcc32dbgpthr_dev-2.0-12.i386.rpm`
- ◆ We here:
  - Package is `globus_ssl_utils`
  - Flavor is gcc compiler, 32 bit, debugging on, pthreads enabled.
  - Subpackage is dev - ie the static library(s)
  - Source tarball version is 2.0 - this is Globus's version number
  - RPM release is 12 - this is our number, may reflect patches we've applied
  - i386 is the binary platform (they've been built on a PIII box in fact)



## Practical differences 1.1.3 vs 2.0

- ◆ Globus 2.0 has no need for a deploy stage: just install the binary RPM's.
- ◆ Globus 2.0 has separate setup scripts for some packages. However, these **should not be used for Testbed1** since we supply the same functionality with `globus_*_edgconfig` RPM's and the `/etc/globus.conf` file.
- ◆ No need to create a globus user or group.
- ◆ `$GLOBUS_LOCATION` rather than `$GLOBUS_INSTALL_PATH`
- ◆ Everything goes in `/opt/globus` rather than `/opt/globus`, `/opt/globus-install` or `/usr/local/globus`, `/opt/SSLeay`, `/opt/openldap`
- ◆ Globus 2.0 relies heavily on its own shared libraries – it's simplest to add `/opt/globus/lib` to `/etc/ld.so.conf`



## Security files in 2.0

- ◆ In 1.1.3 some associated packages (eg gsi-ssh and gsincftp) had started to use /etc/grid-security for host certificates, CA details, grid-mapfile etc.
- ◆ In 2.0 there is some flexibility about where these files reside (/opt/globus/etc is still an option or default in some cases.)
- ◆ For the Testbed1 Distribution, we've chosen to concentrate the working copies of all these files in /etc/grid-security
  - This directory is unaffected by removing the /opt/globus tree.
  - Many sysadmins save /etc directories when they want to preserve / backup a machine configuration.
- ◆ Details of each trusted CA are now in separate files in /etc/grid-security/certificates as hashcode and policy files. This replaces the single ca-signing-policy.conf file of 1.1.3.



## LDAP Based Services

### ◆ MDS 2.1

- Major change is support for LDAPv3
- This permits GSI authentication / encryption.
- Globus 2.0alpha only provides this for GIIS though - GSI GRIS available in Globus 2.0 release.
- GRIS and GIIS now both use same port (2167)

### ◆ Globus Replica Catalog

- Now included as a standard part of Globus
- Publishes information about file locations via LDAP
- Used by GDMP



## GASS : introduction of GridFTP

- ◆ GridFTP added to existing GASS framework (ie parallel streams etc.)
- ◆ This means GASS now supports http, https, GSI ftp and GridFTP.
- ◆ `globus-url-copy` provided as general command line tool for file transfers
  - (`gsincftp` - which supports GridFTP - also supplied.)
- ◆ `gsi_wuftpd` provided with GridFTP support and using port 2811
  - If you want to give normal ftp-like access to a filesystem using grid tools, this is the server to use.
  - Can be run from (x)inetd or standalone - in Testbed 1 distribution, we run it standalone since this simplifies postinstall issues.



## Developers' view

- ◆ All of the RPM's are provided in source ('src rpm') as well as binary form at.
  - This means you can easily install them on a development machine, add a patch and then rebuild binary RPM's for your own or your group's use.
  - If you install the RPM system itself on a non-Linux platform, it should be relatively straightforward to rebuild binary RPM's for that platform.
- ◆ Dependency symbols for each subpackage are exported into the local machine's RPM database.
  - This means middleware authors can specify dependencies on pieces of Globus (the static libraries from globus\_ssl\_utils for instance) if you package your software in RPM format.



## The Bottom Line

- ◆ The binary and source RPM's are available from the W\_P6 repository on `datagrid.in2p3.fr`
- ◆ Ideally, you can install Globus 2.0 on a machine with a couple of commands:
  - `wget -nd -r --no-parent`  
`http://datagrid.in2p3.fr/distribution/globus/current/RPM_S`
  - `rpm -iglobus_* i386 rpm`
- ◆ (However, we will provide tailored subsets for client / developer / farm node contexts.)
- ◆ The new, modular approach to configuration files means we only need drop-in, static config files and a minimal amount of automatic setup when daemons startup – see Anders' talk for details.