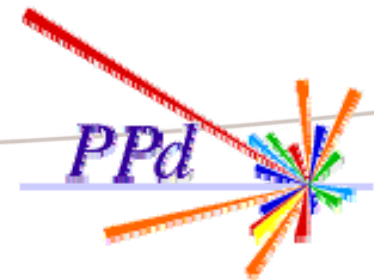




Data Management

Stephen Burke, PPARC/RAL

Jeff Templon, NIKHEF





Overview



- ◆ New features of GDMP 3
 - ◆ Using GDMP
 - ◆ Some problems
 - ◆ Directing jobs to data
 - ◆ Replica Manager
 - ◆ Mass storage
- } Jeff Templon
- } Stephen Burke



GDMP changes from v2



- ◆ GDMP now has a client-server architecture: client commands on a UI or WN talk to a server on an SE, which can in turn talk to servers on other SEs.
- ◆ No need for a gatekeeper on an SE to issue GDMP commands, but the gatekeeper stays anyway. Has proved very useful (see problems).
- ◆ No need to transmit a proxy with a job.
- ◆ GDMP has its own grid map file for each VO, which must contain both users and other SEs.



GDMP Replication Example (1)



- ◆ Put local file on RAL and use GDMP to replicate to CERN
- ◆ First must have correct GDMP config file defined
 - `export GDMP_CONFIG_FILE=/opt/edg/etc/alice/gdmp.conf`
- ◆ Get grid proxy, then start by checking communication to servers:
 - `gdmp_ping` – check if you can talk to local SE
 - Message: The local GDMP server tbn03.nikhef.nl:2000 is listening and you are an authorized user [Mon Sep 2 11:17:45 2002]
 - `gdmp_ping -S gppse05.gridpp.rl.ac.uk` - check if you can talk to remote GDMP server
- ◆ Tell CERN SE to “subscribe” to RAL’s content for Alice VO
 - `gdmp_host_subscribe -S lxshare0393.cern.ch -r gppse05.gridpp.rl.ac.uk`
 - Needs to be done first, so CERN receives notice of RAL’s new file



Replication Example (2)



- ◆ Copy a local file (from UI or WN) to RAL SE
 - Globus-url-copy <file:///home/templon/bigfile.dat>
gsiftp://gppse05.gridpp.rl.ac.uk/flatfiles/05/alice/hairball.45
- ◆ Register file locally at RAL to GDMP system
 - `gdmp_register_local_file -S gppse05.gridpp.rl.ac.uk -R -d /flatfiles/05/alice/`
- ◆ Publish RAL catalog to RC & Subscribed Servers
 - `gdmp_publish_catalogue -S gppse05.gridpp.rl.ac.uk -C`
 - The `-C` is currently quite important (should be default!)



Replication Example (3)



- ◆ Replicate file from RAL to CERN
 - `gdmp_replicate_get -C`
 - Note the `-C` again!
 - Gets ALL the files for your VO, not just yours



Basic GDMP use



- ◆ Copy a file to an SE, e.g. with `globus-url-copy` or `rfcop` or Unix `cp` to an NFS mount.
- ◆ `gdmp_register_local_file` to register a file, or all files under a specified directory, in the GDMP `local_file_catalogue`.
- ◆ `gdmp_host_subscribe` to subscribe one SE to know about the files on another SE - in EDG we normally want to subscribe all pairs of SEs (once and for all).
- ◆ `gdmp_publish_catalogue` to send file information to other SEs, and also write the file into the Replica Catalogue.



More GDMP use



- ◆ `gdmp_replicate_get/_put` to replicate files from one site to another and update the RC.
- ◆ `gdmp_remove_local_file` to delete a file from disk and the RC.
- ◆ `gdmp_ping` to check the connection and authorisation between SEs.
- ◆ `gdmp_get_catalogue` to re-acquire a file list from another SE.
- ◆ New user guide soon, including working examples.



Problems



Normal use of GDMP has no showstoppers, but some things can cause problems:

- ◆ We need to have SE host certificates in both main and VO grid map files. These are now read from the VO server at NIKHEF, but new SEs have to be entered and the map files updated.
- ◆ We need default GDMP and RC config files which are correct, and available to users in a standard way. GDMP config files OK for correctly configured site, but RC not. Not at all clear why we need two files, *see next slide*
- ◆ If things go wrong, error messages are often buried in a log file and can be hard to interpret.



Information is redundant



- ◆ GDMP_REP_CAT_HOST=ldap://grid-vo.nikhef.nl:10389
GDMP_REP_CAT_NAME=AliceReplicaCatalog
GDMP_REP_CAT_MANAGER_CN=Manager
GDMP_REP_CAT_MANAGER_PWD=Alice
GDMP_REP_CAT_CN=dc=eu-datagrid,dc=org
GDMP_REP_CAT_FILE_COLL_NAME=Alice WPI Repcat
- ◆ GDMP_REP_CAT_MANAGER_DN=cn=\${GDMP_REP_CAT_MANAGER_CN},rc=\${GDMP_REP_CAT_NAME},\${GDMP_REP_CAT_CN}
GDMP_REP_CAT_URL=\${GDMP_REP_CAT_HOST}/rc=\${GDMP_REP_CAT_NAME},\${GDMP_REP_CAT_CN}
GDMP_REP_CAT_FILE_COLL_URL=\${GDMP_REP_CAT_HOST}/lc=\${GDMP_REP_CAT_FILE_COLL_NAME},rc=\${GDMP_REP_CAT_NAME},\${GDMP_REP_CAT_CN}
GDMP_REP_CAT_OBJECTIVITY_COLL_URL=\${GDMP_REP_CAT_HOST}/lc=\${GDMP_REP_CAT_OBJFILE_COLL_NAME},rc=\${GDMP_REP_CAT_NAME},\${GDMP_REP_CAT_CN}
- ◆ RC_REP_CAT_MANAGER_DN=cn=Manager, rc=AliceReplicaCatalog,dc=eu-datagrid,dc=org
RC_REP_CAT_MANAGER_PWD=Alice
RC_REP_CAT_URL=ldap://grid-vo.nikhef.nl:10389/rc=AliceReplicaCatalog,dc=eu-datagrid,dc=org
RC_LOGICAL_COLLECTION=ldap://grid-vo.nikhef.nl:10389/lc=Alice WPI Repcat,rc=AliceReplicaCatalog,dc=eu-datagrid,dc=org



Problems (2)



- ◆ We are currently using plain-text passwords for the Replica Catalogues, which hackers could find fairly easily.
- ◆ You need to use the `-C` option with `publish_catalogue` to use the password - if you don't the publication fails but you can't do it again because `GDMP` thinks it worked.
- ◆ `register_local_file` and `replicate_get` are asynchronous: the command returns immediately, so it's hard to know if/when it succeeds (or if you need to do it again!)
- ◆ `remove_local_file` can only delete one file at a time, and is rather sensitive to file permissions etc.
- ◆ If `GDMP`'s "registry" gets confused, things quit working!!



Problems (3)



- ◆ There is no access control and the default for replication is "replicate everything", hence users can replicate large numbers of files by accident.
- ◆ The user has to check that a replication won't run out of disk space.
- ◆ When replicating a batch of files, if one file fails subsequent files are not processed.
- ◆ File IDs used for single-file replication are in an unusual format.



Conclusions



- ◆ Initial comment: as a developer I am very pleased with how much GDMP team has been able to accomplish. However as a user, I am quite unhappy with the general reliability and robustness of the system.
- ◆ Everything works (not a small achievement!)
- ◆ Configuration is fragile
- ◆ Commands are not robust
- ◆ Simple errors can confuse subsystem to point that local manager has to intervene

And now, ladies and gentlemen, Stephen Burke!