
CERN

Administering DPM 1.9.0

Andrea Manzi

CERN

DPM Workshop 2016
LPNHE, Paris

23-24 November 2016

Outline

- Dmlite-shell: the DPM shell
- New commands: quotatoken*,find, getlfn
- drain*/replicate/replicamove commands
- Next steps

dmlite-shell

- Until v 0.7.x shell to administrate Dmlite
  - Admin can manage directories, files, replicas, pools, users, groups, ... (48 commands)
- From v 0.8.x -> DPM administration
  - FS and QuotaToken (ex SpaceToken) management
- Made for admins and developers, not for users
- https://twiki.cern.ch/twiki/bin/view/DPM/DpmAdminDmliteShell

dmlite-shell - what's new

Version 0.8.3 contains:

- New Commands
  - *quotatoken[add, get, del, mod]*
  - *getchecksum (* checksum command deprecated)
  - *find*
  - getlfn

Bug fixes:

- Missing checks on Drain commands, which in combination with a bug in LCGDM-DAV led to file loss!

Enhancements:

- added replica extended attributes to Info command output

Changes:

- Fs* and qryconf commands are now using Dome
- Drain/replicate commands use Dome or Dpm to set the FS as read-only according to the configuration

dmlite-shell – Quota Tokens

Starting with DPM 1.9.0 is it possible to define via dmlite-shell the Quota Tokens for a particular path of the namespace.

- Quotas will be enforced only when the Dome adapter is enabled

Sysadmins have to define the Quota Tokens via the dmlite-shell, this is possible in 2 ways:

- For namespace paths which are not already managed via a SRM spacetoken a new quotatoken needs to be created via the **quotatokenset** command
- For namespace paths managed via a SRM spacetoken, the existing spacetokens need to be 'converted' into Quota Tokens via the **quotatokenmod** command

dmlite-shell – Quota Tokens

- The quota information is now calculated from the folder size attribute associated to the token
- In order to initialize the folder sizes with the correct values the following script needs to be executed once:

dmlite-mysql-dirspaces.py

- To be converted to a shell command..

dmlite-mysql-dirspaces.py --nsconfig=<config> --updatedb

dmlite-shell – Quota Tokens

The **quotatokenset** comman, creates a Quota Token assigning it to a specific namespace path.

- quotatokenset <path> pool <value> desc <value> size <value> groups <value>

The command accepts the following parameter:

- <path> : the path
- pool <poolname> : the pool name associated to the token
- size <size> : the quota size and the corresponding unit of measure (kB, MB, GB, TB, PB), e.g. 2TB , 45GB
- desc <description> : a description of the token
- groups <groups> : a comma-separated list of the groups that have write access to this quotatoken

Example

quotatokenset /dpm.cern.ch/home/dteam pool pool01 size 10TB desc "testing token" group dteam

dmlite-shell – Quota Tokens

The **quotatokenget** command lists the Quota Tokens defined for a given path

- quotatokenget <path> [-s]

The command accepts the following parameters:

- <*path*> : the path
- *-s*: the command will print the quota token associated to the subfolders of the given path (optional)

in order to get all quotatoken defined on the system for instance, the following command can be executed:

quotatokenget / -s

dmlite-shell – Quota Tokens

The **quotatokenmod** command can be used to modify a particular Quota Token.

- *quotatokenmod* <token_id> path <value> pool <value> size <value> desc <value> groups <value>

The command accepts the following parameters:

- *token_id* : the token id
- *path* <path> : the namspace path
- *pool* <poolname> : the pool name associated to the token
- *size* <size> : the quota size and the corresponding unit of measure (kB, MB, GB, TB, PB), e.g. 2TB , 45GB
- *desc* <description> : a description of the token
- *groups* <groups> : a comma-separated list of the groups that have write access to this quotatoken

this command is needed in order to 'convert' an existing SRM spacetoken to a Quota Token.

dmlite-shell – Quota Tokens

The **quotatokendel** command removes an existing Quota Token from the system.

- quotatokendel <path> <pool>

The command accepts the following parameters:

- <path> : the path to remove the quota from
- <pool> : the pool associated to the quota token to remove

i.e.

quotatokendel /dpm/cern.ch/home/dteam pool01

dmlite-shell – GetChecksum

This **getchecksum** command can be used to Get a checksum of the specified type. The checkusm is retrieved from the namespace if available or calculated and the recalculation can be also forced

getchecksum <file> <checksumtype> [<forcerecalc>] [<rfn>]

The command accepts the following parameters:

- < checksumtype > : the type of checksums
- < forcerecalc > : if specified it forces the checksum recalculation
- <rfn> : if specified it uses a specific replica for the recalculation

N.B. Needs Dome + DomeAdapter for recalculation

dmlite-shell – Find

This new function can be used to find a file on the namespace based on substring given as input

- find <name> [<-d>]

The command accepts the following parameters:

- <name> : the substring to look for in the namespace
- -d : retrieves folders instead of files (default is file)

in order to retrieve all the namespace element you can use this options:

find “” or find ”” –d for folders

Examples (also not interactive commands):

- *dmlite-shell -e "find ” "* | *wc –l* -> number of namespace entries
- *dmlite-shell -e "find atlasdatadisk/rucio/mc_12_14TeV "* | *xargs -I {} dmlite-shell -e "acl {}"* ->
  get/change ACL for specific files in a folder
-

N.B. The command latency depends on the size of the DB

dmlite-shell – getlfn

This new function can be used to get the LFN (Namespace entry) from a SFN (disk+ diskpath)

- getlfn <sfn>

Retrieves the LFN associated to the given SFN

Example:

getlfn dpmdisk-trunk.cern.ch:/srv/dpm/volume/atlas/2016-02-01/group.test.hc.NTUP_SMWZ.root.116623.0

/dpm/cern.ch/home/atlas/group.test.hc.NTUP_SMWZ.root

dmlite-shell – drain

- The drain commands in dmlite-shell are making use of HTTP for I/O (while dpm-drain uses RFIO)
  - *drainpool, drainserver, drainfs*
  - Same options as the old drain implementation (plus dryrun)
  - Multithreading (up to 10)
  - Should fix issues with hung threads observed with old drain
  - Of course it requires LCGDM-DAV properly configured on all disknodes
  - **Major bug fixed during the year (needs LCGDM-DAV 0.17)**

dmlite-shell – replicate + replicamove

- *replicate* in dmlite-shell replaces the dpm-replicate client
  - Moved to HTTP as well
  - Allows user to replicate a file to a specified destination
  - Could be simply extended to allow replication between different pool types (e.g. FS to HDFS)
- *replicamove* allows *moving* specific replica folders to a specific server/FS
  - Similar to drain, but with the possibility to specify the destination
  - Equivalent to dpm-move-data in admin tools

Next Steps

- The legacy *dpm-listspaces* command uses DPM/DPNS to get the information about the spaces
- We should re-implement it using dmlite/DOME as a dmlite-shell command to run in a non-interactive way
  - *dmlite-shell -e “listspaces”*
- This command will be used as BDII info-provider as well

Next Steps

- Some commands available in the admin-tools package can be easily included in the shell
  - getlfn recently done
  - dpm-sql-spacetoken* are simply list of a namespace folders with DOME
- dpm-dbck can also be included
  - Alexandr Mikula tool to presented tomorrow as well

Summary

- dmlite-shell has now all commands for DPM administration
- The only available admin tool when moving out from DPM/DPNS
- Feedback from sysadmins is important so please report any issue or suggestion.

Questions?