
ATLAS data and the Tier3:

- access tools and methods for Tier3-based analyses
 - uploading and registering user datasets
-

US ATLAS Transparent Distributed Facility Workshop

March 4, 2008

RENCI, Chapel Hill, NC

Marco Mambelli
University of Chicago

Outline

- Storage of user data
- Moving data
 - Transfer (official and user) files
 - Setup, options, monitoring, performance
 - Request for space reservation
- Publishing data
 - publish a dataset
- Retrieving data
- Software installation

Storage of User Data

- Need for a scalable and reliable storage system for user data (25TB estimated)
- Evaluation of a variety of data storage systems is underway within ATLAS and externally (e.g. HEPiX Storage WG & OSG Storage WG)
- From the report of the ATLAS T3 Taskforce at the ATLAS S&C Workshop (2/29/08)

Storage System	Local Protocol	Load Balancing	Externally Secure	POSIX Access	Single Namespace	Installation Load	Maint Load	Quotas	Cost
NFS	bad	N	N	Y	N	low	high	Y	\$0
Lustre	Y	Y	w/SRM	Y	Y	medium	medium	Y	\$0
GPFS	Y	Y	w/SRM	Y	Y	high	medium	Y	\$\$\$
xrootd	Y	Y	w/SRM	mkdir/rmdir do nothing	Y	medium	low	partitions	\$0
DPM	Y	Y	Y	special commands	Y	medium-high	low- medium	partitions	\$0
dCache	Y	Y	Y	metadata	Y	high	low- medium	partitions	\$0

Type of Storage in OSG



User's Storage



-Desktop
-NFS



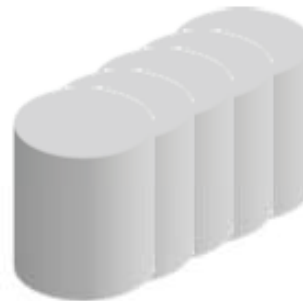
Worker Storage



-/tmp
-Software
-NFS



Storage Element



-Owned
-Leased
-Public



Repository



To stage data

From:

Opportunistic use of storage in OSG - Ted Hasselroth - OSG all hands meeting

Some definitions

- **GUID** Globally Unique Identifier: unique ID of each file and Dataset (DUID, VUID)
- **LFN** Logical File Name: descriptive name of a file unrelated to the path or the name on disk of any of the replicas (copies) of that file
- **PFN** Physical File Name: the URL indicating a single replica of a file (and how to access it).
 - **TURL** Transfer URL, URL to use to copy a file out of a SE
 - **SURL** Storage URL, URL used to refer to a file within a SE (e.g. for pinning, ...)
- **FC** File Catalog, an index to find files and associated metadata
 - **LRC** Local Replica Catalog, file catalog used in OSG (US-ATLAS)
 - **LFC** Local File Catalog, file catalog used in LCG
- **DS** Dataset: set of related files, recorded in DQ2 and AMI
- **DQ2** or **DDM**: ATLAS distributed data management system
 - DQ2 holds the ATLAS central Dataset Catalog
- **AMI** ATLAS Metadata Interface: browsable catalogue of ATLAS Datasets

Some more definitions

- Any file in a Grid Storage Element (SE) is **available** on the Grid. Any user with the URL of the file and the right permissions (valid grid certificate) can get that file using:
 - SRM
 - GridFTP
 - HTTP
 - dCache (with SRM and GridFTP doors)
- A file is **published** if there is an entry in a File Catalog (FC) about that file
- In ATLAS, files must be registered in the LRC (or another valid FC), added to a Dataset and the Dataset must be registered in the DQ2 central catalog
 - Files at Tier 3 centers are currently not visible in DQ2

The client tools

- DQ2 client
 - Allows to interact with dataset subscriptions and to query DQ2 central catalogs (dq2-XXX commands)
- DQ2 enduser tools
 - Includes dq2_XXX commands
- gLite UI (User Interface)
 - Includes lcg-cp, egee-gridftp-XXX
- OSG client (superset of VDT client):
 - Includes tools like voms-proxy-XXX, globus-url-copy, srmcp
- OSG WN Client (for completeness)
 - Subset of OSG Client, not recommended for interactive use (some required features are missing)

Client setup

- **Explicit setup (local or AFS installation):**
 - `$ source /somewhere/glite/etc/profile.d/grid_env.sh`
 - `$ source /somewhere/dq2/dq2.sh`
 - `$ grid-proxy-init`
- **Setup with OSG client**
 - **Easier setup**
 - **More support**
 - `$ source /share/osg-client/setup.sh`
 - `$ source /share/dq2/dq2.sh`
 - `$ grid-proxy-init`
- **Why setup gLite UI in OSG**
 - `dq2_put/get` may use some gLite commands depending on the site they interact with (TiersOfATLASCache.py description): `lcg-lg`, `lcg-rf`, `glite-gridftp-ls`, `lcg-gt`

Transfer files

- globus-url-copy

- Widely available, reliable, many copy options (recursion)
- No support for pinning or space reservation

- srmcp

- Supports request queuing, pinning and space reservation
- Some incompatibility due to different versions or firewall configuration, will improve moving to SRM2

- `srmcp -debug=true`

```
srm://head01.aglt2.org:8443/pnfs/aglt2.org/data/the-file-you-pick file:///tmp/somefile
```

- `srmcp --debug=true -srm protocol version=2`

```
srm://head01.aglt2.org:8443/srm/managerv2?SFN=
pnfs/aglt2.org/data/the-file-you-pick
file:///tmp/somefile
```

Publish datasets generated on a Tier3 to "ATLAS"

- Copy the files into the SE (e.g. MWT2_UC's dCache system)
- Choose a Dataset name
 - Each user dataset in ATLAS has to be named with the prefix `user.FirstnameFamilyname`, as reported in the grid certificate's DN
 - Names cannot be reused
- Register files in the LRC (file catalog)
- Create a dataset definition
- Publish that DS into the DQ2 central catalog
- `dq2_put` performs the previous 3 items as a single command line

Publish a dataset with dq2_put (notes)

- Registers datasets to DQ2 and in the local File Catalog
 - '-d' all files in a directory
 - '-p' all files in a XML Pool File Catalog
- `dq2_put -h` for help and examples
- Documentation:
 - https://twiki.cern.ch/twiki/bin/view/Atlas/UsingDQ2#DQ2_end_user_tools
 - <https://twiki.cern.ch/twiki/bin/view/Atlas/DDMEndUserTutorial>
- Remember to:
 - Define the local SE: `DQ2_LOCAL_ID` (e.g. `export DQ2_LOCAL_ID="MWT2_UC"`)
 - Initialize your Grid proxy
- **Do not use '-d' if you have root files or files with embedded GUID:** `dq2_put` (and `poolFCjob0`) would register the file with a new unrelated guid breaking Athena access (`uuidgen`)
 - `pool insertFileToCatalog filename` adds a file to PFC.xml extracting the embedded GUID

Publish a dataset with dq2_put (use cases)

■ Files are in a SE local to your host

- You are supposed to be able to list directories and run checksumming on the files in the SE (dCache!)
- `dq2_put -v -d /share/data/user/user.MarcoMambelli.regtest0001.local/user.MarcoMambelli.regtest0001.local`

■ Files are in a remote SE

- Files are copied locally (/tmp) to evaluate size and checksum
- `dq2_put -v -g uct2-dc1.uchicago.edu:2811 -d /pnfs/uchicago.edu/data/tier3/user/user.MarcoMambelli.regtest0002.se/user.MarcoMambelli.regtest0002.se`

■ Files are in a remote SE and on your host

- All files in one directory, you need to know paths
- Local evaluation, '-n' remote path in catalog
- `dq2_put -v -d /ecache/marco/local_regtest3/ -n gsiftp://uct2-dc1.uchicago.edu/pnfs/uchicago.edu/data/tier3/user/user.MarcoMambelli.regtest0003.both/user.MarcoMambelli.regtest0003.both`

■ Files are already in the LRC

- Without '-e' or '-f' all checksums are evaluated
- `dq2 put -v -f -p /ecache/marco/local rt4/PoolFileCatalog.xml -n gsiftp://uct2-dc1.uchicago.edu/pnfs/uchicago.edu/data/tier3/user/user.MarcoMambelli.regtest0004.fast/user.MarcoMambelli.regtest0004.fast`

Retrieving a dataset with dq2_get

- dq2_ls lists datasets

- Allows wildcards
- Provides information about the DS: files, replicas
- Important to estimate transfer size
- `dq2 ls user.MarcoMambelli.regtest*`

- dq2_get

- Retrieves Datasets
 - Output Datasets of Pathena jobs
 - Datasets for local processing
- `$ dq2_get -rv`
`csc11.005001.pythia_minbias.evgen.EVNT.v11000401`

More on using dq2_get

- `dq2_get -h` for help
 - `-r` retrieve the dataset from the grid
 - `-d` destination directory
 - `-p` parallel copy threads (do not overdo!)
- The environment can change the behavior
 - `DQ2_COPY_COMMAND` can select the copy command (e.g. `lcg-cp`)
- Documentation
 - https://twiki.cern.ch/twiki/bin/view/Atlas/UsingDQ2#dq2_get

Dataset subscriptions

- Not used directly (for most Tier3)
 - Requires DQ2 site services
- Indirect copy using a close Tier2
 - `dq2-register-subscription --archive -s SWT2 CPB fdr08_run1.0003079.MinBias.merge.AOD.o1_r12_t1 OU`
- Dataset browser
 - <http://gridui02.usatlas.bnl.gov:25880/server/pandamon/query?overview=dslist&redirect=pandamon>
- Monitoring of subscriptions
 - <http://dashb-atlas-data.cern.ch/dashboard/request.py/site>

How the client tools are used

- I surveyed the US-ATLAS users and here are the results
- DQ2 enduser tools
 - Used mainly by end users (scientists) to transfer files
- DQ2 client
 - Mainly used by site admins or production managers
- gLite UI
 - Invoked by DQ2 enduser tools, provides an x509 cert (grid proxy)
- OSG client
 - Used by end users, provides an x509 cert
- Must be setup (they are not in the default environment)
- It is possible to install them locally
- Several of these are available on AFS (from CERN or BNL) and some sites use AFS installation (simplest way to be in business)

Installation: OSG Client

- The OSG client is installed in `/share/osg-client` using the pacman installation (`pacman -get OSG:client`):
- Answer no to all the questions during configuration (except license agreement):
 - Do you agree to the licenses? [y/n] y
 - Would you like to enable the Condor batch system to run automatically? ... n
 - Would you like to setup daily rotation of VDT log files? ... n
 - Do you want to update the CA certification revocation lists (CRLs) automatically? [y/n] n
 - Where would you like to install CA files? ... n (no) - do not install ... n

Installation: gLite UI

- Part of the gLite middleware (released by EGEE). Here is a link on how to install gLite middleware:

<https://twiki.cern.ch/twiki/bin/view/LCG/TarUIInstall>

- Get the packages (`mkdir /share/glite/`, `wget ...`)
- Untar (`tar -zxvf gliteUI_*.tar.gz`)
- edit `./site-info.def` to match your local configuration
 - the installation page explains how
 - some values are LCG specific. I configured only `INSTALL_ROOT`
- `./glite/yaim/bin/yaim -c -s ./site-info.def -n TAR_UI` The command printed some errors (tolerable considering that `MWT2_UC` is not a LCG node)
- Check the content of the setup files in `/share/glite/etc/profile.d/`

- Some additional notes about installing gLite UI on a OSG site are available here:

http://gdsuf.phys.ufl.edu:8080/cmssoft/COCRAB/lcg_ui_installation_for_osg.html

Installation: DQ2 enduser tools

- Check frequently for updates: sometimes new releases are not formally announced
- May be installed in the same directory (`/share/dq2/`)
 - DQ2 enduser tools (`dq2_XXX` commands to get/put files)
 - DQ2 Client (`dq2-XXX` commands, to interact with DQ2 Site services and central server, e.g. register subscriptions)
- The user tools are downloaded from the CVS repository or from one of the AFS installation (usually are in sync, `/afs/usatlas.bnl.gov/Grid/Don-Quijote/dq2_user_client/`) as mentioned in <https://twiki.cern.ch/twiki/bin/view/Atlas/UsingDQ2#Download>
- The setup files are edited to accommodate local needs (`dq2.[c]sh`, `setup.[c]sh.any`)

Installation: DQ2 client

- Check frequently for updates: sometimes new releases of the client software are not formally announced
- The client is installed as recommended in <http://dashb-build.cern.ch/ddm/build/stable/doc/guides/dq2-clientapi/html/user/installSection.html#dq2ByTarballsSection> (To install the development version check <http://dashb-build.cern.ch/ddm/build/unstable/doc/guides/dq2-clientapi/html/user/installSection.html#dq2ByTarballsSection>)
 - `$ mkdir dq2/ 1$ cd dq2/`
 - `$ wget "http://atlas.web.cern.ch/Atlas/GROUPS/DATABASE/project/ddm/releases/dq2.download/downloadDQ2StableClient.py" -O downloadDQ2StableClient.py`
`wget "https://twiki.cern.ch/twiki/bin/viewfile/Atlas/DDMClientInstallation?filename=downloadDQ2Client.py.txt" -O downloadDQ2Client.py`
 - `$ python downloadDQ2Client.py`
- There is a twiki page with installation notes from the clients installation at MWT2_UC (pretty general anyway): <http://twiki.mwt2.org/bin/view/DataServices/InstallingDQ2AtUC>

Using clients on AFS (from BNL or CERN)

■ gLite UI

- `/afs/usatlas.bnl.gov/lcg/current/etc/profile.d/grid-env. [c] sh`
- `/afs/cern.ch/project/gd/LCG-share/current/external/etc/profile.d/grid-env. [c] sh`

■ DQ2 client

- `/afs/usatlas.bnl.gov/Grid/Don-Quijote/dq2_user_client/setup. [c, z] sh.any`
- `/afs/usatlas.bnl.gov/Grid/Don-Quijote/DQ2_0_3_client/dq2.sh`
- `/afs/cern.ch/atlas/offline/external/GRID/ddm/stable/dq2. [c] sh`

■ DQ2 endusertools

- `/afs/cern.ch/atlas/offline/external/GRID/ddm/endusers/setup. [c, z] sh.any`

Summary

- Useful tools to move data
- Tools are evolving
 - Work to integrate gLiteUI and OSG-Client
 - New capability and documentation added
- Check for new features and interact with developers
- Tier3 model is not defined as well
 - Data consumer
 - Provide data visible to ATLAS (DQ2 site services)
- Freedom, no well defined recipes

Where to get help

- Ddm-operation Twiki

- <https://twiki.cern.ch/twiki/bin/view/Atlas/DDMOperationsGroup>

- Mailing list and Hypernews

- usatlas-ddm-l@lists.bnl.gov
- <https://hypernews.cern.ch/HyperNews/Atlas/get/DDMOperationsGroup.html>

- Savannah portal DDM-operation (bug report):

- <https://savannah.cern.ch/projects/dq2-ddm-ops/>
- <https://savannah.cern.ch/projects/atlas-ddm/>