
DQ2 0.3 SiteServices

Patrick McGuigan
mcguigan@uta.edu

Outline

- SiteServices Installation
- Operation
- LRC Impacts
- Management

SiteServices Installation

- Installation Instructions

 - <http://dashb-build.cern.ch/ddm/build/stable/doc/guides/dq2-siteservices/html/admin/index.html>

 - OSG specific recommendations/additions shown later

- USATLAS TWiki page for documenting progress

 - <http://www.usatlas.bnl.gov/twiki/bin/view/Admins/DQ2SiteServicesP1>

- usatlas-ddm-l@lists.bnl.gov

- Pre-installation Checklist

 - Different machine from LRC

 - RHEL4/SL4/SLC4

 - MySQL 5.1

 - See section 2.6 for /etc/my.cnf

 - APT

Installation (cont.)

- As root
- `$ apt-get check`
- `$ apt-get -f remove`

Add gLite/DQ2 repositories to apt configuration (see instructions)

- `$ apt-get update`
- `$ apt-get install dq2-siteservices`
- `$ apt-get install fetch-crl`
- `$ apt-get install lcg-CA`

Configuration

- MySQL account for running DQ2

- OSG recommendation Step 2.6:

- Don't use `dq2user/dqpwd`

- Don't need:

```
GRANT ALL PRIVILEGES ON _dq2.* TO  
  '$DBUSER'@'%' IDENTIFIED BY '$DBPWD';
```

```
$ . /opt/dq2/profile.d/dq2_post_install.sh
```

Configuration

- OSG Recommendation:

- Remove world readable config files
- After running `dq2_post_install.sh` but before modifying config files do:

```
$ chown -R <u>.<g> /opt/dq2/etc
```

```
$ chmod -R o-r /opt/dq2/etc
```

Organization

- `/opt/dq2/`
 - `bin/` contains dq2 client commands
 - `etc/` contains configuration files
 - `profile.d/` contains source file
 - `doc/` html documentation
- `/opt/glite` FTS clients / `glite-gridftp-mkdir`
- `/opt/lcg` LFC client
- `/opt/globus` VDT 1.2.2 as released by gLite
- `/opt/dashboard` Dashboard services (not ARDA)
- `/usr/lib/python2.3/site-packages/dq2` Code

Site Customizations

- Login as <u>.<g>
- Provide contact string for _dq2 write account
/opt/dq2/etc/dq2-agents/dq2-agents.cfg
dbconnection=mysql://<acct>:<pw>@localhost/_dq2
- Provide contact string for LRC write account
/opt/dq2/etc/dq2-info/dq2-info.cfg
catalog_<SITE>=mysql://<l_acct>:<l_pw>@<l_host>:3306/localreplicas
- Provide MyProxy Password
/opt/dq2/etc/dq2-transfertool-fts/dq2-transfertool-fts.cfg
defaultMyProxyPassword=<myproxy_pw>
- Provide site name(s) to be served
/opt/dq2/etc/dq2-service-fetcher/dq2-service-fetcher.cfg
sites=UTA_TEST1,UTA_SWT2

Site Customizations (cont.)

- Provide override for storage location [OPTIONAL]

```
/opt/dq2/etc/dq2-info/dq2-info.cfg
```

```
srms_<SITE>=gsiftp://<gatekeeper>/different/path/
```

- Alter storage location by dataset pattern [OPTIONAL]

```
/opt/dq2/etc/dq2-agents-subscription/dq2-agents-subscription.cfg
```

```
storagePattern={'.*AOD.*':  
    '<protocol>://<host>/different/path/AOD/'}
```

This is a Python dictionary where regular expressions (re module) are keys and storage URL's are values

- Create script file `~/setup.sh` for `<user>` and source it

```
export X509_USER_PROXY="/tmp/x509up_u`id -u`"  
export MYPROXY_SERVER="myproxy.usatlas.bnl.gov"  
export http_proxy=some.host.name  
source /opt/dq2/profile.d/dq2_env.sh
```

Starting Service

- **FIRST TIME ONLY**

```
$ dq2site-recreate-database
```

- **Proxies (30days):**

```
$ grid-proxy-init -valid 720:00
```

```
$ myproxy-init -d -c 720
```

- **Normal operation:**

```
$ dashb-agent-start dq2udpserver
```

```
$ dashb-agent-start dq2agents
```

```
$ dashb-agent-start dq2fetcher
```

Stopping Services

```
$ dashb-agent-stop dq2fetcher  
$ dashb-agent-stop dq2agents  
$ dashb-agent-stop dq2udpserver
```

Catalogs Clients

- DQ2 0.2.12 used dq2 command with operation as parameter
- DQ2 0.3 uses separate commands:
 - Dq2-register-subscription
 - Dq2-delete-subscription
 - Dq2-list-dataset
 - See /opt/dq2/bin

Logging

- Logging is less specific than 0.2.X
- Default configuration puts log in /opt/dq2.log
- Override in */opt/dq2/etc/logging.cfg*
`logging.cfg:args=('/tmp/dq2.log','a',1000000000,5)`
- ARDA Monitoring is using test instance at:
<http://dashb-atlas-data-test.cern.ch/dashboard/request.py/site>

Outstanding Issues

- Allowing subscriber to specify share
- LFC client is generating spurious output to shell
- DQ2 logs contain warning about proxy
- Location of dashboard lock files
- Dashb-agent-stop sometimes fails
- Agents can take some time to stop
- PFN's might be renamed; leaf no longer LFN

Where we are today (6/22)

- BNL/PANDA, BU, AGLT2, UTA_SWT2, SLAC/XRD have latest code
- UTA_SWT2 in production, but ...
- Serious problem seems to exist with how DQ2 is handling Panda's sub datasets
- Can't resume AOD replication
- Much more testing is needed

LRC implications

- New LRC being developed
- Most locations using existing LRC's
 - LRC coupled to a DQ2 0.2.12 installation
 - Sites that rotate NFS storage need to change files on multiple machines
- Installing the current LRC, via pacman, needs to be fixed

Site Management

- DQ2 schedules data movement
- DQ2 does not manage the storage location
- Quotas don't provide mechanism for this
 - What if production system is at quota?
- How to decide what to delete?
- How to automate this so that production can stay below some cap?
- How to manage this for users/groups

cleanse.py

- Walks filesystem; takes leaf names as LFN
- Looks for LFN in local LRC and BNL LRC
- Respects archival flag of local LRC
- Deletes files that are replicated at BNL if downloaded more than 5 days ago

cleanse.py

- Drawbacks:
 - Slow: Not using bulk methods for LRC lookups
 - Deletes LRC first then physical copy
 - Not aggressive
 - Files not in local LRC are not deleted
 - Proposing 3 day limit
 - Files not in BNL LRC are not deleted
 - Proposing 3 week limit
 - Not particularly intelligent
 - Deletes as many files as possible
 - Doesn't order files

deleteDatasetLocal.py

- Deletes a named dataset from storage/LRC
- Issues:
 - ❑ Does not respect archival flag?
 - ❑ Needs to be updated for dq2 0.3

Aborted datasets

- How to delete these from Tier 2's
- Alexei maintains list of datasets that have been aborted
- Wensheng maintains list of guid's that could be at T2
- Several options:
 1. Use deleteDatasetLocal.py by getting dataset names from Alexei's db
 2. Modify cleanse.py to gather list from Wensheng's DB
 3. Have Wensheng's script overload archival flag; modify cleanse.py to delete "D" flags
 4. Modify cleanse.py as mentioned before to delete files not at BNL

My vote is 4

If BNL deletes their file/LRC entry, a copy at Tier2 would be available for deletion within 3 weeks