Getting Started with Dalma

High Performance Computing NYUAD
When storage is shared among many users rules must be set to prevent users to consume disk space excessively at the expense of others. For this purpose HPC systems enforce disk quotas to users and groups of users.

On Dalma disk quotas are enforced on /home and /scratch (not /archive or /fastscratch). There are two quota constraints: the total amount of disk space and the total number of files. Once you reach a quota limit your jobs may be killed. So it is a good practice to check your quota before submitting a job that will generate a lot of data.

*myquota* (Dalma specific) will show your storage utilisation and your limits.
The following data policies are enforced on Dalma:

1. Backup: there is no backup performed by our staff, each user is responsible to archive his/her data.

2. Files count: no user is allowed to have more than 1’000 files per directory on /scratch, /home, and /fastscratch, and 500 on /archive. We periodically inspect storage devices and block user accounts if needed.

3. Files which have not been used for more than 90 days on /scratch and /fastscratch may be deleted. Users will be notified and given a chance to archive their data if they wish.
Backup VS Archive

- A backup is an operation where a disk’s contents is automatically copied to tape
- Archiving is an operation where a disk’s contents is manually moved to a long-term storage device

**backup**: automatically & periodically **COPY** data to tape

**archive**: manually **MOVE** data to long-term storage (tape or other)
Dalma's Archiving Components

- `/home`
  - Perl
  - Python
  - Modules
- `/scratch`
  - Data
- `/fastscratch`
  - Special projects
- `/archive`
  - Disk cache
- "black box"
- DMF robotic tape arm

Manually transfer in/out
What /archive does

- DMF (Data Migration Facility) is an automatic data archiving tool
- It buffers data to a disk cache before copying it (e.g., migration) to the tape drive
- Old data on disk cache is eventually removed to make room for more data
- Once removed from disk cache files still appear in cache – file hierarchy remains, but not data
- Accessing such files automatically triggers their migration back to the disk cache
- Migration to tape drive actually makes two tape copies
- One copy is periodically sent to an outside data recovery site
- The tapes have a data retention of 30+ years
- Each time you modify a file on the disk cache two new copies of the latest version are made to tape
- There are no limits on how many file versions that can be made
- There is no disk quota for the amount of space used on tape or disk cache
DO NOT MISUSE `/archive`

- Don't archive directories with more than 500 files
- Don't archive files that aren't meaningful
- Don't move files in and out
- Don't use it to backup your work weekly / monthly
- Don't request a file restore unless absolutely necessary
- Don't use `/archive` as a way around your `/scratch` disk quota
CORRECT DISK USAGE  /archive

- Periodically clean your /scratch & /fastscratch space
- Once a project is completed move the data over to /archive
- Use tar files to archive directories with large file count
- Use "git" or similar versioning tool to keep track of changes in source code
- Remove any data from /archive which you are sure will not be used ever again
List /archive files

- **dmfls**
- Same options to "ls"
- "-l" option display state of file

```bash
> dmfls -l /archive/zidane/

NOTE: all paths to archive must be absolute (eg /archive/...)

```

REG File not managed by DMF
MIG Migrating
ARC Archiving
DUL Dual-state (both on tape and disk cache)
OFL Offline
UNM Unmigrating
NMG Nonmigratable file
PAR Partial-state file
N/A DMF can't determine state
INV Invalid state
Remove /archive files

- `dmfrm`
- Similar options to "rm"

Remove a file:

```
> dmfrm /archive/beckham/myfile
```

Remove a directory:

```
> dmfrm -r /archive/beckham/mydir
```

**NOTE**: deleted files can be recovered within two weeks
Show /archive disk use

- dmfdu
- Similar options to "du"

# blocks used by a directory / file:
```bash
> dmfdu /archive/ronaldinho
27567617 /archive/ronaldinho/WIEN2K
27567617 /archive/ronaldinho
```

Human readable (-h) and grand total (-c):
```bash
> dmfdu -ch /archive/ronaldinho
27G /archive/ronaldinho/WIEN2K
27G /archive/ronaldinho
27G total
```
Transfer files to /archive

- dmfput

- Copy data to /archive

- Usage: dmfput <source> <destination>

- Audit of directories before archiving

- If destination directory exists then only synchronize with new source content (rsync)

> dmfput /scratch/neymar/mydir /archive/neymar

> dmfput /scratch/neymar/myfile /archive/neymar/dirl

> dmfput /scratch/neymar/dirl/myfile /archive/neymar/dirl/myfile

> dmfput /scratch/neymar/mycrazydir /archive/neymar
Auditing /scratch/neymar/mycrazydir
Directory /scratch/neymar/mycrazydir has too many entries (1000 max=500)
Archiving terminated on account of directories failing to meet limits
Retrieve files from /archive

• dmfget

• Copy data from /archive

• Usage: dmfget <source> <destination>

> dmfget /archive/ibrahimovic/dir1/fileX /scratch/ibrahimovic/dir2
Archiving directories with large number of files

- Very wasteful of resources
- Very very slow to scan (e.g., "ls", "find", etc)
- **DO NOT ATTEMPT TO ARCHIVE SUCH DIRECTORIES**
- Create "compressed tar files" of them instead and archive those tar files

```bash
> tar -czf mycrazydir.tgz mycrazydir
> rm -fr mycrazydir
> dmfput /scratch/messi/mycrazydir.tgz /archive/messi
```
Archiving several versions of the same directory

- Write the data into individual directories

> dmfput /scratch/ronaldo/mydir /archive/ronaldo/mydir.20170822

some time later...

> dmfput /scratch/ronaldo/mydir /archive/ronaldo/mydir.20170905