# Use of DMF at QUT

Ashley Wright
High Performance Computing
And Research Support
a2.wright@qut.edu.au

# About QUT

- Public University in Brisbane CBD
- 2 Main Campus at Gardens Point and Kelvin Grove
- 46,000 Students (2,500 HDR Students)
- 11,000 Staff
- $93 million Research Income.

2013 Statistics
http://www.frp.qut.edu.au/services/reporting/

# About HPC at QUT

- High Performance Computing & Research Support (HPC) provides the QUT research commun
- Resources include HPC hardware, Specialised labs, Video Conferencing and support with Visu

# History of HPC at QUT

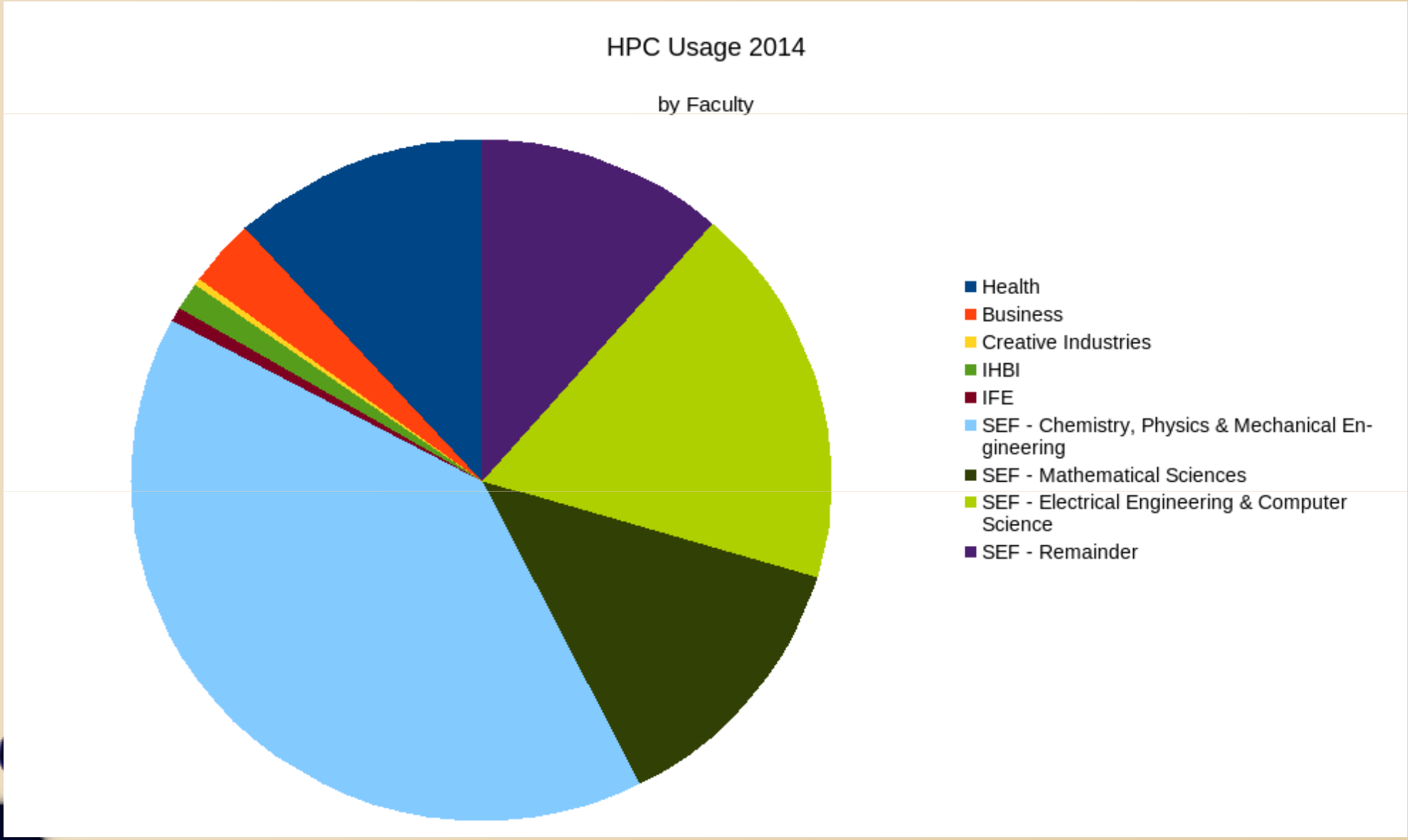
- Initial team formed in 1992
- Started with 3 staff.
- Now 13 staff (12 EFT).

| | |
|---|---|
| 1989 – 1995: | Convex (1, 2 cores), '*Sesame*' |
| 1996 – 2000: | SGI Power Challenge (4, 8, 16 cores), *'Siriu* |
| 2000 – 2007: | SGI Origin 3000 (28, 60, 124, 128 cores), ' |
| 2008 – 2012: | SGI Altix 3400 (96 cores), *'Vega'* |
| 2008 – present: | SGI Altix XE (112, 192, 396, 588, 1292, 154 |

# System Administration

- Two team model.
- HPC and Research Support team handles applications and user support
- Enterprise System Services team handles OS and hardware
- Both teams are in ITS

# Where our Users are From?



# Current HPC Machine

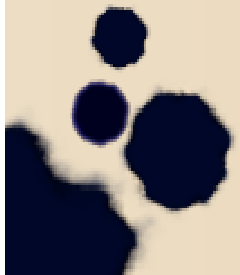
- SGI Lyra Compute Cluster
- 212 Nodes
- 3780 Cores (Xeon X5650, E7-8837, E5-2670, E5-2680v2, E5-2680v3)
- 34TB of RAM  (24/48/64/96/128/256/1024 GB)
- 1.6PB Usable Disk (2x IS5500 and 1x IS5600)
- 2.1PB Usable Tape (2 Libraries, StorageTek T10000D and T2 media)

# DMF Installation

- Active/Passive File Servers (DMF, NFS, CIFS)
- IS5500 (300 Disks) and IS5600 (300 Disks + SSD)
- Oracle Tape Libraries (Shared)
  - Gardens Point
  - Kelvin Grove
- 3 StorageTek T10000D Drives at each site (6 total)
- 110 Tapes (8.5TB / 16TB) at each site

# File System Information

- 190 million active user files.
- 850TB in use.
- Partitions are 100-150TB each
- 30 – 90 million files per partition
- Continuous load between 1,000 – 10,000 IOPS
- About 1-3TB of new data per day
- 2:1 write:read, lots of deletes



## Windows Users

- Bit of a pain point with DMF
- Seem very patient (we wait weeks for data)
- We recall files on their behalf

# DMF File Database

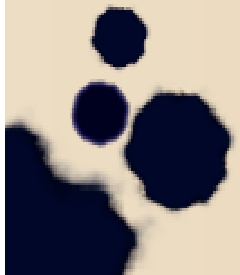
- Idea stolen from DMFUG
- Use dmscanfs.output and dmdump
- Populate a MySQL database
- Used to generate daily reports on user and group usage
- Also used to recall files ordered by tape. (Any search criteria)
- Unfortunately MySQL is too slow.

# Backup

- Don't trust our researchers.
- Backup /home
- DMF migrate & incremental dump is done nightly.
- Retain deleted files for 90 days. (270TB)
- Can take time. 15 – 20 hours.

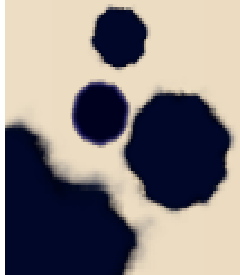
# Quotas

- We implement inode quota in XFS.
  - User created >50 million files in 72 hours.
- About to implement file size quotas.
  - Another User created >70TB of logs.

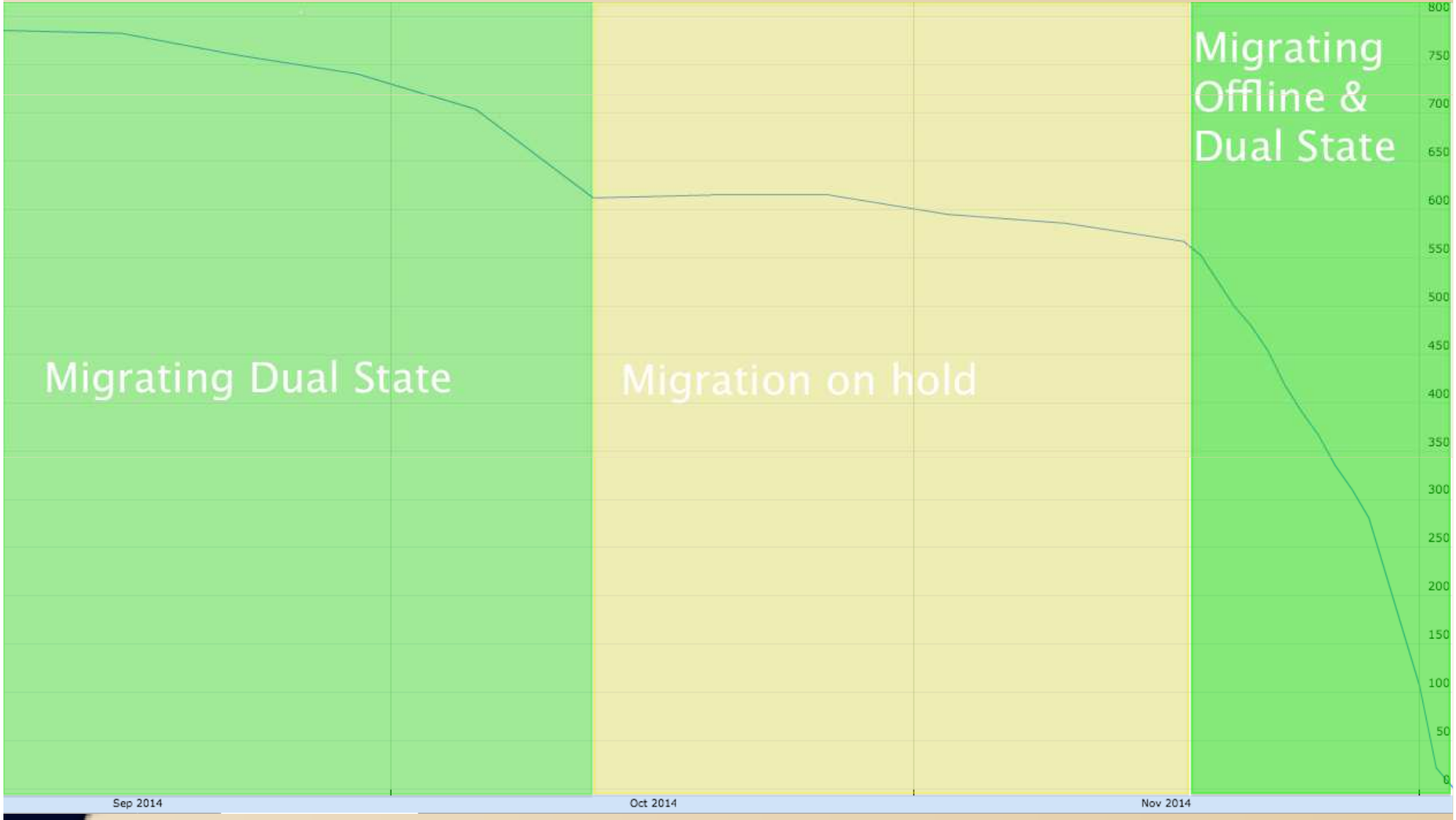


# Tape Migrations

- LTO3 - > LTO5
- LTO5 -> T10000K
- The approach that we took to migrating from the LTO5 drives was to:
    1. Change the migration destination to be only T10000T2 tapes
    2. Run a ‘dmselect -v vg1 | grep DUL | dmmove …’ to push online data to the new tapes
    3. Run a dmmove  for each of the LTO5 drives, migrating the tapes with most data first

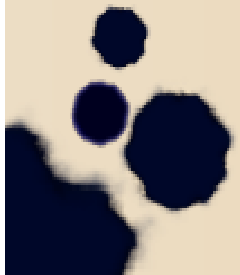


Tape Migrations



## Tape Migrations

- There were some caveats:
- more that 1,000,000 move tasks in the queue slowed DMF and I/O considerably, so new opera
- migration operations got precedence
- moves were killed before nightly migrations
- our MOVE_FS was too small for a number of files.  When these large files were encountered th
- dmselect for DUL state files gets longer and longer as there are fewer DUL state files that fit th



# SystemTap (stap)

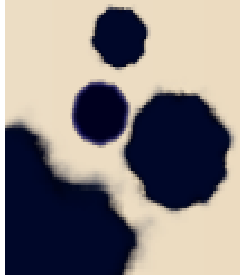
- “SystemTap provides a simple command line interface and scripting language for writing instru
- Can monitor kernel function calls, like "nfsd_vfs_write"
- Allows access to UID, file path, size, IP.
- Unsure of performance impact.

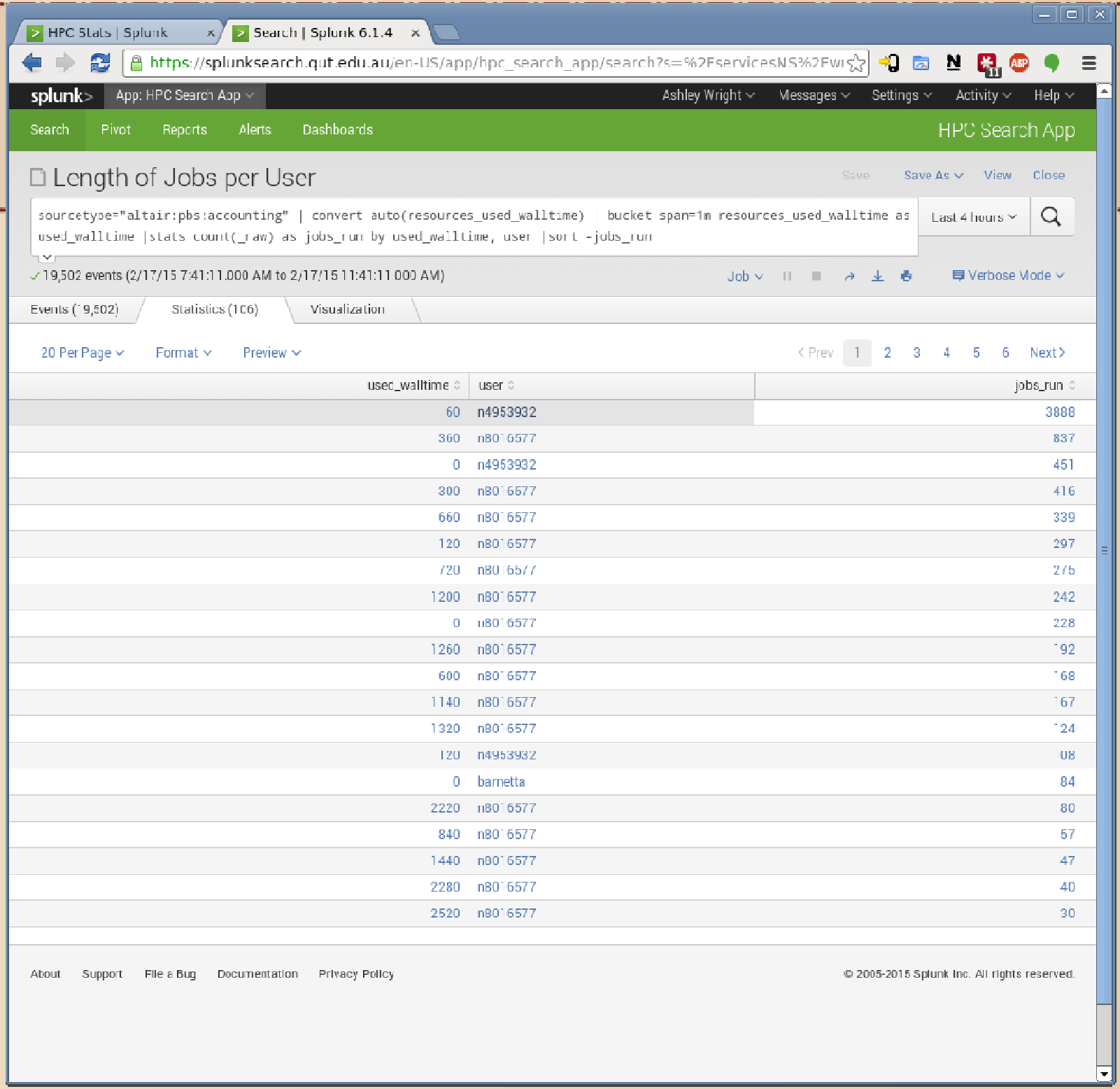
stap

```
probe module("nfsd").function("nfsd_vfs_write").return
{
        uid = $rqstp->rq_cred->cr_uid
        addr = daddr_to_string(addr_from_rqst($rqstp))
        if($file) {
                if($file->f_path->dentry) {
                        fhash = sprintf("%x:%x", $file->f_path->dentry->d_inode, $file->f_path->mnt->m
                        path = file_to_path_hash[fhash]
                        if(nullstr(path)) {
path = sprintf("%s/%s", d_name($file->f_path->mnt->mnt_mountpoint), reverse_path_walk($file
                                file_to_path_hash[fhash] = path
                        }
                }
                stat_hash = sprintf("%d,%s,%s", uid, path, addr)
                nfsd_stat_write[stat_hash] += $cnt[0]
                nfsd_stat_write_count[stat_hash] += 1
        }
```

# Splunk

- Not DMF, but PBS
- Good for searching logs, and correlating by time or other search conditions.
- Query is a mix of SQL-like and Unix pipes.
- Quite easy to use, but bit of a leaning curve.






| used_walltime | user | jobs_run |
|---|---|---|
| 60 | n4953932 | 3888 |
| 360 | n80**6577 | 837 |
| 0 | n4953932 | 451 |
| 300 | n80**6577 | 416 |
| 660 | n80**6577 | 339 |
| 120 | n80**6577 | 297 |
| 720 | n80**6577 | 276 |
| 1200 | n80**6577 | 242 |
| 0 | n80**6577 | 228 |
| 1260 | n80**6577 | **92 |
| 600 | n80**6577 | **68 |
| 1140 | n80**6577 | **67 |
| 1320 | n80**6577 | **24 |
| 120 | n4953932 | 08 |
| 0 | barnetta | 84 |
| 2220 | n80**6577 | 80 |
| 840 | n80**6577 | 57 |
| 1440 | n80**6577 | 47 |
| 2280 | n80**6577 | 40 |
| 2520 | n80**6577 | 30 |

Questions?