Getting the most from the farm at the Sanger Institute

Emyr James
ej4@sanger.ac.uk

Monday 23rd March, 2015

Table of contents

Overview of the Sanger

Monitoring

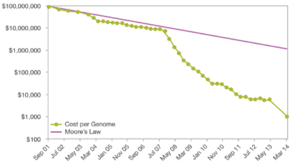
Weekly Farmers Standup Meeting

Lustre Reporting

The Campus

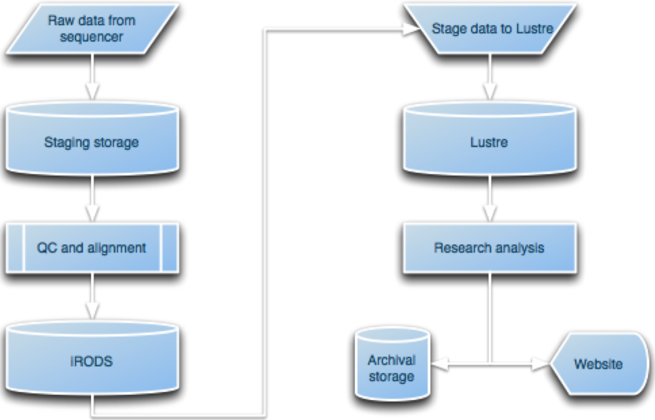
- About 900 staff at the Sanger
- 500 at the EBI
- New sequencing building under construction
- Expect about 30 spinout and startups on site in 2 years

History



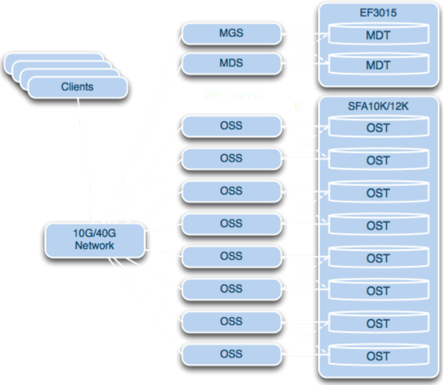
- Set up in 1993
- 1998 - Nematode worm Completed (97Mbp)
- 2003 - Human Genome Complete (2000Mbp)
- 2004 - MRSA Genome
- 2005 - Current Data Centre opens
- 2008 - Next Generation Sequencing, 1000 Genomes Project begons
- 2009 - Joins International Cancer Genome Consortium
- 2010 - UK10K Project begins
- 2013 - UK10K Completed

Typical Workflow

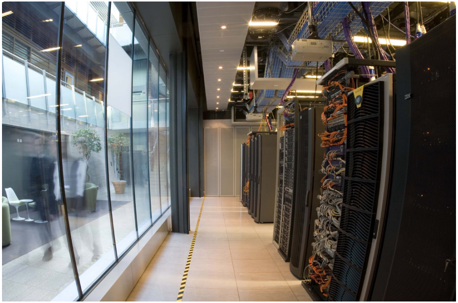


The Cluster

- 11 Lustre Volumes, 2 more imminent, one to be retired
- 250TB/500TB/1PB each
- 6PB total capacity
- DDN Exascaler hardware
- Our own lustre software install
- Aim to deliver 5MB/s for each core
- IB Connected OSS - OST
- 10GigE to clients
- 28PB storage overall (lustre, iRODS, NFS)
- 17,000 cores of compute - mostly HP Blades



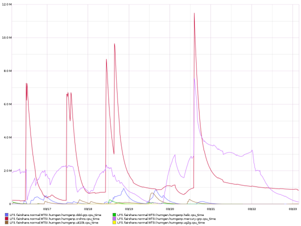
Monitoring

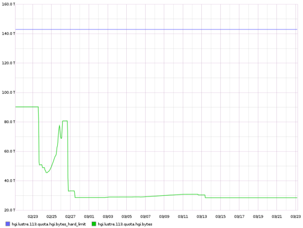


- Ganglia, Opsview / Nagios, Platform LSF / Tableau Analytics in production
- Graphite in test / development
- Hardware ordered for production graphite cluster
- May switch to InfluxDB
- Currently collecting standard metrics (cpu, ram, disk, network)

Application Level Monitoring

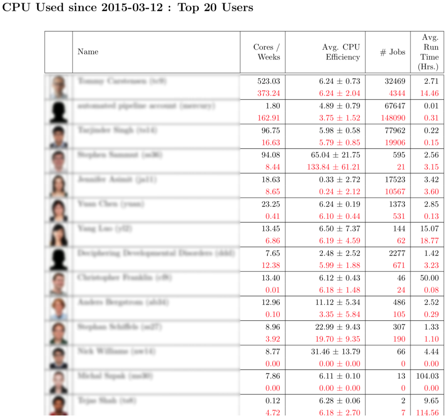
- Graphite makes it easy to add metrics
- Cron which collects Platform LSF Fairshare snapshot every 10 mins
- Has been useful for identifying cpu accounting kernel bug
- Also capturing lustre quota information for groups
- Working on real time analysis for captured data
- Pre-emptive warn of over-quota





The Weekly Report

- Generate weekly usage report for Human Genetics
- python script gets data from LSF Analytics DB (Vertica)
- Gets jpeg image and user name from ldap
- Merges the data with a LaTeXtemplate using pyratemp
- Generates nicely formatted report
- A few copies handed out at the meeting



CPU Used since 2015-03-12 : Top 20 Users

	Name	Cores / Weeks	Avg. CPU Efficiency	# Jobs	Avg. Run Time (Hrs.)
	[redacted]	523.03 373.24	6.24 ± 0.73 6.24 ± 2.04	32469 4344	2.71 14.46
	[redacted]	1.80 162.91	4.89 ± 0.79 3.75 ± 1.52	67647 148090	0.01 0.31
	[redacted]	96.75 16.63	5.98 ± 0.58 5.79 ± 0.85	77962 19906	0.22 0.15
	[redacted]	94.08 8.44	65.04 ± 21.75 133.84 ± 61.21	595 21	2.56 3.15
	[redacted]	18.63 8.65	0.33 ± 2.72 0.24 ± 2.12	17523 10567	3.42 3.60
	[redacted]	23.25 0.41	6.24 ± 0.19 6.10 ± 0.44	1373 531	2.85 0.13
	[redacted]	13.45 6.86	6.50 ± 7.37 6.19 ± 4.59	144 62	15.07 18.77
	[redacted]	7.65 12.38	2.48 ± 2.52 5.99 ± 1.88	2277 671	1.42 3.23
	[redacted]	13.40 0.01	6.12 ± 0.43 6.18 ± 1.48	46 24	50.00 0.08
	[redacted]	12.96 0.10	11.12 ± 5.34 3.35 ± 5.84	486 105	2.52 0.29
	[redacted]	8.96 3.92	22.99 ± 9.43 19.70 ± 9.35	307 190	1.33 1.10
	[redacted]	8.77 0.00	31.46 ± 13.79 0.00 ± 0.00	66 0	4.44 0.00
	[redacted]	7.86 0.00	6.11 ± 0.10 0.00 ± 0.00	13 0	104.03 0.00
	[redacted]	0.12 4.72	6.28 ± 0.06 6.18 ± 2.70	2 7	9.65 114.56

The Meeting

- Human Genetics get together over coffee every thursday at 3.
- Send out the report and try to get at least the top 10 to turn up.
- Stand in a circle and each user says...
  - What they were doing - the science
  - Job submission strategy
    - queue
    - how many jobs
    - memory requirements
    - threading
  - Any problems ?
- Trying to get an idea of what the best practice is
- Identifying areas where
  - We need more documentation
  - May need to improve the systems - any kernel / lustre bugs etc.
  - We could improve tools available to users
- Great for building a community

The Constraints

- Team / project users share quota
- Users can be members of multiple projects
- Project lifetime longer than user tenure
- Long-term need for intermediate project data
- Due to proliferation of projects, quotas overprovisioned
- We need users to tidy up after themselves
- Users need to know where the data is

The Problem

- I need to run some analyses, how much space is available for use by my project?
- You asked us to clean up the disk, where are the oldest large files so I can prioritize them for archiving or deletion?
- My project is near the quota limit. Where is all the space being used ? Who is using it ?
- This is not easy...
  - lfs quota - gives usage but no idea where files are
  - lfs find - stops as soon as you hit "permission denied"
  - find - very hard on the MDS, syntax tricky for users
  - df - can see usage but no granularity
  - du - continues through "permission denied", hits MDS hard, slow, difficult for users
  - agedu - data collection takes a very long time, updated rarely, large list of files in order of last accessed

Towards a solution - mpistat

- Guy Coates found paper on efficient parallel file tree walking using MPI
- Implemented the algorithm with a python class
- Subclassed the walker to make a fast parallel copy program
- I made it do an lstat instead.
- Get full lstat for an entire volume in a practical amount of time (tens of minutes).
- Difficulty in formatting the output - file names with unprintable characters
- Solved by base 64 encoding the path in the tab formatted output file

Summary Report

- Ballpark estimate of cost to store file - £150 per Terabyte per Year
- Calculate a cost for every file based on size and a time
  - ctime - cost to store since creation
  - atime - cost to store since last access - i.e. wastage
- Keep tally of following - totals, by user and by group
  - file sizes
  - file counts
  - zero length files
  - inode type - how many files / directories / symlinks
  - costs
  - files with unprintable characters
  - Example report...

Lustre Treemaps

- Summary all well and good
- Want users to be able to interact with the data
- Treemaps highly suitable
- C++ program...
  - Parses the mpistat output
  - Builds in memory tree, node for each directory
  - Keeps track of accumlators for summaries at each node
  - Embedded http server using facebook proxygen framework
  - Can GET json representation of the tree
- Web Frontend
  - Queries the tree for json of particular subtrees to a given depth (usually 3)
  - Renders treemap using d3.js
- Demo...

Performance and Future Plans

Volume	Files	Size	mpistat	tree build	RAM
scratch114	5.5M	769TiB	23m	17m	5GB
scratch111	16.6M	276TiB	10m	54m	20GB
scratch113	34.6M	649TiB	69m	100m	61GB

- Need gzip encoding of response
- Speed up treebuild - multithreading
- Use key-value store instead of RAM - lmdb
- Real-Time updates - tap into lustre changelog mechanism
- Or use Robin-Hood

Acknowledgemts and References

- Peter Clapham - Platform LSF / Tableau Analytics / Vertica
- Simon Fraser - Local Graphite Guru
- James beal - Resident Lustre Expert
- Matthew Rahtz - Grafana wiz, git-foo
- John Constable - Systems team presence at farmers standup
- Tim Cutts - Sanger overview slides
- Guy Coates - parallel filetree walker, parallel copy
- Parallel Filetree Paper
- Parallel Filetree Website
- Josh Randall - lustre tree front end
- Martin Pollard - investigating lustre changelog mechanism
- HGI Github