Building Bunya: An Evergreen and Bespoke Approach to High-Performance Computing Clusters

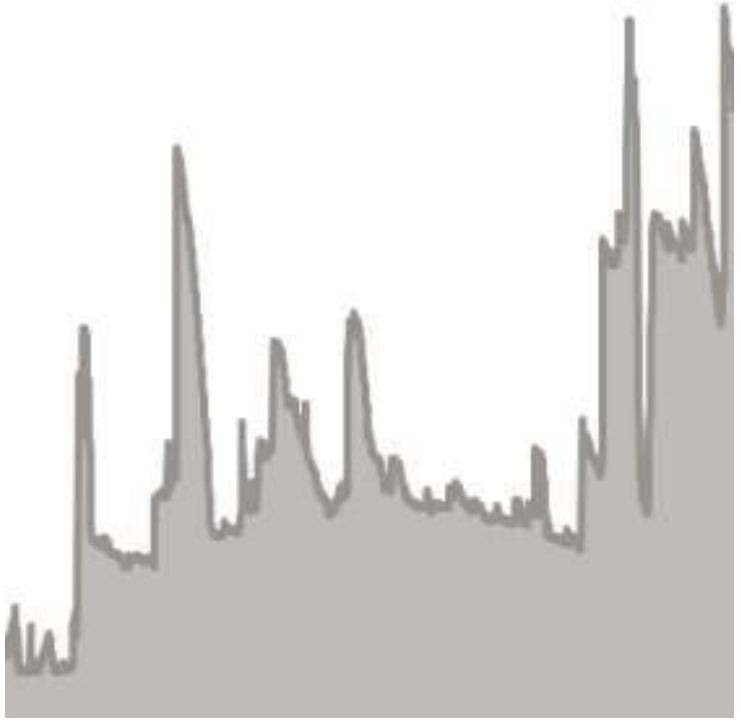
Sarah Walters, Ashley Wright, Owen Powell

Research Computing Centre
The University of Queensland

Challenges

Please, Sir, may I have some more?

- **Ever-growing demand for resources**
- **Broader range of domains are becoming involved in eResearch, including Social Sciences**
- **Big Compute needs Big Data!**



- **“Big workstation on a desk” is no longer viable**
- **Getting hotter, noisier and needing more power**
- **Need maximum value out of expensive hardware – not sitting idle under a desk overnight or when the researcher is on leave**

Challenges

Everybody hates procurement!

- "Buy a big Cray" - these are big, expensive systems, so you can't get them often.
- Need to predict the next several years' workloads, and you're "stuck" with whatever you get whether you got it right or not.
- Big systems like this can take a while to arrive and deploy

- Procurement is a huge overhead
- Going to tender every year isn't practical

- End-users get research grants in small amounts over the lifetime of the system
- Some users will need CPU, some will need GPU, some will be looking for specific architectures.

Challenges

We need experts, we need flexibility, and we needed it yesterday!

- **Distributions like OpenHPC help get you up quickly, but you're locked in to "their way".**
- **What happens when you need a feature from a newer version that isn't in the distribution?**

- **Local teams don't always have the expertise to grow eResearch capability or securely administer servers**
- **Central teams aren't always resourced to build software with quick turn-arounds, so users install locally – but that means they can't share installations.**

- **The knowledge of "how things work" ends up in the heads of key staff … who retire or move on.**

Our Approach

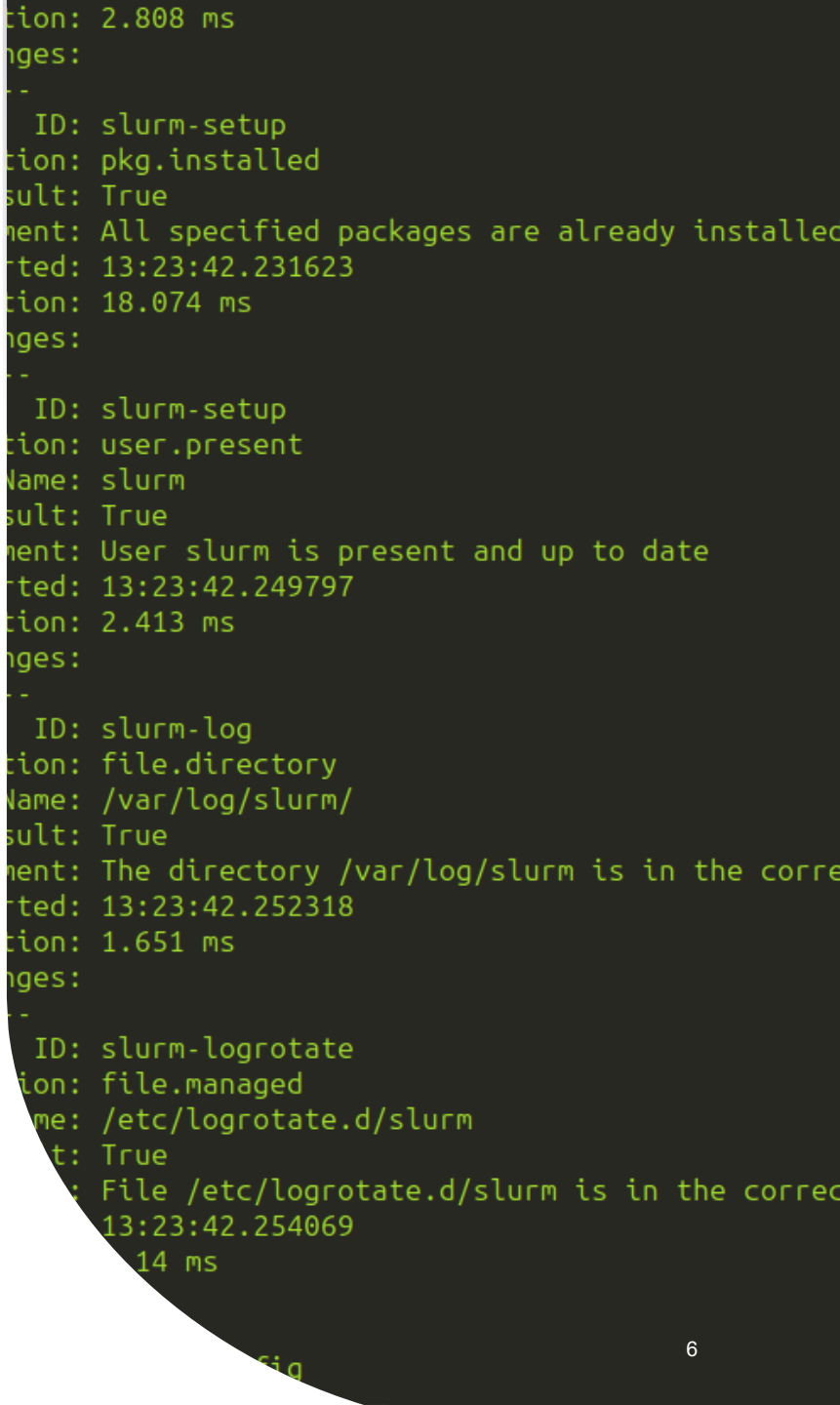
Commodity hardware, done a little differently

- **Off-the-shelf standard servers with 200 Gbps InfiniBand networking – this is still a supercomputer, with the low-latency that implies**
- **Aiming for core performance for price over "the best chip"**
- **Storage is independent - we maximise investment by putting as little storage into the node as possible and having big scratch storage available via RDMA over InfiniBand. This is incredibly fast! We have found that MPI jobs prefer a local /tmp**
- **Vast amounts of RAM (2T on Milan, 1.5T on Genoa, three 4T Milan boxes) to provide for workloads that are not easily supported elsewhere and plenty of room to backfill**
- **We saved money by installing the second phase ourselves**

Our Approach

Servers are not pets!

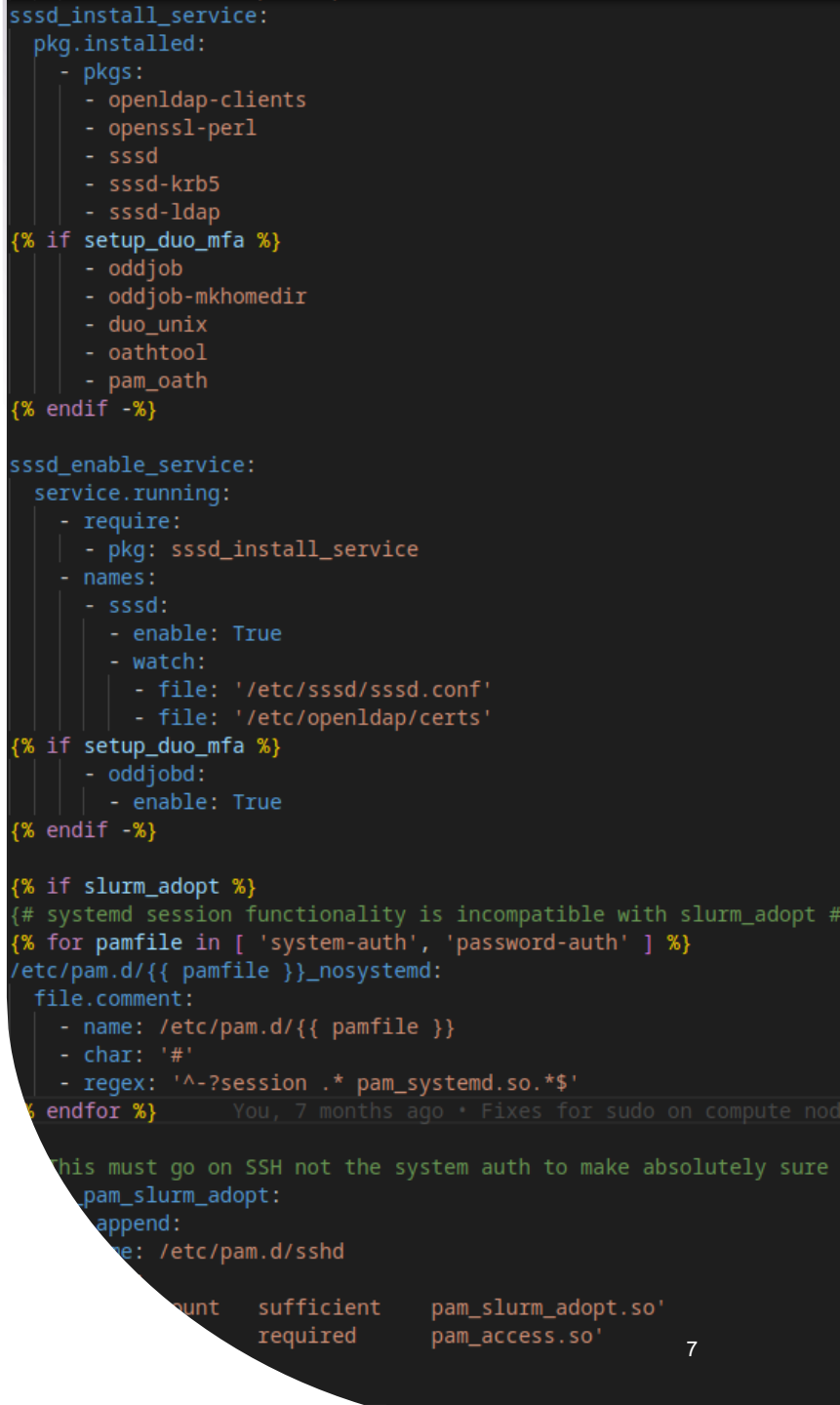
- **Infrastructure-as-Code**
- **Templated bios configuration with vendor tools ensures consistency**
- **Minimal Image – Rocky minimal install with key drivers**
- **SaltStack for Configuration Management**
- **In theory, we can wipe the entire infrastructure and start over from the code (some work still needed here)**
- **All the dependencies are our own - could swap out any component with a change in code. E.g. xCAT image deployment**
- **Reduced time from "servers in racks" to go-live, even on new hardware**



SaltStack

It's not all in our heads!

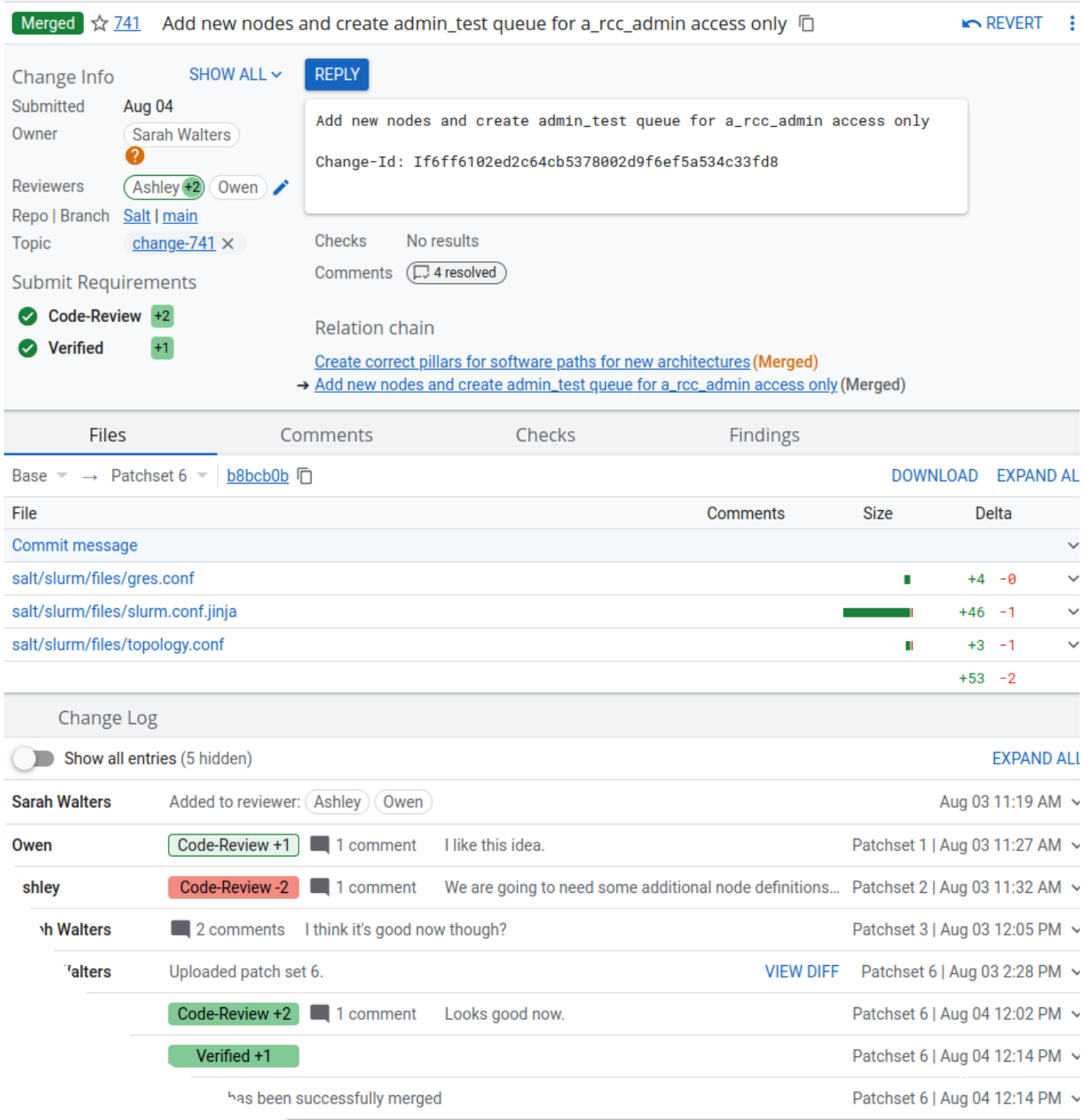
- **Infrastructure as Code isn't a new concept**
- **Reproducable configuration**
- **SaltStack is Python, YAML, and Jinja templating**
- **Central management server**
- **Data "pillars" and Salt "states"**
- **Bootstrap the Salt master from a script**
- **Install a server, set the hostname, bootstrap Salt**
- **Sane certificate distribution for regular host re-imaging**



Code Review

Do it properly the first time!

- **CI/CD pipeline utilising Gerrit and Jenkins - at least two pairs of eyes look over every change, but we can turn a change around in 20 minutes if we need to**
- **Code formatting checkers known as “linters” to enforce good quality code**
- **The history of why a change was made is in the repository**



Software Modules

It's all about what you DO with it!

- **Easybuild - https://easybuild.io/**
- **Over 3000 easyconfigs already available**
- **Users can build their own custom easyconfigs in their scratch area**
- **We are happy to take easyconfig submissions and put them through our Gerrit pipeline**
- **Full test build before adding to Bunya**
- **Some hand-built or commercial software where EasyBuild isn't an option**

```
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/b/binutils/binutils-2.38-GCCcore-11.3.0.eb (module: binutils/.2.38-gcccore
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/n/ncurses/ncurses-6.3-GCCcore-11.3.0.eb (module: ncurses/.6.3-gcccore-11.3
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/p/pkgconf/pkgconf-1.8.0-GCCcore-11.3.0.eb (module: pkgconf/.1.8.0-gcccore-
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/b/bzip2/bzip2-1.0.8-GCCcore-11.3.0.eb (module: bzip2/.1.0.8-gcccore-11.3.0
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/g/GCC/GCC-11.3.0.eb (module: gcc/11.3.0)
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/l/libtool/libtool-2.4.7.eb (module: libtool/.2.4.7)
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/p/pkgconf/pkgconf-1.8.0.eb (module: pkgconf/.1.8.0)
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/a/Autotools/Autotools-20220317.eb (module: autotools/.20220317)
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/o/OpenSSL/OpenSSL-1.1.eb (module: openssl/1.1)
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/x/XZ/XZ-5.2.5-GCCcore-11.3.0.eb (module: xz/.5.2.5-gcccore-11.3.0)
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/c/cURL/cURL-7.83.0-GCCcore-11.3.0.eb (module: curl/.7.83.0-gcccore-11.3.0)
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/l/libarchive/libarchive-3.6.1-GCCcore-11.3.0.eb (module: libarchive/3.6.1-
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/c/CMake/CMake-3.23.1-GCCcore-11.3.0.eb (module: cmake/3.23.1-gcccore-11.3.
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/l/libxml2/libxml2-2.9.13-GCCcore-11.3.0.eb (module: libxml2/.2.9.13-gcccor
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/l/libxslt/libxslt-1.1.34-GCCcore-11.3.0.eb (module: libxslt/1.1.34-gcccore
 * [x] /app/software/EasyBuild/4.8.1/easybuild/easyconfigs/b/bcl2fastq2/bcl2fastq2-2.20.0-GCC-11.3.0.eb (module: bcl2fastq2/2.20.0-gc
 * [ ] /srv/jenkins/work/workspace/Bunya-Auto_21_821_2/easybuild/easyconfigs/s/SpaceRanger/SpaceRanger-2.0.1-GCC-11.3.0.eb (module: s
== Temporary log file(s) /tmp/eb-cpxa1qge/easybuild-fmp7l3c_.log* have been removed.
== Temporary directory /tmp/eb-cpxa1qge has been removed.
eb --robot SpaceRanger-2.0.1-GCC-11.3.0.eb
== Temporary log file in case of crash /tmp/eb-0u0aasdk/easybuild-30g2ltrh.log
== found valid index for /app/software/EasyBuild/4.8.1/easybuild/easyconfigs, so using it...
== resolving dependencies ...
== processing EasyBuild easyconfig /srv/jenkins/work/workspace/Bunya-Auto_21_821_2/easybuild/easyconfigs/s/SpaceRanger/SpaceRanger-2.
== building and installing spaceranger/2.0.1-gcc-11.3.0...
== fetching files...
== ... (took 2 secs)
== creating build dir, resetting environment...
== unpacking...
== ... (took 13 secs)
== patching...
== preparing...
== configuring...
== building...
== testing...
== installing...
== ... (took 7 secs)
== taking care of extensions...
== restore after iterating...
== postprocessing...
== sanity checking...
== cleaning up...
   ... (took 1 secs)
   reating module...
      issions...
         ing...
            : Installation ended successfully (took 27 secs)
               he build can be found in the log file(s) /app/software/SpaceRanger/2.0.1-GCC-11.3.0/easybuild/easybuild-SpaceRanger-2.
                  for 1 out of 1
                     /tmp/eb-0u0aasdk/easybuild-30g2ltrh.log* have been removed.
                        b-0u0aasdk has been removed.
```

Procuring an Evergreen System

Buying stuff is hard – but new stuff every year is awesome!

- **Instead of a once-off purchase, we have a partnership model with our selected vendor**
- **Not locked in, but can purchase with contract variations instead of a full procurement process for three years**
- **Each year we have an annual budget from the University**
- **Negotiate prices on several configurations for a period, generally annually with each major purchase, with an understanding that we can buy individual systems at a similar price for some time after the "big" purchase**
- **Chat with Jake Carroll (UQ CTO Research Computing) if you want to know how to govern things like this legally and contractually**

Heterogeneity

Architecting the architectures

- **Designed to support multiple architectures from the start**
- **EasyBuild software is built across each architecture combination – e.g. epyc3, epyc4, epyc4 with AMD, epyc3 with NVIDIA – for optimal performance**
- **Some hand-crafted or commercial software is architecture-independent or architecture-specific and these are located accordingly**
- **Each architecture has a custom module path configured**
- **Some Slurm partitions (queues) are ‘shared’, and some are not. AMD vs NVIDIA, EPYC3 vs EPYC4 vs Sapphire Rapids.**
- **Features set up in the queue that allow people to target specific hardware – this is important for MPI jobs**

Fairshare

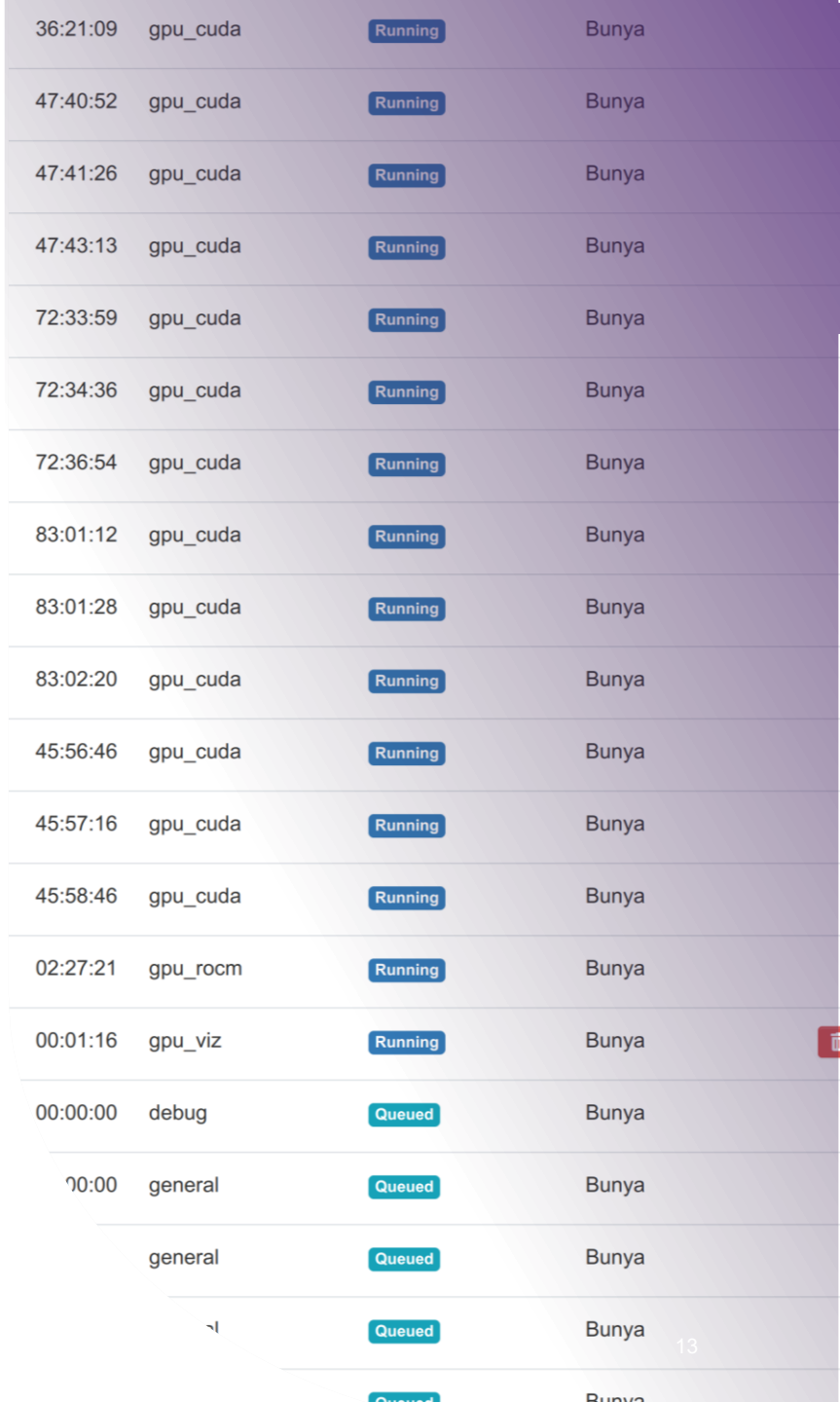
"The miracle is this: The more we share, the more we have" – Leonard Nimoy

- **Every user on Bunya is a member of a Slurm account**
- **Each Slurm account is under its funding "parent"**
- **Each Slurm account gets a total amount of fair-share based on contribution**
- **Bunya uses the Depth-Oblivious Fair-share Factor algorithm to minimise impact of a high-use user on a low-use user in the same account**
- **Usage decays over time**
- **Never have the scenario where allocations for some users are exhausted and parts of the machine are idle**

Researchers Benefit

Maximising Value

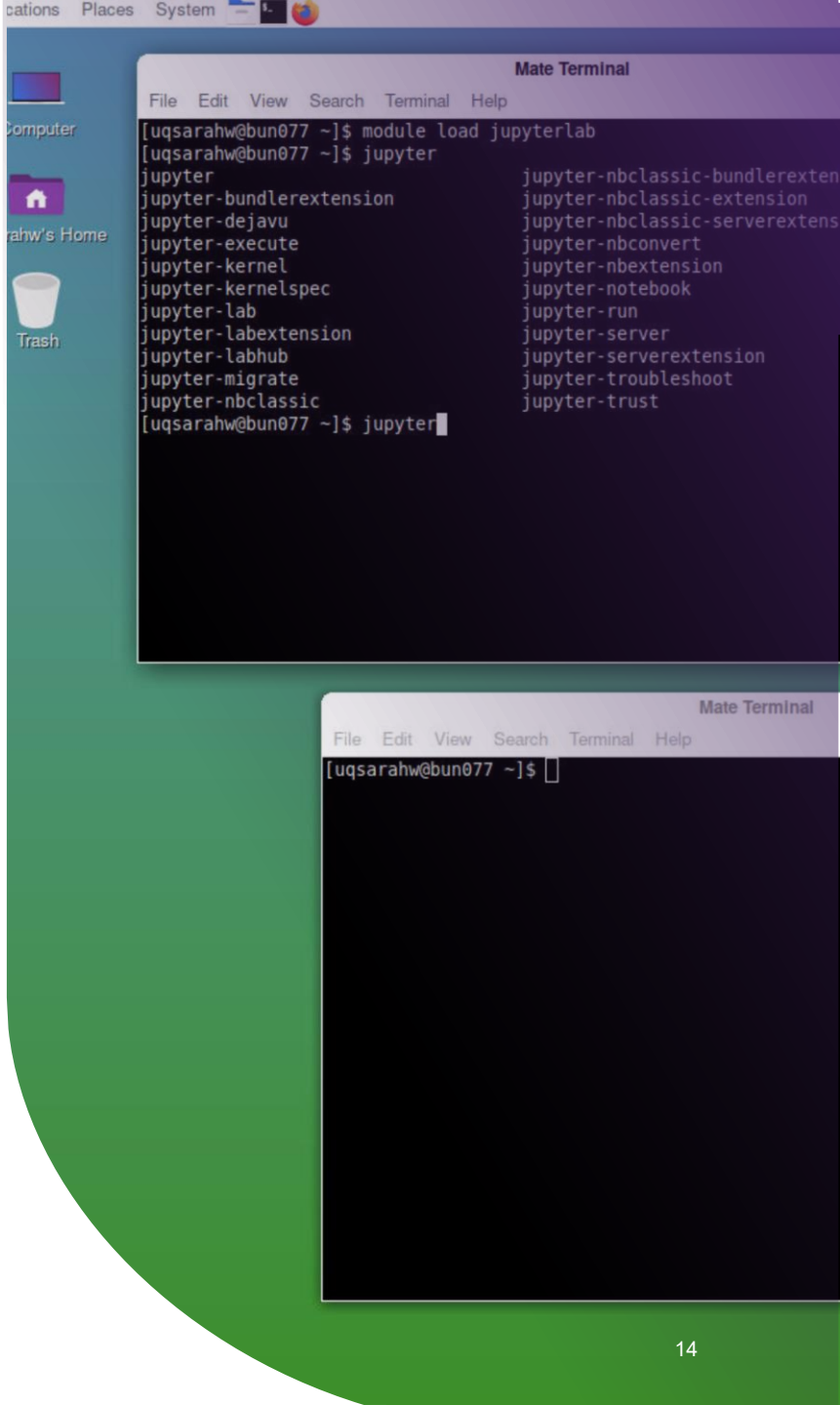
- **Don't need to buy high-powered workstations that sometimes sit idle**
- **Any research group – or entire Institute – can contribute funding and get a fair-share boost. This doesn't penalise other users, because that funding is used to buy more resources. We can provide costings for grant applications.**
- **E.G. Institute for Molecular Bioscience (IMB)**
- **A contribution is calculated as a five-year bonus, but isn't limited to specific hardware – those who bought A100s last year are getting their bonuses on H100s this year**
- **When there is no grant, and there is no departmental budget, the compute is still available**



Researchers Benefit

"Every success story is a tale of constant adaption, revision and change." - Richard Branson

- **Many researchers – especially Early Career researchers – are not familiar with the traditional Linux terminal via SSH**
- **Open OnDemand for a web-based experience that includes a full remote desktop to ease users into the environment**
- **Adaptable software stack**
- **Direct communication between researchers and systems staff to get what they need on the system – both in software and in hardware**
- **Monitor usage and prioritise purchasing whatever is most highly utilised. Annual (or more frequent!) purchases mean that the wait for the most needed resources is shorter.**
- **Access to a variety of hardware: test on different platforms or choose the optimal GPU for the specific workload**



Researchers Benefit

The power of a centrally managed resource

- **Storage is automatically available – plenty of scratch space to work in, and access to the university-wide data fabric. Research Data flows from instruments to the HPC to the desktop, reducing the need to manually move data around.**
- **Sharing systems avoids wastage and brings down the overall environmental cost of running this kind of compute.**
- **No need to have a systems administrator or HPC/computational expert in the research team**
- **Continuity – not only is there more than one person running the system, but the infrastructure as code setup means that the environment can be rebuilt from scratch by people who aren't familiar with our current setup if required**
- **Security – systems engineers know how to secure a system and keep on top of priority patching**

Looking Ahead

"There's a way to do it better – find it." – Thomas Edison

More tools for researchers to access information about their jobs

Trial non-x86 architectures such as NVIDIA's Grace

Virtual replica of compute and management servers for testing

Conclusion

A model that *works*

Tune and change as fast as research does

Deliver more value, faster, with less risk

This makes UQ and our QCIF partners more competitive

Easy to "sell" to senior management