Pro tips for scaling bioinformatics workflows to HPC

Matthew Downton, NCI
Sarah Beecroft, Pawsey
Georgina Samaha, SIH

Australian BioCommons

Today

When to move to HPC?

HPC architecture explained

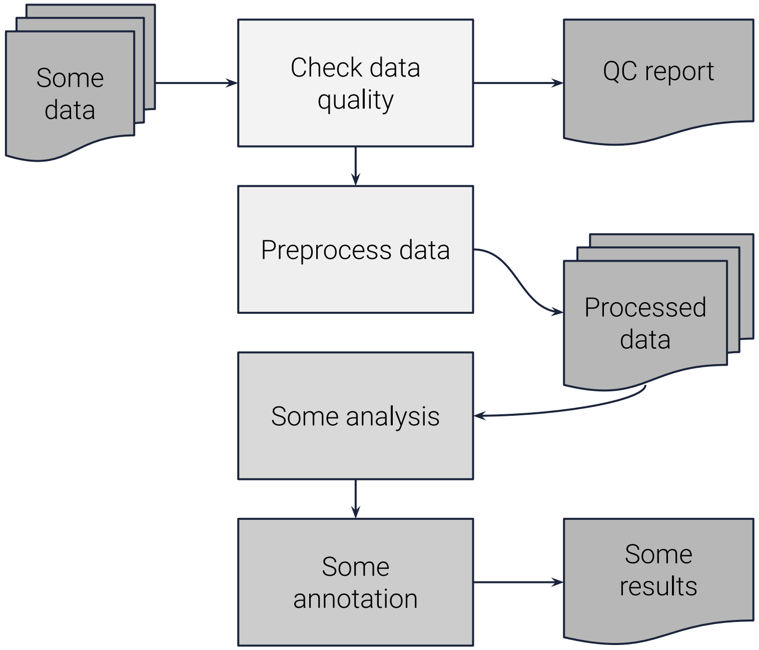
Pro tips for bioinformatic workflows

Where to find support

When should I move my bioinformatics workflows to HPC?

Australian BioCommons

What is a workflow good for?



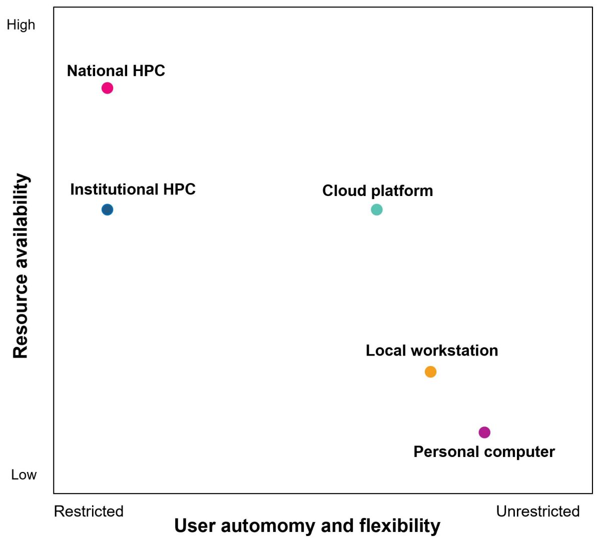
1. Good science requires results be reproducible.
2. Standardised protocols foster collaboration.
3. Some analyses need to be run repeatedly.
4. Automating tasks saves time and resources.

Where can I *do* bioinformatics?

Personal computer	Local workstation	Institutional HPC	Cloud platform	National HPC
Low processing power due to limited cores and memory.	Higher processing power, single nodes.	Multiple nodes allowing for parallel processing.	Scalable, on demand resources.	State of the art processing capacity.
Small storage capacity.	Increased storage capacity for intermediate and final data.	Large-scale storage systems.	Wide range of instance types with varying resources.	Advanced, high-capacity storage systems.
Manual environment management.	Manual environment management.	Job scheduling system distributes workload across the system.	Managed services and pre-configured environments.	Job scheduling system distributes workload across the system and users.
Single user. Admin access.	Few users. Admin access.	Many users. Limited access.	Single-few users. Admin access.	Many users. Limited access.

Australian BioCommons

Things to consider when working on HPC



Benefits

- Enhanced computational power enabling parallel processing
- Designed to handle massive datasets
- Simple scaling of resources for computationally demanding tasks
- Advanced hardware to enhance speed and efficiency

Drawbacks

- Access limitations due to allocation policies, shared resources
- Steep learning curve due to complex resource management techniques
- Dependence on system administrators to configure environment
- Increased code complexity allowing for system architecture

Australian BioCommons

You know it is time to move to HPC when…

- ➔ Your workflow has become computationally intensive.
- ➔ You are approaching a compute resource ceiling.
- ➔ Your workflow needs to be sped up for higher throughput.
- ➔ The size and scale of your datasets have grown.
- ➔ You have data governance obligations.
- ➔ You want to use software that requires specialised hardware.

"My laptop burned a hole on my wooden desk because it was working so hard"- Real user story

Case study: cancer genomics at scale

doi.org/10.1038/s41586-022-05154-6
doi.org/10.1186/s13073-022-01096-w

Project proposal

Research question

- Identify population-specific oncogenic drivers in patients with prostate cancer

Dataset

- Whole genome data for ~200 patients

Bioinformatics

- Mapping to human reference genome
- Short and structural variant discovery

Compute facilities

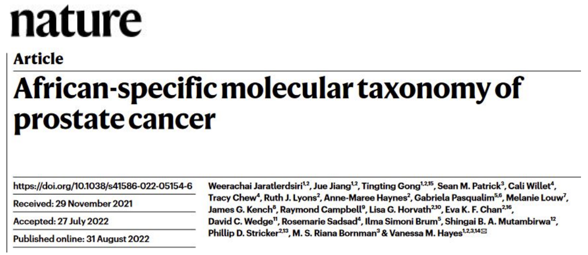
- Existing access to institutional HPC
- Inadequate disk space and resources

Gained access to NCI Gadi via NCMAS allocation scheme

Developed optimised, public workflows to process data

Generated ~200 TB data

Reduced processing time from months to days

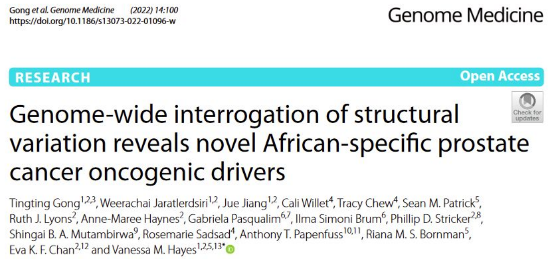

nature

Article

African-specific molecular taxonomy of prostate cancer

https://doi.org/10.1038/s41586-022-05154-6
Received: 29 November 2021
Accepted: 27 July 2022
Published online: 31 August 2022

Weerachai Jaratlerdsiri[1,2], Jue Jiang[1,2], Tingting Gong[1,2,15], Sean M. Patrick[3], Cali Willet[4], Tracy Chew[4], Ruth J. Lyons[2], Anne-Maree Haynes[2], Gabriela Pasqualim[5,6], Melanie Louw[7], James G. Kench[8], Raymond Campbell[9], Lisa G. Horvath[2,10], Eva K. F. Chan[2,16], David C. Wedge[11], Rosemarie Sadsad[4], Ilma Simoni Brum[5], Shingai B. A. Mutambirwa[12], Phillip D. Stricker[2,13], M. S. Riana Bornman[3] & Vanessa M. Hayes[1,2,3,14]


Gong *et al. Genome Medicine* (2022) 14:100
https://doi.org/10.1186/s13073-022-01096-w

Genome Medicine

RESEARCH Open Access

Genome-wide interrogation of structural variation reveals novel African-specific prostate cancer oncogenic drivers

Tingting Gong[1,2,3], Weerachai Jaratlerdsiri[1,2], Jue Jiang[1,2], Cali Willet[4], Tracy Chew[4], Sean M. Patrick[5], Ruth J. Lyons[2], Anne-Maree Haynes[2], Gabriela Pasqualim[6,7], Ilma Simoni Brum[6], Phillip D. Stricker[2,8], Shingai B. A. Mutambirwa[9], Rosemarie Sadsad[4], Anthony T. Papenfuss[10,11], Riana M. S. Bornman[5], Eva K. F. Chan[2,12] and Vanessa M. Hayes[1,2,5,13*]

Australian BioCommons

HPC is a bit different for bioinformatics…

Aspect	Bioinformatics	Other disciplines
Data intensity	Varied dataset sizes and formats. Commonly sequence analysis.	Arrays of data simulations and calculations.
Algorithms	NP-hard problems, dynamic programming, interactive analysis.	Deep learning, large-scale optimisation, dynamic simulations.
Tools and software	Many designed for use on single machines.	Often specialised and built for HPC.
Scale and complexity	Multi-step workflows, resources are data dependent, require multiple tools.	Large-scale data reduction and analyses run by single tools.

Australian BioCommons

HPC architecture explained

Australian BioCommons

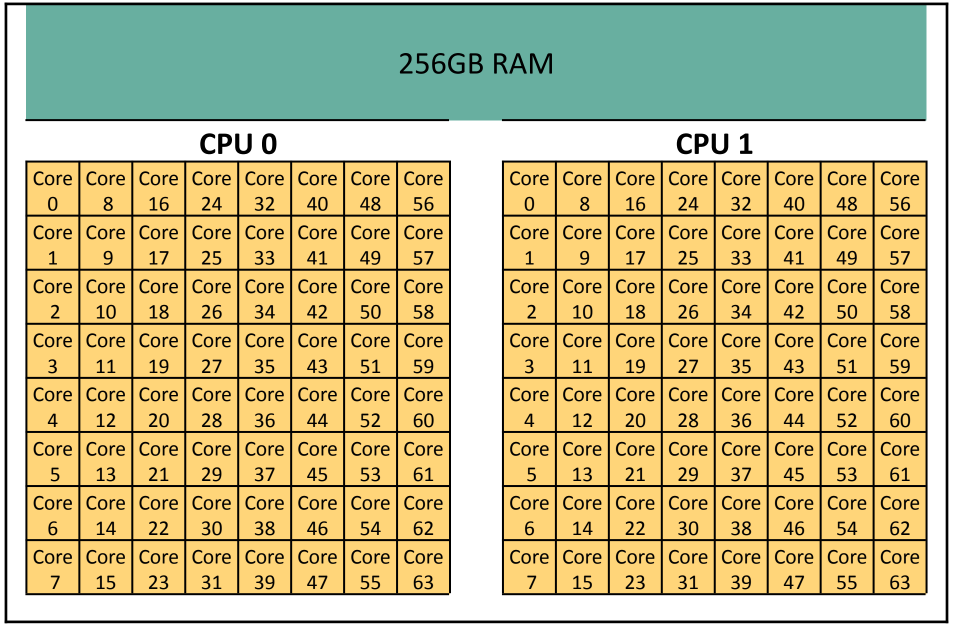
Abridged anatomy of a laptop

- Central processing unit (CPU)
  - Performs processing tasks that make computers useful
- Random Access Memory (RAM)
  - Fast access storage used during processing
  - Analogous to human working memory
- Storage
  - Files live here persistently
  - Analogous to human long-term memory

HPC uses the same basic elements as a laptop

Supercomputers have thousands of cores, grouped into 'nodes'. In this example:

- 2 CPUs
- Each CPU has 64 cores
- 256 GB RAM shared by the 2 CPUs
- A supercomputer uses hundreds of nodes, which share long-term storage


256GB RAM
CPU 0
CPU 1
Core 0 … Core 63 (each CPU, 8 columns: Core 0–7, 8–15, 16–23, 24–31, 32–39, 40–47, 48–55, 56–63)

Simplified view of a single node with 128 cores and 256GB RAM

Anatomy of a supercomputer

Login nodes

- Launch job scripts, interact with scheduler

Scheduler

- Program that handles where/when to run jobs

Data mover nodes

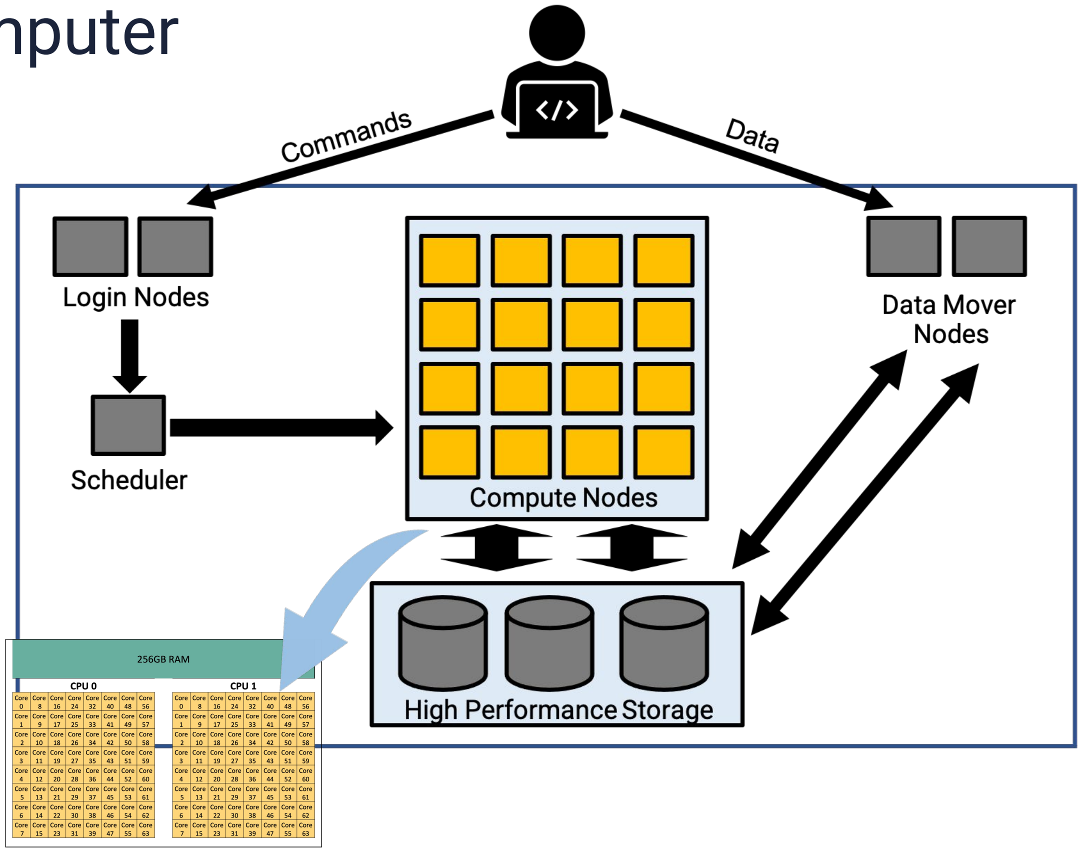
- Specialised for upload/download of data

Compute nodes

- Performs the computation

High performance storage

- Fast temporary read/write of files



CPU 0

Core 0	Core 8	Core 16	Core 24	Core 32	Core 40	Core 48	Core 56
Core 1	Core 9	Core 17	Core 25	Core 33	Core 41	Core 49	Core 57
Core 2	Core 10	Core 18	Core 26	Core 34	Core 42	Core 50	Core 58
Core 3	Core 11	Core 19	Core 27	Core 35	Core 43	Core 51	Core 59
Core 4	Core 12	Core 20	Core 28	Core 36	Core 44	Core 52	Core 60
Core 5	Core 13	Core 21	Core 29	Core 37	Core 45	Core 53	Core 61
Core 6	Core 14	Core 22	Core 30	Core 38	Core 46	Core 54	Core 62
Core 7	Core 15	Core 23	Core 31	Core 39	Core 47	Core 55	Core 63

CPU 1

Core 0	Core 8	Core 16	Core 24	Core 32	Core 40	Core 48	Core 56
Core 1	Core 9	Core 17	Core 25	Core 33	Core 41	Core 49	Core 57
Core 2	Core 10	Core 18	Core 26	Core 34	Core 42	Core 50	Core 58
Core 3	Core 11	Core 19	Core 27	Core 35	Core 43	Core 51	Core 59
Core 4	Core 12	Core 20	Core 28	Core 36	Core 44	Core 52	Core 60
Core 5	Core 13	Core 21	Core 29	Core 37	Core 45	Core 53	Core 61
Core 6	Core 14	Core 22	Core 30	Core 38	Core 46	Core 54	Core 62
Core 7	Core 15	Core 23	Core 31	Core 39	Core 47	Core 55	Core 63

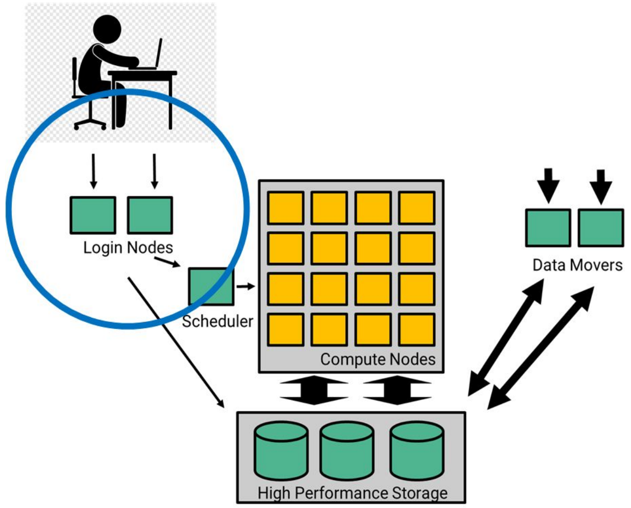
Remote access is via login nodes

Remote access to the supercomputer for administrative work:

- Submit jobs
- Manage workflows
- Check results
- Install software

Many people (~100) share a login node

Do not run programs on the login nodes!



Australian BioCommons

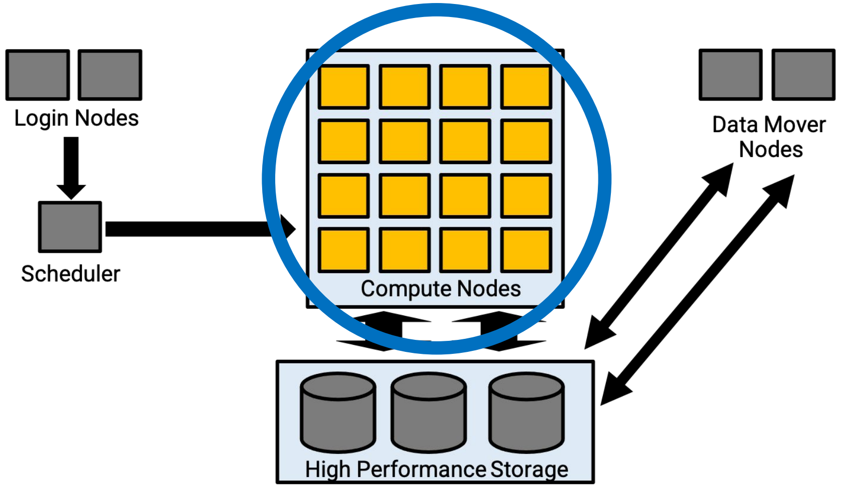
Compute nodes do the heavy lifting

Compute nodes work together:

- To perform large tasks
- Or to obtain a faster execution of your code
- Or to perform many different tasks at the same time

Please delete files when you're done 🙏

These are shared resources, so your jobs are submitted to a queue



Australian BioCommons

Pro tips

Pro-tip 1: Software installation is different on HPC

- HPC is a shared resource
  - No sudo for you!
  - Users can’t do system-wide installations
- You **can** install in your own directories

Australian BioCommons

Pro-tip 1: Software installation is different on HPC

In recommended order of preference:

- Modules are provided by System Administrators
  - Can 'load' or 'unload' them as needed, prevents version conflicts
- Containers
  - Totally isolated software environment, very reproducible and easy
- Conda/Mamba (HINT: If you're going to use Conda, use Mamba instead)
  - Package manager software
- Local installs
  - Specify installation in your directory e.g. /software/your_project/user_name

Australian BioCommons

Pro-tip 1: Software installation is different on HPC

What can you use containers for?

- Avoid installation of packages (e.g. Conda/Pip/apt)
- Handling clashing/multiple versions
- Shipping software you've published
- Fitting in with Nextflow/Snakemake/Cromwell
- Improved automation

Pro-tip 2: Tools to manage your project

Once you have an application accepted, you get quota for the following:

1. *Compute time.* Often measured in Service Units (SUs).
2. *Storage.* Measured in GB/TB and file count. May be of different types.

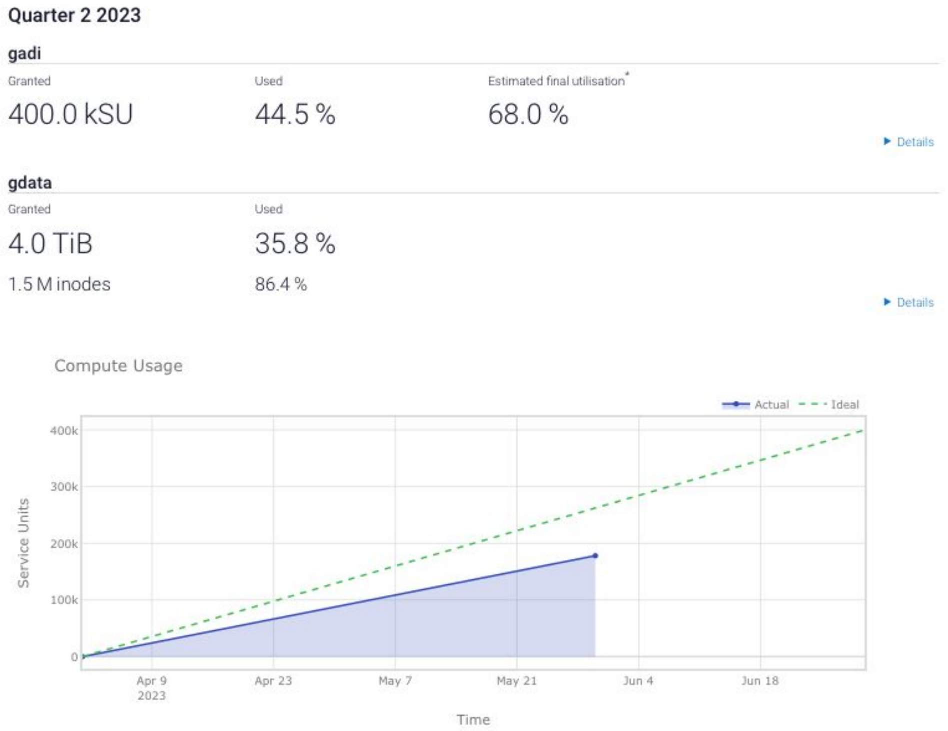
Facilities provide tools and information to help you manage your project well.

1. Quota.
2. Storage.
3. Jobs.

Pro-tip 2: Tools to manage your project - quota

Quarterly quota limits often apply.

- If you don't use all of the quota by the end of quarter, it can't be renewed.
- Important to use the quota evenly over time.
    - Don't get caught in the end of quarter rush!



Quarter 2 2023

gadi

Granted	Used	Estimated final utilisation*
400.0 kSU	44.5 %	68.0 %

▸ Details

gdata

Granted	Used
4.0 TiB	35.8 %
1.5 M inodes	86.4 %

▸ Details

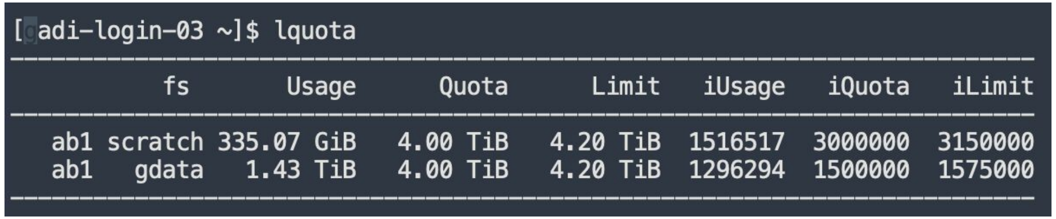
Pro-tip 2: Tools to manage your project - storage

Network filesystems

- High capacity, high throughput
- Quotas on file counts as well as space

Different types of storage.

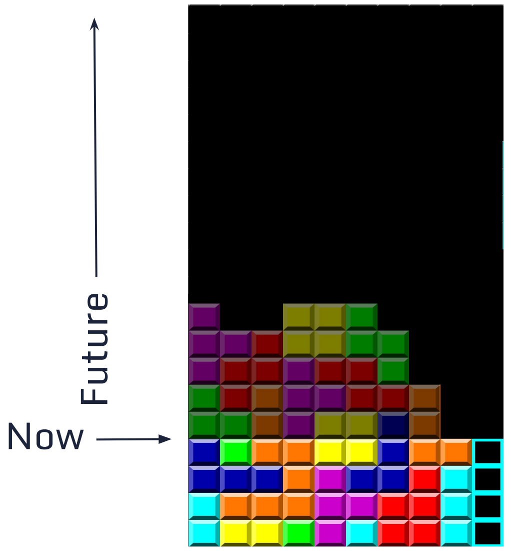
- *Scratch*. Used for calculations, files can be deleted if they are not accessed regularly (i.e. 90 days).
- *Persistent*. Used for storing files over longer periods of time (gdata at NCI).



```
[gadi-login-03 ~]$ lquota
------------------------------------------------------------------------------
           fs        Usage        Quota       Limit   iUsage   iQuota   iLimit
------------------------------------------------------------------------------
   ab1 scratch 335.07 GiB     4.00 TiB    4.20 TiB  1516517  3000000  3150000
   ab1   gdata   1.43 TiB     4.00 TiB    4.20 TiB  1296294  1500000  1575000
------------------------------------------------------------------------------
```

nci-files-report	Detailed report of usage
nci-file-expiry	Files to be deleted from scratch.

Pro-tip 2: Tools to manage your project - jobs



Submitting a batch job tells the scheduler about the 'shape' of your job.

- The scheduler considers all of the queued jobs as a candidate for the next job to start.
- Three dimensions:
    - Number of cores
    - Memory
    - Walltime

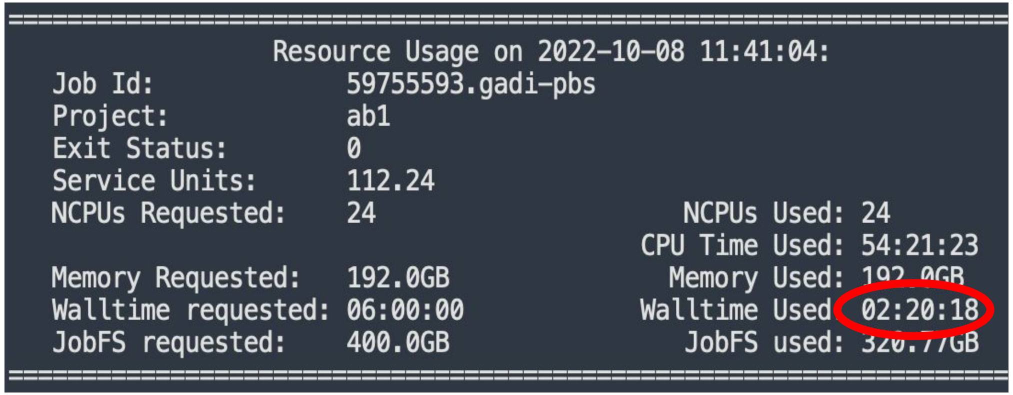
Getting this wrong can lower the system efficiency.

- Longer queue times.
- Underused resources.

Pro-tip 2: Tools to manage your quota - jobs

Facilities provide information to help you see how well resources are used.

- At NCI a 'postscript' is added to the end of standard output for every job.
- Other sites may do the same or have a special command .e.g. seff for slurm.



```
======================================================================================
                  Resource Usage on 2022-10-08 11:41:04:
   Job Id:             59755593.gadi-pbs
   Project:            ab1
   Exit Status:        0
   Service Units:      112.24
   NCPUs Requested:    24                        NCPUs Used: 24
                                              CPU Time Used: 54:21:23
   Memory Requested:   192.0GB                  Memory Used: 192.0GB
   Walltime requested: 06:00:00               Walltime Used: 02:20:18
   JobFS requested:    400.0GB                  JobFS used: 320.77GB
======================================================================================
```

Efficiency = (CPU time used)/(maximum CPU time possible)
= (54.3 core hours)/(2.3 hours * 24 cores)
= 0.98

Australian BioCommons

Pro-tip 3: Work smarter not harder

Now you can manage your project, is there anything else you can do to use the resources well?

- Might want to get more science out of the project.
- Maybe the number of samples has changed.

Performance optimisation is a very large topic. In the following some ideas are presented on how to think about performance and introduce some things to try.

1. Metrics to understand cost.
2. Limits of parallelisation.
3. Scatter-gather patterns.
4. Explore different tools.

Pro-tip 3: Work smarter not harder - metrics

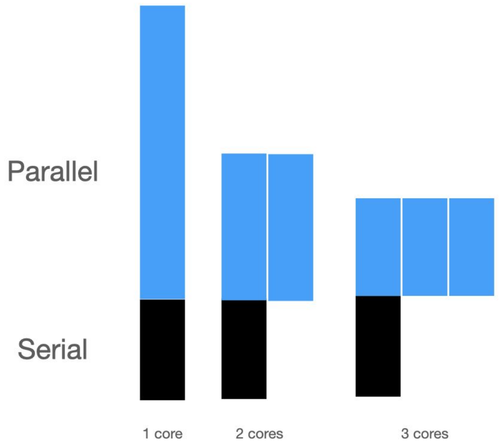
What's important to you? Time, SUs, something else?

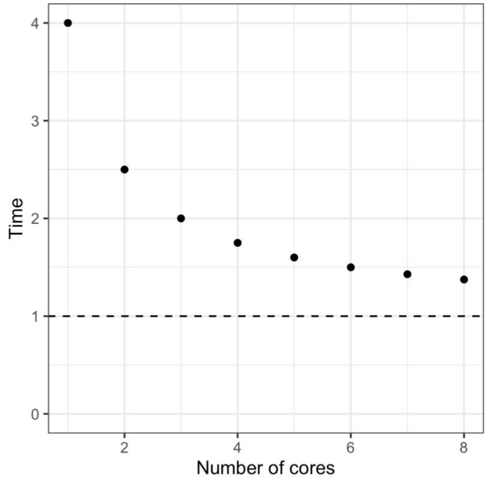
- **Time** (time per sample)
  - It's important to process each sample in the minimum possible time.
  - Tradeoffs: Cost per sample; efficiency.
- **Cost** (SUs per sample)
  - Want to process as many samples as possible.
  - Maybe choose a slower & cheaper queue or reduce number of cores.
- **Throughput** (Samples/day or samples/week)
  - Processing data at the same rate it's generated.

Pro-tip 3: Work smarter not harder - limits to parallelisation

Multithreaded tools may have an option to set the number of threads.

- Why not set it to the largest number possible?
- Only part of the problem can be shared across threads.





Pro-tip 3: Work smarter not harder - scatter-gather

Data parallel workflows apply the same steps to different samples.

- Sometimes the data itself can be split into multiple pieces that are analysed independently.
- At the end of processing these can be stitched back together.

Tools to look at:

- GNU parallel. See also nci-parallel at NCI.
- Array jobs with slurm.
- Dask with python.

Australian BioCommons

Pro-tip 3: Work smarter not harder - different tools

Often there are different software tools that do the same task.

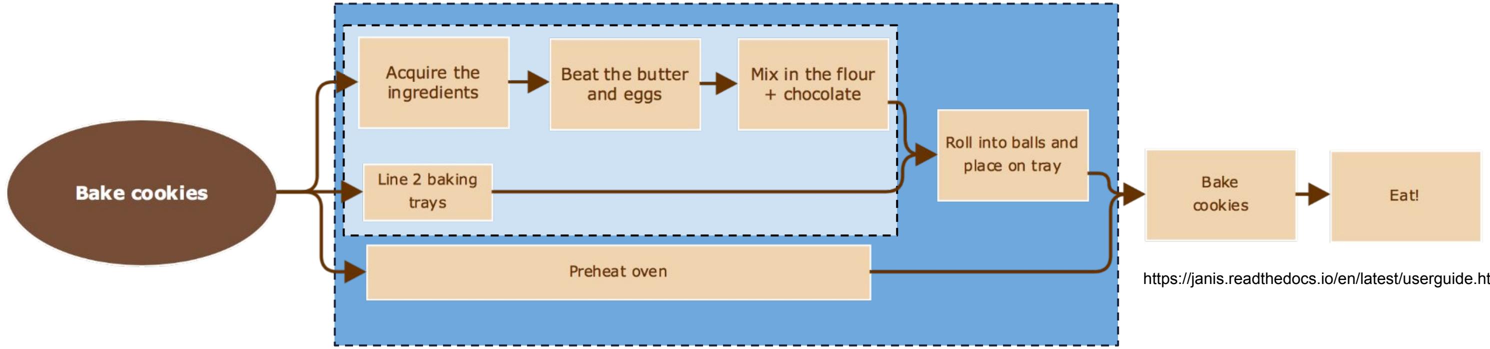
- Find places in your workflow that are time consuming or need a lot of SUs.
    - Are there any alternative tools that could be used?
        - bwa , bwa-mem2, parabricks all do alignment.
        - current code may be in python, but a compiled alternative exists.
- **Consideration 1**: Scientific benchmarking
    - Results need to be identical
    - Results need to be equivalent
- **Consideration 2**: Performance benchmarking
    - Number of service units.
    - Time.

Australian BioCommons

Pro-tip 4: Workflow management tools are your friend

Workflow management tools promote workflow portability, scalability, and reproducibility.

They can dynamically handle process dependencies, simultaneous task execution, and fault tolerance.



https://janis.readthedocs.io/en/latest/userguide.html

Pro-tip 4: Workflow management tools are your friend

nextflow

Supports multiple execution platforms and languages.

Workflow re-entrancy. Each task is cached.

Very active global community.

snakemake

Python-based language.

Automatic dependency resolution and rule-based execution.

Dynamic workflow execution based on input data.

JANIS

Cromwell-based execution engine.

Consistent language for describing workflows.

Framework that can translate between workflow languages.

CWL

Can be executed by different workflow management systems.

Specification language for describing workflows and tools.

Platform agnostic.

{wdl}

Ensures inputs and outputs are compatible.

Task modularity for reuse across workflows.

User defined error-handling and retry mechanism.

Australian BioCommons

Pro-tip 4: Workflow management tools are your friend

What can a workflow management tool do for you on HPC?

- Use multiple software installation methods including containers
- Allow you to port your workflow across different infrastructures and share with others
- Manage software versions and dependencies to ensure your workflow is reproducible
- Spawn jobs to the scheduler and increase throughput
- Checkpoint your workflow- you don’t have to start over if it crashes!
- Create nice resource usage graphs to help you better allocate resources
- Run someone else’s workflows with a bit of upskilling and practice

Australian BioCommons

Where to next?

Australian BioCommons

Getting access to HPCs

What facilities?

- Institutional HPCs
- Australian research computing facilities

How do I get access?

- Contact your institutions ICT or eResearch services
- Merit schemes: Pawsey Partner, ADAPTER NCMAS
- Start up, industry, institutional schemes
- Funded options

Where can I find support?

- Infrastructure Help Desks, documentation, training
- Slack
  - bioinformatics-hpc-au.slack.com
  - nfcore.slack.com
  - nextflow.slack.com
- YouTube
  - @NCIAustralia
  - @PawseySupercomputingCentre
  - @AustralianBioCommonsChannel
- Bioinformatics workflows community
  - https://www.biocommons.org.au/workflows

Bioinformatics @ national HPCs

NCI AUSTRALIA

- Shared bioinformatics software (if89)
- NVIDIA Parabricks GPU-enabled toolkit
- Nextflow Tower service
- Public workflows @ WorkflowHub
- Interactive environments
- NCI data collections

Pawsey

- Globally installed bioinformatics software
- Nextflow Tower service
- Public workflows @ WorkflowHub
- Interactive environments
- Reference datasets.

https://workflowhub.eu/programmes/8

https://australianbiocommons.github.io/tower/

Australian BioCommons

So, is your workflow fit for purpose?

1. Use a software management tool to handle tool dependencies and avoid installation woes.
2. Understand the resource needs of your tools and tasks.
3. Allocate resources responsibly, don't request more than you need.
4. Use workflow checkpointing to manage progression and workflow failure.
5. Tidy up after yourself, disk space is a shared resource.
6. Evaluate your workflow efficiency at the task level using job logs.
7. Ask for help! You are not alone, there are plenty of resources and communities out there.

Australian BioCommons

Thank you

Australian BioCommons