VUW HUB FOR SIMULATION, NUMERICAL METHODS, ANALYTICS AND PROGRAMMING

Rāpoi Updates

SNAP Symposium 2026 — Victoria University of Wellington

Dr Muhammad Ali Hashmi
Research Computing Specialist

What We'll Cover

01
What is Rāpoi?

02
Cluster Hardware

03
Cluster Network

04
Slurm Scheduler

05
Resources & Job Management

06
Usage & Demand

07
Common Mistakes

08
Best Practices

09
Getting Help

What is Rāpoi?

RĀPOI
VUW's High Performance Compute Cluster



HPC Cluster

A collection of large servers (nodes) connected together with shared storage



Slurm Scheduler

Jobs are submitted to a scheduler that allocates CPU, memory, and time



Partitions

Different server/node groups: quicktest, parallel, gpu, bigmem, longrun



Network

Connected via 10G Ethernet for external connections and 52G InfiniBand for node-to-node communication (parallel partition only)

Tip: Most of the time you'll share a node with other users — request only what you need!

Cluster Hardware

AMD Parallel Nodes	GPU Nodes	Big Memory Nodes	Long Run Nodes
amd02n01-amd02n04 amd03n01-amd03n04 amd04n01-amd04n04 amd05n01-amd05n04 amd06n01-amd06n04 amd07n01-amd07n04	NVIDIA A100 GPUs gpu01, gpu02, gpu03 For CUDA/OpenCL workloads	high01–high04 Up to 1 TB RAM NVIDIA Tesla T4 gpu For memory-intensive work	bigtmp01–bigtmp02 30-day max runtime For extended computations 25TB /tmp on each node

The CPU specs are more or less the same for all these nodes
2× AMD EPYC 7702
128 cores / 256 threads
512 GB RAM per node

Login Node: raapoi-login → Submit jobs → Scheduler assigns to compute nodes

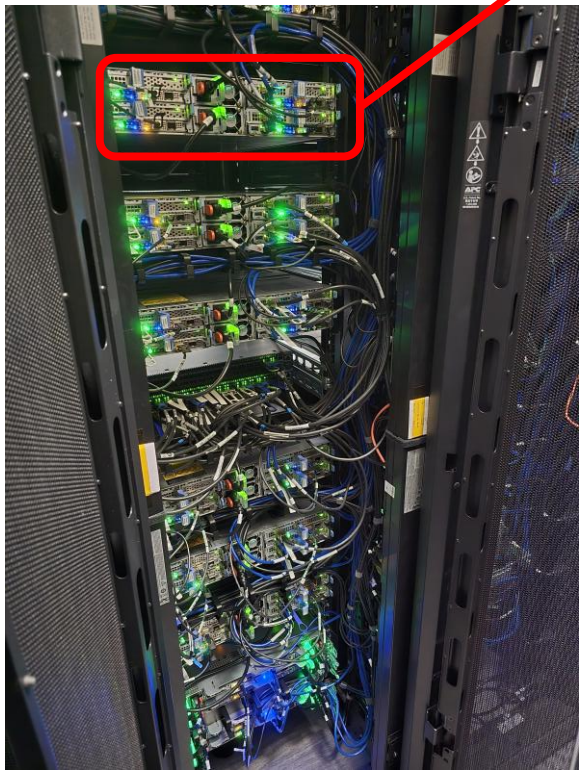
Cluster Racks

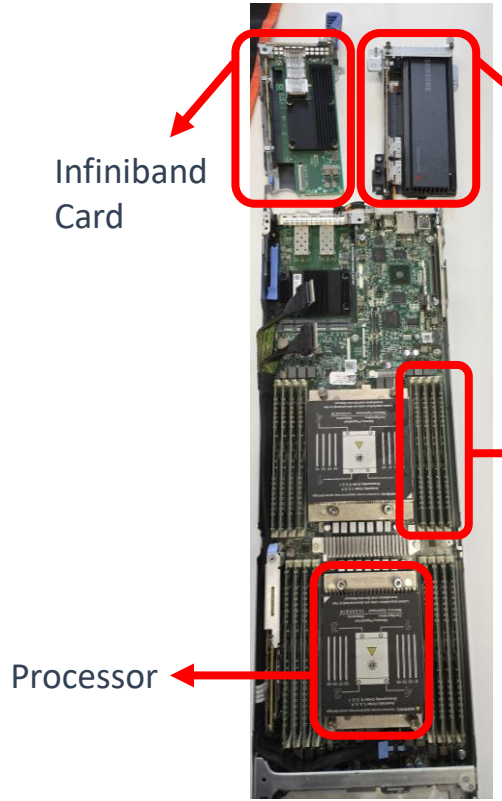
Racks front side



Backside of a rack with parallel nodes

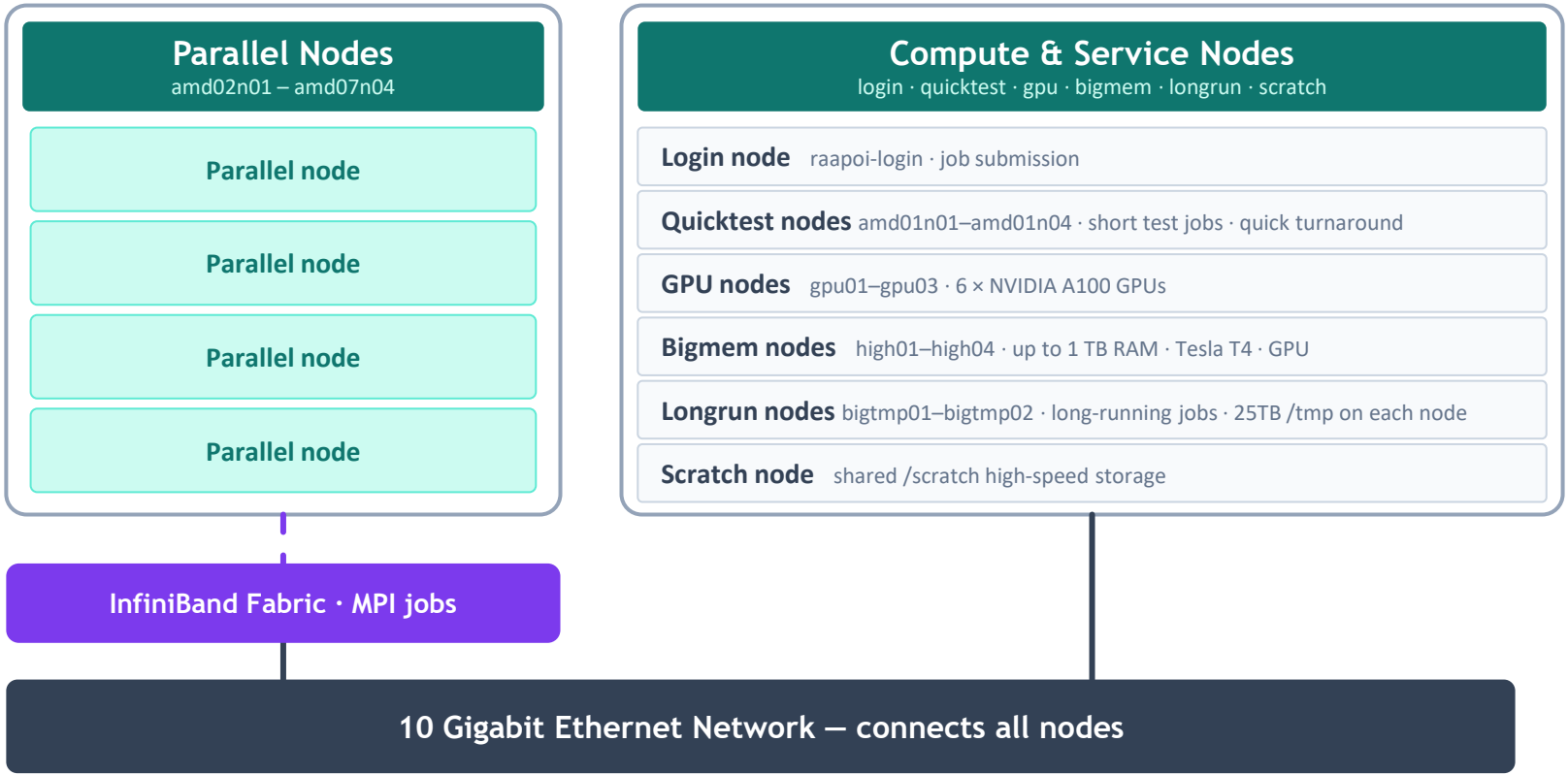
4 parallel nodes





Cluster Network Architecture

Two racks on a shared 10 Gigabit Ethernet backbone; parallel nodes add an InfiniBand fabric for MPI.





Job Scheduling with Slurm

What Slurm does • Interactive sessions • Batch jobs

What is Slurm?

Rāpoi uses Slurm, an open-source job scheduler, to share the cluster fairly between everyone.

Queues Jobs

Your job waits in line until the CPUs, memory and time you asked for are free.

Allocates Resources

Places each job on a compute node in the right partition: quicktest, parallel, gpu, bigmem or longrun.

Enforces Limits

Resources you request are reserved for you, and jobs that exceed their requested memory or time are stopped.

Key Slurm commands

`srun`	`sbatch`	`squeue`	`scancel`
Start an interactive session on a compute node	Submit a script to run in the background	See which jobs are queued or running	Cancel one of your jobs

Two Ways to Run Jobs on Slurm

Interactive jobs: srun

A shell on a compute node: ideal for testing and debugging.

```
$ srun --pty bash
username@amd01n01:~$

# with custom resources
$ srun -J MyTest \
    --cpus-per-task=2 --mem=4G \
    --time=0-05:00:00 --pty bash
```

- Never run programs directly on the login node
- Defaults: 2 CPUs, 2 GB RAM, 1 hour runtime

Batch jobs: sbatch

Run a script in the background: submit it and walk away.

```
#!/bin/bash
#SBATCH --job-name=my_job
#SBATCH --partition=parallel
#SBATCH --cpus-per-task=24
#SBATCH --mem=80G
#SBATCH --time=10-00:00:00

module load COMSOL/6.2.0.415
comsol file.mph

$ sbatch my_job.sh
```

- The job starts when Slurm finds free resources that match your request
- Track it with `vuw-myjobs` or `squeue`

Resource Requests

Other users are prevented from using resources you request — even if you don't use them!

CPUs (N x NN)

```
--ntasks=N
--cpus-per-task=NN
```

Default: 2

If you use more CPUs than requested, your job shares them and runs extremely slowly

Memory

```
--mem=NG
```

Default: 2 GB

If your job exceeds requested memory, Slurm kills it immediately

Time

```
--time=D-HH:MM:SS
```

Default: 1 hour

If your job exceeds wall time, Slurm kills it — request ~20% extra as buffer

Tip: Run a short test job first, then check actual usage with vuw-job-report <jobID>

Managing Jobs

vuw-myjobs (squeue -u <user>)	View your running/pending jobs with resource details
vuw-job-history	Quick view of jobs completed in the last 5 days
vuw-job-report <ID> (sacct -j <ID>)	Detailed report: requested vs. actual memory/CPU usage
scancel <jobID>	Cancel a specific job
vuw-sub filename (vuw-sub –N 1 –P 8 –p comsol64 –q parallel) filename.mph	A script to submit jobs to slurm scheduler in an interactive way

Checking Job Efficiency - Another Example

Always review your job reports to avoid wasting resources!

```
$ vuw-job-report 3338180
```

CPU Efficiency

CPU efficiency is very good ~88%

Memory Efficiency

Requested 34 GB but only used 6.30 GB (18.52%)

Virtual memory efficiency is ~88.5%

* Virt mem can be a mix of RAM-backed memory, mapped shared libraries, memory mapped-files, memory that the program has reserved in its virtual address space, & regions that may not be physically resident in RAM

```
JOB REPORT FOR JOB 3338180

Job ID:              3338180
Cluster:             raapoi
User/Group:          hashmimu/users
State:               COMPLETED
Exit Code:           0:0
Nodes:               1
Requested CPUs:      16
Allocated CPUs:      16
Node List:           high02
Job Wall-clock time: 00:06:48

CPU SUMMARY
-----------
Core Hours:          01:36:00
Total Walltime:      01:48:48
CPU Efficiency:      88.24%
CPU Step:            3338180 (emim_I-C7_T_Cl_b3lyp6311gdp-mwb_gp_freqR)

MEMORY SUMMARY
--------------
Requested Memory:           34.00 GB
Memory Used:                6.30 GB (MaxRSS)
Memory Efficiency:          18.52%
Peak Virtual Memory:        30.09 GB (MaxVMSize)
Virtual Memory Efficiency:  88.50%
MaxRSS Step:                3338180.batch (batch)
MaxVMSize Step:             3338180.batch (batch)
```

Run a short test first, check actual usage, then request ~20% more than needed.



Rāpoi Usage and Demand

Capacity vs demand • Usage by school • Cluster utilisation

Who Is Using Rāpoi? (Jan-Oct 2026)

CPU Usage

~90% of CPU hours used by just four schools

- SCPS: 34.06%
- Robinson Research Institute: 27.12%
- ECS: 22.64%
- ARC: 8.15%

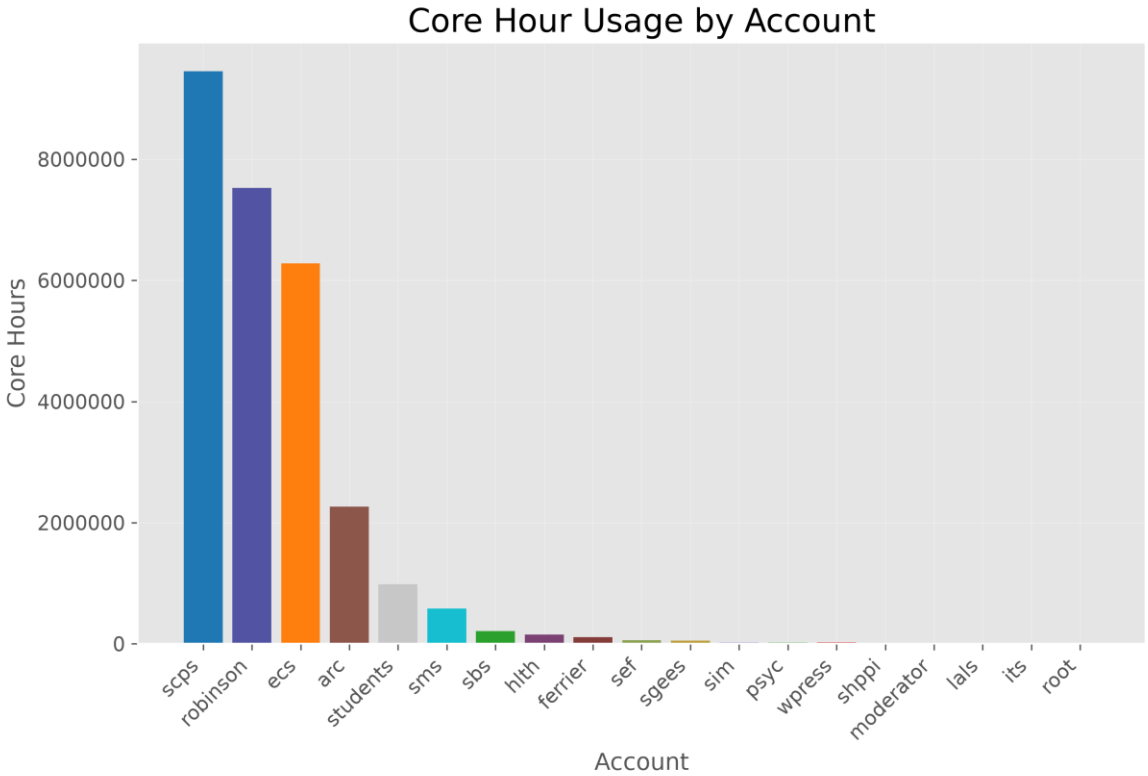
GPU Usage

~90% of GPU hours used by just five schools

- SMS: 32.58%
- ECS: 23.49%
- SEF: 16.28%
- SCPS: 9.05%
- Ferrier Research Institute: 9.02%

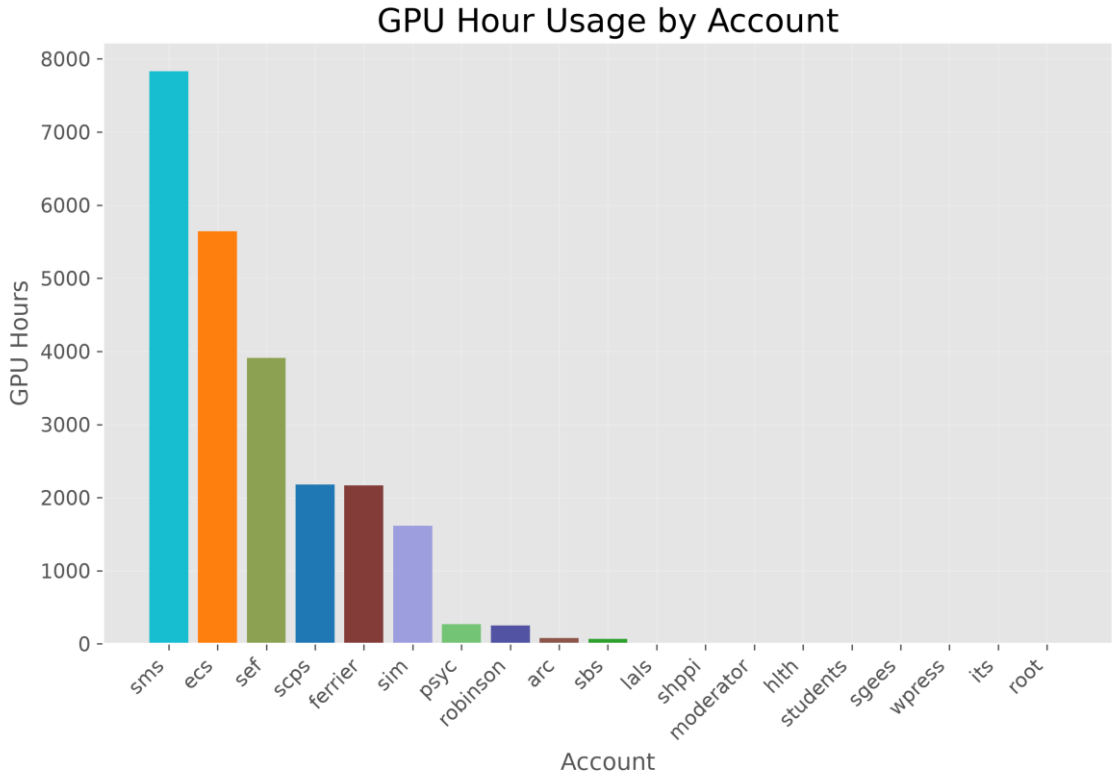
Demand is growing: plan ahead and request only what you need

Rāpoi Usage Stats (Jan-Oct 2026)



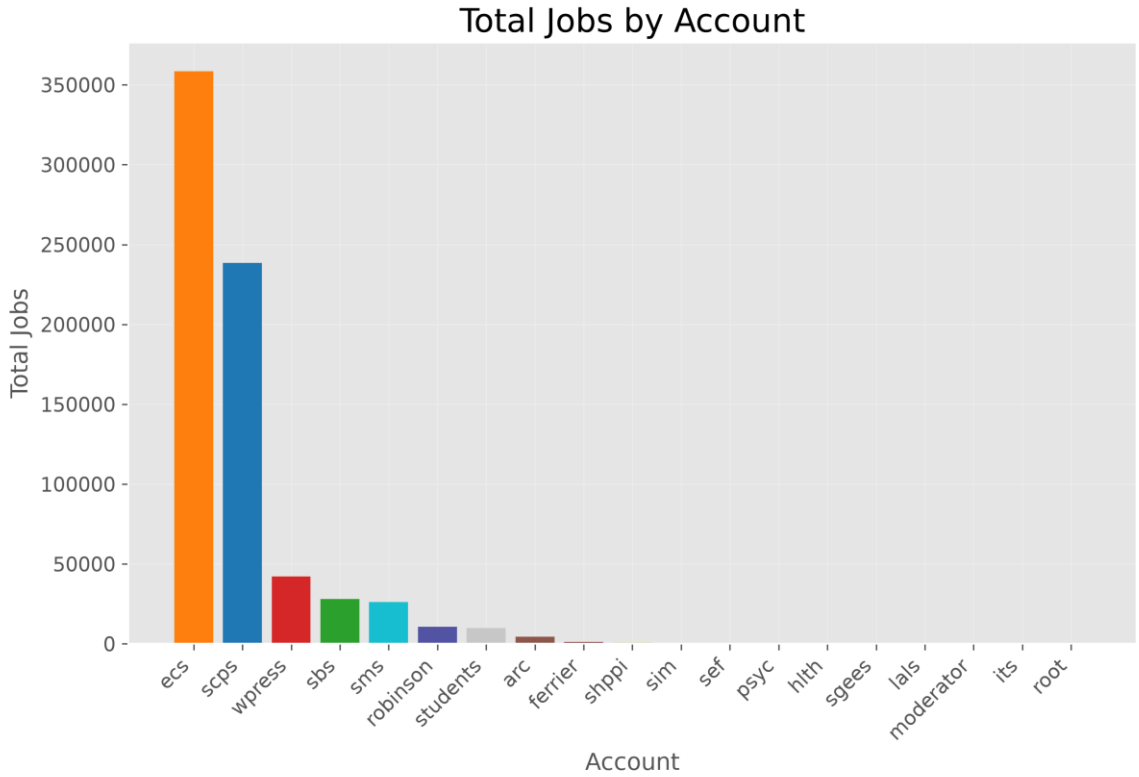
Core Hour Usage by Schools

Rāpoi Usage Stats (Jan-Oct 2026)



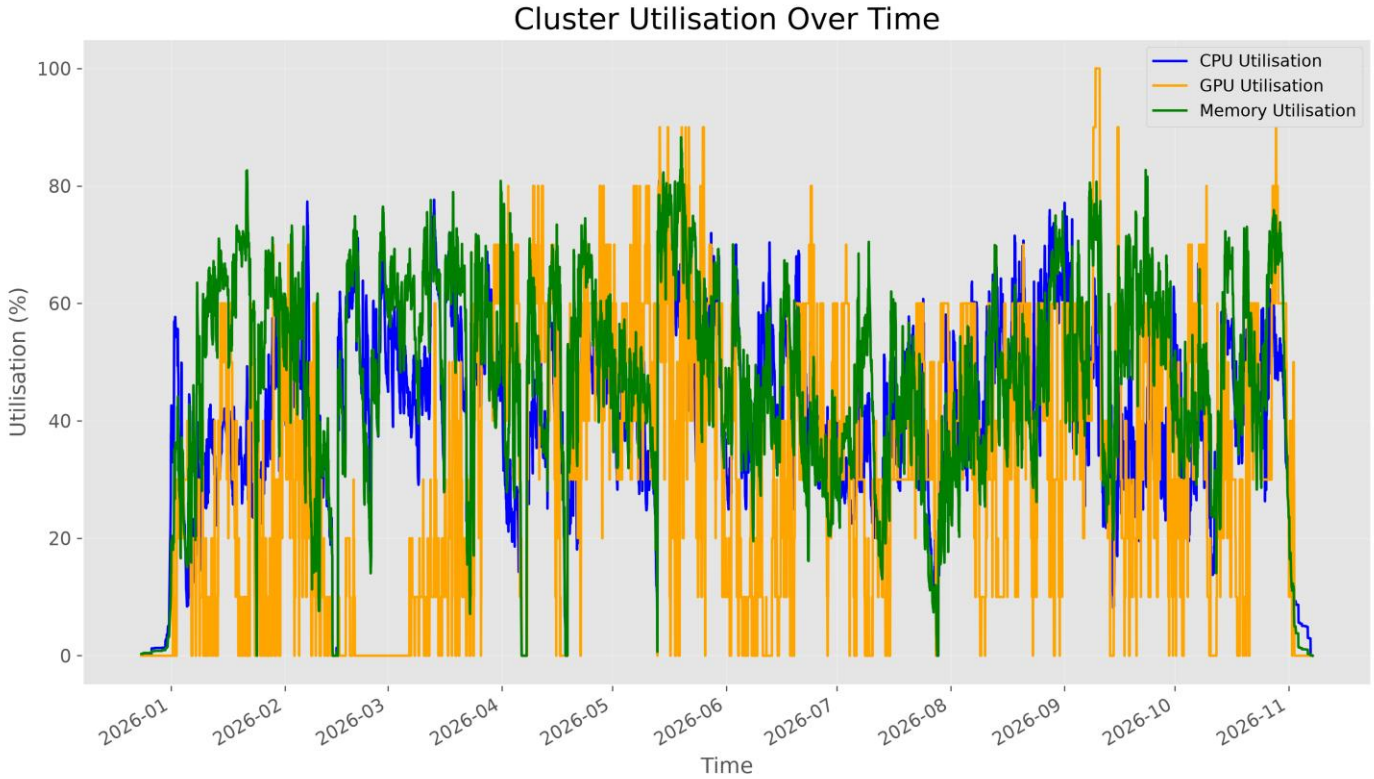
GPU Hour Usage by Schools

Rāpoi Usage Stats (Jan-Oct 2026)



Total Number of Jobs by Schools. 723522 jobs submitted from Jan-Oct

Rāpoi Usage Stats (Jan-Oct 2026)



Overall Cluster Utilisation in 2026

Common Mistakes to Avoid

Don't Do This	Do This Instead
Running programs on the login node	Always use srun or sbatch
Requesting far more resources than needed	Test first, then request ~20% extra
Not checking job reports after completion	Use vuw-job-report to review usage
Storing important data only on scratch	Scratch is not backed up — copy to home or external
Forgetting to clean up /tmp after jobs	Copy results out, remove temp files
Not including module loads in batch scripts	Always load modules in your submit script
Install software locally that already exists	Check what is available using `module spider`

Best Practices Summary

Resources

- Test short jobs first to estimate needs
- Request ~20% more than actual usage
- Review vuw-job-report after every job

Storage

- Code in /home, data in /scratch
- Use /tmp for I/O-heavy operations
- Back up scratch data externally

Jobs

- Include all module loads in scripts
- Use job arrays for parallel tasks
- Set --mail-type for notifications

Etiquette

- Don't hog the login node
- Clean up scratch when done
- Ask on Slack before opening a ticket

Remember: Rāpoi is a shared resource — use it wisely and considerately!

Getting Help

Slack Channel (scan the QR code)

raapoi-help — first stop for questions

Documentation

vuw-research-computing.github.io/raapoi-docs

Service Desk

Account requests & file recovery

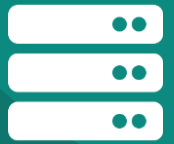
Workshops

I run beginner Rāpoi workshops: contact me to host a workshop on Efficient Use of Rāpoi

Please contact the Rāpoi support team — not Digital Solutions — for HPC issues

SNAP
VUW HUB FOR SIMULATION, NUMERICAL METHODS, ANALYTICS AND PROGRAMMING

Thank You



Questions? Ask us on Slack: raapoi-help