

High-throughput computing (HTC) is growing

HTC: scalable workflows with jobs that have modest compute requirements (often only one or a few CPU cores) such that submissions by all users on the system flow through the scheduler rapidly and efficiently:

- Tens to hundreds of Dask (or Spark) worker jobs
- Single-core analysis submissions using R, Matlab, IDL, Python, NCL, etc...
- Tasks which would be bundled together as “command-file” jobs on Cheyenne (instead submitted as single-CPU jobs on Casper)
- Any of the above configured as part of a job array

Recent improvements on Casper to support HTC

- The CPU-only node count on Casper has been **greatly increased** from 13 to **75!**
- Two additional CPU-only nodes have been added with 1.5 TB each, providing high-memory platform without GPUs

With these new resources, Casper is now the ideal platform at NCAR for running HTC workflows

How do HTC jobs fit into workflows?

Cheyenne

- Large HPC jobs
- Medium CPU MPI jobs
> 4 nodes in job
- Debug & compile jobs

Casper

- GPU jobs of any size
 - CUDA
 - ML/AI
 - Visualization
- Large memory jobs
> 100 GB per node

Why run HTC jobs on Casper?

- ✓ **Casper** has more shared resources than Cheyenne (via its share queue)
- ✓ **Casper** provides resource isolation
- ✓ **Casper** offers higher available memory and NVMe swap as backup
- ✓ **Casper** htc has concurrent use limit of 468 CPU cores vs 18 in Cheyenne share

Known pain points we are working on...

Scheduling jobs between each system natively

- Planning to upgrade PBS version on Cheyenne this summer to enable cross-server submissions and dependencies

Different operating systems and libraries

- Exploring paths forward on this issue

We appreciate any other feedback regarding roadblocks to running high-throughput workflows on Casper

Summary of recent updates and upgrades to Casper

- Migrated from Slurm to PBS for job scheduling
 - Newer version of PBS than on Cheyenne that supports control groups, reserving CPUs, V100 GPUs, and memory for each job
- Migrated JupyterHub to new instance that supports both “persistent login” and batch sessions, and multiple instances
- Switched from 36-job limit to resource-specific limits
- Just this week added a weekend-only queue for large V100 GPU jobs

A lot going on, and we wanted to make sure the Slurm -> PBS transition was clear and well-documented for users...