

# Bon Echo Cluster Access (2026)

---

Access instructions for the Bon Echo (legacy Vaughan) cluster starting from January 2026.

- [Updates to the Bon Echo cluster \(2026\)](#)

# Updates to the Bon Echo cluster (2026)

---

Starting in early 2026, the Scientific Computing is implementing updates to the Bon Echo environment that will affect how users access the cluster and run workflows there.

## Notable changes include:

- The original login nodes (ie. `v.vectorinstitute.ai` and `vlogin.vectorinstitute.ai`) are being retired and replaced.
- All the GPU nodes are getting an OS upgrade to Ubuntu 24. **This means that old Python virtual environments and software packages may no longer work.**
- The system software modules listed under `module avail` (for example `python/3.12.0` or `pytorch2.1-cuda11.8-python3.10`) will be different.
- Many changes to the file system layout.

This document demonstrates all the new login instructions, explains all the changes in detail and shows what to do to update your workflows to work in the new environment.

## Logging Into Bon Echo

---

Log into the new Bon Echo environment by SSH-ing into one of the following login nodes:

- `bonecho.vectorinstitute.ai` (load balancer)
- `blogin01.bonecho.vectorinstitute.ai`
- `blogin02.bonecho.vectorinstitute.ai`
- `blogin03.bonecho.vectorinstitute.ai`

## Updating Your Workflows

---

You'll need to make several updates to your workflow so they work on the upgraded cluster.

### Python Virtual Environments

Any python virtual environments that were created previously using `venv`, `uv`, or `conda` **will not work anymore** on the updated cluster. Similarly, any python environments created on the upgraded Bon Echo cluster will not be backwards compatible to the legacy environment.

You will need to erase old Python virtual environments and recreate them. The following example uses a environment called `my-venv`:

```
# Delete the old environment
rm -rf ~/envs/my-venv

# Create a new environment using an updated python module
module load python/3.12.4
python3 -m venv ~/envs/my-venv
```

```
# Now re-install whatever packages you need
python3 -m pip install torch torchvision pandas
```

## Slurm Configuration

The Slurm cluster has been reconfigured with several changes.

### Time Limits

You must specify a time limit using the `--time=D-HH:MM:SS` or `--time=HH:MM:SS` format. Jobs that do not specify a time limit will get rejected. The maximum time limit is 7 days. Jobs with longer time limits will have access to fewer GPUs. To see the various tiers, run:

```
sinfo --summarize
```

### Requesting Specific GPUs

Previously, to request a specific GPU type you needed to select a partition when submitting a job. *This is no longer supported; specifying a partition will cause your submission to fail.* Instead, use the `gres` flag to select the GPU type and number of resources using the `gres=gpu:<gpu-type>:<num-gpus>` syntax. The following examples show how to request 2x A40 gpus for 1 hour with 4x CPUs and 16G memory:

#### Interactive Jobs:

```
srun --gres=gpu:a40:2 --time=1:00:00 --mem=32G -c 4 --pty bash
```

#### Batch Jobs:

```
#SBATCH --gres=gpu:a40:2
#SBATCH --time=1:00:00
#SBATCH --mem=32G
#SBATCH -c 4
```

### Requesting CPUs

There are no longer any specific flags required to specify CPU-only jobs. Simply omit the `--gres=gpu:...` flag from your submission, and the Slurm scheduler will place it on a CPU-only node.

## System Software Modules

The list of software modules available with `module avail` is different in the new environment. Modules that were available in the old environment (ie. `pytorch2.1-cuda11.8-python3.10`) will not be available anymore, or might be listed under different names.

Use `module avail` to view the new list of modules:

```
coatsworth@blogin02:~$ module avail
-----
----- MPI-dependent avx2 modules
```

```

-----
abyss/2.3.7      (bio)      febio/4.7
kahip/3.16      (D)      netcdf-c++4-mpi/4.3.1      (io)      pcl/1.14.1
(math)      siesta/4.1.5      (chem)
adol-c/2.7.2      ferret/7.6.0      (vis)      lammmps-
omp/20250722      (chem)      netcdf-fortran-mpi/4.6.1      (io)      petsc-64bits/3.21.6
(t)      simnibs/4.1.0
ambertools/23.5      (chem)      fftw-mpi/3.3.10      (math)
libmesh/1.7.5      (math)      netcdf-mpi/4.9.2      (io)      petsc-64bits/3.23.4
(t,D)      slepc-complex/3.20.1
[...]

```

### Warning

The `module avail` command sometimes returns an error message starting with `/cvmfs/soft.computecanada.ca/custom/software/lua/bin/lua: ...anada.ca/custom/software/lmod/lmod/libexec/Cache.lua:340: bad argument #1 to 'next' (table expected, got boolean)`.

This error is related to module caching. Fix it by running `rm -rf ~/.cache/lmod`.

## File System Changes

NAME	OLD LOCATION	NEW LOCATION
Scratch	<code>/scratch/ssd004/scratch/\$USER</code>	<code>/scratch/\$USER</code>
Checkpoint	<code>/checkpoint</code>	N/A, fully removed
Datasets	<code>/datasets</code>	<code>/datasets</code> (same location, but many old datasets have been removed)
Projects	<code>/projects</code>	<code>/projects</code>

## Singularity is now Apptainer

The Singularity project has been renamed to Apptainer (<https://apptainer.org/>). There is no functional difference under the hood, but you'll need to make 2 small changes in your code:

- Replace any Singularity module loads in your shell scripts with: `module load apptainer/1.4.5`
- Change any `singularity` commands in your shell scripts to `apptainer`.

## Support

This guide is maintained by the AI Engineering team. If anything is broken or missing in this document, please contact Mark Coatsworth on Slack or [mark.coatsworth@vectorinstitute.ai](mailto:mark.coatsworth@vectorinstitute.ai).

The Bon Echo cluster is managed by the Scientific Computing team. If you run into any access or technical issues on the cluster, please send an email to [ops-help@vectorinstitute.ai](mailto:ops-help@vectorinstitute.ai).