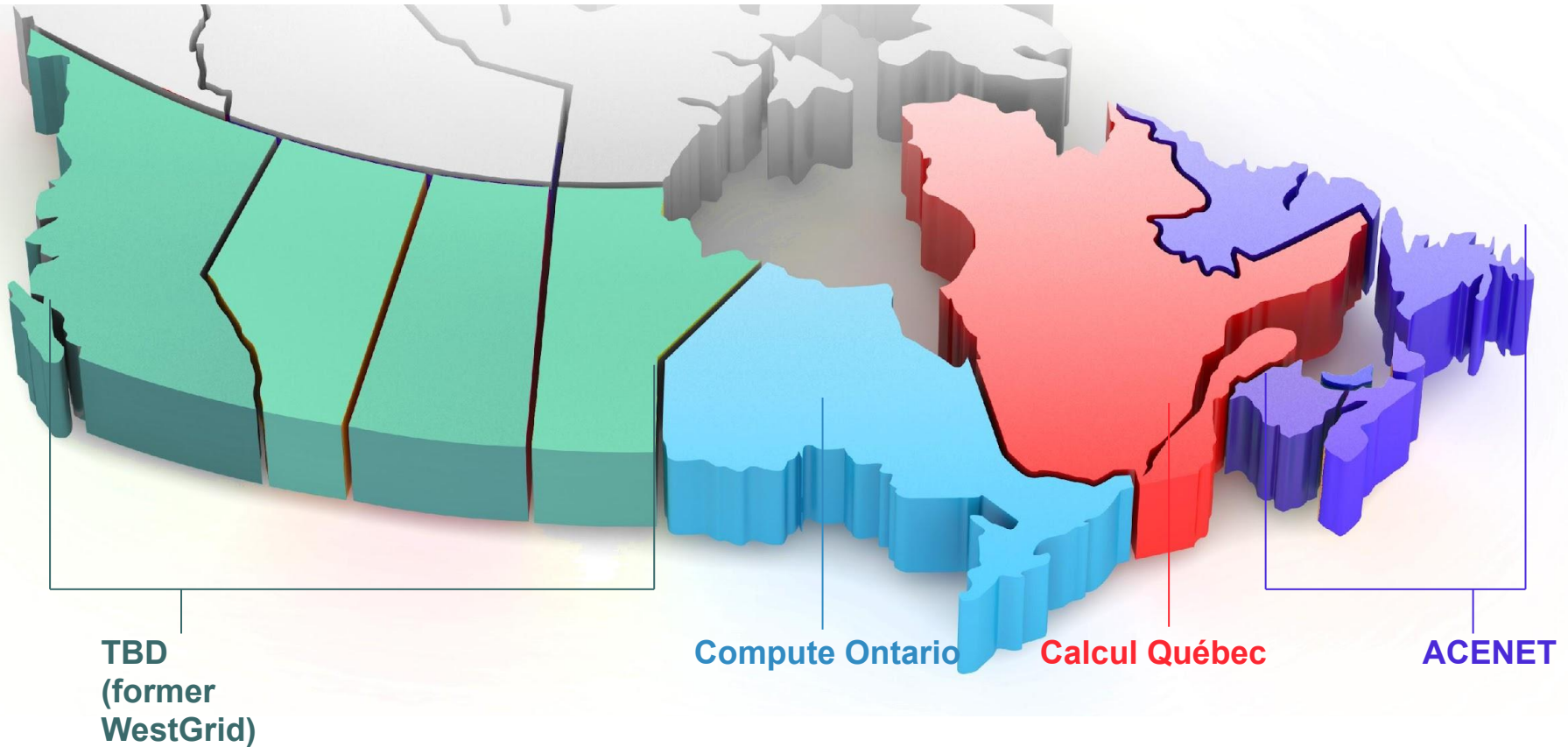


New User Seminar

Single account...



*One can access all national supercomputers
across the country, for free.*



A consortium of 19 Ontario institutions providing advanced computing resources and support...

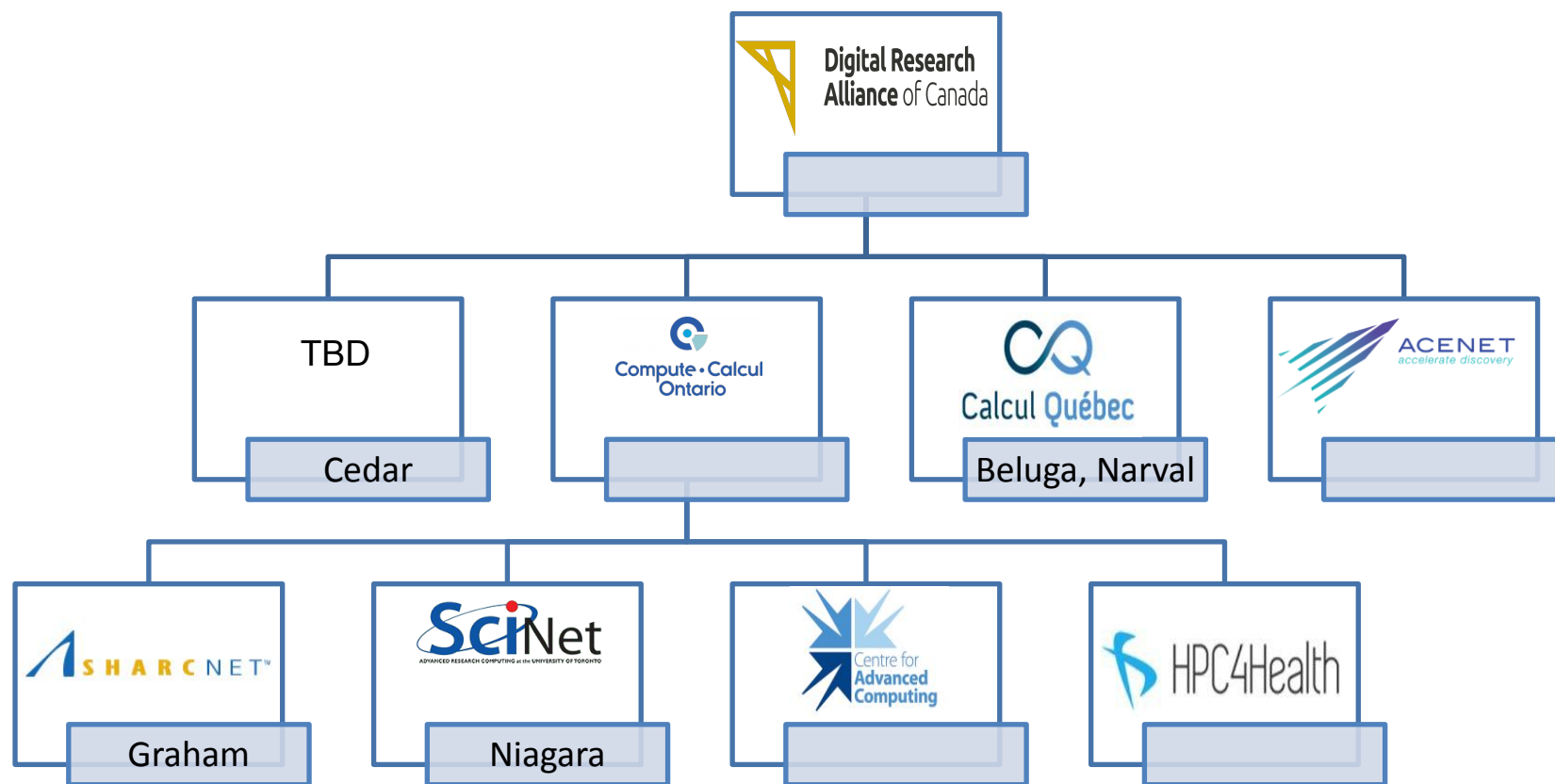
Shared
Hierarchical
Academic
Research
Computing
NETwork



- Member of Alliance and Compute Ontario
- 3,000+ Canadian and international users

- ~50,000 CPU cores
- 500+ GPUs
- 100 Gb/s between nat'l centres

Alliance's Regional Partners



Where to look for information and get help

The screenshot shows a web browser window with the address bar displaying docs.alliancecan.ca/wiki/Technical_documentation. The page title is "Technical documentation". A blue banner states: "This site replaces the former Compute Canada documentation site, and is now being managed by the Digital Research Alliance of Canada." Below this, a message in French says: "Ce site remplace l'ancien site de documentation de Calcul Canada et est maintenant géré par l'Alliance de recherche numérique du Canada." The page includes a search bar, navigation tabs (Main page, Discussion, Read, View source, View history), and a sidebar with links like "Wiki Main Page", "Support", "Getting started", "Getting help", "Running jobs", "Known issues", "System status", "Resources", "Béluga", "Cedar", "Graham", "Narval", "Niagara", "Cloud", "Available software", "The Alliance", "Alliance main page", "CCDB", "Getting An Account", "Acknowledging the Alliance", "Acceptable Use Policy", "Authoring", "Guidelines", "MediaWiki Help", "Recent changes", "Tools", "What links here", "Related changes", "Special pages", "Printable version", "Permanent link", and "Page information". The main content area has a section "Other languages: English • français" and a welcome message: "Welcome to the technical documentation wiki of the Digital Research Alliance of Canada (the Alliance). This is the primary source for users with questions on equipment and services of the Alliance. The focus here is on national services and systems. For documentation on services and systems managed by our regional partners, please use the links provided below. This wiki is a constant work-in-progress and some links might take you to pages which do not yet have content; such pages display like this. Our staff works constantly to improve and expand the available documentation; note however that any of our users is free to add new material and edit existing content." Below this, there are two columns: "Systems and services" and "How-to guides".

Systems and services

- List of current systems
- General-purpose clusters Béluga, Cedar, Graham, Narval
 - System status and upcoming outages
 - Known issues
- Niagara, a cluster designed for large parallel jobs
- Hélios, a GPU cluster
- Available software
- Cloud computing service
- Database servers
- Globus file transfer service
- Nextcloud cloud storage service

How-to guides

- Getting started
 - Getting started with the new national systems (mini-webinar series)
 - Niagara Quick Start Guide
 - Cloud Quick Start Guide
- SSH - How to connect to our compute clusters
- Linux introduction
- Storage and file management
 - Transferring data
 - Scratch purging policy
- Best practices for data migration

<https://docs.alliancecan.ca/>

The screenshot shows the SHARCNET website. The header includes the SHARCNET logo, navigation links (FACILITIES, SUPPORT, ABOUT US), and user links (signed in as bge | sign out, USERS+USAGE, MY ACCOUNT). The main banner features a stylized image of Earth with magnetic field lines and the text "A Forecaster's Guide to Weather in Space" and "Researchers use computational models to better understand solar disturbances of Earth's magnetic environment." Below the banner, there are sections for "EVENTS" and "NEWS".

EVENTS

- Tuesday, September 17
 - 2:00pm New User Seminar
- Tuesday, September 24
 - 2:00pm New User Seminar
- Tuesday, October 1
 - 2:00pm New User Seminar
- Tuesday, October 2
 - 2:00pm New User Seminar

NEWS

- SHARCNET awards Round XIV Dedicated Programming Support
- Career Opportunity: SHARCNET HPC Software Specialist
- SHARCNET Dedicated Programming Support: Round XIV Call fo...
- SHARCNET awards Round XIII Dedicated Programming Support

GitLab Instance at SHARCNET

<https://www.sharcnet.ca/>

Where to look for information and get help

Online events

- New user seminar every Tuesday at 2pm.
- Weekly CO colloquia at noon on Wednesdays.
- Recorded seminars on our youtube channel:
<http://youtube.sharcnet.ca>
- To subscribe to our Events mailing list, send an email to
events+subscribe@sharcnet.ca
- Google for “digital alliance training” to find the calendar for all Alliance events

The screenshot shows the SHARCNET website's Events Calendar for October 2019. The calendar is a grid view with days of the week as columns and dates as rows. Events are listed on specific dates: Tuesday 1st, 8th, 15th, 22nd, and 29th at 2pm (New User Seminar); Wednesday 9th at 12pm (Webinar "Intro"); Wednesday 16th (highlighted in yellow); Wednesday 23rd at 12pm (Webinar "Leve"); and November 1st. The website header includes the SHARCNET logo and navigation links for FACILITIES, SUPPORT, and ABOUT US. The footer indicates the time zone is Eastern Time - Toronto and includes a Google Calendar logo.

Sun	Mon	Tue	Wed	Thu	Fri	Sat
29	30	Oct 1 2pm New User Semi	2	3	4	5
6	7	8 2pm New User Semi	9 12pm Webinar "Intro"	10	11	12
13	14	15 2pm New User Semi	16	17	18	19
20	21	22 2pm New User Semi	23 12pm Webinar "Leve"	24	25	26
27	28	29 2pm New User Semi	30	31	Nov 1	2

Where to look for information and get help

Interactive help

- Ticketing system via support@tech.alliancecan.ca or help@sharcnet.ca
- E-mail us - check staff contact info on <https://www.sharcnet.ca/>
- Phone us
- Office visit*

Use of systems

Installation of software

Access to commercial software and site licence

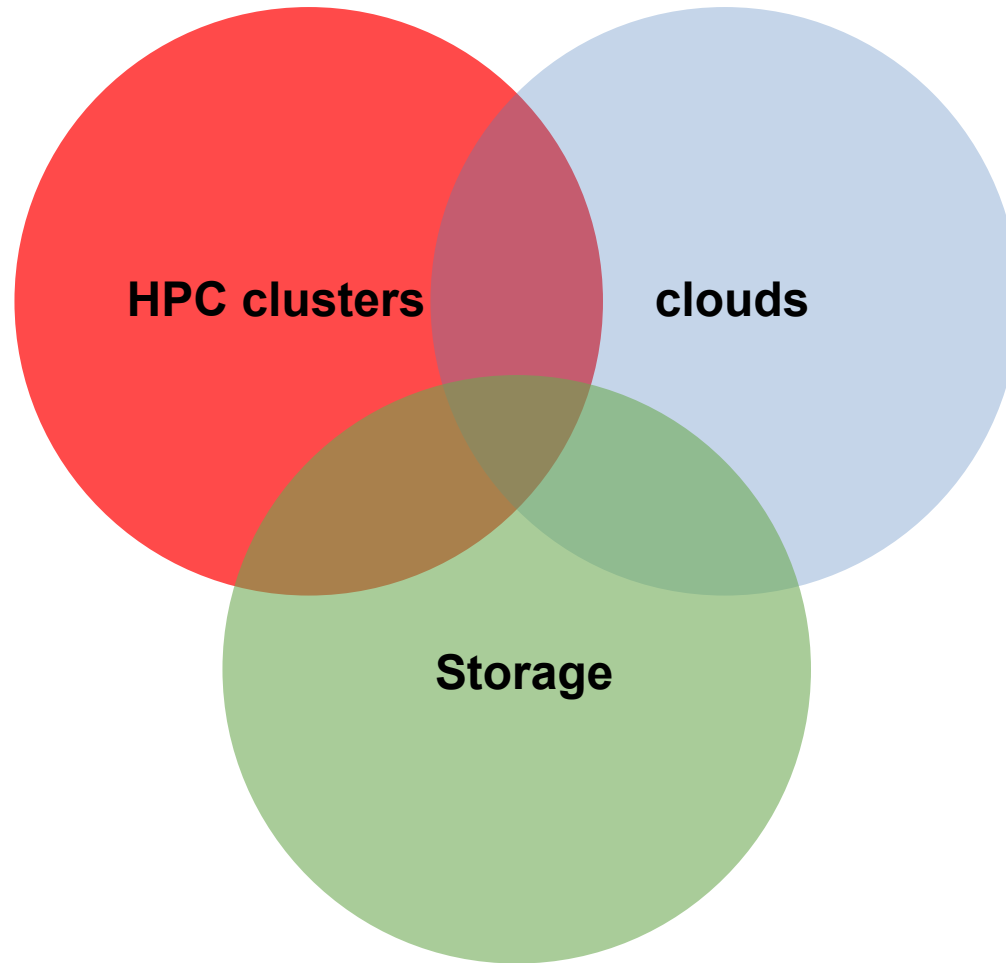
Debugging and optimizing code

Programming

RAC applications

...

The facilities and resources



The facilities and resources

Clusters across the country

- cedar.computeCanada.ca
- graham.computeCanada.ca
- narval.computeCanada.ca
- niagara.computeCanada.ca
- beluga.computeCanada.ca

Cloud services

- arbutus.cloud.computeCanada.ca
- cedar.cloud.computeCanada.ca
- graham.cloud.computeCanada.ca
- east.cloud.computeCanada.ca

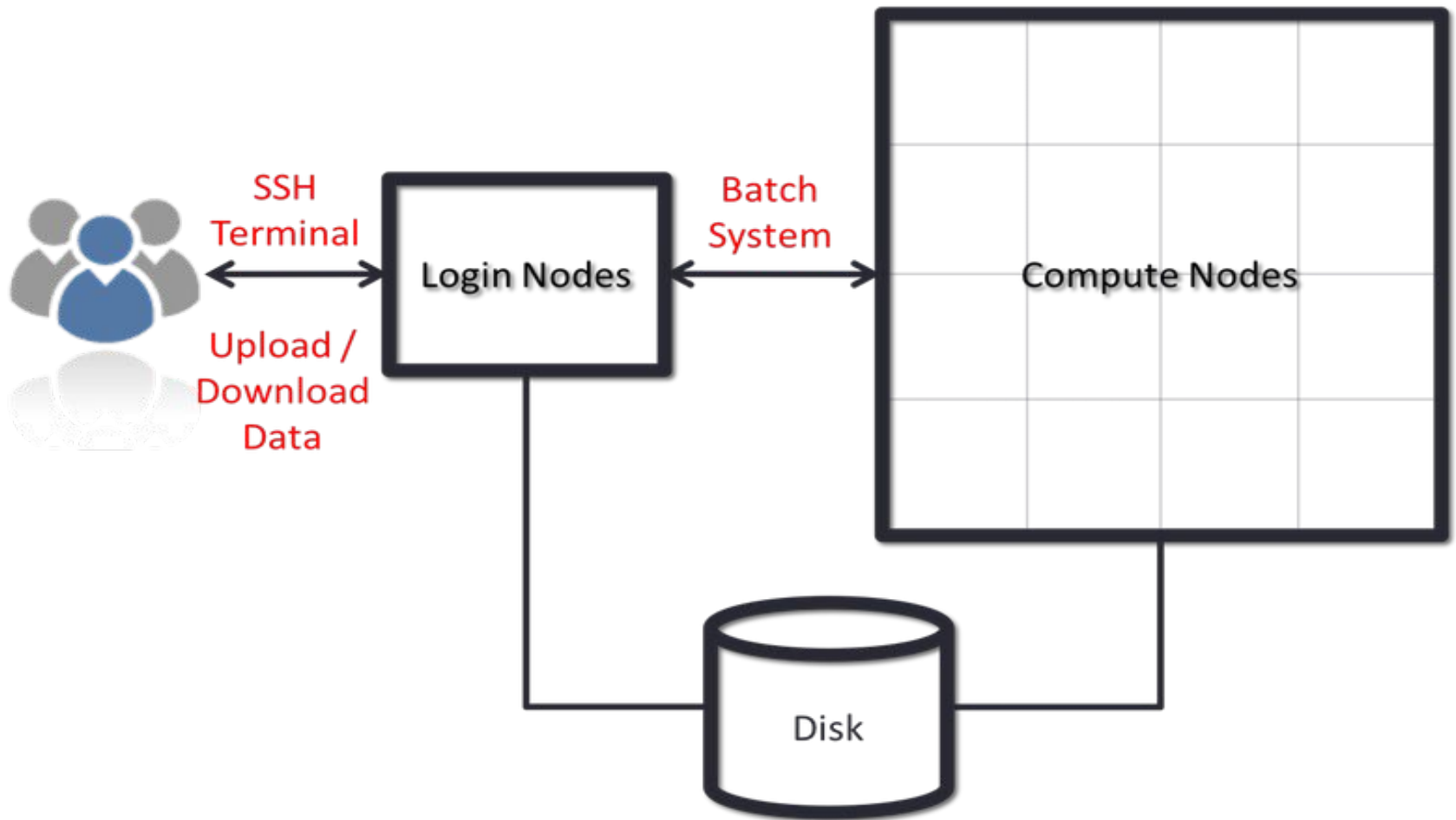
NB. Files are NOT shared across systems. You need move and copy files around as needed.

The computing environment

Cluster computing environment

- OS: 64-bit Linux - CentOS etc.
- Languages: C/C++, Fortran, R, Python, Matlab/Octave, Java, Julia, CUDA, etc.
- Access to a variety of software packages
- Parallel development support:
 - **MPI**, **OpenMP**, Pthreads, **CUDA**, OpenACC, OpenCL, DDT
 - **C++**: Language support for multithreading (since C++-11 standard)
 - **Fortran**: Language support for parallel programming (since 2003 standard)
 - **Julia**: Parallel processing constructs, shared and distributed objects
- Data science support:
 - R, Python, Julia, Spark, DASK, etc.
- Batch computing via slurm

Login nodes vs. compute nodes



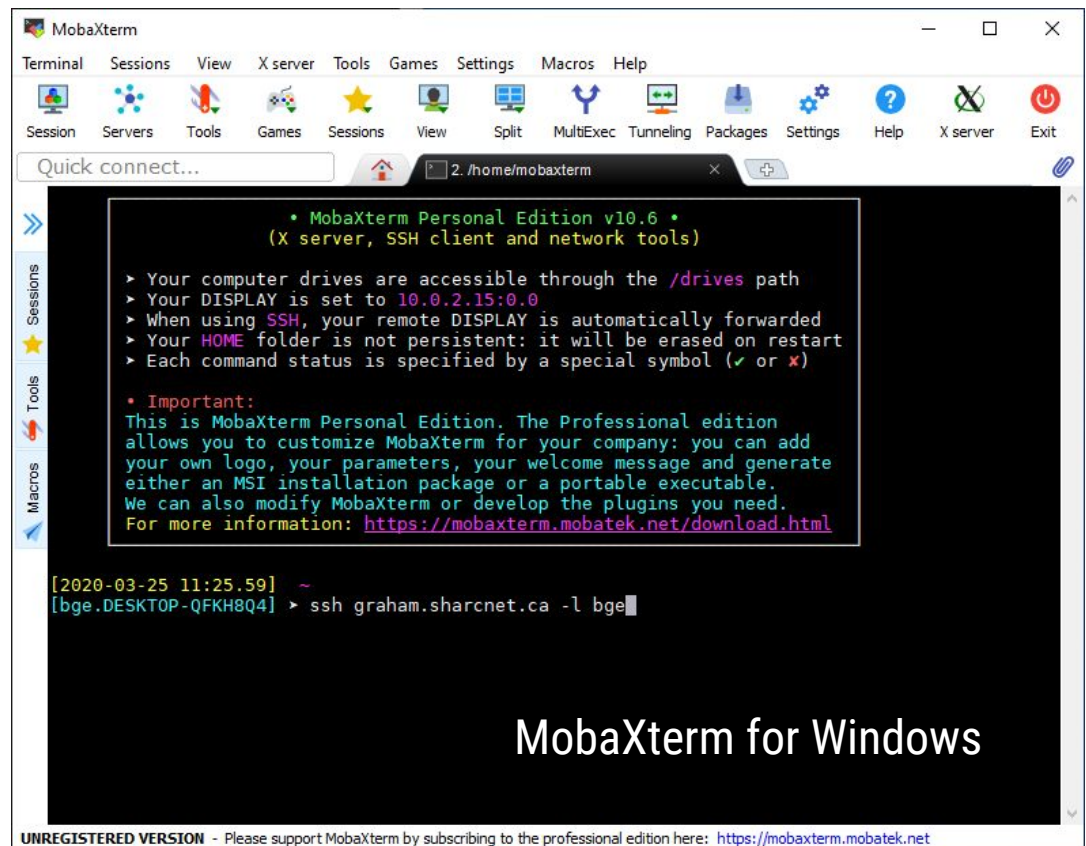
Connecting to clusters via SSH

For Windows users we recommend free software **MobaXterm**. It combines:

- SSH client (to login to systems)
- SFTP client (to copy files)
- Xwindow server (to run graphical applications)

Linux and Mac users can use command line tools ssh, scp, rsync etc.

For Mac only, running graphical applications remotely requires free software **XQuartz**.

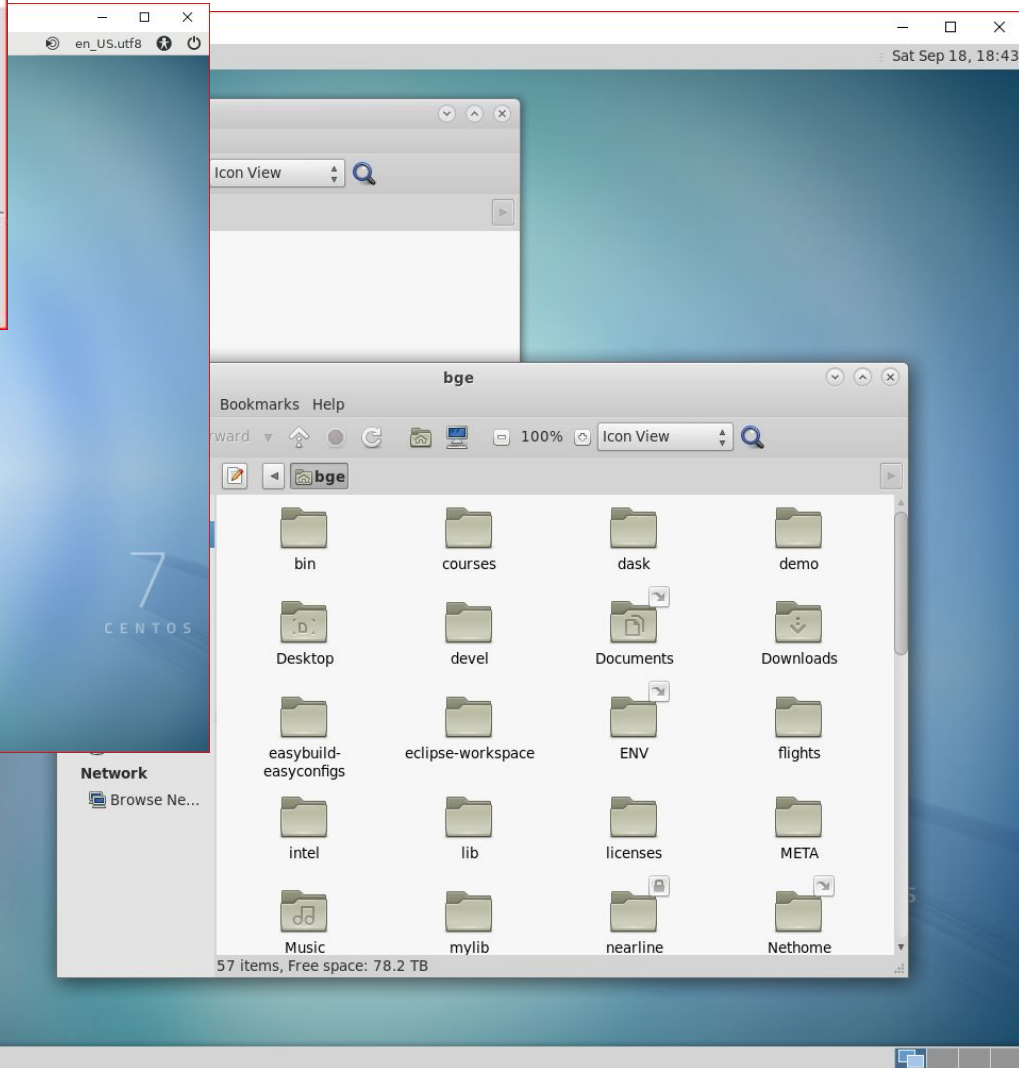
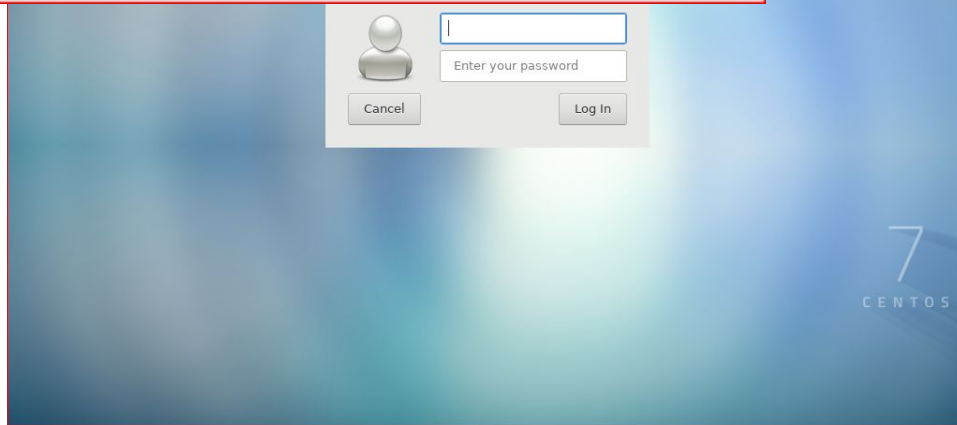


Transferring large amounts of files using Globus

Transferring data between your computer and clusters and between clusters via Globus (see <https://docs.alliancecan.ca/wiki/Globus>):

- Uses different protocols, generally has higher throughput.
- Web based.
- Enables file transfer between any two computers, known as end-points.
- Local computer needs to run Globus Connect Personal for transferring files between the local computer and remote end-point.
- Allows non Alliance users share data (via arrangements).

Connecting to GUI desktop



Accessing and managing files

Everyone has the access to the following file systems

- **/home**: 50G, 0.5m files; **backed up** regularly
- **/project**: 1T (extendable to 10T) per group, 0.5m files; **backed up**.
- **/scratch**: 20T per user, 1m files, up to 100T; 2 months of life.
- **nearline**: to store files not currently in use, but may be needed later.
5000 files (approx. 10T) per group. **NOT available on compute nodes**.

*NB: Please **DO NOT** store everything, remove the files no longer in use to save space.*

Tips for project space

- Do not preserve file permissions when copying files to Project! If you do, you will likely get a “Not enough of disk space” error on Project.
- Specifically, **don't do** the following when project is the destination, and source is not in project:

```
$ cp -p
```

```
$ mv
```

```
$ rsync -a
```

Tips for **nearline** space

- Quota limit is an “ingest” limit. After it has been consumed, you can add more files up to your file count limit.
- Find your file count by typing:
find /nearline/YOUR_ACCOUNT/ | wc -l
- Don't edit files in nearline! If it is on tape, your editor might hang.
- File size, some guideline:
 - At least 10G to 20G per file. Ideal file size is 100G to 500G. Up to 2T per file is acceptable.

What software packages are available?

Available software - CC Doc

docs.alliancecan.ca/wiki/Available_software

English Alliance CCDB Log In

Page Discussion Read View source View history Search CC Doc

Available software

This site replaces the former Compute Canada documentation site, and is now being managed by the Digital Research Alliance of Canada.

Ce site remplace l'ancien site de documentation de Calcul Canada et est maintenant géré par l'Alliance de recherche numérique du Canada.

Other languages: English • français

A current list of the software available on Compute Canada national systems is below. This list changes frequently as new software is added. You can request the installation or updating of a particular program or library by contacting **Technical support**. If you wish to use the Compute Canada software environment on your own system, please see **accessing CVMFS**.

Contents [hide]

- 1 Notes
 - 1.1 Niagara
- 2 List of globally-installed modules
- 3 Site-specific software

Notes

Except for basic system programs, you access most software by loading a **module**. See [Using modules](#) for more on how to use the Lmod module system. Note that some prerequisite modules are loaded by default.

Here are a few things to know about the available software:

- Most **Python** modules are not installed as (Lmod) modules. They are instead provided as binary **wheels**, stored on the Compute Canada systems under `/cvmfs/soft.computeCanada.ca/custom/python/wheelhouse/`. One such package is **TensorFlow**. For instructions on how to install or list Python packages, see the **Python** page.
- Similarly, most **R** or **Perl** packages are not installed either. We recommend installing them in your personal or group file space. See the **R** and **Perl** pages for instructions on how to do so.
- A page discusses **symbolic algebra software** like Mathematica and Sage.
- Note that **Docker** is not available on Compute Canada clusters but **Singularity** is available by loading the module `singularity`. Docker containers can be converted to Singularity as discussed [here](#).
- Some of the software packages listed below are not immediately usable because they require you to have a license. You may need to be granted access to them by us. Attempting to load the module for one of these will give you instructions on what to do to obtain access.

Check for software while on a cluster

\$ module avail

\$ module spider keyword

Running jobs using a slurm script - *myjob.sh*

Submitting a serial job

```
#!/bin/bash
#SBATCH --time=00-01:00:00 # DD-HH:MM
#SBATCH --account=def-user
module load python/3.6
python simple_job.py 0 output 10
```

To see what account groups you have access to, use command **sshare -U** or **salloc** by itself

sbatch *myjob.sh*

Submitting a series of jobs

```
#!/bin/bash
#SBATCH --time=01:00
#SBATCH --account=def-user
#SBATCH --array=1-200

python simple_job.py $SLURM_ARRAY_TASK_ID output
```

META package

https://docs.alliancecan.ca/wiki/META: A_package_for_job_farming

(Google for “meta digital alliance”)

A more universal and convenient way to manage a large number of similar jobs.

Running jobs using a slurm script - *myjob.sh*

Submitting a threaded job

```
#!/bin/bash
#SBATCH --account=def-user
#SBATCH --time=0-03:00
#SBATCH --cpus-per-task=32
#SBATCH --ntasks=1
#SBATCH --mem=20G

export \
  OMP_NUM_THREADS=$SLURM_CPUS_PER_TASK

./myprog.exe
```

sbatch *myjob.sh*

Submitting a parallel job

```
#!/bin/bash
#SBATCH --account=def-user
#SBATCH --time=5-00:00
#SBATCH --ntasks=100
#SBATCH --mem-per-cpu=4G

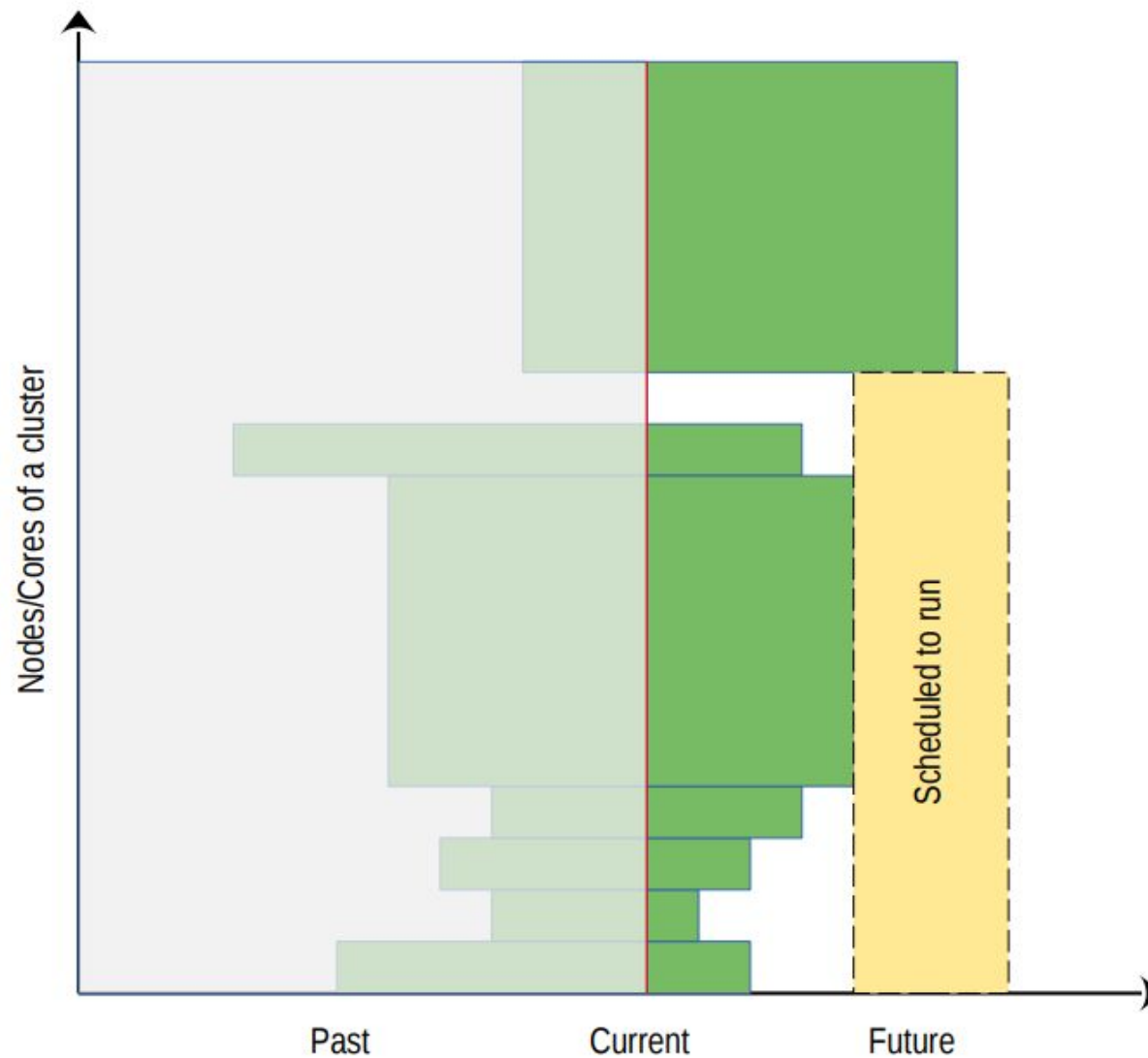
srun ./mympprog.exe
```

Viewing your jobs

Commonly used slurm commands

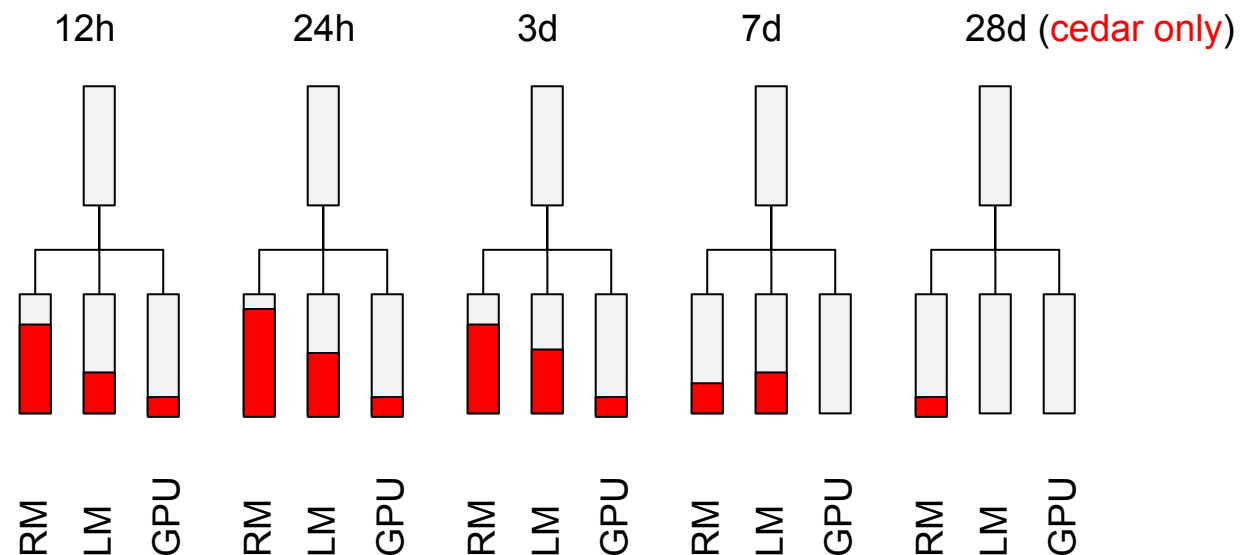
- `squeue -u you`
- `sacct`
- `scancel`
- `seff`

How scheduler works



Why my jobs don't start

On entry, each of your jobs is “placed in one of the partitions” by its attributes (cores, memory, runtime, etc.) you specified



You may use command **partition-stats** to see the system status.

What can be done about wait times

- Requesting more resources (runtime, CPU cores, memory) than what the job process requires can result in a longer queue times.

Tip: request only what the job needs, with a bit of leeway for time and memory.

- The recent usage of an account is calculated independently on each of the Alliance general purpose systems (Graham, Beluga, and Cedar) and the availability of the resources varies across systems.

Tip: use multiple systems when appropriate.

- More resources are available to full-node jobs. If your job can efficiently use multiples of 32 cpu cores (graham) it gains access to a larger set of nodes if it is submitted as a full-node job.

Tip: use --nodes=N and --ntasks-per-node=32 sbatch arguments for full-node jobs.

- Less than 20% of all resources are available via default accounts.

Tip: If a project needs more than the default level usage, a larger target share of the system can be obtained through the annual Resources Allocation Competition (RAC)

Common mistakes to avoid

- Do not run significant programs on login nodes, nor run programs directly on compute nodes.
- Do not specify a job run time blindly (say, 7 days), or more memory than needed for your program
- Try not to create millions of tiny files, or large amounts (> GB) of uncompressed (eg. ASCII) output
- Do not let your jobs access (read/write) files frequently (more than 10 files per second from all of your running jobs).

Q&A

