

# Table of Contents

<b><u>Computing Policies</u></b> .....	<b>1</b>
<u>Pleiades Front-End Usage Guidelines</u> .....	1
<u>Columbia Front-End Usage Guidelines</u> .....	3

# Computing Policies

## Pleiades Front-End Usage Guidelines

**Summary:** Use the Pleiades front-end systems (PFEs) and the bridge nodes for file editing, compiling, short debugging/testing sessions, and batch job submissions.

---

The PFEs and the bridge nodes are the front-end systems to Pleiades. They provide an environment that allows you to get quick turnaround while performing file editing, file transferring, compiling, short debugging/testing sessions, and batch job submission via PBS to a subset of the Pleiades compute nodes.

**WARNING:** The new Pleiades front-ends (pfe[20-27]) use the Intel Sandy Bridge processors. If you use a PGI compiler to build your executable, be aware that by default the executable is optimized for Sandy Bridge and will not necessarily execute on Harpertown, Nehalem-EP, or Westmere processors. To generate a single executable that will work on all Pleiades processor types, use the option

`-tp=penryn-64,nehalem-64,sandybridge-64` during compilation with PGI compilers. See [PGI Compilers and Tools](#) for more information.

You cannot "ssh" to the compute nodes except for the subset of nodes your PBS job is running on.

The bridge nodes are recommended for the following functions:

### Pre- and/or Post-Processing

The large amount of memory on the bridge nodes allows pre- and post-processing applications such as [Tecplot](#), [IDL](#), and [Matlab](#) to run faster than on the PFEs. Note that the bridge nodes have the same software as the PFEs. For a list of available applications, run the command `module avail`.

### File Transfers Between Pleiades and Columbia

Both the Pleiades Lustre filesystems `/nobackup` and the Columbia CXFS filesystems `/nobackup` are mounted on the bridge nodes. To copy files between the Pleiades Lustre and Columbia CXFS filesystems, log into a bridge node and use the `cp`, `mcp`, or `shiftpc` command to perform the transfer.

### File Transfers to Mass Storage

The Pleiades /nobackup filesystems are mounted on Lou2. Thus, the easiest way to transfer files between Pleiades and Lou2 is to initiate a command such as **cp**, **mcp**, **tar**, or **shifc** on Lou2. For example:

```
lou% mcp /nobackup/username/foo $HOME
```

If you initiate the transfer on Pleiades, the commands **scp**, **bbftp**, **bbscp**, and **shifc** are available to do the transfers between a Pleiades front-end or bridge node and Lou. Since **bbscp** uses almost the same syntax as **scp**, but performs faster than **scp**, we recommend using **bbscp** in cases where you do not require the data to be encrypted. For very large file transfers, we recommend the Shift utility, developed at NAS.

See also [File Transfer Overview](#), and [File Transfer Commands](#).

File transfers from the Pleiades compute nodes to Lou must go through one of the PFEs or bridge nodes first. See [Streamlining File Transfers from the Pleiades Compute Nodes to Lou](#) for more information.

When sending data to Lou, keep your largest individual file size under 1 TB, as large files will occupy all of the tape drives, preventing other file restores and backups.

## Additional Restrictions on Front-end Systems

- No MPI (Message Passing Interface) jobs are allowed to run on the PFEs or the bridge nodes
- A job on bridge[1-2] should not use more than 56 GB; when it does, a courtesy email is sent to the owner of the job
- A job on bridge[3-4] should not use more than 192 GB; when it does, a courtesy email is sent to the owner of the job

Before starting a large-memory session, it is a good idea to check to make sure there is enough memory available. You can run the command **top**, hit "M", and check under the "RES" column for other large memory applications that may be running.

# Columbia Front-End Usage Guidelines

The front-end system, cfe2, provide an environment that allows users to get quick turnaround while performing the following: file editing, file management, short debugging and testing sessions, and batch job submission to the compute systems.

Running long and/or large (in terms of memory and/or number of processors) debugging or production jobs interactively or in the background of cfe2 is considered to be inconsiderate behavior to the rest of the user community. If you need help submitting such jobs to the batch systems, please contact a the Control Room at (650) 604-4444 or (800) 331-USER or send e-mail to: support@nas.nasa.gov

Jobs that cause significant impact on the system load of the Columbia front-end machine (cfe2) are candidates for removal in order to bring the front-end systems back to a normal and smooth environment for all users. A cron job regularly monitors the system load and determines if job removal is necessary. The criteria for job removal are described below. Owners of any removed jobs will receive a notification e-mail.

1. To be eligible for removal, the number of processors a front-end interactive job uses can be one (1) or more. Exceptions to this are those programs, utilities, etc. common to users and/or NASA missions that are listed in an "exception file". Examples of these would be: **bash**, **cp**, **cs**, **em**, **gz**, **rs**, **sc**, **sf**, **sh**, **ss**, **ta**, and **tc**. Users can submit program names to be added to this exception file by mailing requests to: support@nas.nasa.gov.
2. For qualifying processes, the CPU time usage of each process in a job has, on the average, exceeded a threshold defined as: (20 min x 8 / number of processes for the job). That is, a baseline for removal is a job with 8 processors running for more than 20 minutes. The maximum amount of time allowed for each processor in a job is scaled using the formula: 20 min x 8 cpu / number-of-processes. Therefore, the following variations are possible:
  - ◆ 160 minutes = (20 \* 8) / 1 cpu
  - ◆ 80 minutes = (20 \* 8) / 2 cpu
  - ◆ 40 minutes = (20 \* 8) / 4 cpu
  - ◆ 20 minutes = (20 \* 8) / 8 cpu
  - ◆ 10 minutes = (20 \* 8) / 16 cpu
  - ◆ 5 minutes = (20 \* 8) / 32 cpu
  - ◆ 2.5 minutes = (20 \* 8) / 64 cpu

The conditions of removal are subject to change, when necessary.