

PGI Compilers and Tools

Category: Compilers

Summary: PGI compilers and tools are installed as modules under the system directory /nasa on Pleiades. By default, the PGI compilers generate code that is optimized for the compilation host.

PGI Compiling

As an option to using the Intel compilers, you can use the compilers and program development tools from PGI ([The Portland Group, Inc.](http://www.pgroup.com)) for Fortran, C, and C++. Several versions of the PGI compilers and tools are installed under the system directory /nasa on Pleiades as modules. To see what is available, do:

```
pfe20% module avail comp-pgi
comp-pgi/10.6 comp-pgi/11.0 comp-pgi/11.6
comp-pgi/12.3 comp-pgi/12.4 comp-pgi/12.5
```

The following command will load version 12.5:

```
pfe20% module load comp-pgi/12.5
```

Newer versions may be installed under `/u/scicon/tools/modulefiles`. Use the following command to see what is available.

```
pfe20% cd /u/scicon/tools/modulefiles
pfe20% ls -l /u/scicon/tools/modulefiles/pgi*
pfi_11.10
pfi_12.4
pfi_12.6
pfi_12.8
```

Follow the example below to load version 12.8:

```
pfe20% module load /u/scicon/tools/modulefiles/pfi_12.8
```

Several MVAPICH2 modules built with the PGI compilers are also available under `/u/scicon/tools/modulesfiles`; some have MPI CUDA calls enabled and some do not. Both will work with MPI codes with sections compiled to run on the [GPU](#), but the modules with `_cuda_` in their names also allow some MPI calls to use the GPU memory. See [this MVAPICH web page](#) for more information. At the time of publication the two latest modules are:

mvapich2_1.8_pgi_12.8
mvapich2_1.8_pgi_12.8_cuda_4.1

Future versions will probably be installed under the system modules tree /nasa, but not under the /u/scicon tree.

Using the PGI Compilers

PGI provides various commands for different languages or functions, as shown in this table.

Command	Language or Function
pgfortran	PGI Fortran
pgf77	Fortran 77
pgf90 or pgf95	Fortran 90/95/F2003
pghpf	High Performance Fortran
pgcc	ANSI C99 and K&R (Kernighan and Ritchie) C
pgCC or pgcpp	ANSI C++ with cfront features
pgdbg	Source code debugger (supports OpenMP and MPI)
pgprof	Performance profiler (supports OpenMP and MPI)

NOTE: By default, the PGI compilers generate code that is optimized for the type of processor on which compilation is performed (the compilation host). Be aware that the processors on Pleiades are forward-compatible, but not backward-compatible. Thus a code compiled and optimized on a newer-generation processor, such as Sandy Bridge, will not necessarily execute correctly on previous-generation processors, such as Harpertown, Nehalem-EP, or Westmere. This could be an issue if you compile and optimize your code on the new Pleiades Sandy Bridge front-end nodes (pfe[20-27]) or the Sandy Bridge compute nodes (through a PBS session) and later want to use the same executable to run on the Harpertown, Nehalem-EP, or Westmere nodes.

If you want to build an executable that targets a specific processor type on Pleiades, use the `-tp` flag:

<code>-tp= sandybridge-64</code>	Intel SandyBridge architecture Core processor, 64-bit mode
<code>-tp= nehalem-64</code>	Intel Nehalem architecture Core processor, 64-bit mode (including Nehalem-EP and Westmere)
<code>-tp= penryn-64</code>	Intel Penryn architecture Pentium processor, 64-bit mode (including Harpertown)

TIP: Using the `-tp=penryn-64,nehalem-64,sandybridge-64` option will generate a single executable where the code is optimized for the Intel Penryn (Harpertown), Nehalem (Nehalem-EP and Westmere), and Sandy Bridge architectures. The choice of which optimized copy to execute is made at run time depending on the machine executing the code.

PGI recommends that for best performance on processors which support SSE instructions (including all Pleiades processor types), use `pgfortran`, even for FORTRAN 77 code, use the `-fastsse` option.

For more information about the PGI compilers, see the **pgfortran**, **pgcc**, **pgCC man pages** or use the command **pgfortran -help**, **pgcc -help**, or **pgCC -help**. Information about the PGI debugger and performance analysis tool can be found in the **pgdbg** and **pgprof man pages**.

Article ID: 365

Last updated: 18 Dec, 2012

Computing at NAS -> Porting & Developing Applications -> Compilers -> PGI Compilers and Tools

<http://www.nas.nasa.gov/hecc/support/kb/entry/365/?ajax=1>