

Compilers and related tools

Apache ANT Java development software

Apache Ant is a Java library and command-line tool whose mission is to drive processes described in build files as targets and extension points dependent upon each other. The main known usage of Ant is the build of Java applications. ANT is accessed via Modules.

```
> module load ant
> module load ...your_needed_java_ver...
```

Eclipse IDE

Eclipse is a multi-language software development environment comprising an integrated development environment (IDE) and an extensible plug-in system.

Load the following:

```
> module load java/1.6.0_64bit
> module load eclipse/1.4.1-x86_64
```

To run on the login node:

```
> eclipse
```

or on a compute node:

```
> bsub -lp -q int_public6 eclipse
```

Python Compiler

Installed python modules: matplotlib, numpy, Networkx, Biopython

To check docs online while logged into cluster:

```
> pydoc matplotlib
```

Complete [Python docs](#)

A guide to Python [Modules](#)

Perl Compiler

Perl is a stable, cross platform programming language. Perl is extensible. There are over 500 third party modules available from the Comprehensive Perl Archive Network ([CPAN](#)).

For perl debug tools and examples see [this link](#).

Portland Compilers

Portland Group compilers are available for use on the cluster. Fortran, C and C++ compilers and development tools enable use of networked compute nodes of Intel x64 processor-based workstations and servers to tackle serious scientific computing applications. PGI compilers offer world-class performance and features including auto-parallelization for multi-core, OpenMP directive-based parallelization, and support for the PGI Unified Binary™ technology.

Portland products are not part of the default environment on the head node, but they can be accessed by use of the module command. Under modules Portland products are listed as: pgi

Portland documentation is on the vendor website or on the cluster in the install tree found at: /opt/pgi/

MPI related

OpenMPI, Mvapich and Mvapich2 are available on the cluster as a loadable module. Once the corresponding module is loaded, your environment will provide access to the various MPI compilers.

```
> module load openmpi
```

For example, OpenMPI provides the following:

```
mpic++ mpicxx mpicc mpiCC mpif77 mpif90
```

Likewise for mvapich and mvapich2.

Java

The cluster has jdk version 1.6.0_07 for x86_64 hardware installed and under module control.

For command line options:

```
-bash-3.2$ java -h
```

Local documentation via man pages:

```
-bash-3.2$ man java
```

For useful troubleshooting and other Java docs:

Tuning

Docs

GCC (C, C++, Fortran) compilers

The cluster 64-bit login node requires Gnu GCC 64-bit compiler and as a result becomes the default native compiler. No Module setup is required. Documentation is available at GCC online documentation or from the following man pages:

```
> man gcc
> man g77
> man gfortran
```

Note, the system default install of **gcc** is ver. 4.1. This can be an issue for some types of dependencies. For example, the gfortran option for **openmp** usage requires **gcc** ver. 4.7 or newer. You may access this version by the following:

```
> module load gcc
```

G95

G95 is a stable, production Fortran 95 compiler available for multiple cpu architectures and operating systems. There are two versions on the cluster. The module command makes a distinction. Using "module load g95" gets the 32-bit version and "module load g95-64" gets the 64-bit version.

Valgrind Memory analysis and profiling tool

The Valgrind distribution currently includes six production-quality tools: a memory error detector, two thread error detectors, a cache and branch-prediction profiler, a call-graph generating cache profiler, and a heap profiler. It also includes two experimental tools: a heap/stack/global array overrun detector, and a SimPoint basic block vector generator. Valgrind is available on the cluster login/headnode only.

Yap compiler

YAP is a high-performance Prolog language compiler.

Lisp compiler

Steel Bank Common Lisp (SBCL) is an open source (free software) compiler and runtime system for ANSI Common Lisp. It provides an interactive environment including an integrated native compiler, a debugger, and many extensions.

cmake build tools

CMake is a cross-platform, open-source build system and tools.

Intel compilers

Tufts licenses the Intel compilers for use on the cluster. Under modules it has three components: icc, idb and ifc.

As usual, all software under module control needs to be loaded via modules. For example to see what is available:

```
> module available
```

Access is via the following two commands:

```
idb - Intel debugger
ifort - Intel fortran compiler
icc - Intel C compiler
```

To access icc:

```
> module load icc
> icc ....your options....
```

Local Fortran documentation in HTML format can be found at:

```
>firefox file:///opt/intel/fce/10.1.017/doc/main_for/index.htm
```

or via manpages depending on what Module is loaded:

```
> man icc
> man ifc
> man idb
```

Fortran quick reference is available by typing:

```
> man ifort
> ifort -help
```

Thread programming

In shared memory multiprocessor architectures, such as SMPs, threads can be used to implement thread based parallelism. The cluster compute nodes are smp nodes with 8 cores. C and C++ compilers will have one or more compile options for threads. The cluster has pthreads installed and for more information see the man pages:

```
-bash-3.2$ man pthreads
```

Click here for a [Thread tutorial](#)

The cluster's gcc compiler supports openmp threads. Check the gcc docs for related information. Note the possible need to access a compute node in an Exclusive Host LSF manner. This would allow one's threads to use all cores on one host. A simple example such as:

```
-bash-3.2$ bsub -queue express -R bigmem -x ./yourcodeexecutable
```

would obtain a node for exclusive host access with large memory.

Text Editing tools:

emacs, vi, vim, nano, nedit

Firefox browser:

A web browser is provided to allow viewing of locally installed software product documentation. Access to the internet is restricted.