Modulefiles Instruction

SANDIA, INC.
Technical Support Group. 2019

1. module Instruction

1.1. module Explanation

- Module is a tool to easily configure the application environment to improve the usability of the application

1.2. module command

1.2.1. module avail

- Shows available modules.

```
[sandia@dirac45 ~]$ module avail

------------------------------------ /opt/ohpc/pub/moduledeps/intel-18.0.3 -------------------------------------
   mvapich2/2.2    openmpi/3.1.1

---------------------------------------- /opt/ohpc/pub/modulefiles ----------------------------------------
   EasyBuild/3.6.1    cmake/3.11.1    hwloc/1.11.10          llvm5/5.0.1    prun/1.2
   autotools          gnu7/7.3.0      intel/18.0.3.222 (L)   papi/5.6.0     sge/8.1.8 (L)

  Where:
   L:  Module is loaded

Use "module spider" to find all possible modules.
Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".


[sandia@dirac45 ~]$
```

1.2.2. module load [modulefile path], module add [modulefile path]

- Select the module you want to use.

```
[sandia@dirac45 ~]$ module load openmpi/3.1.1
[sandia@dirac45 ~]$
[sandia@dirac45 ~]$ which mpirun
/opt/mpi/intel-18.0.3/openmpi/3.1.1/bin/mpirun
```

1.2.3. module list

- Shows the currently loaded module.

```
[sandia@dirac45 ~]$ module list

Currently Loaded Modules:
  1) sge/8.1.8   2) intel/18.0.3.222   3) openmpi/3.1.1
```

1.2.4. module rm [modulefile path]

- Delete the loaded module.

```
[sandia@dirac45 ~]$
[sandia@dirac45 ~]$ module rm openmpi/3.1.1
[sandia@dirac45 ~]$
```

1.2.5. module switch (swap) [modulefile1 path] [modulefile2 path]

- Delete the environment defined in modulefile1 and apply the environment defined in modulefile2.

```
[sandia@dirac45 ~]$ which mpirun
/opt/mpi/intel-18.0.3/openmpi/3.1.1/bin/mpirun
[sandia@dirac45 ~]$ module switch openmpi/3.1.1 mvapich2/2.2
[sandia@dirac45 ~]$ which mpirun
/opt/mpi/intel-18.0.3/mvapich2/2.3/bin/mpirun
```

1.2.6.module display [modulefile path]

- Shows the environment variables and values defined in modulefile.

```
[sandia@dirac45 ~]$ module display mvapich2/2.2
-------------------------------------------------------------------------------------------------------------
   /opt/ohpc/pub/moduledeps/intel-18.0.3/mvapich2/2.2:
-------------------------------------------------------------------------------------------------------------
whatis("Name: mvapich2 built with intel toolchain ")
whatis("Version: 2.2 ")
whatis("Category: runtime library ")
whatis("Description: OSU MVAPICH2 MPI implementation ")
whatis("URL: http://mvapich.cse.ohio-state.edu/overview/mvapich2/ ")
prepend_path("PATH","/opt/mpi/intel-18.0.3/mvapich2/2.3/bin")
prepend_path("MANPATH","/opt/mpi/intel-18.0.3/mvapich2/2.3/share/man")
prepend_path("LD_LIBRARY_PATH","/opt/mpi/intel-18.0.3/mvapich2/2.3/lib")
prepend_path("MPI_DIR","/opt/mpi/intel-18.0.3/mvapich2/2.3")
prepend_path("PKG_CONFIG_PATH","/opt/mpi/intel-18.0.3/mvapich2/2.3/lib/pkgconfig")
family("MPI")
help([[
This module loads the mvapich2 library built with the intel toolchain.

Version 2.2
```

1.2.7. module purge

- Unload all currently loaded module files.

1.2.8. Frequently used module registration

```
[sandia@dirac52 ~]$ cat ~/.bashrc
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
        . /etc/bashrc
fi

# Uncomment the following line if you don't like systemctl's auto-paging feature:
# export SYSTEMD_PAGER=

# User specific aliases and functions

module load sge/8.1.8 intel/18.0.3.222
[sandia@dirac52 ~]$
```

- This is an example of defining module contents directly in a .bashrc file.
This is an example that allows you to load frequently used modules from each user's .bashrc file by default.
If you input a module that you use frequently as above in the .bashrc file for each user, it always loads the specified module

1.2.9. Module example of use

```
[sandia@dirac45 ~]$ module list
No modules loaded
[sandia@dirac45 ~]$
[sandia@dirac45 ~]$ module avail

---------------------------------- /opt/ohpc/pub/modulefiles ----------------------------------
   EasyBuild/3.6.1    cmake/3.11.1    hwloc/1.11.10       llvm5/5.0.1    prun/1.2
   autotools          gnu7/7.3.0      intel/18.0.3.222    papi/5.6.0     sge/8.1.8

Use "module spider" to find all possible modules.
Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".


[sandia@dirac45 ~]$
[sandia@dirac45 ~]$ module load sge/8.1.8 intel/18.0.3.222
[sandia@dirac45 ~]$
[sandia@dirac45 ~]$ module list

Currently Loaded Modules:
  1) sge/8.1.8   2) intel/18.0.3.222


[sandia@dirac45 ~]$ module avail

---------------------------- /opt/ohpc/pub/moduledeps/intel-18.0.3 -----------------------------
   fftw/3.3.7/double    fftw/3.3.7/single (D)    mvapich2/2.2    openmpi/3.1.1    vasp/5.4.4

---------------------------------- /opt/ohpc/pub/modulefiles ----------------------------------
   EasyBuild/3.6.1    cmake/3.11.1    hwloc/1.11.10           llvm5/5.0.1    prun/1.2
   autotools          gnu7/7.3.0      intel/18.0.3.222 (L)    papi/5.6.0     sge/8.1.8 (L)

  Where:
   D:  Default Module
   L:  Module is loaded

Use "module spider" to find all possible modules.
Use "module keyword key1 key2 ..." to search for all possible modules matching any of the "keys".


[sandia@dirac45 ~]$ module load openmpi/3.1.1
[sandia@dirac45 ~]$
[sandia@dirac45 ~]$ module list

Currently Loaded Modules:
  1) sge/8.1.8   2) intel/18.0.3.222   3) openmpi/3.1.1


[sandia@dirac45 ~]$
```

1.2.10. Using Modulefiles in Job Scheduler

```
#$-------------------------------------------------------
### MPI JOB
#$-------------------------------------------------------
#
#
# Remove all module environments.
 module purge

# Load Intel Compiler environments.
 module load sge/8.1.8 intel/18.0.3.222
```

- When you use SGE, you need to call modulefile as above in the script to submit job normally. After removing all the module files loaded by module purge, you must load the module file to be used for the job.