PlaFRIM

PlaFRIM 2, Modules

Nathalie Furmento

15.12.2015

1

Introduction

Modules

- Dynamic modification of a user's environment via modulefiles.
- Each modulefile contains the information needed to use the specific library/compiler/... by altering or setting shell environment variables such as PATH, MANPATH, etc.
- Different versions of applications.
- Metamodules can be used to load an entire suite of different applications.

Compilers: GNU & Intel

- GNU:
  gcc is available without loading any module, but we advise to load it as module.

```
module load compiler/gcc
→ gcc g++ gfortran
$ gcc -O2 -Wall toto.c -o toto
```

- Intel:

```
module load compiler/intel
→ icc, icpc, ifort
$ ifort hello.f90 -o hello
```

  (MKL libs, optimized libraries for maths functions)

2

The official modules

The official modules

- Maintained by the technical team

The official modules

- Maintained by the technical team
- Available in `/cm/shared/modulefiles`

The official modules

- Maintained by the technical team
- Available in `/cm/shared/modulefiles`
- Update policy: at least the last 3 versions of a module are kept:
  - `mpi/intel-mpi/64/4.1.3 mpi/intel-mpi/64/5.0.3 mpi/intel-mpi/64/5.1.1`
  - `mpi/openmpi/gcc/1.10.0-tm mpi/openmpi/gcc/1.8.1 mpi/openmpi/gcc/1.8.4-tm mpi/openmpi/gcc/1.8.5-tm mpi/openmpi/gcc/1.8.6-tm`

The official modules

```
------------------------------------------------- /cm/shared/modulefiles ------------------------------
cmgui/7.0                              compiler/open64/4.5.2.1              mpi/intel-mpi/64/5.1.1/109
compiler/cuda/6.5/blas/6.5.14          default-environment                  mpi/intel-mpi/mic/4.1.3/049
compiler/cuda/6.5/fft/6.5.14           intel/mkl/64/11.1/2013_sp1.3.174     mpi/mpich/ge/gcc/64/3.1
compiler/cuda/6.5/nsight/6.5.14        intel/mkl/64/11.2/2015.5.223         mpi/mpich/ge/open64/64/3.1
compiler/cuda/6.5/profiler/6.5.14      intel/mkl/64/11.2/2016.0.0           mpi/mpiexec/0.84_432
compiler/cuda/6.5/toolkit/6.5.14       intel-cluster-checker/2.2.2          mpi/mvapich2/gcc/64/2.2.1
compiler/cuda/7.0/blas/7.0.28          intel-cluster-runtime/ia32/3.7       mpi/openmpi/gcc/1.10.0-tm
compiler/cuda/7.0/fft/7.0.28           intel-cluster-runtime/intel64/3.7    mpi/openmpi/gcc/1.10.0-tm-mlx
compiler/cuda/7.0/nsight/7.0.28        intel-cluster-runtime/mic/3.7        mpi/openmpi/gcc/1.10.1-tm
compiler/cuda/7.0/profiler/7.0.28      intel-tbb-oss/ia32/43_20150424oss    mpi/openmpi/gcc/1.8.1
compiler/cuda/7.0/toolkit/7.0.28       intel-tbb-oss/intel64/43_20150424oss mpi/openmpi/gcc/1.8.4-tm
compiler/gcc/4.8.4                     iozone/3_420                         mpi/openmpi/gcc/1.8.5-tm
compiler/gcc/4.9.0                     iperf/3.0.1                          mpi/openmpi/gcc/1.8.6-tm
compiler/gcc/4.9.0.1                   magma/2.11.13                        slurm/14.03.0
compiler/gcc/4.9.2                     magma/2.17.12                        tools/debug/ddt/5.0.1
compiler/gcc/5.1.0                     magma/2.19.5                         tools/debug/ddt/5.1.43967
compiler/intel/64/2013_sp1.3.174       mpi/intel-mpi/32/4.1.3/049           tools/irods/3.3.1
compiler/intel/64/2015.5.223           mpi/intel-mpi/64/4.1.3/049
compiler/intel/64/2016                 mpi/intel-mpi/64/5.0.3/048
```

3

The users modules

The users modules

- All users can be asked to be added to the Unix group `plafrim-dev` ... (ticket to plafrim-support@inria.fr)

The users modules

- All users can be asked to be added to the Unix group `plafrim-dev` ... (ticket to plafrim-support@inria.fr)
- ... and install modules in `/cm/shared/dev/modulefiles`

The users modules

- All users can be asked to be added to the Unix group `plafrim-dev` ... (ticket to plafrim-support@inria.fr)
- ... and install modules in `/cm/shared/dev/modulefiles`

- But there is some rules to follow ...

The users modules

```
------------------------------------------------- /cm/shared/dev/modulefiles -------------------------------------------------
benchmark/bonnie++/1.97.1                  genome/cutadapt/1.7.1              linalg/chameleon/0.9.1/gcc/cuda-magma-mkl-mpi-starpu
benchmark/hpl/2.1                          genome/fastqc/0.11.2               linalg/chameleon/0.9.1/gcc/mkl-mpi-starpu
benchmark/iozone/3_420                     genome/flash/1.2.11                linalg/cuda65/blas/6.5.14
benchmark/netperf/2.6.0                    genome/jspecies/1.2.1              linalg/lapack/gcc/64/3.5.0
bigdata/drill/1.1.0                        genome/kmergenie/1.6950            linalg/lapack/open64/64/3.5.0
bigdata/hadoop/2.6.1                       genome/mcl/14-137                  linalg/magma/1.6.2/gcc/cuda-mkl
bigdata/spark/1.5.1                        genome/minia/1.6906                linalg/maphys/0.9.2/gcc/mkl-mpi-pastix-scotch
build/ac269-am114-lt246-m41417             genome/mira/3.2.1                  linalg/openblas/dynamic/0.2.8
build/cmake/3.2.1                          genome/mira/4.0.2                  linalg/pastix/5.2.2.22/gcc/mkl-mpi-scotch
compiler/java/jdk1.8.0_60                  genome/mummer/3.23                 linalg/petsc/3.6.2
compiler/tcltk/8.4.20                      genome/orthomcl/1.2                linalg/plasma/gcc/2.7.1
dataencoding/jansson/2.7                   genome/samtools/1.2                partitioning/scotch/int32/6.0.4
editor/nano/2.4.1                          genome/tophat/2.1.0                partitioning/scotch/int64/6.0.4
fft/cuda65/fft/6.5.14                      genome/trim_galore/0.3.7           runtime/starpu/1.1.5/gcc/cuda-fxt-mpi
fft/fftw2/openmpi/gcc/64/double/2.1.5      genome/vcftools/b240116cfa         runtime/starpu/1.1.5/gcc/fxt-mpi
fft/fftw2/openmpi/gcc/64/float/2.1.5       gis/ferret/6.93                    scm/git/2.3.5
fft/fftw2/openmpi/open64/64/double/2.1.5   gis/gdal/2.0.0                     soft/openfoam/2.3.0
fft/fftw2/openmpi/open64/64/float/2.1.5    gis/libtiff/4.0.4                  statistics/R/3.2.2
fft/fftw3/openmpi/gcc/64/3.3.3             hardware/hwloc/1.11.0              tools/matlab/MCR_R2015a
fft/fftw3/openmpi/open64/64/3.3.3          hardware/hwloc-mic/1.11.0          tools/matlab/R2015a
formal/pari/openmpi/2.7.4                  hardware/libpciaccess/0.13.4       tools/module_cat/1.0.0
formal/pari/pthread/2.7.4                  io/hdf5/1.6.10                     tools/refprop/1.0
genome/bioperl/1.6.924                     io/hdf5_18/1.8.12                  trace/eztrace/1.0
genome/blast/2.2.31+                       io/netcdf/gcc/64/4.3.1.1           trace/eztrace/1.0-intel
genome/bowtie2/2.2.6                       io/netcdf/open64/64/4.3.1.1        trace/eztrace/1.1
genome/bwa/0.7.12                          linalg/blas/gcc/64/1               trace/fxt/0.3.1
genome/corset/1.04                         linalg/blas/open64/64/1            trace/htop/1.0.3
```

The users modules

```
------------------------------------------------- /cm/shared/dev/modulefiles -------------------------------------------------
benchmark/bonnie++/1.97.1                  genome/cutadapt/1.7.1              linalg/chameleon/0.9.1/gcc/cuda-magma-mkl-mpi-starpu
benchmark/hpl/2.1                          genome/fastqc/0.11.2               linalg/chameleon/0.9.1/gcc/mkl-mpi-starpu
benchmark/iozone/3_420                     genome/flash/1.2.11                linalg/cuda65/blas/6.5.14
benchmark/netperf/2.6.0                    genome/jspecies/1.2.1              linalg/lapack/gcc/64/3.5.0
bigdata/drill/1.1.0                        genome/kmergenie/1.6950            linalg/lapack/open64/64/3.5.0
bigdata/hadoop/2.6.1                       genome/mcl/14-137                  linalg/magma/1.6.2/gcc/cuda-mkl
bigdata/spark/1.5.1                        genome/minia/1.6906                linalg/maphys/0.9.2/gcc/mkl-mpi-pastix-scotch
build/ac269-am114-lt246-m41417             genome/mira/3.2.1                  linalg/openblas/dynamic/0.2.8
build/cmake/3.2.1                          genome/mira/4.0.2                  linalg/pastix/5.2.2.22/gcc/mkl-mpi-scotch
compiler/java/jdk1.8.0_60                  genome/mummer/3.23                 linalg/petsc/3.6.2
compiler/tcltk/8.4.20                      genome/orthomcl/1.2                linalg/plasma/gcc/2.7.1
dataencoding/jansson/2.7                   genome/samtools/1.2                partitioning/scotch/int32/6.0.4
editor/nano/2.4.1                          genome/tophat/2.1.0                partitioning/scotch/int64/6.0.4
fft/cuda65/fft/6.5.14                      genome/trim_galore/0.3.7           runtime/starpu/1.1.5/gcc/cuda-fxt-mpi
fft/fftw2/openmpi/gcc/64/double/2.1.5      genome/vcftools/b240116cfa         runtime/starpu/1.1.5/gcc/fxt-mpi
fft/fftw2/openmpi/gcc/64/float/2.1.5       gis/ferret/6.93                    scm/git/2.3.5
fft/fftw2/openmpi/open64/64/double/2.1.5   gis/gdal/2.0.0                     soft/openfoam/2.3.0
fft/fftw2/openmpi/open64/64/float/2.1.5    gis/libtiff/4.0.4                  statistics/R/3.2.2
fft/fftw3/openmpi/gcc/64/3.3.3             hardware/hwloc/1.11.0              tools/matlab/MCR_R2015a
fft/fftw3/openmpi/open64/64/3.3.3          hardware/hwloc-mic/1.11.0          tools/matlab/R2015a
formal/pari/openmpi/2.7.4                  hardware/libpciaccess/0.13.4       tools/module_cat/1.0.0
formal/pari/pthread/2.7.4                  io/hdf5/1.6.10                     tools/refprop/1.0
genome/bioperl/1.6.924                     io/hdf5_18/1.8.12                  trace/eztrace/1.0
genome/blast/2.2.31+                       io/netcdf/gcc/64/4.3.1.1           trace/eztrace/1.0-intel
genome/bowtie2/2.2.6                       io/netcdf/open64/64/4.3.1.1        trace/eztrace/1.1
genome/bwa/0.7.12                          linalg/blas/gcc/64/1               trace/fxt/0.3.1
genome/corset/1.04                         linalg/blas/open64/64/1            trace/htop/1.0.3
```

4

Module naming policies

Categories

- In order to increase and to ease the use of the modules on the platform, modules are grouped within categories. Each module belongs to a specific category, which can be for example `trace` or `statistics`.

Categories

- In order to increase and to ease the use of the modules on the platform, modules are grouped within categories. Each module belongs to a specific category, which can be for example `trace` or `statistics`.

- The module naming policy is as follows:

  category/*module*/*option*/*version*

Categories

- In order to increase and to ease the use of the modules on the platform, modules are grouped within categories. Each module belongs to a specific category, which can be for example `trace` or `statistics`.

- The module naming policy is as follows:

  category/*module*/*option*/*version*

- the number of options being between 0 and as many as you want.

  ```
  partitioning/scotch/int32/6.0.4
  partitioning/scotch/int64/6.0.4
  ```

File system

- Modules files go in `/cm/shared/dev/modulefiles` by following the naming policy.
  Module `fxt` in the category `trace` with version `0.3.1` and without any specific option
  $\rightarrow$ `/cm/shared/dev/modulefiles/trace/fxt/0.3.1`

File system

- Modules files go in `/cm/shared/dev/modulefiles` by following the naming policy.
  Module `fxt` in the category `trace` with version `0.3.1` and without any specific option
  → `/cm/shared/dev/modulefiles/trace/fxt/0.3.1`
- Installation module files go in `/cm/shared/dev/apps` with the same naming policy.
  → `/cm/shared/dev/apps/trace/fxt/0.3.1`

File system

- Modules files go in `/cm/shared/dev/modulefiles` by following the naming policy.
  Module `fxt` in the category `trace` with version `0.3.1` and without any specific option
  → `/cm/shared/dev/modulefiles/trace/fxt/0.3.1`
- Installation module files go in `/cm/shared/dev/apps` with the same naming policy.
  → `/cm/shared/dev/apps/trace/fxt/0.3.1`
- Make sure all the files can be read by anyone on the platform.
- write a real description in whatis, the command `module show/whatis xxx` should say something more meaningful than `loads the xxx environment`

tools/module_cat

- `module load tools/module_cat`

tools/module_cat

- `module load tools/module_cat`
- You can then list all the modules whose name contains a specific string.

  `module_grep mkl`

```
intel/mkl/64/11.1/2013_sp1.3.174
intel/mkl/64/11.2/2015.5.223
intel/mkl/64/11.2/2016.0.0
linalg/chameleon/0.9.1/gcc/cuda-magma-mkl-mpi-starpu
linalg/chameleon/0.9.1/gcc/mkl-mpi-starpu
linalg/magma/1.6.2/gcc/cuda-mkl
linalg/maphys/0.9.2/gcc/mkl-mpi-pastix-scotch
linalg/pastix/5.2.2.22/gcc/mkl-mpi-scotch
```

tools/module_cat

- `module load tools/module_cat`
- You can then list all the modules whose name contains a specific string.

  `module_grep mkl`
- List all the categories

  `module_list`

```
-------------------- /cm/shared/dev/modulefiles
benchmark editor runtime trace bigdata fft hardware
scm build formal io soft compiler genome linalg
statistics dataencoding gis partitioning tools
```

tools/module_cat

- `module load tools/module_cat`
- You can then list all the modules whose name contains a specific string.
  `module_grep mkl`
- List all the categories
  `module_list`
- List the content of a given category
  `module_list hardware`

```
-------------------- /cm/shared/dev/modulefiles
hardware/hwloc hardware/hwloc-mic
hardware/libpciaccess
```

tools/module_cat

- Restrict MODULEPATH to a specific set of categories

```
module_restrict mpi compiler && echo $MODULEPATH
```

```
:/cm/shared/modulefiles/compiler:
/cm/shared/modulefiles/mpi:
/cm/shared/dev/modulefiles/compiler
```

tools/module_cat

- Restrict MODULEPATH to a specific set of categories

  `module_restrict mpi compiler && echo $MODULEPATH`
- Add some categories

  `module_add tools && echo $MODULEPATH`

```
[module_add]adding/cm/shared/modulefiles/tools
[module_add] adding /cm/shared/dev/modulefiles/tools
/cm/shared/dev/modulefiles/tools:
/cm/shared/modulefiles/tools:
/cm/shared/modulefiles/compiler:
/cm/shared/modulefiles/mpi:
/cm/shared/dev/modulefiles/compiler
```

tools/module_cat

- Restrict MODULEPATH to a specific set of categories

  `module_restrict mpi compiler && echo $MODULEPATH`
- Add some categories

  `module_add tools && echo $MODULEPATH`
- Remove some categories

  `module_rm mpi compiler && echo $MODULEPATH`

```
/cm/shared/dev/modulefiles/tools:
/cm/shared/modulefiles/tools:
```

tools/module_cat

- Restrict MODULEPATH to a specific set of categories
  `module_restrict mpi compiler && echo $MODULEPATH`
- Add some categories
  `module_add tools && echo $MODULEPATH`
- Remove some categories
  `module_rm mpi compiler && echo $MODULEPATH`
- Help for the different options are displayed by calling
  `module_help`

module_<command> with <command> being:

help	- display help
list [<**category**>]	- list categories (from the top level by default, or the given category)
init	- list categories from the initial setup
reset	- reset initial MODULEPATH
add <**category1**> ... <**categoryn**>	- add given category/ies to MODULEPATH
rm <**category1**> ... <**categoryn**>	- remove given category/ies from MODULEPATH
restrict <**category1**> ... <**categoryn**>	- only keep given category/ies in MODULEPATH

5

Other questions

Other questions

- the migration of the users modules from PlaFRIM1 to PlaFRIM2 has to be done by **you**. Take the opportunity to do some cleanup!

Other questions

- the migration of the users modules from PlaFRIM1 to PlaFRIM2 has to be done by **you**. Take the opportunity to do some cleanup!
- Software should preferably be compiled on compute nodes, as frontal nodes may not have all necessary libraries/drivers/... (e.g cuda)