Software available on the TeideHPC and AnagaGPU clusters

Before reading the next section, we recommend taking a look at the section "Modules Tool"

The Environment Modules system uses a set of commands that include:

- `module overview` : Shows a general summary list of the available modules.
- `module avail` : Shows all modules available in the system to load.
- `module avail [module]` : Shows the available modules with that name.
- `module spider [module]` : Search for a specific module.
- `module load [module]` : Loads a specific module, setting the environment variables necessary to use the software that the module represents.
- `module unload [module]:` Unloads a module, removing the environment variables associated with the software from the user session.
- `module swap [current_module] [new_module]` : Swaps one loaded module for another, which is useful for quickly switching software versions.
- `module purge` : Removes all loaded modules, clearing the user's environment of any configuration made by modules.
- `module help [module]` : Provides help information for a specific module, which may include information about how to use the software, its environment variables, etc.
- `module whatis [module]` : Shows a brief description of what a module does.
- `module show [module]` : Shows information about what the module does to the user's environment; that is, what environment variables it adjusts and how.

In TeideHPC there are 2 clusters, TeideHPC and AnagaGPU

For this reason, the set of available modules **depends on the Cluster you are going to use.**

Each cluster has its own login nodes and **does not have the same software installed.**

Software execution is prohibited on login nodes

On the TeideHPC and AnagaGPU login nodes, it is prohibited to execute any application except common Linux commands.

ml* is an abbreviation of *modules

To avoid common errors when writing the *modules* commands such as *modles,moduls, mdules, etc* it is possible to use the abbreviation **ml**.

`ml spider ...` `ml load ...` `ml purge` ...

Meaning of some significant modules.

- **GCCcore**: basic set of compilers from the GNU Compiler Collection: C, C++, Objective-C, Fortran.
- **GCC**: collection of GCCcore compilers and libraries (GDB, binutils, glibc)
- **foss**: The foss module consists entirely of the common term *FOOS*, which is short for *Free and Open source software*.
- **gompi**: toolchain that groups together tools compiled with GCC and OpenMPI
- **gompic**: toolchain that groups together tools compiled with GCC+CUDA and OpenMPI
- **intel-compilers**: Classic Intel C, C++ & Fortran compilers and oneAPI.
- **iimpi**: Intel C/C++ and Fortran compiler with Intel MPI support.

Example of using modules

```
module overview
or ml overview
```

```
----------------/share/easybuild/software/common/modules/all ------------------------
EasyBuild (2)   Go (1)   Mamba (1)   Miniconda3 (2)   Singularity (1)   Squashfs (1)   slurm
(1)

----------------/share/easybuild/software/x86_64/modules/all ------------------------
ADMIXTURE            (1)   Imath                (1)    UCX              (2)   jupyter-server  (1)
ATK                  (1)   Infernal             (1)    UDUNITS          (2)   libGLU          (2)
ATLAS                (1)   JasPer               (4)    USEARCH          (1)   libarchive      (3)
AdmixTools           (2)   Java                 (5)    UnZip            (2)   libcerf         (1)
Armadillo            (2)   JsonCpp              (1)    WPS              (2)   libdeflate      (1)
```

```
Arrow               (2)   JupyterLab          (2)    WRF              (2)   libdrm         (2)
Autoconf            (4)   LAME                (1)    X11              (2)   libepoxy       (1)
Automake            (4)   LAMPLD              (2)    XALT             (1)   libevent       (1)
Autotools           (5)   LAPACK              (1)    XML-LibXML       (2)   libfabric      (2)
BCFtools            (1)   LERC                (1)    XZ               (2)   libffi         (2)
BEDTools            (1)   LLVM                (2)    Xerces-C++       (1)   libgd          (1)
```

```
module avail
or ml av
```

```
----------------- /share/easybuild/software/common/modules/all -----------------
   EasyBuild/4.7.0        Go/1.18.3         Miniconda3/22.11.1-1        Singularity/3.11.0    slurm/
teide
   EasyBuild/4.8.2 (D)    Mamba/4.14.0-0    Miniconda3/23.5.2-0  (D)    Squashfs/4.3

---------------- /share/easybuild/software/x86_64/modules/all --------------------
   ADMIXTURE/1.3.0                                  UCC/1.1.0-GCCcore-12.2.0
   ATK/2.38.0-GCCcore-12.2.0                        UCX/1.11.2-GCCcore-11.2.0
   ATLAS/0.9.9-foss-2022b                           UCX/1.13.1-GCCcore-12.2.0                    (D)
   AdmixTools/7.0.2-foss-2021b                      UDUNITS/2.2.28-GCCcore-11.2.0
   AdmixTools/7.0.2-foss-2022b               (D)    UDUNITS/2.2.28-
GCCcore-12.2.0                  (D)
   Armadillo/10.5.3-foss-2022b                      USEARCH/11.0.667-i86linux32
   Armadillo/11.4.3-foss-2022b               (D)    UnZip/6.0-GCCcore-11.2.0
   Arrow/6.0.0-foss-2021b                           UnZip/6.0-GCCcore-12.2.0                     (D)
   Arrow/11.0.0-gfbf-2022b                   (D)    WPS/3.9.1-intel-2021b-dmpar
   Autoconf/2.69-GCCcore-11.2.0                     WPS/4.1-intel-2021b-dmpar
(D)
   Autoconf/2.71-GCCcore-11.2.0                     WRF/3.9.1.1-intel-2021b-dmpar
   Autoconf/2.71-GCCcore-12.2.0                     WRF/4.4.1-foss-2022b-dmpar
(D)
   Autoconf/2.71                             (D)    X11/20210802-GCCcore-11.2.0
   Automake/1.16.2-GCCcore-11.2.0                   X11/20221110-
GCCcore-12.2.0                  (D)
   Automake/1.16.4-GCCcore-11.2.0                   XALT/3.0.1
...
```

```
module spider wrf
```

```
----------------------------------------------------
  WRF:
----------------------------------------------------
    Description:
      The Weather Research and Forecasting (WRF) Model is a next-generation mesoscale
numerical weather
      prediction system designed to serve both operational forecasting and atmospheric
research needs.

     Versions:
        WRF/3.9.1.1-intel-2021b-dmpar
        WRF/4.4.1-foss-2022b-dmpar

----------------------------------------------------
```

```
  For detailed information about a specific "WRF" package (including how to load the
modules) use the module's full n
ame.
  Note that names that have a trailing (E) are extensions provided by other modules.
  For example:

     $ module spider WRF/4.4.1-foss-2022b-dmpar
```

```
module load WRF/3.9.1.1-intel-2021b-dmpar
module list
```

```
Currently Loaded Modules:
  1) GCCcore/11.2.0                         11) intel/2021b
  2) zlib/1.2.11-GCCcore-11.2.0             12) libtirpc/1.3.1-GCCcore-11.2.0
  3) binutils/2.37-GCCcore-11.2.0           13) JasPer/2.0.24-GCCcore-11.2.0
  4) intel-compilers/2021.4.0               14) Szip/2.1.1-GCCcore-11.2.0
  5) numactl/2.0.14-GCCcore-11.2.0          15) HDF5/1.10.7-iimpi-2021b
  6) UCX/1.11.2-GCCcore-11.2.0              16) OpenSSL/1.1
  7) impi/2021.4.0-intel-compilers-2021.4.0  17) cURL/7.78.0-GCCcore-11.2.0
  8) imkl/2021.4.0                          18) netCDF/4.7.4-iimpi-2021b
  9) iimpi/2021b                            19) netCDF-Fortran/4.5.3-iimpi-2021b
 10) imkl-FFTW/2021.4.0-iimpi-2021b         20) WRF/3.9.1.1-intel-2021b-dmpar
```

- **foss**

The foss module consists entirely of the common term *FOOS*, which is short for *Free and Open source software*. 2021a, 2021b, 2022b, etc indicates the generation.

This module consists of the following packages and is responsible for preloading them.

- binutils
- GNU GCC (C), g++ (C++) and gfortran (Fortran) compiler
- Open MPI Library
- FlexiBLAS library with OpenBLAS + LAPACK as backend
- ScaLAPACK Bookstore
- FFTW (Fourier transform (DFT)) library

```
module spider foss
module load foss/2021b
module list
```

```
Currently Loaded Modules:
  1) GCCcore/11.2.0                 8) libpciaccess/0.16-GCCcore-11.2.0  15) OpenMPI/4.1.1-
GCC-11.2.0
  2) zlib/1.2.11-GCCcore-11.2.0      9) hwloc/2.5.0-GCCcore-11.2.0        16) OpenBLAS/0.3.18-
GCC-11.2.0
```

```
  3) binutils/2.37-GCCcore-11.2.0   10) OpenSSL/1.1                    17) FlexiBLAS/3.0.4-
GCC-11.2.0
  4) GCC/11.2.0                     11) libevent/2.1.12-GCCcore-11.2.0    18) gompi/2021b
  5) numactl/2.0.14-GCCcore-11.2.0  12) UCX/1.11.2-GCCcore-11.2.0         19) FFTW/3.3.10-
gompi-2021b
  6) XZ/5.2.5-GCCcore-11.2.0        13) libfabric/1.13.2-GCCcore-11.2.0   20) ScaLAPACK/
2.1.0-gompi-2021b-fb
  7) libxml2/2.9.10-GCCcore-11.2.0  14) PMIx/4.1.0-GCCcore-11.2.0         21) foss/2021b
```

- **gompi**:

The *gompi* module preloads all software compiled with GNU GCC and OpenMPI. 2021a, 2021b, 2022b, etc indicates the generation.

```
$ module load gompi/2021b
$ module list
```

```
Currently Loaded Modules:
  1) GCCcore/11.2.0                 7) libxml2/2.9.10-GCCcore-11.2.0     13) libfabric/1.13.2-
GCCcore-11.2.0
  2) zlib/1.2.11-GCCcore-11.2.0     8) libpciaccess/0.16-GCCcore-11.2.0  14) PMIx/4.1.0-
GCCcore-11.2.0
  3) binutils/2.37-GCCcore-11.2.0   9) hwloc/2.5.0-GCCcore-11.2.0        15) OpenMPI/4.1.1-
GCC-11.2.0
  4) GCC/11.2.0                    10) OpenSSL/1.1                       16) gompi/2021b
  5) numactl/2.0.14-GCCcore-11.2.0 11) libevent/2.1.12-GCCcore-11.2.0
  6) XZ/5.2.5-GCCcore-11.2.0       12) UCX/1.11.2-GCCcore-11.2.0
```

- **intel**:

This common module preloads Intel compilers and libraries. 2021a, 2021b, 2022b, etc indicates the generation.

- Intel C/C++/Fortran compilers (icc, icpc and ifort)
- binutils
- GCC, which serves as the basis for Intel compilers.
- Intel MPI Library
- Intel Math Kernel Library (MKL) for BLAS/LAPACK/FFT functionality

```
module load intel/2022b
module list
```

```
Currently Loaded Modules:
  1) GCCcore/12.2.0                5) numactl/2.0.16-GCCcore-12.2.0          9) iimpi/2022b
  2) zlib/1.2.12-GCCcore-12.2.0    6) UCX/1.13.1-GCCcore-12.2.0             10) imkl-FFTW/
2022.2.1-iimpi-2022b
  3) binutils/2.39-GCCcore-12.2.0  7) impi/2021.7.1-intel-compilers-2022.2.1  11) intel/
```

```
2022b
 4) intel-compilers/2022.2.1       8) imkl/2022.2.1
```