



National HPC

SW and SW development

UNIVERSITY OF BERGEN





Outline

- Setup
- SW available
- Requesting SW
- Compiling SW
- Running SW
- Optimisation
- Debugging
- Profiling





Modules

- No modules by default
- Toolchains and dependencies
- Hidden system modules
 - `module --show_hidden av`
- Sticky modules





```
[aoltu@login-1-2 ~]$ module li
```

Currently Loaded Modules:

1) StdEnv (S)

Where:

S: Module

```
[aoltu@login-1-2 ~]$ module load intel/2018a
```

```
[aoltu@login-1-2 ~]$ module li
```

```
2.28 [aoltu@login-1-2 ~]$ module reset
```

Resetting modules to system default

```
[aoltu@login-1-2 ~]$ module li
```

Currently Loaded Modules:

1) StdEnv (S)

Where:

S: Module is Sticky, requires --force to unload or purge

Where:

S: Module is Sticky, requires --force to unload or purge

H: Hidden Module

```
[aoltu@login-1-2 ~]$
```



Modules

- Modules
 - module av
 - module load / unload
 - module switch
 - module list
 - module purge
- No default versions
- Autocompletion





EasyBuild

- Framework to install and build scientific software
 - Reproducibility and efficiency
 - Has its own + and -
 - Stable release cycle
 - Some challenges to have compilers outside of release cycle
 - Loads whole toolchain with dependencies





SW available

- https://documentation.sigma2.no/development/which_software_is_installed.html
- <https://documentation.sigma2.no/apps/allmodules.html>
- module av





Requesting new SW

- support@metacenter.no





Compiling SW

- module load intel/2018a
- Any extra library





Compilers

- Regular
 - ifort - Fortran
 - icc - C
 - icpc - C++
- MPI
 - mpiifort - Fortran
 - mpiicc - C
 - mpicpc - C++





Linking

- pkg_config compatible - automatic
- Otherwise may vary
 - LD_LIBRARY_PATH - automatic
 - Still needs -l
 - \$EBROOT*Software*





Running

- Queue - script or interactive
- mpirun for Intel MPI compiled
- srun - for other compilers





Fine control

- Number of cores per nodes and so on
 - SBATCH directives
 - mpirun will pick up automatically





Optimisation

- Xeon(R) CPU E5-2683 v4
 - AVX2
 - FMA
- Intel
 - -fast
 - -xHost -O3
 - or *-xcode*
- GNU : `—march=native`





Debugging

- `ulimit -c unlimited` - for core dumps
 - `gdb executable core`
- Totalview
 - `-newUI` for modern UI
 - ~512 cores





Finding a sweet spot

- Make a short test case
- Try on different number of cores
- Find your sweet spot :)





Profiling

- ARM Performance Reports
 - <https://documentation.sigma2.no/development/performance.html>
- ARM MAP





Current compiler recommendation for CCSM/NORES

- Issues with Intel 2017
- Intel 2016 current recommended compiler
 - module load intel/2016a
 - module load netCDF-Fortran/4.4.3-intel-2016a
 - module load PnetCDF/1.8.1-intel-2016a
 - module load CMake/3.5.2-intel-2016a
- Recipe to build WRF on Fram
- Testing of Intel 2018a





Matlab and other licensed software

- Matlab – bring you own license (UiB)
 - export
MLM_LICENSE_FILE=home_inst_matlab_server
- <https://documentation.sigma2.no/apps/matlab.html>





UNIVERSITY OF BERGEN