

Idun FAQ

- **General Questions**
 - How to login to IDUN?
 - Can I ssh to IDUN from off campus?
 - What is the difference between cpus, cores and hyperthreading?
 - Fluent licenses
 - I get an error message that contains ^M
 - I get a locale error message on login: /usr/bin/manpath: can't set the locale; make sure \$LC_* and \$LANG are correct
- **Applications**
 - Installation of an R package fails: Error: ERROR: no permission to install to directory ?/share/apps/software/MPI/GCC/5.4.0-2.26/OpenMPI/1.10.3/R/3.3.3/lib64/R/library?
- **Application Development**
 - What does it mean that Lmod says that the module exists but cannot be loaded?
 - What do the error "Tcl command execution failed..." when loading a module file mean?
 - Where are the mpicc and mpif90 compiler wrappers located?
 - My code fails to link with the message "...relocation truncated to fit..."
 - How do I get support for C++11?
 - I want to use a new compiler, but the module depends on an older one: ERROR:151: Module 'openjpeg/2.1.0' depends on one of the module(s) 'intelcomp/15.0.1 '
- **Running Jobs**
 - How can several jobs be executed on one node, for example having one job for each core on the node?
 - Job dependencies
 - Is hyperthreading enabled?
 - What is the estimated start time for my queued job?
 - How do I get information about running and completed jobs?
 - Interactive job (eq partition EPIC):
 - My OMP_NUM_THREADS or MKL_NUM_THREADS setting does not appear to be recognized.

General Questions

How to login to IDUN?

See [Log into IDUN](#)

Can I ssh to IDUN from off campus?

It is not possible to ssh directly into IDUN from off-campus except in special circumstances. We recommend that you use VPN. You can also log in using ssh via login.ansatt.ntnu.no or login.stud.ntnu.no.

What is the difference between cpus, cores and hyperthreading?

Each node has two CPU's with ten physical cores. Each core can execute two threads(hyperthreading). With hyperthreading each node has twenty logical cores.

Fluent licenses

The fluent license server is on the IVT-faculty. The server is lisens01.ivt.ntnu.no. To get information about licenses, do:

```
$ module load fluent
$ lmstat -a -c 1055@lisens01.ivt.ntnu.no
```

I get an error message that contains ^M

If you receive an error message with the ^M character, for example:

```
"/bin/bash^M: bad interpreter: No such file or directory"
```

then you have copied a text file from Windows to IDUN. This text file therefore contains Windows line-end characters, which are different from UNIX line-end characters. You have to convert the text file to UNIX format by using the dos2unix command:

```
$ dos2unix filename
```

I get a locale error message on login: /usr/bin/manpath: can't set the locale; make sure \$LC_* and \$LANG are correct

You will need to change the locale setting in your Mac terminal preferences. In your Mac Terminal select "Terminal" -> "Preferences" -> "Advanced". Uncheck "Set locale environment variables on startup". You need to stop/kill all your terminal sessions for the change to take effect. On startup the locale environment variables will be unset, and any Linux session following a ssh login, will not inherit the locale setting from the Mac Terminal.

Applications

Installation of an R package fails: Error: ERROR: no permission to install to directory ?/share/apps/software/MPI/GCC/5.4.0-2.26/OpenMPI/1.10.3/R/3.3.3/lib64/R/library?

If you execute `devtools::install_github(<package>)`, the installation fails due to missing write permissions under `/share/apps`. You will need to install the `<package>` in your home directory. You do this by making your own subdirectory for R packages, and call `withr::with_libpaths()` from R:

```
devtools::install_github("QTCAT/qtcat")
```

```
Error: ERROR: no permission to install to directory ?/share/apps/software/MPI/GCC/5.4.0-2.26/OpenMPI/1.10.3/R/3.3.3/lib64/R/library?  
Error: Command failed (1)
```

```
$ mkdir ~/myRpckgs/lib  
  
# From within R do. Remember to replace QTCAT/qtcat with the packages you  
need  
> withr::with_libpaths( "~/myR/lib", install_github( "QTCAT/qtcat" ) ).libPaths  
( "~/myR/lib" )
```

Application Development

What does it mean that Lmod says that the module exists but cannot be loaded?

```
$ module load Python/2.7.14  
Lmod has detected the following error: These module(s) exist but cannot be  
loaded as  
requested: "Python/2.7.14"  
Try: "module spider Python/2.7.14" to see how to load the module(s).
```

Applications and libraries are loadable from a hierarchical software tree. The hierarchy has two main tool-chains foss (Free and Open Source Software) and intel. The software in the foss branch are built with GNU compilers. The software in the Intel branch are built with Intel compilers.

``module spider Python/2.7.14`` will show you the software that Python-2.7.14 depends on

```

$ module spider Python/2.7.14
-----
Python: Python/2.7.14
-----
Description:
  Python is a programming language that lets you work more quickly and
  integrate your systems
  more effectively.
Other possible modules matches:
  ScientificPython, netcdf4-python
You will need to load all module(s) on any one of the lines below
before the "Python/2.7.14" module is available to load.
GCC/6.4.0-2.28 OpenMPI/2.1.1
GCC/6.4.0-2.28 OpenMPI/2.1.2
icc/2017.4.196-GCC-6.4.0-2.28 impi/2017.3.196
icc/2018.1.163-GCC-6.4.0-2.28 impi/2018.1.163
ifort/2017.4.196-GCC-6.4.0-2.28 impi/2017.3.196
ifort/2018.1.163-GCC-6.4.0-2.28 impi/2018.1.163

Help:

Description
=====
Python is a programming language that lets you work more quickly and
integrate your systems
more effectively.

More information
=====
- Homepage: http://python.org/

Included extensions
=====
arff-2.1.1, bitstring-3.1.5, blist-1.3.6, cryptography-2.0.3, Cython-
0.26.1,
dateutil-2.6.1, deap-1.0.2, decorator-4.1.2, docopt-0.6.2, ecdsa-0.13,
enum34-1.1.6, funcsigns-1.0.2, joblib-0.11, mock-2.0.0, mpi4py-2.0.0,
netaddr-0.7.19, netifaces-0.10.6, nose-1.3.7, numpy-1.13.1, pandas-
0.20.3,
paramiko-2.2.1, paycheck-1.0.2, pbr-3.1.1, pip-9.0.1, pycrypto-2.6.1,
pyparsing-2.2.0, pytz-2017.2, scipy-0.19.1, setuptools-36.5.0, six-
1.11.0,
virtualenv-15.1.0

-----
To find other possible module matches do:
module -r spider '.*Python/2.7.14.*'

```

By doing `module load GCC/6.4.0-2.28 OpenMPI/2.1.2` the prerequisite software for loading Python/2.7.14 becomes available. An alternative is to load the proper toolchain. For Python/2.7.14 this would be either foss/2018a or intel/2018a.

A tool-chain is loaded with `module load foss/2018a`.

What do the error "*Tcl command execution failed...*" when loading a module file mean?

Some modules have dependencies that require other modules to be loaded first. E.g. trying to load the 'boost/1.53.0' module will result in:

This means you must load 'intelcomp/13.0.1' before loading 'boost/1.53.0'.

Where are the *mpicc* and *mpif90* compiler wrappers located?

You need to load the mpt module file before building MPI applications:

My code fails to link with the message *"...relocation truncated to fit..."*

This happens if your code need more than 2GB of static data. Compile the code with the `-mmodel=medium -shared-intel` options.

How do I get support for C++11?

You will need to do a module load of `gcc/6.2.0` after you have loaded the Intel Compiler. If you only do module load `intelcomp/17.0.0`, `icpc -v` will report compatibility with GCC 4.3.0. GCC 4.3.0 has not implemented the C++11 standard. After a 'module load `gcc/6.2.0`' will 'icpc -v' report compatibility with GCC 6.2.0 which has implemented the C++11 standard.

I want to use a new compiler, but the module depends on an older one: ERROR: 151: Module 'openjpeg/2.1.0' depends on one of the module(s) 'intelcomp/15.0.1'

Modules can have prerequisites for older compilers. You may still use the module with a new compiler. You load the newer compiler by doing a 'module switch <load module> <new module>':

Running Jobs

How can several jobs be executed on one node, for example having one job for each core on the node?

This can accomplished with a [Job Array](#). Below is a script where 20 jobs are executed in parallel - a array of jobs. The batch job generate a Task Id for each task. The Task Id can be used for differentiating between the individual jobs/task.

Here it is assumed that the script is started from the subdirectory which contains the 'run' subdirectory. The shell command 'sbatch jobscript' will start the array job. The scripts under/run/case[0-19] is executed in parallel. The range 0-19 is just an example. Change the range to whatever fit the actual case.

Note that most nodes on IDUN have 20 cores. If a range of 1-24 is specified, task 1 to 20 will run on one node, the next 4 tasks will be executed on a second node.

```
#!/bin/bash
#SBATCH --job-name=arrayJob
#SBATCH --output=arrayJob_%A_%a.out
#SBATCH --error=arrayJob_%A_%a.err
#SBATCH --array=0-20
#SBATCH --time=00:15:00
#SBATCH --partition=WORKQ
#SBATCH --ntasks=1
#SBATCH --mem-per-cpu=3000
#####
# Begin work section #
#####
# Print this sub-job's task ID
echo "My SLURM_ARRAY_TASK_ID: " $SLURM_ARRAY_TASK_ID
./run/case${SLURM_ARRAY_TASK_ID}/yourprogram
```

Job dependencies

Job dependencies are used to defer the start of a job until the specified dependencies have been satisfied. They are specified with the `--dependency` option to `sbatch` or `swarm` in the format

```
sbatch --dependency=<type:job_id[:job_id][,type:job_id[:job_id]]> ...
```

Dependency types:

after: jobid[:jobid...]	job can begin after the specified jobs have started
afterany: jobid[:jobid...]	job can begin after the specified jobs have terminated
afternotok: jobid[:jobid...]	job can begin after the specified jobs have failed

<code>afterok:jobid[:jobid...]</code>	job can begin after the specified jobs have run to completion with an exit code of zero.
---------------------------------------	--

Is hyperthreading enabled?

No, hyperthreading is disabled by default.

What is the estimated start time for my queued job?

Adding option `--start` to command `squeue` shows the estimated start time of queued jobs:

```
$ squeue --start
```

How do I get information about running and completed jobs?

To list all running jobs:

```
$ squeue -u <username>
```

To list all jobs that are idle with estimated start times:

```
$ squeue --start -u <username>
```

Interactive job (eq partition EPIC):

```
srun --nodes=1 -p EPIC --time=01:00:00 --pty bash -i
```

My `OMP_NUM_THREADS` or `MKL_NUM_THREADS` setting does not appear to be recognized.

If your application is built with the single dynamic MKL library, `mkl_rt`, you need to specify at runtime if you want to use the threaded or sequential mode of MKL. By default, specified in the Intel compiler modulefile, sequential mode is used. If you want to use Intel threading, specify this in the jobscript using the `MKL_THREADING_LAYER` variable, e.g. (bash syntax):

```
module load intel/2017a
export MKL_THREADING_LAYER=INTEL
export OMP_NUM_THREADS=20
export MKL_NUM_THREADS=20
```

Notice, always set environment variables after 'module load' commands so that your own variable settings are not reset.