



E E S S I

EUROPEAN ENVIRONMENT FOR
SCIENTIFIC SOFTWARE INSTALLATIONS

Software layer
optimized and
tested
installations of
applications +
dependencies

Compatibility layer
levelling the ground across
Linux distros

Filesystem layer
stream packages globally

Host operating system
network, drivers, slurm, ...



Type **eessi** in your login terminal (**Bash** only):

```
1 zen trainee00@l55:~$ eessi
2 Found EESSI repo @ /cvmfs/software.eessi.io/versions/2023.06!
3 archdetect says x86_64/amd/zen3
4 Prepending /cvmfs/software.eessi.io/versions/2023.06/software/linux...
5 Environment set up to use EESSI (2023.06), have fun!
6 {EESSI 2023.06} trainee00@l55:~$
```

There is **no way** to get *out* of EESSI, you have to login again.

Note how the prompt now shows the **{EESSI 2023.06}** env.

Type `ml ov` to show all modules in an overview:

```
1 {EESSI 2023.06} trainee00@l55:~$ ml ov
2
3 ---/cvmfs/software.eessi.io/versions/2023.06/... ---
4 Abseil                (2)    Eigen                (3)
5 ALL                   (1)    elfutils            (2)
6 amdahl                (1)    ELPA                (2)
```

Where (2) means two versions are installed.

Type `q` to get out of the list, or `page down` all the way.

Even if it looks like `modules`, EESSI uses its own system called **lmod**.

Use `ma` or `module avail` to find installations of a package:

```
1 {EESSI 2023.06} trainee00@l55:~$ module avail mpi
```

Here package names are **not case sensitive**, or search for part of a name:

```
3 {EESSI 2023.06} trainee00@l55:~$ module avail matplot
```

EESSI will only show packages that fit the **current** CPU.

Use `ml` or `module load` to load your package:

```
1 {EESSI 2023.06} trainee00@l55:~$ ml amdahl
```

Now package names are **case sensitive**:

- ▶ **amdahl**
- ▶ **Abseil**
- ▶ **ScaLAPACK**,
- ▶ **GROMACS**

When a package is loaded the **first** time, this might need a while, it is fetched from CERN once.

Use `ml` alone, or `module list` to list your package:

```
1 {EESSI 2023.06} trainee00@l55:~$ ml
2 Currently Loaded Modules:
3 1) GCCcore/12.3.0
4 2) GCC/12.3.0
5 3) numactl/2.0.16-GCCcore-12.3.0
6 4) libxml2/2.11.4-GCCcore-12.3.0
7 5) libpciaccess/0.17-GCCcore-12.3.0
```

Modules load their dependencies automatically.

Type `module purge` to remove all loaded modules:

```
1 {EESSI 2023.06} trainee00@l55:~$ module purge
```

Recommended in job scripts before loading any modules (reproducibility).

Use `module rm` or `module unload` to remove/unload a list of modules:

```
2 {EESSI 2023.06} trainee00@l55:~$ module rm <modulename> <modulename...>
```

```
3 {EESSI 2023.06} trainee00@l55:~$ module unload <modulename> <module...>
```


Use `module show` to show the variables added to your environment by a module:

```
1 {EESSI 2023.06} trainee00@l55:~$ module show amdahl
2 -----
3     /cvmfs/software.eessi.io/versions/2023.06/software/linux/x86_64/...
4 -----
5     ...
6     load("gompi/2023a")
7     load("mpi4py/3.1.4-gompi-2023a")
8     prepend_path("CMAKE_PREFIX_PATH", "/cvmfs/software.eessi.io/versions...
9     setenv("EBROOTAMDAHL", "/cvmfs/software.eessi.io/versions/2023.06/so...
10    ...
```

No need to load a module before inspecting it.

- ☐ Login to VSC-4/5
- ☐ Start EESSI
- ☐ How many **Armadillo** packages are there?
- ☐ Find a **GROMACS** package.
- ☐ Which **mpi** packages are available?
- ☐ Load a **matplotlib** *python* extension.
- ☐ List the loaded modules.