



Spack

on top of

E E S S I

EUROPEAN ENVIRONMENT FOR
SCIENTIFIC SOFTWARE INSTALLATIONS

Wed 20 May 2026

Kenneth Hoste (Ghent University)

Disclaimers



- I'm not a Spack expert...
- For difficult questions, please talk to Massimiliano or Harmen
- Most of the work presented here was done by Loris Ercole, while he was still at CECAM
- Last time I played extensively with Spack (early 2018), I found a major bug in it...

Using Spack to install software on top of EESSI



- Spack can be made aware of software installations provided by EESSI, via a `packages.yaml` file that specifies them as “external packages”; see also https://spack.readthedocs.io/en/latest/packages_yaml.html#external-packages
- As a result, Spack can resolve dependencies via EESSI, rather than installing them again
- Proof-of-concept was implemented by Loris Ercole (CECAM) using QuantumESPRESSO, see <https://eessi.io/docs/blog/2026/02/05/Spack-on-top-of-EESSI-best-of-both-worlds>
- Scripts and detailed setup instructions available via <https://github.com/lorisercole/spood>

Using Spack to install software on top of EESSI



Hands-on demo - Part 1: Set up Spack (*note: patched version is currently required!!*)

See also <https://github.com/lorisercole/spood#installation--use>

```
git clone https://github.com/lorisercole/spack
```

```
cd spack
```

```
git checkout eessi
```

```
cd -
```

```
source spack/share/spack/setup-env.sh
```

```
spack --version
```

Using Spack to install software on top of EESSI

Hands-on demo - Part 2: Initialize EESSI (2023.06), get spood, run demo

```
git clone https://github.com/EESSI/spood  
cd spood  
source /cvmfs/software.eessi.io/versions/2023.06/init/lmod/bash  
./quick_start.sh
```

Note: Here be dragons, see also <https://github.com/EESSI/spood/issues/2>

Key part is `packages.yaml`, which tells Spack about software installed in EESSI:

https://github.com/EESSI/spood/blob/develop/examples/ext_install/packages.yaml

Using Spack to install software on top of EESSI



Hands-on demo - Example output of successful run

```
⇒ Installing quantum-espresso-7.5-zxky252cemgoipfhhqoolnlow2s7qfrd [17/17]
[100%] 72.48 MB @ 21.0 MB/s
⇒ No patches needed for quantum-espresso
⇒ quantum-espresso: Executing phase: 'cmake'
⇒ quantum-espresso: Executing phase: 'build'
⇒ quantum-espresso: Executing phase: 'install'
⇒ quantum-espresso: Successfully installed quantum-espresso-7.5-zxky252cemgoipfhhqoolnlow2s7qfrd
Stage: 5.73s. Cmake: 11.61s. Build: 3m 4.04s. Install: 1.35s. Post-install: 1.04s. Total: 3m 23.88s
[+] /tmp/vsc40023/EESSI-webinar/spood/demo_spood/opt/linux-cascadelake/quantum-espresso-7.5-zxky252cemgoipfhhqoolnlow2s7qfrd

5. Verify the installation:
$ ldd /tmp/vsc40023/EESSI-webinar/spood/demo_spood/opt/linux-cascadelake/quantum-espresso-7.5-zxky252cemgoipfhhqoolnlow2s7qfrd

linux-vdso.so.1 (0x00007ffc0293e000)
libfftw3.so.3 ⇒ /cvmfs/software.eessi.io/versions/2023.06/software/linux/x86_64/intel/haswell/software/FFTW/3.3.10-GC
libfftw3_omp.so.3 ⇒ /cvmfs/software.eessi.io/versions/2023.06/software/linux/x86_64/intel/haswell/software/FFTW/3.3.1
libgomp.so.1 ⇒ /tmp/vsc40023/EESSI-webinar/spood/demo_spood/opt/linux-cascadelake/gcc-runtime-13.2.0-vvcl5hwifb2atnvu
libgfortran.so.5 ⇒ /tmp/vsc40023/EESSI-webinar/spood/demo_spood/opt/linux-cascadelake/gcc-runtime-13.2.0-vvcl5hwifb2a
libm.so.6 ⇒ /cvmfs/software.eessi.io/versions/2023.06/compat/linux/x86_64/lib64/libm.so.6 (0x000014efe6520000)
libmvec.so.1 ⇒ /cvmfs/software.eessi.io/versions/2023.06/compat/linux/x86_64/lib64/libmvec.so.1 (0x000014efe6108000)
libopenblas.so.0 ⇒ /cvmfs/software.eessi.io/versions/2023.06/software/linux/x86_64/intel/haswell/software/OpenBLAS/0.
libgcc_s.so.1 ⇒ /tmp/vsc40023/EESSI-webinar/spood/demo_spood/opt/linux-cascadelake/gcc-runtime-13.2.0-vvcl5hwifb2atnv
libc.so.6 ⇒ /cvmfs/software.eessi.io/versions/2023.06/compat/linux/x86_64/lib64/libc.so.6 (0x000014efe502f000)
/cvmfs/software.eessi.io/versions/2023.06/compat/linux/x86_64/lib64/ld-linux-x86-64.so.2 (0x000014efe6912000)

===== End of Demo =====
```



Spack on top of EESSI in CI (GitHub Actions)

- It is very easy to use EESSI in a CI environment like GitHub Actions
- Especially using [eessi/github-action-eessi](https://github.com/eessi/github-action-eessi)
- Likewise for GitLab runners, see [eessi/gitlab-eessi](https://github.com/eessi/gitlab-eessi)
- So we can also easily use Spack on top of EESSI in CI
- GitHub Actions workflow to use Spack on top of EESSI:

https://github.com/EESSI/spood/blob/develop/.github/workflows/test_build.yml

Spack on top of EESSI in CI (GitHub Actions)



```
steps:
- uses: actions/checkout@v6
- uses: eessi/github-action-eessi@v3
  with:
    eessi_stack_version: '2023.06'
- name: Build and test our code
  run: |
    # from a Python virtual environment
    git clone -b releases/v1.1 https://github.com/spack/spack.git

    # apply patch
    cd spack
    curl -L "https://github.com/spack/spack/compare/develop...lorisercole:spack:eessi.diff" | git apply -3

    # activate spack
    source ./share/spack/setup-env.sh
    # now go back
    cd ..
    ./quick_start.sh demo_dir
shell: bash
```

Next steps



- Make sure that Spack doesn't need to be patched anymore, cfr. <https://github.com/spack/spack/pull/51873> and <https://github.com/spack/spack/compare/develop...lorisercole:spack:eessi>
- More testing: other `spack install` commands, functional & performance tests, ...
- Auto-generate the `packages.yaml`, and ship it in EESSI repository
- Come up with a maintainable way of mapping software names from EESSI to Spack
- Figure out how to deal with “variants”...
- EESSI-extend module for Spack?