EESSI

EUROPEAN ENVIRONMENT FOR SCIENTIFIC SOFTWARE INSTALLATIONS

MultiScale

Beyond the Basics

9th EasyBuild User Meeting 2024 (2024-04-25)

Kenneth Hoste (HPC-UGent)

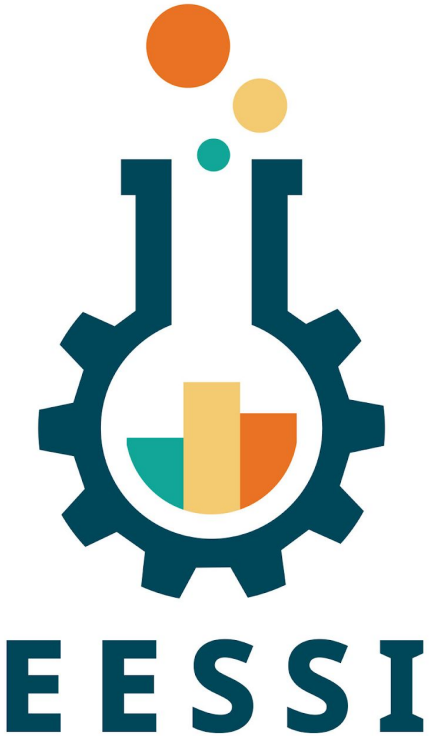
kenneth.hoste@ugent.be

GHENT UNIVERSITY

VLAAMS SUPERCOMPUTER CENTRUM

- EESSI test suite
- NVIDIA GPU support in EESSI
- EESSI in GitHub Actions (CI)

eessi.io/docs



EESSI test suite - Why?

- Ensure quality of the software installations provided by EESSI.
- Provide more insights into performance of the built software:
  - Not only right after the build, but also over time.
  - Performance aspects cover both single-node and multi-node scaling.
- Indicate system inefficiencies with respect to change in system software at the OS level.
- Provide end-users (scientists) with indicative performance numbers of the software for various hardware architectures (both CPU and accelerators).

EESSI test suite - What & how?

eessi.io/docs/test-suite

- A suite of **portable** tests for scientific software, using ReFrame
- EasyBuild already covers unit tests and smoke tests for various applications
- Type of tests to be covered under the test suite:
  - *Integration tests*: Application tests on the build node.
  - *System tests*: Application tests on a completely different system than the build node.
  - *Periodic tests*: Same as system tests, run periodically to check the performance status of the built applications in the central software stack.

EESSI test suite - What is already there?

- Version 0.2.0 includes tests for **GROMACS**, **TensorFlow**, **OSU Micro Benchmarks**.
- The test-suite is **integrated in the CI pipeline** used to build and deploy the software.
    - Filtering various tests based on the build (WIP).
- Each software contains **multiple tests** based on the size of the test and scalability.
- The **node configuration is abstracted** in each of the tests, auto-detect output and the config file inputs that the end user provides are used to determine the scales and the type of tests that can be run.
- **Periodic tests** are constantly running in AWS Magic Castle cluster and EuroHPC systems such as Vega apart from site specific systems in Belgium and Netherlands.

EESSI test suite - What is planned/WIP?

- **More tests** (WIP): LAMMPS, QuantumESPRESSO, CP2K, OpenFOAM, ESPResSo, …
- Inserting **performance references** based on system arch and configuration.
- **Test dashboard** with results from periodic tests running on multiple systems.
- Integrating more low/mid level application tests that normally act as dependencies rather than end-user applications such as OpenBLAS, MKL etc.
- Further documentation, training sessions, etc.

EESSI test suite - Contributors

- Test suite is being developed in scope of MultiXscale WP 1 (SURF, UGent, RUG, UB)
- Several external contributors already (Vrije Universiteit Brussel, CECAM, …)
- See also MultiXscale deliverable D1.2 “Plan for the design of a portable test suite” https://zenodo.org/records/10451718

More info (installation, configuration, usage) via eessi.io/docs/test-suite

EESSI test suite - Demo

- eessi.io/docs/test-suite
- Installation
- Configuration
- Usage

NVIDIA GPU support in EESSI

- Detailed documentation available at eessi.io/docs/gpu
- Problems:
  - 1) We don't know where the NVIDIA GPU driver libraries are in host OS…
  - 2) We can not redistribute the full CUDA installation due to EULA (only runtime libs)
- In EESSI, we provide scripts to deal with both these problems:
  - 1) **`link_nvidia_host_libraries.sh`** to link GPU driver libraries "into" EESSI (requires write access to `/cvmfs/software.eessi.io/host_injections` variant symlink)
  - 2) **`install_cuda_host_injections.sh`** installs full CUDA to subdirectory of `/cvmfs/software.eessi.io/host_injections`

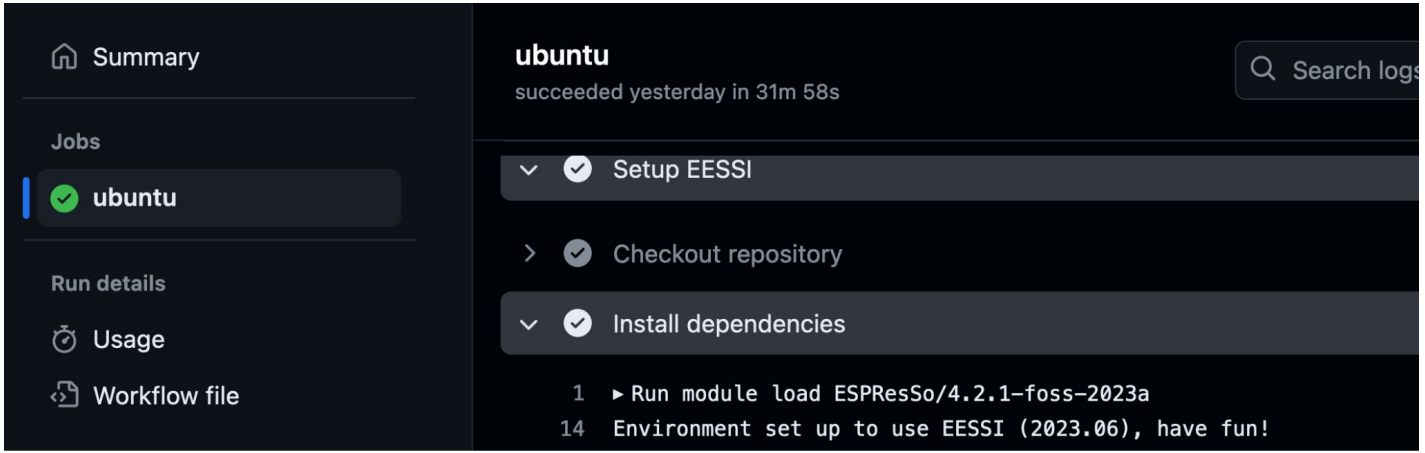
Leveraging EESSI in CI environment

Using EESSI in GitHub Actions is trivial (and works *really* well):

```yaml
name: ubuntu_gromacs
on: [push, pull_request]
jobs:
 build:
   runs-on: ubuntu-latest
   steps:
   - uses: actions/checkout@v2
   - uses: eessi/github-action-eessi@v3
     with:
       eessi_stack_version: '2023.06'
   - name: Test EESSI
     run: |
       module load TensorFlow/2.13.0-foss-2023a
       python test_with_tensorflow.py
     shell: bash
```

github.com/EESSI/github-action-eessi

Leveraging EESSI GitHub Action



https://github.com/pyMBE-dev/pyMBE/blob/main/.github/workflows/testsuite.yml

https://github.com/pyMBE-dev/pyMBE/actions/runs/8815523092/job/24197651600

Acknowledgements

- Funded by the European Union. This work has received funding from the European High Performance Computing Joint Undertaking (JU) and countries participating in the project under grant agreement No 101093169.

- Thanks to Amazon Web Services (AWS) and Microsoft Azure for generously sponsoring the EESSI project with cloud credits, feedback, and guidance.

Website: eessi.io

GitHub: github.com/eessi

Documentation: eessi.github.io/docs

YouTube channel: youtube.com/@eessi_community

Paper (open access): doi.org/10.1002/spe.3075

EESSI support portal: gitlab.com/eessi/support

Monthly online meetings (first Thursday, 2pm CEST)

Join our mailing list & Slack channel