

Multiscale

ISC'24 - Hamburg - 16 May 2024

Lara Peeters (Ghent University, Belgium)

lara.peeters@ugent.be

What is MultiXscale?

- Collaboration between scientific partners (CECAM nodes) who deliver **3 pilot use cases**, and technical partners (EESSI members) who provide the tools to allow application software to be seamlessly used on any available hardware
- MultiXscale targets improving the
 - **Productivity** of scientists who develop and/or use open source codes
 - **Performance** of those codes on EuroHPC hardware (and beyond)
 - **Portability** of the codes and workflows from laptop to server to cloud to HPC
- The focus of MultiXscale is multi-scale modelling and leveraging the opportunities that EuroHPC offers
- Advances the transition towards use of exascale resources for the community

What is MultiXscale?

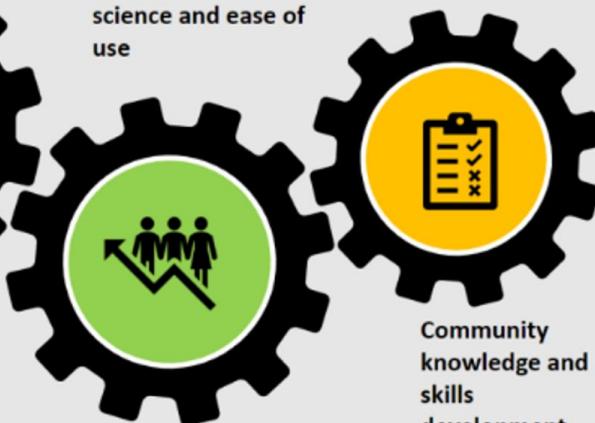
MultiXscale – Performance, Portability, Productivity

Accelerate development for exascale



Application co-design for exascale

Attract user community with groundbreaking science and ease of use



Community knowledge and skills development

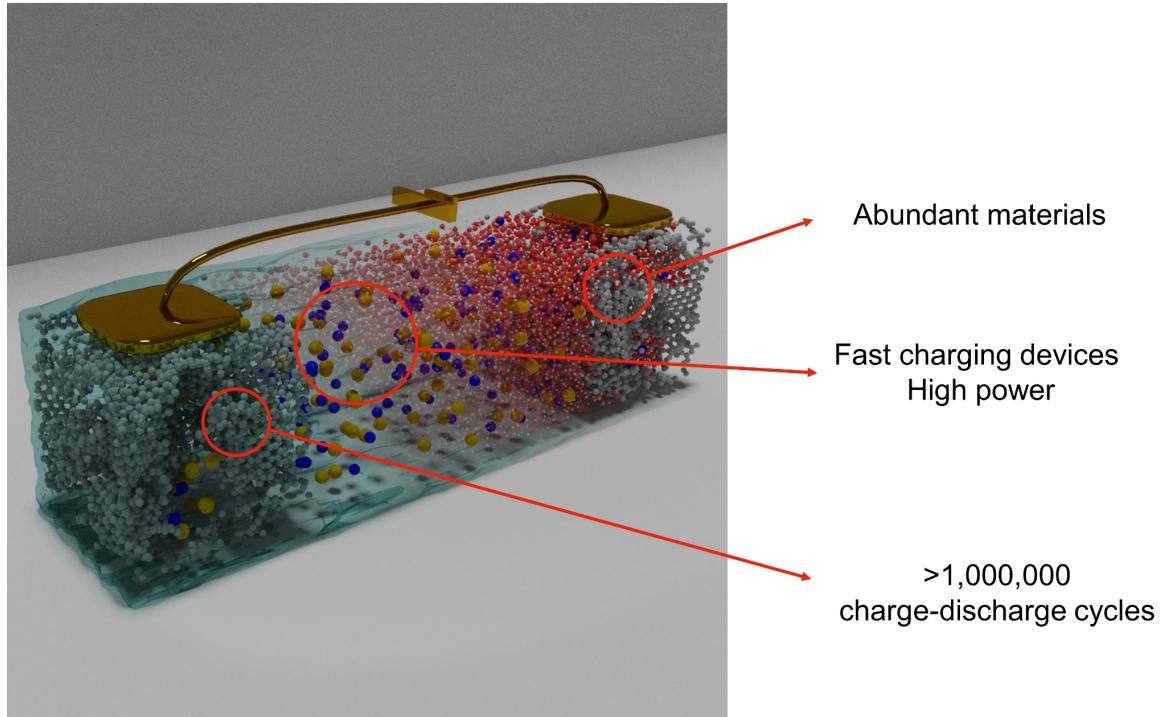
Deploy scientific workflows across exascale systems



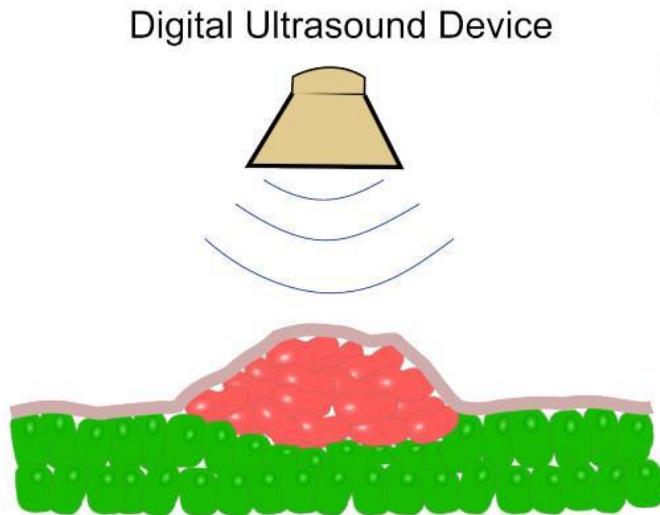
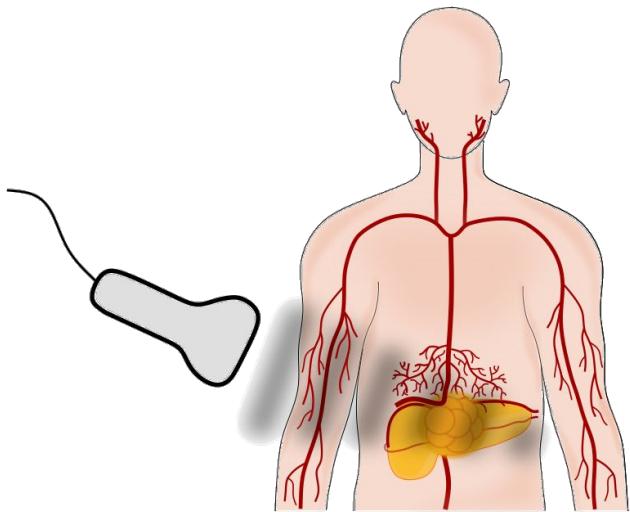
Pilot: Batteries for sustainable energy



- Carbon/carbon supercapacitors are energy storage devices which are particularly attractive for their superior power density.
- Predicting materials with enhanced performance requires extremely large scales.

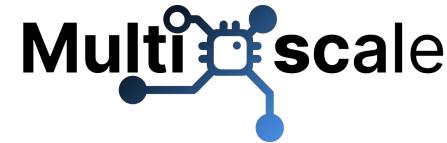


Pilot: Biomedical Applications

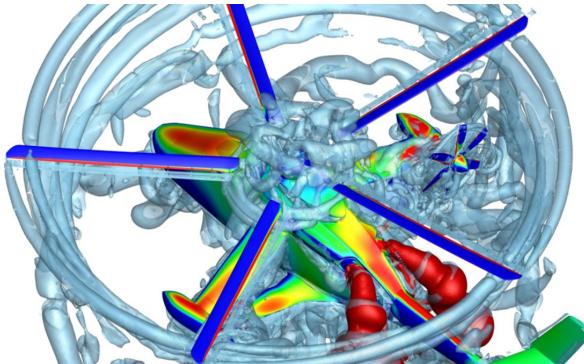


- Create an accurate “digital twin” for the ultrasound process
- Use the digital twin to improve protocols: controlled testing and rational optimization of ultrasound parameters, such as frequency and amplitude

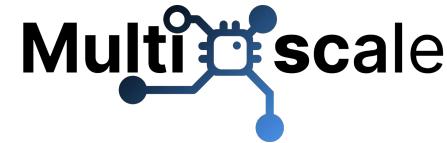
Pilot: Design of innovative helicopters for civil applications



- An extremely complex task which requires proper modelling of the fluid flows around the helicopter body and detailed understanding of the mechanical forces sustained during flight operations.
- Coupling between aerodynamic and structural modelling is limited by the high computational cost associated to high-fidelity fluid dynamics methodologies, especially once applied to moving bodies, as in the case of helicopter rotors.



Scientific motivation requiring technical innovations



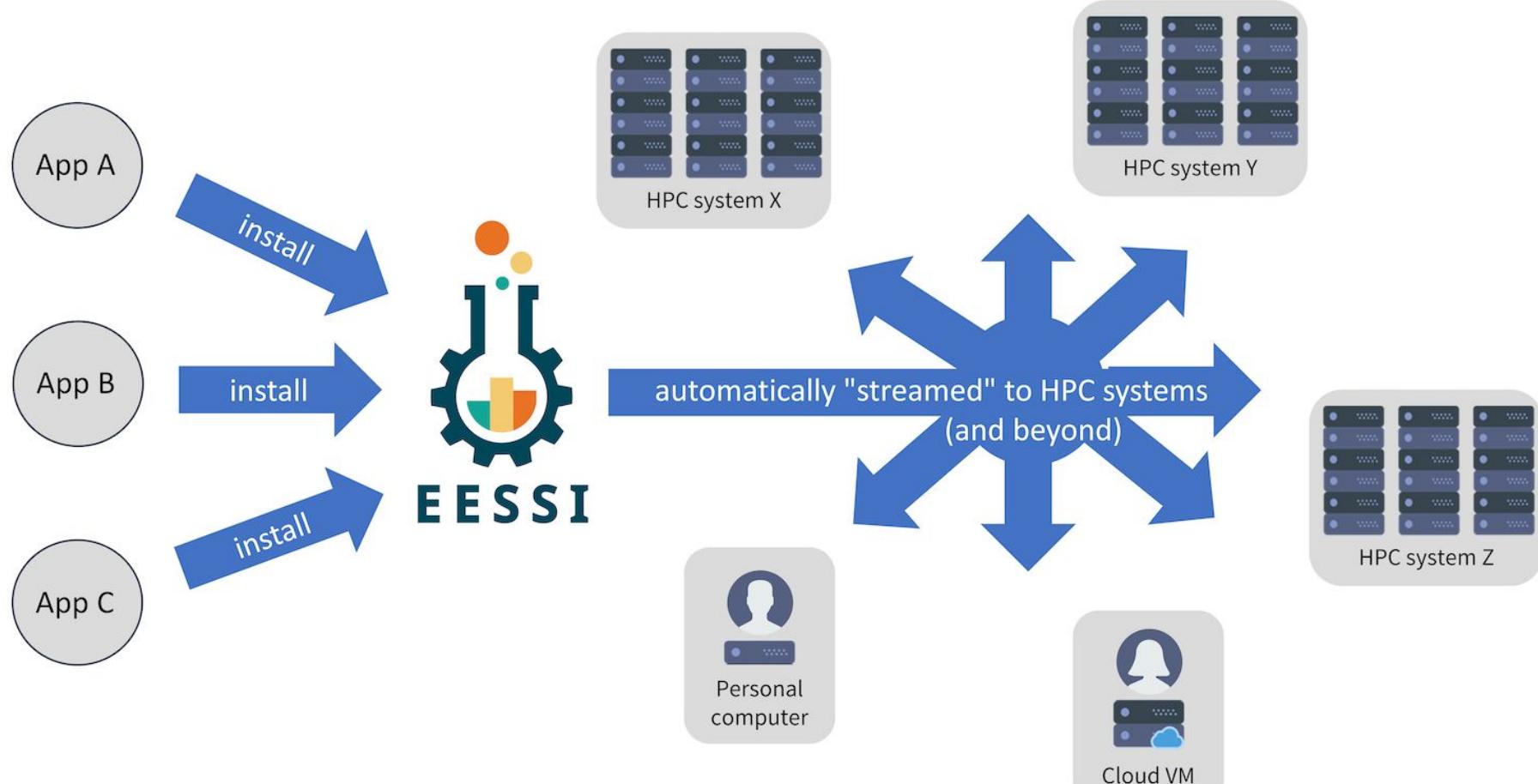
- Each pilot is dealing with phenomena that manifest at multiple scales, and as a result each utilizes multiple methods and a variety of software applications
- The challenge is coupling these together, and making sure they can take advantage of large scale resources
- In Europe, the largest scale resources are to be found (for free!) in **EuroHPC**
- Having access to resources is not enough, we need to get the scientific workflows there and make it as easy as possible for the scientists to use



The role of EESSI in Multi^{scale}

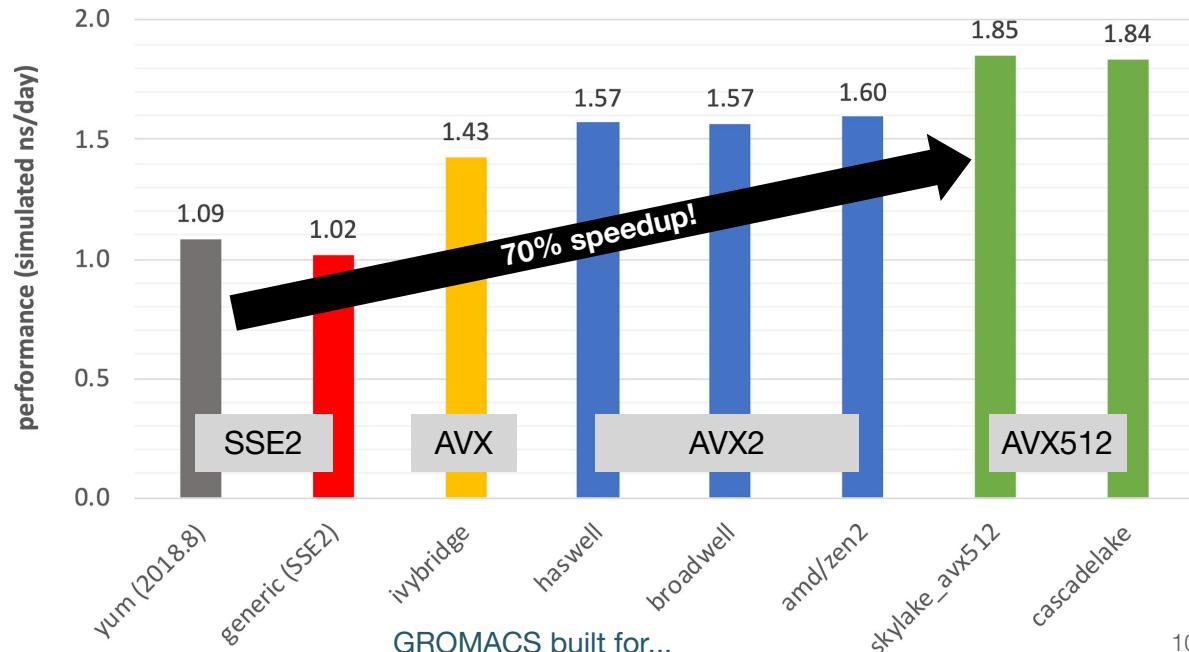


EESSI as a shared software stack



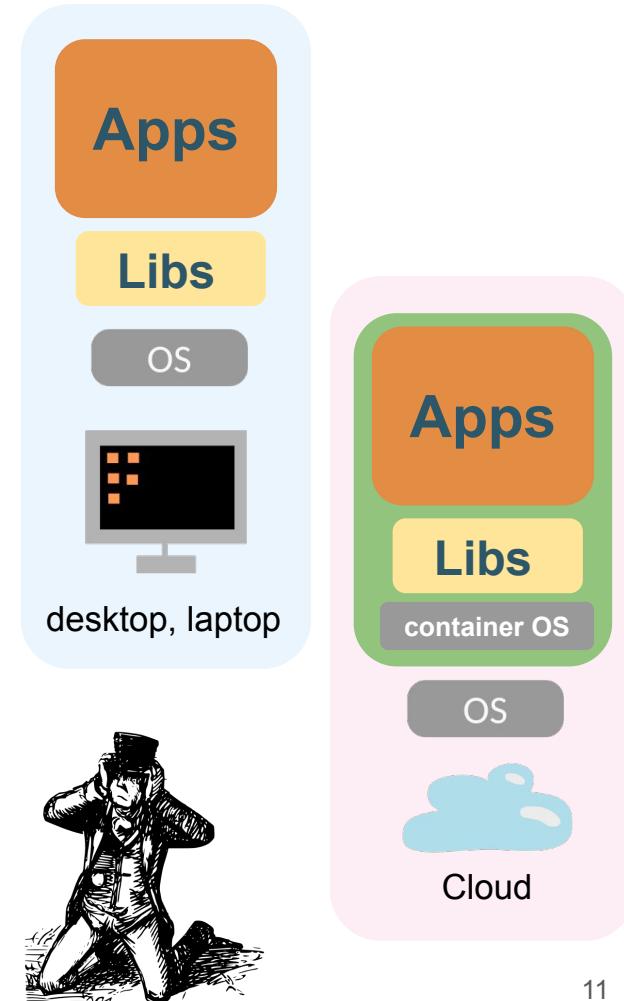
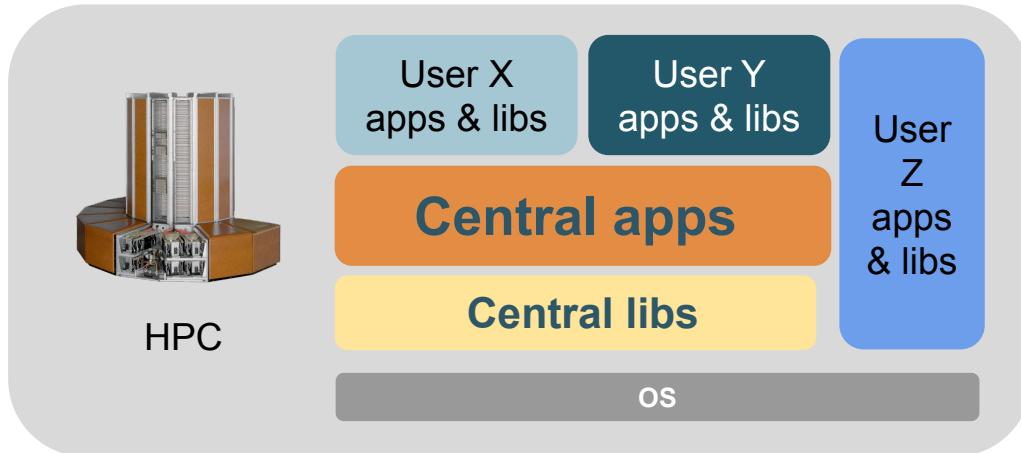
Optimized scientific software installations

- Software should be optimized for the system it will run on
- Impact on performance is often significant for scientific software
- Example: GROMACS 2020.1
(PRACE benchmark, Test Case B)
- Metric: (simulated) ns/day,
higher is better
- Test system: dual-socket
Intel Xeon Gold 6420
(Cascade Lake, 2x18 cores)
- **Performance of different GROMACS binaries,
on exact same hardware/OS**



We need to collaborate more...

- **Too much software** for a single support team to handle
- Different systems (CPU, GPU, OS, ...) => different problems
- Existing tools (EasyBuild, Spack) are **not sufficient anymore...**
- Lots of **duplicate work** across HPC sites and scientists
- **Diverse software stacks** across different platforms



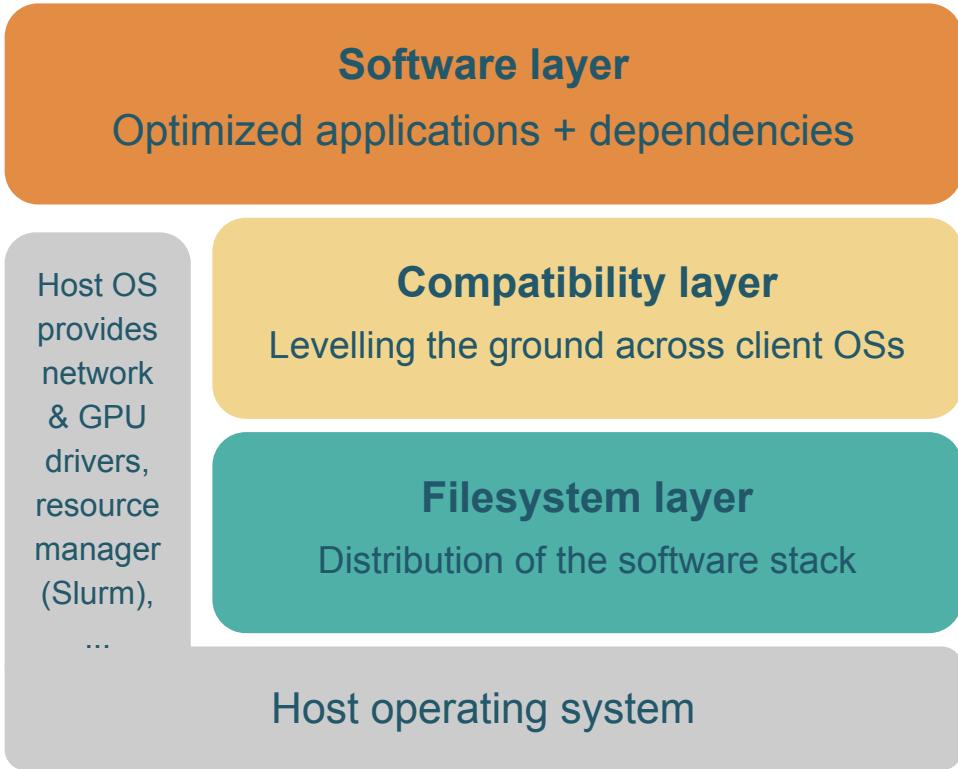
European Environment for Scientific Software Installations

- Public repository of (optimized!) scientific software installations
- Avoid duplicate work by collaborating on a shared software stack
- Uniform way of providing software to users, regardless of the system they use!
- Should work on any Linux OS (incl. WSL) and system architecture
 - From laptops and personal workstations to HPC clusters and cloud
 - Support for different CPUs (AMD, Intel, Arm, RISC-V), interconnects, GPUs, etc.
- Focus on performance, automation, testing, collaboration



<https://eessi.io>

<https://eessi.io/docs>



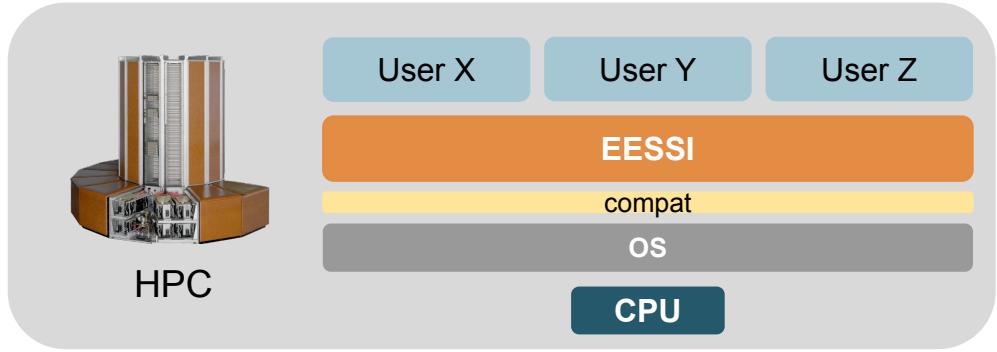
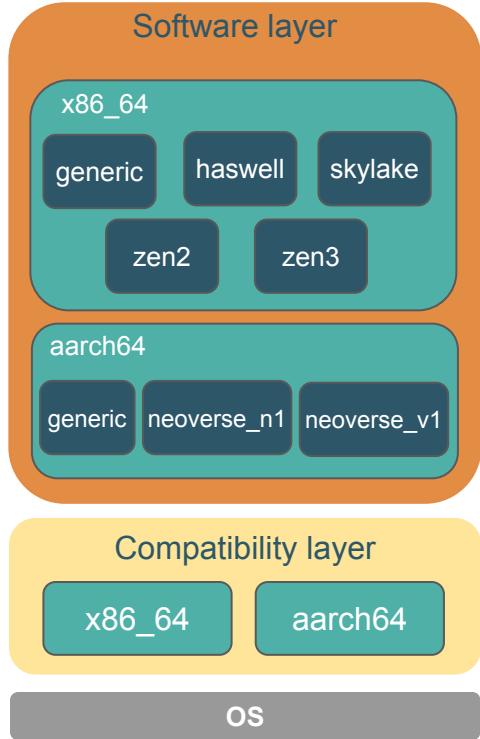
Tutorial “Best Practices for CernVM-FS in HPC”



- <https://multixscale.github.io/cvmfs-tutorial-hpc-best-practices>
- Held online on 4 Dec 2023 (~3 hours), **recorded & available on YouTube**
- Over 200 registrations, ~125 attending the meeting
- Lecture + hands-on demos
- Topics:
 - Introduction to CernVM-FS + EESSI
 - Configuring CernVM-FS: client, Stratum 1 mirror server, proxy server
 - Troubleshooting problems
 - Benchmarking of start-up performance



EESSI to the rescue!



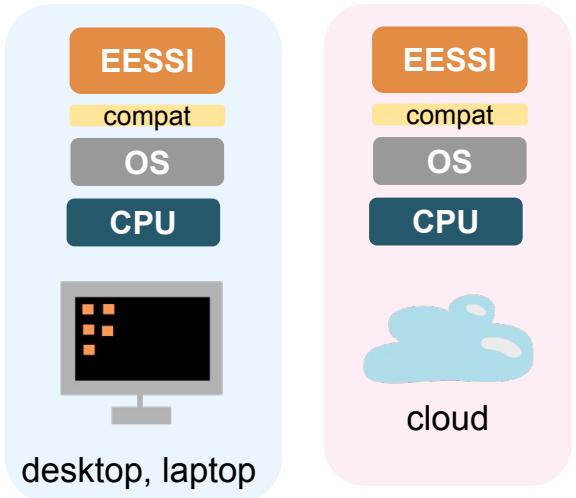
**Shared repository of
(optimized) scientific
software installations**

**Same software stack
everywhere!**

eessi.io

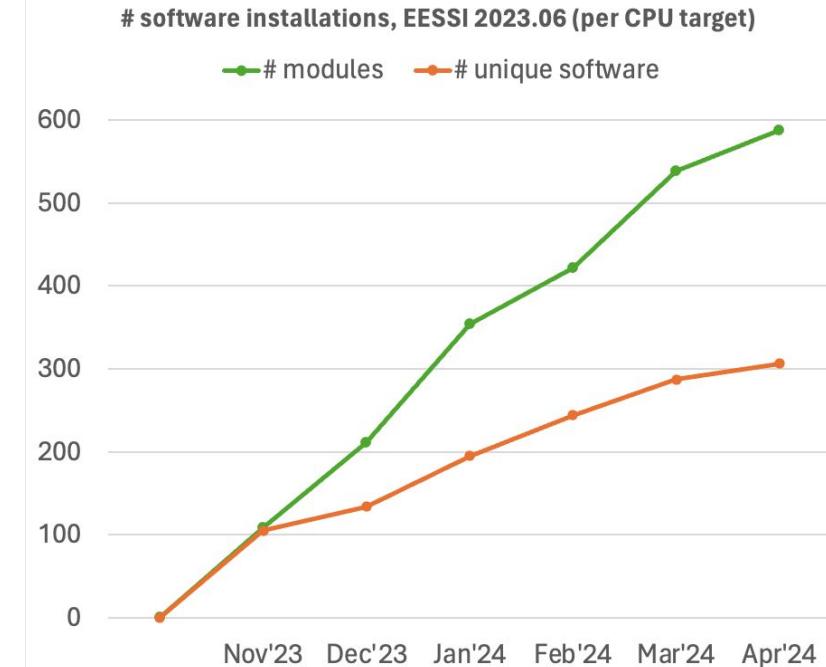
eessi.io/docs

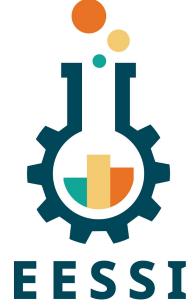
eessi.io/docs/support



Overview of installed software

- ~600 software installations available per CPU target, and increasing every day
 - Including ESPResSo, GROMACS, LAMMPS, OpenFOAM, PyTorch, R, QuantumESPRESSO, TensorFlow, WRF
 - eessi.io/docs/available_software (coming soon!)
- Includes (**but is not limited to!**) applications specific to the MultiXscale CoE
- Focus on recent compiler toolchains: currently targeting foss/2023a and foss/2023b





Getting access to EESSI

- Native installation of CernVM-FS (*requires admin privileges*)
eessi.io/docs/getting_access/native_installation
- Using a container (via Apptainer)
eessi.io/docs/getting_access/eessi_container
- Via cvmfsexec
github.com/cvmfs/cvmfsexec

To check whether you have access to EESSI:

```
ls /cvmfs/software.eessi.io
```

Accessing EESSI via CernVM-FS



```
# Native installation
# Installation commands for RHEL-based distros
# like CentOS, Rocky Linux, Almalinux, Fedora, ...

# install CernVM-FS
sudo yum install -y
https://ecsft.cern.ch/dist/cvmfs/cvmfs-release/cvmfs-release-latest.noarch.rpm
sudo yum install -y cvmfs

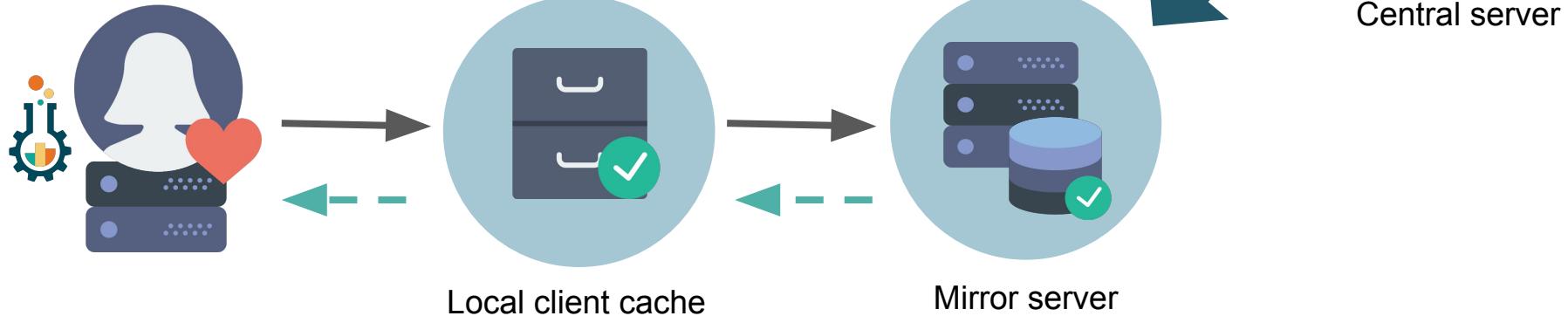
# create client configuration file for CernVM-FS
# (no proxy, 10GB local CernVM-FS client cache)
sudo bash -c "echo 'CVMFS_CLIENT_PROFILE=\"single\"' > /etc/cvmfs/default.local"
sudo bash -c "echo 'CVMFS_QUOTA_LIMIT=10000' >> /etc/cvmfs/default.local"

# Make sure that EESSI CernVM-FS repository is accessible
sudo cvmfs_config setup
```

Alternative ways of accessing EESSI are available, via a container image, via cvmfsexec, ...
eessi.io/docs/getting_access/native_installation - eessi.io/docs/getting_access/eessi_container

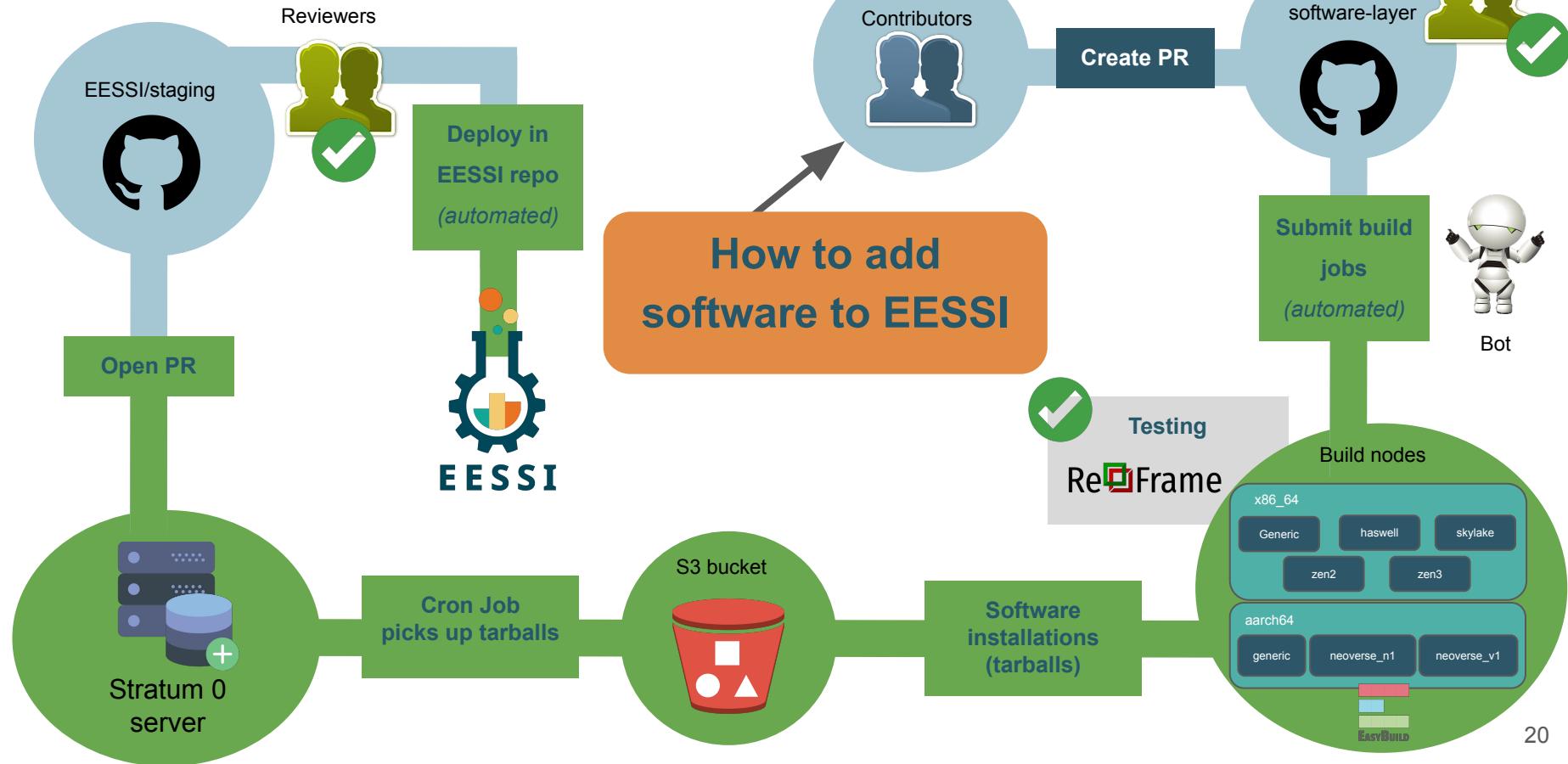
The EESSI User Experience

```
$ source /cvmfs/software.eessi.io/versions/2023.06/init/bash  
{EESSI 2023.06} $ module load GROMACS/2024.1-foss-2023b  
{EESSI 2023.06} $ gmx mdrun ...
```



EESSI provides **on-demand streaming**
of (scientific) software (like music, TV-series, ...)

EESSI as community project



EESSI test suite

ReFrame
eessi.io/docs/test-suite



- Ensure quality of the software installations provided by EESSI
- A suite of **portable** tests for scientific software, using ReFrame
- Version 0.2.0 includes tests for **GROMACS**, **TensorFlow**, **OSU Micro Benchmarks**.
- See also MultiXscale deliverable D1.2 “**Plan for the design of a portable test suite**”
<https://zenodo.org/records/10451718>
- Work-in-progress:
 - More tests: 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



Leveraging EESSI in CI environment

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

```
name: ubuntu_gromacs
on: [push, pull_request]          github.com/EESSI/github-action-eessi
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
```



Leveraging EESSI GitHub Action



The screenshot shows a GitHub Actions job summary for the 'ubuntu' job. The left sidebar lists 'Summary', 'Jobs' (with 'ubuntu' selected), 'Run details', 'Usage', and 'Workflow file'. The main area shows the 'ubuntu' job status as 'succeeded yesterday in 31m 58s'. It details three steps: 'Setup EESSI' (successful), 'Checkout repository' (successful), and 'Install dependencies' (successful). The 'Install dependencies' step shows command-line output:

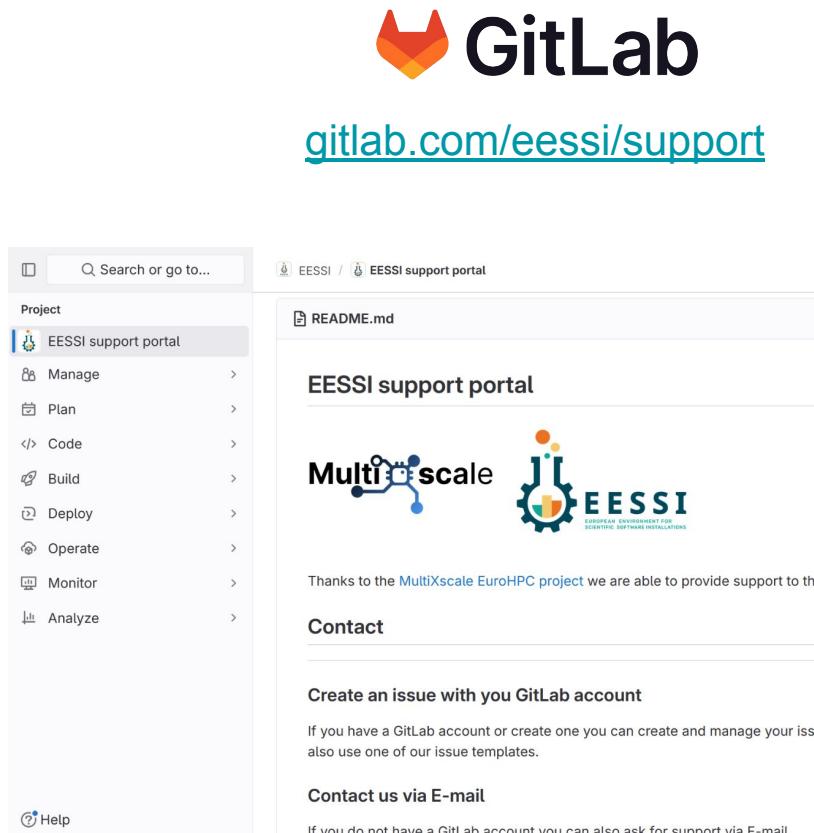
```
1 ► Run module load ESPResSo/4.2.1-foss-2023a
14 Environment set up to use EESSI (2023.06), have fun!
```

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

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

Getting support for EESSI

- Via GitLab, or via email: support@eessi.io
- Report problems
- Ask questions
- Request software
- Get help with contributing
- Suggest features
- Confidential tickets possible (security issues, ...)



The screenshot shows the GitLab interface for the EESSI support portal. On the left is a sidebar with navigation links: Project, Manage, Plan, Code, Build, Deploy, Operate, Monitor, and Analyze. The main area displays a README.md file with the following content:

```
README.md
```

EESSI support portal

Thanks to the [MultiXscale EuroHPC project](#) we are able to provide support to the u

Contact

Create an issue with your GitLab account

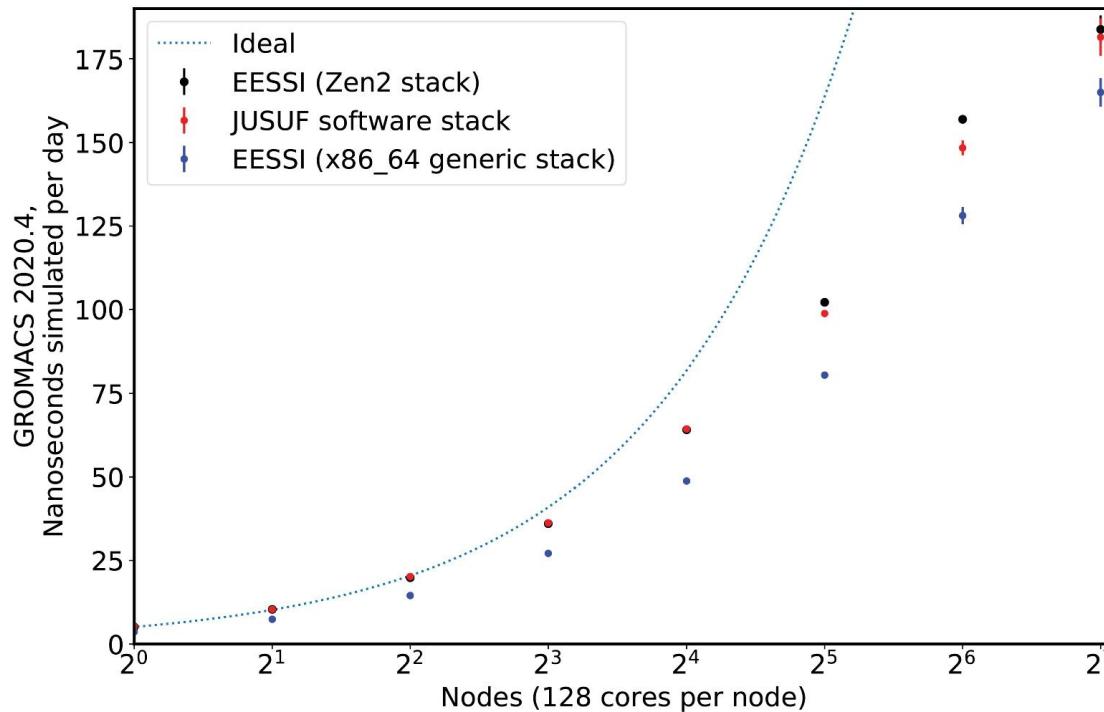
If you have a GitLab account or create one you can create and manage your issue also use one of our issue templates.

Contact us via E-mail

If you do not have a GitLab account you can also ask for support via E-mail.

Dedicated support team, thanks to EuroHPC Centre-of-Excellence





Paper includes **proof-of-concept performance evaluation** compared to system software stack, performed at JUSUF @ JSC using GROMACS 2020.4, up to 16,384 cores (CPU-only)

EESSI in a nutshell

- **On-demand streaming of optimized scientific software installations**
- **Works on any Linux distribution** thanks to EESSI compat layer
- **Uniform software stack** across various systems: laptop, HPC, cloud, ...
- Community-oriented: **let's tackle the challenges we see together!**



Acknowledgements



Co-funded by
the European Union



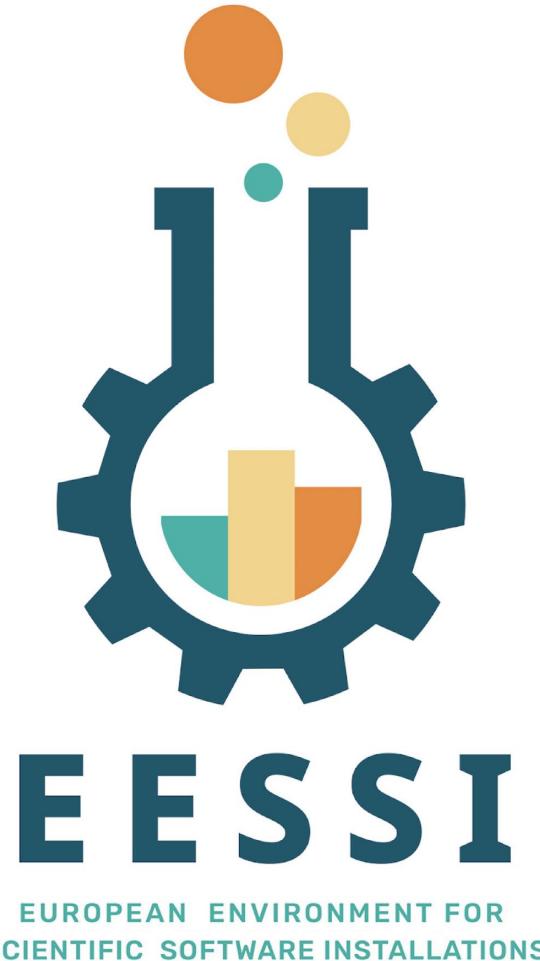
EuroHPC
Joint Undertaking

- 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.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

Multiscale



Co-funded by
the European Union



Web page: multixscale.eu

Facebook: [MultiXscale](#)

Twitter: [@MultiXscale](#)

LinkedIn: [MultiXscale](#)

Youtube: [@MultiXscale](#)

