EESSI

EUROPEAN ENVIRONMENT FOR SCIENTIFIC SOFTWARE INSTALLATIONS

Kenneth Hoste (HPC-UGent, Belgium)

13 Sept 2022

CernVM workshop @ Amsterdam

whoami

kenneth.hoste@ugent.be

@boegel (GitHub)

@kehoste (Twitter)

EasyBuild Slack

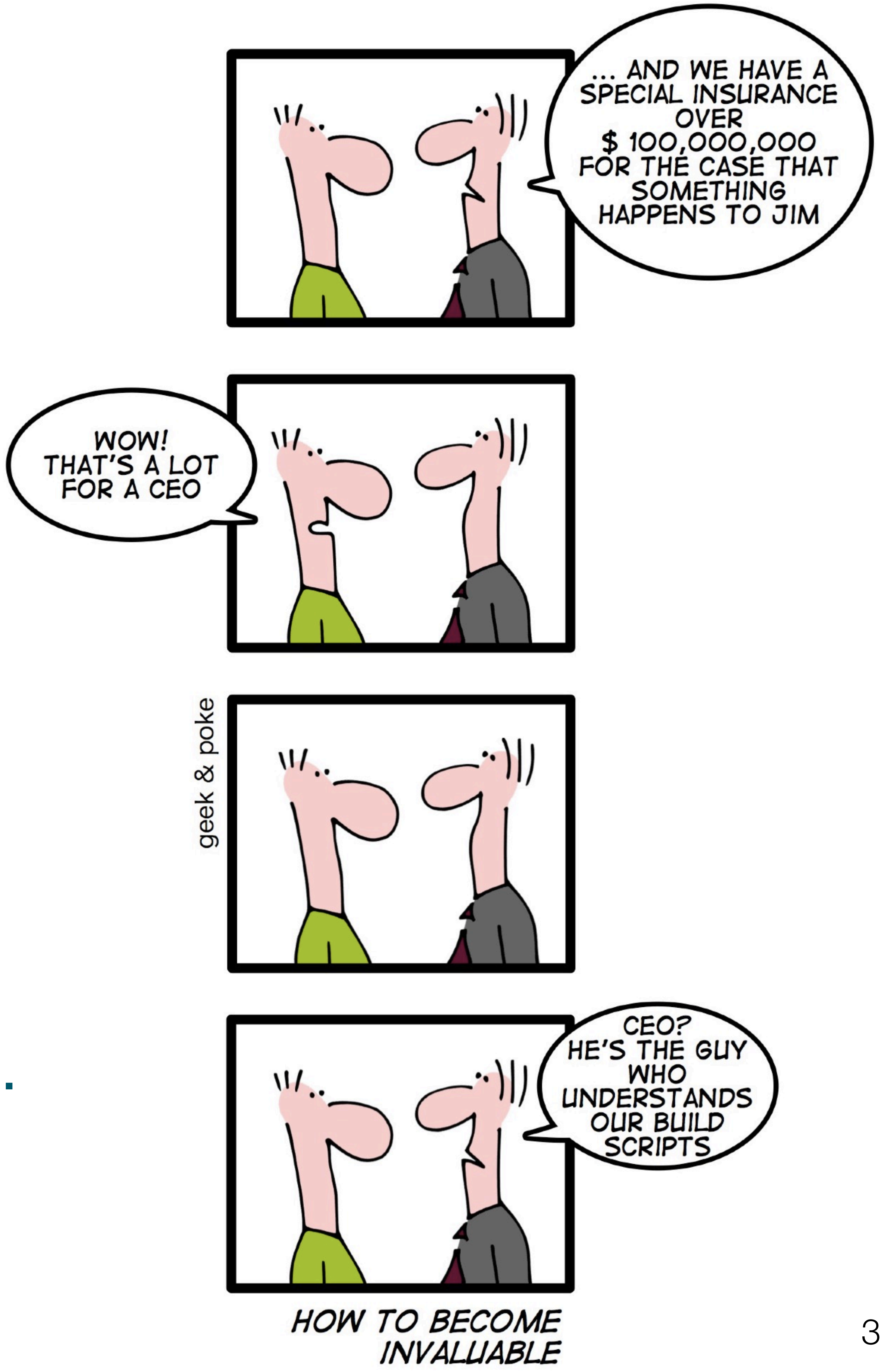
LinkedIn

- Computer scientist with Masters + PhD from Ghent University (BE)
- Joined HPC team at Ghent University in October 2010
- Main tasks: user support, incl. installation of scientific software
- Inherited maintenance of EasyBuild in 2011
- Slowly also became **EasyBuild lead developer & release manager** ...
- I like beer, loud music, FOSS, stickers, dad jokes
- I don't like C++ compiler errors, CMake, SCons, Bazel, TensorFlow, OpenFOAM, ...

Getting Scientific Software Installed

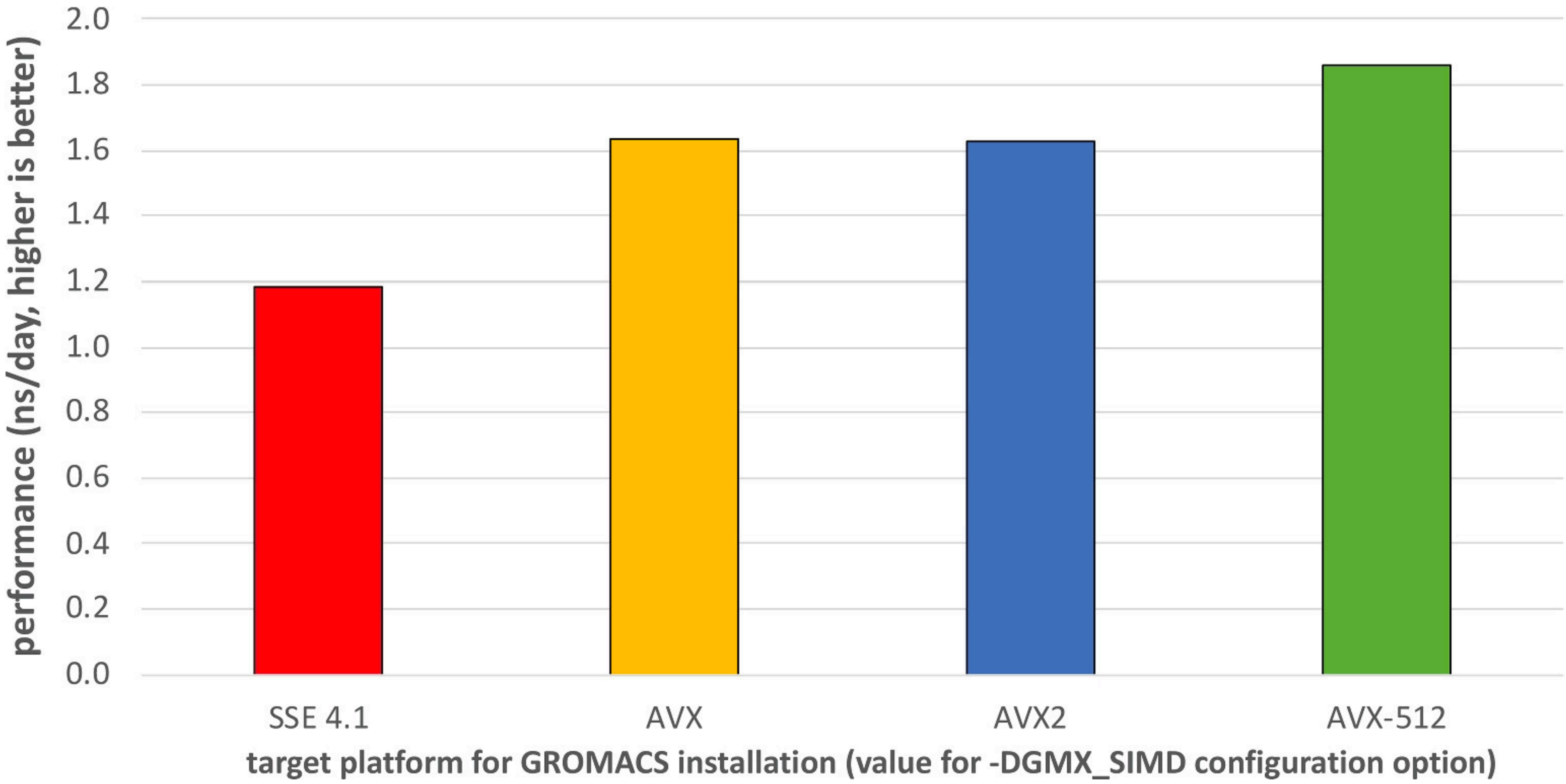
Installing scientific software is (still) a tremendous problem for High Performance Computing (HPC) sites all around the world.

- Ideally built from source - performance is key!
- Tedious, time-consuming, frustrating,
  sometimes simply not worth the (manual) effort...
- **Huge burden on researchers & HPC support teams**
- Installation requests are ~15% of HPC-UGent helpdesk tickets,
  but they consume ***way*** more than 15% of our user support time...
- Very little collaboration among HPC sites (until recently)


... AND WE HAVE A SPECIAL INSURANCE OVER $ 100,000,000 FOR THE CASE THAT SOMETHING HAPPENS TO JIM
WOW! THAT'S A LOT FOR A CEO
geek & poke
CEO? HE'S THE GUY WHO UNDERSTANDS OUR BUILD SCRIPTS
HOW TO BECOME INVALUABLE

http://geekandpoke.typepad.com/geekandpoke/2010/05/how-to-become-invaluable.html

Don’t sacrifice performance for “mobility of compute”



- Performance evaluation of GROMACS with different binaries on an Intel Cascade Lake system
- Impact of target ISA on GROMACS: **57% speedup** on a system that support AVX-512 instructions!
- When using containers (and CernVM-FS?) performance is often sacrificed for "mobility of compute"...

(all technical details in EESSI paper, see doi.org/10.1002/spe.3075)

easybuild to the rescue

easybuild.io

docs.easybuild.io

youtube.com/c/easybuilders

twitter.com/easy_build

easybuild.io/tutorial

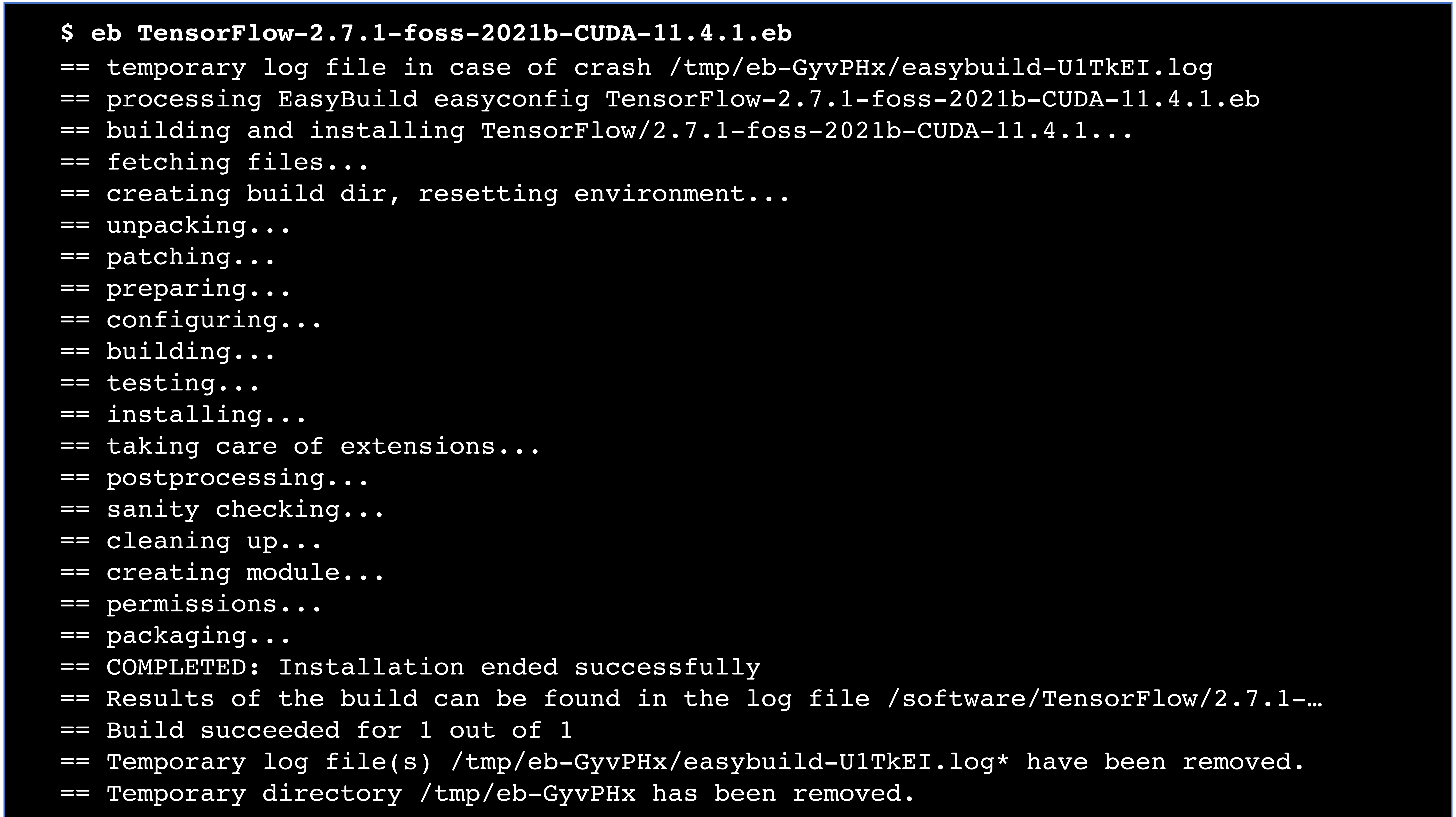
easybuild.io/eum

easybuild.io/tech-talks

- **Tool to make installation of scientific software easier** (preferably from source)
- Created in 2009, available as open source software (GPLv2) since 2012 via GitHub + PyPI
- Focus on Linux & HPC systems, specific attention to ***performance*** of installed software
- By default: optimise for CPU microarchitecture of system on which software is being compiled
- Implemented in Python, supports both Python 2.7 and 3.5+
- Integrates with environment modules tool (Tcl-based Environment Modules + Lmod)
- Supports different compiler toolchains + over 2,700 software packages (excl. extensions)
- Created by HPC-UGent team, now supported & developed by a **worldwide community**...

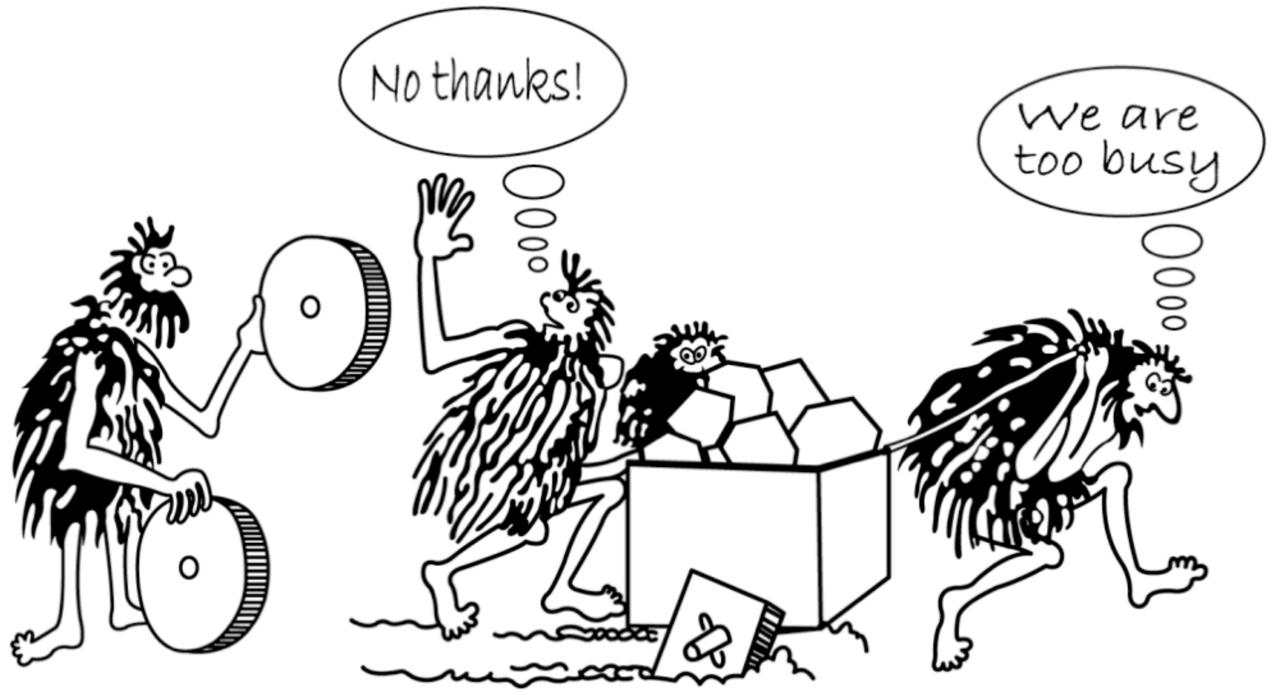
Installing TensorFlow from source, with ease

easybuild.io



```
$ eb TensorFlow-2.7.1-foss-2021b-CUDA-11.4.1.eb
== temporary log file in case of crash /tmp/eb-GyvPHx/easybuild-U1TkEI.log
== processing EasyBuild easyconfig TensorFlow-2.7.1-foss-2021b-CUDA-11.4.1.eb
== building and installing TensorFlow/2.7.1-foss-2021b-CUDA-11.4.1...
== fetching files...
== creating build dir, resetting environment...
== unpacking...
== patching...
== preparing...
== configuring...
== building...
== testing...
== installing...
== taking care of extensions...
== postprocessing...
== sanity checking...
== cleaning up...
== creating module...
== permissions...
== packaging...
== COMPLETED: Installation ended successfully
== Results of the build can be found in the log file /software/TensorFlow/2.7.1-…
== Build succeeded for 1 out of 1
== Temporary log file(s) /tmp/eb-GyvPHx/easybuild-U1TkEI.log* have been removed.
== Temporary directory /tmp/eb-GyvPHx has been removed.
```

Landscape of computational science is changing



- **Explosion of available scientific software applications**
    - Broader scope in terms of scientific domains (bioinformatics, AI, ...)
    - Fuelled by recent shifts in scientific community
        - Pressure to publish code along with research articles + popularity of GitHub
        - Wider adoption of HPC across scientific domains (GPUs in bioinformatics, ...)
- **Increasing interest in using cloud infrastructure** (both private and commercial)
- **Increasing variety in processor (micro)architectures**
    - Intel + AMD, Arm, POWER, soon also RISC-V? (cfr. European Processor Initiative, ...)
    - In addition, GPUs (NVIDIA, AMD, soon Intel?) and other accelerators (TPUs, IPUs, ...)
- **In strong contrast: available manpower in HPC support teams...**

It is time to take the next step…

- EasyBuild helps a lot with managing scientific software stacks, but it's not enough anymore...
- **HPC sites are *still* losing way too much time in getting scientific software installed**
  - Installation requests for new software often require significant investment & expertise
  - Problems occur due to site-specific differences in system setup & configuration
  - Small details (OS packages, disk partitioning, ...) can cause trouble
- **Situation is getting *worse*, not better**: more software, increasing variety in hardware, ...
- **There is a lot of potential for more extensive collaboration beyond installing software**
  - Ensuring correctness of installations w.r.t. scientific results they produce
  - Evaluating performance of provided installations + performance monitoring
  - ReFrame developed by CSCS is definitely a step in the right direction!

Introducing EESSI…

- *European Environment for Scientific Software Installations*
- Collaborative project, by and for the computational science community
- **Shared central stack of (optimised) scientific software installations**
- Uniform way of providing software to users, regardless of system they use
- Should work regardless of OS and system architecture (HPC, cloud, laptop,…)
- Focus on **performance**, automation, testing, collaboration

EESSI

EUROPEAN ENVIRONMENT FOR
SCIENTIFIC SOFTWARE INSTALLATIONS

www.eessi-hpc.org

eessi.github.io/docs (try out pilot repository!)

Goals of EESSI, and how we can achieve them

CernVM-FS

Global distribution of scientific software installations, transparent for end users

“Netflix for scientific software”

Goals of EESSI, and how we can achieve them

CernVM-FS

Global distribution of scientific software installations, transparent for end users

“Netflix for scientific software”

gentoo

Abstraction from OS, one software stack (per type of CPU) to rule them all

“Containers without the containing”

Goals of EESSI, and how we can achieve them

EESSI

CernVM-FS

Global distribution of scientific software installations, transparent for end users

“Netflix for scientific software”

gentoo

Abstraction from OS, one software stack (per type of CPU) to rule them all

“Containers without the containing”

Provide optimised software installations for specific CPUs
+ automatically select best suitable

Don’t sacrifice performance for mobility of compute!

easybuild Lmod arch spec

Goals of EESSI, and how we can achieve them

CernVM-FS

gentoo

EESSI

Global distribution of scientific software installations, transparent for end users

“Netflix for scientific software”

Abstraction from OS, one software stack (per type of CPU) to rule them all

“Containers without the containing”

Provide optimised software installations for specific CPUs + automatically select best suitable

Don’t sacrifice performance for mobility of compute!

Test provided software installations, monitor performance over time

Go (well) beyond just installing software: ensure correctness, performance, …

easybuild Lmod archspec

ReFrame

High-level overview of EESSI

Software layer

optimized installations of applications + dependencies

Compatibility layer

levelling the ground across Linux distros

("containers without the containing")

Filesystem layer

global distribution of the software stack

Host OS provides network & GPU drivers, resource manager (Slurm), ...

host operating system (any Linux distribution)

Testing

Heavily inspired by compute canada software stack

From zero to science in 3 steps

- Step 1: Get access to EESSI repository, either through:
  - system-wide CernVM-FS installation (requires admin privileges)
  - container with CernVM-FS + EESSI configuration pre-installed
- Step 2: Set up environment: source EESSI init script
- Step 3: Load module(s) and run!

eessi.github.io/docs/pilot

github.com/eessi/eessi-demo

```
# Assumption: EESSI is accessible (CernVM-FS is installed + EESSI configuration is in place)

# Step 2: set up environment (CPU architecture is detected automatically!)
$ source /cvmfs/pilot.eessi-hpc.org/latest/init/bash

# Step 3: load module(s) to activate software (check with ‘module avail’), and run!
[EESSI pilot 2021.12] $ module load GROMACS
[EESSI pilot 2021.12] $ gmx mdrun ...
```

Use cases

- A **uniform software stack** across HPC clusters, clouds, servers, and laptops
- Can be leveraged in **continuous integration** (CI) environments for software testing
- Significantly facilitates setting up **infrastructure for HPC training**
  - Using throwaway clusters in the cloud (Azure, AWS, …)
  - Participants can easily get access to same software environment "at home"
- **Enhanced collaboration** with software developers and application experts
  - Work towards "vetting" of scientific software installations included in EESSI
  - Relieve developers from burden of getting their software (properly) installed
- **Community software** and **portable workflows**

Current status of EESSI

EESSI

status.eessi-infra.org

- Started as an idea early 2020, effort kickstarted shortly after
- Monthly online meetings since mid 2020
- Currently an unfunded "side project", actively pursuing dedicated project funding...
- **Work-in-progress, proof-of-concept pilot repository available** - eessi.github.io/docs/pilot
  - Supports `x86_64` (Intel, AMD), `aarch64` (Arm 64-bit), `ppc64le` (POWER)
  - Included software: Bioconductor, GROMACS, OpenFOAM, R, TensorFlow, ...
  - Targets: Intel Haswell + Skylake, AMD Rome + Milan, AWS Graviton2 + 3, POWER9 (partial),
    + `generic` fallback for `aarch64` (Raspberry Pi!), `ppc64le`, `x86_64`
- Sponsored credits provided by both AWS and Azure to support development
- Actively developing monitoring, tooling, test suite, GPU support, contribution workflow, etc.

Future work

- **Make EESSI ready for production use**
  - Switch to eessi.io domain (more neutral, EESSI is not specific to HPC)
  - New Stratum-0 (dedicated hardware + Yubikey) + Stratum-1 servers (managed by small team?)
  - Follow up on (impact of) necessary security updates in compatibility layer (glibc, …)
  - Stabilise support for NVIDIA GPUs + implement bot to automatically build/deploy more software
  - Include more scientific software (all open source software that EasyBuild supports?)
- **Add support for RISC-V CPUs** (`riscv64`)
  - CernVM-FS already works on RISC-V (?), Gentoo Prefix bootstrapping works (cfr. GSoC project)
  - Next steps: build EESSI compat layer + see which scientific software works on top
- Also support other types of GPUs: AMD, Intel, …

Funding via MultiXscale EuroHPC project

- EuroHPC Centre of Excellence:
  Exascale-oriented application co-design and delivery for multiscale simulations

- Total funding: € 6,000,000 (2023-2027, 16 partners incl. HPC-UGent)

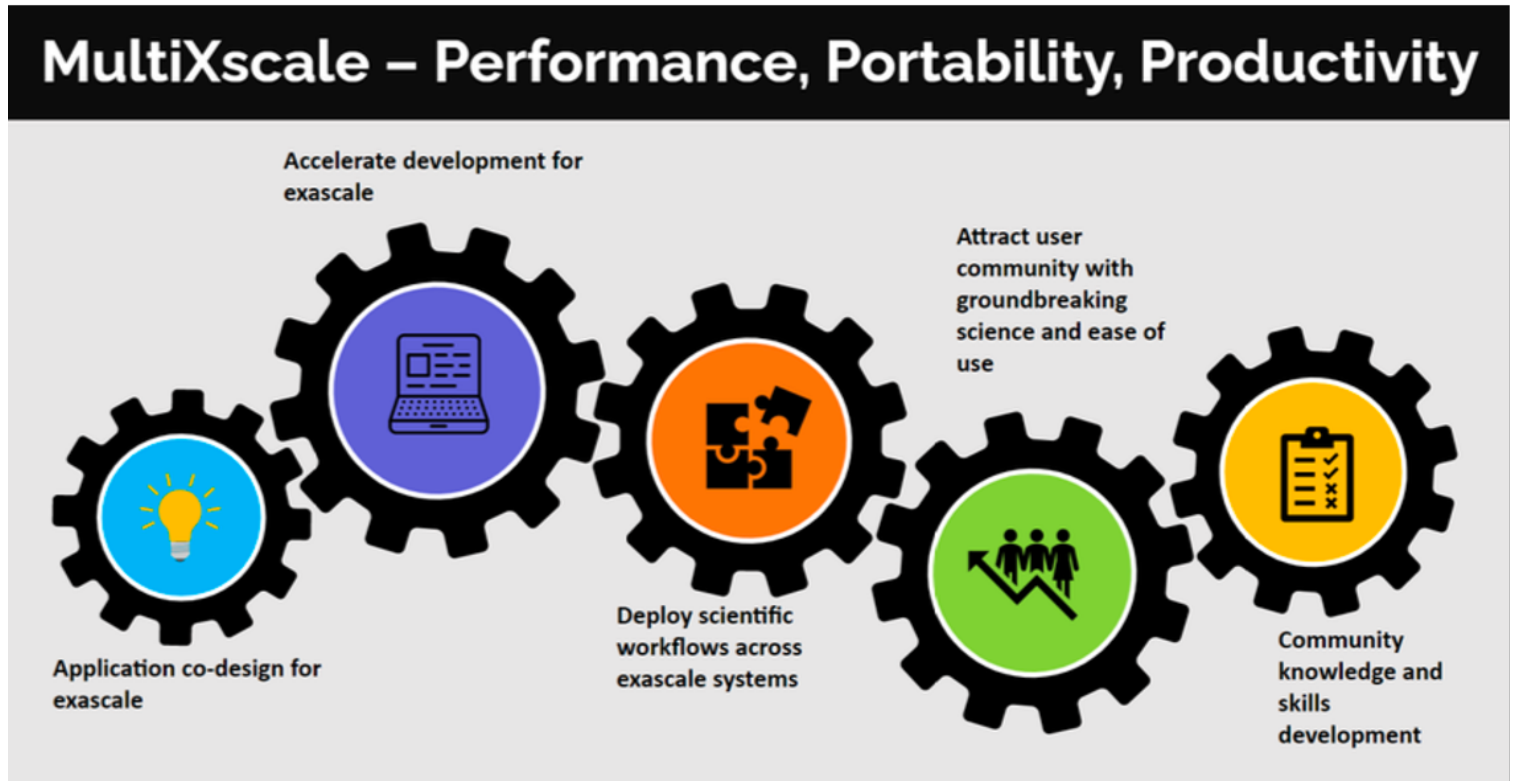
- Two aspects:

  - **Services to facilitate development and deployment of scientific software** =>

  - Development of software for efficient large-scale multiscale simulations

- Project proposal was accepted for funding, currently working towards grant agreement

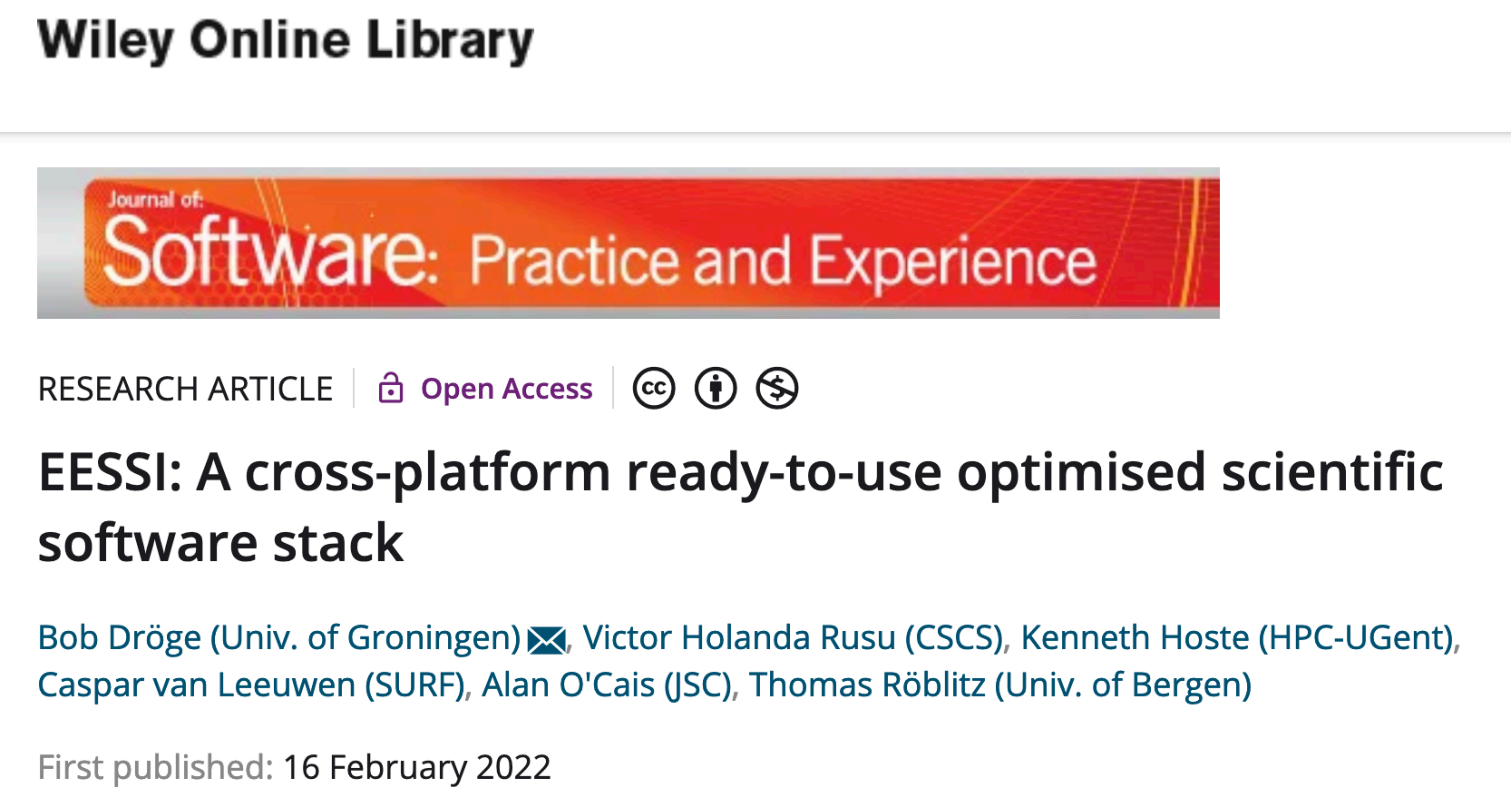
- Expected timeframe: Feb’23 - Jan’28


MultiXscale – Performance, Portability, Productivity
Accelerate development for exascale
Attract user community with groundbreaking science and ease of use
Application co-design for exascale
Deploy scientific workflows across exascale systems
Community knowledge and skills development

EESSI paper (open access)

doi.org/10.1002/spe.3075

Detailed overview of motivation, project goals, design, challenges & limitations, use cases, etc.



Wiley Online Library

Journal of: Software: Practice and Experience

RESEARCH ARTICLE | Open Access

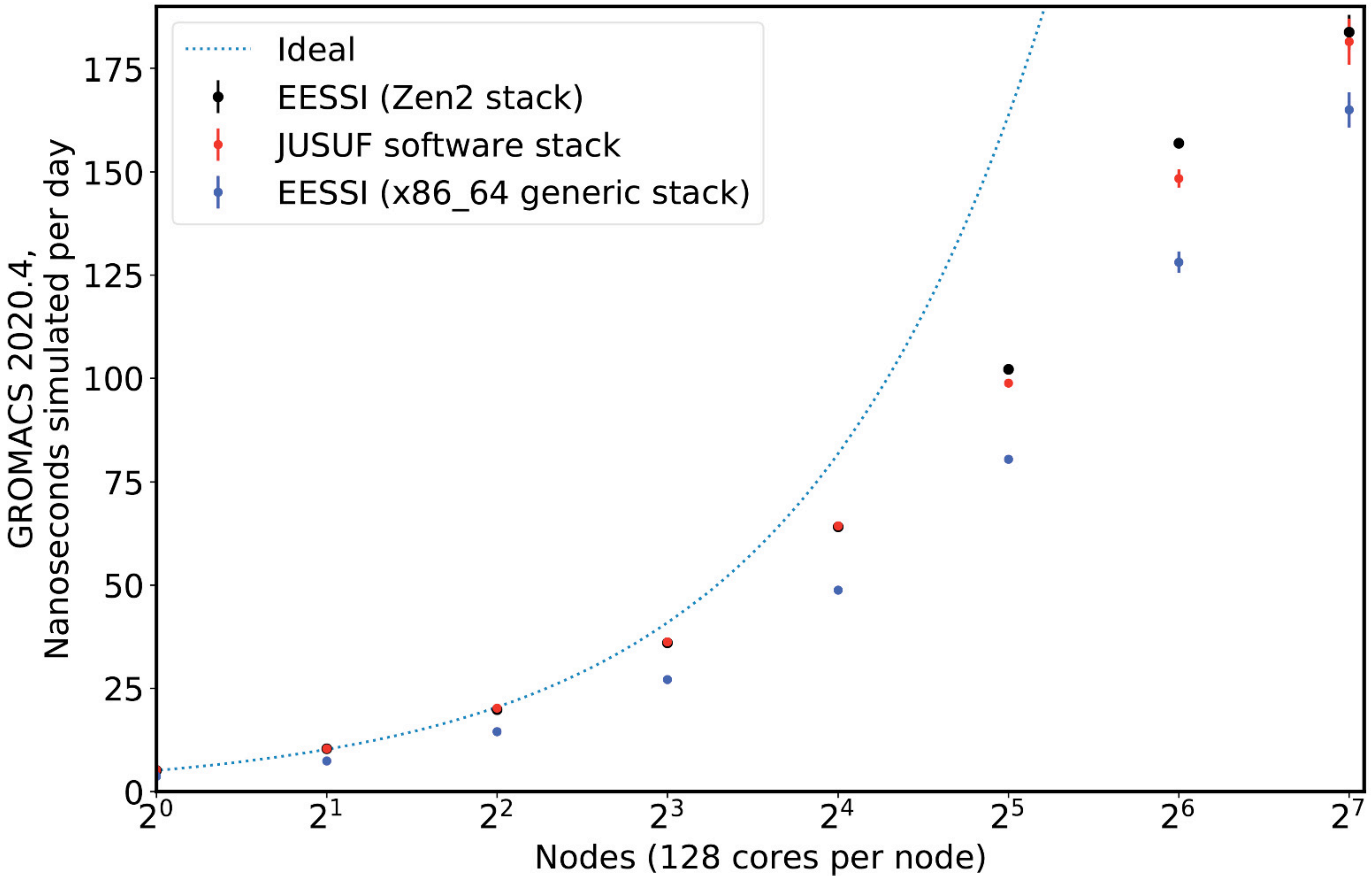
EESSI: A cross-platform ready-to-use optimised scientific software stack

Bob Dröge (Univ. of Groningen), Victor Holanda Rusu (CSCS), Kenneth Hoste (HPC-UGent), Caspar van Leeuwen (SURF), Alan O'Cais (JSC), Thomas Röblitz (Univ. of Bergen)

First published: 16 February 2022

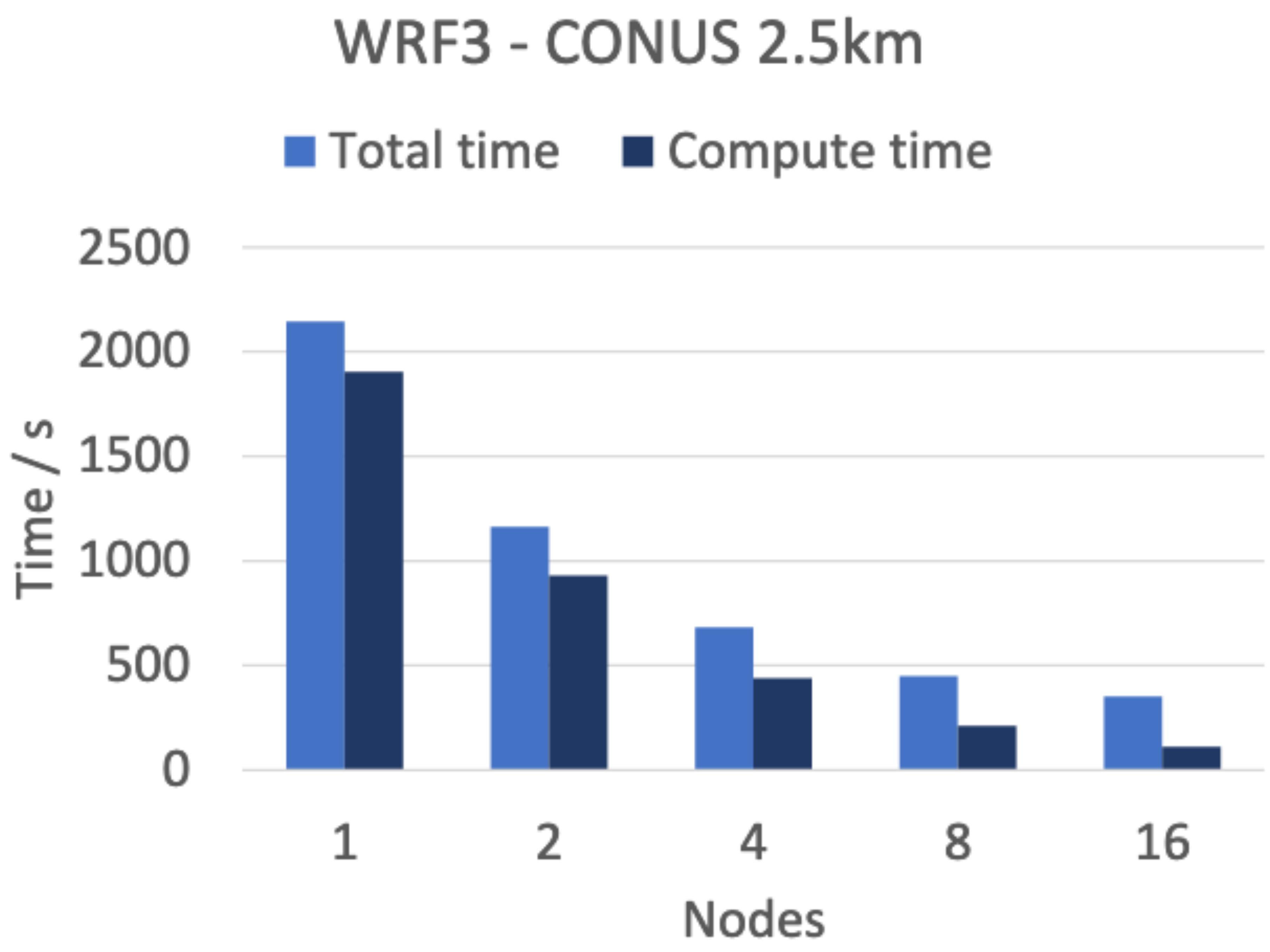
Large-scale GROMACS run via EESSI on JUSUF (JSC)

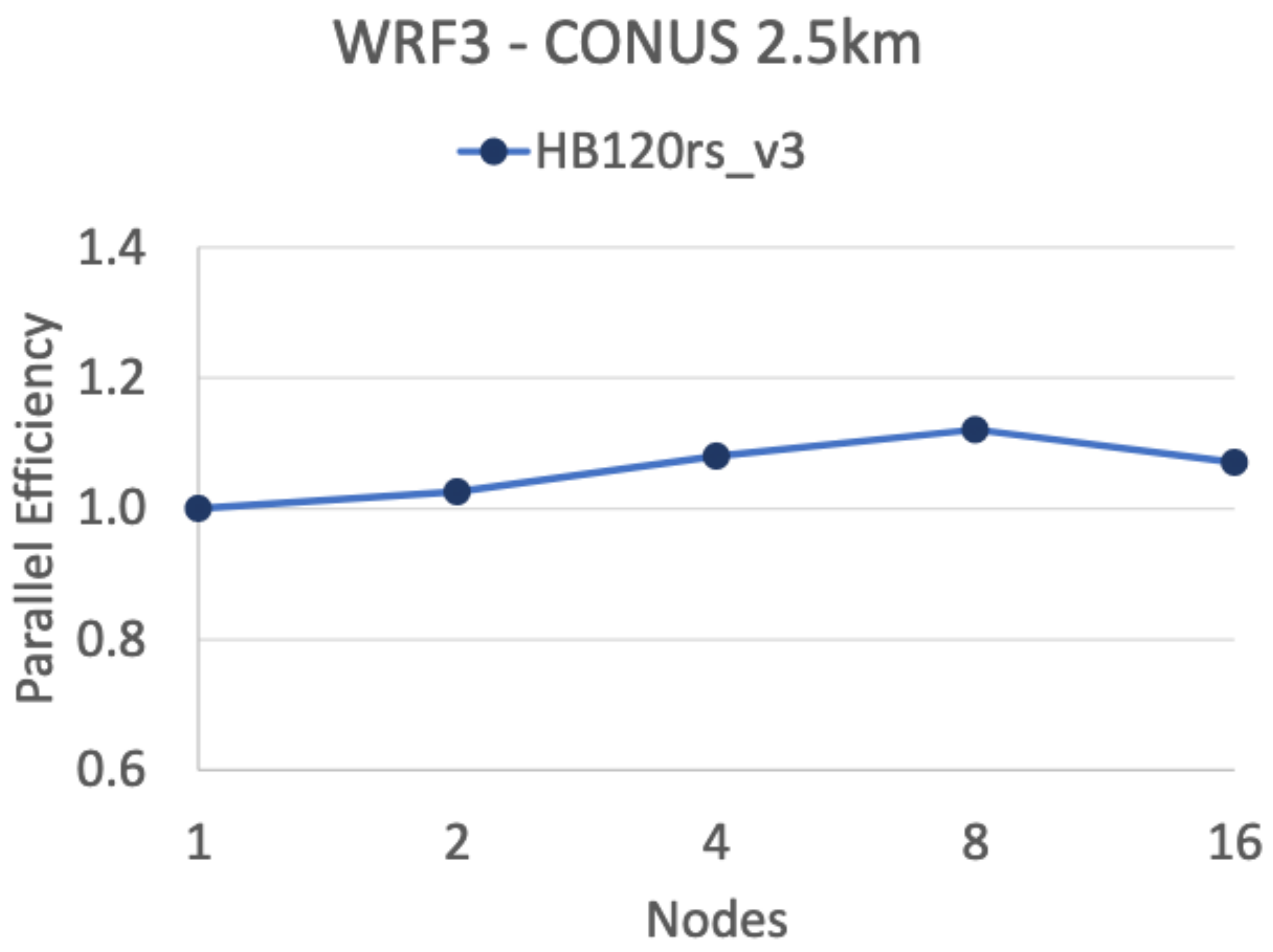
doi.org/10.1002/spe.3075



- 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)

Running WRF in Azure via EESSI





- Performance experiment by Hugo Meiland & Davide Vanzo (Microsoft Azure), for their EUM'22 talk "Leveraging EESSI for WRF simulations at scale on Azure HPC"
- WRF v3.9.1.1 with `foss/2020a` on `HB120rs_v3` (120 cores, AMD EPYC 7V13, HDR Infiniband)
- Essentially using EESSI pilot repository as is, no changes needed to leverage fast interconnect!

EESSI Community Meeting

- After dozens of online meetings, finally a physical (hybrid) community meeting! \o/
- **Wed-Fri 14-16 2022 in Amsterdam**
- Venue: meeting room at restaurant Polder, across the street from Amsterdam Science Park
- Agenda: mix of presentations (introductory & beyond) + hackathon/discussion sessions
- Live streaming of all presentations via the (brand new) EESSI YouTube channel
- All info via https://eessi.github.io/docs/meetings/2022-09-amsterdam
- **If you would like to attend one or multiple days, let me know!**

EESSI

EUROPEAN ENVIRONMENT FOR SCIENTIFIC SOFTWARE INSTALLATIONS

eessi-hpc.org

eessi.github.io/docs

github.com/EESSI

Join our mailing list & Slack channel

eessi-hpc.org/join

Twitter: @eessi_hpc

Status page: status.eessi-infra.org

Paper: doi.org/10.1002/spe.3075

EESSI-related talks at EUM'22:

Getting started - WRF in Azure - Adding software

Monthly online meetings, open to anyone interested

(first Thursday, 14:00 CE(S)T)