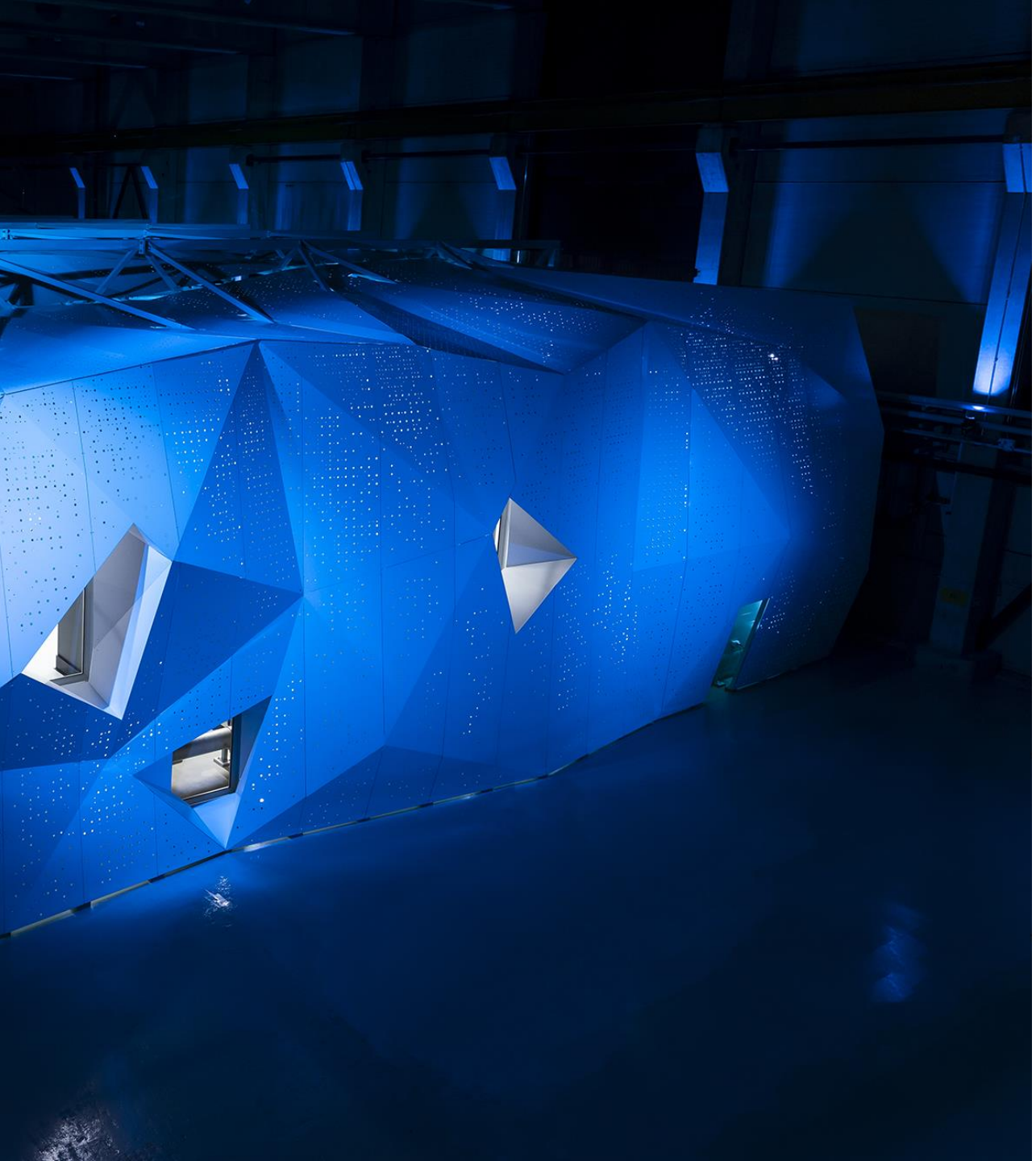


LUMI

The Supercomputer of the North

Orian Louant – Uliège
LUMI User Support Team & Logisticien CECI

orian.louant@uliege.be

CECI User Meeting – Apr. 21th, 2023

Who am I

```
$ pinky -l olouant
Login name: olouant                     In real life:  Orian Louant
(orian.louant@uliege.be)
Directory: /home/ulg/ace/olouant        Shell:  /bin/bash
```

- My background is in computational chemistry (UNamur)
- *“Logisticien CECI”* since 2020
- Based in the University of Liège
- Since 2021, I’m a member of the LUMI User Support Team (part time)

LUMI

LUMI (Large Unified Modern Infrastructure) is a petascale supercomputer

One of the fastest supercomputers in the world: currently **#3 in the Top500** and **#1 in Europe**

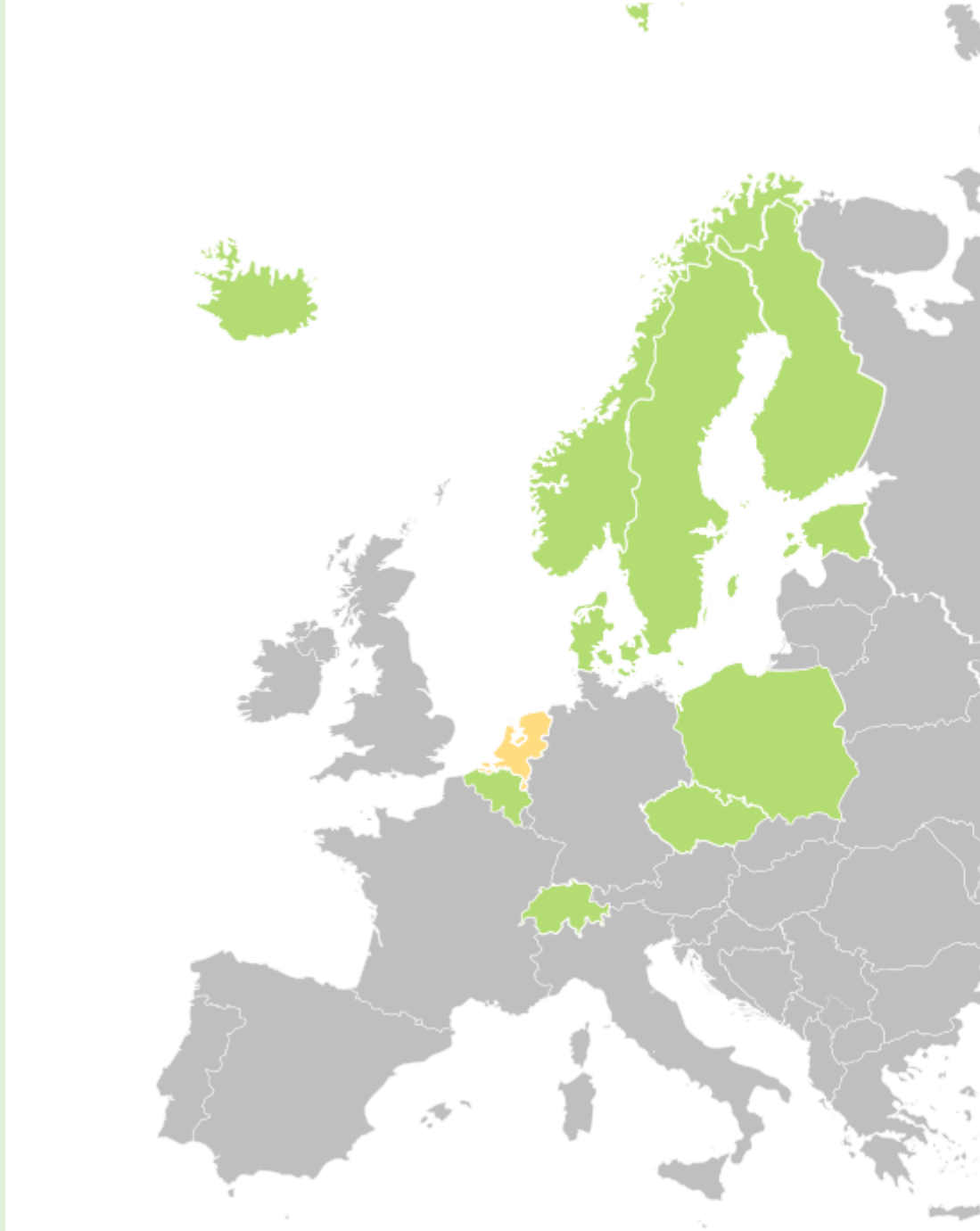
309 Pflops HPL linpack, to be improved in Phase 3 (375 PFlops)

Today's agenda

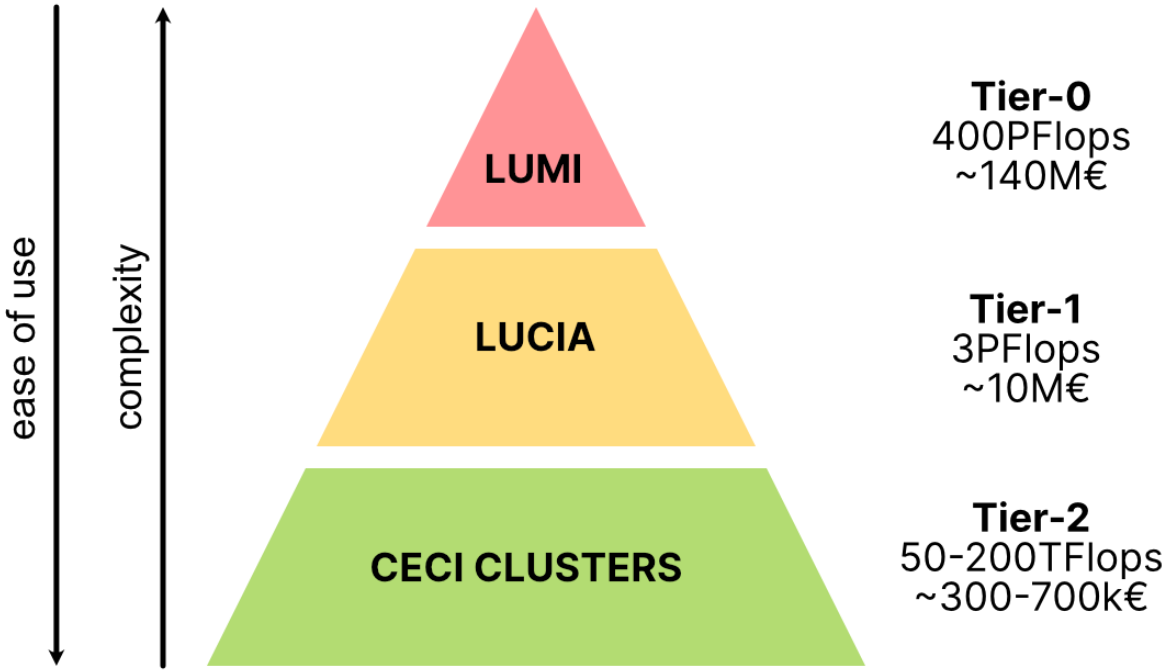
- Hardware overview
- Programming environment and software
- Getting access
- Getting help

The LUMI Consortium

- LUMI was acquired by the **EuroHPC Joint Undertaking** (50%) and **a consortium of 10 countries** (50%)
- Soon to be 11 countries with **The Netherland** joining the consortium
- **~200M€ total budget** with compute time allocated in proportion of the investment of each stakeholder
- Installed in **Kajaani**, Finland
- **Belgium** is the second-largest contributor in the consortium with a **7.651% share**



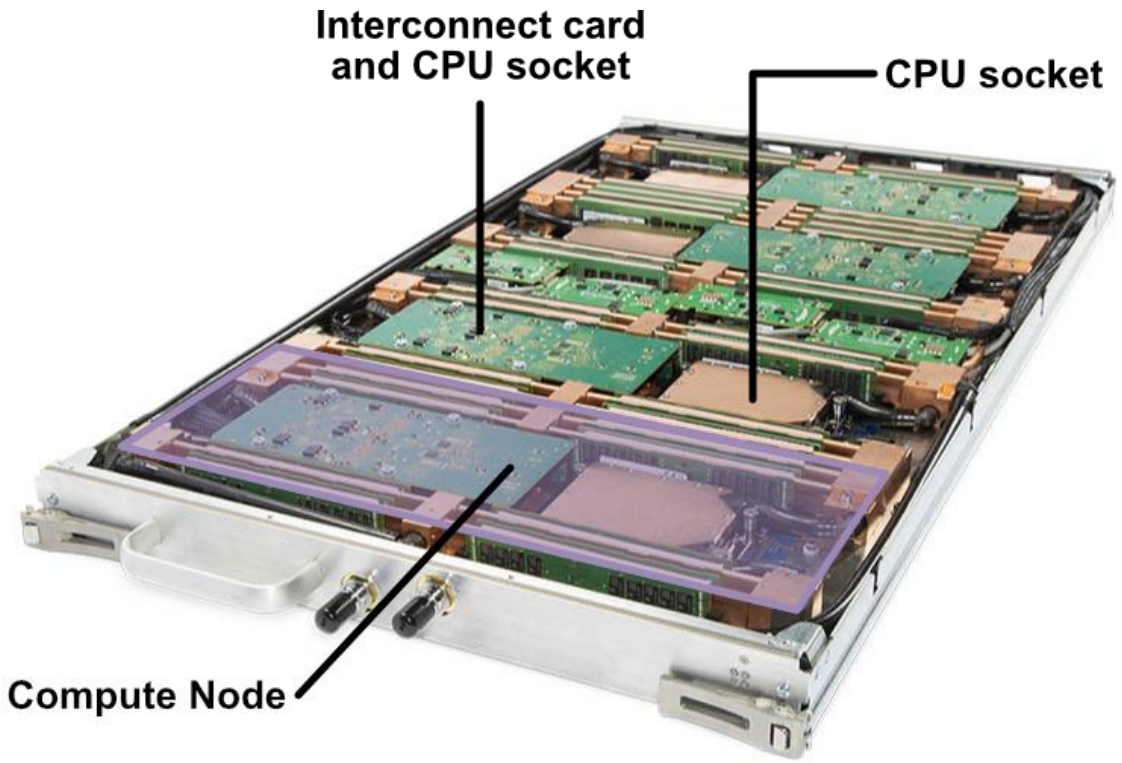
Get access to LUMI



LUMI uses **exa-scale technology**: it's the little brother of Frontier, currently #1 in the top 500

- Its role is to enable ground-breaking science and help European researcher **prepare for exa-scale**
- Suited **for large-scale calculations** with high limits on the size of a job:
  - 512 nodes for LUMI-C
  - 1024 nodes for LUMI-G
  - Full machine "hero" runs once a month
- Requires prior HPC experience to use it at its full potential
- Made up of millions of parts and the chance one of these components will fail is high

LUMI-C: the CPU partition

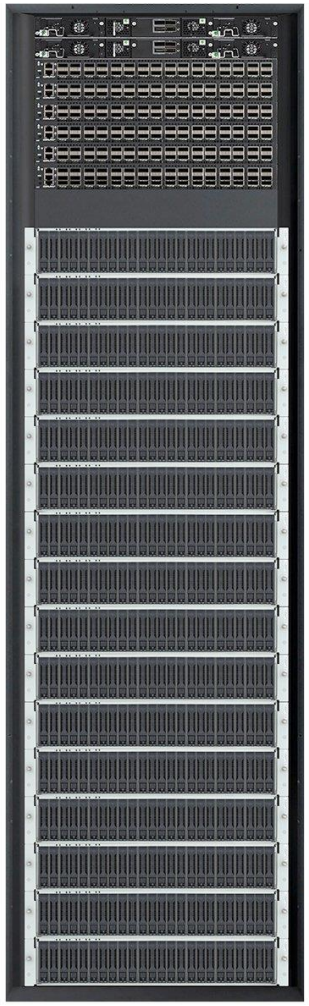


- 1536 nodes with **2×64 cores AMD “Milan” CPUs** (128 cores/nodes).
  - 1376 nodes with 256GB of memory
  - 128 nodes with 512GB of memory
  - 32 nodes with 1TB of memory
- One **Slingshot-11 network interconnect** (200Gb/s) per node
- No disks on the compute nodes

LUMI-G: the GPU partition

- **2560 nodes** with one **64 cores AMD “Trento” CPU** and **4 AMD MI250x GPU Multichip modules**.
- The MI250x modules have 128GB of HBM2 memory at 3.2TB/s. Each module has **2 Graphics Compute dies (GCDs)**, each with 64GB of HBM2 memory at 1.6 TB/s.
- Each GCD has **one infinity fabric link** (8 links total) to one of the CCD (8 cores + L3 cache) on the CPU.
- GCDs within a module are connected to each other with a **quad infinity fabric links**. GCDs between modules are linked to each other with **single or double infinity fabric links**.
- Each MI250x module is connected to a **Slingshot-11 network interconnect** (200Gb/s)

Storage



LUMI has several storage options for a total capacity of 117 PB:

- **LUMI-P:** 4 × 20 PB Lustre filesystem
- **LUMI-F:** 7 PB of flash-based Lustre filesystem
- **LUMI-O:** 30 PB object-storage

User home directory: 20 GB, 100k files

Every project gets allocated a project directory as well as scratch on hard disk and flash-based filesystems:

- **Project directory:** 100 GB, 100k files
- **Scratch:** 50 TB, 2M files
- **Flash scratch:** 2 TB, 1M files

Programming Environments

LUMI is an HPE Cray system and is based on the **Cray Programming environments (also available on Lucia)**:

- **PrgEnv-cray:** clang based C/C++ compilers and proprietary Cray Fortran compiler
- **PrgEnv-amd:** AMD ROCm clang/flang based compiler for AMD GPUs
- **PrgEnv-gnu:** gcc and gfortran compilers (CPU only)
- **PrgEnv-aocc:** AMD optimizing C/C++ and Fortran compilers (CPU only)

- Cray MPI is the only MPI implementation supported on LUMI (based on MPICH)
- Full ROCm installation
- The programming environment comes with commonly used libraries:
  - LibSci (BLAS, LAPACK, ScaLAPACK)
  - HDF5 and netCDF
  - FFTW3
- Parallel debugger (gdb4hpc, DDT) and profiler (Cray perftools, MAP)

Programming models for LUMI-G

- LUMI is an AMD system, **no CUDA** but AMD provides a CUDA like language: **HIP**.
- HIP kernel language is identical to CUDA.
- For runtime calls, porting is usually a matter of replacing “cudaXXX” with “hipXXX”
  - Ex: `cudaMalloc` becomes `hipMalloc`
- HIP code can be compiled on LUMI with the AMD and Cray compilers
- No equivalent to CUDA Fortran but a interface between HIP and Fortran is available: **hipfort**

Directive Based Models

- OpenMP offload available with the Cray and AMD compilers (C/C++ and Fortran)
- OpenACC is available but **only for the Cray Fortran compiler**.

High-Level Frameworks

- Kokkos
- Raja
- Alpaka
- SyCL
- …

Software Policy

- **LUMI is used by a wide variety of users with different needs:** impossible to maintain and install all software centrally as it is impossible to test and maintain a complex software environment for a every field of science using LUMI. Only a very limited set of libraries/software are installed by default.

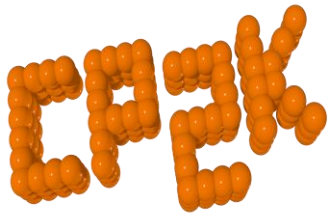
- **You must build your software stack yourself:** we have set up EasyBuild to make that process as easy as possible.

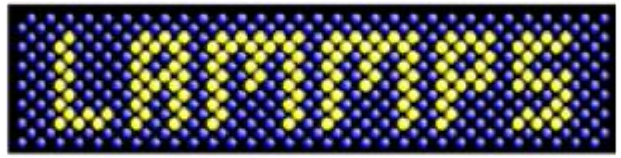
```
$ module load LUMI/22.08
$ module load partition/C
$ module load EasyBuild-user
$ eb -S NAMD
CFGS1=/appl/lumi/LUMI-EasyBuild-contrib/easybuild/easyconfigs
 * $CFGS1/n/NAMD/NAMD-2.14-cpeGNU-22.08-MPI.eb

$ eb --robot --parallel=16 NAMD-2.14-cpeGNU-22.08-MPI.eb
<easybuild output>
$ module load NAMD/2.14-cpeGNU-22.08-MPI
```

GPU software on LUMI

- LUMI is an AMD system: CUDA applications will not run out of the box!
- The following software packages are known to be able to run on the LUMI-G partition:
  - CP2K
  - GROMACS (SyCL and AMD's HIP port)
  - LAMMPS
  - NAMD
  - Amber
  - PyFR
  - nekRS
- As the porting effort progress we hope to see more CUDA code become available for LUMI





LUMI for AI

- PyTorch and TensorFlow have been ported to use AMD GPUs:
  - Quite easy to install using Python wheels
  - Recommended to use containers
  - Most PyTorch extensions can be compiled for AMD GPUs with minimal or no porting effort thanks to the built-in “hipification” tool
- Horovod can be used for distributed training
- JAX available as Python wheel or container

How to get access to LUMI

LUMI is accessible for free to CECI users via the call for compute time organized under the umbrella of the Belgian NCC. Multiple call for compute time are open each year (3-4 call/year)

https://www.enccb.be/GettingAccess

Preparatory/development access: to test the scalability of a code/workflow or perform development to port a code or workflow.
Maximum of 500 CPU kH or 25 GPU kH.

Regular access: for large production run.
Maximum of 10 CPU MH or 500 GPU kH.

Long/medium term goal of an allocation from the Belgian national share of LUMI should be to prepare for a subsequent proposal via the EuroHPC JU track

Getting Help (LUMI User Support Team)

Support team staffed with 9 FTE (15 members) coming from the consortium countries.

- **For Belgium:**
  - Kurt Lust (Flanders, VSC, UAntwerp)
  - Orian Louant (Wallonia-Brussel Federation, CECI, ULiège)
- **Centralized helpdesk:** the LUMI support team supports all users of LUMI regardless their country of origin

The LUMI user support team is responsible for the Level-2 support:

- User support: https://lumi-supercomputer.eu/user-support/need-help/
- User documentation: https://docs.lumi-supercomputer.eu/
- User training: https://lumi-supercomputer.eu/events/
- Software build recipes: https://lumi-supercomputer.github.io/LUMI-EasyBuild-docs/
- Porting program and Hackathon

Getting Help

- **Before you apply:** contact CECI support if you need help writing your application for compute time
- **Level-1 support:** issues regarding your allocation and user account
    - Each consortium country manage the allocation process for its share. For Belgium, this is done by the VSC (for Flanders) and the **CECI (for Wallonia-Brussel Federation)**
    - Once your proposal is accepted, allocation will be created by the CECI resource allocator (David Colignon).
        - If you need help regarding your allocation, connecting to LUMI or add member to your project, email David with myself in CC
- **Level-3 support:** optimization, profiling and debugging of code or porting a CPU code to GPU
    - LUMI User Support Team: Hackathons and application readiness program (need to be selected)
    - **Feel free to contact me as well!**

EuroHPC JU systems

Pre-exascale systems

LUMI
(Finland)

Mare Nostrum 5
(Spain)

Leonardo
(Italy)

Peta-scale systems

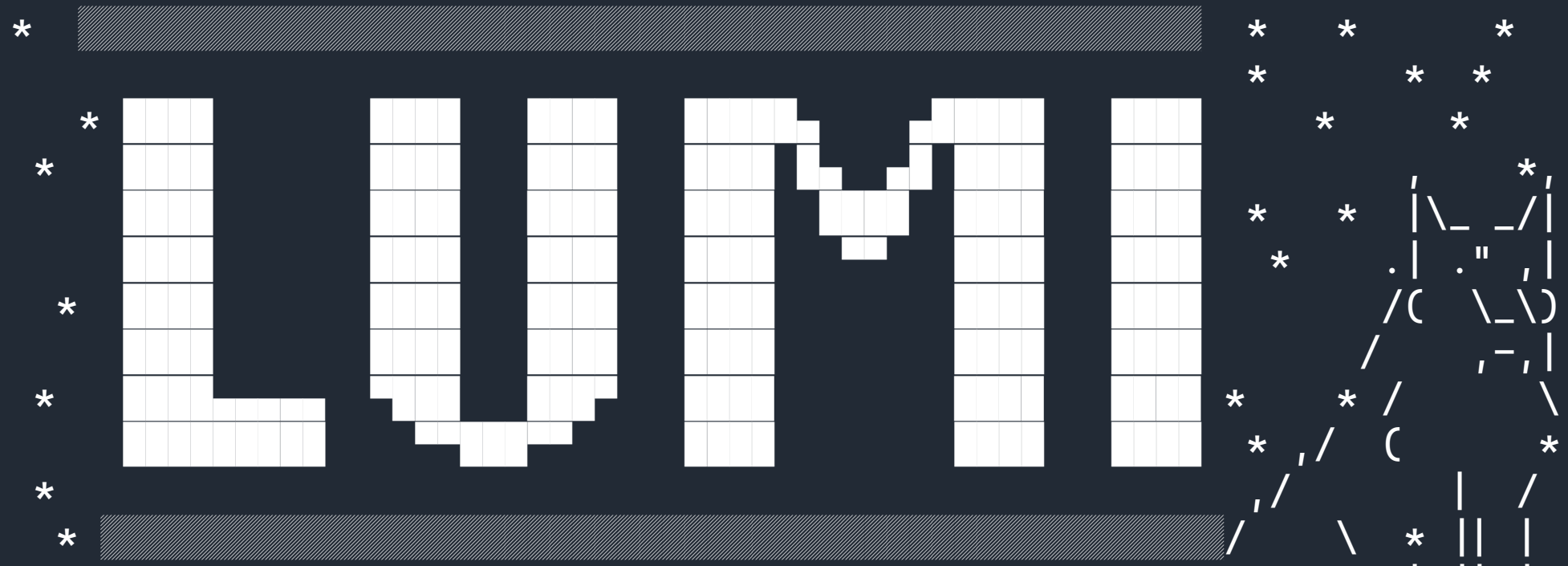
Karolina
(Czech Republic)

Meluxina
(Luxembourg)

Vega
(Slovenia)

Discoverer
(Bulgaria)

Deucalion
(Portugal)



User guide and support
https://docs.lumi-supercomputer.eu
https://lumi-supercomputer.eu/user-support

❄ (14:41) uan01 [~] $ echo See you on LUMI!