

# LUMI



## LUMI User Support Team



## LUMI will have a support team

Current **8 FTE** with “application expert” background  
Each country nominates one person.

+ support from HPE Center of Excellence, **4 full-time experts**

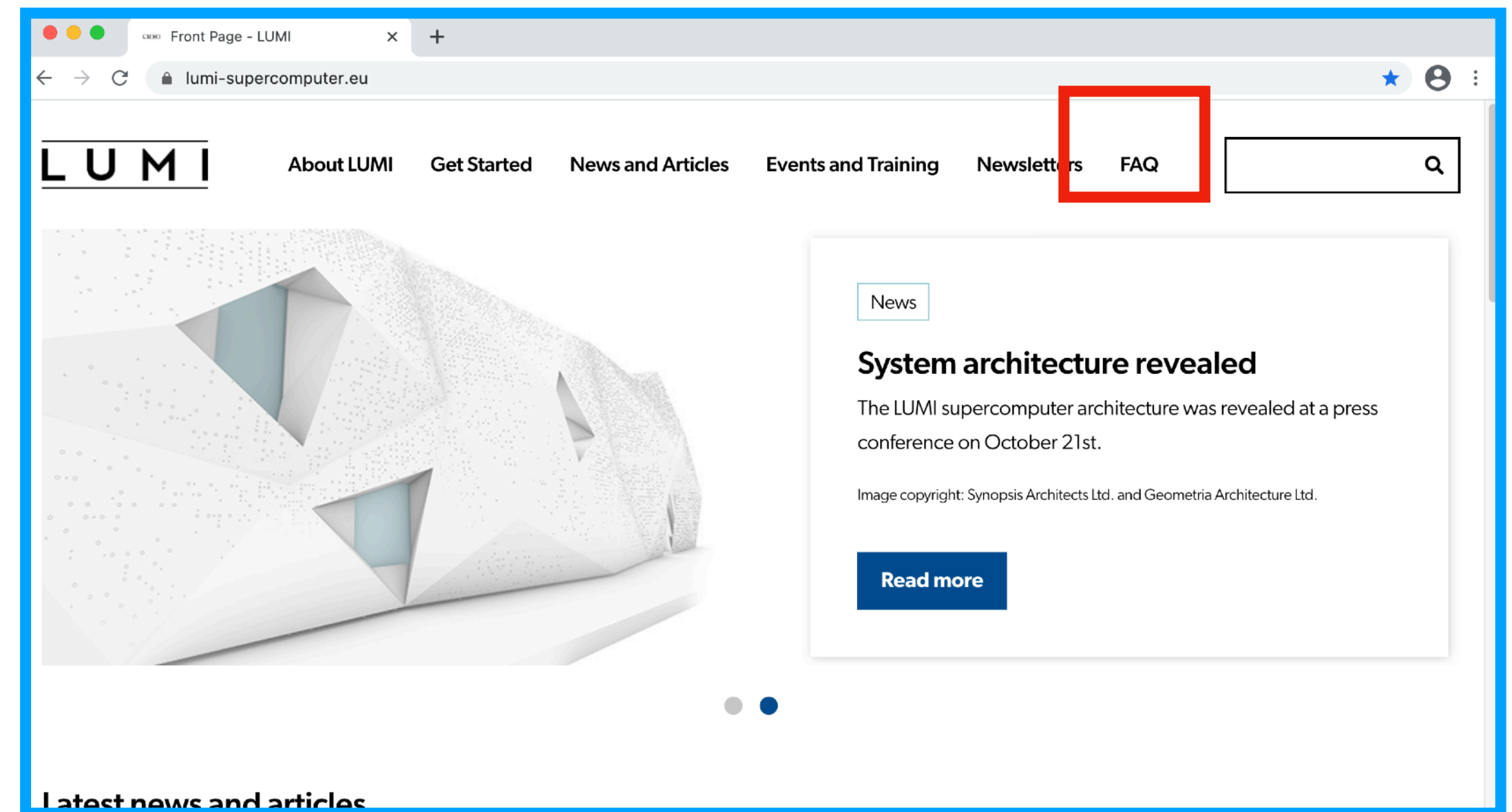
- There will be a support portal on the web page:  
[support@lumi-supercomputer.eu](mailto:support@lumi-supercomputer.eu)\*

\* or similar looking, TBD

- Mondays-Fridays 08.00-18.00 CET
- You can ask for help just like you are used to when running on a SNIC supercomputing center.
- We will also write documentation for the system, and work **proactively** with support.

<https://lumi-supercomputer.eu>

Support will be  
here in the future



## What we can help with

- **Installing software:**
  - Compilers, choosing flags, fixing errors
  - Finding the right library to speed up the computation
  - Smaller code fixes.
- **Troubleshooting:**
  - Why did my program crash?
  - Investigating numerical issues like instability and bad convergence
- **Best practice advice:**
  - How to use scientific software, which algorithms, best convergence parameters etc
- **HPC workflow / logistics:**
  - How to use the batch system
  - Moving and staging data and calculations.

### Please ask for help!

- The LUMI Support might have seen the same problem that you have 10 time before and we can solve it in 5 minutes.
- We are also interested in feedback. Annoyed about something? Tell us.

# How to start preparing

## Migration Path to LUMI

Today	LUMI	Comments
Intel Compilers	Cray PE, GCC, AOCC, clang	LUMI has AMD CPUs. Intel compilers may work, but not optimally. This includes Intel MKL library.
NVIDIA CUDA C/C++	AMD HIP C/C++	HIP comes with tools that can convert CUDA code to HIP, but manual changes ( $\approx 25\%$ of CUDA code) are still needed and new performance tuning.
OpenACC	OpenMP (offload)	Unclear support, maybe only in Cray compilers?
OpenMP (offload)	OpenMP (offload)	No problems! Note that OpenMP 4.5/5.0 not fully implemented in gcc yet.
OpenCL	OpenCL	AMD's ROCm has OpenCL runtime.

If you have not run on a Cray supercomputer before, try to get access to one before.  
In Sweden, we have the *Beskow* system at PDC.