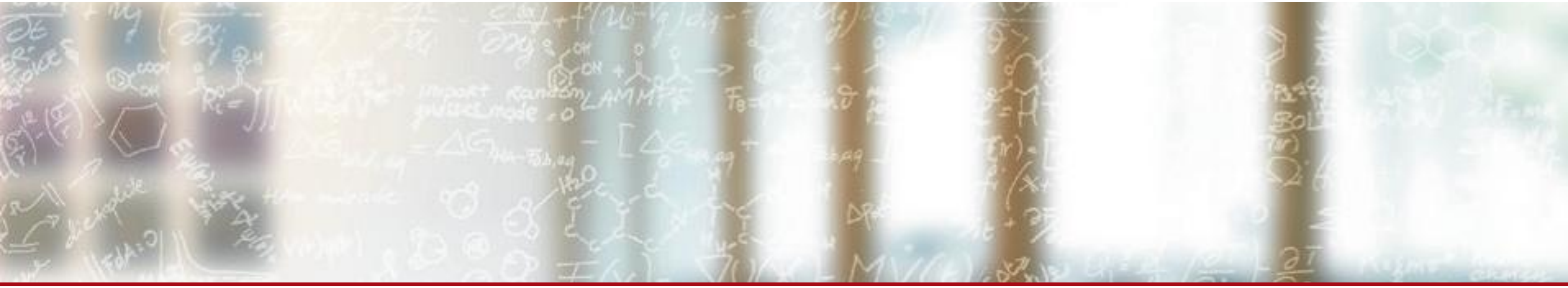
CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich



Leveraging libfabric to compare containerized MPI applications performance over Slingshot 11

CUG 2023

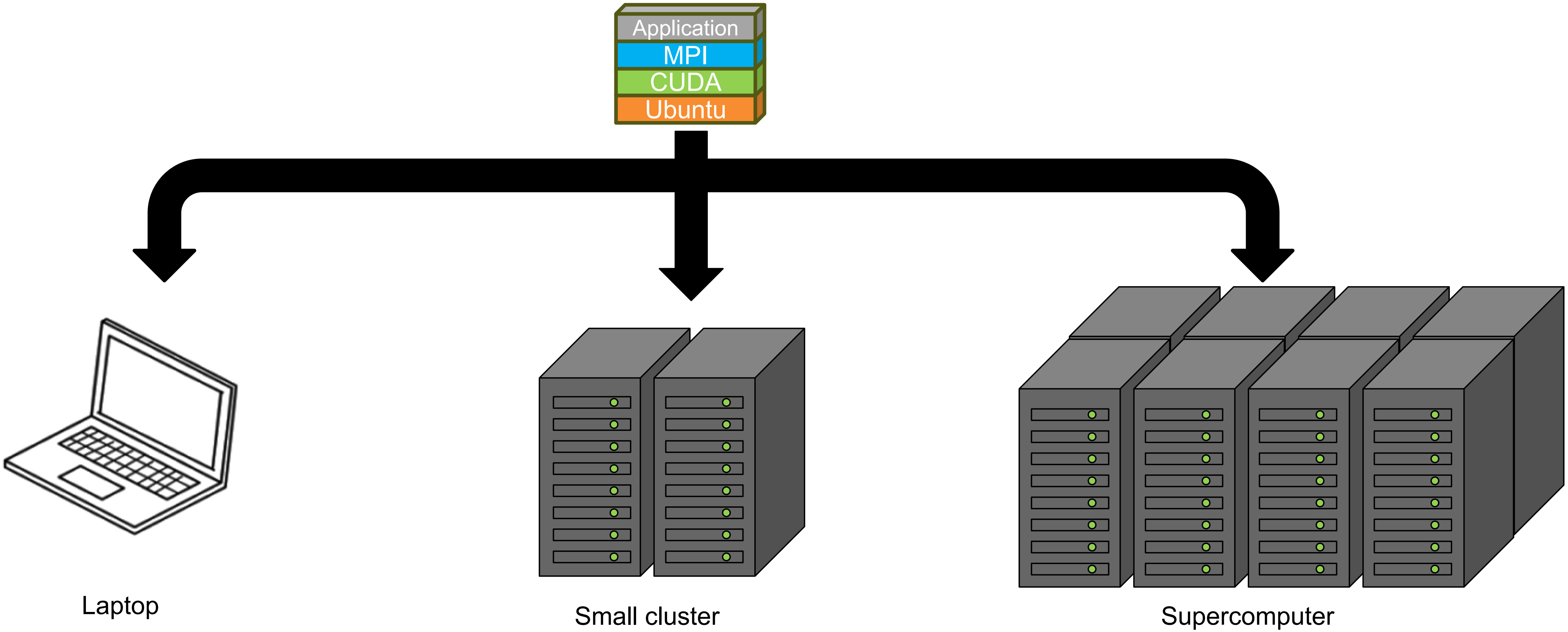
Alberto Madonna, ETH Zurich / CSCS

May 9th, 2023

Table of Contents

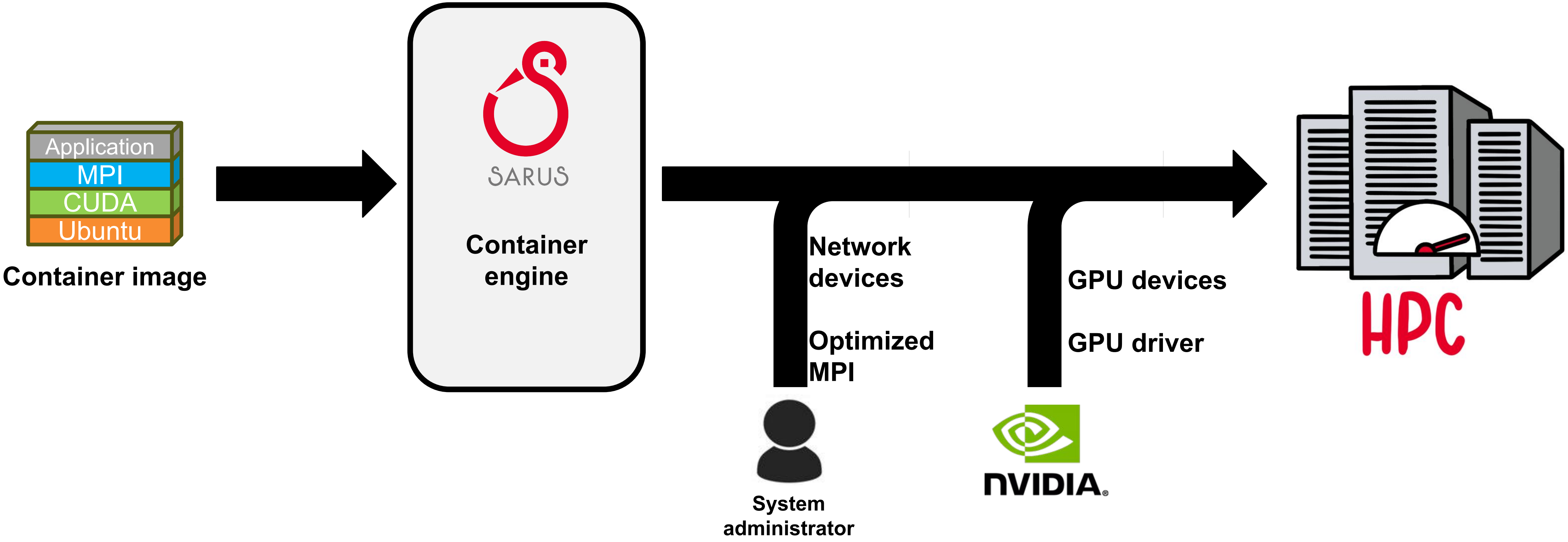
Performance portability for HPC containers

- Take advantage of the portability of images. Don't rebuild images for each system



Performance portability for HPC containers

- Augment images with host resources at container creation time: portable ***images***, HPC ***containers***



Ideal solutions to enable MPI performance portability should…

① Be independent from the MPI implementation

- Allow developers to use the best MPI flavor for their application
- Allow computing providers to accommodate users regardless of their chosen MPI implementation

② Minimize modifications to the container image software stack

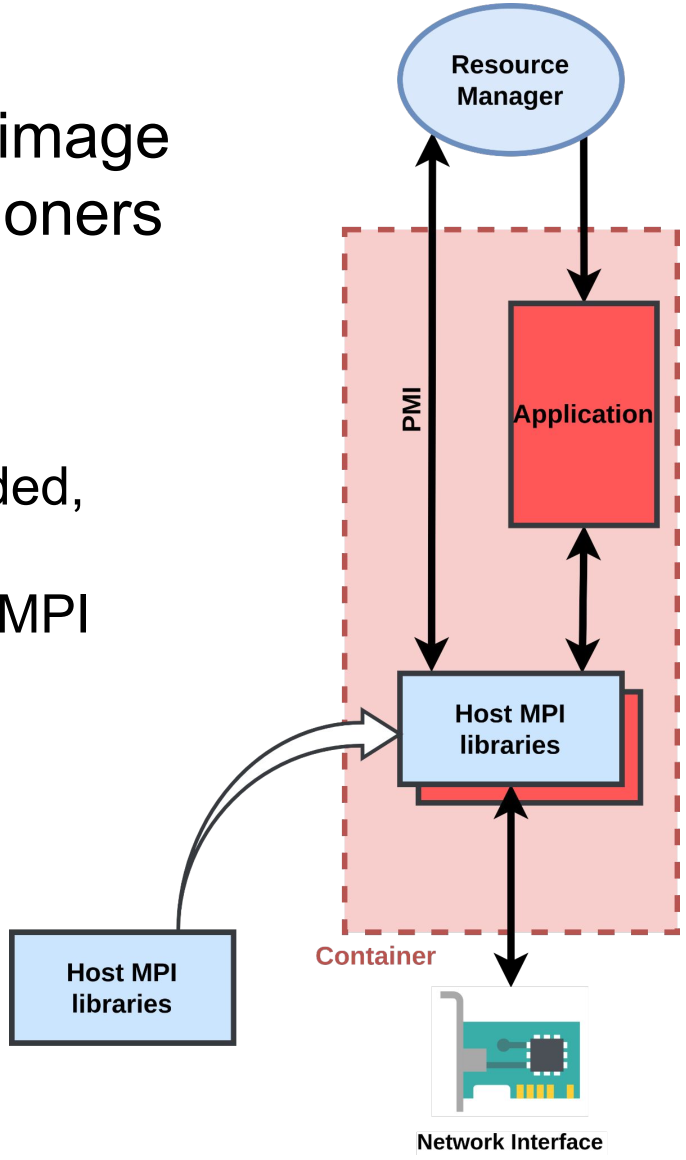
- Improve workflow reproducibility

Established approach: MPI libraries replacement

- Completely replace the MPI libraries of the container image
- Implemented in various forms by HPC tools or practitioners

- Pros:
  - ✓ Transparently matches the host's PMI implementation
  - ✓ Injected host libraries are usually optimized or vendor-provided, allowing to achieve native performance
  - ✓ Seamlessly enables complex features not present in image MPI (e.g. GPUDirect RDMA)

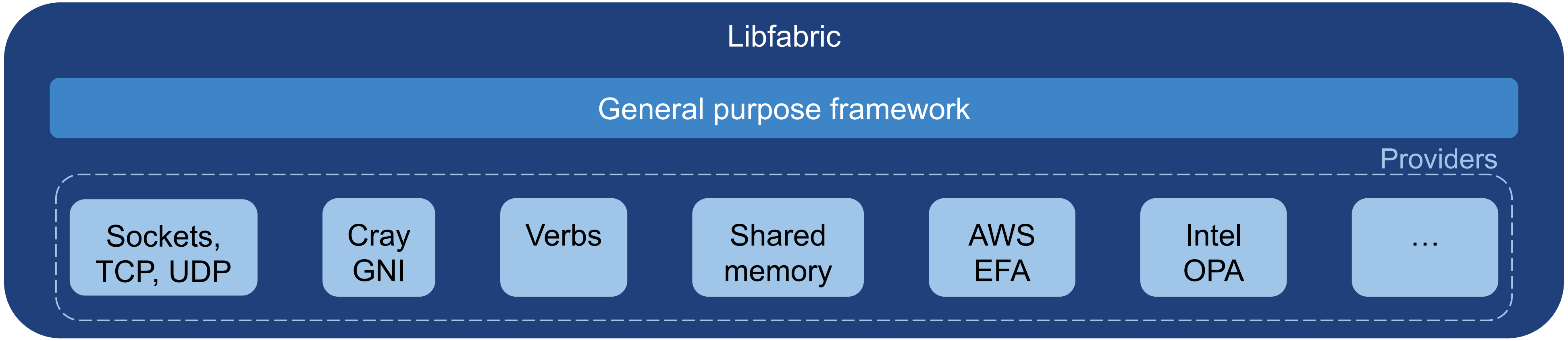
- Cons:
  - ✕ Requires same family of MPI implementation ① (MPICH or OpenMPI)
  - ✕ Requires ABI compatibility
  - ✕ Extensive amount of dependencies to inject ②



Libfabric

"A framework focused on exporting fabric communication services to applications"[1]

- Can act as middleware between MPI libraries and the network hardware
- Provides a unified, high-level interface for callers
- Under the hood uses optimised code paths and dynamic hardware selection for best performance
- Fabric diversity is supported through different ***providers***:



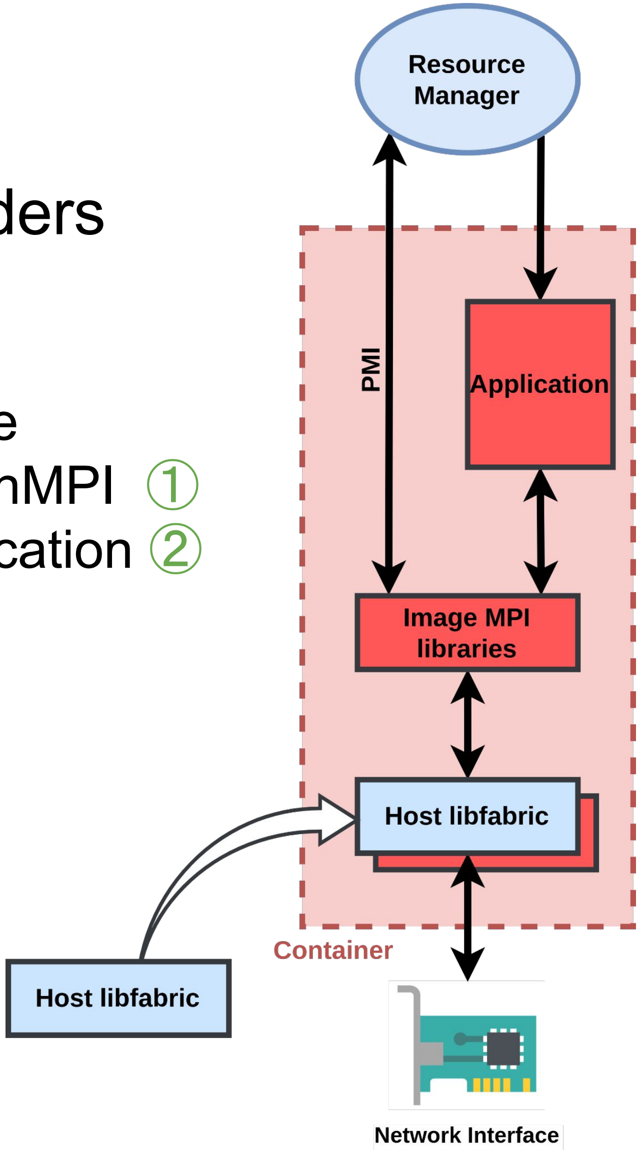
[1] https://ofiwg.github.io/libfabric/

Technique: libfabric replacement

- Replace image libfabric instead of the MPI library
- Host libfabric would feature interconnect-specific providers

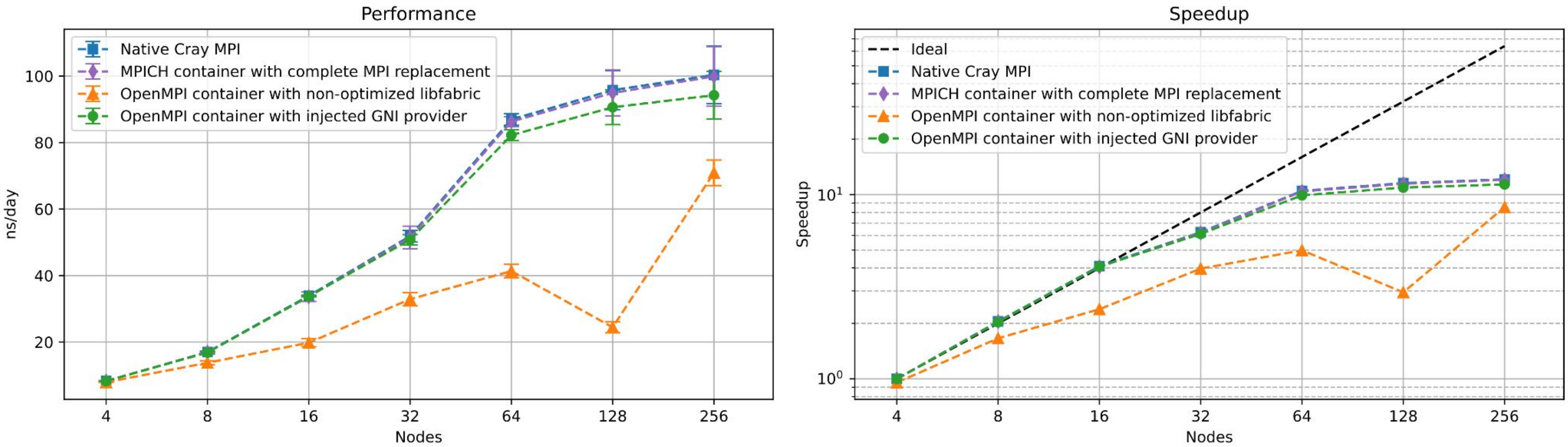
- Pros:
  - ✓ Hardware-matching provider enables near-native performance
  - ✓ MPI implementation agnostic: supports both MPICH and OpenMPI ①
  - ✓ Preserves original image MPI and ABI interface with the application ②
  - ✓ Less dependencies to inject ②

- Cons:
  - ✗ Requires image MPI to be built with libfabric support
  - ✗ Requires libfabric ABI compatibility
  - ✗ Image MPI must support PMI used by the host
  - ✗ Vendor-specific MPI optimizations may not be available



http://doi.org/10.1109/CANOPIE-HPC56864.2022.00010

Libfabric-enabled OpenMPI container on Piz Daint



Software: GROMACS 2021.5, CUDA 11.0, libfabric 1.15.1
Test case: PRACE Unified European Applications Benchmark Suite, GROMACS Test Case B
System: Piz Daint hybrid partition (Intel Xeon E5-2690 v3, NVIDIA Tesla P100, Cray Aries Interconnect)

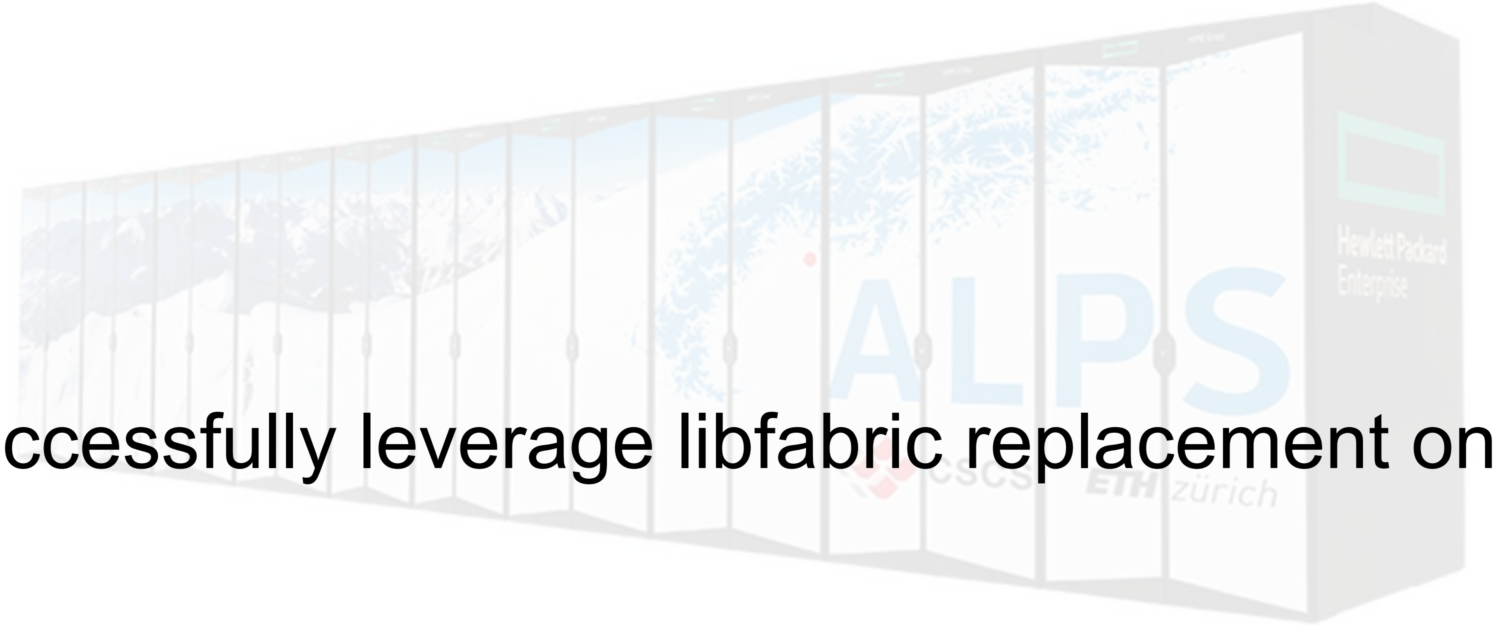
“Libfabric-based Injection Solutions for Portable Containerized MPI Applications”
A. Madonna (ETHZ/CSCS), T. Aliaga (ETHZ/CSCS), CANOPIE-HPC 2022 workshop:
http://doi.org/10.1109/CANOPIE-HPC56864.2022.00010

Alps Research Infrastructure

Includes:

- HPE Cray EX Supercomputer
- HPE Slingshot High-speed Interconnect

Alps Research Infrastructure



Can we successfully leverage libfabric replacement on Slingshot??

Includes:

- HPE Cray EX Supercomputer
- HPE Slingshot High-speed Interconnect

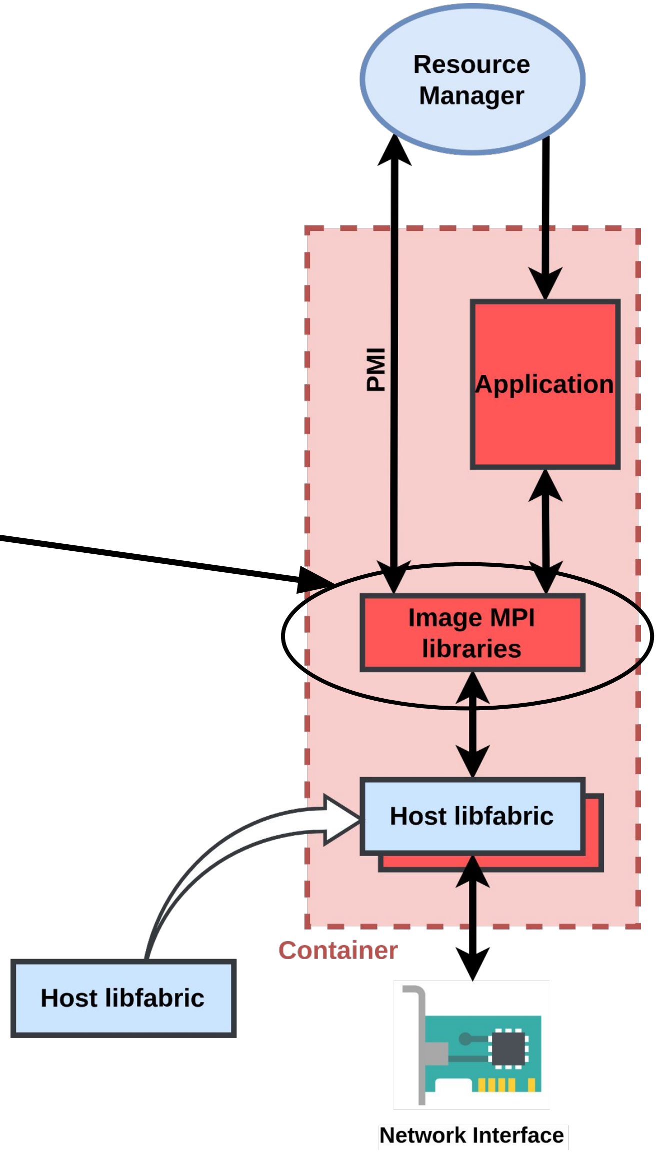
Experimental setup

- Host system: Alps Infrastructure - CPU production partition
  - HPE Cray EX supercomputer @ CSCS
  - Compute nodes: 2 x AMD EPYC 7742 64-core CPU
  - HPE Slingshot 11 interconnect with Dragonfly topology
  - HPE-provided libfabric 1.15.0.0 with “CXI” custom provider for Slingshot 11
  - Native Cray MPICH 8.1.12
  - Container engine: Sarus 1.5.1

- Container images base elements:
  - Ubuntu 22.04
  - Libfabric 1.14.1
  - One of the following MPI implementations:
    - OpenMPI 4.1.4
    - MPICH 4.1
    - MVAPICH2 3.0a (only for synthetic benchmarks)

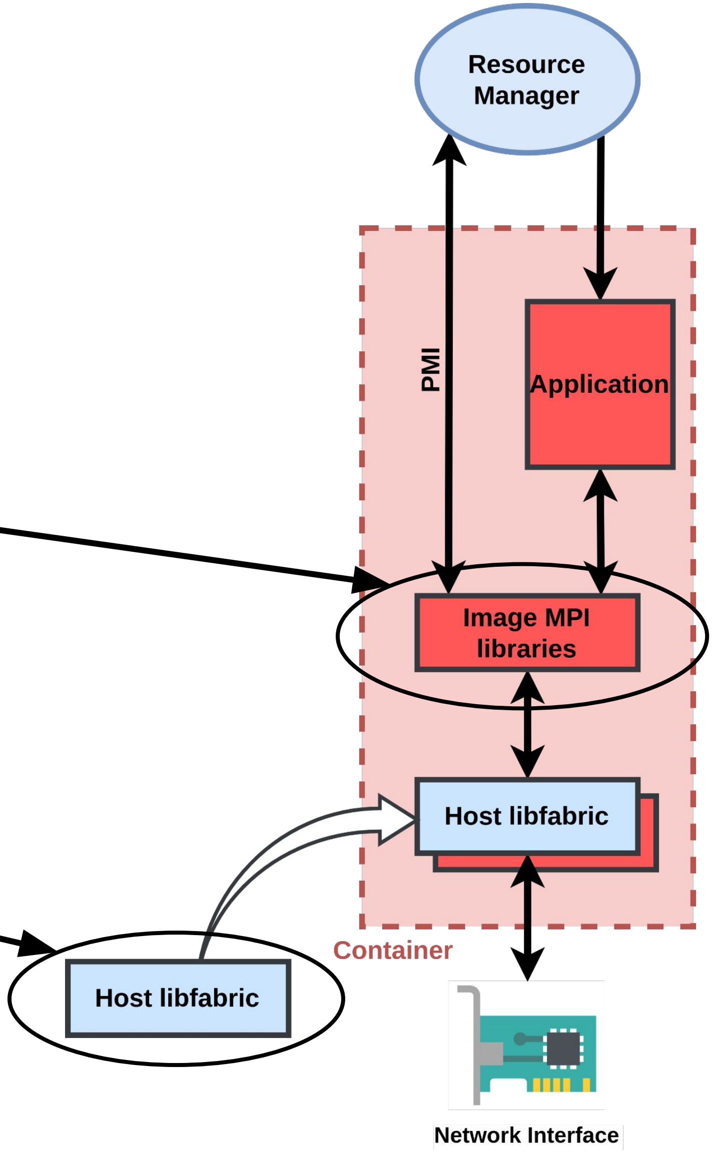
Libfabric replacement on Slingshot 11

- Image MPI libraries
  - Built with the image software stack/compiler
  - In these experiments, one of:
    - OpenMPI 4.1.4
    - MPICH 4.1
    - MVAPICH2 3.0a



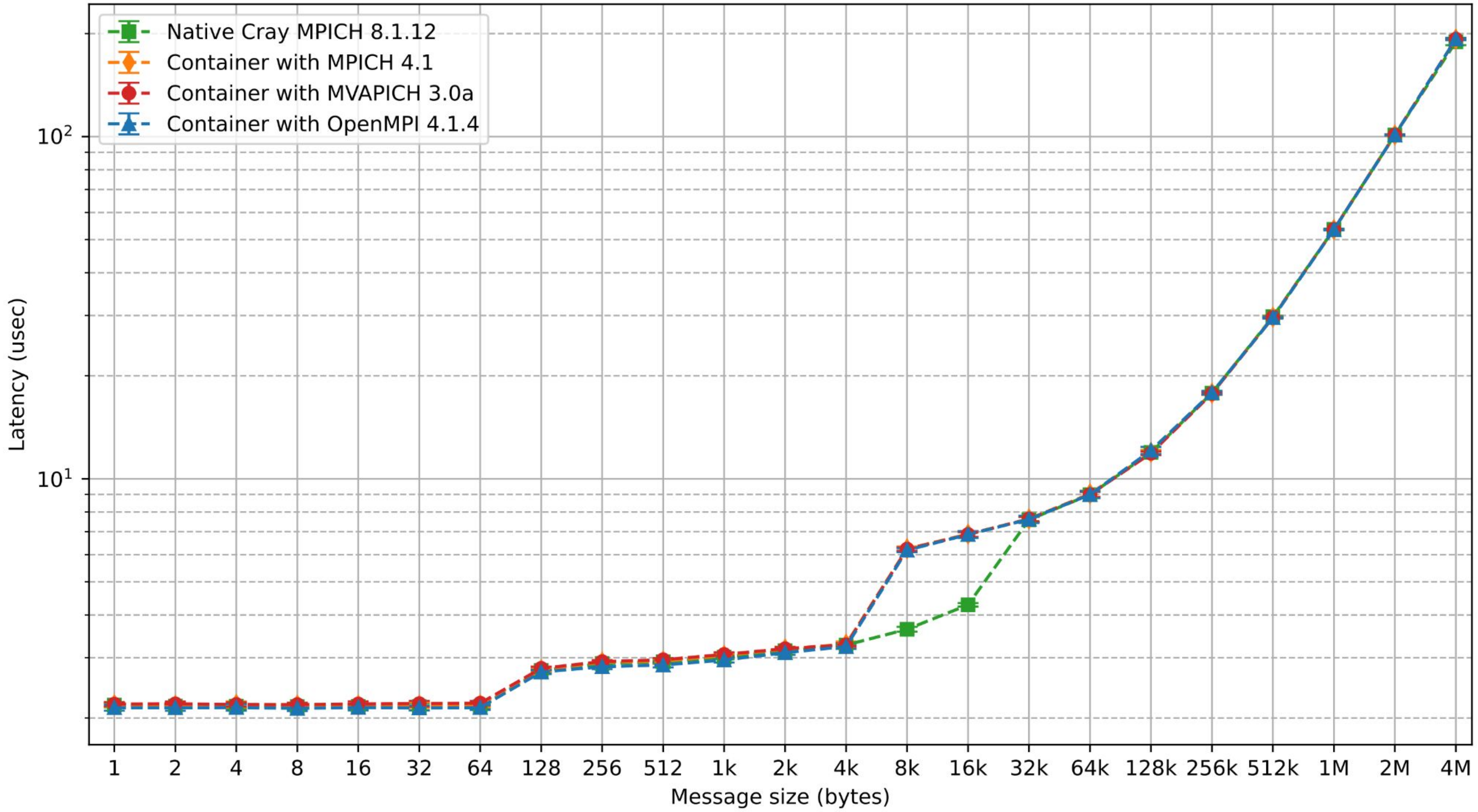
Libfabric replacement on Slingshot 11

- Image MPI libraries
  - Built with the image software stack/compiler
  - In these experiments, one of:
    - OpenMPI 4.1.4
    - MPICH 4.1
    - MVAPICH2 3.0a
- HPE-provided libfabric with CXI custom provider for Slingshot 11



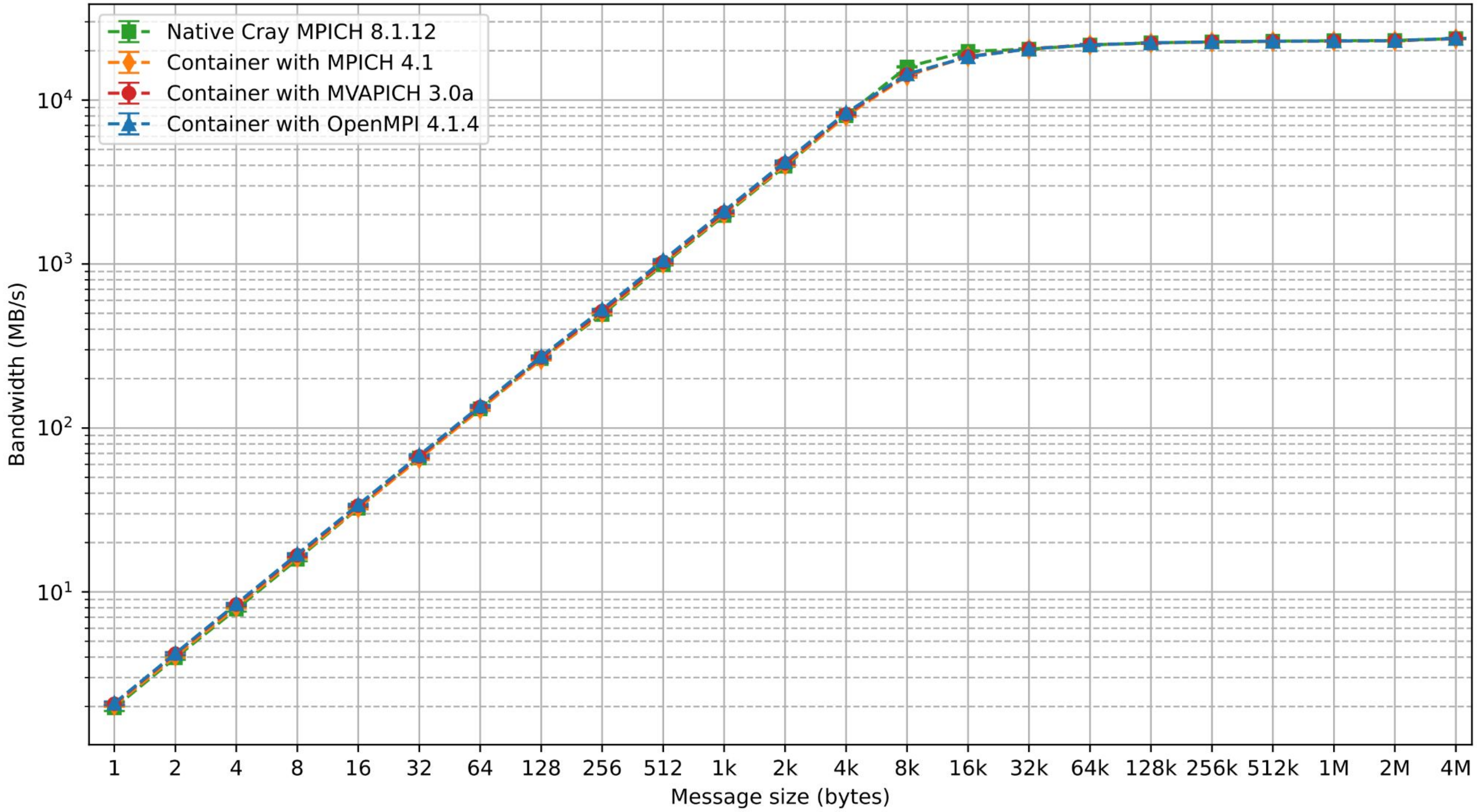
Point-to-point synthetic benchmarks

OSU Latency



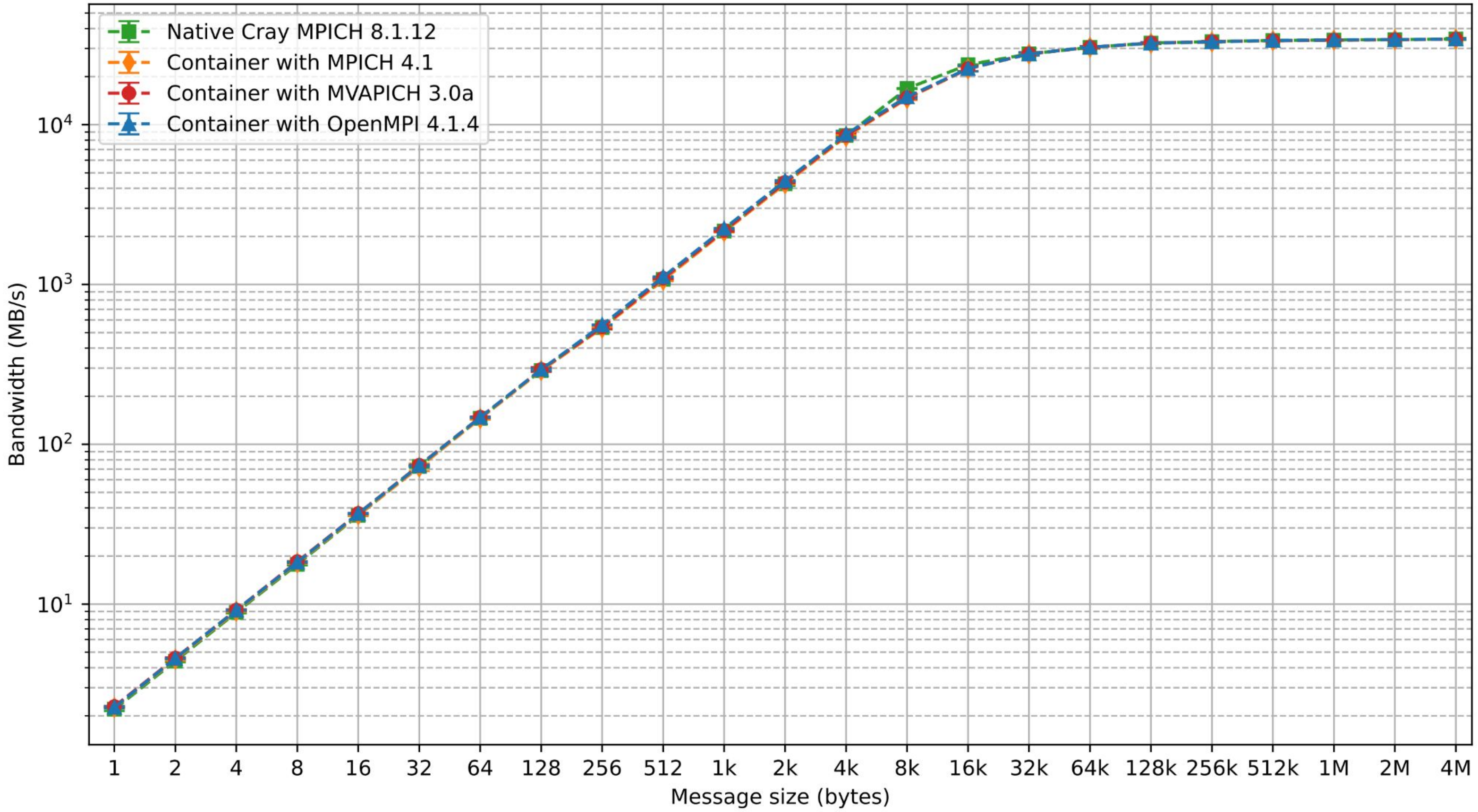
Software: OSU Micro-Benchmarks 6.2, HPE libfabric 1.15.0.0
Test case: `osu_latency` benchmark (2 physical nodes, 30 repetitions)
System: Alps Infrastructure - CPU partition (2 x AMD EPYC 7742, HPE Slingshot 11 Interconnect)

OSU Bandwidth



Software: OSU Micro-Benchmarks 6.2, HPE libfabric 1.15.0.0
Test case: `osu_bw` benchmark (2 physical nodes, 30 repetitions)
System: Alps Infrastructure - CPU partition (2 x AMD EPYC 7742, HPE Slingshot 11 Interconnect)

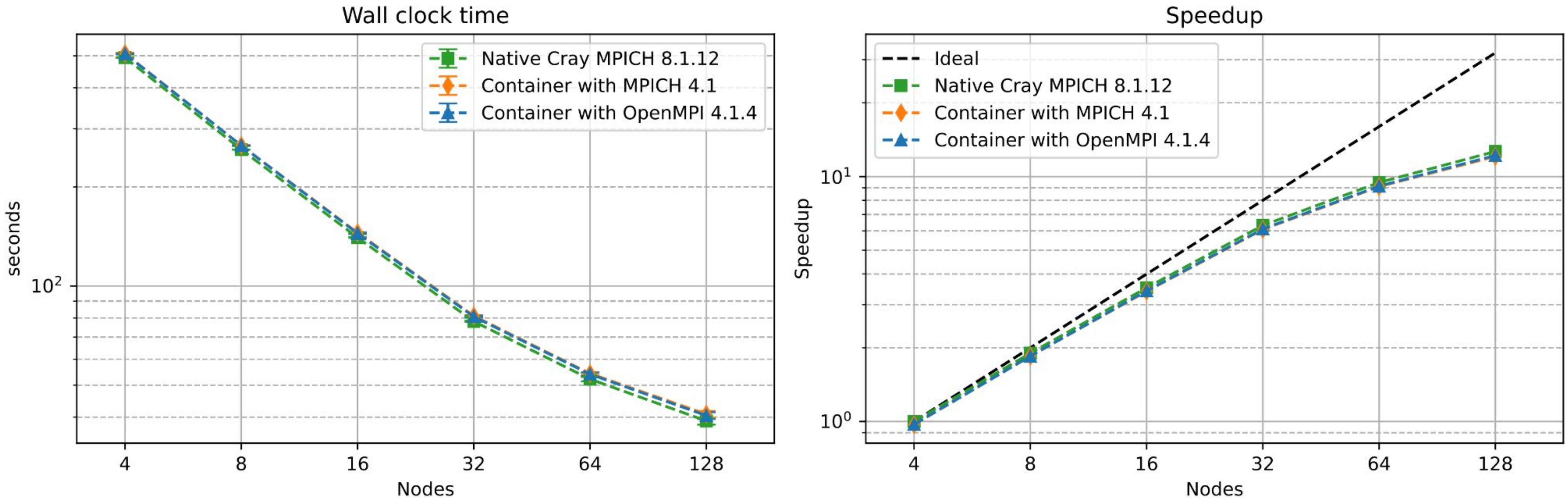
OSU Bi-directional Bandwidth



Software: OSU Micro-Benchmarks 6.2, HPE libfabric 1.15.0.0
Test case: `osu_bibw` benchmark (2 physical nodes, 30 repetitions)
System: Alps Infrastructure - CPU partition (2 x AMD EPYC 7742, HPE Slingshot 11 Interconnect)

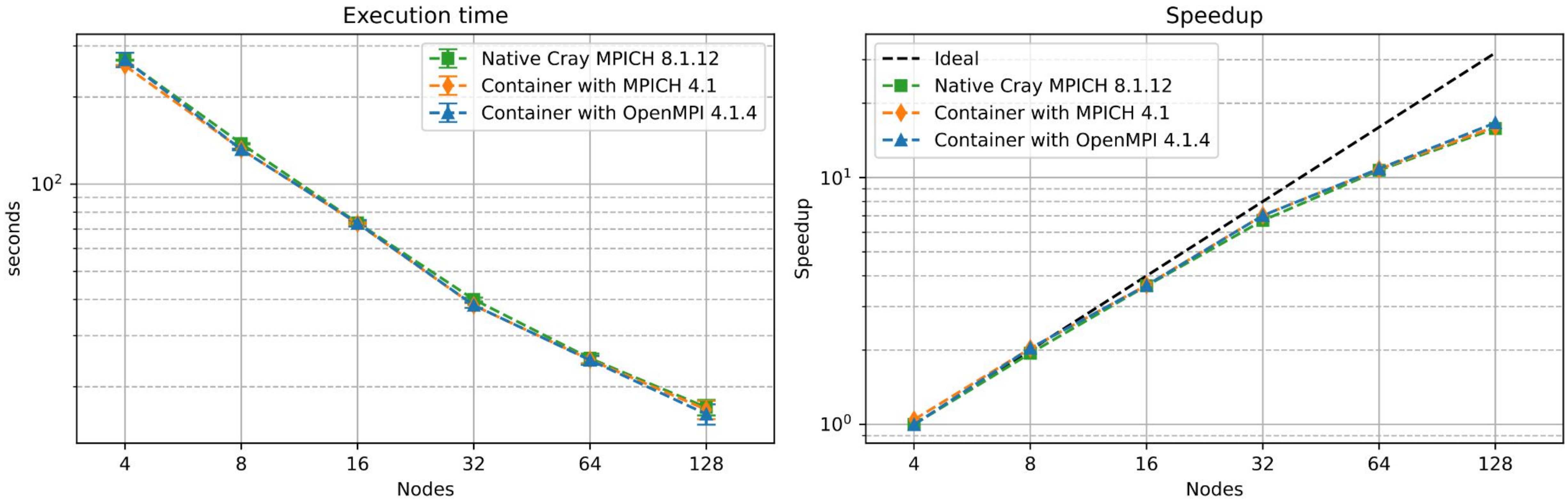
Real-world scientific applications

GROMACS (Classical Molecular Dynamics)



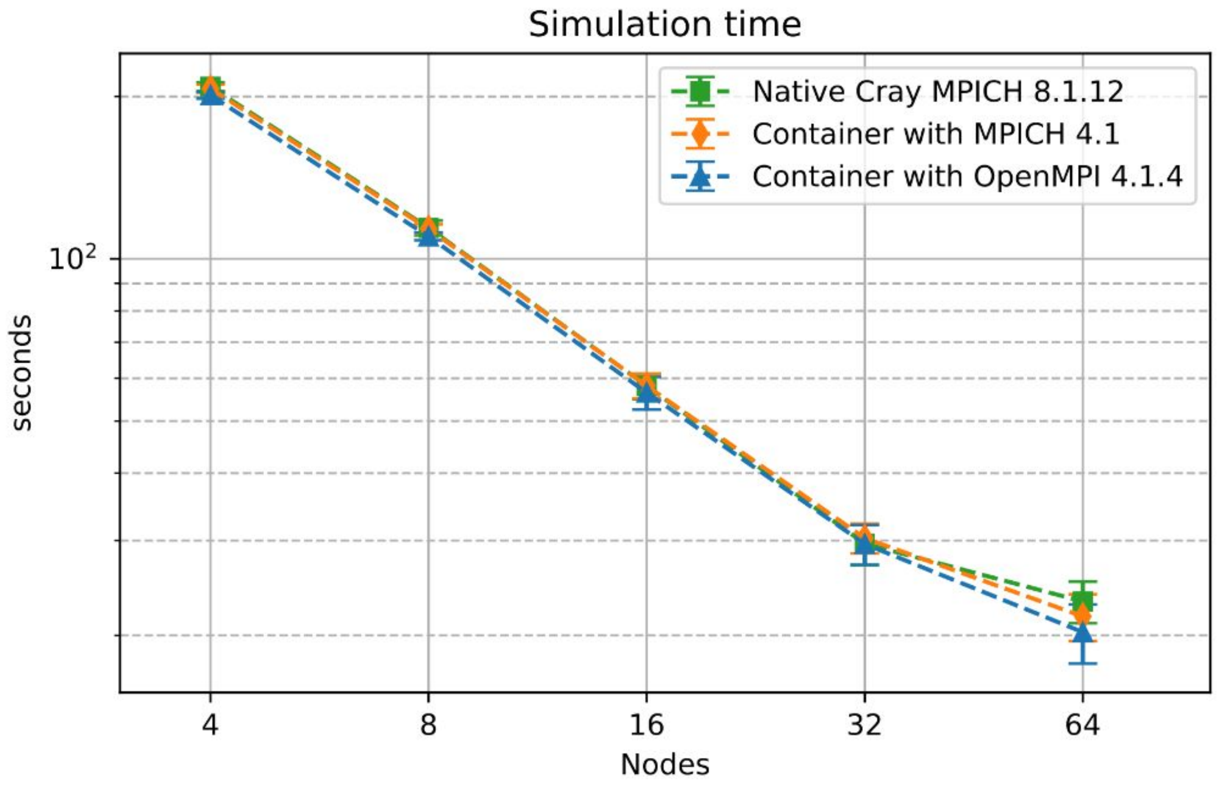
Software: GROMACS 2021.5, HPE libfabric 1.15.0.0
Test case: PRACE Unified European Applications Benchmark Suite, GROMACS Test Case B (16 ranks per node, 30 repetitions)
System: Alps Infrastructure - CPU partition (2 x AMD EPYC 7742, HPE Slingshot 11 Interconnect)

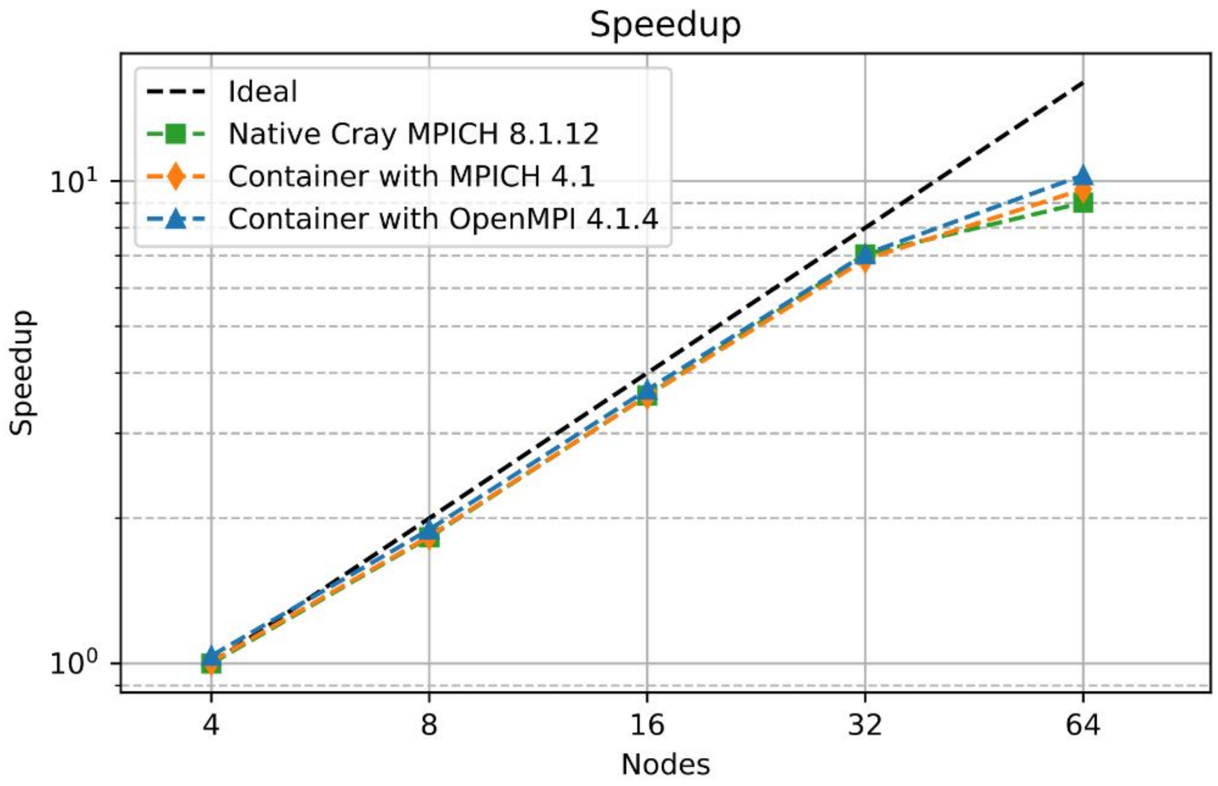
SPH-EXA (Smoothed Particle Hydrodynamics)



Software: SPH-EXA v0.7, HPE libfabric 1.15.0.0
Test case: Sedov spherical blast wave (2 ranks per node, 30 repetitions)
System: Alps Infrastructure - CPU partition (2 x AMD EPYC 7742, HPE Slingshot 11 Interconnect)

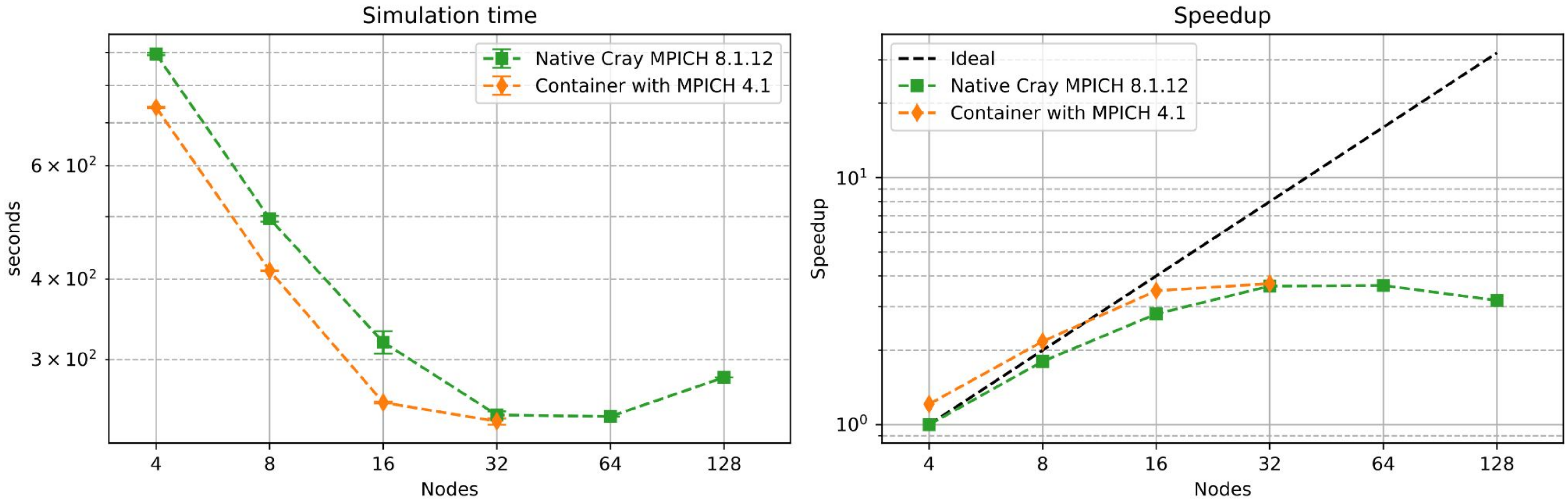
PyFR (Flux Reconstruction CFD)





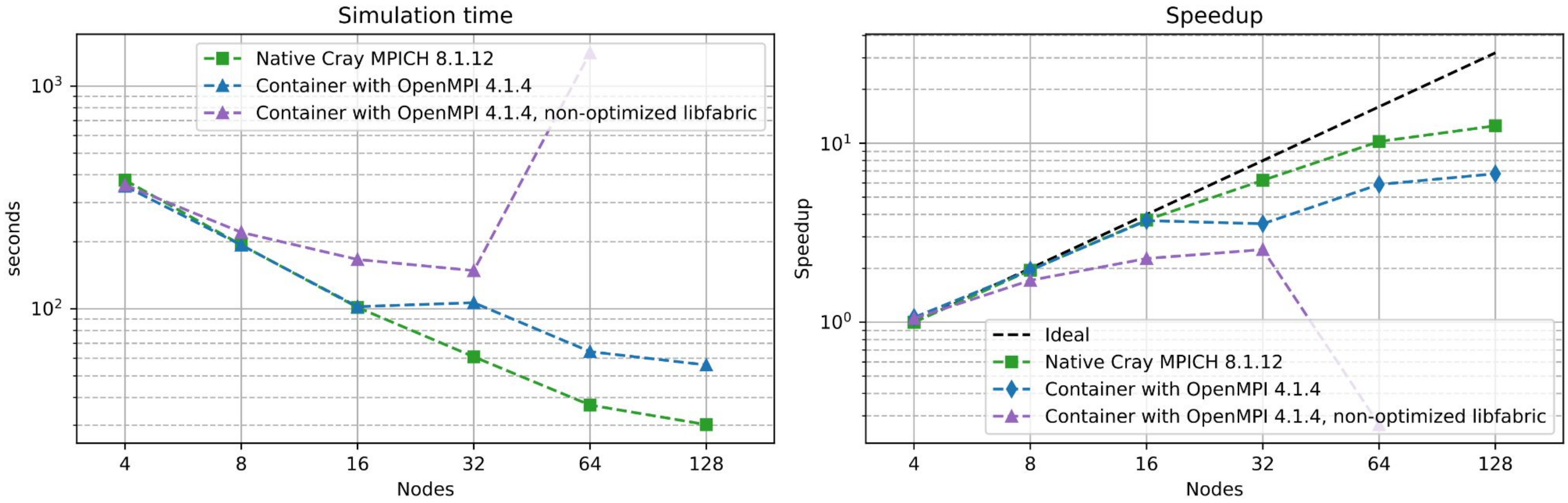
Software: PyFR 1.15.0 (OpenMP backend), HPE libfabric 1.15.0.0
Test case: SD7003 airfoil (2 ranks per node, 30 repetitions)
System: Alps Infrastructure - CPU partition (2 x AMD EPYC 7742, HPE Slingshot 11 Interconnect)

QuantumESPRESSO (Electronic Structure Computation)



Software: Quantum ESPRESSO 7.1, HPE libfabric 1.15.0.0
Test case: Si511Ge (64 ranks per node, 30 repetitions at 4-32 nodes, 1 repetition at 64-128 nodes)
System: Alps Infrastructure - CPU partition (2 x AMD EPYC 7742, HPE Slingshot 11 Interconnect)

CP2K (Quantum Chemistry and Solid State Physics)



Software: CP2K 9.1.0, HPE libfabric 1.15.0.0
Test case: Linear-scaling DFT - 2048 H2O molecules (16 ranks per node, 1 repetition)
System: Alps Infrastructure - CPU partition (2 x AMD EPYC 7742, HPE Slingshot 11 Interconnect)

Closing remarks

Conclusions

- Libfabric replacement **can** work on Slingshot 11 using HPE's proprietary libfabric
  - Compatible with different containerized MPIs
  - Reduces complexity compared to full MPI replacement
  - Enables near-native performance

- Right now, it does not *always* work: outcome depends on application, use case, MPI implementation (your mileage may vary)

- Communication frameworks (e.g. libfabric, UCX) have great potential for containers in HPC and deserve more consideration

Future work

- More testing!
  - Applications
  - Test cases
  - MPI implementations as they develop (e.g. OpenMPI 5, MPICH 4.x, MVAPICH2 3.0)

- Consolidate the approach and integrate into user-facing tools

- Explore more complex use cases:
  - Communication collectives libraries (e.g. NCCL, RCCL)
  - RDMA
  - MPI I/O

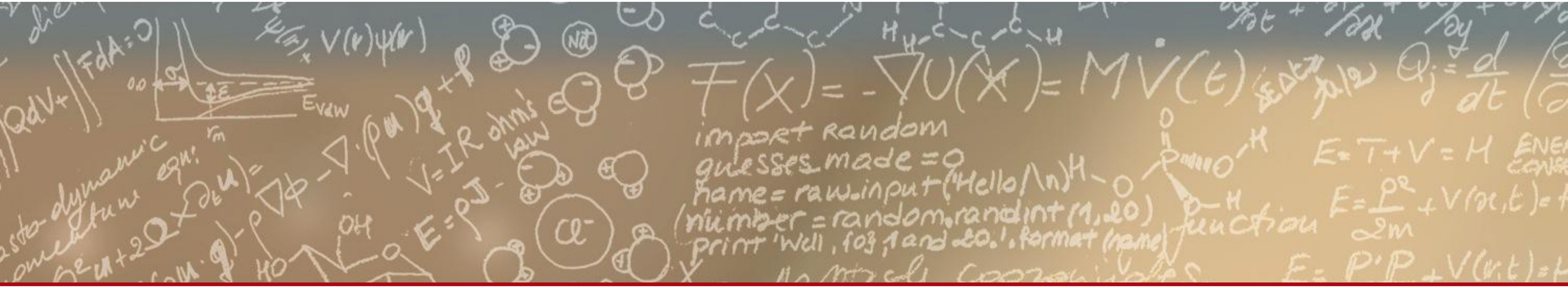
Acknowledgements

- Thanks to Dr. S. Keller (ETHZ / CSCS) for advice and assistance with the SPH-EXA test case

- Thanks to Dr. A. Kozhevnikov (ETHZ / CSCS) for advice and assistance with the QuantumESPRESSO test case

CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich



Thank you for your attention.

CSCS
Centro Svizzero di Calcolo Scientifico
Swiss National Supercomputing Centre

ETH zürich

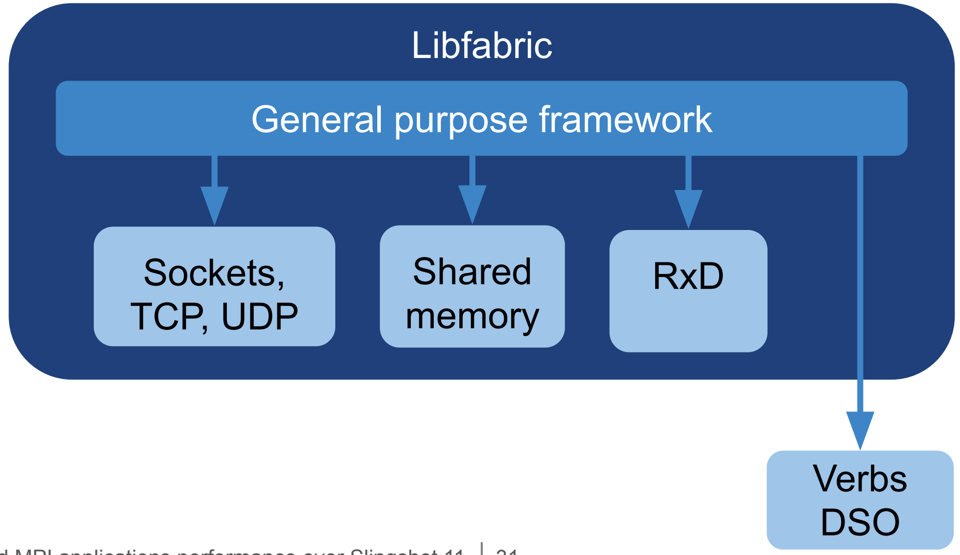
Backup slides

Dynamically linked providers!

- Providers can be compiled either as built-in or as external dynamic shared objects (DSO), for example:

```
*** Built-in providers:   perf shm rxd rxm tcp udp efa sockets
*** DSO providers:        verbs
```

- DSO providers can be loaded at runtime by a libfabric library which was not originally compiled with them



Technique: fabric provider injection

- Inject a runtime-loadable provider built on the host to augment the original libfabric from the container image

- Pros:
  - ✓ No library replacements, only additions!
  - ✓ Least amount of dependencies to inject: only add the hardware-specific resources the image is missing
  - ✓ Minimizes modifications to the image software stack

- Cons:
  - ✗ Requires image MPI to be built on libfabric
  - ✗ Image MPI must support PMI used by the host
  - ✗ Vendor-specific MPI optimizations may not be available

- ⚠️ Lack of clarity about compatibility between external providers and core libfabric