Portable, Streaming HPC Containers for the EBRAINS Software Distribution

Krishna Kant Singh[1] Eleni Mathioulaki[2] Dennis Terhorst[3] Rajalekshmi Deepu[1] Eric Müller[4]

[1]Simulation & Data Lab Neuroscience, Jülich Supercomputing Centre, Forschungszentrum Jülich, Germany [2]ATHENA Research & Innovation Center, Marousi, Greece [3]Institute for Advanced Simulation (IAS-6), Forschungszentrum Jülich, Germany [4]Kirchhoff Institute for Physics, Ruprecht-Karls-Universität Heidelberg, Germany

EBRAINS

https://ebrains.eu/esd

1. Introduction — reproducibility lives in the dependency graph

Reproducibility in computational neuroscience depends not only on the simulator but on its *entire dependency graph*. NEST 3.9 carries 72 transitive dependencies (e.g., OpenMPI, Boost, jemalloc, Python). Deploying such workflows across HPC is hindered by site-specific dependencies, heterogeneous OS, and microarchitecture-specific compiler optimisations. The EBRAINS Software Distribution (ESD) tracks dependencies rigidly via *Spack*, yet synchronising identical environments from ESD cloud deployments onto specific HPC hosts still leads to dependency hell.

We present a zero-install approach:
portable, HPC-enabled Apptainer images that reach near-bare-metal performance — and ask: *how much does portability cost?*

2. Our approach — HPC-enabled, streaming containers

Runtime Portability — PMIx bridges container-host MPI

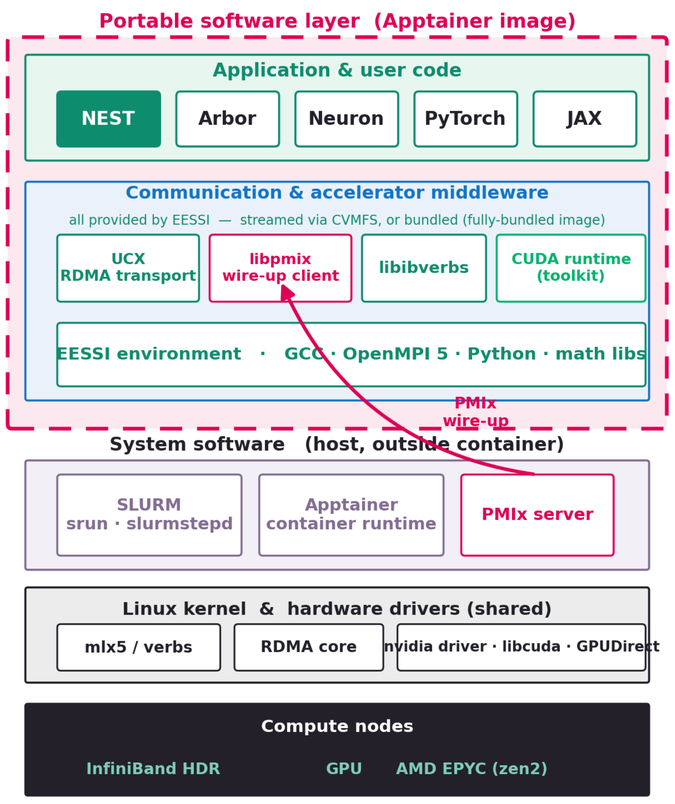
The image encapsulates OpenMPI 5.x, UCX, verbs and a PMIx client. At runtime the client connects to the host-side PMIx server exposed by SLURM, bridging container MPI to the host fabric, process manager and scheduler simultaneously.

Compile-time Portability — Multi-architecture Builds

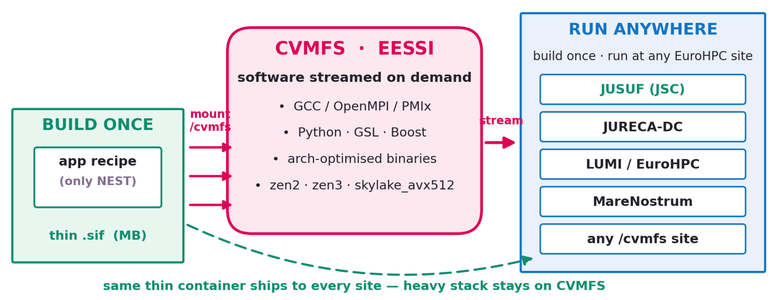
QEMU user-mode emulation inside Apptainer produces architecture-optimised builds (AVX2 & AVX-512, Intel/AMD) from a single host.

Toward "Software Streaming"

The dependency stack moves onto CernVM-FS: the image ships only the application; the EESSI stack is *streamed on demand*, so one thin image runs across EuroHPC sites.



Host / container split; magenta = PMIx wire-up.

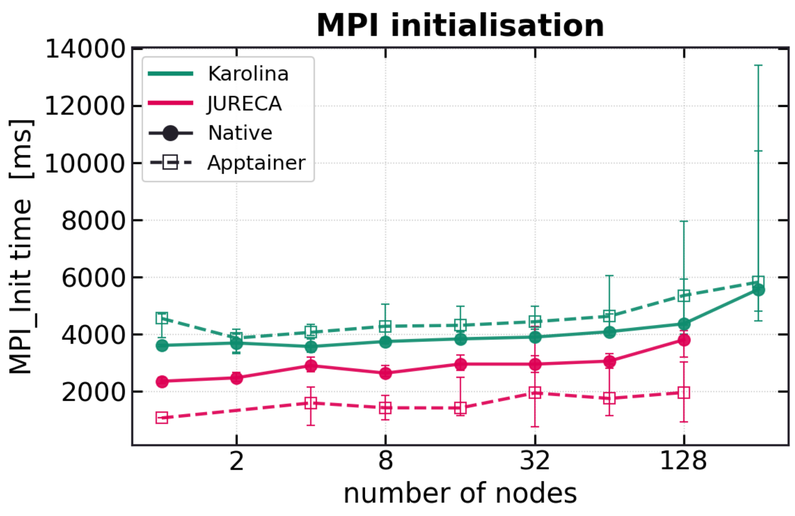


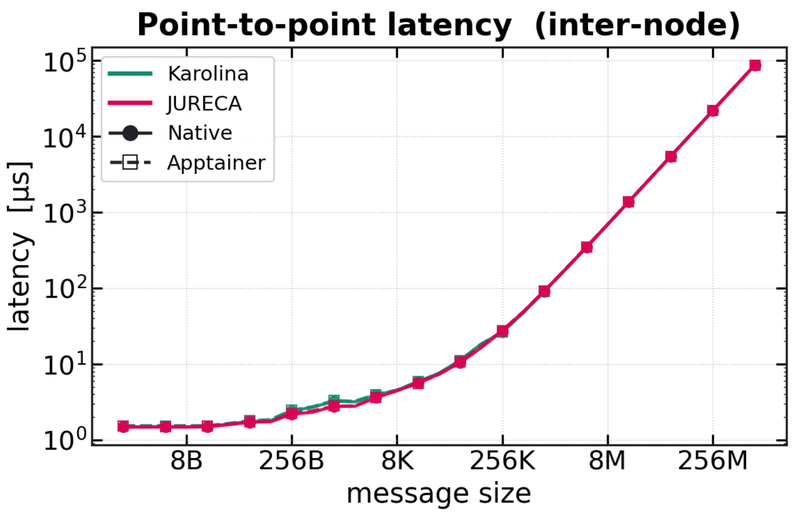
Build once → stream stack from CernVM-FS → run at any EuroHPC site.

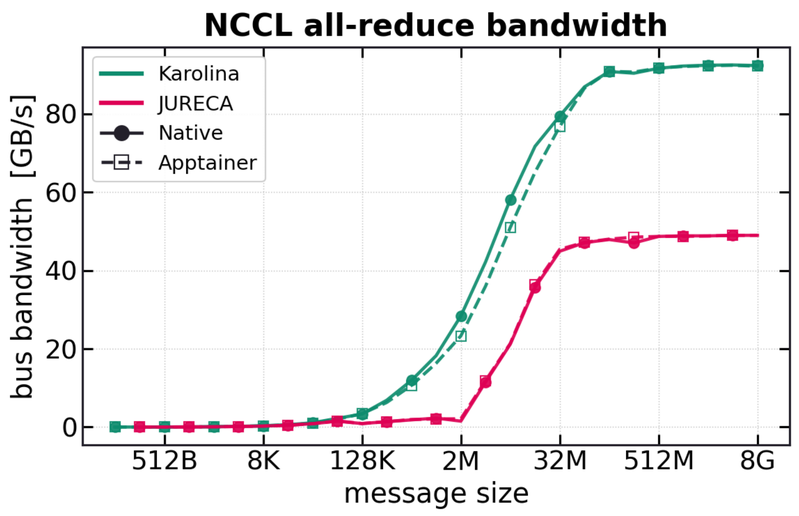
Dependency provenance across deployment models

Software component	Bare-metal	CVMFS	Bundled
OS userspace	host	in image	in image
Application: NEST	host module	in image	in image
MPI & comm. (OMPI, UCX)	host module	from /cvmfs	in image
Base libraries (Python, GCC)	host module	from /cvmfs	in image
PMIx server · fabric · drivers	host	host	host

3. Communication micro-benchmarks — container vs native (Karolina & JURECA)

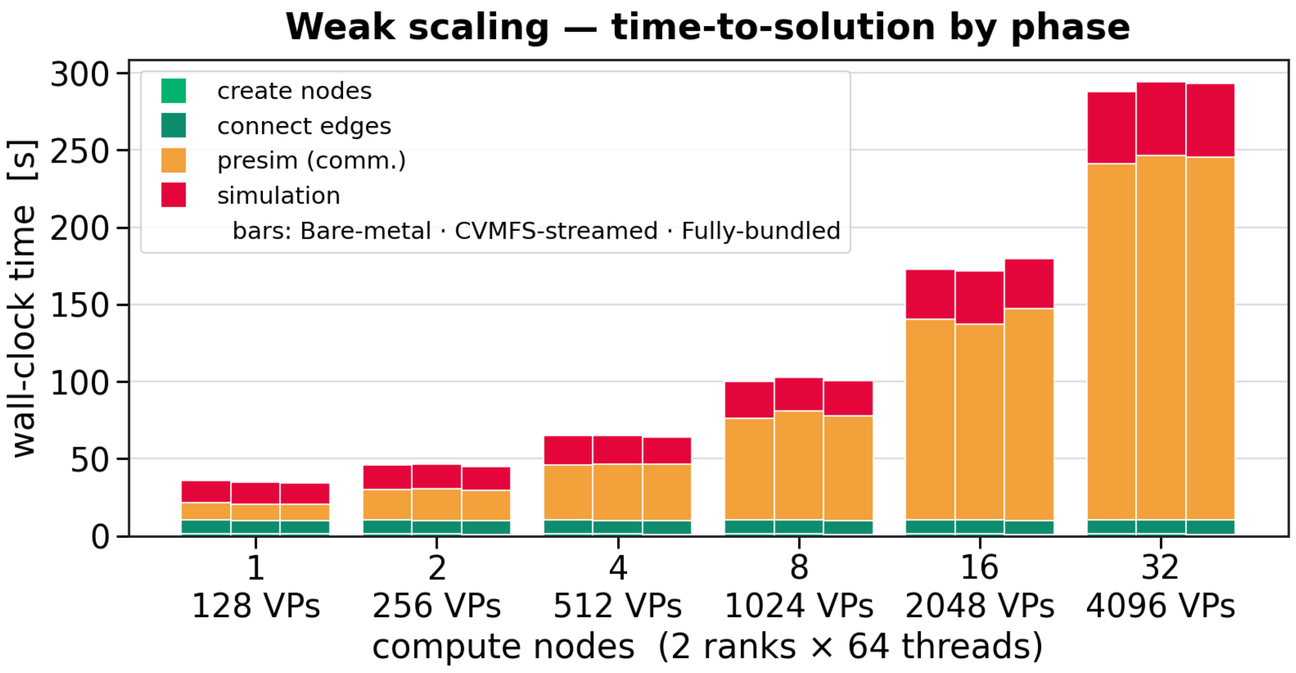


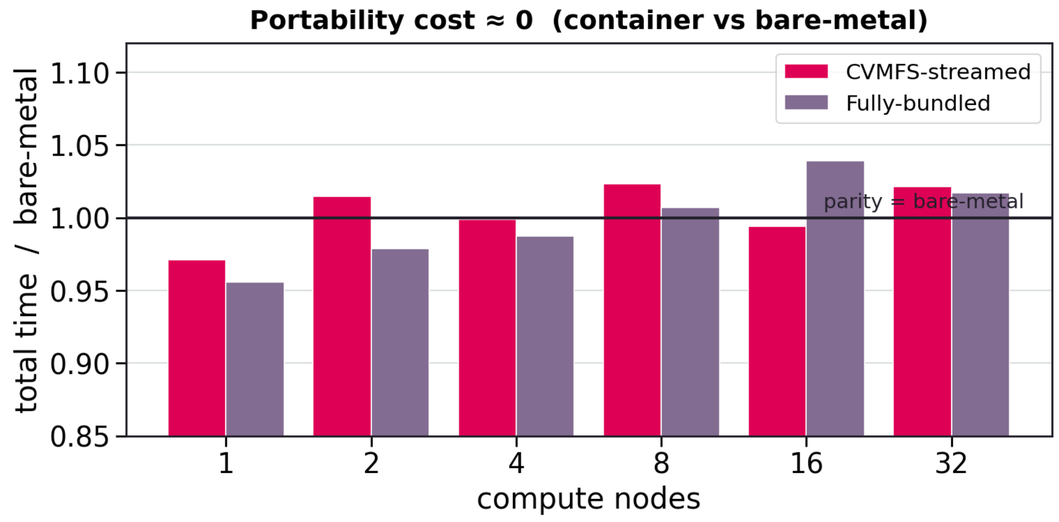




Point-to-point latency and NCCL bus bandwidth are indistinguishable from native;
MPI_Init is on par (on JURECA the container is even *faster*, insulated from a host hwloc regression).

4. NEST scaling — three deployment models on JUSUF (1–32 nodes)





Portability cost ≈ 0: both CVMFS-streamed and fully-bundled images stay within ±4% of bare-metal ParaStationMPI across 1–32 nodes.

5. Conclusion

- Portability is essentially free: containerised MPI matches native at the fabric (OSU, NCCL) and application (NEST) level.
- Build once, run anywhere: PMIx wire-up + QEMU multi-arch + CernVM-FS streaming give one thin, EuroHPC-portable image.
- A practical, zero-install path for the ESD toward on-demand HPC deployment.

References

1. Singh, Krishna Kant, *et al.* "HPC Containers for EBRAINS: Towards Portable Cross-Domain Software Environment." arXiv preprint arXiv:2603.12044 (2026).
2. J. Albers *et al.*, "A Modular Workflow for Performance Benchmarking of Neuronal Network Simulations," *Front. Neuroinform.* **16**:837549, 2022.
3. EESSI — *European Environment for Scientific Software Installations*, eessi.io.

Co-funded via EC Horizon Europe grant 101147319 (EBRAINS 2.0)