Zero to LAMMPS in Two Minutes

Scientific Achievement

- **E4S Cache-backed Image (E4S CBI):** a minimal container (Spack, MPICH) backed by a Spack build cache that can access all of the hundreds of libraries, tools, and apps in E4S
- Any E4S library, tool, or application available in "real time" — curated, reproducible, ubiquitous
- On a laptop: 42 s to pull the image, 75 s to install LAMMPS from the cache
- **Result: a fully functional parallel LAMMPS in under two minutes**

Significance and Impact

- Removes the build-and-install barrier between a researcher and a running application
- Simulation is the entry point for many AI-for-science workflows, including surrogate training
- **Scales unchanged from a laptop quick start to leadership-class runs and the cloud**

Technical Approach

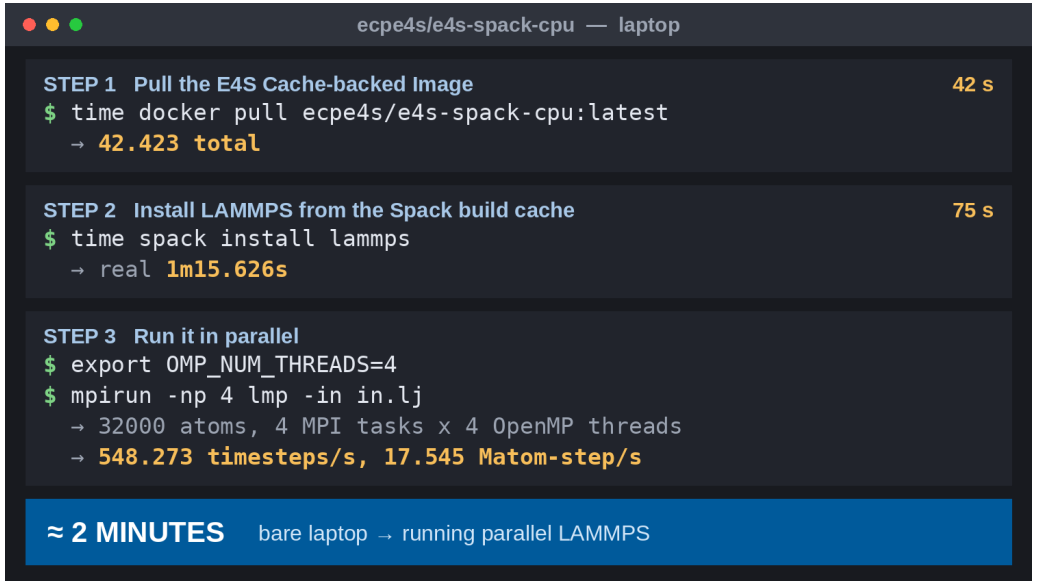
- The Cache-backed Image stays small; the build cache supplies the rest of E4S on demand
- Cached binaries replace source builds, cutting a multi-hour install to 75 seconds
- LAMMPS in.lj: 32,000 atoms, 4 MPI ranks × 4 OpenMP threads, 17.5 Matom-step/s

PESO

https://pesoproject.org

Argonne NATIONAL LABORATORY · BERKELEY LAB · Brookhaven National Laboratory · kitware · Lawrence Livermore National Laboratory · Los Alamos NATIONAL LABORATORY · OAK RIDGE National Laboratory · Pacific Northwest NATIONAL LABORATORY · ParaTools · Sandia National Laboratories · SUSTAINABLE HORIZONS INSTITUTE · Team Libra · UNIVERSITY OF OREGON

Stewarding and advancing the current and future ecosystem of HPC-AI computing software



Captured on a laptop: a 42-second docker pull of the E4S Cache-backed Image (ecpe4s/e4s-spack-cpu), a 75-second spack install lammps from the E4S build cache, and a four-rank, four-thread LAMMPS run — about two minutes end to end. E4S CBIs are available for all major CPU and GPU platforms. User content runs essentially unchanged from laptop to cloud to leadership-class systems.

The E4S Quick Start guide walks a new user through the same four steps — pull the Docker image, supply MPI code, launch the container, then compile and run — and the E4S Guide Bot answers questions interactively along the way. (Source: e4s.io)

PIs: Michael Heroux, (ParaTools, Inc.), Lois Curfman McInnes (Argonne)
E4S lead: Sameer Shende (U of Oregon/ParaTools)
E4S Cache-backed Image: ecpe4s/e4s-spack-cpu; stack curated by E4S/Spack
ASCR Program: Next-Generation Scientific Software Technologies
ASCR PM: David Rabson
E4S website: https://e4s.io
PESO website: https://pesoproject.org LAMMPS: https://www.lammps.org