This paper describes objective technical results and analysis. Any subjective views or opinions that might be expressed in the paper do not necessarily represent the views of the U.S. Department of Energy or the United States Government.

Containers and MPMD for CTH/SABLE

Anthony M. Agelastos

2023 Trilinos User Group Meeting
Containers: Current Architecture

Features

• Transmissible
  • Everything contained (minus the controlled ModSim application) is open source or there are agreements in place allowing for redistribution
  • Adherence to GPL monitored

• Tailored
  • Different flavors of UBI + multi-stage builds enable developer- and user-facing containers

• Portable
  • Successfully ran simulations and generated output atop Microsoft Windows, Apple macOS (Intel), and Red Hat Enterprise Linux

Current capability provides desktop-class simulation capabilities
Containers: Forward-Looking Architecture

Enhancements

- **Transmissible**
  - Map out ABI compatibility across Open MPI and MPICH-based MPI implementations

- **Portable**
  - Migrate to Podman (and alternatives required on target HPC platforms)
  - Begin understanding performance and deployment tradeoffs for Tier 1 HPC platforms at SNL, LANL, LLNL
  - Begin deploying containers broadly

- **Future-leaning**
  - Want to test newer versions of RHEL through UBI

R&D targeted at reducing no. of builds while minimizing performance impact
Multiphysics Coupling

- Immersed FEM to couple Lagrangian solid mechanics (e.g., *Sierra/SM*) to Eulerian shock physics (e.g., *SABLE*)

- This is currently managed via an application, i.e., *TeMPI*, between the production apps to facilitate the communications and to perform the overlap calculations in a neutral location

- The applications communicate through MPI multiple program multiple data (MPMD) execution model
  - OpenMPI: `orterun -n 3 appA : -n 1 TeMPI : -n 3 appB`
TeMPI Shim

- Creates and manages MPI communicators for an arbitrary number of MPMD-coupled applications (including no coupling)
  - Single, header-only file (i.e., easy to add to project)
  - C library (portable and able to work with Fortran, C/C++, etc.)
  - Automatically creates intra- and inter-communicators that support sending:
    - within the app
    - across pairs of apps
    - across the world
    - to root ranks of each app
  - Provides facilities for applications to create, store, and access custom MPI communicators
  - Scales better than `MPI_Init()` (tested up to 57,344 MPI ranks)

- Currently embedded within TeMPI, SABLE, and Sierra/SM
- Will add to aforementioned container R&D to understand best practices for containerized MPMD workflows
Questions?