ASC DevOps Initiative

ASC Unified Environment and Trilinos

Presented by Scott Warnock

Content also provided by ASC DevOps Technical Leadership Team, ASC DevOps Management Team, & Andrew Younge

TUG, 26 October 2022, Albuquerque, NM
AGENDA

- Overview of ASC DevOps Initiative
  - Where Trilinos Fits
  - ASC DevOps & Trilinos FY23
ASC DevOps’ focus is to minimize \( \frac{\text{time for CI/CD cycle}}{\text{(ASC-wide DevOps efficiency)}} \) through ecosystem-wide, collaborative R&D.
Containers Collaboration Framework

ASC DevOps Container Project

Runtime Support
- How do we build containers?
- Currently, on track for Podman within RHEL. Docker is still the technology for macOS and Windows.

Base Images
- Decide on a minimum set of base container images and intended use cases (e.g., RHEL, UBI) to support external deployments whereas Alpine will only support external.

Env Images
- Create "env" images that add only the libraries and build utilities needed to run app build steps (3). These are ideal as small as possible without too much effort w.r.t. breaking things on the base image.

Dev Images
- Create "developer" images that add PE stop relevant base images (e.g., some of these will mimic CTS/1/10G, some will mimic CEEREL, some ATIS-1/Trinity, some ASC13). These will contain relevant compilers, MPI, and other TPUs utilities for building and debugging the application.

Testing
- Learn how to use images from (3) and (4) across the wide spectrum of needs and use cases. For example, what is the best way to use Trilium/loopy to debug? Another example, how would we analyze a Sierra container and a CTH container to run Zappac and post process results with a Python container?

Deployment Solutions
- Investigate ways of heavily compresing (4) to have something that is truly minimal.

Supercomputers
- Now that sizes of images are understood, compute estimates for infrastructure needed to handle production rollout of this capability assuming all apps are interested (Because they are...)

1. Image Curation
- Container Registry
- HPC Container runtime
- Podman
- Docker for Win

2. Security & Validation
- Security
- RHEL, UBI
- Alpine

3. CI
- ATSE: Val-1 & 2
- CTS1/2
- Cray PE

4. CI Runners
- CTH
- Sierra
- Tensorflow
- LAM/MPI

5. Deployment
- CTH deploy-Astra
- LAM/MPI: Portmular-DuAlT

6. Supercomputers
- El Capitan
- Portmular

7. Planned or Ongoing Container Sub-projects
- Planned
- Ongoing
- Complete

Collaboration

Scope

Execution Sequence
ASC DevOps and Trilinos

1. Adopt Supported Compilers + MPI
2. Align on selected TPLs

ASC DevOps 3-Year Roadmap

ASC Unified Environments (AUE)
Tooling, Compilers + MPI

ASC DO Evaluation Space

ASC Unified Environments (AUE) TPLs
ASC DevOps and Trilinos FY23

- Setting up Planning Team
- Understand current state
- Map Trilinos builds by team
- Look for commonality
- Develop technical path forward
- FY23 Planning is seen as risk mitigation
- One hour a week
- All stakeholders in the room