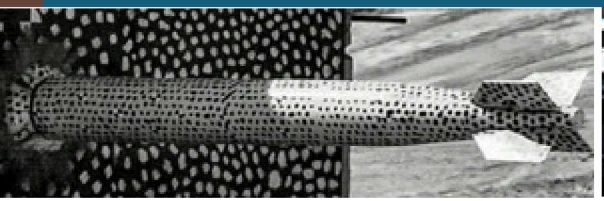
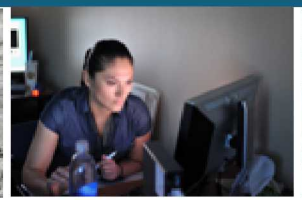
SAND2020-6326C

Supercontainers and the Future of HPC





PRESENTED BY

Andrew J. Younge

Sandia National Laboratories

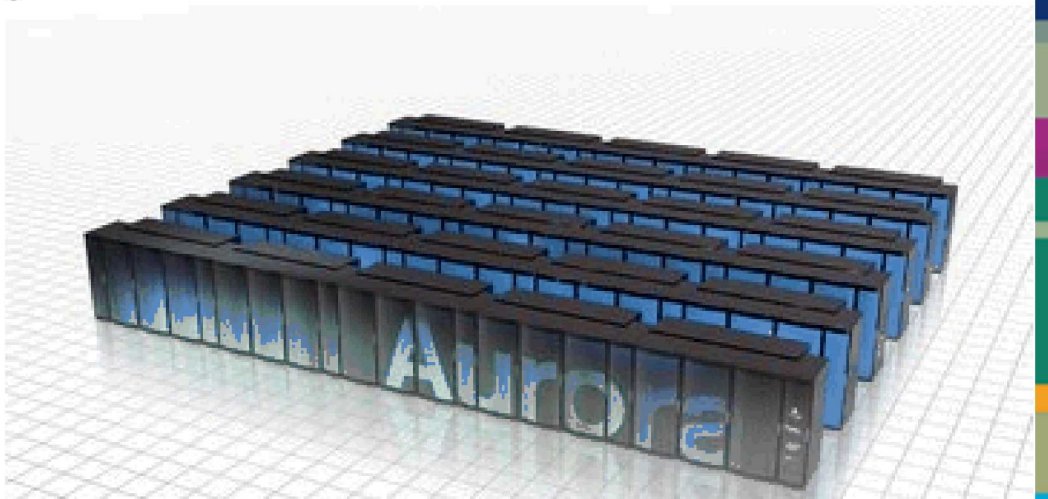
High Performance Container Workshop – 2020

(virtual)

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

DOE Exascale Computing Project

- The fastest supercomputers today solve problems at the petascale
  - 1 quadrillion (10^{15}) operations each second
  - We need to make the jump to Exascale (10^{18})
  - Many simulations fundamentally need this computational power
    - precision medicine, regional climate, additive manufacturing, the conversion of plants to biofuels, the relationship between energy and water use, physics in materials discovery and design, fundamental forces of the universe, …
- The Exascale Computing Project is the US plan to get there
  - Multi-laboratory DOE-wide effort
  - ~3-5B funding for developing next generation technologies needed for Exascale
    - Applications, Software technologies, and Hardware & Integration
  - Culminates in 3 Exascale machines:
    - Aurora –2021 – Argonne National Laboratories
    - Frontier – 2022 – Oak Ridge National Laboratories
    - El Capitan – 2023 – Lawrence Livermore National Laboratory



ECP Supercontainers

- Joint DOE effort - Sandia, LANL, LBNL, LLNL, U. of Oregon
- Ensure container runtimes will be scalable, interoperable, and well integrated across DOE
  - Enable container deployments from laptops to Exascale
  - Assist Exascale applications and facilities leverage containers most efficiently
- Three-fold approach
  - Scalable R&D activities
  - Collaboration with related ST and AD projects
  - Training, Education, and Support
    - https://github.com/supercontainers/sc19-tutorial/
- Activities conducted in the context of interoperability
  - Portable solutions
    - Optimized E4S container images for each machine type
    - Containerized ECP that runs on Astra, A21, El-Capitan, …
  - Work for multiple container implementations
    - Not picking a “winning” container runtime
  - Multiple DOE facilities at multiple scales

SUPERCONTAINERS

Extreme-scale Scientific Software Stack (E4S) https://e4s.io

- E4S is a Spack-based distribution of ECP software
  - Tested for interoperability and portability to multiple architectures
  - Focuses on ST & related software
  - A modular, interoperable, and deployable stack
- E4S is a community effort to provide open source software packages for developing, deploying, and running scientific applications on HPC platforms.
- E4S provides both source builds and containers of a broad collection of HPC software packages for multiple platforms (x86_64, ppc64le, aarch64).
- E4S exists to accelerate the development, deployment and use of HPC software, lowering the barriers for HPC users.
- E4S provides containers and turn-key, from-source builds pf popular HPC software

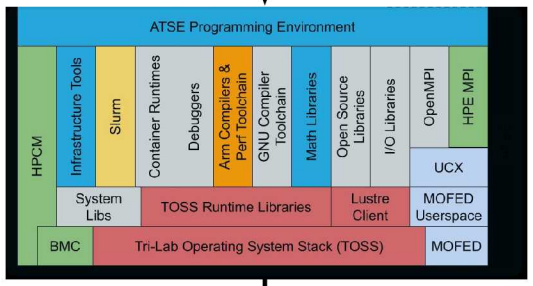
E4S

Un-privileged Container Builds

- Build containers directly on the supercomputer
  - Important for non-x86 or specialized hardware, tools, software, ..
  - Building block for CI/CD pipelines
- Podman and Buildah to provide container builds while maintaining user-level permissions
  - User namespaces
  - Set uid/gid mappers
  - Overlay & FUSE for mount
- Launch scalable job

  w/ Singularity & Slurm

```
podman build -t "gitlab.doe.gov/atse/astra:1.2.4" .
```



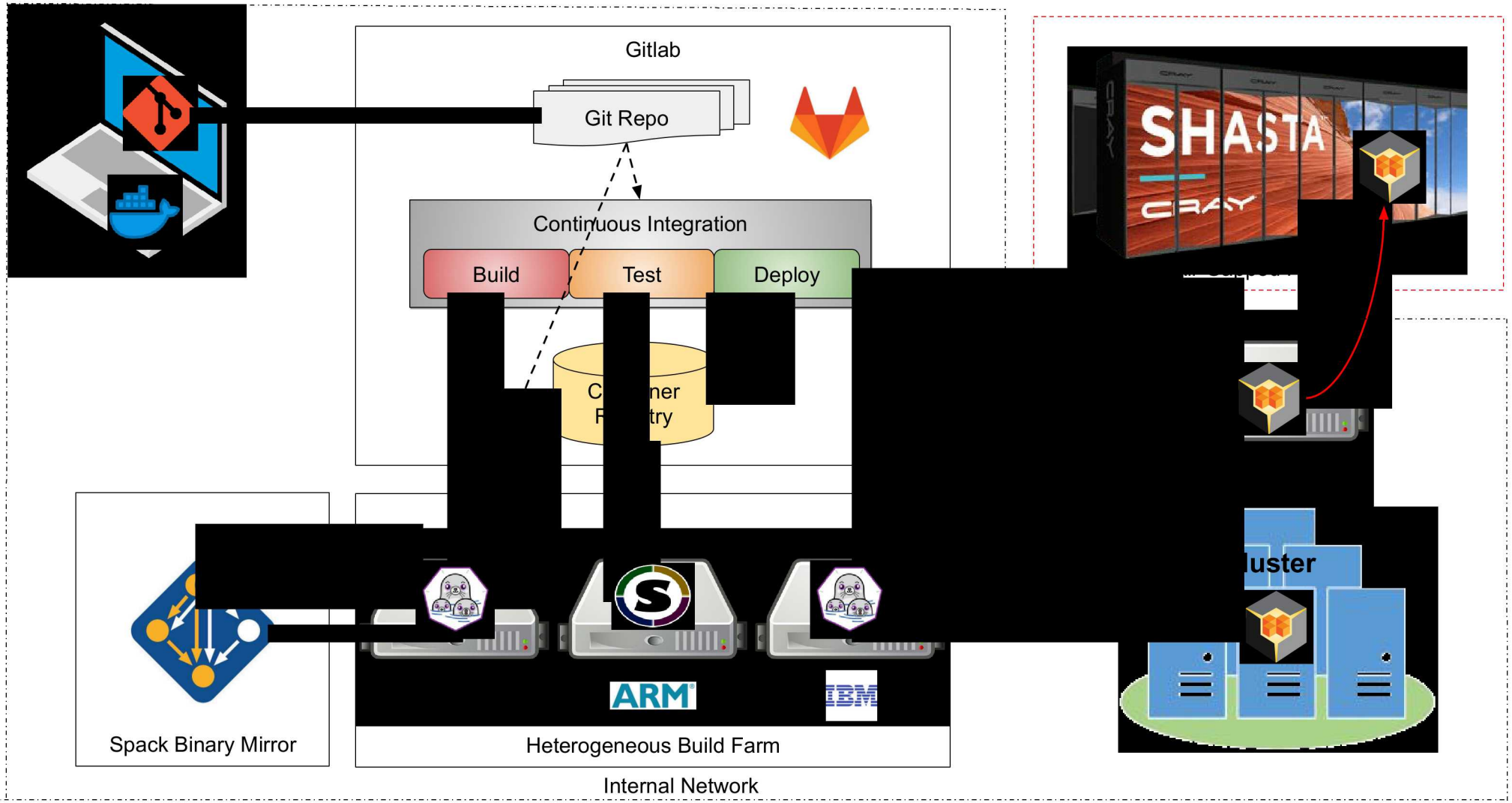

```
podman push gitlab.doe.gov/atse/astra:1.2.4
```

```
singularity build atse-astra-1.2.4.sif docker://gitlab.doe.gov/atse/astra:1.2.4
```


```
salloc -N 2048 && mpirun -np $NP singularity exec atse-astra-1.2.4.sif /app
```

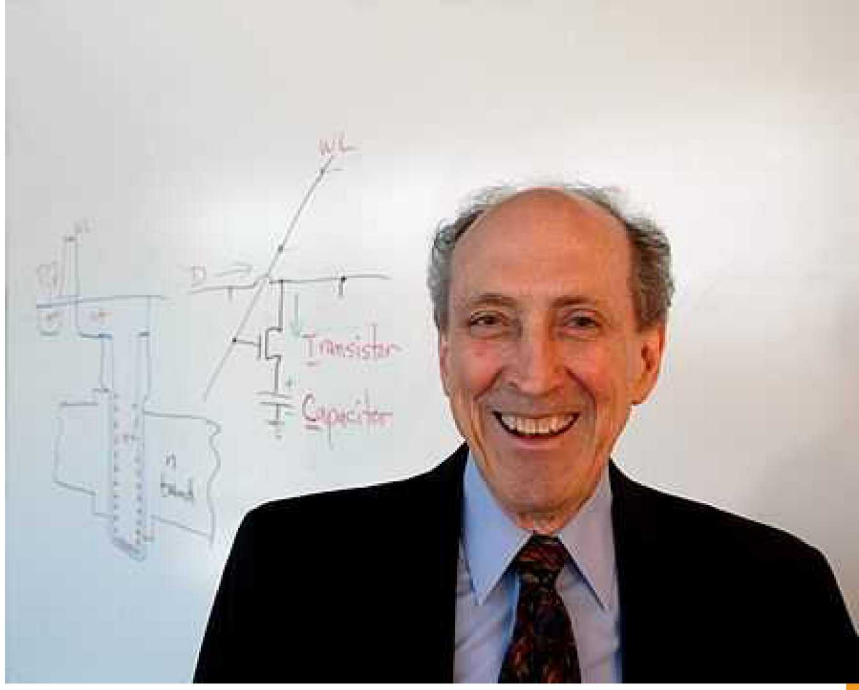
Future Containerized CI Pipeline

- As a *developer* I want to *generate container builds from code pull requests* so that *containers are used to test new code on target Exascale machines.*



An outlook?

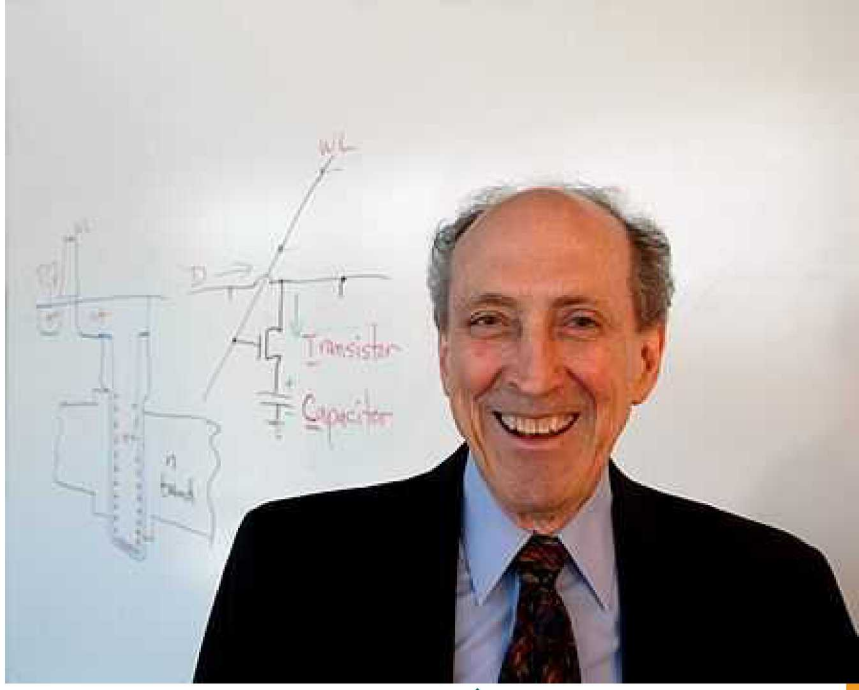
Post-Exascale HPC



- Everybody says Moore's Law is dead
  - Maybe we start to believe them?
  - How serious are they?
- HPC will still exist
- Expect HPC to evolve
  - Extremely heterogeneous platforms with diverse architectures
    - System-level heterogeneity, not GPU architectures
    - Level of heterogeneity & customization to fit computation
  - On-demand capability-class workload ensembles
    - More than just big simulations
    - More than just big DL training
    - Definitely not the edge

Post-Exascale HPC

- Everybody says Moore's Law is dead
  - Maybe we start to believe them?
  - How serious are they?
- HPC will still exist
- Expect HPC to evolve
  - Extremely heterogeneous platforms with diverse architectures
    - System-level heterogeneity, not GPU architectures
    - Level of heterogeneity & customization to fit computation
  - On-demand capability-class workload ensembles
    - More than just big simulations
    - More than just big DL training
    - Definitely not the edge

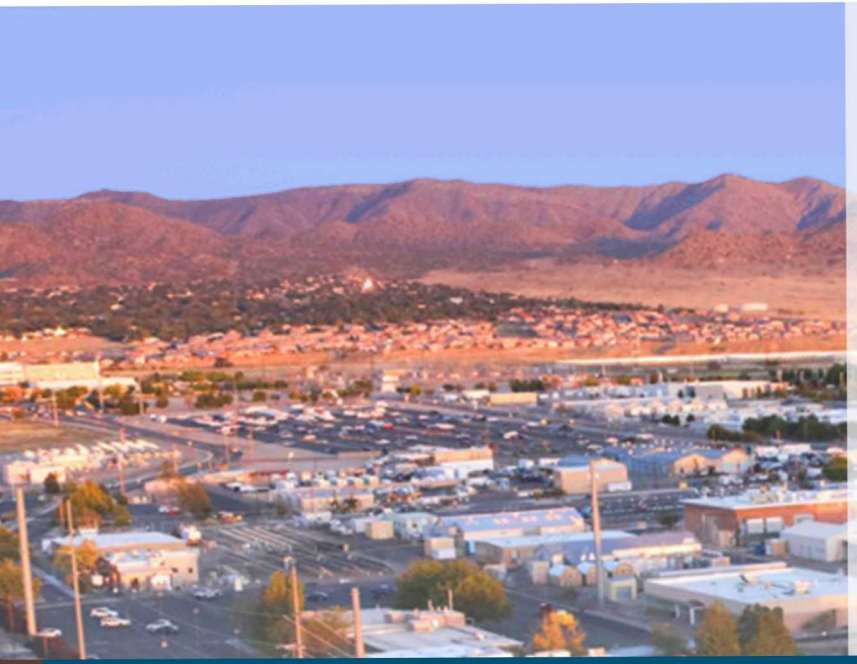


Robert Dennard

Wait... What about containers?

- Future HPC systems will demand the flexibility of containers
- We need to change **what** and **how** we schedule & orchestrate
  - Batch isn't working
  - Kubernetes is not a lift-n-shift solution
- Containers as basic scheduling singleton
  - Complete workflow ensembles will be the new norm
    - Our workflows will be diverse
    - And so will the computational requirements
  - Provide on-demand functionality with queuing capability
    - These is not mutually exclusive
    - HPC nodes are getting fat, let's make better use of them
  - Containers for extremely diverse components
    - Workflows spanning GPUs, CPUs, FPGAs, custom ASICs, SmartNICs, …
    - Continuous Integration will be the on-ramp to these architectures

SUPERCONTAINERS



Thanks

ajyoung@sandia.gov