



# Containers and Data-Centric Computing

Jay Lofstead

[glofst@sandia.gov](mailto:glofst@sandia.gov)

March 25, 2020

Supercomputing Frontiers Europe 2020



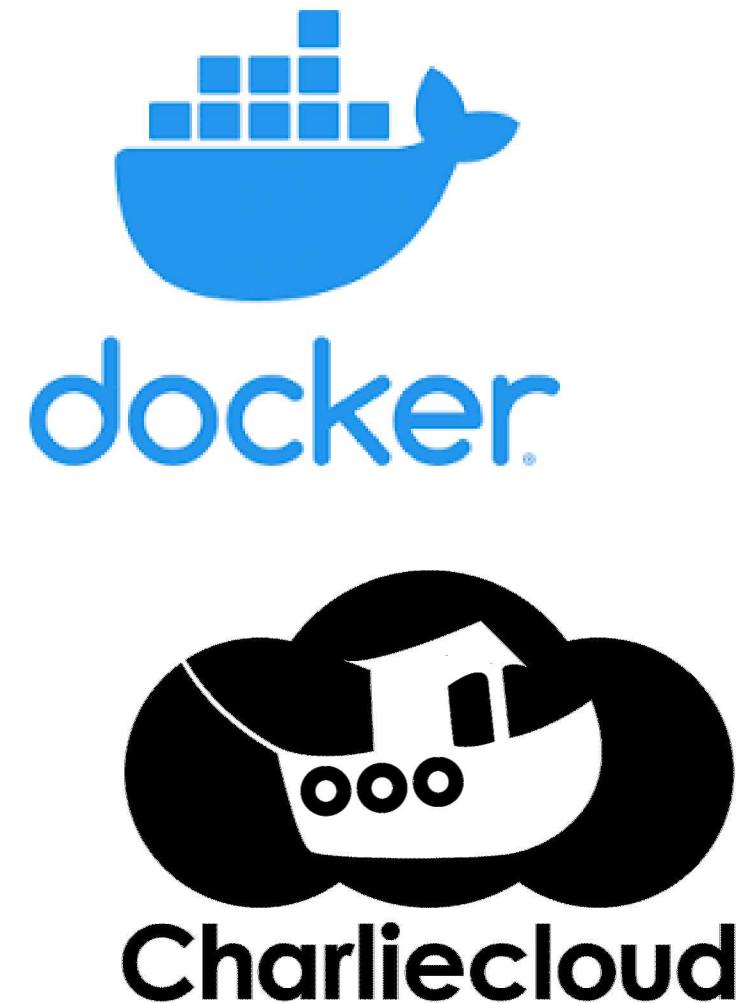
U.S. DEPARTMENT OF  
**ENERGY**



Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and 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.

# Containers Are Great for Compute

- Self-contained
- Largely portable
- Consistent processing
- Uniquely identifiable
- Public container repos!



# Large Scale Storage is Great

- Store PB to EB
- Parallel and distributed access
- Fast aggregate performance
- Familiar interfaces (mostly)



IBM  
**Spectrum  
Scale**



**BeeGFS®**



**ceph**

**lustre®**  
File System

# What About Output?

- Containerized application identifiable
  - Can save the hash ID and/or the container(s) themselves
- Data has a location and file name(s) [identifiers or keys]
- Resulting model or visualization is useful, but provenance is separate
  - Separate database, or better, part of the system
  - Linked via pointing to a location in the storage system

# A Different Approach

- PASS[1] offers a system managed database to try to track provenance in the storage system.
- It demonstrates automatic system collection, but acknowledges not portable
- Why not make a container environment that manages that data?

[1] Muniswamy-Reddy, Kiran-Kumar, et al. "Provenance-aware storage systems." *Usenix annual technical conference, general track*. 2006.

# FUSEd Storage Interfaces

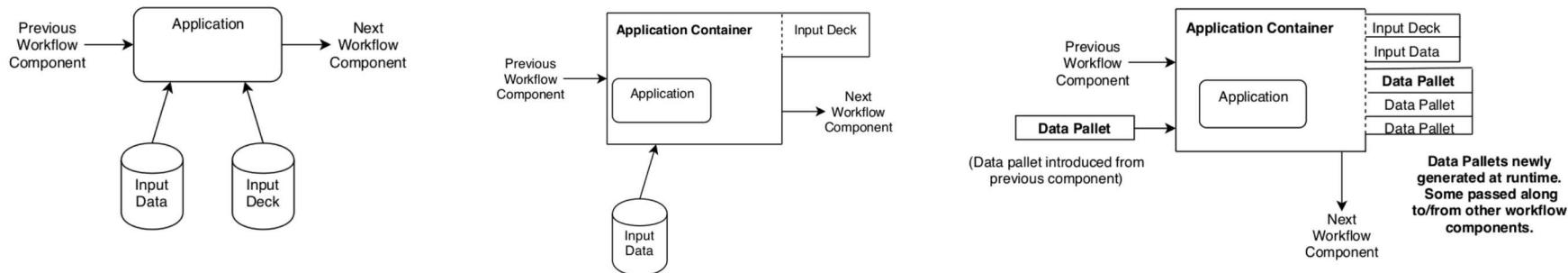
- Typical Storage Containers offer access to shared storage
- Use a new Storage Container that incorporates the provenance based on a portable database
  - Eliminate linking to the paths on a single system based on container mounting
  - Adds provenance portability

# Challenges

- Containers rely on system-installed drivers (in large part)
- Additional interface currently requires user intervention
  - Can we extend containers to just do this and look for the provenance database?
- How do we scale from single node to massively parallel?

# Data Pallets[2]

- My earlier work is called Data Pallets[2]



- Can we manage the data on the fly as a unit?
  - In short, OS restrictions says no, but we can package after the fact.

[2] Lofstead, Jay, Joshua Baker, and Andrew Younge. "Data pallets: containerizing storage for reproducibility and traceability." *International Conference on High Performance Computing*. Springer, Cham, 2019.

# Pull Containers into FS Entities

- Linking the two ideas:
  - Make storage systems “containers aware” capable of tracking based on the unique hash code rather than location or name
  - Collect provenance for files based on the containers
  - Provide a “storage container” to make accessible on other platforms
- Resulting provenance is more strongly connected and more portable since it does not rely on the names and paths on a source system being universal
- Storage Container interface offers a portable, standard interface for accessing the data independent of the storage backend

# Questions?

- Work is ongoing