ABSTRACT

Over the last 5 years, container technologies have gone from a novelty, to a niche, to a necessity for many research communities. As both infrastructure and cyberinfrastructure providers, the authors have worked to support their user communities in their adoption and use of these technologies. While Docker [1] is still the dominant mainstream container technology, the space has begun to fragment as containers move past the hype and into broad usage. Alternative image formats and container runtimes have emerged, providing compelling use cases and advantages over Docker for compute containers [2][3][4][5].

Compute containers have provided scientists a way to easily improve reproducibility and simplify the process of sharing apps with others in the field. While the concept of pre-packaged application images is similar to the way many scientists have leveraged virtualization technologies in create prepackaged machine images for use in cloud environments, the use of containers reduces the overhead of moving software across incompatible cloud providers and, ideally, enables portability between local, HPC, and cloud compute resources.

While several efforts are underway attempting to build up libraries of user-provided application codes, the Bioconda [6] project has taken a more aggressive approach by pairing Docker containers with the Bioconda repository of bioinformatics tools. The BioContainers project has generated containers for over 2000 scientifically relevant applications [7]. By building off an existing, popular polyglot dependency management system, they have taken a significant step forward in making containers approachable for anyone currently using Bioconda. Ironically, many of their academic users are still hampered by a lack of compatible container runtime environments on academic compute clusters and supercomputing systems. Those systems cannot support Docker because of issues such as existing security restrictions, incompatible versions of the Linux kernel, and hardware incompatibilities, just to name a few. These same systems are, however, increasingly choosing to support Singularity.

For the last year, the Texas Advanced Computing Center has worked closely with researchers in the CyVerse [8], Agave [9], and DesignSafe [10] projects to develop a library of containerized applications in both Docker and Singularity image formats that can be used pervasively across HPC, cloud, and local resources. Additionally, by creating a gateway that enables researchers to run these containers on academic cyberinfrastructure, we bridge the disconnect between app discovery and execution that remains unmet by existing registry services like [11-14], execution services like [15-18], and infrastructure services like [19-22]. In this talk, we present initial insights and progress adding the first 2000 images to our library, and the advantages of supporting dual container runtimes. As time permits, we will demonstrate how Agave’s Application Exchange leverages the library to enable one click, reproducible execution of any code in the

Containerizing Conda
Experiences Building 2000 Applications As Portable Docker And Singularity Images

John Fonner
Texas Advanced Computing Center
University of Texas at Austin
Austin, TX USA
jfonner@tacc.utexas.edu

Rion Dooley
Texas Advanced Computing Center
University of Texas at Austin
Austin, TX USA
dooley@tacc.utexas.edu
BioConda repository on HPC, Cloud, and local resources.

Keywords—Docker; Singularity; Agave; App Exchange; reproducibility; portability; containers; BioConda

REFERENCES