SINGULARITY USER GROUP

March 12-13, 2019

Presented By

SDSC SAN DIEGO SUPERCOMPUTER CENTER + Sylabs.io

AGENDA

Tuesday, March 12, 2019

8:00 - 8:30	**Registration and Continental Breakfast**
8:30 - 9:00	**Opening Keynote** **Singularity: Past, Present, and Future!** *Gregory Kurtzer, CEO, Sylabs Inc.*
9:00 - 10:00	**Morning Hands-on Session** **Creating and Running Software Containers with Singularity** *Eduardo Arango, Software Engineer, Sylabs Inc.*
10:00 - 10:15	**Break**
10:15 – 12:15	**What's New in Singularity 3.0** Dave Godlove, Product Manager, Sylabs Inc. **LiCO and Singularity - Driving the Convergence of HPC & AI** JJ Falkanger, Sr. AI Software & Solutions Manager, Lenovo **System Telemetry and I/O Profiling for Cloud Migration and Containers** Rosemary Francis, CEO, Ellexus **Combining VMware vSphere w/ Singularity Containers Brings Together the Best of Both Worlds** Mohan Potheri, Solutions Lab & HPC Solutions Architect, VMware
12:15 - 1:00	**Lunch**
1:00 – 3:00	**PMIx: Bridging the Container Boundary** Ralph Castain, Intel **Using Singularity Containers for Research Workflows and Reproducible Science** BJ Lougee, Federal Reserve Bank of Kansas City **Contributed Talk (TBD)**
3:00 - 3:15	**Break**
3:15 – 6:15	**Best Practices for Containerizing InfiniBand** Parav Pandit, Mellanox **From Containerizing Testbeds for HPC Applications to Exascale Supercontainers** Andrew Younge, Sandia **FuncX: A Function Serving Platform for HPC** Rick Wagner, University of Chicago **Supporting Singularity Containers on the ORNL's Summit Leadership Computing Platform** Geoffrey Vallee, ORNL **Contributed Talk (TBD)**
6:15 - 8:00	**Evening Reception**

Wednesday, March 13, 2019

8:00 - 8:30	**Continental Breakfast**
8:30 – 10:30	**Building a Testing Infrastructure for Singularity - A Complex CGO Project** Eduardo Arango, Software Engineer, Sylabs Inc. **Managing Large-Scale Cosmology Simulations with Parsl and Singularity** Rick Wagner, University of Chicago **Using Singularity Containers to Enable Scalable Interactive Workflows on Harvard Odyssey HPC Cluster** Francesco Pontiggio, Harvard FAS Research Computing **NVIDIA HPC Container Efforts: An Overview** Adam Simpson, NVIDIA
10:30 - 10:45	**Break**
10:45 - 12:15	**Simplify Your Containers with Spack!** Gregory Beckner, Lawrence Livermore National Laboratory **Kubernetes the Next Research Program** Bob Killen, University of Michigan **Contributed Talk (TBD)**
12:15 - 1:00	**Lunch**
1:00 -3:00	**Status Update on the Integration of Singularity and Kubernetes** Michael Bauer, Software Engineer, Sylabs Inc. **Managing and/or Orchestrating Containerized Workloads** Ian Lumb, Technical Writer, Sylabs Inc. **Singularity and High Throughput Computing** Dave Dykstra, Fermilab **Providing a National Software Repository for Interactive HPC** Lance Wilson, Jafar Lie-Monash University
3:00 – 3:15	**Break**
3:15 – 3:45	**Closing Keynote** **Enabling Scientific Computing with Singularity Containers on SDSC Comet** Shawn Strande & Mahidhar Tatineni, SDSC
3:45 – 4:30	**Town Hall**

Agenda is subject to change, please visit our website at https://www.eiseverywhere.com/ehome/sug2019/839014/ to stay up to date!

A special thank you to our sponsor!

Lenovo

Featured Speakers

LiCO and Singularity - Driving the Convergence of HPC & AI

JJ Falkanger, Sr. AI Software & Solutions Manager, Lenovo

Scientific researchers and commercial enterprises have been increasingly applying high performance computing (HPC) and artificial intelligence (AI) techniques for drivinginsights and business value, often in the same computing environment. This convergence demands flexible tools to effectively utilize the resources and workflows. Lenovo Intelligent Computing Orchestration (LiCO) provides an intuitive web-based interface for managing and deploying both AI and HPC workloads on the same cluster. Built upon an OpenHPC management stack, LiCO enables even inexperienced users to utilize cluster resources, thanks to Singularity container management and job template integration. This session covers the capabilities of LiCO, and the many benefits for users Singularity provides as part of the architecture.

System Telemetry and I/O Profiling for Cloud Migration and Containers

Rosemary Francis, CEO, Ellexus

Many organisations are looking to move their compute architectures to the cloud and into new technologies such as containerisation. However, dependence of legacy systems and large amounts of shared storage can make it hard to migrate to more flexible compute environments. Without a clear list of dependencies, for example, it can take a lot of time to containerise applications and to size storage appropriately for the application in the new environment.

This presentation will use industry case studies to show attendees how they can get a clear picture of their shared storage requirements in order to develop an informed go-to-cloud and containerisation strategy with I/O profiling tools from Ellexus and Singularity containers. By using system telemetry, attendees will gain an understanding of how to manage hybrid cloudarchitectures. We will describe techniques and best-practices for use by small teams, for corporation-wide infrastructure strategy or those just starting to look at Singularity.

Combining VMware vSphere w/ Singularity Containers Brings Together the Best of Both Worlds

Mohan Potheri, Solutions Lab & HPC Solutions Architect, VMware

VMware vSphere provides excellent capabilities for isolating workloads and sharing HW. Singularity containers are an excellent packaging mechanism for applications. We will look at how vSphere virtual machines and Singularity can be combined effectively for sharing GPUs in machine learning applications. We will discuss some experiments we have done in image processing and the 3X gains we saw in processing throughput by using this solution.

Featured Speakers

PMIx: Bridging the Container Boundary
Ralph Castain, Intel

High-Performance Computing (HPC) applications have historically executed in static resource allocations, using programming models that ran independently from the resident system management stack (SMS). The Process Management Interface - Exascale (PMIx) community is committed to establishing scalable workflow orchestration by defining an abstract set of interfaces by which not only applications and tools can interact with the resident SMS, but also the various SMS components can interact with each other. PMIx has been widely adopted by both HPC programming libraries (e.g., MPI, OpenSHMEM) and SMS vendors (e.g., Slurm, JSM, PBSPro), and work is underway to extend support into non-HPC areas (e.g., Spark, Kubernetes). This presentation will provide an overview of the project and how it impacts the container community, and introduce a new community project (Epyx) specifically tailored to supporting containers in PMIx-enabled environments.

Using Singularity Containers for Research Workflows and Reproducible Science
BJ Lougee, Federal Reserve Bank of Kansas City

InfiniBand is the leading interconnect accelerating HPC applications, delivers high bandwidth, low latency, RDMA and In-Network Computing advantages. The InfiniBand technology enables to bypasses the kernel, and offloads the network transport to the Host Channel Adapter (HCA). In the case of non-RDMA networks, when using containerized application, only system call level ABI compatibility needed between the Linux kernel and the container. However, a containerized application which uses RDMA, needs to ensure ABI compatibility between (a) application and kernel, (b) kernel and HCA and (c) user space stack and HCA. This talk highlights some of the best practices for InfiniBand in such environment. In this talk we will also discuss how to containerize InfiniBand for Singularity platform in an easy manner.

From Containerizing Testbeds for HPC Applications to Exascale Supercontainers
Andrew Younge, Sandia

This talk introduces the ECP Supercomputing Containers Project, named Supercontainers, which represents a consolidated effort across the DOE and NNSA to use a multi-level approach to accelerate adoption of container technologies for Exascale. A major tenant of the project is to ensure that container runtimes are well poised to take advantage of future HPC systems, including efforts to ensure container images can be scalable, interoperable, and well integrated into Exascale supercomputing across the DOE. The project will focus on foundational system software research needed for ensuring containers can be deployed at scale and will provide enhanced user and developer support for enabling containerized Exascale applications and software are both efficent and performant. Furthermore, these activities will be conducted in the context of interoperability, effectively generating portable solutions that work for HPE applications across DOE facilities ranging from laptops to Exascale platforms.

Featured Speakers

FuncX: A Function Serving Platform for HPC

Rick Wagner, University of Chicago

There is a growing need to support the execution of short-duration tasks on High Performance Computing (HPC) infrastructure. To address this need we have developed FuncX: a function serving platform that is designed to provide on-demand execution of stateless functions directly from existing HPC infrastructure. FuncX aims to remove the barriers that exist between functions and large-scale computing infrastructure by simplify execution of function-based workloads, enabling integration of HPC capabilities in applications and analysis pipelines, and ultimately increasing the usability of computing infrastructure. FuncX uses the Parsl scripting library to provide low-latency, scalable, and reliable execution of tasks across heterogeneous computing environments. It securely isolates function executions within containerized Singularity execution environments and leverages Globus, as an identity and access management framework and to automate the movement of functions and data. In this talk I will describe the use cases that have motivated the development of FuncX, present the design of the FuncX execution framework, and present the prototype implementation and early experiences using the system.

Supporting Singularity Containers on the ORNL's Summit Leadership Computing Platform

Geoffrey Vallee, ORNL

Leadership computing systems traditionally have been designed for the execution of parallel simulations with the assumptions that these applications optimized will have an optimal access to the network and other hardware resources such as accelerators in order to guarantee maximum performance. However, this approach imposes application teams to adapt their codes to the target platforms, sometimes leading to major investments in terms of time and manpower when a new systems is delivered.

A natural way to try to address this limitation and potentially limit the cost to support a new computing platform is to let application teams define, instantiate and run their customized environments. Historically, virtual machines were considered to achieve that goal but were proven to impose an execution overhead that was too high for HPC workloads. Fortunately, the venue of new technologies such as container shave proven a drastically lower execution overhead (at the cost of some isolation capabilities, e.g., the kernel is shared by the host and the guests) without imposing overbearing limitations on the customization of execution environments.

In this study, we present our experience with supporting Singularity on leadership computing systems at ORNL, from the creation of images to the execution of Singularity containers for the execution of MPI applications, highlighting its benefits and limitations. Furthermore, with the venue of architectures such as Summit, it is clear that the hardware architecture trend is to provide bigger but fewer computer nodes, creating a new need for the support of MPI+X applications. We therefore also present our preliminary experience for the execution of MPI+OpenMP applications in Singularity containers on Summit.

Featured Speakers

Managing large-scale cosmology simulations with Parsl and Singularity

Rick Wagner, University of Chicago

In preparation for the Large Synoptic Survey Telescope (LSST), we are working with dark energy researchers to simulate images that are similar to the raw exposures that will be generated from the telescope. To do so, we use the imSim software package (https://github.com/LSSTDESC/imSim) to create images based on catalogs of astronomical objects and by taking into account systematic effects of the atmosphere, optics, and telescope. In order to produce data comparable to what the LSST will create, we must scale the imSim workflow to process tens of thousands of instance catalogs, each containing millions of astronomical objects, and to simulate the output of the LSST's 189 telescope sensors. To address these needs, we have developed a Parsl-based workflow that coordinates the execution of imSim on input instance catalogs and for each sensor. We package the imSim software inside a Singularity container so that it can be developed independently, packaged to include all dependencies, trivially scaled across thousands of computing nodes, and seamlessly moved between computing systems. The Parsl workflow is responsible for processing instance catalogs, determining how to pack simulation workloads onto compute nodes, and orchestrating the invocation of imSim in the Singularity containers deployed to each node. To date, the simulation workflow has consumed more than 30M core hours using 4K nodes (256K cores) on Argonne's Theta supercomputer and 2K nodes (128K cores) on NERSC's Cori supercomputer. The use of Singularity not only enabled efficient scaling and seamless conversion to support other container technologies, but it was also an integral part of our development process. It significantly simplified the complexity of developing and managing the execution of a workflow as part of a multi-institution collaboration and furthermore it removed much of the difficulties associated with execution on heterogeneous supercomputers.

Using Singularity containers to enable scalable interactive workflows on Harvard Odyssey HPC cluster

Francesco Pontiggio, Harvard FAS Research Computing

Harvard FAS Research Computing has recently deployed a web-portal to allow users to run on the HPC cluster Odyssey interactive GUI jobs,including remote desktop sessions, notebook oriented workflows, GUI desktop software and other applications with a web-based frontend like for example RStudio server. Trying to deploy and run those applications out of our standard cluster software repository resulted in numerous challenges.
We encountered substantial difficulties in terms of ease of development and we faced compatibility issues with some specific software packages not designed to run in multi-tenant HPC environments. Moreover we faced severe limitations in the application startup performance, due to some aspect of our storage infrastructure not originally designed to support those type of workflows. While a slow startup is not a particularly relevant factor in the context of typical batch jobs run on our system, it becomes an important factor in negatively impacting the user experience when scheduling and accessing interactive applications.
Using Singularity containers proved crucial for an efficient deployment of our new portal. It gave us flexibility in the application development and allowed us to circumvent intrinsic limitations of specific applications. Moreover the use of containers allowed us to leverage our high performance lustre storage, particularly suited fast read access of large size files, allowing a significant performance boost in the applications startup time.

Featured Speakers

Simplify your containers with Spack!
Gregory Beckner, Lawrence Livermore National Laboratory

Spack is an open-source package manager for HPC. Its simple, templated Python DSL allows the same package to be built in many configurations, with different compilers, flags, dependencies, and dependency versions. Spack allows HPC end users to automatically build any of over 3,000 community-maintained packages, and it enables software developers to easily manage large applications with hundreds of dependencies. These capabilities also enable Spack to greatly simplify HPC container builds.

This presentation will give an overview of Spack, including recent developments and a number of items on the near-term roadmap. We will focus on container use cases of interest to the Singularity community, including: optimized binary packages, improved developer workflows with Spack environments, and better support for large-scale HPC facility deployment. We will also discuss how Spack can be used to simplify container recipes, and some ideas for making Spack automatically generate multi-stage container builds.

Kubernetes the next research program
Bob Killen, University of Michigan

Kubernetes has become the defacto standard as a platform for container orchestration. Its ease of extending and many integrations has paved the way for a wide variety of data science and research tooling to be built on top of it.

From all encompassing tools like Kubeflow that make it easy for researchers to build end-to-end Machine Learning pipelines to specific orchestration of analytics engines such as Spark;Kubernetes has made the deployment and management of these things easy. This presentation will showcase some of the larger research tools in the ecosystem and go into how Kubernetes has enabled this easy form of application management.

Singularity and High Throughput Computing
Dave Dykstra, Fermilab

Singularity is used heavily in production by multiple High Energy Physics (HEP) projects (the LHC's CMS, the Open Science Grid Virtual Organization, and several Fermilab projects) on hundreds of thousands of CPU cores worldwide based on High Throughput Computing (HTC, also known as grid computing). Many more projects are making plans to also use it. The HEP projects rely on singularity for both controlling the software environment and for isolating jobs from end users that are run by unprivileged pilot jobs. They exclusively use the singularity sandbox mode, with images unpacked in the CernVM FileSystem (CVMFS) which securely distributes the images by means of a very efficient on-demand caching infrastructure. They are currently rolling out a conversion to using non-setuid singularity, based on RHEL 7.6 which introduced full support for unprivileged namespaces. This talk will introduce the Singularity User Group to this important singularity user community.

Featured Speakers

Providing a national software repository for interactive HPC

Lance Wilson, Jafar Lie-Monash University

The characterisation community within Australia has been working towards a federated analytics platform to provide researchers with the tools, compute and storage needed to make the most of national investments in instruments. The current user base i ~1000
across disciplines from neuroscience to neutron and synchrotron science. One of the highlighted needs was for consistency between software stacks provided on different HPC systems. The initial work has been an extension of the existing container first philosophy on the MASSIVE HPC facility (Goscinski, 2014) which provides the Characterisation Virtual Laboratory (https://www.cvl.org.au). The repository for the build files is public (https://github.com/Characterisation-Virtual-Laboratory/CharacterisationVL-Software) and is run in the same way as a software development project. In addition to the build files being public, the repository is linked to singularity hub (https://singularity-hub.org/collections/1396), such that the containers are easily available to anyone in the characterisation research community. The containers are a diverse mix of single applications through to workflows, highly dependent on the intended use by the research community.
Containers have also simplified aspects of user support, for single applications that are particularly difficult to install on complex HPC systems. It has also created a new support aspect in workflows or software aggregation tools, where the expectation from the software developers is that they are installed locally in the path. Overall however software support is now more convenient for both the administrators and researchers especially where we partner with software developers to provide containers.