Scaling Research with the NSF Nautilus HyperCluster

A Tutorial and Case Study

University of Missouri

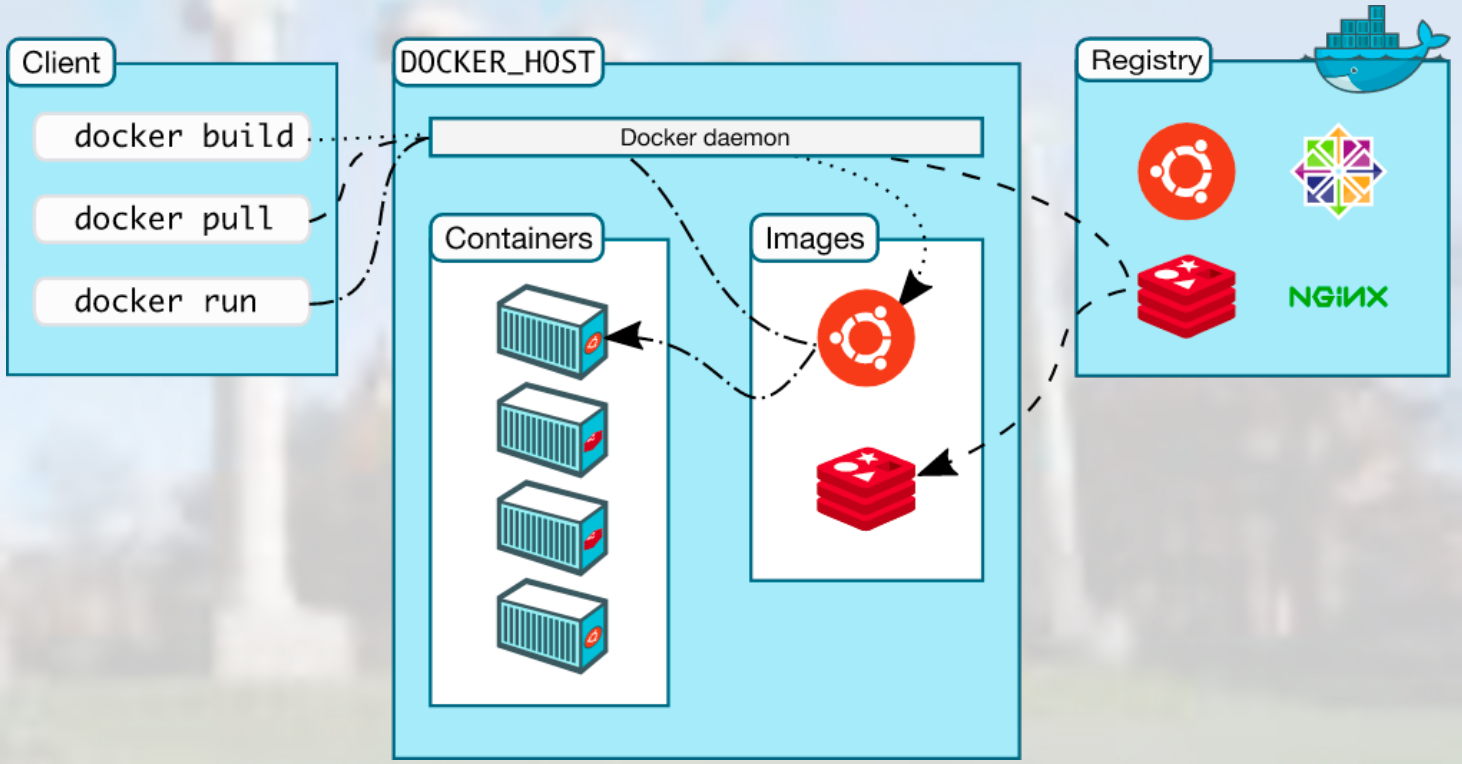
November 4, 2022

Overview

- Introduction
  - Docker
    - Containerization
    - Key Concepts: Images, Containers, Registries
  - Kubernetes
    - Container Orchestration
    - Key Concepts: Nodes, Namespaces, Pods, Jobs
- NSF Nautilus HyperCluster
  - Resources Available
  - System Overview
  - By the Numbers
- Using Nautilus: Tutorial
  - Getting Started
  - Using Persistent Storage
    - S3 Cloud Storage Integration
  - Creating a Pod
  - Creating a Batch Job
  - Using Nautilus for Deep Learning
  - Automating Deep Learning Job Deployments

Docker

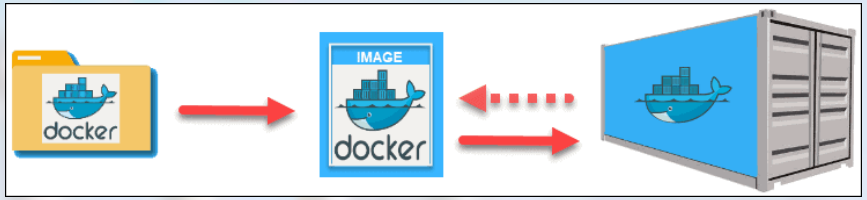
- **What is Docker?**
  - Docker is a set of platform as a service products that use OS-level virtualization to deliver software in packages called containers.[1]
  - You can think of Docker containers as mini-VMs that contain all the packages, both at the OS and language-specific level, necessary to run your software.
- **Why Docker?**
  - Docker enables predictable and reliable deployment of software.
  - Docker containers are portable!
    - local development computers, compute clusters, internal compute servers, cloud infrastructure, and more!
- *Docker containers are how software is deployed in **Kubernetes**!*

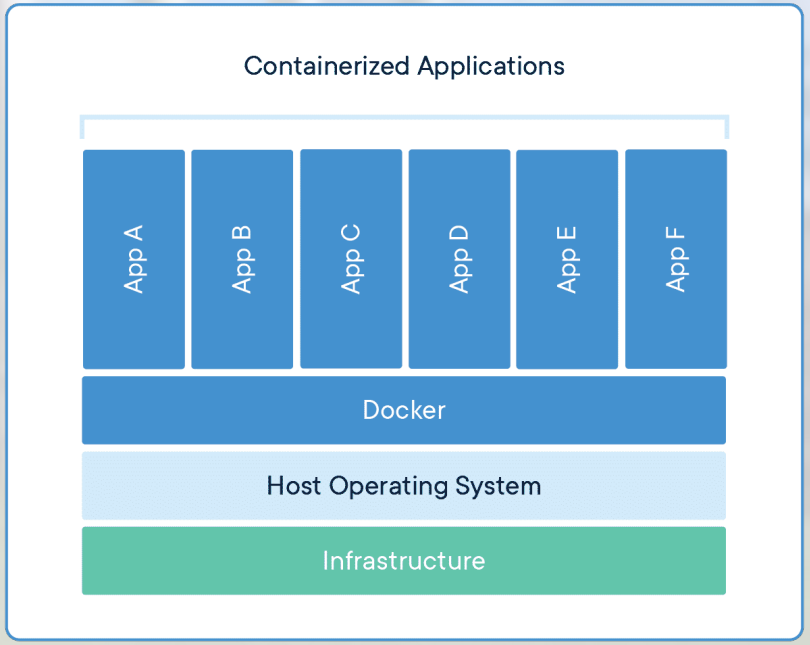


1. https://en.wikipedia.org/wiki/Docker_(software)
2. Image: https://docs.docker.com/get-started/overview/

Key Docker Concepts

- **Dockerfile** - A list of commands and instructions describing how to build an Image
- **Image** - Standard unit of software that packages up code and its dependencies so the application runs reliably from one computing environment to another.
  - Includes everything needed to run an application: code, runtime, system tools, system libraries and settings.
- **Container** - An instantiated runtime of a docker image, containing all necessary software for a given application to run, both at the OS and language-specific level
  - Images become containers at runtime
- **Registry** - a service for storing private container images

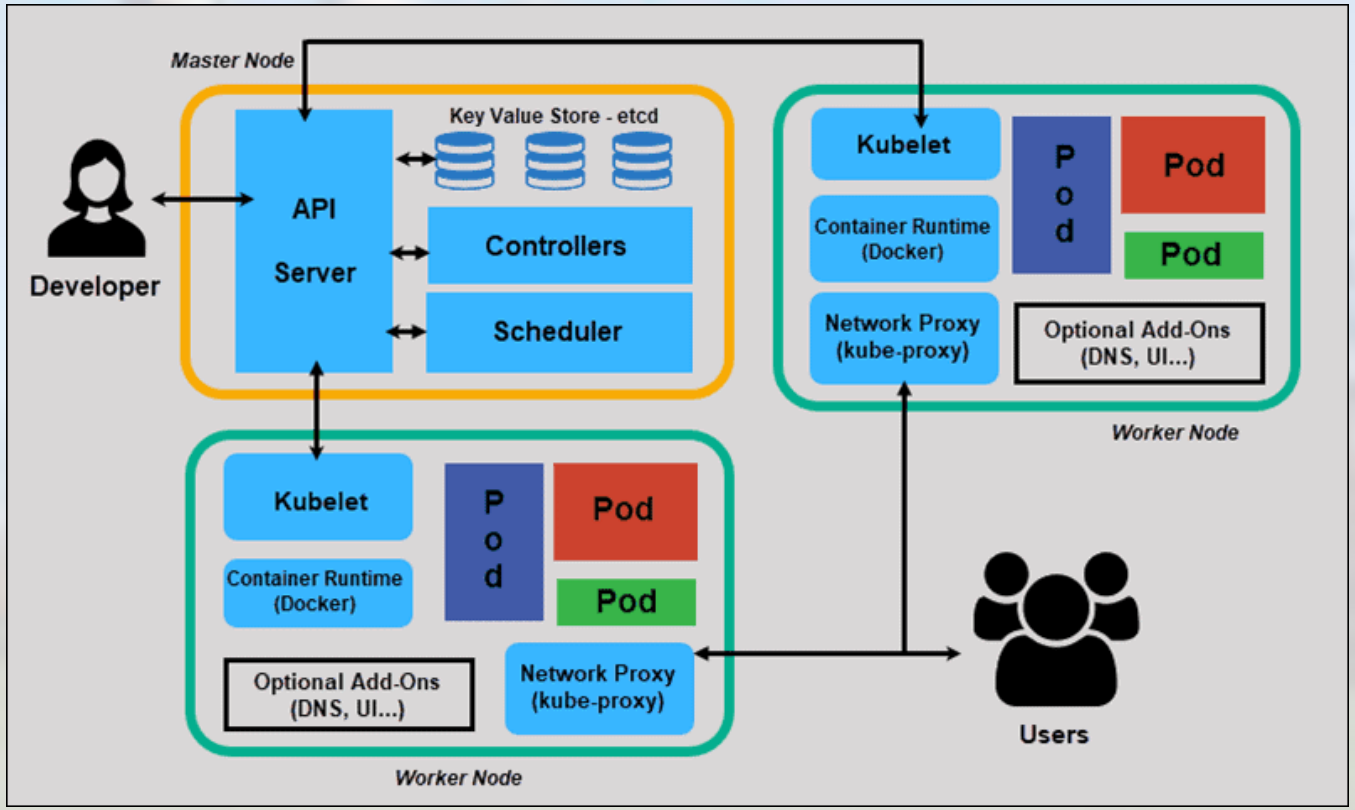




https://www.docker.com/resources/what-container/
https://phoenixnap.com/kb/docker-image-vs-container

Kubernetes

- Kubernetes, also known as K8s, is an open-source system for automating deployment, scaling, and management of containerized applications.[1]
- Kubernetes enables container orchestration and serves as the backbone of the NSF Nautilus HyperCluster



1. https://kubernetes.io/
2. Image: https://phoenixnap.com/kb/understanding-kubernetes-architecture-diagrams
3. Logo: https://commons.wikimedia.org/wiki/File:Kubernetes_logo_without_workmark.svg

Key Kubernetes Concepts

- **Nodes** - a node is a physical machine where containers are deployed. Each node must run a container runtime (in Nautilus, it is 'Containerd').
- **Namespaces** - provide a way for K8s to partition cluster resources across multiple or many users in an exclusive way.
- **Pods** - pods are the basic scheduling unit of K8s. Pods consist of one or more containers running inside. Each pod has a unique IP address to enable micro services or applications
- **Jobs** – long running processes that require more than 6 hours of runtime or have more demanding compute that 2 CPU cores. 8 GB or RAM, and 1 GPU
- **Services** - a set of pods working together

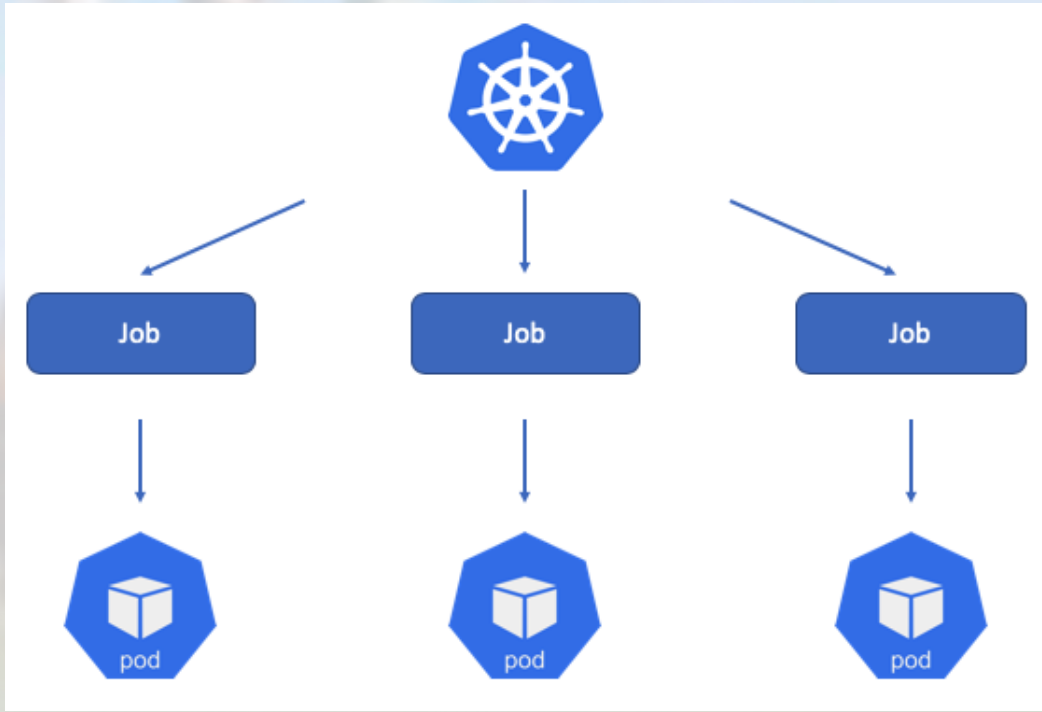
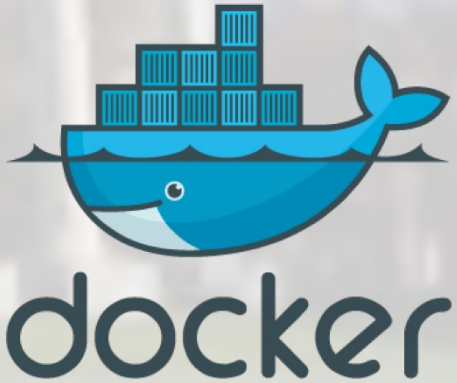


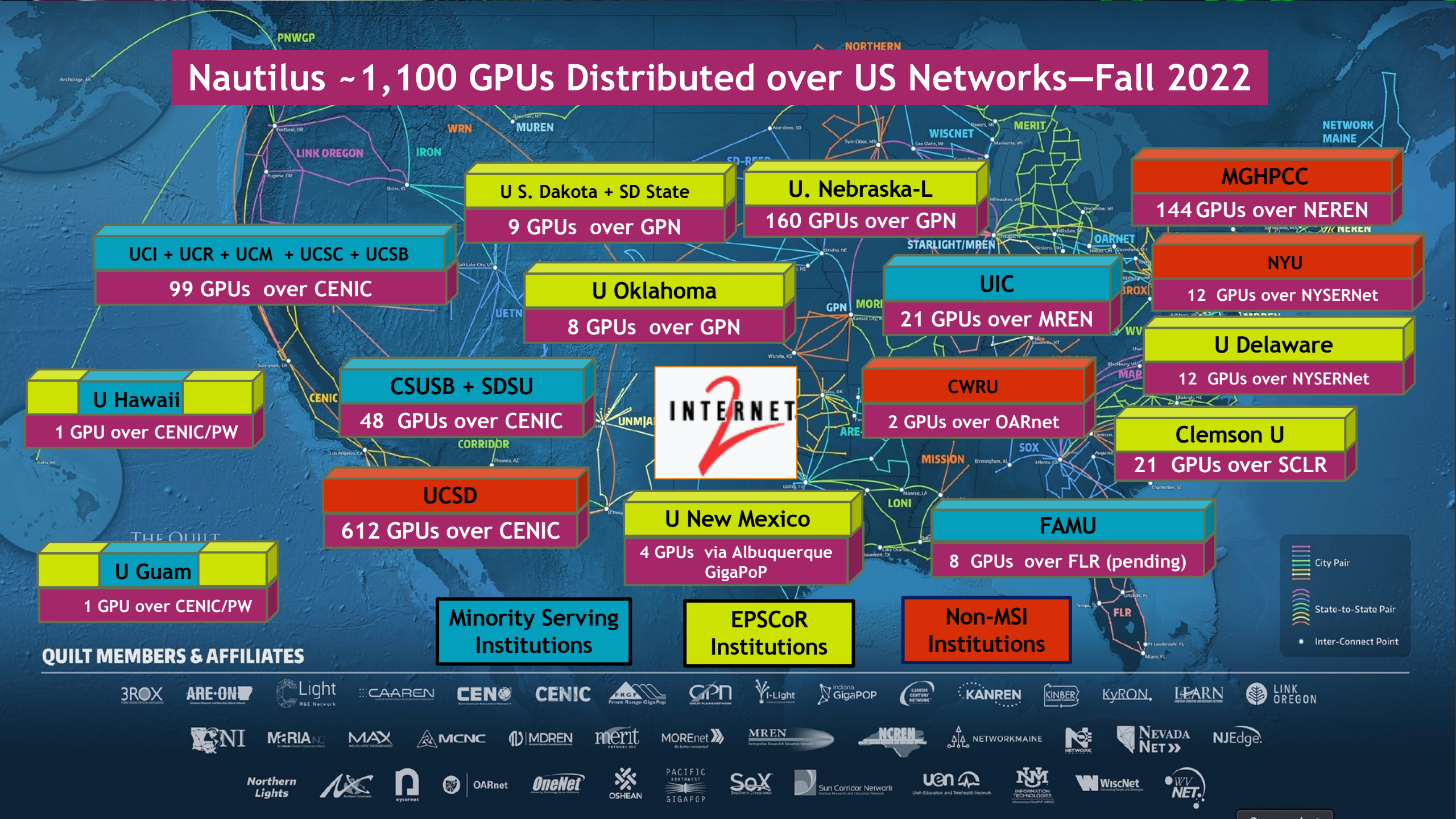
Image: https://aws.plainenglish.io/kubernetes-deep-dive-job-and-cronjob-5ffed1c5fa4e

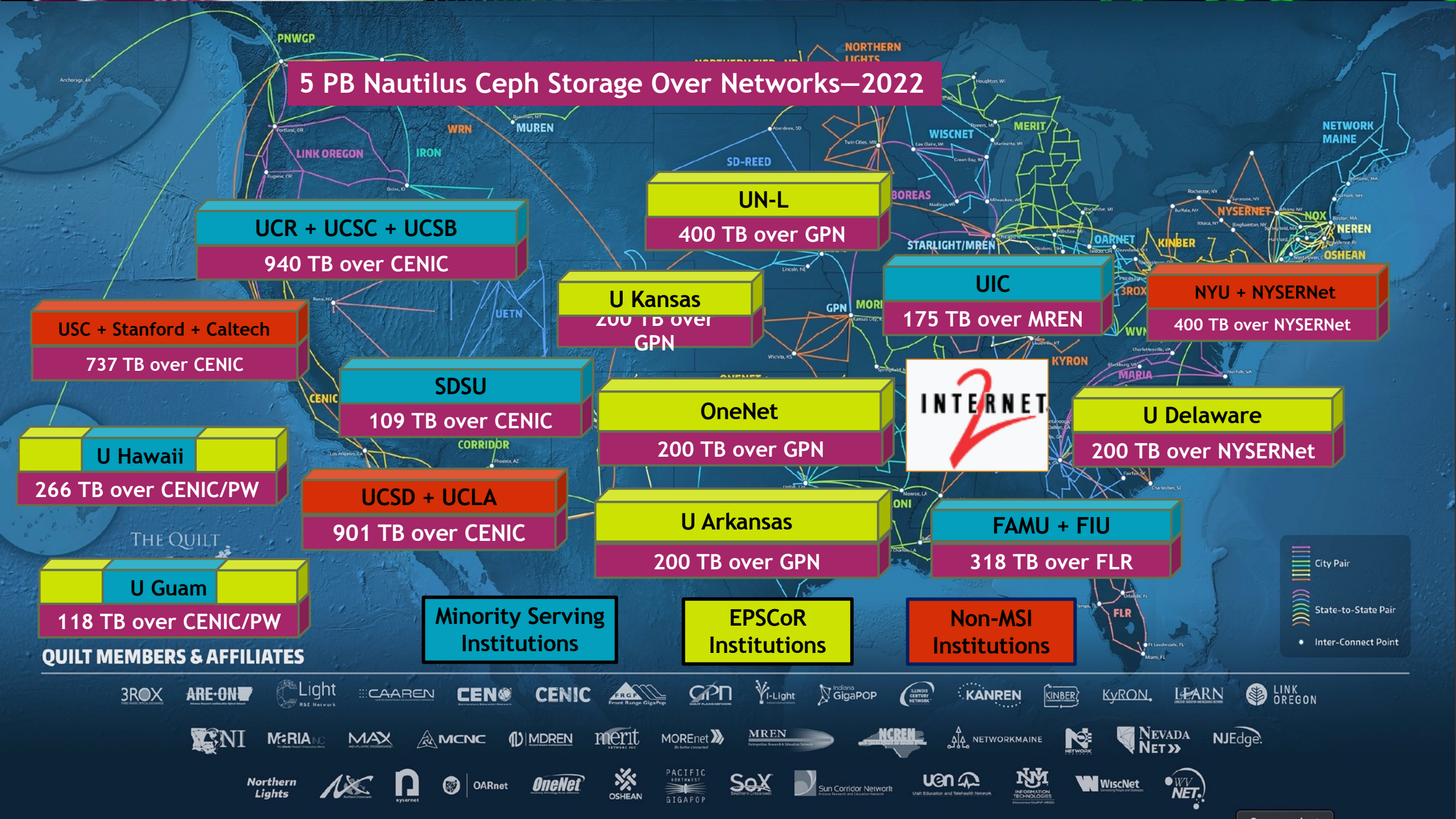
NSF Nautilus HyperCluster

- The NSF Nautilus HyperCluster is a Kubernetes cluster with vast resources that can be utilized for various research purposes:
  - Prototyping research code
  - S3 cloud storage for data and models
  - Accelerated small-scale research compute
  - Scaling research compute for large scale experimentation
- Resources readily available:
  - Nodes: 263
  - Logical CPU Cores: 2,114
  - RAM: 12.47 TB
  - GPUs: 158



Nautilus Logo: https://portal.nrp-nautilus.io/
Docker Logo: https://developers.redhat.com/blog/2014/05/15/practical-introduction-to-docker-containers
Kubernetes Logo: https://commons.wikimedia.org/wiki/File:Kubernetes_logo_without_workmark.svg


Nautilus ~1,100 GPUs Distributed over US Networks—Fall 2022
U S. Dakota + SD State
9 GPUs over GPN
U. Nebraska-L
160 GPUs over GPN
MGHPCC
144 GPUs over NEREN
UCI + UCR + UCM + UCSC + UCSB
99 GPUs over CENIC
U Oklahoma
8 GPUs over GPN
UIC
21 GPUs over MREN
NYU
12 GPUs over NYSERNet
U Delaware
12 GPUs over NYSERNet
U Hawaii
1 GPU over CENIC/PW
CSUSB + SDSU
48 GPUs over CENIC
INTERNET2
CWRU
2 GPUs over OARnet
Clemson U
21 GPUs over SCLR
UCSD
612 GPUs over CENIC
U New Mexico
4 GPUs via Albuquerque GigaPoP
FAMU
8 GPUs over FLR (pending)
U Guam
1 GPU over CENIC/PW
Minority Serving Institutions
EPSCoR Institutions
Non-MSI Institutions
City Pair
State-to-State Pair
Inter-Connect Point
QUILT MEMBERS & AFFILIATES


5 PB Nautilus Ceph Storage Over Networks—2022
UCR + UCSC + UCSB
940 TB over CENIC
USC + Stanford + Caltech
737 TB over CENIC
U Hawaii
266 TB over CENIC/PW
U Guam
118 TB over CENIC/PW
SDSU
109 TB over CENIC
UCSD + UCLA
901 TB over CENIC
UN-L
400 TB over GPN
U Kansas
GPN
OneNet
200 TB over GPN
U Arkansas
200 TB over GPN
UIC
175 TB over MREN
NYU + NYSERNet
400 TB over NYSERNet
U Delaware
200 TB over NYSERNet
FAMU + FIU
318 TB over FLR
INTERNET2
THE QUILT
QUILT MEMBERS & AFFILIATES
Minority Serving Institutions
EPSCoR Institutions
Non-MSI Institutions
City Pair
State-to-State Pair
Inter-Connect Point
PNWGP
NORTHERN LIGHTS
WRN
MUREN
LINK OREGON
IRON
SD-REED
BOREAS
WISCNET
MERIT
NETWORK MAINE
NYSERNET
NOX
NEREN
OSHEAN
STARLIGHT/MREN
OARNET
KINBER
3ROX
UETN
GPN
MORI
WVN
KYRON
MARIA
CENIC
CORRIDOR
ONI
FLR

Container Orchestration in Nautilus

1. Users push code to VCS
   - https://gitlab.nrp-nautilus.io/
2. GitLab CI/CD build Docker image and push to Container Registry
3. Users create pod/job using published Docker image
   - KubeCTL Command Line Tool
4. Kubernetes schedules the pod/job onto a node with required resources

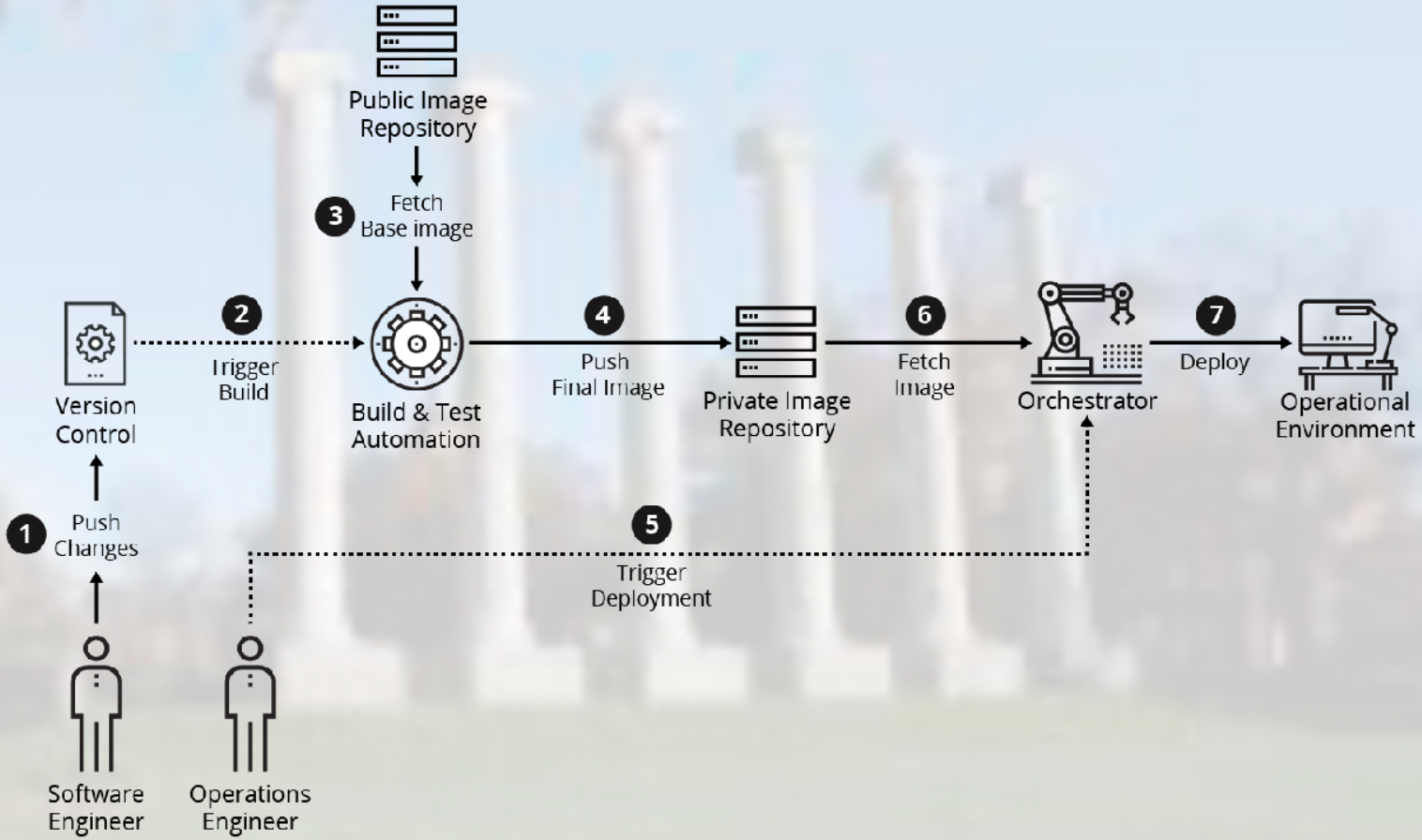


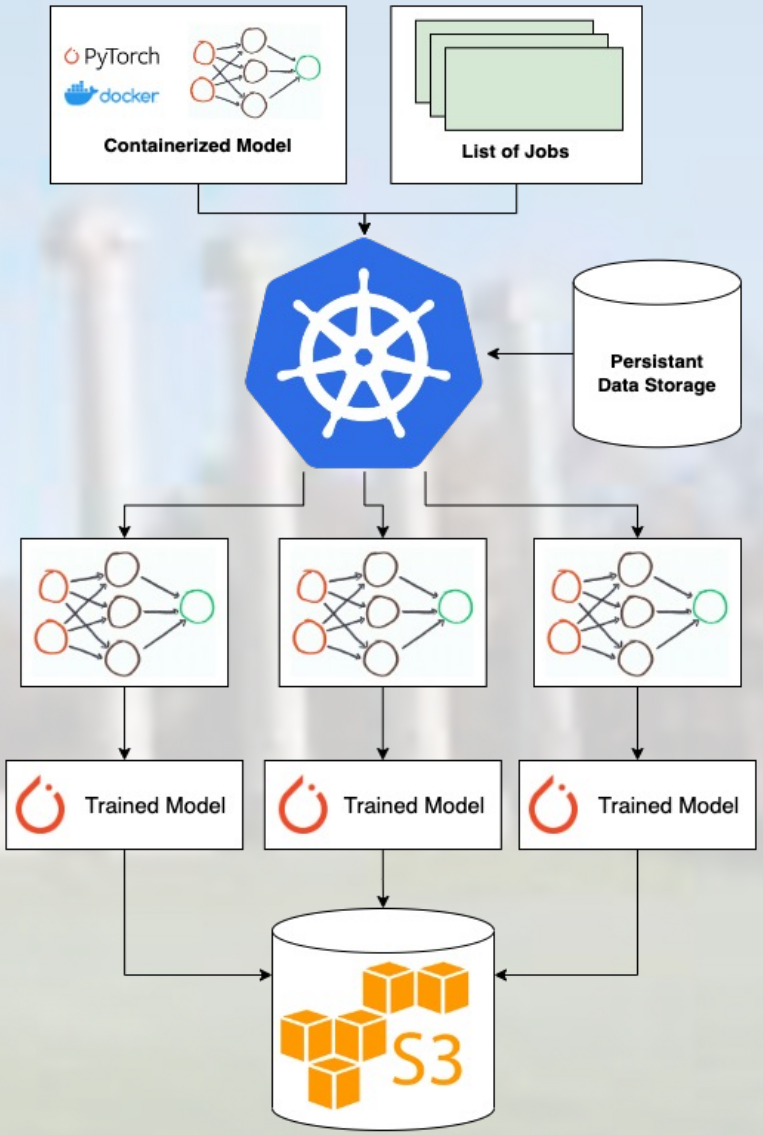
Figure 4: Model Container Supply Chain

Image: https://insights.sei.cmu.edu/blog/7-quick-steps-to-using-containers-securely/

Case Study System Overview:

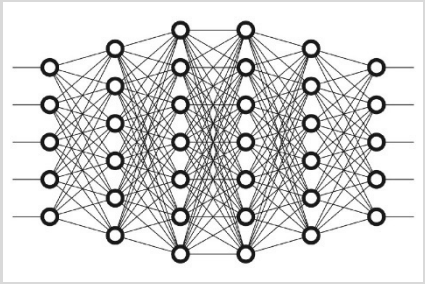
Nautilus for Accelerated Deep Learning Research

- Using containerized model definition and list of jobs
- Mounted persistent data storage to each pod
- Each GPU job produces an associated trained model
- Automation currently performed via environment variables and bash, but more sophisticated methods in development
- Models are later sync'd to Nautilus S3 bucket for later use in evaluation or other ML applications



Deep Learning on Nautilus: By the Numbers

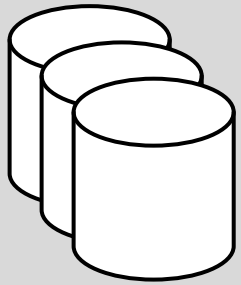
Nautilus



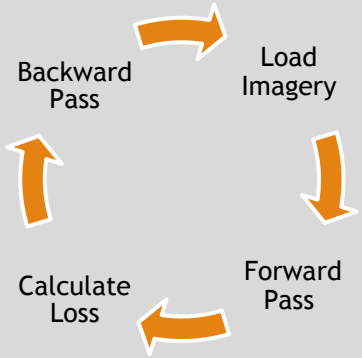
9 Deep Neural Architectures

27 Trained Models

124.74 TB Imagery Processed



3 Open Source HR-RSI Datasets



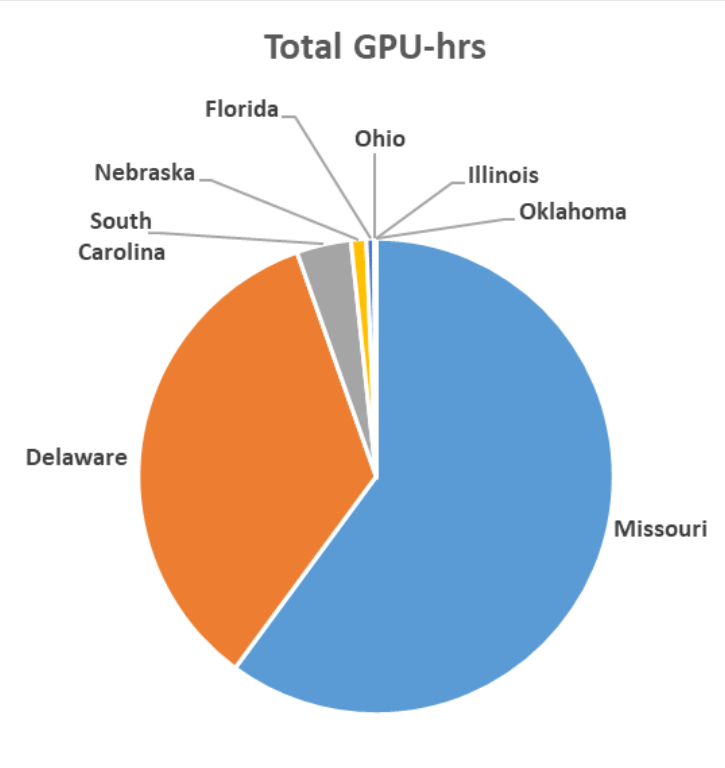
- 8,100 Epochs
- 30M iterations
- 1.7B parameters

Wall Clock Time: 76 days, 10 hours

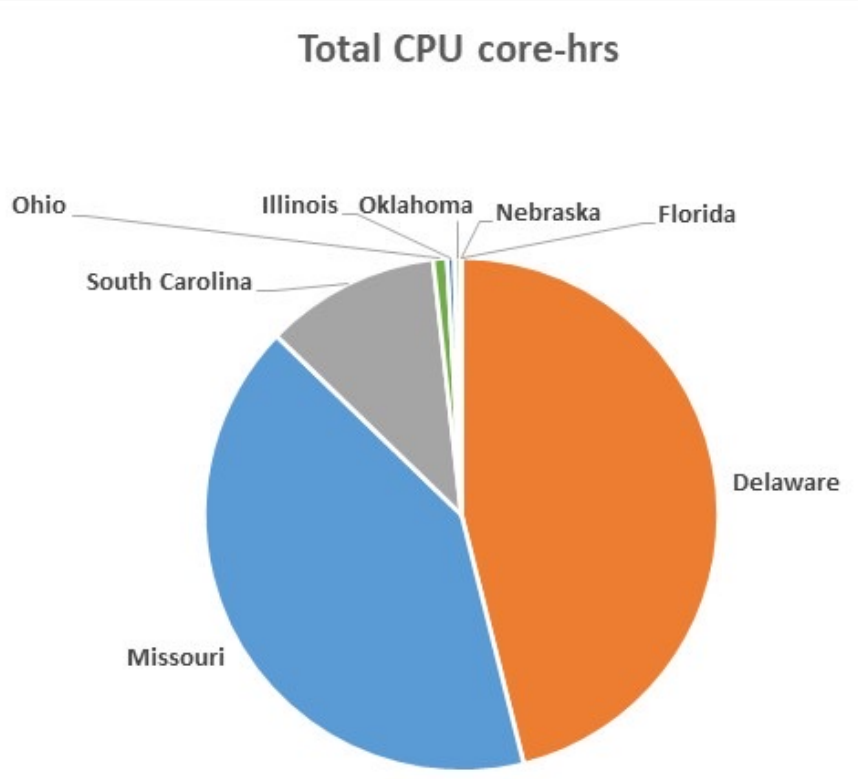
https://www.freecodecamp.org/news/want-to-know-how-deep-learning-works-heres-a-quick-guide-for-everyone-1aedeca88076/
https://commons.wikimedia.org/wiki/File:PyTorch_logo_icon.svg
https://www.clipartmax.com/download/m2K9A0m2b1m2i8i8_4-o-clock-clip-art-clock-face-clip-art/

Non-California Nautilus PI Namespace 2021 Usage by State: "Big MO!"

Data/Plots provided by Larry Smarr (PI, National Research Platform & father of US Super Computing Centers)



17,217 GPU-hrs



28,088 CPU core-hrs

University of Missouri

Grant Scott, UMC
Helped Organize the UMC PRP Usage