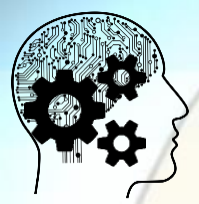


NVIDIA GPU Cloud (NGC)

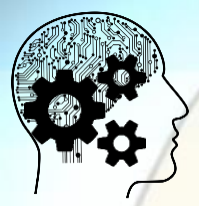
Dr. Sarwan Singh



Agenda

- NVIDIA GPU Cloud (NGC)
- Docker
- Containers
- Virtualization
- Virtual Machines Vs Docker





Grand Challenges Require Massive Computing



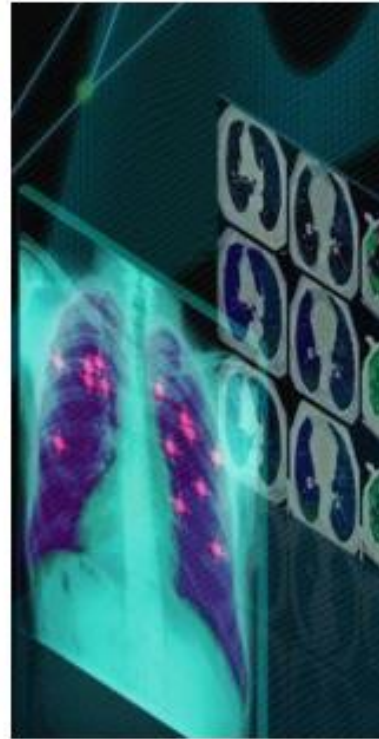
AUTONOMOUS DRIVING



ASTROPHYSICS



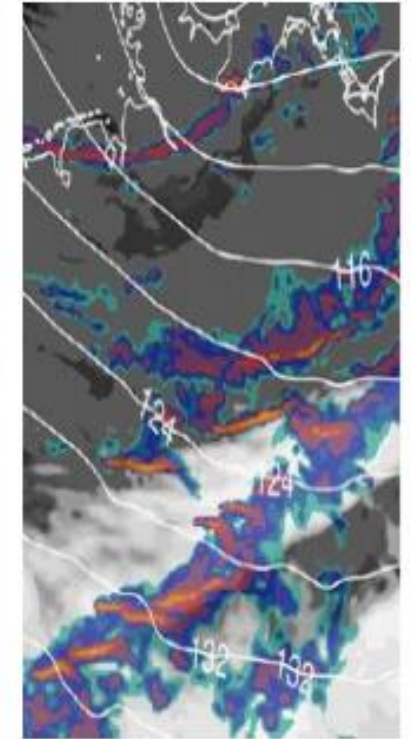
GENOMICS



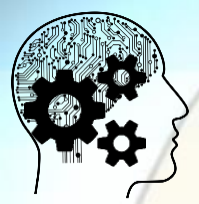
MEDICAL IMAGING



NUCLEAR FUSION



WEATHER



Challenges Utilizing AI & HPC Software

EXPERTISE



Building AI-centric solutions requires expertise

INSTALLATION



Complex, time consuming, and error-prone

OPTIMIZATION



Requires expertise to optimize framework performance

PRODUCTIVITY

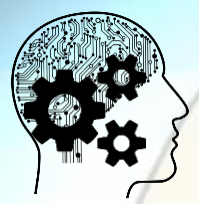


Users limited to older features and lower performance

MAINTAINENCE



IT can't keep up with frequent software upgrades



NGC – Simplifying AI & HPC Workflows

EMBEDDING EXPERTISE



Deliver greater value,
faster

FASTER DEPLOYMENTS



Eliminates installations.
Simply Pull & Run the
app

OPTIMIZED SOFTWARE



Key DL frameworks
updated monthly for perf
optimization

HIGHER PRODUCTIVITY



Better Insights and faster
time-to-solution

ZERO MAINTENANCE



Empowers users to
deploy the latest versions
with IT support

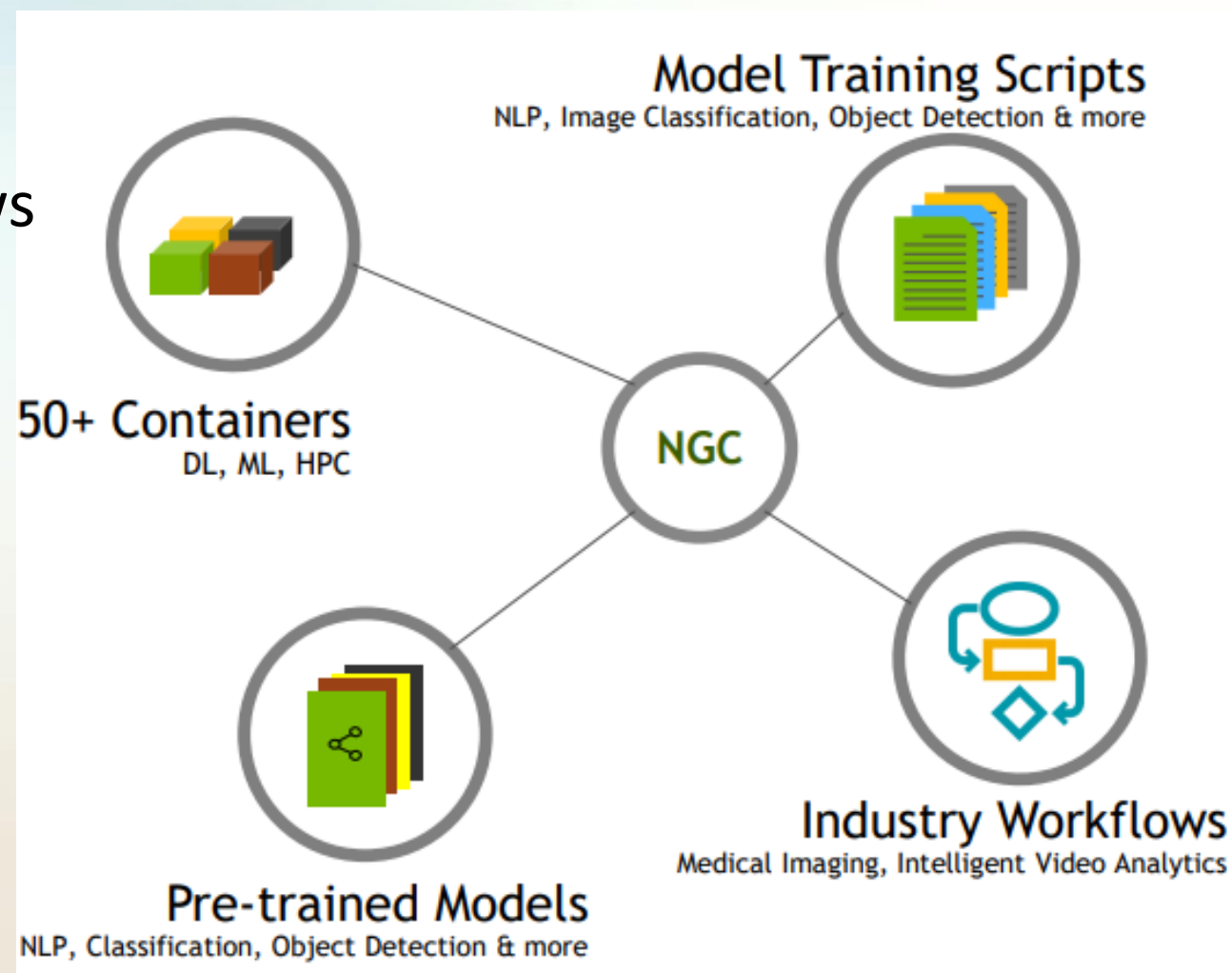


The new NGC

GPU-optimized Software Hub.

Simplifying DL, ML & HPC Workflows

- ✓ Simplify Deployments
- ✓ Innovate Faster
- ✓ Deploy anywhere





Terminology

Docker

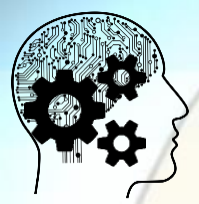
an open source tool that enables developers to deploy an application inside an isolated sandbox (known as a container), that is run on the host operating system

Images

an executable package that includes everything needed to run an application, e.g. the code, dependencies, runtime, configuration files, environment variables, etc. It is the blueprint that forms the basis of a container.

Container

a runtime instance of an image. The act of running a Docker image creates a Docker container. The container runs the actual application that was packaged inside it.



Container

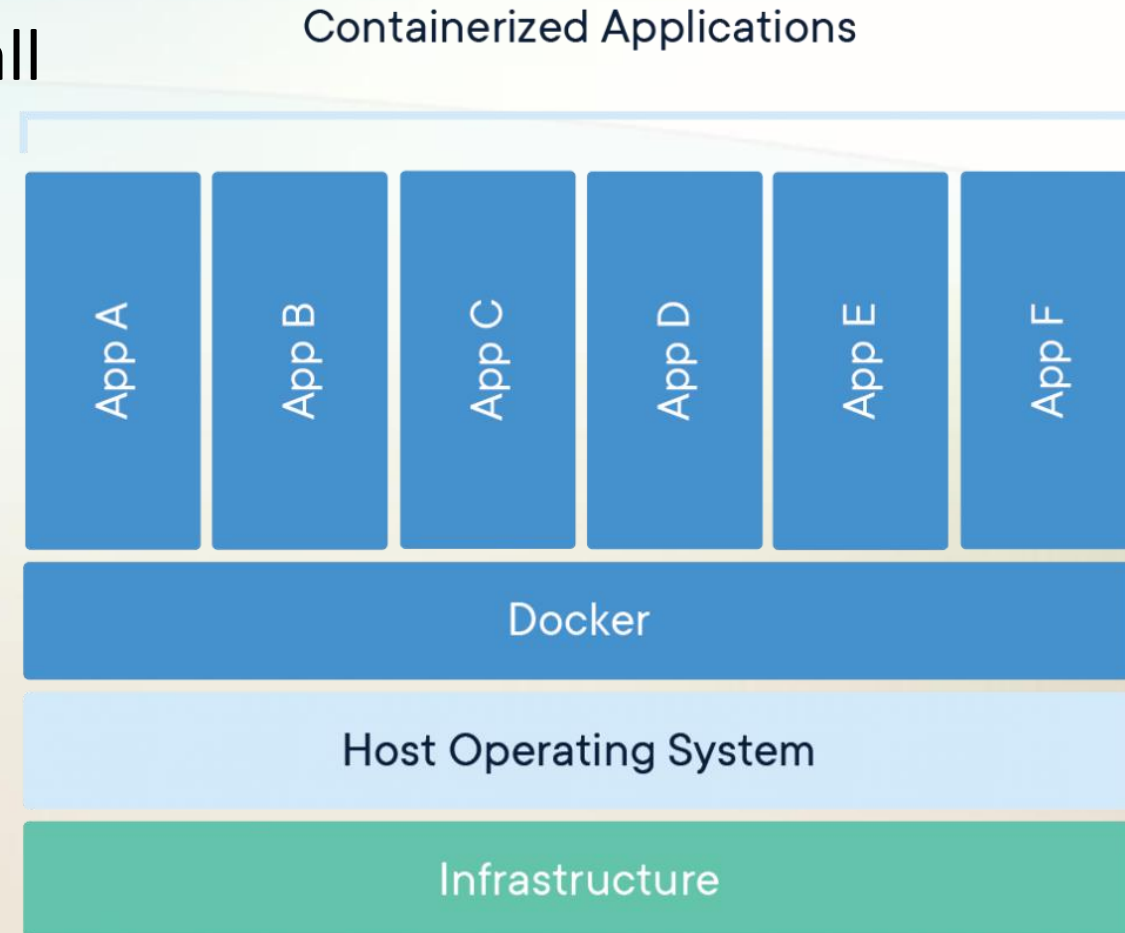
A standardized unit of software

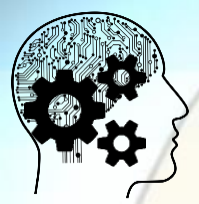
“Package Software into
Standardized Units for
Development,
Shipment and
Deployment”



Container

- A container is a standard unit of software that packages up code and all its dependencies so the application runs quickly and reliably from one computing environment to another.
- A Docker container image is a lightweight, standalone, executable package of software that includes everything needed to run an application: **code, runtime, system tools, system libraries and settings.**



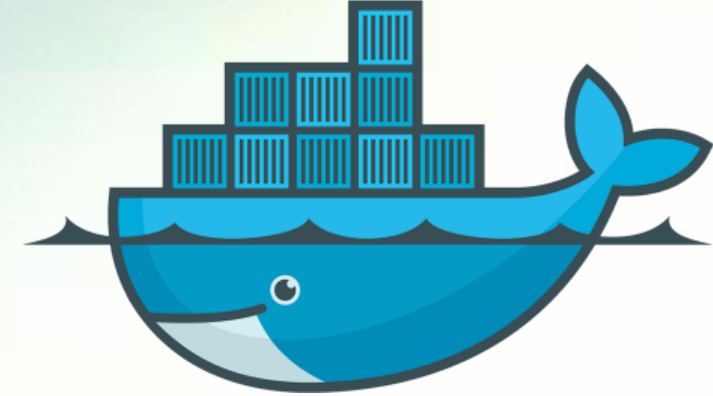


Container

- Containers offer a logical packaging mechanism in which applications can be abstracted from the environment in which they actually run.
- This decoupling allows container-based applications to be deployed easily and consistently, regardless of whether the target environment is a private data center, the public cloud, or even a developer's personal laptop.
- Containerization provides a clean separation of concerns, as developers focus on their application logic and dependencies, while IT operations teams can focus on deployment and management without bothering with application details such as specific software versions and configurations specific to the app.



Docker

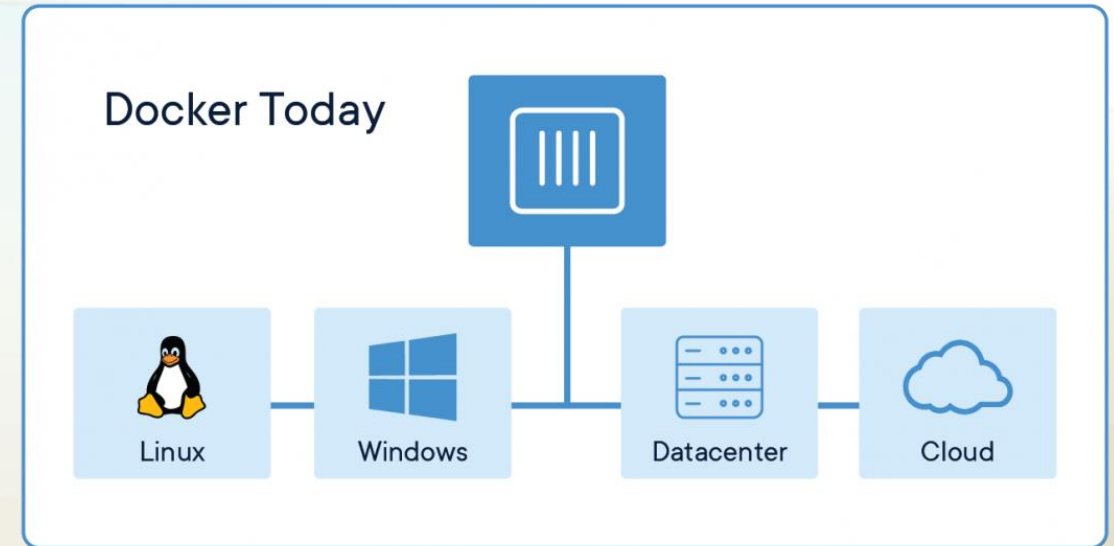


- Docker container technology was launched in 2013 as an open source Docker Engine.
- Docker Engine is the industry's de facto container runtime that runs on various Linux (CentOS, Debian, Fedora, Oracle Linux, RHEL, SUSE, and Ubuntu) and Windows Server operating systems.
- Docker creates simple tooling and a universal packaging approach that bundles up all application dependencies inside a container which is then run on Docker Engine.
- Docker Engine enables containerized applications to run anywhere consistently on any infrastructure, solving “**dependency hell**” for developers and operations teams, and eliminating the “**it works on my laptop!**” problem.



Docker

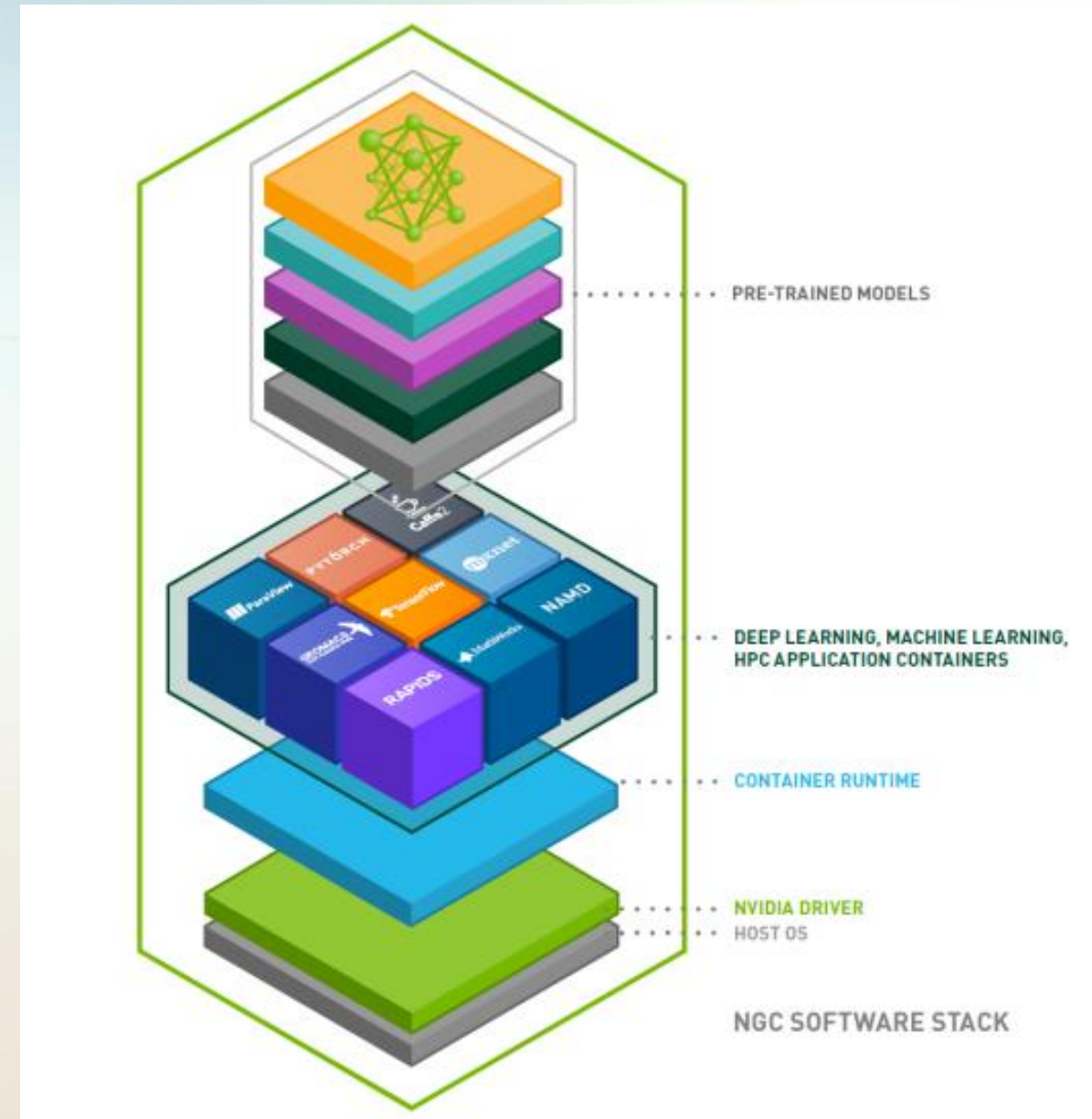
- Success in the Linux world drove a partnership with Microsoft that brought Docker containers and its functionality to Windows Server (sometimes referred to as **Docker Windows containers**).

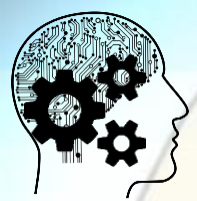




Why Containers

- **Simplifies Deployments** - Eliminates complex, time-consuming builds and installs
- **Get started in minutes** – Simply Pull & Run the app
- **Portable** - Deploy across various environments, from test to production with minimal changes





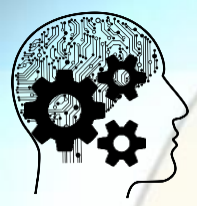
NGC Containers: Accelerating Workflows

WHY CONTAINERS

- **Simplifies Deployments** - Eliminates complex, time-consuming builds and installs
- **Get started in minutes** - Simply Pull & Run the app
- **Portable** - Deploy across various environments, from test to production with minimal changes

WHY **NGC** CONTAINERS

- **Optimized for Performance** - Monthly DL container releases offer latest features and superior performance on NVIDIA GPUs
- **Scalable Performance** - Supports multi-GPU & multi-node systems for scale-up & scale-out environments
- **Designed for Enterprise & HPC environments** - Supports Docker & Singularity runtimes
- **Run Anywhere** - Pascal/Volta/Turing-powered NVIDIA DGX, PCs, workstations, servers and top cloud platforms



GPU-Optimized Software Containers

Over 50 Containers on NGC



DEEP LEARNING

TensorFlow | PyTorch | more



MACHINE LEARNING

RAPIDS | H2O | more



INFERENCE

TensorRT | DeepStream | more



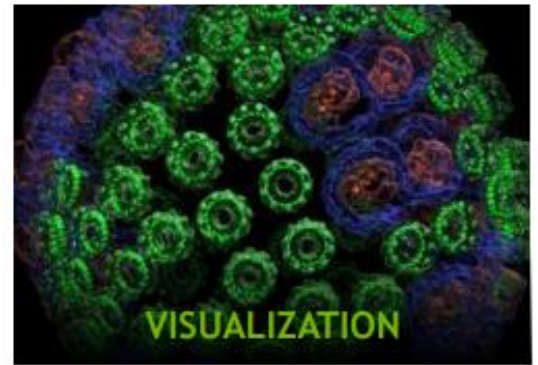
HPC

NAMD | GROMACS | more



GENOMICS

Parabricks

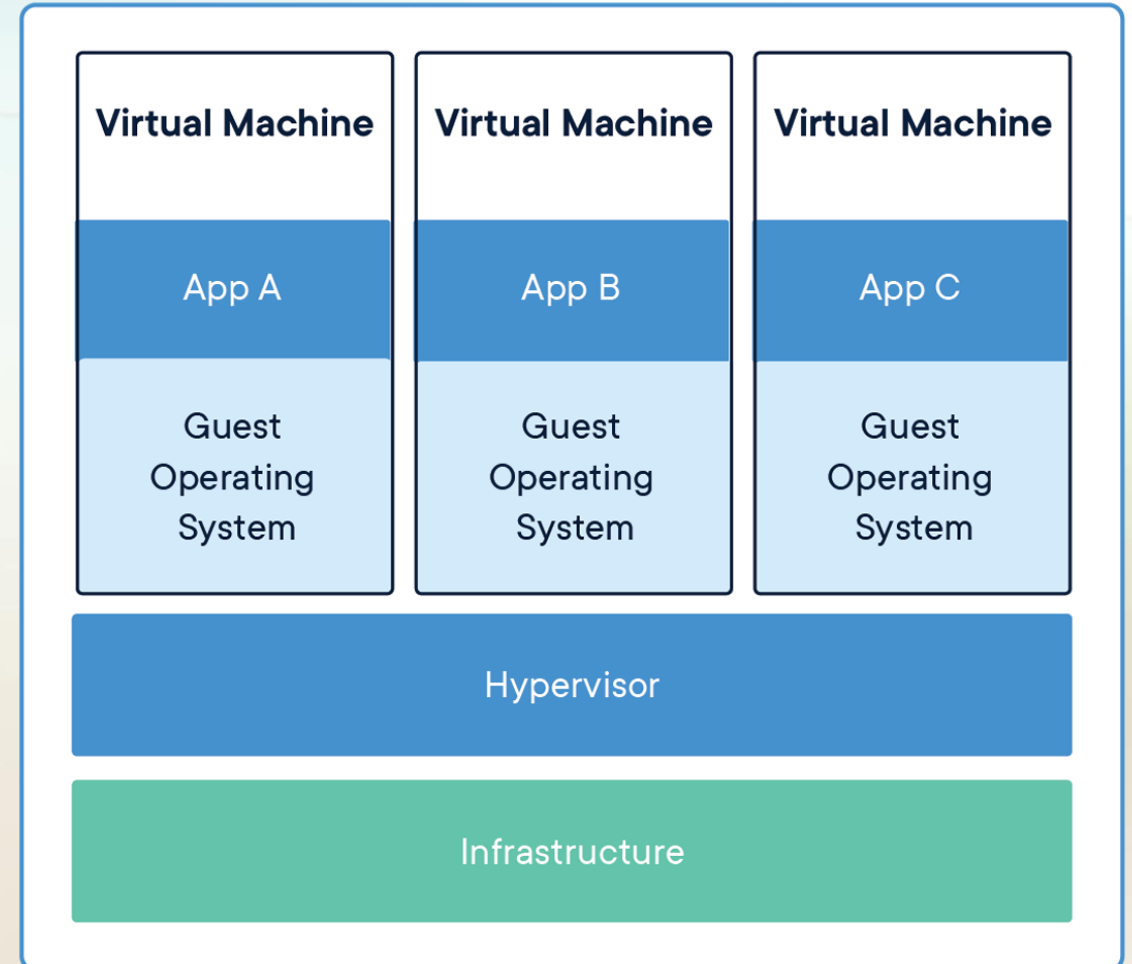
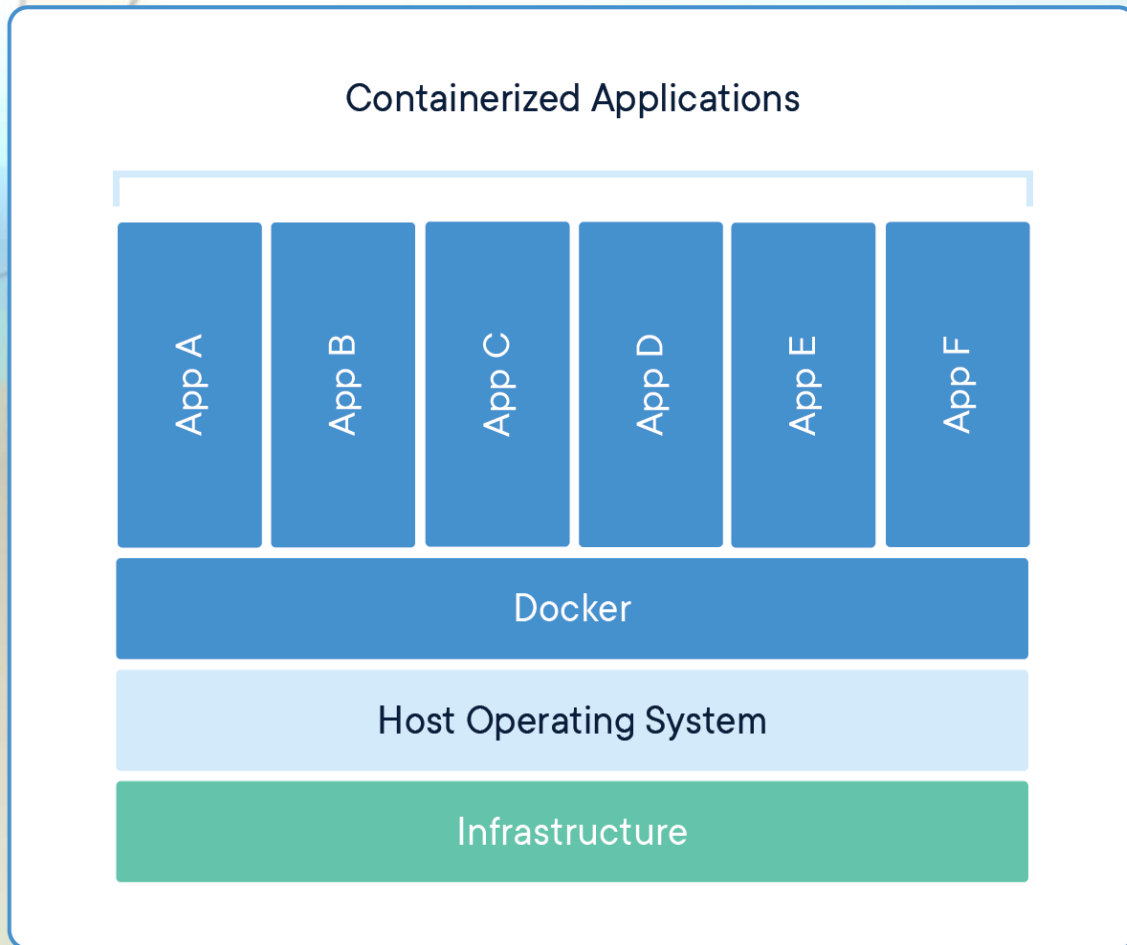


VISUALIZATION

ParaView | IndeX | more



Containers vs Virtual Machines





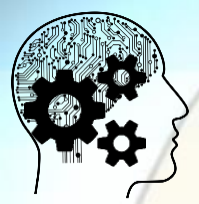
Containers vs Virtual Machines

CONTAINERS

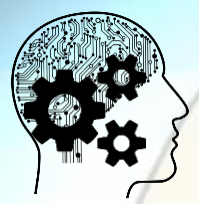
- Containers are an abstraction at the app layer that packages code and dependencies together.
- Multiple containers can run on the same machine and share the OS kernel with other containers, each running as isolated processes in user space.
- Containers take up less space than VMs (container images are typically tens of MBs in size), can handle more applications and require fewer VMs and Operating systems.

VIRTUAL MACHINES

- Virtual machines (VMs) are an abstraction of physical hardware turning one server into many servers.
- The hypervisor allows multiple VMs to run on a single machine. Each VM includes a full copy of an operating system, the application, necessary binaries and libraries - taking up tens of GBs.
- VMs can also be slow to boot.



	Container Benefits	Virtual Machine Benefits
Consistent Runtime Environment	✓	✓
Application Sandboxing	✓	✓
Small Size on Disk	✓	
Low Overhead	✓	



The Google way...

- From Gmail to YouTube to Search, everything at Google runs in containers. Containerization allows our development teams to move fast, deploy software efficiently, and operate at an unprecedented scale. Each week, we start over several billion containers.



