



HCS
C O M P A N Y

CONTAINER & CLOUD-NATIVE ROADSHOW

OPERATIONS TRACK

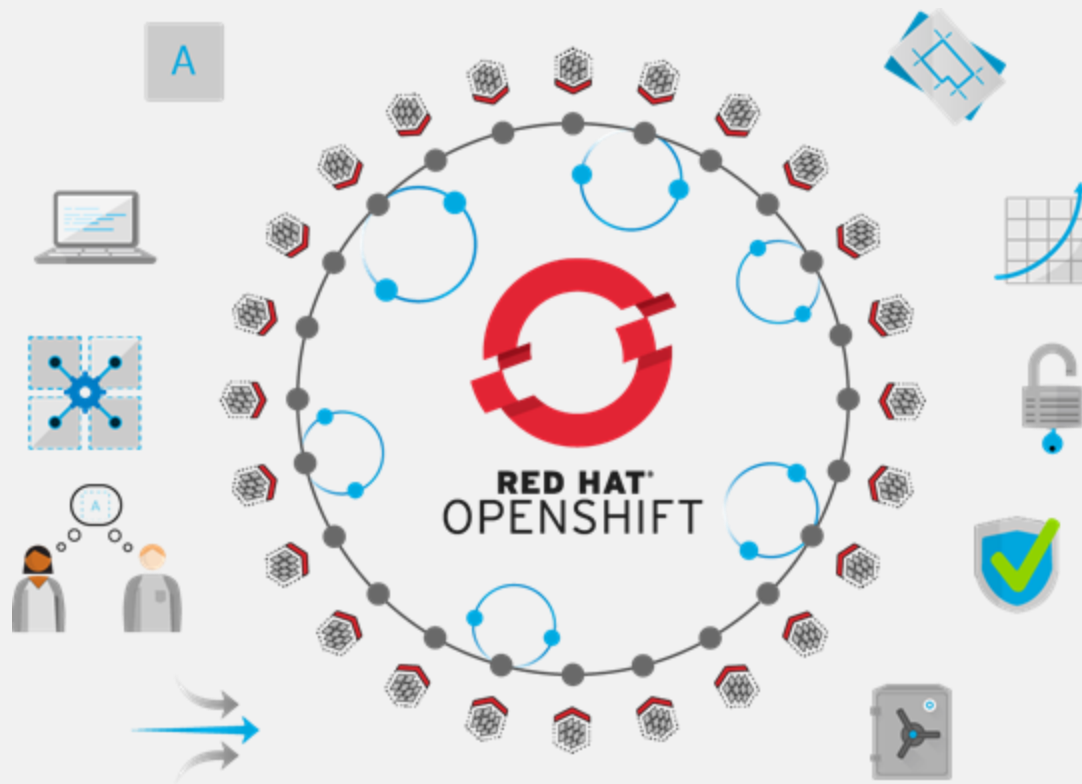




Instructors : Marcel, Kees and Jan.

Agenda

- Introductions – Marcel
- Linux Containers – Marcel
- OpenShift Architecture – Kees
- Container Native Storage / Ceph Architecture – Jan
- Labs!







HCS
C O M P A N Y

LINUX CONTAINERS



WHAT ARE CONTAINERS?

It Depends Who You Ask

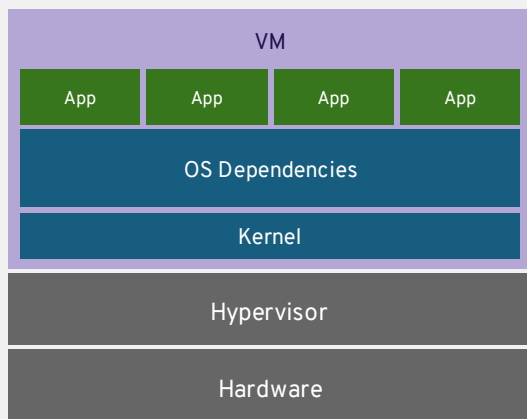
INFRASTRUCTURE

APPLICATIONS

- Application processes on a shared kernel
- Simpler, lighter, and denser than VMs
- Portable across different environments
- Package apps with all dependencies
- Deploy to any environment in seconds
- Easily accessed and shared

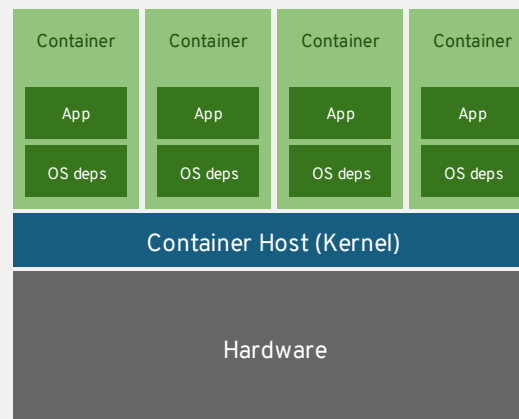
VIRTUAL MACHINES AND CONTAINERS

VIRTUAL MACHINES



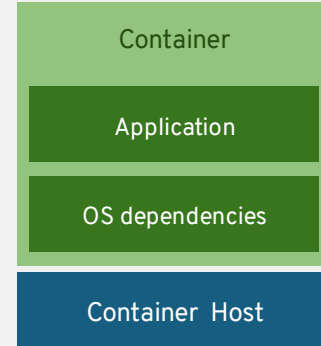
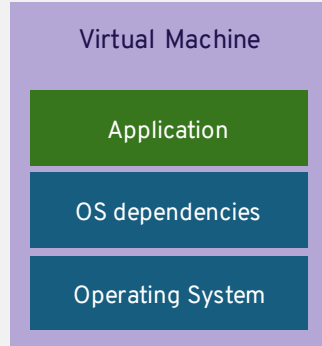
virtual machines are isolated
apps are not

CONTAINERS



containers are isolated
so are the apps

VIRTUAL MACHINES AND CONTAINERS

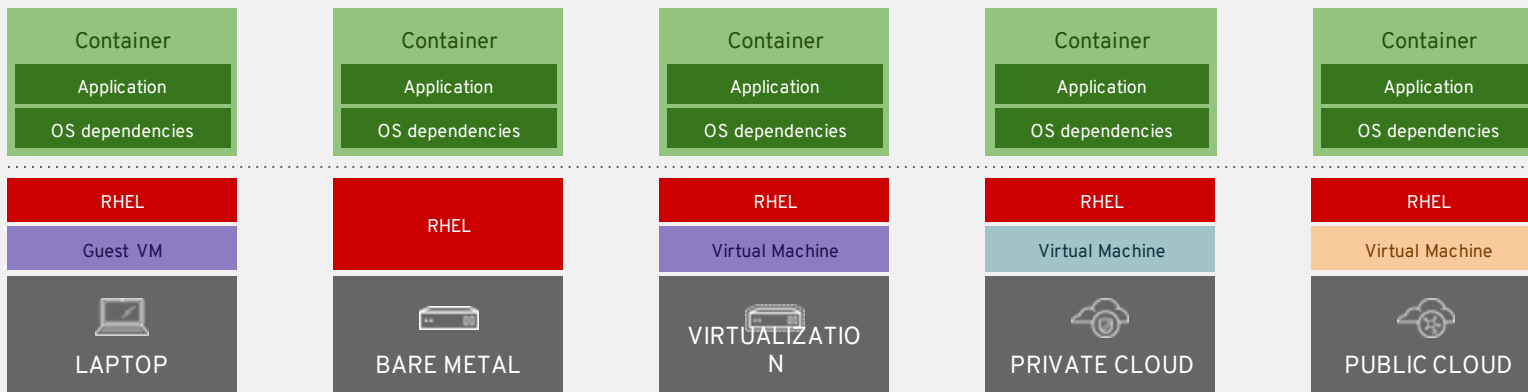


- + VM Isolation
- Complete OS
- Static Compute
- Static Memory
- High Resource Usage

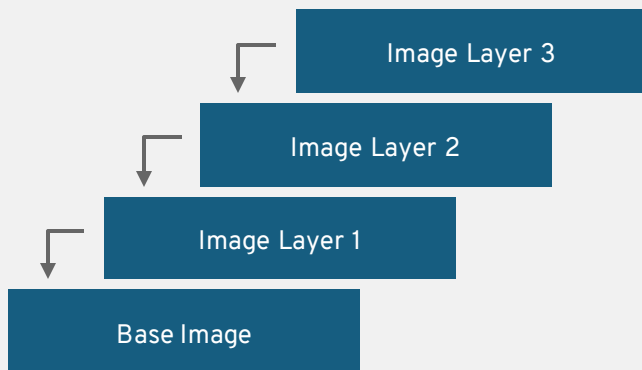
- + Container Isolation
- + Shared Kernel
- + Burstable Compute
- + Burstable Memory
- + Low Resource Usage

APPLICATION PORTABILITY WITH CONTAINERS

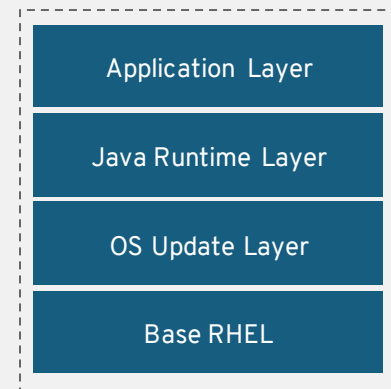
RHEL Containers + RHEL Host = Guaranteed Portability
Across Any Infrastructure



RAPID SECURITY PATCHING USING CONTAINER IMAGE LAYERING



Container Image Layers



Example Container Image



cri-o

A lightweight, OCI-compliant container runtime

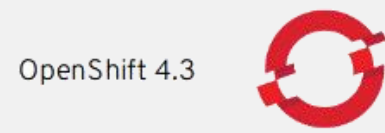
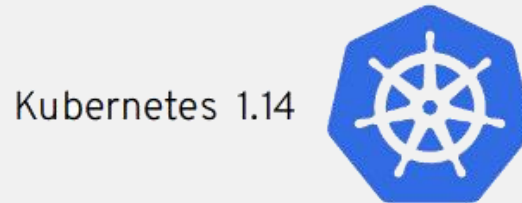
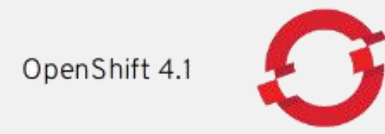
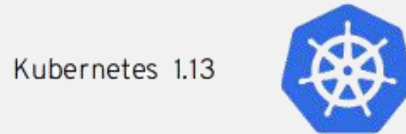
Optimized for
Kubernetes

Any OCI-compliant
container from any
OCI registry
(including docker)

Improve Security and
Performance at scale

CRI-O Support in OpenShift

CRI-O tracks and versions identical to Kubernetes, simplifying support permutations





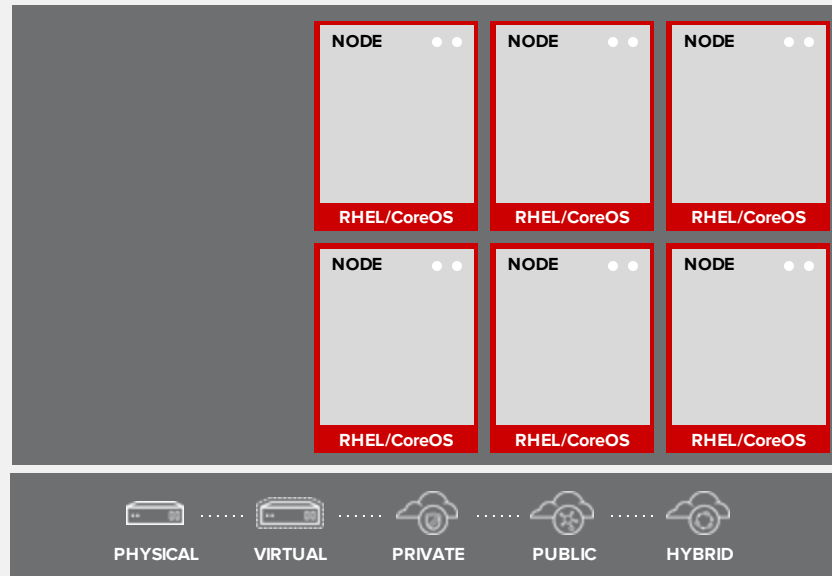
HCS
C O M P A N Y

OPENSIFT ARCHITECTURE

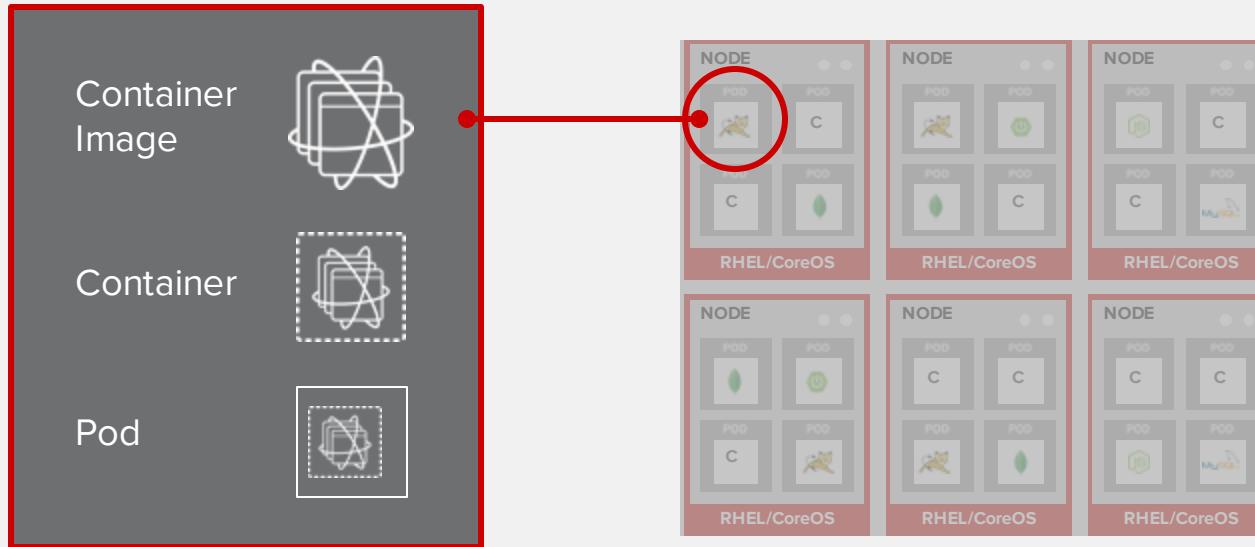
YOUR CHOICE OF INFRASTRUCTURE



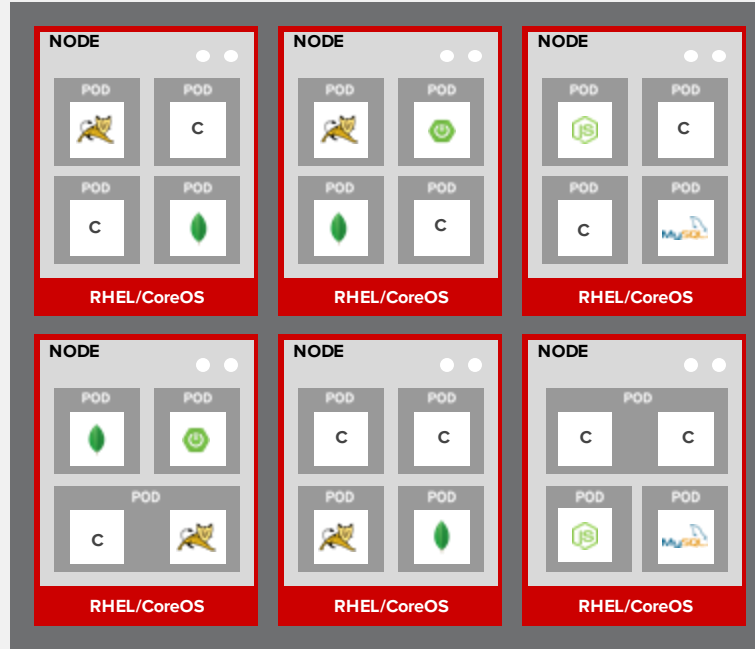
NODES RHEL INSTANCES WHERE APPS RUN



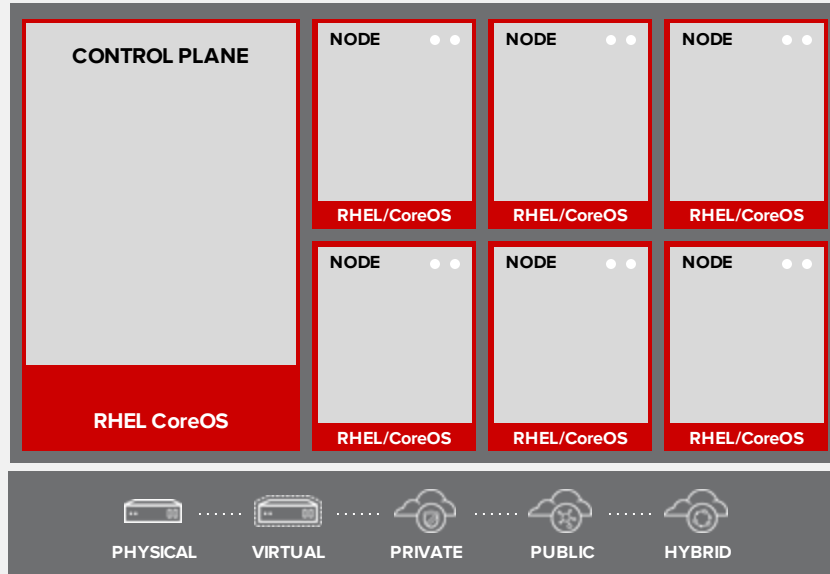
APPS RUN IN CONTAINERS



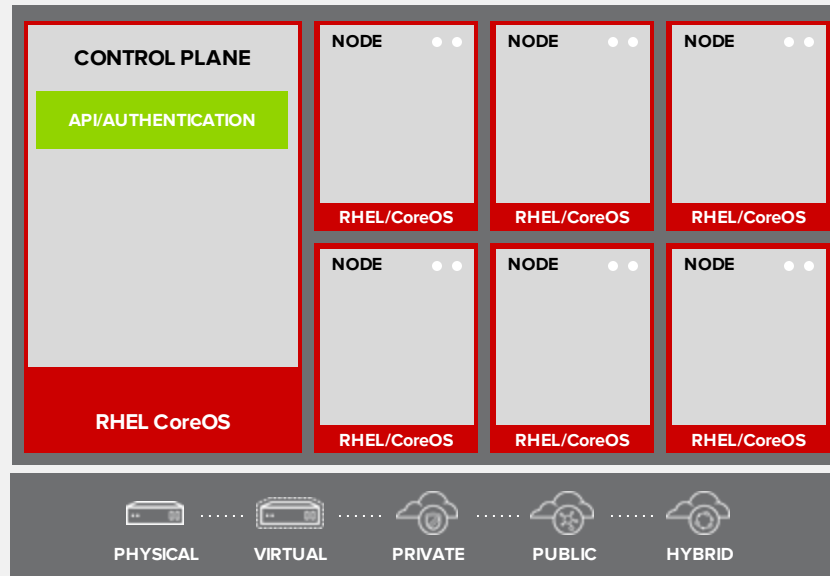
PODS ARE THE UNIT OF ORCHESTRATION



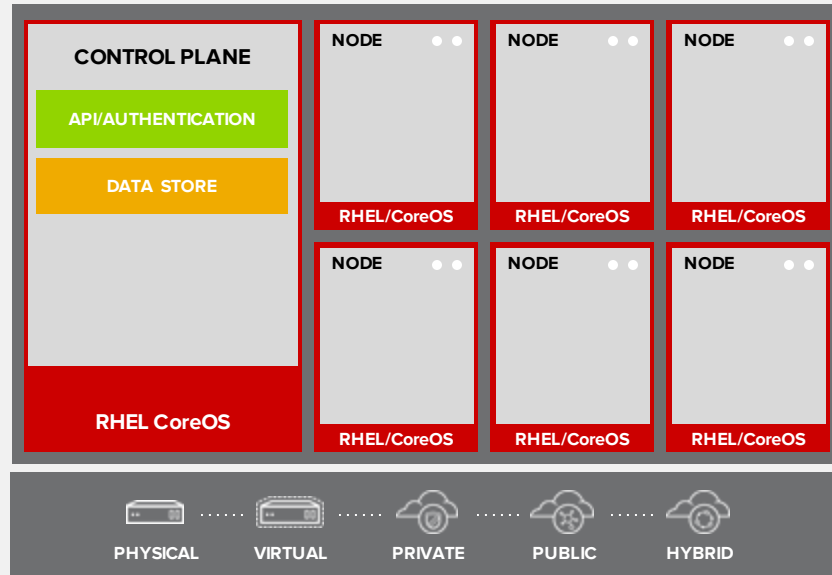
THE CONTROL PLANE



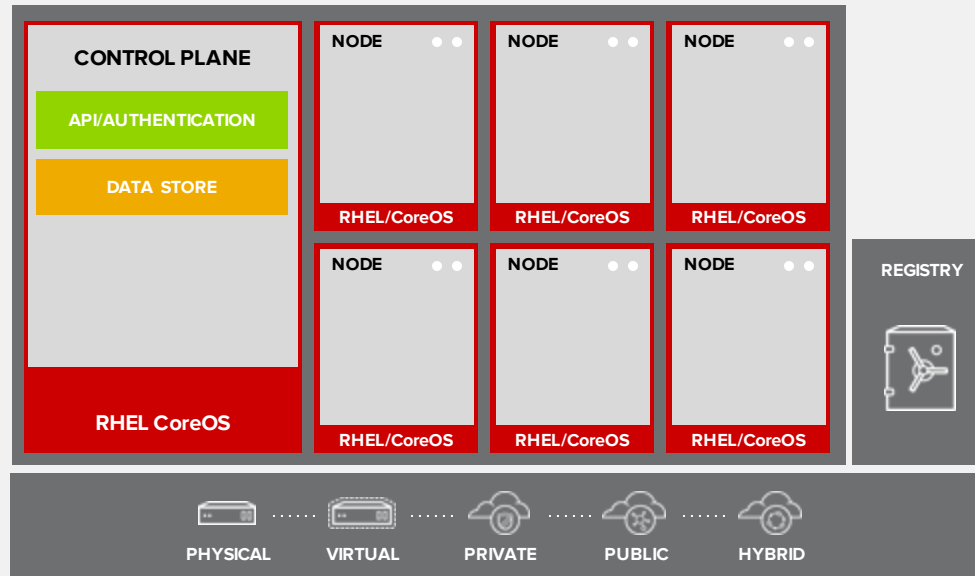
API AND AUTHENTICATION



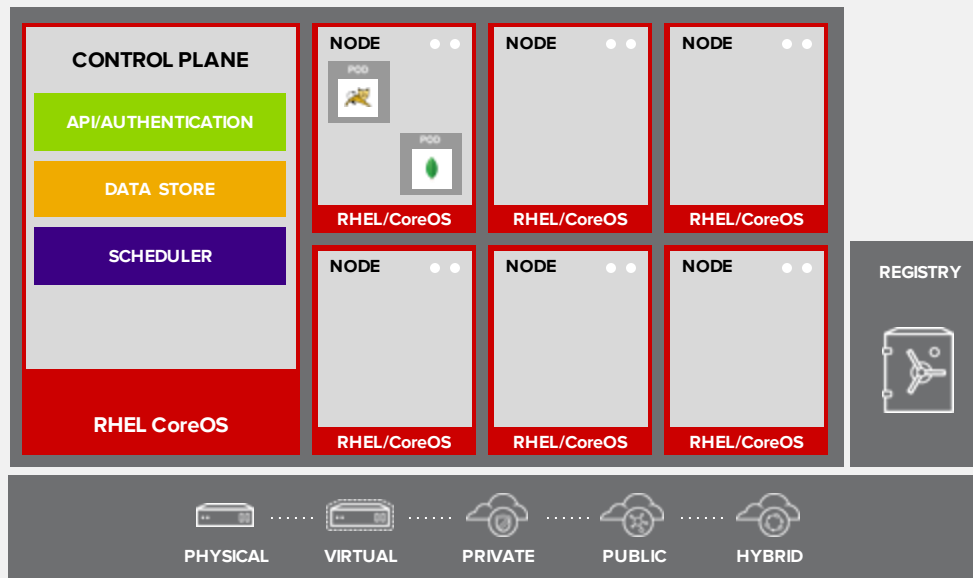
DESIRED AND CURRENT STATE



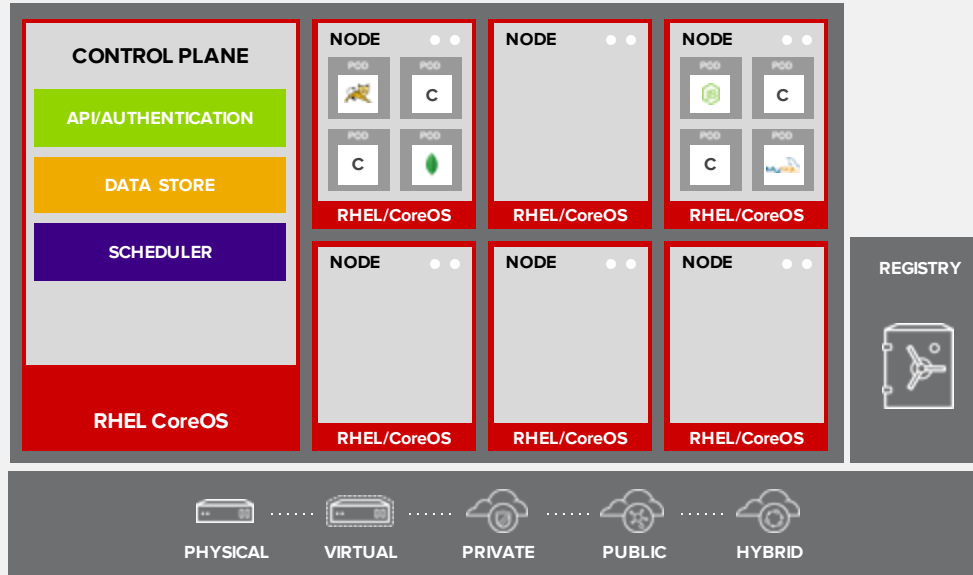
INTEGRATED CONTAINER REGISTRY



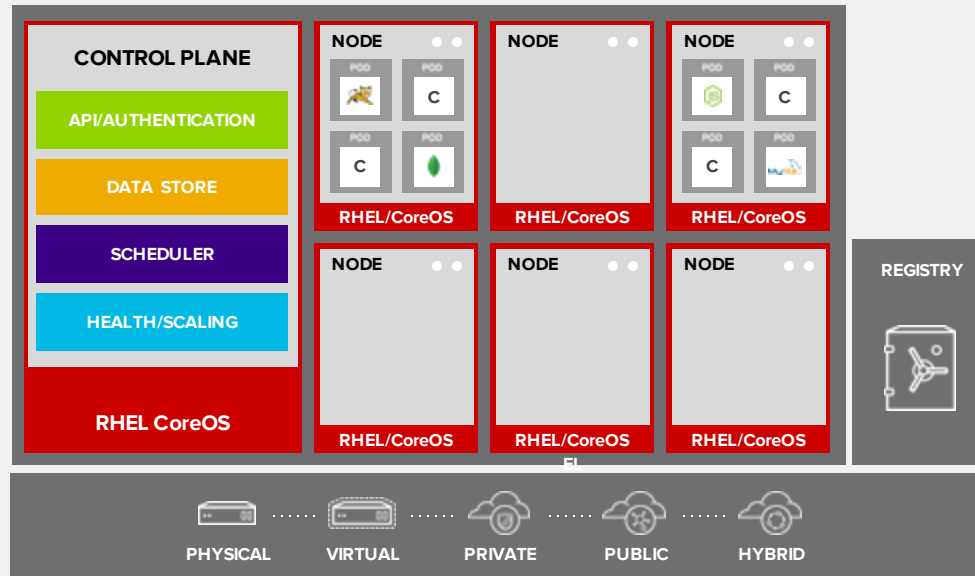
ORCHESTRATION AND SCHEDULING



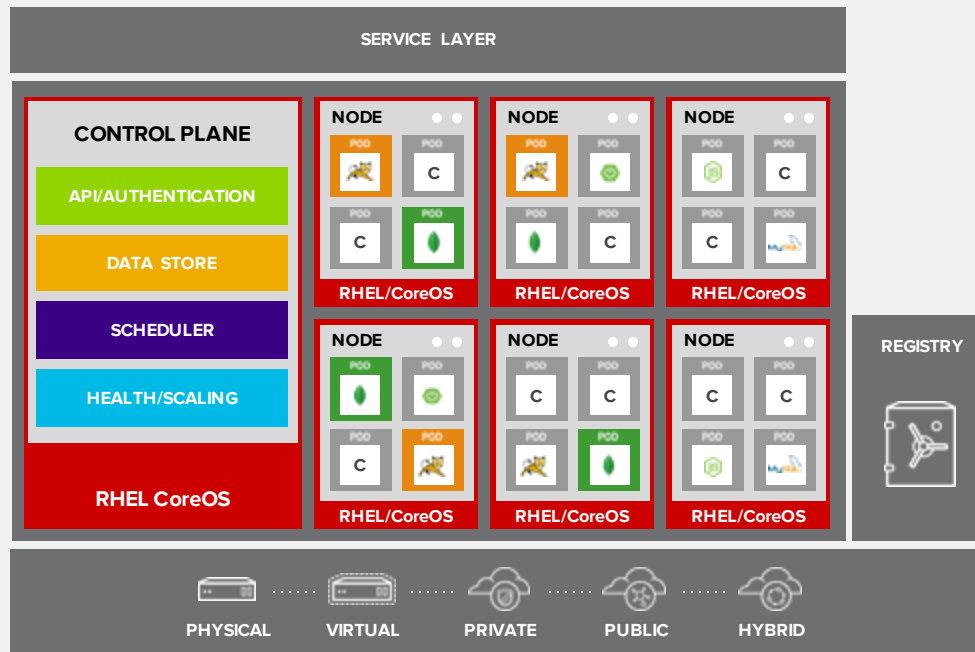
PLACEMENT BY POLICY



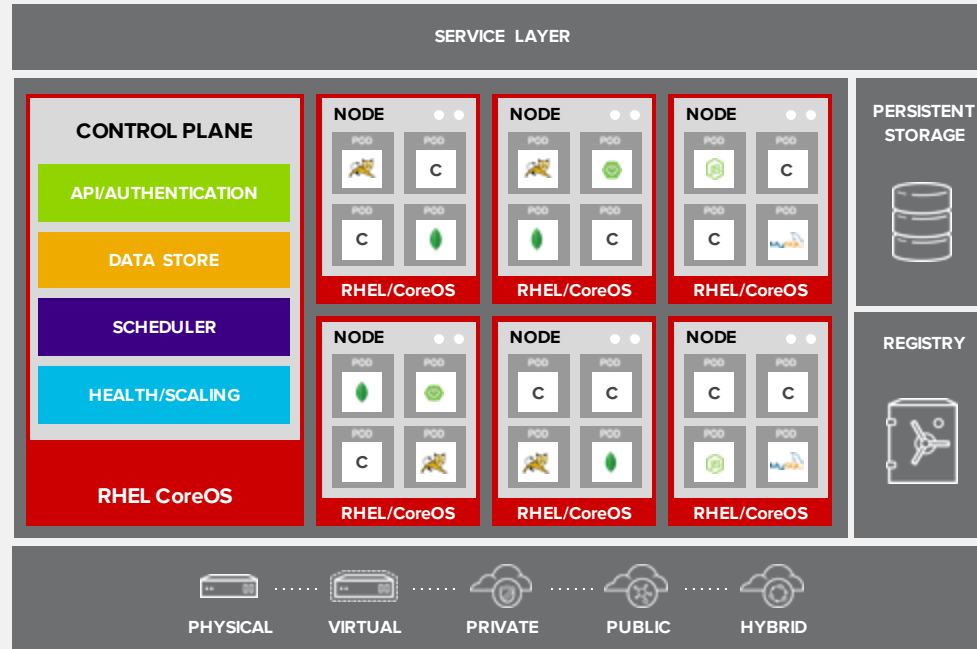
AUTOSCALING PODS



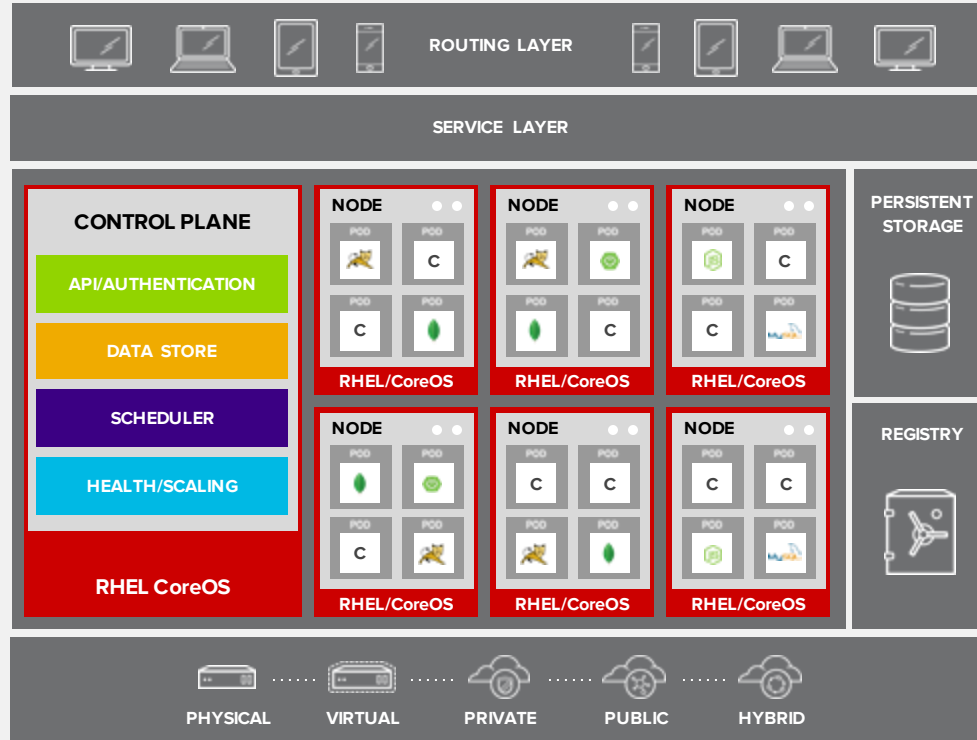
SERVICE DISCOVERY



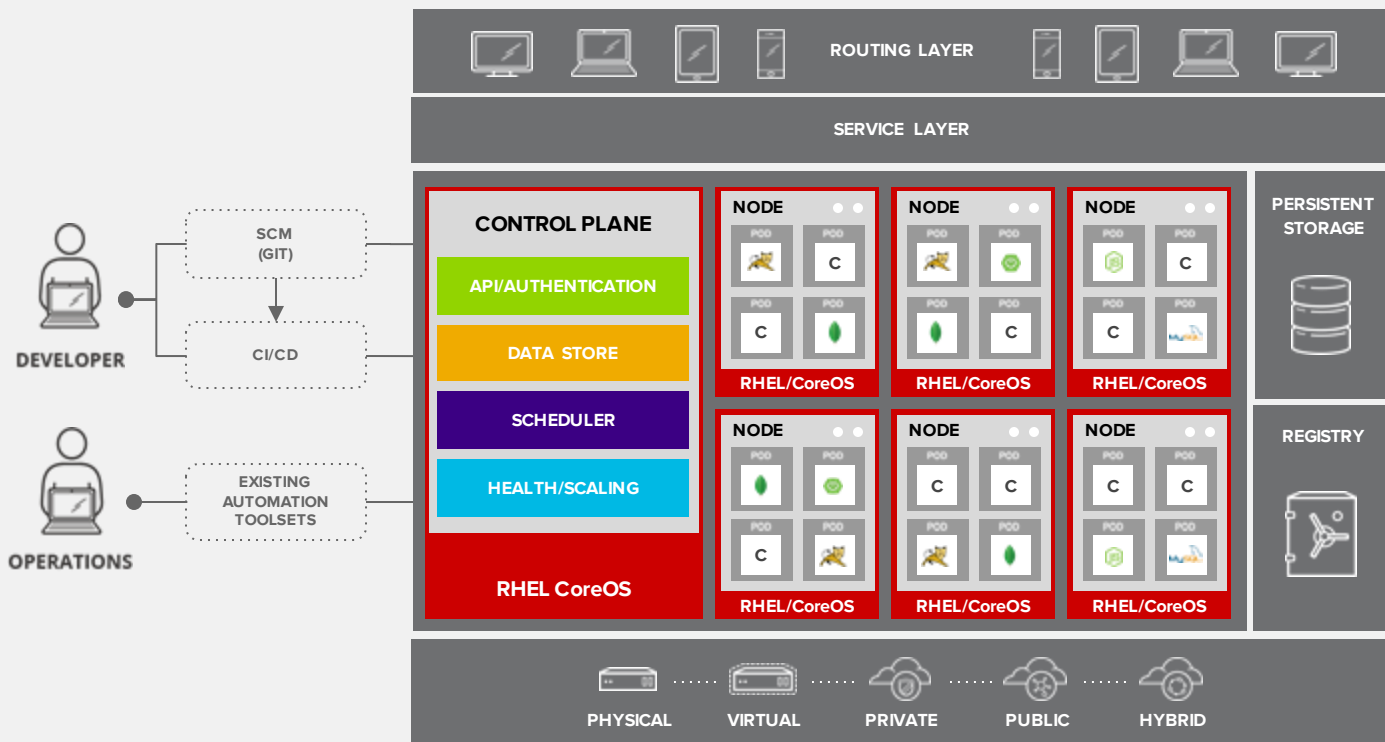
PERSISTENT DATA IN CONTAINERS



ROUTING AND LOAD-BALANCING



ACCESS VIA WEB, CLI, IDE AND API





HCS
C O M P A N Y

CONTAINER NATIVE STORAGE

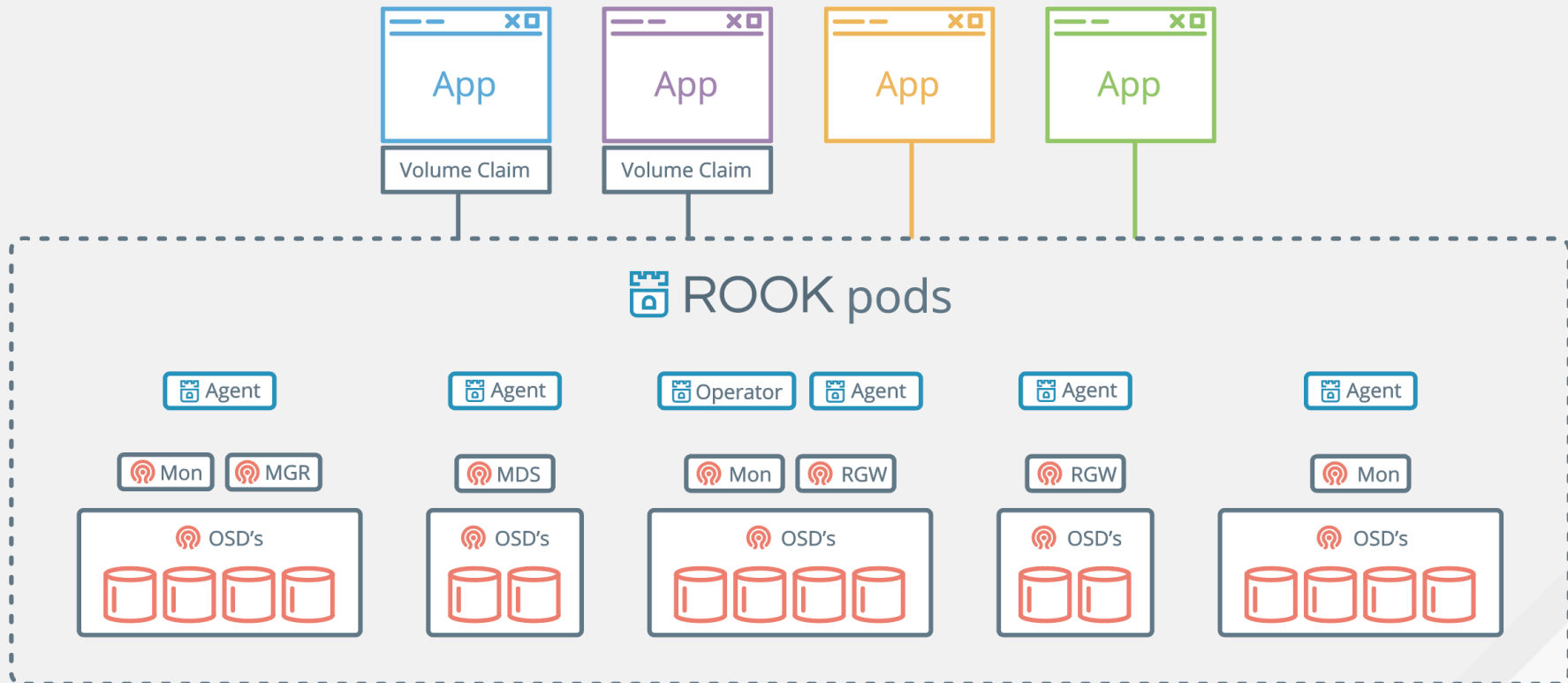


OCS 4 Component Overview

- META OPERATOR
- MACHINE API and MACHINE SETS
- ROOK (CEPH) OPERATOR
- MCG (NOOBAA) OPERATOR



Rook Architecture



OCS SIZING GUIDELINES

- OCS NODES NEED TO HAVE MINIMUM OF 16vCPU AND 64GB RAM
- MINIMUM NODES # - The MINIMUM amount of storage nodes is 3
- PV SUPPORTED # - Out-of-the-Box OCS4.2 supports up to 1500 PVs
- ADDITIONAL NODES - Each additional node enables for +500 PVs
- MAXIMUM NODES # - The MAXIMUM number of nodes in a cluster is 10
- MAXIMUM PV # - The MAXIMUM number of PVs can scale to 5000 PVs