

accenture

OCTOBER 18-19, 2018

BRATISLAVA, SLOVAKIA

Accenture conference on emerging
technologies and open source

OPENSLOVA'18

Organizers

accenture

STU
FEI

STU
FIIT



General partner



Media partner



Partners



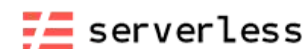
SIEMENS



Pivotal



Supporters



ABOUT ME



- Luigi Fugaro
- EMEA Middleware Architect @RedHat
- And you can find me:



@foogaro



@foogaro



@foogaro



@foogaro



That's me

Internet



OPENSLAVA'18

Name: OpenSlava

Password: Open2018



Agenda



OPENSLAVA'18

- **Presentation**
- **Lab**
- **Q/A**

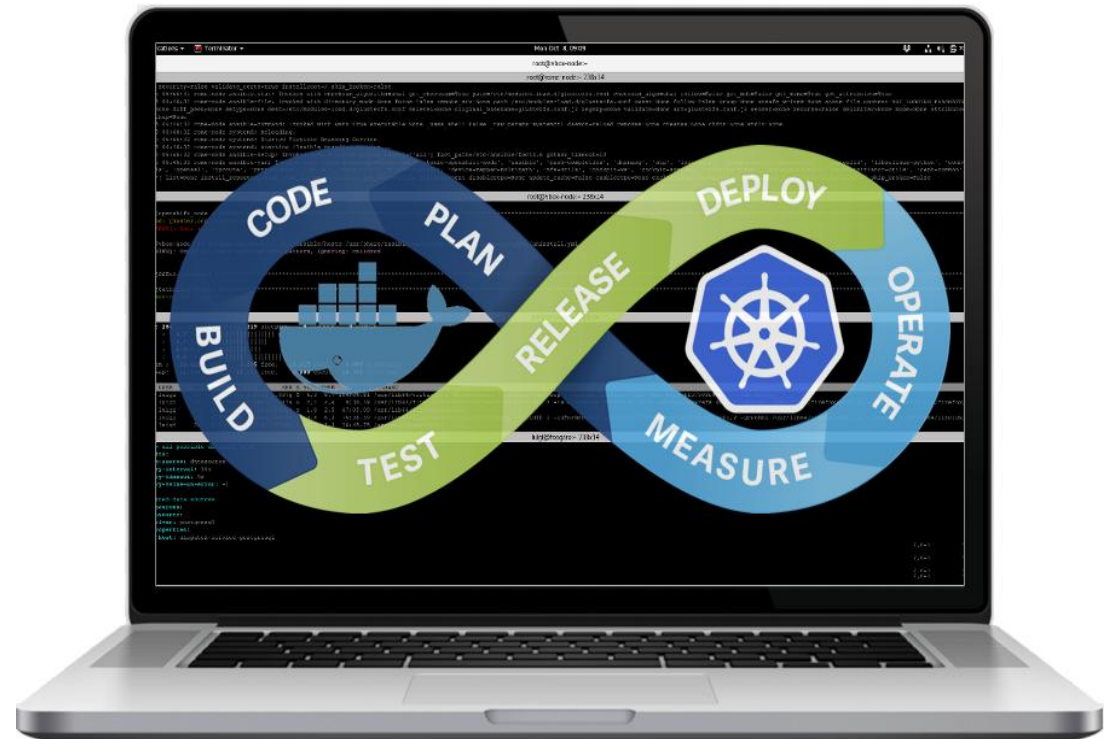


Agenda



OPENS�AVA'18

- Presentation
- Lab
- Q/A



Agenda



OPENSLAVA'18

- **Presentation**
- **Lab**
- **Q/A**



Cloud Native Applications



Why?

- We need to scale
- Cloud scales by design and definition

Cloud Native Applications



How?

- Containers
- Orchestrator

Cloud Native Applications



The problem

- Monolith applications
- Applications are stateful
- Applications need to be scale-aware

Cloud Native Applications



Solution

- Microservice approach
- 12-factor applications
- Enterprise support

Cloud Native Applications



OPENS�AVA'18

Monolith



Cloud Native Applications



OPENS�AVA'18

Monolith

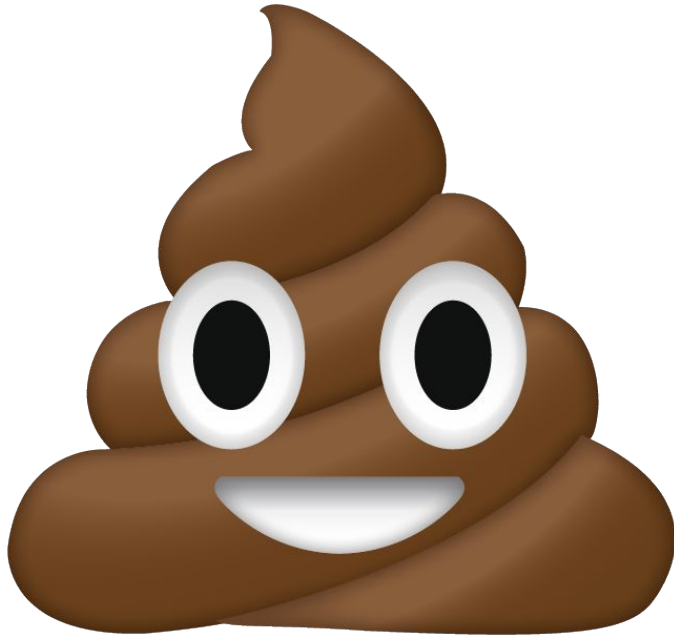


Cloud Native Applications

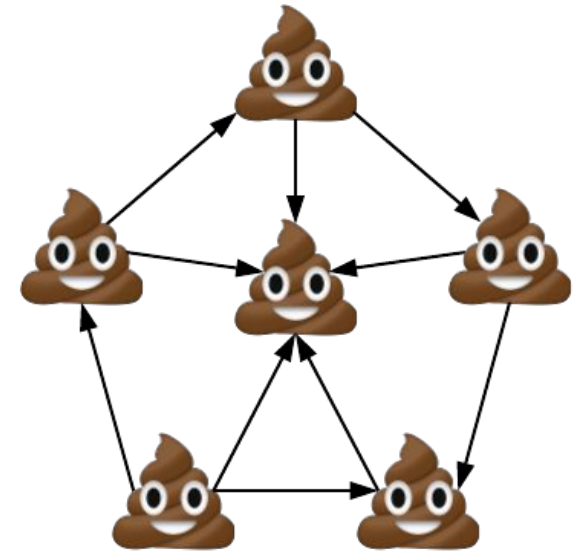


OPENS�AVA'18

Monolith



Microservices



Cloud Native Applications



12-Factor application

Cloud Native Applications



12-Factor application

1.Codebase

2.Dependencies

3.Configuration

4.Backing Services

5.Build, Release, Run

6.Processes

7.Port Binding

8.Concurrency

9.Disposability

10.Dev/Prod Parity

11.Logs

12.Admin Processes

Cloud Native Applications



12-Factor application

- 1.Codebase
- 2.Dependencies
- 3.Configuration
- 4.Backing Services
- 5.Build, Release, Run
- 6.Processes

13. Security

- 7.Port Binding
- 8.Concurrency
- 9.Disposability
- 10.Dev/Prod Parity
- 11.Logs
- 12.Admin Processes

Cloud Native Applications



12-Factor application

- 1.Codebase
- 2.Dependencies
- 3.Configuration
- 4.Backing Services
- 5.Build, Release, Run
- 6.Processes

13. Security

14. ???

- 7.Port Binding
- 8.Concurrency
- 9.Disposability
- 10.Dev/Prod Parity
- 11.Logs
- 12.Admin Processes

Cloud Native Applications



12-Factor application

1.Codebase

2.Dependencies

3.Configuration

4.Backing Services

5.Build, Release, Run

6.Processes

7.Port Binding

8.Concurrency

9.Disposability

10.Dev/Prod Parity

11.Logs

12.Admin Processes

13.Security

14.Next...

Cloud Native Applications



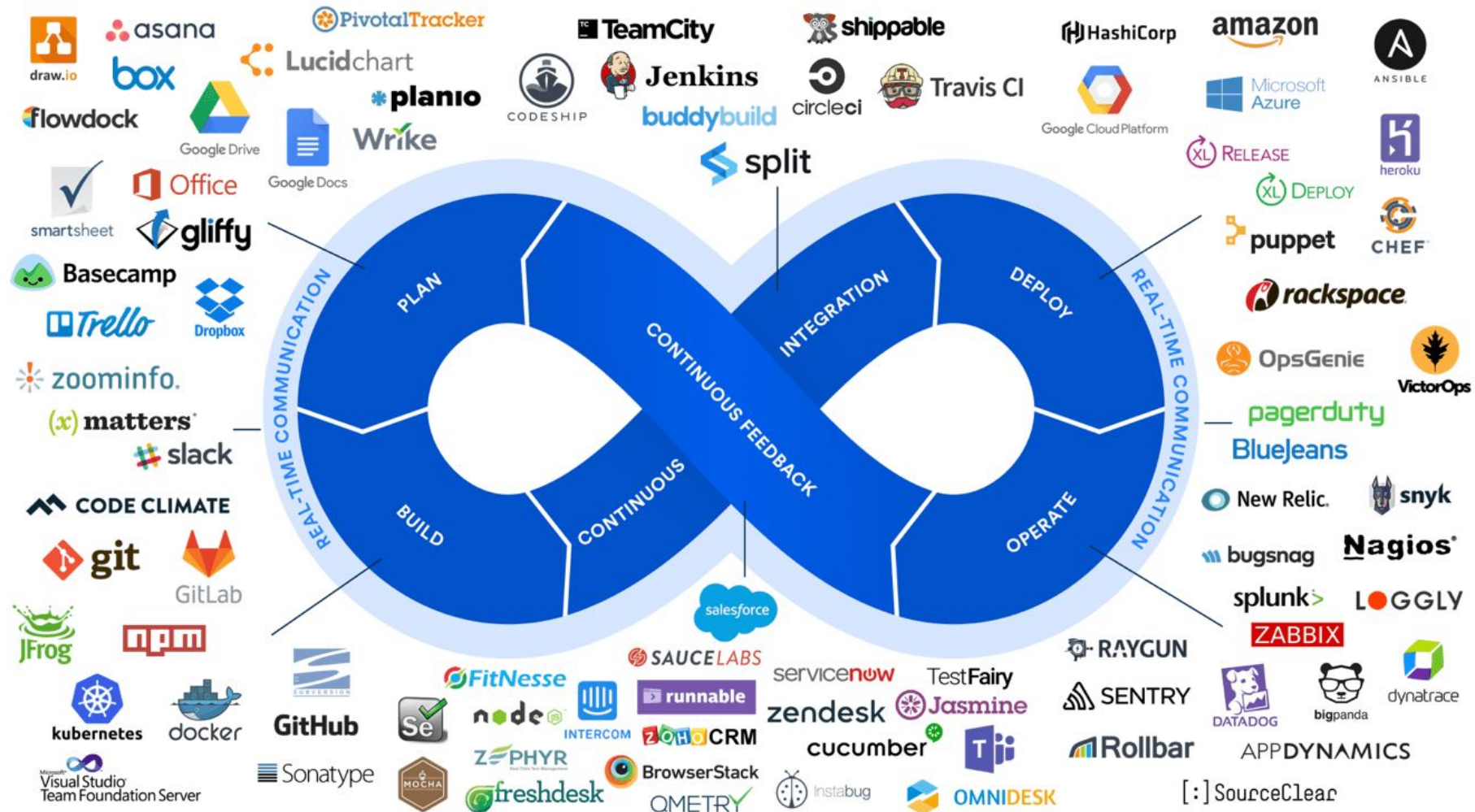
OPENSLAVA'18

The right tool for the right job

Cloud Native Applications



OPENS�AVA'18



Cloud Native Applications



**Which are the de-facto standards in terms of
DevOps?
(hopefully opensource)**

Cloud Native Applications



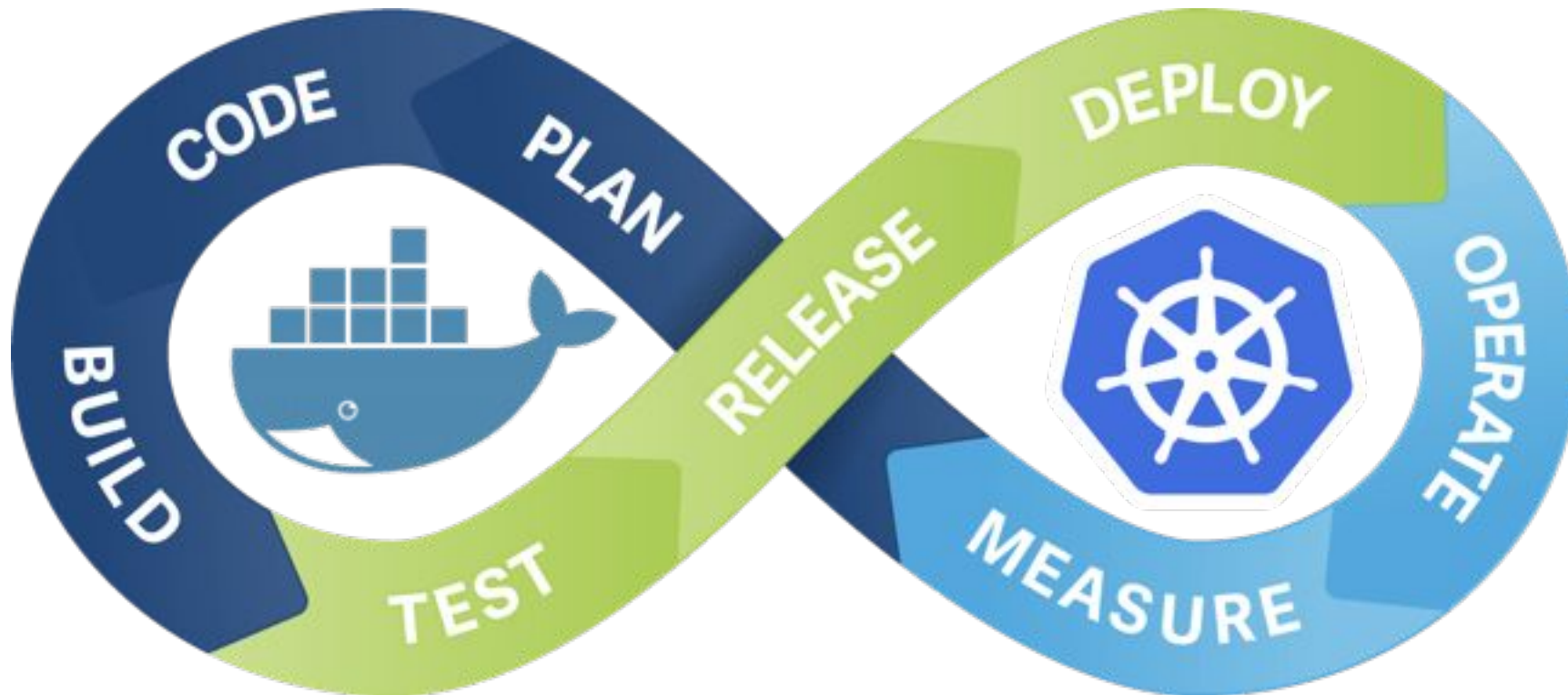
OPENSSLAVA'18



Cloud Native Applications



OPENSLAVA '18



Cloud Native Applications



OPENS�AVA'18

Do we have a platform that merges the right tool for the right job?

Cloud Native Applications



OPENS�AVA'18

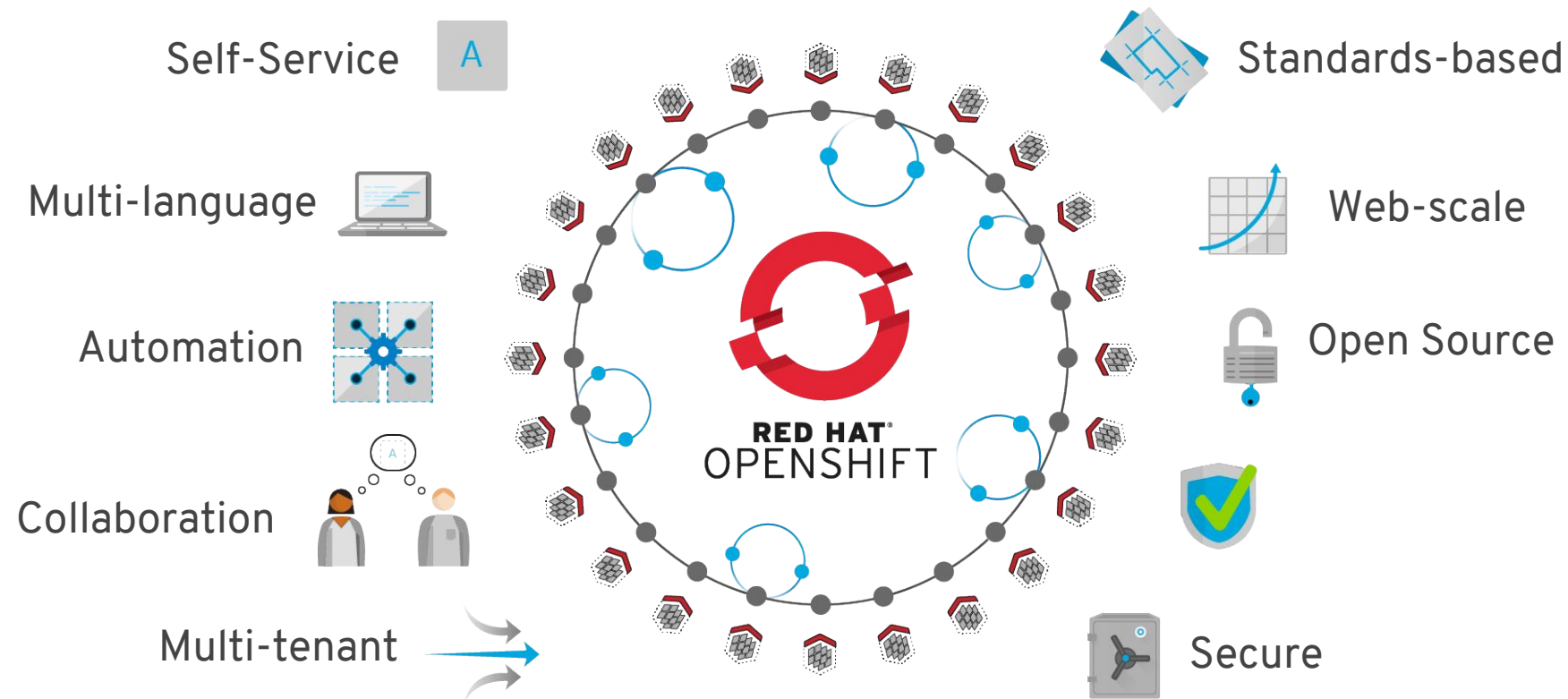


RED HAT®
OPENSİIFT
Container Platform

Cloud Native Applications



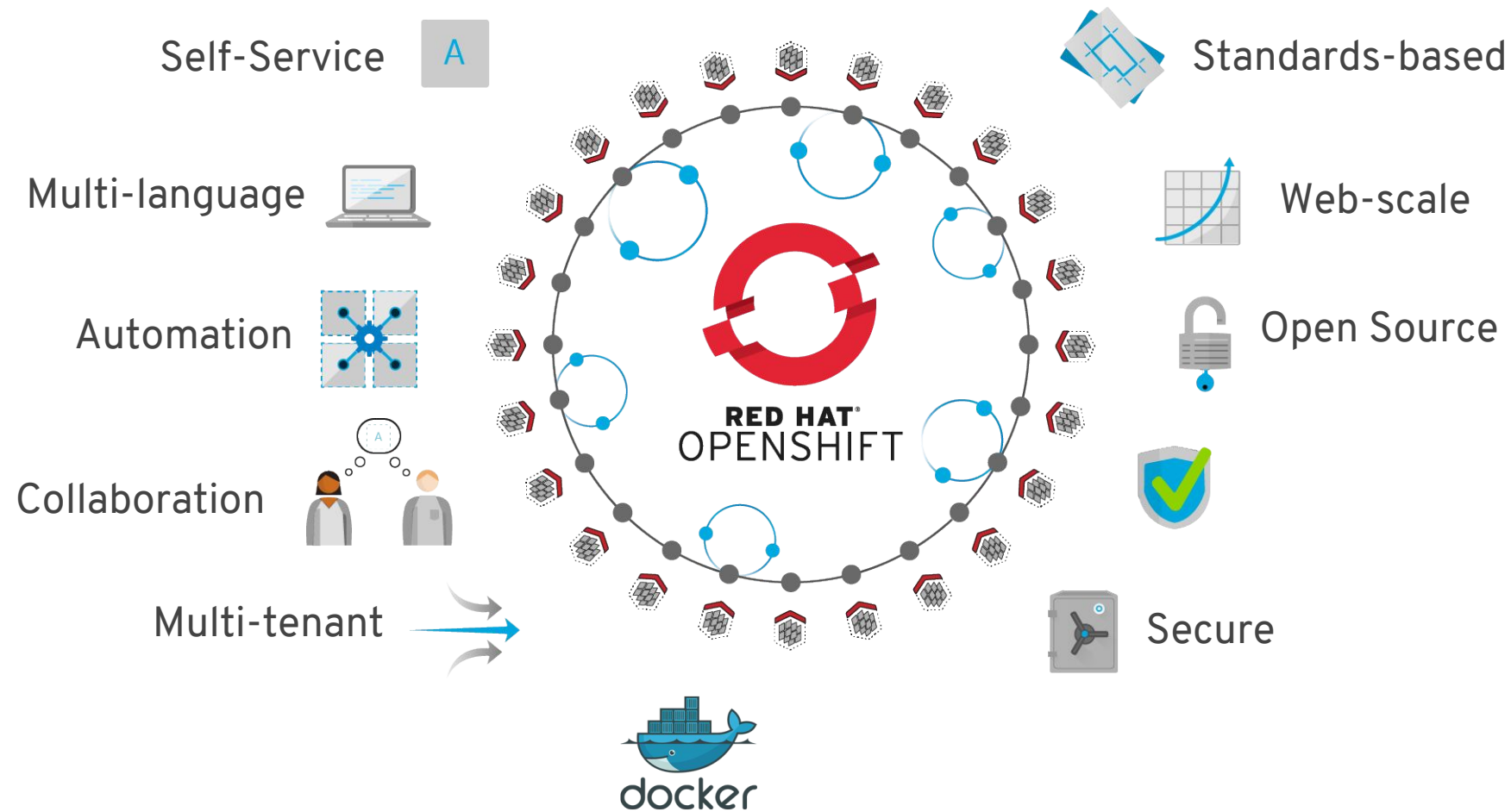
OPENS�AVA'18



Cloud Native Applications



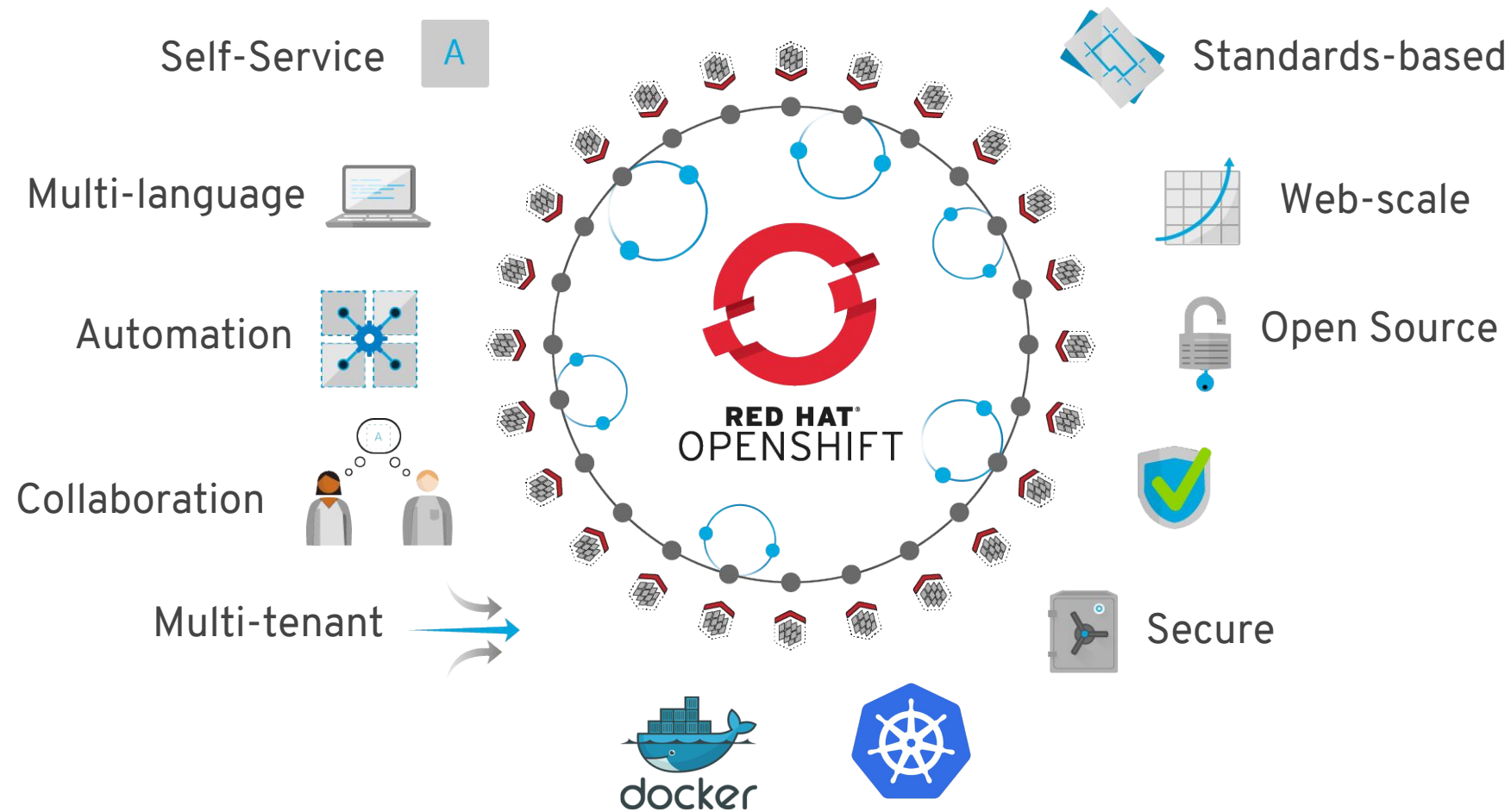
OPENS�AVA'18



Cloud Native Applications



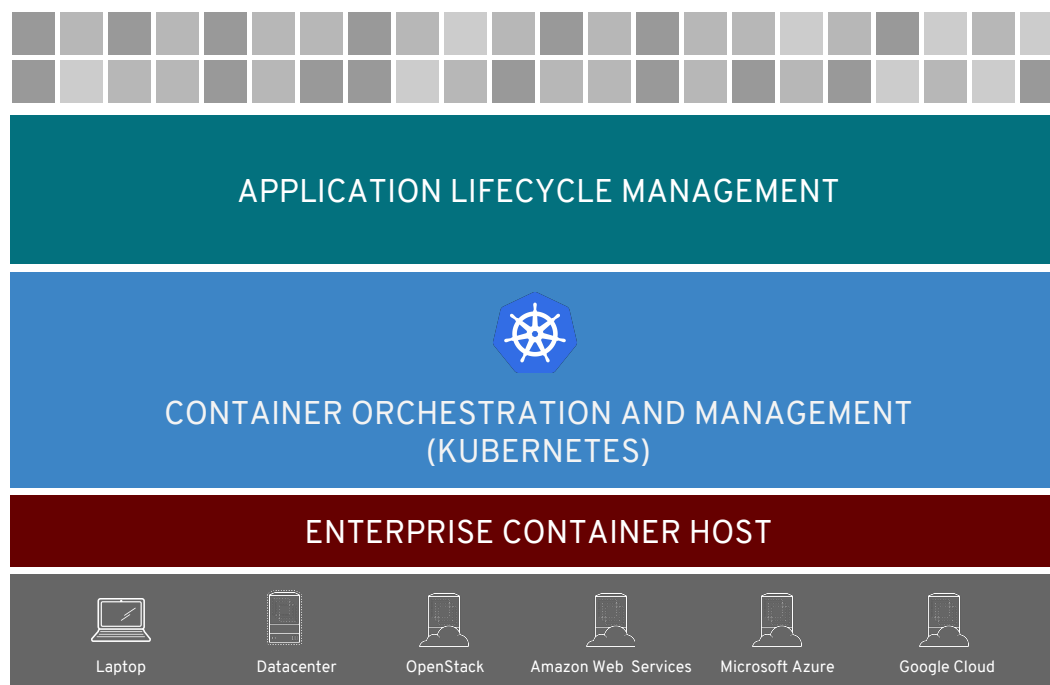
OPENS�AVA'18



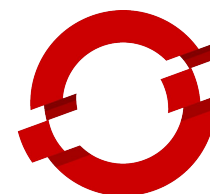
Cloud Native Applications



OPENS�AVA'18



ANY
CONTAINER



RED HAT
OPENSŁIFT

ANY
INFRASTRUCTURE

Cloud Native Applications



CRI-O is an implementation of the Kubernetes CRI (Container Runtime Interface) to enable using OCI (Open Container Initiative) compatible runtimes.

**Optimized for
Kubernetes**

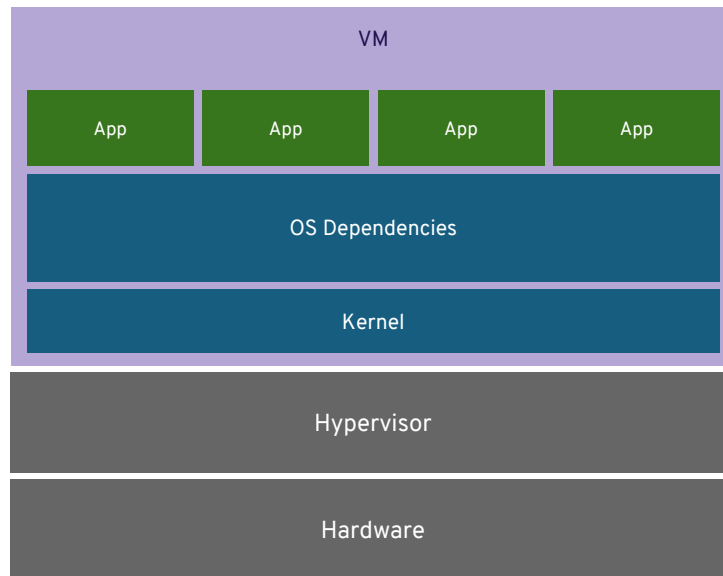
**Any OCI-compliant
container from any
OCI registry
(including docker)**

**Improve Security
and Performance at
scale**

Cloud Native Applications

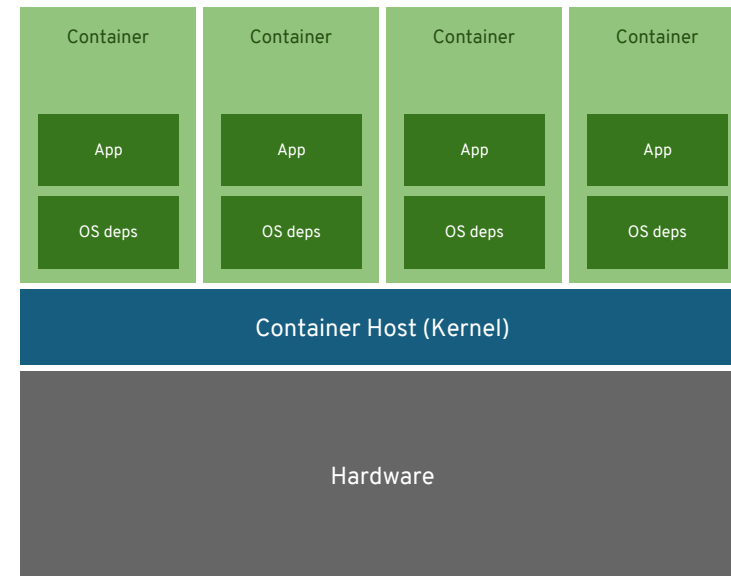


VIRTUAL MACHINES



VM virtualizes the hardware

CONTAINERS

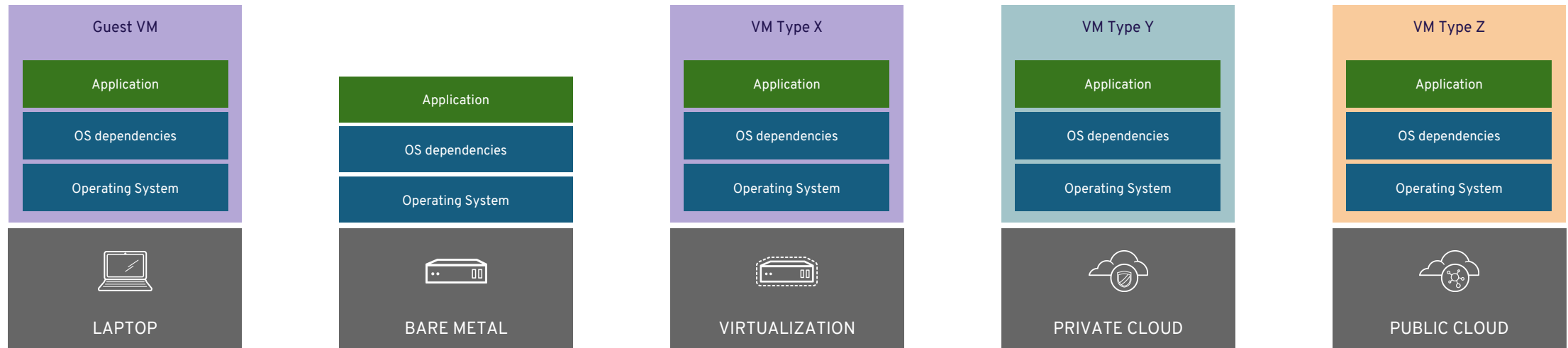


Container virtualizes the process

Cloud Native Applications



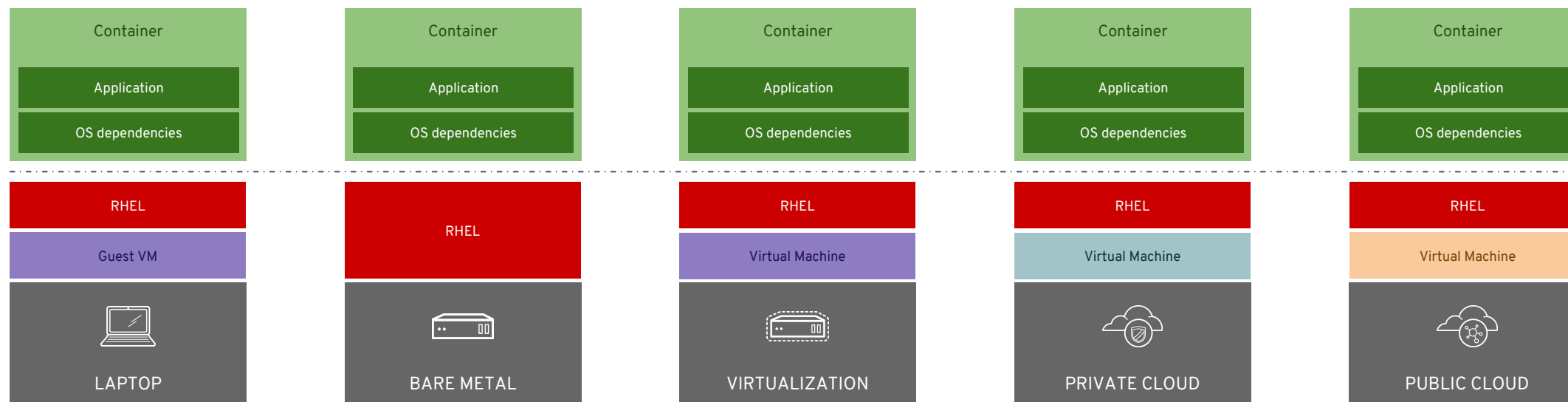
Virtual machines are **NOT** portable across hypervisor and do **NOT** provide portable packaging for applications



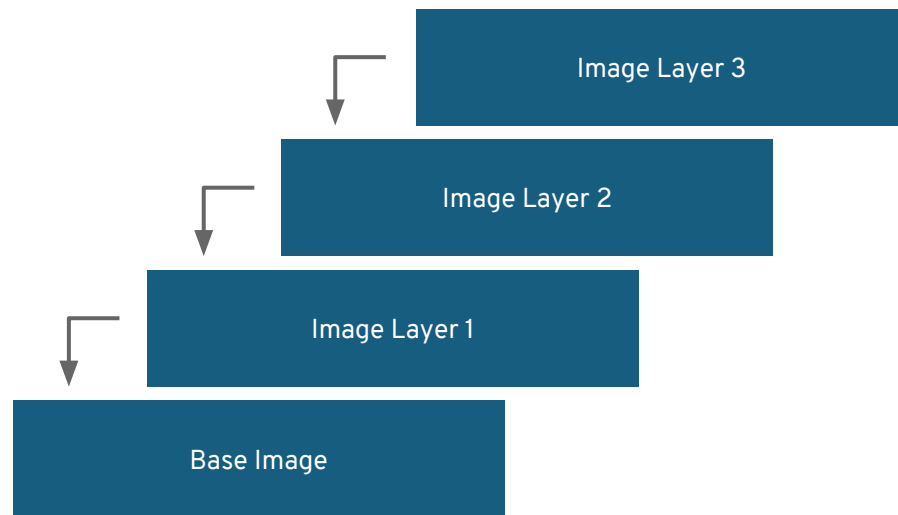
Cloud Native Applications



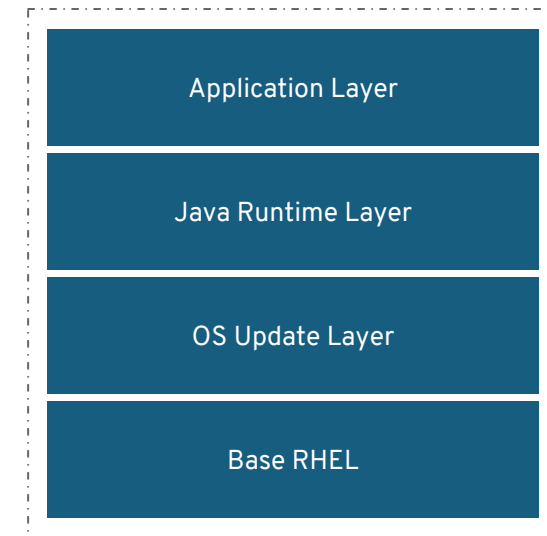
RHEL Containers + RHEL Host = Guaranteed Portability
Across Any Infrastructure



Cloud Native Applications



Container Image Layers



Example Container Image

Cloud Native Applications



A container is the smallest compute unit

A solid teal square representing a container.

CONTAINER

Cloud Native Applications



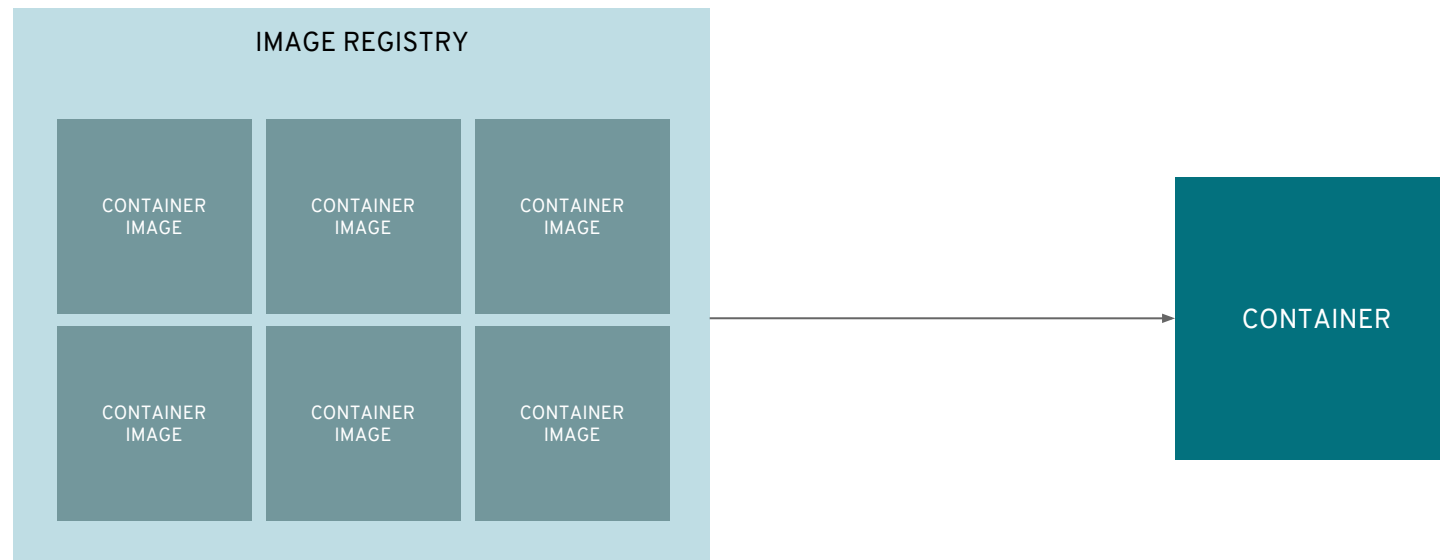
Containers are created from container images



Cloud Native Applications



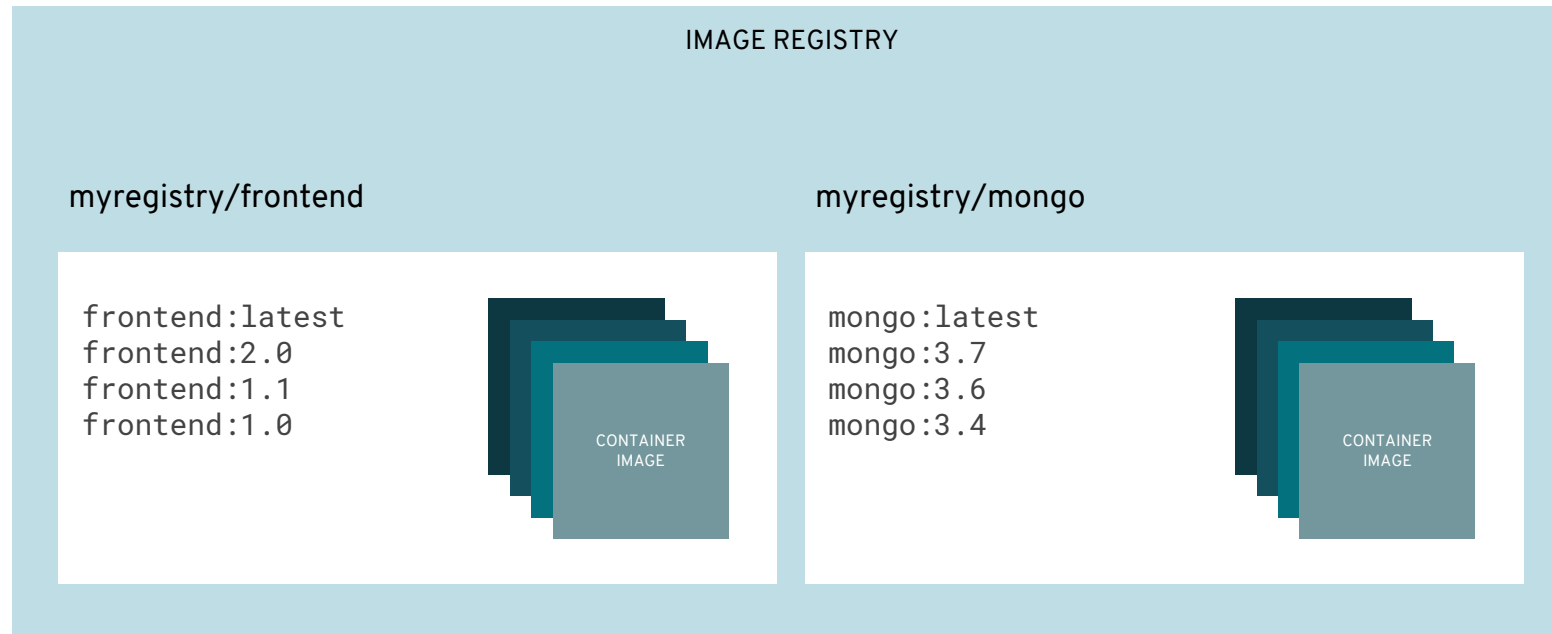
Container images are stored in an image registry



Cloud Native Applications



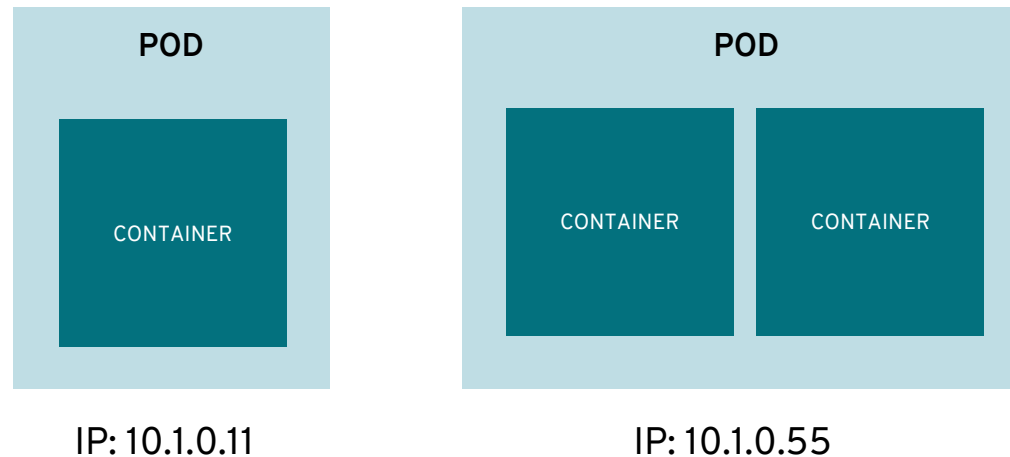
An image repository contains all versions of an image in the image registry



Cloud Native Applications



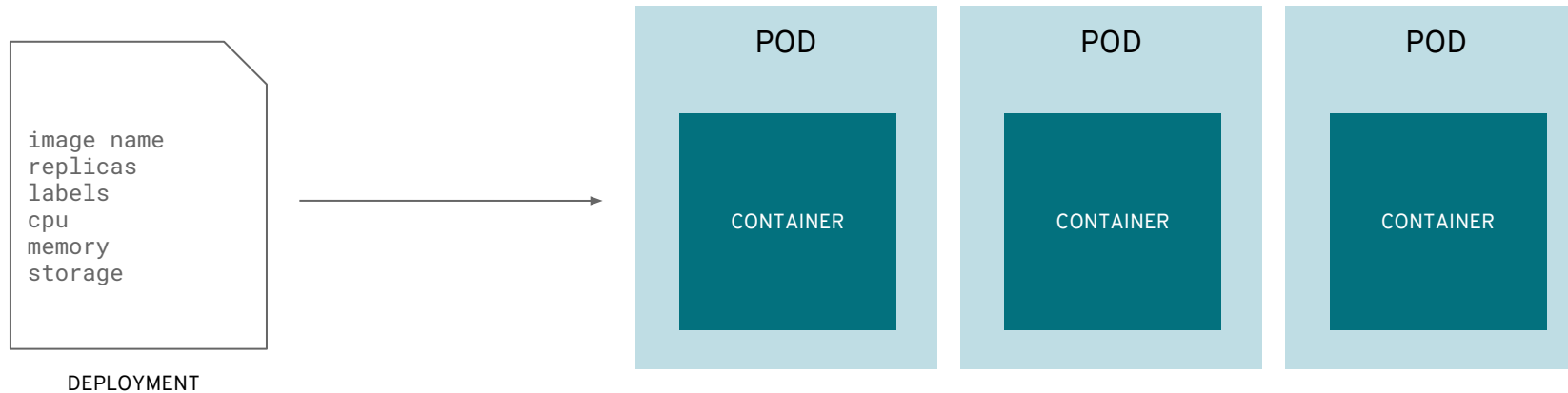
Containers are wrapped in pods which are units of deployment and management



Cloud Native Applications



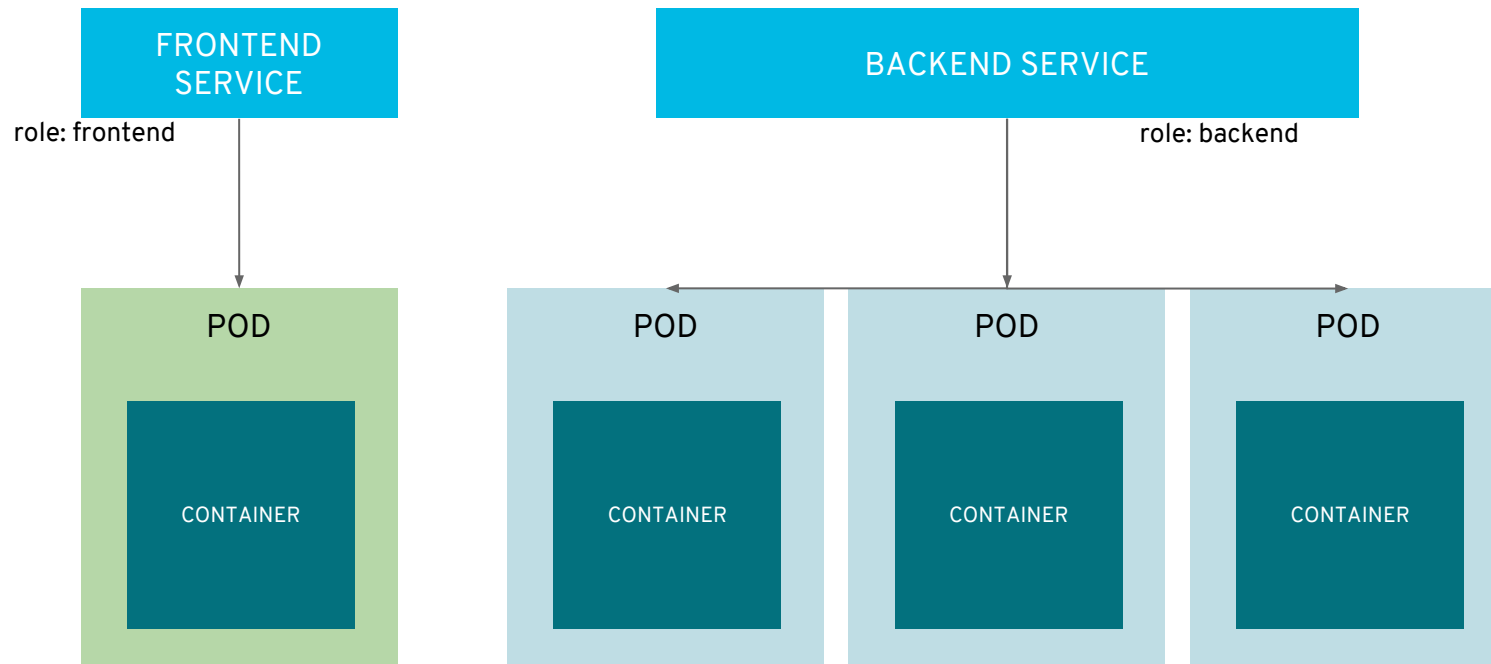
Pods configuration is defined in a deployment



Cloud Native Applications



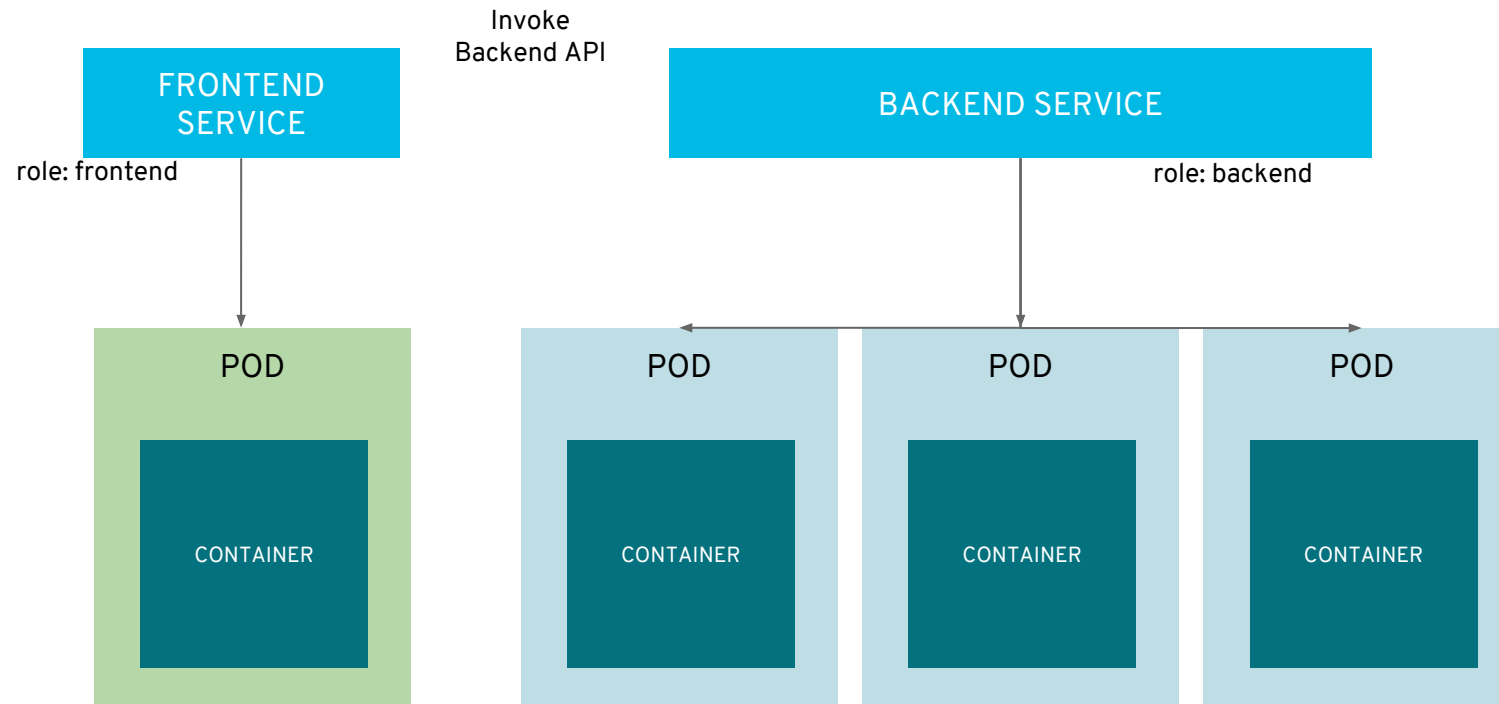
Services provide internal load-balancing and service discovery across pods



Cloud Native Applications



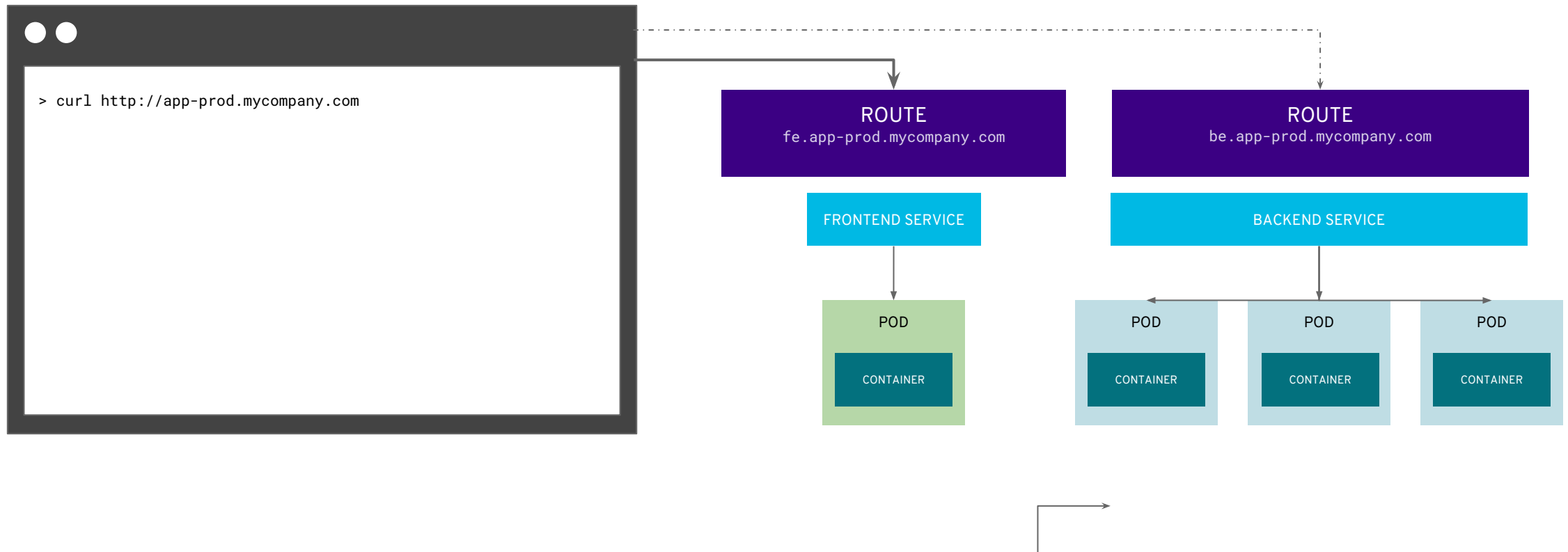
Apps can talk to each other via services



Cloud Native Applications



Routes add services to the external load-balancer and provide readable urls for the app



Cloud Native Applications



OpenShift Architecture

Cloud Native Applications



OPENSSLAVA'18


PHYSICAL


VIRTUAL


PRIVATE

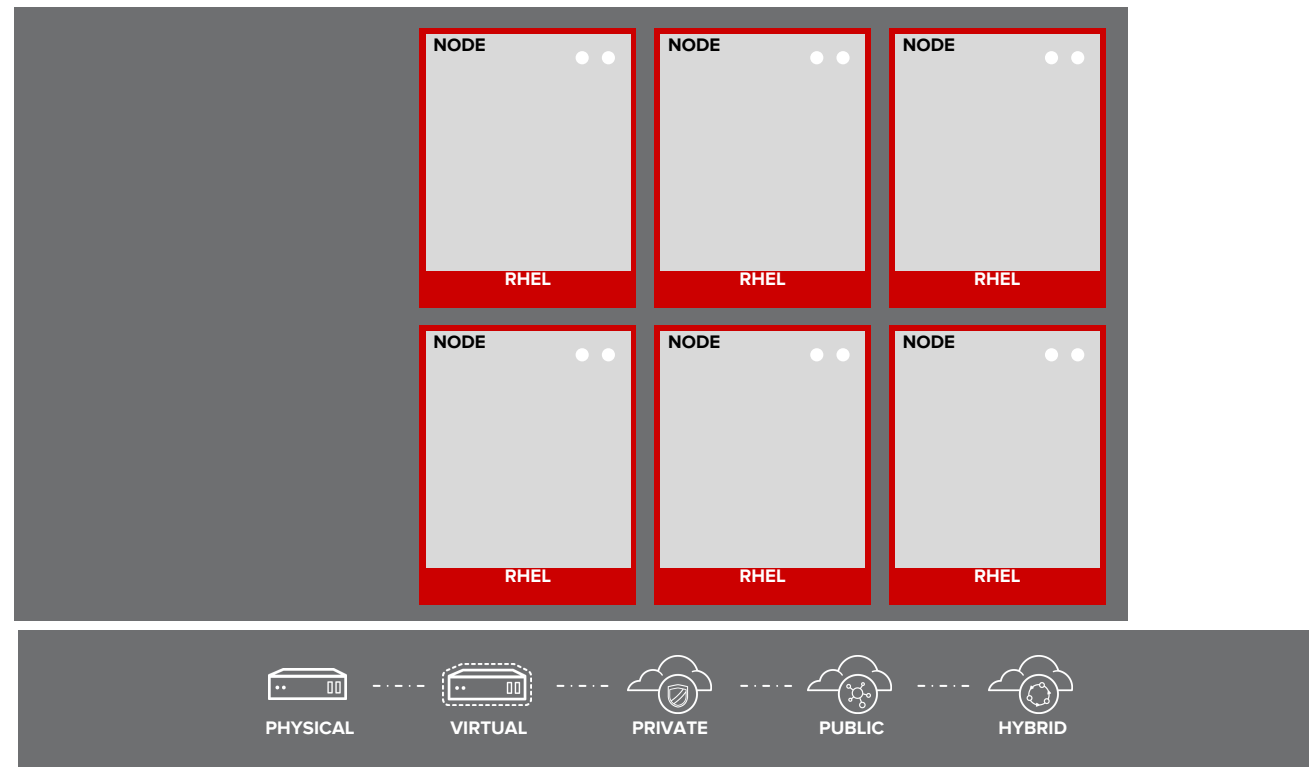

PUBLIC


HYBRID

Cloud Native Applications



OPENS�AVA'18



Cloud Native Applications



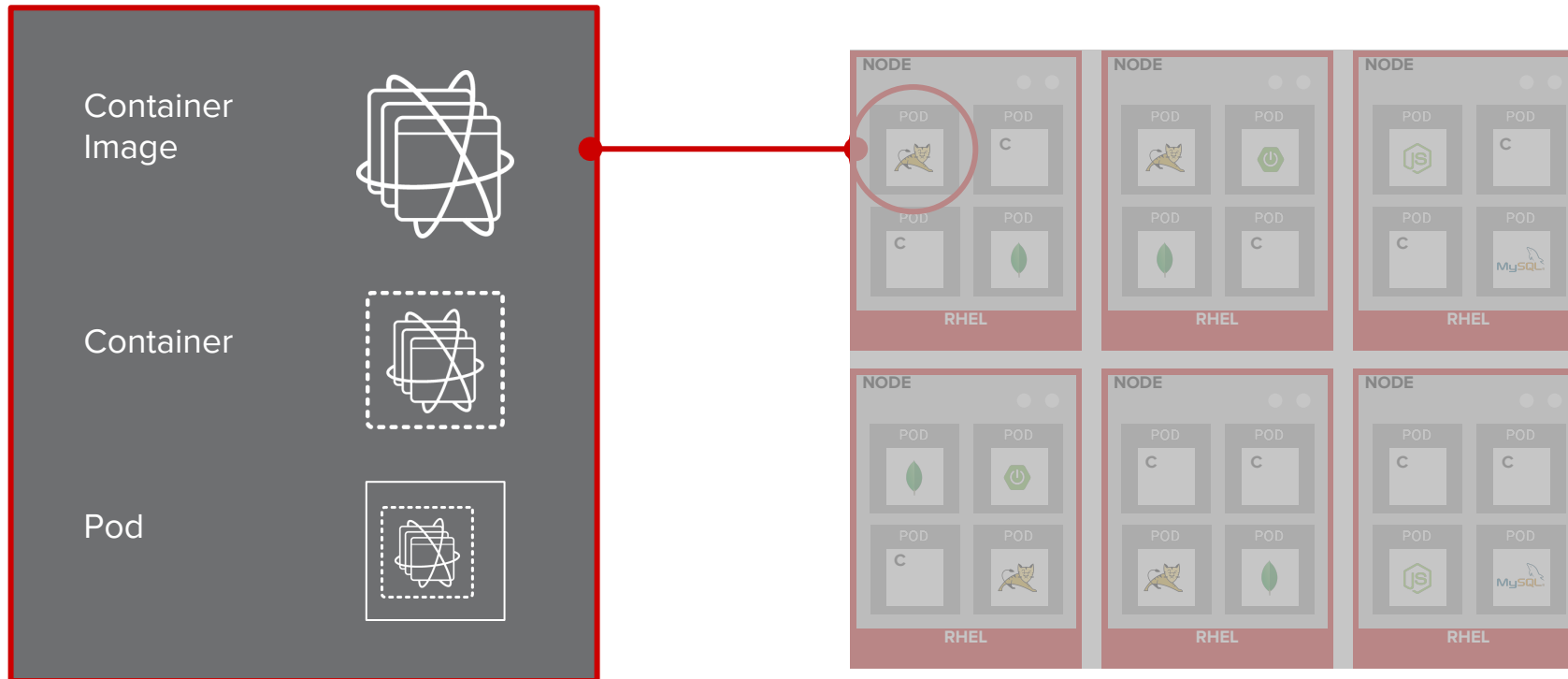
OPENS�AVA'18



Cloud Native Applications



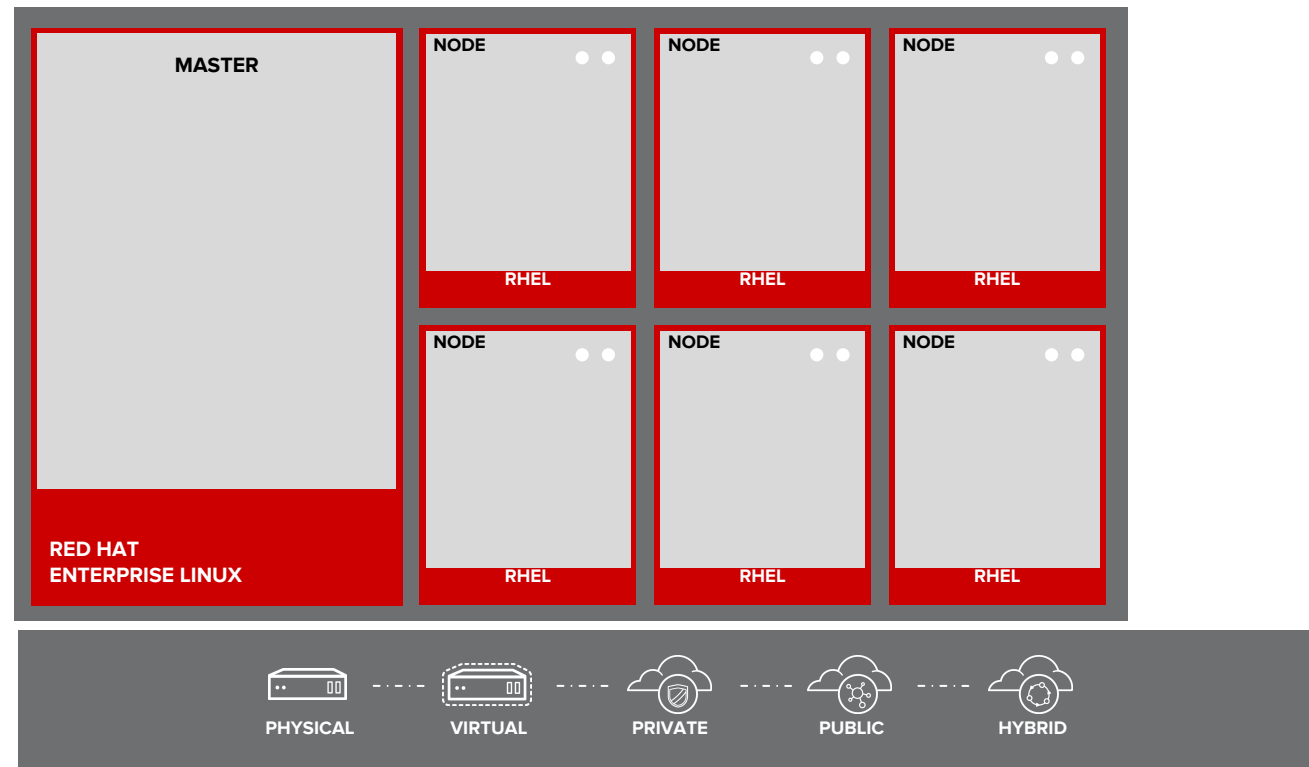
OPENS�AVA'18



Cloud Native Applications



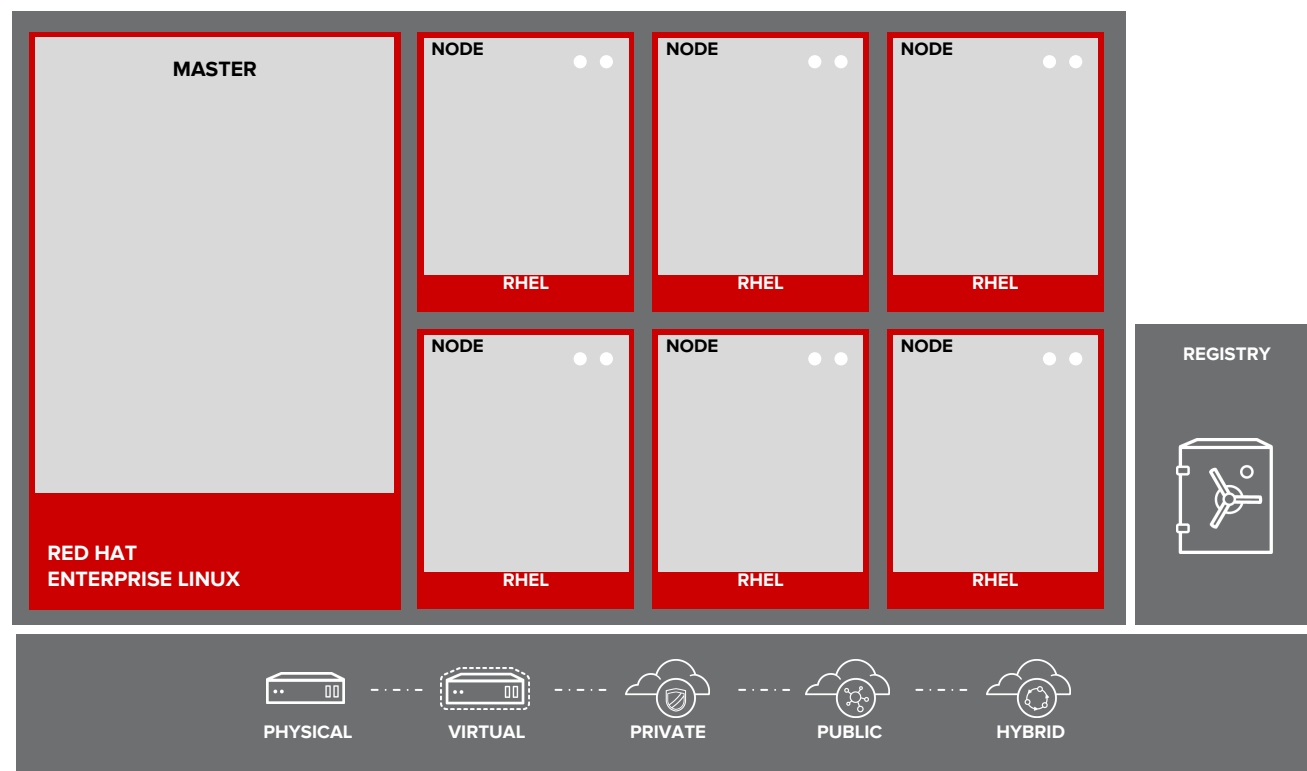
OPENS�AVA'18



Cloud Native Applications



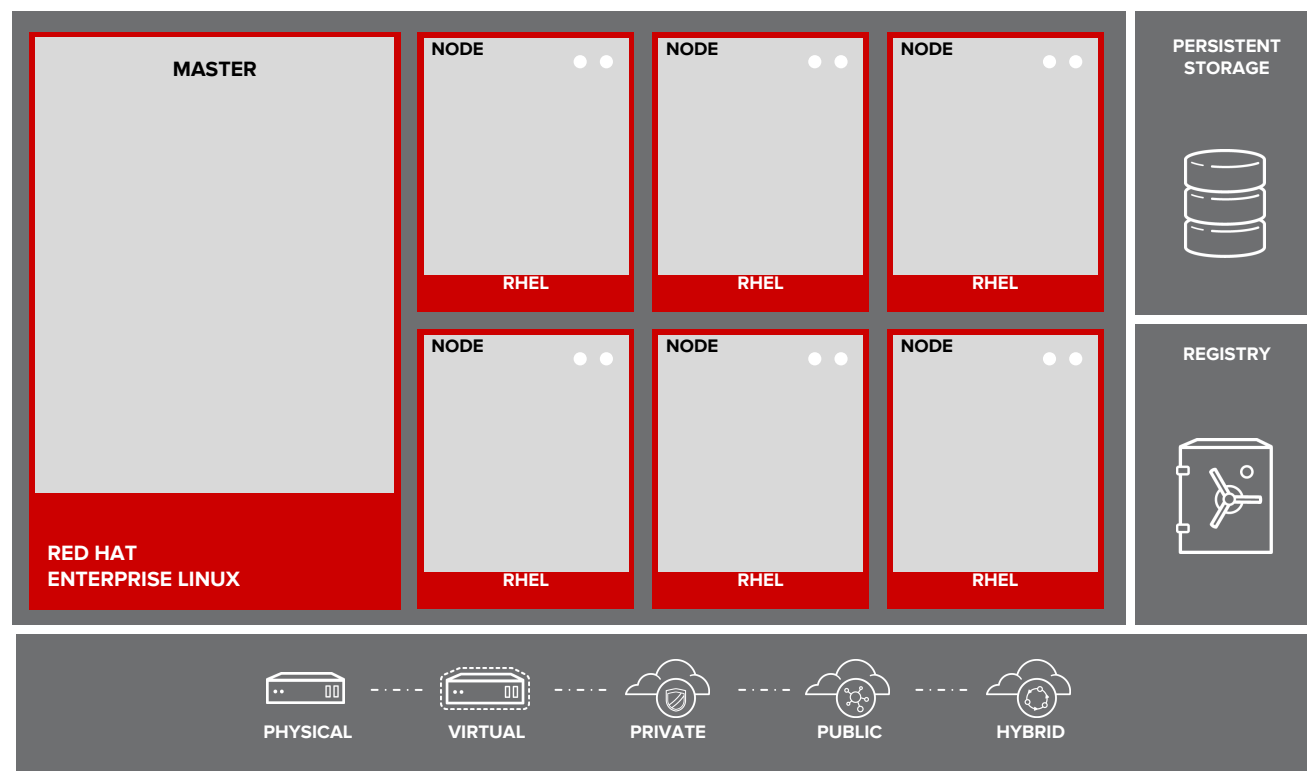
OPENS�AVA'18



Cloud Native Applications



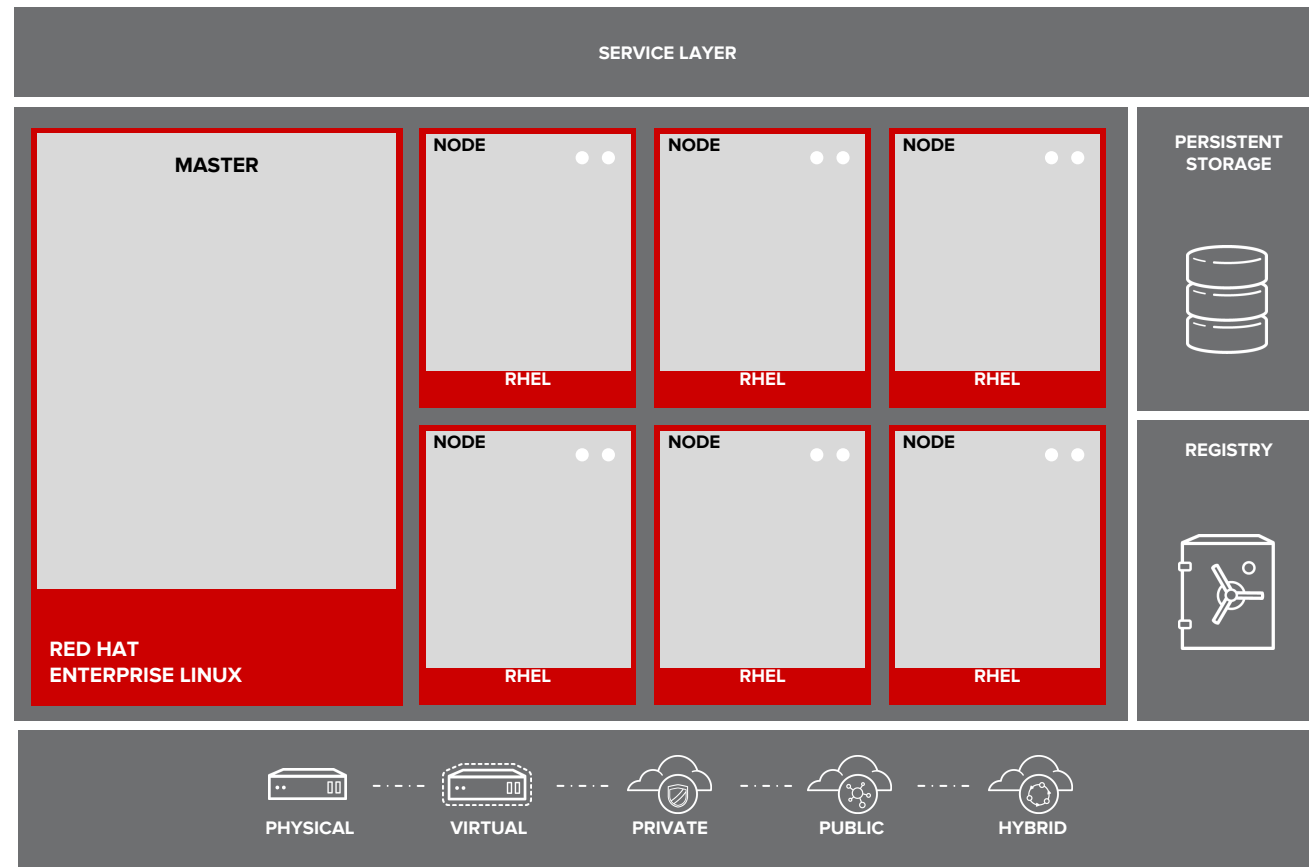
OPENSLOVA'18



Cloud Native Applications



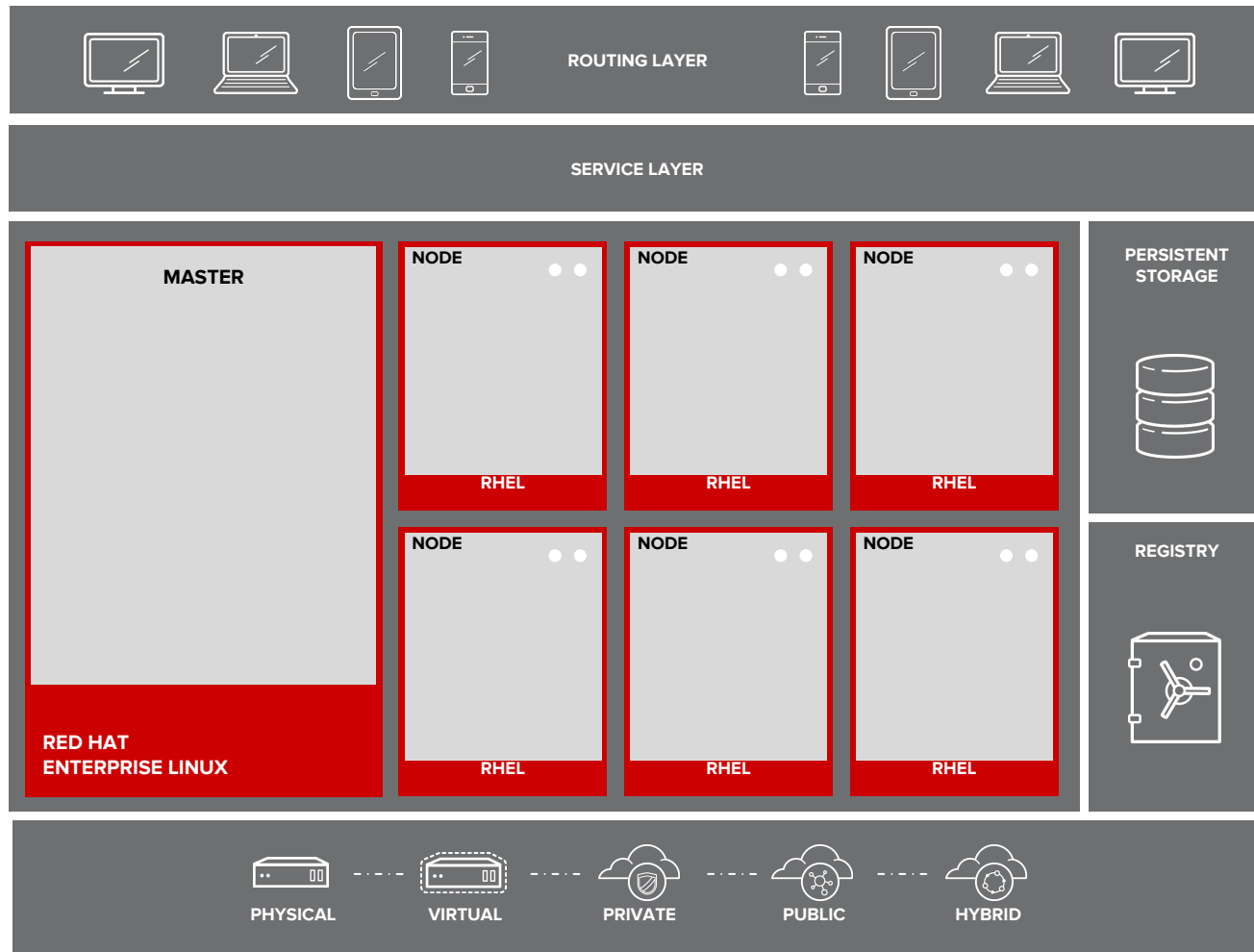
OPENS�AVA'18



Cloud Native Applications



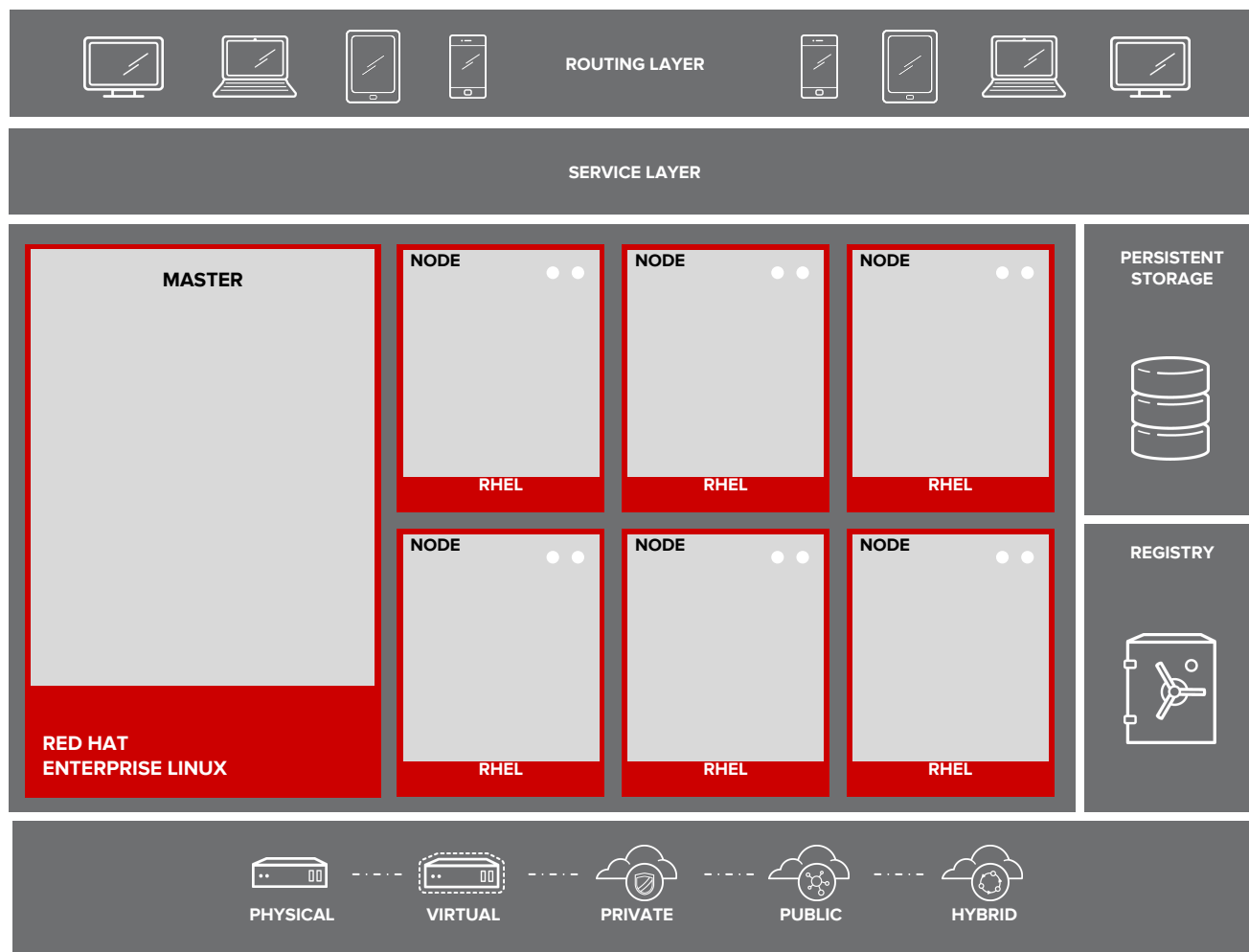
OPENS�AVA'18



Cloud Native Applications



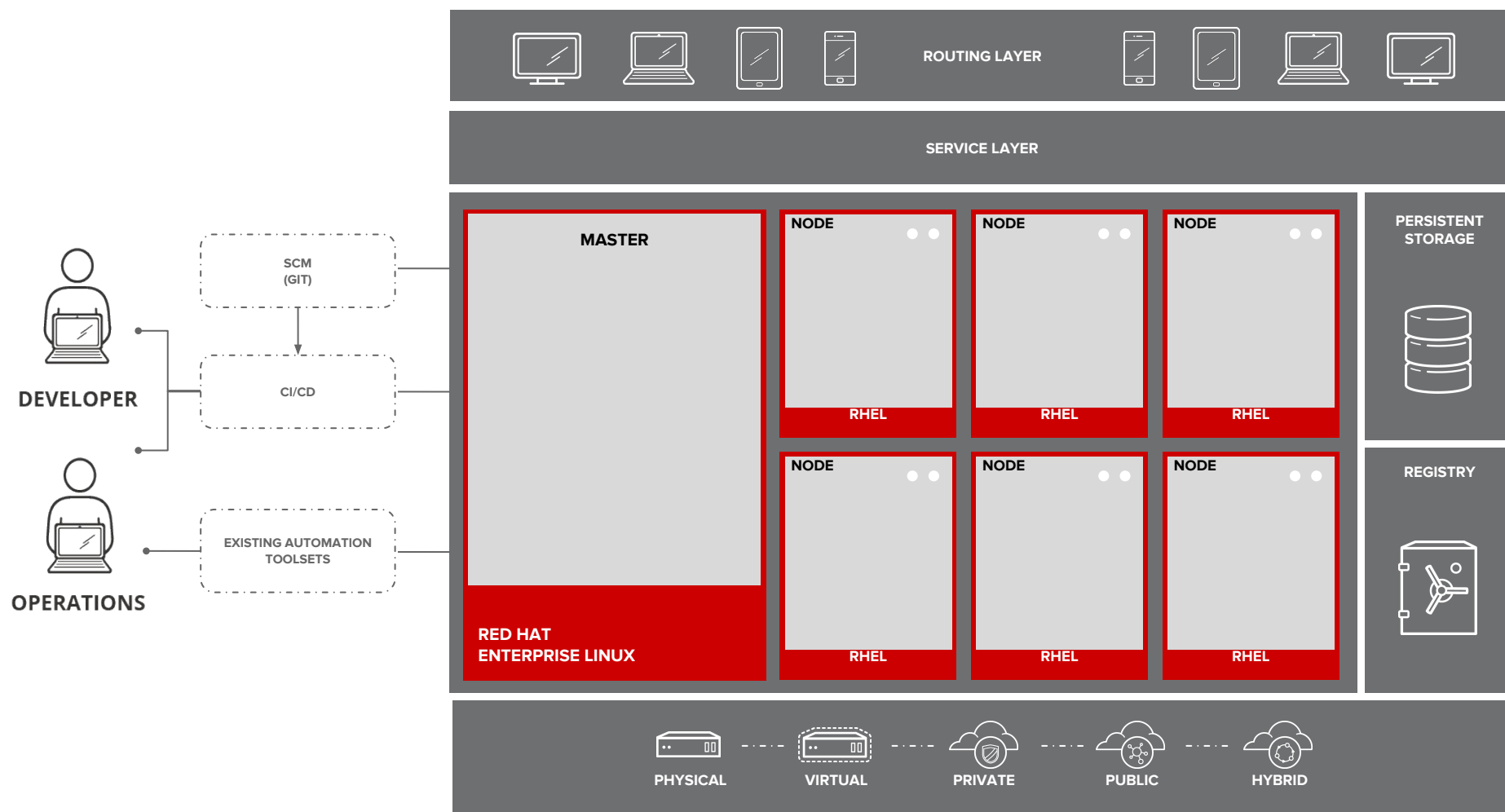
OPENS�AVA'18



Cloud Native Applications



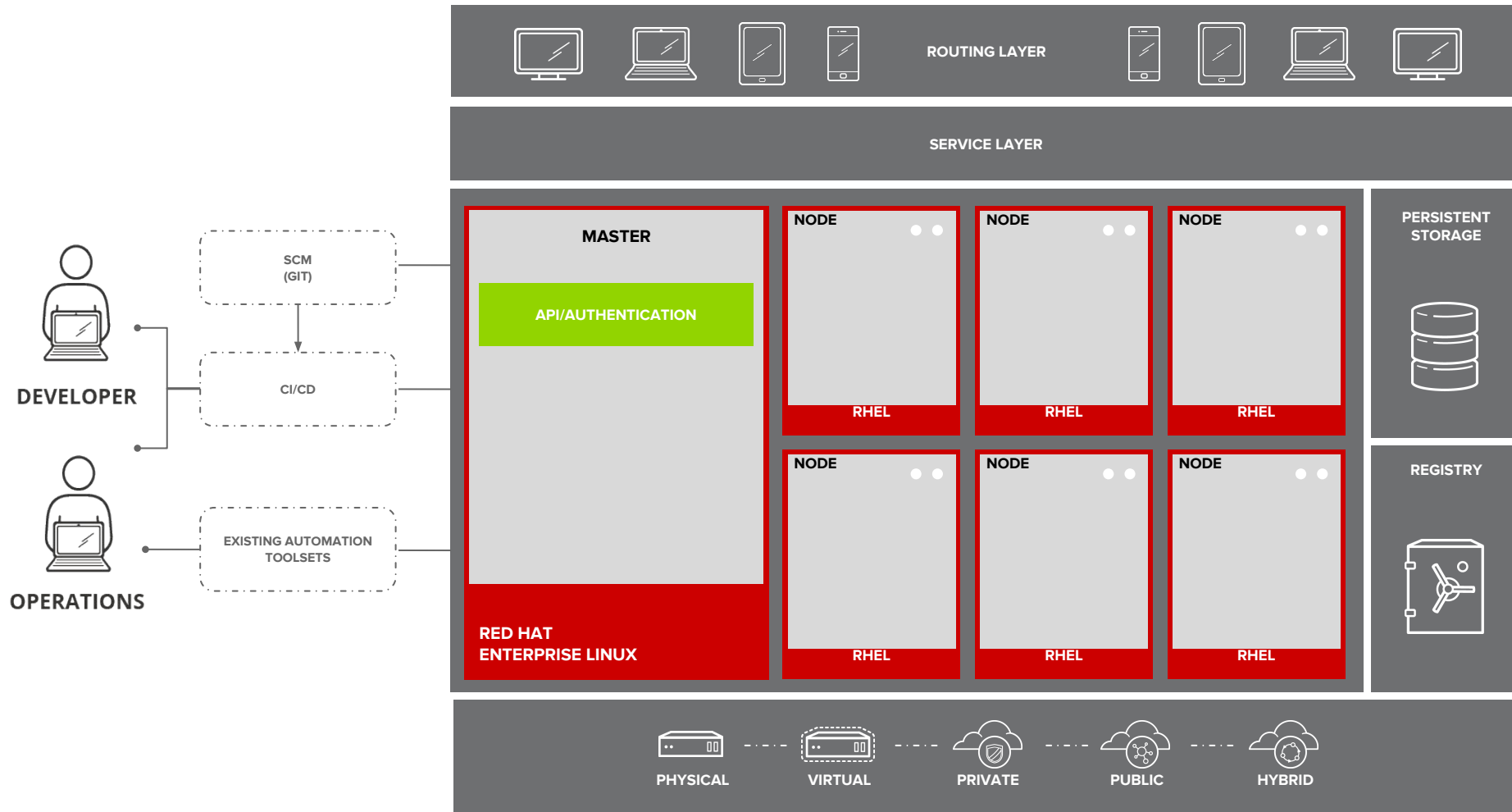
OPENS�AVA'18



Cloud Native Applications



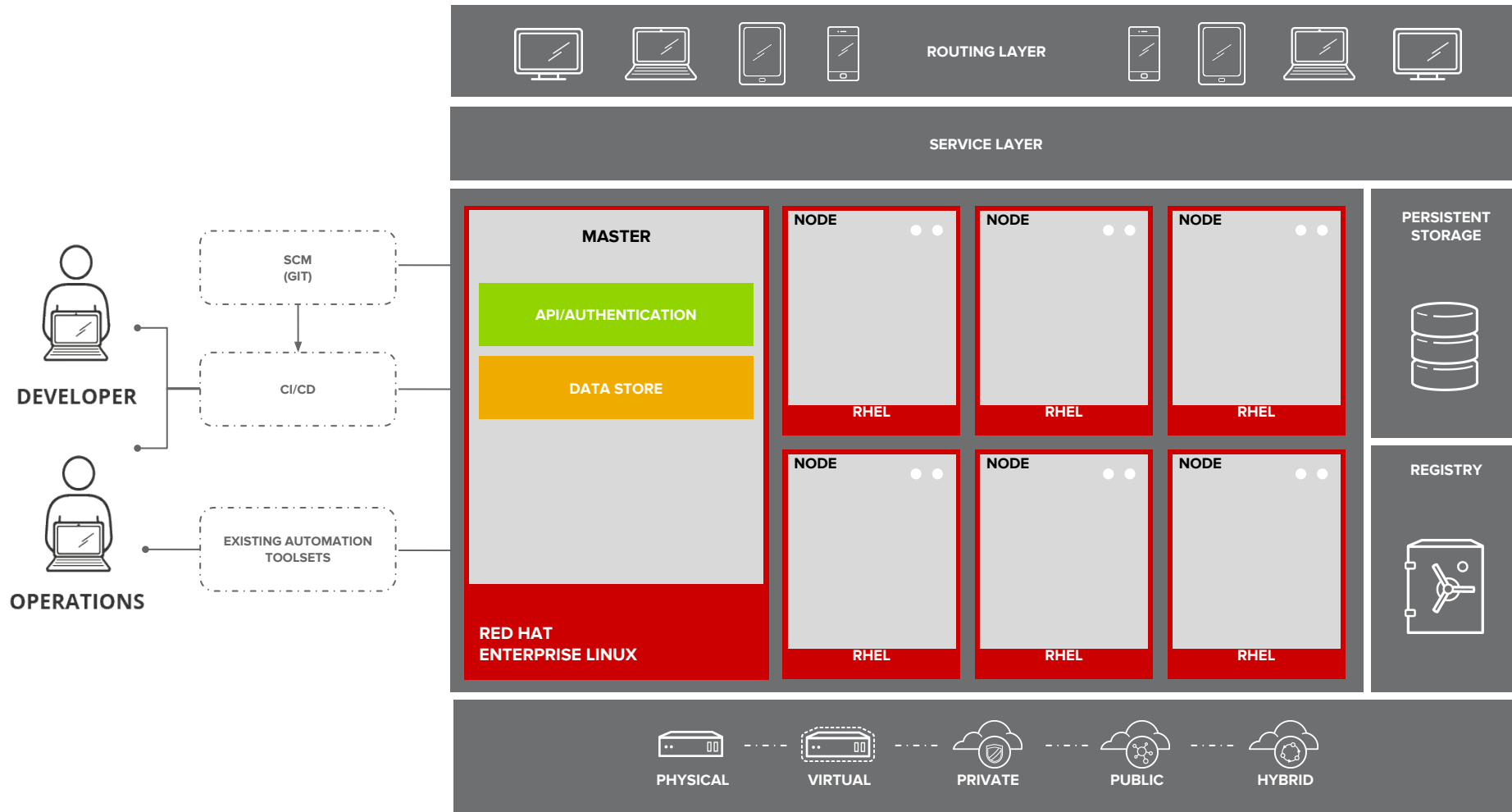
OPENS�AVA'18



Cloud Native Applications



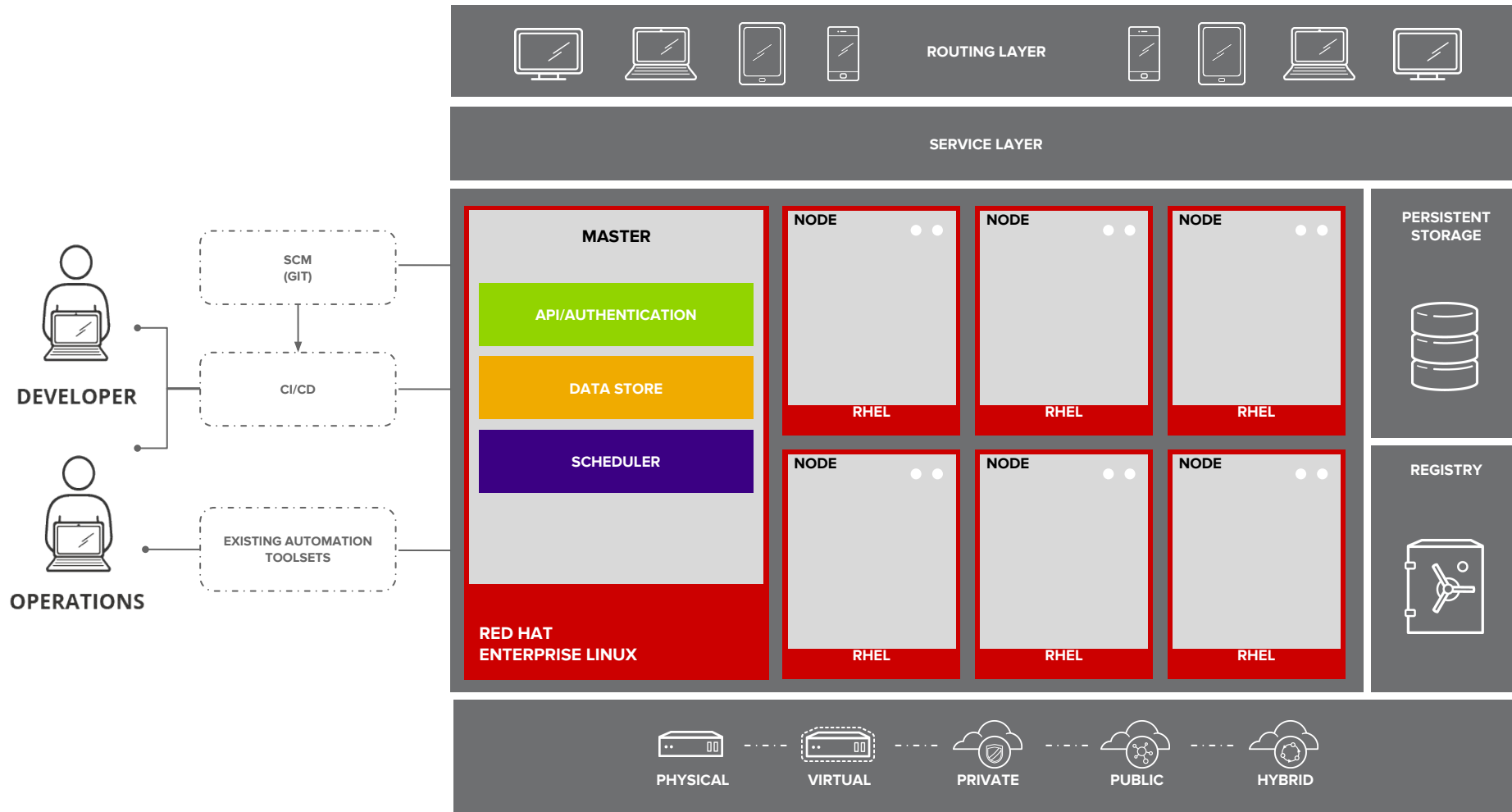
OPENS�AVA'18



Cloud Native Applications



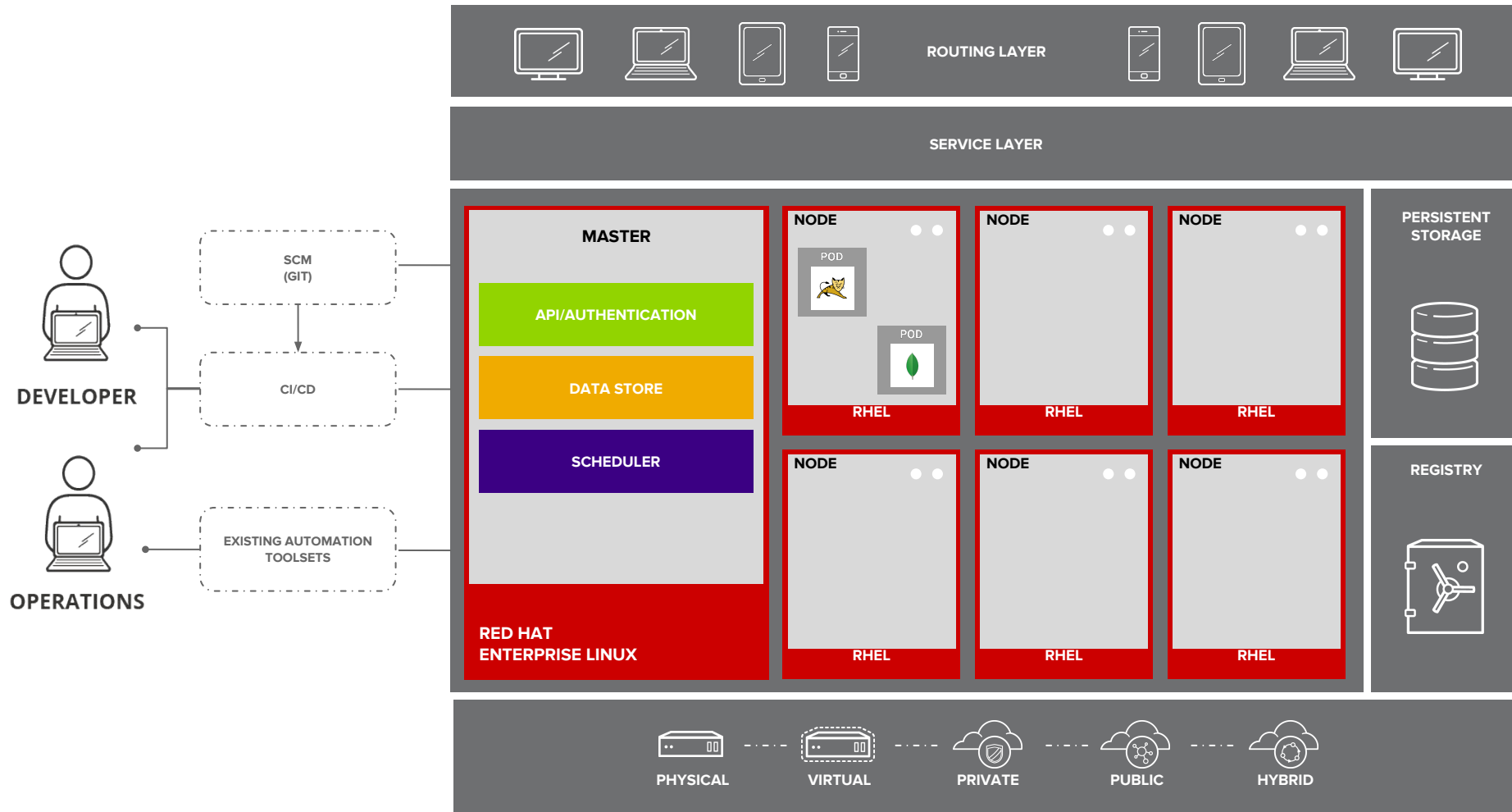
OPENS�AVA'18



Cloud Native Applications



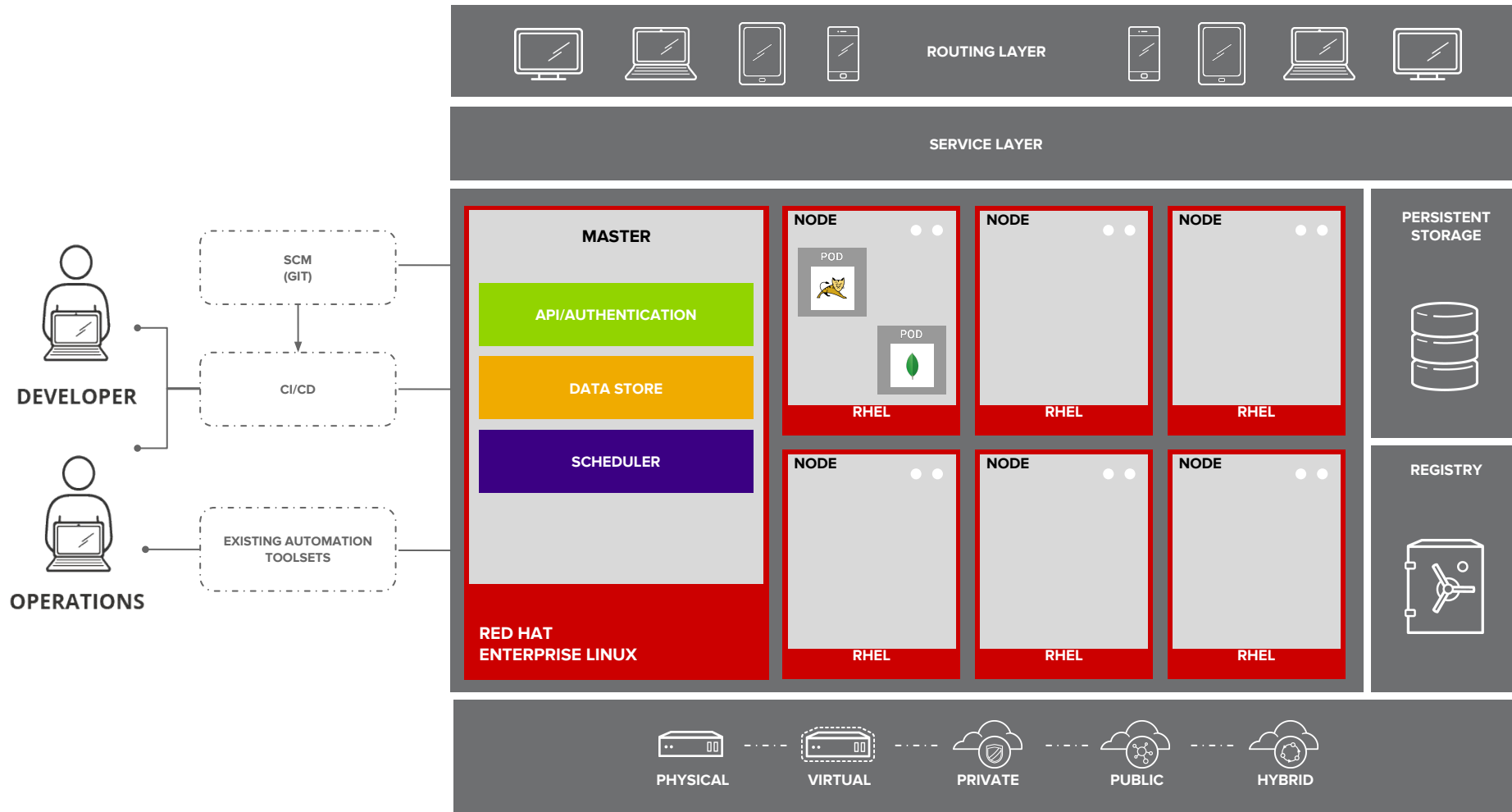
OPENS�AVA'18



Cloud Native Applications



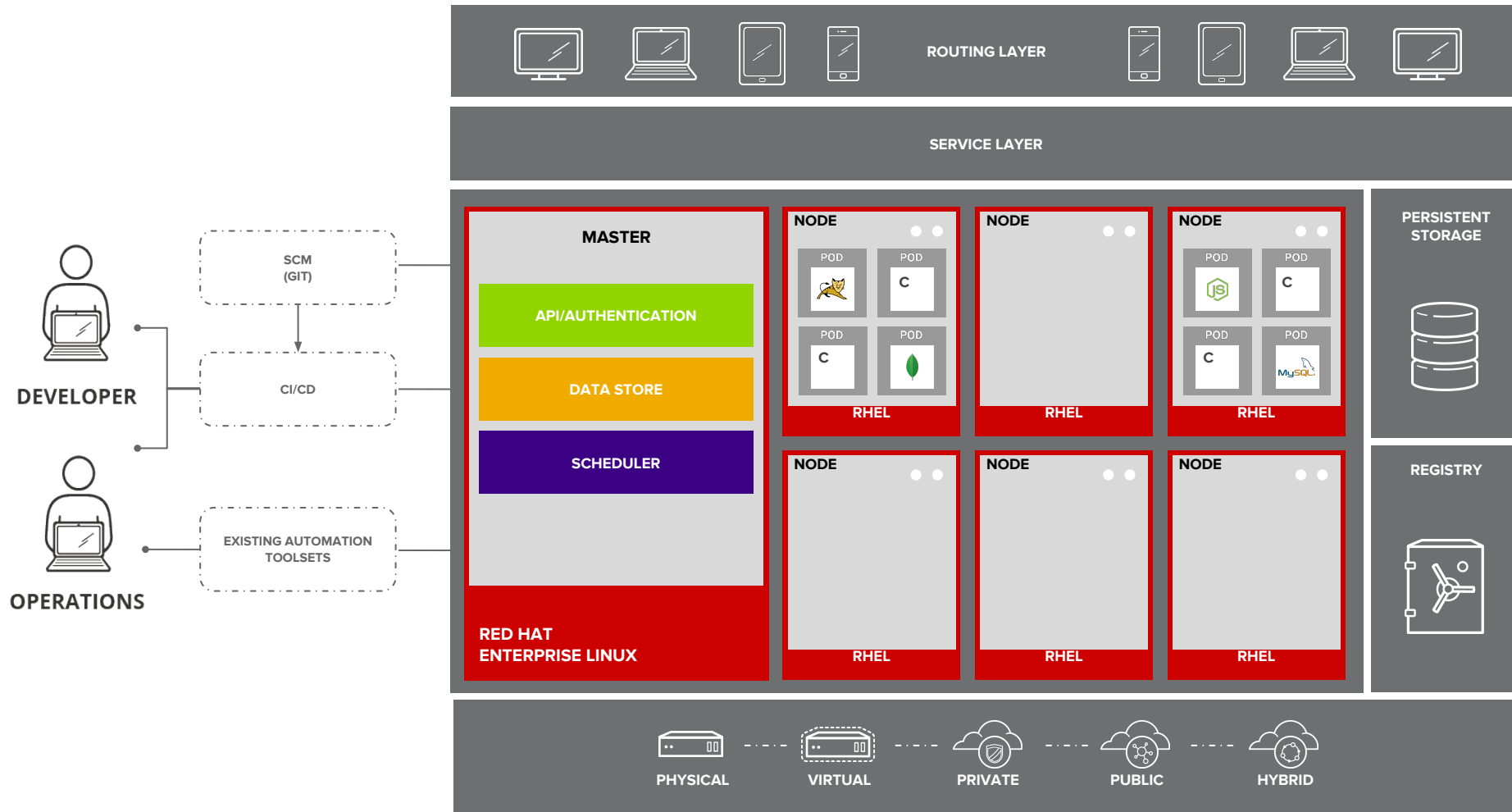
OPENS�AVA'18



Cloud Native Applications



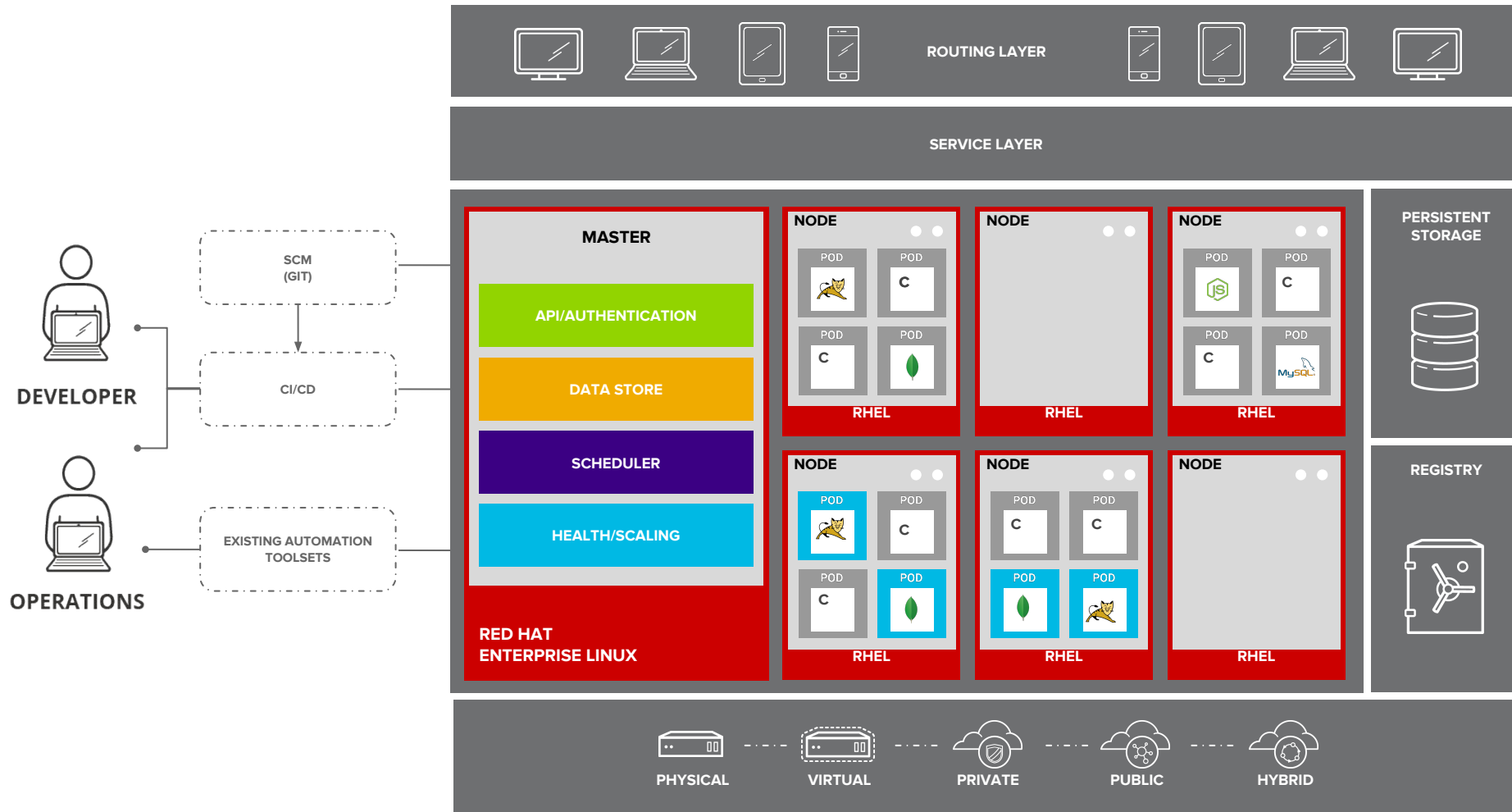
OPENS�AVA'18



Cloud Native Applications



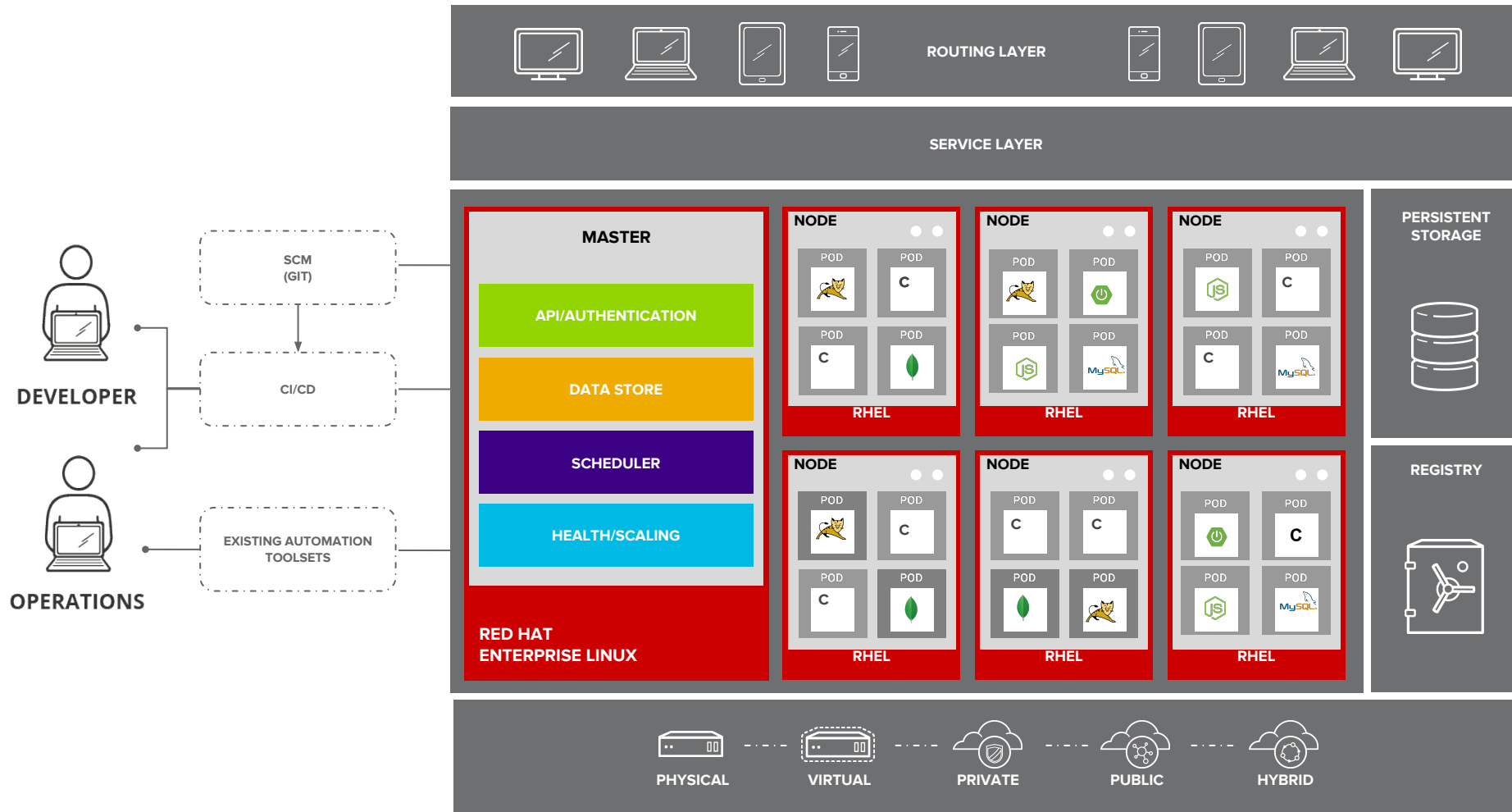
OPENS�AVA'18



Cloud Native Applications



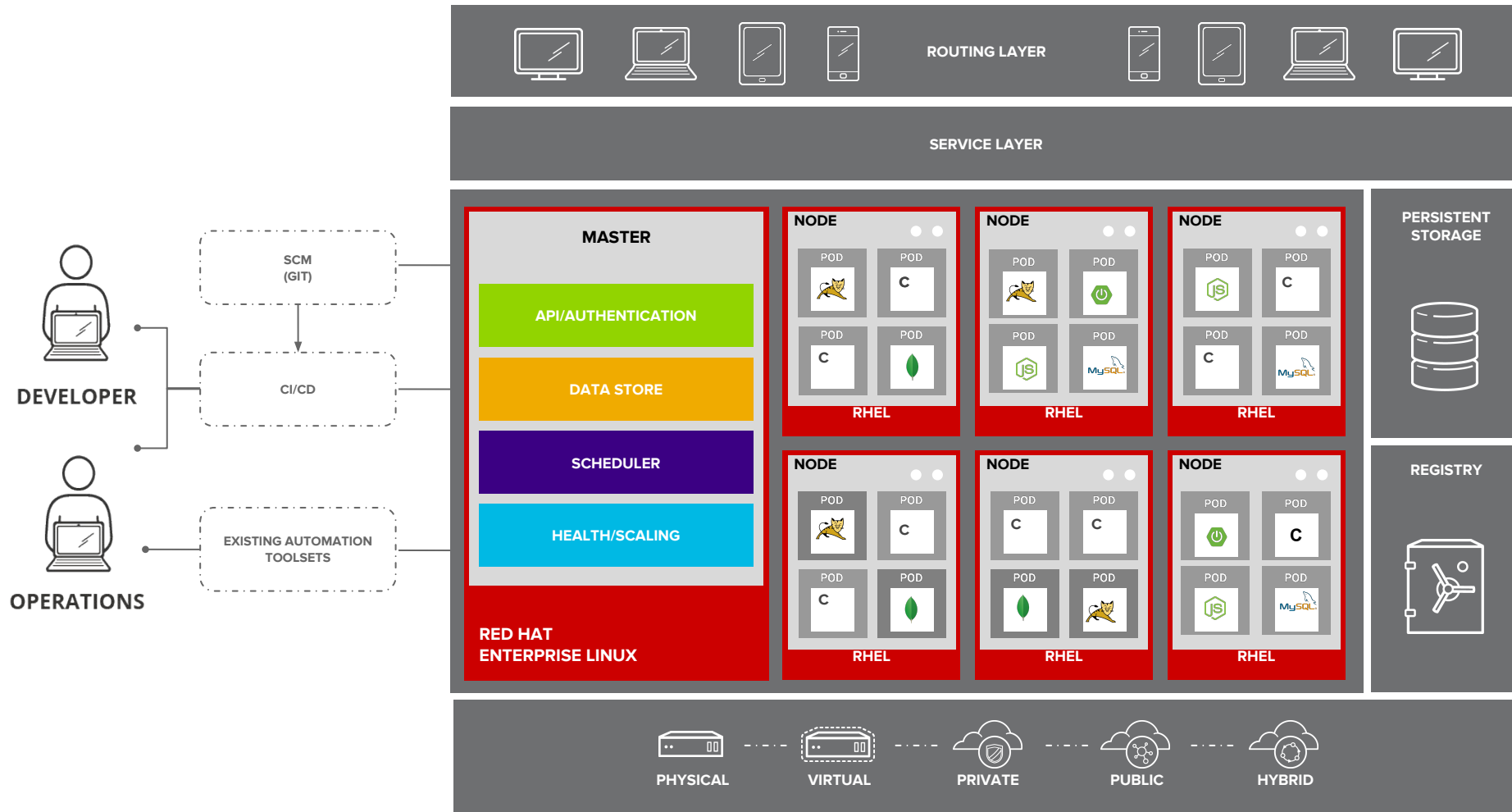
OPENS�AVA'18



Cloud Native Applications



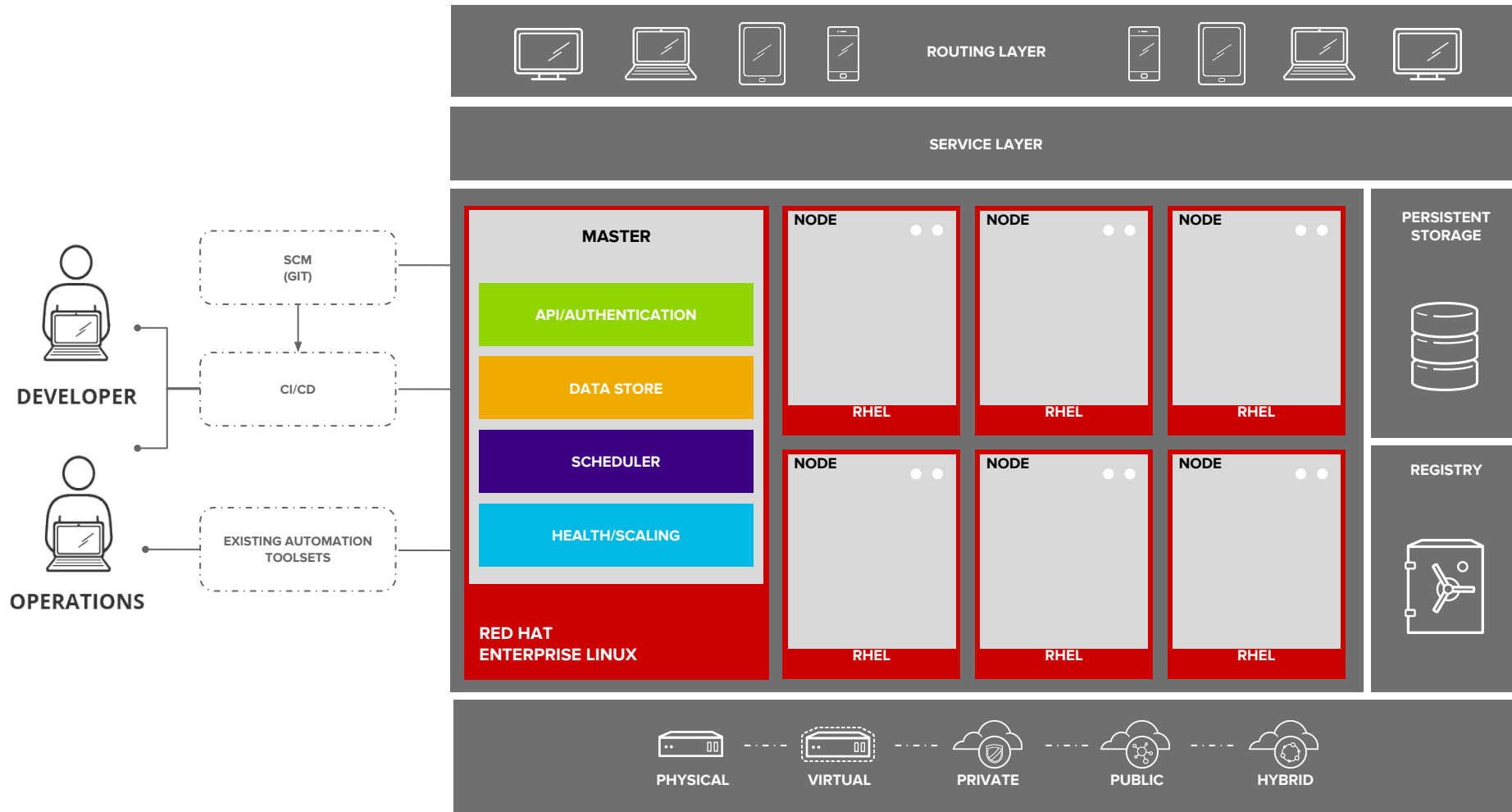
OPENS�AVA'18



Cloud Native Applications



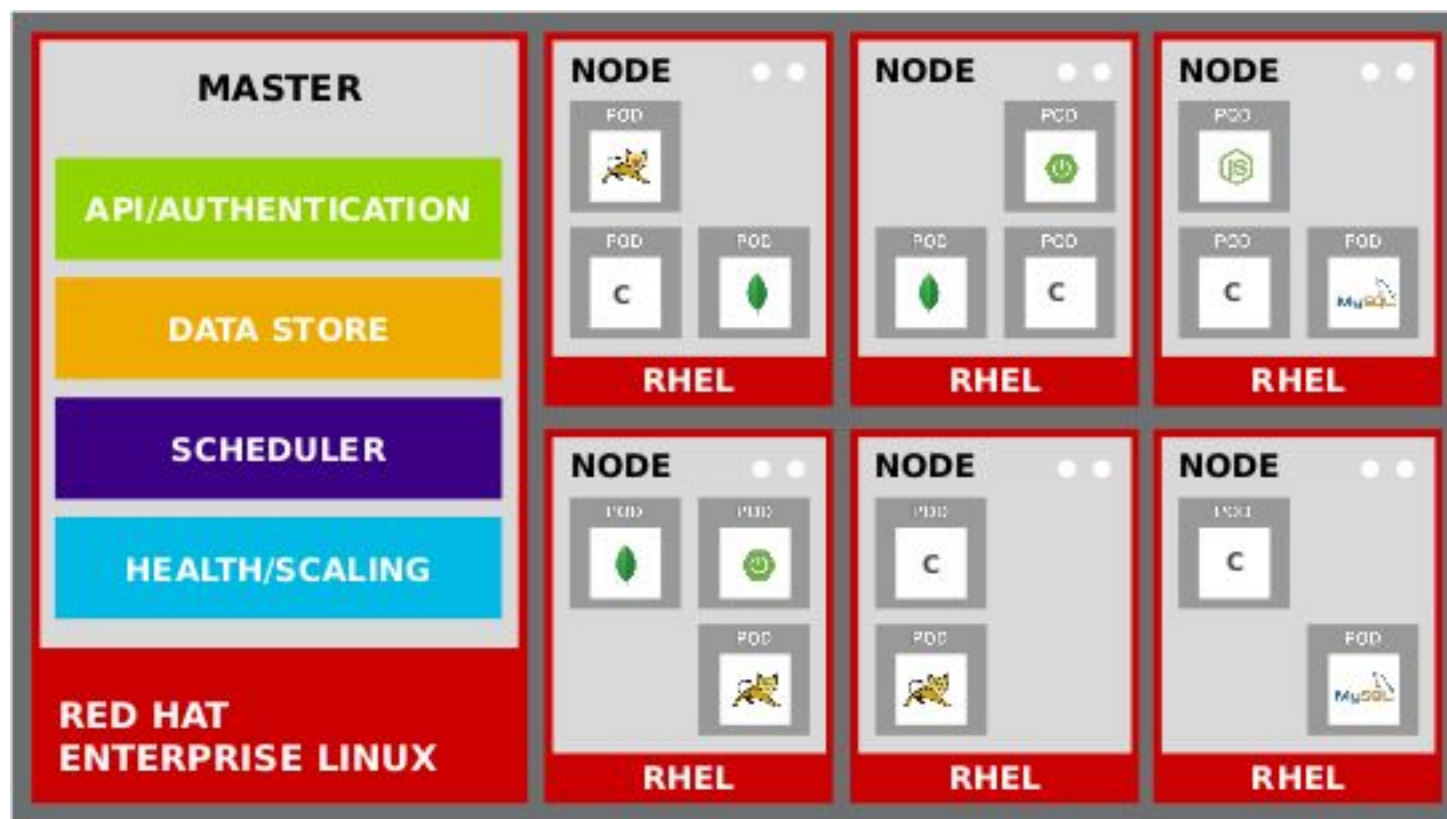
OPENS�AVA'18



Cloud Native Applications



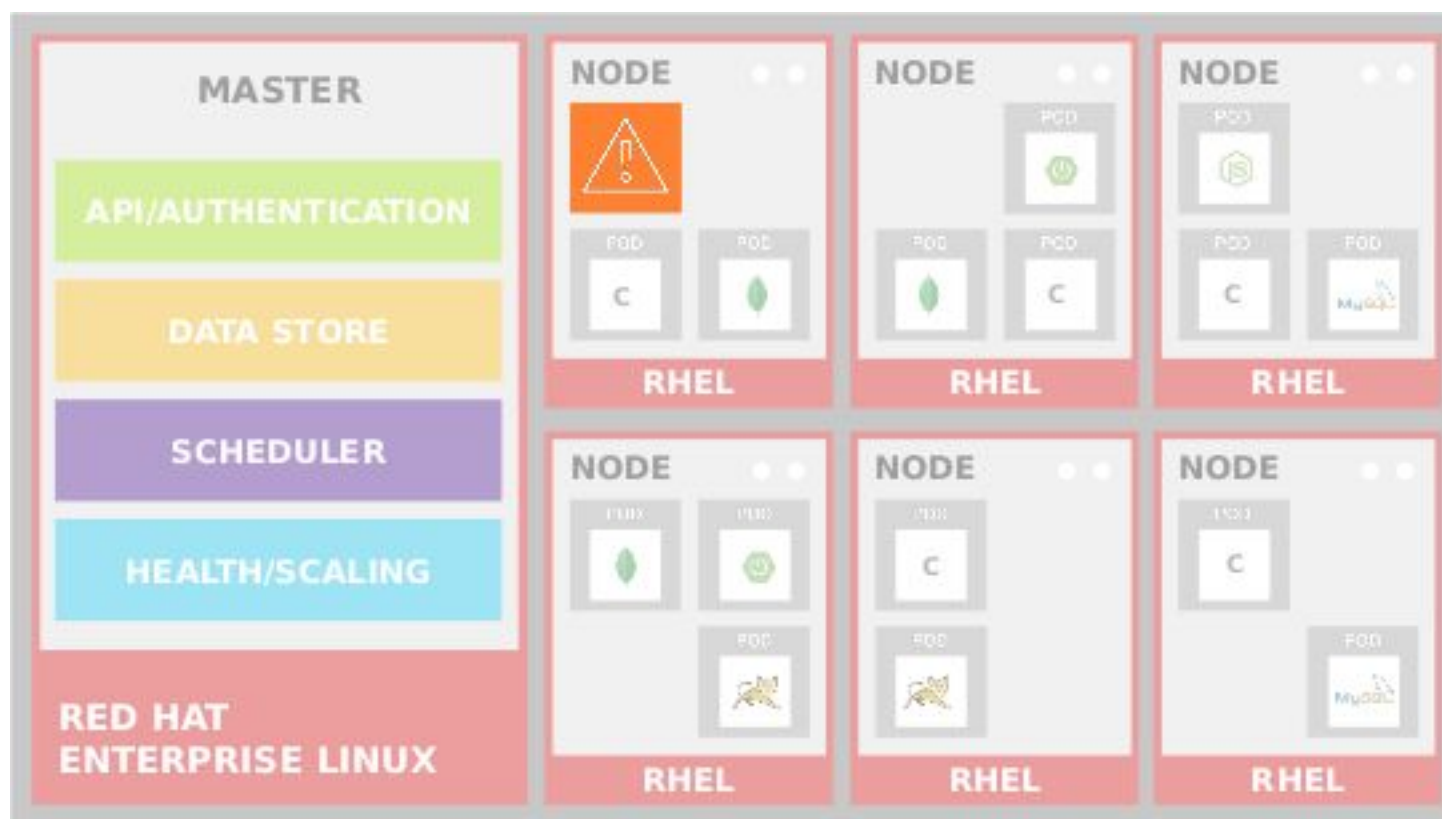
OPENS�AVA'18



Cloud Native Applications



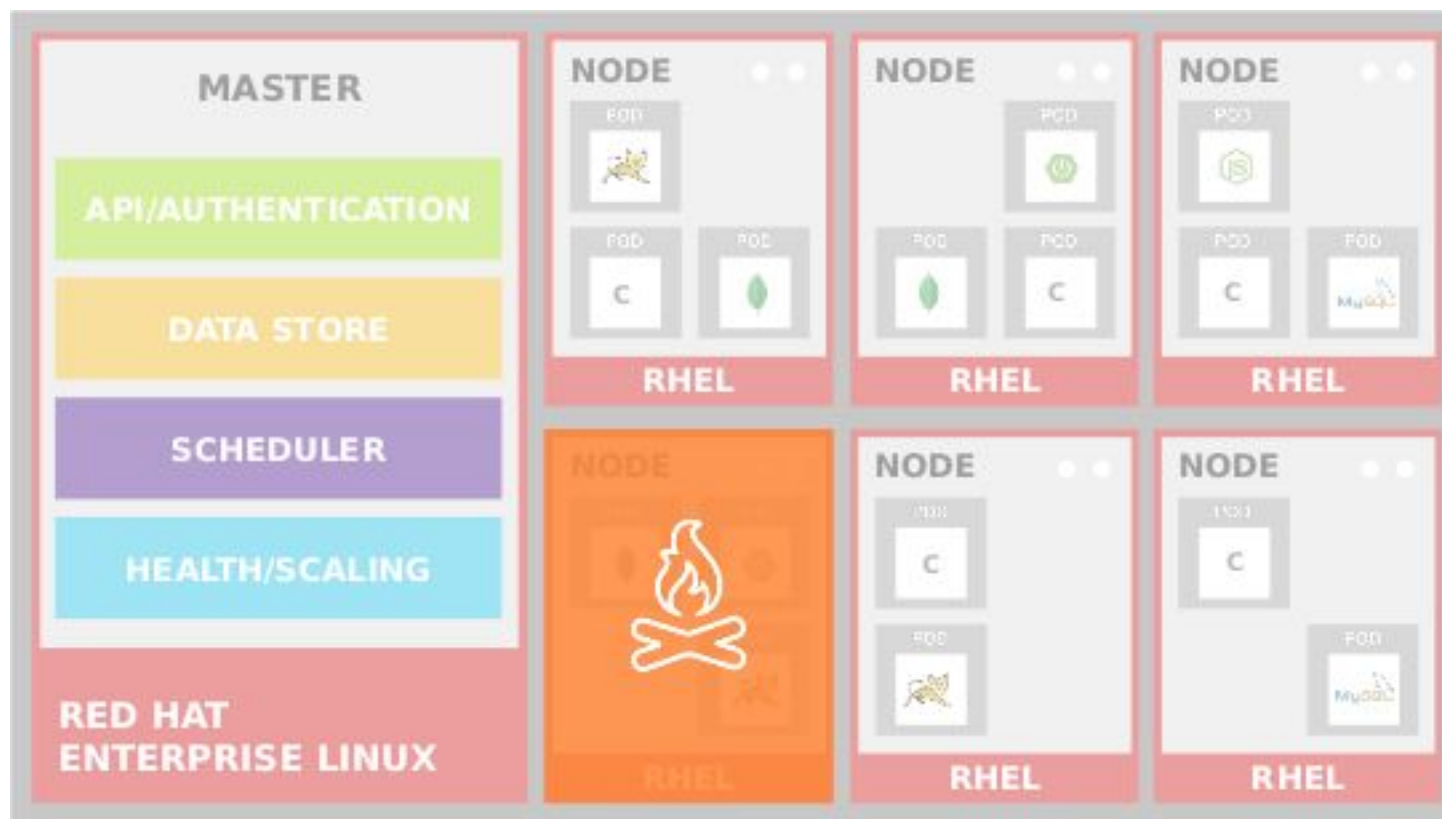
OPENS�AVA'18



Cloud Native Applications



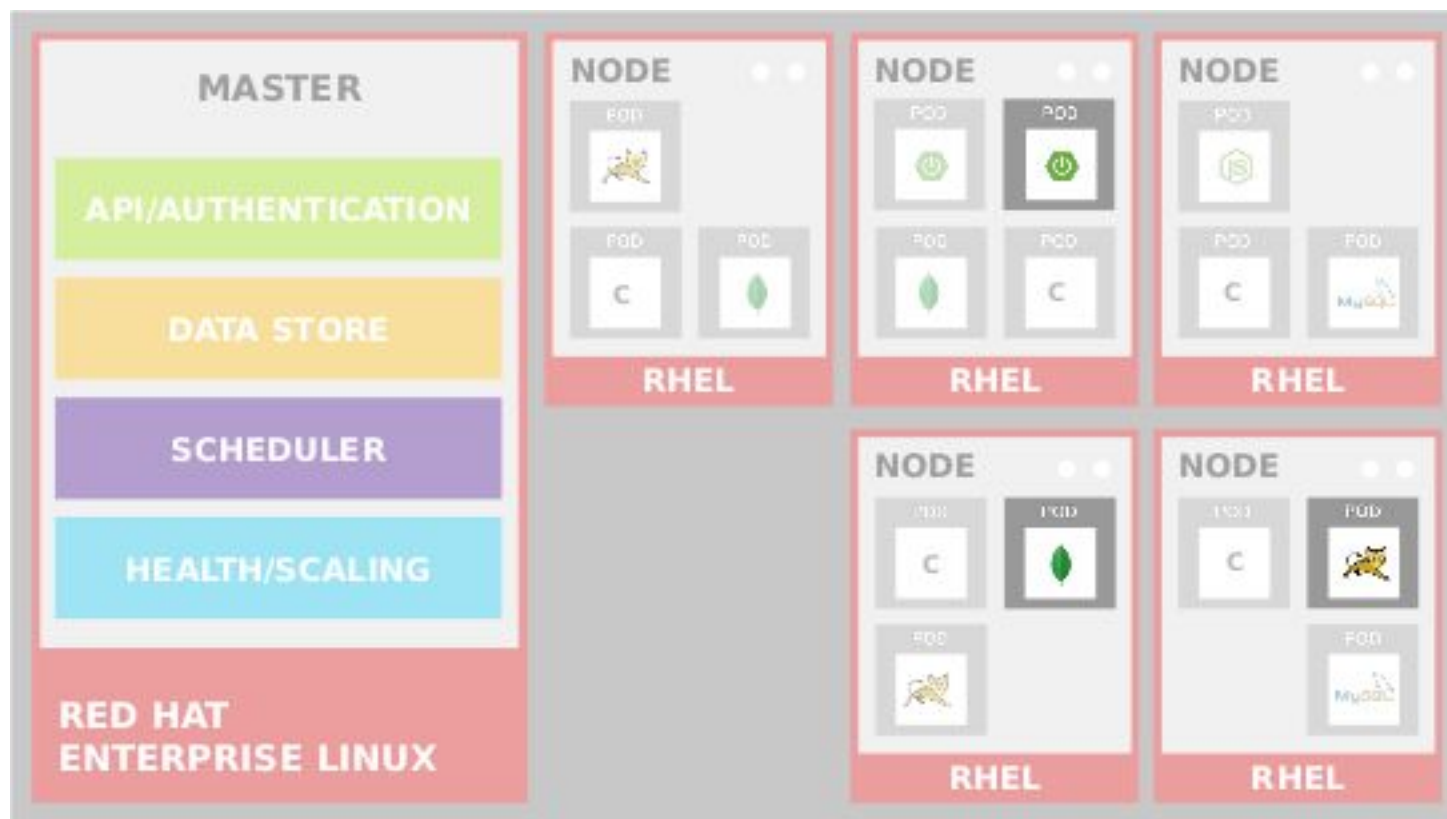
OPENS�AVA'18



Cloud Native Applications



OPENS�AVA'18



Cloud Native Applications

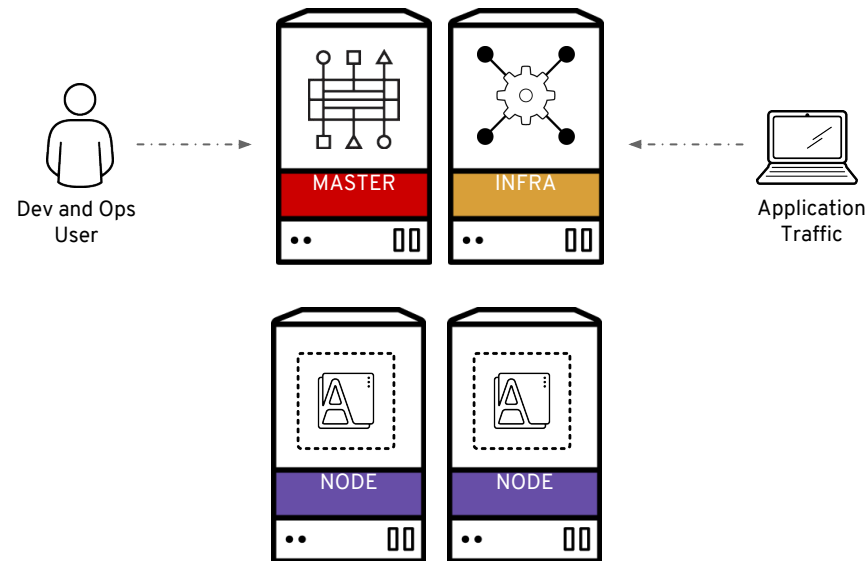


OpenShift Installation Architecture

Cloud Native Applications



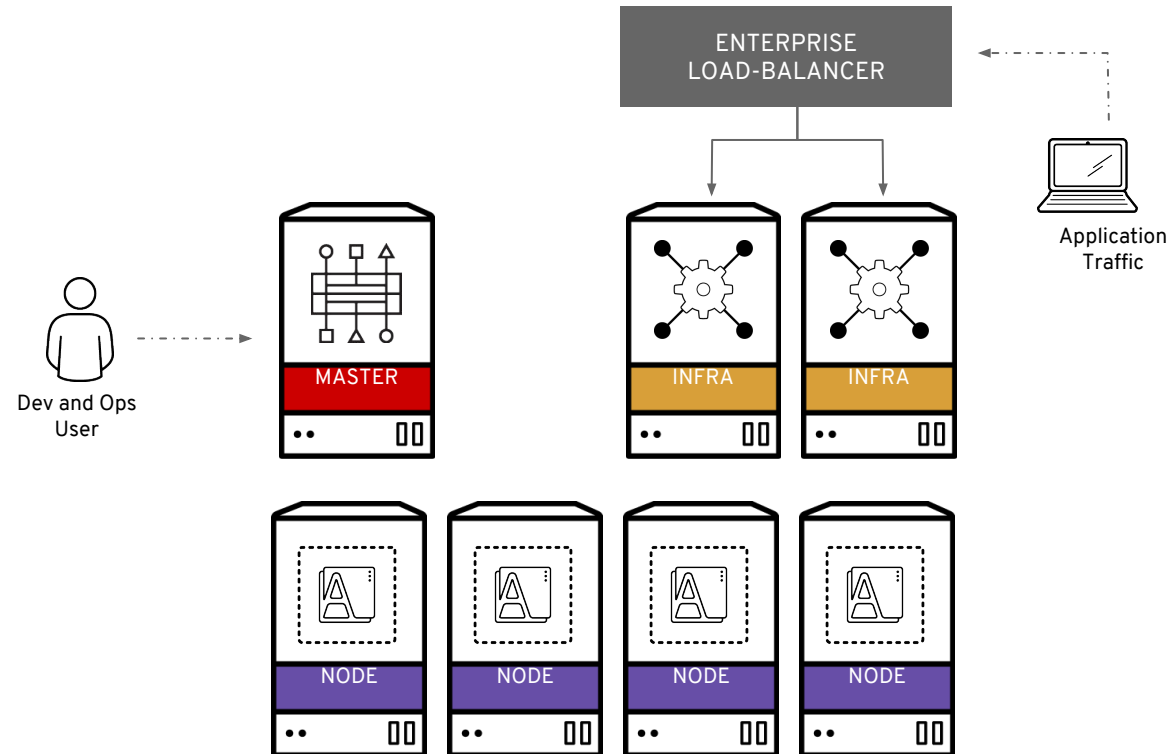
Proof-of-Concept Architecture



Cloud Native Applications



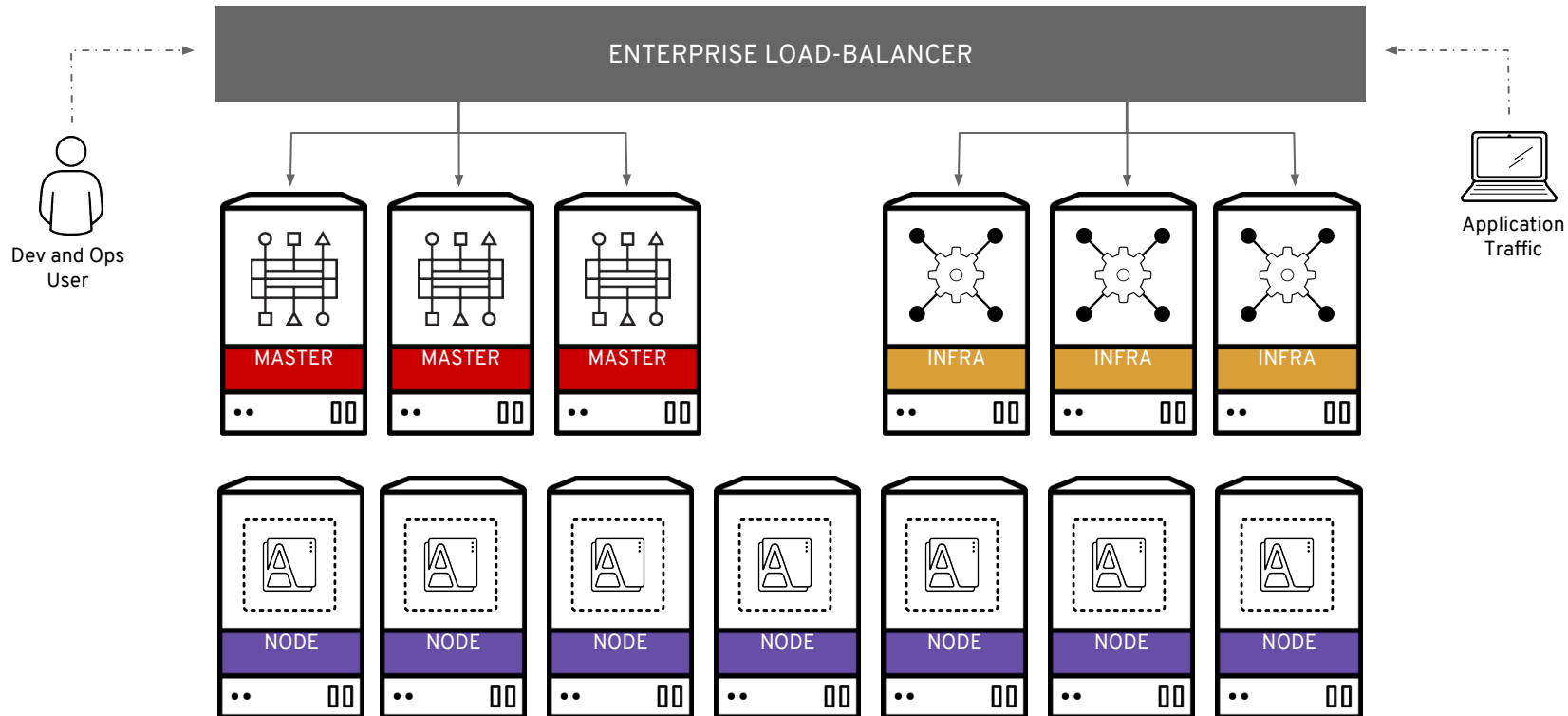
High-Availability Architecture



Cloud Native Applications



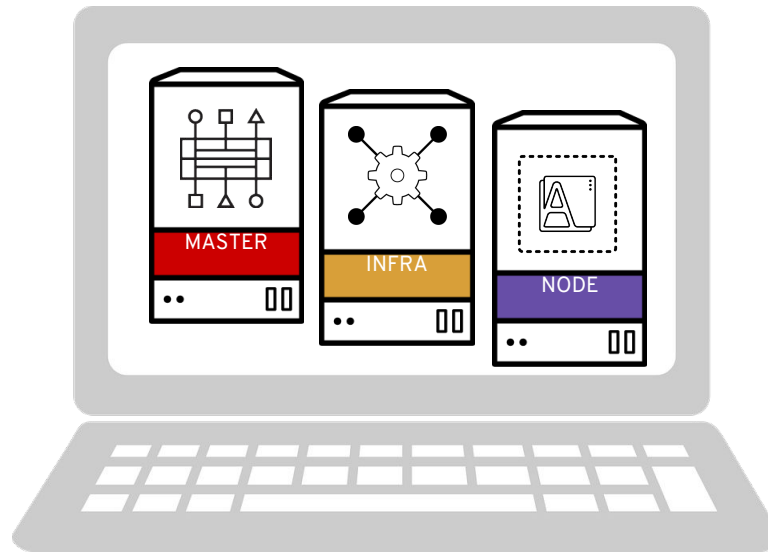
Full High-Availability Architecture



Cloud Native Applications



A laptop with at least 8GB of RAM to host the master, the infra and the compute nodes.



```
$ oc cluster up --logging=false --metrics=true ...
```

Cloud Native Applications



LAB and Q/A (anytime you want/need)

What you need?

Internet

Root/Admin access to your PC

Docker

JDK 1.8+

OC – OpenShift CLI tool

<https://github.com/openshift/origin/releases>

Cloud Native Applications



Presentation and labs available at:

<https://github.com/foogaro/openslava-2018>

Grazie

Ciao