Containers

Soluções Dell EMC utilizando Red Hat

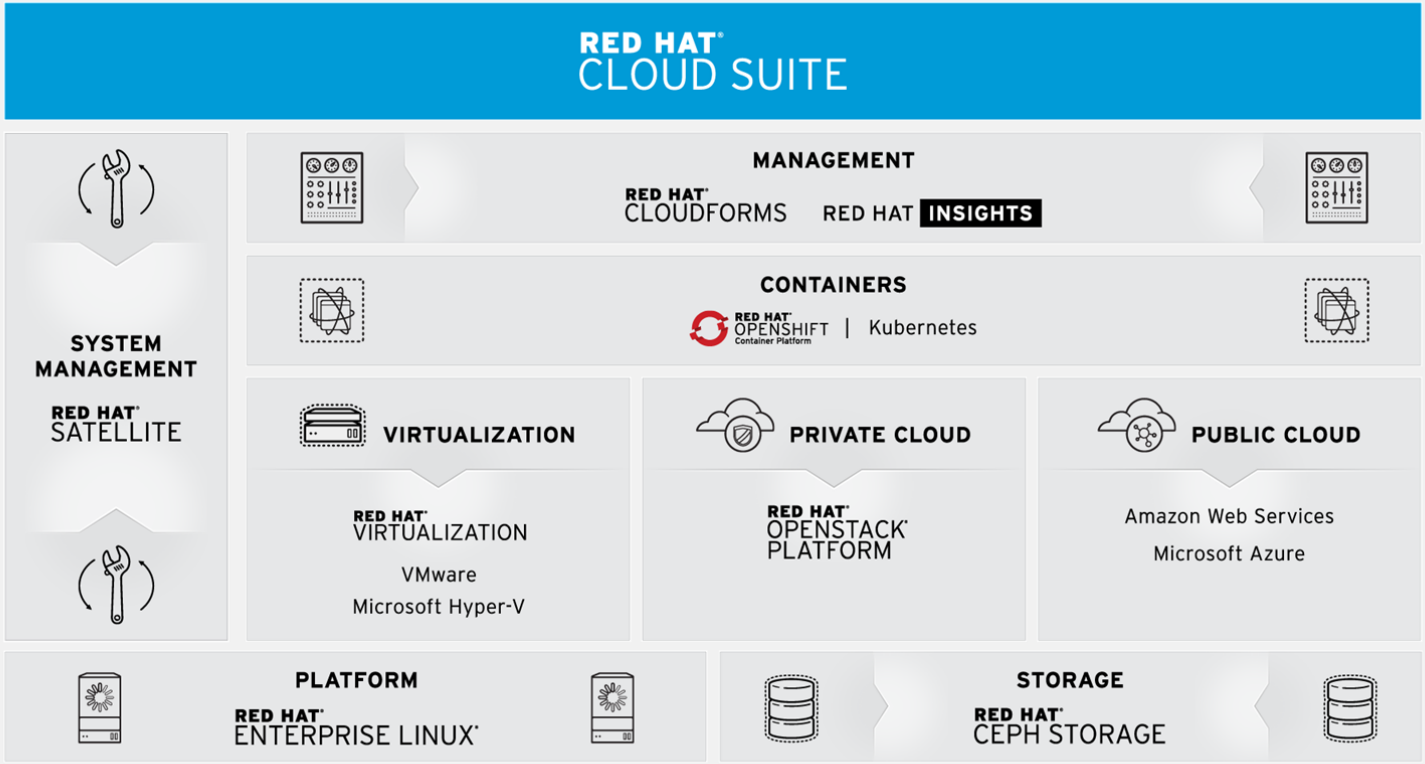
Estratégia de uso de containers, automação e gerência de infraestrutura

Caio Candido

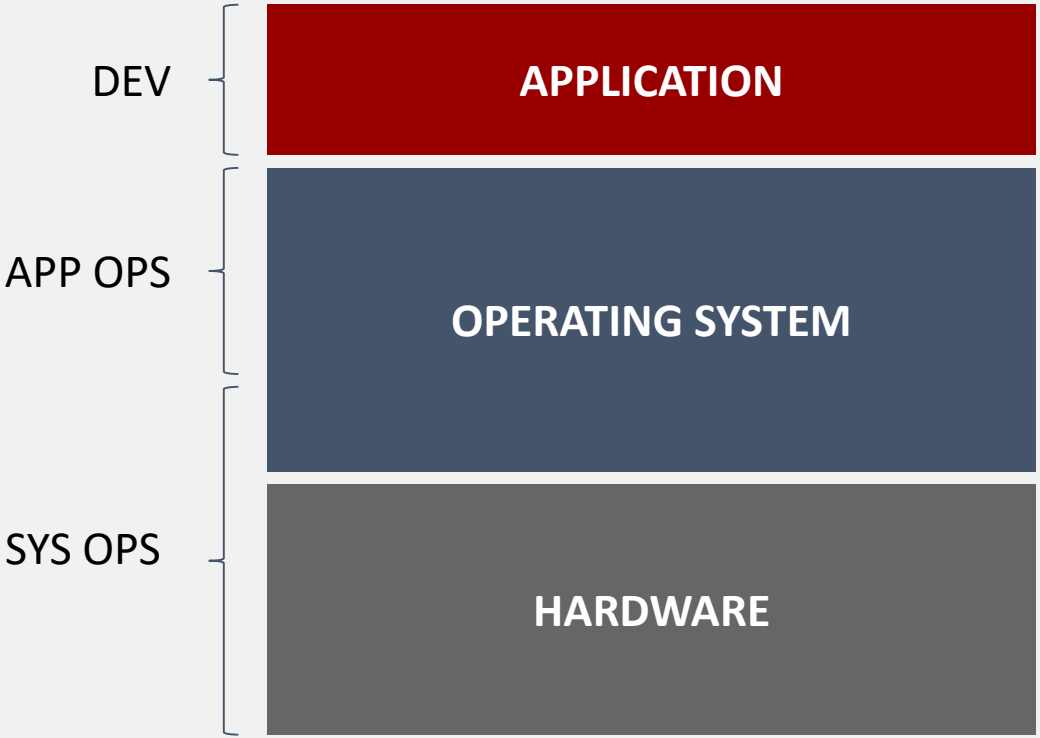
Senior Technology Strategist, Latin America Ready Solutions & Red Hat Specialist

powered by

Softwares para sua estratégia on premises e cloud

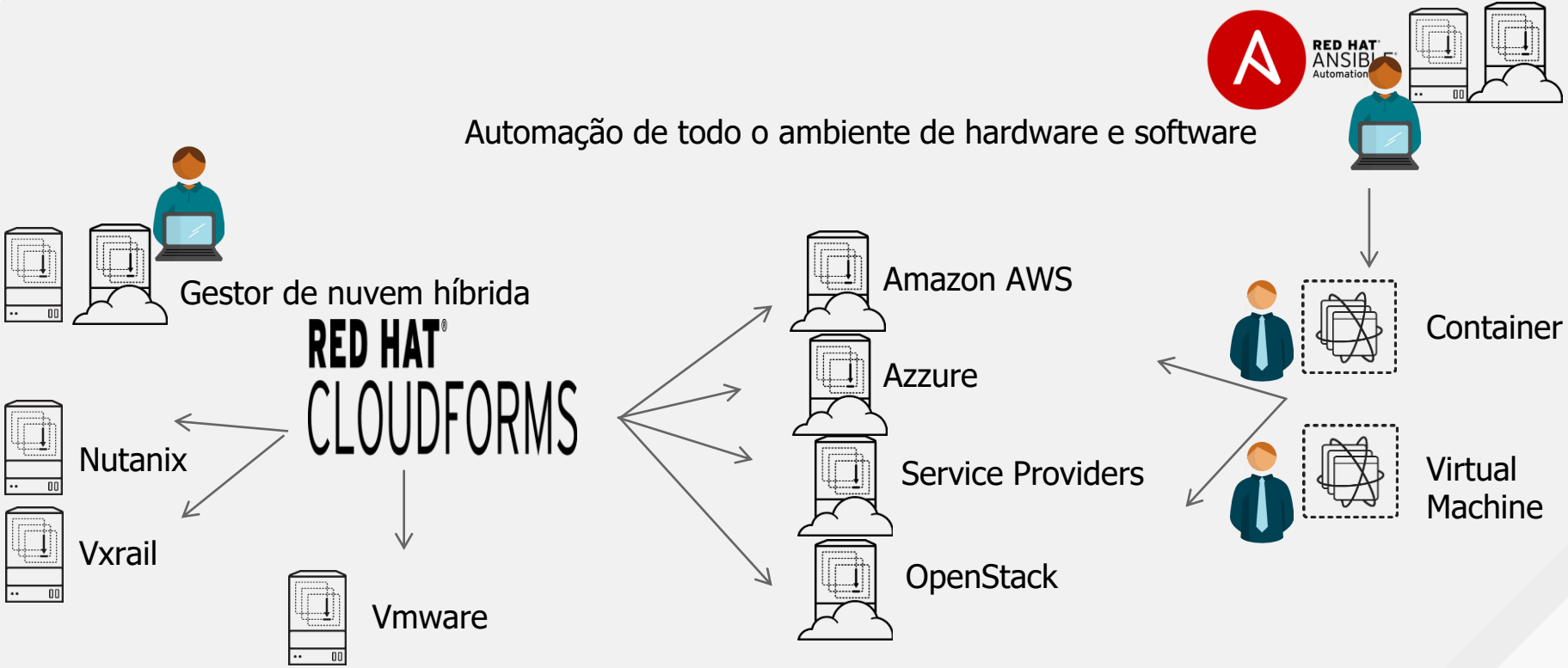


Definindo os time e esforços

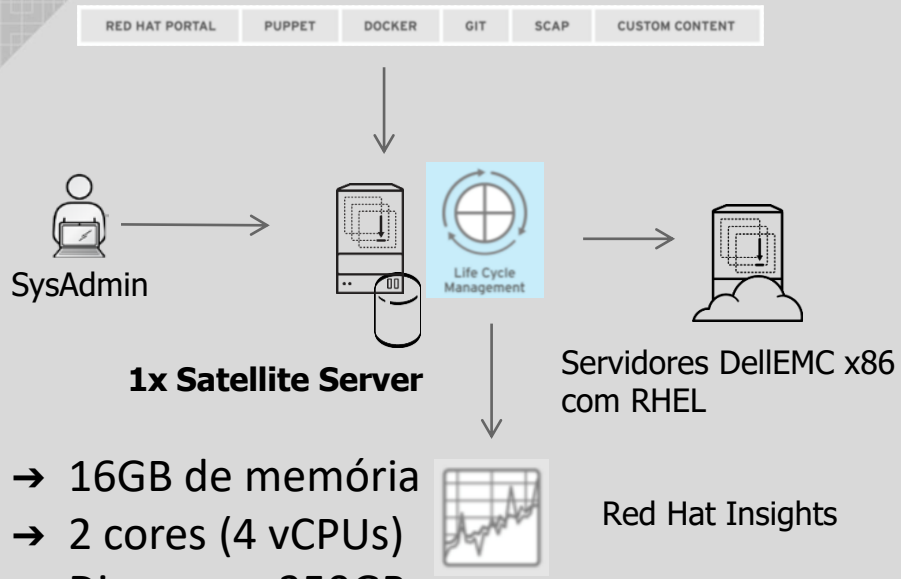


- Application Runtime
- Application Services
- Application Workflow
- Container Scheduling
- Container Orchestration
- Container Engine
- IaaS
- Operating System
- Virtual / Physical Infrastructure

Satellite+ CloudForms + Ansible



Arquitetura inicial – Satellite



- ➔ 16GB de memória
- ➔ 2 cores (4 vCPUs)
- ➔ Disco com 350GB
- ➔ vNIC x2 Network

Mount Point	Size	Função
/	20 GB	Arquivos do Sistema e Serviços
/var/log	4 GB	Logs do Sistema
/var	3 GB	Variáveis
/tmp	1 GB	Arquivos temporários do SO
/home	2 GB	Arquivos de usuários não admin
/var/cache/pulp	30 GB	Temporário usado durante sincronização
/var/lib/qpidd	10 GB	Suporte para até 30.000 content hosts
/var/lib/pgsql	10 GB	Banco de Dados
/var/lib/pulp	200 GB	Repositórios
/var/lib/mongodb	50 GB	Base para conteúdo do Satellite
/opt	5 GB	Serviços Opcionais

DELL EMC redhat

CloudForms

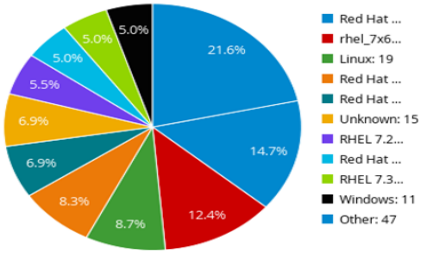
RED HAT CLOUDFORMS MANAGEMENT ENGINE

Cloud Operators | CF42_ui2

- Cloud Intel
- Red Hat Insights
- Services
- Compute
- Configuration
- Networks
- Middleware
- Storage
- Control
- Automate
- Optimize

Global | VMware | Cloud | Containers | Health | Compliance | Consumption

Guest OS Information



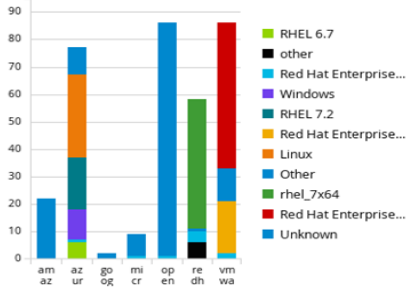
Updated March 07, 2017 00:00 | Next March 08, 2017 00:00

Top CPU Consumers (Last Hour)

VM Name	CPU Usage	Allocated vCPUs	VM Vendor
overcloud1-telus	21.4%	8	vmware
manageiq-euwe-2	18.3%	4	redhat
manageiq-euwe-3	14.0%	4	redhat
Lenovo XClarity Adminis trator - Do not delete	9.1%	2	vmware
vcenter6	8.0%	4	vmware

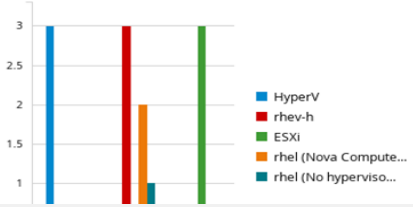
Updated December 21, 2016 20:17 | Next March 07, 2017 08:45

Guest OS Count



Updated February 02, 2017 21:49 | Next March 07, 2017 09:00

Virtual Infrastructure Platforms



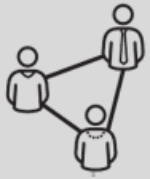
Projects by Number of Pods Widget

Project Name	Number of Pods
demo-project	12
demo-project	7
default	3
openshift-infra	3

Vendor and Guest OS Chart

DELL EMC redhat

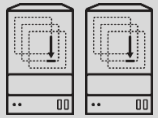
Arquitetura inicial - CloudForms



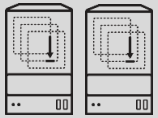
Usuários

SysAdmin

Balanceador de Carga

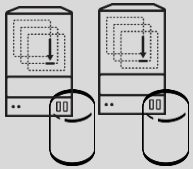


Master UI



4x CFME: workers

- ➔ 12GB de memória
- ➔ 2 cores (4 vCPUs)
- ➔ Disco com 60GB



2x Master Database

- ➔ 16GB de memória
- ➔ 2 cores (4 vCPUs)
- ➔ Disco com 60GB
- ➔ Disco adicional de 100GB DB

DELL EMC

redhat.

Como o Ansible pode fazer parte da sua estartégia?

Ansible é um projeto de comunidade Open Source mantido pela Red Hat. É uma **linguagem de automação simples** que permite descrever perfeitamente sua infraestrutura como código, através de playbooks

Ansible é uma solução de automação que permite implementar o gerenciamento de configuração e orquestração.

Como começamos esta jornada?

Os diferenciais do Ansible

SIMPLES

Automação legível

Habilidades de codificação dispensáveis

Tarefas executadas em ordem

Rápida produtividade

PODEROSO

Deploy de aplicação

Gerenciamento de configurações

Orquestração

Automação de rede

Orquestra ciclo de vida da aplicação

SEM AGENTE

Arquitetura sem agente

OpenSSH e WinRM

Sem atualização de agente

Começo imediato

Maior eficiência e segurança

As diferenças entre Ansible e Ansible Tower

O QUE É O ANSIBLE TOWER?

Ansible Tower é um **framework corporativo** para controlar, garantir e gerenciar sua automação Ansible - com uma **UI** e **API RESTful**.

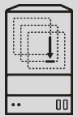
- **Controle de acesso** baseado em **roles**
- **Deploy** completo de aplicações pressionando **botão de deployment**
- Todas automações com **log centralizado**

Principais Diferencias do Ansible Tower Trabalhando com DevOps

Dashboard Integrado	Criação de Playbooks automatizados
Gerenciamento das Tarefas	Execução remota
Serviços de Mensagens	Acesso e Segurança garantida
Agendamento das Tarefas	Visibilidade de tudo
Auto Escaling e inventários	Suporte e Desenvolvimento Global

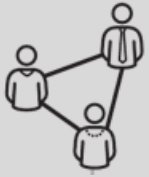
Arquitetura inicial - Ansible

SysAdmin



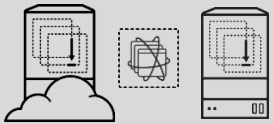
Git Server/ repositório

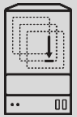
Usuários



Automação do Ambiente x86

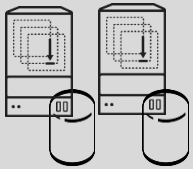
Servidores DellEMC x86 com RHEL





1x Ansible Tower

- ➔ 8GB de memória
- ➔ 1 cores (4 vCPUs)
- ➔ Disco com 60GB

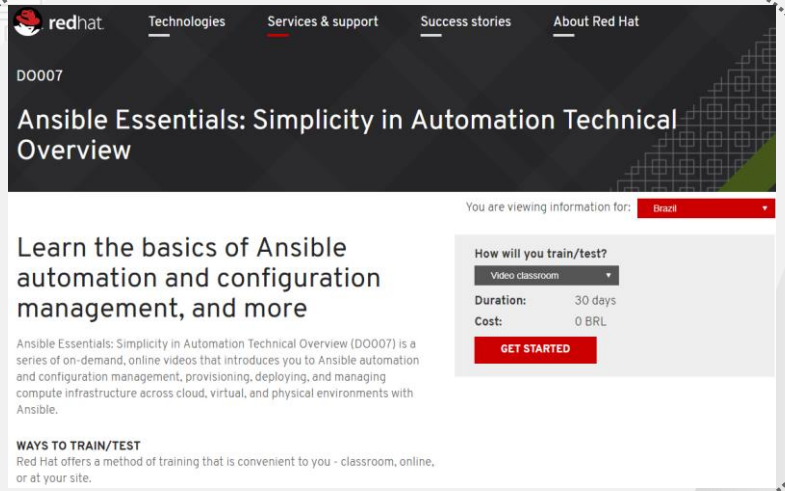


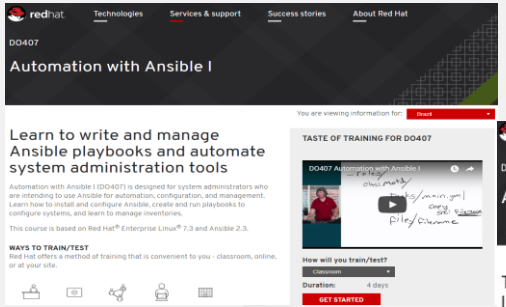
1x Ansible Database

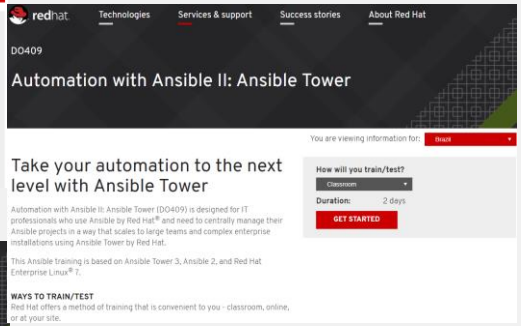
- ➔ 16GB de memória
- ➔ 2 cores (8 vCPUs)
- ➔ Disco adicional de 120GB DB

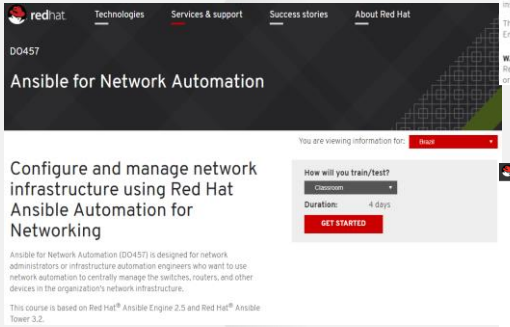
DELL EMC redhat.

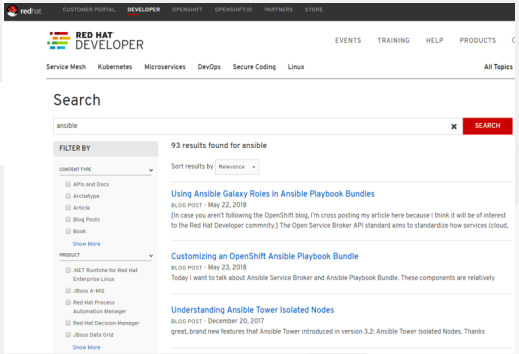
Como posso obter mais conhecimento?



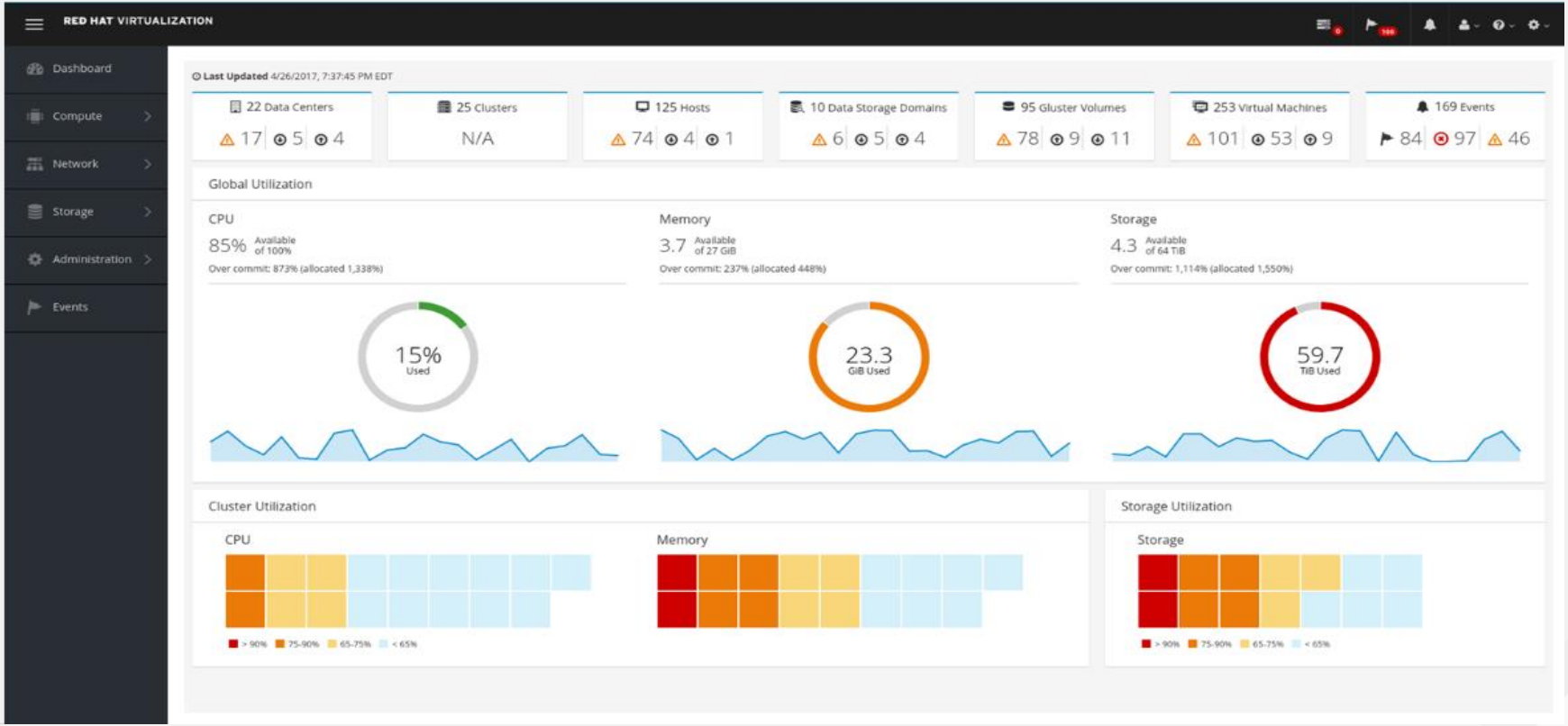






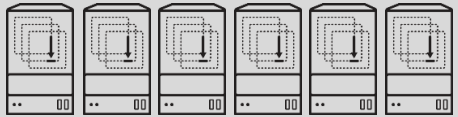


RHV-M


RED HAT VIRTUALIZATION
Dashboard
Compute
Network
Storage
Administration
Events
Last Updated 4/26/2017, 7:37:45 PM EDT
22 Data Centers
17 5 4
25 Clusters
N/A
125 Hosts
74 4 1
10 Data Storage Domains
6 5 4
95 Gluster Volumes
78 9 11
253 Virtual Machines
101 53 9
169 Events
84 97 46
Global Utilization
CPU
85% Available of 100%
Over commit: 873% (allocated 1,338%)
15% Used
Memory
3.7 Available of 27 GiB
Over commit: 237% (allocated 448%)
23.3 GiB Used
Storage
4.3 Available of 64 TiB
Over commit: 1,114% (allocated 1,550%)
59.7 TiB Used
Cluster Utilization
CPU
Memory
> 90% 75-90% 65-75% < 65%
Storage Utilization
Storage
> 90% 75-90% 65-75% < 65%

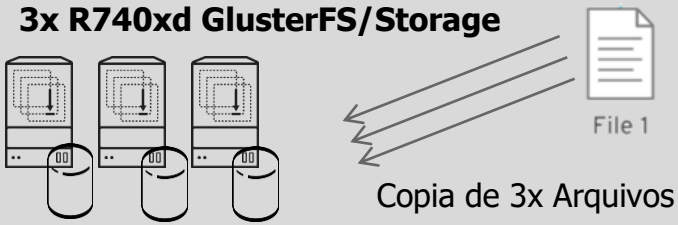
Arquitectura Default disponible para toda LATAM con X86

6x R640 Infra Nodes com RHV



- Servidor 14ª servidores DellEMC PowerEdge **R640**
- **2x Processador Intel Xeon Gold 6148**
- **128GB RAM** | 4 x pentes de memória de 128GB RDIMM, Dual Rank
- **2 x Discos de 2.4TB 10K RPM SAS 12Gbps 512e 2.5in Hot-plug Hard Drive**
- Tarjeta de control BOSS + com 2 M.2 Sticks 240G (RAID 1),FH
- Control PERC HBA330 RAID, 12Gb Minicard - **NO RAID**
- **OpenStack Cloud Compute NOD**
- **Network Intel X710 DP 10Gb DA/SFP+, + I350 DP 1Gb Ethernet**, Network Configuración de RAID no configurada

3x R740xd GlusterFS/Storage

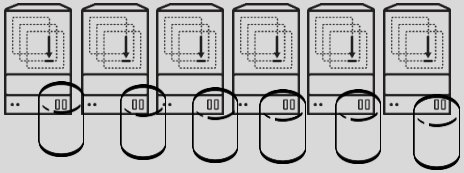


- Servidor 14ª servidores DellEMC PowerEdge **R740XD**
- **2x Processador Intel Xeon Gold 5118**
- **128GB RAM** | 4 x pentes de memória de 128GB RDIMM, Dual Rank
- **16 x Discos de 8TB 10K RPM SAS 12Gbps 512e 3.5in Hot-plug Hard Drive**
- Tarjeta de control BOSS + com 2 M.2 Sticks 240G (RAID 1),FH
- Control PERC HBA330 RAID, 12Gb Minicard - **NO RAID**
- **OpenStack Cloud Compute NOD**
- **Intel X710 Quad Port 10Gb Direct Aattach, SFP+, Converged Network Adapter**
- **Network Intel X710 DP 10Gb DA/SFP+, + I350 DP 1Gb Ethernet**, Network Configuración de RAID no configurada

DELL EMC redhat.

Arquitectura Hiperconvergente disponible para toda LATAM con x86

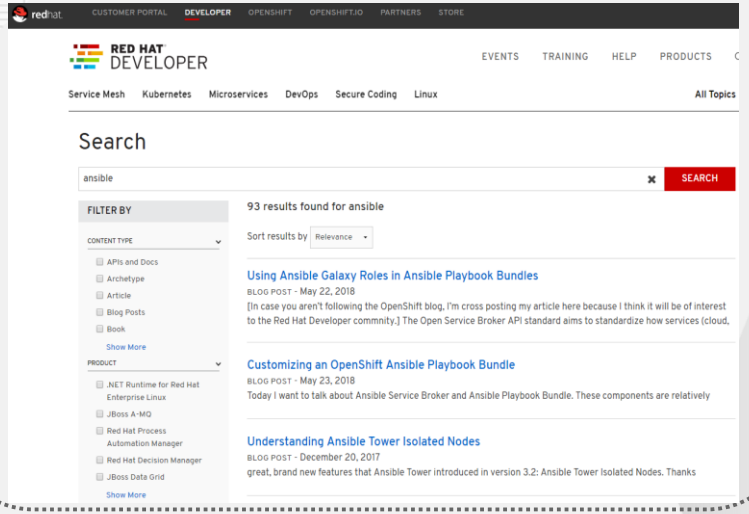
6x R740xd Infra Nodes com RHV



Copia de 3x Arquivos

- Servidor 14ª servidores DellEMC PowerEdge **R740XD**
- **2x Processador Intel Xeon Gold 6148**
- **128GB RAM** | 4 x pentes de memória de 128GB RDIMM, Dual Rank
- **18 x Discos de 2.4TB 10K RPM SAS 12Gbps 512e 2.5in Hot-plug Hard Drive**
- Tarjeta de control BOSS + com 2 M.2 Sticks 240G (RAID 1),FH
- Control PERC HBA330 RAID, 12Gb Minicard - **NO RAID**
- **Intel X710 Quad Port 10Gb Direct Aattach, SFP+, Converged**
- **Network Intel X710 DP 10Gb DA/SFP+, + I350 DP 1Gb Ethernet**,
- Configuración de RAID no configurada

Como obter mais conhecimento?



https://www.redhat.com/en/services/training/rh024-red-hat-linux-technical-overview

https://www.redhat.com/en/services/training/rh018-virtualization-and-infrastructure-migration-technical-overview

https://www.redhat.com/en/services/training/do092-developing-cloud-native-applications-microservices-architectures

https://www.redhat.com/en/services/training/do080-deploying-containerized-applications-technical-overview

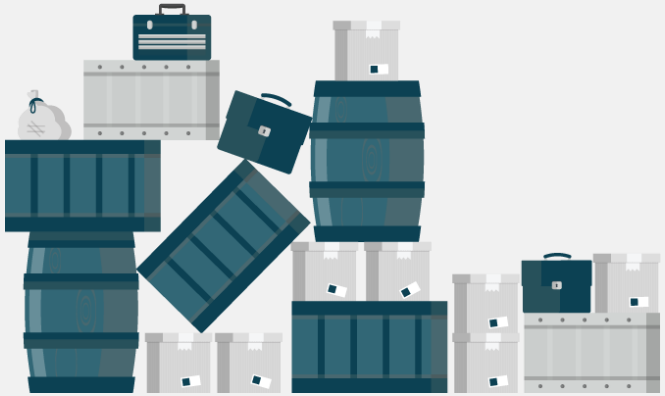
https://www.redhat.com/en/services/training/do007-ansible-essentials-simplicity-automation-technical-overview

https://www.redhat.com/en/services/training/cl010-red-hat-openstack-technical-overview

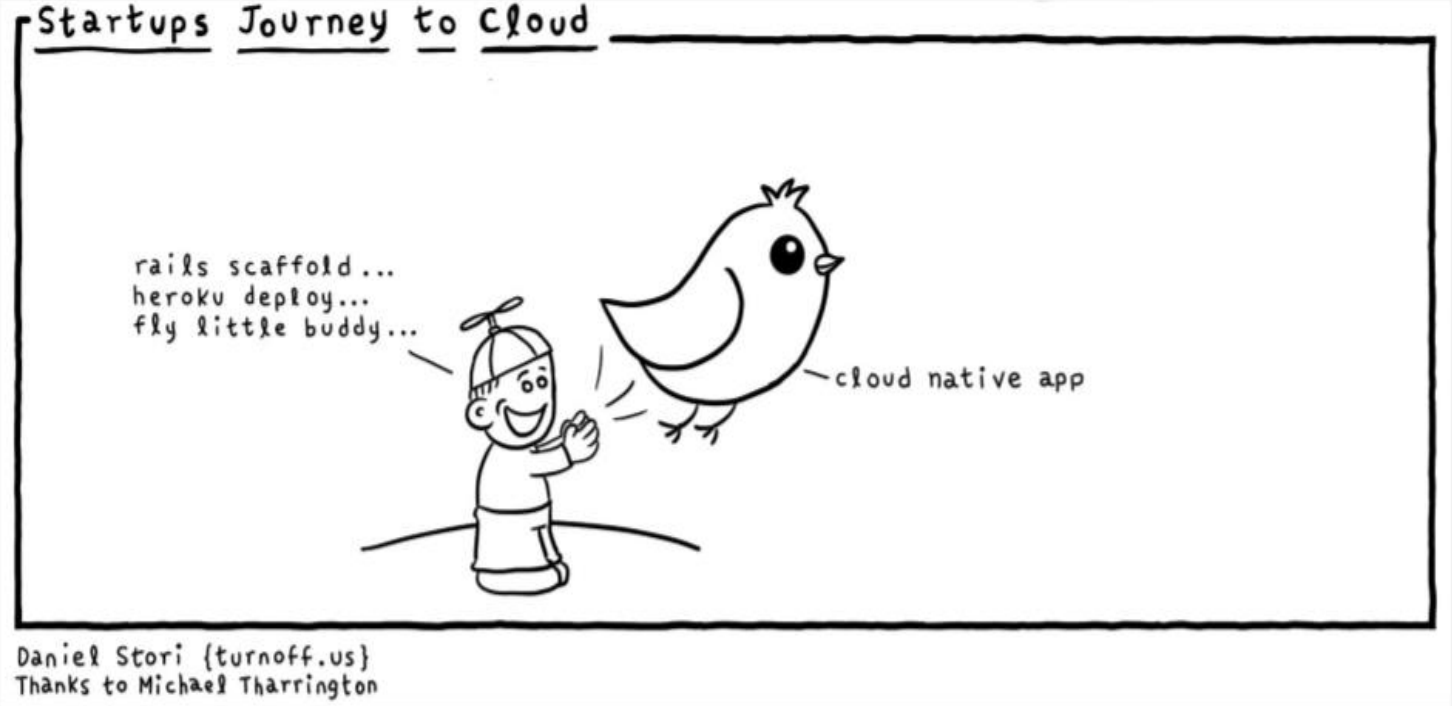
Os Problemas na engenharia de software

Aplicações possuem um processo de instalação, configuração e integrações complexo que deve ser realizado a cada novo deploy.

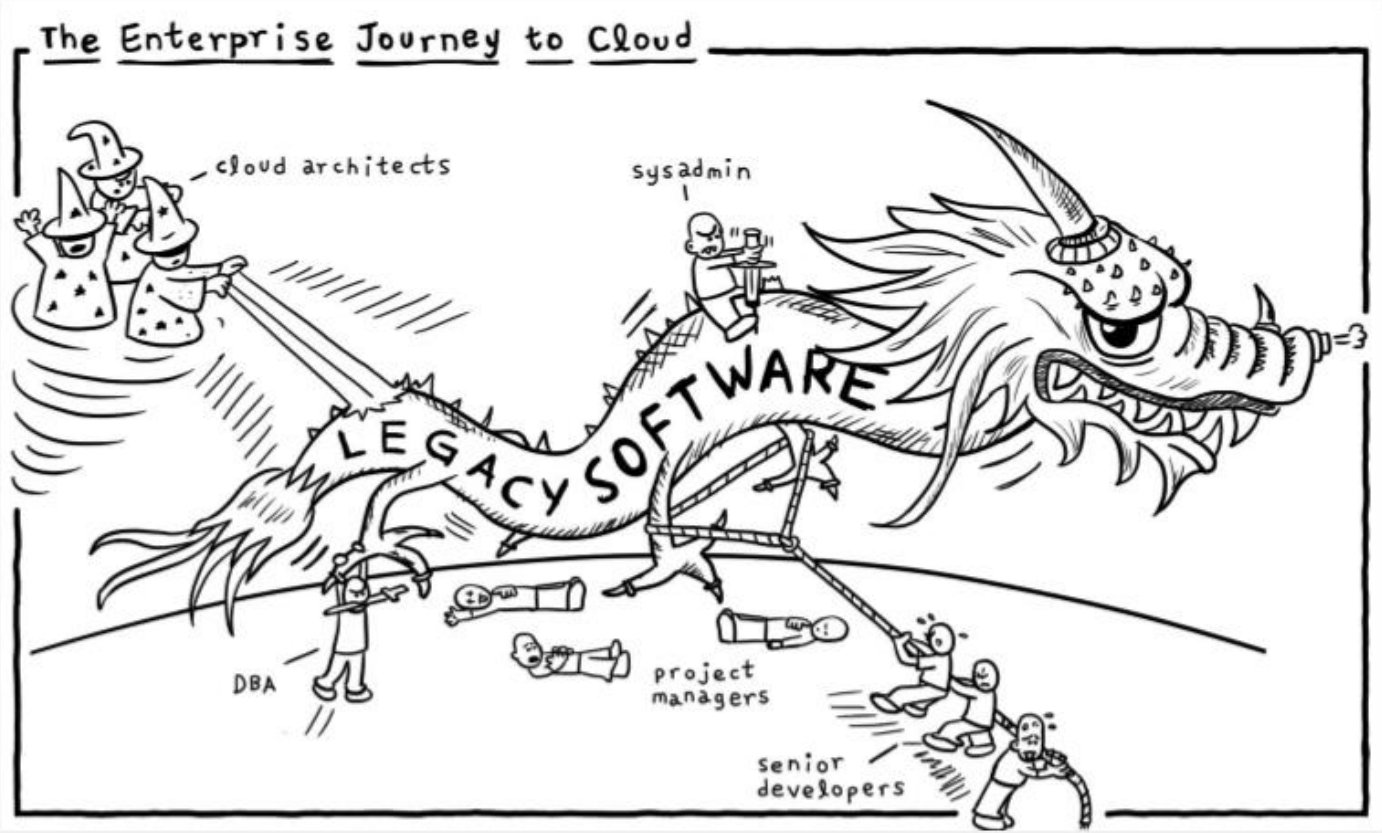
Além disso, precisamos gerenciar ambientes de maneiras diferentes para prover a infra-estrutura por Ops e Dev consumir.



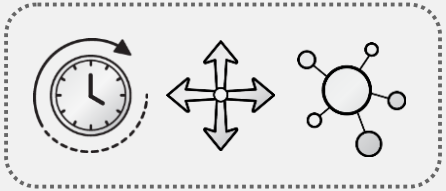
Jornada para a cloud (Expectativa)



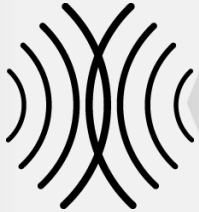
Jornada para a cloud (Realidade)



O Desafio



DESENVOLVEDORES



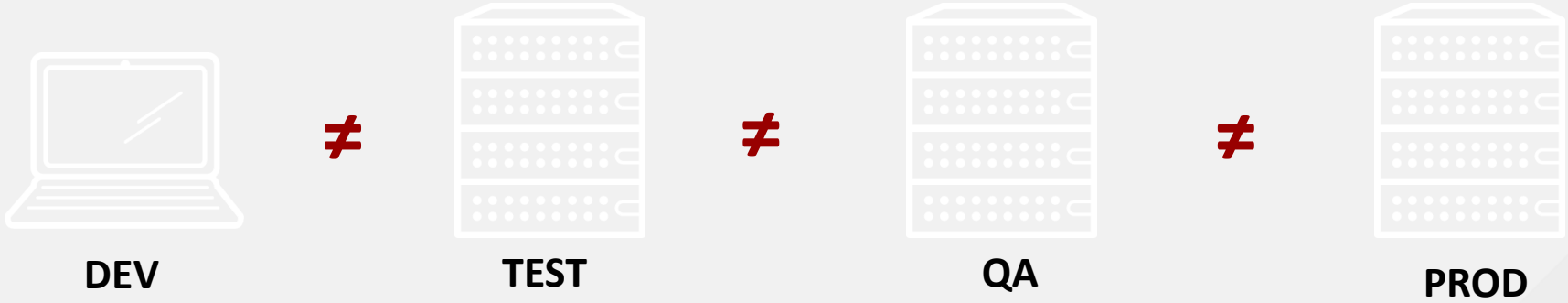
OPERAÇÕES DE TI

Automação em todo lugar: Para onde iremos

RED HAT ANSIBLE Automation — Ferramenta de automação de ambientes - Ops

RED HAT CLOUDFORMS — Ferramenta Multicloud, conectar de forma hibrida nossos ambientes - Ops

RED HAT SATELLITE — Ferramenta de provisionamento e gestão de ambientes - Ops



A Solução

Adotar uma estratégia de containers irá permitir que as aplicações sejam facilmente compartilhadas e implantadas.

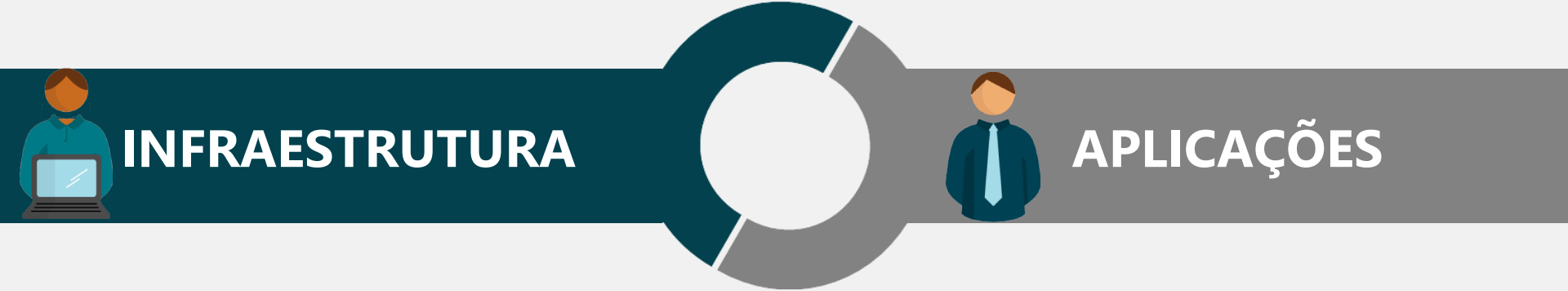
Além do uso de ferramentas que ajudam a entregar uma esteira DevOps.

Quando pensamos em DevOps …

- Tudo como código
- Automação em todo lugar
- Continuous Integration/Delivery
- A aplicação é sempre "releasable"
- Monitoramento de Aplicações
- Rápido Feedback
- Rebuild vs. Repair
- Delivery pipeline

O que são containers?

Depende, para quem você pergunta ...

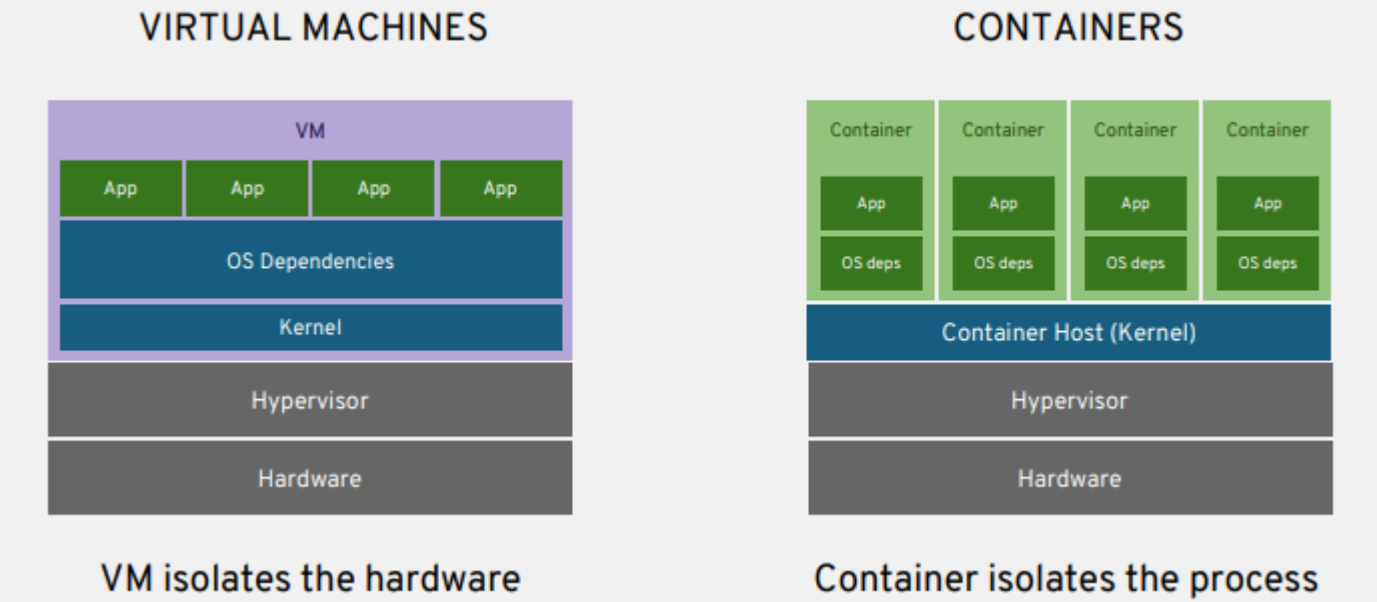


- É uma aplicação rodando em um "Sandbox" em um ambiente onde apenas o Kernel do Linux é compartilhado
- São mais simples, leves e densos que máquinas virtuais
- Portável para qualquer ambiente

- É um pacote com minha aplicação e todas as suas dependências
- Implantação (deploy) para qualquer ambiente em segundos e permite o CI/CD
- Acesso fácil e compartilhado de ambientes containerizados

Máquinas virtuais e containers

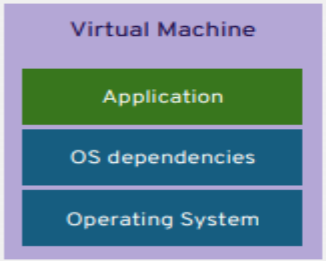
As diferenças de uso na entrega de Máquinas e/ou Containers

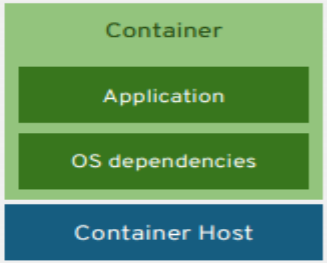


Pensando em desempenho

Entrega de uma máquina virtual e um container

VIRTUAL MACHINES AND CONTAINERS



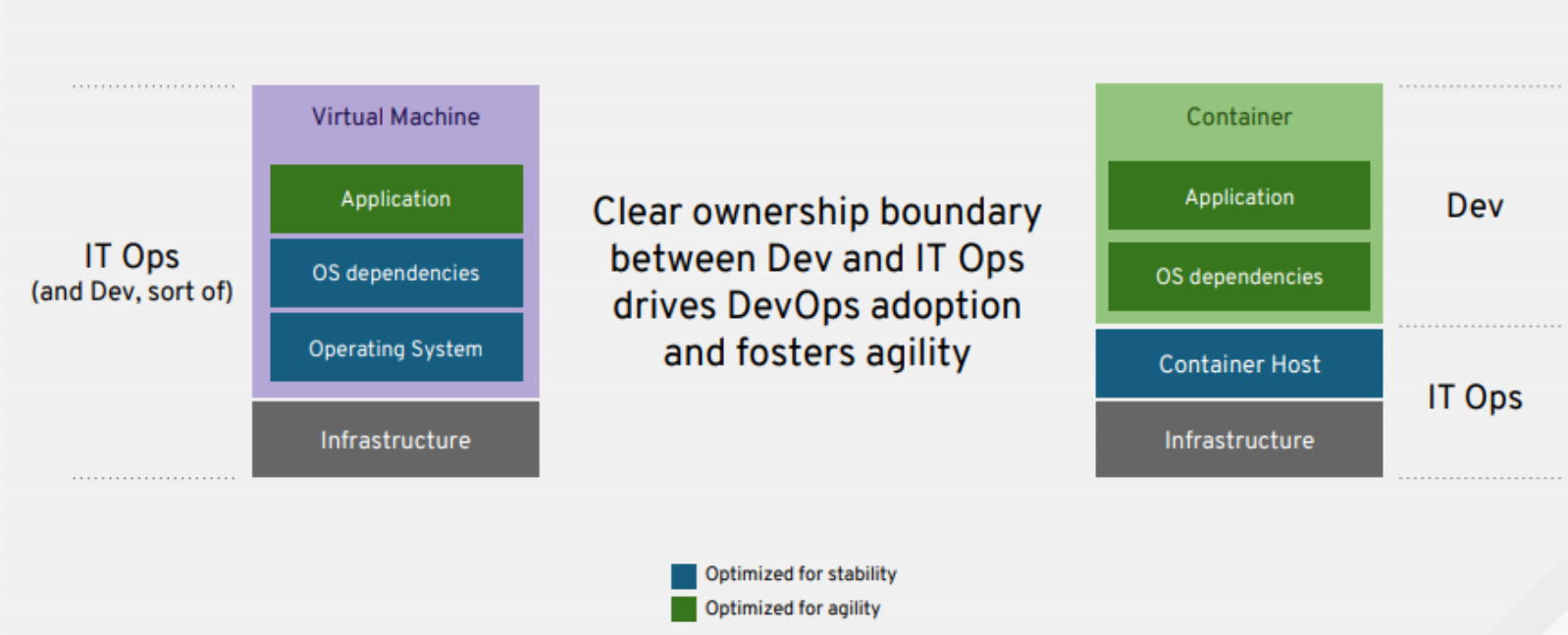


- ➕ VM Isolation
- ➖ Complete OS
- ➖ Static Compute
- ➖ Static Memory
- ➖ High Resource Usage

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

Adoção de um modelo baseado em containers

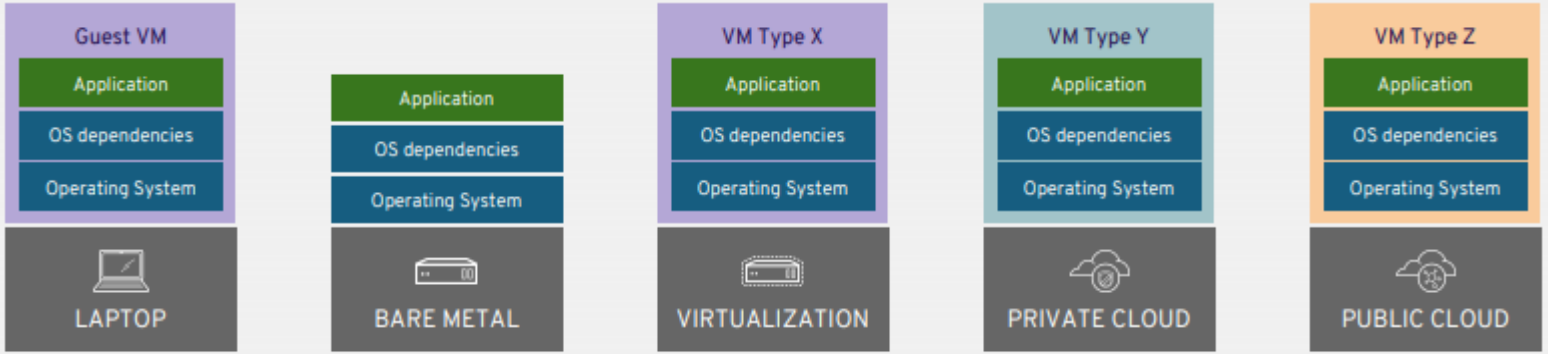
Pensando nos futuros pápeis e responsábilidades do Dev e do Ops



O que o container nos ajuda no dia-a-dia?

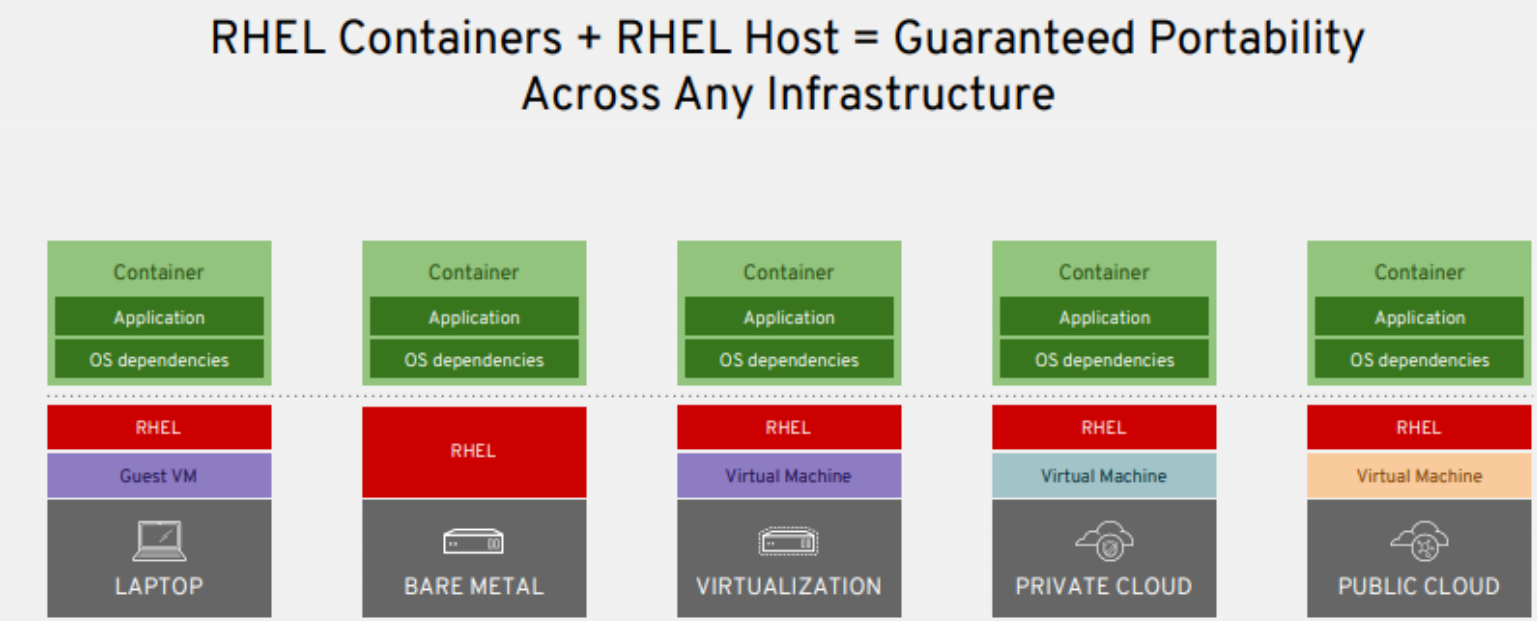
Aqui não possuimos uma estratégia de infra ágil

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

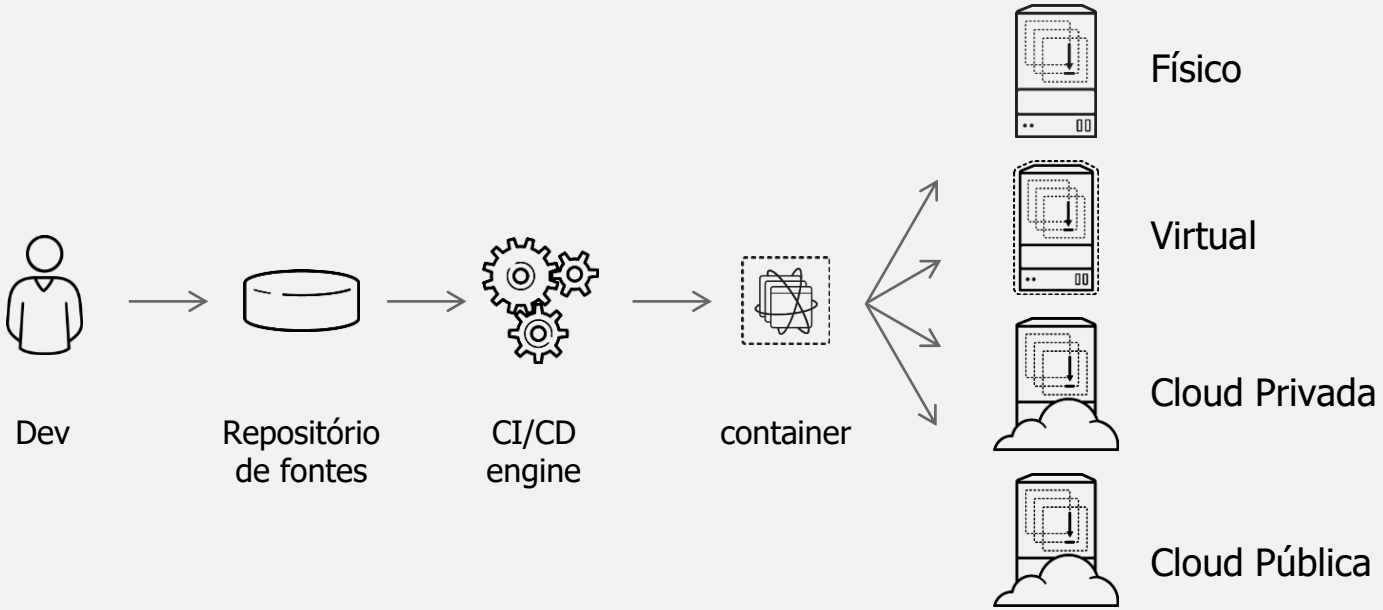


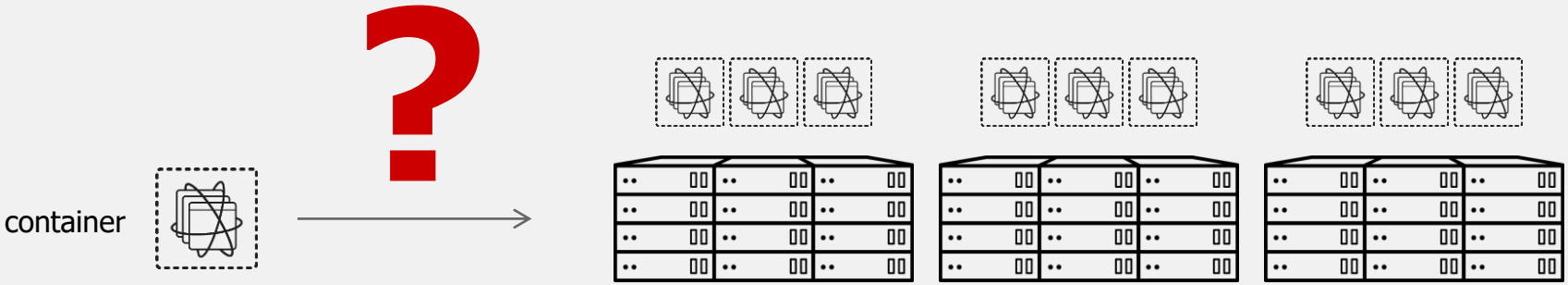
Você não precisa “matar” sua infra atual

Entenda como o container pode te ajudar



Docker é uma tecnologia de software que fornece contêineres, promovido pela empresa Docker, Inc. O **Docker** fornece uma camada adicional de abstração e automação de virtualização de nível de sistema operacional no Windows e no Linux.


Dev
Repositório de fontes
CI/CD engine
container
Físico
Virtual
Cloud Privada
Cloud Pública

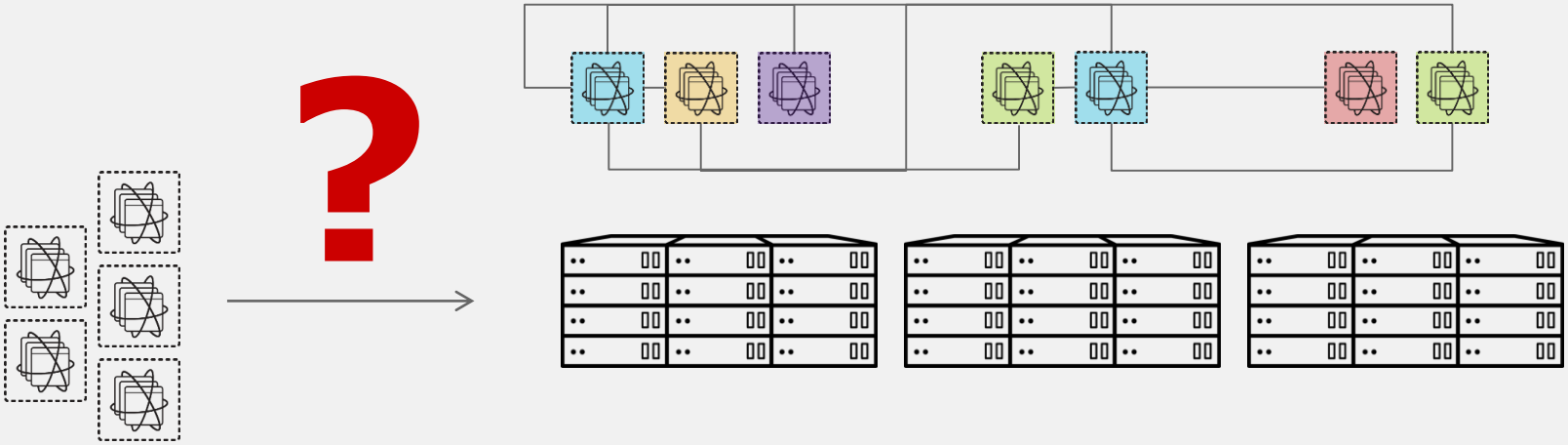

container

```
$ docker build -t app:v1 .

$ docker run app:v1
```

```
$ docker build -t app/frontend:v1 .
$ docker build -t app/backend:v1 .
$ docker build -t app/database:v1 .
$ docker build -t app/cache:v1 .
$ docker build -t app/messaging:v1 .
```

```
$ docker run app/frontend:v1 link-to-backend
$ docker run app/frontend:v1 link-to-backend
$ docker run app/backend:v1 link-to-db-cache-messaging
$ docker run app/backend:v1 link-to-db-cache-messaging
$ docker run app/database:v1
$ docker run app/cache:v1 link-to-db
$ docker run app/messaging:v1
```



Não é suficiente! Necessitamos ir além de containers!

Scheduling
Decide onde devemos fazer o deploy dos containers

Segurança
Controla quem pode fazer o que

Lifecycle and health
Mantém os containers rodando a despeito de falhas

Scaling
Estala os containers Up e Down

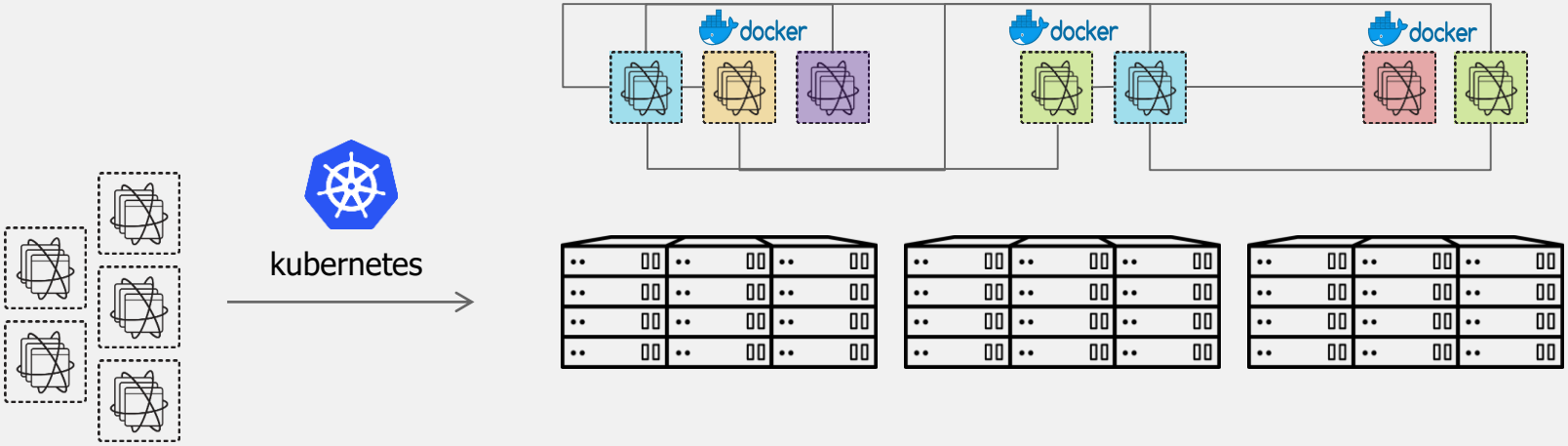
Discovery
Localiza outros containers na rede

Persistência
Mantém os dados além do ciclo de vida do container

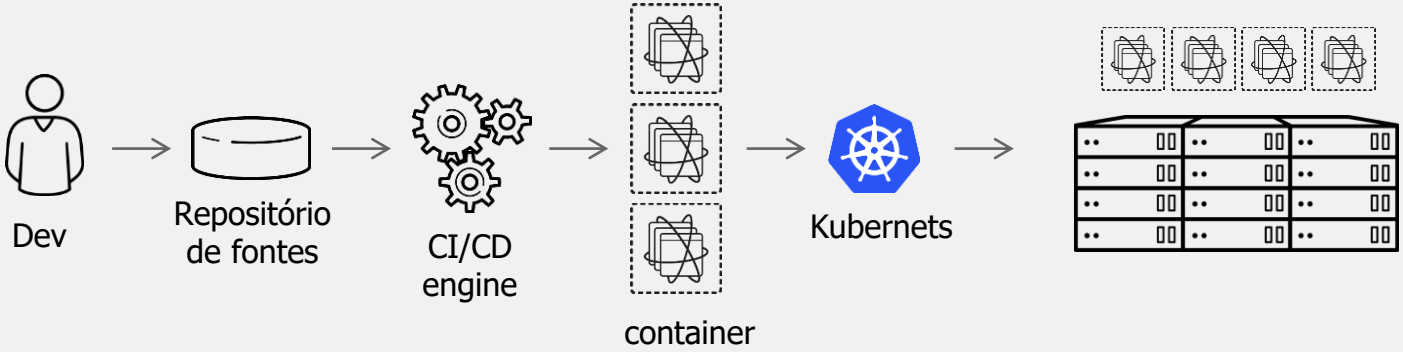
Monitoring
Visibilidade de como os containers estão rodando

Agregação
Compor Aplicações com multiplos containers

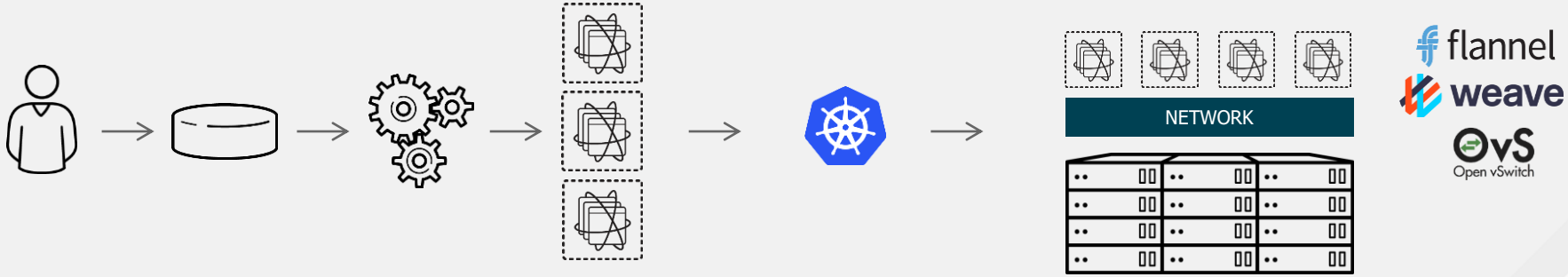
Kubernetes é um sistema OpenSource para automação de deployment, operações, e scaling de aplicações conteinerizadas através de múltiplos hosts. O **Kubernets** é um Gerenciador (a nível)de Containers muito poderoso.


kubernetes
docker
docker
docker

DevOps utilizando Kubernets

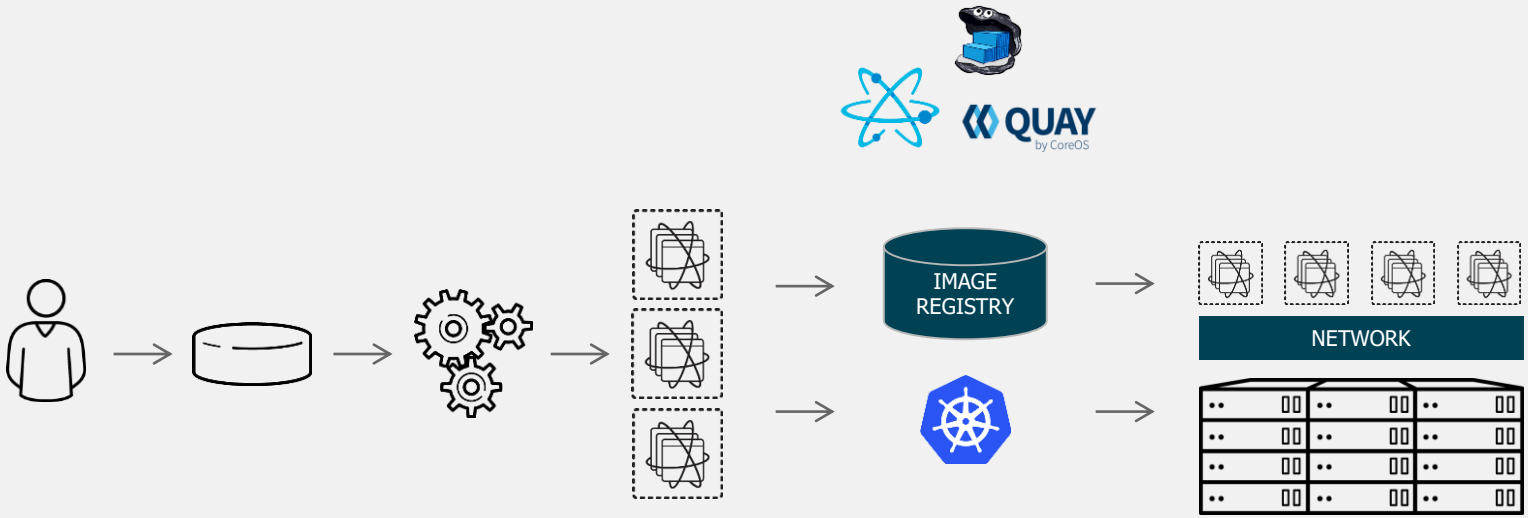


DevOps utilizando Kubernets



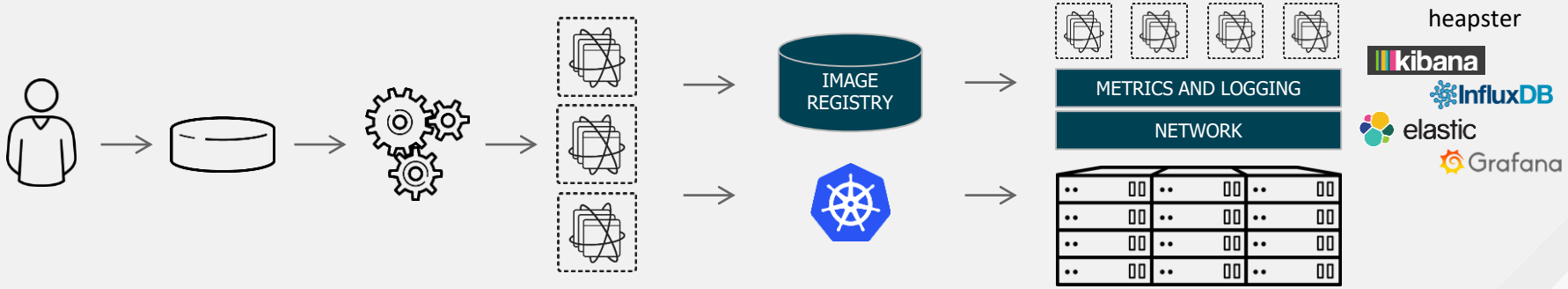
Não é suficiente! Precisamos de networking!

DevOps utilizando Kubernets



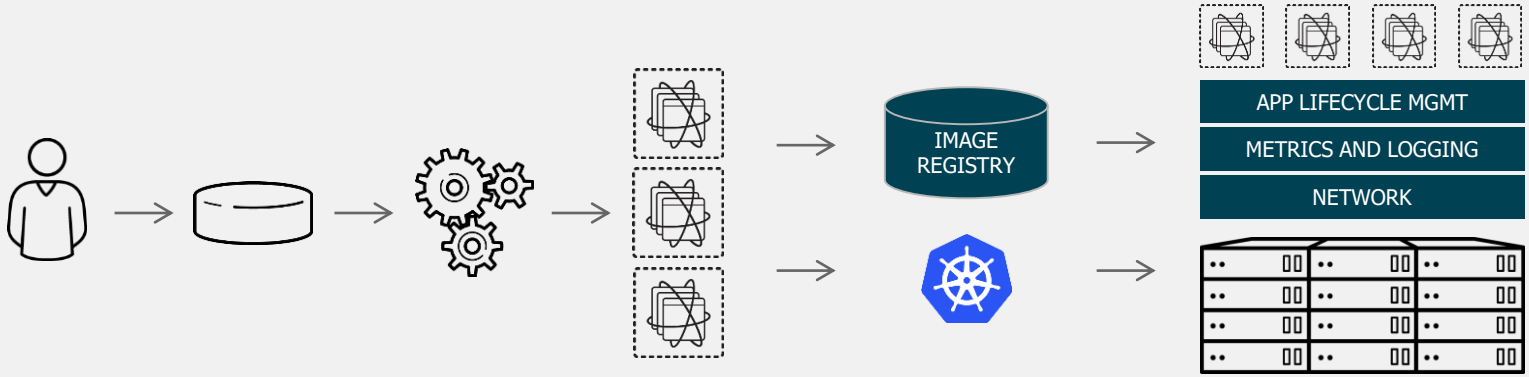
Não é suficiente! Necessitamos de registro das imagens!

DevOps utilizando Kubernets



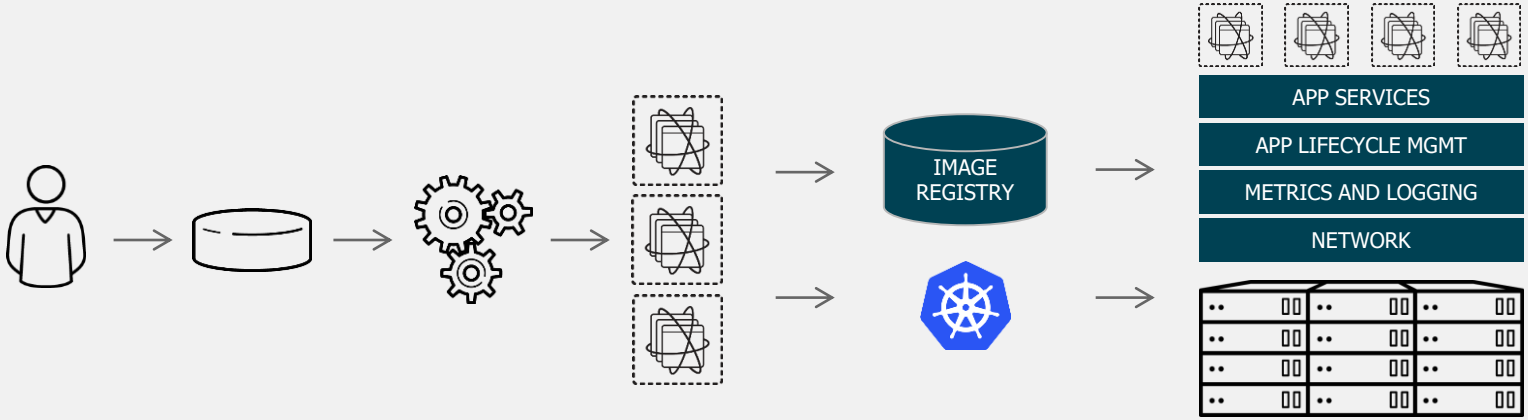
Não é suficiente! Necessitamos de métricas e logging!

DevOps utilizando Kubernets



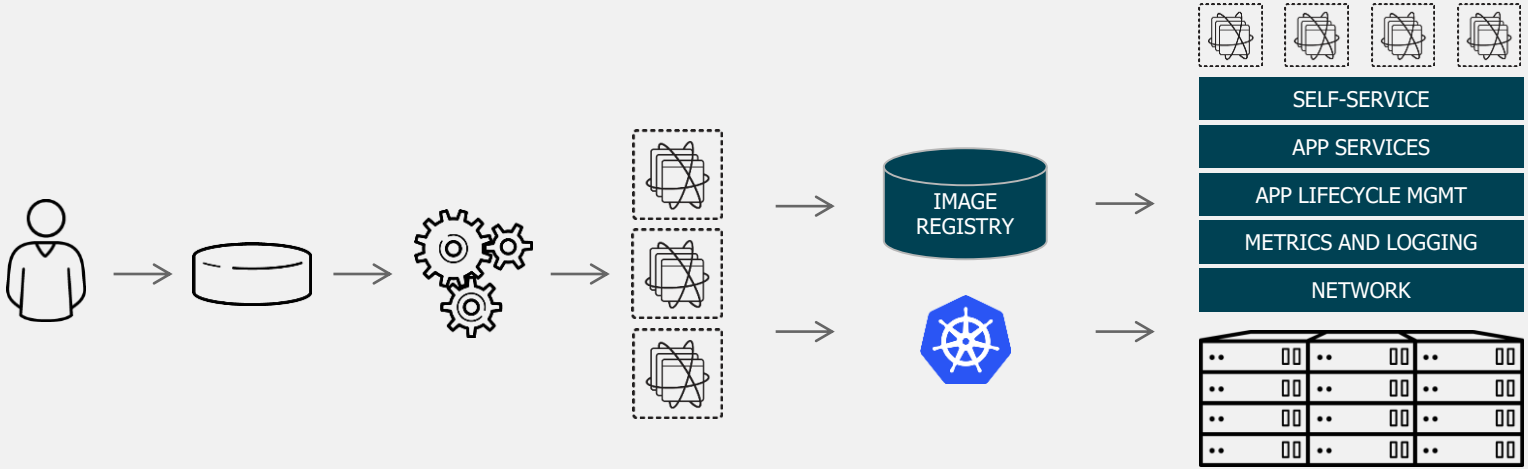
Não é suficiente! Necessitamos de gerenciamento de ciclo de vida!

DevOps utilizando Kubernets



Não é suficiente! Necessitamos de App Services, exemplo: DBs e messageria

DevOps utilizando Kubernets



Não é suficiente! Necessitamos de portal de self-service!

Não é suficiente! Necessitamos ir além de Kubernets!

Multi-tenancy	Teams and Collaboration
Routing & Load Balancing	Quota Management
CI/CD Pipelines	Image Build Automation
Role-based Authorization	Container Isolation
Capacity Management	Vulnerability Scanning
Infrastructure Visibility	Chargeback

RED HAT® OPENSHIFT

Tradicional, Stateful & Microservices como base para Apps

Red Hat OpenShift Container Platform

Red Hat Enterprise Linux & Atomic Host

PaaS

Business Automation	Integration	Data	Web & Mobile	3rd party frameworks
CONTAINER	CONTAINER	CONTAINER	CONTAINER	CONTAINER

CaaS

LIFECYCLE AUTOMATION (Self-service, CI/CD, Image Stream)

CONTAINER MANAGEMENT (Monitoring, Capacity, Policies)

CONTAINER INFRASTRUCTURE SERVICES (Orchestration & Scheduling, Storage, Registry, Security, Networking)

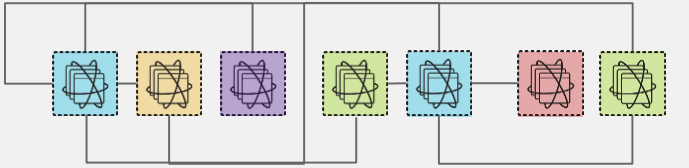
IaaS

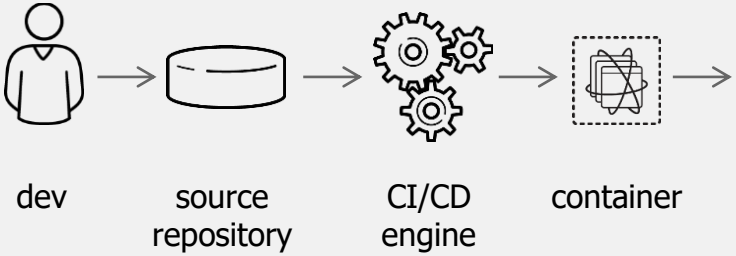
ENTERPRISE-GRADE CONTAINER OS

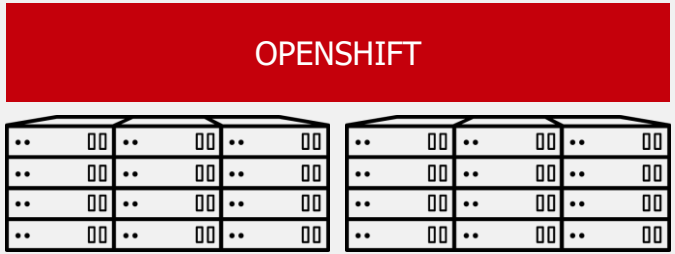
Physical | Virtual | Private | Public

DELL EMC redhat.

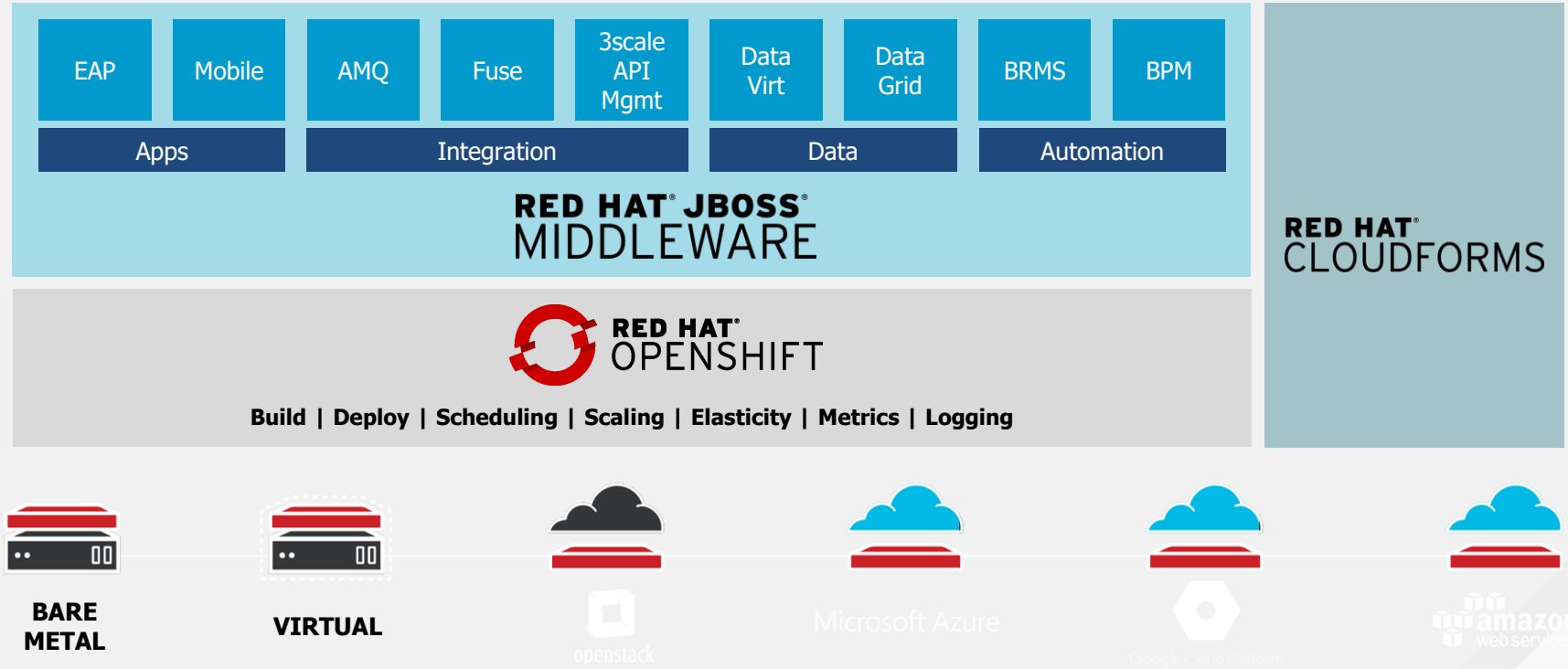
RED HAT® OPENSHIFT







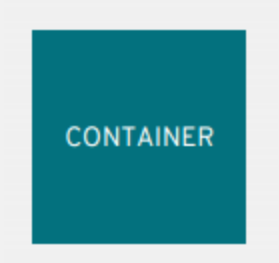
PaaS


EAP
Mobile
AMQ
Fuse
3scale API Mgmt
Data Virt
Data Grid
BRMS
BPM
Apps
Integration
Data
Automation
RED HAT® JBOSS®
MIDDLEWARE
RED HAT®
CLOUDFORMS
RED HAT®
OPENSHIFT
Build | Deploy | Scheduling | Scaling | Elasticity | Metrics | Logging
BARE METAL
VIRTUAL
openstack
Microsoft Azure
Google Cloud Platform
amazon web services

Implementação baseada em Open Container Initiative do Kubernetes **Container Runtime Interface.** Arquitetura mínima e segura. Otimizado para trabalhar com o **Kubernetes**. Executa qualquer imagem compatível com OCI (incluindo **Docker**).

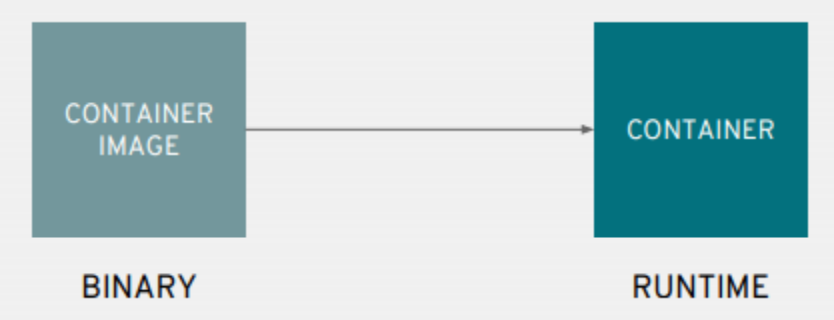
Container

Um contêiner é a menor unidade de um sistema operacional na computação



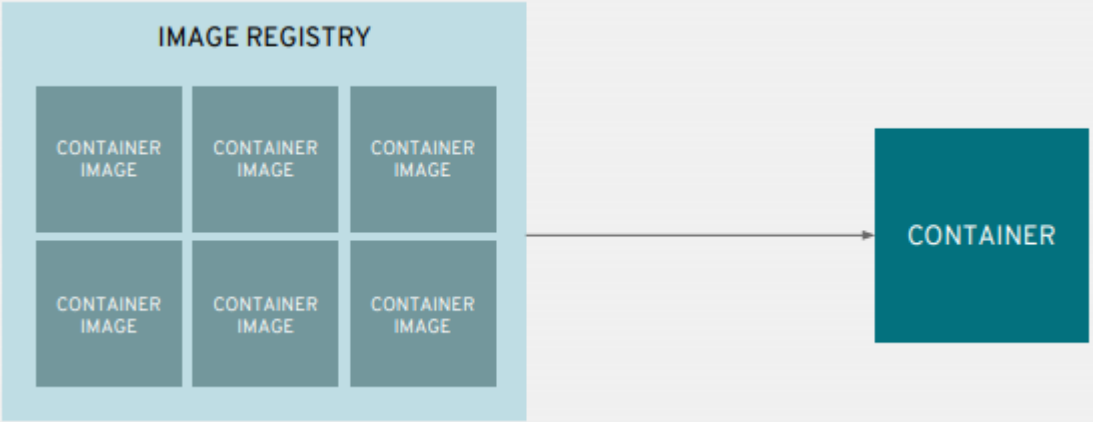
Image

Contêineres são criados a partir de imagens de contêiner



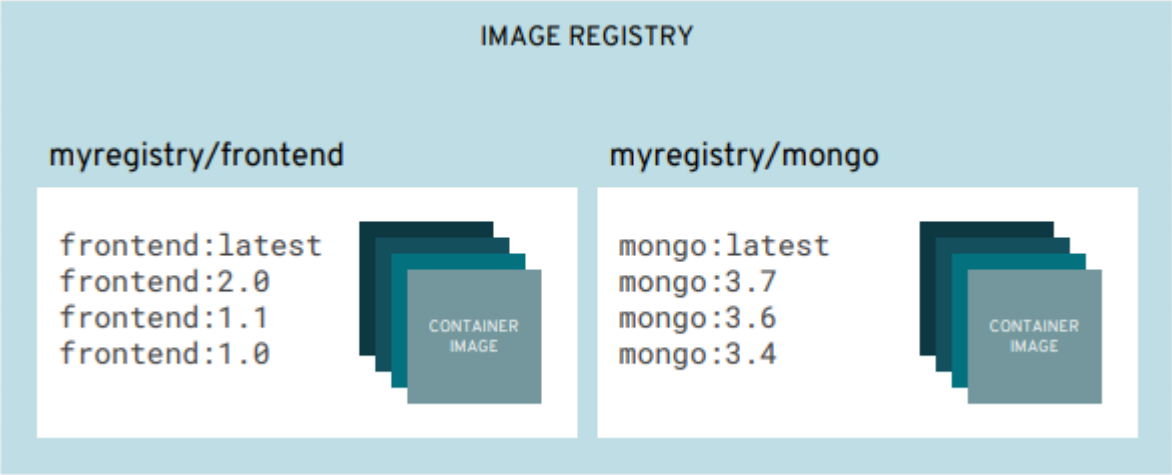
Registry

As imagens do contêiner são armazenadas em um registro de imagem



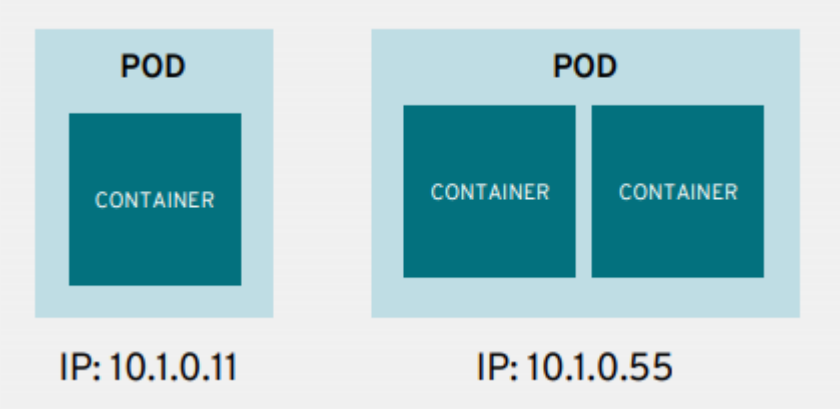
Repository

Um repositório de imagens contém todas as versões de uma imagem no registro de imagens



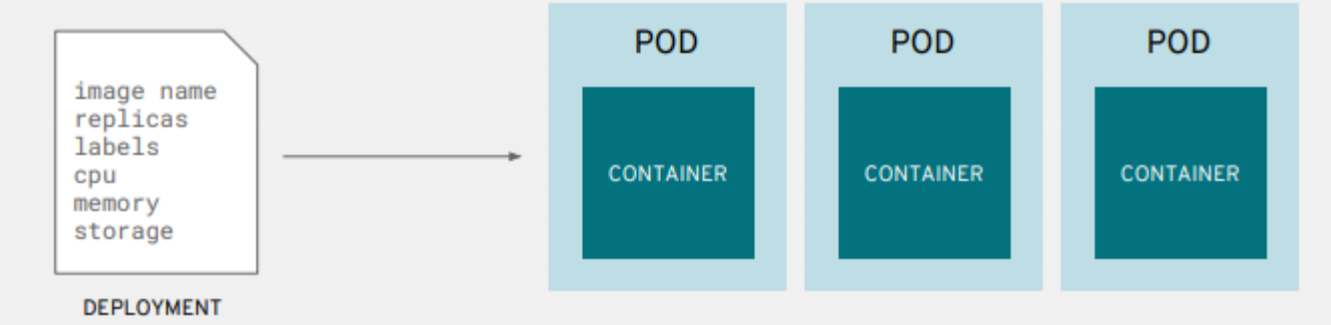
IP

Contêineres são agrupados em grupos de unidades de implantação e gerenciamento



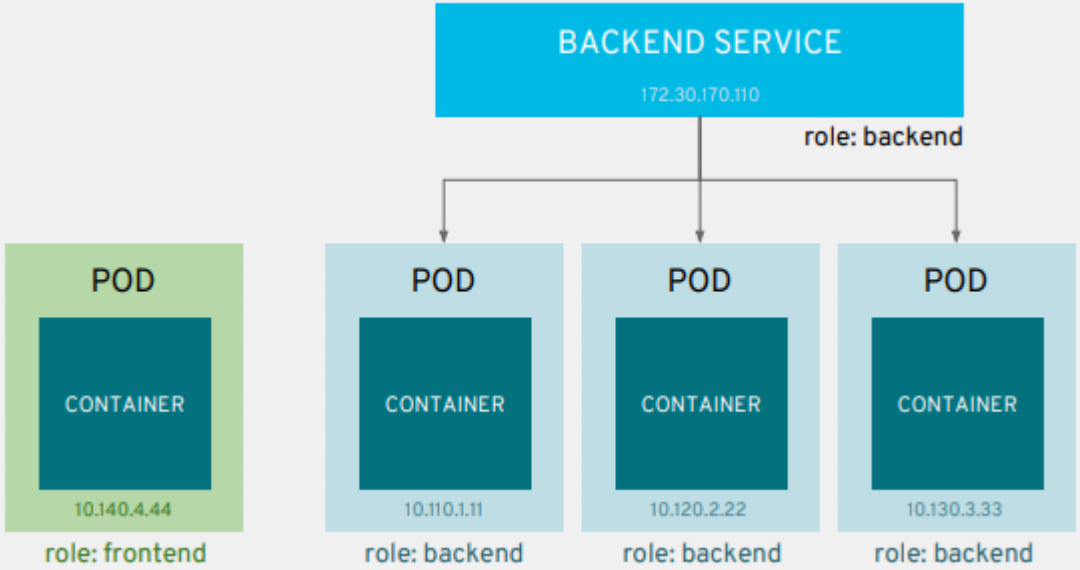
Deployment

A configuração de pods é definida em uma implementação



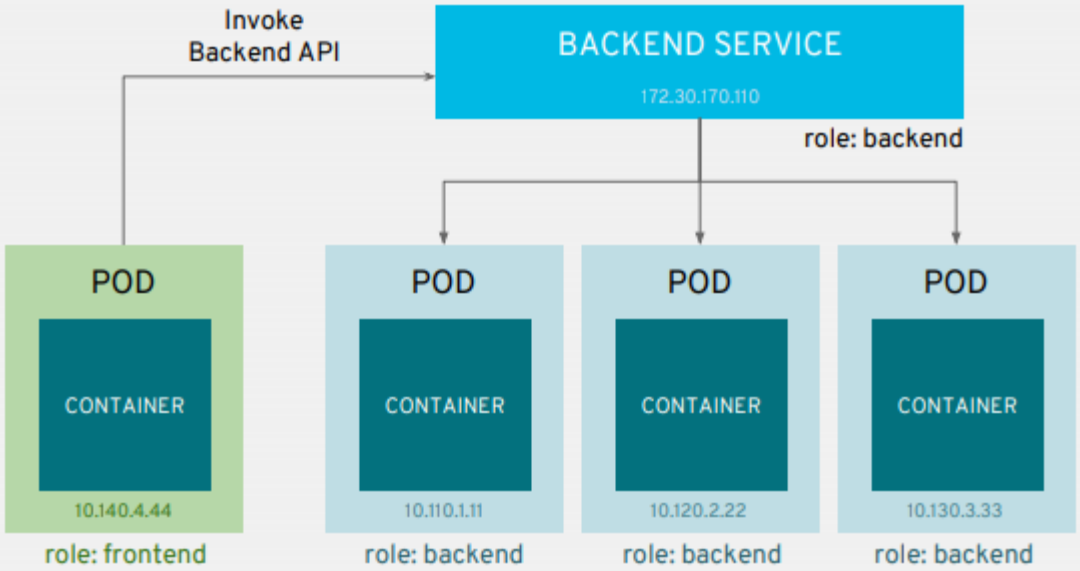
Backend Service

Serviços fornecem balanceamento de carga interno e descoberta de serviço em pods



Backend API

Os aplicativos podem conversar entre si por meio de serviços

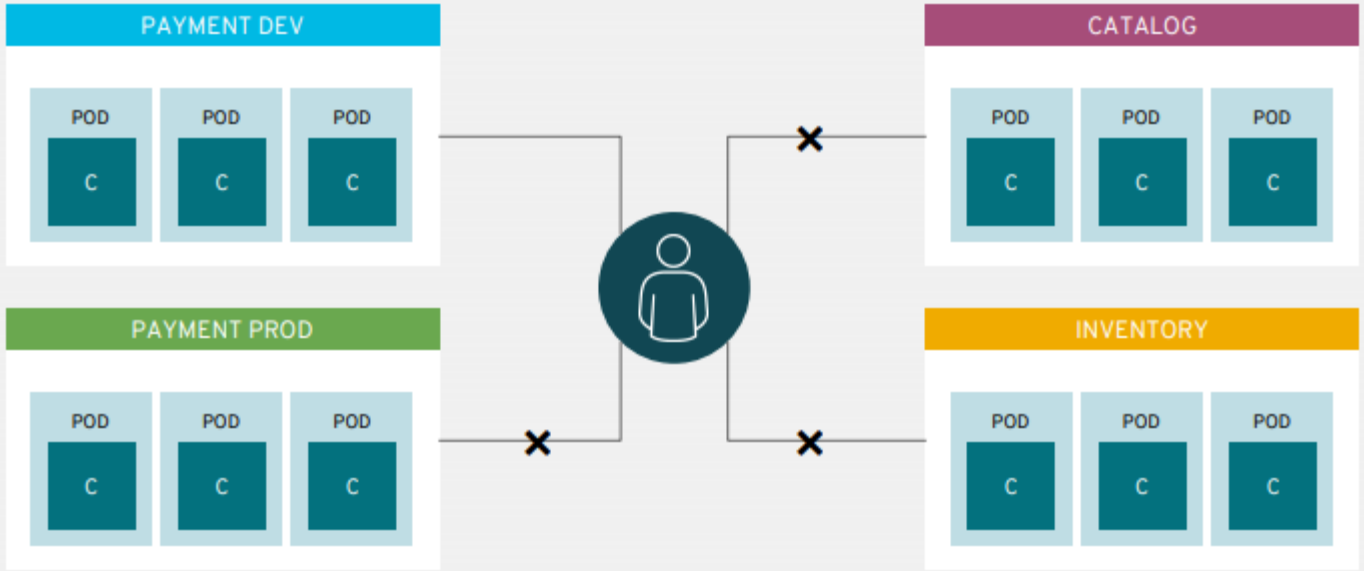


Route

Rotas adicionam serviços ao balanceador de carga externo e fornecem URLs legíveis para o aplicativo

POD

Projetos isolam aplicativos em ambientes, equipes, grupos e departamentos



Arquitetura inicial - OpenShift

Balanciador de carga

- Provido pela Ailos

2x Infra Nodes

- 8GB de memória
- 2 cores (4 vCPUs)
- Disco com 45GB (/var com 20GB)
- Disco adicional 50GB

3x Master Nodes

- 32GB de memória
- 2 core (4 vCPUs)
- Disco com 65GB (/var com 40GB)

3x App Nodes

- 16GB de memória
- 4 cores (8 vCPUs)
- Disco com 45GB (/var com 20GB)
- Disco adicional 50GB (Images)

Replicas de 3x Arquivos

File 1

4x OCS/Storage

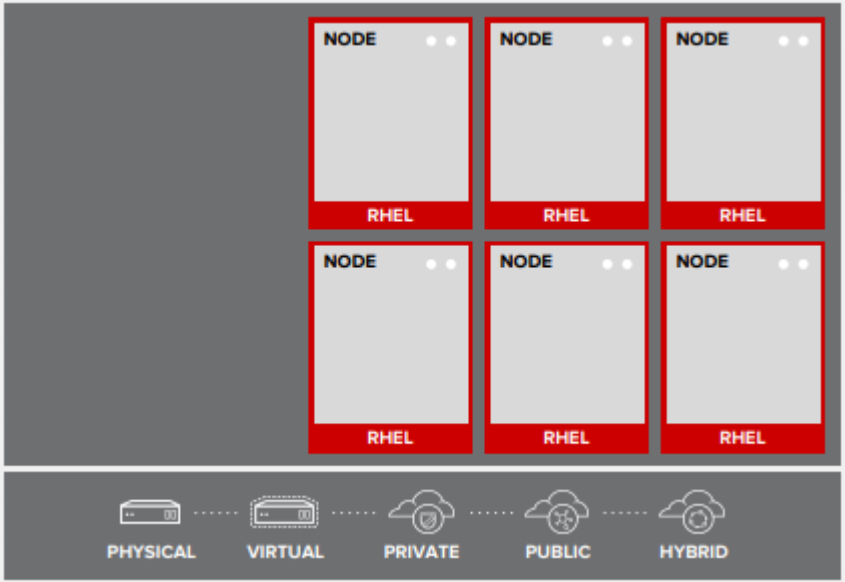
- 32GB de memória
- 2 cores (4 vCPUs)
- Disco com 200GB
- Ponto de montagem / (50GB)
- Ponto de montage /var (150GB)
- 1 vNIC Gigabit Ethernet

DELL EMC

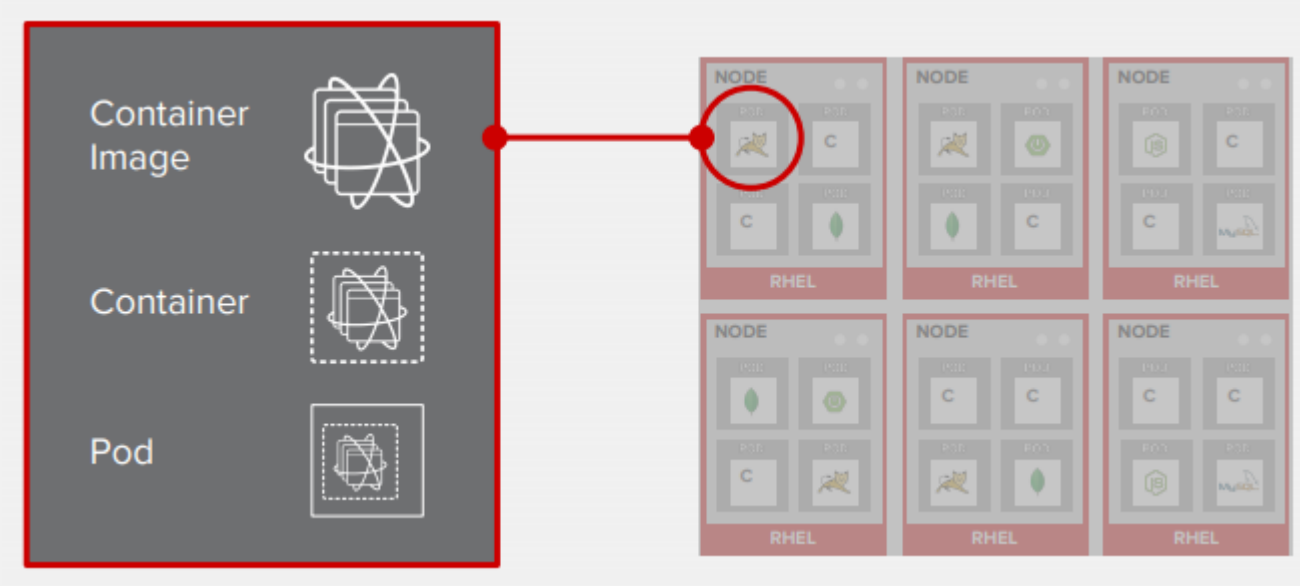
redhat.

Podemos ter mais de uma alternativa para Infra

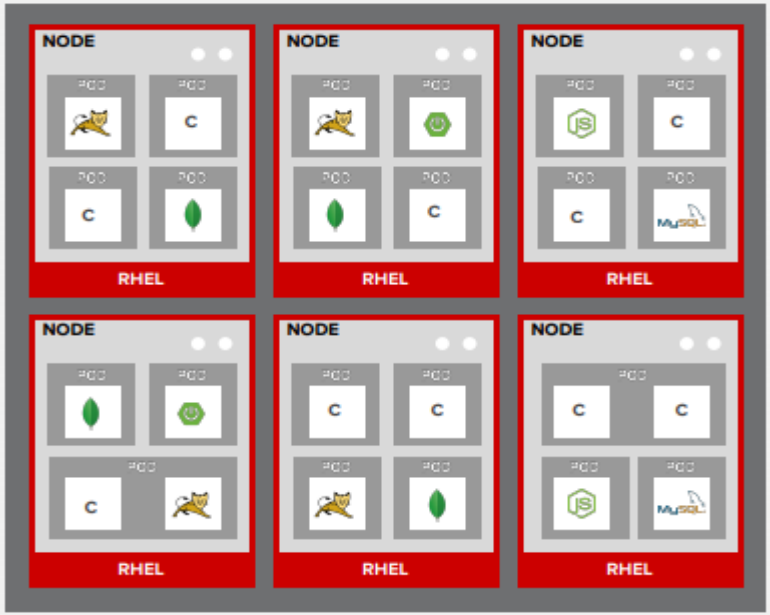
Instâncias de Nodes com RHEL onde o app funciona



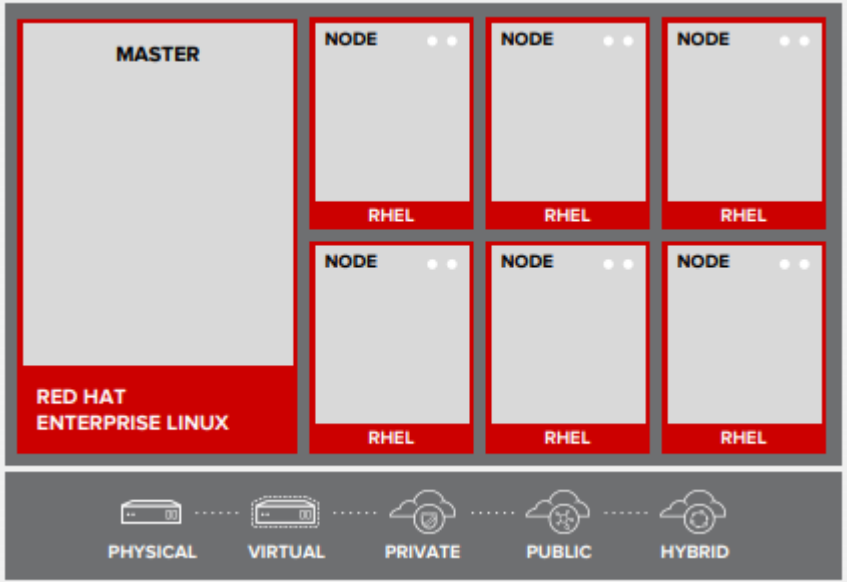
Apps funcionando em Containers



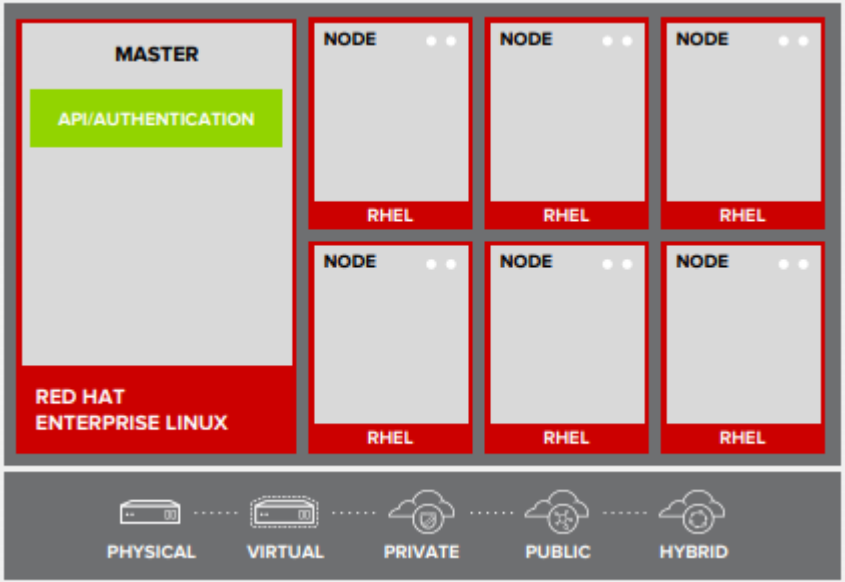
Pods são as unidades de orquestração



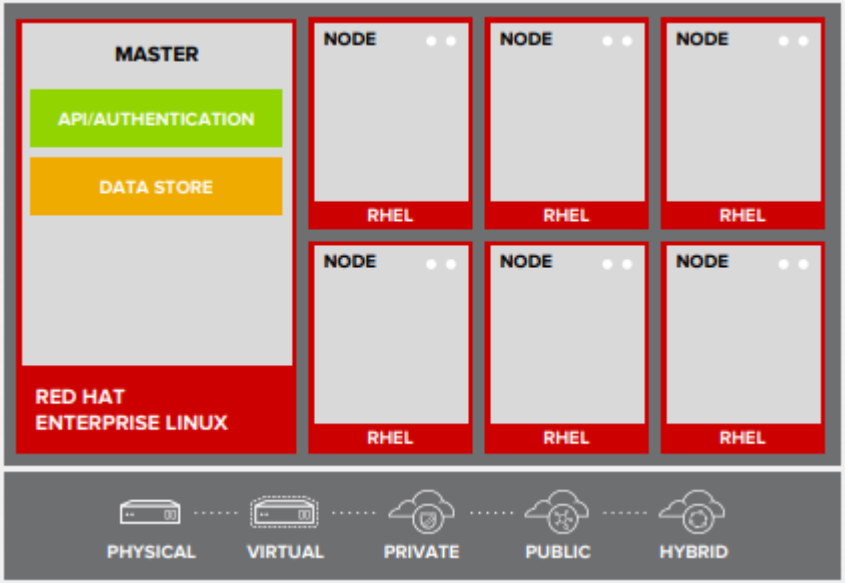
O Master faz todo o controle



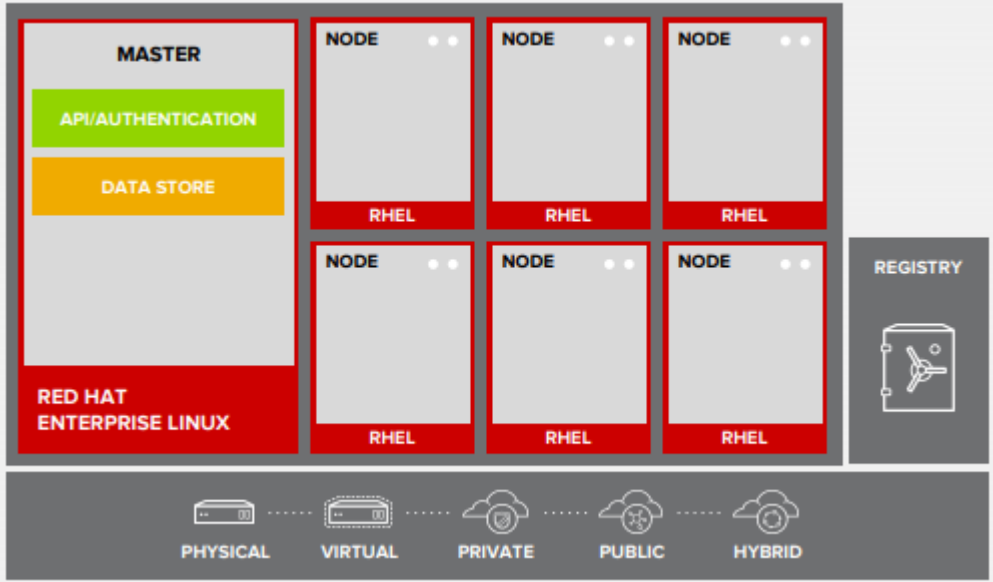
API de identificação



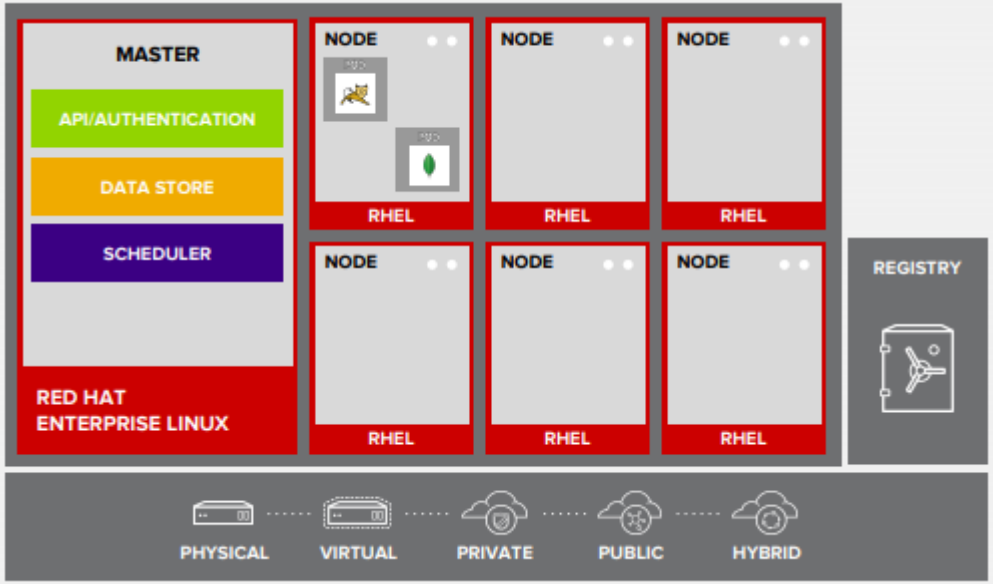
Estado valido e desejado



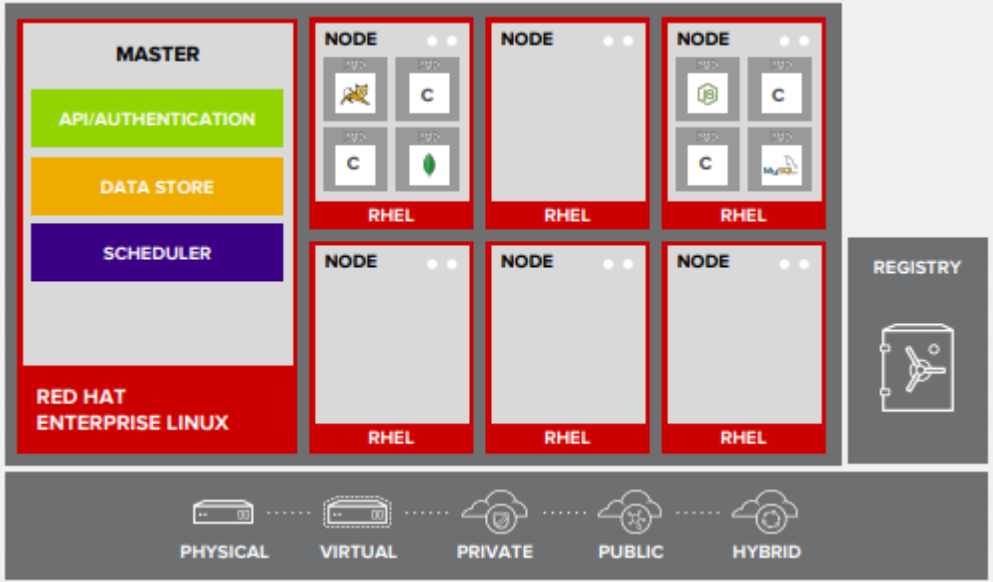
Integração com o container Registry



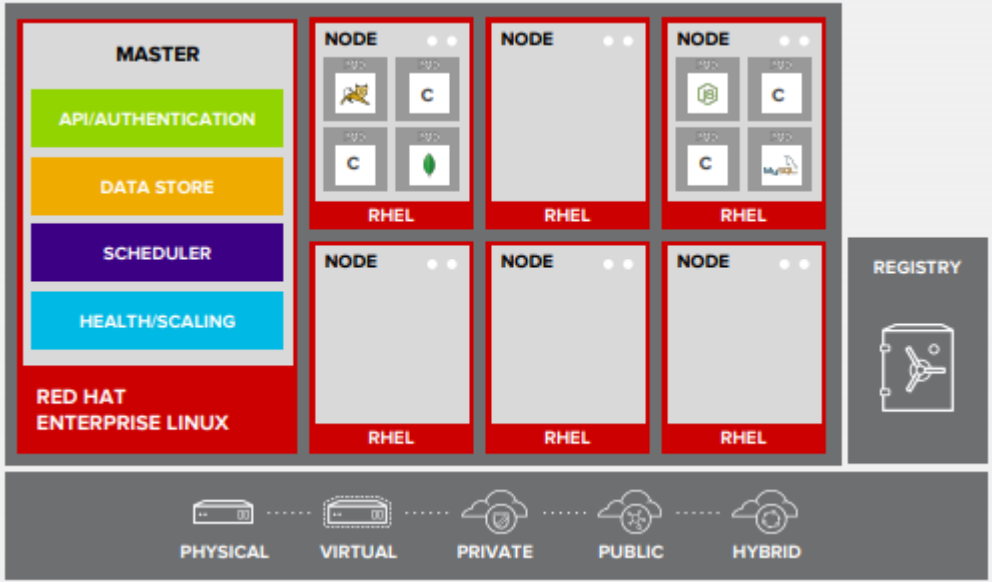
Orquestração e Agendamento



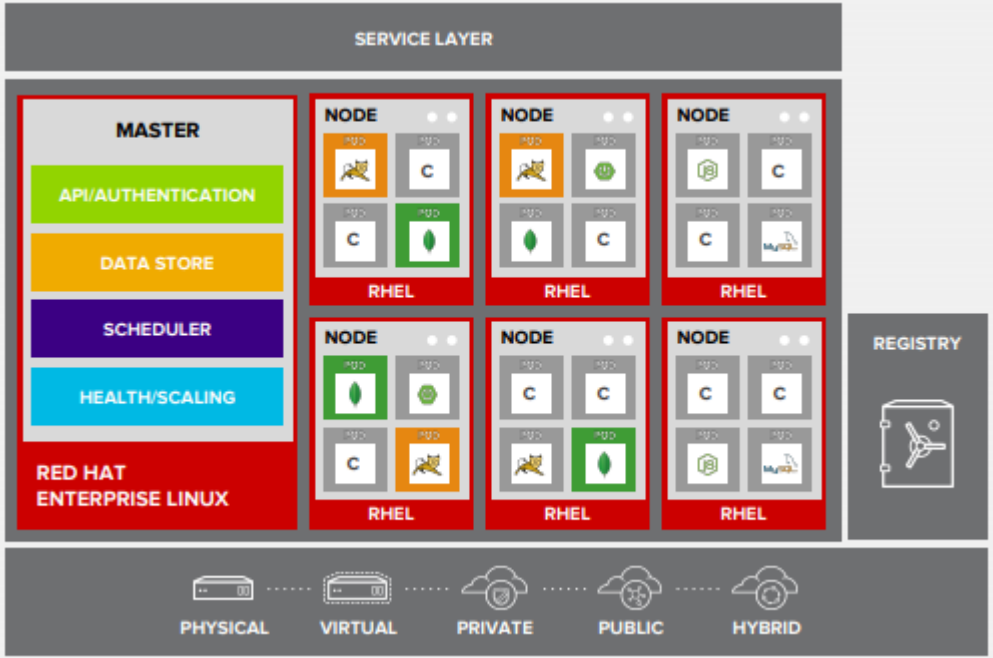
Inserção de políticas



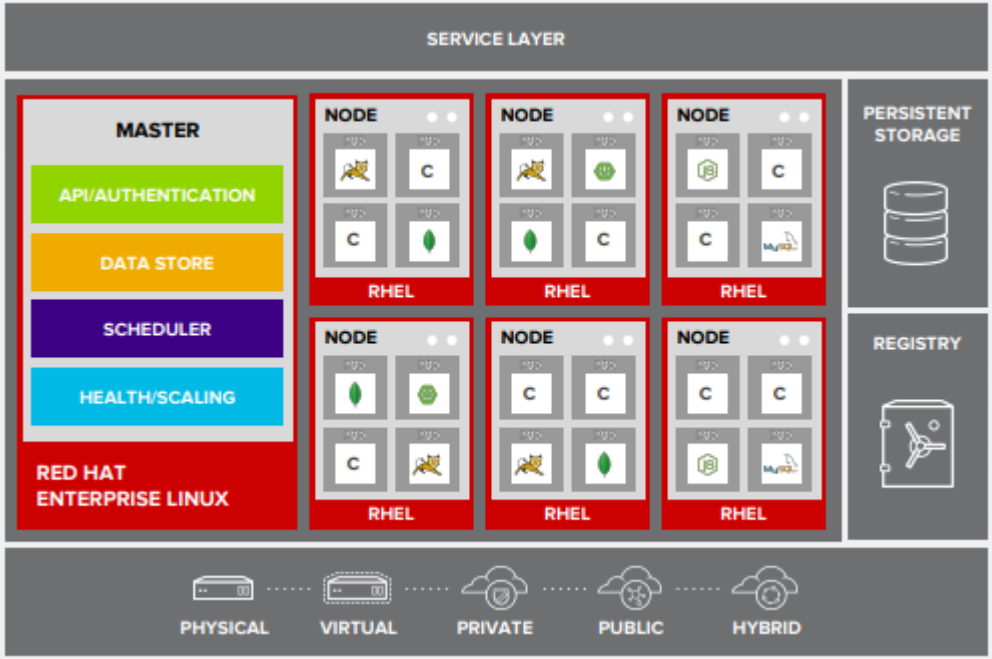
Escalação de Pods



Serviços e Descovery

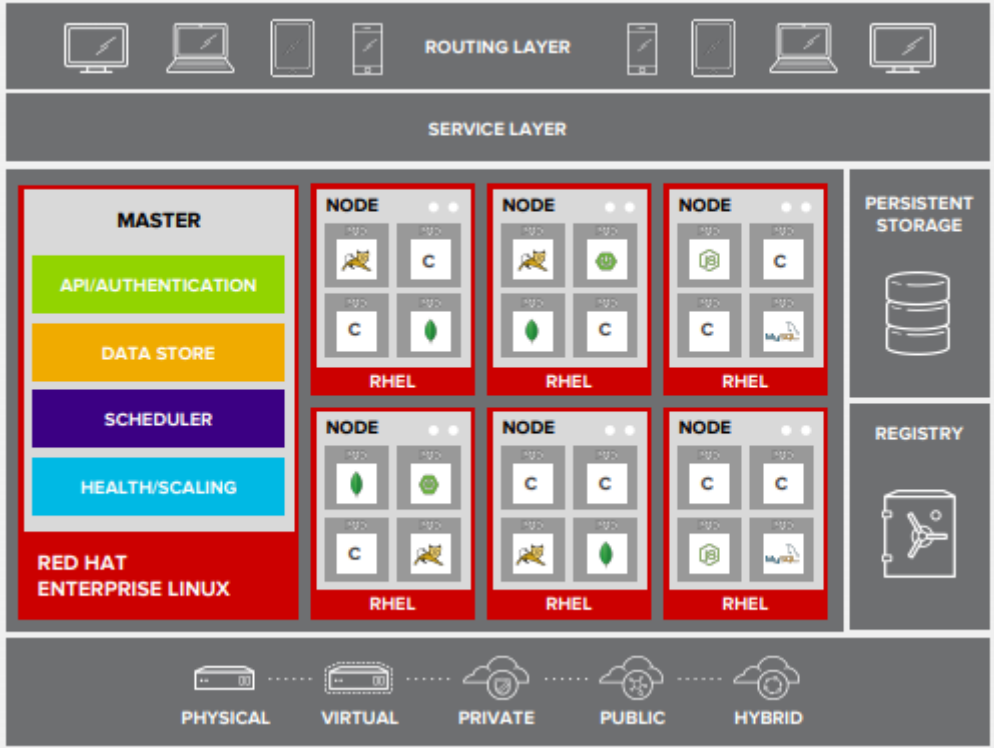


Dados persistentes dos containers

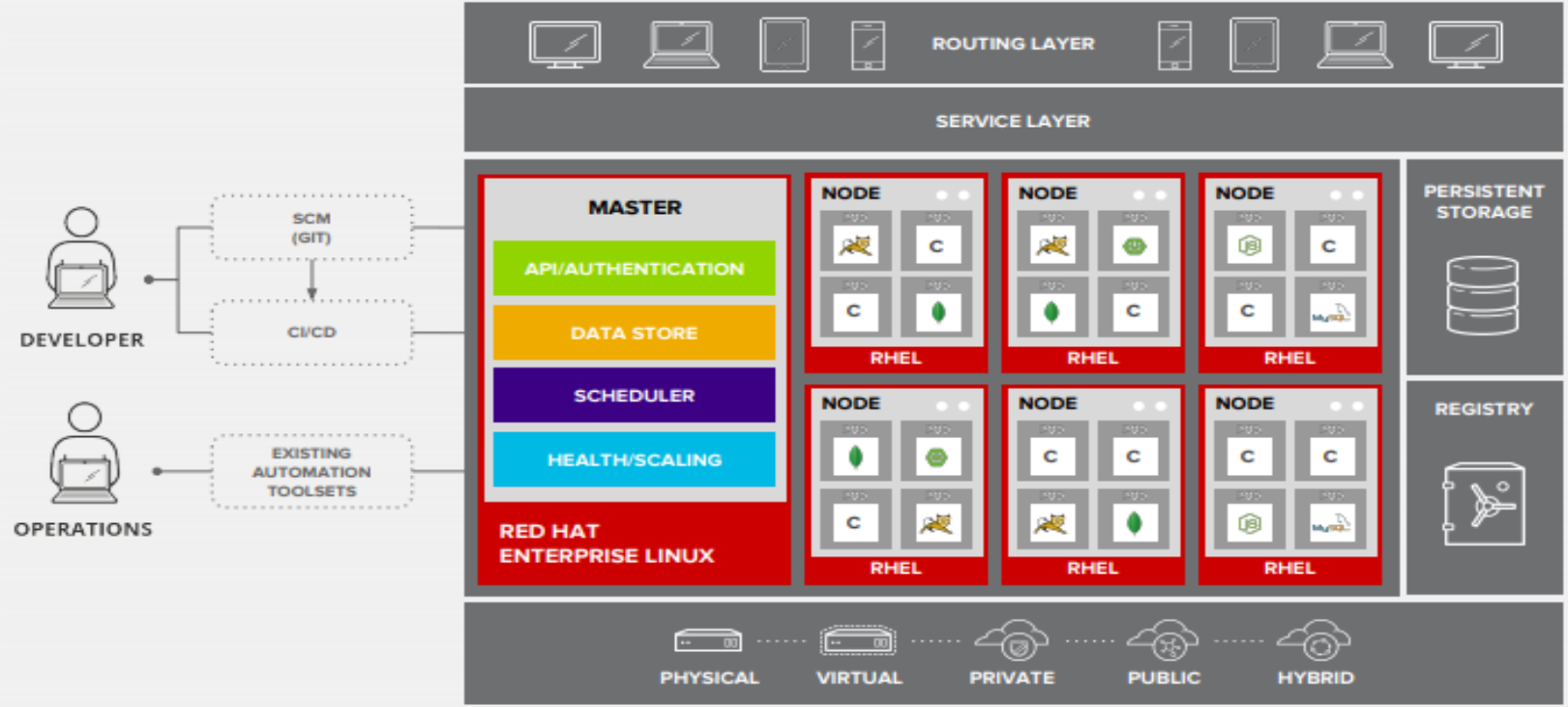


Roteamento e Load Balance

Texto



Acesso via Web, CLI, IDE e API



Estratégia de Esteira e Pipeline

Jenkins

Jenkins ▸ tooling/tasks-pipeline ▸

- Back to Dashboard
- Status
- Changes
- Build Now
- Delete Pipeline
- Configure
- Move
- Full Stage View
- Pipeline Syntax

Build History trend

find

- #3 Apr 14, 2017 4:39 PM
  OpenShift Build tooling/tasks-pipeline-3
- #2 Apr 14, 2017 4:35 PM
  OpenShift Build tooling/tasks-pipeline-2
- #1 Apr 14, 2017 4:32 PM
  OpenShift Build tooling/tasks-pipeline-1

RSS for all RSS for failures

Pipeline tooling/tasks-pipeline

Project name: tooling-tasks-pipeline

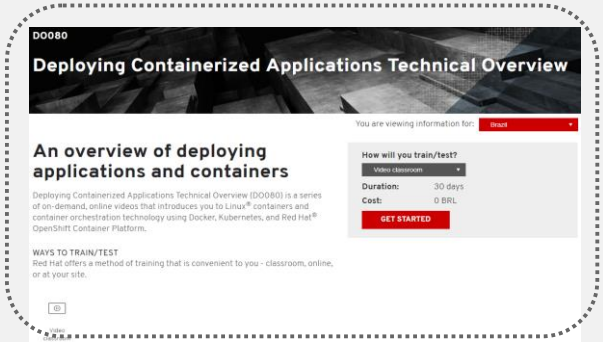
Recent Changes

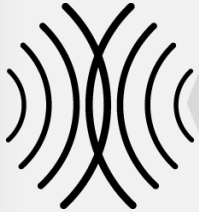
Stage View

	Build	Push to Nexus	Deploy DEV	Deploy STAGE
Average stage times:	21s	8s	1min 43s	7s
#3 Apr 14 13:39 No Changes	46s	8s	1min 43s	7s (paused for 5s)
#2 Apr 14 13:35 No Changes	7s failed			
#1 Apr 14 13:32 No Changes	9s failed			

DELL EMC redhat.

Get Start!





DESENVOLVEDORES

OPERAÇÕES DE TI

English

Single-page HTML

Getting Started

1. Overview

1.1. Introduction

1.1.1. Why Should I Use OpenShift?

2. Install OpenShift Container Platform

2.1. Overview

2.1.1. Prerequisites

2.1.2. Attach OpenShift Container Platform Subscription

2.1.3. Set Up Repositories

2.1.4. Install the OpenShift Container Platform Package

2.1.5. Set up Password-less SSH Access

2.1.6. Run the Installation Playbooks

2.2. Interact with OpenShift Container Platform

2.3. Understand Roles and Authentication

Red Hat Training

A Red Hat training course is available for OpenShift Container Platform

GETTING STARTED

OPENSHIFT
CONTAINER
PLATFORM
3.11

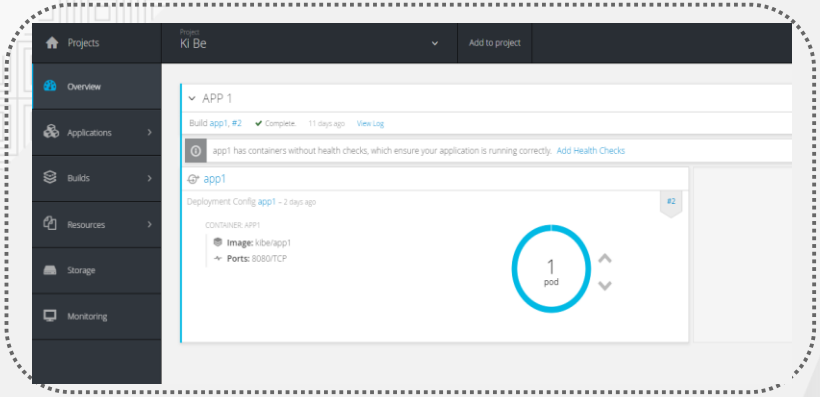
Getting Started with OpenShift Container Platform 3.11

Red Hat OpenShift Documentation Team

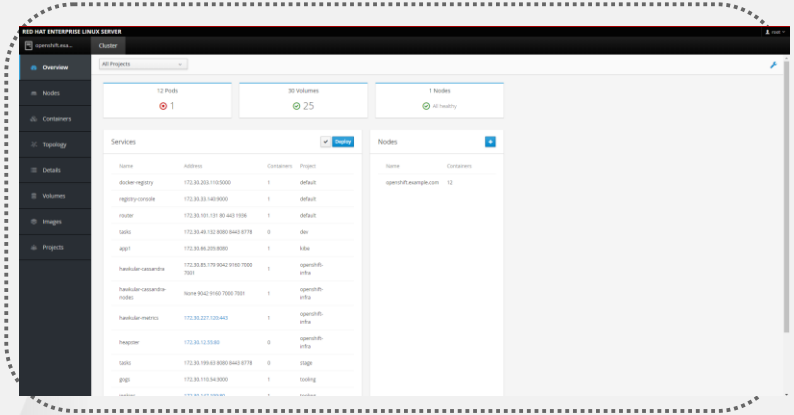
Legal Notice

Abstract

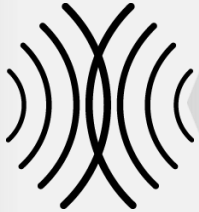
Whether you are a developer or a platform administrator, you can get started with OpenShift using the topics in this book. Administrators can use the installation utility and an interactive CLI tool, to quickly install and configure a new OpenShift instance across multiple hosts. Developers can use the OpenShift CLI or the web console to log in to an existing OpenShift instance and start creating applications.



Como?



DESENVOLVEDORES



OPERAÇÕES DE TI

RED HAT OPENSHIFT Container Platform

OPENSHIFT CONTAINER PLATFORM

Username caio_candido-dell.com

Password ••••••••

Log In

Welcome to the OpenShift Container Platform.

Projects

Project
Ki Be

Add to project

Overview

Applications

Deployments

Stateful Sets

Pods

Services

Routes

Builds

Resources

Storage

Monitoring

APP 1

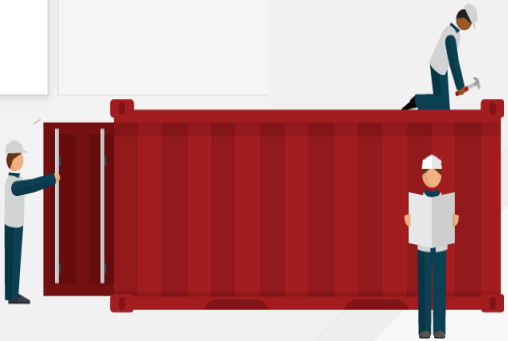
11 days ago View Log

out health checks, which ensure your application is running correctly. Add Health Checks

ago

#2

1
pod



DELL EMC redhat.

RED HAT ENTERPRISE LINUX SERVER

root

openshift.exa...

Cluster

- Overview
- Nodes
- Containers
- Topology
- Details
- Volumes
- Images
- Projects

All Projects

Select an object to see more details.

Pod

Pods contain one or more containers that run together on a node, containing your application code.

Replication Controller

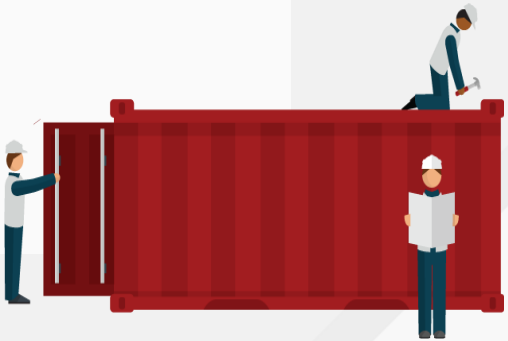
Replication controllers dynamically create instances of pods from templates, and remove pods when necessary.

Service

Services group pods and provide a common DNS name and an optional, load-balanced IP address to access them.

Node

Nodes are the machines that run your containers.



Projects | Project: Ki Be | Add to project | demo

- Overview
- Applications
- Builds
- Resources
- Storage
- Monitoring

Monitoring

All | Filter by name | Hide older resources

Pods

app1-2-x4c7t created 2 days ago	Running	kibe/app1 9d320c3

Deployments

app1-2 created 2 days ago	Active, 1 replica	kibe/app1 9d320c3

Stateful Sets

There are no stateful sets in this project.

Builds

app1-2 created 11 days ago	Complete , ran for 7m 46s	Merge pull request #61 from bparees/sc_metadata 6f2ebfb

Events

3 warnings | View Details

pods/app1-2-x4c7t Started container with docker id 01b882e62094	Started 16 minutes ago
pods/app1-2-x4c7t Created container with docker id 01b882e62094; Security:...	Created 17 minutes ago
pods/app1-2-x4c7t Successfully pulled image "172.30.203.110:5000/kibe/app...	Pulled 17 minutes ago
pods/app1-2-x4c7t pulling image "172.30.203.110:5000/kibe/app1@sha256:9... 3 times in the last 19 minutes	Pulling 17 minutes ago
pods/app1-2-x4c7t Failed to pull image "172.30.203.110:5000/kibe/app1@sha... 2 times in the last 19 minutes	Failed 18 minutes ago
pods/app1-2-x4c7t Error syncing pod, skipping: failed to "StartContainer" for "... 2 times in the last 19 minutes	Failed sync 18 minutes ago
pods/app1-2-x4c7t Error syncing pod, skipping: network is not ready: [SDN po... 3 times in the last 20 minutes	Failed sync 20 minutes ago

RED HAT ENTERPRISE LINUX SERVER

root

openshift.exa...

Cluster

- Overview
- Nodes
- Containers
- Topology
- Details
- Volumes
- Images
- Projects

All Projects

Containers

Name	Pod	Project	Node	Status
registry	docker-registry-2-ws4mf	default	openshift.example.com	running
registry-console	registry-console-1-dzp57	default	openshift.example.com	running
router	router-2-9jhc9	default	openshift.example.com	running
sti-build	app1-1-build	kibe	openshift.example.com	terminated
sti-build	app1-2-build	kibe	openshift.example.com	terminated
app1	app1-2-x4c7t	kibe	openshift.example.com	running
hawkular-cassandra-1	hawkular-cassandra-1-x282k	openshift-infra	openshift.example.com	running
hawkular-metrics	hawkular-metrics-mxbjk	openshift-infra	openshift.example.com	running
gogs	gogs-1-nmwzn	tooling	openshift.example.com	running
jenkins	jenkins-1-b24t3	tooling	openshift.example.com	
nexus	nexus-1-vsbpp	tooling	openshift.example.com	
postgresql	postgresql-gogs-1-z1g74	tooling	openshift.example.com	

Projects

Project
Ki Be

Add to project

demo

Overview

Applications

Builds

Resources

Storage

Monitoring

APP 1

http://app1-kibe.cloudapps.example.com

Build app1, #2 Complete. 11 days ago View Log

app1 has containers without health checks, which ensure your application is running correctly. Add Health Checks

app1

Deployment Config app1 – 2 days ago

#2

CONTAINER: APP1

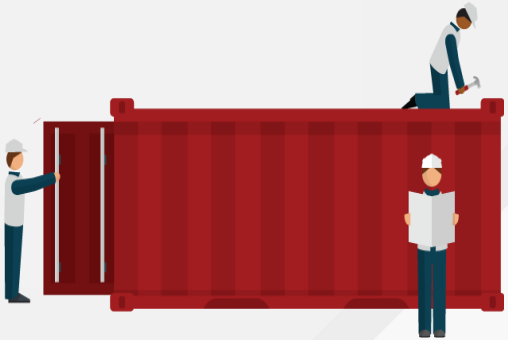
Image: kibe/app1

Ports: 8080/TCP

3
pods

No grouped services.

No services are grouped with app1.



RED HAT ENTERPRISE LINUX SERVER

root

openshift.exa...

Cluster

Overview

Nodes

Containers

Topology

Details

Volumes

Images

Projects

Project: kibe

Select an object to see more details.

Pod

Pods contain one or more containers that run together on a node, containing your application code.

Replication Controller

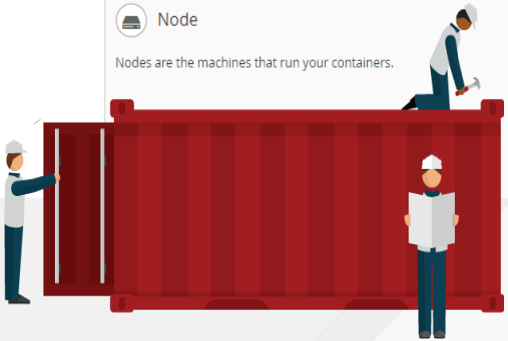
Replication controllers dynamically create instances of pods from templates and remove pods when necessary.

Service

Services group pods and provide a common DNS name and an optional, load-balanced IP address to access them.

Node

Nodes are the machines that run your containers.



Projects | Project: Ki Be | Add to project | demo

Overview
Applications
Builds
Resources
Storage
Monitoring

Deployments » app1 » #2

app1-2 created 2 days ago

Actions

app | app1 | openshift.io/deployment-config.name | app1

Details | Environment | Metrics | Logs | Events

Status: Active

Save | Expand

Stop Following

```
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 10.128.1.4. Set the 'ServerName' directive globally to suppress this message
AH00558: httpd: Could not reliably determine the server's fully qualified domain name, using 10.128.1.4. Set the 'ServerName' directive globally to suppress this message
[Mon Mar 05 03:23:16.359346 2018] [auth_digest:notice] [pid 1] AH01757: generating secret for digest authentication ...
[Mon Mar 05 03:23:16.625290 2018] [http2:warn] [pid 1] AH02951: mod_ssl does not seem to be enabled
[Mon Mar 05 03:23:16.625984 2018] [lbmethod_heartbeat:notice] [pid 1] AH02282: No slotmem from mod_heartmonitor
[Mon Mar 05 03:23:36.176140 2018] [mpm_prefork:notice] [pid 1] AH00163: Apache/2.4.18 (Red Hat) configured -- resuming normal operations
[Mon Mar 05 03:23:36.176186 2018] [core:notice] [pid 1] AH00094: Command line: 'httpd -D FOREGROUND'
```

app1-kibe.cloudapps.example.com

Welcome to your CakePHP application on OpenShift

How to use this example application

For instructions on how to use this application with OpenShift, start by reading the Developer Guide.

Deploying code changes

The source code for this application is available to be forked from the OpenShift GitHub repository. You can configure a webhook in your repository to make OpenShift automatically start a build whenever you push your code:

1. From the Web Console homepage, navigate to your project
2. Click on Browse > Builds
3. Click the link with your BuildConfig name
4. Click the Configuration tab
5. Click the "Copy to clipboard" icon to the right of the "GitHub webhook URL" field
6. Navigate to your repository on GitHub and click on repository settings > webhooks > Add webhook
7. Paste your webhook URL provided by OpenShift
8. Leave the defaults for the remaining fields — that's it!

After you save your webhook, if you refresh your settings page you can see the status of the ping that Github sent to OpenShift to verify it can reach the server.

Note: adding a webhook requires your OpenShift server to be reachable from GitHub.

Working in your local Git repository

If you forked the application from the OpenShift GitHub example, you'll need to manually clone the repository to your local system. Copy the application's source code Git URL and then run:

Managing your application

Documentation on how to manage your application from the Web Console or Command Line is available at the Developer Guide.

Web Console

You can use the Web Console to view the state of your application components and launch new builds.

Command Line

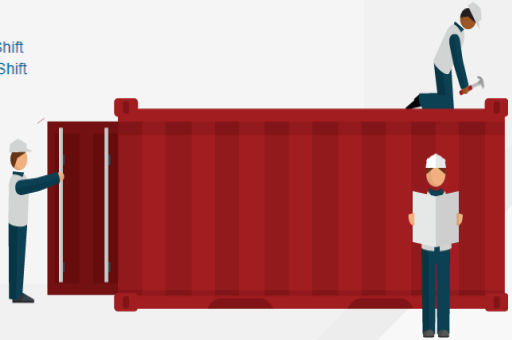
With the OpenShift command line interface (CLI), you can create applications and manage projects from a terminal.

Development Resources

- OpenShift Documentation
- Openshift Origin GitHub
- Source To Image GitHub
- Getting Started with PHP on OpenShift
- Stack Overflow questions for OpenShift
- Git documentation

Request information

Page view count: No database configured



Install OpenShift Container Platform

```
# subscription-manager register
# subscription-manager refresh
# subscription-manager list --available
# subscription-manager attach --pool=<pool_id>

# subscription-manager repos --enable="rhel-7-
server-rpms" \
    --enable="rhel-7-server-extras-rpms" \
    --enable="rhel-7-server-ose-3.11-rpms" \
    --enable="rhel-7-server-ansible-2.6-rpms"
```

```
# yum -y install wget git net-tools bind-utils
iptables-services bridge-utils bash-completion
kexec-tools sos psacct

# yum -y update
# reboot

# yum -y install openshift-ansible

# yum -y install cri-o docker
```

```
$ ssh-keygen

$ for host in master.openshift.example.com \
    node.openshift.example.com; \
    do ssh-copy-id -i ~/.ssh/id_rsa.pub $host; \
    done

$ cd /usr/share/ansible/openshift-ansible
$ ansible-playbook -i <inventory_file>
playbooks/prerequisites.yml

$ cd /usr/share/ansible/openshift-ansible
$ ansible-playbook -i <inventory_file>
playbooks/deploy_cluster.yml
```

Configure OpenShift Container Platform

```
$ oc login -u system:admin
$ oc get nodes
oauthConfig:
  ...
  identityProviders:
  - challenge: true
    login: true
    name: htpasswd_auth provider
    provider:
      apiVersion: v1
      kind: HTPasswdPasswordIdentityProvider
      file: /etc/origin/master/htpasswd
```

```
# yum -y install httpd-tools

# touch /etc/origin/master/htpasswd

# htpasswd -b /etc/origin/master/htpasswd admin
redhat

# master-restart api

# master-restart controllers
```

```
$ oc adm policy add-cluster-role-to-user cluster-admin admin

$ oc login -u admin

$ oc project default

$ oc delete all -l router=router

$ oc adm router --replicas=1 --service-account=router
```

Creating a Project

```
$ oc login -u=<username> -p=<password> --server=<your-openshift-server> --insecure-skip-tls-verify
$ oc login <https://api.your-openshift-server.com> --token=<tokenID>
$ oc new-project <projectname> --description="<description>" --display-name="<display_name>"
$ oc new-app openshift/ruby-20-centos7~https://github.com/<your_github_username>/ruby-ex
```

```
$ oc logs -f bc/ruby-ex

$ oc status

$ oc expose service ruby-ex
$ oc get route

$ oc edit bc/ruby-ex
triggers
- github:
    secret: Q1tGY0i9f1ZFihQbX07S
    type: GitHub

$ oc describe bc ruby-ex
```

RED HAT OPENSHIFT Container Platform

OPENSHIFT CONTAINER PLATFORM

Username caio_candido-dell.com

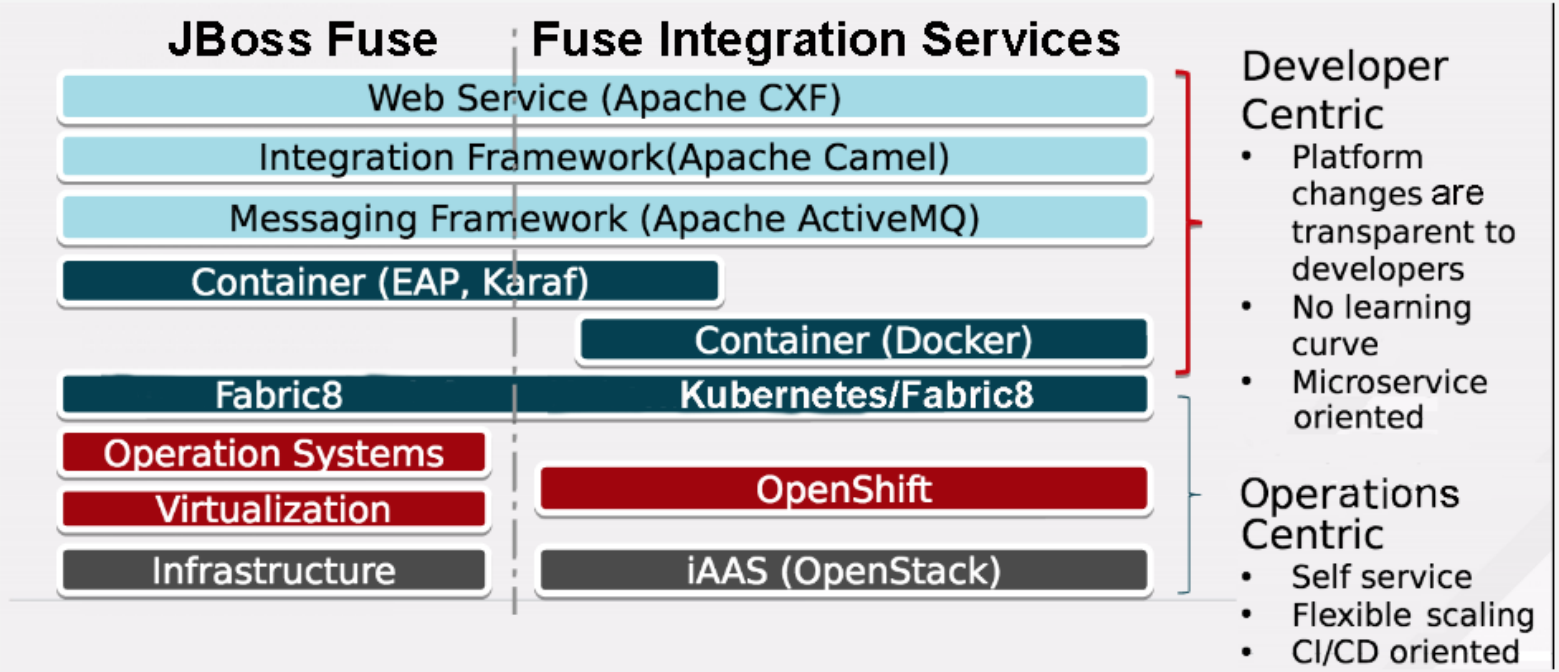
Password ••••••••

Log In

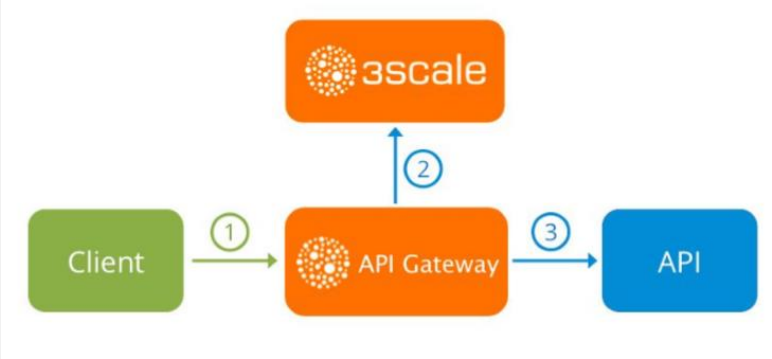
Welcome to the OpenShift Container Platform.

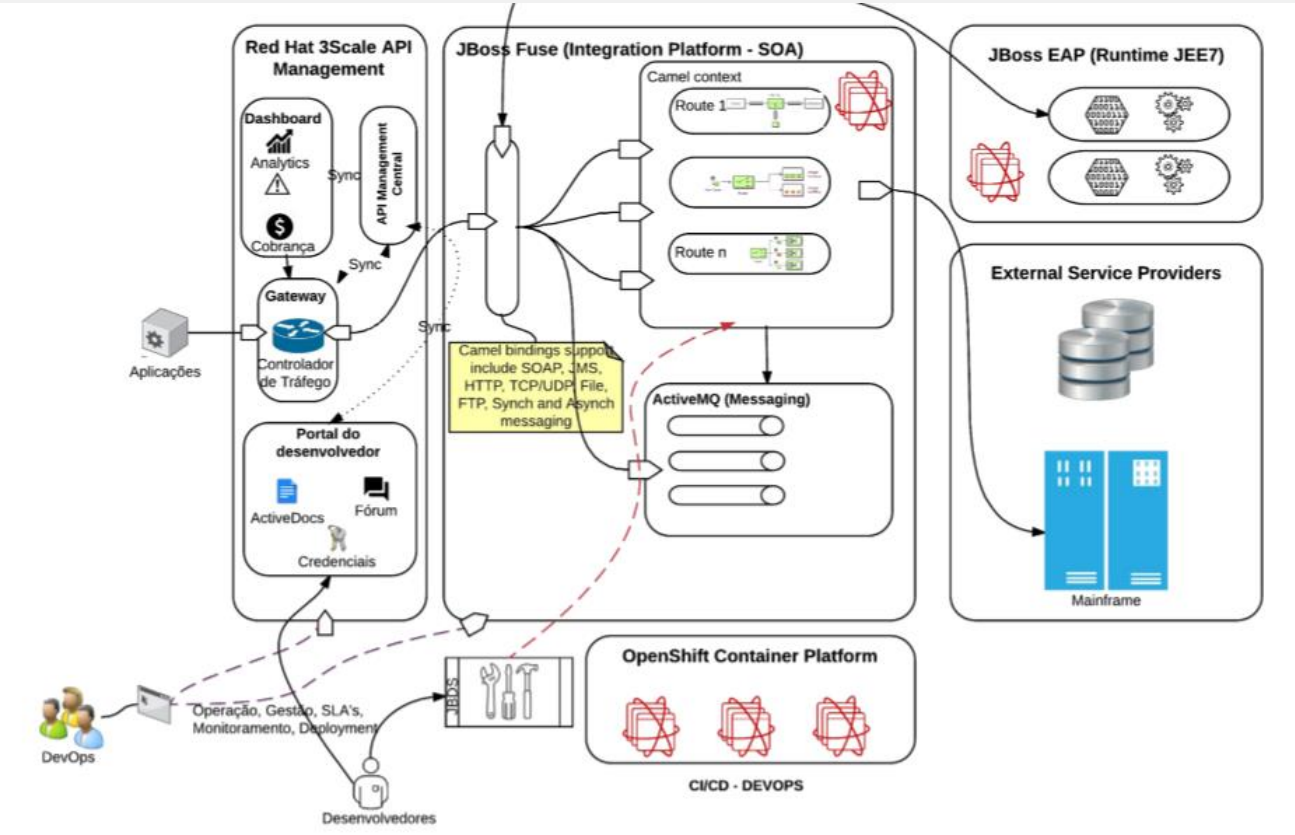
Fuse Integration Services

Architecture

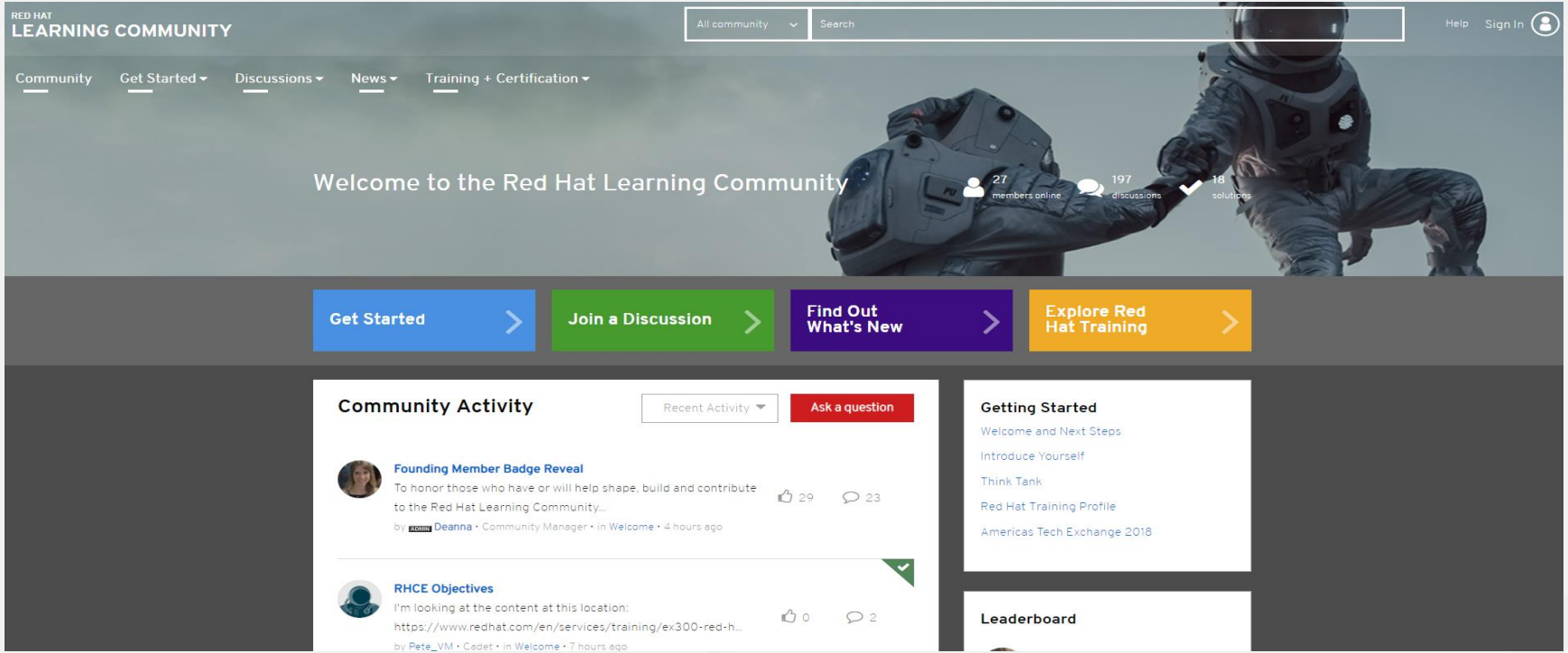



CHALLENGERS
LEADERS
Google (Apigee)
CA Technologies
IBM
Software AG
MuleSoft
Amazon Web Services
SAP
TIBCO Software
Red Hat (3scale)
Axway
Oracle
Microsoft
Cloud Elements
WSO2
Sensedia
SmartBear
Kong
Tyk
digitalML
Torry Harris Business Solutions
Dell Boomi
Kony
ABILITY TO EXECUTE
NICHE PLAYERS
VISIONARIES
COMPLETENESS OF VISION
As of April 2018
© Gartner, Inc


3scale
2
Client
1
API Gateway
3
API


Red Hat 3Scale API Management
Dashboard
Analytics
Cobrança
Sync
API Management Central
Sync
Gateway
Controlador de Tráfego
Aplicações
Sync
Portal do desenvolvedor
ActiveDocs
Fórum
Credenciais
JBoss Fuse (Integration Platform - SOA)
Camel context
Route 1
Route n
Camel bindings support include SOAP, JMS, HTTP, TCP/UDP, File, FTP, Synch and Asynch messaging
ActiveMQ (Messaging)
JBoss EAP (Runtime JEE7)
External Service Providers
Mainframe
OpenShift Container Platform
CI/CD - DEVOPS
DevOps
Operação, Gestão, SLA's, Monitoramento, Deployment
Desenvolvedores

Como aprender tudo isso?



Como começar hoje?

https://learn.openshift.com

RED HAT OPENSHIFT

PRODUCTS LEARN COMMUNITY SUPPORT FREE TRIAL REPORT AN ISSUE

Interactive Learning Portal

Our Interactive Learning Scenarios provide you with a pre-configured OpenShift® instance, accessible from your browser without any downloads or configuration. Use it to experiment, learn OpenShift and see how we can help solve real-world problems.

Foundations of OpenShift START COURSE	Building Applications On OpenShift START COURSE	Subsystems, Components, and Internals START COURSE
OpenShift Playgrounds START COURSE	Service Mesh Workshop with Istio START COURSE	Serverless Scenarios with OpenShift Cloud Functions START COURSE

DELL EMC redhat.

Obrigado!

powered by

Caio Candido,
caio.candido@dell.com
https://www.linkedin.com/in/ccandido/

RED HAT CERTIFIED ARCHITECT

111-200-002