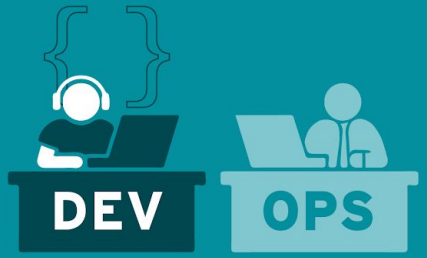


Upgrading your developer superpowers with Kubernetes and OpenShift

DevNexus, Atlanta

redhat.

Presenters

Steve Pousty (theSteve0)

Director of Developer Advocacy,
Red Hat Middleware

US

Jorge Morales

Field Product Manager &
Developer Advocate

Spain

Agenda

09:00 – 10:30 Introduction (theSteve0 and Jorge)

10:30 – 10:45 Refreshment Break

10:45 – 12:00 Labs (theSteve0 and Jorge)

12:00 – 13:00 Lunch

13:00 – 15:00 Labs (theSteve0 and Jorge)

15:00 – 15:30 Refreshment Break

15:30 – 18:00 Recap, demos and Q&A (theSteve0 and Jorge)

Goals

1. Introduction to OpenShift/Kubernetes (assume Docker/Container knowledge)
2. HANDS ON
3. Have fun and survive

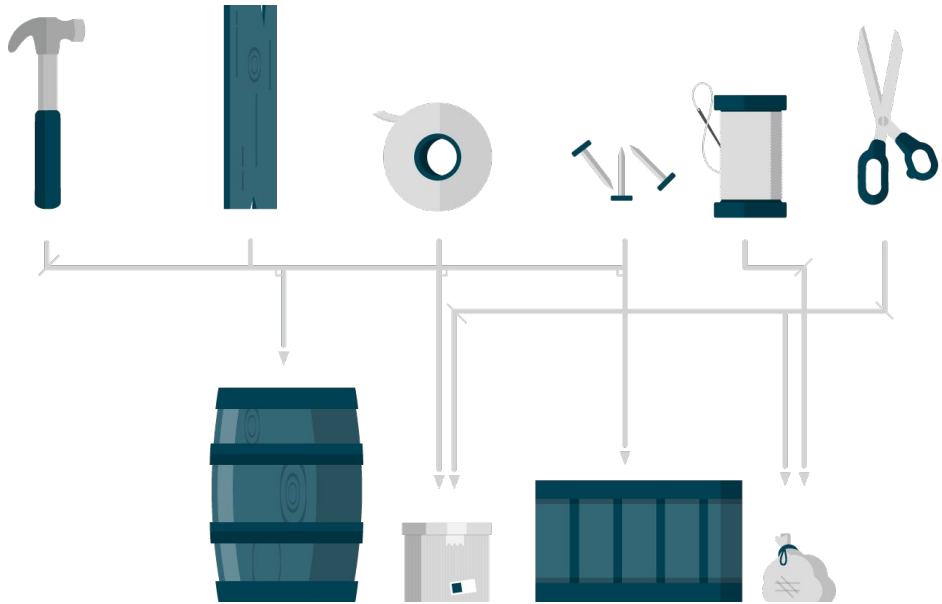
OPENSHIFT OVERVIEW

So what is the point of OpenShift

THE PROBLEM

Every application delivered by I.T.:

- Can have different requirements
- Can use different languages, databases, and tools.



THE PROBLEM

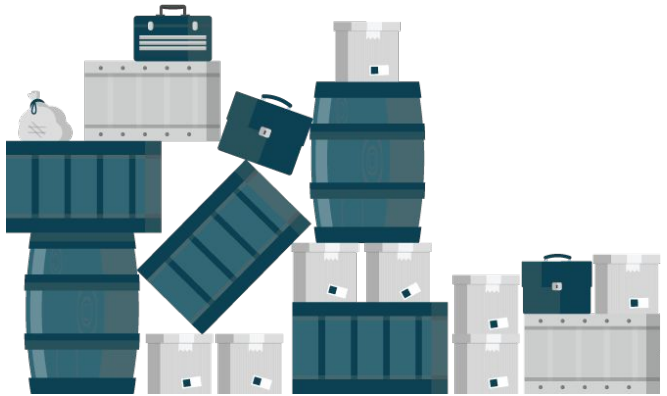
To deploy, configure, manage, and maintain this complexity takes:

- People
- Expertise
- The right systems, infrastructure, and architecture

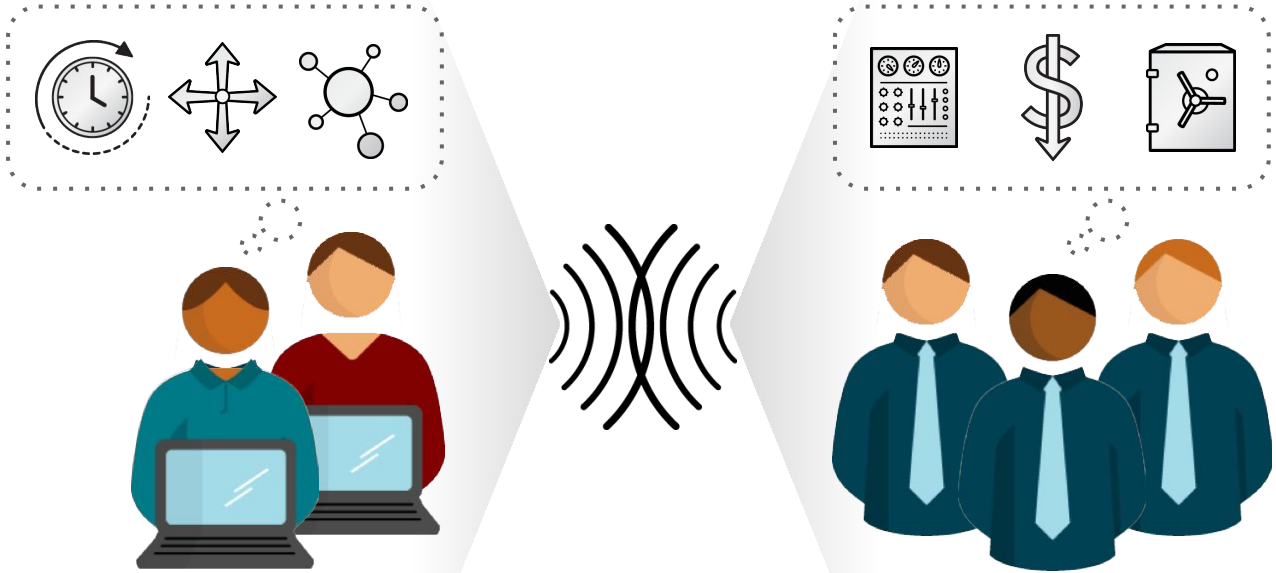
This costs time and money.

THE PROBLEM

Applications require complicated installation and integration every time they are deployed

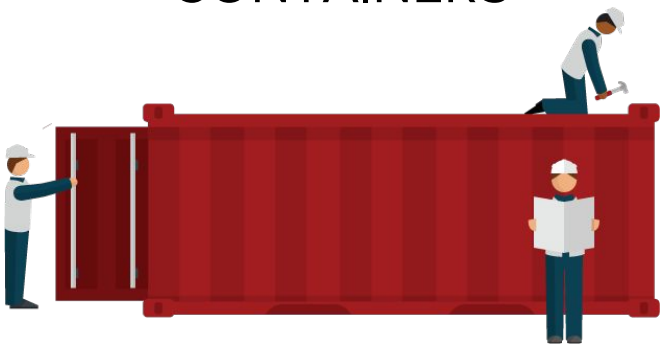


THE PROBLEM



THE SOLUTION

CONTAINERS



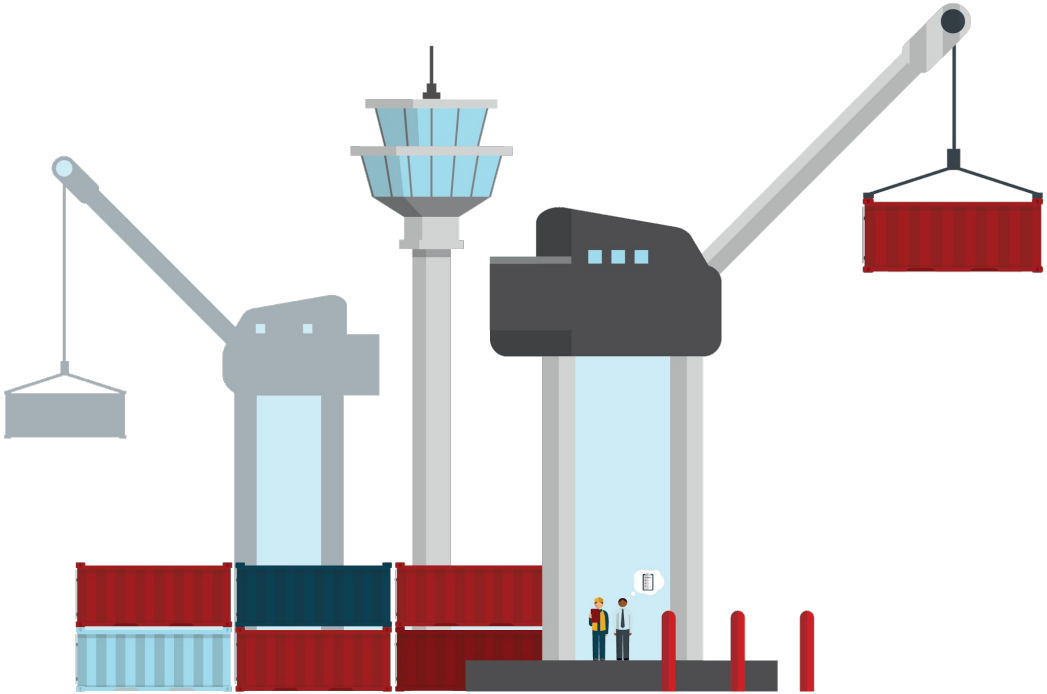
Adopting a container-based strategy will allow applications to be easily shared and deployed.

APPLICATIONS REQUIRE **MANY** CONTAINERS.

HOW DO YOU DELIVER AND MANAGE THEM AT SCALE?

BUILD, DEPLOY, AND MANAGE AT SCALE

WITH RED HAT OPENSHIFT CONTAINER PLATFORM



The industry's most secure and comprehensive enterprise-grade container platform based on industry standards, Docker and Kubernetes.

RED HAT® OPENSHIFT Container Platform

Self-Service

Multi-language

Automation

Collaboration

Seamless

Standards-based

Web-scale

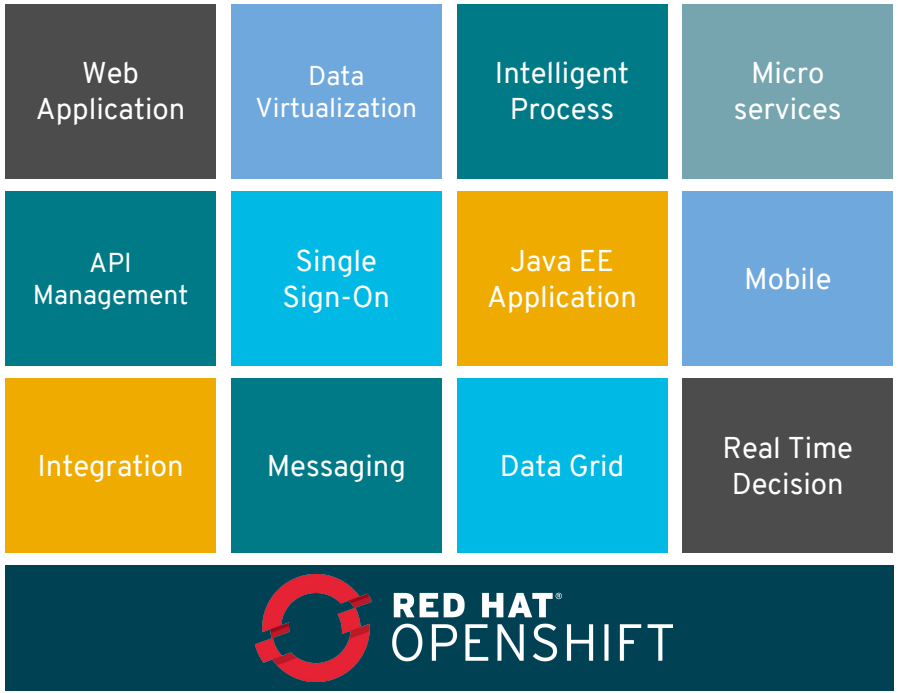
Open Source

Enterprise Grade

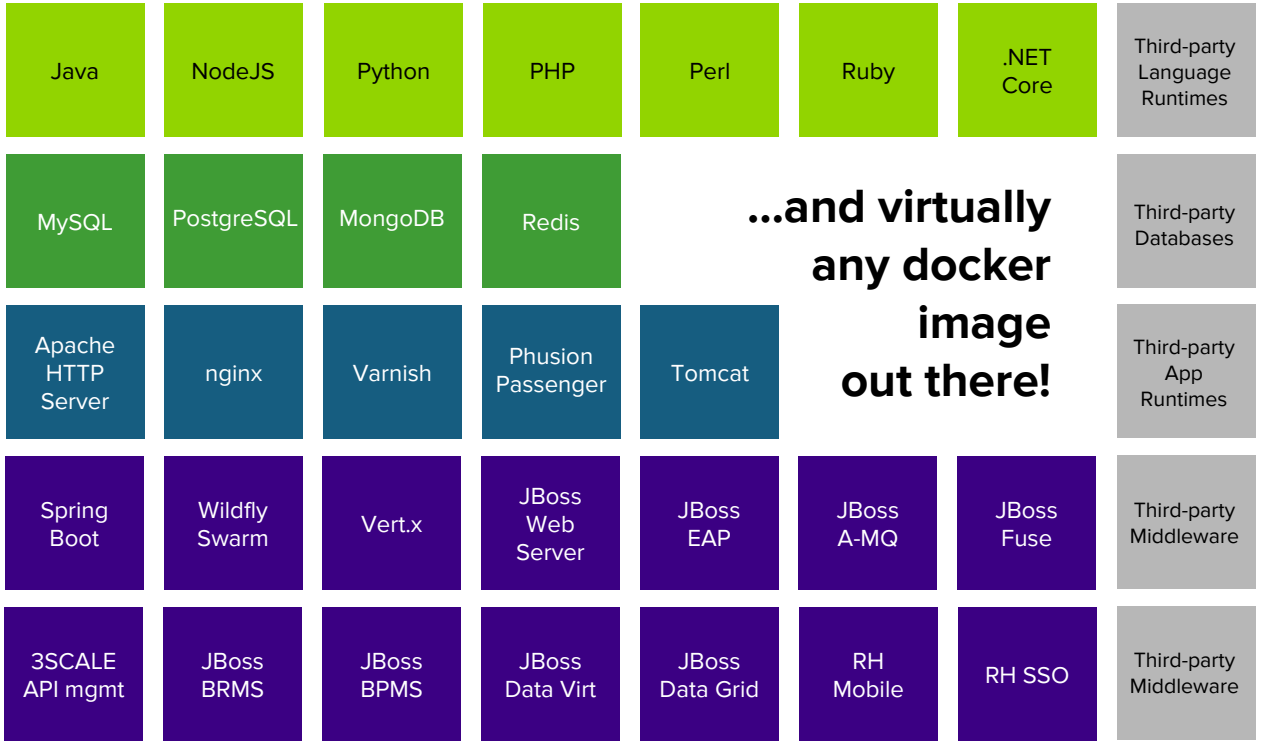
Secure

redhat.

A PLATFORM THAT GROWS WITH YOUR BUSINESS



TRUE POLYGLOT PLATFORM



CrunchyData

GitLab

Iron.io

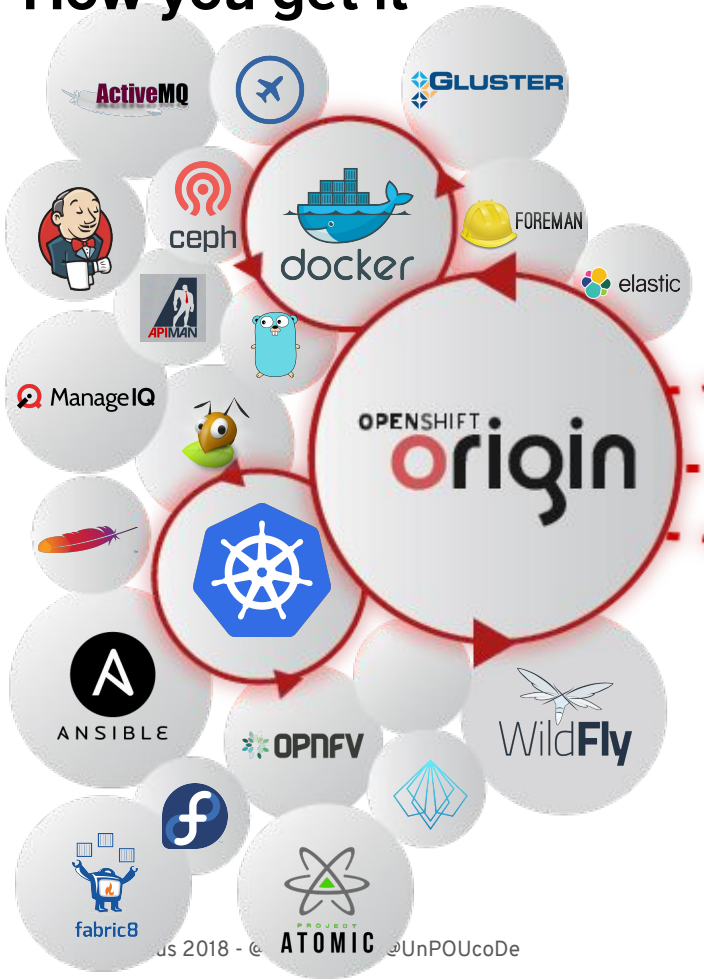
Couchbase

Sonatype

EnterpriseDB

and many more

How you get it



RED HAT® OPENSHIFT Container Platform

RED HAT® OPENSHIFT Dedicated

RED HAT® OPENSHIFT Online

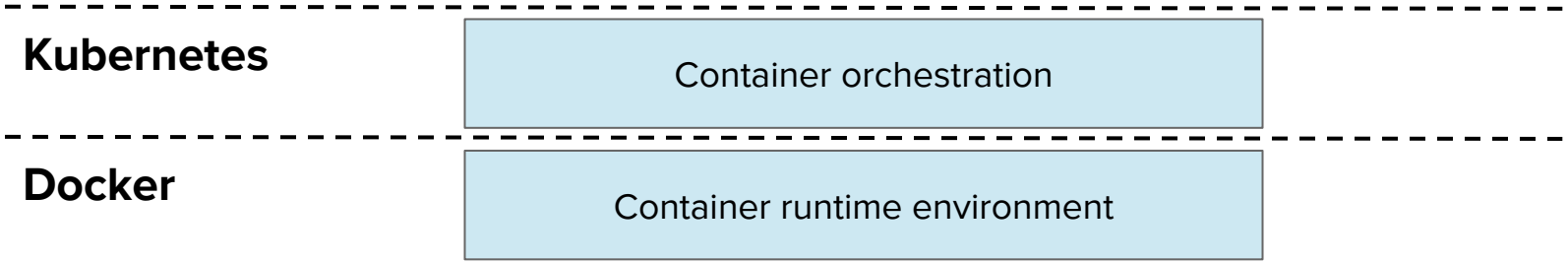
redhat.

Containers...

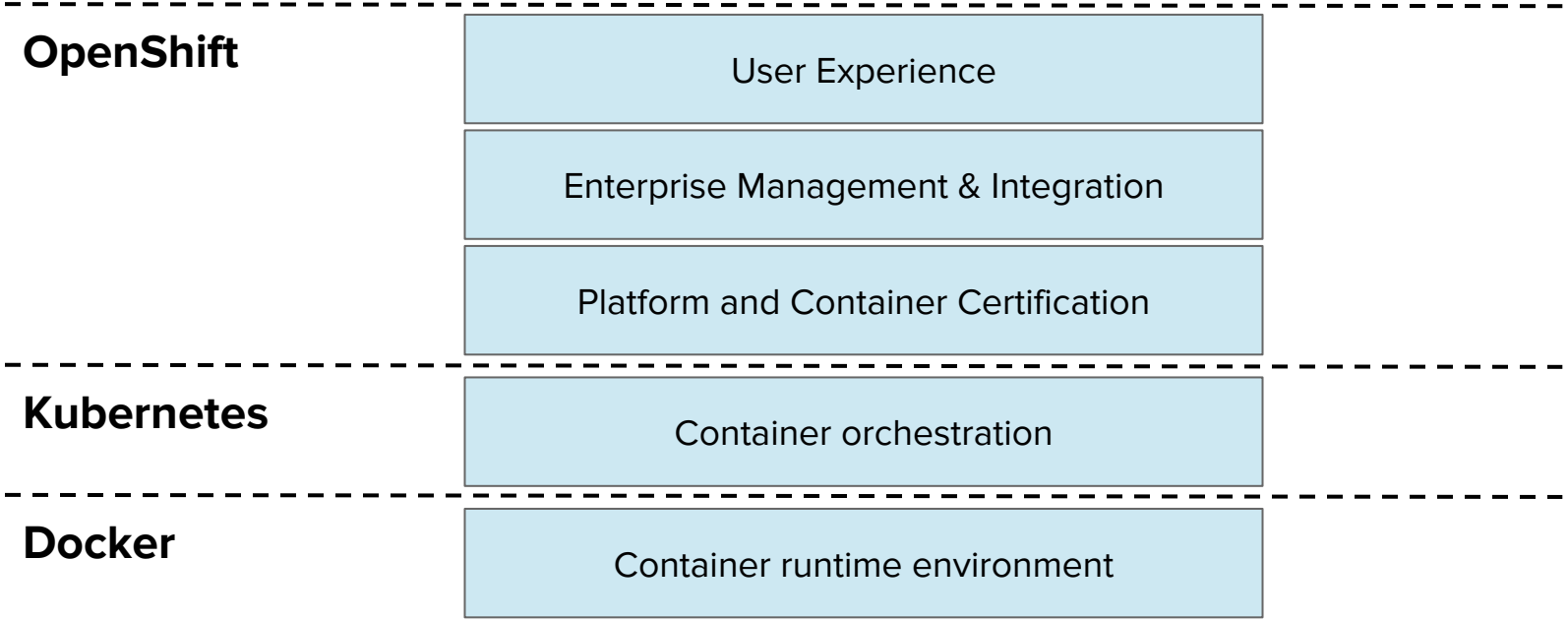
Docker

Container runtime environment

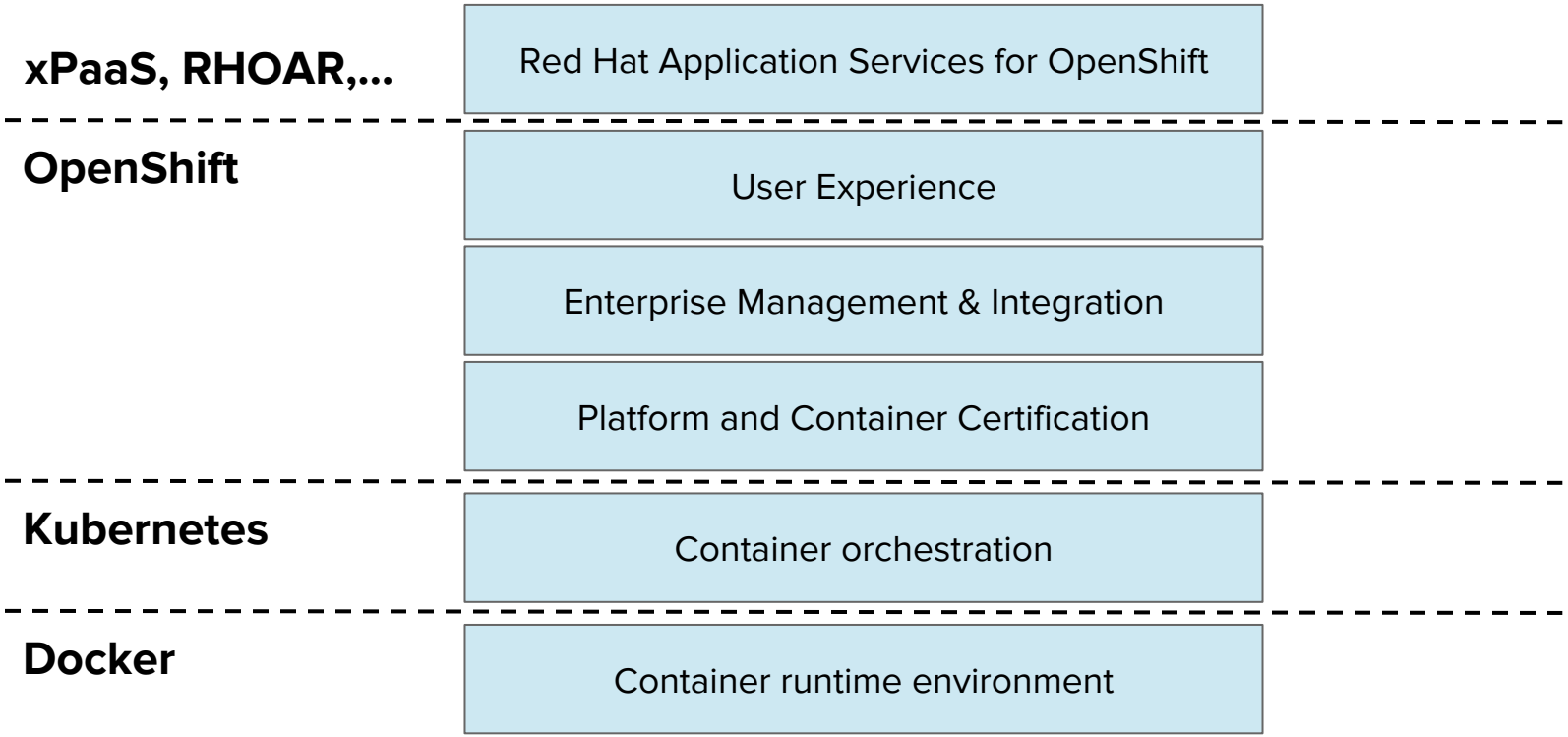
Orchestrated Containers...



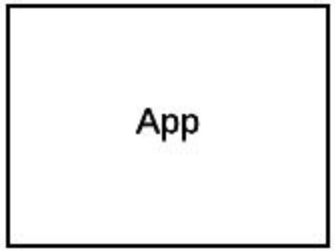
Orchestrated Containers for the Enterprise

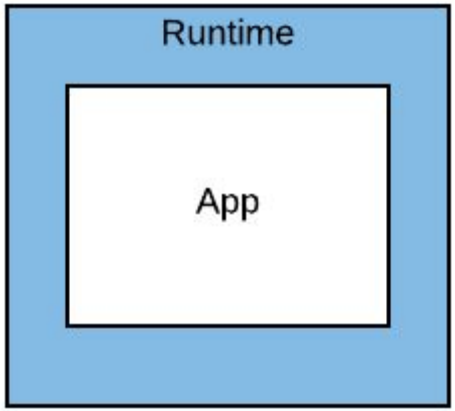


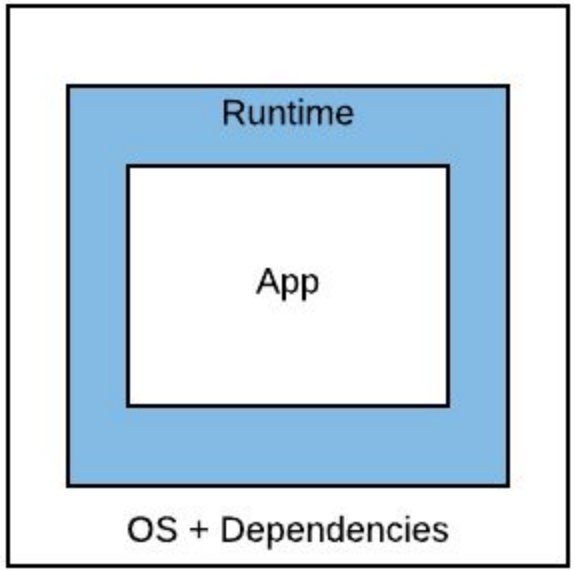
Orchestrated Containers for the Enterprise

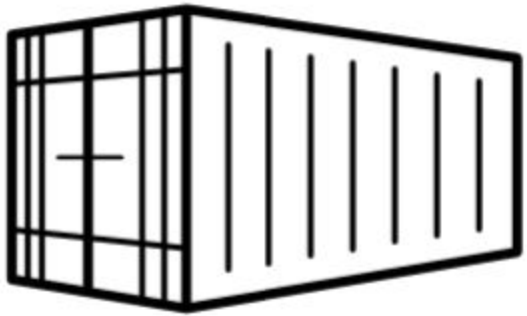


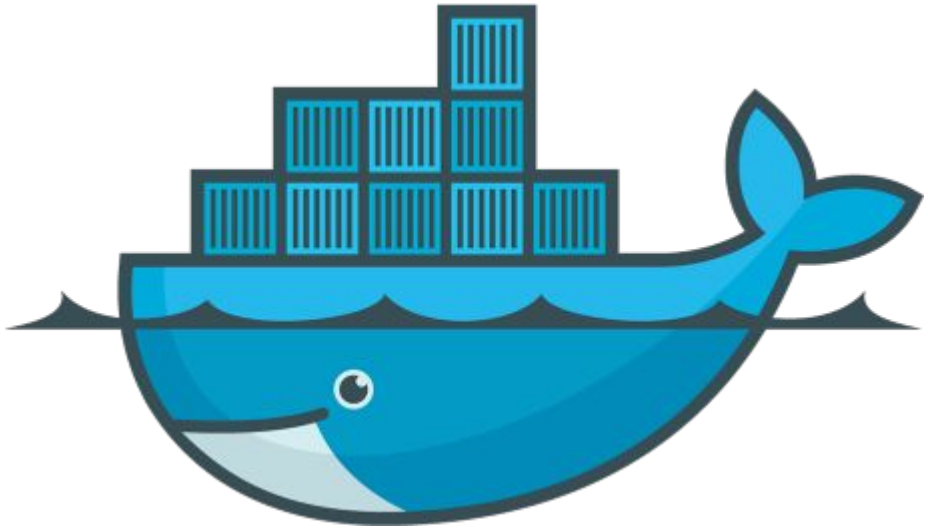
OPENSHIFT CONCEPTS OVERVIEW

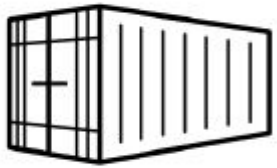




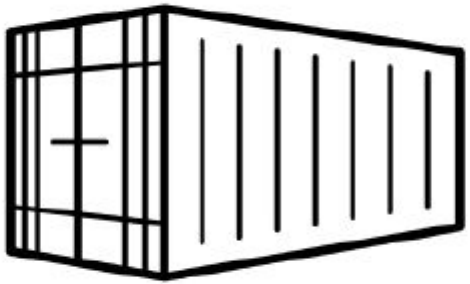






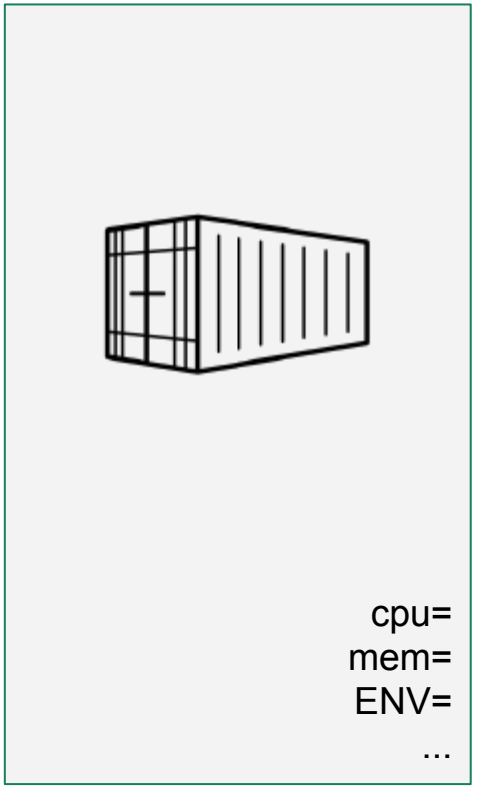


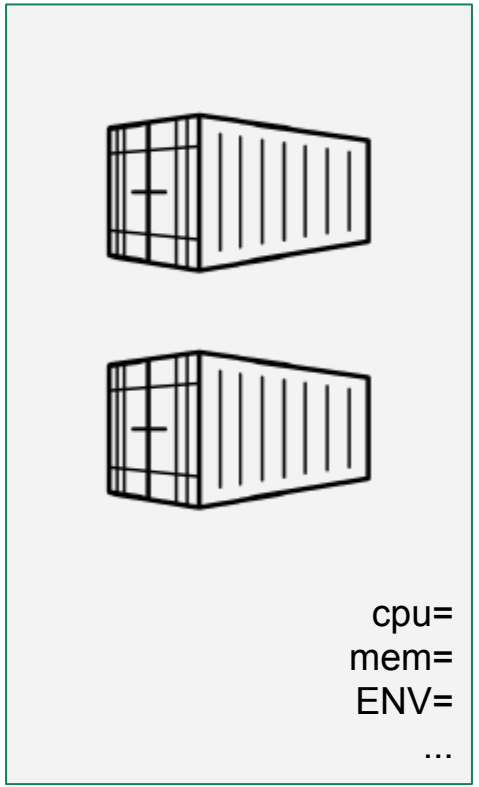
1 cpu
512 MB

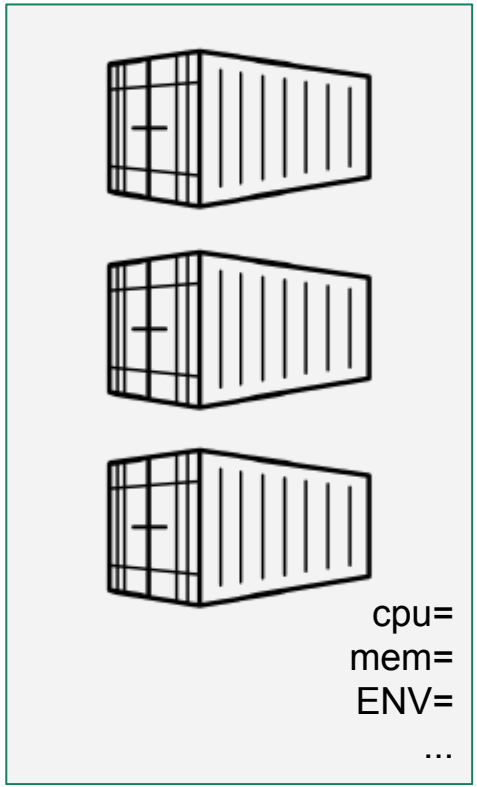


2 cpu
1 GB

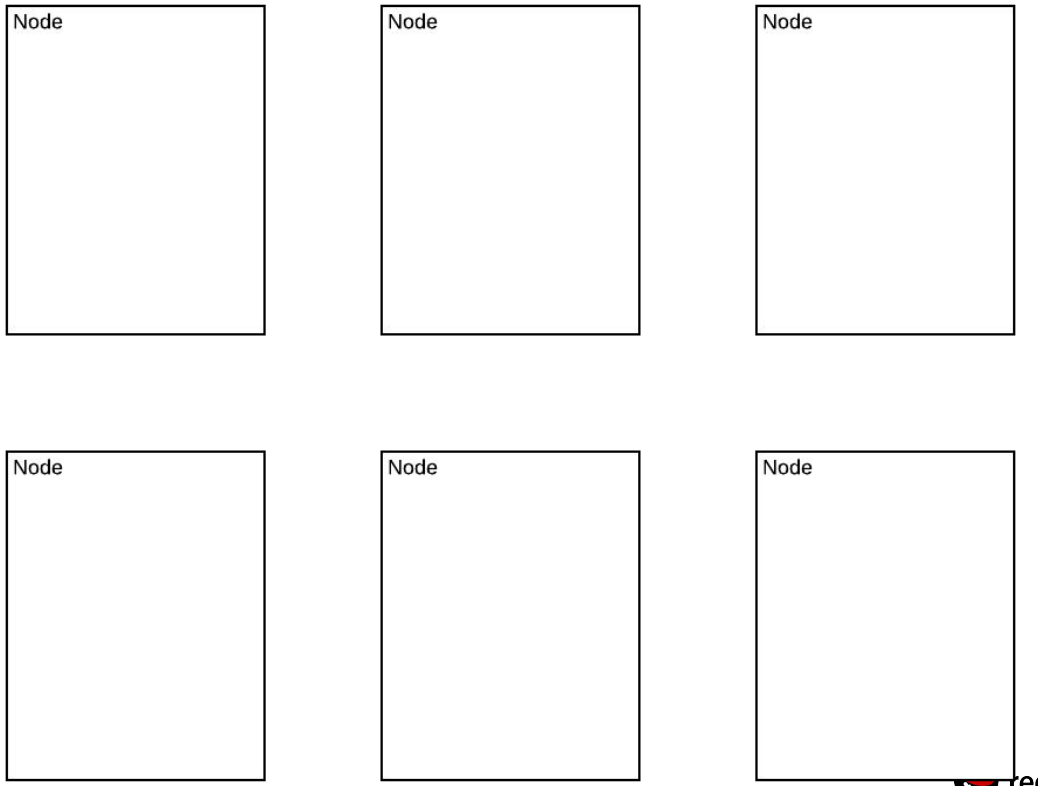
Pod



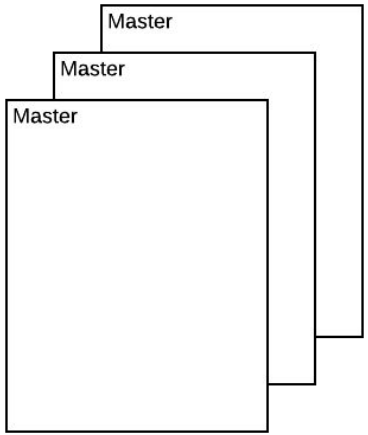


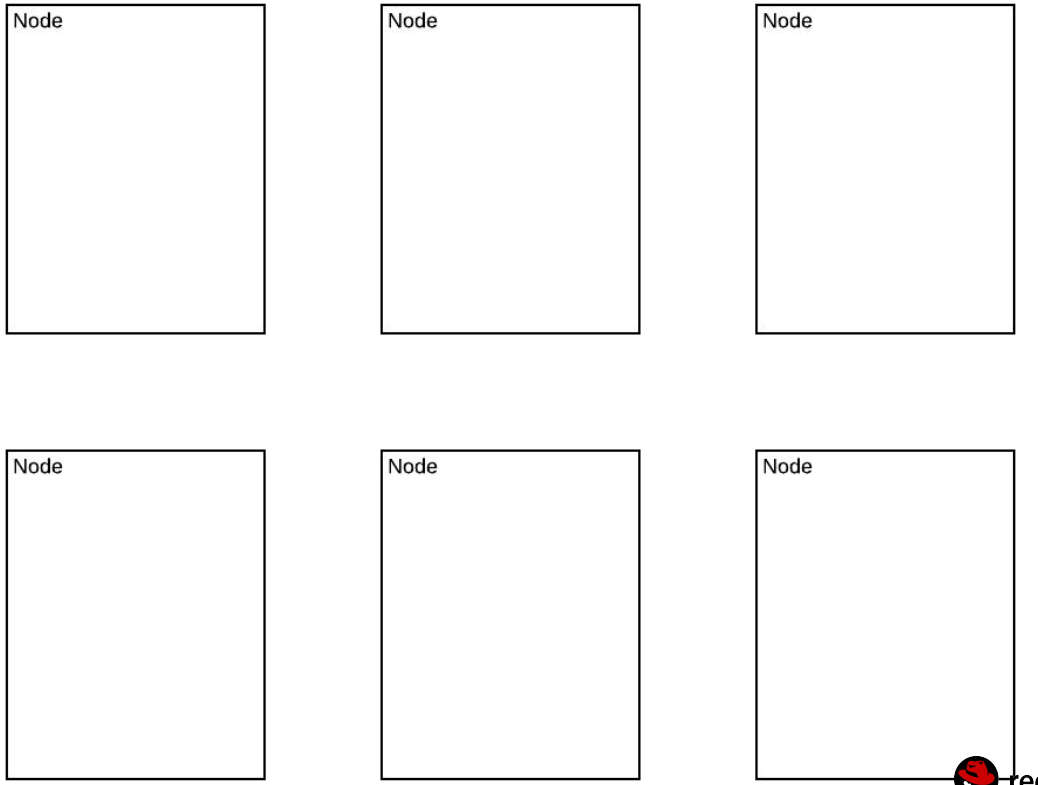


Nodes

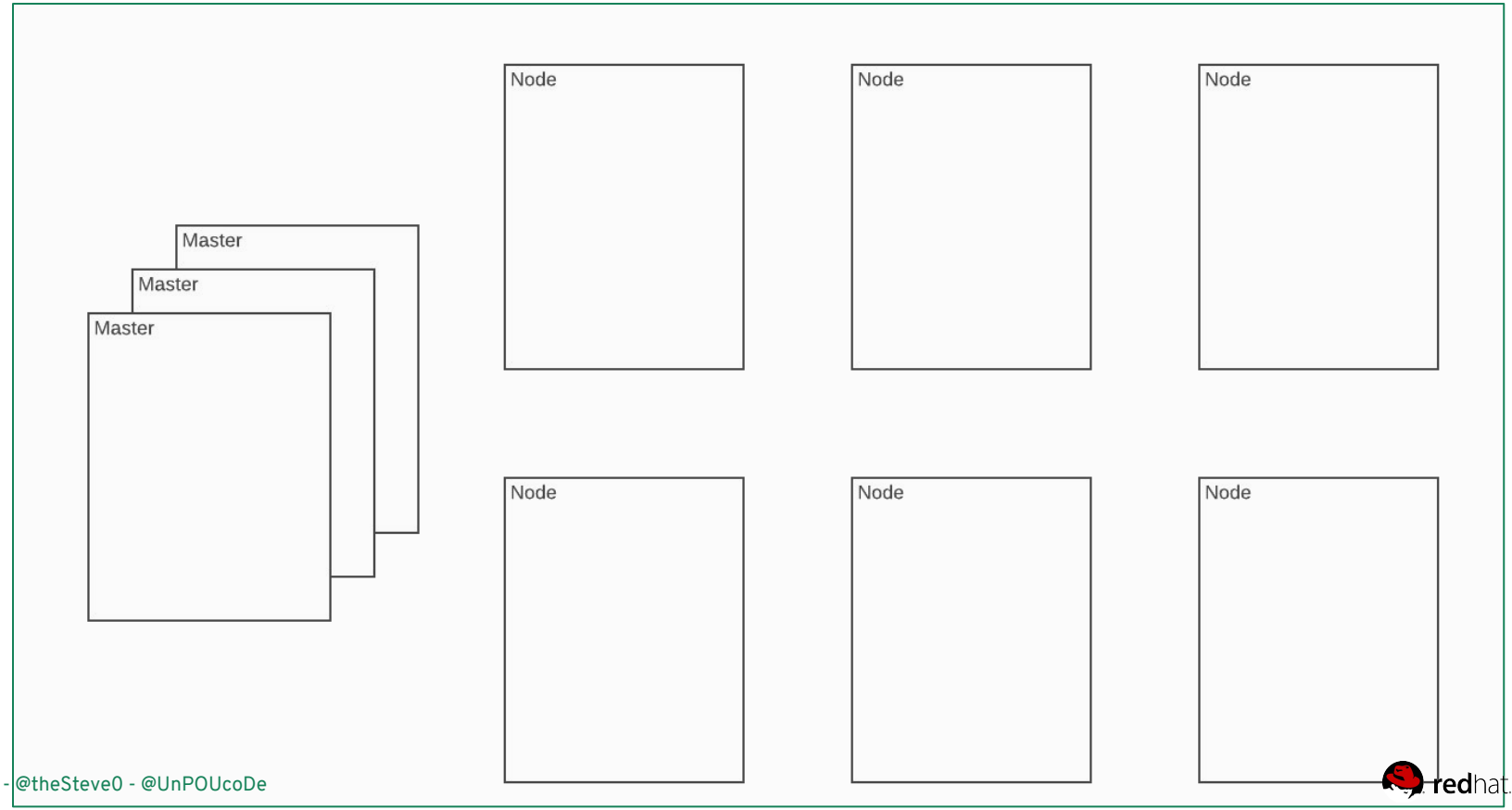


Master nodes

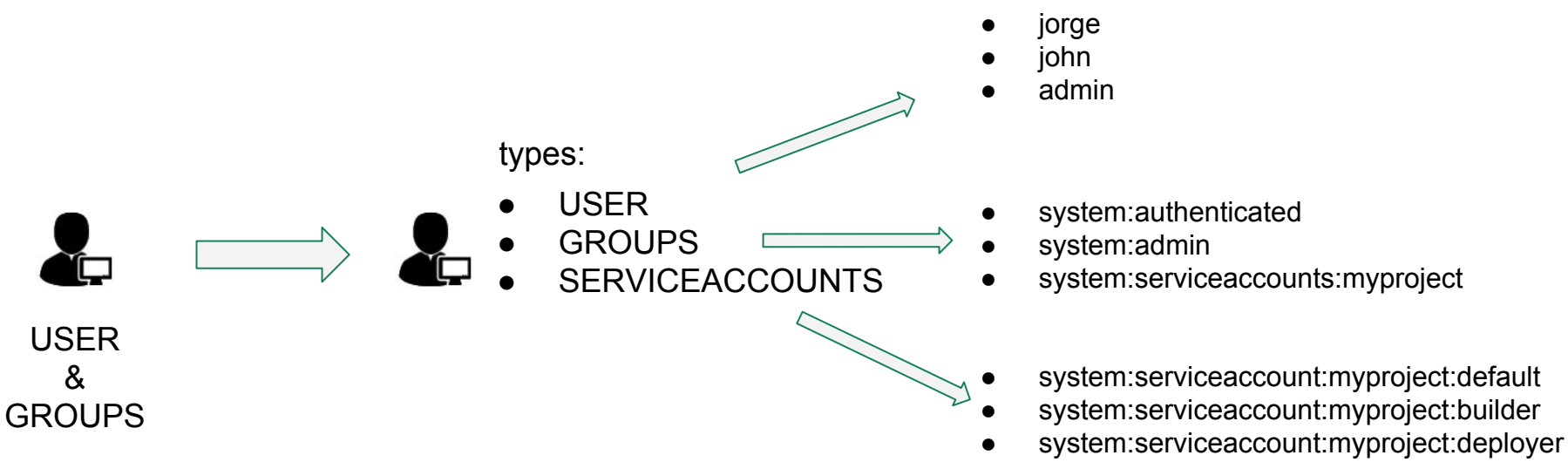




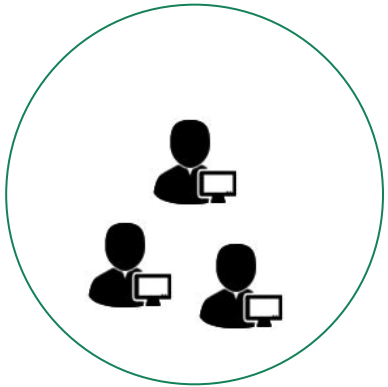
OpenShift Cluster



Users and Groups



Scopes / Multitenancy



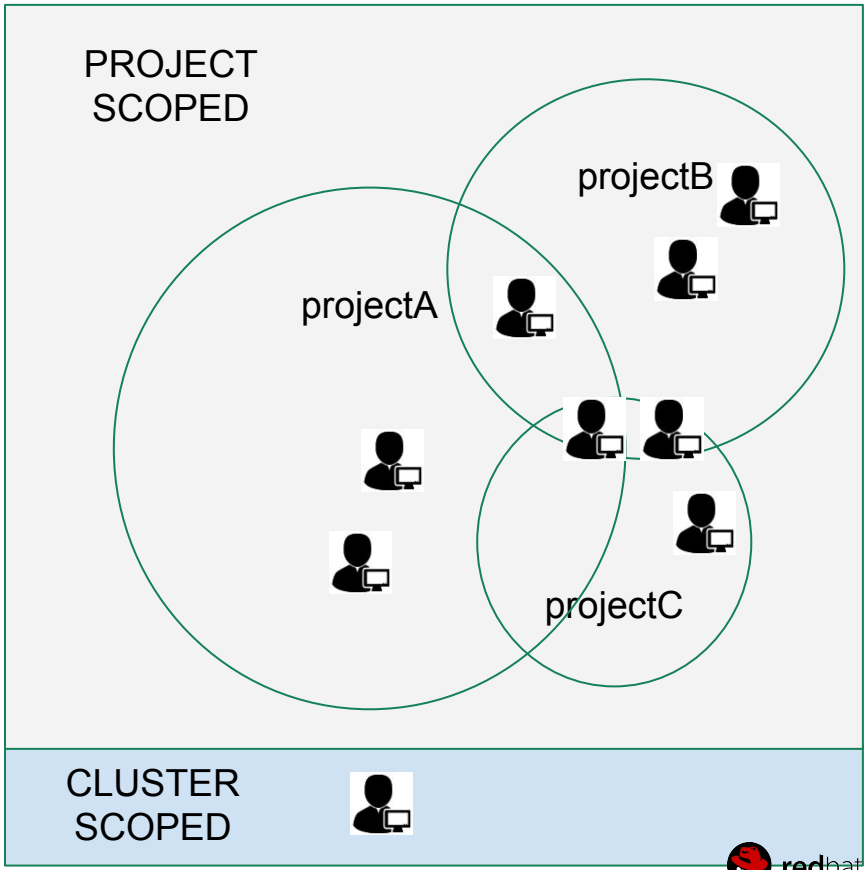
PROJECT (a.k.a namespace)

Constrains:

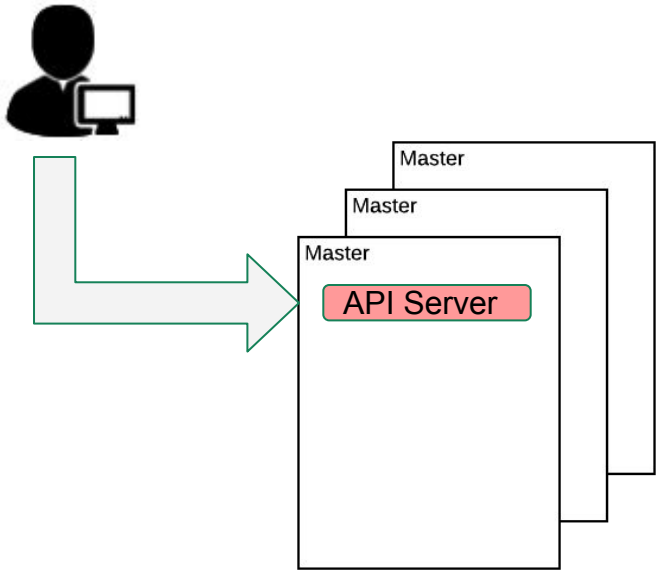
- Resources
- Visibility

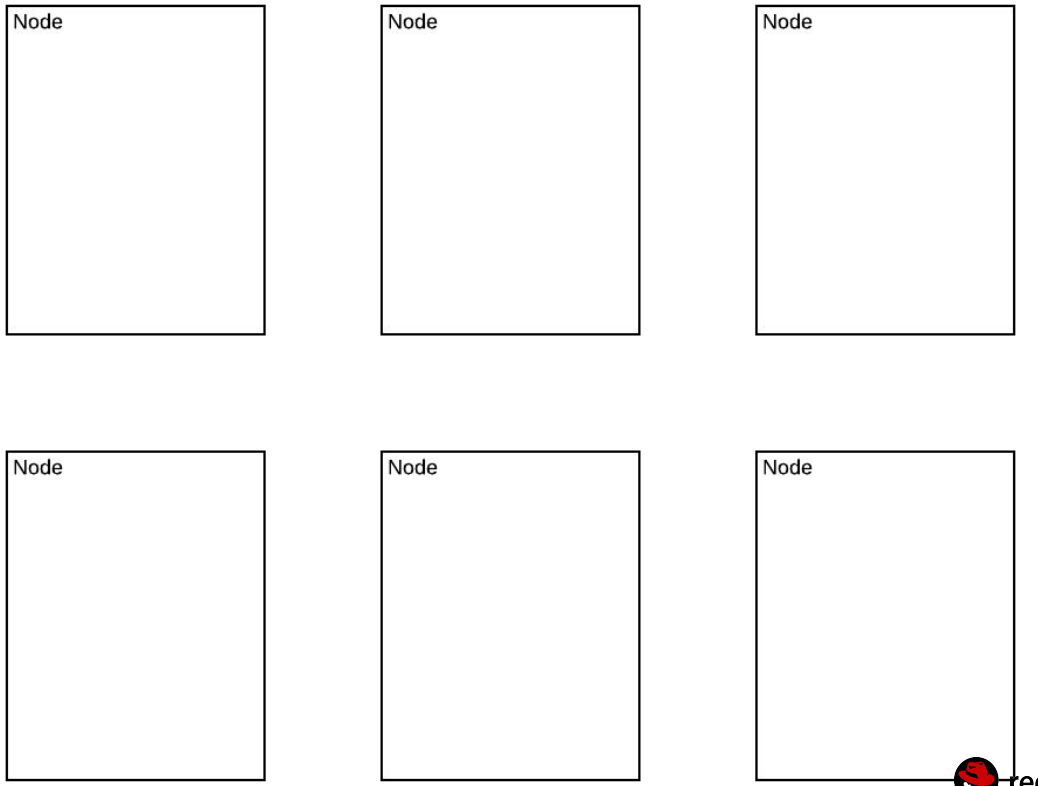
View project

View all

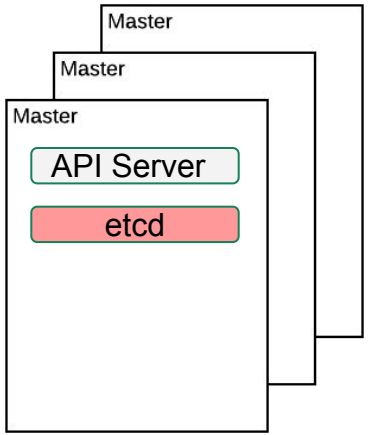


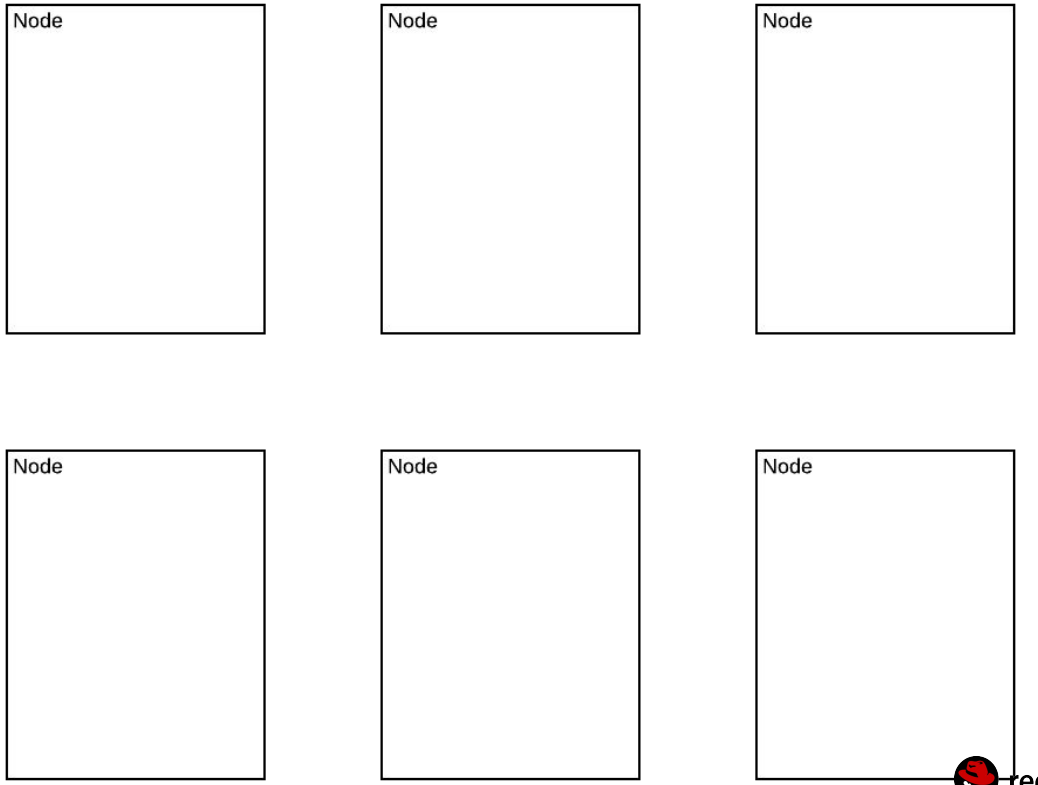
Authn & Authz

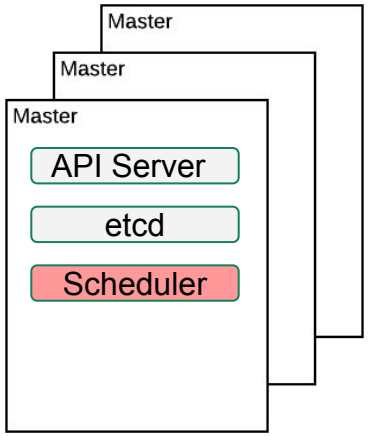




Store state







Node

Node

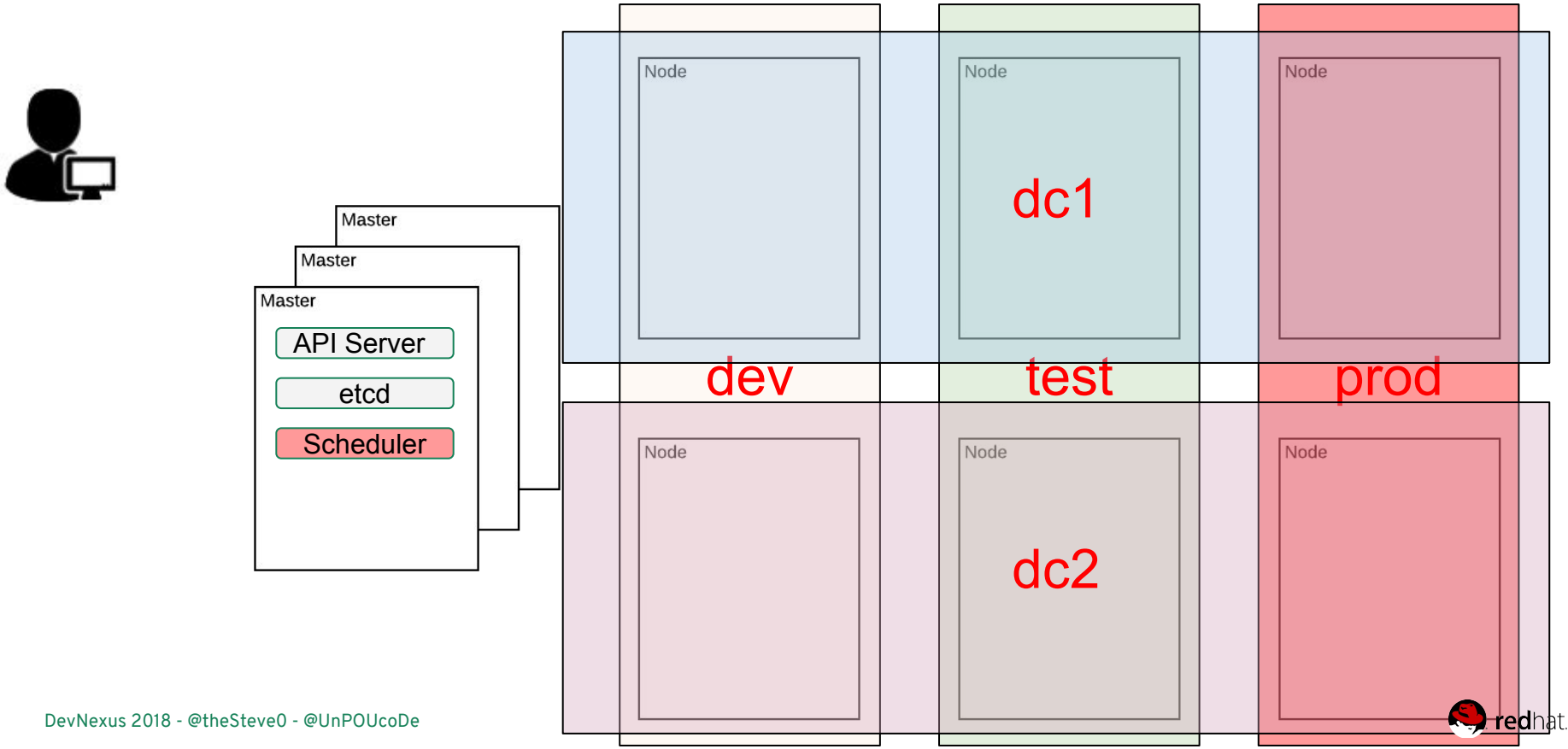
Node

Node

Node

Node

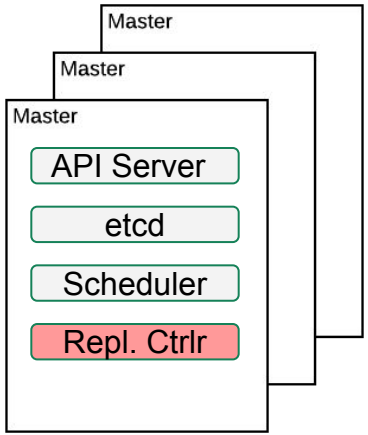
Workload placement

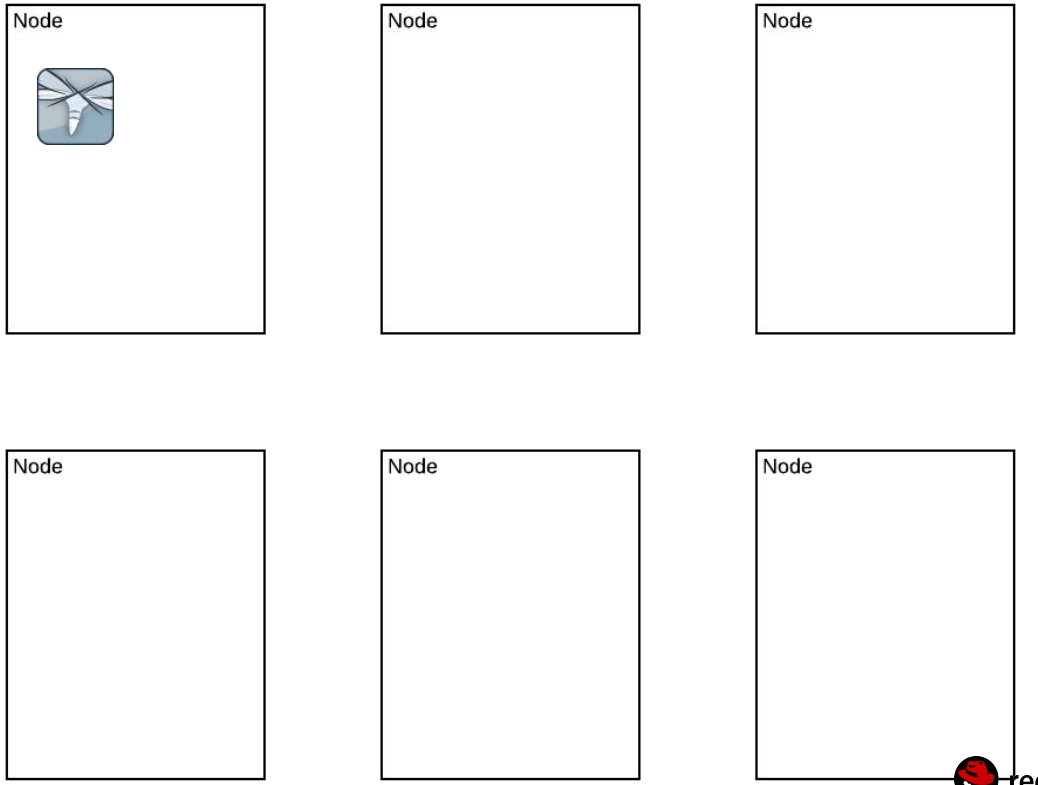


Workloads

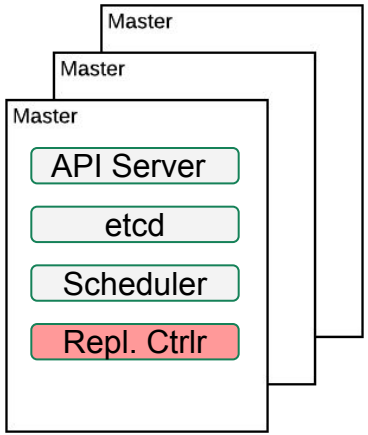
- Stateless
  - **ReplicaSet** (also known as ReplicationController)
- Stateful
  - **StatefulSet** (previously known as PetSet)
- Batch
  - **Jobs**
  - **CronJobs**
- Daemon
  - **DaemonSet**

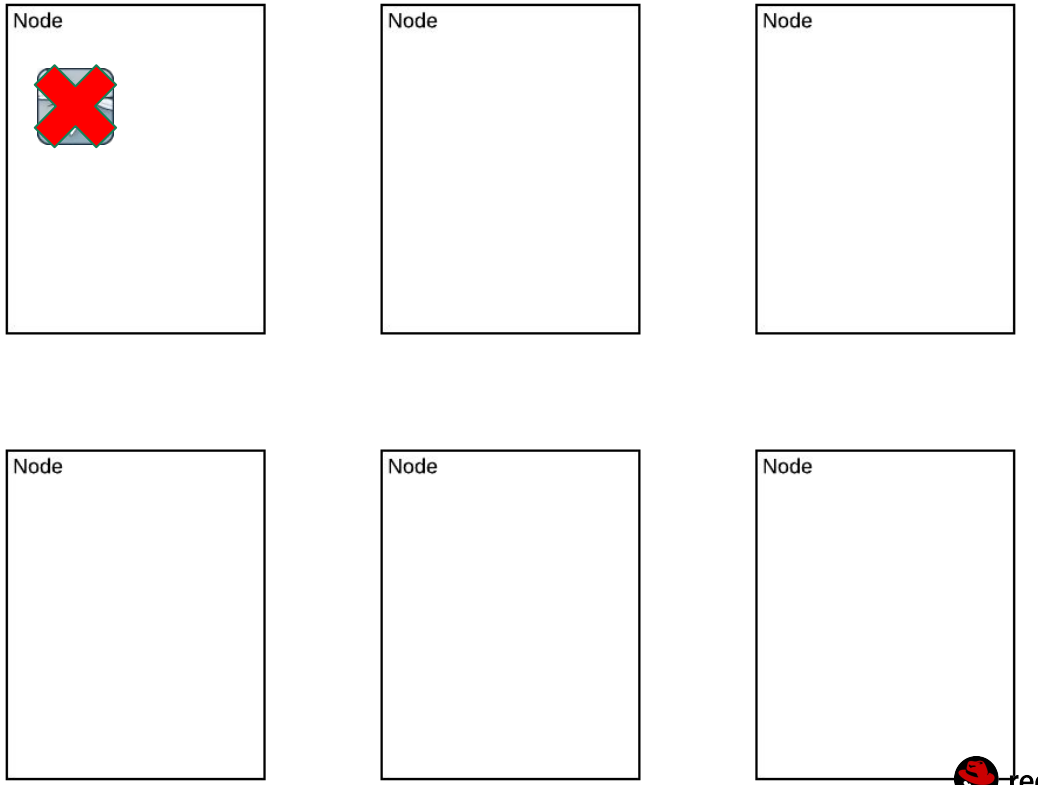
Workload execution guarantees



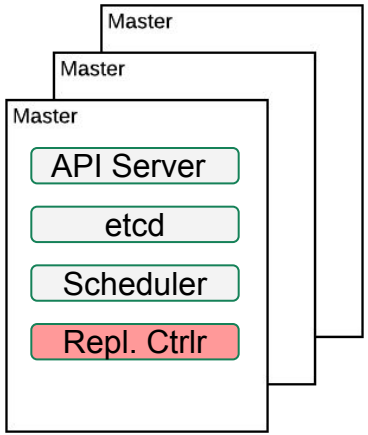


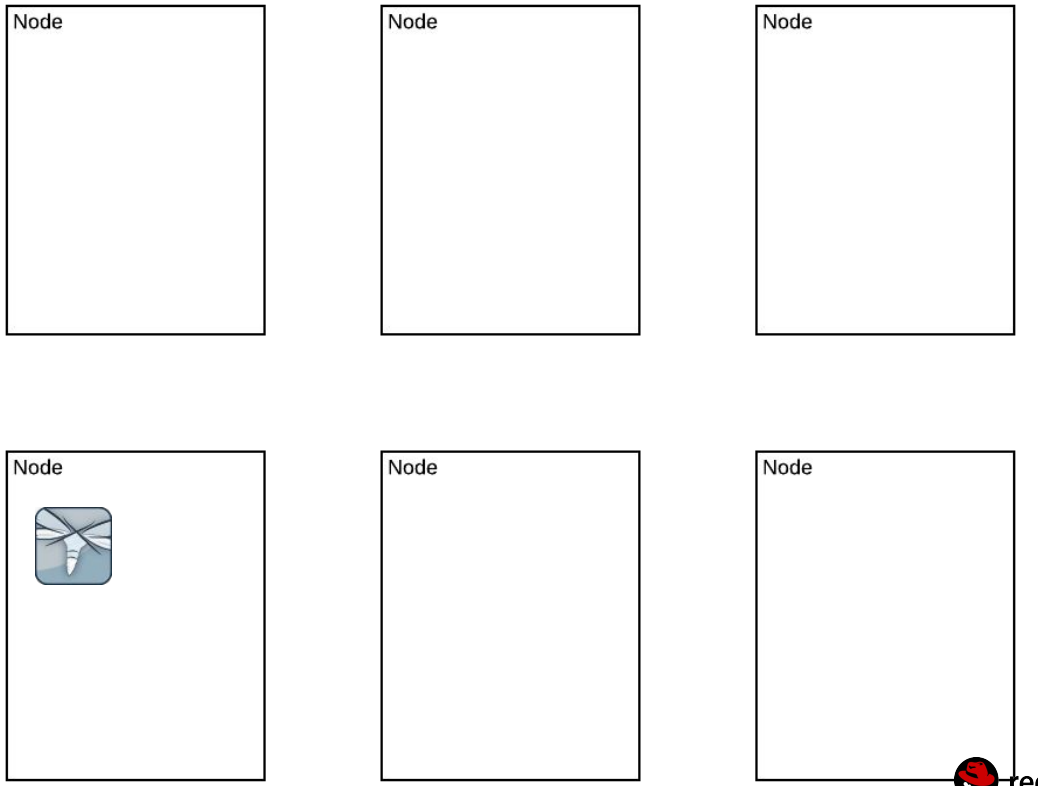
Workload execution guarantees



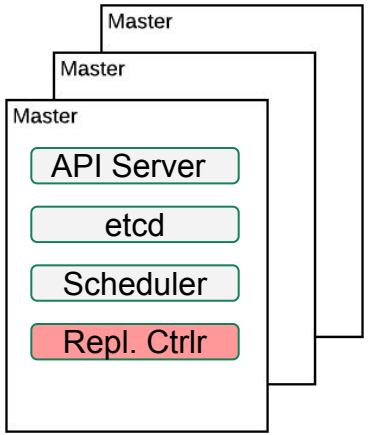


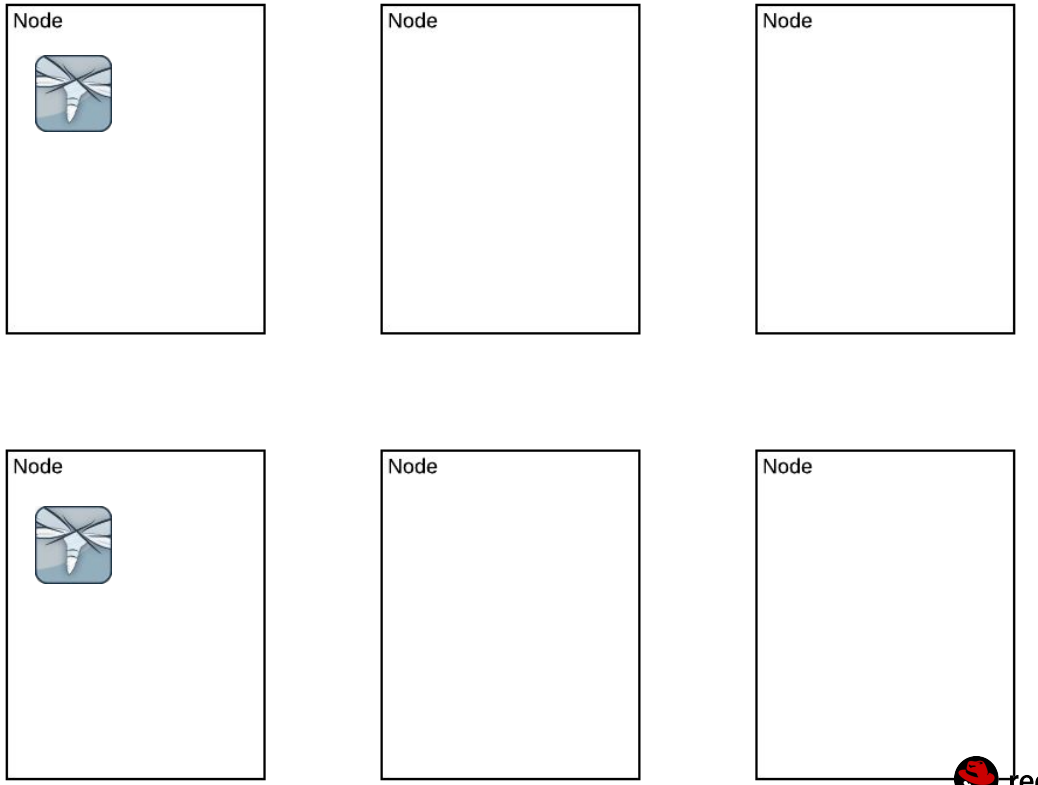
Workload execution guarantees



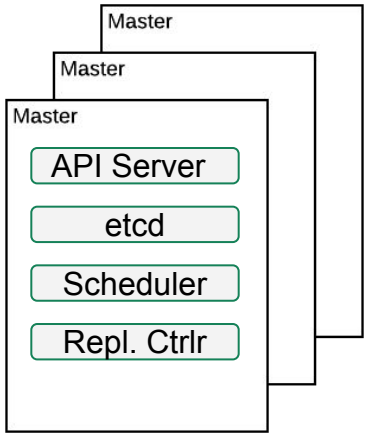


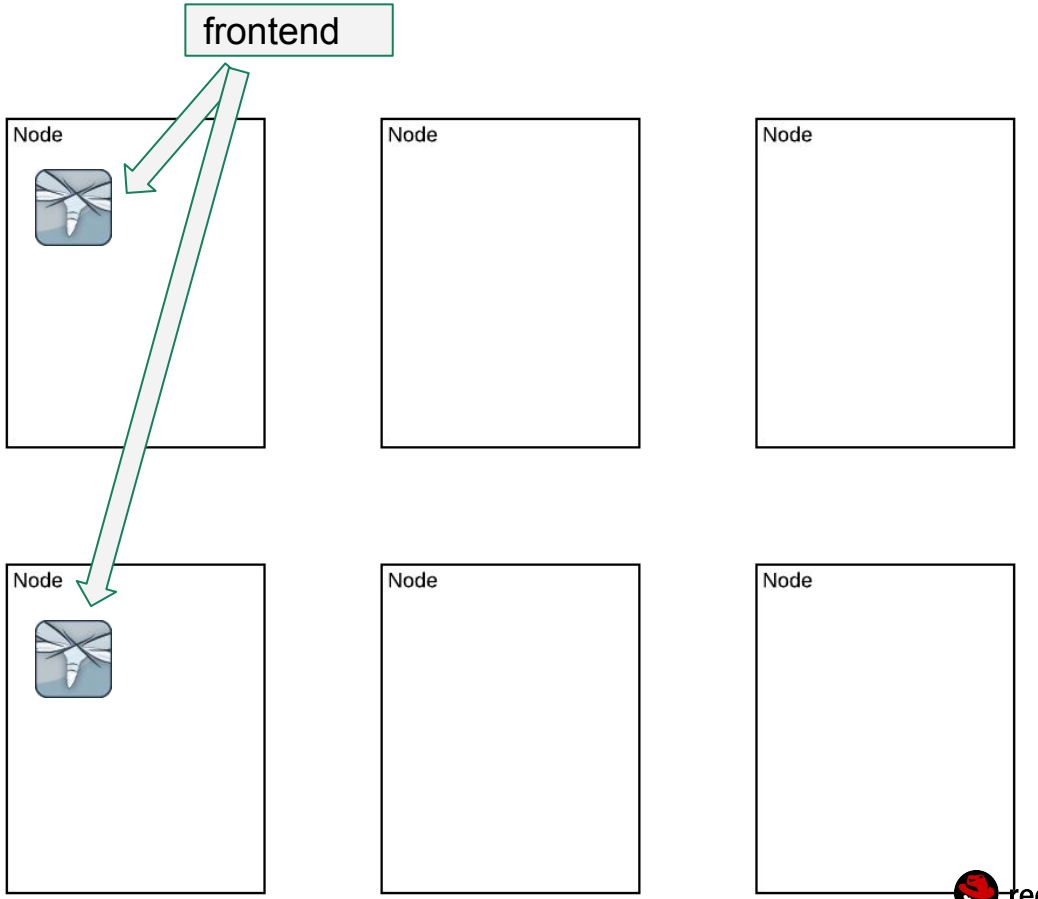
Scalability



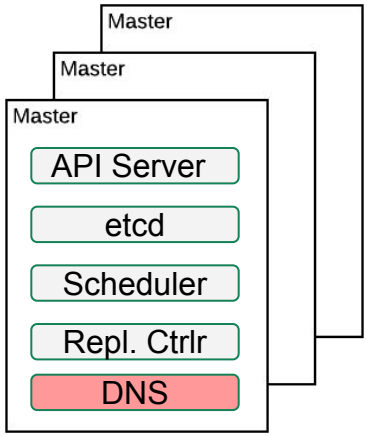


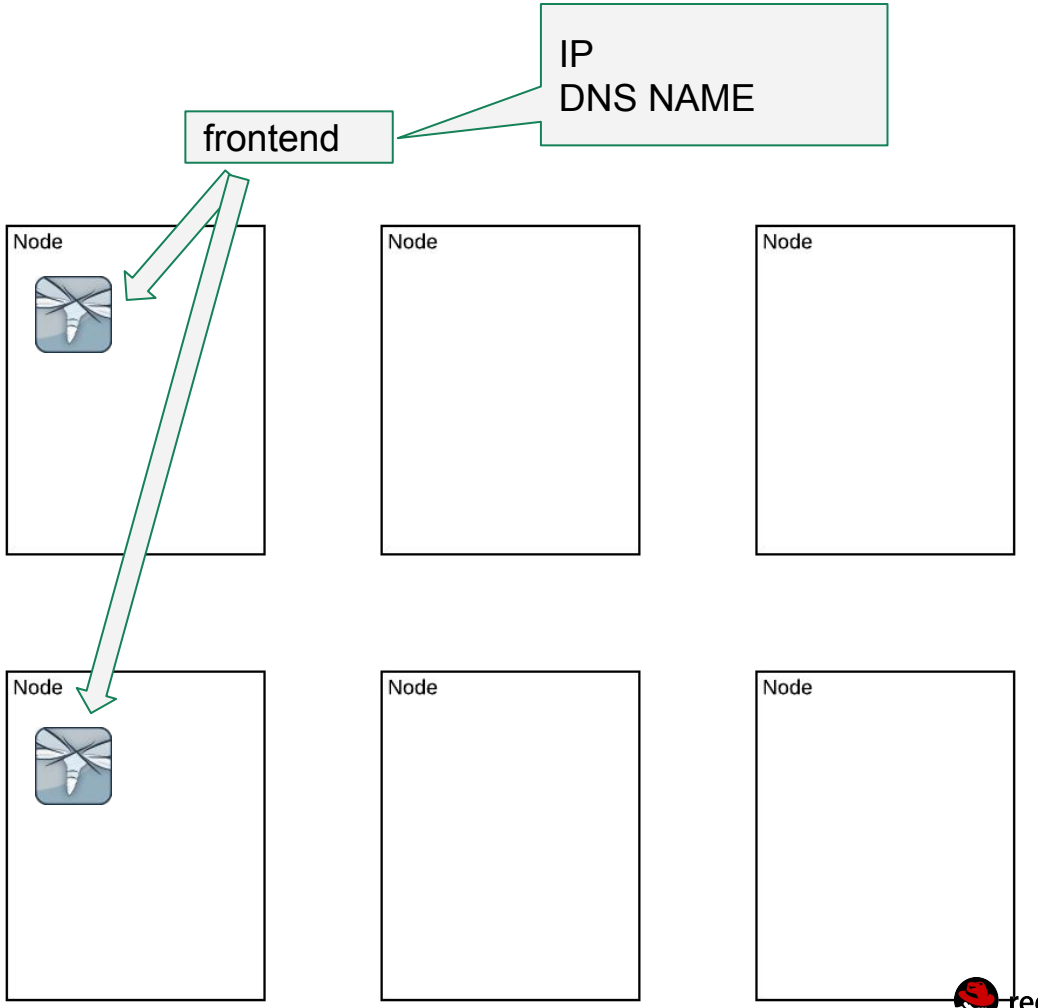
Service

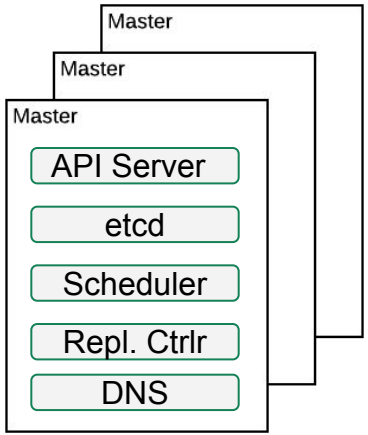




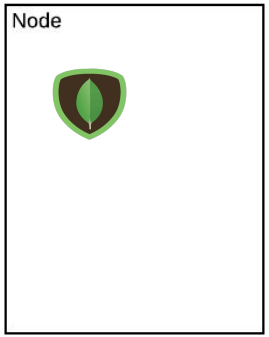
Service

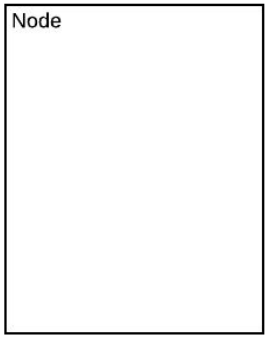




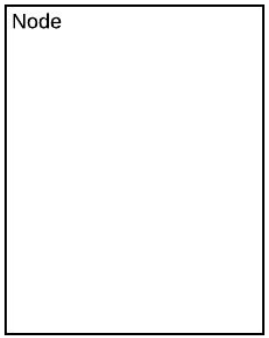

Master
Master
Master
API Server
etcd
Scheduler
Repl. Ctrlr
DNS

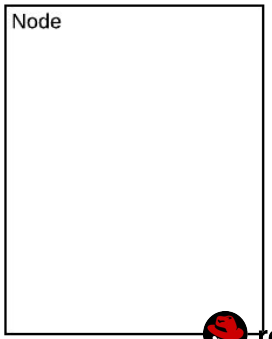

Node


Node

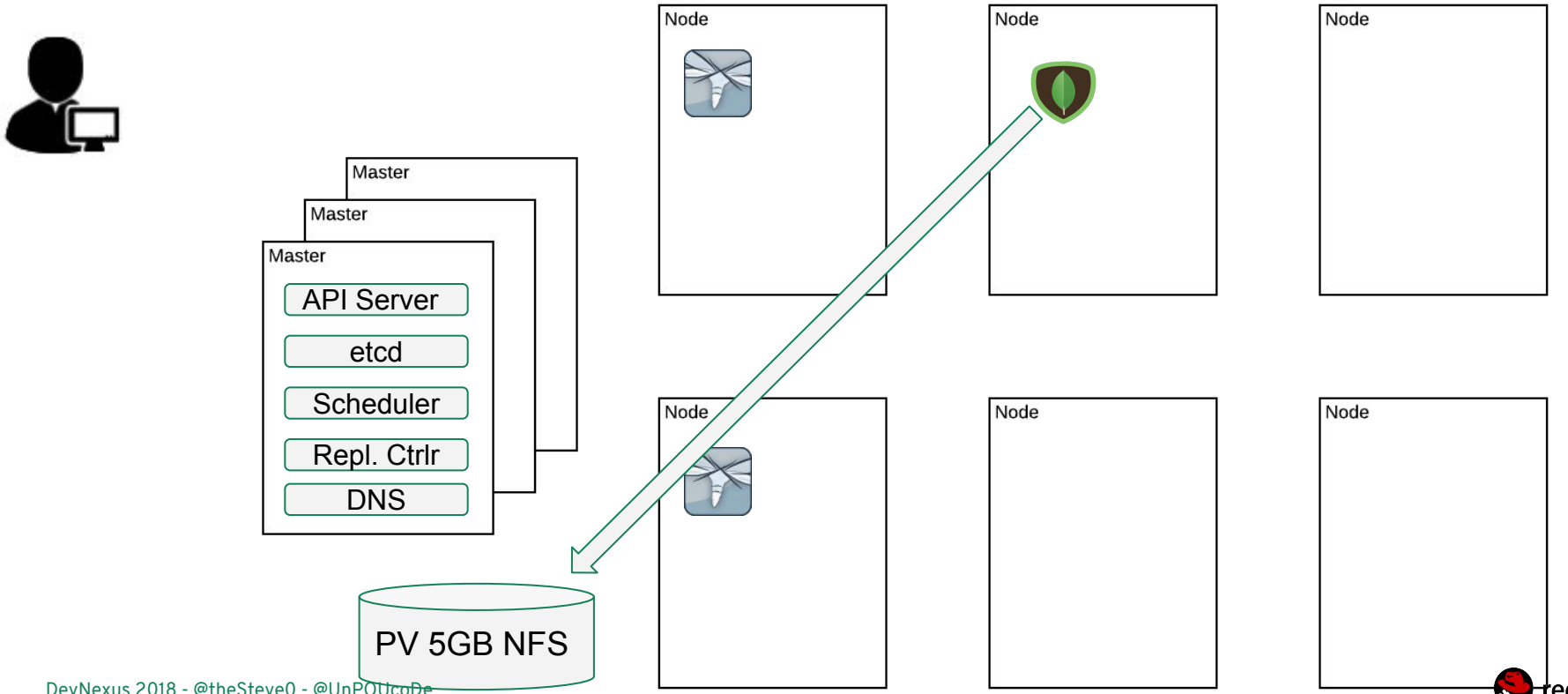

Node


Node

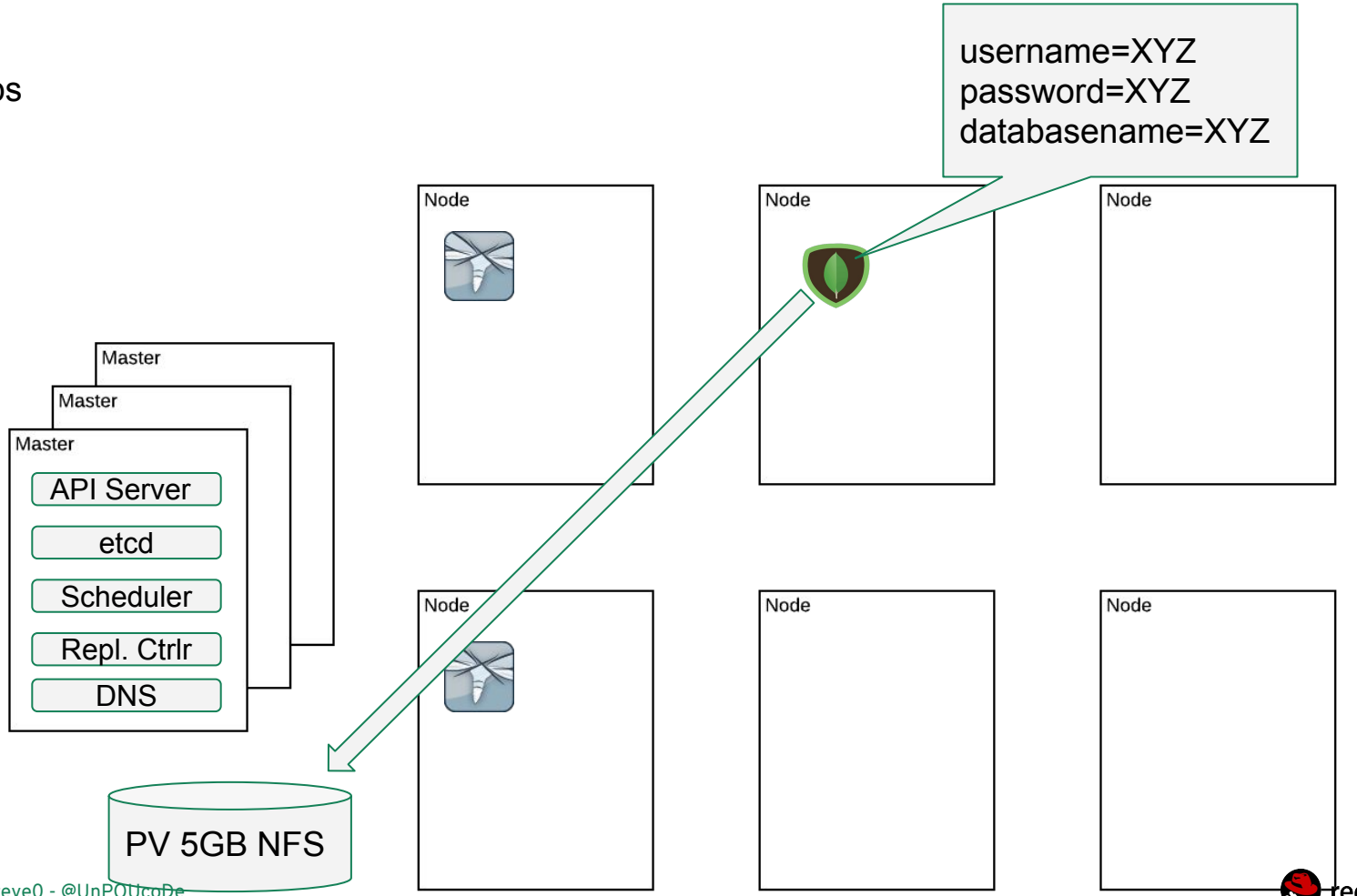

Node


Node

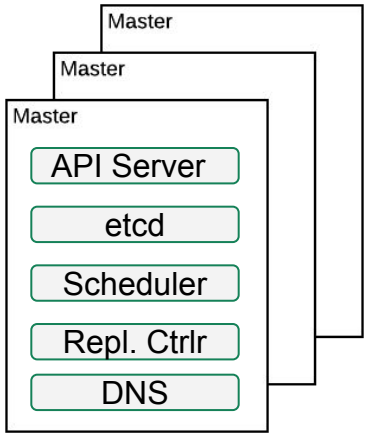
PersistentVolume and PersistentVolumeClaim

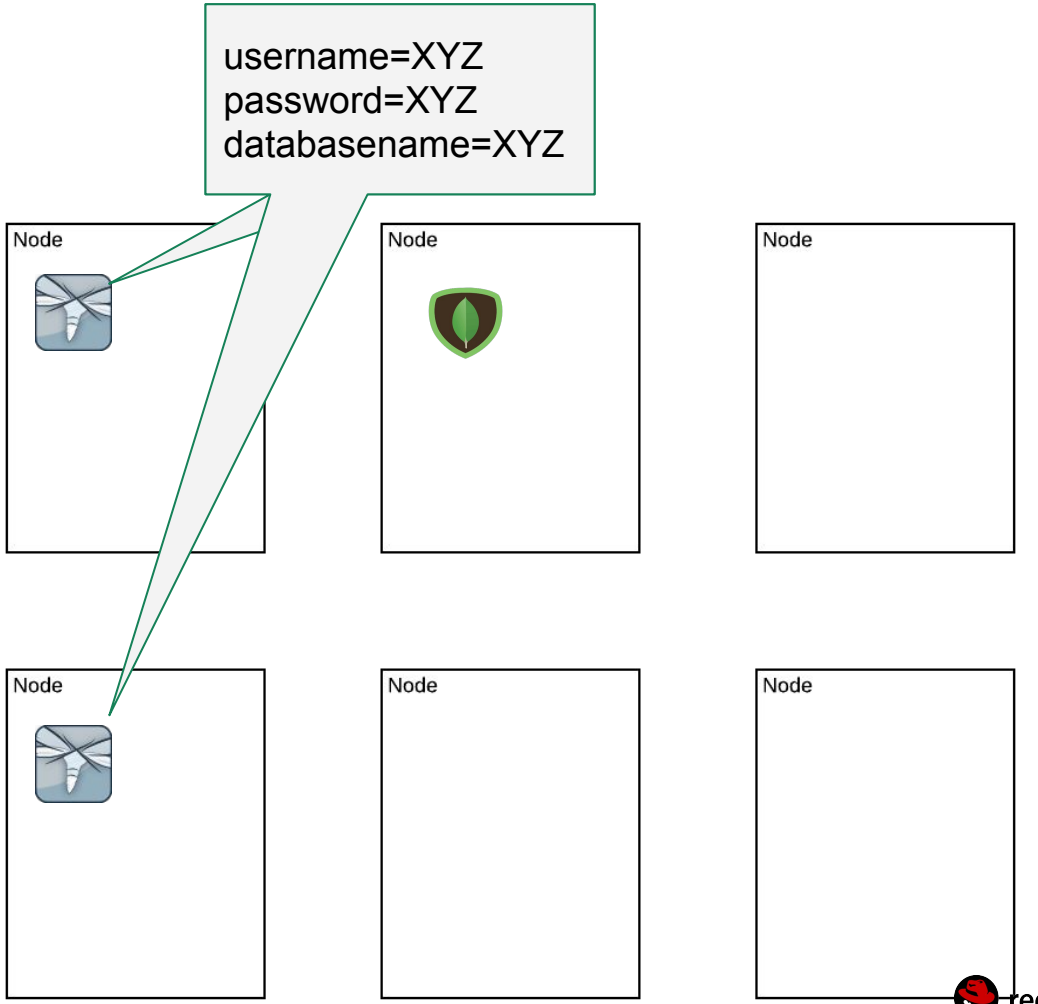


Configuring apps

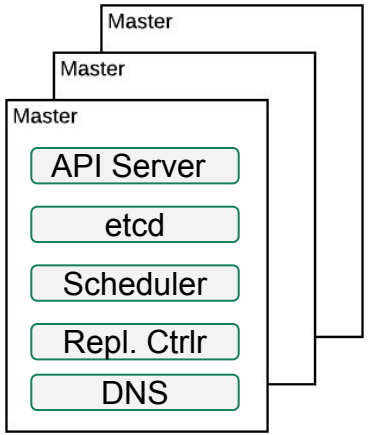


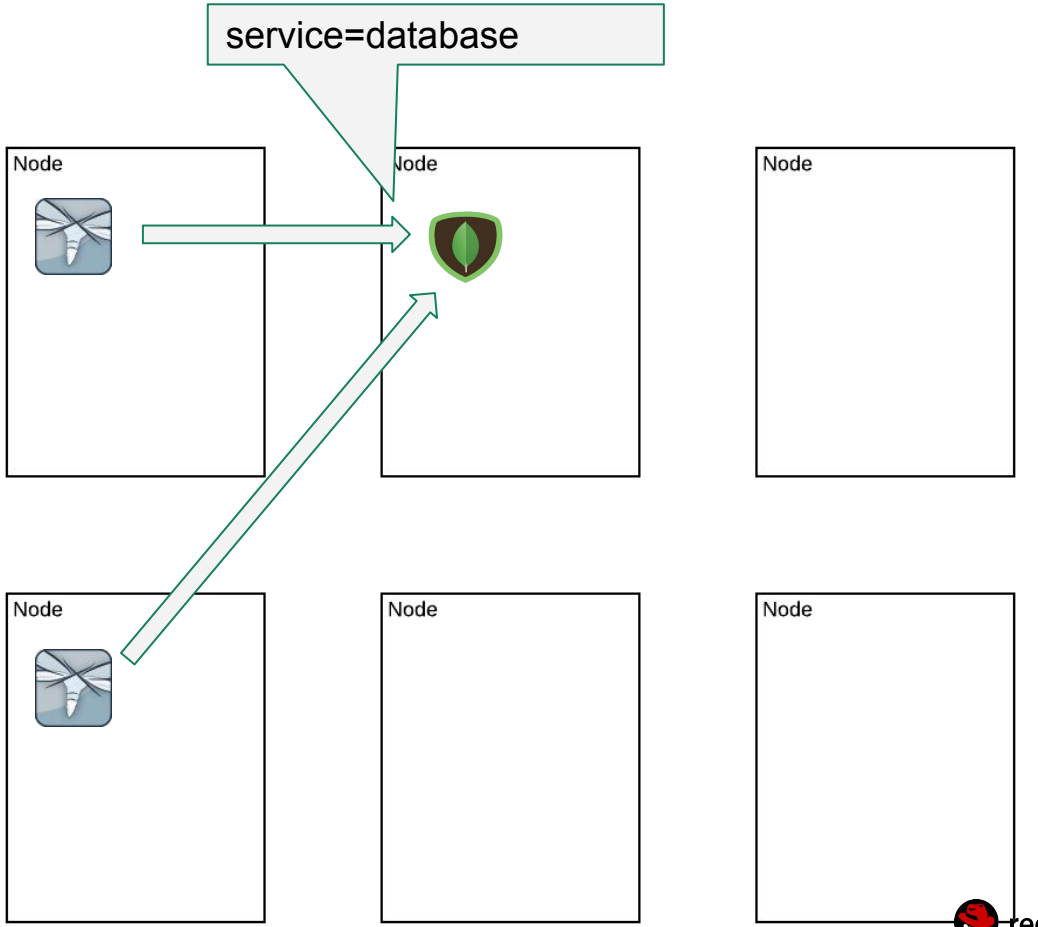
Linking services



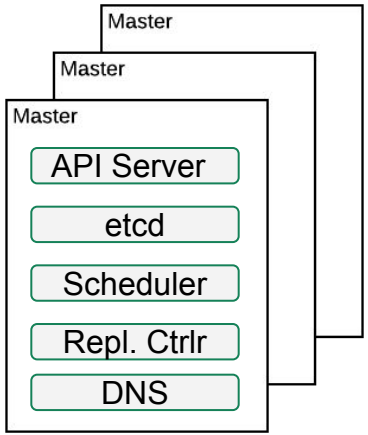


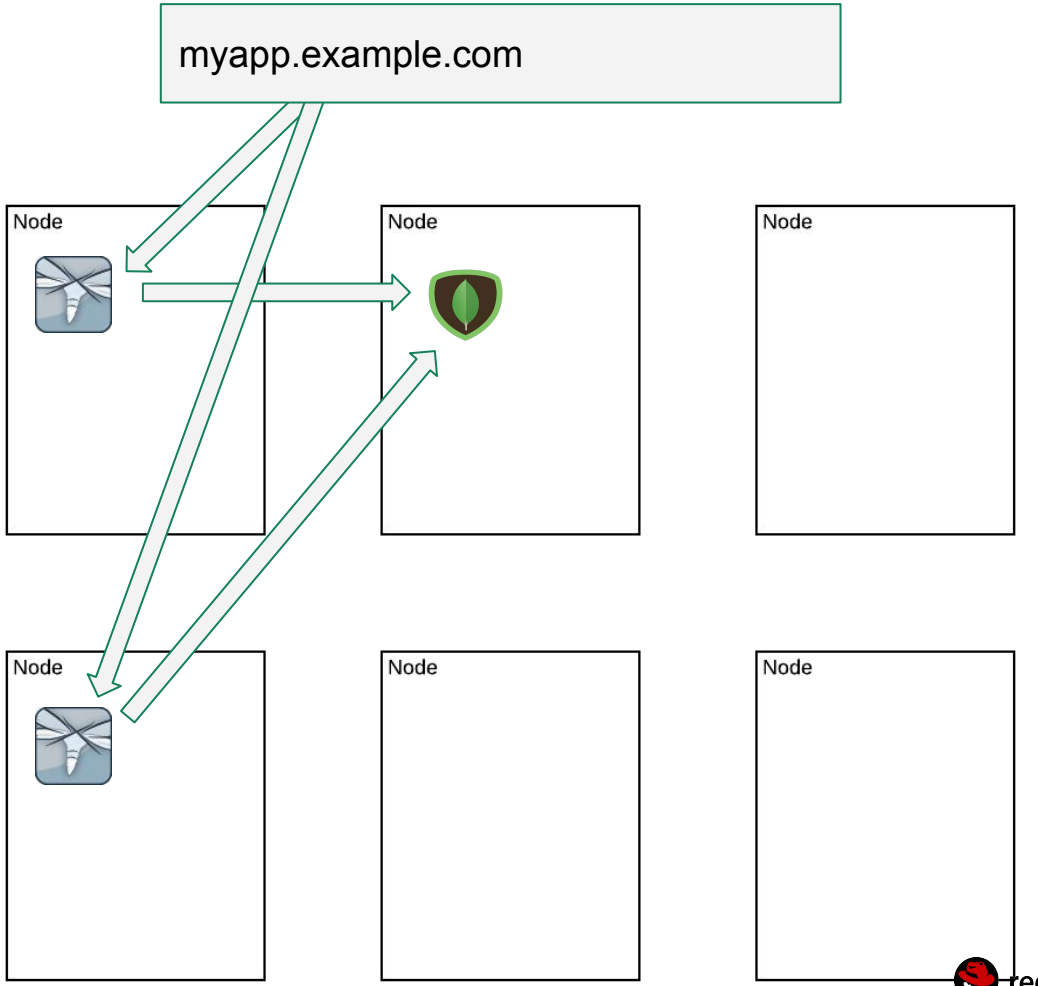
Linking services





Routing





S2I Walk Through

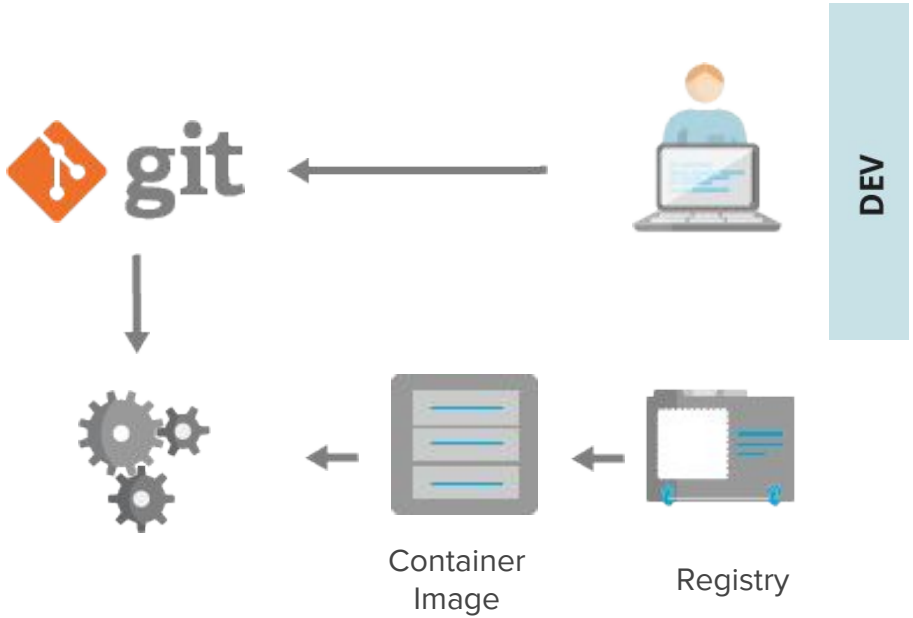
Code

Developers can leverage existing development tools and then access the OpenShift Web, CLI or IDE interfaces to create new application services and push source code via GIT. OpenShift can also accept binary deployments or be fully integrated with a customer's existing CI/CD environment.

S2I Walk Through

Build

OpenShift automates the Docker image build process with Source-to-Image (S2I). S2I combines source code with a corresponding Builder image from the integrated Docker registry. Builds can also be triggered manually or automatically by setting a Git webhook. Add in Build pipelines



S2I Walk Through

Deploy

OpenShift automates the deployment of application containers across multiple Node hosts via the Kubernetes scheduler. Users can automatically trigger deployments on application changes and do rollbacks, configure A/B deployments & other custom deployment types.

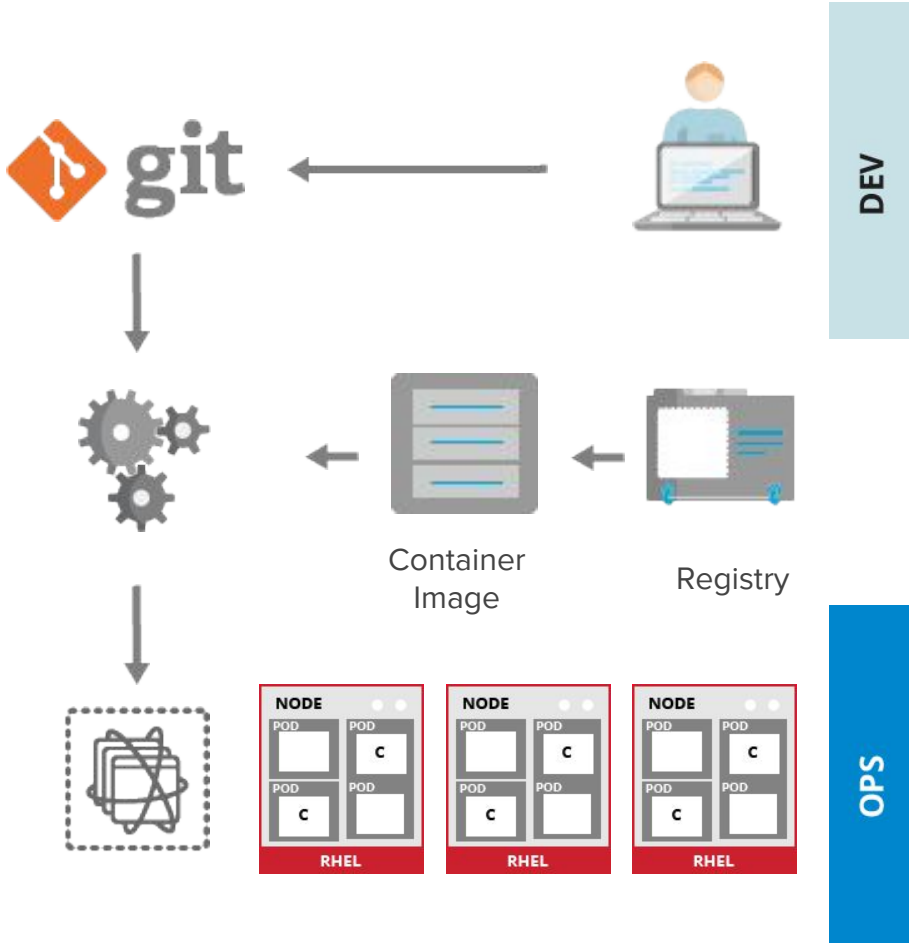
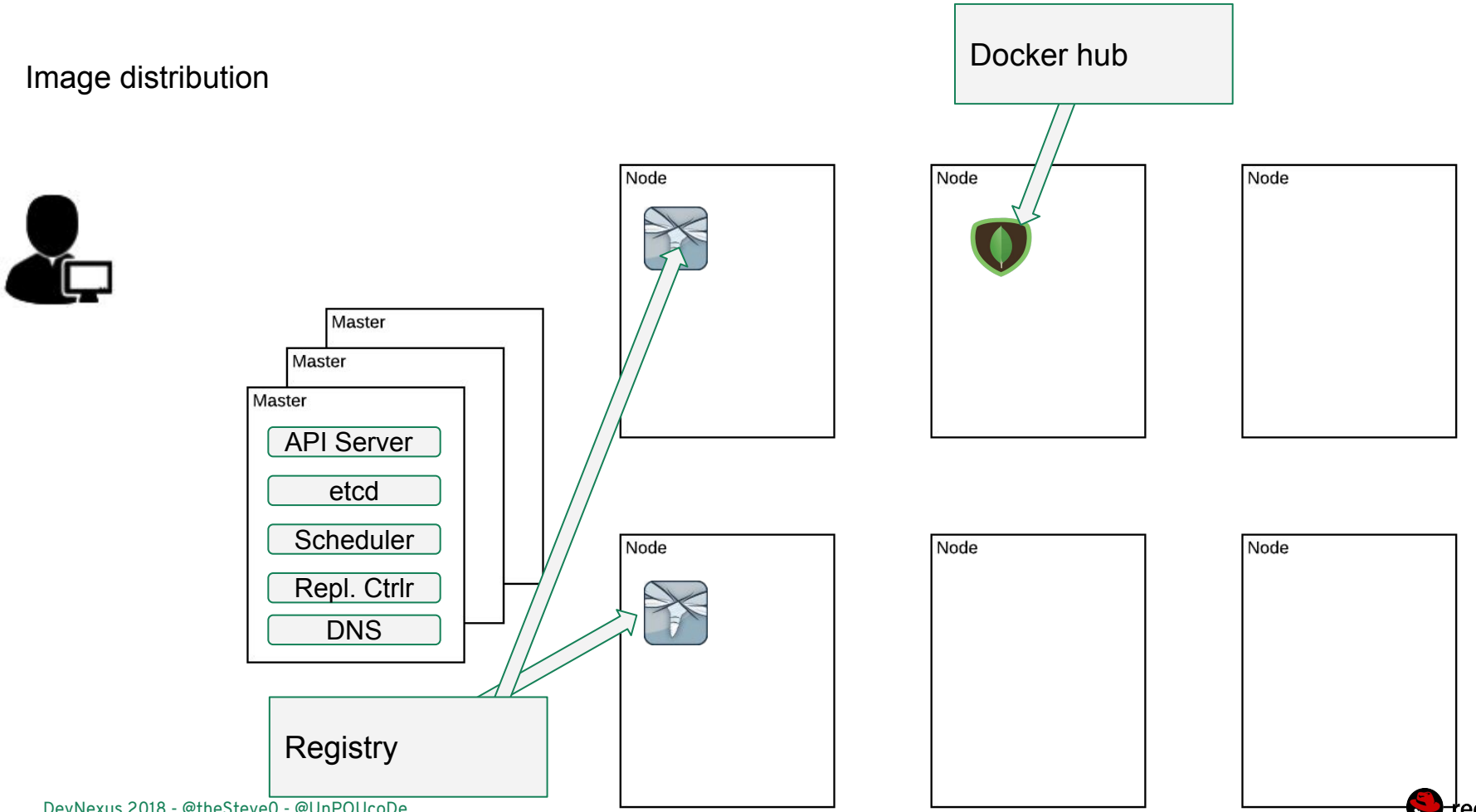


Image distribution



WORKSHOP

LAB GUIDE

Lab 1. Environment Overview

Lab 2. Architecture Overview of the ParksMap Application

Lab 3. Installing the *oc* client tool

Lab 4. Exploring the CLI and Web Console

Lab 5. Deploying our First Docker Image

Lab 6. Scaling

Lab 7. Creating Routes

Lab 8. Exploring OpenShift's Logging Capabilities

Lab 9. Role-Based Access Control

Lab 10. Remote Operations

LAB GUIDE

Lab 11. Deploying Java Code

Lab 12. Adding a Database (MongoDB)

Lab 13. Application Health

Lab 14. Using Source 2 Image for Code Changes

Lab 15. Using Application Templates

Lab 16. Binary Deploy

Lab 17. Remote Debugging

Lab 18. Clustering Stateful Java EE Applications

Lab 19. Further resources

How to Start?

Link on next slide ;-)

How it Goes?

You will do the labs at your own pace.

Instructors will have provided an introduction to OpenShift, and then we will regroup do the exercise together and consolidate concepts.

Having an Issue?

Raise your hand. An instructor will come to you.

Hands-on-Labs:

bit.ly/openshift-devnexus18

LABS EXPLAINED BY US

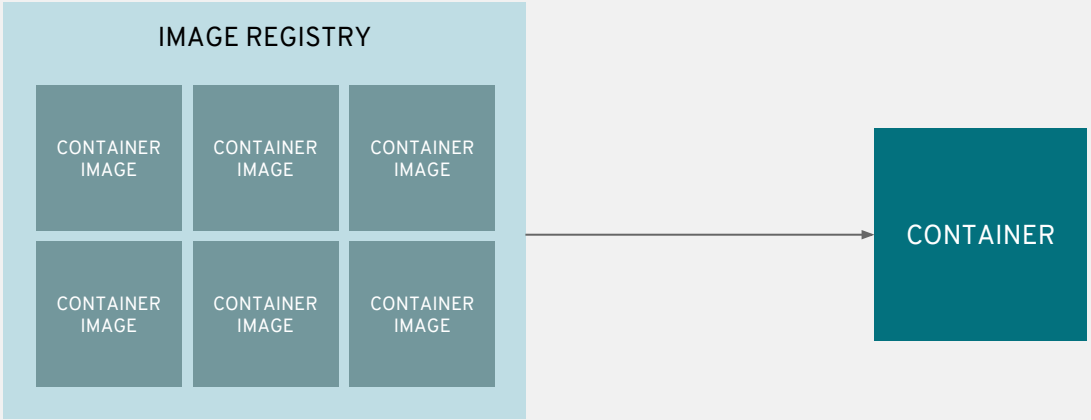
Image: https://linpack-for-tableau.com/news/linpack-for-tableau-demo/

OPENSHIFT CONCEPTS SUMMARY

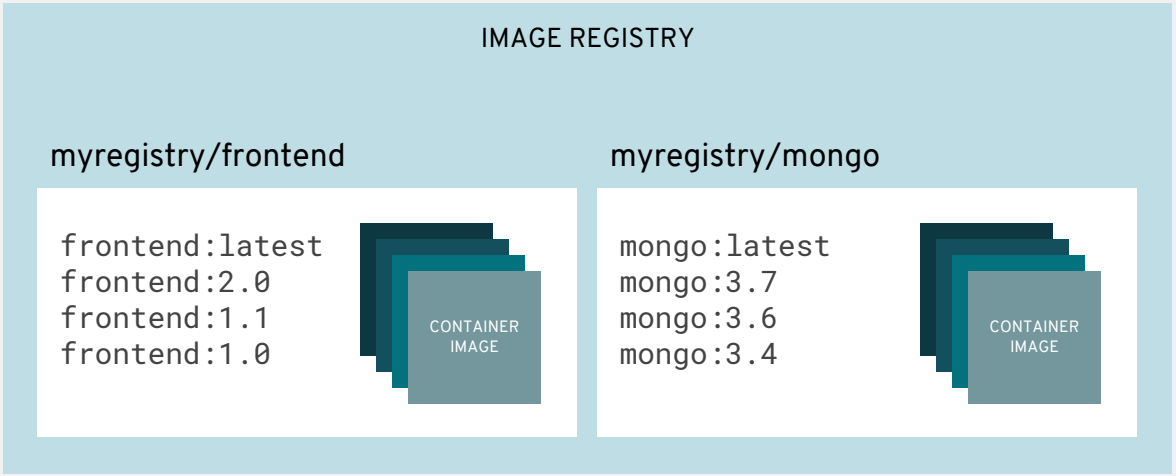
A container is the smallest compute unit

containers are created from container images

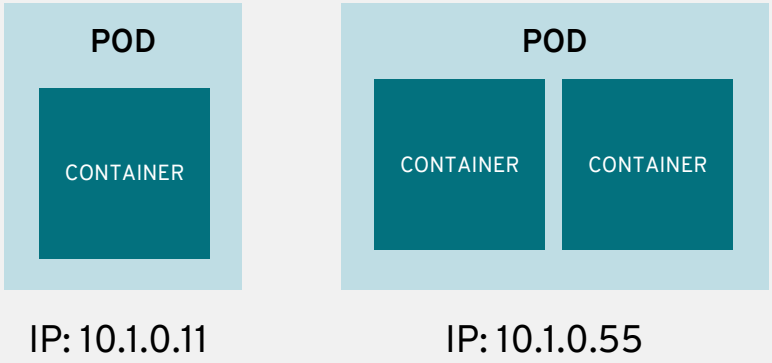
container images are stored in an image registry



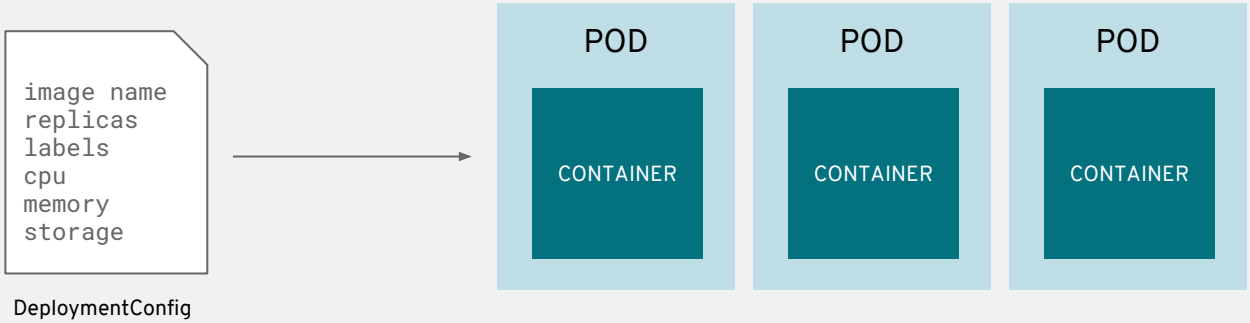
an image repository contains all versions of an image in the image registry



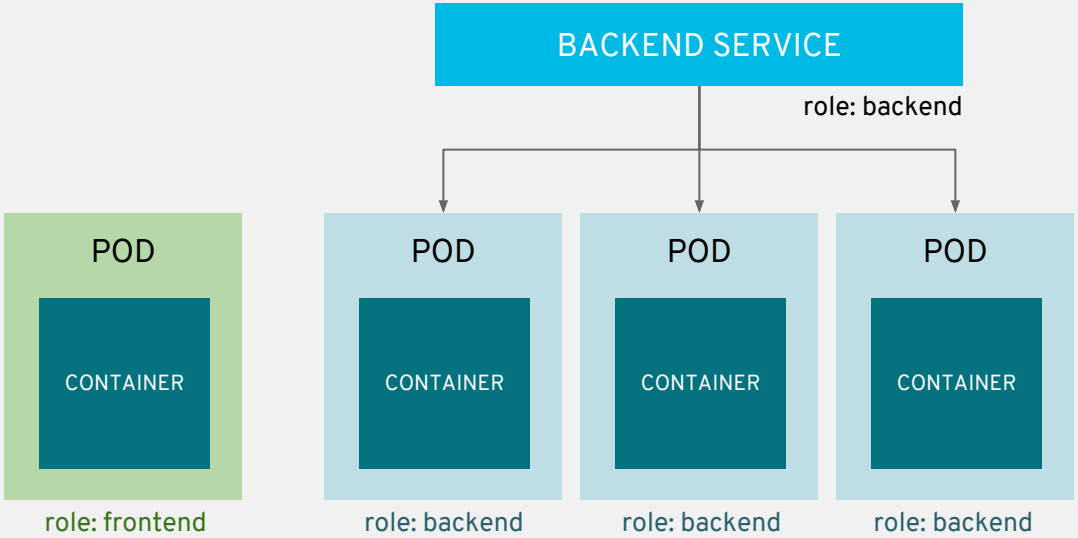
containers are wrapped in pods which are units of deployment and management



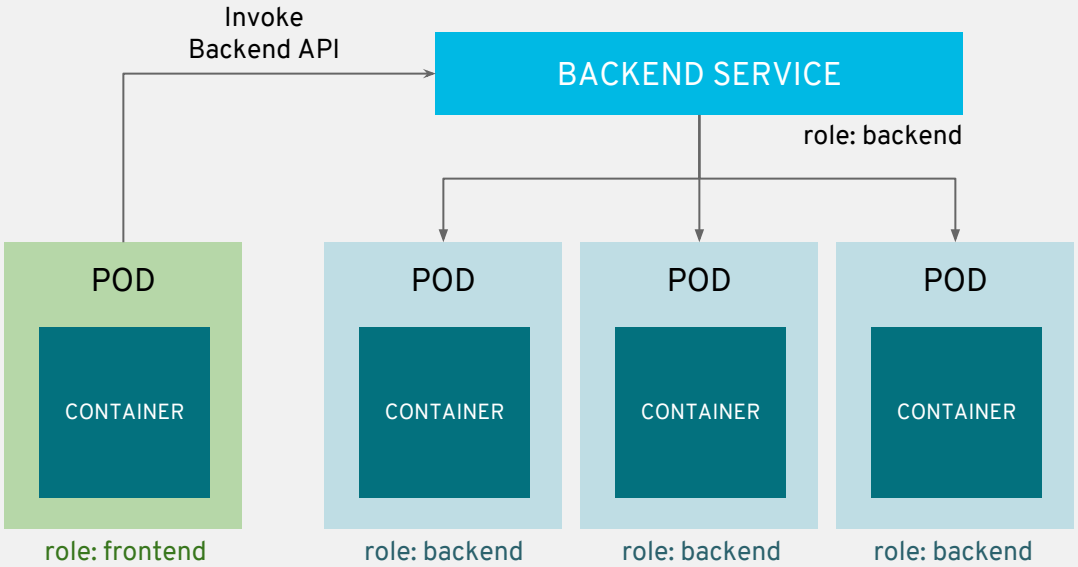
pods configuration is defined in a `DeploymentConfig`



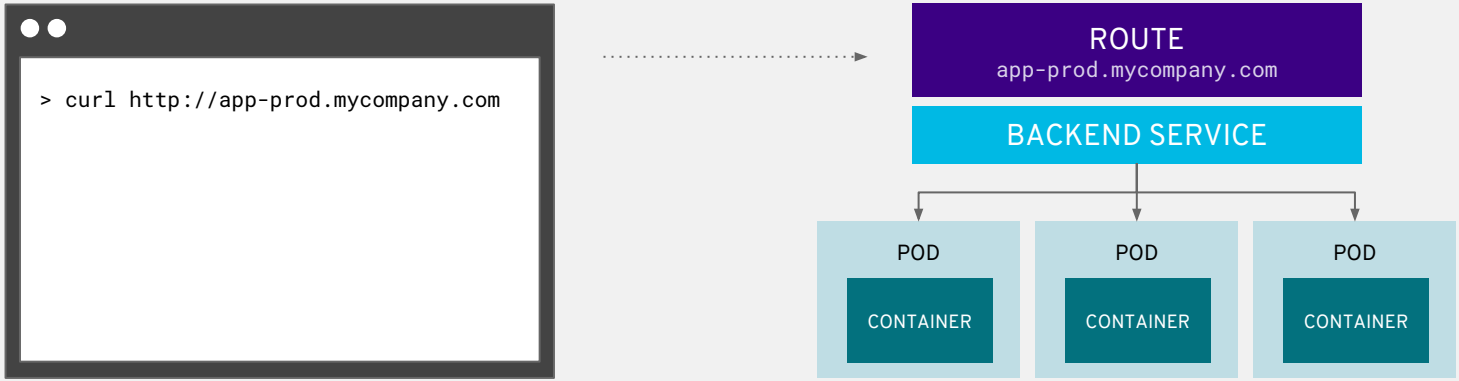
services provide internal load-balancing and service discovery across pods



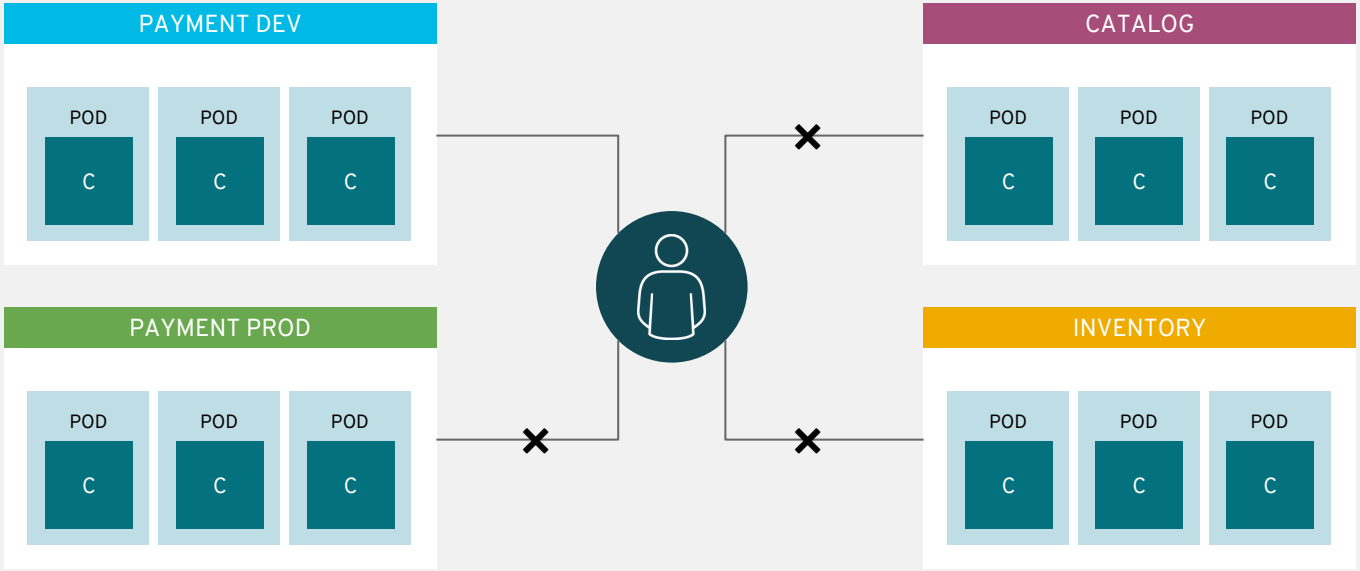
apps can talk to each other via services



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

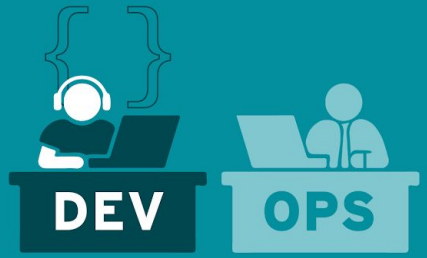


projects allow managing apps in isolation from other environments, teams, groups and departments



Take Homes

1. The new hotness is Dev and Ops getting along
2. You too can now use containers without being a container expert
3. So much to be gained by Automate all the things!!!!
4. But remember OpenShift is just a TOOL that helps - not the end all and be all
    a. Tech is easy - peopling is hard



THANK YOU

plus.google.com/+RedHat

facebook.com/redhatinc

linkedin.com/company/red-hat

twitter.com/RedHatNews

youtube.com/user/RedHatVideos

redhat.