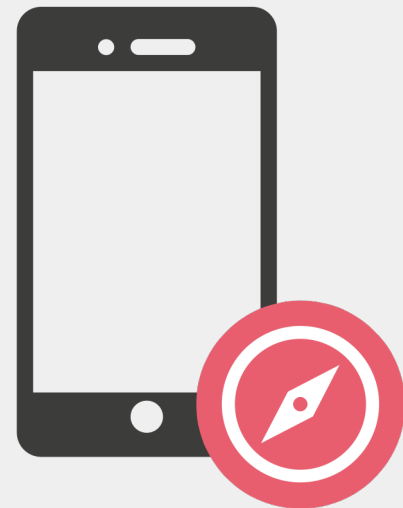


OpenShift Pt.1

COMPASS NUM-APP

21.04.2021



Objectives

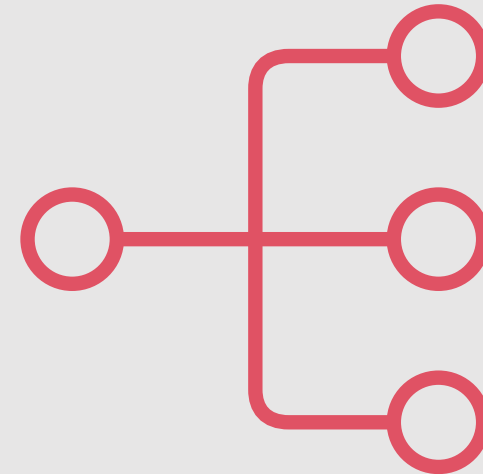
After this session, you should be able to:

- understand the concept of containers and related concepts
- work with containers
- explain the basics of Kubernetes & OpenShift



Agenda

- Introduction to Container Technology
- Container Architecture & Lifecycle & Networking
- Working with containers
- Introduction to Dockerfiles and working with images
- Introduction to Kubernetes & OpenShift



Necessary Preparation

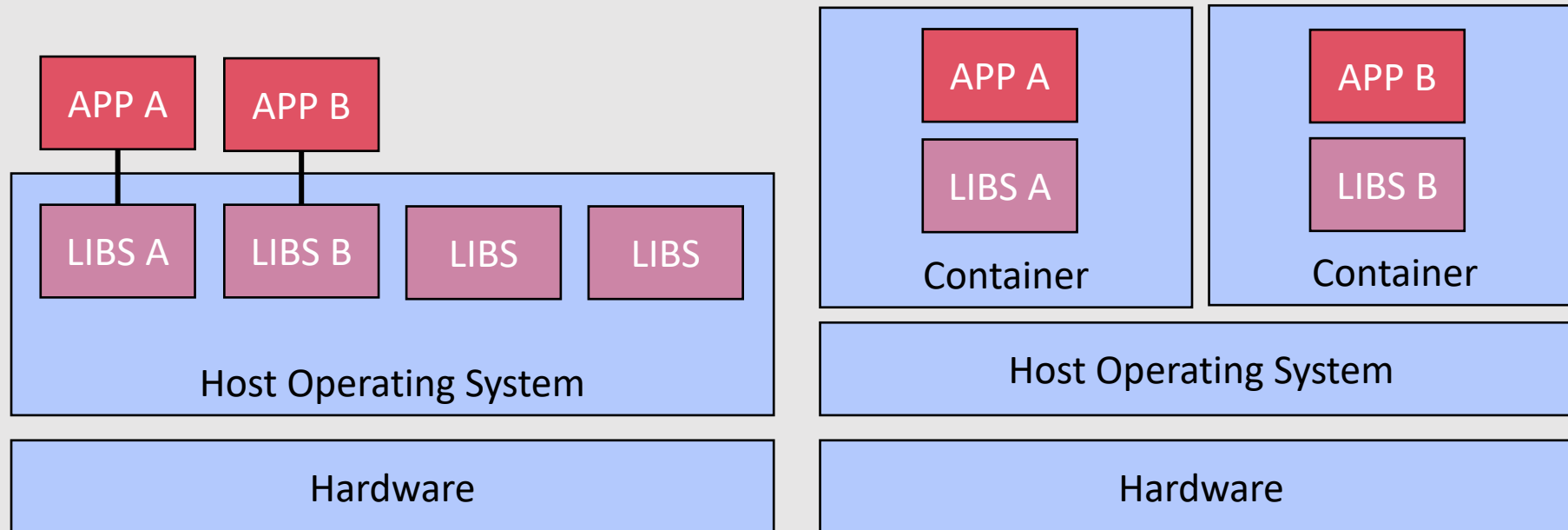
Installation of Docker (Container Engine)



<https://www.docker.com/>

Introduction to Container Technology

Traditional OS vs. Containers



A container is a set of one or more processes that are isolated from the rest of the system.

Advantages of Containers



- Efficiency
- Elasticity
- Reusability
- Application portability
- Low hardware footprint
- Environment isolation
- Quick Deployment
- Multiple environment deployment

Quiz Time



[This Photo](#) by Unknown Author is licensed under [CC BY](#)

Quiz Time

Choose 2

Which examples might run in a container?

- A. A Python app accessing services such as a MySQL database, an FTP server & a web server on a single physical host
- B. A Java EE app with an Oracle database & a message broker running on a single VM
- C. An I/O monitoring tool responsible for analyzing the traffic & block data transfer
- D. An app responsible for taking snapshots of all memory CPU caches for monitoring purposes

Quiz Time

Choose 2

Which examples might run in a container?

- A. A Python app accessing services such as a MySQL database, an FTP server & a web server on a single physical host**
- B. A Java EE app with an Oracle database & a message broker running on a single VM**
- C. An I/O monitoring tool responsible for analyzing the traffic & block data transfer
- D. An app responsible for taking snapshots of all memory CPU caches for monitoring purposes

Quiz Time

Choose 2

Which use cases are best suited for containers?

- A. A software provider needs to distribute software that can be reused by other companies in a fast and error-free way.
- B. A company is deploying applications on a physical host and would like to improve its performance by using containers.
- C. Developers at a company need a disposable environment that mimics the production environment so that they can quickly test the code they develop.
- D. A financial company is implementing a CPU-intensive risk analysis tool on their own containers to minimize the number of processors needed.

Quiz Time

Choose 2

Which use cases are best suited for containers?

- A. A software provider needs to distribute software that can be reused by other companies in a fast and error-free way.**
- B. A company is deploying applications on a physical host and would like to improve its performance by using containers.
- C. Developers at a company need a disposable environment that mimics the production environment so that they can quickly test the code they develop.**
- D. A financial company is implementing a CPU-intensive risk analysis tool on their own containers to minimize the number of processors needed.

Quiz Time

Choose 2

A company runs their PHP & Python apps on the same host. Both apps share a set of custom libraries from the OS. The latest update that was applied to them because of a request from the Python dev team broke the PHP app. Which architectures would provide a better for the apps?

- A. Deploy each app to different VMs & apply the custom shared libraries individually to each VM host.
- B. Deploy each app to different containers & apply the custom shared libraries individually to each container.
- C. Deploy each app to different VMs & apply the custom shared libraries to all VM hosts.
- D. Deploy each app to different containers & apply the custom shared libraries to all containers.

Quiz Time

Choose 2

A company runs their PHP & Python apps on the same host. Both apps share a set of custom libraries from the OS. The latest update that was applied to them because of a request from the Python dev team broke the PHP app. Which architectures would provide a better for the apps?

- A. Deploy each app to different VMs & apply the custom shared libraries individually to each VM host.**
- B. Deploy each app to different containers & apply the custom shared libraries individually to each container.**
- C. Deploy each app to different VMs & apply the custom shared libraries to all VM hosts.
- D. Deploy each app to different containers & apply the custom shared libraries to all containers.

Quiz Time

Choose 3

Which app types can be packaged as containers for immediate consumption?

- A. A VM hypervisor
- B. A frontend application
- C. A database
- D. A local file system recovery tool
- E. A web server

Quiz Time

Choose 3

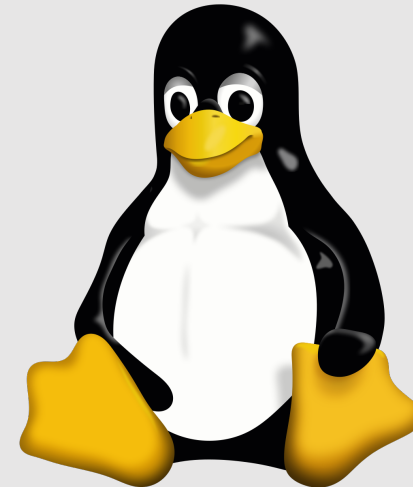
Which app types can be packaged as containers for immediate consumption?

- A. A VM hypervisor
- B. A frontend application**
- C. A database**
- D. A local file system recovery tool
- E. A web server**

Container Architecture & Lifecycle & Networking

Architecture – Linux Features

- Namespaces
- Control Groups
- Seccomp
- SELinux (Security-Enhanced Linux)



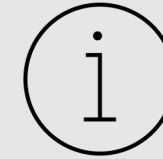
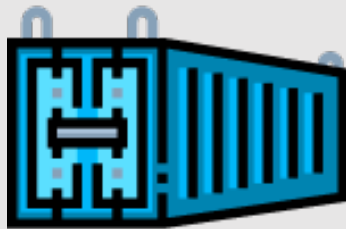
Goal

Enabling processes to run isolated while still accessing system resources

Architecture - Container



What is a **Container**?



A set of one or more processes that are isolated from the rest of the system.

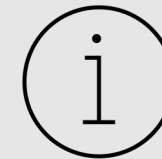
Architecture - Image

~~Container runs a single binary file~~

Container runs an image



What is a **Container**?

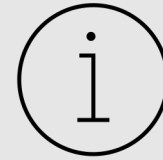


An immutable file-system bundle
containing all required dependencies.

Architecture – Image Registry



What is an **Image Registry**?



A public or private service where images can be stored, searched and retrieved.

Examples:

- Red Hat Container Catalog (<https://registry.redhat.io>)
- Docker Hub (<https://hub.docker.com>)
- Red Hat Quay (<https://quay.io/>)

Container images

Container images offer lightweight and self-contained software to enable deployment at scale.

Home > Software > Container images

postgresql

Search

1 - 15 of 82

Provider

Search

☐ Crunchy Data

☐ EnterpriseDB

☐ Fujitsu

☐ IBM

☐ OwIDQ

See more


Category

Search

☐ Application Delivery

☐ Backup & Recovery

☐ Business Intelligence

 Red Hat


rhscL/postgresql-12-rhel7

PostgreSQL 12 SQL database server

by Red Hat, Inc.

PostgreSQL is an advanced Object-Relational database management system

Updated a month ago

 Red Hat


rhscL/postgresql-10-rhel7

PostgreSQL 10 SQL database server

by Red Hat, Inc.

PostgreSQL is an advanced Object-Relational database management system

Updated 15 days ago

 UBIX


ubix/postgres

postgres

by UBI

PostgreSQL is an advanced Object-Relational database management system

Updated 15 days ago

 Red Hat


rhscL/postgresql-13

PostgreSQL 13 SQL database server

by Red Hat, Inc.

PostgreSQL is an advanced Object-Relational database management system

Updated 15 days ago

 Red Hat


rhel8/postgresql-96

PostgreSQL 9.6 SQL database server

by Red Hat, Inc.

PostgreSQL is an advanced Object-Relational database management system

Updated 15 days ago

 Red Hat

rhel8/postgresql-12

PostgreSQL 12 SQL database server

by Red Hat, Inc.

PostgreSQL is an advanced Object-Relational database management system

Updated 15 days ago

https://catalog.redhat.com/software/containers/rhscL/postgresql-10-rhel7/5aa63541ac3db95f196086f1

Red Hat Ecosystem Catalog

HardwareSoftwareCloud & service providers

Resources

Help


All Red Hat

Home > Software > Container images > PostgreSQL 12 SQL database server

Standalone Image

PostgreSQL 12 SQL database server

rhscL/postgresql-12-rhel7

Provided by  Red Hat

Architecture amd64 Tag latest

latest

1

1-31

Overview

Security

Packages

Dockerfile

Get this image

Description

Usage

This container image provides a containerized packaging of the PostgreSQL postgres daemon and client application. The postgres server daemon accepts connections from clients and provides access to content from PostgreSQL databases on behalf of the clients. You can find more information on the PostgreSQL project from the project Web site (<https://www.postgresql.org/>).

Usage

For this, we will assume that you are using the rhscL/postgresql-12-rhel7 image, available via postgresql:10 imagstream tag in Openshift. If you want to set only the mandatory environment variables and not store the database in a host directory, execute the following command:

```
$ podman run -d --name postgresql_database -e POSTGRES_USER=user -e POSTGRES_PASSWORD=pass -e POSTGRES_DATABASE=db -p 5432:5432 rhscL/postgresql-12-rhel7
```

This will create a container named postgresql_database running PostgreSQL with database db and user with credentials user:pass.

Note: user postgres is reserved for internal usage

Published

a month ago

Release category

Generally Available

Health index

A

Size

122.2 MB

(336.0 MB uncompressed)

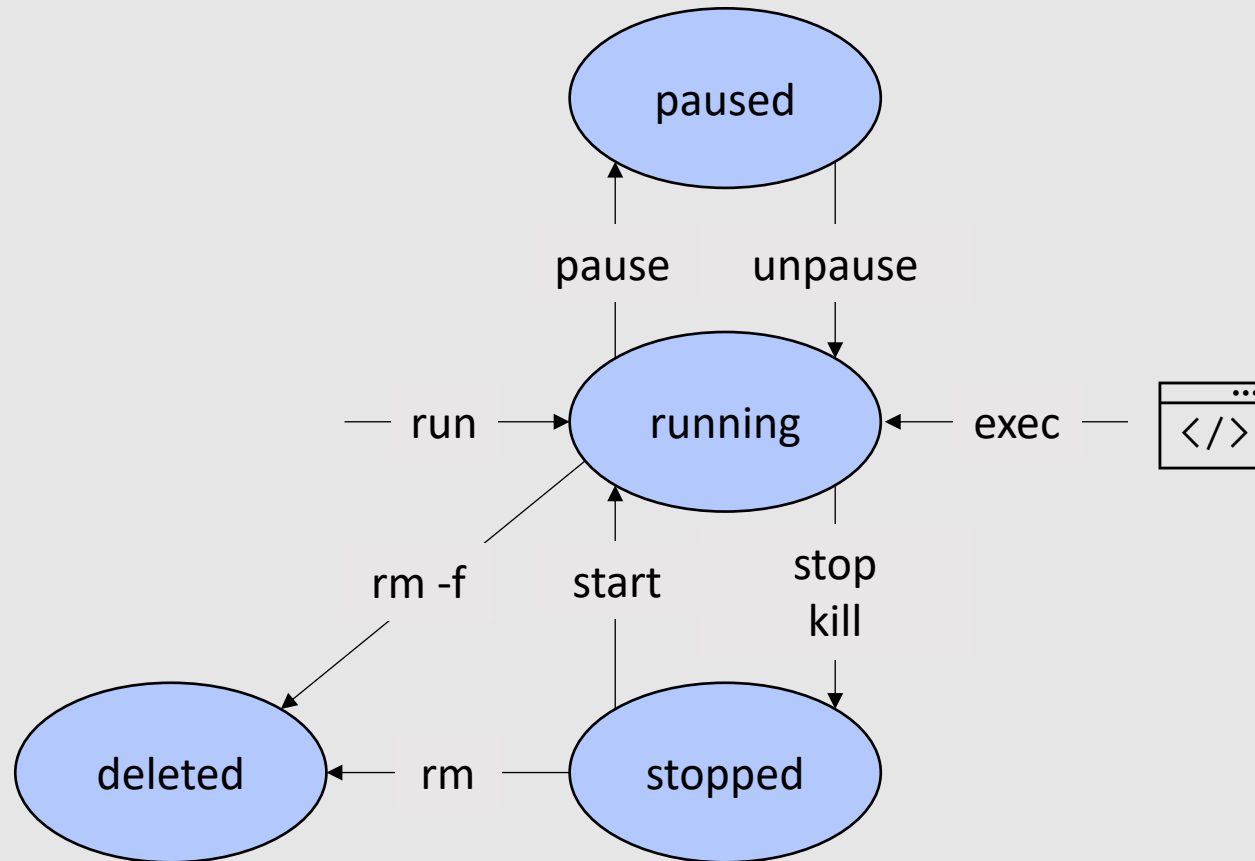
Digest

b6116381...

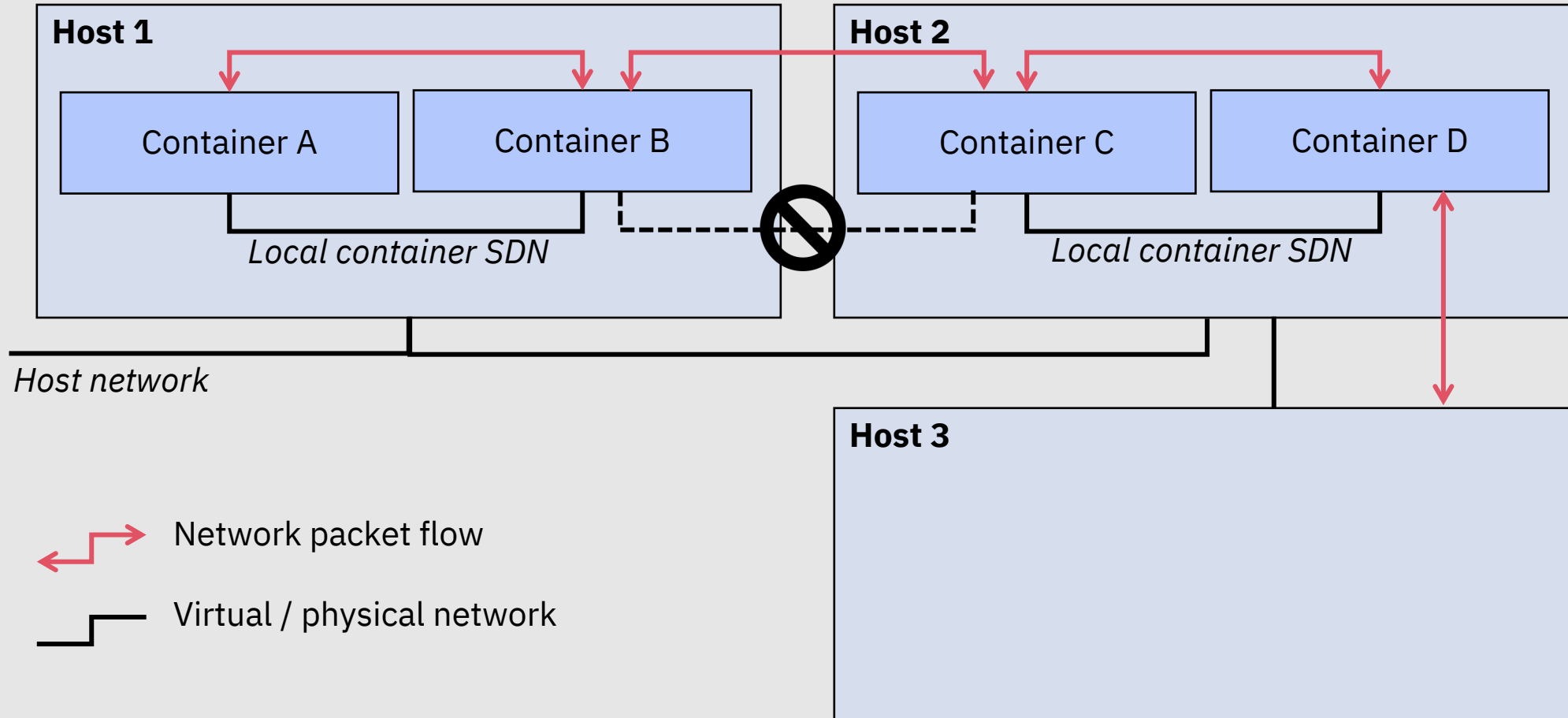
Have feedback?

https://registry.redhat.io

Container Lifecycle



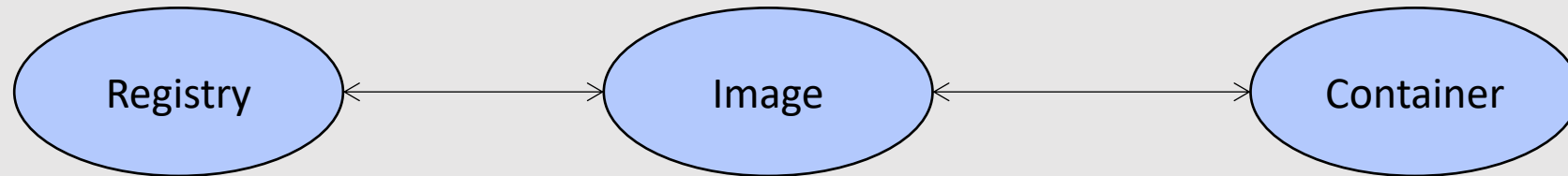
Networking



Working with containers

Managing Containers

Containers, images & image registries require interaction



Docker



- docker **pull**: retrieve an image from a registry
- docker **run**: create a container from an image
- docker **ps**: list running containers
- docker **ps -a**: list running and exited containers
- docker **exec**: access a running container
- docker **stop**: stop a running container gracefully
- docker **kill**: immediately stop running container
- docker **images**: list all local images
- docker **rm**: delete stopped container
- docker **rmi**: delete local image
- docker **inspect**: obtain details about the container

Things to Keep in Mind

1. A container runs in isolation
2. Container storage is said to be *ephemeral*
3. Images are immutable

Demo



Time for a break

Exercise



Start simple container

1. *docker pull busybox*
2. *docker images*
3. *docker run busybox*
4. *docker run busybox echo "hello from busybox"*

Start container in background

1. *docker run -P -d nginxdemos/hello*
2. *docker ps*
3. Open localhost:5500x
4. *docker stop container_name*

Introduction to Dockerfiles and working with images

Dockerfile

“A Dockerfile is a text document that contains all the commands a user could call on the command line to assemble an image.”

Dockerfile – Example

```
1 FROM python:3.7
2
3 COPY / .
4
5 RUN apt-get update && apt-get install -y \
6     build-essential \
7     python3-dev \
8     swig \
9     && pip install -r requirements.txt
10
11 CMD [ "python", "./downloader.py" ]
```

- Define base image
- Copy files into container
- Execute command inside container (install dependencies)
- Define command to be executed when image is run

Demo



Time for a break

Introduction to Kubernetes & OpenShift

Objectives

After this session, you should be able to answer these questions:

- What are the limitations of Linux containers and why do we need orchestration?
- What is Kubernetes?
- What is Red Hat OpenShift Container Platform?

Limitations of Containers

Production environment requirements:

- Communication between large number of containers
- Resource limits
- Need to increase/decrease number of running containers
- Quick response to service degradation
- Roll out of new service releases

Kubernetes - Overview

“Kubernetes is a portable, extensible, open-source platform for managing containerized workloads and services, that facilitates both declarative configuration and automation.”

Kubernetes - Features

Kubernetes provides:

- Service discovery and load balancing
- Horizontal scaling
- Health-checks and self-healing
- Automated rollout
- Secrets and configuration management

Kubernetes - Cluster

Kubernetes cluster

Control plane

kube-apiserver

kube-scheduler

kube-controller-manager



etcd

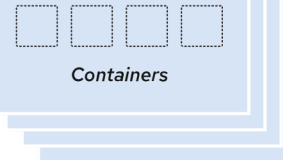
Compute machines

kubelet

kube-proxy

Container runtime

Pod

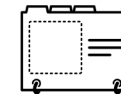


Containers

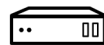
Persistent storage



Container registry



Underlying infrastructure



Physical



Virtual



Private



Public



Hybrid

OpenShift - Overview

What is Red Hat OpenShift Container Platform?

- Built on top of Kubernetes
- Set of modular components and services
- Provides production relevant capabilities for
 - Monitoring and Auditing
 - Security
 - Multitenancy
 - Application life-cycle management
 - Many more...

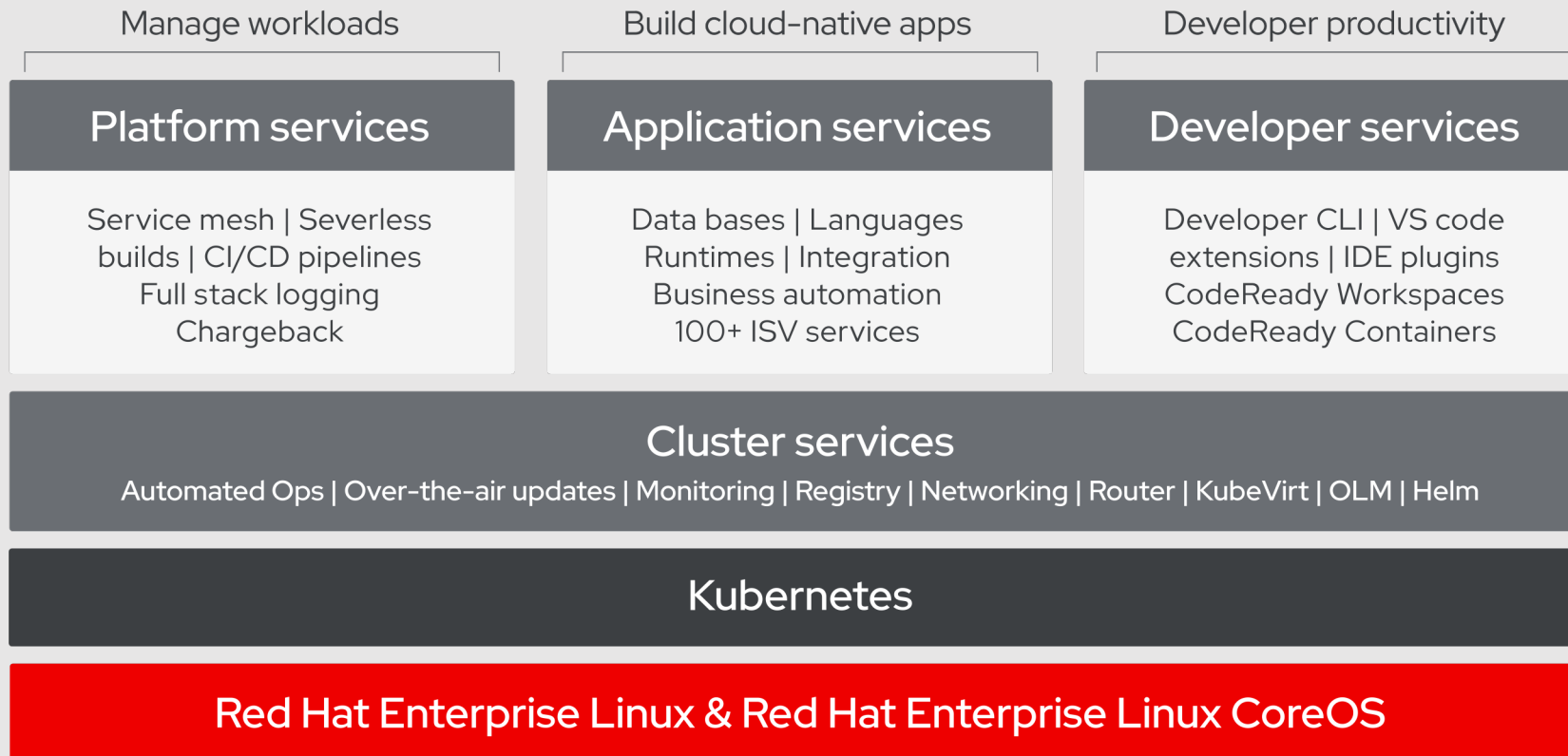
OpenShift - Features

Features provided by OpenShift:

- Integrated developer workflow
- Routes
- Metrics and logging
- Unified UI

OpenShift – Platform

Red Hat OpenShift



Relevant OpenShift Resources

- Pod
 - A Pod is a group of one or more containers deployed to a single node.
- Service
 - A *Service* is a set of replicated pods. It decouples work definitions from the pods.
- Route
 - A *Route* is a load balancing mechanism used to expose services externally.
- Build Config
 - A Build Configuration (BC) defines a build process for new container images.
- Deployment Config
 - A Deployment Configuration (DC) defines the template for a pod and manages deploying new images or configuration changes.
- Secret
 - A Secret is an object that contains a small amount of sensitive data such as a password, a token, or a key.

Relevant OpenShift Resources

- Config Map
 - A ConfigMap (CM) is an API object used to store non-confidential data in key-value pairs.
- Image
 - An *Image* is a portable package containing all content, binaries, and configuration data that define a container instance
- Image Stream
 - An image stream comprises one or more Docker images identified by tags. It presents a single virtual view of related images, similar to a Docker image repository
- Persistent Volume
 - A *PersistentVolume* (PV) is a piece of storage in the cluster.
- Persistent Volume Claim
 - A *PersistentVolumeClaim* (PVC) is a request for storage by a user.

Quiz Time



[This Photo](#) by Unknown Author is licensed under [CC BY](#)

Quiz Time

Choose 3

Which of the following statements are correct regarding container limitations?

- A. Containers are easily orchestrated in large numbers.
- B. Lack of automation increases response time to problems.
- C. Containers do not manage application failure inside them.
- D. Containers are not load-balanced.
- E. Containers are heavily isolated packaged applications.

Quiz Time

Choose 3

Which of the following statements are correct regarding container limitations?

- A. Containers are easily orchestrated in large numbers.
- B. Lack of automation increases response time to problems.**
- C. Containers do not manage application failure inside them.**
- D. Containers are not load-balanced.**
- E. Containers are heavily isolated packaged applications.

Quiz Time

Choose 2

Which of the following statements are correct regarding Kubernetes?

- A. Kubernetes is a container.
- B. Kubernetes can only use Docker containers.
- C. Kubernetes is a container orchestration system.
- D. Kubernetes simplifies management, deployment, and scaling of containerized applications.
- E. Applications managed in a Kubernetes cluster are harder to maintain.

Quiz Time

Choose 2

Which of the following statements are correct regarding Kubernetes?

- A. Kubernetes is a container.
- B. Kubernetes can only use Docker containers.
- C. Kubernetes is a container orchestration system.**
- D. Kubernetes simplifies management, deployment, and scaling of containerized applications.**
- E. Applications managed in a Kubernetes cluster are harder to maintain.

Quiz Time

Choose 3

Which of the following statements are true regarding Red Hat OpenShift?

- A. OpenShift provides additional features to a Kubernetes infrastructure.
- B. Kubernetes and OpenShift are mutually exclusive.
- C. OpenShift hosts use Red Hat Enterprise Linux as the base operating system.
- D. OpenShift simplifies development incorporating a Source-to-Image technology and CI/CD pipelines.
- E. OpenShift simplifies routing and load balancing.

Quiz Time

Choose 3

Which of the following statements are true regarding Red Hat OpenShift?

- A. OpenShift provides additional features to a Kubernetes infrastructure.**
- B. Kubernetes and OpenShift are mutually exclusive.
- C. OpenShift hosts use Red Hat Enterprise Linux as the base operating system.
- D. OpenShift simplifies development incorporating a Source-to-Image technology and CI/CD pipelines.**
- E. OpenShift simplifies routing and load balancing.**

Links

Links

Containers

<https://www.docker.com/resources/what-container>

Q&A

Q&A

*What questions
do you have?*

