Cloud 301 OpenShift

Virtual Participant Guide

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Contents

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Facilities and Logistics

Please Turn Off Any Other Electronic Devices

Close any open applications other than WebEx and turn off other electronic devices

WebEx functionality will be reviewed prior to the class starting

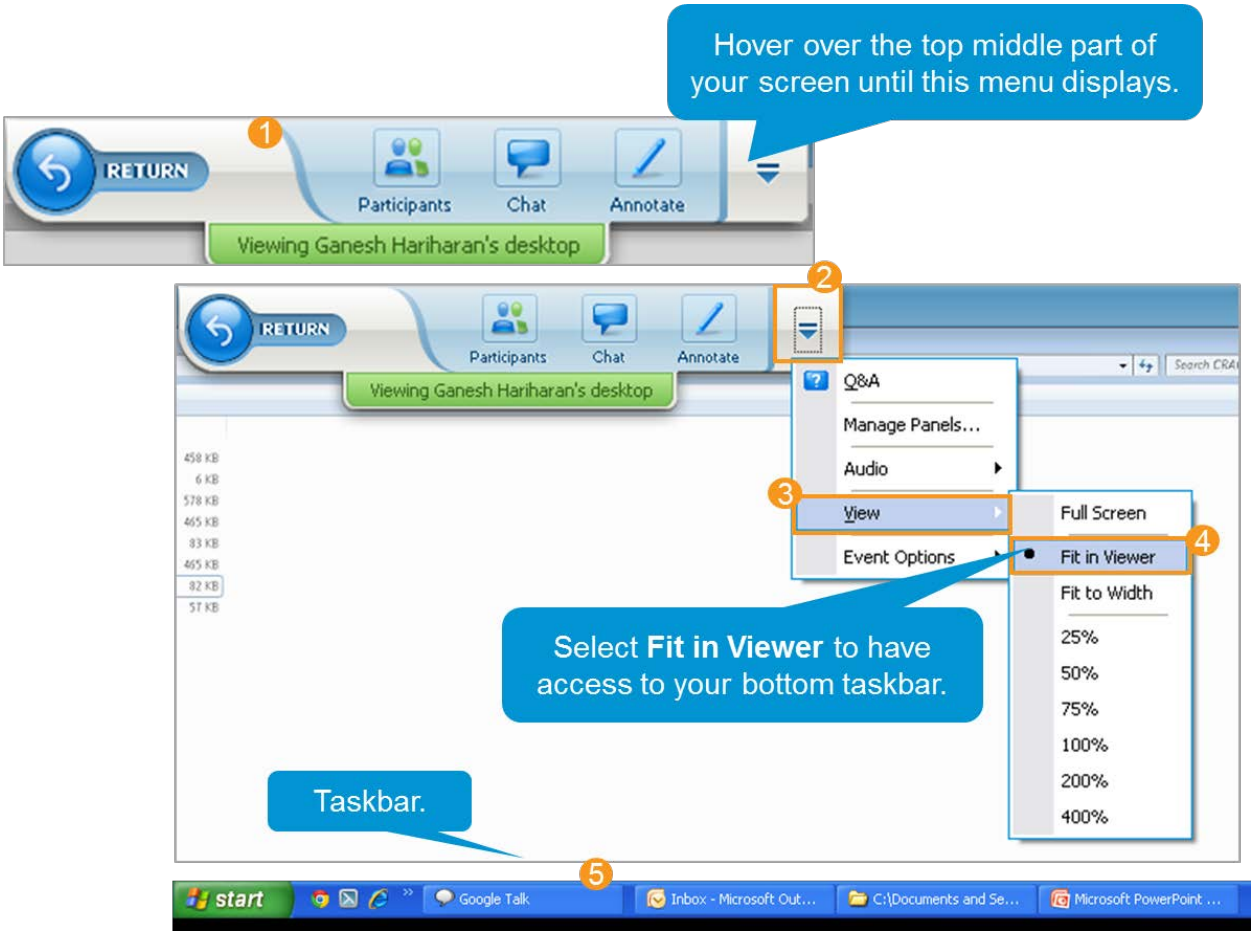
Ensure that the instructor has you signed in for the class

Breaks and Lunch

Q & A

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

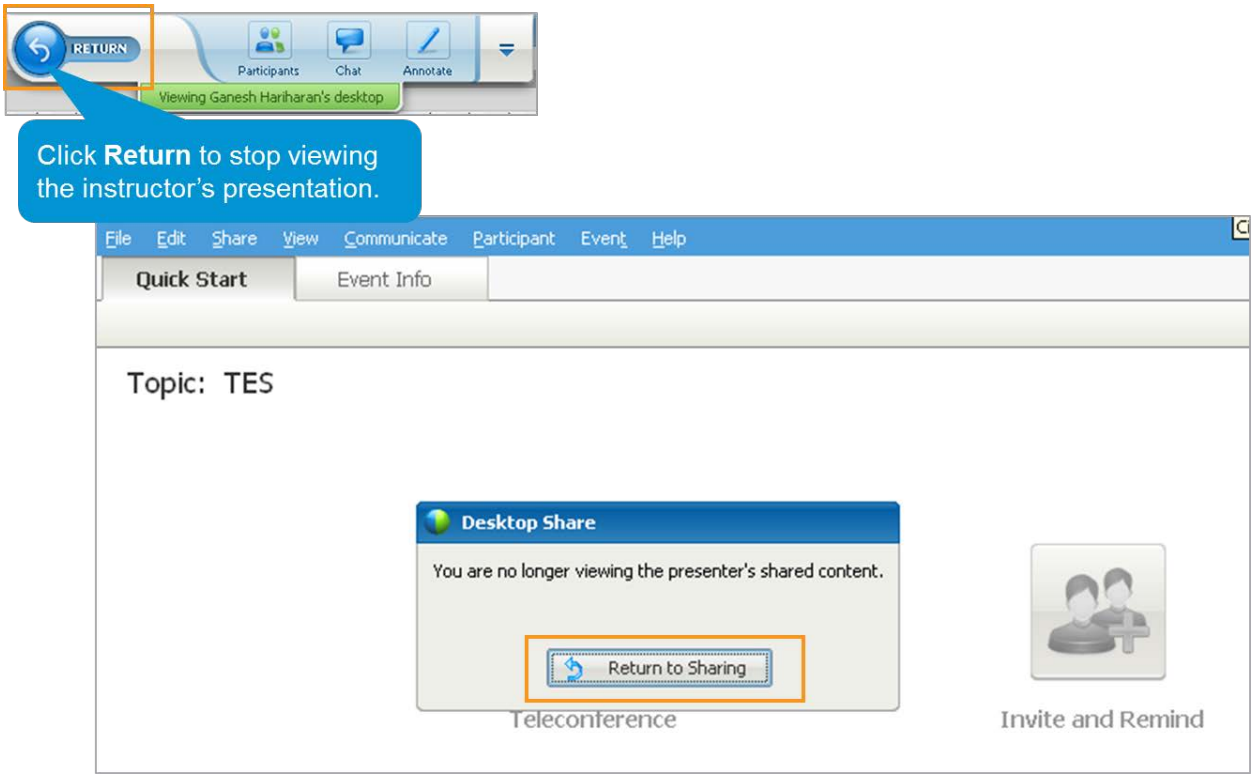
WebEx Logistics



- When you attend a WebEx session and the presenter shares their screen, the session may take over your full computer screen. If this occurs, you may not be able to access the task bar at the bottom of your computer screen that shows all of your open applications and your **Start** menu.
- If this happens, you can adjust the screen fit to allow you access to the task bar while viewing the presentation:
    1. To begin, hover over the top middle part of the screen until the drop-down menu shown appears.
    2. Click the drop-down arrow that appears to the right of the pull-down menu.
    3. Hover over the **View** option.
    4. Select **Fit in Viewer**.
    5. Once you make this selection, the task-bar will reappear at the bottom of your computer screen.

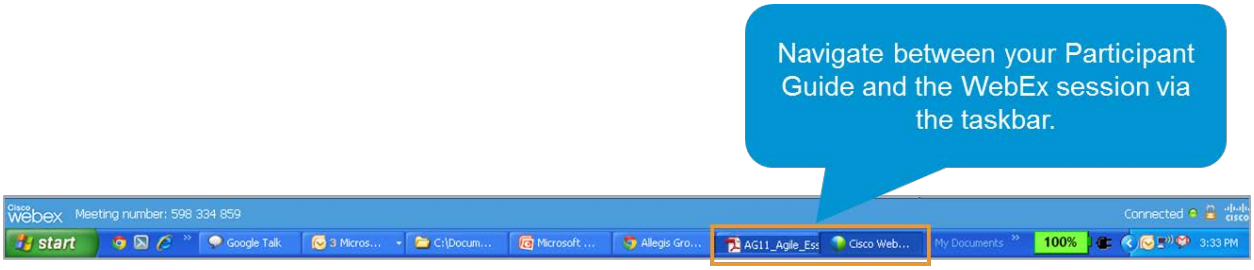
Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

WebEx Logistics (Cont.)



- If you would like to stop viewing the presentation at any time, click **Return.**
- To return to the presentation, click **Return to Sharing.**

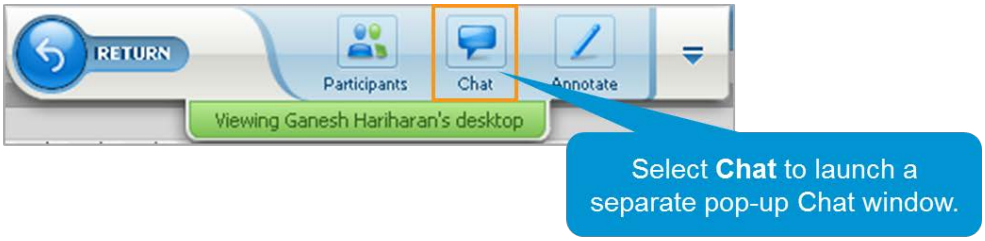
WebEx Logistics (Cont.)

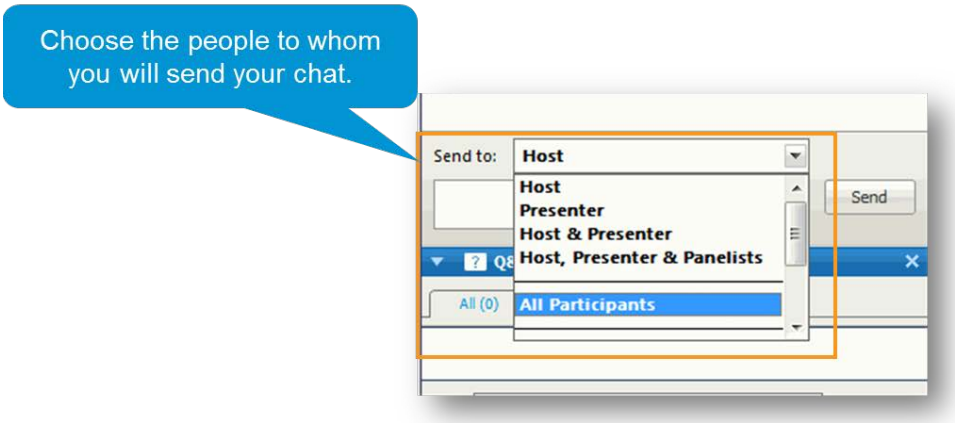


- During this class, if you are using an electronic (PDF) version of the Participant Guide, you may need to switch between viewing the WebEx presentation and your Participant Guide.
- Once you have the Participant Guide open, you can navigate between the two via the links that appear in your taskbar.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Chat Window

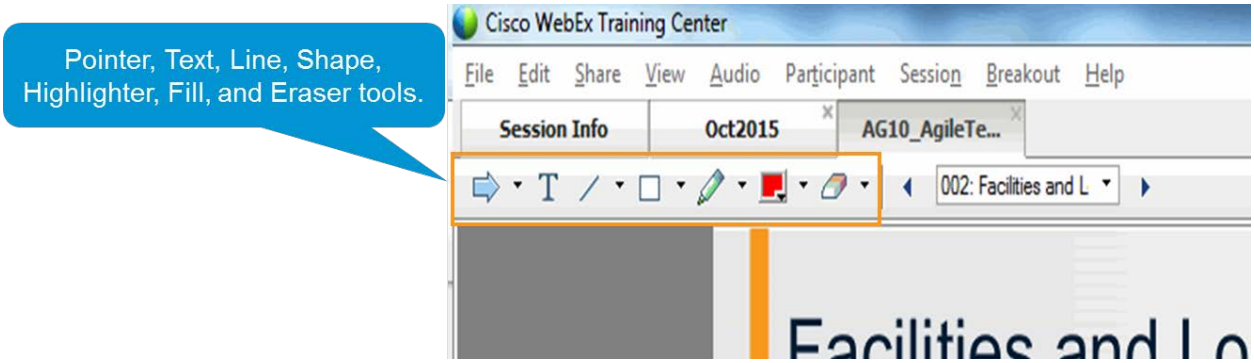




- During this session, you will need to use the Chat panel to answer knowledge check questions, and to ask the instructor any questions you may have.
- To access the Chat panel while the session is launched, select **Chat** from the menu we introduced on the previous page—remember that this menu appears when you hover over the top middle portion of your screen.
- After you click **Chat**, a new Chat window will open. You can move this window anywhere on your screen so you can easily access it while you are watching the presentation.
- You can choose the person or group of people to whom you will send your chat by making a selection from the **Send to:** drop-down menu. All interactivities require responses to be sent to All Participants.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Annotation Tools

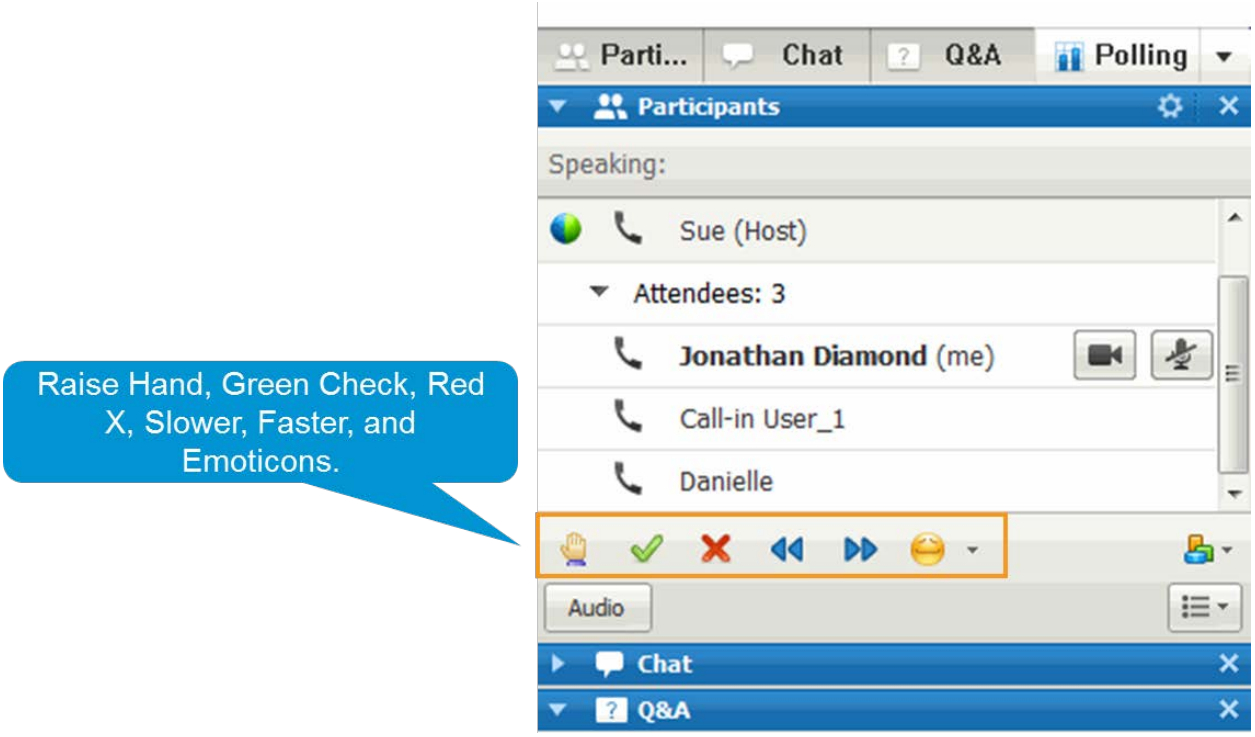


The **Annotation Tools** are used to interact with slide content. Each virtual interactivity will provide specific instructions for how you should participate using the tools available. The **Annotation Tools** include the following:

- **Pointer**: Lets you point out text and graphics on shared content. The pointer displays an arrow with your name and annotation color. Clicking this button again and then clicking the Close button turns off the text tool.
- **Text** Lets you type text on shared content. Attendees can view the text you have entered after you type it and click your mouse in the content viewer, outside the text box. To change the font, on the **Edit** menu, choose **Font**. Clicking this button again and then clicking the Close button turns off the text tool.
- **Line** Lets you draw lines and arrows on shared content. For more options, click the downward-pointing arrow. Clicking this button again and then clicking the Close button turns off the Line tool.
- **Rectangle** Lets you draw shapes, such as rectangles and ellipses on shared content. For more options, click the downward-pointing arrow. Clicking this button again and then clicking the Close button turns off the Rectangle tool.
- **Highlighter** Lets you highlight text and other elements in shared content. For more options, click the downward-pointing arrow. Clicking this button again and then clicking the Close button turns off the Highlighter tool.
- **Annotation Color** Displays the Annotation Color palette. Select a color to annotate shared content. The Annotation Color palette closes.
- **Eraser** Erases text and annotations or clears pointers on shared content. To erase a single annotation, click it in the viewer. For more options, click the downward-pointing arrow. Clicking this button again and clicking the Close button turns off the Eraser tool.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Participant Window



The **Participants Window** has various tools that can be used including:

- **Raise Hand**: Lets you get the instructors attention to ask a question or volunteer to respond.
- **Green Check/Red X**: Used to respond to Yes/No or True False questions from instructor.
- **Slower/Faster:** Used to ask the instructor to speed up or slow down their pace of instruction.
- **Emoticons**: Various emoticons can be used for communication purposes. The Break icon is commonly used in virtual training sessions.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Interactivity: Dream Vacation

Question: Where is your dream vacation location?

Instructions: Use Pointer tool.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Course Agenda

Topics:

- Module 1: An Introduction to OpenShift
- Module 2: Deploying and Managing Applications
- Module 3: Complex Deployments
- Module 4: Troubleshooting Applications

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

This page intentionally left blank.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift Intro

Module 1

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Module Objectives

After this module, participants will be able to:

- Describe the purpose and use cases for OpenShift
- Understand the OpenShift architecture
- Identify the features within OpenShift
- Identify an OpenShift web console
- Describe the Command Line Interface (CLI)

In addition to the objectives you identified in taking this module, these are the ones we expect to achieve based on the material provided.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

What is OpenShift?

- OpenShift is an open-source container platform solution offered by Red Hat®, which allows developers to quickly develop, maintain, and deploy containers on a server.
- OpenShift uses both Kubernetes and Docker to function as a standalone container platform solution.
- OpenShift is a type of cloud service classified as a Platform as a Service (PaaS).
- The OpenShift website describes their own service in this way:

Build, deploy, and scale on any infrastructure

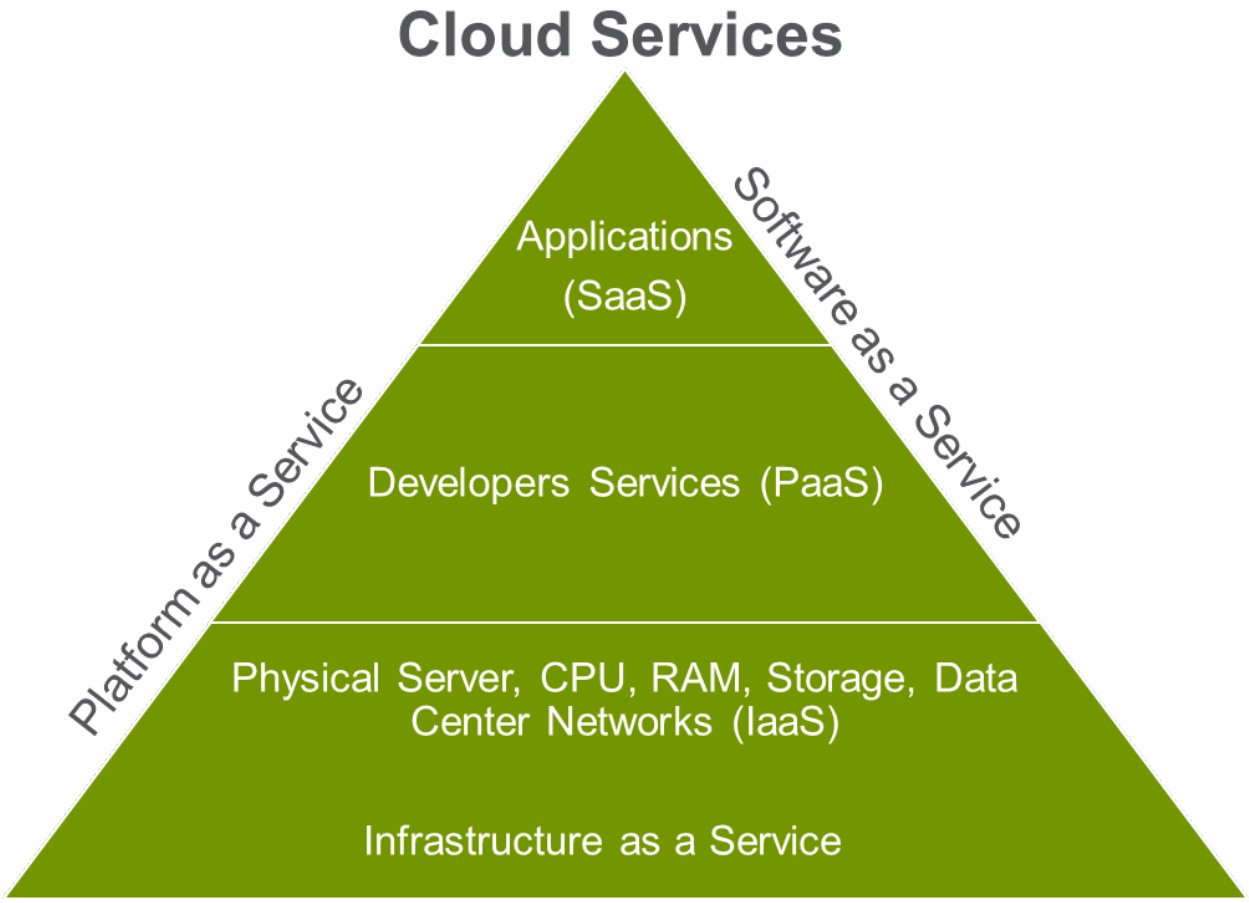
Automate the build, deployment, and management of applications so that you can focus on writing the code for your business, startup, or next big idea.

OpenShift is a RedHat initiative. It is a public, Open Source cloud which helps you maintain, deploy, and use containers. Its goal is to standardize scalable applications. For that, it uses Kubernetes and Docker, that is, the latest and greatest technologies. Its value-add, however, is to create a generic mechanism for deploying and scaling on any infrastructure.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Types of Cloud Services

- Infrastructure as a Service (IaaS)
- Platform as a Service (PaaS)
- Software as a Service (SaaS)



Cloud vendors generally offer different options of their services and most of the Cloud vendors will give you different options of services which may include IaaS, PaaS, and or SaaS.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

IaaS, PaaS, and SaaS

Infrastructure as a Service (IaaS)

IaaS is a type of cloud service which provides:

- Servers and server related hardware
- Virtualization
- Storage
- Networking.

If you utilize an IaaS service you will need to provide the:

- Applications
- Data
- Runtime
- Middleware
- Operating system.

Examples of IaaS service include:

- Amazon Elastic Compute Cloud (EC2), Microsoft Azure VM, Google Compute Engine (GCE)

Platform as a Service (PaaS)

PaaS is a type of cloud service which provides:

- Runtime
- Middleware
- Operating system
- Server and server related hardware
- Virtualization
- Storage
- Networking

If you utilize a PaaS service you will need to provide the:

- Applications
- Data

Examples of PaaS include:

- OpenShift, Heroku, Amazon Elastic Beanstalk, Google App Engine

Software as a Service (SaaS)

SaaS is a type of cloud service which provides:

- Application
- Data
- Runtime
- Middleware
- Operating system
- Server and server related hardware
- Virtualization
- Storage
- Networking

This includes software which is available on a subscription basis and is centrally hosted.

Examples of SaaS include:

- Salesforce.com, Microsoft Office 365 Cloud, Google Apps

What does OpenShift add to Docker and Kubernetes on which it is built? It is a Platform-as-a-Server (PaaS) offering, filling in the missing parts.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Interactivity: Matching Question

Question: Match each type of cloud service with examples of it.

Instructions: Raise Hand to volunteer and the n use the Line tool.

PaaS	Microsoft Office 365 Cloud, Google Apps
SaaS	Microsoft Azure VM, Amazon EC2
IaaS	OpenShift, Google App Engine

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

What is a Container?

A container image is a lightweight executable software package which contains:

- The software code
- Runtime
- System libraries
- Settings

It is comparable to a lighter version of a virtual machine.

Running applications using containers helps the application be compatible with different types of systems.

First, we need to look at Docker on which OpenShift is built, and more generically, review what a container is. Here is a little summary as a review, as we covered containers in the Essentials course.

To deliver an application there initially were two choices:

1. Deliver just the application without any supporting libraries and rely on the environment.
2. Deliver an application in a virtual machine with all of the environment, plus the operating system.

The first approach suffered from library dependency management hell, the second one was overkill. Containers are in between. They rely on the underlying operating system but provide the libraries that are needed by the application beyond the operating system.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Docker, Kubernetes, and OpenShift

- **Docker**, **Kubernetes**, and **OpenShift** work together to get your apps up and running.
- OpenShift is an all-in-one solution that uses Docker and Kubernetes to deploy functioning applications on the cloud or on your on-premises server.

Many people find it hard to grasp, what else is needed when you have Docker and Kubernetes already. The answer is that OpenShift deploys Docker and Kubernetes solution in any place, be it a public cloud or on-premises server. Docker is the container, k8s is the way to organize containers, and OpenShift sits on top.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

What is Docker and how is it Different than OpenShift?

Docker

Docker is a software solution which provides the framework and capability to convert existing applications and infrastructure into individualized containers to enable them to be compatible on any type of system.

OpenShift uses Docker-based container applications as a part of its container platform solution, but OpenShift also:

- Provides a system for the containers to operate.
- Deploys the containers into functioning applications on the cloud or on on-premises infrastructures.

OpenShift also allows you to submit the source code of an application and it uses Docker to create a container for you.

You can also deploy applications which are already in container image files.

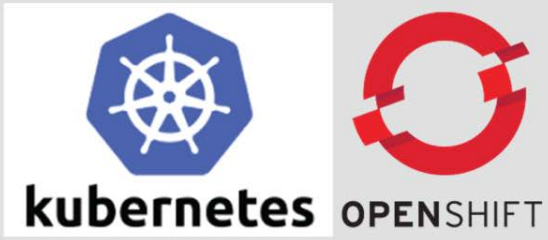
This slide gives another answer to what OpenShift offers above Docker. The short answer is that OpenShift organizes Docker deployments in any environment or cloud. Docker gives you a way to take an application and deploy it into a container. It reuses underlying services. OpenShift uses Docker, but it gives you a system and deploys those containers into a functional application, either in the cloud or on-prem. Optum uses it on-prem.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

What is Kubernetes and how is it Different than OpenShift?

Kubernetes

- Kubernetes is an open-source container orchestration system which allows the applications and containers to communicate and function in a single unit.
- OpenShift uses Kubernetes as a part of the container platform, but OpenShift also provides a solution to deploy and manage the Kubernetes nodes on the server.



Another answer to the above question is that OpenShift provides a solution to deploy and manage Kubernetes nodes.

For more on OpenShift, see: https://www.openshift.com/learn/topics/kubernetes/

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Interactivity: Poll Question

Question: Due to the cloud's nature as a shared resource; security is a key concern. Select the three (3) cloud potentially vulnerable areas that have become a priority for organizations contracting with a cloud computing provider.

A. Anti-Virus software, Firewalls, Stronger Passwords
B. Identity Management, Privacy, Access Control
C. Data Compliance, Data Governance, Industry-Specific Data
D. Data at Rest, Data in Use, Data in Motion

OpenShift Services

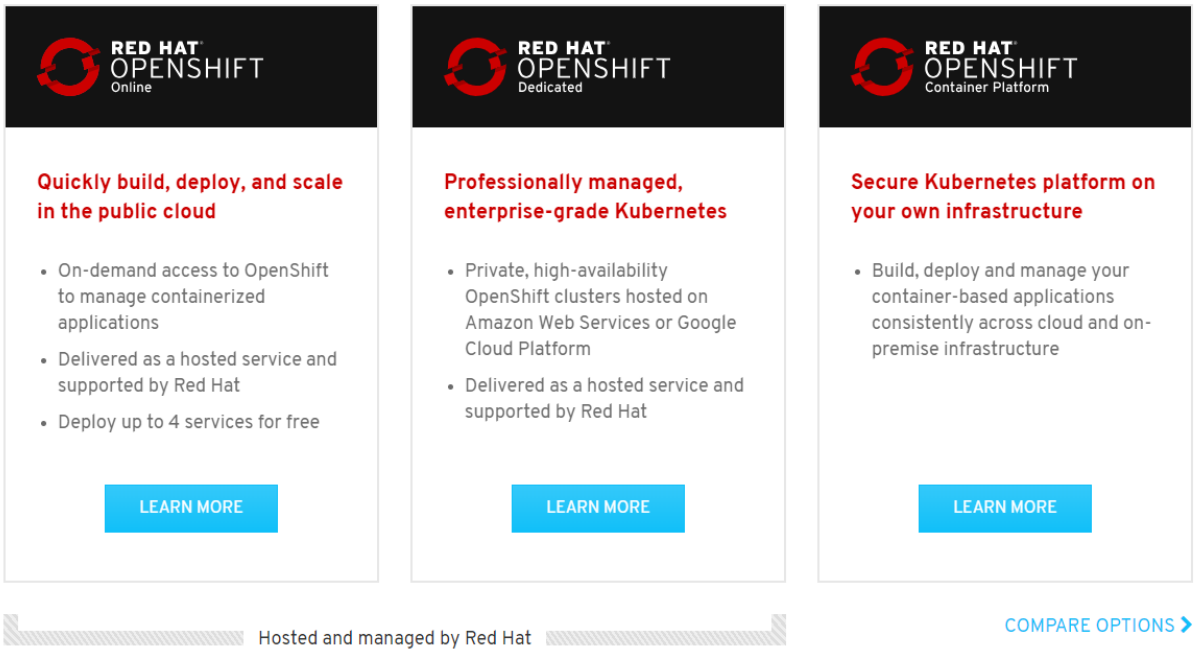
OpenShift offers several types of services:

- OpenShift Online
- OpenShift Dedicated
- OpenShift Container Platform
- OpenShift Origin

In accordance with the professed goal of OpenShift, to simplify the deployment in every environment, the possible environments are Online, Dedicated, and Container Platform. There is also a community version of OpenShift called OpenShift Origin.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Types of Services that OpenShift Offers



These services are what OpenShift offers to organize deployments. Optum hosts it on premises. OpenShift Origin (OKD) is the Upstream OpenSource project that the commercial offerings are based on. The Optum internal offerings include both OKD and OCP.

OpenShift Online

OpenShift Online:

- A web-based service which allows developers to use OpenShift as a container platform solution on the Red Hat® public server. There are two plans:
  - Starter plan: A free version for testing purposes where you can deploy 1 project on their public cloud.
  - Pro: A limited professional plan where you can run 10 projects on their public cloud for $50.00 / month.

As can be seen from this slide, OpenShift is actually a web-based service, not only software. It can be a free layer or a paid layer, which starts at a moderate $50/month for up to 10 projects.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Prerequisites to Running OpenShift Online and OpenShift Dedicated

It doesn't take much to get applications running with OpenShift Online or OpenShift Dedicated. Here are a few things you would need:

- A GitHub account
- Computer system with internet connectivity
- OS: Windows / Linux / MAC
- The source code of the applications to run or applications in a Docker container file image format
- The source code or container file images need to be in an accessible location such as a GitHub repository
- Git software installed on the local computer

There will also be separate developer requirements for the development of your applications.

For the OpenShift lab exercises, participants will only need a GitHub account and computer system with internet activity to start the exercise.

To complete the OpenShift Lab: *Using Command Line Interface to Automatically Rebuild Your Application*, participants will need a text editor other than notepad installed to their system.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift Dedicated and Container Platform

OpenShift Dedicated

An enterprise solution which provides large-scale private clusters for use with OpenShift services on Amazon Web Services (AMS) or Google Cloud Platform (GCP).

- Pricing starts at $48,000/year.
- Please contact OpenShift sales directly for customized pricing.
- The private clusters with AMS or GCP are included.
- Supports connectivity with your internal network.
- No specific hardware requirements specified since it is cloud-based.

OpenShift Container Platform

- A solution to use OpenShift to "build, deploy and scale" your application framework in both on-premises infrastructure as well as on the cloud.
- Works by creating an integrated container registry called OpenShift Container Registry (OCR) to operate on the cloud or on on-premises hosting systems.

Playing along with the goals of deploying anywhere, OpenShift can deploy on AWS or on GCP. These prices are more significant. However, if we are talking large deployments, they are less than half the salary of a support person.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift Community (OKD)

Let's review the terminology:

- Docker:
  - Standardized Linux container packaging for applications and their dependencies
- Kubernetes:
  - Standardized Linux container packaging for applications and their dependencies
- OKD:
  - A distribution of Kubernetes
  - Optimized for continuous application development and multi-tenant deployment
- OKD adds:
  - Developer and operations-centric tools on top of Kubernetes
  - Enables rapid application development
  - Easy deployment and scaling
  - Long-term lifecycle maintenance for small and large teams.
- OKD is what powers RedHat's OpenShift

Here we see a summary of the features of OKD to help put it in perspective. Compared to the other components:

- OKD is a distribution of Kubernetes that adds CI/CD
- Kubernetes is what controls Docker deployments
- Docker is a container technology

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Interactivity: Poll Question

Question: Which of the following is a distribution of Kubernetes that is optimized for continuous application development?

A. OKD
B. Docker
C. OCR
D. SaaS

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift Architecture

What Services Does OpenShift Container Platform Provide?

Docker: Packages the lightweight container images.

Kubernetes: Orchestrates the containers and helps to manage the clusters.

OpenShift Container Platform adds the following:

- Management of:
  - Source code
  - Builds
  - Deployment of applications
  - Images - container files
- Scaling of the applications
- User tracking
- Network infrastructure

Many people ask, “What does OpenShift add to Kubernetes?” The answer is that OpenShift puts Kubernetes into the real-world framework. It adds these services:

- Management of the source code
- Building, deployment, and image creation

It also takes care of application scaling. Scaling of the applications means deploying and managing of the increasing number of instances of the application. It provides facilities for user tracking, and it builds the network infrastructure.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Infrastructure Components of OpenShift Container Platform

- Containers and images
- Pods
- Services
- Nodes
- Replication Controllers
- Routes
- User

Infrastructure components:

- Containers and images - such as Docker container images.
- Pods - a Kubernetes subunit.
- One or more container in a functioning virtual unit - this allows the containers to communicate with each other.
- Services - define sets of pods and policies of when/how to access them.
- Node - a single server unit consisting of pods and services.
- Replication Controllers - monitor multiple instances of replicas of pods to make sure they are running. If there are insufficient replicas of a pod operating, the replication controller will increase the number of functioning pods to the required number.
- Routes - allow users to access a service by a hostname, such as a web address.
- User - humans or machines/software interacting with the OpenShift Container Platform.

Information obtained from these sites.

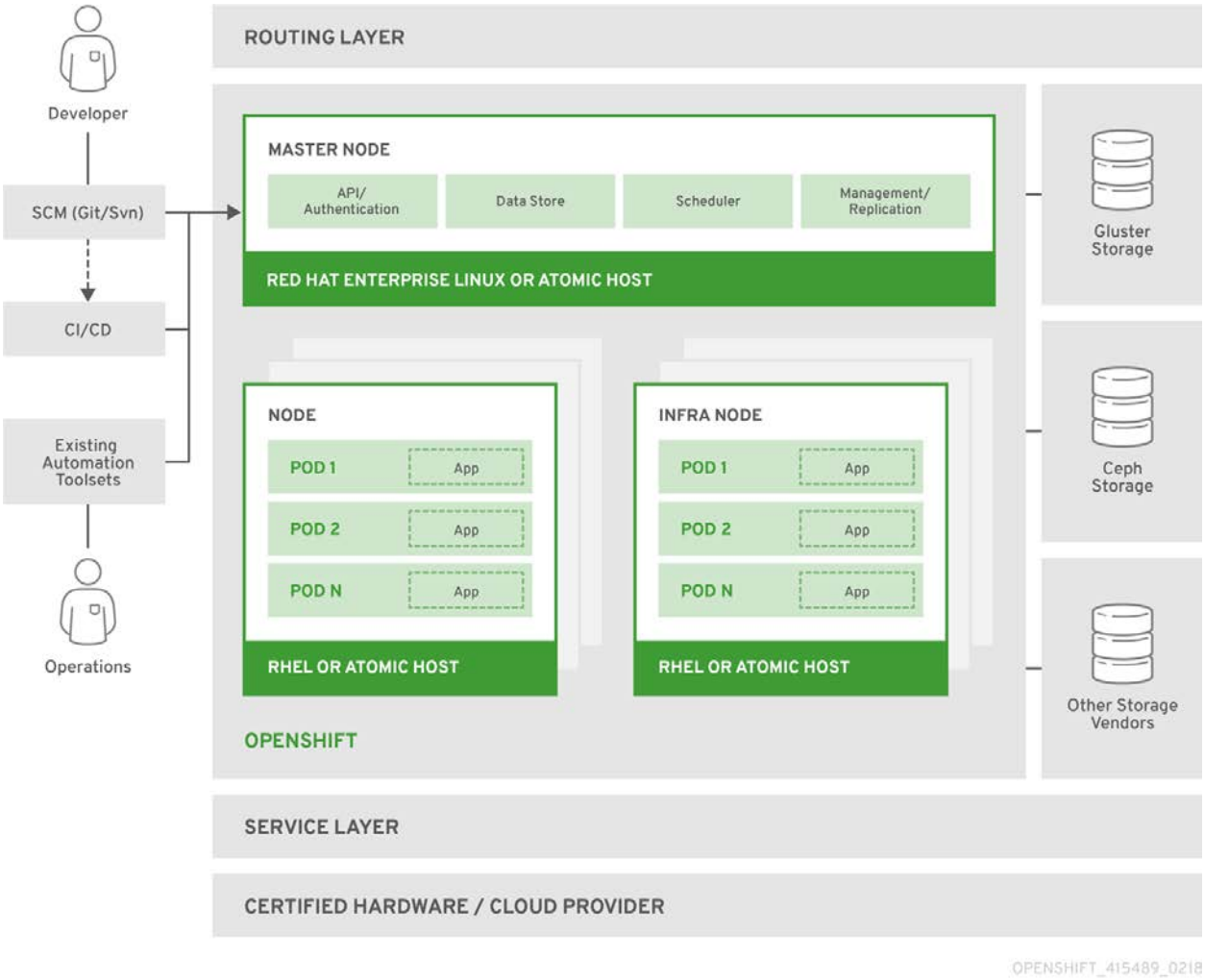
- https://docs.OpenShift.com/container-platform/3.9/architecture/core_concepts/index.html
- https://docs.OpenShift.com/container-platform/3.9/architecture/core_concepts/pods_and_services.html#architecture-core-concepts-pods-and-services

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

- https://kubernetes.io/docs/concepts/services-networking/service/#defining-a-service
- https://docs.OpenShift.com/container-platform/3.9/architecture/core_concepts/deployments.html#architecture-core-concepts-deployments

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Diagram of OpenShift Container Platform Infrastructure



The OpenShift container permits deployment of various services, as illustrated in the above diagram. It also illustrates the architecture of how multiple services are brought together.

This diagram illustrates the architecture of the container platform infrastructure as implemented by OpenShift.

Each worker node is built on RHEL, or on an atomic host. There are also infrastructure nodes with the same structure.

There is also a master node that controls the deployment and contains the following services:

- API
- DataStore
- Scheduler

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

- Maintenance

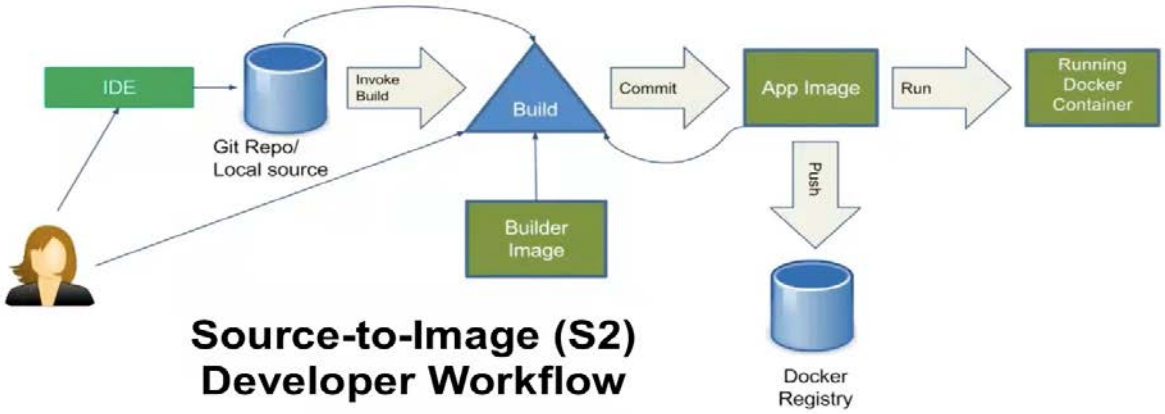
The developer stores his configurations in the SCM, from which CI/CD takes its information.

The operation folks then add automation tools that interact with the CI/CD.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Source to Image

- OpenShift allows you to upload a container image which you have already made using Docker.
- OpenShift also allows you to upload the source code for supported programming languages to OpenShift and OpenShift will directly convert it into container images.
- Source-to-Image (S2I) is a tool with OpenShift, which allows a user to easily convert the source code to a Docker container image file.



Source-to-Image is a service that allows you to reuse the work you have already done in Docker and bring it into OpenShift.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Interactivity: Poll Question

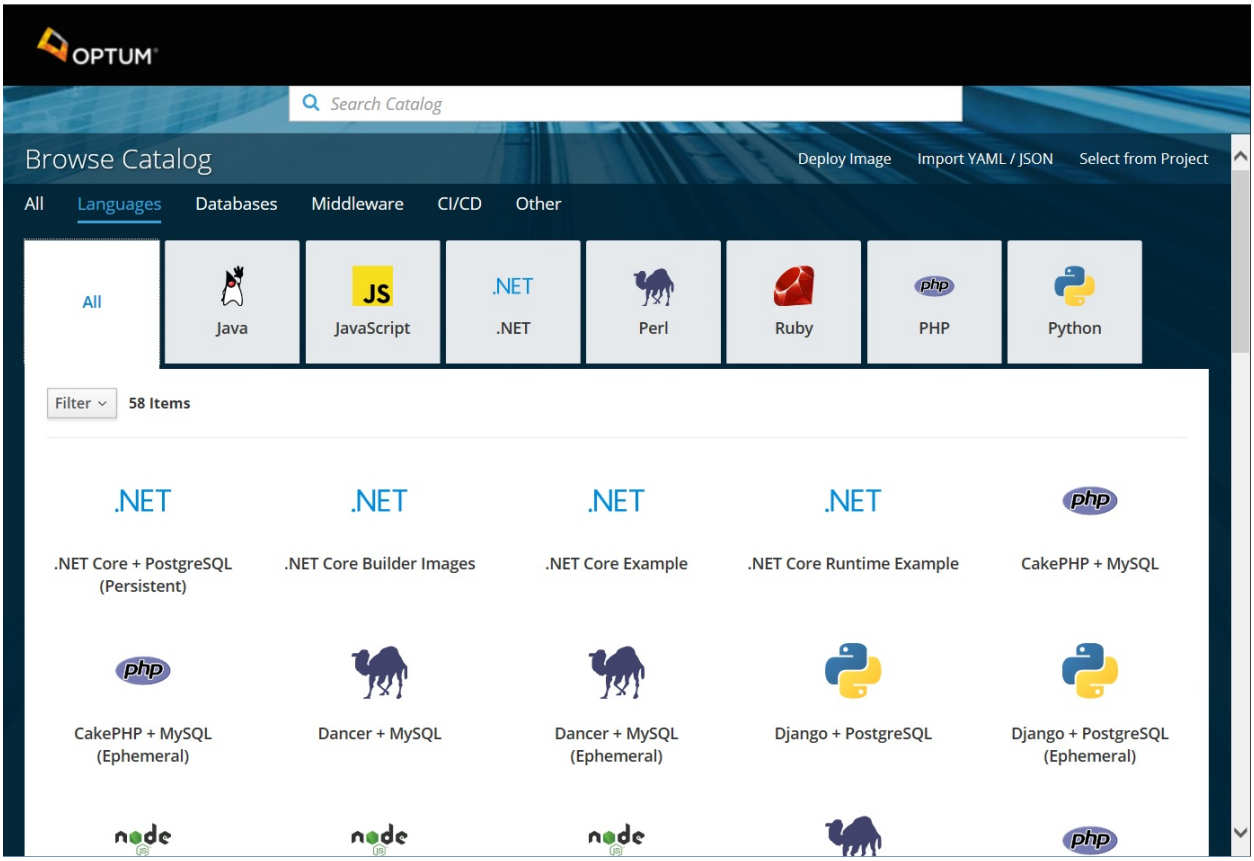
Question: OpenShift allows you to a container image and what?

A. Source code for supported programming languages
B. Pods to hold containers
C. User images
D. Clusters

OpenShift Features

Languages Catalog

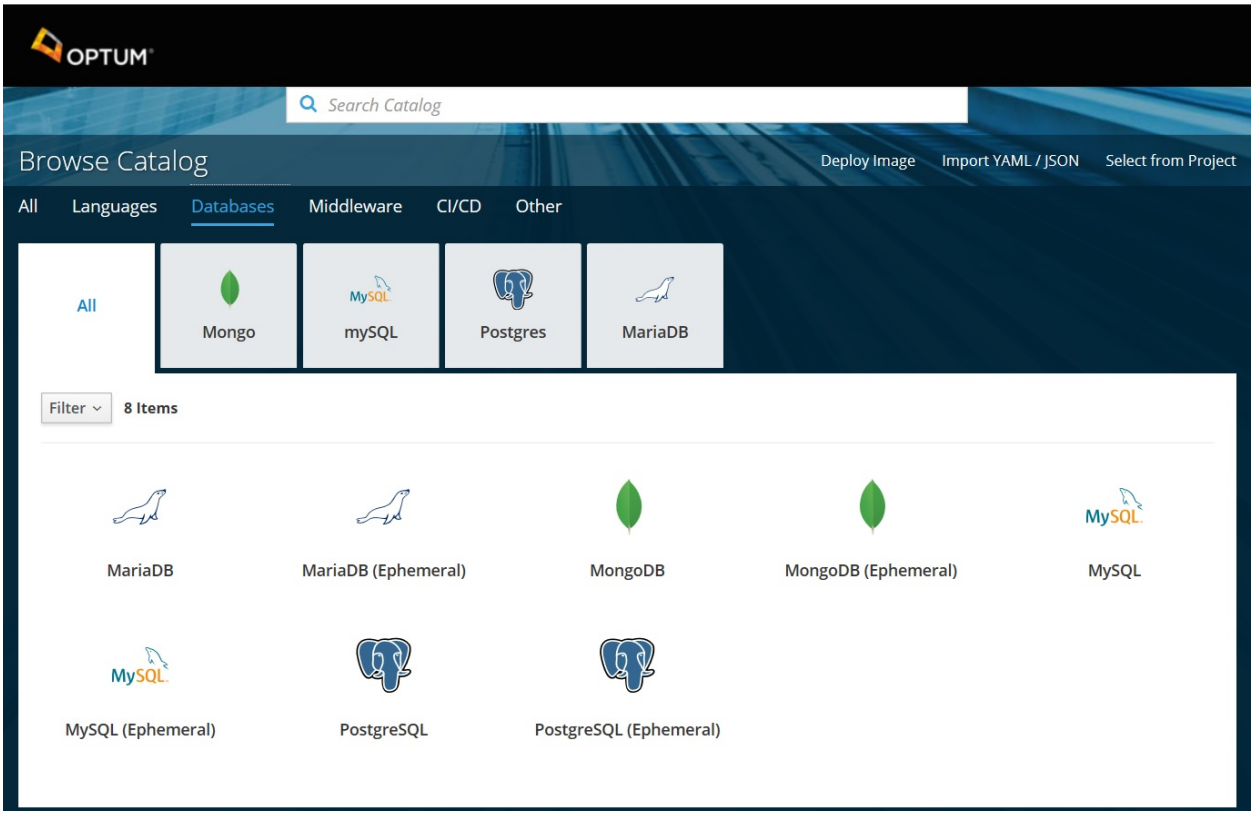
A catalog for languages for applying your source code.



As you can see, most modern languages can be used with OpenShift. The table above provides the list of supported languages, but more can be added with time.

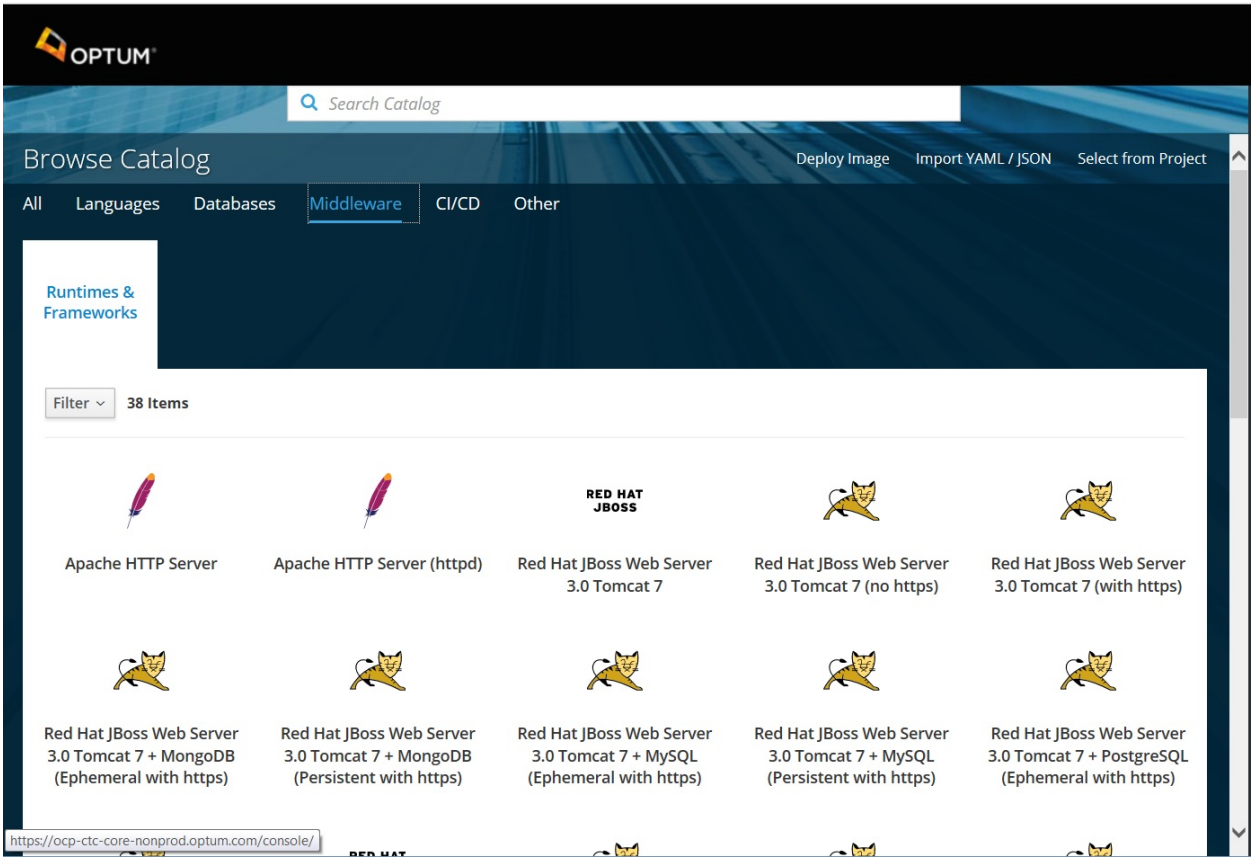
Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Database Catalog



Just as with languages, you have a choice of the database that will store the back-end information for your application. MongoDB is the largest and most well-known NoSQL among developers. This slide presents the current list of choices for the back-end database. Cassandra is another choice.

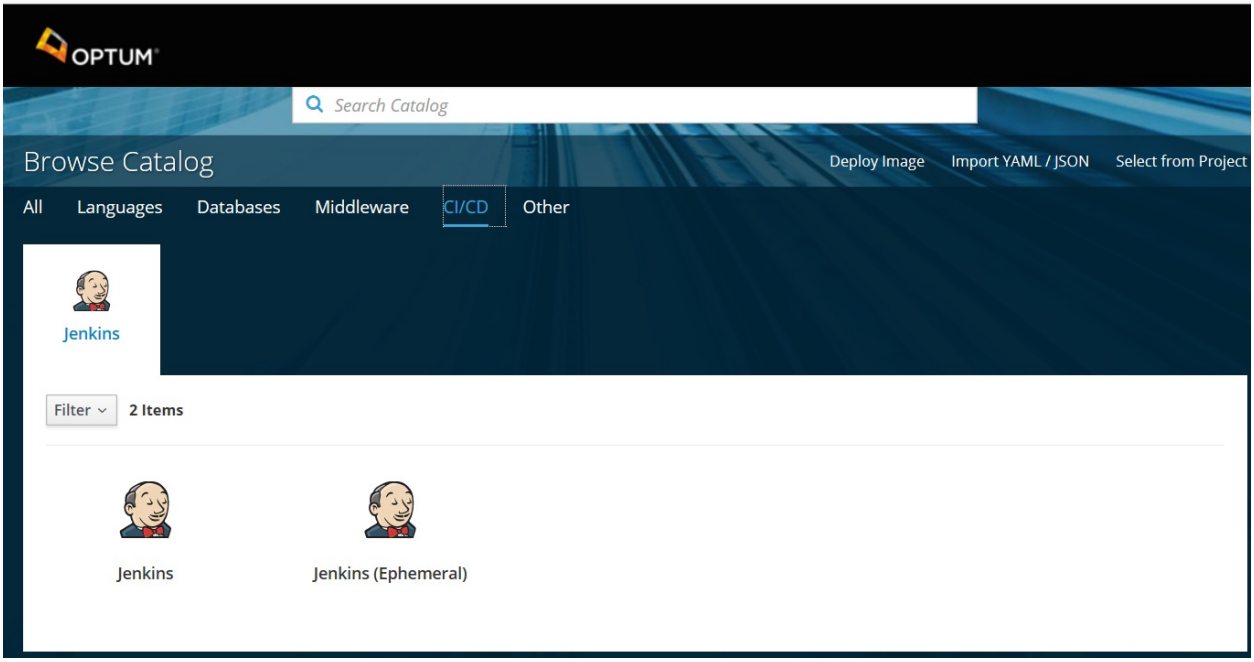
Middleware Catalog



Middleware is the next choice in implementing your application with OpenShift. This slide presents the current four choices in middleware.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

CI/CD Catalog



CI/CD stands for continuous integration and continuous delivery. Jenkins Pipeline is an example of CI/CD. Jenkins is currently the only choice for Continuous Integration and Continuous Delivery. That is OK, however, since Jenkins is

- Open source and therefore
- Free
- Well-tested
- De-facto standard in CI/CD

One cannot over-emphasize the importance of continuous testing in the modern software development, and especially with automated deployments like OpenShift. Optum has an enterprise Jenkins deployment that can be used, which minimizes the need to use a Jenkins within OpenShift.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Support for Many Types of Technologies & Languages

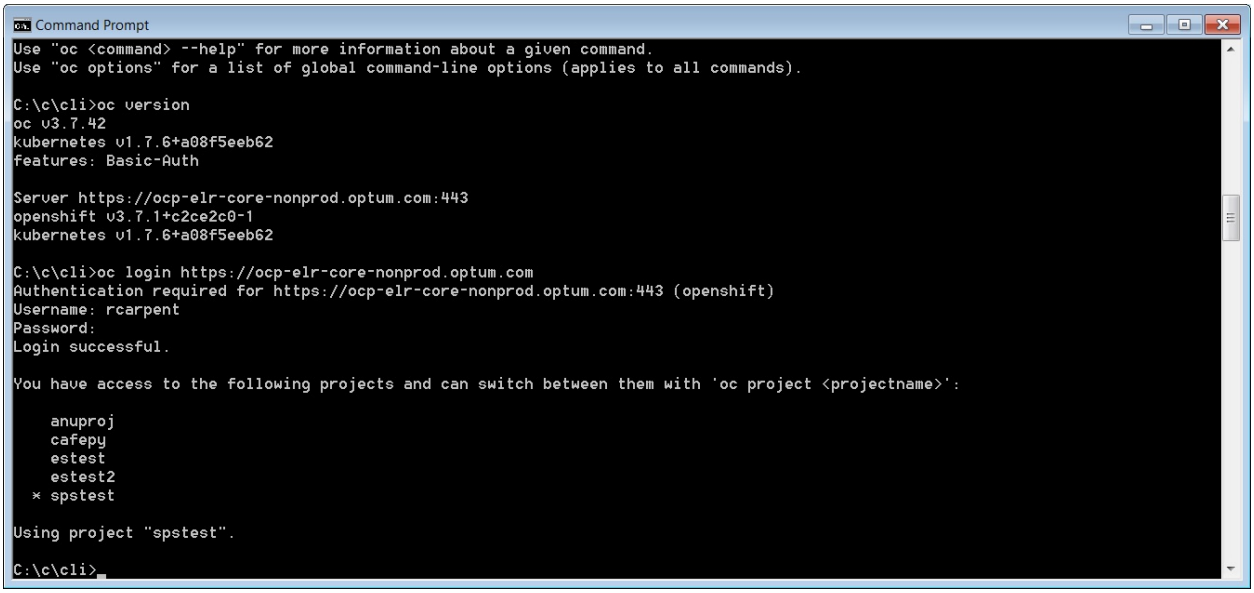
OpenShift Online supports many types of container files, programming languages, databases, and other technologies, including the following and more:

In today's world, you want to support as many development languages as possible. This diagram shows eight, and there is more. See the complete list of languages supported at this link.
https://www.OpenShift.com/products/features/containers/#online3

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Command Line Interface for OpenShift

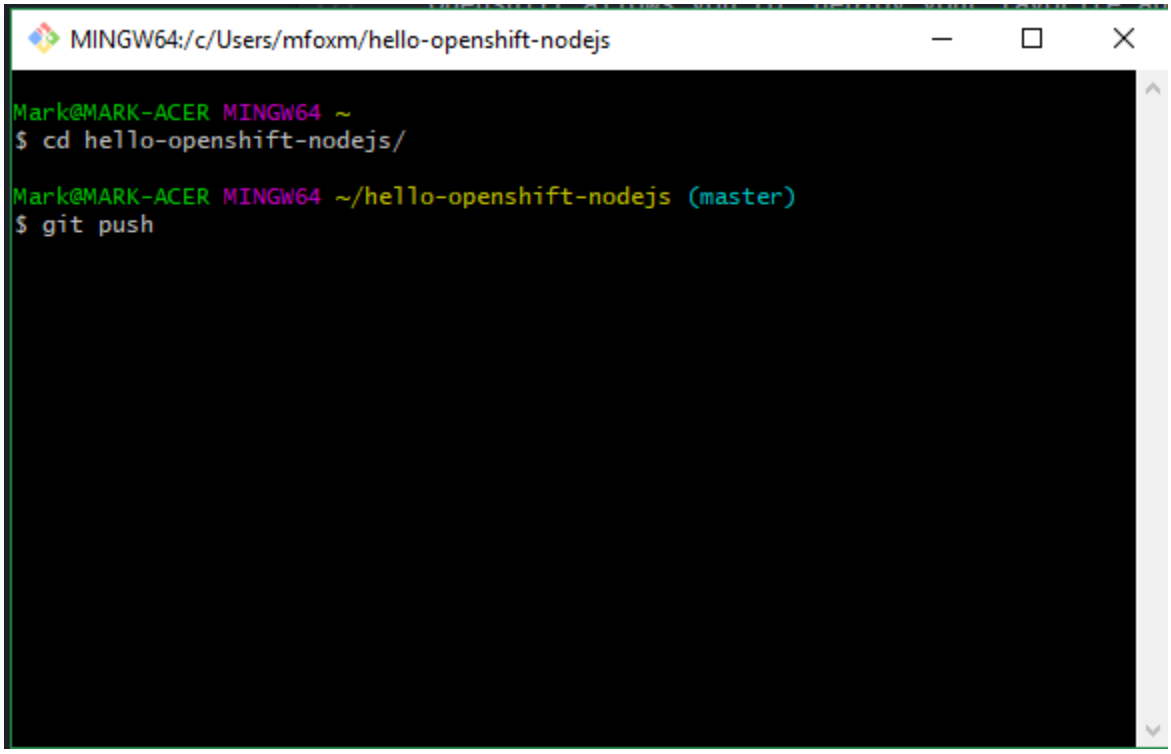
A command line interface allows for more powerful management of your apps.



OpenShift is fully enabled with the command-line interface. Here is an example of how you would start the deployment from the command line. Remember that the OC command line can be installed from the app store, which we saw earlier.

Easily Update Your Changes Using a "git push" Command

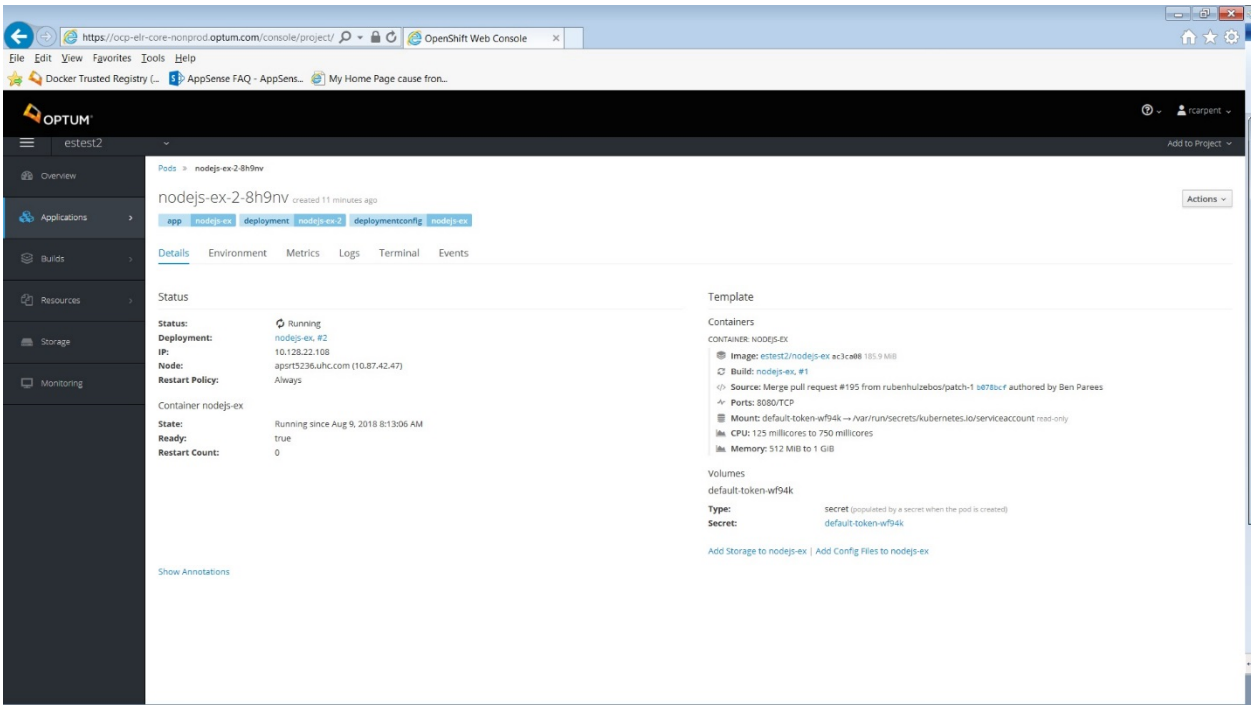
OpenShift allows you the ability to automatically rebuild the applications whenever you push your updates to the repository.



The major operations will be done with Git and GitHub.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Tools Available to Easily Deploy, Monitor, and Manage Applications



Here is the deployment screen that you will see.

OpenShift Reviews

Matt D'Angelo at Business.com (5/23/2018) "Best Cloud Computing Services."

- "OpenShift by Red Hat® is a platform as a service (PaaS) designed for developers. Many typical systems administration tasks, like virtual server provisioning, configuration and scaling, are automated so developers can spend more time on code and less time configuring operating systems and installing libraries and packages.

This comment emphases the importance and the usefulness of OpenShift for developers.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift Reviews (Cont.)

Aivann Carlo Cariaga with Upwark (5/3/2018) "Great for solo and large Enterprises."

- "OpenShift is a great way to start, especially their free tier. It is excellent for software engineers who want to learn PaaS. I have tested many PaaS system from Heroku, IBM Bluemix, to Azure Microsoft. It is great for solo developers and large enterprises."

As a platform-as-a-service, OpenShift gets positive reviews from developers, commenting on its use in the enterprise.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Discussion: OpenShift Pros and Cons

Questions:

- What are the value-add propositions of OpenShift?
- What does OpenShift add for modern software development practices?
- What languages are available for OpenShift implementations?

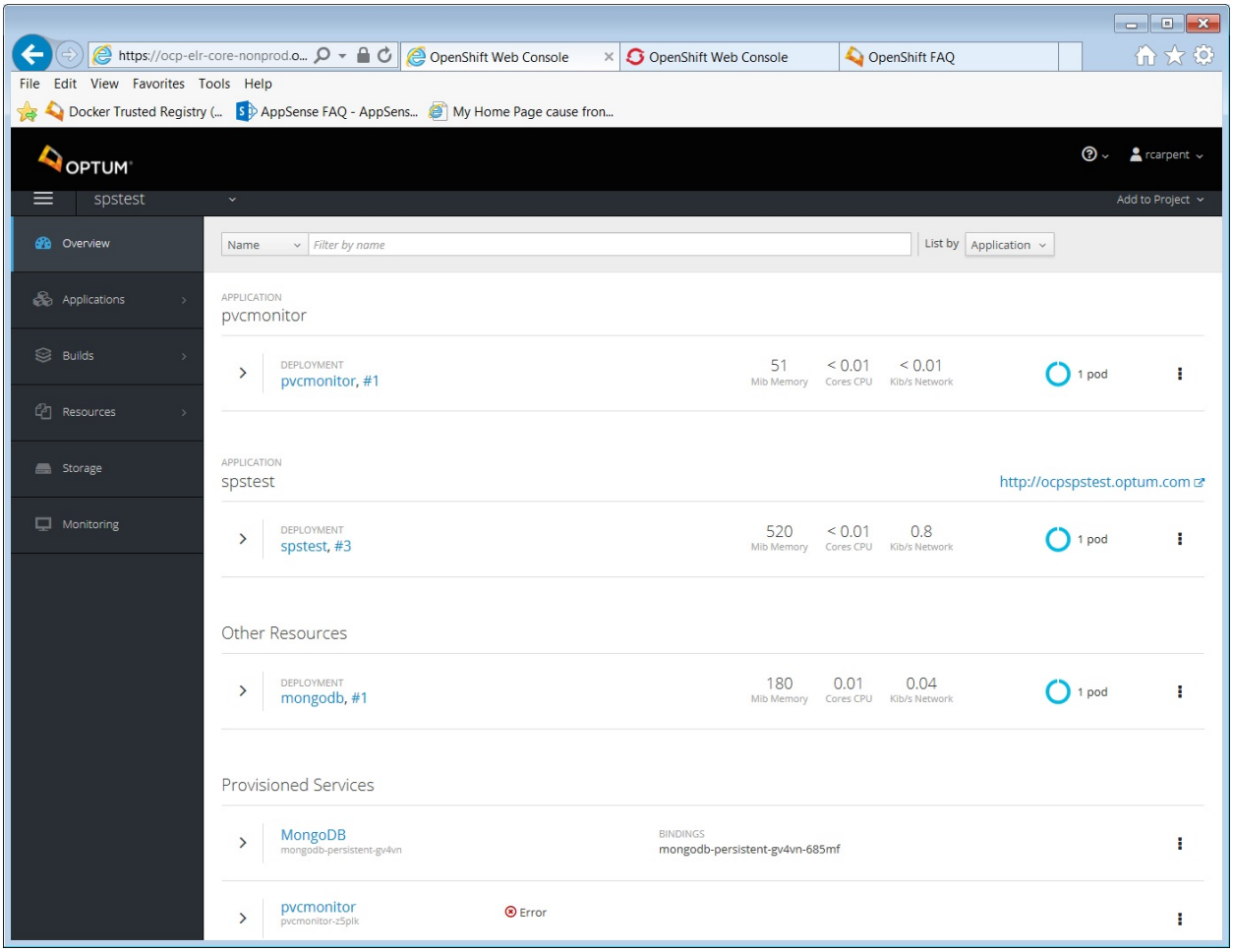
Instructions: Enter your answer into Chat (All Participants).

Take a moment to think about each question and write down your answers.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Preview of OpenShift Web Console

Web Console Project View



Description of numbers on the image above. These allow you to:

1. Project Selector - switch between projects
2. Search Bar - search through the catalog
3. Add to Project (top right) - create a new application or service
4. Top Right - view notifications
5. Overview Tab (current view) - see details of current project
6. Applications Tab - review and work with application components
7. Builds Tab - review and take action on project builds and images
8. Resources Tab - review resources and quota

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

9. Storage Tab - review and manage storage for your project
10. Monitoring Tab - review logs and notifications
11. Catalog Tab - go to the catalog.

Image and information obtained from OpenShift site at this link.
https://docs.OpenShift.com/online/architecture/infrastructure_components/web_console.html#architecture-infrastructure-components-web-console

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Meet the Command Line Interface (CLI)

Introduction to Command Line Interface (CLI)

- OpenShift offers a command line interface to perform any action you need for the deploying of your applications and services.
- All of the OpenShift related commands are prefaced with "oc".
- On Linux/Mac, you will use the terminal to type "oc" commands to interface with OpenShift.
- On Windows, you will need to open a command prompt "as an administrator" to allow you to run the "oc" commands.
- You can check the version of your oc.exe application by typing:

```
1 oc version
```

The oc client can be installed from the app store, available at either http://appstore, or at the following link:

http://appstore.uhc.com/AppList/AppSearch?searchText=OpenShift%20CLI&category=Java%20(OpenShift%20CLI)&OptumDev=1

OpenShift Lab: *Introduction to OpenShift Command Line Interface* helps to introduce some of the basic CLI commands.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Login and Logout using CLI

- Many of the "oc" commands require for you to be logged in to your OpenShift account by using the oc login command.
- To login to CLI access, you must first log in to the Web Console and copy the login link containing your session token information, then paste that information into the command prompt/terminal.
- Whenever you are done working with the CLI, you can log out by simply typing:

```
1 oc logout
```

In the next module there will be a lab that will help give specific instructions on how to login.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Discussion: OpenShift Development Aspects

Questions:

- What security practices are implemented in OpenShift?
- What does OpenShift add as a PAAS?

Instructions: Enter your answer into Chat (All Participants).

Take a moment to think about each question and write down your answers.

Using the Help Command to Get Help from CLI

- Using the Command Prompt or Terminal "oc" command you can get help by typing this command: `oc help` .
- The help command gives you a list of possible commands.
- If you want more help on a specific command:
  - Type the "oc" command and add `--help` .
- If you want help on the new-app command, for example, type: `oc new-app --help .`
- For additional info on the object you're looking at, type: `oc explain.`

When typing the help command, if you need, you can increase the size of the command prompt or terminal window so you can read it more easily.

You can use the --help flag to get help on any of the available commands.

Additionally, there's an oc explain command that gives you more insight on the specific object you're looking for.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Managing Projects with OpenShift CLI

- In OpenShift, a project allows you to create a separate workspace containing a group of applications and other objects.
- To check a list of available projects type `oc projects` .
- To check which project you are on, type `oc project` .
- Switching Projects:

  For example, if you have two projects: project1 and project2.

  If you are on project1, switch to project2 with this command: `oc project project2` .
- Creating a new project by typing: `oc new-project my-project` .

When typing the above commands, replace "my-project" with the name of your project.

Objects includes applications, services, resources, stored data, etc.

The OpenShift Online free starter plan allows you to have only one project a time.

If you have any existing projects, the "oc" command will assign you to one project at a time. All of your commands will automatically affect the project which you are working on. An oc new-project "project name" is only allowed if you are the cluster admin; Users will only have project admin rights.

OpenShift Lab: *Introduction to OpenShift Command Line Interface* will review the project related commands.

Learn more about projects at
https://docs.OpenShift.com/enterprise/3.2/admin_guide/managing_projects.html

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Lab: Getting Started with OpenShift Online

Overview:

Time: 20 Minutes

In this lab, you will:

- Install Git software.
- Sign up for OpenShift Online starter plan.
- Get familiar with the OpenShift website.

Instructions:

- Open the Student Lab Manual and follow the steps to perform the lab.

Please find the lab instructions in the student lab manual.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Lab: Installing a Sample App

Overview:

Time: 30 Minutes

In this lab, you will:

- Fork an app on the Github repository.
- Use Git to clone the application on your local system.
- Create a project using the OpenShift web console.
- Install the application on the OpenShift online server.
- Configure automatic builds for your application.

Instructions:

- Open the Student Lab Manual and follow the steps to perform the lab.

Please find the lab instructions in the student lab manual.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Module Summary

Now that you have completed this module, you should be able to:

- Describe the purpose and use cases for OpenShift
- Understand the OpenShift architecture
- Identify the features within OpenShift
- Identify an OpenShift web console
- Describe the Command Line Interface (CLI)

In addition to the objectives you identified in taking this module, these are the ones we expect to achieve based on the material provided.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift: Deploying and Managing Applications

Module 2

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Module Objectives

After this module, participants will be able to:

- Deploy an application in OpenShift
- Configure an application.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Deploying an Application in OpenShift

Options for Sources of Applications

When you want to create and deploy an application, OpenShift gives you several possible options for the possible sources of the application. You can do one of the following:

Upload the source code directly to OpenShift.

Use an existing template.

Load an application from an existing container file image.

Read more information about the options for creating an app at this link.
https://docs.OpenShift.com/online/dev_guide/application_lifecycle/new_app.html

Read more about templates at this link.
https://docs.OpenShift.com/enterprise/3.0/dev_guide/new_app.html#specifying-a-template

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

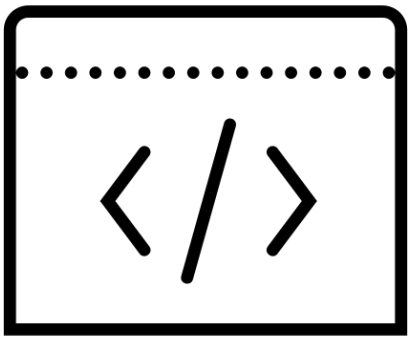
Languages and Resources Available in OpenShift

Included languages:

- PHP
- Python
- Perl
- Node.js
- Ruby
- Java

Supported Runtimes:

- Java
- .NET core



While these and the included features on the following slides are included out of the box, OpenShift is not limited to these. Many different languages and run time environments that can be setup in a docker image can be run on OpenShift.

For more information see these links. https://www.OpenShift.com/products/features/containers/ and https://docs.OpenShift.com/online/dev_guide/migrating_applications/support_guide.html

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

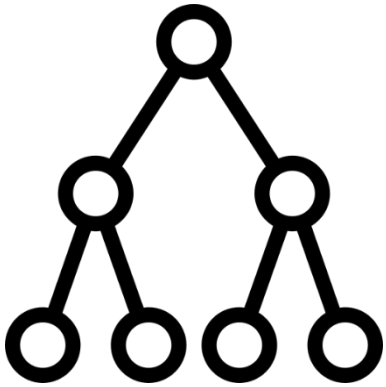
Included Frameworks and Databases

Included Frameworks:

- Jenkins

Included Databases:

- MySQL
- MongoDB
- MariaDB
- PostgreSQL
- Redis

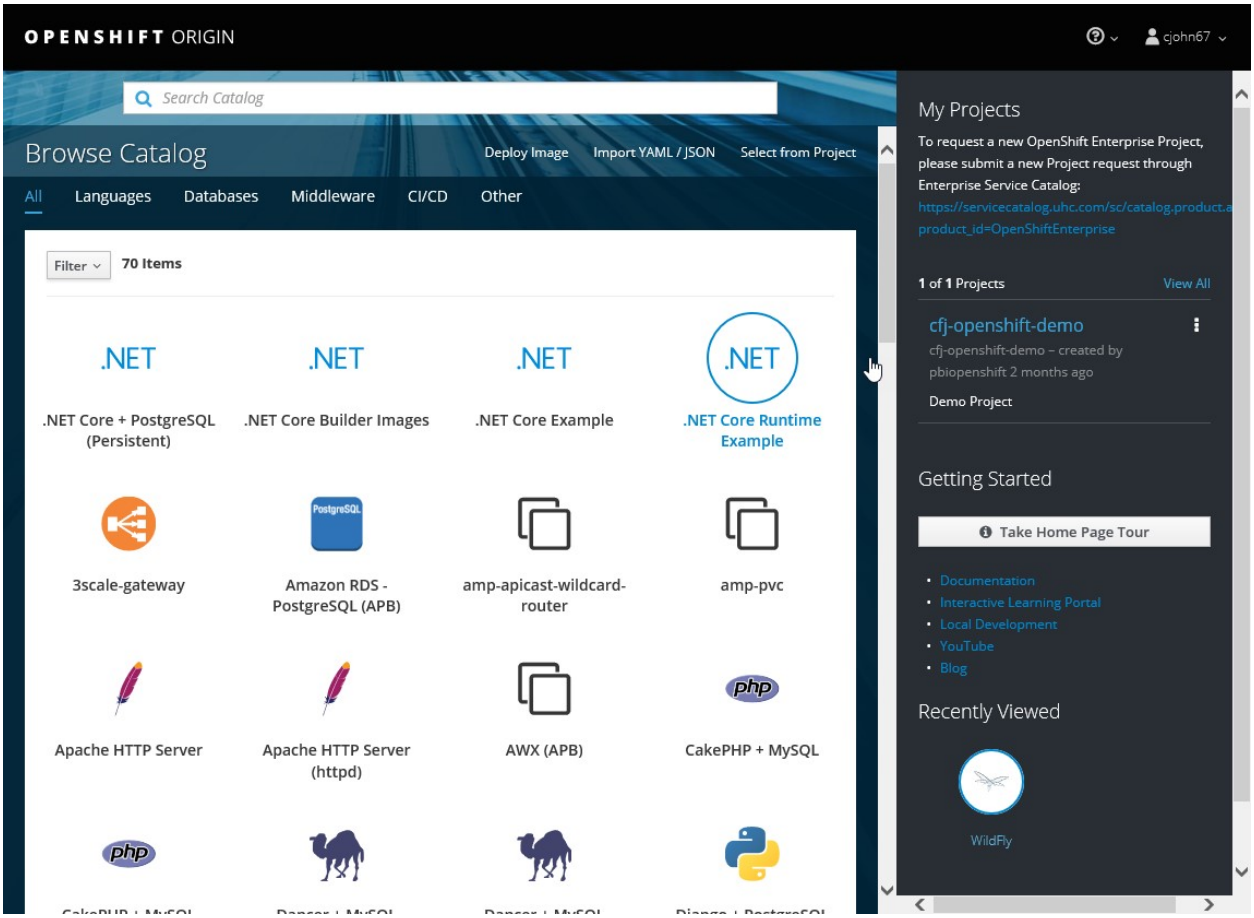


Read more about databases at this link.
https://docs.OpenShift.com/online/dev_guide/migrating_applications/database_applications.html#migrating-database-applications-supported-databases

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Creating an Application from Source Code with Web Console

Both the OpenShift Web Console and the OpenShift CLI have tools to automatically build your desired application using Source-to-Image (S2I).



You can add an application or another object to your OpenShift project by selecting the type of item in the catalog on the OpenShift Web Console. This will open up a window in which you will specify the configuration data and the repository URL or path to the source code.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Creating an Application from Source Code with CLI

The command line interface (CLI) has a command called "oc new-app", which allows you to create or upload the source code to your project.

You can specify a local repository for the source of your application with this command.

```
oc new-app /path/to/your/source/code
```

With oc new-app you can specify a Github repository URL instead. For example:

```
oc new-app https://github.com/username/hello-openshift-nodejs
```

You can optionally specify a name for all resources associated with the app by using the --name argument. This will apply the name nodejs-example.

```
oc new-app https://github.com/username/hello-openshift-nodejs --name nodejs-example
```

If using CLI you must first use the oc login command before you can use the oc new-app command. Instructions will be provided in *Lab: Introduction to OpenShift Command Line Interface*.

The oc new-app command automatically detects the language of the source file.

For the oc new-app commands instead of /path/to/your/source/code you will put your actual local path of the source code.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Pipeline Builds

Instead of using a Source-to-Image (S2I) method of building your application, there is the option of using a Jenkins Pipeline to build of your app.

A Jenkins Pipeline is a set of plugins which allows you to automatically update to the server any changes you make to the source code of your applications

The Pipeline workflow information needs to be documented in a Jenkins file which will be either in the Build config file for your app, or in a separate file which is referenced to in the Build config file.

You can build either with pipelines or with Jenkins.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Interactivity: True or False?

Question: True or False. Use the oc new-app command to automatically detect the language of the source file.

Instructions: Use ✓ (True) or **X** (False)

Automatic Detection

- When you start to create a new app, OpenShift automatically detects the type of application and it also searches for a Jenkinsfile. If it finds it will automatically select the Pipeline Build strategy.
- You can manually override this using the CLI with the oc new-app command by using a strategy flag to set it to either a Pipeline build, or S2I build using the following examples.

```
1 oc new-app https://github.com/username/example --strategy=source
```

Or

```
1 oc new-app https://github.com/username/example --strategy=pipeline
```

OpenShift searches for a Jenkins file in the root directory of the source repository. If it finds it, OpenShift will automatically select the Pipeline Build Strategy. If not, it will automatically select the Source-to-Image (S2I) strategy.

Also, you can use oc rollout latest dc/<name> -n <project>

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Create an Application Using a Template

- You can use a template file for your desired type of application, database, or resource to create a new application.
- The Web Console has a catalog with different templates in it to start with by clicking on the object in the catalog you want to add.
- If using the CLI, you can specify a local template using `-f` before the template.
- Using `oc new-app` with a local template. Replace the /path/to/nodejs.json with the local path and name of the template file.

```
1 oc new-app -f /path/to/nodejs.json
```

The process of application creation starts with a template. You can do the creation either from a Web Console or from a command line, using the command line tools (CLI) which we described earlier. The CLI was installed in one of the previous labs.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Create an Application Using a Container Image

- Creating an application using a container image is a similar process to using `the oc new-app` command with a template.
- Direct the `oc new-app` command to the container image file instead of the template.
- Specify the name or url or path of the container image you want to add. You can specify the Dockerhub name as in these examples.

```
1 oc new-app mysql
```

Or

```
1 oc new-app centos/mongodb-26-centos7
```

Another way to build an application is by using a container image. You can do it with the CLI by providing the URL of the Docker image. Optum uses an internal Docker registry "Docker Trusted Registry" at docker.optum.com.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Defining Environmental Variables

- To specify values for the parameters in your template file when creating an app, you will use the -p flag like this example:

```
1 oc new-app -f /path/to/nodejs-mongodb.json -p MONGODB_USER=admin
```

- If you want to set environmental variables when loading a container image, use the -e flag, like this example:

```
1 oc new-app centos/mongodb-26-centos7 -e MONGODB_USER=admin
```

The environmental variables control the building of your application. It is also a best practice for security, and this ties back to the 12 Factors we discussed in the Essentials course. Shown here are some examples of setting these environmental variables. Optum has an artifact repository that an environment variable commonly needs to be set for builds:

--build-env MAVEN_MIRROR_URL=http://repo1.uhc.com/artifactory/repo

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Checking the Status

- You can easily check the status of your applications in the Web Console under the project section.
- Using CLI, you can request the status of your applications by simply typing: `oc status` .
- Add the `-v flag` to get more detailed information: `oc status -v .`
- To list services for your app type: `oc` get services .
- For a complete list of everything under your project, type: `oc get all` .

The current status of the application can be checked by either using the dashboard or the command line (CLI). There are many aspects of the current application that you can get with checking the status, found at this link (https://docs.OpenShift.com/online/getting_started/beyond_the_basics.html#getting-started-beyond-the-basics).

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Configuring an Application

Defining Environmental Variables

- After you have imported the source, template, or image, you can define environmental variables using the "oc set env" command like this example:

```
1 oc set env dc/registry STORAGE_DIR=/local
```

After you have imported the source, template, or image, you can define environmental variables using the "oc set env" command, shown here.

Configuring Routes

In order for your application to be accessible from a web browser, you will need to configure a route for the application.

How to configure a route:

- Get the name of your services by typing: `oc get services` .
- Next, type the "`oc expose`" command for your service as in this example:

```
1 oc expose service/hello-openshift-nodejs
```

Find the route by typing: `oc get routes` .

This will give you a web address, copy the web address and paste it in your browser address bar and hit enter to load the page.

Naturally, you want your application to be used, and for that you need to make it accessible from the internet. This is accomplished by exposing the services you provide through a route. The above command shows how to do this. Recall that we introduced port bindings in the 12 Factor Methodology earlier, and this is another instance of it staying in our minds as we develop.

This will generate a URL which you can test.

There are more details for exposing with friendly name in the "OpenShift Application Access Guide" available at: https://www.optumdeveloper.com/content/odv-optumdev/optum-developer/en/developer-centers/hosting/hosting-openshift/internet-access-guide.html

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Interactivity: Poll Question

Question: To make your application Internet accessible, you need to expose your services. This is accomplished via what?

A. Template
B. Image
C. Route
D. Container

Automatic Builds

OpenShift allows you to automatically rebuild the application on the server:

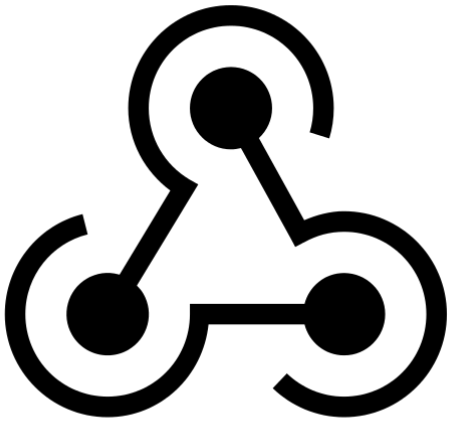
- By a Jenkins Pipeline
- By using a webhook.

Jenkins is a CI (Constant Integration) tool. You can use it for deployment as well. In a Jenkins pipeline, you will set up a rebuild operation. Alternatively, a webhook will cause the script to be executed every time that a given URL is hit.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Webhooks

Webhooks are a type of trigger which cause your application on OpenShift to be automatically rebuilt under certain specified conditions, such as using "git push" to update the repository of your application.



The simplest way to use a webhook involves copying the webhook link from the OpenShift web console, and pasting it on your GitHub website account. Once this is successfully set up, changes to the application on the server will be automatically updated when you push your changes to the GitHub repository.

OpenShift Lab: *Installing a Sample App*, Step 4 describes the process of applying the webhook to your GitHub account.

It is also possible to get the webhook URL and secret using the command line interface.

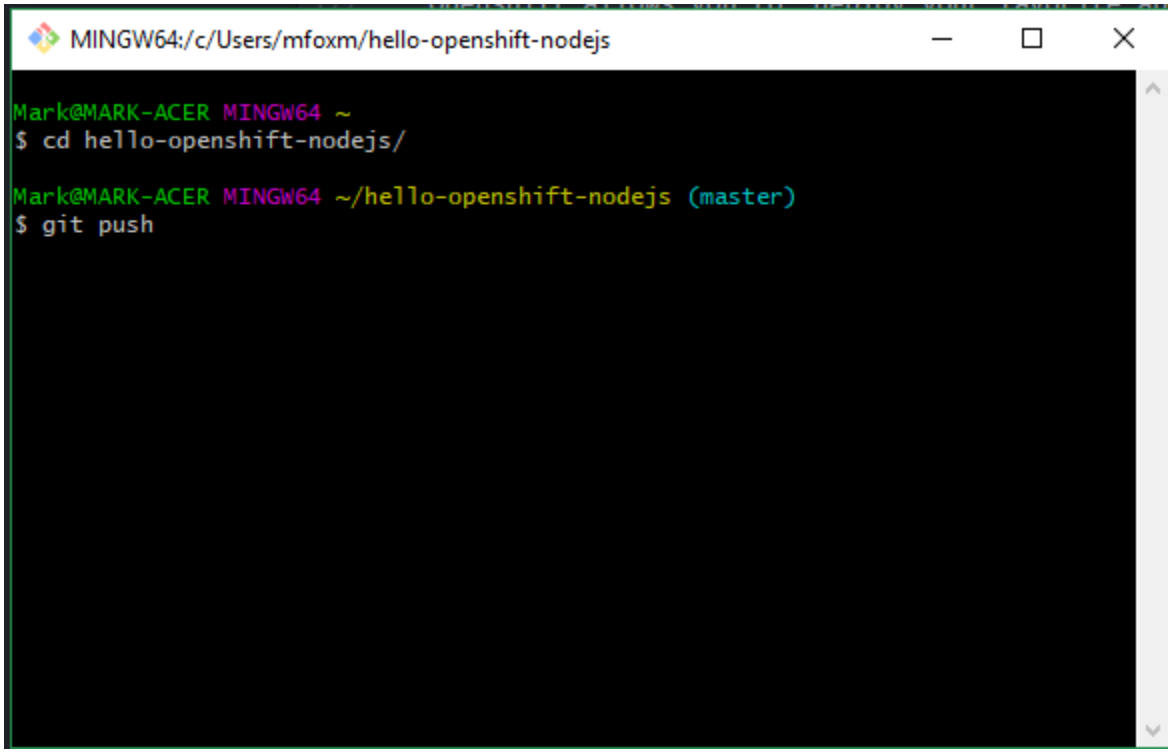
You can also use other repositories besides GitHub if you prefer.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Git Push

You can make changes to your application on your local computer and then use your "`Git Bash`" command line or terminal to "`git push`" to update your changes on the repository.

If you have automatic builds configured, this will automatically update the application on the server.



The ‘push’ operation on GitHub is registered by GitHub and can cause an action. One of these actions is a ‘webhook’ for anyone who is listening. The service is provided for all levels of accounts in GitHub including the free level.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

BuildConfig Files

A BuildConfig is a file which is created automatically when you create a new app on the Web Console or using the "`oc new-app`" command.

A BuildConfig contains configuration settings for creating a new build and contains information about triggers for rebuilding the app.

BuildConfig is a template for creating new applications. You can change but for starters you will probably use the default.

About BuildConfig Files

The following types of triggers can be used to rebuild the app:

- A Github or generic webhook
- Image change trigger
- Configuration change triggers

A Buildconfig file is generally in .yaml or .json formats and can be edited with the CLI and a text editor.

A Buildconfig file can also be edited directly on the Web Console in the Builds section.

An image change trigger will automatically rebuild the app when a new image is available.

A Configuration change trigger will automatically rebuild the app when a configuration file, such as BuildConfig file is updated.

OpenShift Lab: *Using Command Line Interface to Automatically Rebuild Your Application* will walk through the process of editing BuildConfig and DeploymentConfig files.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

DeploymentConfig Files

A DeploymentConfig file is automatically created when you create a new application using the Web Console or the "`oc new-app`" command.

The DeploymentConfig file specifies details about the:

- Pods
- Replication controllers
- Triggers
- Health checks

The DeploymentConfig file can be edited directly in the Web Console under Applications or through the CLI and a text editor.

OpenShift Lab: *Using Command Line Interface to Automatically Rebuild Your Application* will walk through the process of editing BuildConfig and DeploymentConfig files.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Persistent Volume and Persistent Volume Claims

Persistent Volumes (PV) are available storage resources on the cluster.

Persistent Volume Claims (PVC) request a specific amount of data to be used as storage and once it is setup, it safeguards that PV storage space to be only used for the applications which have requested it.

If your application requires stored data, such as a database, then the PVC will provision storage space for the application.

A PersistentVolumeClaim file is a configuration file with details about the configuration of the PVC. It can be edited using CLI and a text editor.

OpenShift Lab: *Using Web Console and Command Line Interface to View and Edit Configuration Files* will walk through the process of creating a PVC and editing the configuration file.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Mounting a PVC

Configuring a pod to use a persistent volume requires the following steps:

- Create a PVC
- Create a pod
- Set up access control

In the Optum environment, the application team only needs to create the PVC and mount it. Mounting will usually be done in the deploymentConfig configuration, rather than to a pod.

Each of the above steps requires editing the configuration files and running Linux commands. For the exact description, refer to the following instruction:

https://kubernetes.io/docs/tasks/configure-pod-container/configure-persistent-volume-storage/

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Interactivity: Matching Question

Question: Match each step in the process of mounting a PVC with its step number.

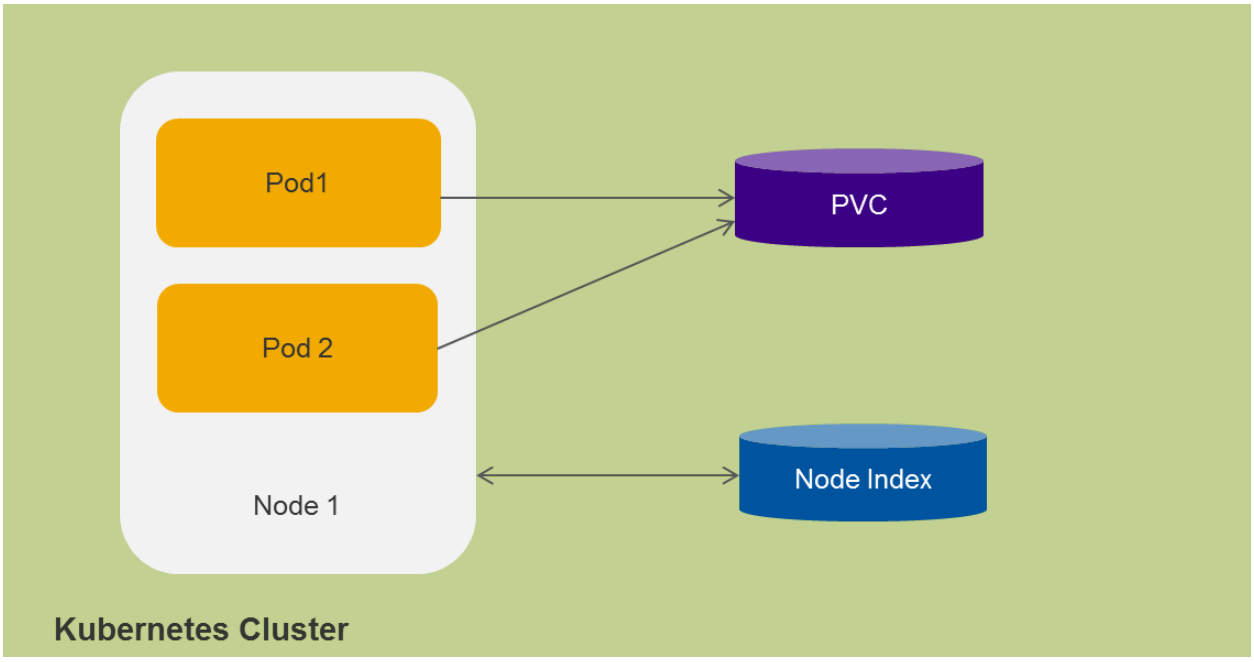
Instructions: Raise Hand to volunteer and then use the Line tool.

Step 1	Create a PVC.
Step 2	Set up access control.
Step 3	Create a pod.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

PVC with Multiple Pods

Multiple pods are able to connect to the same PVC



This diagram shows that multiple pods in one server are able to connect to the same PVC.

Secrets and Creating a Secret

- A secret file is an object which contains sensitive data such as a user id and password for services.
- You can generate a secret by creating a yaml text file on your local computer in the secret format like the one below.

```yaml
apiVersion: v1
kind: Secret
metadata:
  name: mysecret
type: Opaque
data:
  username: dXNlci1uYW1l
  password: cGFzc3dvcmQ=
```

- If you named the file secret.yaml, you can use command line to create a secret file:
`oc create -f secret.yaml .`

Once you have created a secret file, the pods can refer to that secret in the pod files. You can also easily create a secret file through the Web Console under Resources/Secrets.

In order to inspect the information contained in a secret, one can use the following commands:

- docker secret ls (list secrets)
- docker secret inspect (list detailed information on one or more secrets)

Secrets are encrypted during transit and at rest. For secret utilization, you grant privileges to services. A given secret is only accessible to those services that have been granted explicit access to it, and only while those service tasks are running.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Health Checks

Health Checks are a way to ensure that the applications are operating correctly.

There are two types of health checks:

- A readiness probe checks if the application is able to handle requests.

  The application needs a route for the readiness probe to check it.

  If the readiness probe fails, the pod is removed as an available endpoint for the service.

  If the pod later passes the readiness probe check, it will be added back into rotation for the service. A liveness probe checks if a container is still running.

- If the liveness probe fails, it will automatically kill the container.

Health Checks are a must, and form the first layer of monitoring. The existing Health Checks are listed here. You can create more and you can combine them within your overall monitoring. For that, you can use Nagios.

A best practice is to always configure these using checks specific to the application being deployed. All three types (port check, URL, script) are available.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Configuring Health Checks

Health checks can be configured under the Web Console under deployment, or in CLI. For example:

```
oc set probe --readiness --get-url=http://my.website.com:8080
```

Set initial delay for health checks using the "`oc set probe`" command as in this example:

```
oc set probe dc --all --readiness --initial-delay-seconds=30
```

You can also add or edit health checks by editing the DeploymentConfig file.

This can also be done through the Web UI and through the OpenShift API. Using the oc command with –loglevel=9 will turn on verbose logging that includes showing the API calls that are made.

For more information, see the API documentation at https://docs.openshift.com/container-platform/3.7/rest_api/index.html

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Discussion: OpenShift Complex Deployments

Questions:

- What are examples of complex deployments?
- Name some of the best practices for autoscaling.
- What are health checks?

Instructions: Enter your answer into Chat (All Participants).

Take a moment to think about each question and write down your answers.

Lab: Introduction to OpenShift Command Line Interface (CLI)

Overview:

Time: 20 Minutes

In this lab, you will:

- Log in to OpenShift using CLI.
- Delete existing project or projects using the CLI.
- Create a project using the CLI.
- Deploy an example application from source code.
- Configure a route for your application.
- Log out of OpenShift CLI.

Instructions:

- Open the Student Lab Manual and follow the steps to perform the lab.

Please find the lab instructions in the file student lab manual.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Lab: Using CLI to Automatically Rebuild Your Application

Overview:

Time: 30 Minutes

In this lab, you will:

- Clone an application from the Github repository.
- Log in to OpenShift using the CLI.
- Delete existing webhooks from your Github account.
- Obtain a webhook URL using the CLI.
- Add a webhook using the URL you have obtained.
- Edit the source code of your application.
- Deploy an example app from source code.
- Use "git push" to update changes.

Instructions:

- Open the Student Lab Manual and follow the steps to perform the lab.

Please find the lab instructions in the student lab manual.

Lab: Using Web Console and CLI to View and Edit Configuration Files

Overview:

Time: 20 Minutes

In this lab, you will:

- Learn how to view and edit configuration files using the web console.
- Learn how to configure health checks using the web console.
- Learn how to view and edit configuration files using the CLI.
- Learn how to configure health checks using CLI.
- Learn how to create and edit secret files.
- Learn how to create a PersistentVolumeClaim (PVC) and edit the PVC object definition file.
- Learn how to create pods and edit pod object definition files.

Instructions:

- Open the Student Lab Manual and follow the steps to perform the lab.

Please find the lab instructions in the student lab manual.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Module Summary

Now that you have completed this module, you should be able to:

- Deploy an application in OpenShift.
- Configure an application.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

This page intentionally left blank.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift: Creating Complex Deployments

Module 3

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Module Objectives

After this module, participants will be able to:

- Use the best practices of scaling applications.

In addition to the objectives you identified in taking this module, these are the ones we expect to achieve based on the material provided.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Scaling an Application

Why Scale an Application?

Multiple copies of authentication components	For a web service, it is necessary to provide multiple copies of the application components so more than one user can use the application simultaneously.
Increase or decrease in replicas of pods and resources	Scaling an application means to increase or decrease the replicas of the pods and resources based on the needed number.
Automatic scaling	OpenShift has features built-in which help you to automatically scale the application.

Now, let's review some of the components of the architecture used in OpenShift. There are three basic design patterns.

1. Multiple copies of the application
2. Scale the resources up and down
3. Make scaling automatic

Pods, Replicas, and Services

Pods	Replicas	Services
A pod is a Kubernetes subunit that consists of: • One or more containers. • Shared network and storage. • A single IP address and port space. • Specifications of how to run the containers.	A replica is multiple instances of the same type of pod.	Although pods can be either running or not running, a service is always on and available. A service is a representation of all the replicas of a particular application or function. A service defines the operation of the pods and sets policies of when to access the pods. Each service is accessed by a single IP address.

OpenShift builds on top of Kubernetes, so reviewing the terminology is important. Above we have defined the three levels of abstraction. The Pod is a collection of containers. Replica are multiple pods. Finally, services are the result of building your app and opening it through a specific URL.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Interactivity: Matching Question

Question: Match each OpenShift component with its description.

Instructions: Raise Hand to volunteer and then use the Line tool.

Replica	One or more containers
Service	Representation of all the replicas of a particular application or function
Pod	Multiple instances of the same type of pod

Routes and Load Balancers

- A route provides an accessible web address to a particular service, so that it can be reached through the network or Internet.
- OpenShift has a software load balancer that is built-in to the routing layer of the OpenShift architecture. It is the load balancer that maps out, provides, and controls the routes to the services.
- The load balancer is a key component that also helps to automatically scale your applications.

A load balancer implements the “Decoupling” architectural design pattern. It also serves for scaling the application under increasing load.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Replication Controllers

- A replication controller (RC) is a supervisor for long-running pods.
- A replication controller will automatically increase or control the number of replicas of the application based on the need.
- DeploymentConfig files contain the details of the replication controllers and number of replicas for the deployment.
- You can check for a list of replication controllers using the command below.

```
1 oc get replicationcontrollers
```

A replication controller (RC) works together with the load balancer for application scaling. It is not a pod.

Stateless and Stateful Applications

Stateless Applications

- An application that is "stateless" is one that does not need to store data or access stored data other than the application itself.
- If you are deploying only stateless applications, much of the scaling process of the application is automatic.

Stateful Applications

- A "stateful" application stores and accesses data as part of its operations. Any application which utilizes a database would be a stateful application.
- If you deploy stateful applications then you will need to also provide persistent storage which is able to scale up and down with the usage of the application.

You are given the freedom of choice to create either stateful or stateless applications. If you choose a stateful application, you should keep in mind the following:

1. Stateless applications are easier.
2. The best design pattern is to store the state in a NoSQL database, for scalability and speed of access.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

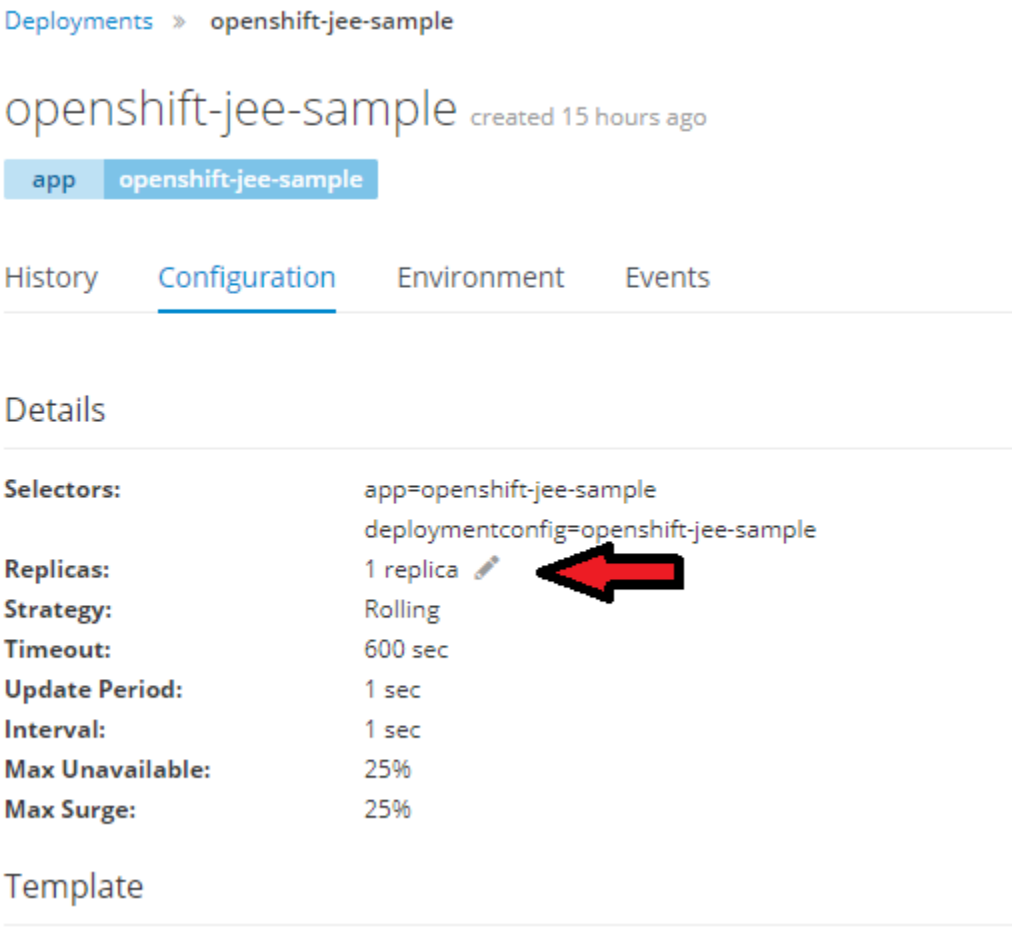
Scaling an Application by Editing the Deployment Configuration

The deployment configuration contains details of the replication controllers. You can manually scale an application by simply editing the deployment configuration file.

In the example pictured here, you would change the number of replicas by changing the number "1" on line 16 to the desired number and save and update the DeploymentConfig file.

Changing the Number of Replicas with the Web Console

You can change the number of replicas using the web console.



When you are in the Web Console managing your project, you can change the number of replicas:

Click **Applications**, then click **Deployments**.

Click the name of the deployment you want to change.

Click the **Configuration** tab.

Then, you can click the **pencil** next to the "1 replica" and put in the number of replicas you would like.

Keep in mind that the actual number of replicas which will be running depend on the amount of the server resources available and on the need.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Changing the Number of Replicas

How to change the number of replicas:

- You can scale an application up and down manually by simply changing the number of replicas in the DeploymentConfig file.
- You can also scale an application by changing the number of replicas the deployment configuration settings section on the web console.
- You can use the `oc scale` command in the CLI as in this example:

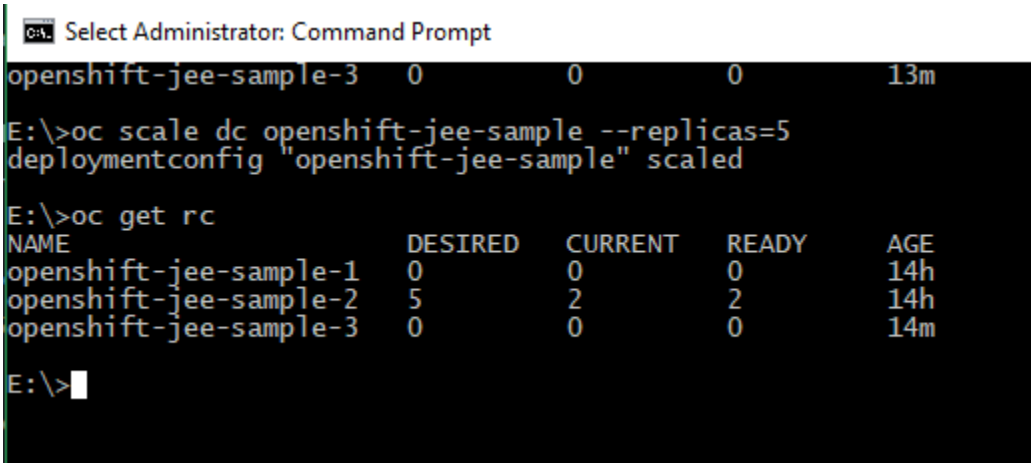
```
1 oc scale dc my-example-app --replicas=3
```

Keep in mind that the actual number of replicas which will be running depend on the amount of the server resources available and on the need.

How to Get Information about the Replication Controllers using the CLI

Using CLI, you can refer to the replication controllers by "`rc`."

Type the command: `oc get rc` to see the names of the replication controllers and the number of replicas running.



You can view or save a copy of the replication controller configuration files in the same way as other configuration files, using the "oc edit rc" and "oc get rc" commands.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Stateful Sets

- Deployment manages Pods that are created from the same container spec.
- By contrast, a StatefulSet also manages Pods created from the same container spec but each Pod has a unique identity.
- Pods in a StatefulSet are not interchangeable; each has its own unique identifier.
    - This identifier persists across any rescheduling.

- In summary
    - StatefulSet is the workload API object used to manage stateful applications.

The details on managing StatefulSets can be found here,
https://kubernetes.io/docs/concepts/workloads/controllers/statefulset/

The simplest way to install a database with a stateful set is to create it using a template designed for a stateful set database.

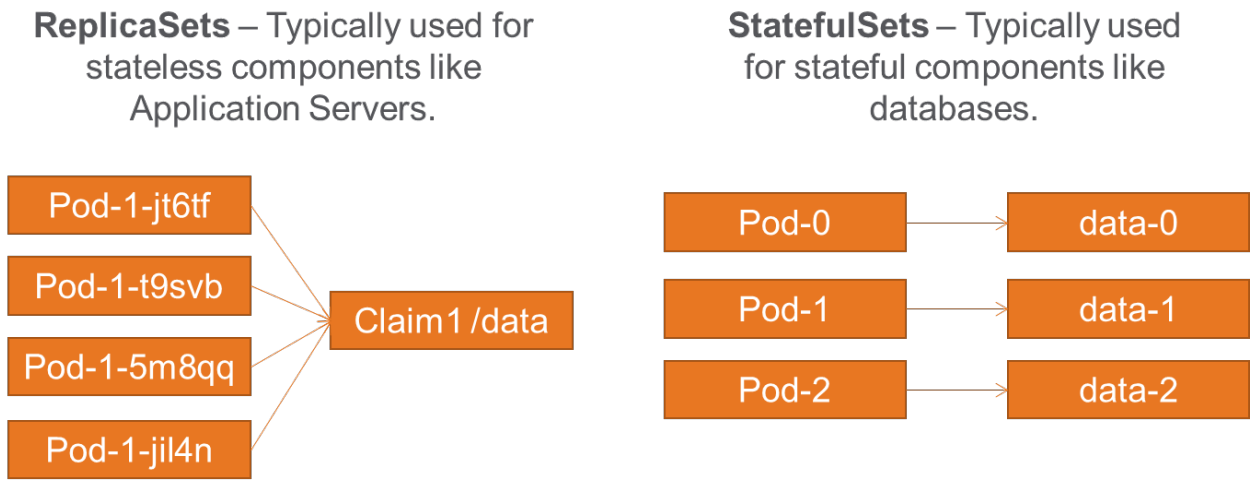
You can use the "oc scale" command to scale the database, but you need to specify "statefulsets" in the command as in this example:

"oc scale statefulsets mongodb --replicas=5"

If you create a stateful set of a MongoDB database, it will create multiple replicas of the MongoDB pods.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Sets and PVCs



Pods in replica sets have random names. ReplicaSets can optionally have a PVC mounted to the pods. If done, all pod instances will mount the same PVC and have shared access to the data.

Pods in a StateFul set have a well-known name. StatefulSets have a VolumeClaimTemplate to create an individual PVC specific to each pod.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

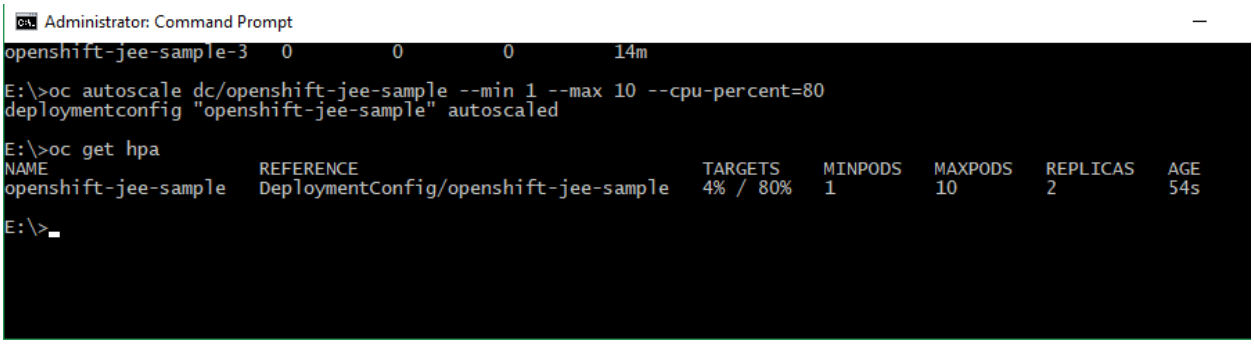
Autoscaling

OpenShift allows you to automatically scale the application by specifying the:

- Minimum number of pods
- Maximum number of pods
- Maximum percentage of server CPU usage that you would like to use.

You can easily set up autoscaling in the deployment section of the web console.

Alternatively, use the `oc autoscale` command as shown in the following example:



Use the "oc get hpa" command to check the status of the autoscaling function. HPA stands for "horizontal pod autoscaler."

Autoscaling Best Practices

Here are the best practices gleaned from various autoscaling systems

- Identify the metrics that affect the performance
  - It may be CPU, but it may also be RAM, or some application metric
  - Base your autoscaling on these metrics
- Scale up fast, scale down slow
  - Usually, you need to scale up fast because you want the performance to be satisfactory
  - However, scaling down should not be rushed: it may be that the demand will increase while we are scaling down
  - This behavior is known as “thrashing”
- However, there are exceptions to the above rule
  - Usually you measure CPU
  - But you may measure the size of a certain queue
  - These two measurements might be in conflict

Autoscaling is a standard cloud question and every cloud provider has their answer. You can compare Kubernetes policies and best practices with those of Amazon auto scaling groups.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Clustering an Application

A cluster is defined as a group of servers connected together which share their resources to act as one functional unit.

Your cluster in OpenShift would be the largest functional unit consisting of a Master node, other nodes, storage, as well as routing layer.

Clustering is when a group of applications or servers need to interact with a single database. When there are many users and many replicas of a pod, this needs to be done efficiently.

Clustering applications are today the best practice of scalability. That is, instead of buying a bigger computer, you buy more of the smaller, less expensive servers. Thus, your price for clustering goes linearly as opposed to quadratically or exponentially with the hardware-only approach.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Discussion: Review of the Major Concepts

Questions:

- What is the purpose of scaling application?
- What is a pod and what is a replica?
- How is stateful application different from stateless in terms of pods?

Instructions: Enter your answer into Chat (All Participants).

Take a moment to think about each question and write down your answers.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Configuring Pods and Pod Scheduling

Configuring Pods

In OpenShift, the pods have object definition files that can be edited and modified to configure the functioning of the pod.
The pod object definition file contains various settings and specifications including:

- Network Information such as the IP address and ports
- Resources allocated to the pod
- Names of containers
- PVC and volumes allocated to the pod
- Replication controller specifics
- Name of the pod scheduler policy
- Pod restart policy.

Pod Scheduling

Policies for the operation of pods are specified in a default pod schedule file which is automatically deployed.

Each pod object definition file will refer to the name of the scheduler being used to determine these policies.

You can create custom .yaml file which has different policies and priorities for the operation of the pods. This is called a custom scheduler.

OpenShift Lab: *Using Web Console and Command Line Interface to View and Edit Configuration Files* describes how to view and edit the pod object definition files.

OpenShift online documentation has the text for the default scheduler on the website at the link below. The site explains each parameter in the file. You can copy the default configuration and paste it in a new yaml file with a text editor. Pod configuration is most commonly going to be done in the deployment configuration, not in a pod specific configuration file.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Pod Restart Policy

The pod object definition file specifies the pod restart policy. By simply editing the pod object definition file, you can configure each pod to restart based on three options.

- Always - continuously tries to restart a container in a pod which has successfully exited.
- On Failure - tries to restart a container in a pod which had failed to start for up to 5 minutes.
- Never - does not try to restart a container on a pod whether it exited successfully or failed to start.

The decision of how to restart a pod upon failure depends on the maturity of the application. If your application is not yet mature, failure is likely due to the software imperfections. In this case, restarting the application does not help, and your pod restart policy should be ‘never.’

By contrast, if it is a mature and debugged application which is usually running, you can use the ‘Always’ policy because the failure can be attributed to other components and are likely not to repeat.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Routes in OpenShift

Routes

OpenShift has a default, built in router which automatically handles the routing requests with a few basic commands or with the web console.

Routine protocols are limited to HTTP and HTTPS. Services using other protocols cannot be exposed via a route.

Use the `oc expose` command to assign an accessible route to a service.

The `oc expose` command lets you add arguments to specify:

- A custom address as a route
- Ports
- Protocol
- and more

The `oc get routes` command will list the routes already assigned.

The web console allows you to create and edit routes under the Applications, Routes tab.

For help on the "oc expose" command, type "oc expose --help" into the CLI.

Custom routes are not allowed in OpenShift Online starter plan, but they are allowed in the OpenShift Online Pro plan.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Discussion: Applying Kubernetes and OpenShift

Questions:

- What problems are solved in your organization through Kubernetes?
- What does OpenShift add in your practice to the Kubernetes-based solutions?

Instructions: Enter your answer into Chat (All Participants).

Take a moment to think about each question and write down your answers.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Lab: Working with Complex Deployments

Overview:

Time: 20 Minutes

In this lab, you will:

- Scale an application.
- Autoscale an application.
- Practice creating pods with an affinity rule.
- Cluster an application.
- Create a route in OpenShift.

Instructions:

- Open the Student Lab Manual and follow the steps to perform the lab.

Please find the lab instructions in the file student lab manual.

Lab: Working with Application Clustering

Overview:

Time: 20 Minutes

In this lab, you will:

- Install an example application which uses application clustering.

Instructions:

- Open the Student Lab Manual and follow the steps to perform the lab.

Please find the lab instructions in the file student lab manual.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Module Summary

Now that you have completed this module, you should be able to:

- Use the best practices of scaling applications.

In addition to the objectives you identified in taking this module, these are the ones we expect to achieve based on the material provided.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

This page intentionally left blank.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift: Troubleshooting Applications

Module 4

Module Objectives

After this module, participants will be able to:

- Recognize the resources available for support.
- Debug an application with OpenShift.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Resources Available for Support

OpenShift Technical Support Available offered by Red Hat®

- Optum has a support plan with RedHat
- Internal support is available by opening a ticket with the OpenShift group
- An OpenShift User Group can be accessed through HubConnect Community
- To locate information and discuss questions or problems, utilize Flowdock OpenShift
- OpenShift Container Platform offers vendor support
- OpenShift Origin (OKD) does not have vendor support
- Red Hat® also offers training for OpenShift Container Platform customers

Multiple levels of OpenShift support ensure that you can get the level that fits your needs exactly.

Optum has a support plan with RedHat; however, internal support is available by opening a ticket with the OpenShift group. On HubConnect Community (https://hubconnect.uhg.com/groups/openshift-enterprise-users-group), there is an OpenShift User Group. With Flowdock, an incident ticket to the OpenShift team can also be utilized to locate information and to discuss questions and problems.

OpenShift Origin (OKD) does not have vendor support; however, OpenShift Container Platform does offer vendor support.

For OCP, there is an option to get a Red Hat account and open a ticket. For information on doing this on OptumDeveloper, this link walks developers through how OpenShift is supported, between the commercial and open source offerings. It also provides information on the local user community:

https://www.optumdeveloper.com/content/odv-optumdev/optum-developer/en/developer-centers/hosting/hosting-openshift/support.html

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift has Extensive Documentation Available Online

- OpenShift has some tutorials to get you started including the "Basic Walkthrough," and the "Beyond the Basics" tutorials.
- You can easily search the online documentation on the OpenShift website.
- OpenShift has a blog on its website where various authors write tutorials, access to example applications, and other helpful information.

OpenShift provides multiple tutorials that make sure that the system is used by as wide an audience of developers as possible. The blogs on the OpenShift website are a good way to keep current in the latest and greatest developments.

For more, see the OptumDeveloper OpenShift getting started tutorial:

https://www.optumdeveloper.com/content/odv-optumdev/optum-developer/en/getting-started/hosting-getting-started/getting-started-openshift.html

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

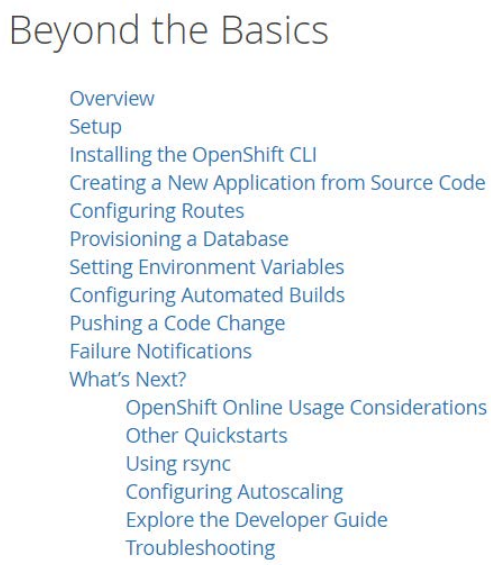
OpenShift Documentation has Many Helpful Tutorials and Links

The "Beyond the Basics" tutorial contains many helpful links including the following:

- Tutorials and templates for applications using: Ruby, Python, Node.js, PHP, PERL, and Java.
- Links to the GitHub repository for images and templates.

Other tutorials include:

- QuickStart templates with repository links
- Ruby on Rails
- Setting up a Nexus Mirror
- OpenShift Pipeline builds
- Binary builds

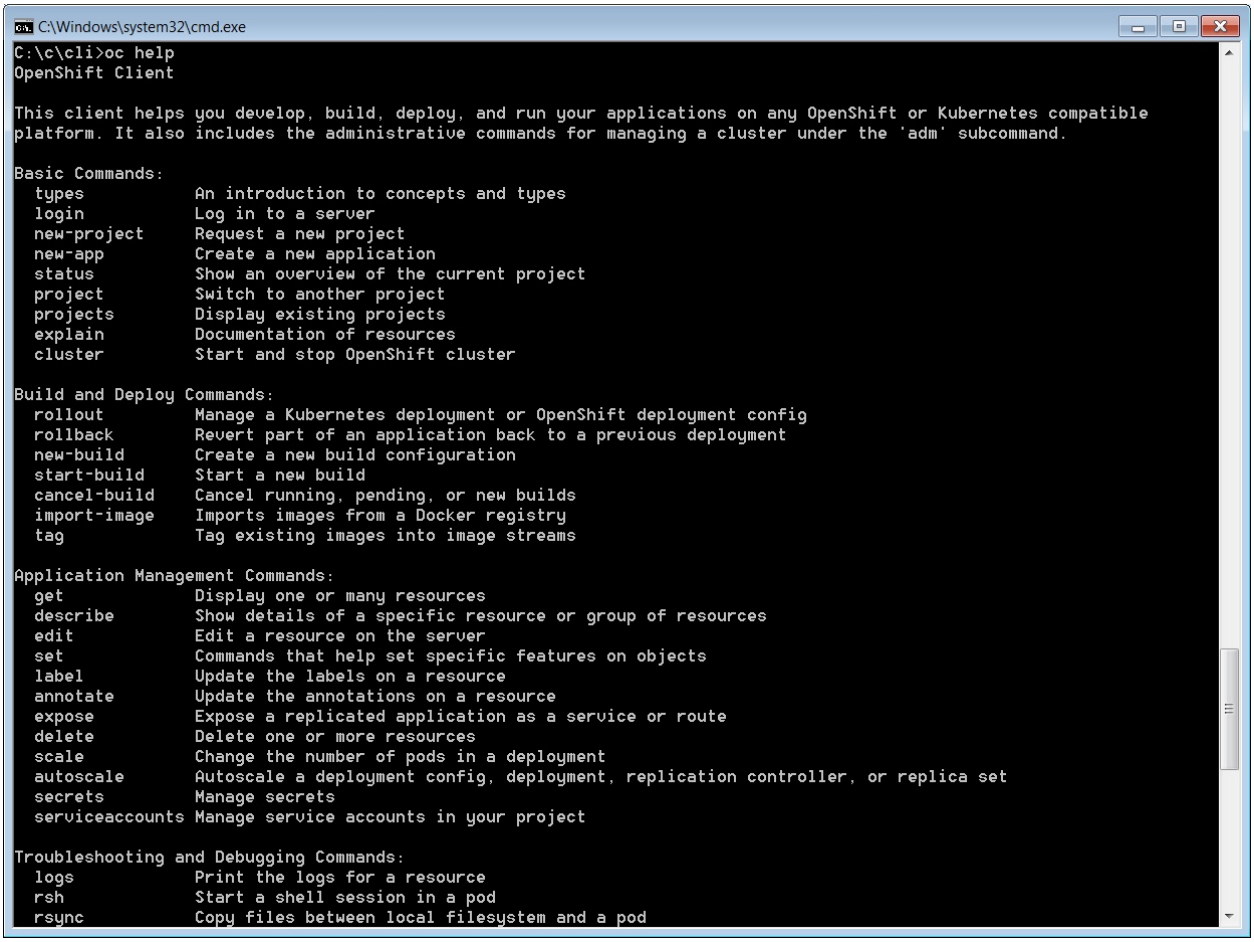


The above tutorials explain how to use the template in multiple different languages, such as Ruby, Python, Node.js, and even PHP and Perl. Furthermore, you can use OpenShift with Ruby on Rails.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift CLI Reference Documentation

The `oc --help` command and specific help for each command is built-in to the oc.exe command.



In your work, you will often need to reference the manual for CLI which is found here:

https://docs.OpenShift.com/online/cli_reference/index.html.

There you will find the instructions for installing the CLI in the first place, on Windows, Mac, and Linux. You will also find the configuration after the installation.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

OC Rollout Command Cheat Sheet

Here is a list of Build/Deploy oc commands.

`oc rollout`	Manage deployments from deployment configuration
`oc rollout latest`	Start a new deployment with the latest state
`oc rollout undo`	Perform a rollback operation
`oc rollout history`	View historical information for a deployment configuration
`oc rollout status`	Watch the status of a rollout until complete
`oc tag`	Tag existing images into image streams
`oc start-build`	Start a new build from a build configuration
`oc cancel-build`	Cancel a build in progress
`oc import-image`	Pull in images and tags from an external Docker registry
`oc scale`	Change the number of pod replicas for a deployment

There are a variety of OpenShift cheat sheets, select one that fits your style. Here is an example of an OpenShift cheat sheet that has additional oc commands:

http://design.jboss.org/redhatdeveloper/marketing/openshift_cheatsheet/cheatsheet/images/openshift_cheat_sheet_r3v1.pdf

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

OC Explain Command Cheat Sheet

Here is a list of `oc explain` commands.

`oc explain pod`	Explain the construction of the pod
`oc explain pod.spec.affinity`	Affinity is a group of affinity scheduling rules

These are just a few commonly used oc explain commands, it is considered the ground truth regarding pod construction.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

OC Get Command Cheat Sheet

Here is a list of `oc get` commands.

`Oc get -f`	File to get
`Oc get -L`	Accept comma-separated list of labels
`Oc get -o`	Specify output format
`Oc get -a`	Show all resources
`Oc get sort-by`	Sort by resource name
`Oc get -r`	Recursive. Useful when you want to manage related manifests organized within the same directory.

These are several commonly used oc get commands.

Interactivity: Poll Question

Question: Which command would you use to change the number of pod replicas for a deployment?

A. `oc explain pod`

B. `oc scale`

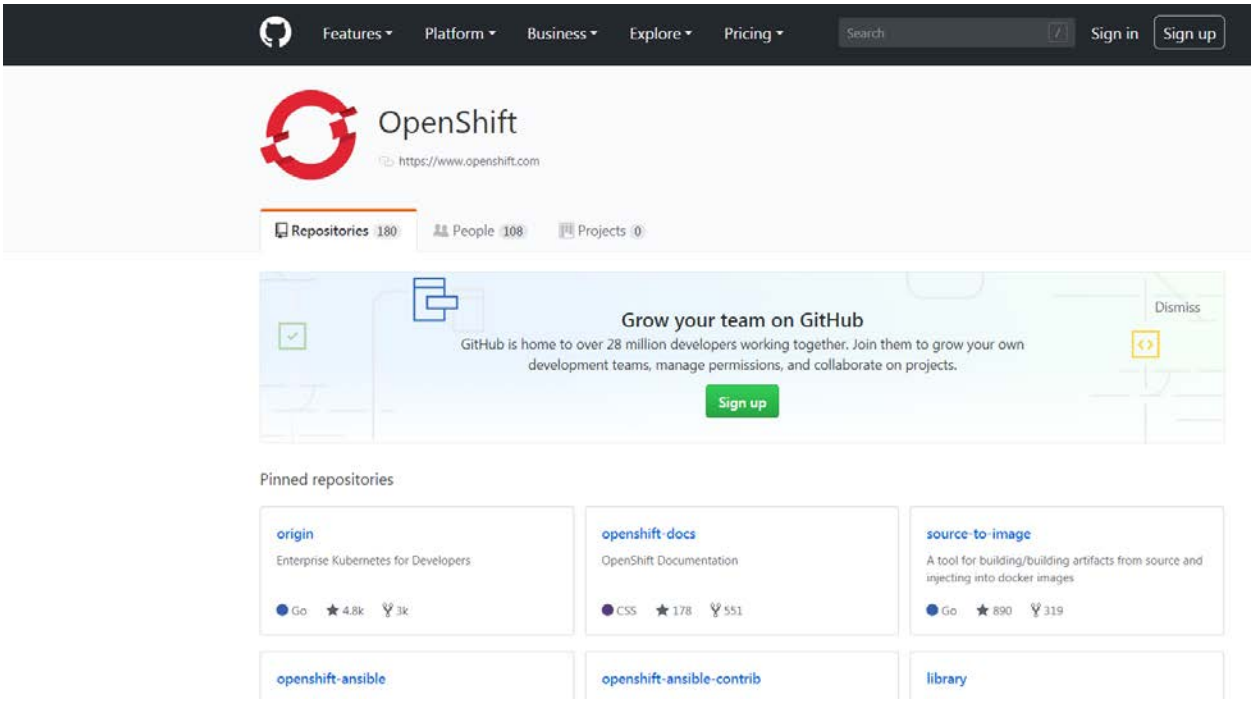
C. `oc rollout`

D. `oc tag`

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

OpenShift Github Repository

A larger catalog of templates, container images, example applications, and documentation is available at the Github repository at the following link: https://github.com/OpenShift



Please notice that you will find many OpenShift repositories on GitHub. Only one is authoritative – as referenced here. The others are forks of this main one. It is recommended to fork it yourself and use your own copy. Then, if you need to add something, you can later submit a pull request.

OpenShift in Action

- "OpenShift in Action" is a very helpful textbook guide through OpenShift.
- It is a highly recommended resource and is available in ebook format.
- There is a GitHub repository with example applications from the book at this website: https://github.com/OpenShiftinaction

Reference for the recommended book:

Duncan, J., Osborne, J. (2018). OpenShift In Action. Manning Publications: Shelter Island, NY. www.allitebooks.com

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

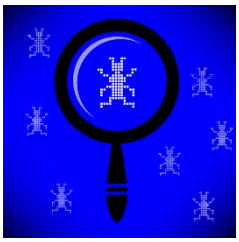
Debugging an Application with OpenShift

Debugging Using the OpenShift CLI

The `oc debug` command opens up a debug pod with a command shell where you can manually debug an existing pod or deployment using shell commands.

An example of the debug command to debug the deployment wildfly-app is shown below:

```
1 oc debug dc/wildfly-app
```



At the shell prompt, type `help` for a list of commands.

To exit the shell, simply type `exit` .

The debugging in this mode implies a good knowledge of OpenShift CLI.

Open a Command Shell into a Container

- The `oc rsh` command is similar to the `oc` debug command; it opens up a command shell into a container.
- First, type `oc get pods` .
- Choose the name of a pod listed, for example: `wildfly-app-2-cctcc` .
- To open the command shell, type the command `oc rsh wildfly-app-2-cctcc` .

At the shell prompt, type "help" for a list of commands.

To exit the shell, simply type "exit".

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Debug using Eclipse

You can also debug using Eclipse or any other software that supports remote debugging.

In order to debug, you will need to open the debug port using this command below.

```
1 oc set env dc/openshift-jee-sample DEBUG=true
```

Get the name of your pods by typing `oc get pods` .

Use the active-pod name in the example below to forward the port to your debugger.

```
1 oc port-forward my-pod-name 8787:8787
```

You can then access the Java application remotely at port 8787.

The debugging process for OpenShift is similar to other application development. The main idea is to connect to the JVM remotely. This capability is provided by the JVM and is similar to Spark debugging. Any IDE will work, not only Eclipse. We can recommend trying IntelliJ.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Interactivity: Poll Question

Question: What command would you use to get the name of your pods when debugging using Eclipse?

A. `oc pod retrieve`

B. `oc pods`

C. `oc pod name`

D. `oc get pods`

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Viewing Logs in OpenShift

You can easily view logs of various types in the CLI using the `oc logs` command.

With `oc logs`, you can view the log data of:

- pods
- builds
- build configurations
- deployment configurations

See the following example. Replace "`my-pod-name`" with the actual name of the pod.

```
1 oc logs pod/my-pod-name
```

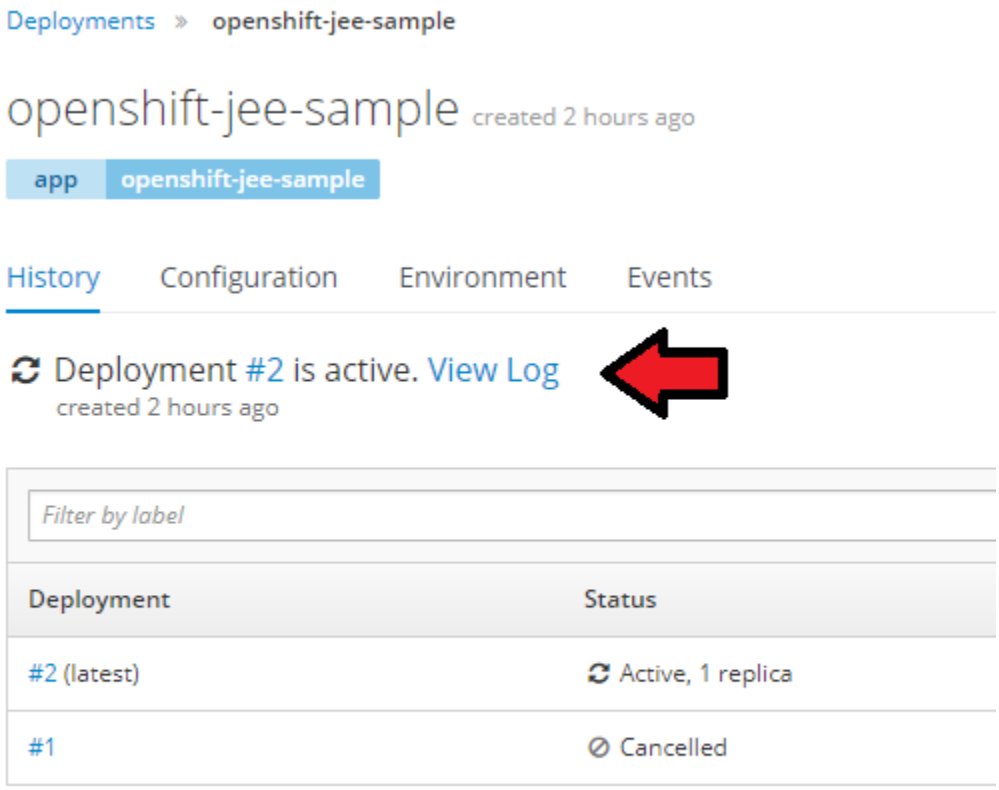
The default setting of "oc logs" will print historical logs up to present on the screen.

If you add the ""-f" argument, it will stream the log onto the console adding real time log data. But then you cannot use that window to execute other commands.

Please see this link for more information on the "oc debug" and "oc logs" commands.
https://docs.OpenShift.com/online/cli_reference/basic_cli_operations.html#debug

Viewing Logs using the Web Console

You can also view the historical and streaming logs using the OpenShift Web Console.



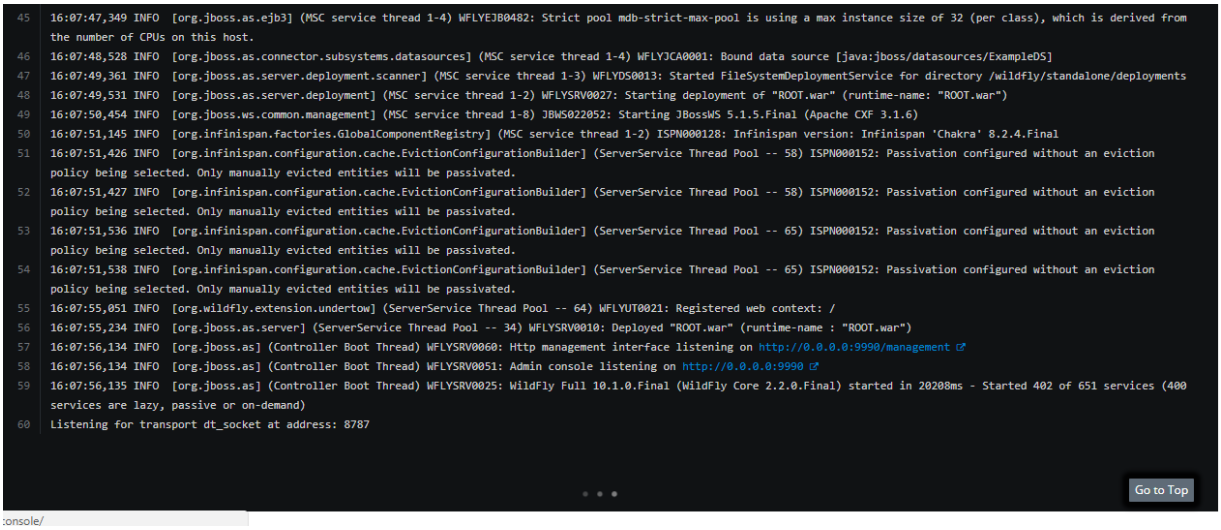
To view the deployment configuration log from the web console, click Applications then Deployments and select the deployment application name.

Then click on the "View log" link.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

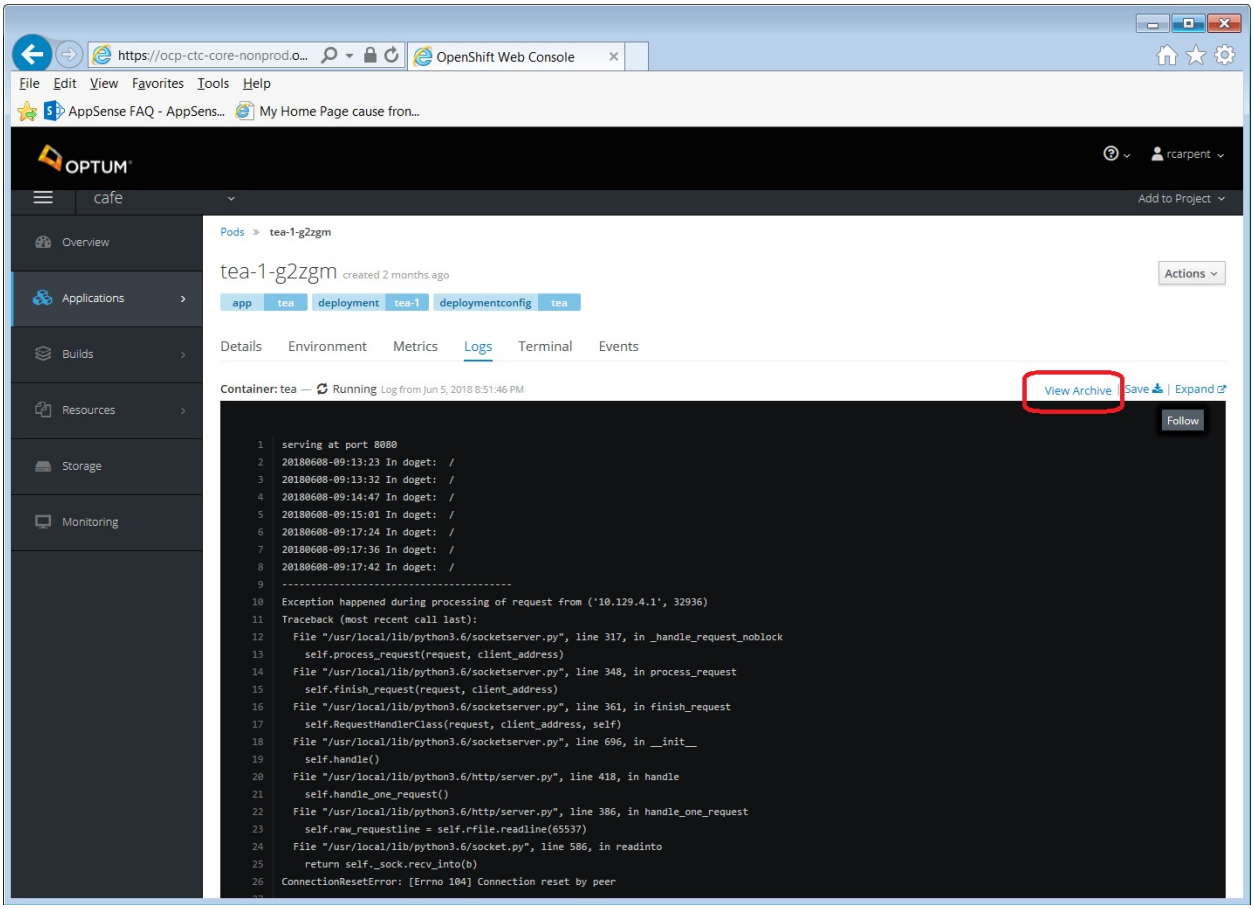
A View of a Log from the Web Console

This is a view of a deployment configuration log.



The log shows no errors or warnings, only the INFO entries. This signifies a completely clean deployment.

View Archive Logs in the Web Console



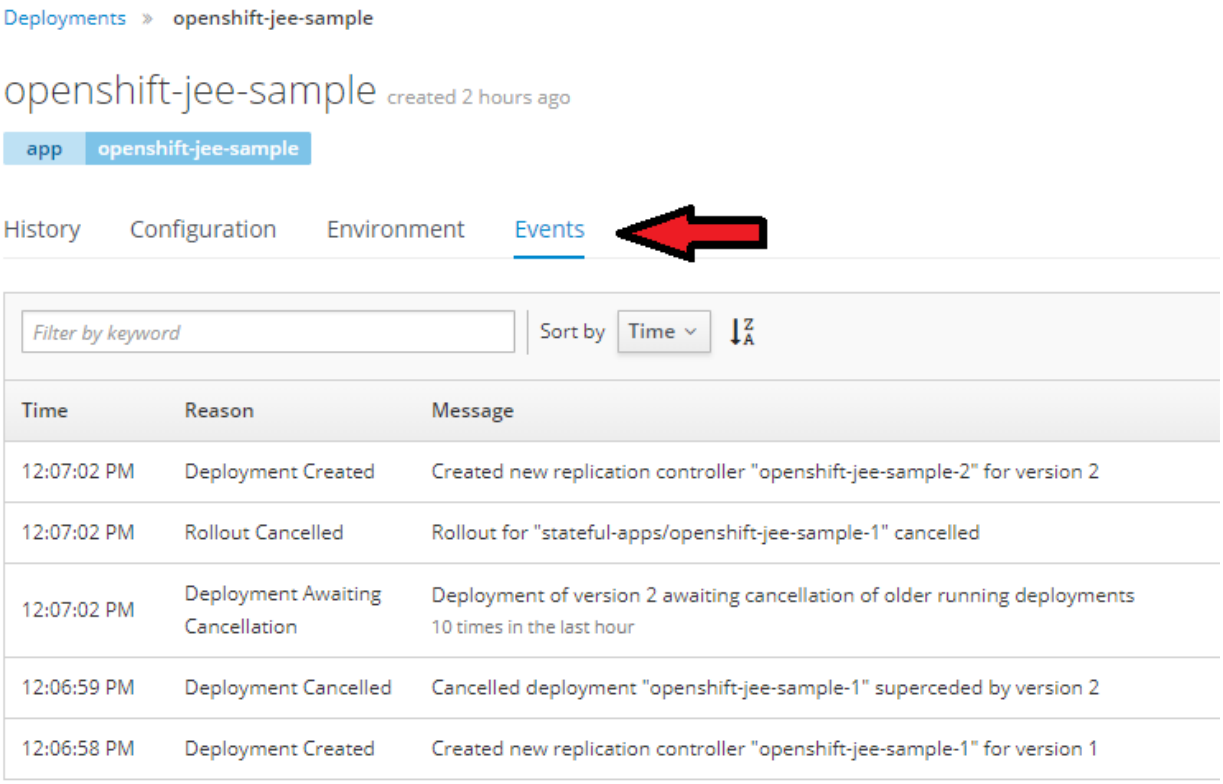
OpenShift has a system to view logs that have been archived.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Viewing Events

"Events" refers to those events which affect API objects on your OpenShift Cluster.

This is a list of events for a deployment of an application.



You can easily view events from the Web Console by clicking the name of a deployment, build, pod, or other resource and clicking on the "Event" tab.

You can request events in the CLI by typing the following command and replace my-project with the actual name of your project.

"oc get events -n my-project"

View this link for more information on events. https://docs.OpenShift.com/online/dev_guide/events.html

Discussion: Review the Concepts

Questions:

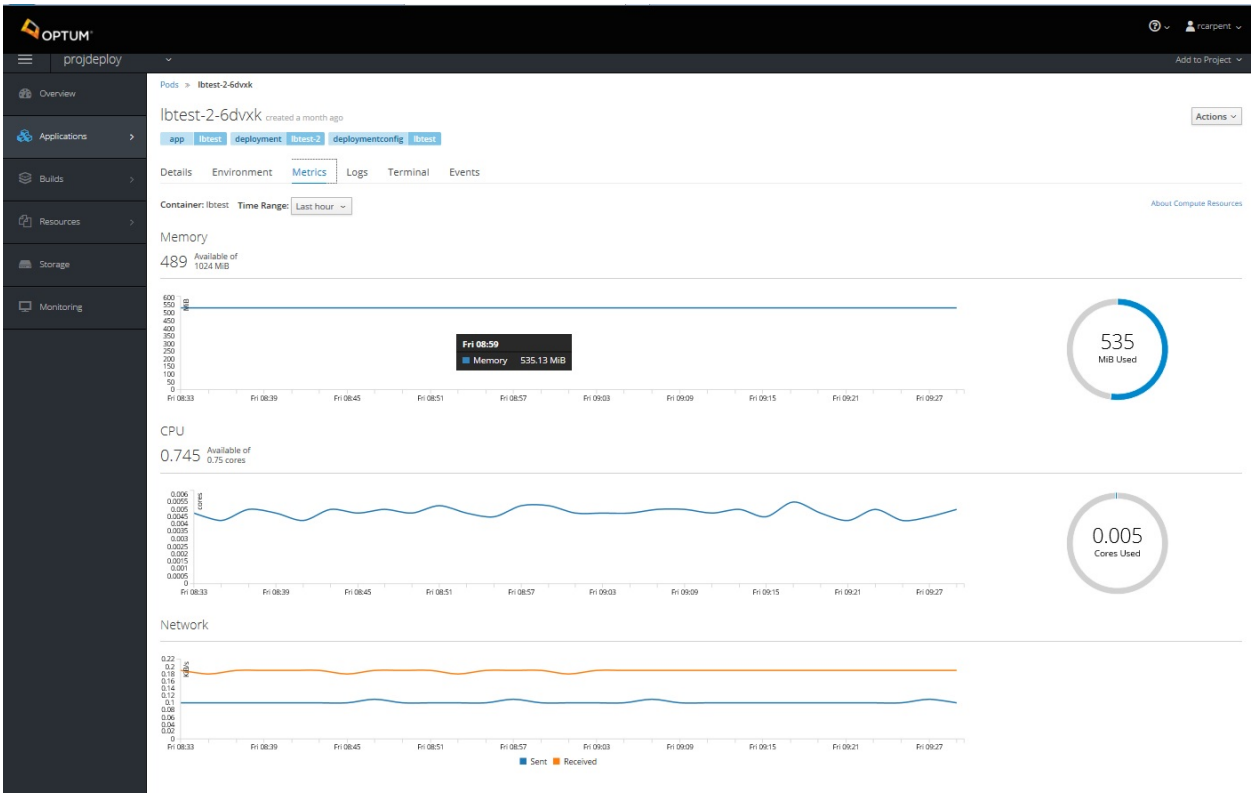
- What are the technical support options offered by RedHat?
- How is debugging done with CLI?
- How do you view logs in OpenShift?

Instructions: Enter your answer into Chat (All Participants).

Take a moment to think about each question and write down your answers.

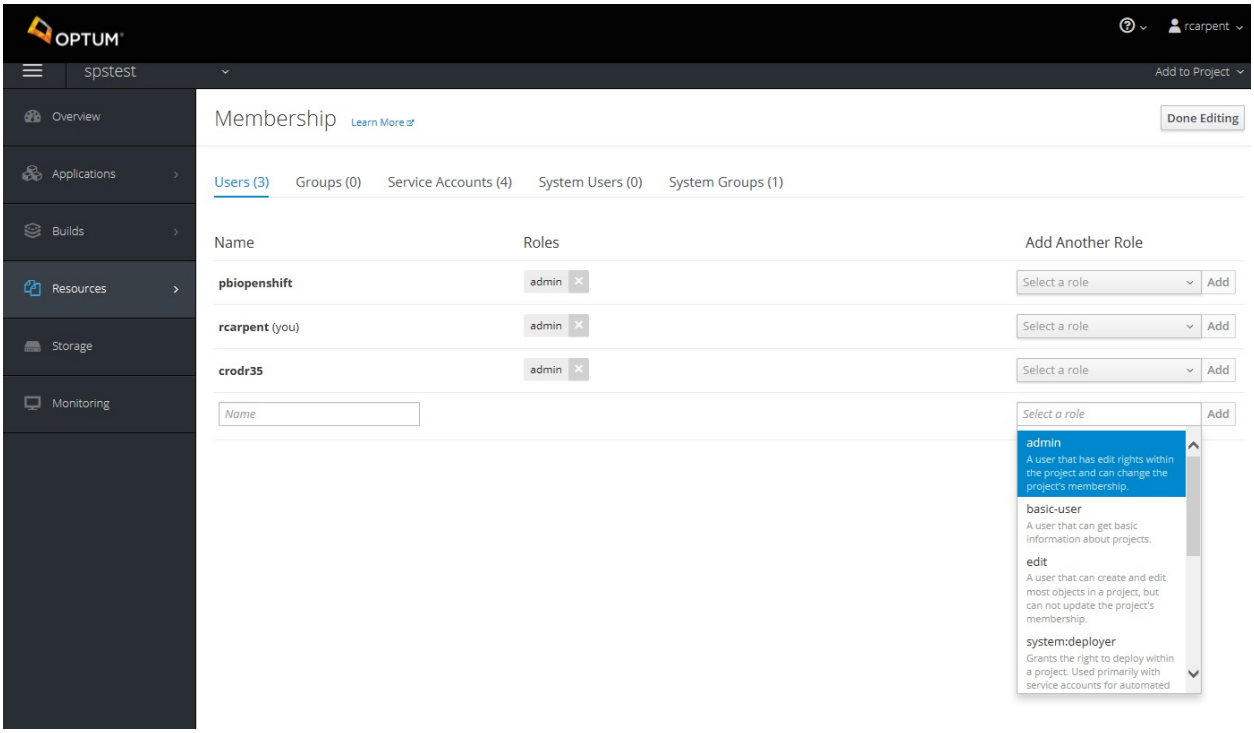
Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Metrics



Here is a reference Optum's metrics.

Membership



Another metric to track would be membership, and that includes the different types of members, individual users, service accounts, and more.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Discussion: Discuss the Practical Aspects of OpenShift

Questions:

- How is OpenShift used in your organization?
- What do you like about it?
- What don't you like?

Instructions: Enter your answer into Chat (All Participants).

Take a moment to think about each question and write down your answers.

Lab: Troubleshooting Applications

Overview:

Time: 20 Minutes

In this lab, you will:

- Install an example Wildfly application.
- View events and logs.
- Use the oc debug and oc rsh commands.
- Take the optional tutorial about debugging using Eclipse.

Instructions:

- Open the Student Lab Manual and follow the steps to perform the lab.

Please find the lab instructions in the file student lab manual.

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum

Module Summary

Now that you have completed this module, you should be able to:

- Recognize the resources available for support.
- Debug an application with OpenShift.

Confidential property of Optum. Do not distribute or reproduce without express permission from Optum.
Confidential and proprietary. | Optum

Course Summary

Now that you have completed this course, you should be able to:

- Describe basic features of OpenShift
- Deploy and manage applications using OpenShift
- Manage more complex deployments using OpenShift
- Troubleshoot applications that were deployed using OpenShift

Confidential property of Optum. Do not distribute without express permission from Optum.
Confidential and proprietary. | Optum