

Oracle Container Engine for Kubernetes

Level 100

Jamal Arif

Oracle Cloud Infrastructure

October, 2019

Safe harbor statement

The following is intended to outline our general product direction. It is intended for information purposes only, and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions.

The development, release, timing, and pricing of any features or functionality described for Oracle's products may change and remains at the sole discretion of Oracle Corporation.



Objectives

After completing this lesson, you should be able to understand:

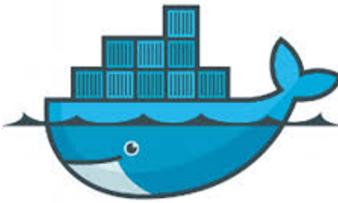
- Containers, Docker container engine
- Orchestration systems and Kubernetes
- Oracle Container Engine for Kubernetes
- Creating a K8s cluster in OCI using ‘quickstart’

Key Containers / Orchestration Use Cases



	Share	Container Use Cases	Orchestration Use Cases
Development	65%	Developer productivity; Consistent appstacks in Dev, Test & Production	Automated deploys to accelerate application release cadence
CI/CD/DevOps	48%	Containerized dependencies; Container registries;	Rolling updates and reversals
Operations	41%	Standardized environments for dev, testing and operations	Resilient, self-healing systems; High Availability; Elastic Scalability
Refactor Legacy Apps	34%	Refactor from N-tier to portable containerized applications	Run distributed, stateful apps on scale-out infrastructure
Migrate to Cloud	33%	Move entire appstacks and see them run identically in the cloud	Cloud bursting; Reduce infrastructure costs by avoiding over-provisioning
New Microservice Apps	32%	Create small purpose-built services that can be assembled to scalable custom applications	Dynamically manage large-scale microservices infrastructure

Docker and Kubernetes



Docker Containers

- Popular, easy to use tooling targeting developer productivity
- De facto standard container runtime and image format
- Used for developer on-boarding and 1st generation application management

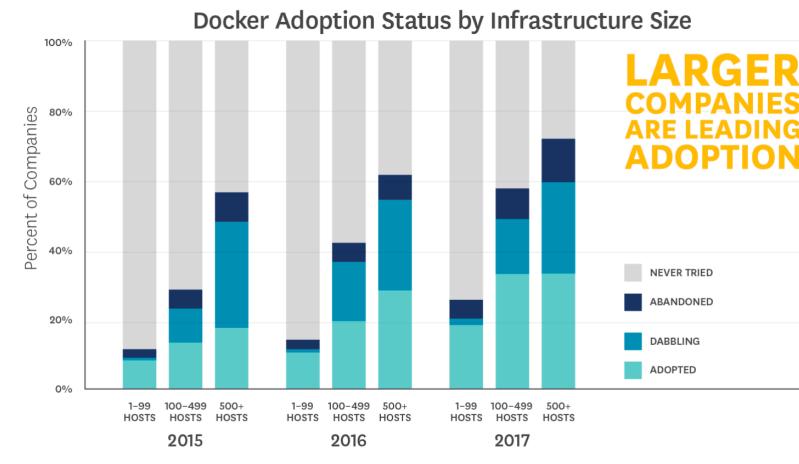


Kubernetes Orchestration

- Production grade container management targeting DevOps and operations, with widespread adoption
- Complex but powerful toolset supporting cloud scale applications
- Rich operations feature set, autoscaling, rolling upgrades, stateful apps and more.

Docker & Kubernetes Lead the Market

Containers (Docker)

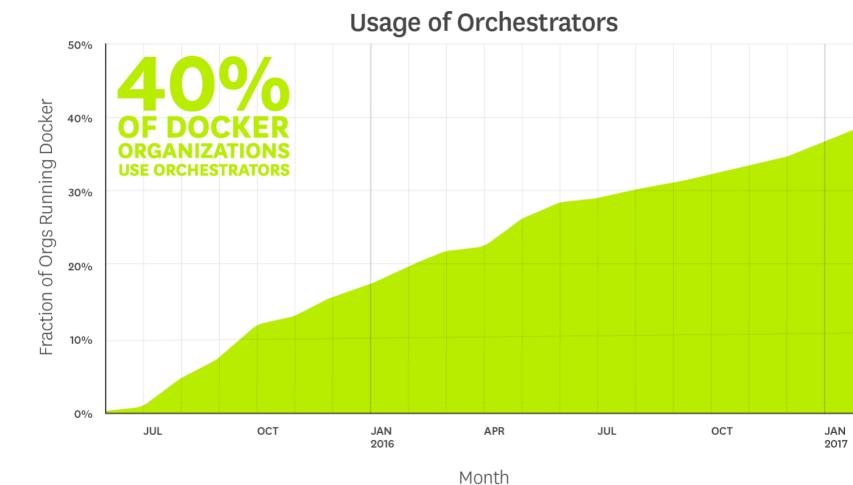


LARGER COMPANIES ARE LEADING ADOPTION

60% of enterprise companies (500+ hosts) use Docker

15% of all the hosts at these companies run Docker

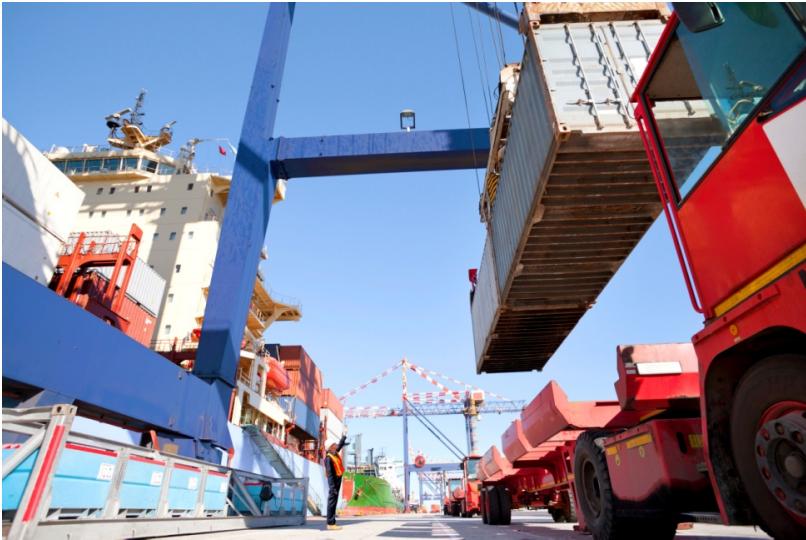
Orchestration (Kubernetes)



40% of Docker users also use orchestrators

80% of these orchestration users prefer Kubernetes

Container Orchestration And Containers as a Service (CaaS)



- Multi-container apps
- Scheduling
- Service Discovery
- Maintaining Desired State

- Orchestration as a service
- Hosted Container Runtime
- Minimize operational overhead



Container Engine for Kubernetes - OKE

Three Ways to Run Kubernetes on Oracle Cloud Infrastructure

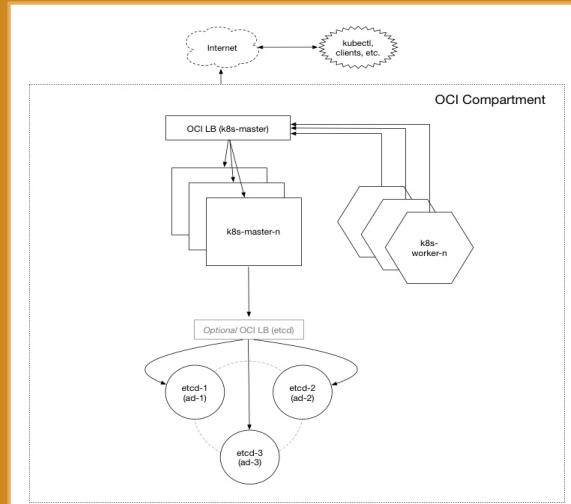
Roll Your Own, Pre-Built Installer, Managed Service

Oracle Cloud
Infrastructure



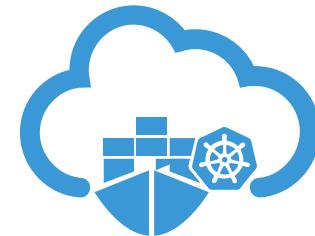
Roll-Your-Own
Container Management

Quickstart Experience
(OSS Terraform Installer on GitHub)



Pre-Built Kubernetes
Installer

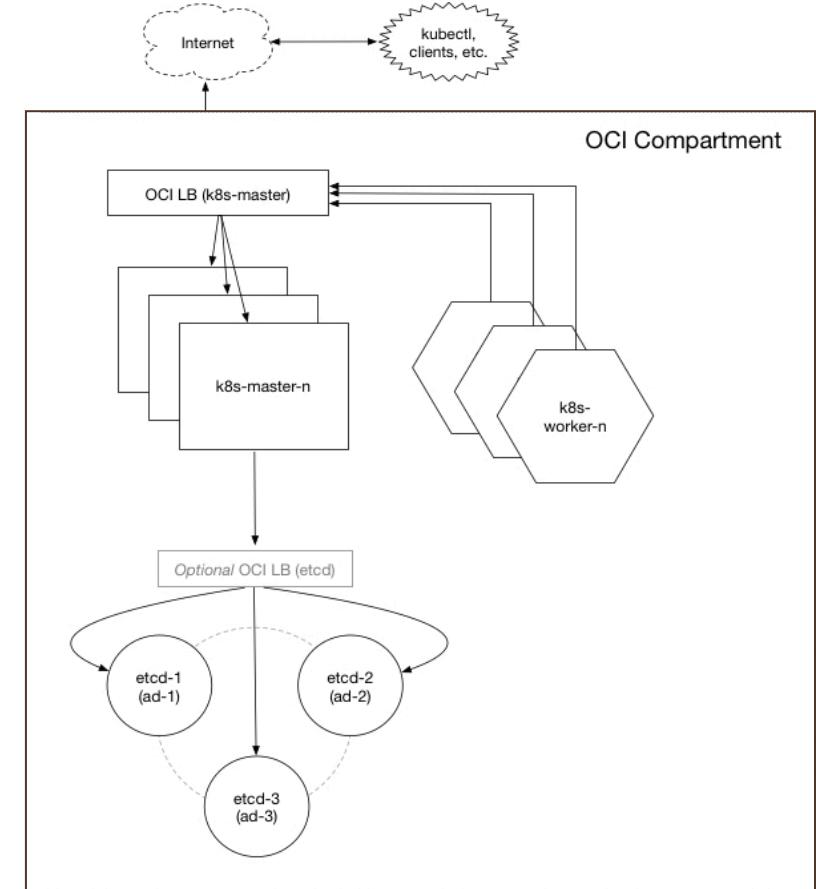
Container Engine for
Kubernetes (OKE)



Enterprise Class Managed
Kubernetes Service

Terraform Kubernetes Installer for OCI

- Open source, based on Terraform
 - Oracle developed for Kubernetes on OCI
 - Available now on Github
 - <https://github.com/oracle/terraform-kubernetes-installer>
- Key Highlights
 - Highly available Kubernetes cluster configured in your OCI tenancy and compartment
 - Creates VCN, subnets, LBs and instances for control plane
 - Specify number and shape of nodes for your cluster
 - Scale your cluster as needed
- <https://blogs.oracle.com/developers/get-a-highly-available-kubernetes-cluster-on-oracle-cloud-infrastructure-in-minutes>





Introducing Container Engine for Kubernetes - OKE

What is It?

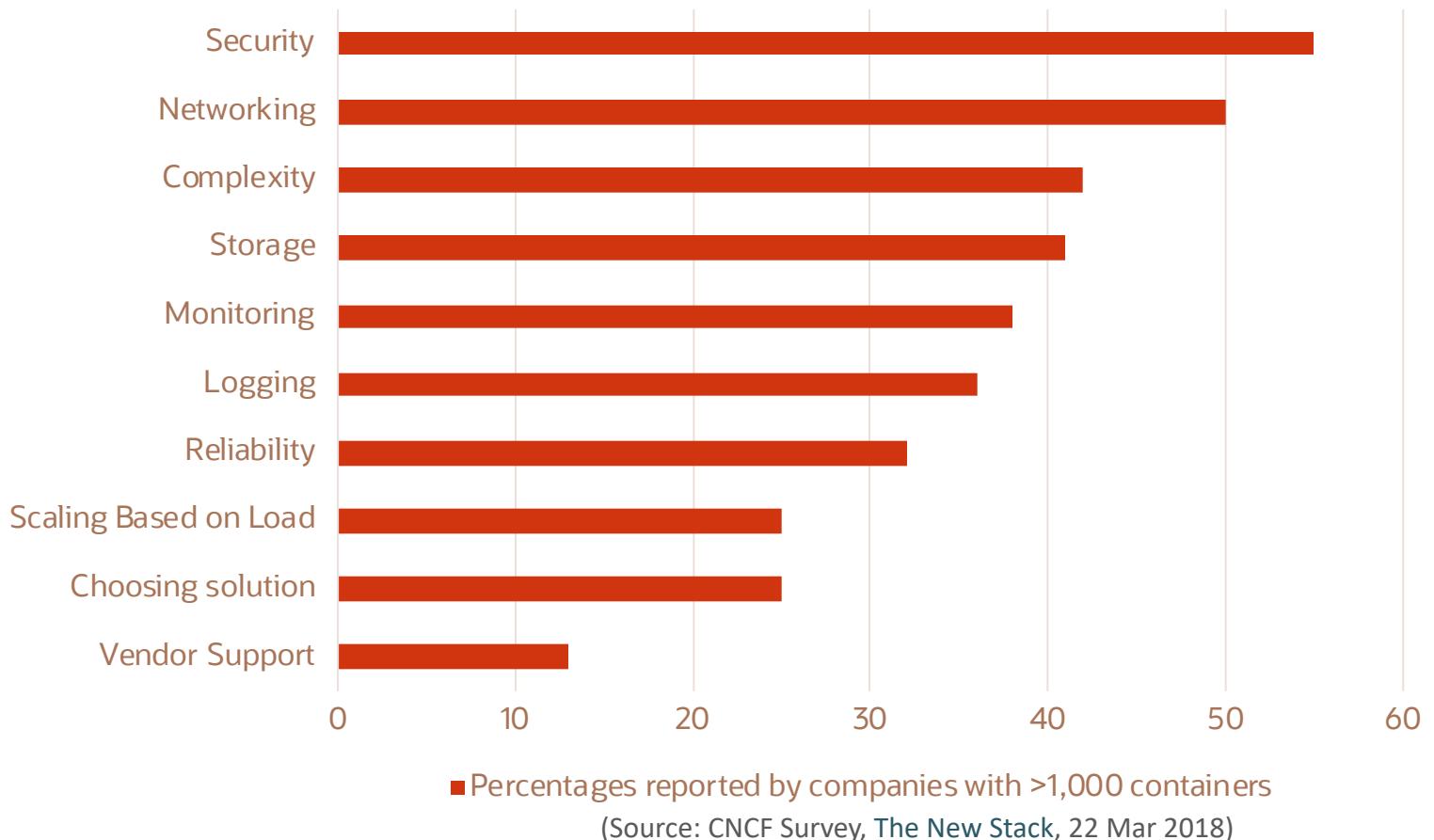
What Problems
Does it Solve?

Key Benefits

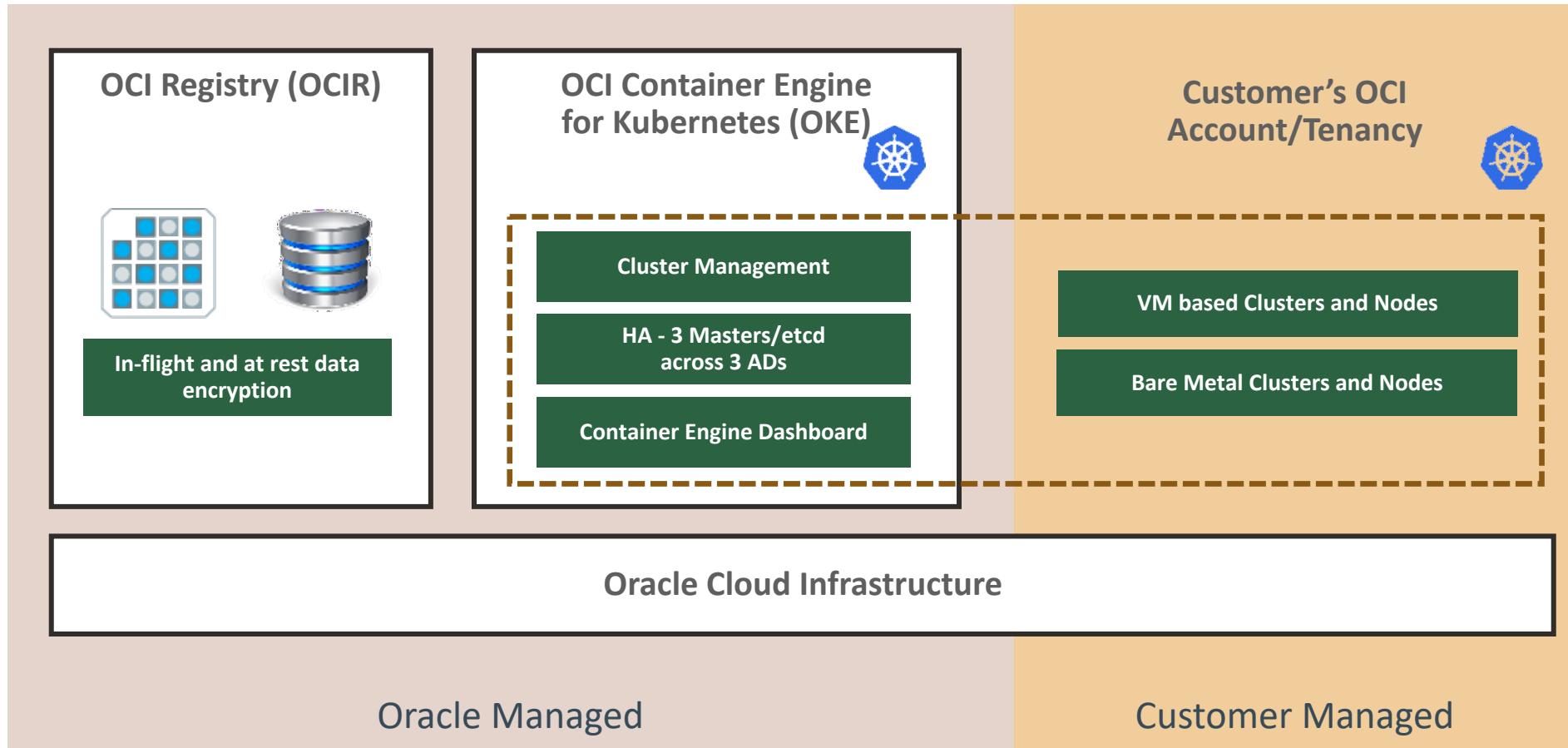
- Managed Kubernetes container service to deploy and run your own container based apps
- Tooling to create, scale, manage & control your own standard Kubernetes clusters instantly
- Too complex, costly and time consuming to build & maintain environments
- Too hard to integrate Kubernetes with a registry and build process for container lifecycle management
- Too difficult to manage and control team access to production clusters
- Enables developers to get started and deploy containers quickly. Gives DevOps teams visibility and control for Kubernetes management.
- Combines production grade container orchestration of open Kubernetes, with control, security, IAM, and high predictable performance of Oracle's next generation cloud infrastructure

Kubernetes Challenges

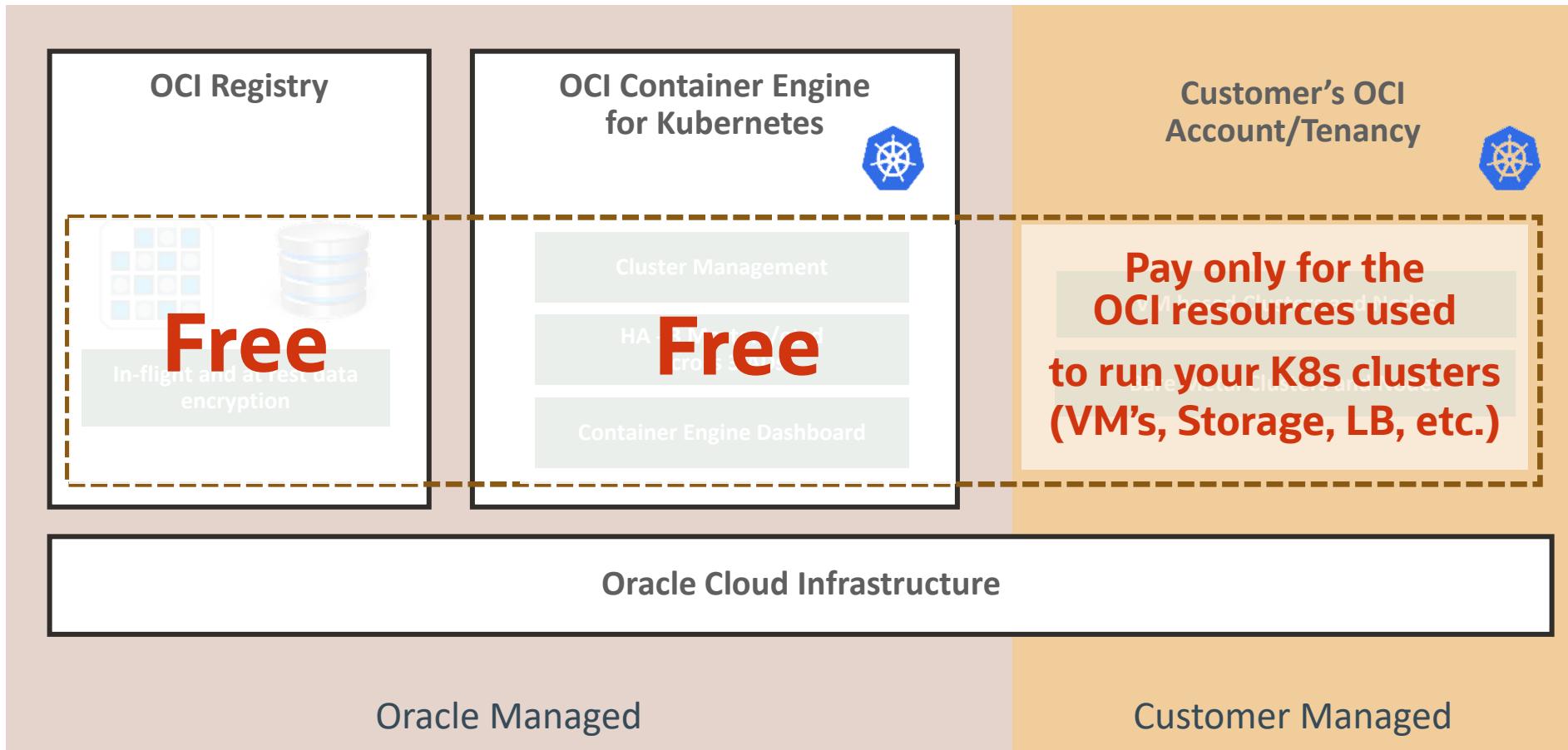
- Managing Kubernetes Infrastructure, upgrading, security
- Container networking & persistent storage
- Managing Teams & Access
- CI/CD Integration, automated testing, conditional release



Working with OKE and OCIR on OCI



OKE/OCIR Pricing and Packaging



Oracle Container Engine (OKE) and Registry



Container Native

- **Standard Docker & Kubernetes**
 - Deploy standard & open upstream Docker and Kubernetes versions for compatibility across environments
- **Registry Integration**
 - Full Docker v2 compatible private registry to store and manage images
- **Container Engine**
 - Deploy and operate containers and clusters
- **Full integration to cloud networking and storage**
 - Leverage the enterprise class networking, load balancing and persistent storage of Oracle Cloud Infrastructure

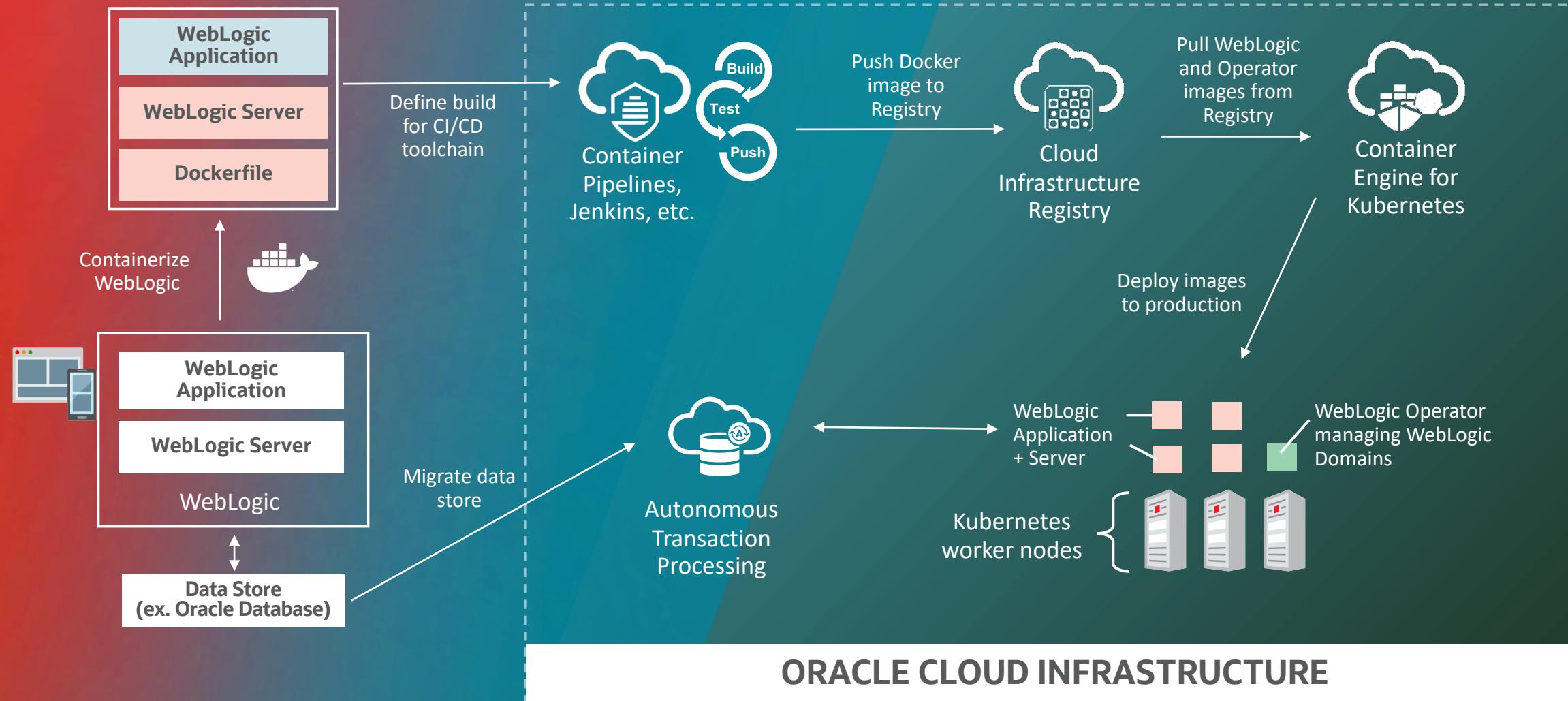
Developer Friendly

- **Streamlined Workflow**
 - Use your favorite CI to push containers to the registry, then Kubernetes to deploy to clusters and manage operations
- **Full REST API**
 - Automate the workflow, create and scale clusters through full REST API
- **Built In Cluster Add-Ons**
 - Kubernetes Dashboard, DNS & Helm
- **Open Standards**
 - Docker Based Runtime
 - Worker Node SSH Access
 - Standard Kubernetes

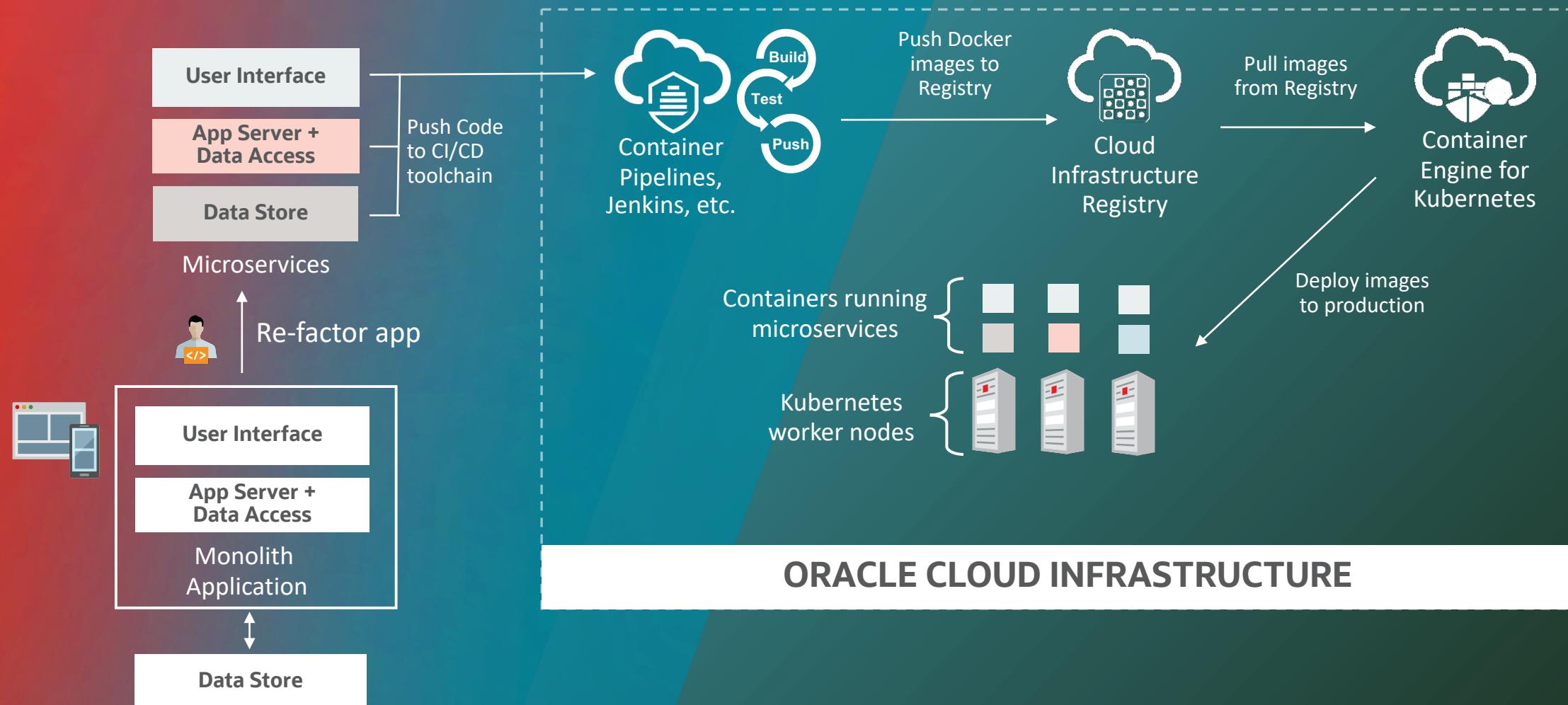
Enterprise Ready

- **Simplified Cluster Operations**
 - Fully managed, highly available registry, master nodes and control plane
 - One-click Quick Create for secure Private Worker Nodes/Subnets
- **Full Bare Metal Performance and Highly Available IaaS**
 - Combine Kubernetes with bare metal shapes for raw performance
 - Deploy Kubernetes clusters across multiple Availability Domains for resilient applications
- **Team Based Access Controls**
 - Control team access and permissions to clusters

Containers Use Case: Lift & Shift WebLogic Application



Containers Use Case: Refactor an Existing Application





Creating an OKE Cluster in OCI

Pre-requisites for creating a K8s Cluster via Quickstart

- Monthly universal Credits have limit of 3 clusters per OCI region with 1000 nodes in a cluster and Pay-as-you-go or Promo accounts have a limit for One Cluster (by default)
- Must also have compute Instance Quota (Required) – to launch k8s worker nodes in an AD or across ADs for HA
- Required Policy in the root compartment of your tenancy
allow service OKE to manage all-resources in tenancy
- To launch a K8s cluster, user must be either part of the Admin group or a group to which a policy grants the appropriate Container Engine for Kubernetes permissions.
- Policies can be created for users which are not part of the admin group
- For Example: To enable users in group 'dev-team' to perform any operation on cluster-related resources →
allow group dev-team to manage cluster-family in tenancy

Note: Policies must also grant the group 'dev-team' Networking permissions of VCN_READ and VCN_CREATE, SUBNET_READ and SUBNET_CREATE, COMPARTMENT_INSPECT, INTERNET_GATEWAY_CREATE, NAT_GATEWAY_CREATE, ROUTE_TABLE_UPDATE, SECURITY_LIST_CREATE: Details [here](#)

OKE Quickstart

Step 1: Navigate to Menu → Developer Services → Container Clusters (OKE) → Create Cluster

The screenshot shows the Oracle Cloud Developer Services interface for creating a Container Cluster (OKE). The top navigation bar includes 'Developer Services' (selected), 'Container Clusters (OKE)', 'Marketplace', and 'Registry (OCIR)'. Below the navigation is the Oracle Cloud logo and a search bar. The main content area is titled 'Clusters in tutorials Compartment' and contains a yellow warning box for 'Clusters Requirements'. It lists required privileges: List, Get, and Create VCNs; List, Get, and Create Subnets; List Availability Domains; Create Internet Gateways; Create NAT Gateways; Update Route Tables; and Create Security Lists. It also notes that tenancy must allow service OKE to manage resources. A note states: 'Without the above privileges and policies, various errors will be presented and the cluster service will not function correctly.' A 'Create Cluster' button is at the bottom left, and a table below shows no existing clusters.

Name	Status	Node Pools	VCN	Version	Created
No clusters exist. Create one to get started.					

Showing 0 item(s)

OKE Quickstart

Step 2: Cluster Creation

Name of the Cluster

The version of Kubernetes to run on the master nodes and worker nodes of the cluster. Either accept the default version or select a version of your choice. Amongst other things, the Kubernetes version you select determines the default set of admission controllers that are turned on in the created cluster (the set follows the recommendation given in the [Kubernetes documentation](#) for that version).

Cluster Creation [help](#) [close](#)

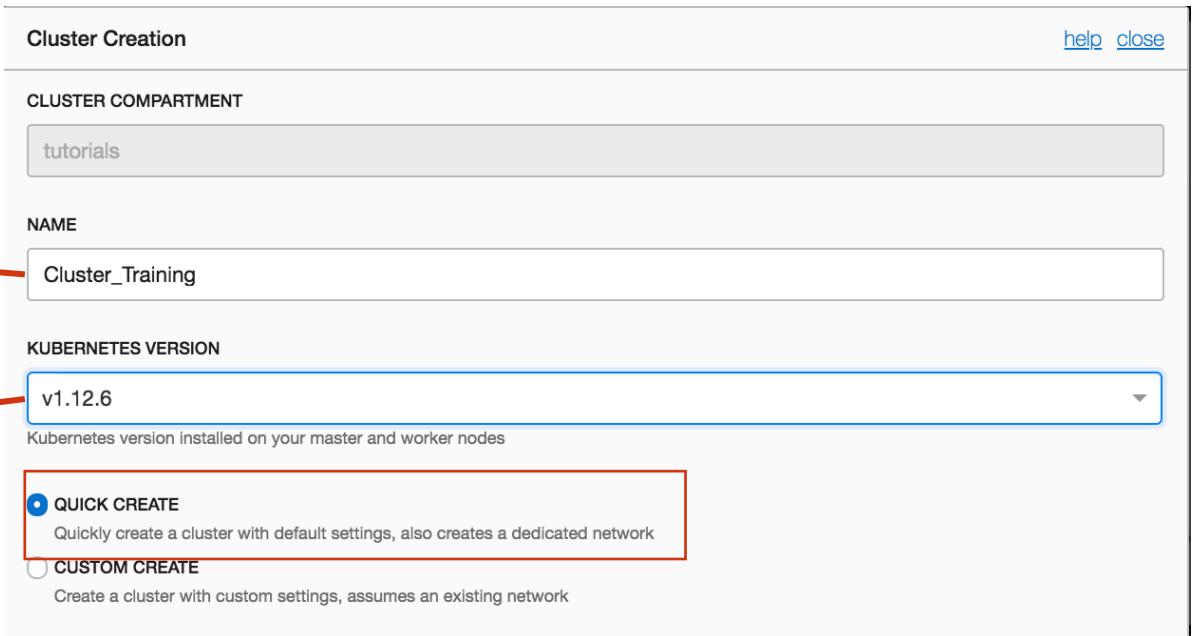
CLUSTER COMPARTMENT
tutorials

NAME
Cluster_Training

KUBERNETES VERSION
v1.12.6 ▼
Kubernetes version installed on your master and worker nodes

QUICK CREATE
Quickly create a cluster with default settings, also creates a dedicated network

CUSTOM CREATE
Create a cluster with custom settings, assumes an existing network



OKE Quickstart (contd...)

Step 2: Cluster Creation

New network resources for the cluster are created automatically, the worker nodes in a 'quick cluster' can be created in private subnets or public. A NAT gateway is created in case of private subnets.

Shape: The compute shape to use for each node in the node pool.

Quantity per Subnet: The number of worker nodes to create for the node pool in each private subnet.

Public SSH Key: (Optional) The public key is installed on all worker nodes in the cluster, and you can use this key to access the worker nodes (Connect via Bastion Host since worker nodes are in Private subnets)

Kubernetes Labels: One or more labels (in addition to a default label) to add to worker nodes in the node pool to enable the targeting of workloads at specific node pools.

The screenshot shows the OCI console interface for creating a Kubernetes cluster. It consists of two main sections: 'Create Virtual Cloud Network' and 'Create Node Pool'.

Create Virtual Cloud Network: This section is titled 'Create Virtual Cloud Network'. It displays a message: 'A new VCN network will be created for you in order to have a functioning cluster'. Below this, it shows 'COMPARTMENT: Demos' and 'RESOURCE CREATION: 1 VCN, 1 service lb subnet and 1 worker node subnet'. There are two radio button options: 'PRIVATE' (selected) and 'PUBLIC'. The 'PRIVATE' option is described as 'The Kubernetes worker nodes that are created will be hosted in private subnet(s)'. The 'PUBLIC' option is described as 'The Kubernetes worker nodes that are created will be hosted in public subnet(s)'.

Create Node Pool: This section is titled 'Create Node Pool'. It includes fields for 'NAME: pool1', 'COMPARTMENT: Kubernetes', 'KUBERNETES VERSION: v1.12.6', and 'IMAGE: Oracle-Linux-7.5'. It has sections for 'SHAPE' (set to 'VM.Standard1.2'), 'QUANTITY PER SUBNET' (set to '1'), 'PUBLIC SSH KEY' (optional), and 'Kubernetes Labels'. The 'PUBLIC SSH KEY' section contains a placeholder 'Input SSH public key' and a note: 'In order to access your private nodes with a public SSH key you will need to set up a bastion host (a.k.a. jump box). [Learn more about setting up a bastion host](#)'. The 'Kubernetes Labels' section shows a table with one row: 'name' under 'KEY' and 'pool1' under 'VALUE'. A note below states: 'Nodes added to this node pool will automatically get one or more Kubernetes labels applied, enabling users to target Kubernetes workloads in a specific pool'. A 'Another Pair' button is located at the bottom right of the labels table.

OKE Quickstart (contd...)

Step 2: Cluster Creation

Kubernetes Dashboard Enabled: Select if you want to use the Kubernetes Dashboard to deploy and troubleshoot containerized applications, and to manage Kubernetes resources. See [Starting the Kubernetes Dashboard](#).

Tiller (Helm) Enabled: Select if you want Tiller (the server portion of Helm) to run in the Kubernetes cluster. With Tiller running in the cluster, you can use Helm to manage Kubernetes resources.



Additional Add Ons

- KUBERNETES DASHBOARD ENABLED
- TILLER (HELM) ENABLED

- VIEW DETAIL PAGE AFTER THIS CLUSTER IS REQUESTED

Create

Cancel

K8s Cluster in minutes ..

Cluster details

TestCluster

[Access Kubeconfig](#) [Delete Cluster](#)

Cluster Details

Cluster Information

Cluster Status: ✓ Active

Node Pools: 1

Cluster Id: ...cydgobwmm4t [Show](#) [Copy](#)

Compartment: jamalarif (root)/Kubernetes

Kubernetes Version: v1.12.6

Kubernetes Address: ...com:6443 [Show](#) [Copy](#)

Launched: Wed, 13 Mar 2019 17:28:23 GMT

Created By: sardar.jamal.arif@oracle.com

Network Information

VCN Name: [oke-vcn-quick-TestCluster-20190313172636](#)

VCN Id: ...m2oll7sa [Show](#) [Copy](#)

Compartment: jamalarif (root)/Kubernetes

Pods CIDR: 10.244.0.0/16

Services CIDR: 10.96.0.0/16

Service LB Subnet 1: ...US-ASHBURN-AD-1 [Show](#) [Copy](#)

Service LB Subnet 2: ...US-ASHBURN-AD-2 [Show](#) [Copy](#)



K8s Cluster in minutes ..

Node Pool details

Node Pools

Add Node Pool

To access private nodes with a public SSH key, [set up a bastion host](#). Dismiss

pool1		Actions ▾		
Details	Labels			
Kubernetes Ver: v1.12.6	Image Name: Oracle-Linux-7.5	Nodes Per Subnet: 1		
Shape: VM.Standard1.2	Total Worker Nodes: 3	Number of Subnets: 3		
Node Pool Id: ...ayzsg43g Show Copy				
Hide Node Details				
Instance Name ▾	Compute Node State	Subnet	Public IP	Kubernetes Ver
oke-cydgobwmm4t-nzdazsg43g-shkj2fwiyfq-0	ACTIVE	oke-subnet-quick-TestCluster-20190313172636-fyhg:US-ASHBURN-AD-1 (Private)	Unavailable	v1.12.6
oke-cydgobwmm4t-nzdazsg43g-suzpischafa-0	ACTIVE	oke-subnet-quick-TestCluster-20190313172636-fyhg:US-ASHBURN-AD-3 (Private)	Unavailable	v1.12.6
oke-cydgobwmm4t-nzdazsg43g-szhfpwjuqa-0	ACTIVE	oke-subnet-quick-TestCluster-20190313172636-fyhg:US-ASHBURN-AD-2 (Private)	Unavailable	v1.12.6

Showing 3 Item(s)



Accessing the K8s Cluster - Dashboard

The screenshot shows the Kubernetes Dashboard's "Getting Started" page. On the left, there's a sidebar with links: Resources, Node Pools, Work Requests, and Getting Started (which is highlighted with a blue border). A red arrow points from the "Getting Started" link in the sidebar to the "Getting Started" section on the right.

Getting Started

Kubernetes Dashboard

You can use the Kubernetes Dashboard to get an overview of applications running in your cluster. It also provides information on the state of Kubernetes resources in your clusters, and on any errors that may have occurred.

1. `kubectl proxy`
2. Dashboard will be available at:
<http://localhost:8001/api/v1/namespaces/kube-system/services/https:kubernetes-dashboard:/proxy/>

Quick Start: Deploy Sample App

1 Access Kubeconfig File
To get started, learn how to download the `kubeconfig` file for this cluster by clicking below. This file will contain a series of authentication mechanisms and cluster connection information.
[Access Kubeconfig](#)

2 Check Version
Verify that kubernetes is available by entering the following command in your terminal
1. `kubectl version`

3 Deploy Application
Deploy a sample hello world application by running the following command in your terminal.
1. `kubectl create -f https://k8s.io/docs/tasks/run-application/deployment.yaml`

Accessing the K8s Cluster - Dashboard

Kubernetes

Search

+ CREATE | 🔍

☰ Overview

Cluster

- Namespaces
- Nodes
- Persistent Volumes
- Roles
- Storage Classes

Namespace

default ▾

Overview

Workloads

Cron Jobs

Daemon Sets

Deployments

Jobs

Pods

Replica Sets

Replication Controllers

Stateful Sets

Discovery and Load Balancer

Ingresses

Services

Config and Storage

Config Maps

Persistent Volume Claim

Secrets

Settings

About

Workloads Statuses

Deployments

Pods

Replica Sets

Deployments

Name	Labels	Pods	Age	Images
jenkins-master	app:jenkins	1 / 1	3 days	jenkins
nginx	run:nginx	3 / 3	3 days	nginx

Pods

Name	Node	Status	Restarts	Age
jenkins-master-6bd4f5cb76-6jvw8	10.0.10.2	Running	0	3 days
nginx-cdb6b5b95-ckbfv	10.0.12.2	Running	0	3 days
nginx-cdb6b5b95-sh25z	10.0.10.2	Running	0	3 days
nginx-cdb6b5b95-m4t6x	10.0.11.2	Running	0	3 days

Replica Sets

Name	Labels	Pods	Age	Images
jenkins-master-6bd4f5cb76	app:jenkins-master pod-template-hash:6bd4f5cb76	1 / 1	3 days	jenkins
nginx-cdb6b5b95	pod-template-hash:cdb6b5b95 run:nginx	3 / 3	3 days	nginx

Discovery and Load Balancing

Services

Name	Labels	Cluster IP	Internal endpoints	External endpoints	Age
jenkins-master	app:jenkins	10.96.189.152	jenkins-master:80 TCP jenkins-master:30076 TCP	129.213.193.160:80	3 days
kubernetes	component:apiserver provider:kubernetes	10.96.0.1	kubernetes:443 TCP	-	5 days

Accessing the K8s Cluster with kubectl

How to Access Kubeconfig [help](#) [close](#)

You must have [downloaded and installed](#) the OCI CLI and [configured](#) it for use.

To access the kubeconfig for your cluster, run the following commands:

```
1. mkdir -p $HOME/.kube
2. oci ce cluster create-kubeconfig --cluster-id
   ocid1.cluster.oc1.iad.aaaaaaaaaftdcztbgrtggobzgbsgcnrgy2wmmdcgu4wezjygcyclgobwmm4t --file
   $HOME/.kube/config --region us-ashburn-1
```

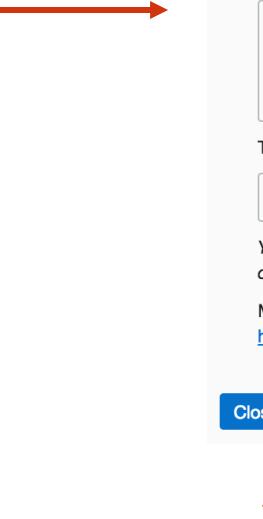
To set your KUBECONFIG environment variable to the file for this cluster, use:

```
export KUBECONFIG=$HOME/.kube/config
```

You may have to add this to your shell initiation script if you wish to persist this change. For more information on managing kubeconfig files, please refer to the official [Kubernetes documentation](#).

More information on the available commands for OCI's Container Engine for Kubernetes CLI can be found [here](#).

[Close](#)



```
[sararif-mac:~ sararif]$ kubectl cluster-info
Kubernetes master is running at https://cydgobwmm4t.us-ashburn-1.clusters.oci.oraclecloud.com:6443
KubeDNS is running at https://cydgobwmm4t.us-ashburn-1.clusters.oci.oraclecloud.com:6443/api/v1/namespaces/kube-system/services/kube-dns:dns/proxy

To further debug and diagnose cluster problems, use 'kubectl cluster-info dump'.
[sararif-mac:~ sararif]$ kubectl get nodes -o wide
NAME      STATUS    ROLES   AGE     VERSION   EXTERNAL-IP   OS-IMAGE           KERNEL-VERSION   CONTAINER-RUNTIME
10.0.10.2  Ready     node    5d      v1.12.6  <none>        Oracle Linux Server 7.5  4.14.35-1818.2.1.el7uek.x86_64  docker://18.9.1
10.0.11.2  Ready     node    5d      v1.12.6  <none>        Oracle Linux Server 7.5  4.14.35-1818.2.1.el7uek.x86_64  docker://18.9.1
10.0.12.2  Ready     node    5d      v1.12.6  <none>        Oracle Linux Server 7.5  4.14.35-1818.2.1.el7uek.x86_64  docker://18.9.1
sararif-mac:~ sararif$
```

<https://kubernetes.io/docs/reference/kubectl/kubectl/>

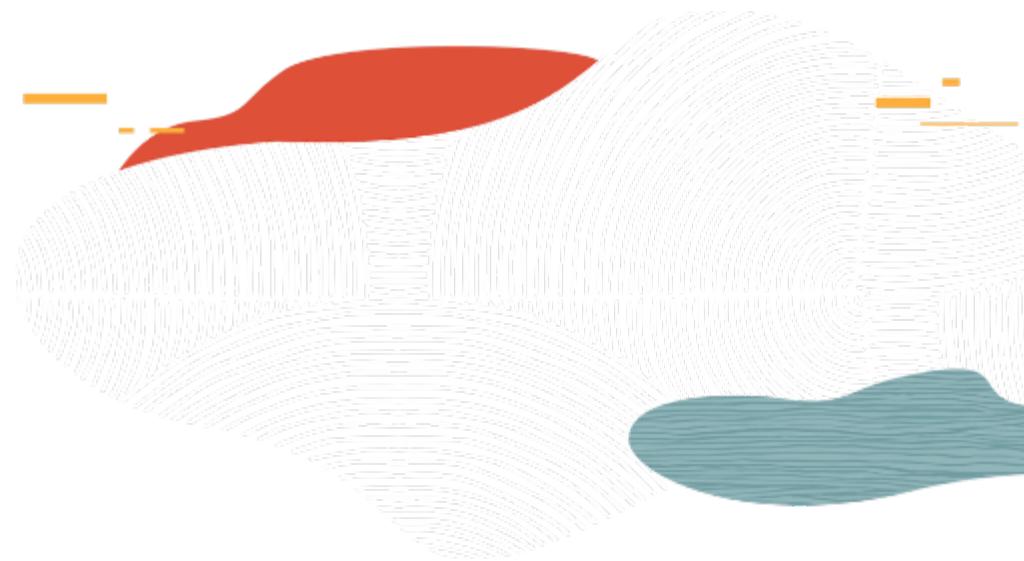


DEMO

<http://bit.ly/30cIn3I>

Summary

- OCI Container engine for Kubernetes is a managed Kubernetes service
- K8s service is itself free, you only pay for the resources you use for your worker nodes
- Create a highly available Kubernetes cluster using quickstart in minutes on OCI



Oracle Cloud always free tier:
oracle.com/cloud/free/

OCI training and certification:
cloud.oracle.com/en_US/iaas/training
cloud.oracle.com/en_US/iaas/training/certification
education.oracle.com/oracle-certification-path/pFamily_647

OCI hands-on labs:
ocitraining.qloudable.com/provider/oracle

Oracle learning library videos on YouTube:
youtube.com/user/OracleLearning