ORACLE

OCI Compute Services

Rohit Rahi

Oracle Cloud Infrastructure

Feb 2020

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, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Agenda

Bare Metal

VMs

Scaling

Container Engine
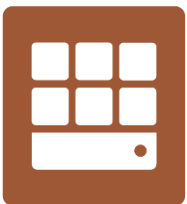
Functions

OCI Compute Services

Bare Metal	Dedicated Virtual Hosts	Virtual machines	Container Engine	Functions
Code	Code	Code	Code	Code
App Container	App Container	App Container	App Container	
Language Runtime	Language Runtime	Language Runtime		
Operating System	Operating System	Operating System		
Virtualization				
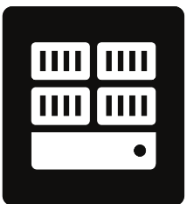
Bare Metal, VM and Dedicated Hosts

Bare Metal (BM)

Direct Hardware Access –
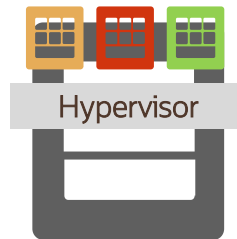customers get the full bare
metal server
(single-tenant server)

Virtual Machine (VM)

A hypervisor to virtualize the
underlying bare metal server into
smaller VMs
(multi-tenant VMs)

Dedicated VM Hosts (DVH)
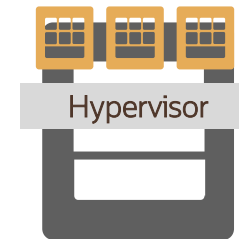
Run your VMs instances
on dedicated bare metal
servers (single-tenant VMs)

No VMs

Bare Metal Server

VMs (multi-tenant)

Hypervisor

Bare Metal Server

VMs (single-tenant)
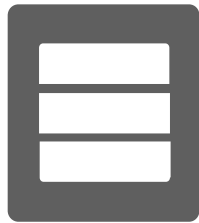
Hypervisor

Bare Metal Server

VM compute instances runs on the same hardware as a Bare Metal instances, leveraging the same cloud-optimized
hardware, firmware, software stack, and networking infrastructure

Bare Metal use cases

Direct Hardware Access with all the Security, Capabilities, Elasticity and Scalability of OCI

Hypervisor

Workloads that are
Performance-intensive

Workloads that are
not virtualized

Workloads that require
a specific hypervisor

Workloads that
require BYO Licensing

VM use cases

Hypervisor

Use VMs when you want to control all aspects of an environment

Use VMs when you want to deploy a legacy app running on Windows or Linux

You can use VMs to move applications from on-premises to Oracle Cloud Infrastructure

VMs require work – OS patch management, security configuration, monitoring, application configuration and scaling to handle variable traffic
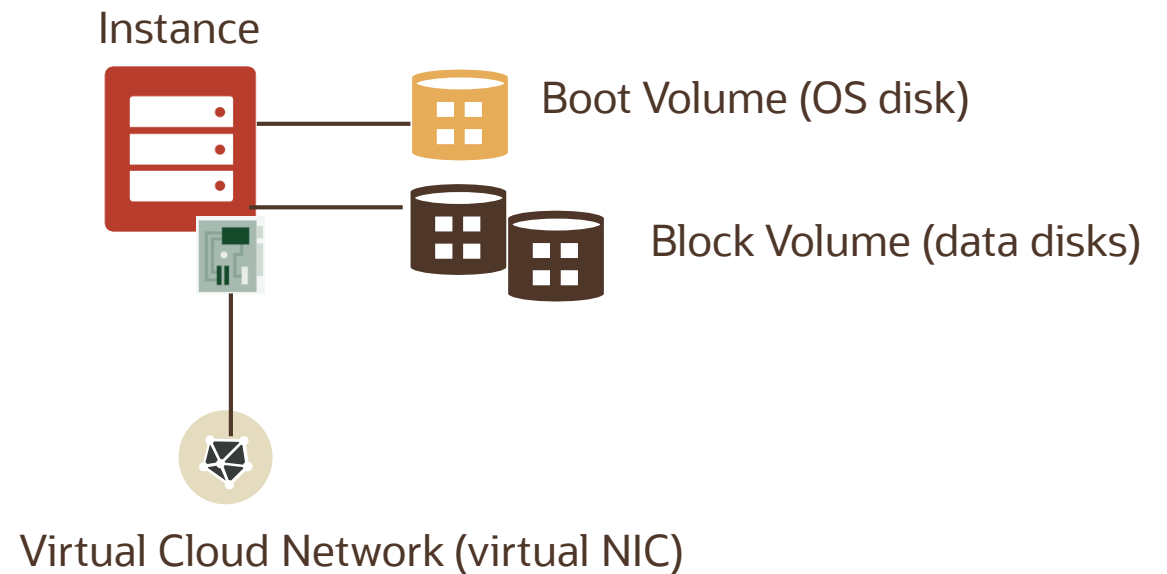
Instance basics

Various instance sizes for every workloads (CPU, RAM, Bandwidth)

Supports both Intel and AMD processors with industry leading price/performance

Provide GPU and HPC instance options

Instances are placed on virtual network with powerful connectivity options (incl. on-premises)

Compute instances depend on other OCI services such as Block Volume and VCN

Instance

Boot Volume (OS disk)

Block Volume (data disks)

Virtual Cloud Network (virtual NIC)

Instance basics
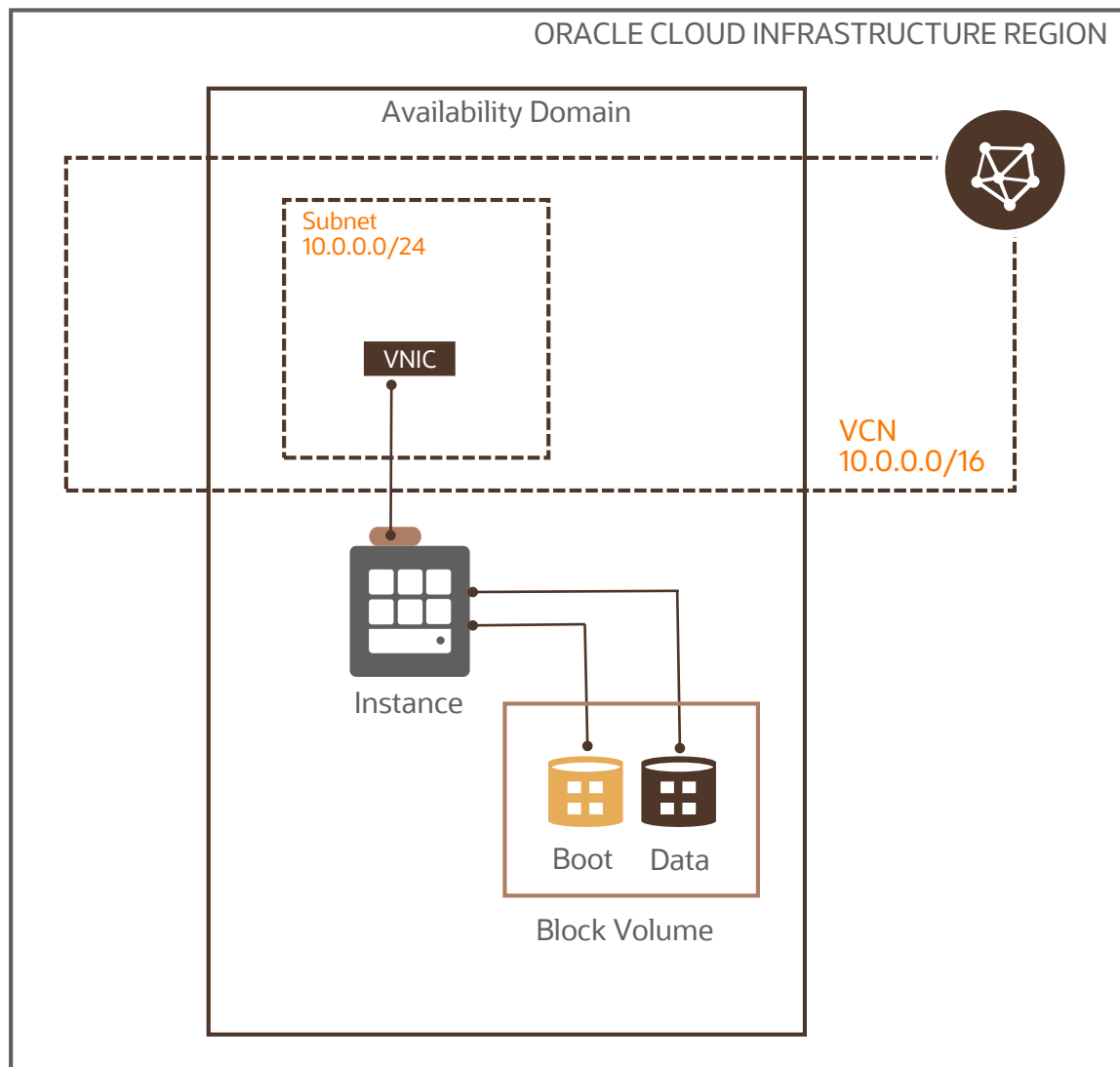
Vertical Scaling

- Scale-up and Scale-down instance shape supported

- New shape must have the some hardware architecture.

- Downtime is required. The instance must be stopped before resize it

Autoscaling

Enables large scale deployment of VMs from a single gold image with automatic configuration
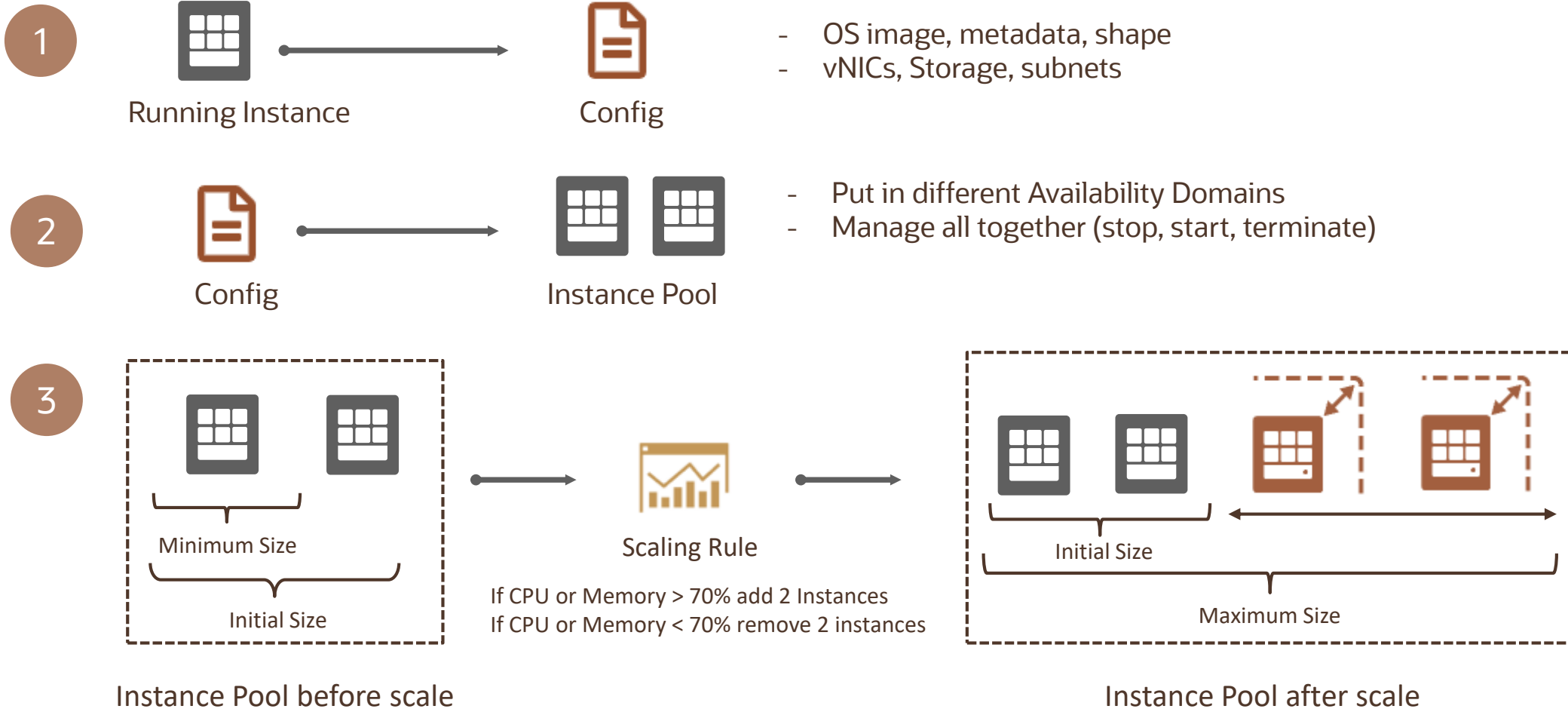
Referred to as scale-out or scale-in

If one VMs fails in the Autoscaling group, others will keep working

Match traffic demand by adding or removing VMs automatically (supports auto scaling based on metrics – CPU or Memory utilization

No extra cost for using Autoscaling

Autoscaling

1

Running Instance → Config

- OS image, metadata, shape
- vNICs, Storage, subnets

2

Config → Instance Pool

- Put in different Availability Domains
- Manage all together (stop, start, terminate)

3

Minimum Size

Initial Size

Instance Pool before scale

Scaling Rule

If CPU or Memory > 70% add 2 Instances
If CPU or Memory < 70% remove 2 instances

Initial Size

Maximum Size

Instance Pool after scale

Containers v/s VMs
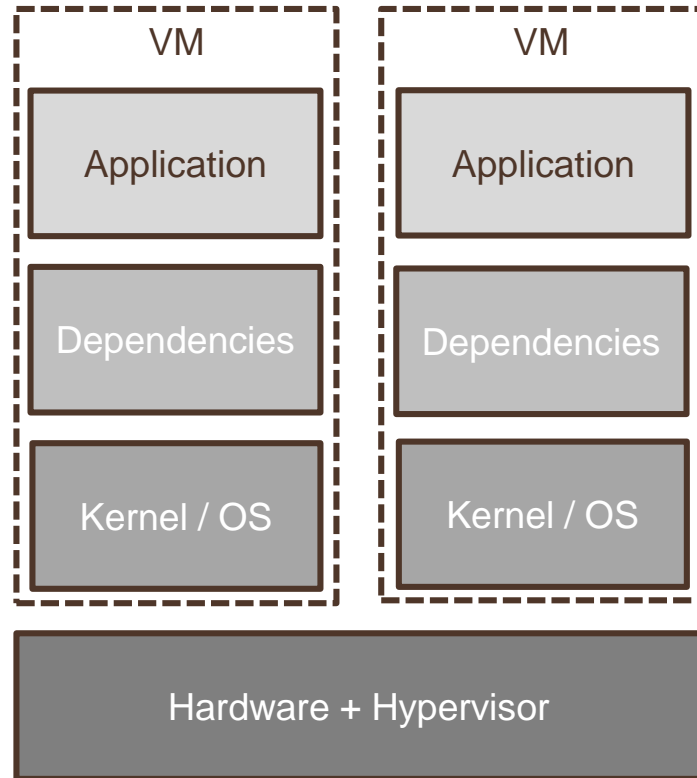
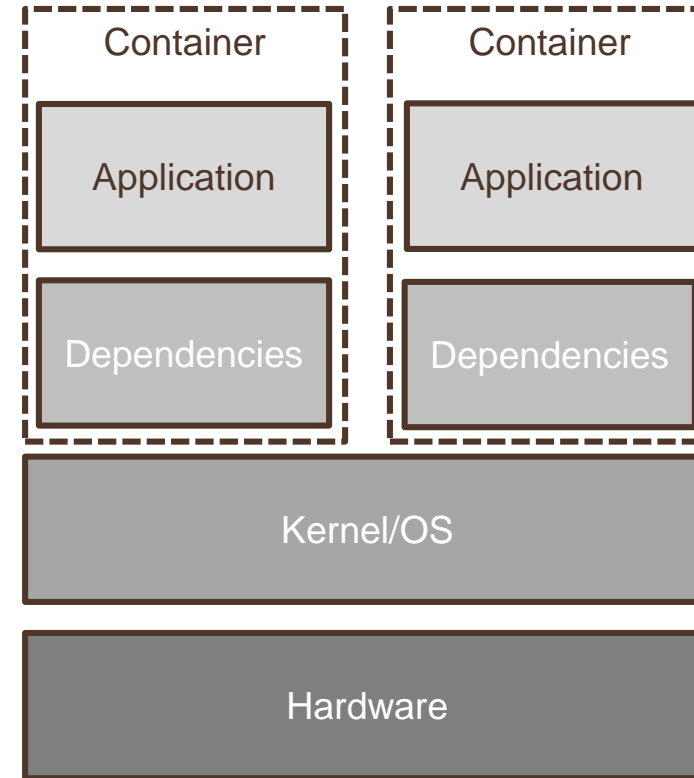Each virtual machine (VM) includes the app, the necessary binaries and libraries and an entire guest operating system

Containers include the app & all of its dependencies, but share the kernel/OS with other containers. Containers are not tied to any specific infrastructure and can run anywhere

How to deploy Containers?

- Manually SSH into machines and run Docker
    - Pro: Simple and easily understood
    - Con: Not automated, no reproducible, doesn't scale, doesn't self heal
- Scripting or config management tools
    - Pro: Integrates with existing environments, easily understood
    - Con: Doesn't scale, doesn't self heal, no scheduling mechanism
- Orchestration Systems
    - Pro: Automated, reproducible, self healing, scalable
    - Con: additional tooling and training required, some overhead

Oracle Kubernetes Engine

Kubernetes is an open source system for automating deployment, scaling and management of containerized applications

OKE is a fully-managed, scalable, and highly available service that you can use to deploy your containerized applications in OCI
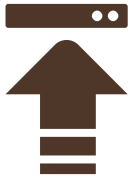
OCIR is a managed Docker container registry service and can be used to pull images for k8s deployments
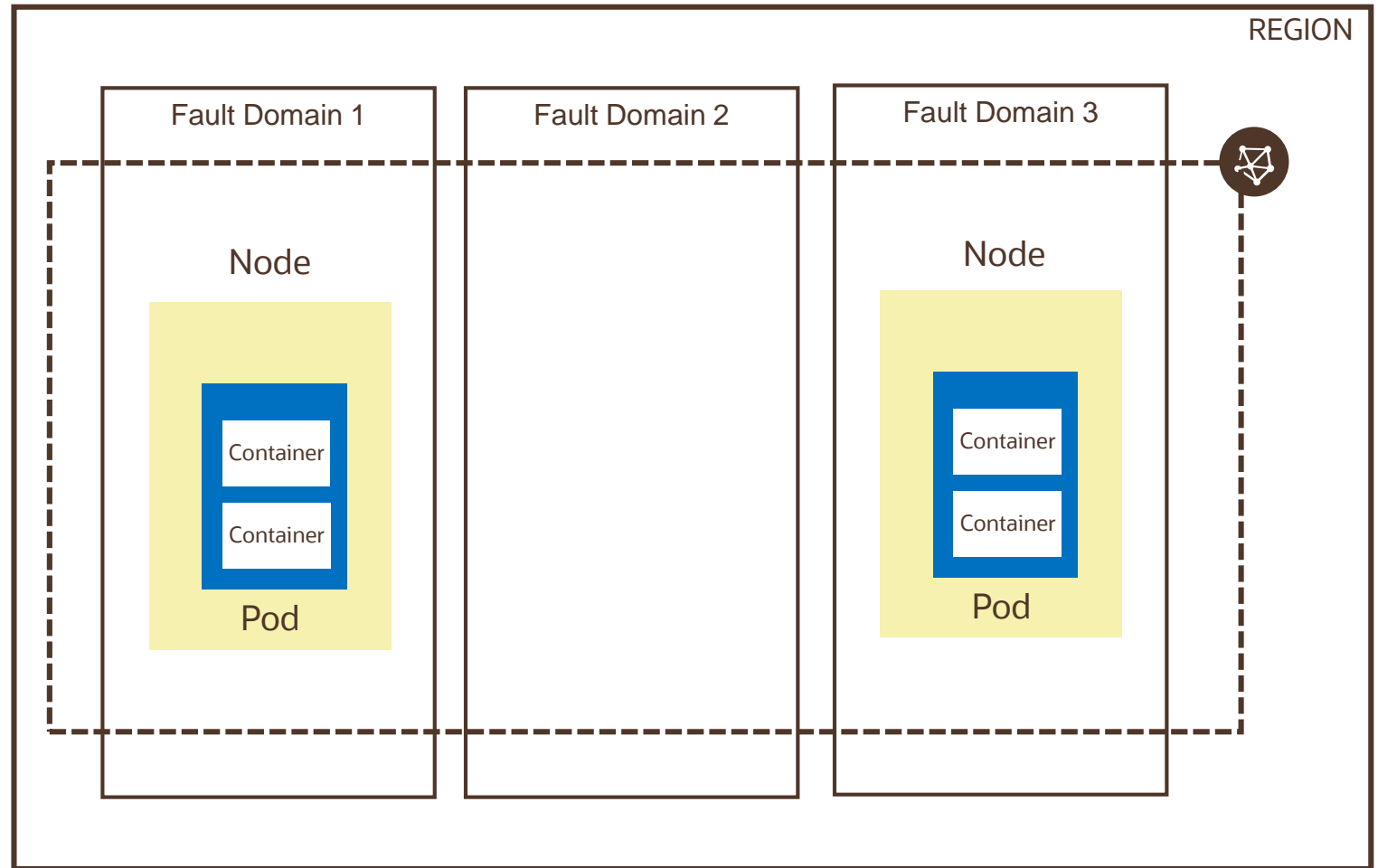
Functions

In Oracle Functions, functions are:

- small but powerful blocks of code that generally do one simple thing
- stored as Docker images in a specified Docker registry
- invoked in response to a CLI command or signed HTTP request

Push container
to registry

Configure function
trigger

Code runs only
when triggered

Pay for code
execution time only

Summary

Bare Metal

VMs

Scaling

Container Engine

Functions

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

Thank you