



# Oracle Cloud Infrastructure Cloud Operations Professional: Hands-on Workshop

Student Guide – Volume II

D1111263GC10

Learn more from Oracle University at [education.oracle.com](https://education.oracle.com)



**Copyright © 2025, Oracle and/or its affiliates.**

**Disclaimer**

This document contains proprietary information and is protected by copyright and other intellectual property laws. The document may not be modified or altered in any way. Except where your use constitutes "fair use" under copyright law, you may not use, share, download, upload, copy, print, display, perform, reproduce, publish, license, post, transmit, or distribute this document in whole or in part without the express authorization of Oracle.

The information contained in this document is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

**Restricted Rights Notice**

If this documentation is delivered to the United States Government or anyone using the documentation on behalf of the United States Government, the following notice is applicable:

**U.S. GOVERNMENT END USERS:** Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software" or "commercial computer software documentation" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

**Trademark Notice**

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

**Third-Party Content, Products, and Services Disclaimer**

This documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

1003042025

## Table of Contents

<b>Module 09: Terraform - Infrastructure as Code</b>	<b>14</b>
Infrastructure as Code	15
What is Infrastructure as Code?	16
Why Use Infrastructure as Code?	17
Objectives	18
Benefits and Overview	19
Terraform	20
Terraform Concepts	21
Terraform Commands	22
Terraform Commands	23
Variables	25
Variables	26
terraform.tfvars	28
Provider	29
Provider	30
Resources	34
Resources	35
Outputs	37
Outputs	38
Modules	41
Modules	42
State	44
State	45
Preparing The Environment	47
Terraform Setup	48
Parameters Evaluation Order	50
Provider Configuration	51
Environment Variables Linux	52

Environment Variables Windows	53
Security Token	54
Terraform Workflow	55
Your First Terraform Configuration	56
Terraform Configuration File Structure	57
main.tf	58
Change Infrastructure - Updating Your Configuration Files	59
Modifying Terraform Files	60
Splitting the Configuration	61
Incorporating Modules	62
Creating the Module	63
Creating the Module - Continued	64
Incorporating Modules	65
<b>Module 10: OCI Resource Manager</b>	<b>66</b>
Introduction and Concepts	67
Resource Manager Concepts	68
Configuration Source Providers	69
Configuration	71
Stacks	73
Actions	75
Jobs	77
Templates	79
All Together	81
Creating Your First Stack	82
Creating Your First Stack	83
Using Source Providers	84
Configuration Source Providers	85
Using a Configuration Source Provider	86
Importing Existing Infrastructure	87
Importing Existing Infrastructure	88
Drift Detection	89

Drift	90
Using Drift Detection	91
<b>Templates</b>	<b>92</b>
Templates	93
Creating Private Templates	94
<b>Remote Exec and Endpoints</b>	<b>95</b>
Resource Manager Endpoints	96
Creating the Endpoint with Terraform	97
<b>Module 11: Deploy a Monolithic Architecture</b>	<b>98</b>
Case Study Architecture	99
Case Study: Mastodon	100
Instance Architecture	104
Mastodon Architecture	112
VCN Deep Dive: Gateways and Routing	113
Virtual Networking	114
Conceptual Design	115
CIDR Blocks	116
Conceptual Design	118
VCN Gateways	123
VCN Layout and Gateways	131
Demo: Setting up a VCN	132
Demo: Stack Creation	133
Demo: VCN Creation Terraform	134
VCN Deep Dive: Access Control	135
Virtual Networking	136
VCN Access Control	137
Stateful rules allow responses	150
Demo: Securing a VCN	153
Compute Deep Dive: The Instance Life Cycle	154
Recap...	155
Next...	156

Instance Life Cycle	157
Demo: Provision a compute instance with Terraform	158
Compute Deep Dive: Provisioning and Sourcing	159
Instance Lifecycle	160
Approach 1 Our example: Redis	165
Approach 2 Our example: PostgreSQL	166
Approach 3 Our example: Ruby on Rails	167
Approach 1 Our example: Redis	168
Example Workflow 1	169
Example Workflow 2	170
Example Workflow 3	171
Example Workflow 1	172
1 Provisioning	173
2 Source	178
Compute Deep Dive: Bootstrapping with Cloud-init	181
Instance Life Cycle	182
Approach 1 Our example: PostgreSQL123	185
Approach 2 Our example: Redis	186
Approach 3 Our example: Ruby on Rails	187
Example Workflow 1	188
Example Workflow 2	190
Bootstrapping	191
Compute Deep Dive:Fine-tuning with Ansible	193
Instance Life Cycle	194
Approach 1 Our example: PostgreSQL	197
Approach 2 Our example: Redis	198
Approach 3 Our example: Ruby on Rails	199
Example Workflow 2	200
Example Workflow 3	202
Fine-tuning	203
File Storage Deep Dive	208
Object Storage Deep Dive	218

Object Storage Bucket	221
Versioning	225
Lifecycle Management	235
<b>Module 12: Secrets and Encryption</b>	<b>240</b>
OCI Key Management Service (KMS)	241
OCI Encryption Options	242
OCI KMS encryption portfolio	245
Choosing the right OCI KMS offering	246
OCI KMS offers	247
Encryption Basics	248
Encryption at rest and in-transit	250
Symmetric Encryption	251
Asymmetric Encryption	252
Encryption Concepts	253
Hardware Security Module (HSM)	254
Vault Introduction	255
OCI Vault	256
Vaults	257
Keys	258
Master and Data Encryption Keys	259
Master Encryption Keys: Protection Modes	260
Wrapping Keys	261
Rotating Keys	262
Demo: Vault Basics Part 1	263
Demo: Vault Basics Part 2	264
Import and Export Keys	265
Cryptographic and Management Endpoints	266
Cryptographic and Management Endpoints	267
Crypto Operations	268
Importing Keys or Key Versions	269
Exporting Keys or Key Versions	270

OCI Services Integration with Vault	271
Encryption Using Oracle-Managed Keys	273
Encryption Using Customer-Managed Keys	274
OCI Object Storage Integration with Vault	275
Back up and Replicate Vaults and Keys	276
Backing Up Vaults and Keys	277
Restoring Vaults and Keys	279
Cross-Region Replication	280
Demo: OCI services integration with Vault	281
Secrets	282
What's a Secret?	283
Secrets	284
Secrets Rules	286
Demo: Secrets	287
<b>Module 13: Disaster Recovery</b>	<b>288</b>
High Availability	289
High Availability Concepts	290
Availability Domains	291
Fault Domains	292
Avoiding Single Points of Failure	293
Regional and AD-Specific Subnets	294
Load Balancer	295
Virtual IP	296
Compute	297
Compute: Autoscaling	299
Storage: Object Storage	300
Storage: Block Volume	301
Storage: File Storage	302
High Availability for OCI: Connectivity	303
IPSec VPN Redundancy Models (Multiple CPE)	304
Redundant FastConnect	305

Demo: Secrets	306
Demo: High Availability Workshop Part 02	307
Disaster Recovery	308
Disaster Recovery Terminology	309
Disaster Recovery RTO and RPO	310
Disaster Recovery Options	311
Backup and Restore Architecture	312
Standby Architecture	313
Active/Active Architecture	314
Disaster Recovery for OCI	315
Disaster Recovery Using Multiple Regions	316
Disaster Recovery Using Multiple Regions	317
Database Strategies for DR	318
Overview	319
Disaster Recovery Operational Challenges in OCI	320
How a typical DR runbook	321
Full Stack DR orchestrates recovery with a single click	323
Recovery made easy for many business systems	324
Capitalize on your existing effort	325
Flexible, highly scalable, highly extensible and customizable	326
Recovery point and recovery time objectives	327
Core Concepts	328
FSDR components and concepts	333
Sample Scenario	334
DR Protection Group	335
Peer Association	337
Members	338
DR Plans	340
DR Groups	342
DR Plan Groups	344
DR Plans	345

Failover in Action	346
Switchover in Action	347
Start Drill in Action	348
Stop Drill in Action	349
Requirements	350
Movable instance vs Non-movable instance	351
Preparing for Full Stack Disaster Recovery	352
Demo –Setup	359
Preparing Mushopfor Full Stack DR	360
MuShopScenario after deployment	361
Demo –DR plan Pre-Check and execution	362
MuShopScenario after plan setup	363
MuShopScenario after Switchover	364
<b>Module 14: Troubleshooting</b>	<b>365</b>
Troubleshooting	365
Oracle Cloud Infrastructure Troubleshooting	366
Objectives	367
SSH Connection	368
Instance Console Connections	369
Troubleshooting Performance	370
Oracle Cloud Infrastructure Troubleshooting	371
Objectives	372
IPSec connection testing	373
FastConnect Redundant Connections	374
Load Balancer Health Status	375
Health Check	376
Oracle Cloud Infrastructure Troubleshooting	377
Objectives	378
Block Storage Backup Copy – Common Errors	379
Block Storage Recovery steps	380
Block Storage Multi-Attach	381

Block Storage Volume Resize	382
<b>Local NVMe Device Failures</b>	<b>383</b>
Local NVMeDevice	384
RAID with Local NVMeDevice	385
When a Device Fails	386
What if the Availability Domain fails?	387
Backups to Block Storage/File Storage	388
Replicate to another Compute Instance	389
<b>Troubleshoot and Attach Orphaned Mount Targets</b>	<b>390</b>
Mount Target and File Systems	391
Troubleshoot File Systems	393
<b>Module 15: Observability &amp; Management</b>	<b>396</b>
What is Observability	397
Traditional Monitoring	398
Challenges with Traditional Monitoring	399
Definition: Observability	400
Comparing Monitoring and Observability	401
Introducing Observability and Management Services	402
Observability & Management Services	403
Use Case: Observability and Management in DevOps	404
Monitoring Service Overview	405
OCI Monitoring Service: Getting Started	406
Monitoring Capabilities	407
Monitoring Service Workflow	408
Demo: Monitoring Concepts	409
Monitoring Concepts	410
Metrics	411
Intervals and Resolutions	412
Statistics	413
Alarms	414
Metric Query Components	415

Notifications Service .....	416
Overview .....	417
Notifications Service:Creating a Topic .....	418
Demo: Notifications Service .....	419
Alarms .....	420
Alarms Workflow .....	421
Best Practices .....	422
Demo: Alarms .....	423
Access and Limits .....	424
Ways to Access Monitoring .....	425
IAM Policies for Access .....	426
IAM Policies with Restricted Access .....	427
Limits of Monitoring Service .....	428
Metric Queries .....	429
Building Metric Queries .....	430
Sample Queries .....	431
Nested Queries .....	432
Demo: Metric Queries .....	433
Logging Service: Overview .....	434
OCI Logging Service .....	435
Types of Logs .....	436
Service Flow .....	437
Logging Concepts .....	438
Log Groups .....	439
Logging Concepts .....	440
Service Logs .....	441
Service Log Format .....	442
Object Storage Logs .....	443
Load Balancer Logs .....	444
VCN Flow Logs .....	445
Demo: Service Logs .....	446

Custom Logs	447
Custom Log Ingestion	448
Using Unified Monitoring Agent	449
Agent Communication Workflow	450
Agent Configuration	451
Demo: Custom Logs	452
Access & Explore Logs	453
IAM Policies	454
Searching Logs	455
Viewing Log Events	456
Logging Queries	457
Log Search	458
Logging Query Specification	459
Log Streams	460
Fields	461
Data Types	462
Tabular Operators	463
Scalar Operators	464
Demo: Logging Queries	465
Connector Hub	466
Overview and Key Concepts	467
Connectors Workflow	468
Take Actions for Use Cases	469



# Terraform- Infrastructure as Code

## Oracle Cloud Infrastructure

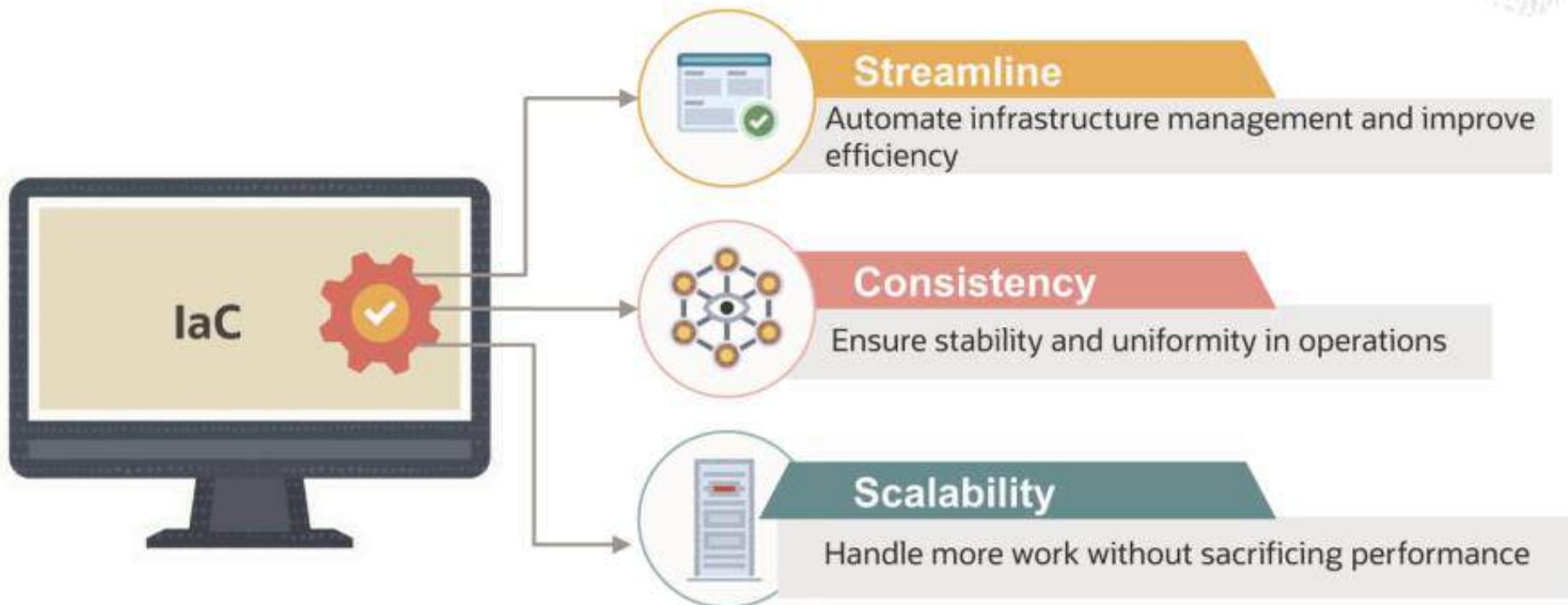
# Infrastructure as Code

### What is Infrastructure as Code?

## What is Infrastructure as Code?



# Why Use Infrastructure as Code?



# Objectives

A stylized illustration of a person in an orange shirt and brown pants climbing a green mountain. A white grid path leads up the mountain. Above the mountain is a large, colorful cloud composed of various patterns like wood grain and abstract shapes in shades of brown, orange, and yellow. A small white airplane is flying through the clouds above the mountain.

Terraform Basics

Terraform with OCI

OCI Resource Manager

## Oracle Cloud Infrastructure

# Understanding Terraform Concepts

### Benefits and Overview

# Terraform Benefits



# Terraform Concepts

Terraform Commands

Provider

Resources

Variables

Outputs

Modules

State



Oracle Cloud Infrastructure

# Understanding Terraform Concepts

## Terraform Commands



# Terraform Commands

---

# Terraform Commands



## version



Show the current Terraform version

## init



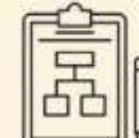
Prepare your working directory for other commands

## validate



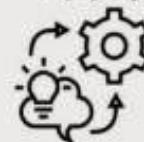
Check whether the configuration is valid

## plan



Show changes required by the current configuration

## apply



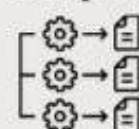
Create or update infrastructure

## destroy



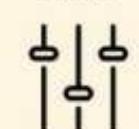
Destroy previously created infrastructure

## output



Show output values from your root module

## fmt



Reformat your configuration in the standard style

Oracle Cloud Infrastructure

# Understanding Terraform Concepts

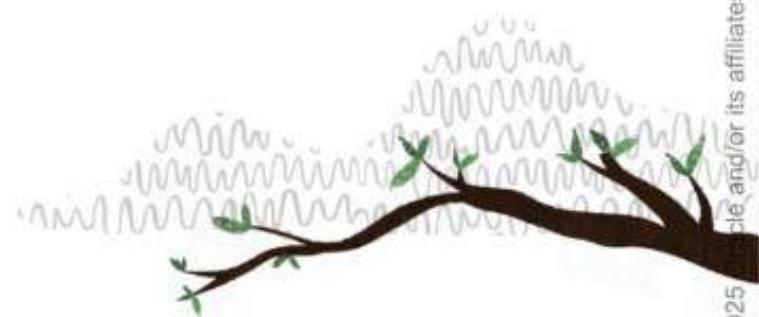
## Variables



# Variables

---

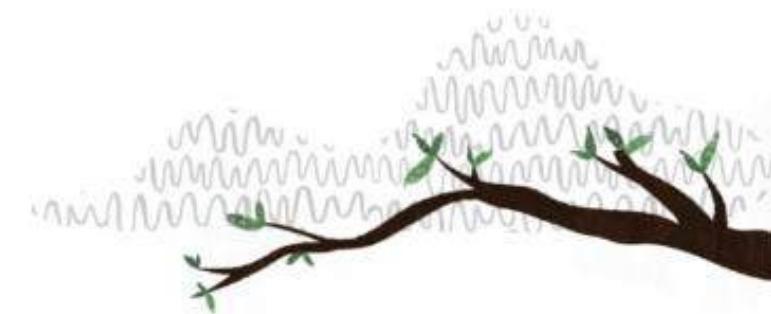
# Variables



```
variable "compartment_id" {
    description = "OCI Compartment OCID"
    type        = string
    default     = "default_compartment_id"
    validation {
        condition = length(var.compartment_id) > 0
        error_message = "Compartment ID must be provided."
    }
}
variable "availability_domain" {
    description = "OCI Availability Domain Name"
}
```

```
resource "oci_core_instance" "example_instance" {
    availability_domain = var.availability_domain
    compartment_id      = var.compartment_id
    shape               = "VM.Standard2.1"
    display_name        = "ExampleInstance"
    # Other configuration settings...
}
```

# terraform.tfvars



```
# terraform.tfvars  
compartment_id = "your_compartment_id_here"  
availability_domain = "your_availability_domain_here"
```

```
resource "oci_core_instance" "example_instance" {  
    compartment_id      = var.compartment_id  
    availability_domain = var.availability_domain  
    shape               = "VM.Standard2.1"  
    display_name        = "ExampleInstance"  
    # Other configuration settings...}
```

Oracle Cloud Infrastructure

# Understanding Terraform Concepts

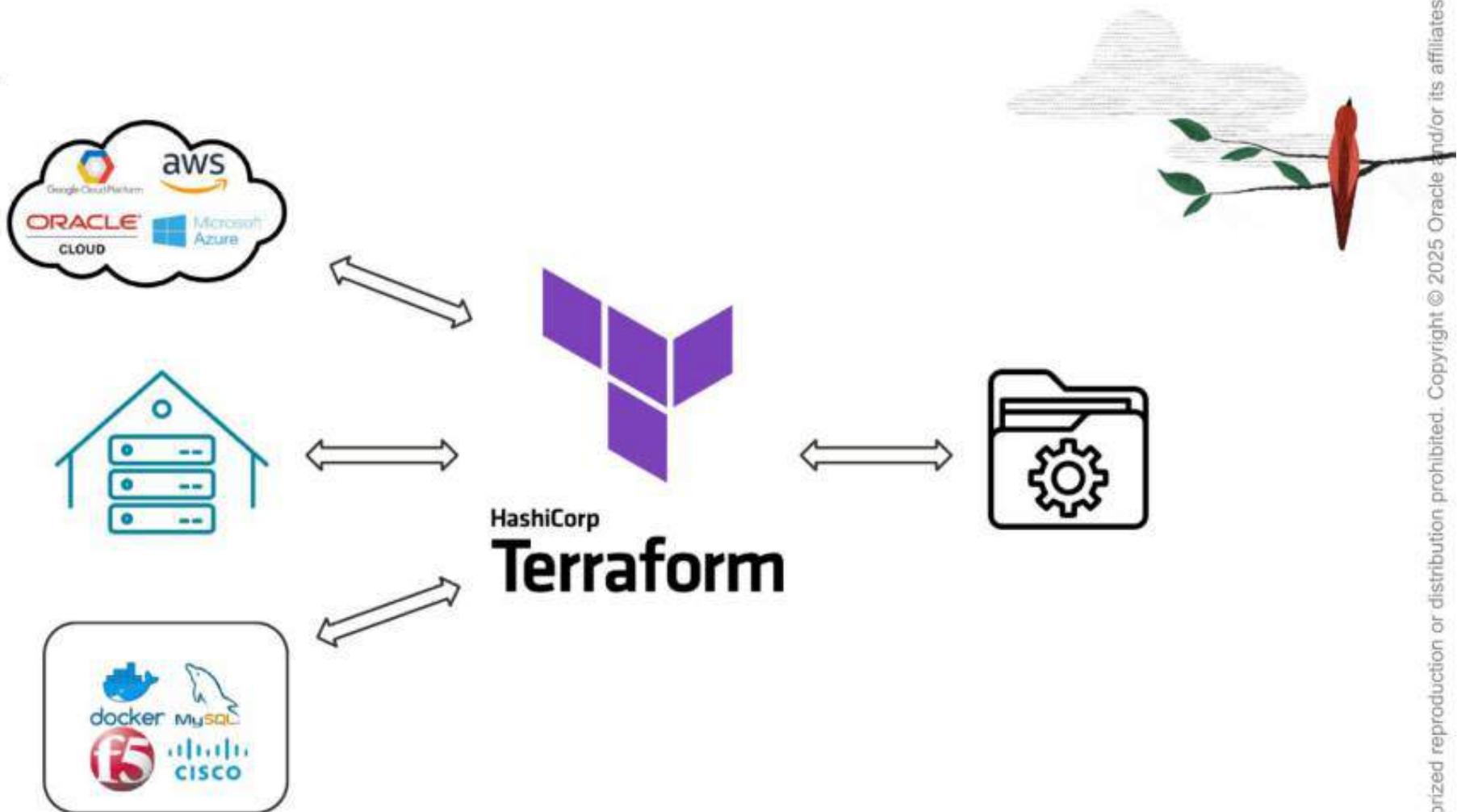
—  
**Provider**



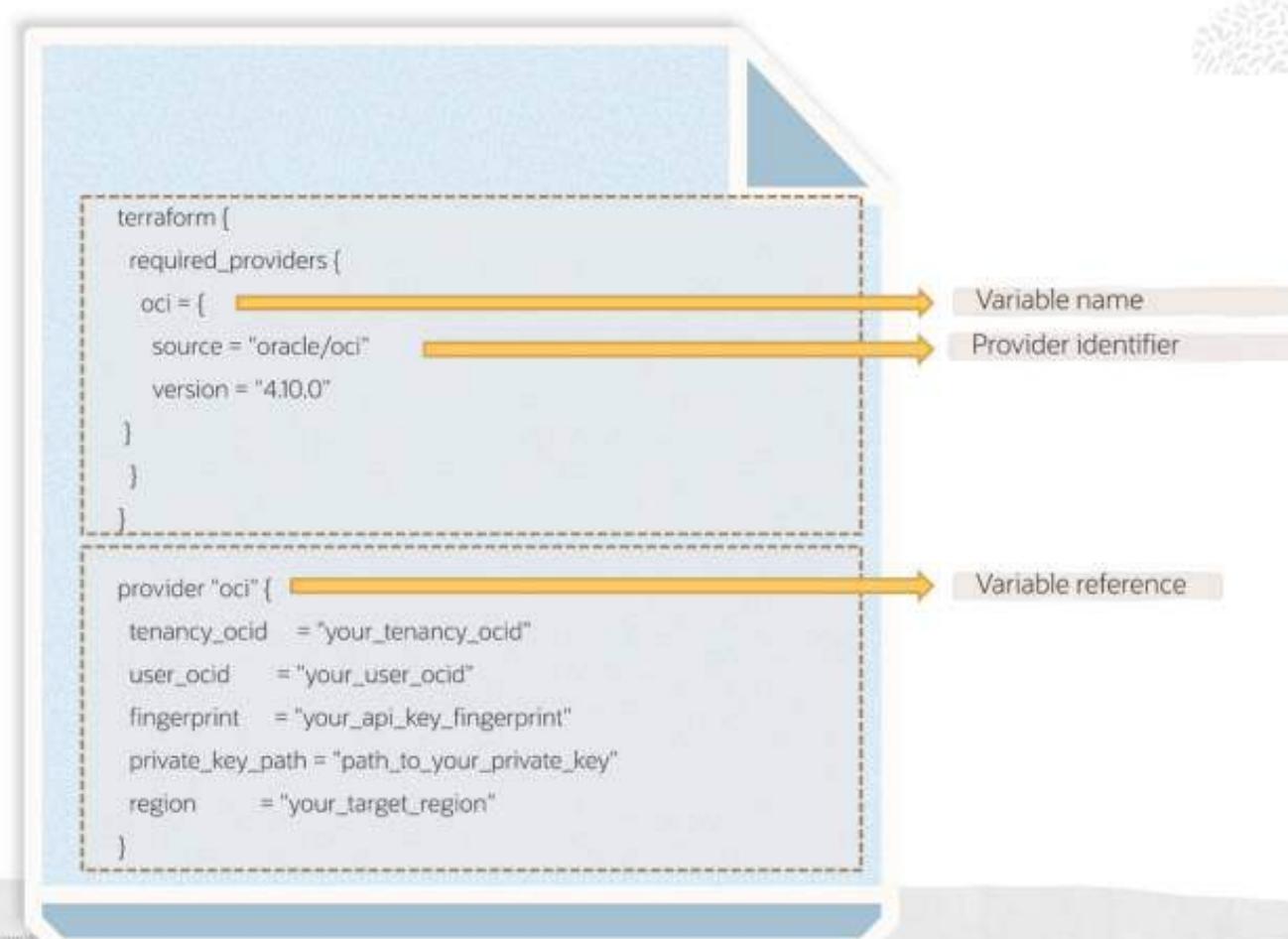
# Provider

---

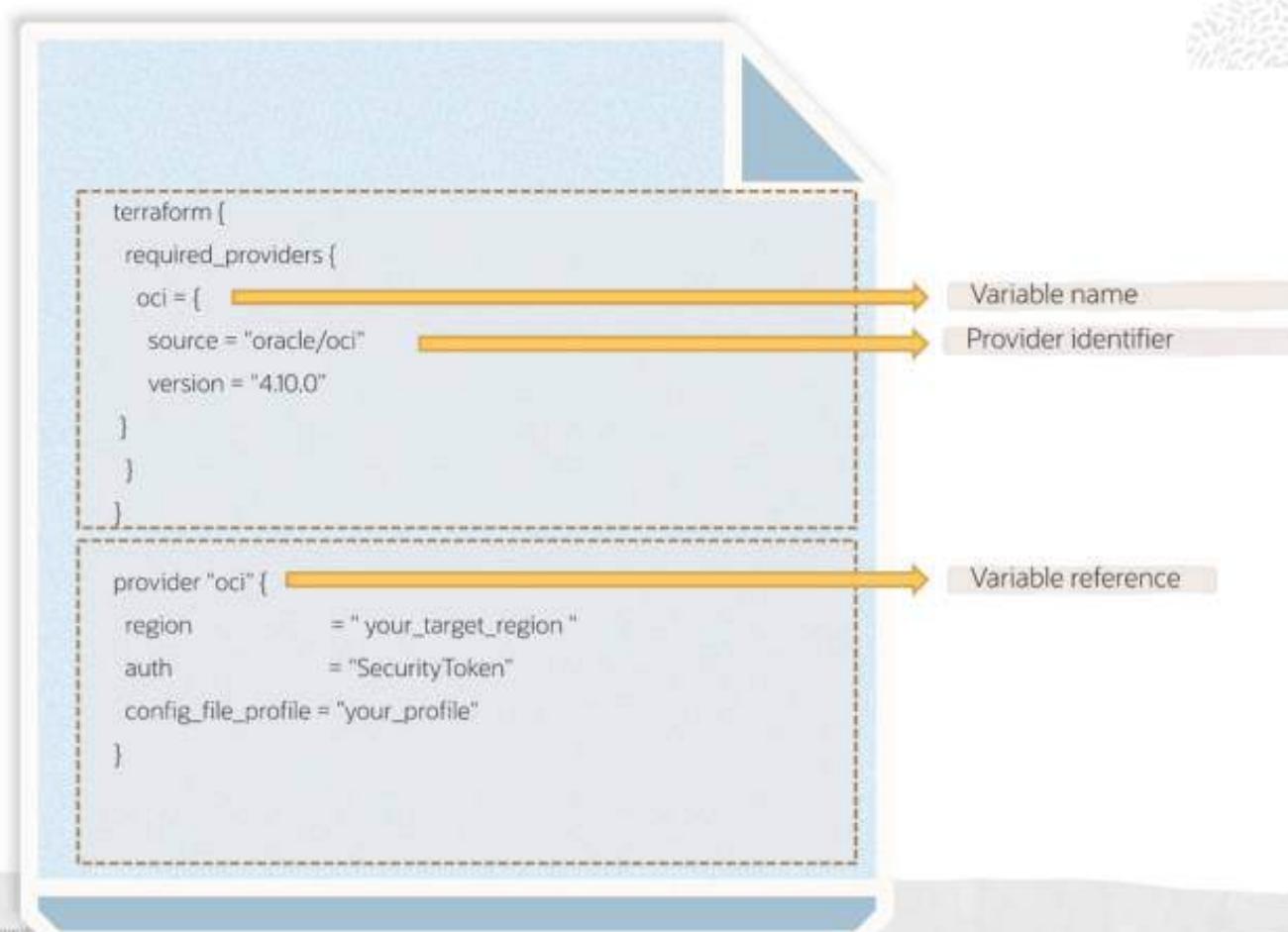
## Provider



# Provider



# Provider



## Oracle Cloud Infrastructure

# Understanding Terraform Concepts

### Resources



# Resources

---

# Resources

---

- Resource block
  - Resource type
  - Resource name
- Configurations
  - Attributes
    - Attribute name
    - Attribute value

```
resource "oci_core_instance" "example_instance" {  
    availability_domain = "instance_availability_domain"  
    compartment_id      = "instance_compartment_id"  
    shape               = "VM.Standard2.1"  
    display_name        = "ExampleInstance"  
    image_id            = "instance_image_id"  
    subnet_id           = "target_subnet_id"  
  
    metadata {  
        ssh_authorized_keys = "your_ssh_public_key"  
    }  
}
```

<https://registry.terraform.io/providers/oracle/oci/latest>

Oracle Cloud Infrastructure

# Understanding Terraform Concepts

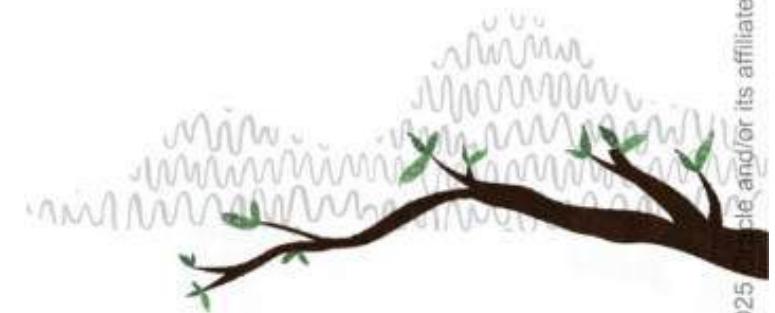
## Outputs



# Outputs

---

# Outputs

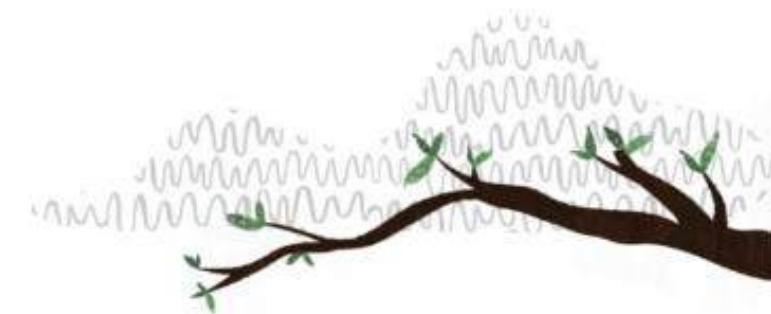


```
resource "oci_core_subnet" "example_subnet" {
    cidr_block = var.subnet_cidr_block
    compartment_id = var.compartment_id
    vcn_id = oci_core_vcn.test_vcn.id
}

output "subnet_id" { value =
oci_core_subnet.example_subnet.id }
```

```
resource "oci_core_instance" "example_instance" {
    availability_domain = "your_availability_domain"
    compartment_id      = "your_compartment_id"
    shape               = "VM.Standard2.1"
    display_name        = "ExampleInstance"
    image_id            = "your_image_id"
    # Using output from another resource
    subnet_id           = oci_core_subnet.example_subnet.id
}
```

# Outputs



```
resource "oci_core_subnet" "example_subnet" {
    cidr_block = var.subnet_cidr_block
    compartment_id = var.compartment_id
    vcn_id = oci_core_vcn.test_vcn.id
}

output "subnet_id" { value =
oci_core_subnet.example_subnet.id }
```

```
resource "oci_core_instance" "example_instance" {
    availability_domain = "your_availability_domain"
    compartment_id      = "your_compartment_id"
    shape               = "VM.Standard2.1"
    display_name        = "ExampleInstance"
    image_id            = "your_image_id"
    # Using output from another resource
    subnet_id           = oci_core_subnet.example_subnet.id
}
```

## Oracle Cloud Infrastructure

# Understanding Terraform Concepts

### — Modules



# Modules

---

# Modules



```
# modules/compute-instance/main.tf
```

```
variable "instance_name" {  
    description = "Name of the compute instance"  
}
```

```
variable "image_id" {  
    description = "OCI image ID for the instance"  
}
```

```
variable "subnet_id" {  
    description = "OCI subnet ID for the instance"  
}
```

```
resource "oci_core_instance" "example_instance" {  
    availability_domain = var.availability_domain  
    compartment_id     = var.compartment_id  
    shape              = var.instance_shape  
    display_name       = var.instance_name  
    image_id           = var.image_id  
    subnet_id          = var.subnet_id  
}
```

```
# Other configurations...  
}
```

```
# main.tf
```

```
module "example_instance" {  
    source = "./modules/compute-instance"  
  
    instance_name = "ExampleInstance"  
    image_id     = "your_image_id"  
    subnet_id    = "your_subnet_id"  
}
```

## Oracle Cloud Infrastructure

# Understanding Terraform Concepts

### State



# State

---

# State

---

- `terraform.tfstate`
  - State tracking
  - Resource dependency
  - Resource attributes
  - Locking mechanism
  - Sensitive data
  - Remote state backends



```
{  
  "version": 4,  
  "terraform_version": "1.8.2",  
  "serial": 1,  
  "lineage": "2dddb0cb-ee0e-b81f-88c1-5b969f147dbc",  
  "outputs": {},  
  "resources": [  
    {  
      "mode": "managed",  
      "type": "oci_core_vcn",  
      "name": "internal",  
      "provider":  
        "provider[\"registry.terraform.io/oracle/oci\"]",  
      "instances": [  
        {  
          "schema_version": 0,  
          "attributes": {  
            "byoipv6cidr_blocks": [],  
            "byoipv6cidr_details": null,  
            "cidr_block": "172.16.0.0/20",  
            "cidr_blocks": [  
              "172.16.0.0/20"  
            ]  
          }  
        }  
      ]  
    }  
  ]  
}  
: press SPACE to continue
```

Oracle Cloud Infrastructure

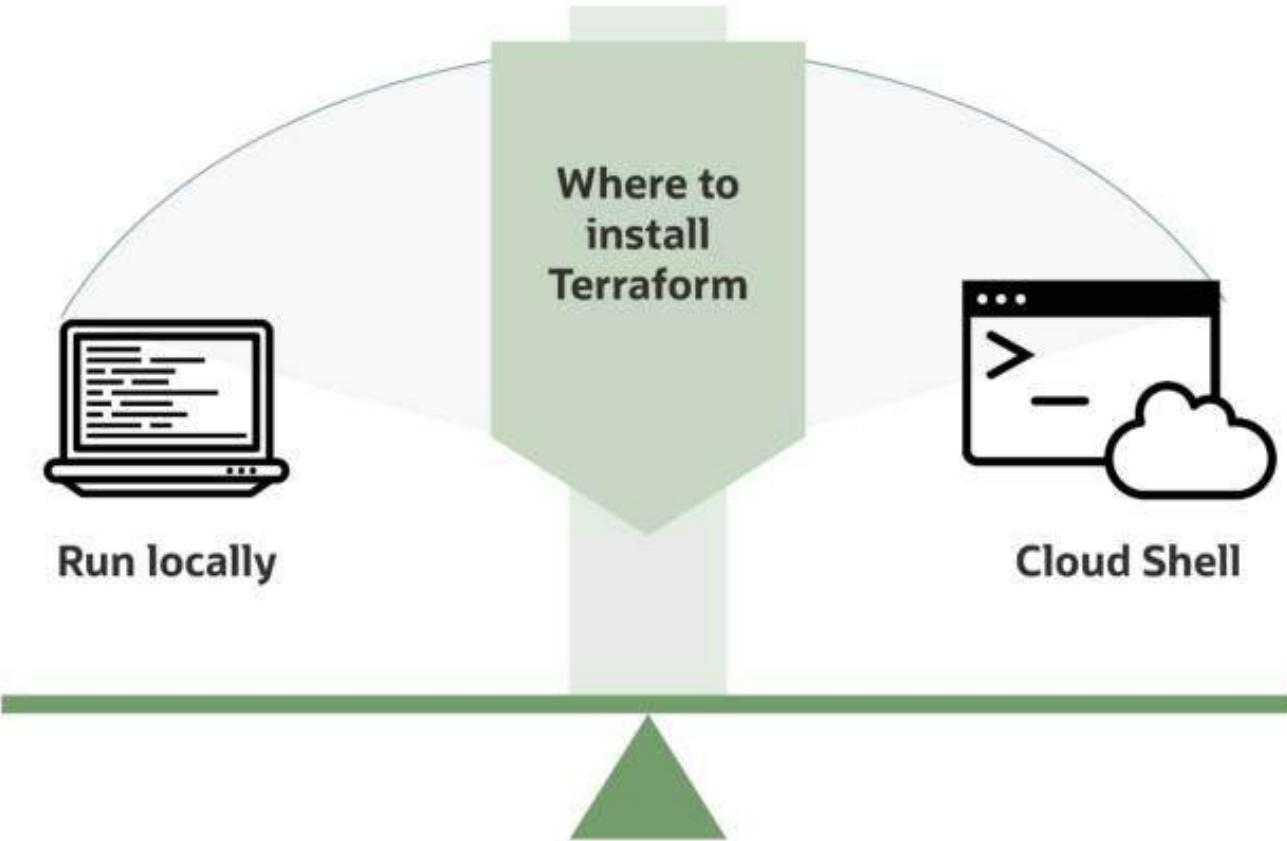
# Your First Terraform Configuration

## Prepared The Environment

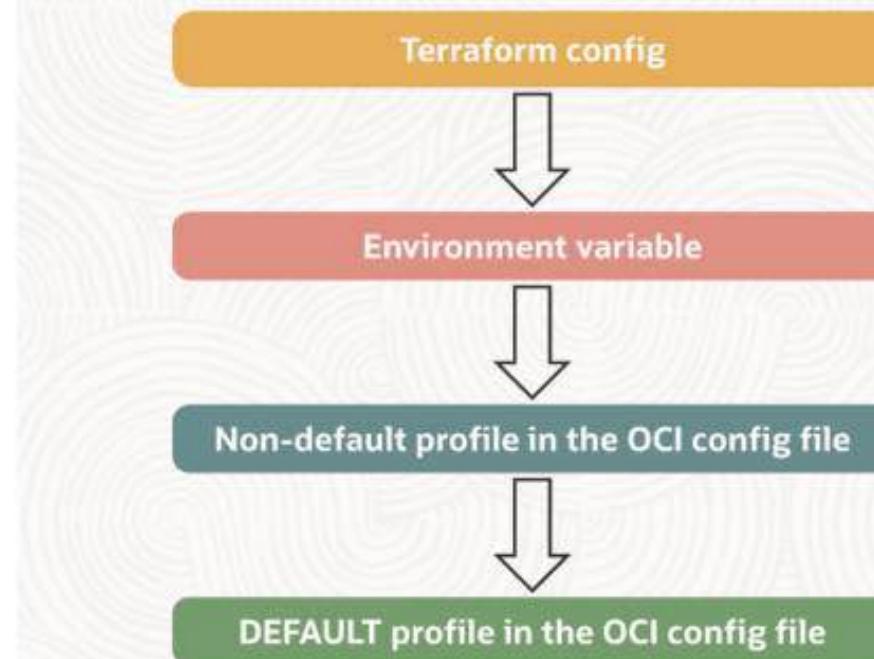


# Terraform Setup

---



# Parameters Evaluation Order



# Provider Configuration



```
provider "oci" {  
    tenancy_ocid      = var.tenancy_id  
    user_ocid         = var.user_id  
    private_key_path = "~/.ssh/priv.key"  
    fingerprint       = A1:B2:C3:D4:E5:F6"  
    region            = "ca-toronto-1"  
}
```

# Environment Variables

## Linux



```
export OCI_DEFAULT_CERTS_PATH=<certificates_path>
export TF_VAR_tenancy_ocid=<tenancy_OCID>
export TF_VAR_compartment_ocid=<compartment_OCID>
export TF_VAR_user_ocid=<user_OCID>
export TF_VAR_fingerprint=<key_fingerprint>
export TF_VAR_private_key_path=<private_key_path>
export TF_VAR_region=<region>
export SER_AGENT_PROVIDER_NAME=<custom_user_agent>
export OCI_SDK_APPEND_USER_AGENT=<custom_user_agent>
export TF_APPEND_USER_AGENT=<custom_user_agent>
```

# Environment Variables

## Windows



```
setx OCI_DEFAULT_CERTS_PATH=<certificates_path>
setx TF_VAR_tenancy_ocid <tenancy_OCID>
setx TF_VAR_compartment_ocid <compartment_OCID>
setx TF_VAR_user_ocid <user_OCID>
setx TF_VAR_fingerprint <key_fingerprint>
setx TF_VAR_private_key_path <private_key_path>
setx TF_VAR_region=<region>
setx USER_AGENT_PROVIDER_NAME=<custom_user_agent>
setx OCI_SDK_APPEND_USER_AGENT=<custom_user_agent>
setx TF_APPEND_USER_AGENT=<custom_user_agent>
```

# Security Token



```
provider "oci" {  
    region      = "<region>"  
    auth        = "SecurityToken"  
    config_file_profile = "my-profile"  
}
```

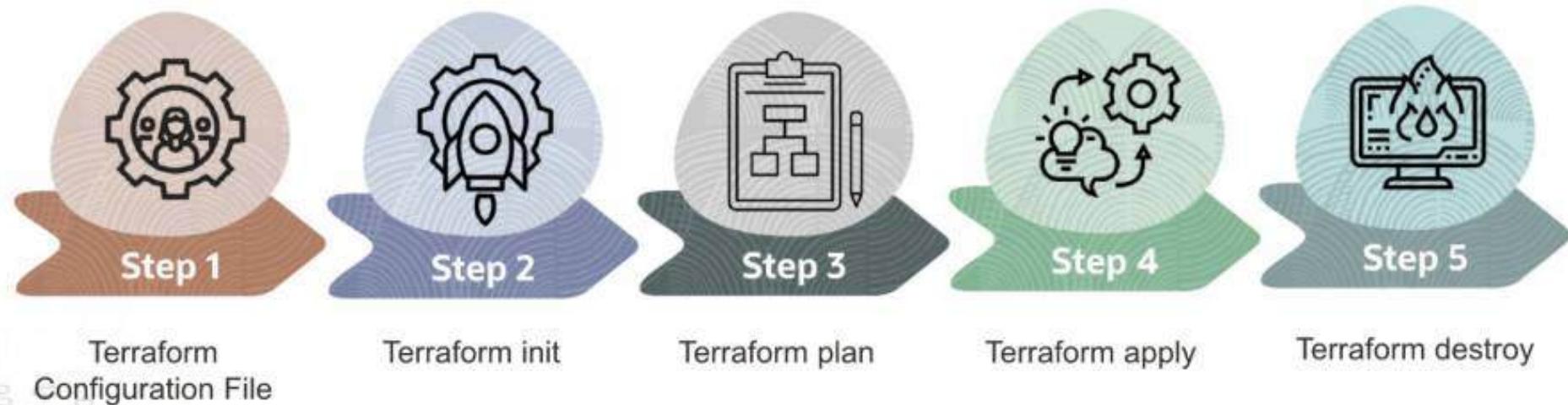
When you authenticate, it will create a profile, that will be added to the terraform command

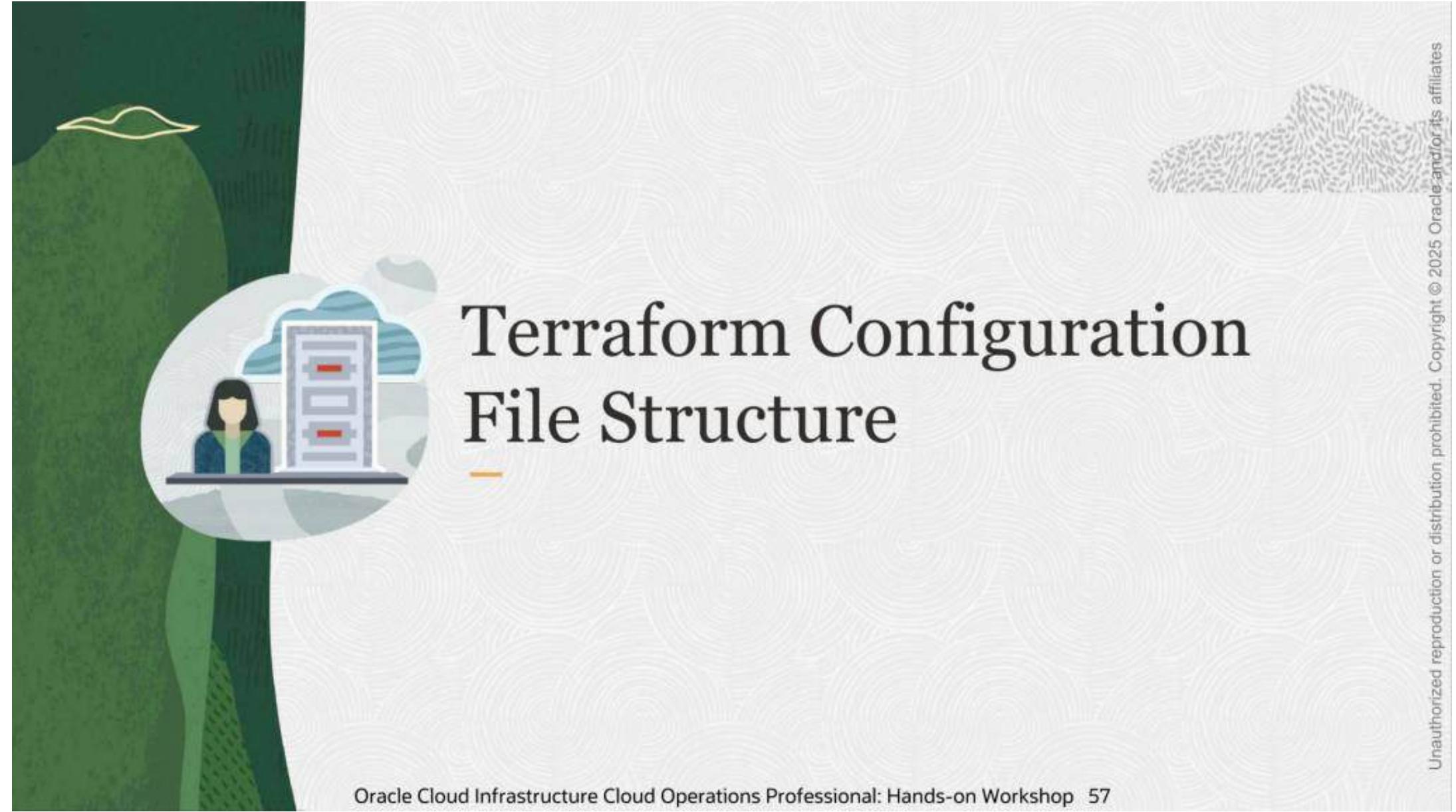
Oracle Cloud Infrastructure

# Your First Terraform Configuration

## Terraform Workflow

# Your First Terraform Configuration



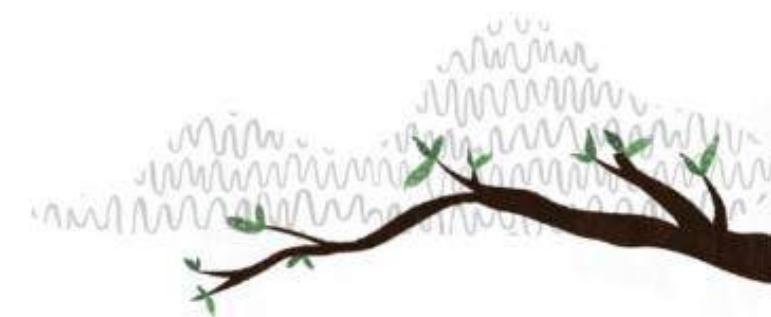


# Terraform Configuration File Structure

---

# main.tf

```
terraform {  
    required_providers {  
        oci = {  
            source = "oracle/oci"  
        }  
    }  
}  
  
provider "oci" {  
    # tenancy_ocid = "<tenancy OCID>"  
    # user_ocid = " ocid1.compartment.oc1..aaaaa"  
    # private_key_path = "~/keys/priv.pem"  
    # fingerprint = "AA:11:BB:22"  
    region = "<region name>"  
}  
  
resource "oci_core_virtual_network" "vcn01" {  
    compartment_id = "ocid1.compartment.oc1..aaaaa"  
    cidr_block = "10.0.0.0/16"  
    dns_label = "vcn01"  
    display_name = "vcn01"  
}
```



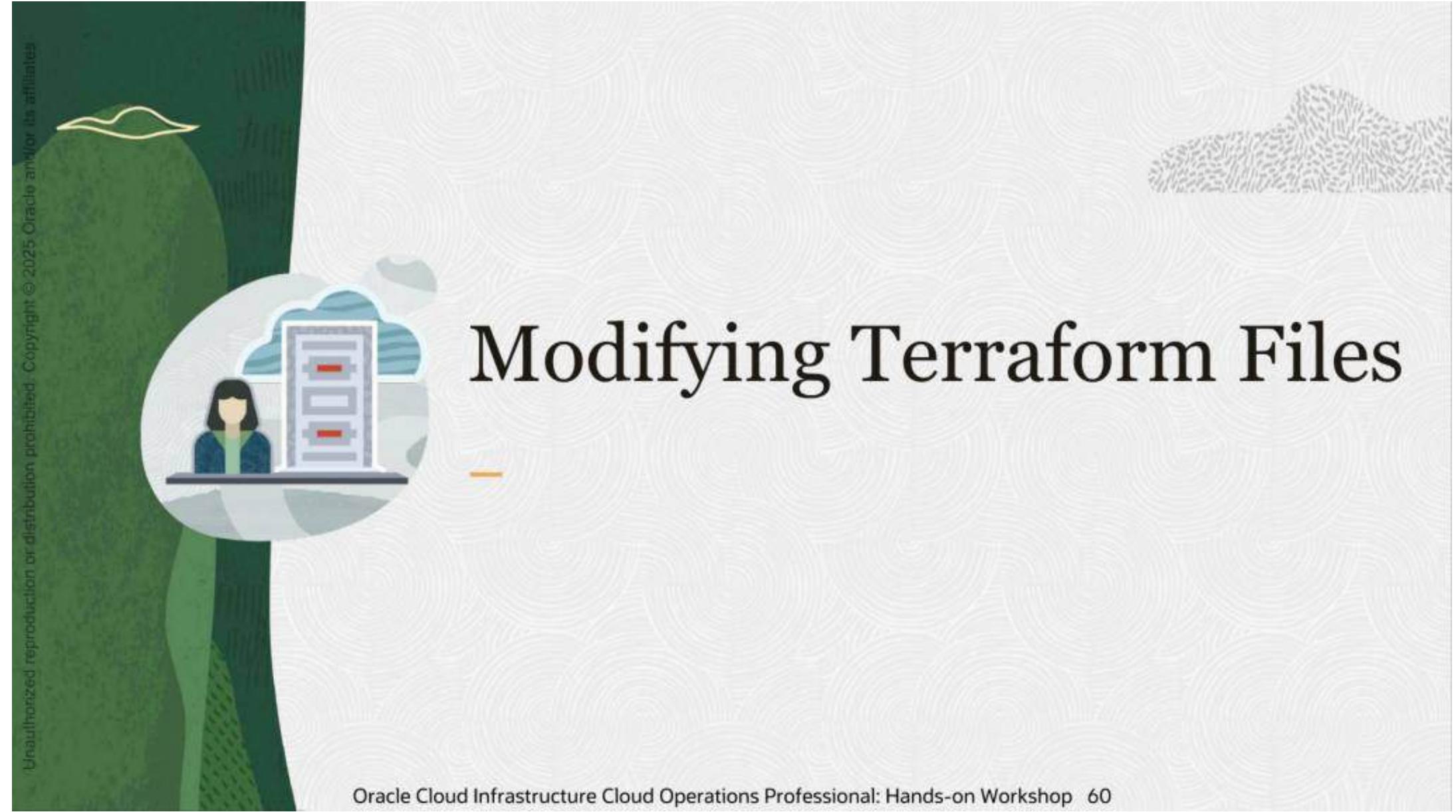
All the blocks can be on a single **.tf** file.

If you are not using Cloud Shell,  
uncomment the lines in the  
provider block and adjust the  
values

## Oracle Cloud Infrastructure

# Your First Terraform Configuration

### Change Infrastructure - Updating Your Configuration Files



# Modifying Terraform Files

---

# Splitting the Configuration

```
# variables.tf

variable "region" {
  description = "region where you have"
  type        = string
  default     = "us-sanjose-1"
}

variable "user_ocid" {}
variable "privatekey_path" {}
variable "fingerprint" {}
variable "tenancy_ocid" {}
variable "compartment_ocid" {}
```

```
# main.tf

terraform {
  required_providers {
    oci = {
      source = "oracle/oci"
    }
  }
}

provider "oci" {
  # tenancy_ocid = "<tenancy OCID>"
  # user_ocid = " ocid1.compartment.oc1..aaaa"
  # private_key_path = "~/keys/priv.pem"
  # fingerprint = "AA:11:BB:22"
  region = "<region name>"
}

resource "oci_core_virtual_network" "vcn01" {
  compartment_id = "ocid1.compartment.oc1..aaaa"
  cidr_block = "10.0.0.0/16"
  dns_label = "vcn01"
  display_name = "vcn01"
}
```

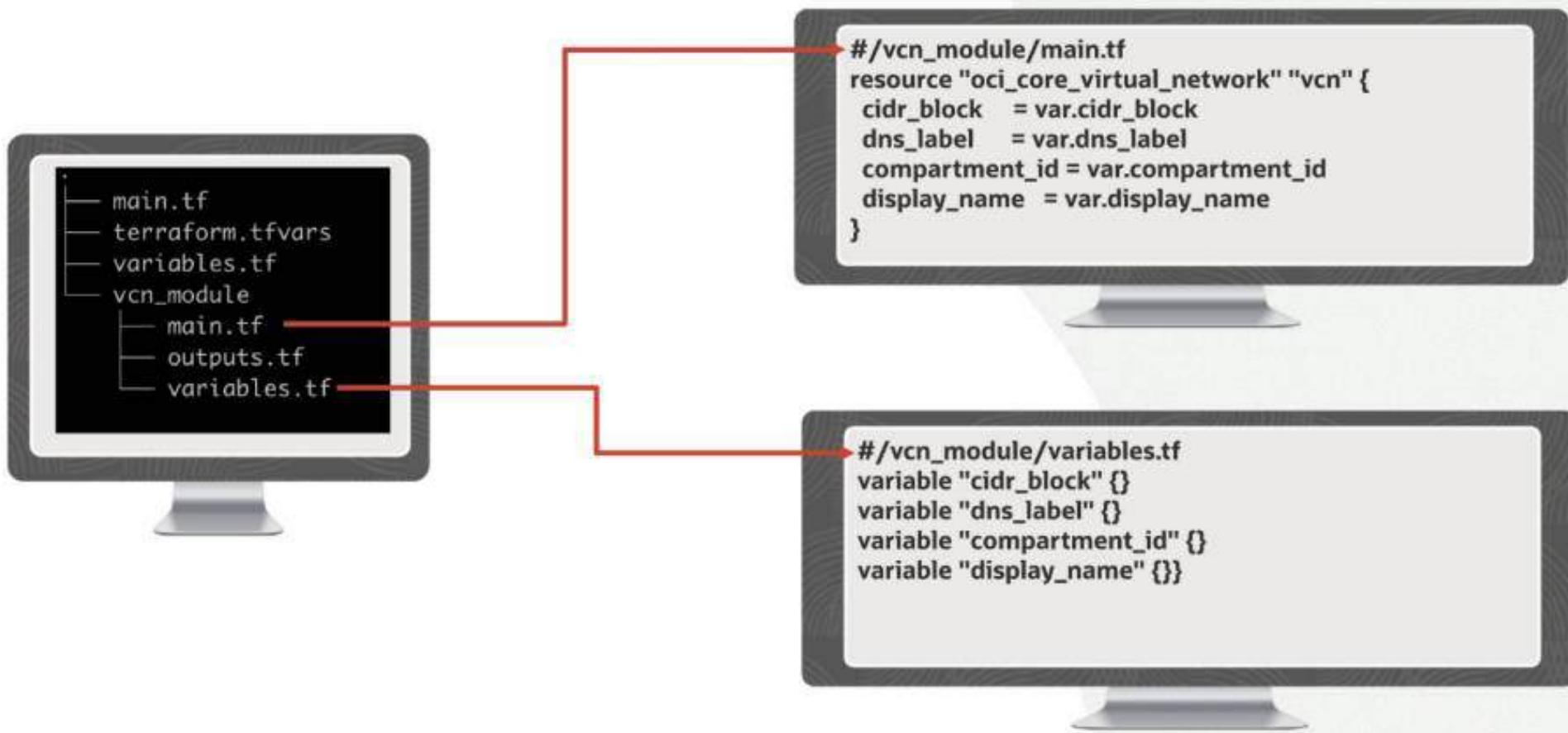
```
variables {
  tenancy_ocid = var.tenancy_ocid
  user_ocid = var.user_ocid
  privatekey_path = var.privatekey_path
  fingerprint = var.fingerprint
  region = var.region
```



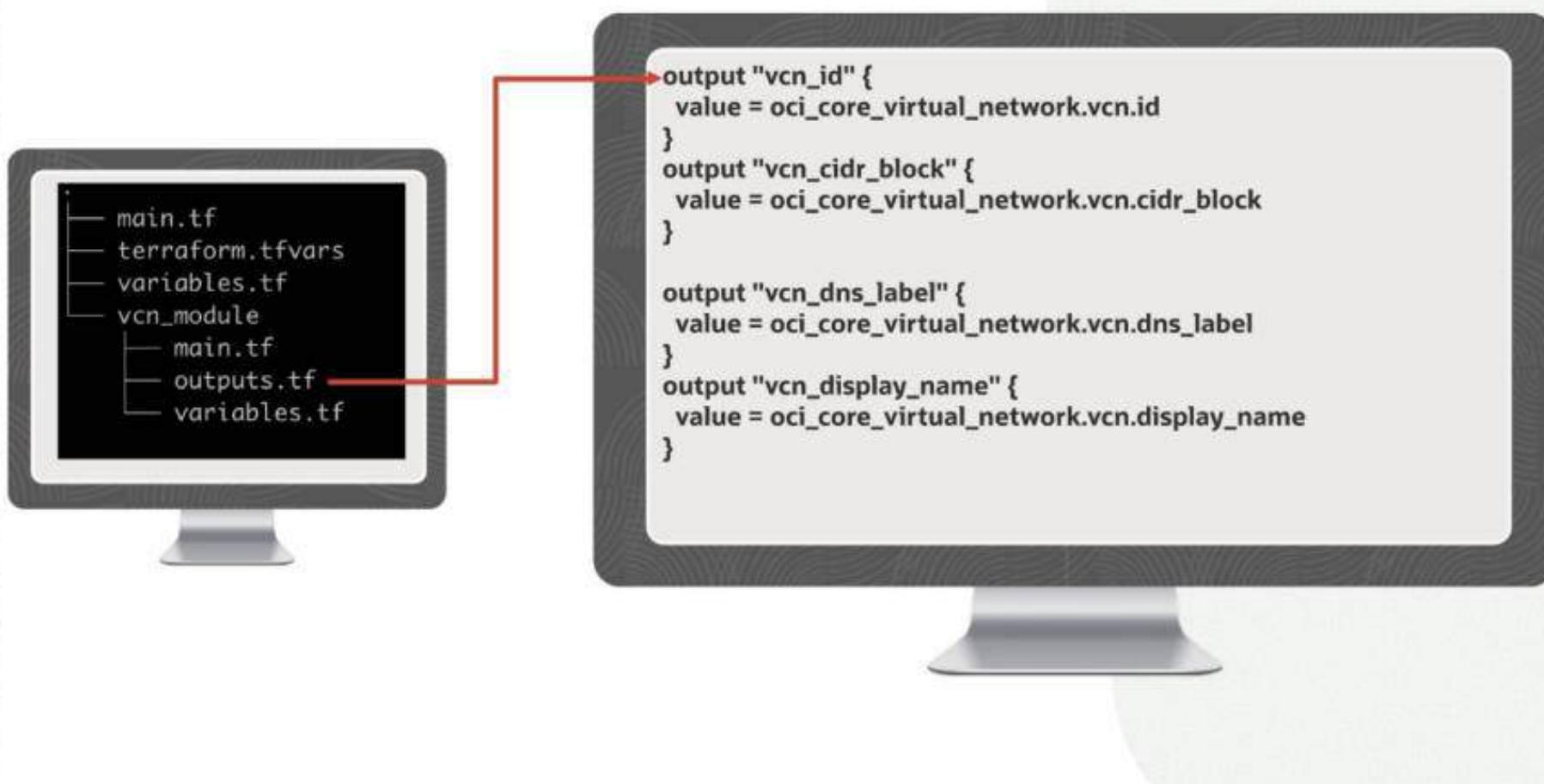
# Incorporating Modules

---

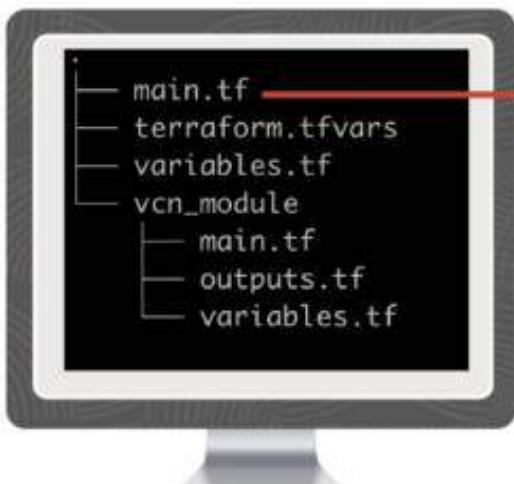
# Creating the Module



# Creating the Module - Continued



# Incorporating Modules



```
#main.tf
module "vcn01" {
  source          = "./vcn_module"
  vcn_name        = var.vcn01_display_name
  vcn_cidr_block  = var.vcn01_cidr_block
  compartment_id  = var.compartment_ocid
}

module "vcn02" {
  source          = "./vcn_module"
  vcn_name        = "vcn02"
  vcn_cidr_block  = "172.0.0.0/16"
  compartment_id  = var.compartment_ocid
}
```

# OCI Resource Manager

# Oracle Cloud Infrastructure OCI Resource Manager

---

## Introduction and Concepts

# Resource Manager Concepts

Configuration Source Provider

Configuration

Stacks

Actions

Jobs

Templates





# Configuration Source Providers

---

## Configuration Source Providers

Streamlined access to external configuration data, facilitating efficient management and integration of settings across your cloud infrastructure.



Bitbucket  
Server



Bitbucket  
Cloud



GitHub



GitLab



# Configuration

---

# Configuration

```
# variables.tf

variable "region" {
    description = "region where you have OCI tenancy"
    type        = string
    default     = "us-sanjose-1"
}
variable "user_ocid" {}
variable "privatekey_path" {}
variable "fingerprint" {}
variable "tenancy_ocid" {}
variable "compartment_ocid" {}
```

```
# network.tf

resource "oci_core_vcn" "internal" {
    dns_label  = "internal"
    cidr_block = "172.16.0.0/20"
    compartment_id = var.compartment_ocid
    display_name = "My internal VCN"
```

```
#provider.tf

terraform {
    required_providers {
        oci = {
            source = "oracle/oci"
        }
    }
}

provider "oci" {
    tenancy_ocid = var.tenancy_ocid
    user_ocid = var.user_ocid
    private_key_path = var.privatekey_path
    fingerprint = var.fingerprint
    region = var.region
}
```

```
# network.tf

resource "oci_core_vcn" "internal" {
    dns_label  = "internal"
    cidr_block = "172.16.0.0/20"
    compartment_id = var.compartment_ocid
    display_name = "My internal VCN"
```



# Stacks

---

# Stacks Configuration



```
# variables.tf
variable "region" {
    description = "region where you have OCI tenancy"
    type        = string
    default     = "us-sanjose-1"
}
variable "user_ocid" {}
variable "privatekey_path" {}
variable "fingerprint" {}
variable "tenancy_ocid" {}
variable "compartment_ocid" {}
```

**Your Stack**

```
# network.tf
resource "oci_core_vcn" "internal" {
    dns_label  = "internal"
    cidr_block = "172.16.0.0/20"
    compartment_id = var.compartment_ocid
    display_name = "My internal VCN"
```

```
# provider.tf
terraform {
    required_providers {
        oci = {
            source = "oracle/oci"
        }
    }
}
provider "oci" {
    tenancy_ocid = var.tenancy_ocid
    user_ocid = var.user_ocid
    private_key_path = var.privatekey_path
    fingerprint = var.fingerprint
    region = var.region
}
```

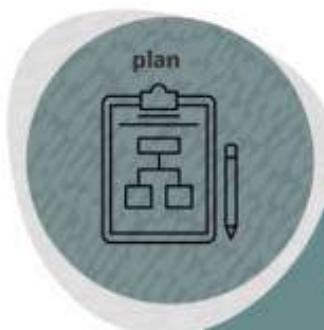
```
# network.tf
resource "oci_core_vcn" "internal" {
    dns_label  = "internal"
    cidr_block = "172.16.0.0/20"
    compartment_id = var.compartment_ocid
    display_name = "My internal VCN"
```



# Actions

---

# Actions



Show changes required by the current configuration



Create or update infrastructure



Destroy previously created infrastructure



Change the configuration



Detect differences between the configuration and the infrastructure



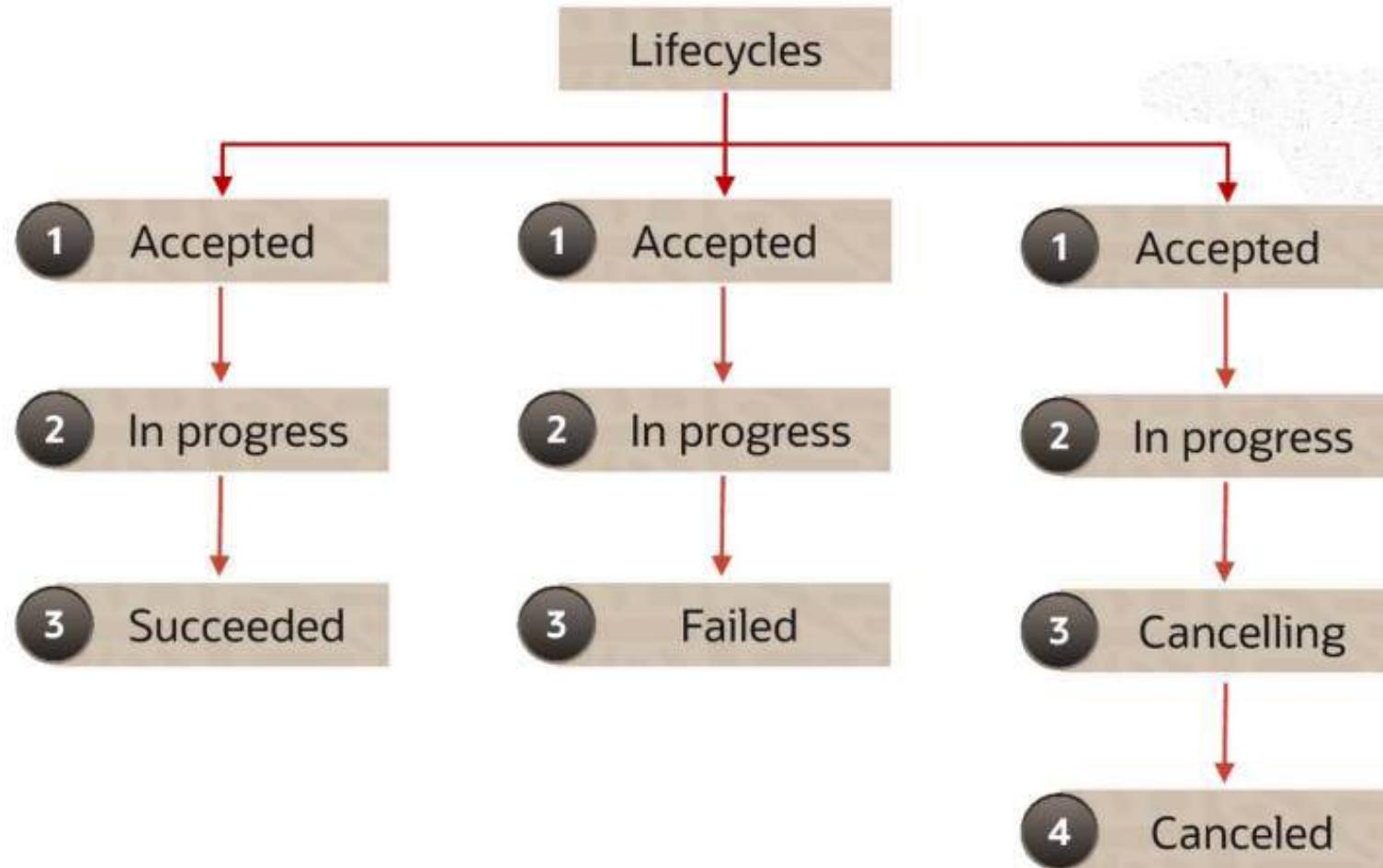
Import state to reflect existing infrastructure



# Jobs

---

## Jobs

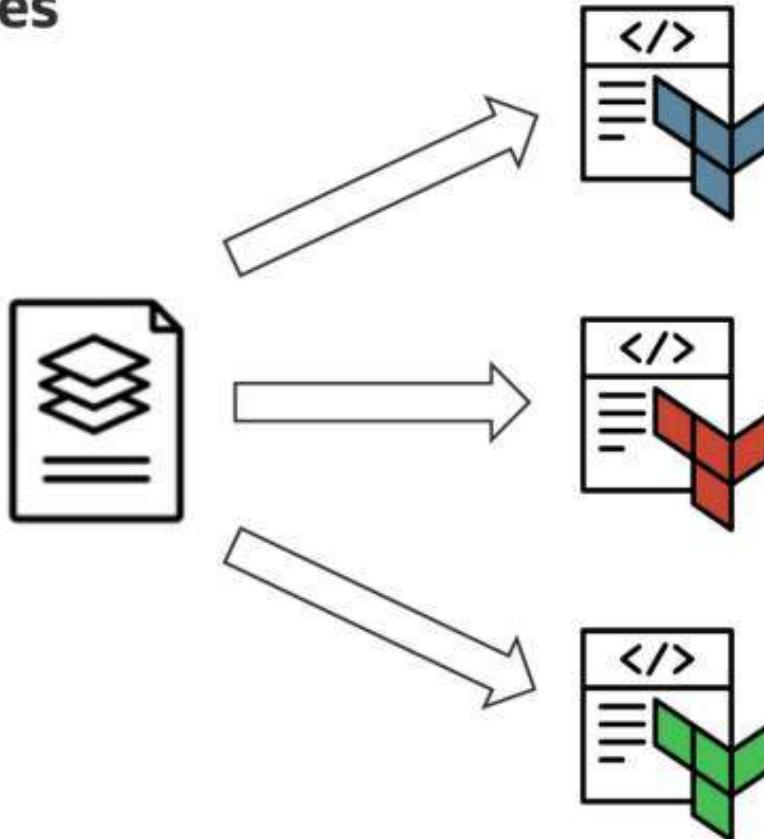




# Templates

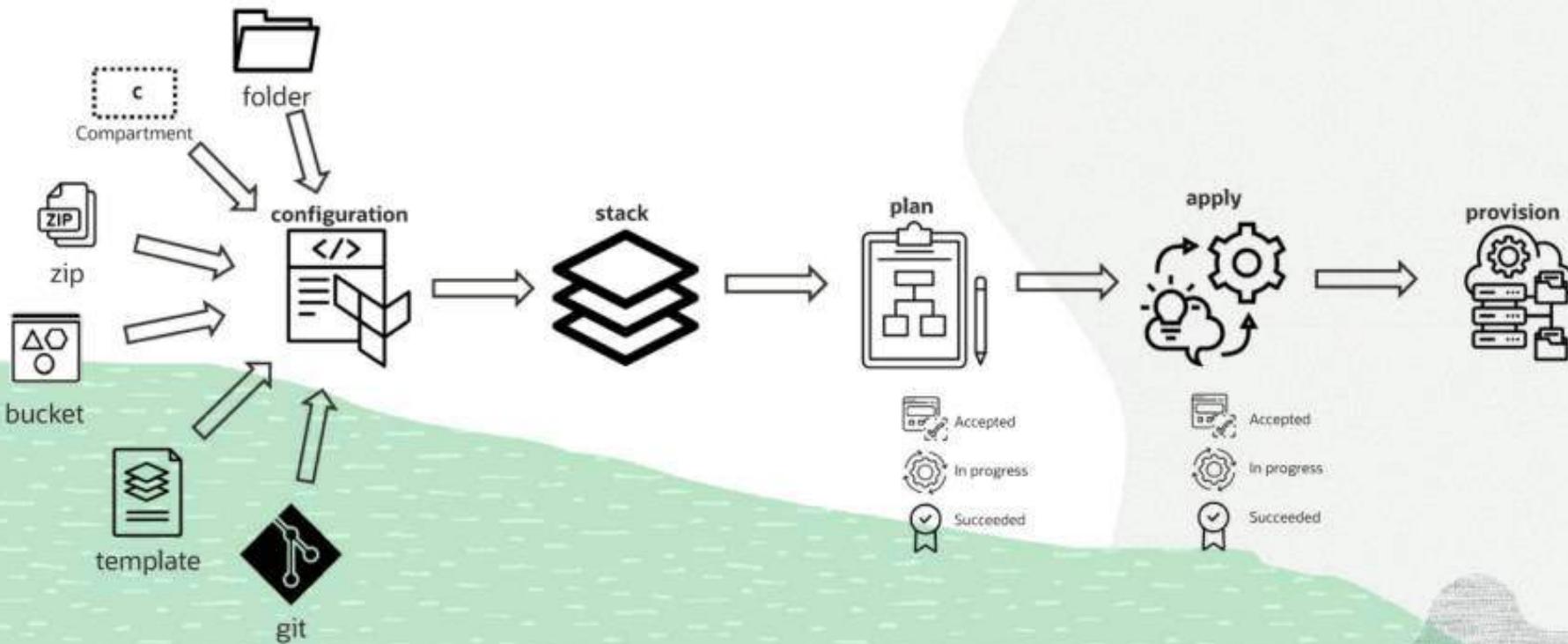
---

## Templates



Templates offer standardized configurations enabling automated deployment and management

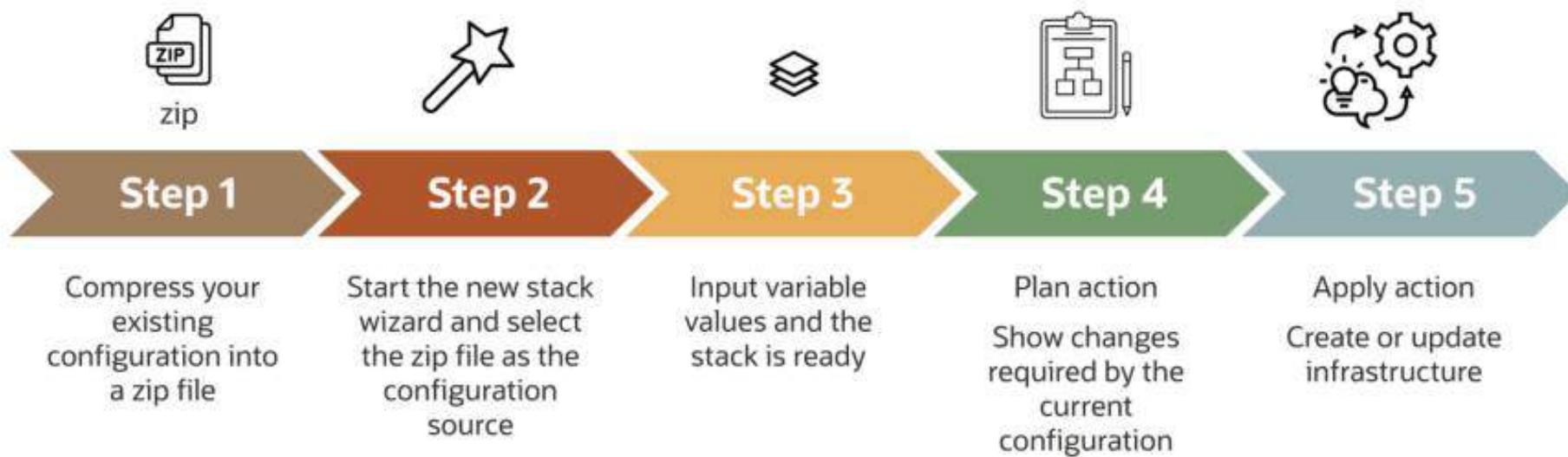
# All Together



# Oracle Cloud Infrastructure OCI Resource Manager

## Creating Your First Stack

# Creating Your First Stack



# Oracle Cloud Infrastructure OCI Resource Manager

## Using Source Providers

## Configuration Source Providers

Streamlined access to external configuration data, facilitating efficient management and integration of settings across your cloud infrastructure.



Bitbucket  
Server



Bitbucket  
Cloud



GitHub



GitLab

# Using a Configuration Source Provider



## Step 1

Create a repository that contains your configuration

## Step 2

Create the source provider

## Step 3

Start the new stack wizard and select the source provider as the configuration source

## Step 4

Input variable values if present and the stack is ready

## Step 5

Plan action  
Show changes required by the current configuration

## Step 6

Apply action  
Create or update infrastructure

# Oracle Cloud Infrastructure Infrastructure as Code

## Importing Existing Infrastructure



# Importing Existing Infrastructure

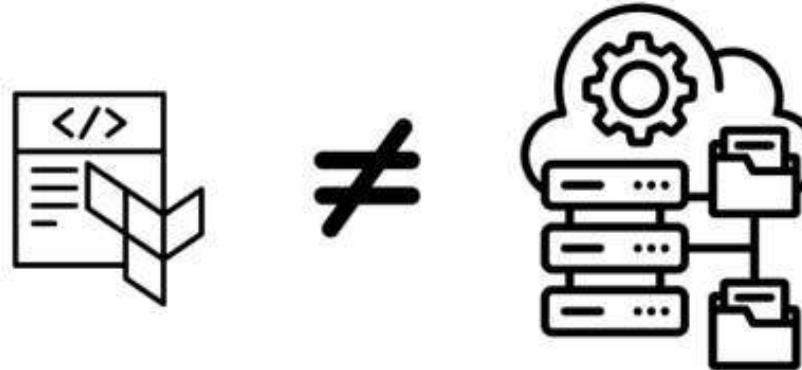


# Oracle Cloud Infrastructure Infrastructure as Code

## Drift Detection

## Drift

---



Drift is the variance between the current state of deployed resources and the desired state defined in the Terraform configuration.



## Using Drift Detection



### Step 1

Run the drift detection action. A report will be created if drift is detected

### Step 2

Inspect the report and find the differences

### Step 3

Edit your configuration, incorporating the changes

### Step 4

Run the plan action

### Step 5

Run the apply actions

### Step 6

Run the drift detection action to ensure there is no drift

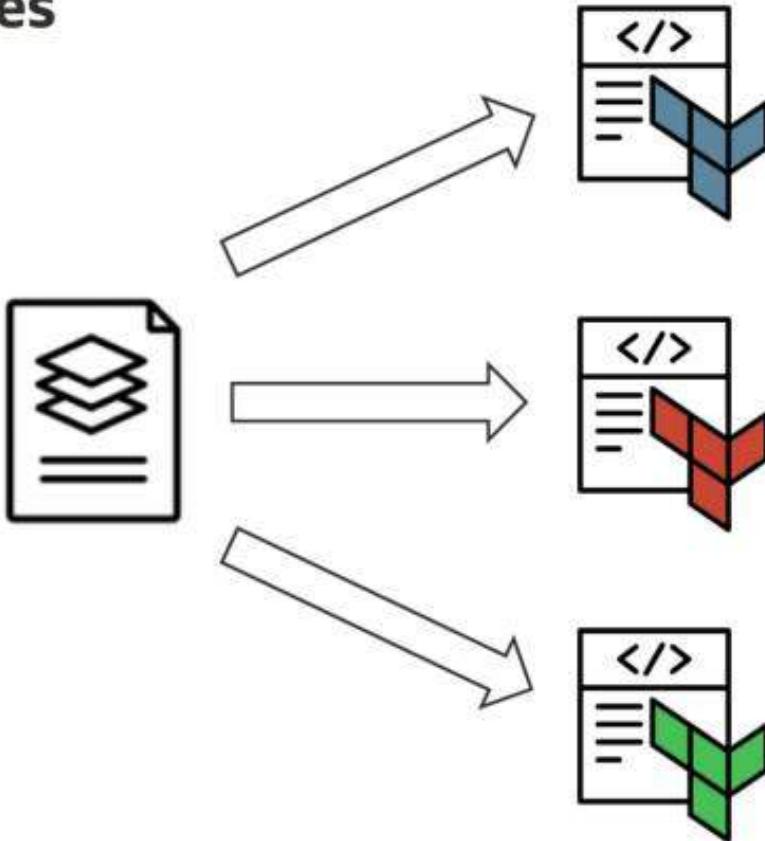
# Oracle Cloud Infrastructure

## Infrastructure as Code

---

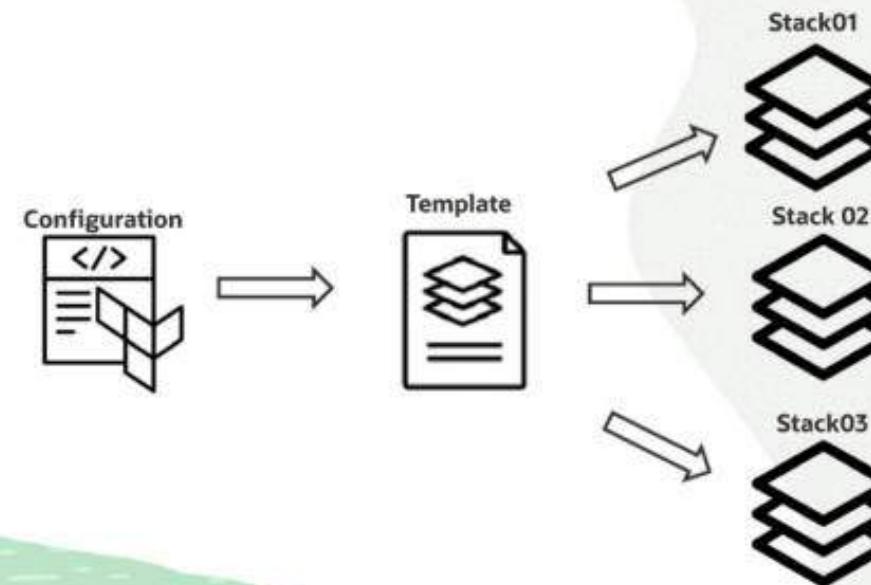
### Templates

# Templates



Templates offer standardized configurations enabling automated deployment and management

# Creating Private Templates



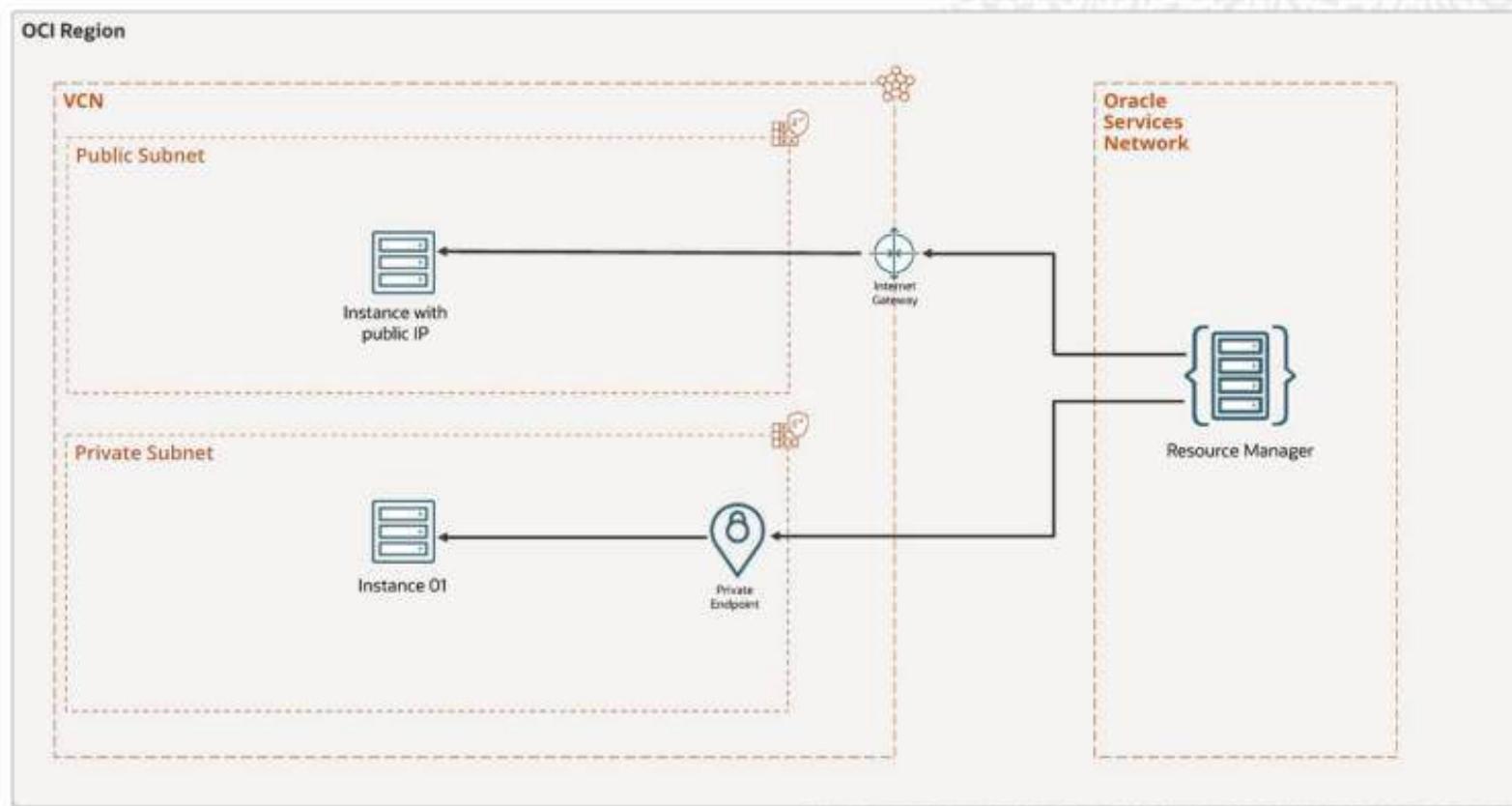
# Oracle Cloud Infrastructure

## Infrastructure as Code

---

### Remote Exec and Endpoints

# Resource Manager Endpoints



# Creating the Endpoint with Terraform

```
// The RMS private endpoint resource. Requires a VCN with a private subnet  
  
resource "oci_resourcemanager_private_endpoint" "rms_private_endpoint" {  
    compartment_id = var.compartment_ocid  
    display_name   = "rms_private_endpoint"  
    description   = "rms_private_endpoint_description"  
    vcn_id        = module.my_vcn01.vcn_id  
    subnet_id     = module.my_vcn01.subnet_ids["Subnet-1"]  
}
```

```
// Resolves the private IP of the customer's private endpoint to a NAT IP. Used as the host  
// address in the "remote-exec" resource  
  
data "oci_resourcemanager_private_endpoint_reachable_ip" "test_private_endpoint_reachable_ip" {  
    private_endpoint_id = oci_resourcemanager_private_endpoint.rms_private_endpoint.id  
    private_ip         = oci_core_instance.instance01.private_ip  
}
```

```
resource "null_resource" "remote-exec" {  
    depends_on = [oci_core_instance.instance01]  
  
    provisioner "remote-exec" {  
        connection {  
            agent = false  
            timeout = "3m"  
            user = "opc"  
            host = data.oci_resourcemanager_private_endpoint_reachable_ip.test_private_endpoint_reachable_ip.ip_address  
            private_key = tls_private_key.public_private_key_pair.private_key_pem  
        }  
        // write to a file on the compute instance via the private access SSH connection  
        inline = [  
            "echo 'remote exec showcase' > ~/remoteExecTest.txt"  
        ]  
    }  
}
```



# Deploy a Monolithic Architecture



# Case Study Architecture

---

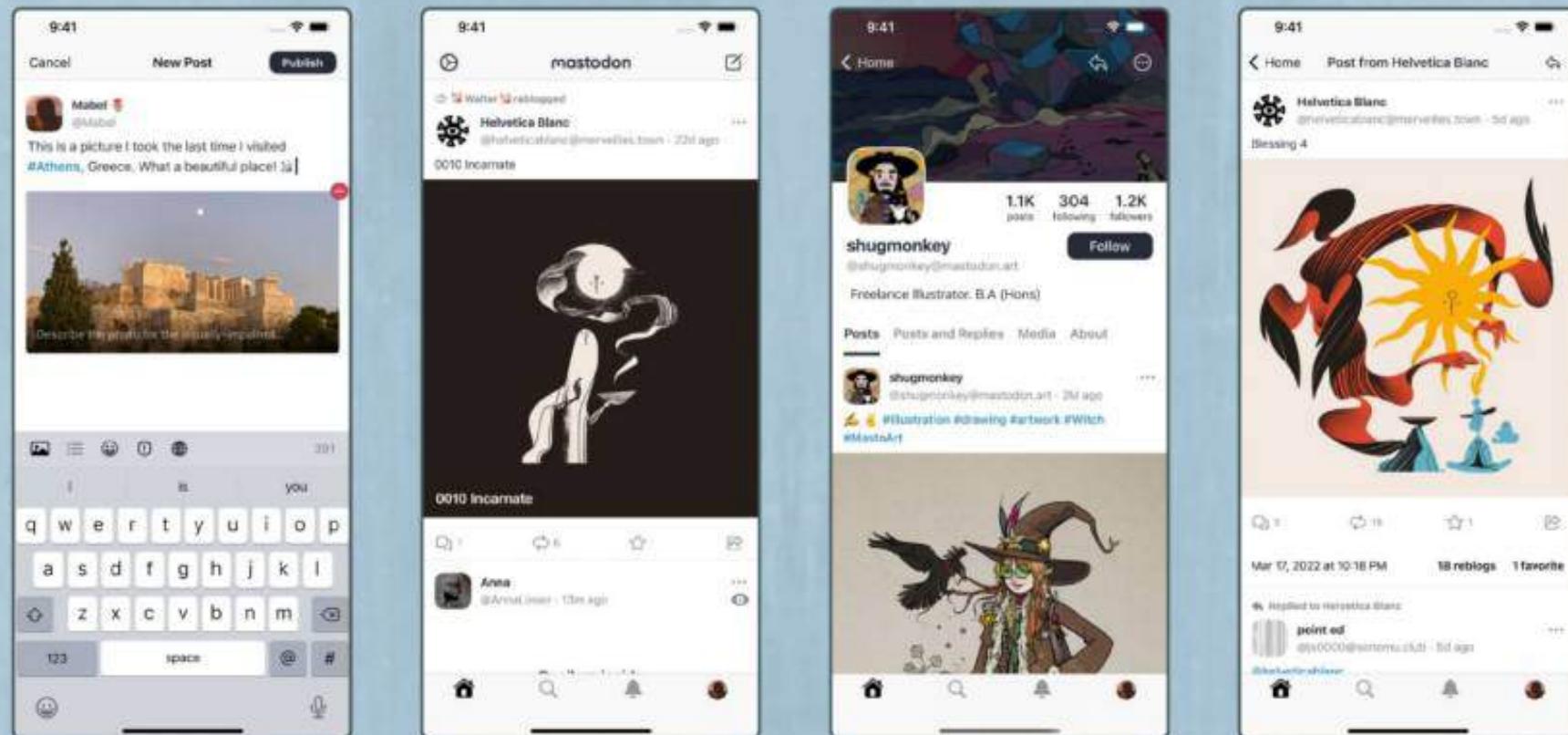
## OCI Cloud Operations



## Case Study: Mastodon



# Case Study: Mastodon



## Case Study: Mastodon

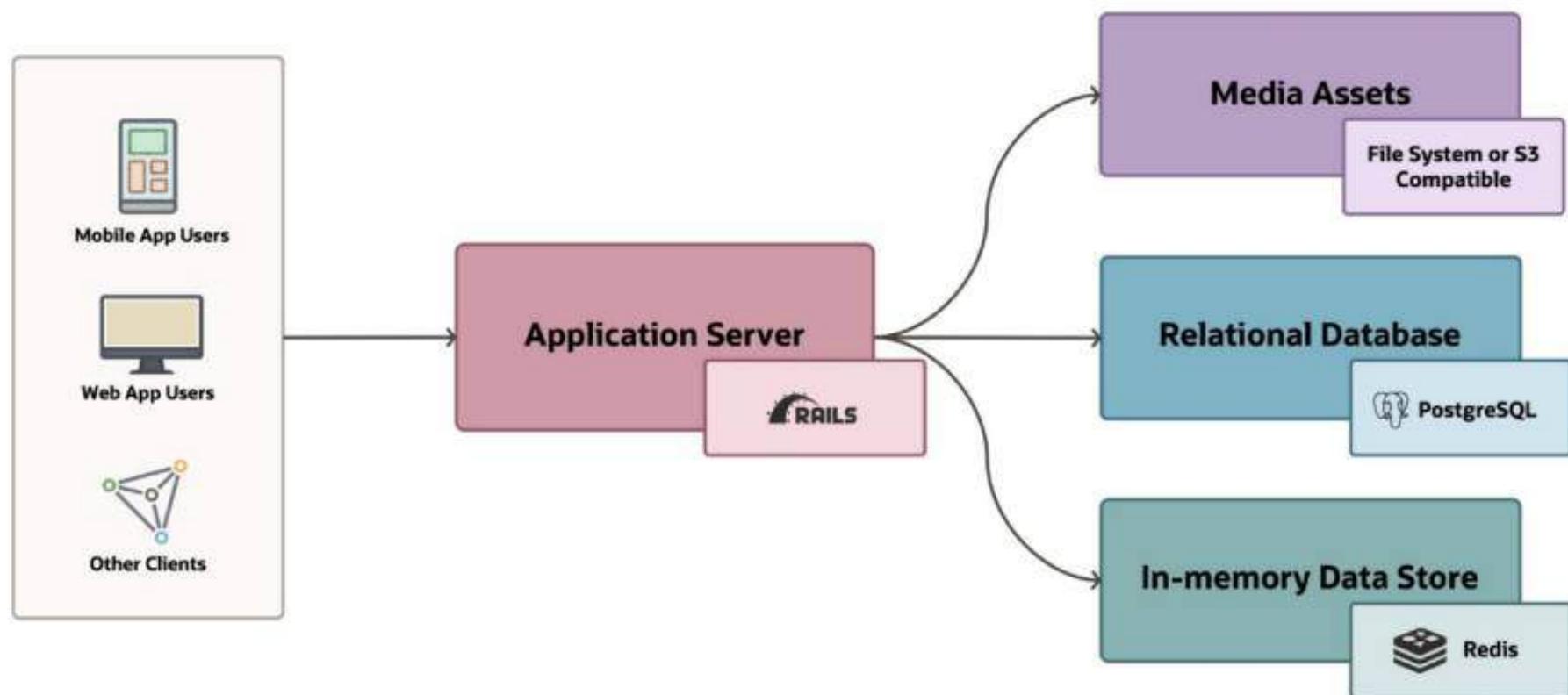
---



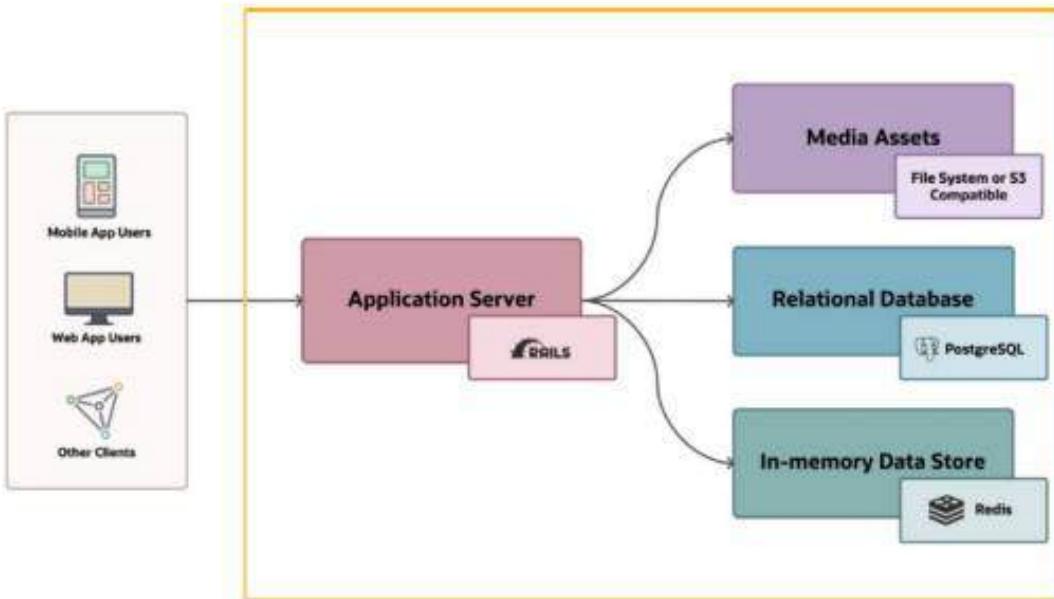
## Case Study: Mastodon



## Instance Architecture



## Instance Architecture

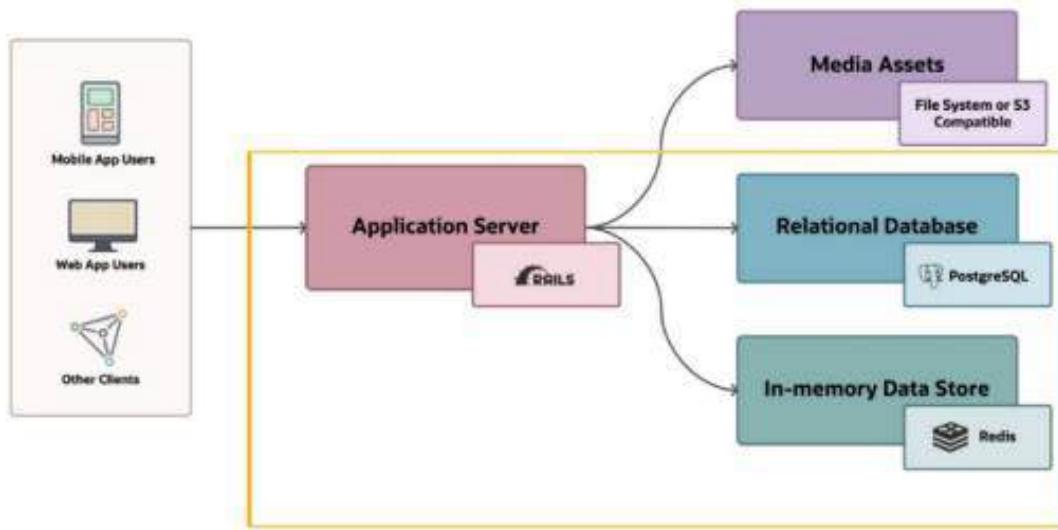


OCI Services



Virtual Networking

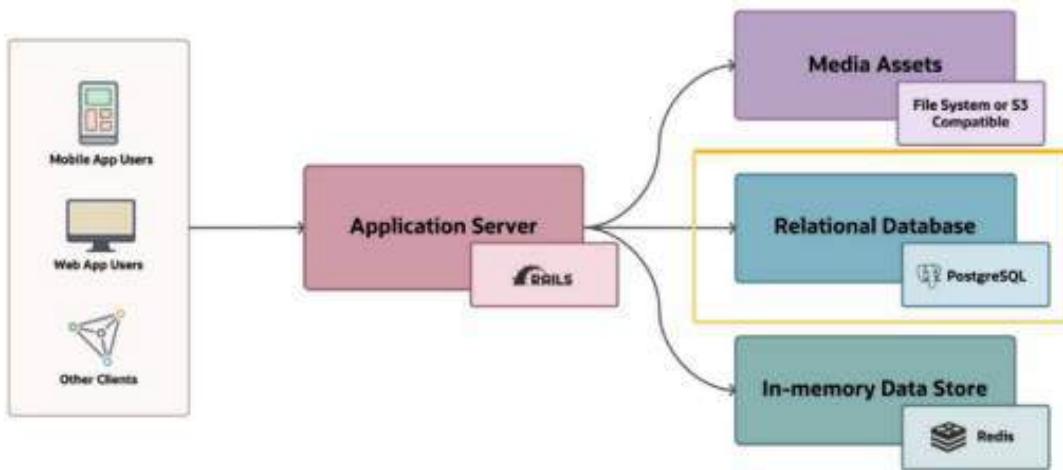
## Instance Architecture



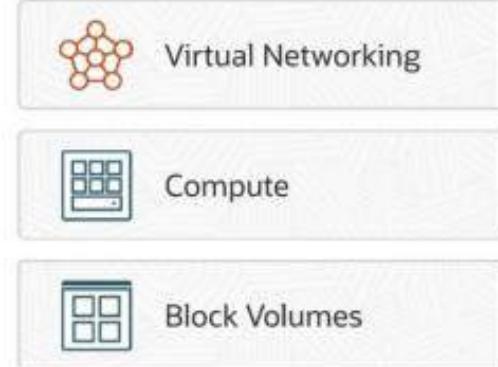
### OCI Services



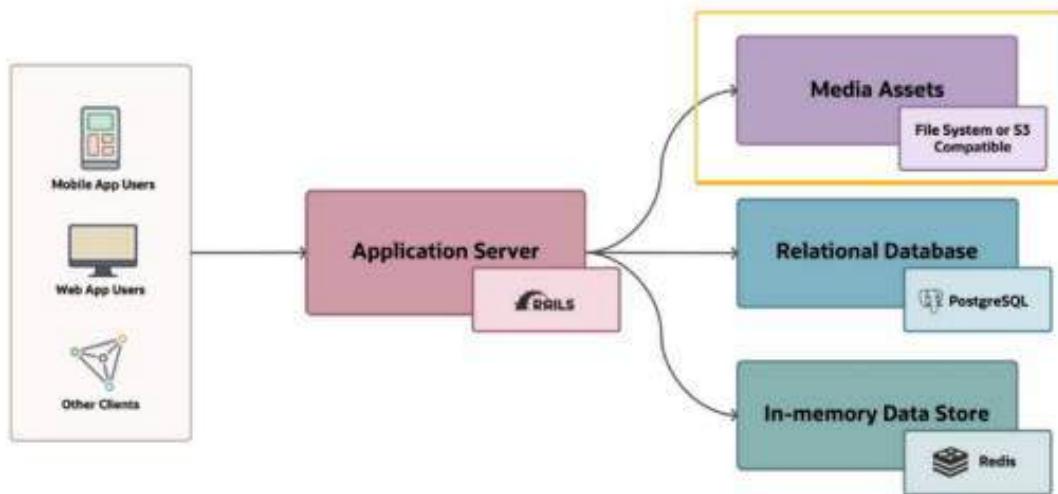
## Instance Architecture



### OCI Services



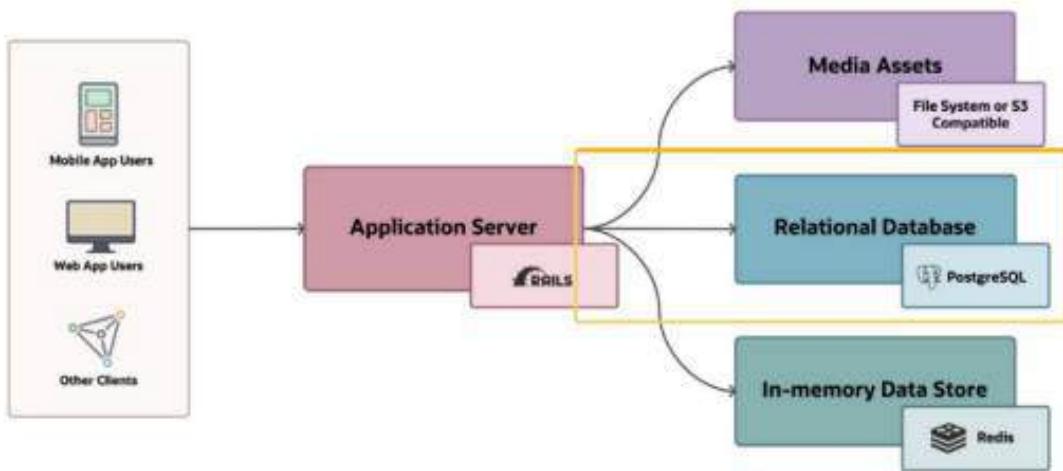
## Instance Architecture



### OCI Services



## Instance Architecture

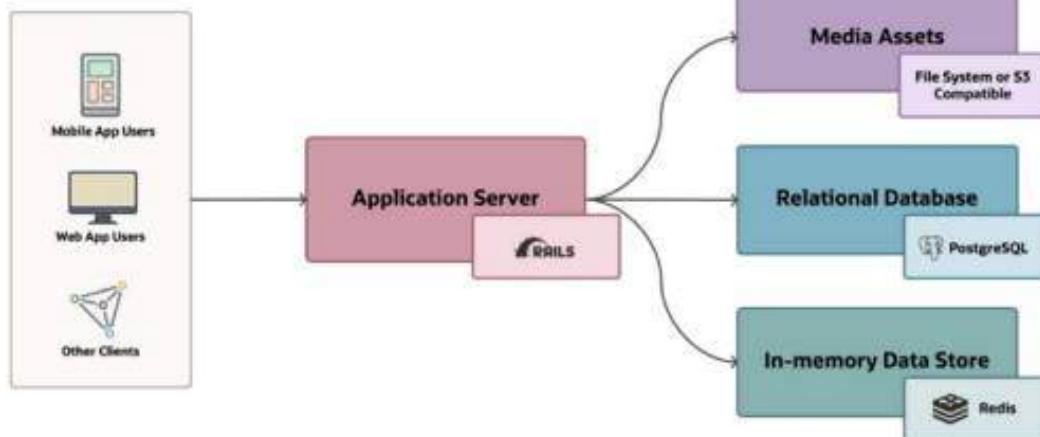
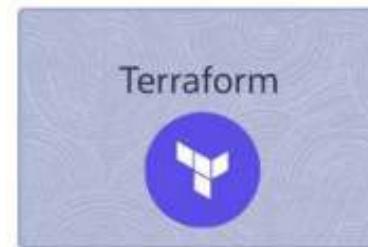
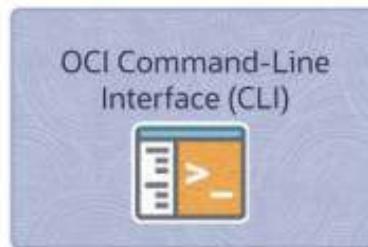
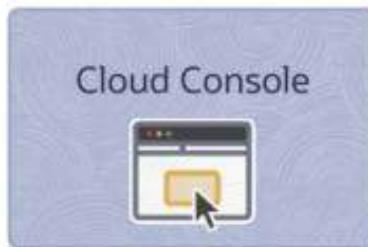


### OCI Services



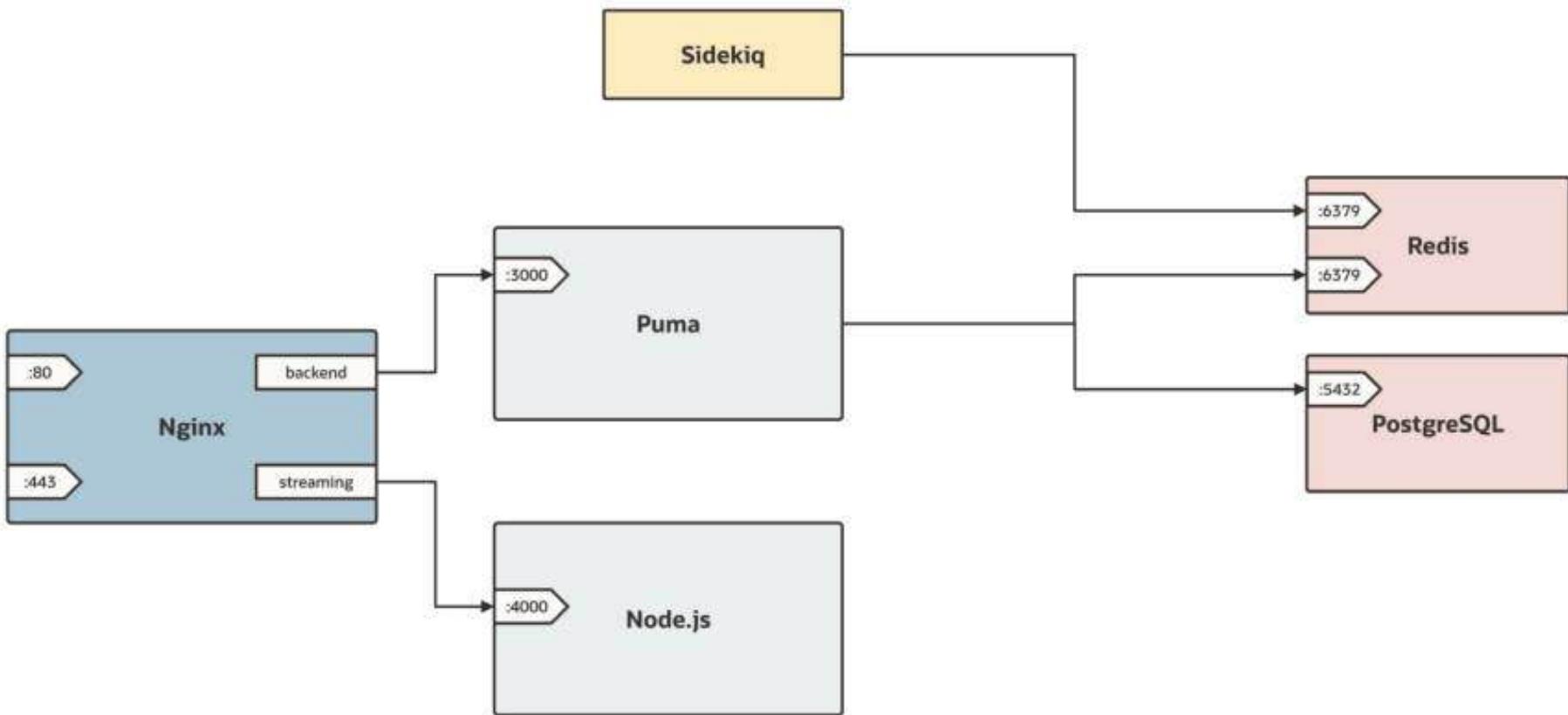
## Instance Architecture

### Methods

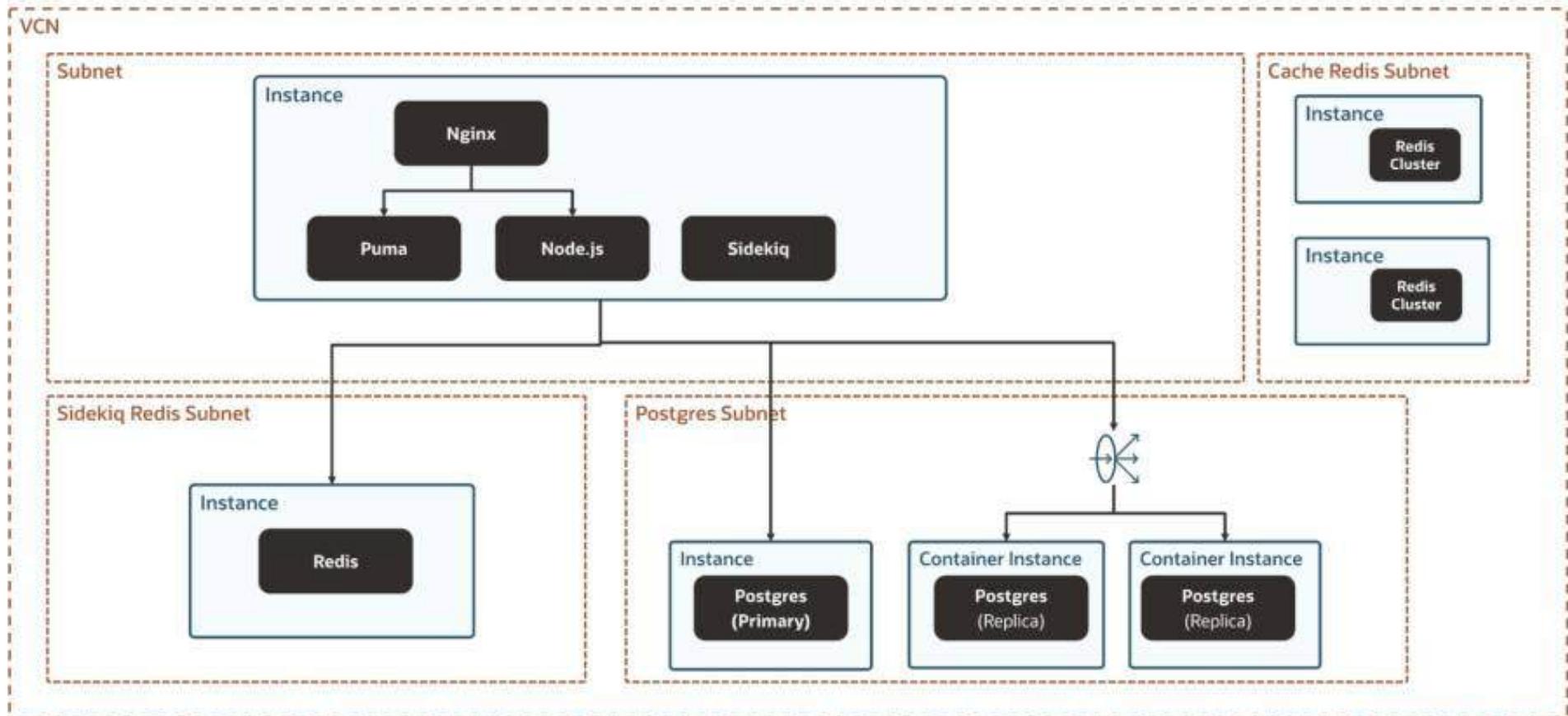


### OCI Services





# Mastodon Architecture





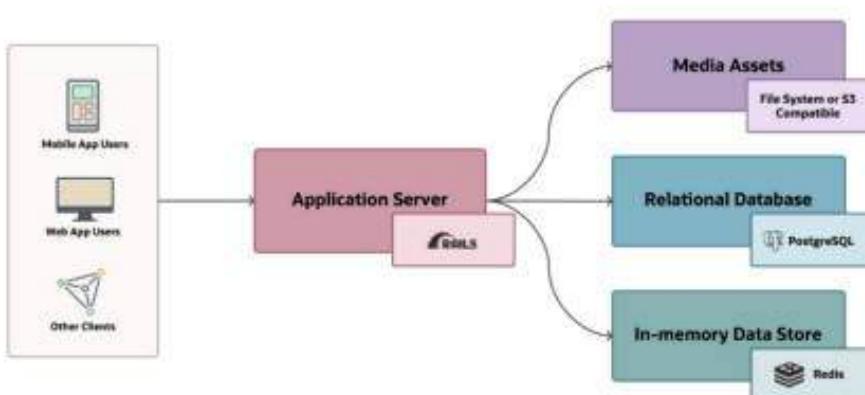
# VCN Deep Dive: Gateways and Routing

---

**OCI Cloud Operations**

## Virtual Networking

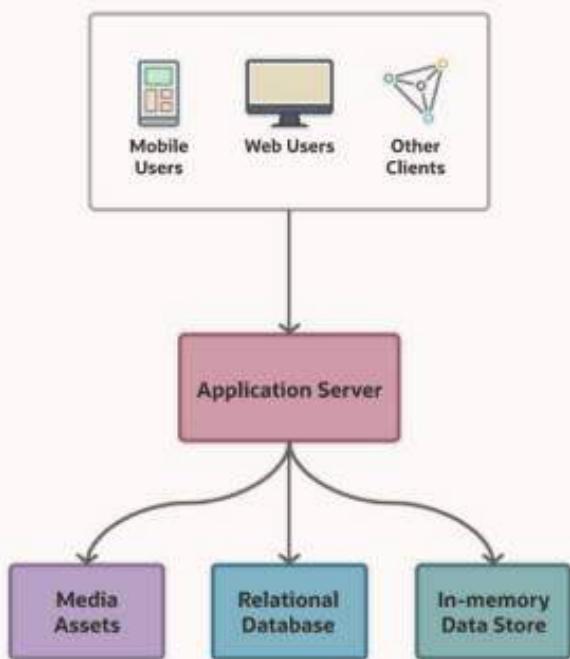
### Conceptual Design



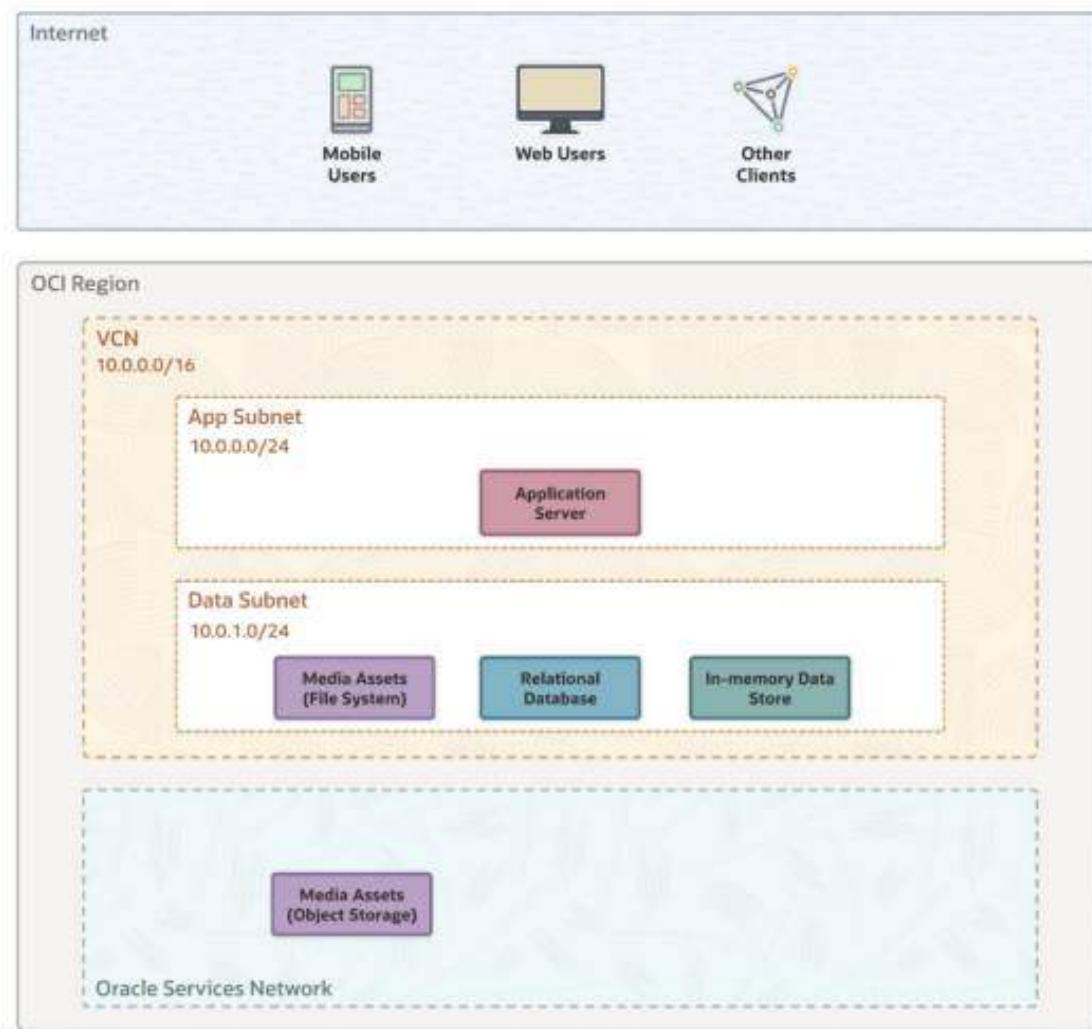
### Our approach to networking

- 1 Layout
- 2 Gateways and Routing
- 3 Access and Security

## Conceptual Design

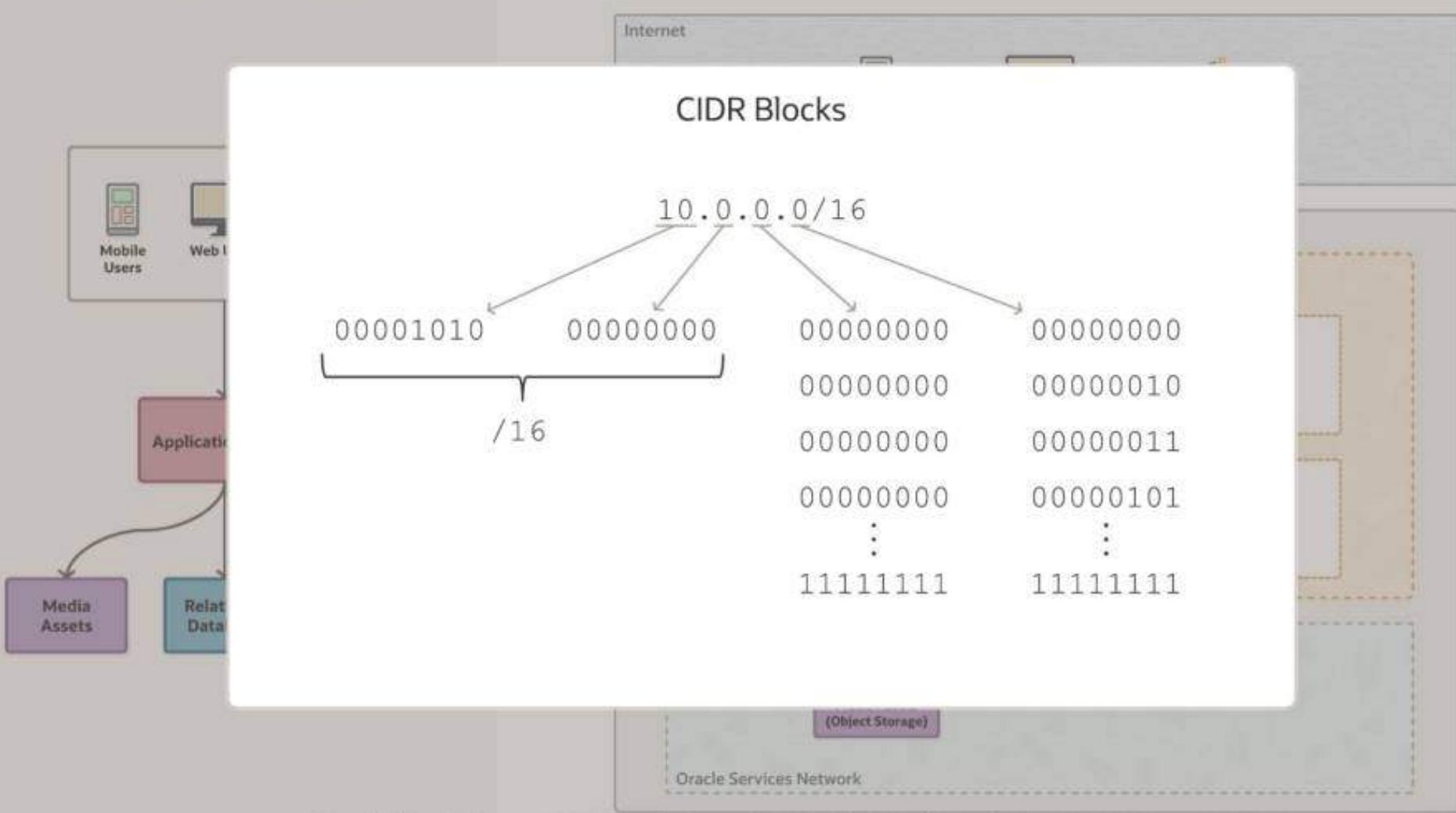


## VCN Layout



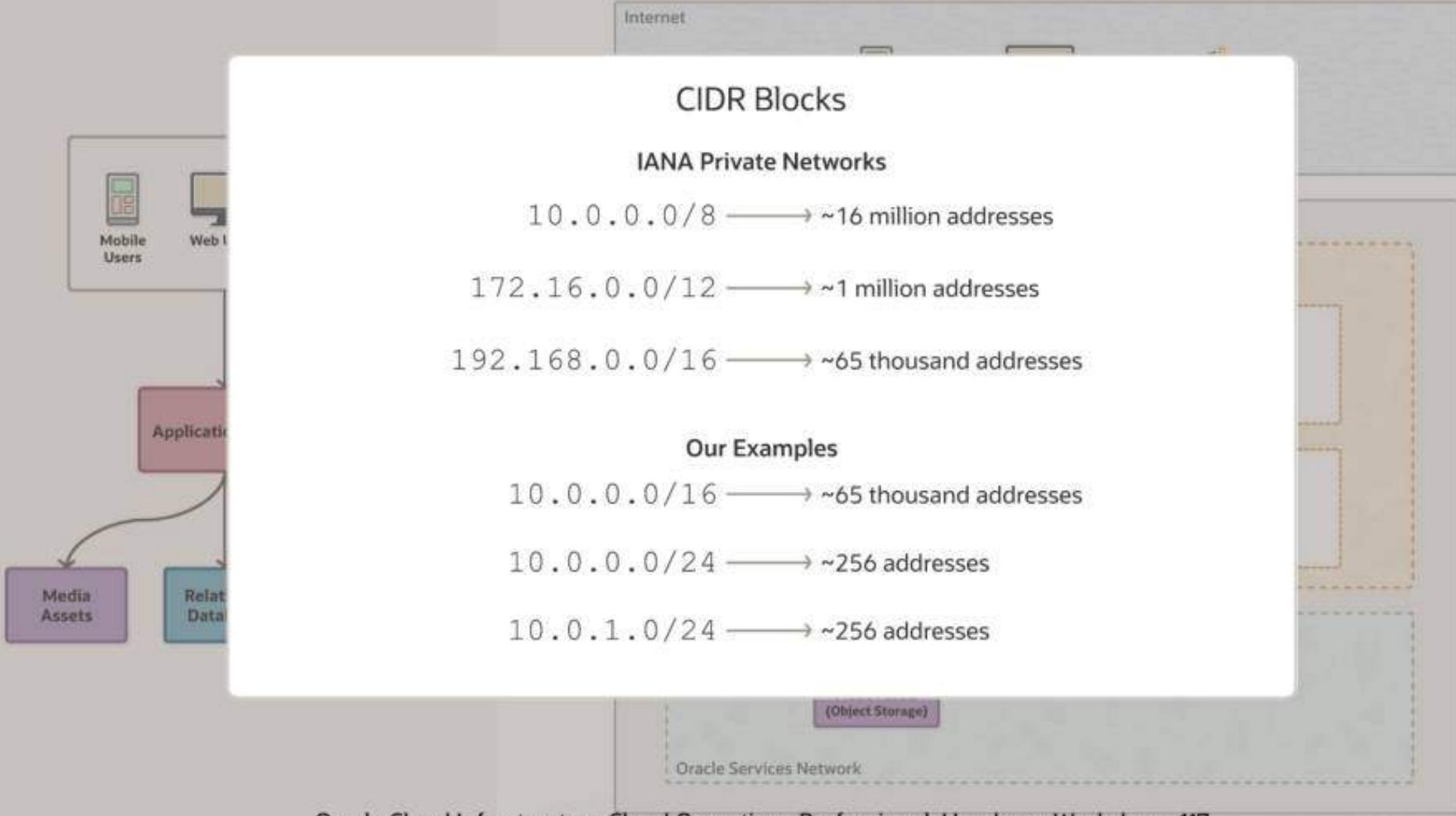
## Conceptual Design

## VCN Layout

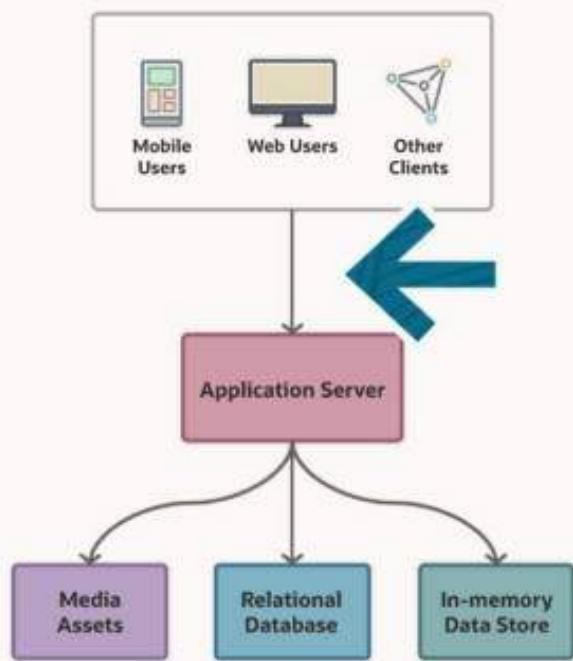


## Conceptual Design

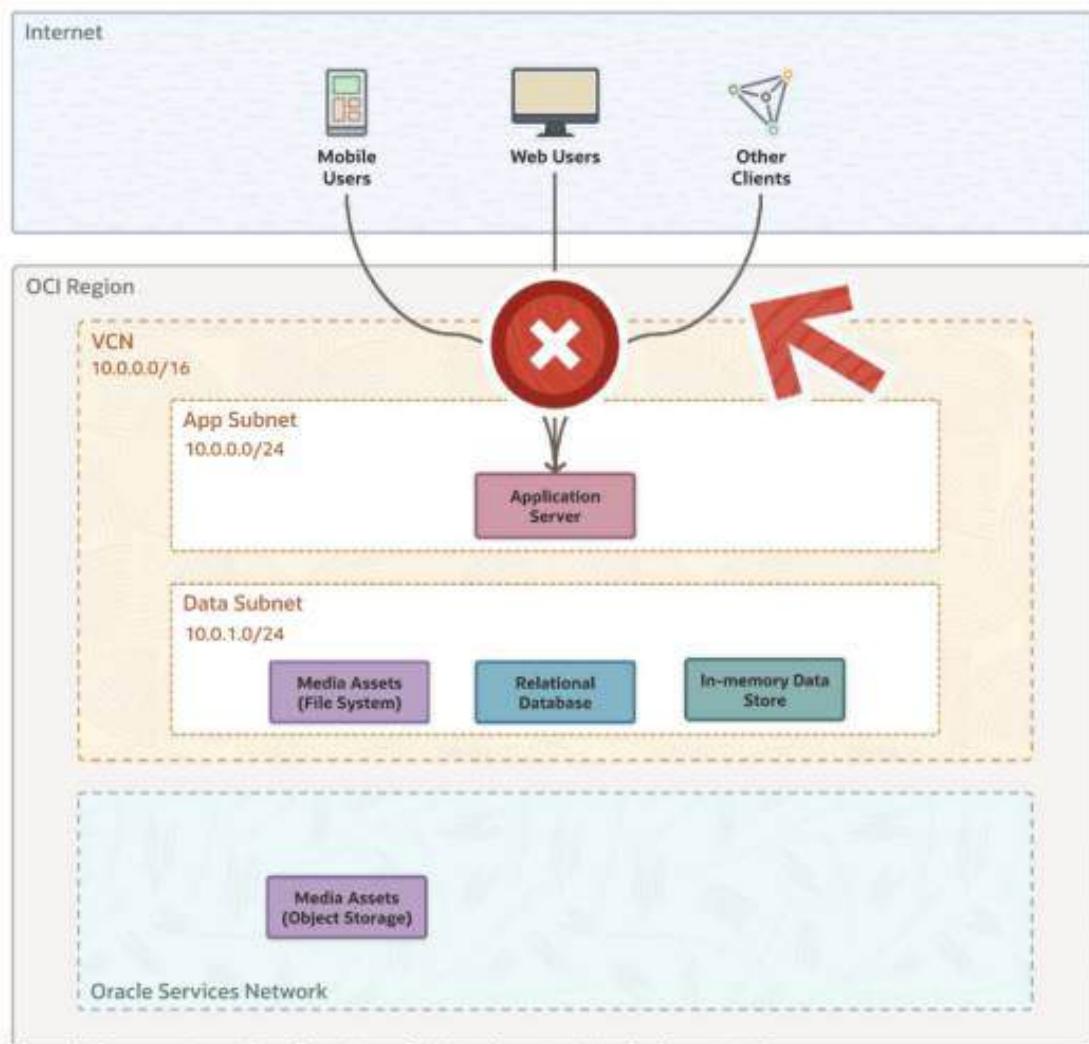
## VCN Layout



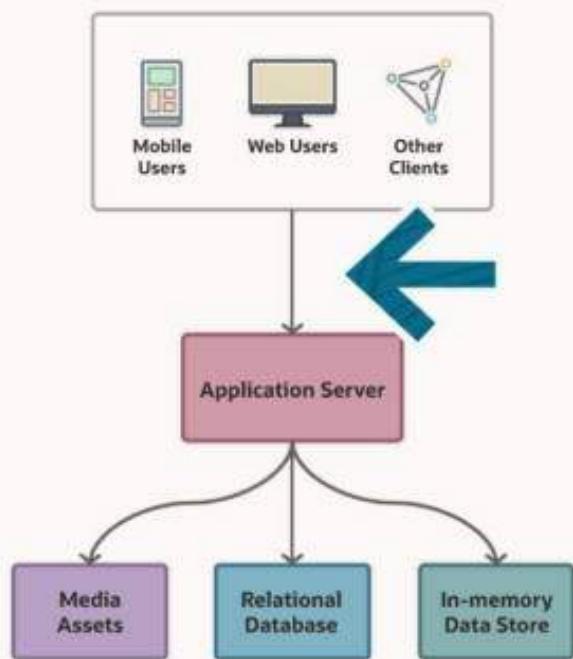
## Conceptual Design



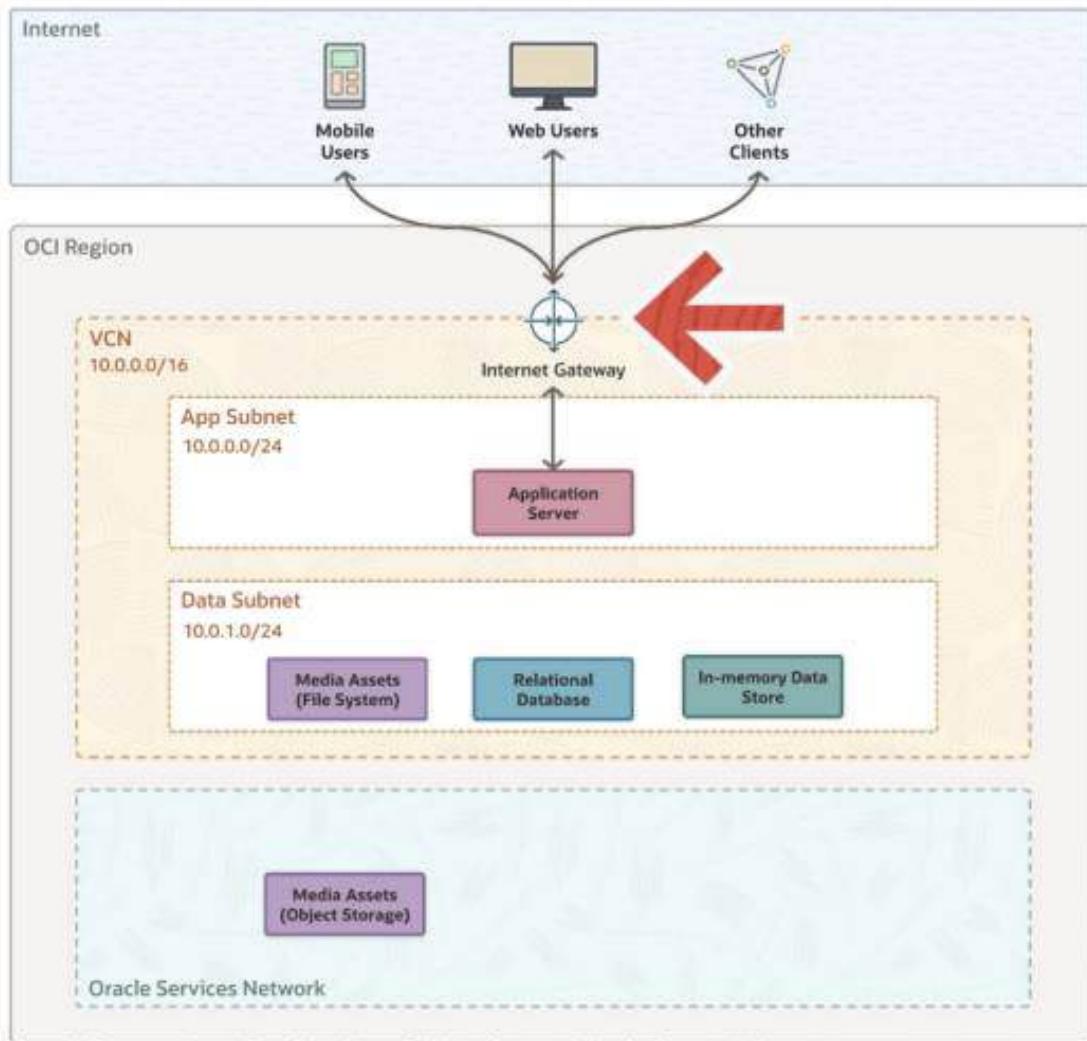
## VCN Gateways



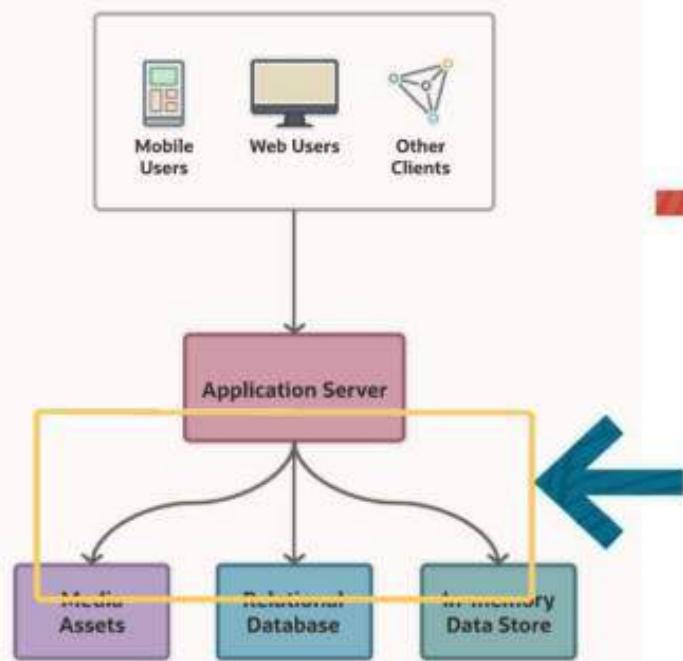
## Conceptual Design



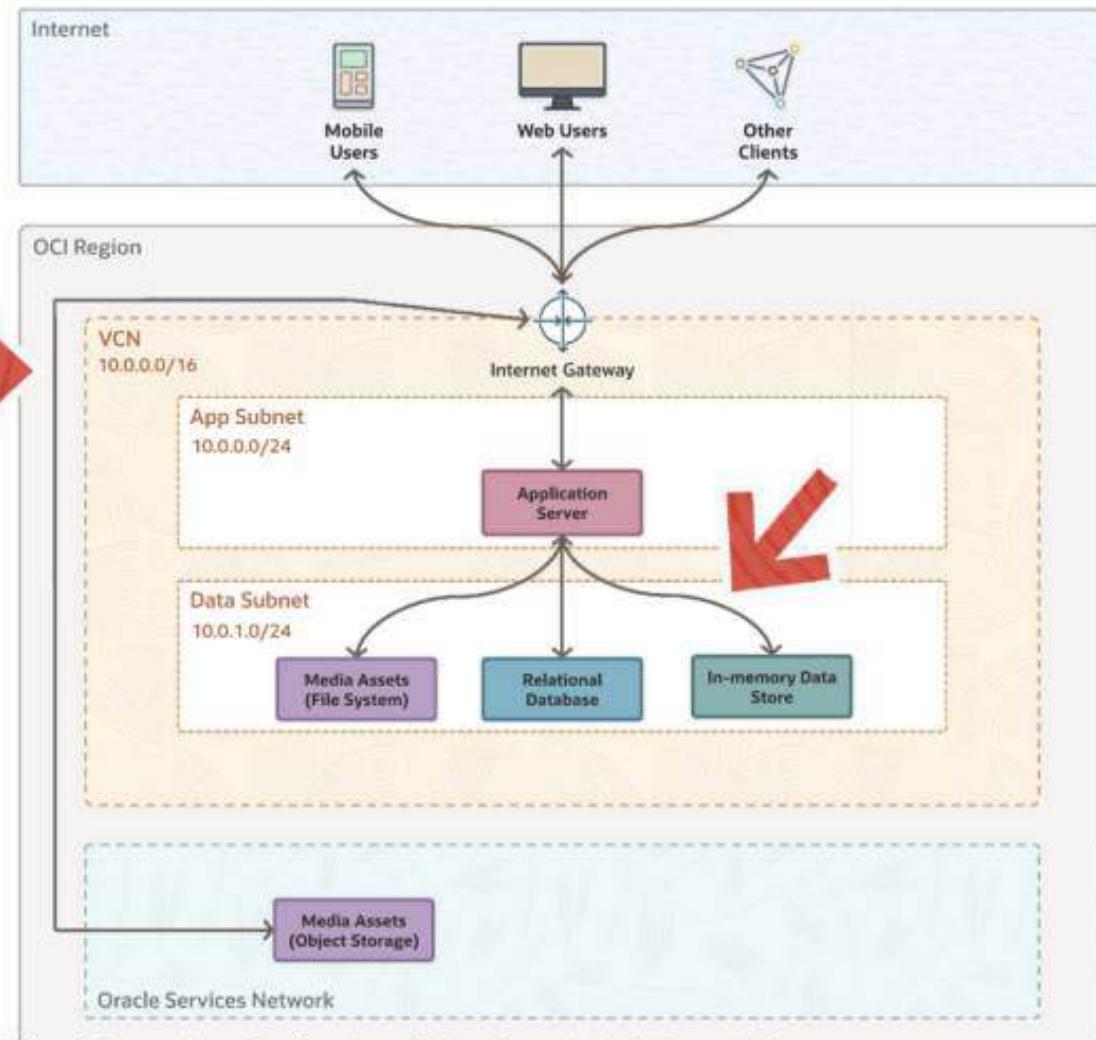
## VCN Gateways



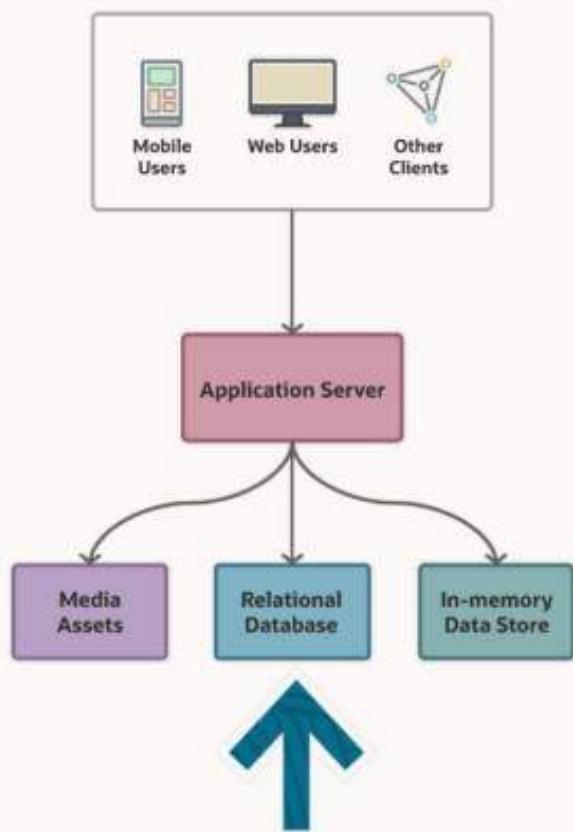
## Conceptual Design



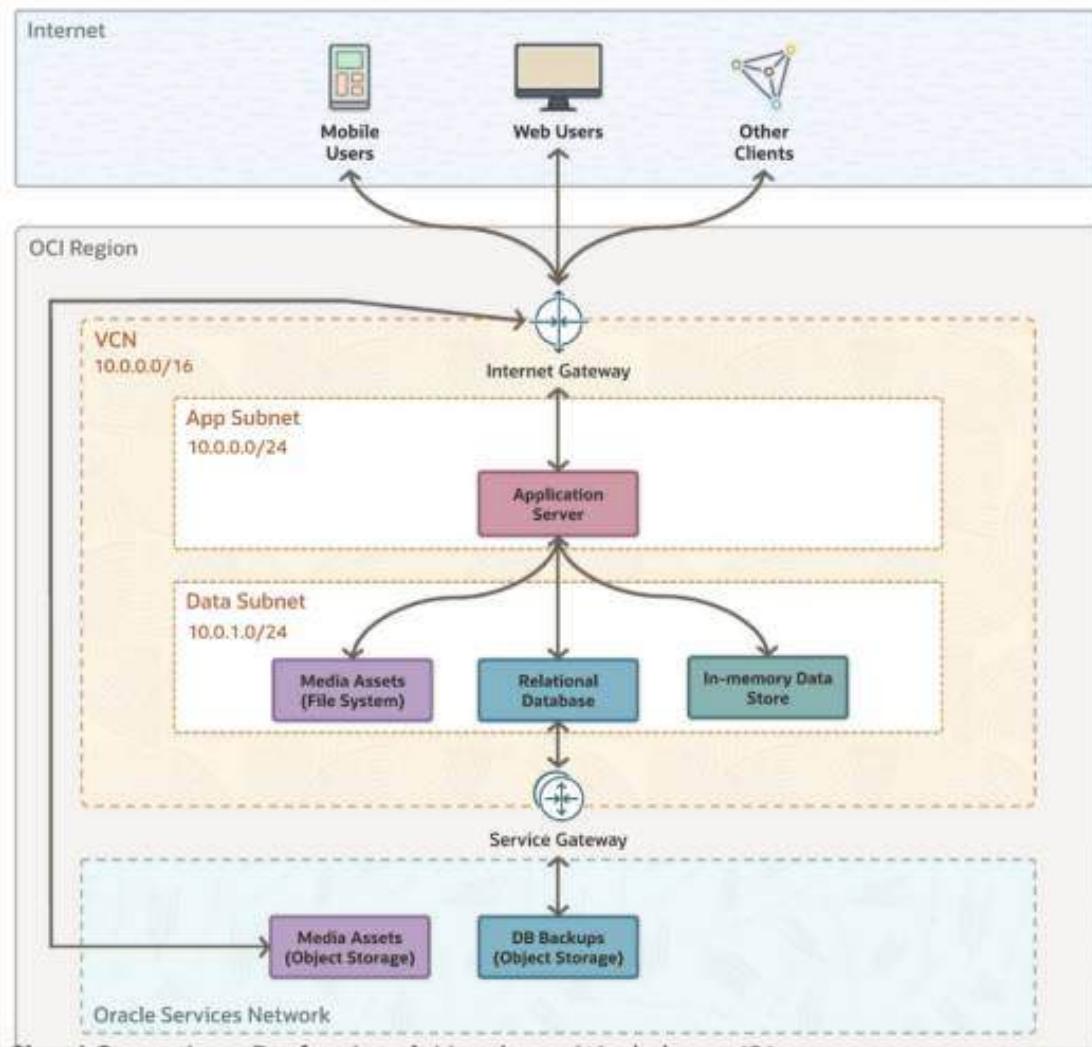
## VCN Gateways



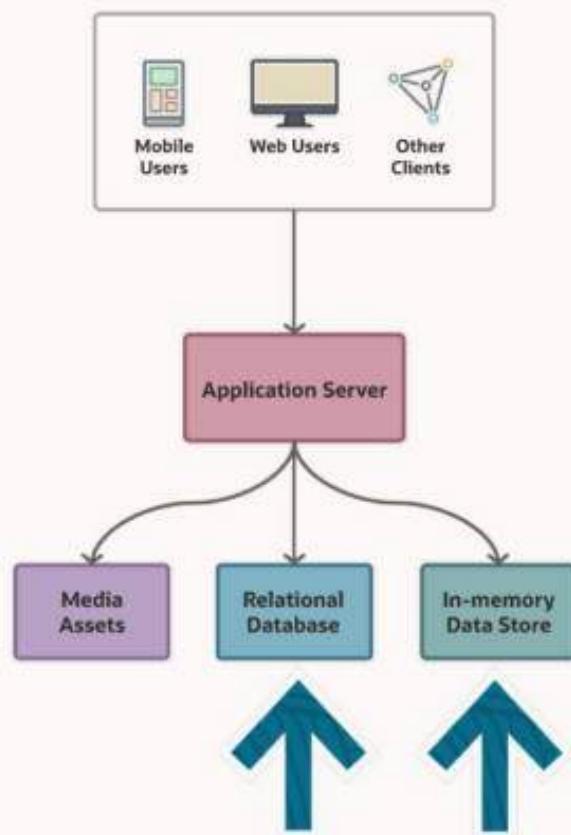
## Conceptual Design



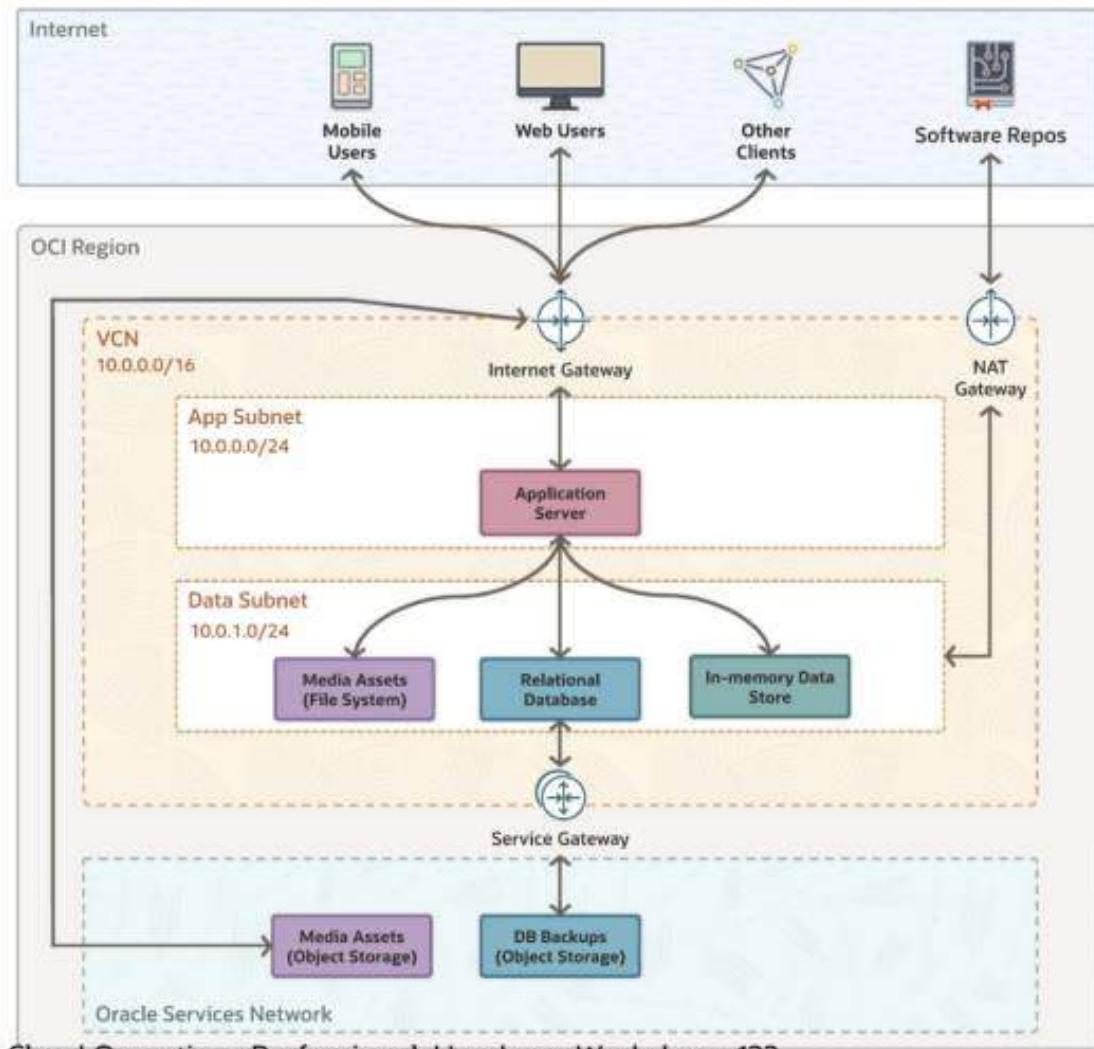
## VCN Gateways



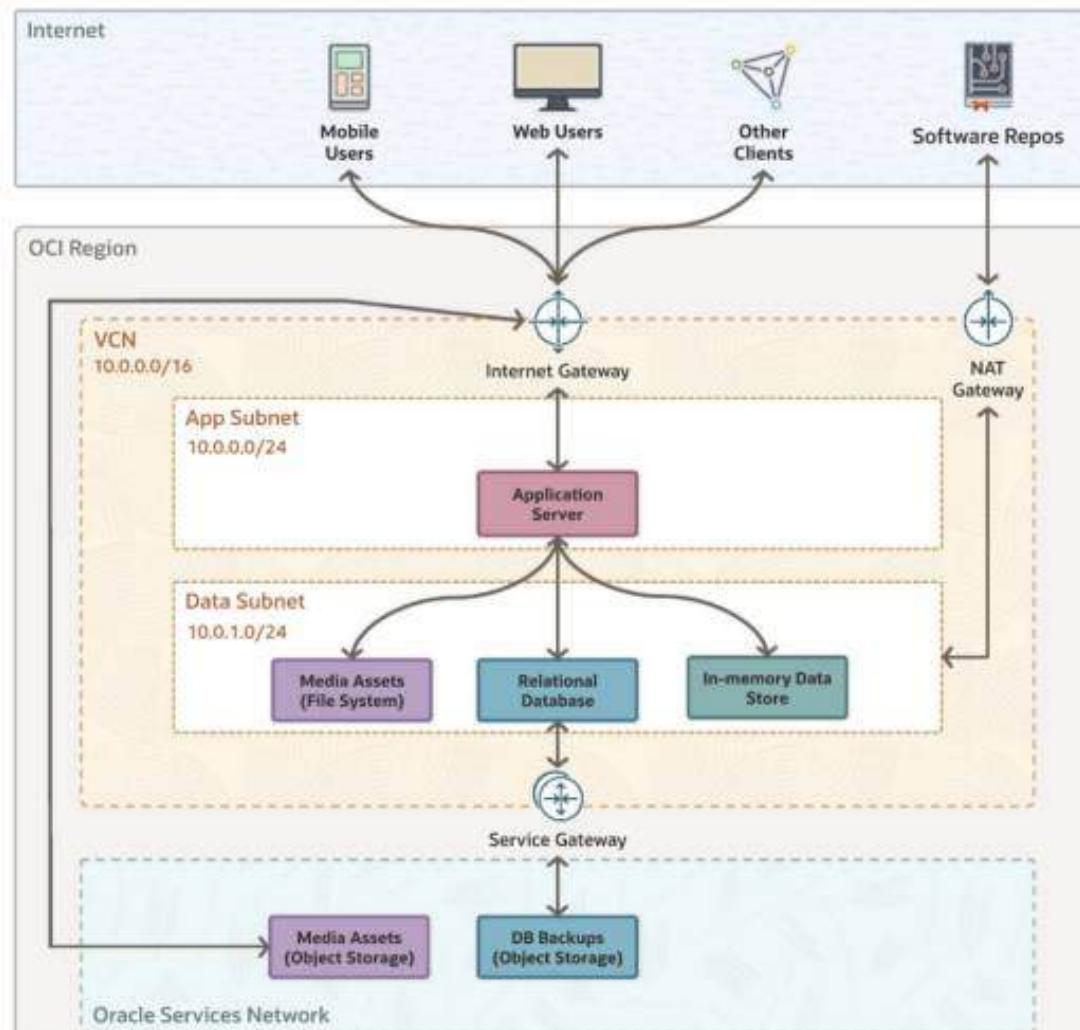
## Conceptual Design



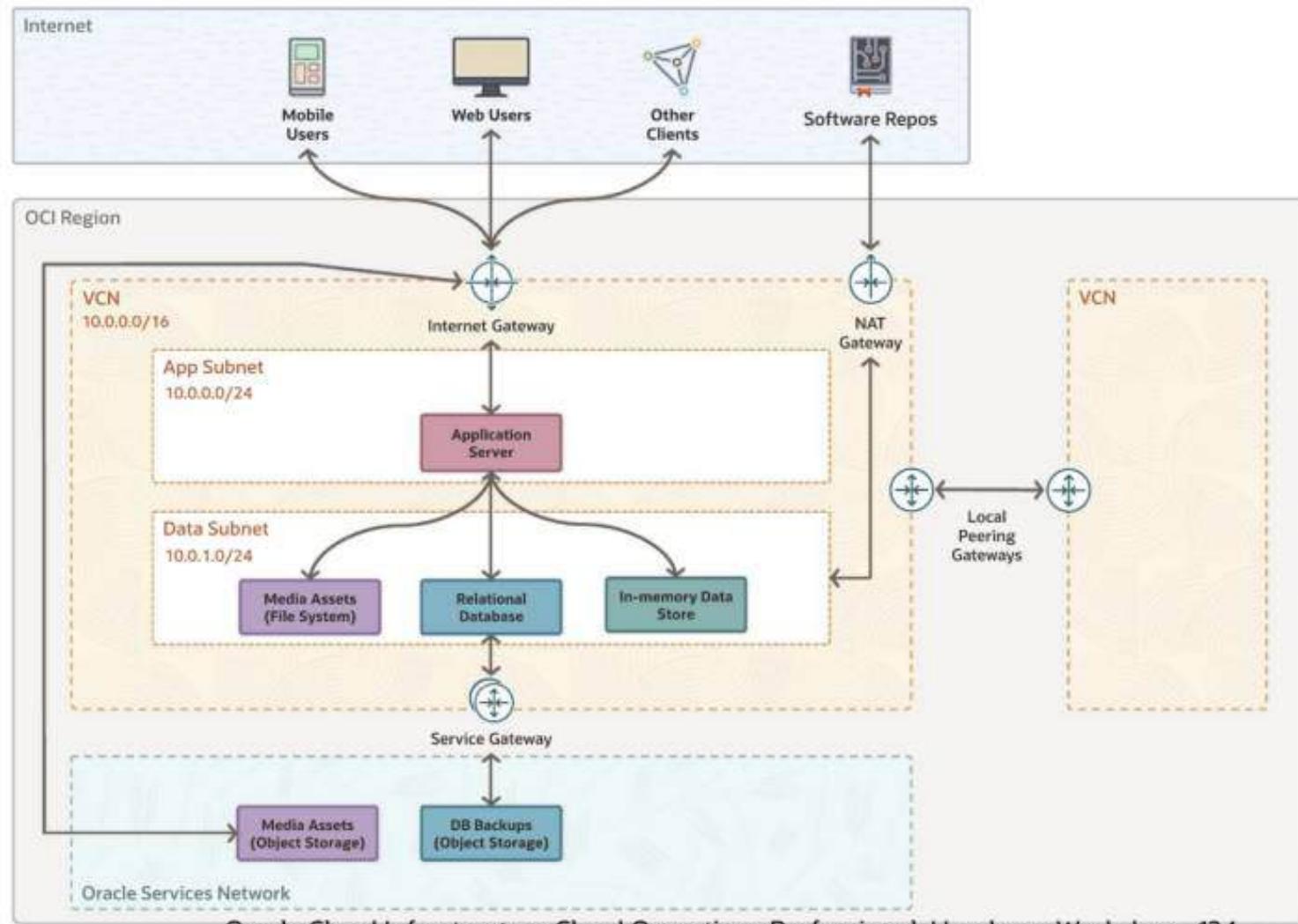
## VCN Gateways



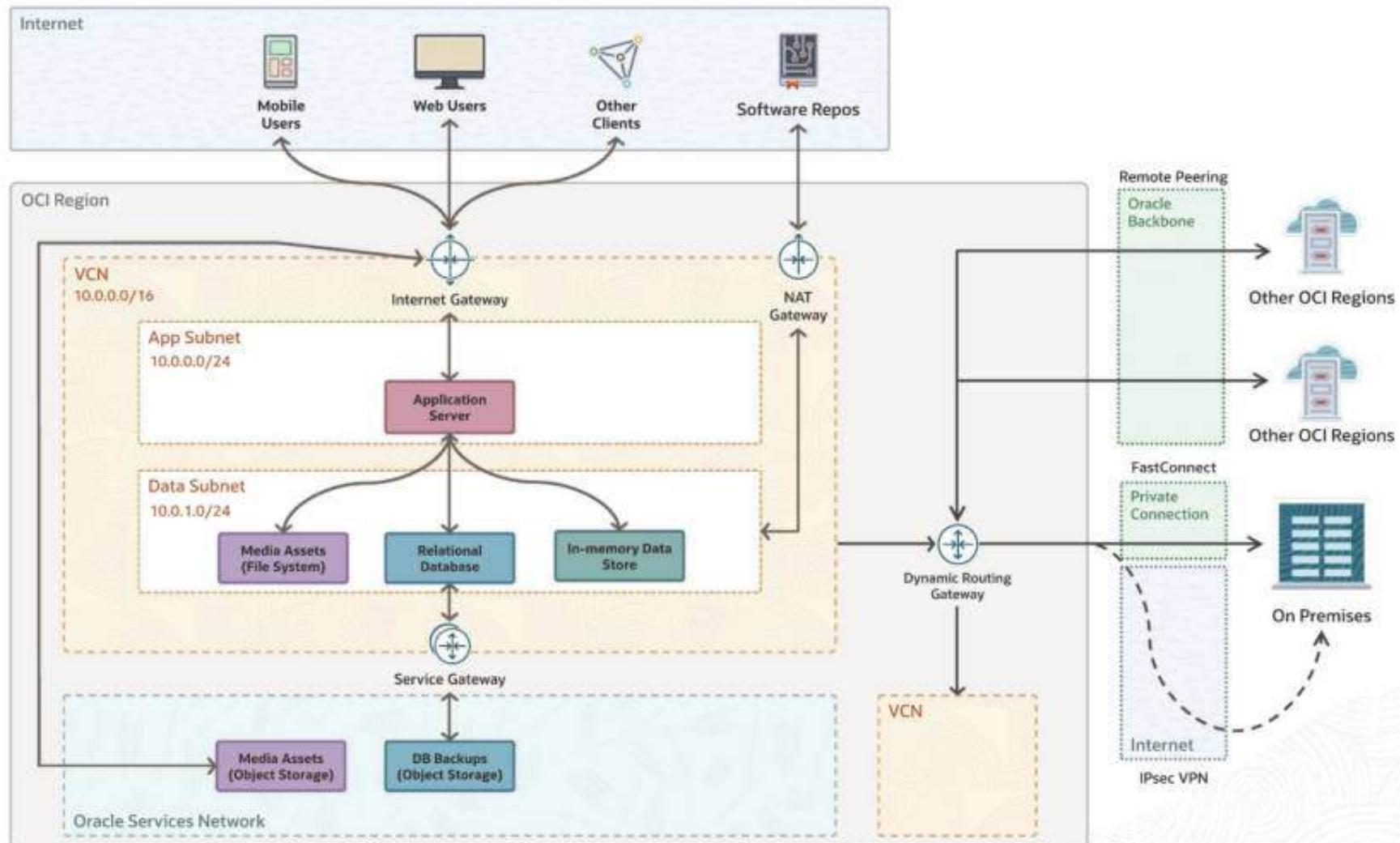
## VCN Gateways



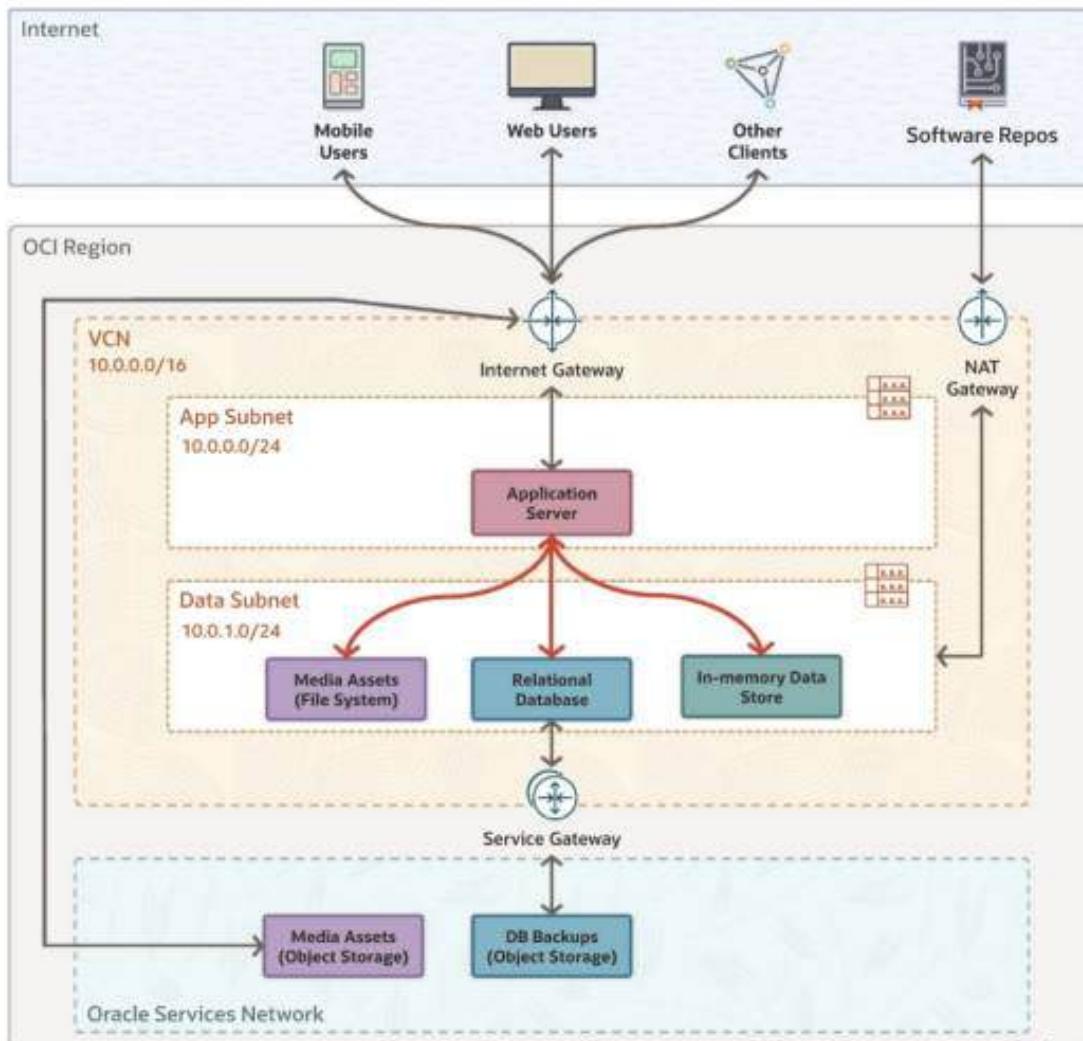
## VCN Gateways



## VCN Gateways



## VCN Gateways

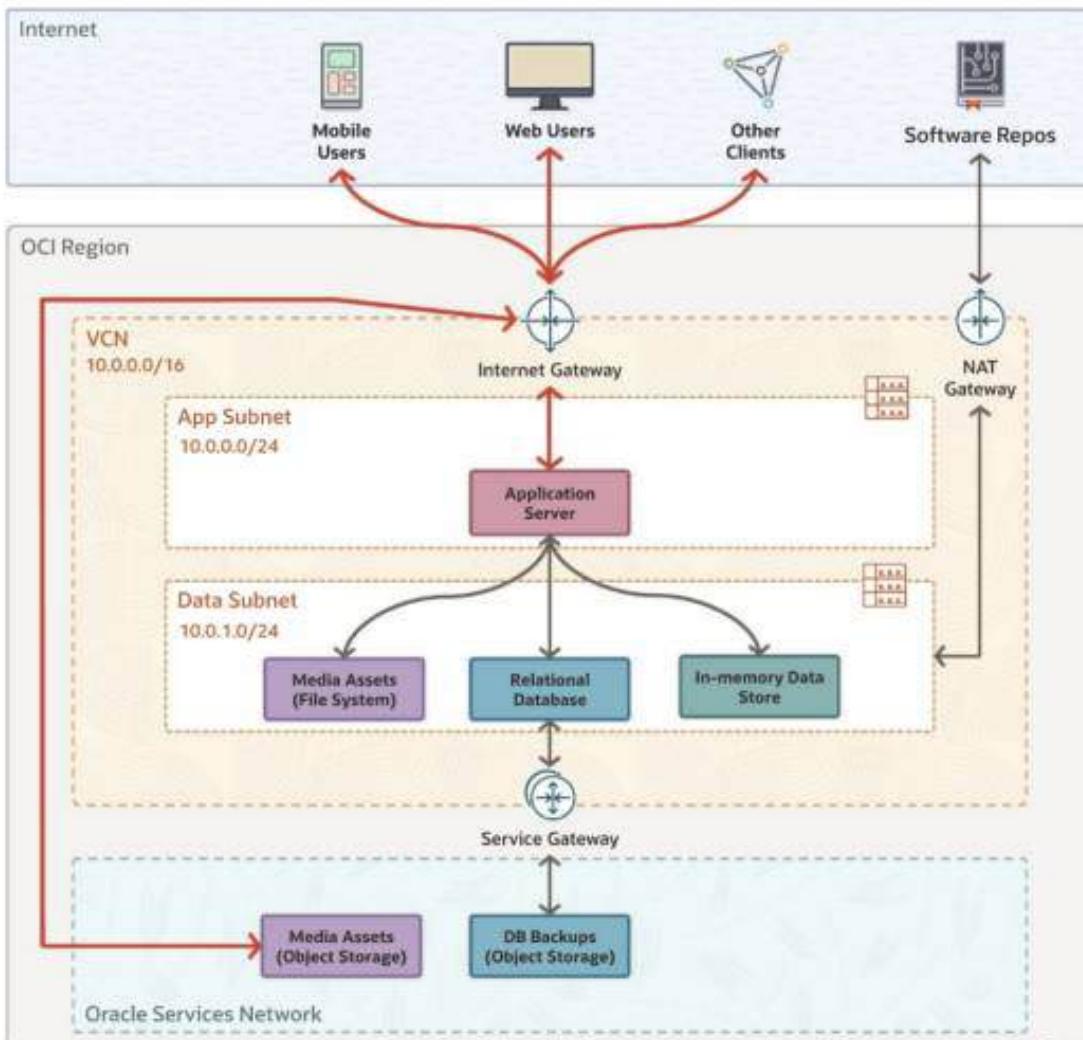


## VCN Routing

Public Route Table

Destination	Target
VCN	Implicit Rules
0.0.0.0/0	Internet Gateway

## VCN Gateways

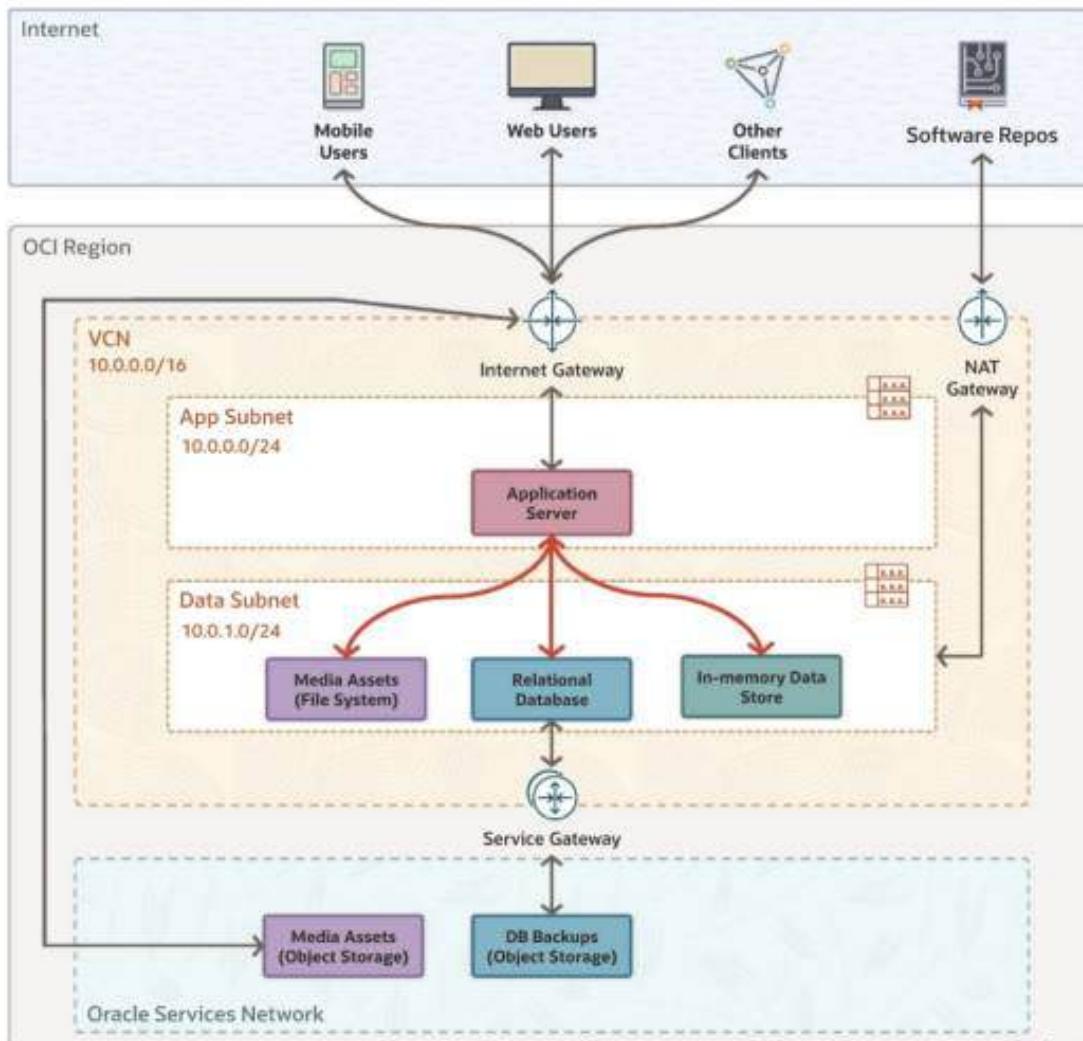


## VCN Routing

Public Route Table

Destination	Target
VCN	Implicit Rules
0.0.0.0/0	Internet Gateway

## VCN Gateways



## VCN Routing

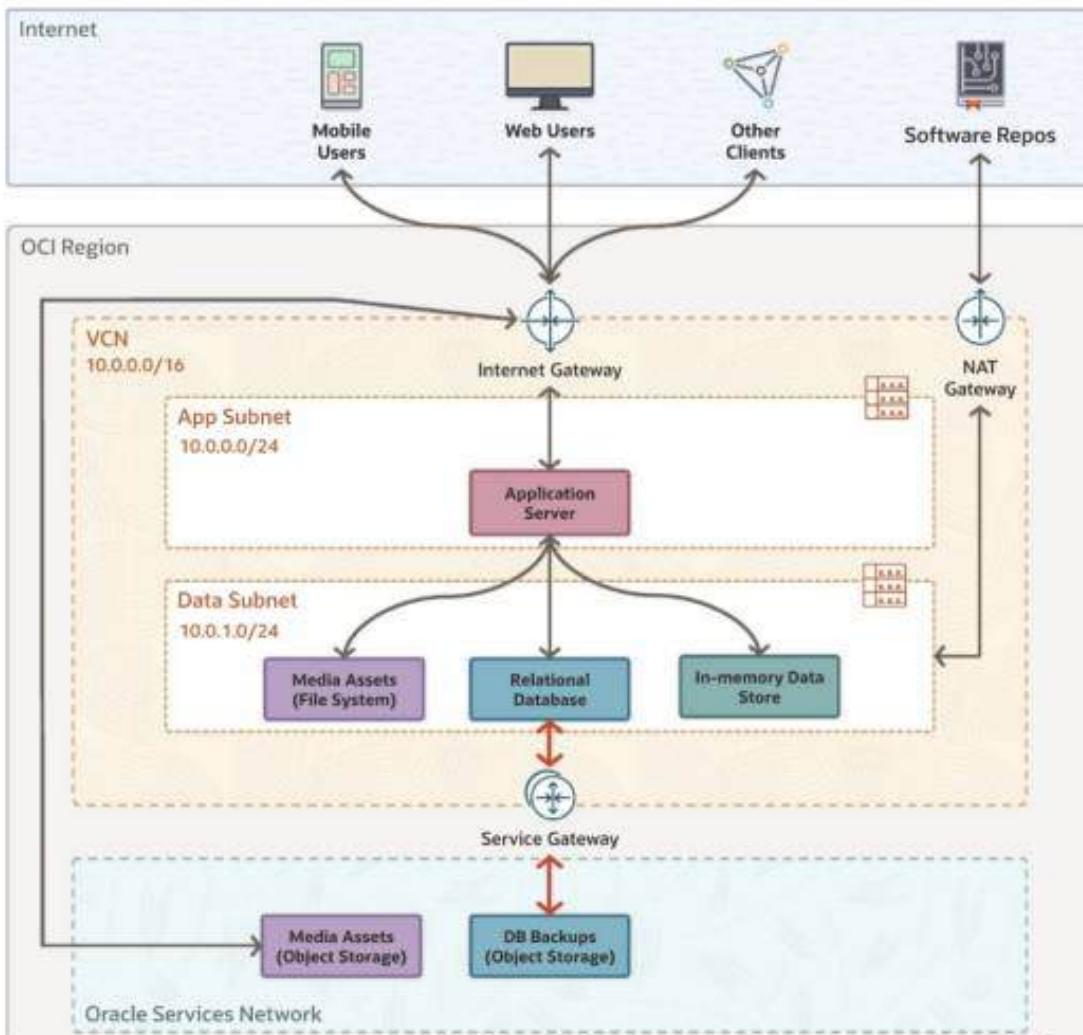
Public Route Table

Destination	Target
VCN	Implicit Rules
0.0.0.0/0	Internet Gateway

Private Route Table

Destination	Target
VCN	Implicit Rules
Oracle Services	Service Gateway
0.0.0.0/0	NAT Gateway

## VCN Gateways



## VCN Routing

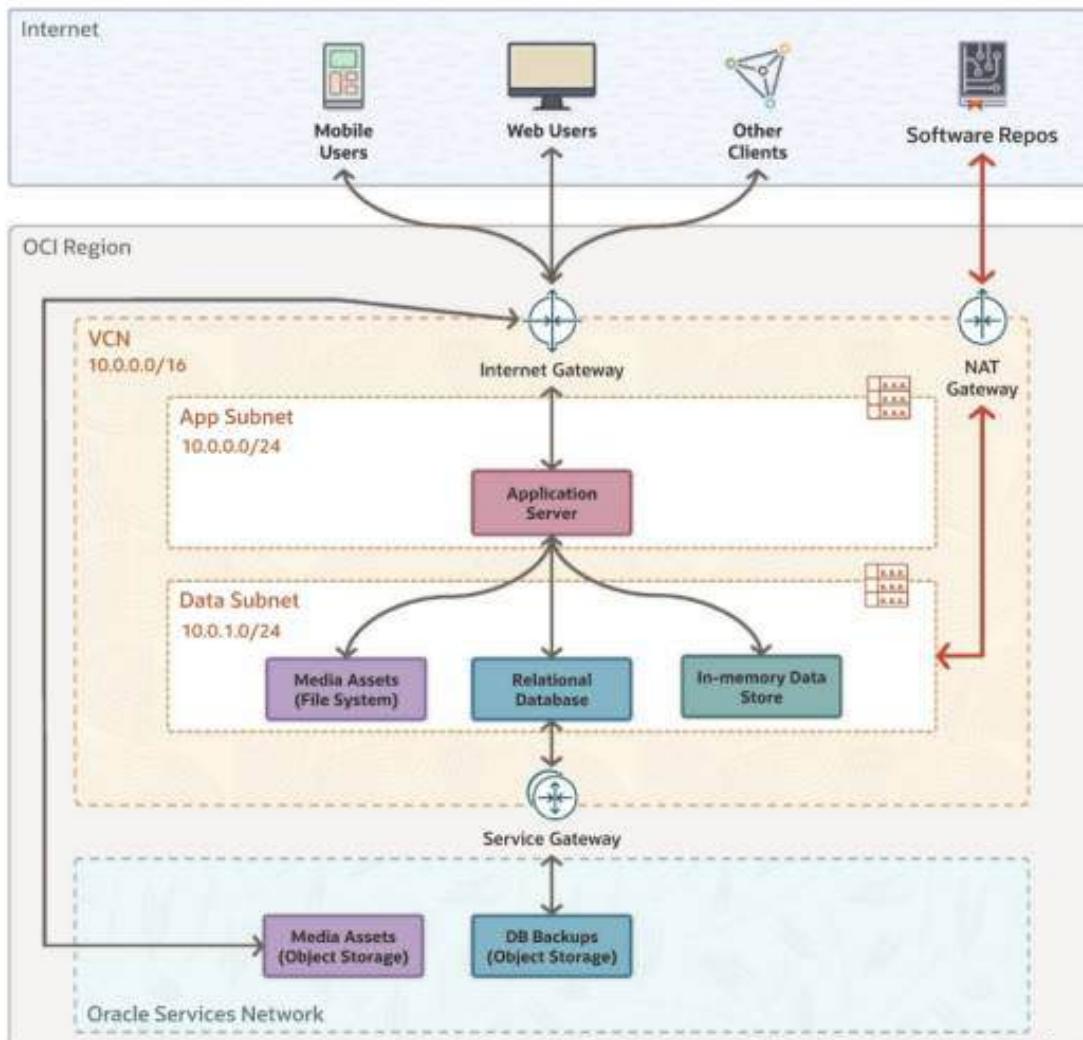
Public Route Table

Destination	Target
VCN	Implicit Rules
0.0.0.0/0	Internet Gateway

Private Route Table

Destination	Target
VCN	Implicit Rules
Oracle Services	Service Gateway
0.0.0.0/0	NAT Gateway

## VCN Gateways



## VCN Routing

Public Route Table

Destination	Target
VCN	Implicit Rules
0.0.0.0/0	Internet Gateway

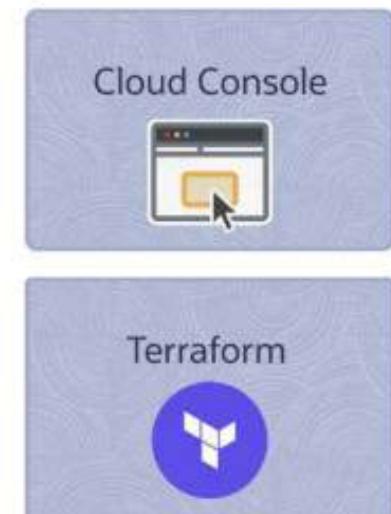
Private Route Table

Destination	Target
VCN	Implicit Rules
Oracle Services	Service Gateway
0.0.0.0/0	NAT Gateway

## VCN Layout and Gateways



Up next...





## Demo: Setting up a VCN

### OCI Cloud Operations



## Demo: Stack Creation

### OCI Cloud Operations



## Demo: VCN Creation Terraform

---

### OCI Cloud Operations

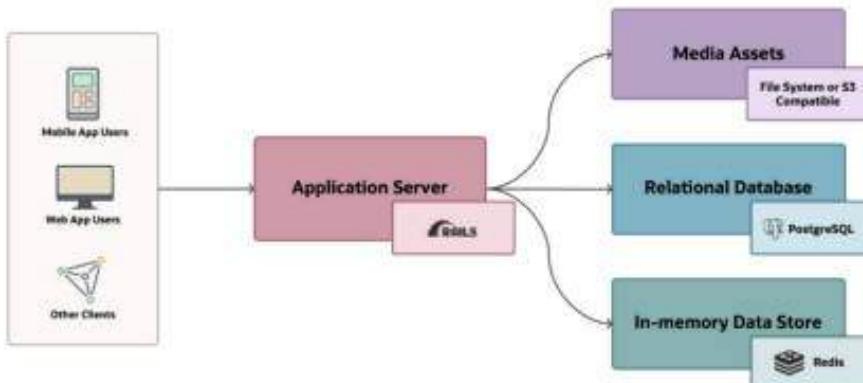


# VCN Deep Dive: Access Control

## OCI Cloud Operations

# Virtual Networking

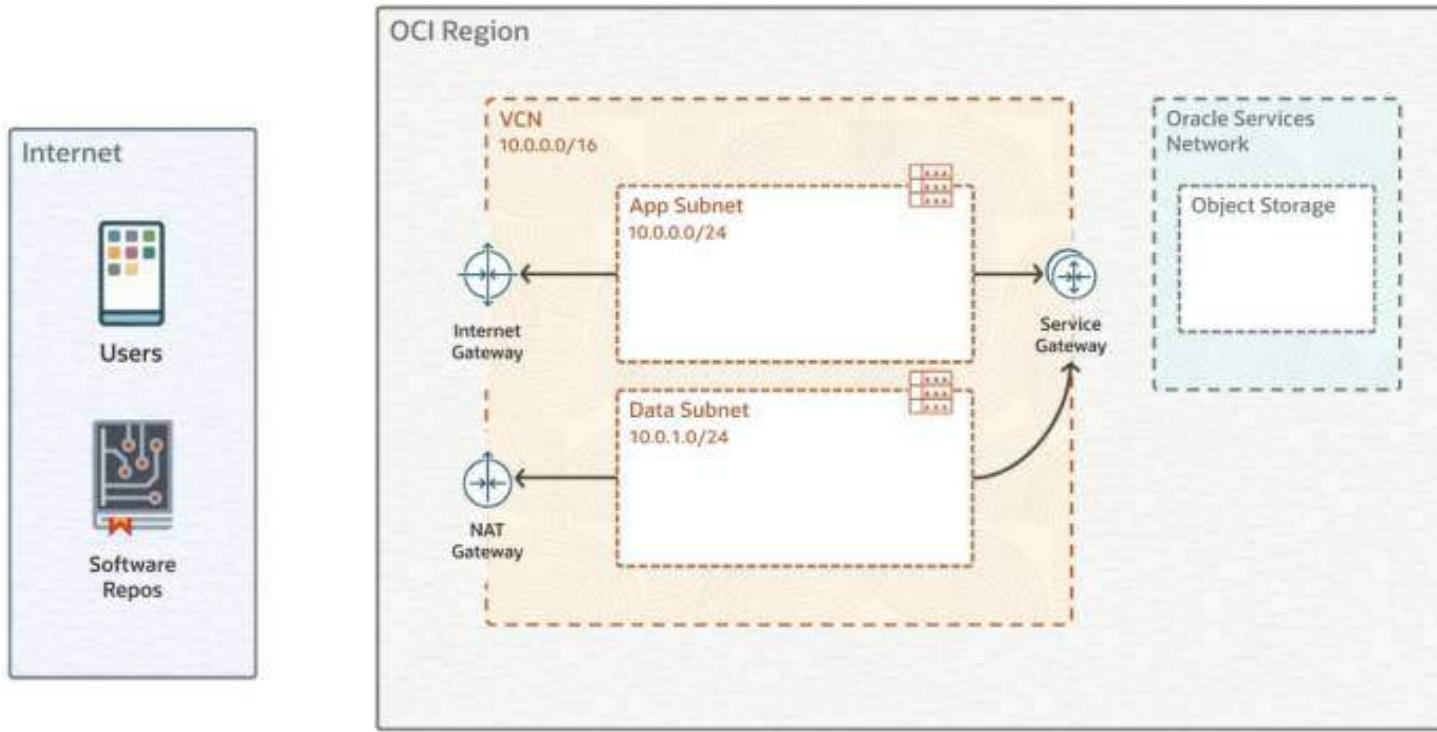
## Conceptual Design



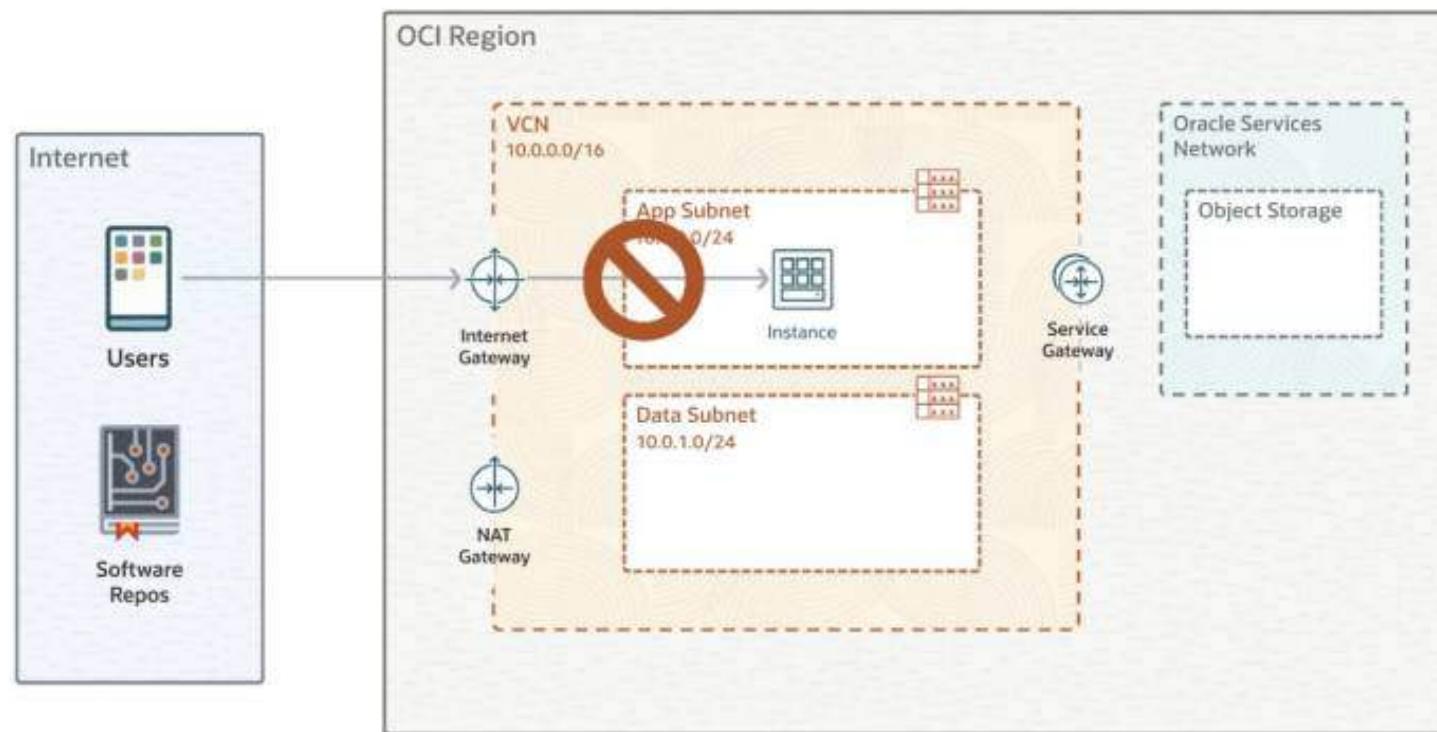
## Our approach to networking

- 1 Layout
- 2 Gateways and Routing
- 3 Access and Security

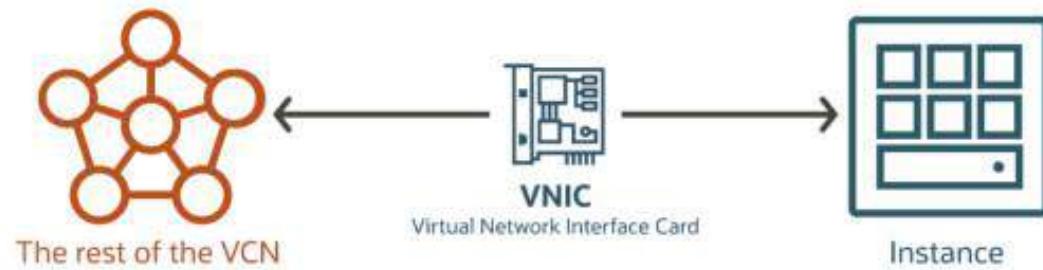
## VCN Access Control



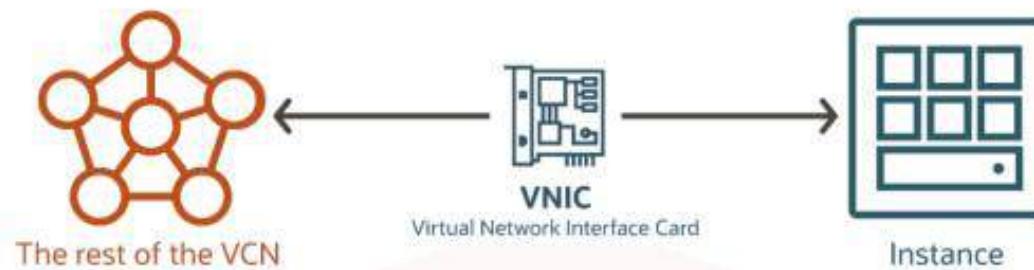
## VCN Access Control



## VCN Access Control



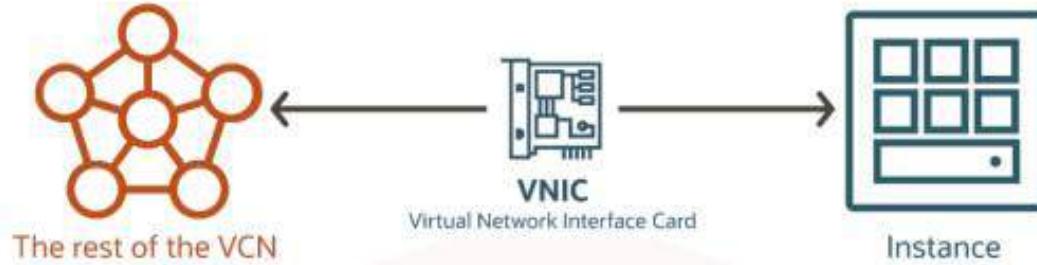
## VCN Access Control



### Allowlist Ingress

Source	Protocol	
CIDR Block	TCP	
NSG	UDP	
Service	ICMP	
	:	
	IP (all)	

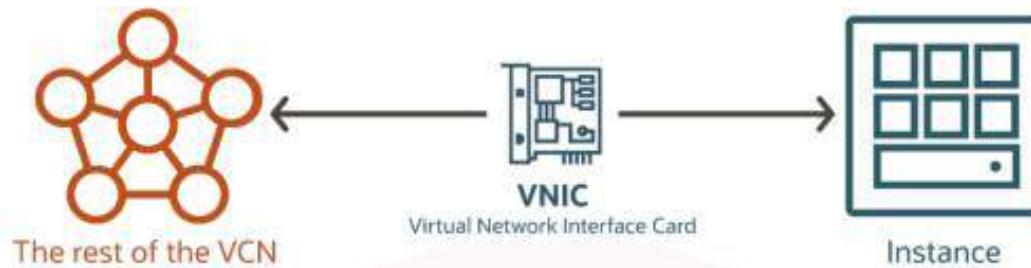
## VCN Access Control



### Allowlist Ingress

Source	Protocol	Protocol Details		
		Source Port (TCP/UDP)	Destination Port (TCP/UDP)	Type & Code (ICMP)
CIDR Block	TCP	80,443	80,443	
NSG	UDP			
Service	ICMP			Type 8
	:			
	IP (all)			

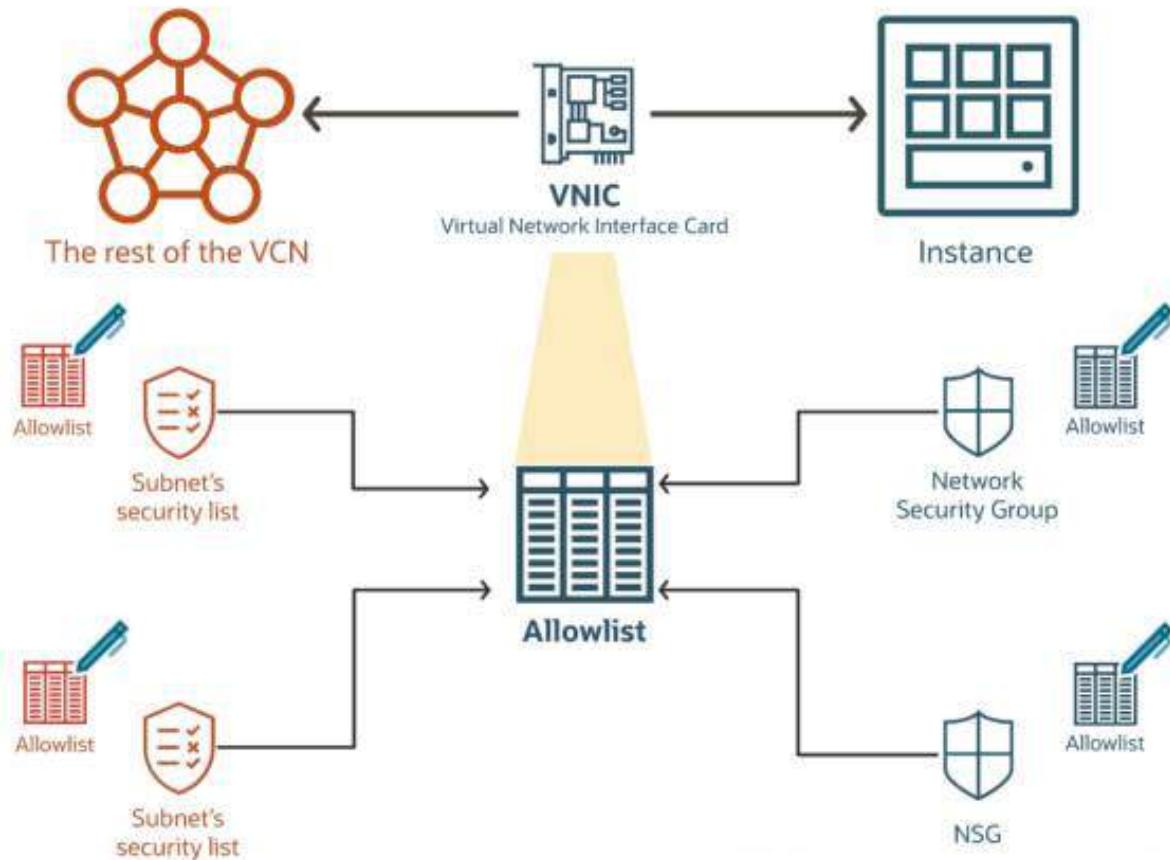
## VCN Access Control

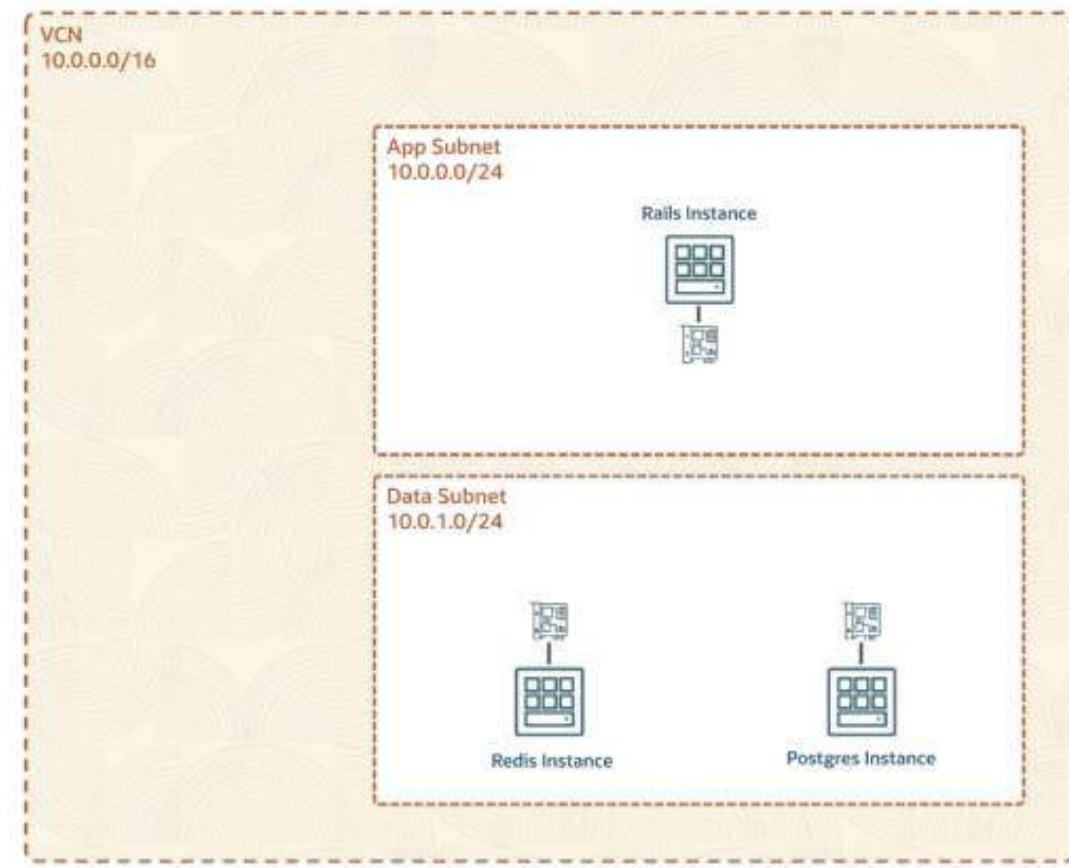
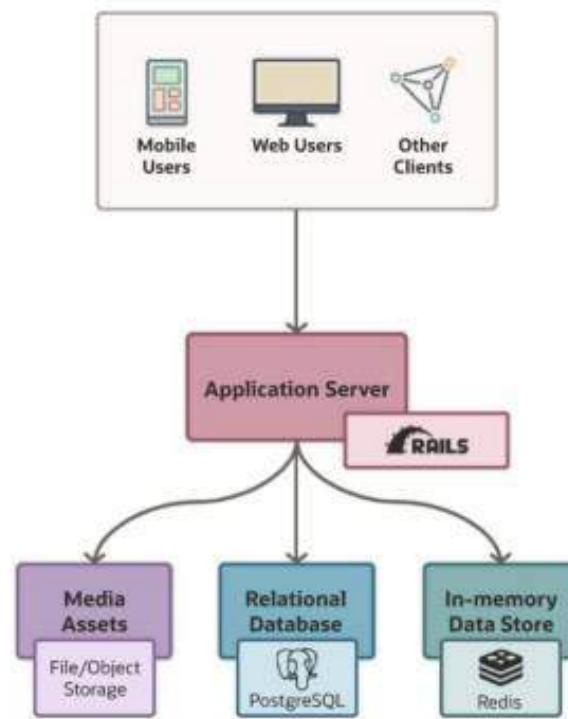


### Allowlist Ingress

Source	Protocol	Protocol Details		
		Source Port (TCP/UDP)	Destination Port (TCP/UDP)	Type & Code (ICMP)
HTTP(S) from anywhere	0.0.0.0/0	TCP	80,443	
HTTP(S) from the VCN	10.0.0.0/16	TCP	80,443	
Ping from the VCN	10.0.0.0/16	ICMP		Type 8
MySQL from the VCN	10.0.0.0/16	TCP	3306	
Postgres from the VCN	10.0.0.0/16	TCP	5432	

## VCN Access Control



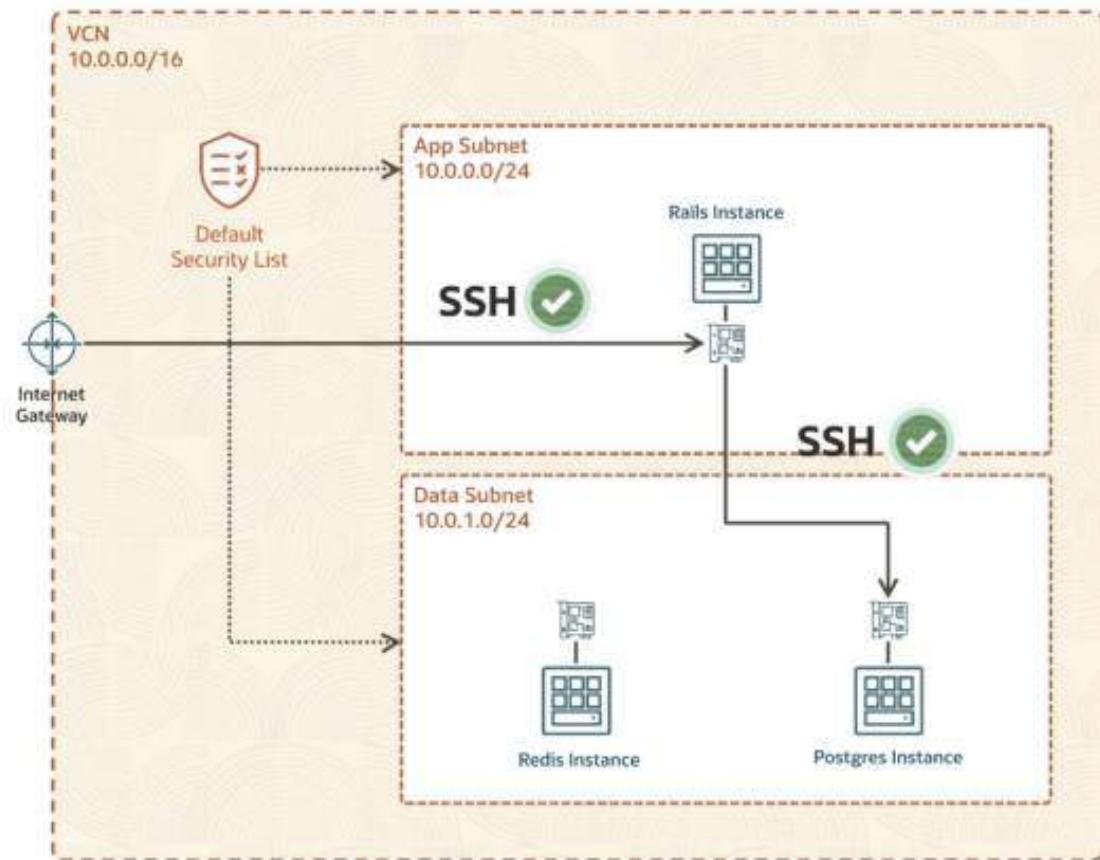


### Default Security List (Ingress)

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	22 (SSH)	
0.0.0.0/0	ICMP		Type 3/Code 4
10.0.0.0/16	ICMP		Type 3

### Default Security List (Egress)

Destination	Protocol	Destination Port	Type/Code
0.0.0.0/0	All	All	All



**Default Security List (Ingress)**

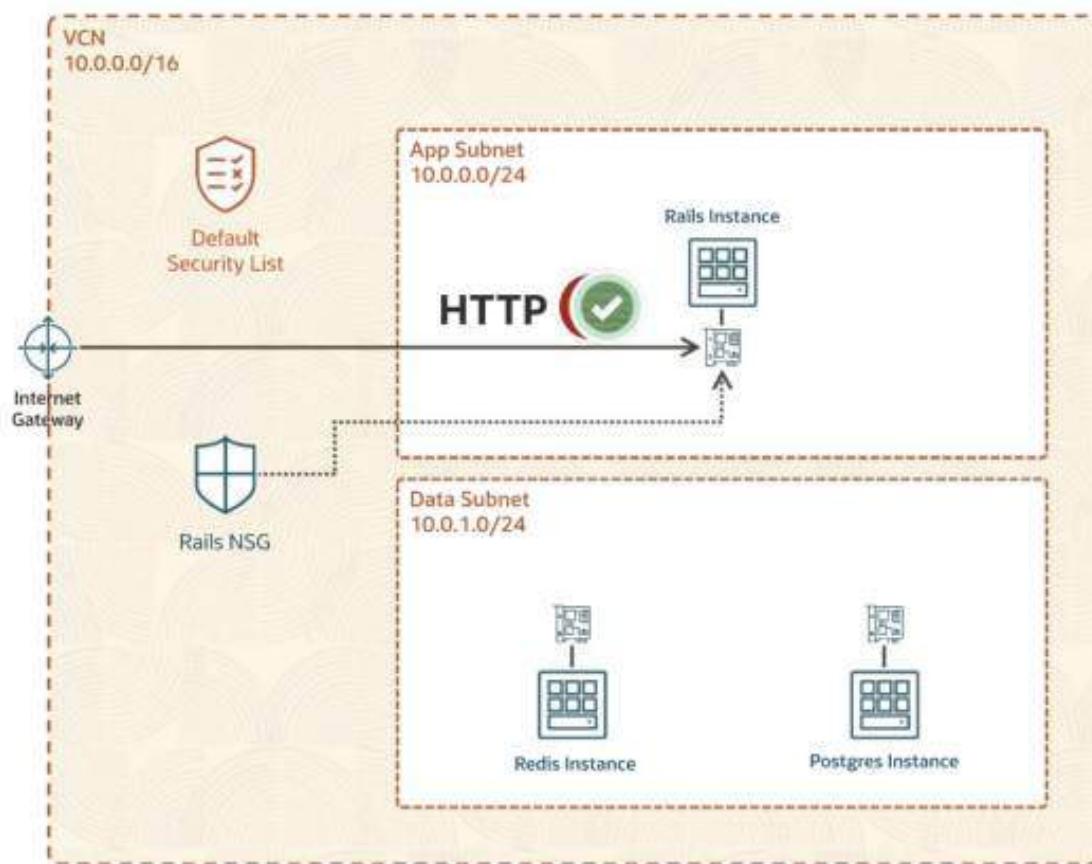
Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	22 (SSH)	
0.0.0.0/0	ICMP		Type 3/Code 4
10.0.0.0/16	ICMP		Type 3

**Default Security List (Egress)**

Destination	Protocol	Destination Port	Type/Code
0.0.0.0/0	All	All	All

**Rails NSG**

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	80,443	



#### Default Security List (Ingress)

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	22 (SSH)	
0.0.0.0/0	ICMP		Type 3/Code 4
10.0.0.0/16	ICMP		Type 3

#### Default Security List (Egress)

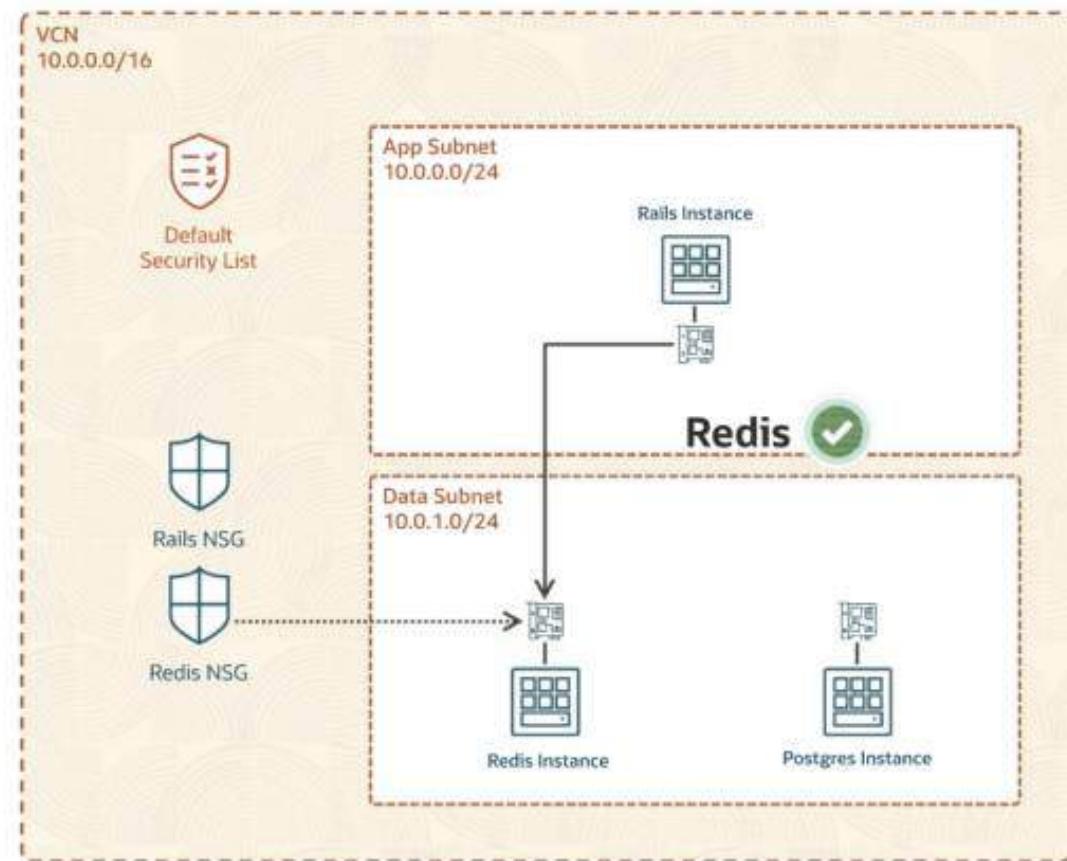
Destination	Protocol	Destination Port	Type/Code
0.0.0.0/0	All	All	All

#### Rails NSG

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	80,443	

#### Redis NSG

Source	Protocol	Destination Port	Type/Code
10.0.0.0/16	TCP	6379	



**Default Security List (Ingress)**

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	22 (SSH)	
0.0.0.0/0	ICMP		Type 3/Code 4
10.0.0.0/16	ICMP		Type 3

**Default Security List (Egress)**

Destination	Protocol	Destination Port	Type/Code
0.0.0.0/0	All	All	All

**Rails NSG**

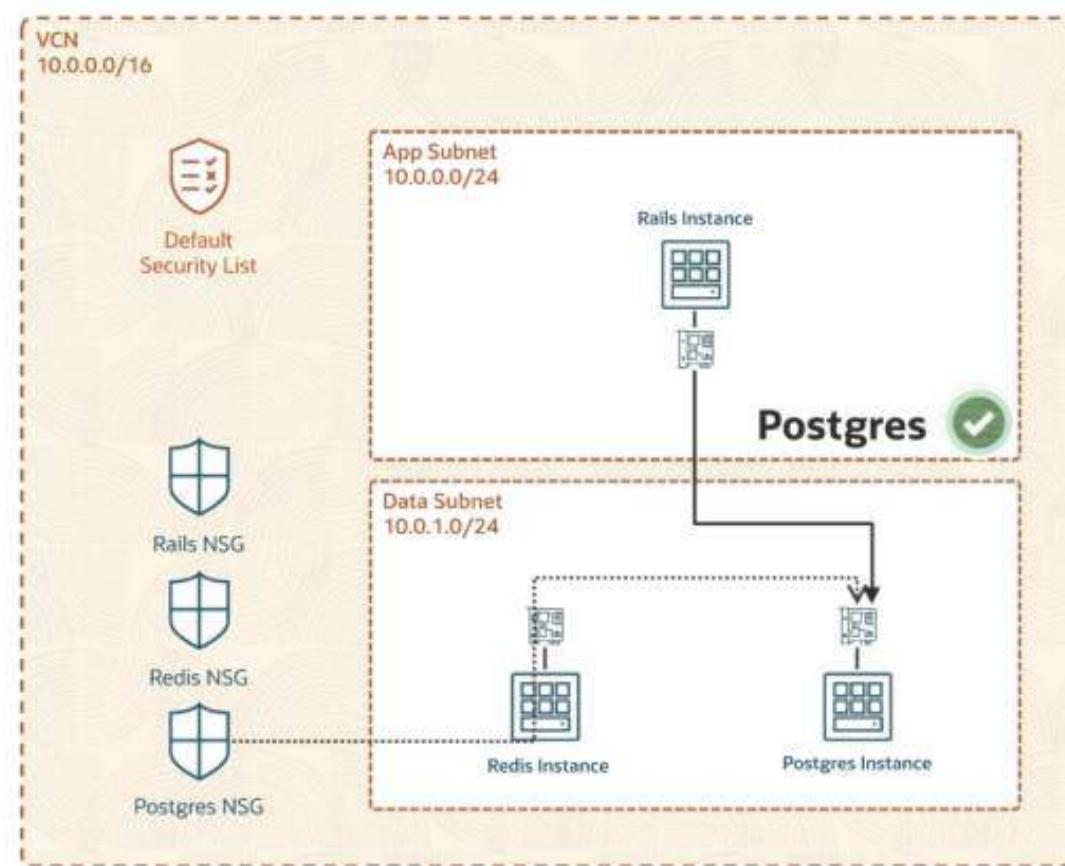
Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	80,443	

**Redis NSG**

Source	Protocol	Destination Port	Type/Code
10.0.0.0/16	TCP	6379	

**Postgres NSG**

Source	Protocol	Destination Port	Type/Code
10.0.0.0/16	TCP	5432	



#### Default Security List (Ingress)

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	22 (SSH)	
0.0.0.0/0	ICMP		Type 3/Code 4
10.0.0.0/16	ICMP		Type 3

#### Default Security List (Egress)

Destination	Protocol	Destination Port	Type/Code
0.0.0.0/0	All	All	All

#### Rails NSG

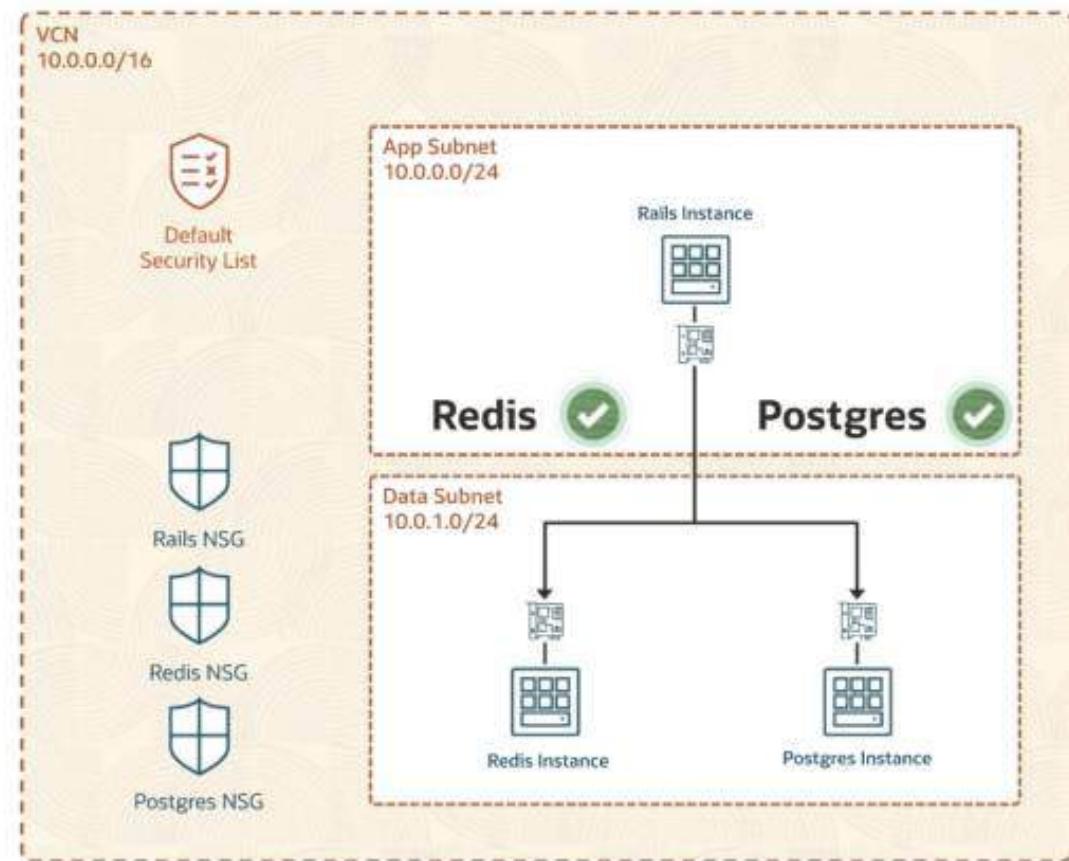
Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	80,443	

#### Redis NSG

Source	Protocol	Destination Port	Type/Code
Rails NSG	TCP	6379	

#### Postgres NSG

Source	Protocol	Destination Port	Type/Code
Rails NSG	TCP	5432	



**Default Security List (Ingress)**

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	22 (SSH)	
0.0.0.0/0	ICMP		Type 3/Code 4
10.0.0.0/16	ICMP		Type 3

**Default Security List (Egress)**

Destination	Protocol	Destination Port	Type/Code
0.0.0.0/0	All	All	All

**Rails NSG**

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	80,443	

**Redis NSG**

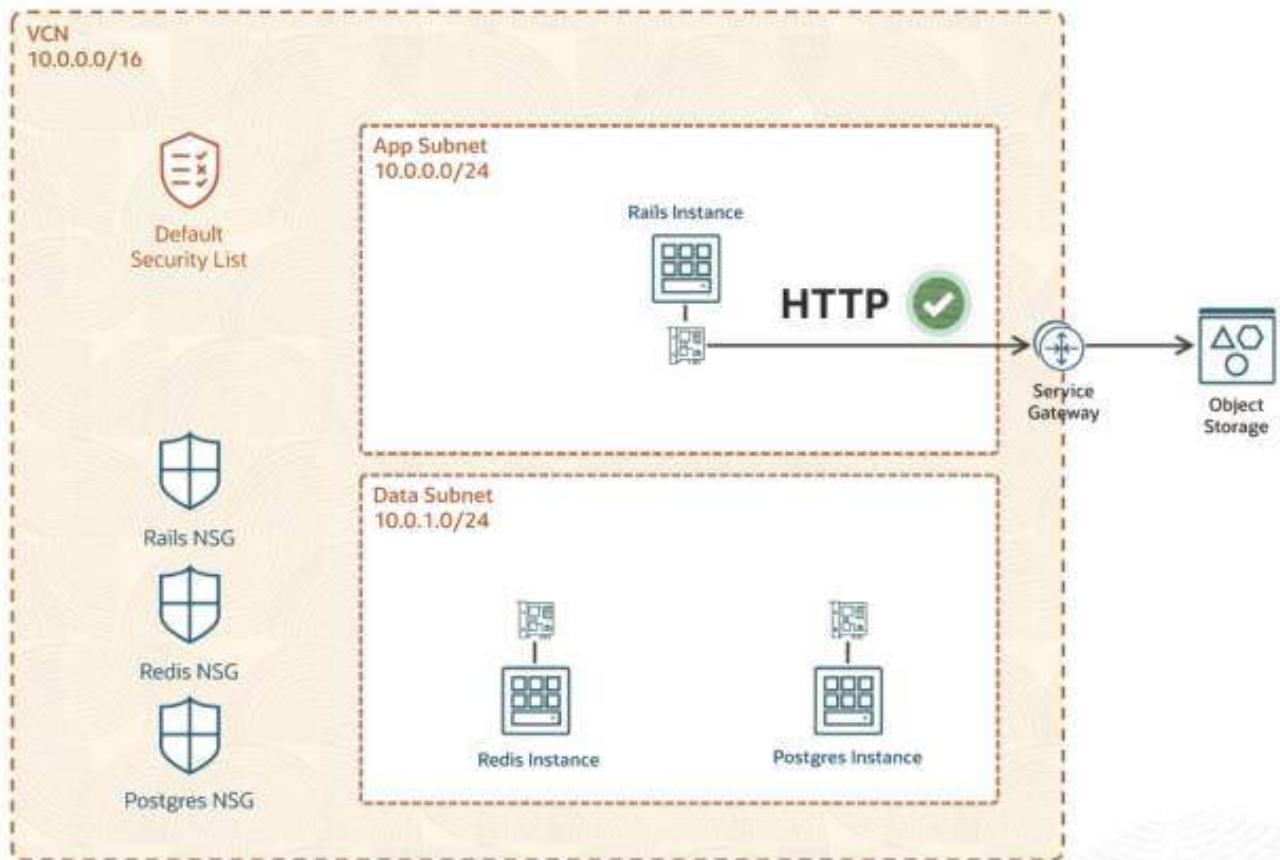
Source	Protocol	Destination Port	Type/Code
Rails NSG	TCP	6379	

**Postgres NSG**

Source	Protocol	Destination Port	Type/Code
Rails NSG	TCP	5432	



Stateful rules allow responses.



#### Default Security List (Ingress)

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	22 (SSH)	
0.0.0.0/0	ICMP		Type 3/Code 4
10.0.0.0/16	ICMP		Type 3

#### Default Security List (Egress)

Destination	Protocol	Destination Port	Type/Code
0.0.0.0/0	All	All	All

#### Rails NSG

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	80,443	

#### Redis NSG

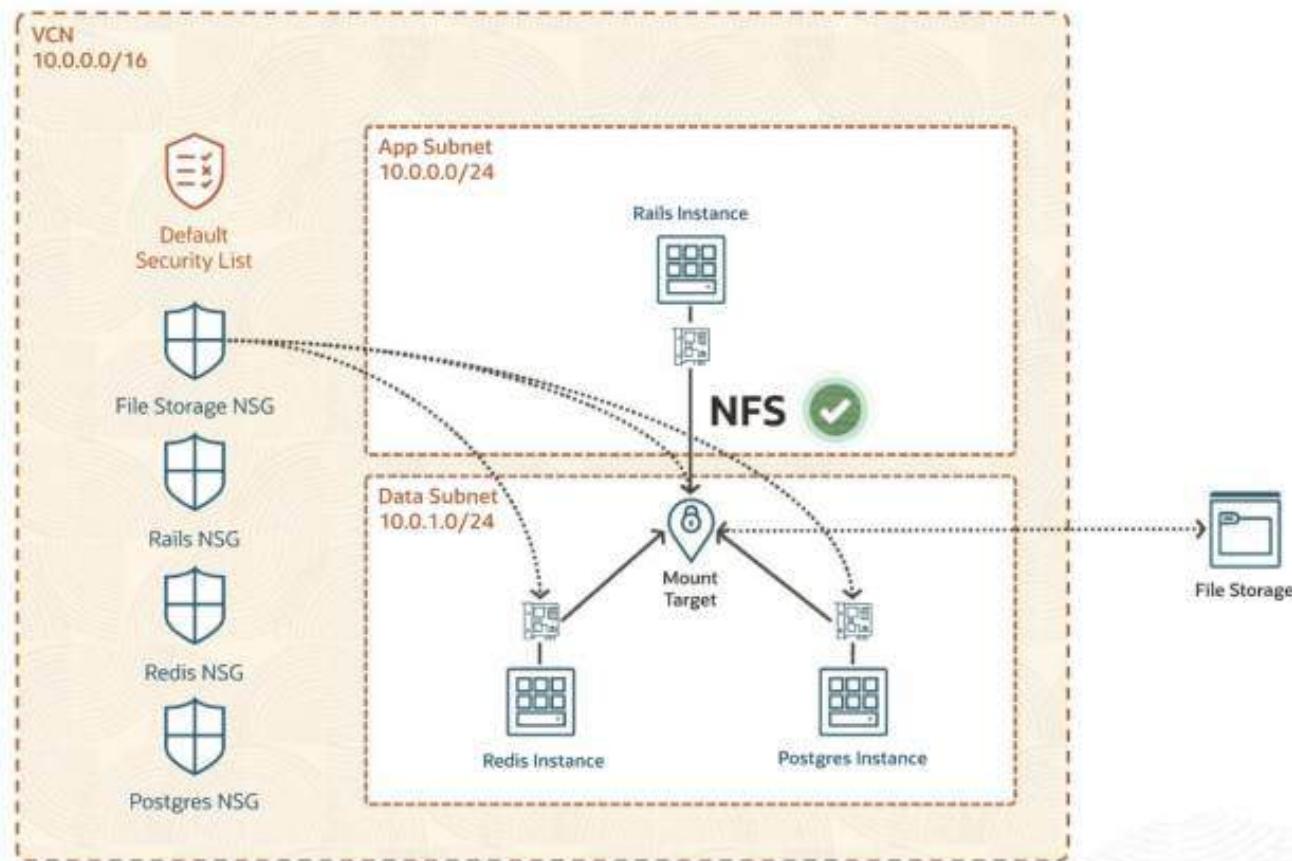
Source	Protocol	Destination Port	Type/Code
Rails NSG	TCP	6379	

#### Postgres NSG

Source	Protocol	Destination Port	Type/Code
Rails NSG	TCP	5432	

#### File Storage NSG

Source	Protocol	Destination Port	Type/Code
10.0.0.0/16	TCP	111,2048-2050	
10.0.0.0/16	UDP	111	



**Default Security List (Ingress)**

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	22 (SSH)	
0.0.0.0/0	ICMP		Type 3/Code 4
10.0.0.0/16	ICMP		Type 3

**Default Security List (Egress)**

Destination	Protocol	Destination Port	Type/Code
0.0.0.0/0	All	All	All

**Rails NSG**

Source	Protocol	Destination Port	Type/Code
0.0.0.0/0	TCP	80,443	

**Redis NSG**

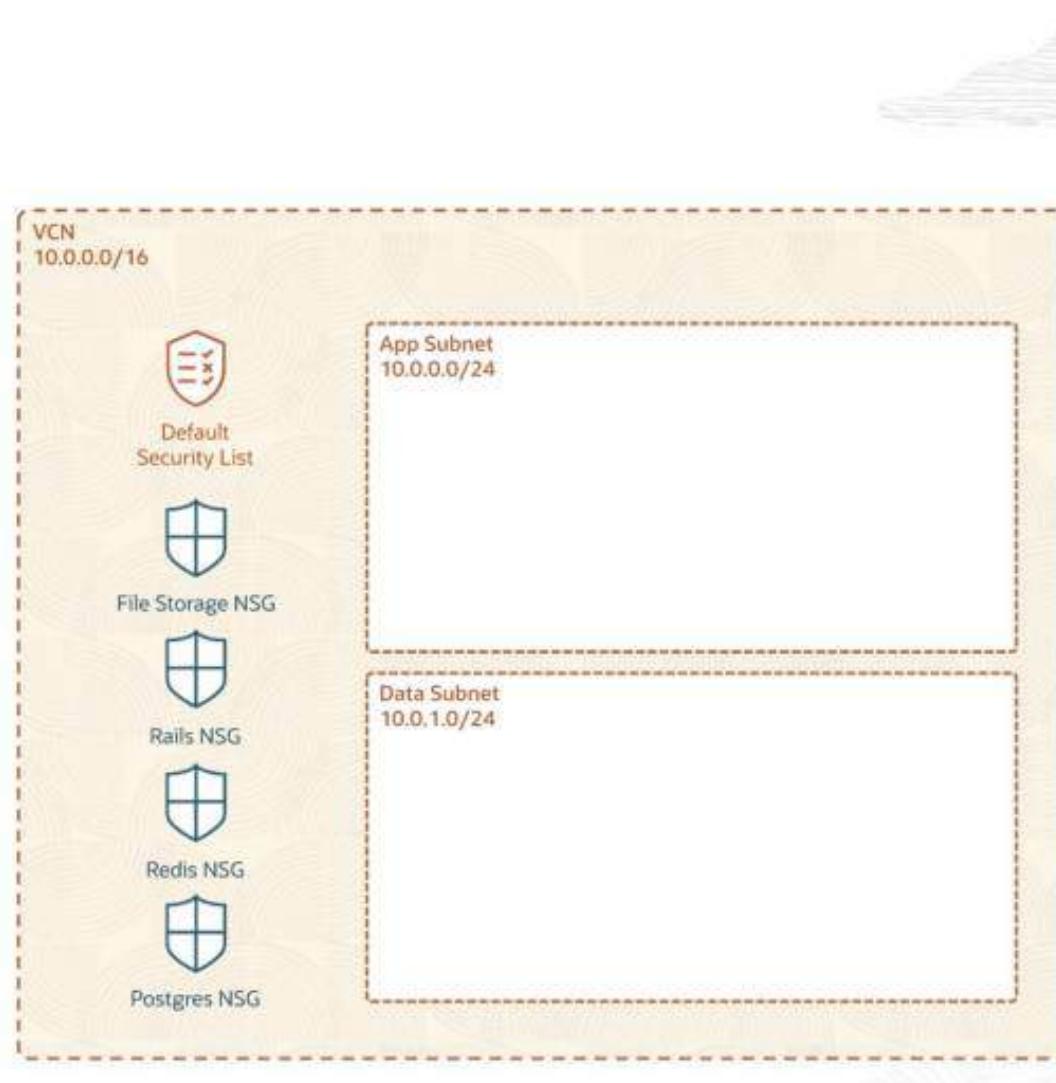
Source	Protocol	Destination Port	Type/Code
Rails NSG	TCP	6379	

**Postgres NSG**

Source	Protocol	Destination Port	Type/Code
Rails NSG	TCP	5432	

**File Storage NSG**

Source	Protocol	Destination Port	Type/Code
10.0.0.0/16	TCP	111,2048-2050	
10.0.0.0/16	UDP	111	



## Demo: Securing a VCN

---

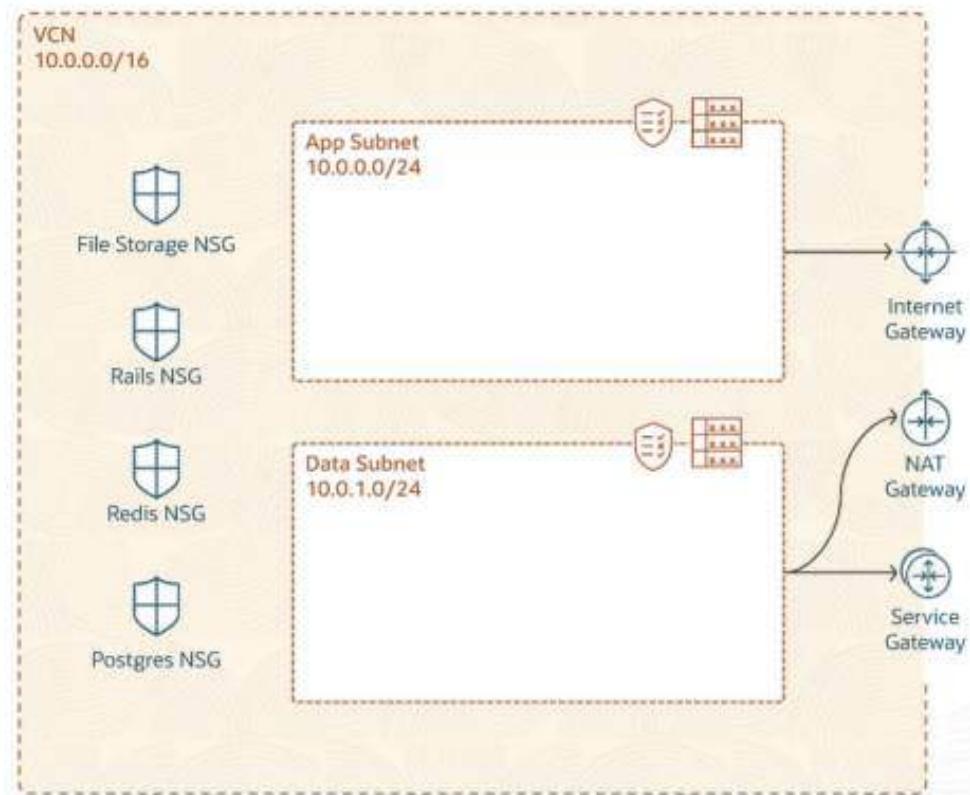
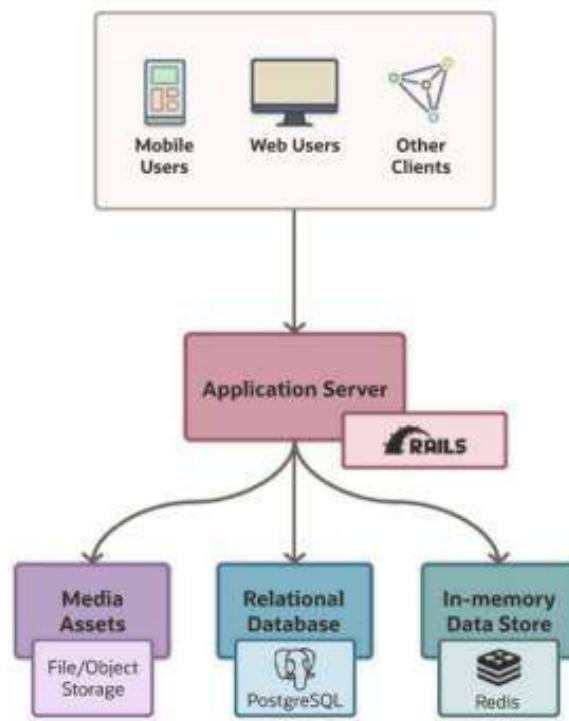
### OCI Cloud Operations



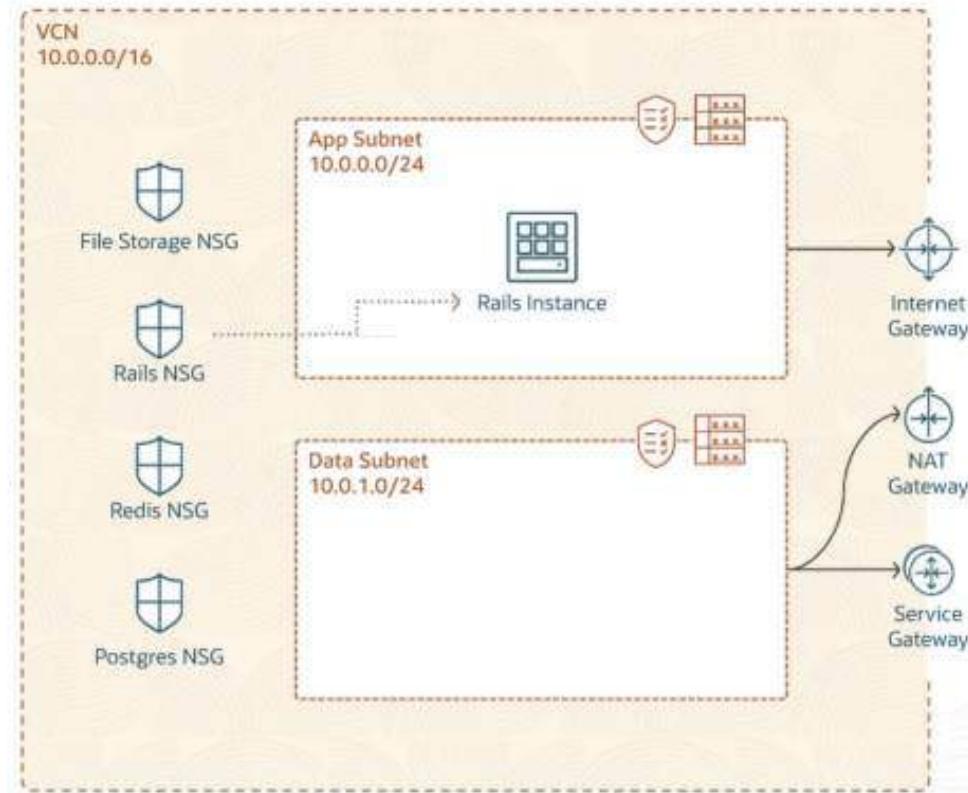
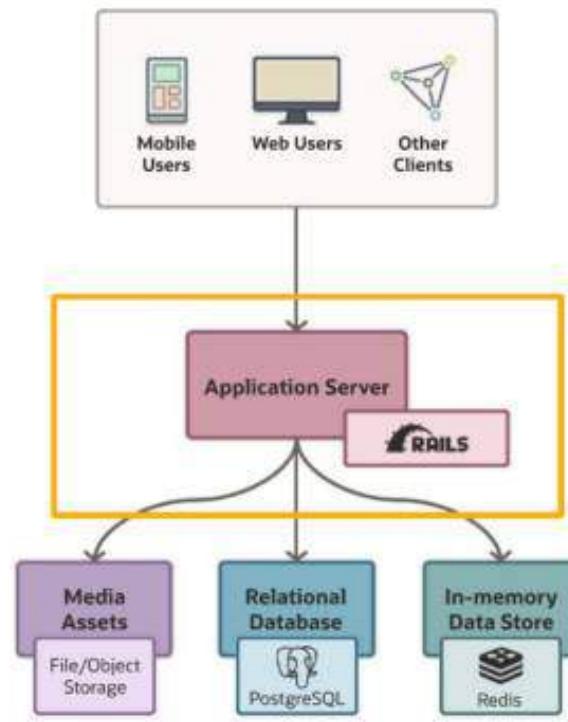
# Compute Deep Dive: The Instance Life Cycle

## OCI Cloud Operations

# Recap...

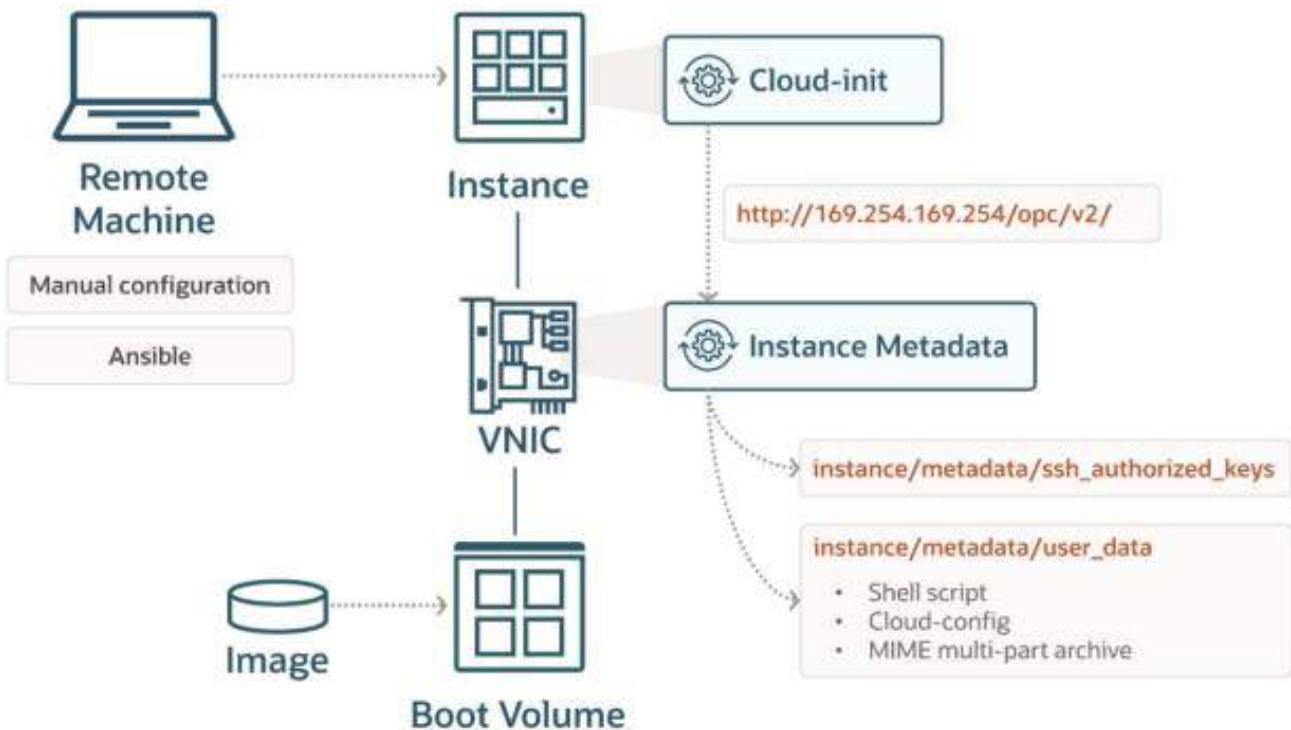


# Next...



## Instance Life Cycle

- 1 Provision shape
- 2 Use source details
- 3 Run cloud-init with instance metadata
- 4 Remote configuration





## Demo: Provision a compute instance with Terraform

---

### OCI Cloud Operations



# Compute Deep Dive: Provisioning and Sourcing

---

**OCI Cloud Operations**

Recap...

## Instance Lifecycle

### 1 Provisioning

Allocate shape and network

### 2 Sourcing

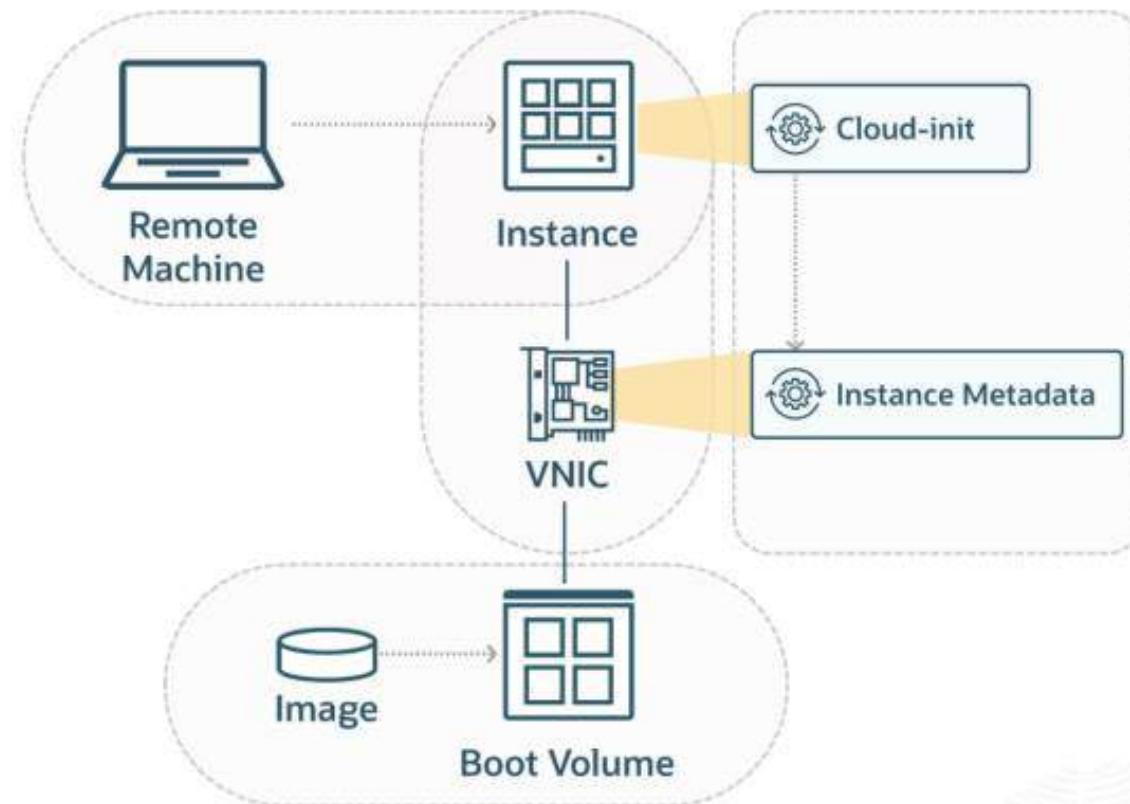
Instantiate from an image or connect existing boot volume

### 3 Bootstrapping

Run cloud-init with instance metadata

### 4 Configuration

Remotely fine-tune manually or via configuration management tools



Recap...

## Instance Lifecycle

### 1 Provisioning

Allocate shape and network

### 2 Sourcing

Instantiate from an image or connect existing boot volume

### 3 Bootstrapping

Run cloud-init with instance metadata

### 4 Configuration

Remotely fine-tune manually or via configuration management tools

### Infrastructure

CPU and memory

Network bandwidth

Storage attachment

### Base System

Operating system

Pre-installed software

### Access Control

Network firewall

Users, groups, and permissions

### System Maintenance

Storage mount/format

Software updates

### Data Management

Files and data

Environment variables

### Application Deployment

System services

Application installation

Application settings

## 1 Provisioning

### Infrastructure

CPU and memory

Network bandwidth

Storage attachment

## 2 Sourcing

### Base System

Operating system

Pre-installed software

## 3 Bootstrapping

### Access Control

Network firewall

Users, groups, and permissions

## 4 Configuration

### System Maintenance

Storage mount/format

Software updates

### Data Management

Files and data

Environment variables

### Application Deployment

System services

Application installation

Application settings



## 1 Provisioning

CPU and memory    Network bandwidth    Storage attachment

## 2 Sourcing

## 3 Bootstrapping

## 4 Configuration

### Base System

Operating system    Pre-installed software

### Access Control

Network firewall    Users, groups, and permissions

### System Maintenance

Storage mount/format    Software updates

### Data Management

Files and data    Environment variables

### Application Deployment

System services    Application installation    Application settings

Immutability



Flexibility

## 1 Provisioning

CPU and memory   Network bandwidth   Storage attachment

## 2 Sourcing

Operating system   Pre-installed software

## 3 Bootstrapping

## 4 Configuration

### Access Control

Network firewall

Users, groups, and permissions

### System Maintenance

Storage mount/format

Software updates

### Data Management

Files and data

Environment variables

### Application Deployment

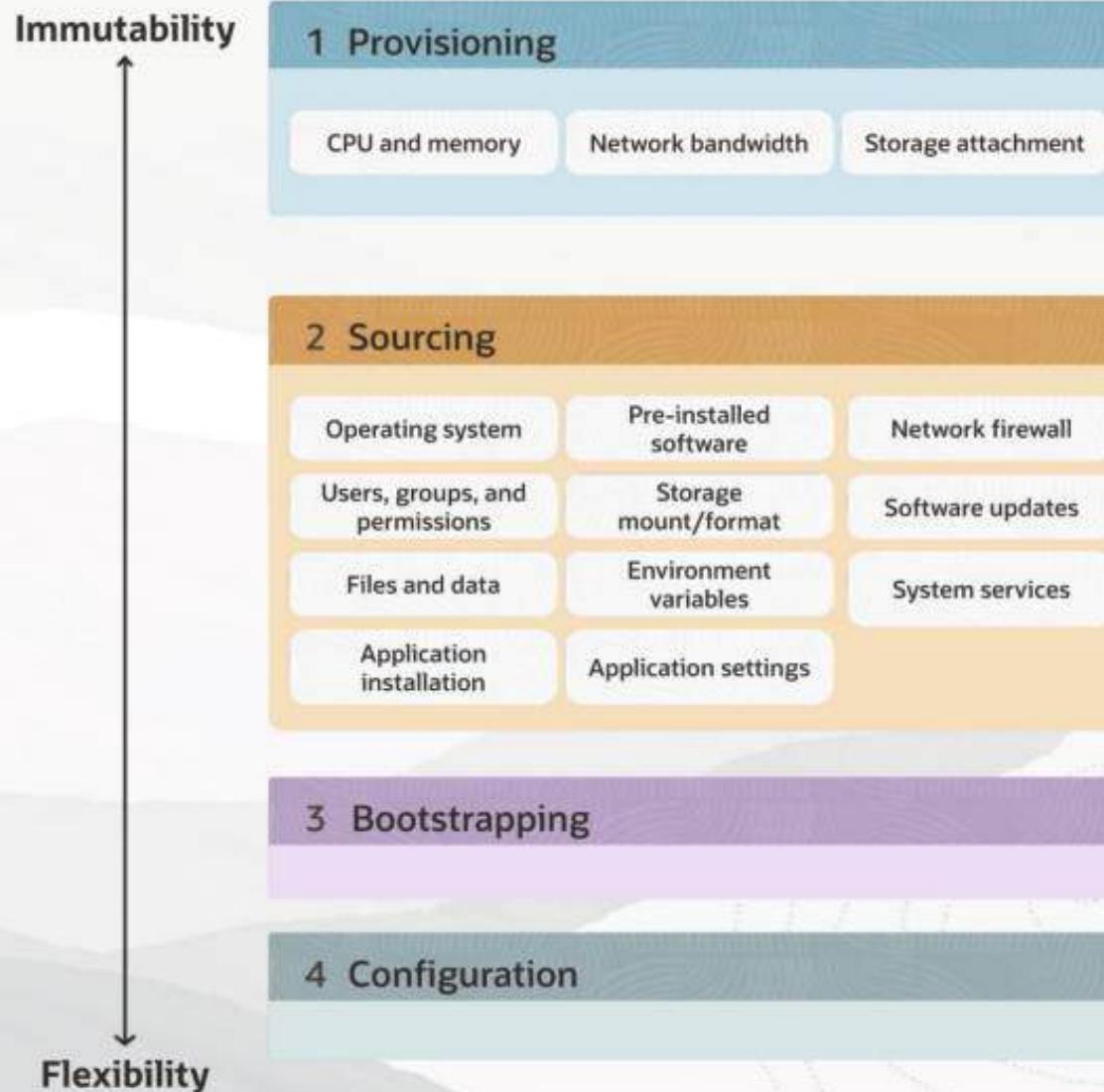
System services

Application installation

Application settings

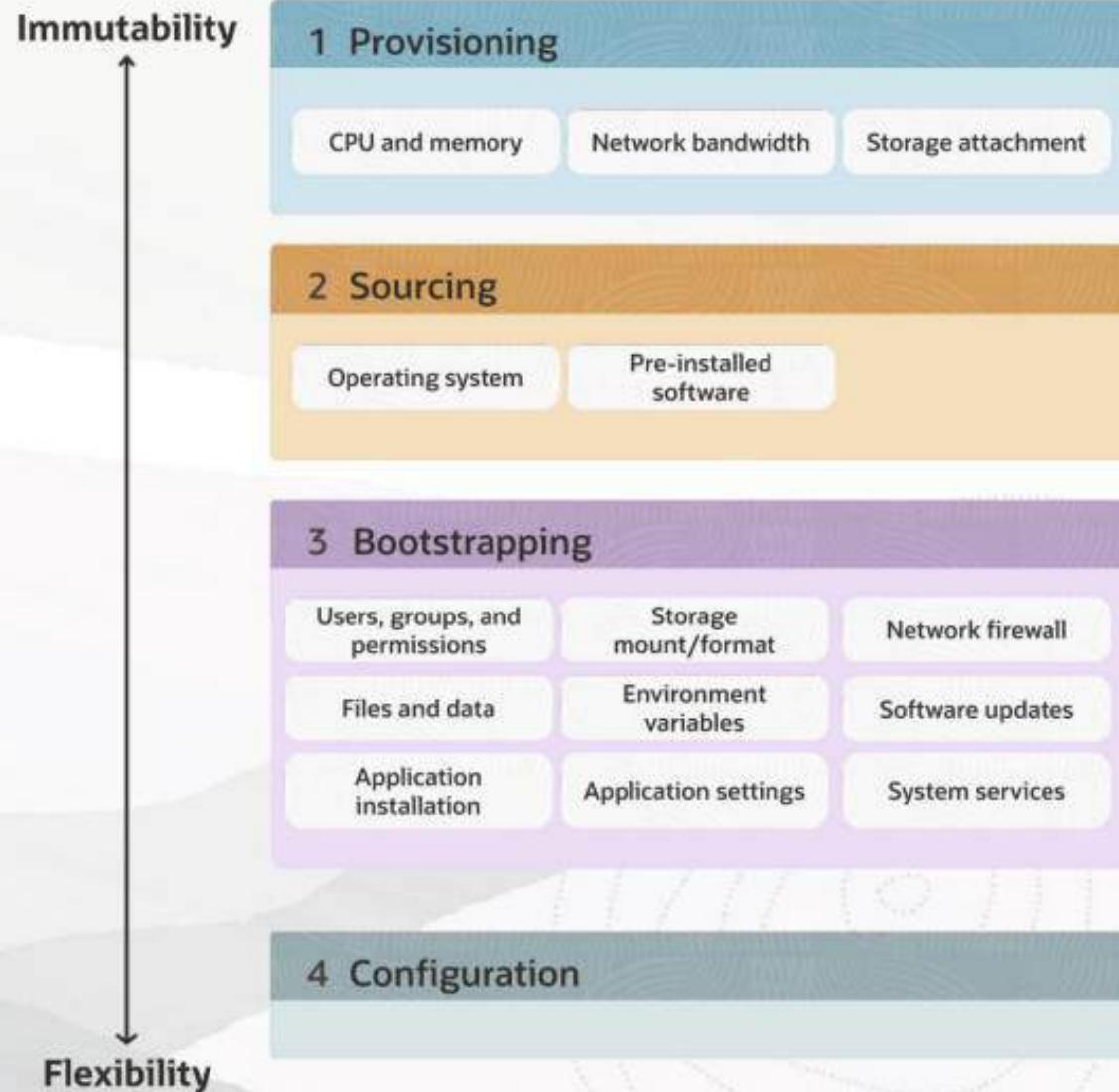
# Approach 1

Our example: Redis



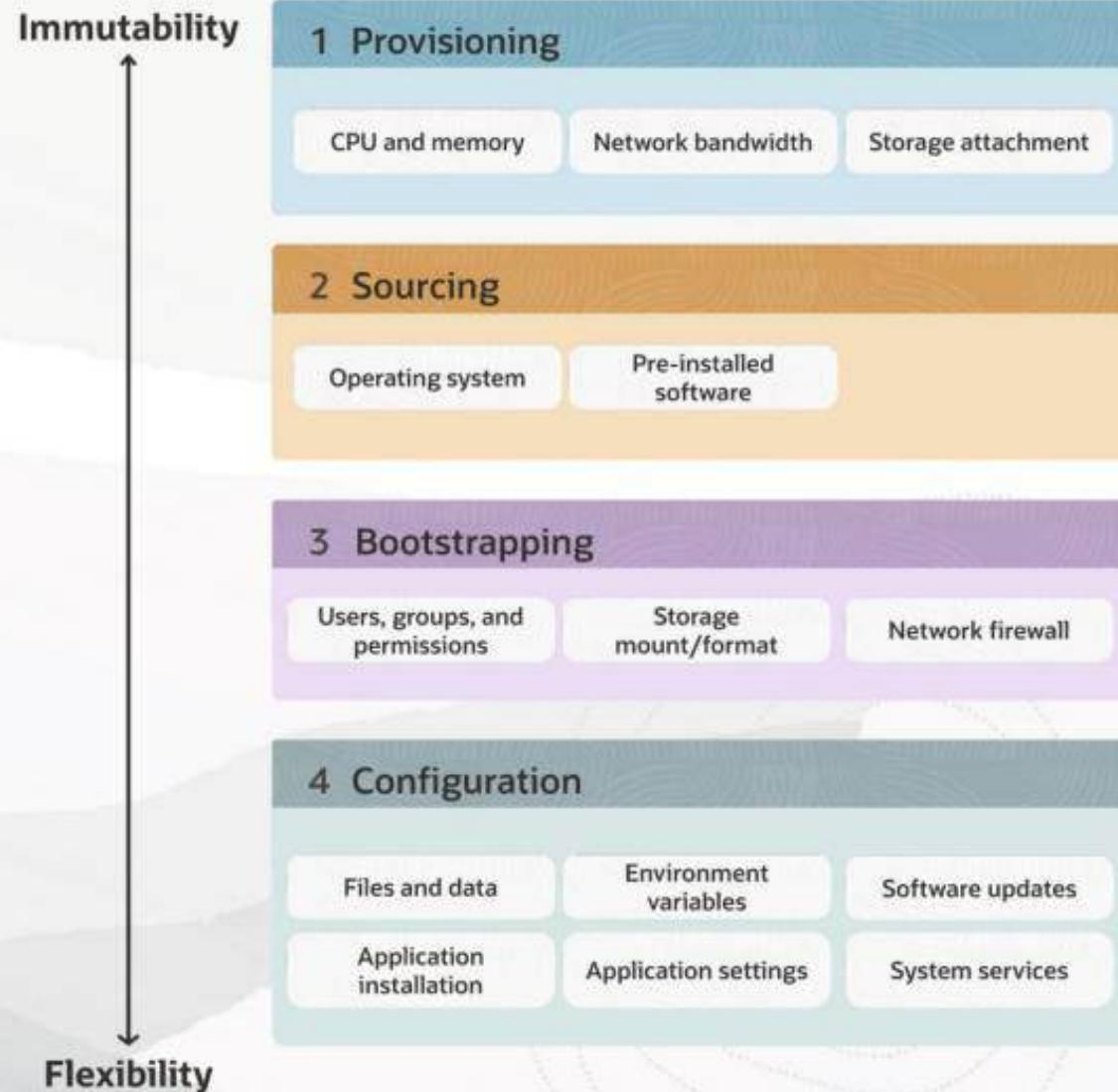
## Approach 2

Our example: PostgreSQL



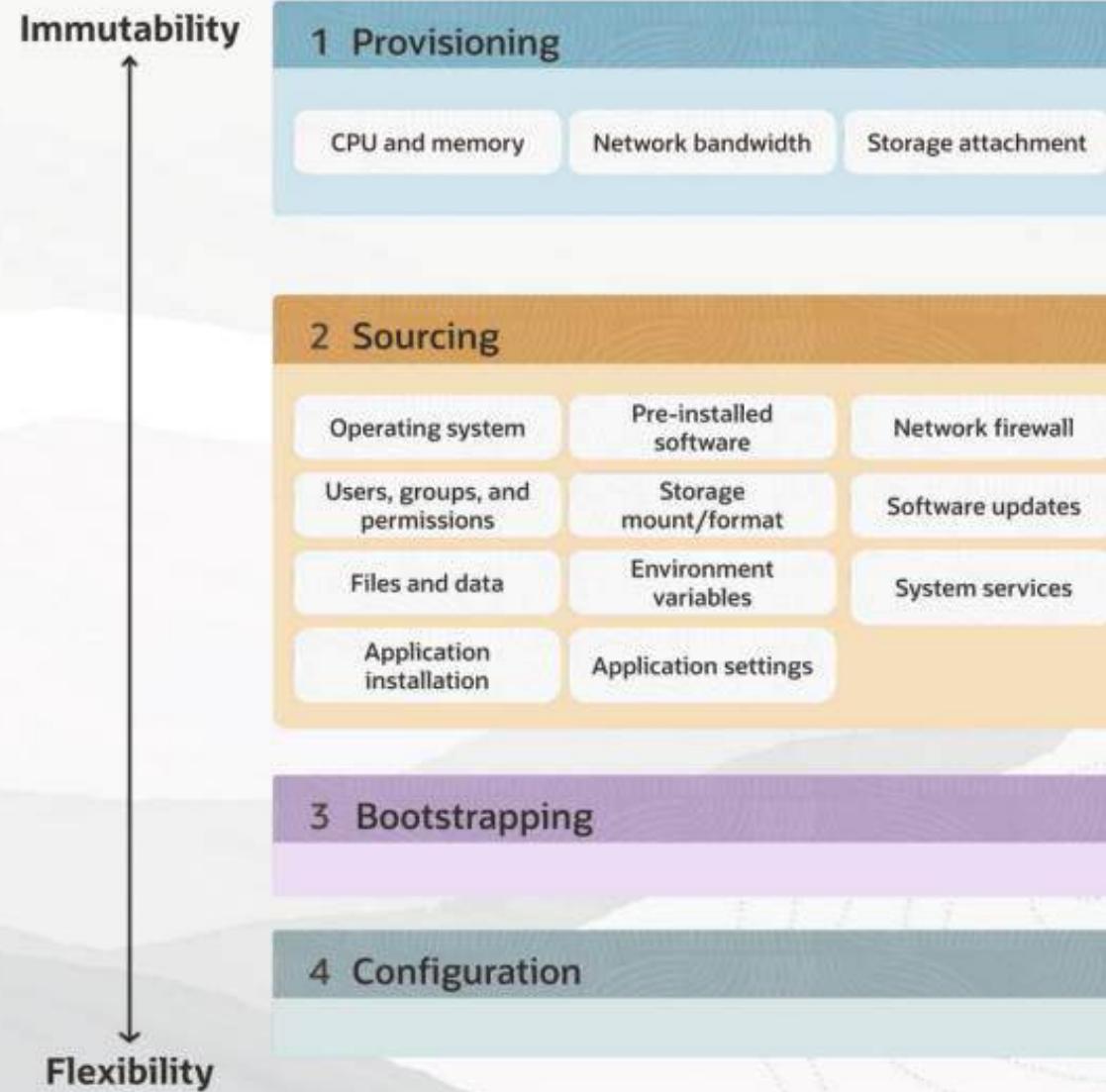
## Approach 3

Our example: Ruby on Rails

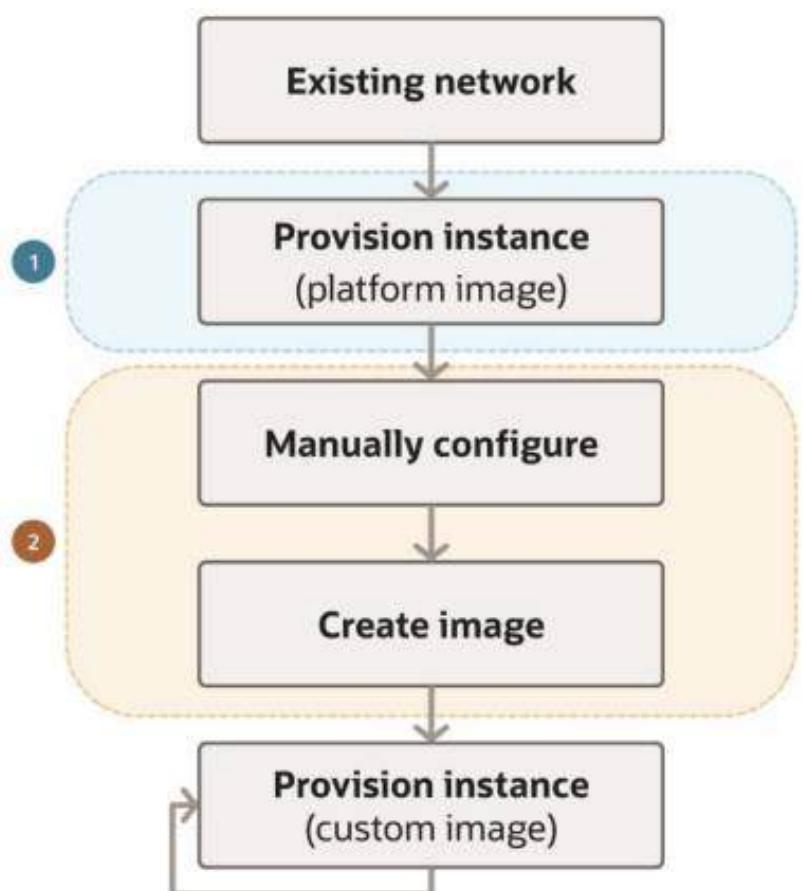


## Approach 1

Our example: Redis



## Example Workflow 1



### 1 Provisioning

CPU and memory   Network bandwidth   Storage attachment

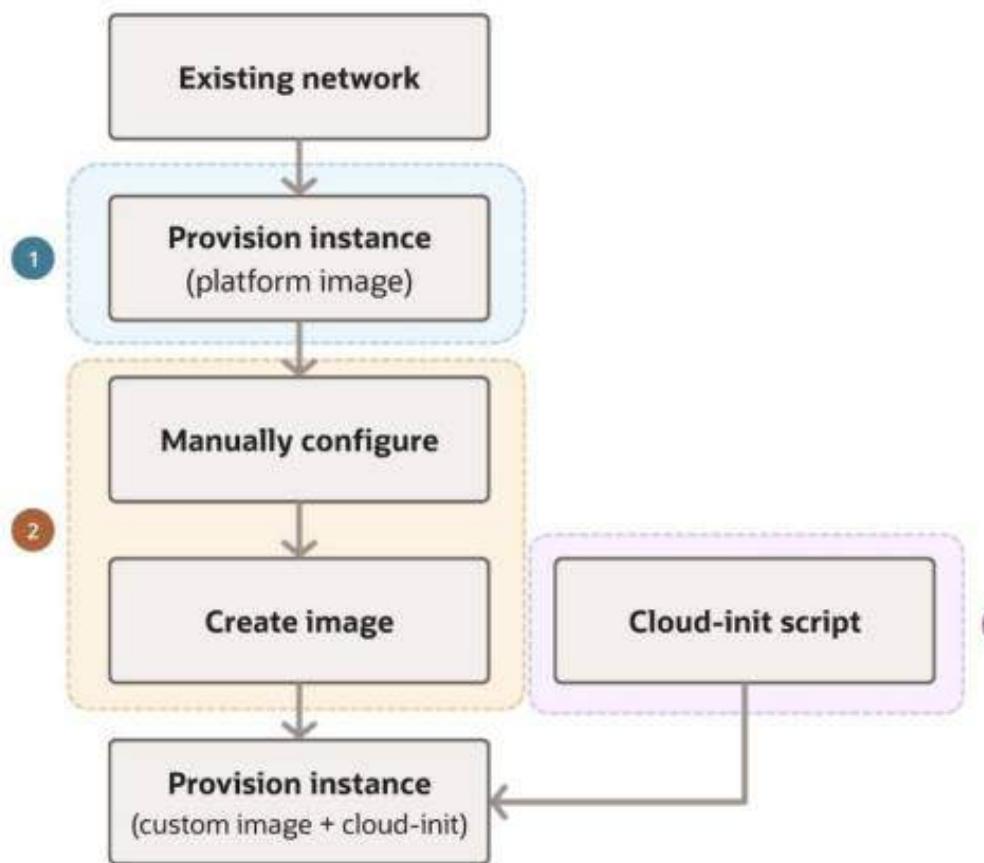
### 2 Sourcing

Operating system	Pre-installed software	Network firewall
Users, groups, and permissions	Storage mount/format	Software updates
Files and data	Environment variables	System services
Application installation	Application settings	

### 3 Bootstrapping

### 4 Configuration

## Example Workflow 2



### 1 Provisioning

CPU and memory   Network bandwidth   Storage attachment

### 2 Sourcing

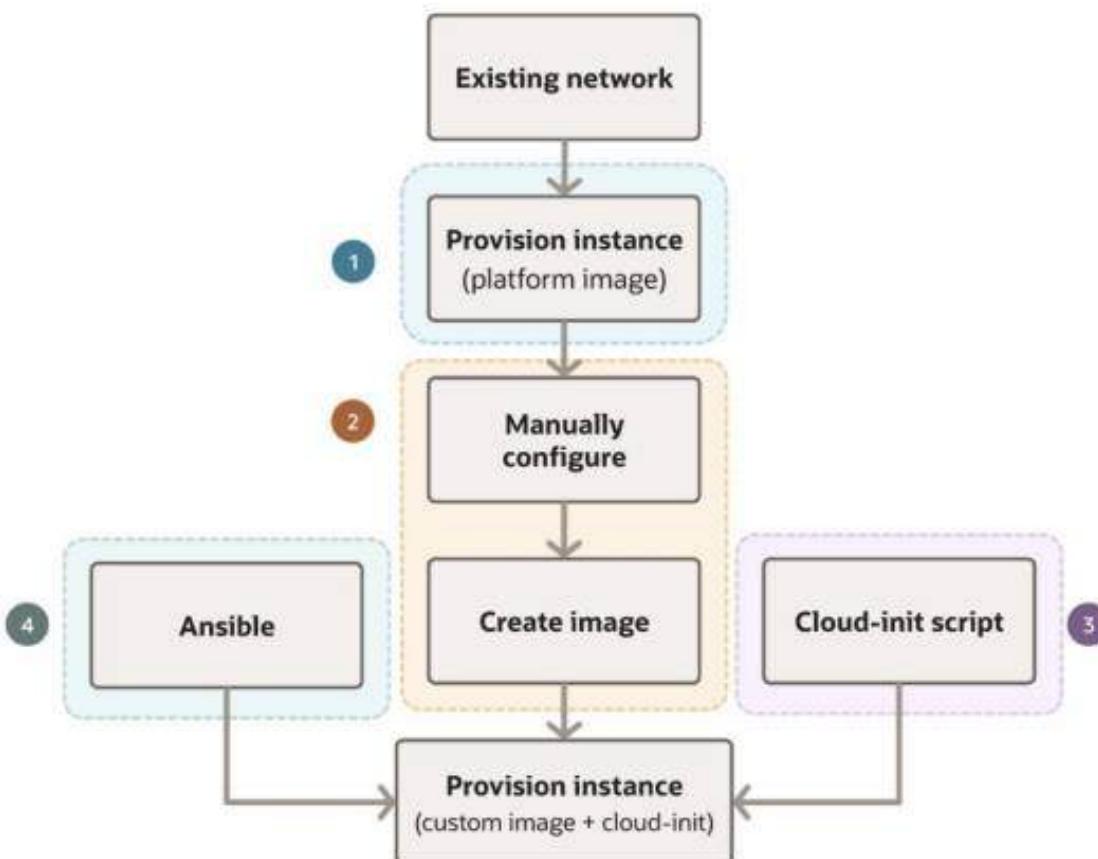
Operating system   Pre-installed software

### 3 Bootstrapping

Users, groups, and permissions	Storage mount/format	Network firewall
Files and data	Environment variables	Software updates
Application installation	Application settings	System services

### 4 Configuration

## Example Workflow 3



### 1 Provisioning

CPU and memory   Network bandwidth   Storage attachment

### 2 Sourcing

Operating system   Pre-installed software

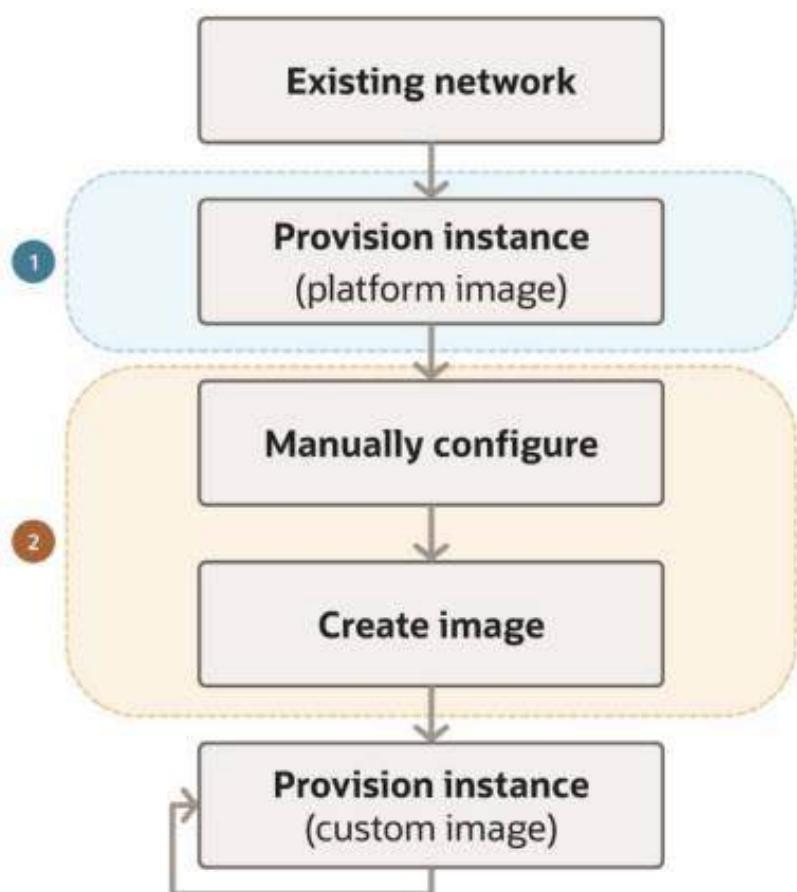
### 3 Bootstrapping

Users, groups, and permissions   Storage mount/format   Network firewall

### 4 Configuration

Files and data	Environment variables	Software updates
Application installation	Application settings	System services

## Example Workflow 1



### 1 Provisioning

CPU and memory   Network bandwidth   Storage attachment

### 2 Sourcing

Operating system	Pre-installed software	Network firewall
Users, groups, and permissions	Storage mount/format	Software updates
Files and data	Environment variables	System services
Application installation	Application settings	

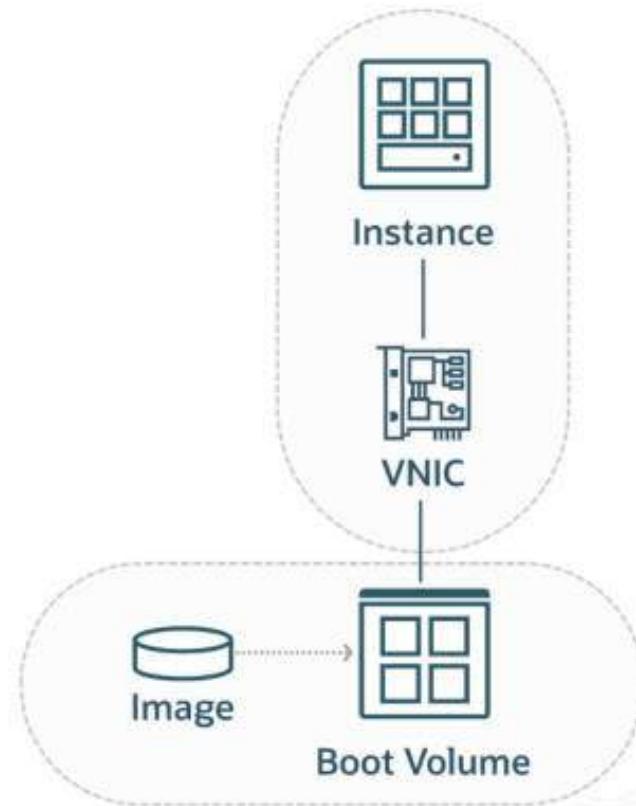
### 3 Bootstrapping

### 4 Configuration

# 1 Provisioning

## Essential Fields

Compartment	IAM location
Availability domain	Physical location
Subnet	Network location
Shape	Hardware
Source	Base operating system



# 1 Provisioning

## Essential Fields

Compartment

Availability domain

VNIC Subnet

Shape

Source

## CLI Example

```
oci compute instance launch
  --compartment-id ocid1.compartment.oc1..example1
  --availability-domain Uocm:PHX-AD-1
  --subnet-id ocid1.subnet.oc1.phx.example2
  --shape VM.Standard.A1.Flex
  --shape-config '{"ocpus": 4, "memoryInGBs": 16}'
  --source-details
    '{"sourceType": "image", "imageId": "ocid1.image.phx.example3"}'
```

# 1 Provisioning

## Essential Fields

Compartment

Availability domain

VNIC Subnet

Shape

Source

## Terraform Example

```
resource "oci_core_instance" "example_instance" {
    compartment_id      = "ocid1.compartment.oc1..example1"
    availability_domain = "Uocm:PHX-AD-1"
    create_vnic_details {
        subnet_id = "ocid1.subnet.oc1.phx.example2"
    }
    shape = "VM.Standard.A1.Flex"
    shape_config {
        ocpus       = 4
        memory_in_gbs = 16
    }
    source_details {
        source_type = "image"
        source_id   = "ocid1.image.oc1.phx.example3"
    }
}
```

# 1 Provisioning

## Optional Fields

Display name

Hostname

NSGs

Metadata

More...

## CLI Example

```
oci compute instance launch
--compartment-id ocid1.compartment.oc1..example1
--availability-domain Uocm:PHX-AD-1
--subnet-id ocid1.subnet.oc1.phx.example2
--shape VM.Standard.A1.Flex
--shape-config '{"ocpus": 4, "memoryInGBs": 16}'
--source-details
'{"sourceType": "image", "imageId": "ocid1.image.phx.example3"}'
--display-name "Friendly Name"
--hostname-label "ex-instance-1"
--nsg-ids
'["ocid1.networksecuritygroup.oc1.phx.example4"]'
--metadata
'{"authorized_keys": "ssh-rsa abcde", "user_data": "#!cloud-config"}'
```

# 1 Provisioning

## Optional Fields

Display name

Hostname

NSGs

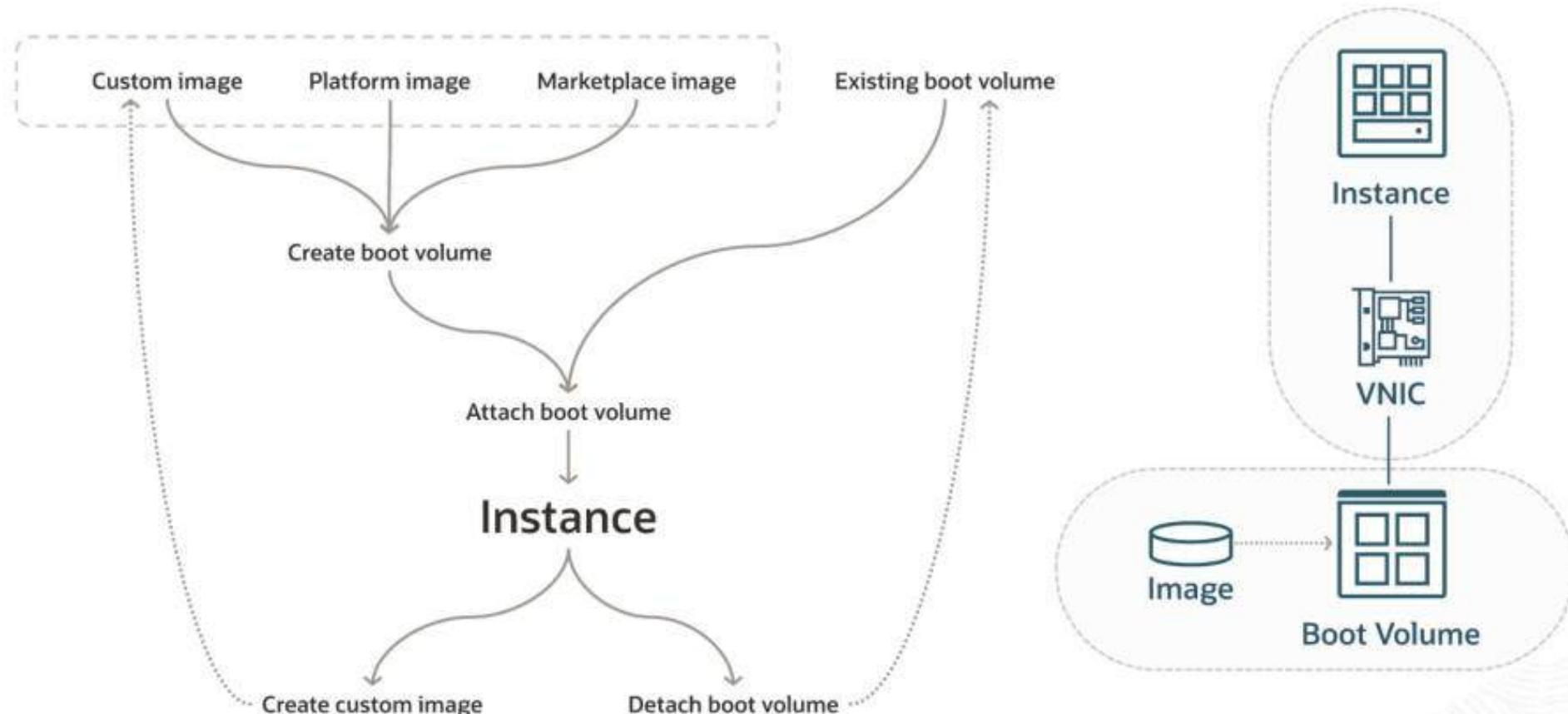
Metadata

More...

## Terraform Example

```
resource "oci_core_instance" "example_instance" {
    compartment_id      = "ocid1.compartment.oc1..example1"
    availability_domain = "Uocm:PHX-AD-1"
    create_vnic_details {
        subnet_id = "ocid1.subnet.oc1.phx.example2"
        hostname_label = "ex-instance-1"
        nsg_ids = [
            "ocid1.networksecuritygroup.oc1.phx.example4"
        ]
    }
    shape = "VM.Standard.A1.Flex"
    shape_config {
        ocpus      = 4
        memory_in_gbs = 16
    }
    source_details {
        source_type = "image"
        source_id   = "ocid1.image.oc1.phx.example3"
    }
    metadata = {
        ssh_authorized_keys = "ssh-rsa abcde"
        user_data = ""
    }
    display_name = "Friendly Name"
}
```

## 2 Source



## 2 Source

### OCI CLI: Image Source

```
oci compute instance launch
  --compartment-id ocid1.compartment.oc1..example1
  --availability-domain Uocm:PHX-AD-1
  --subnet-id ocid1.subnet.oc1.phx.example2
  --shape VM.Standard.A1.Flex
  --shape-config '{"ocpus": 4, "memoryInGBs": 16}'
  --source-details
    '{"sourceType": "image", "imageId": "ocid1.image.phx.example3"}'
```

### OCI CLI: Boot Volume Source

```
oci compute instance launch
  --compartment-id ocid1.compartment.oc1..example1
  --availability-domain Uocm:PHX-AD-1
  --subnet-id ocid1.subnet.oc1.phx.example2
  --shape VM.Standard.A1.Flex
  --shape-config '{"ocpus": 4, "memoryInGBs": 16}'
  --source-details
    '{"sourceType": "bootVolume", "bootVolumeId": "ocid1.bootvolume.phx.example3"}'
```

## 2 Source

### Terraform: Image Source

```
resource "oci_core_instance" "example_instance" {
  compartment_id      = "ocid1.compartment.oc1..example1"
  availability_domain = "Uocm:PHX-AD-1"
  create_vnic_details {
    subnet_id = "ocid1.subnet.oc1.phx.example2"
  }
  shape = "VM.Standard.A1.Flex"
  shape_config {
    ocpus        = 4
    memory_in_gbs = 16
  }
  source_details {
    source_type = "image"
    source_id   = "ocid1.image.oc1.phx.example3"
  }
}
```

### Terraform: Boot Volume Source

```
resource "oci_core_instance" "example_instance" {
  compartment_id      = "ocid1.compartment.oc1..example1"
  availability_domain = "Uocm:PHX-AD-1"
  create_vnic_details {
    subnet_id = "ocid1.subnet.oc1.phx.example2"
  }
  shape = "VM.Standard.A1.Flex"
  shape_config {
    ocpus        = 4
    memory_in_gbs = 16
  }
  source_details {
    source_type = "bootVolume"
    source_id   = "ocid1.bootvolume.oc1.phx.example5"
  }
}
```



# Compute Deep Dive: Bootstrapping with Cloud-init

---

**OCI Cloud Operations**

Recap...

## Instance Life Cycle

### 1 Provisioning

Allocate shape and network

### 2 Sourcing

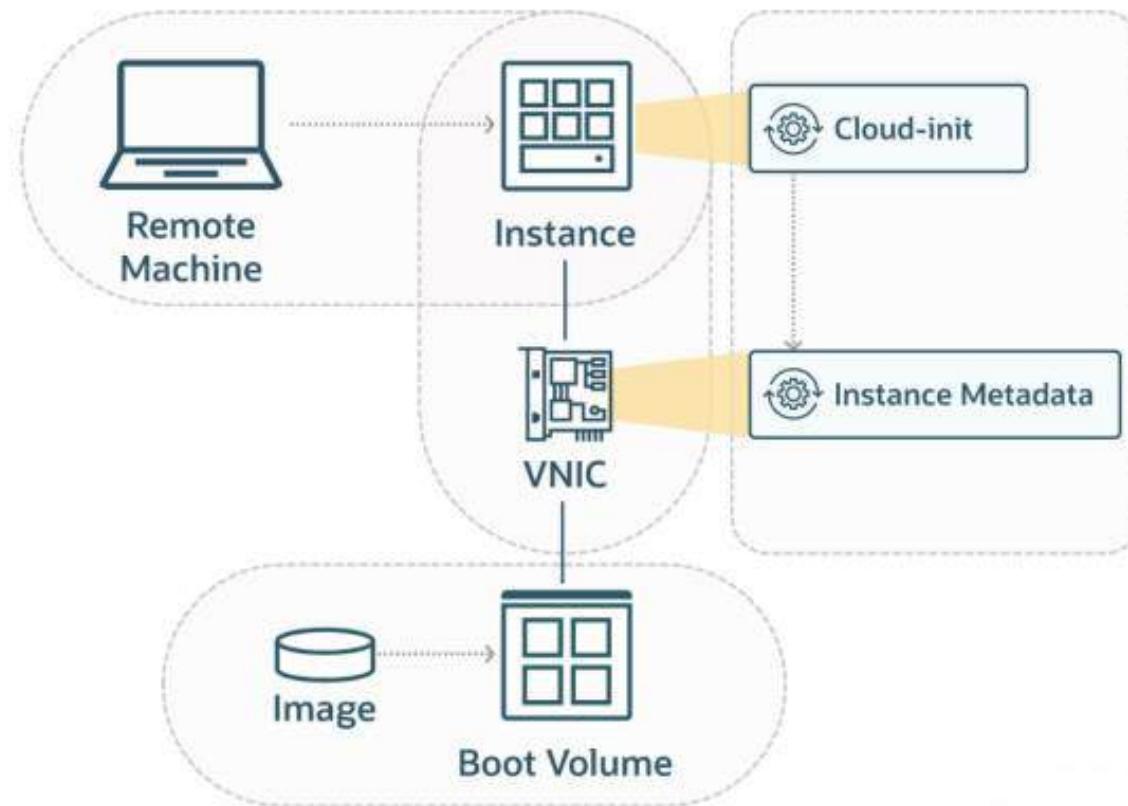
Instantiate from an image or connect existing boot volume

### 3 Bootstrapping

Run cloud-init with instance metadata

### 4 Fine-tuning

Remotely fine-tune manually or via configuration management tools



Recap...

## Instance Life Cycle

### 1 Provisioning

Allocate shape and network

### 2 Sourcing

Instantiate from an image or connect existing boot volume

### 3 Bootstrapping

Run cloud-init with instance metadata

### 4 Fine-tuning

Remotely fine-tune manually or via configuration management tools

### Infrastructure

CPU and memory

Network bandwidth

Storage attachment

### Base System

Operating system

Pre-installed software

### Access Control

Network firewall

Users, groups, and permissions

### System Maintenance

Storage mount/format

Software updates

### Data Management

Files and data

Environment variables

### Application Deployment

System services

Application installation

Application settings

## 1 Provisioning

### Infrastructure

CPU and memory

Network bandwidth

Storage attachment

## 2 Sourcing

### Base System

Operating system

Pre-installed software

## 3 Bootstrapping

### Access Control

Network firewall

Users, groups, and permissions

## 4 Fine-tuning

### System Maintenance

Storage mount/format

Software updates

### Data Management

Files and data

Environment variables

### Application Deployment

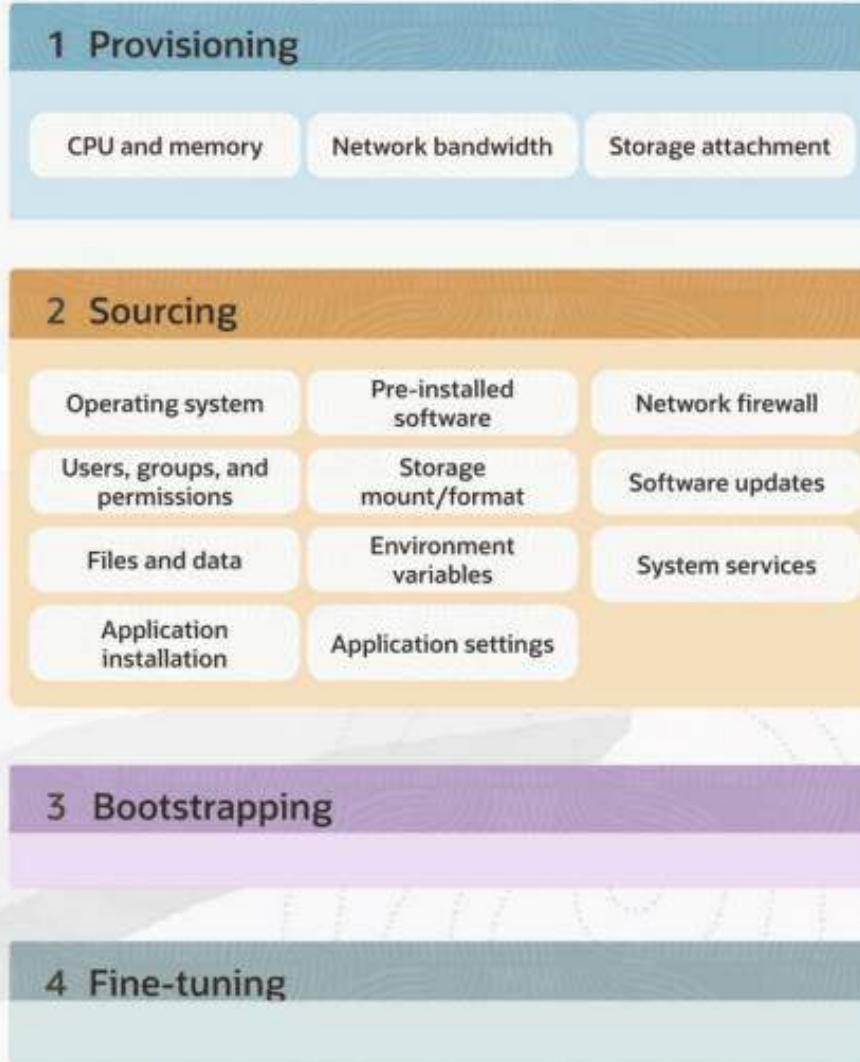
System services

Application installation

Application settings

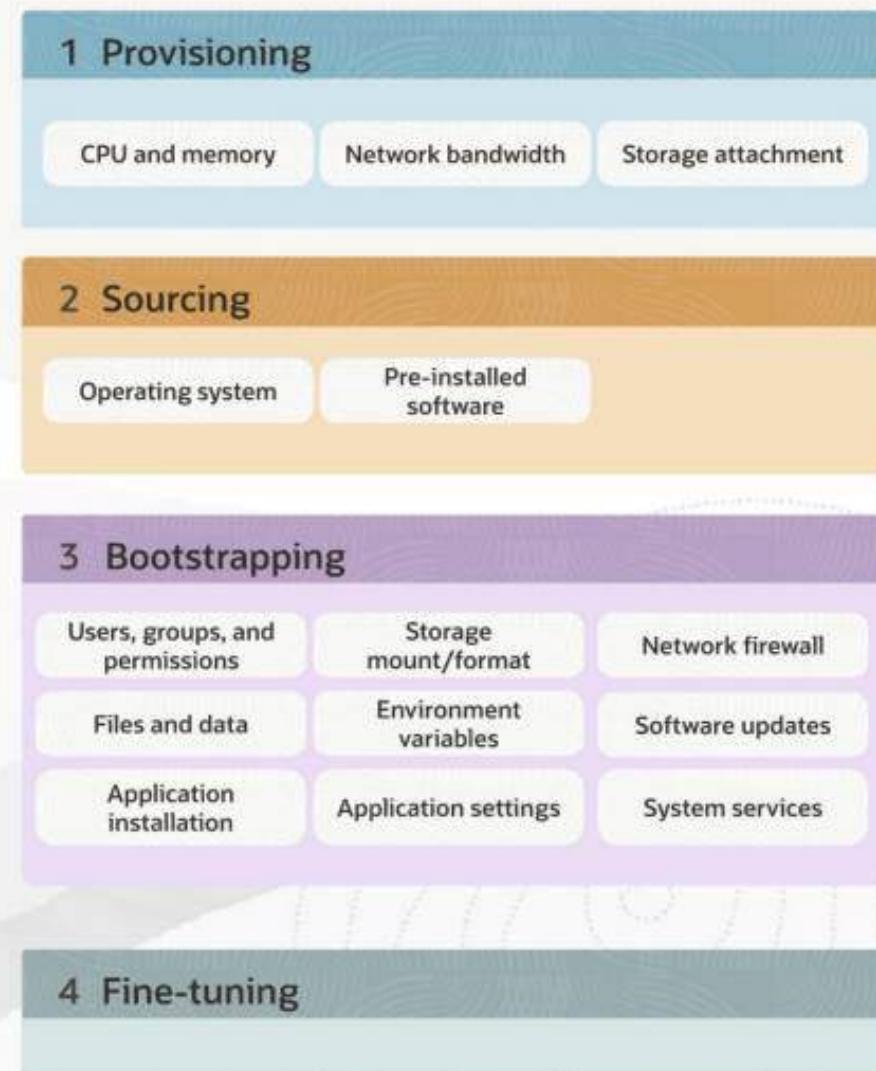
# Approach 1

Our example: PostgreSQL



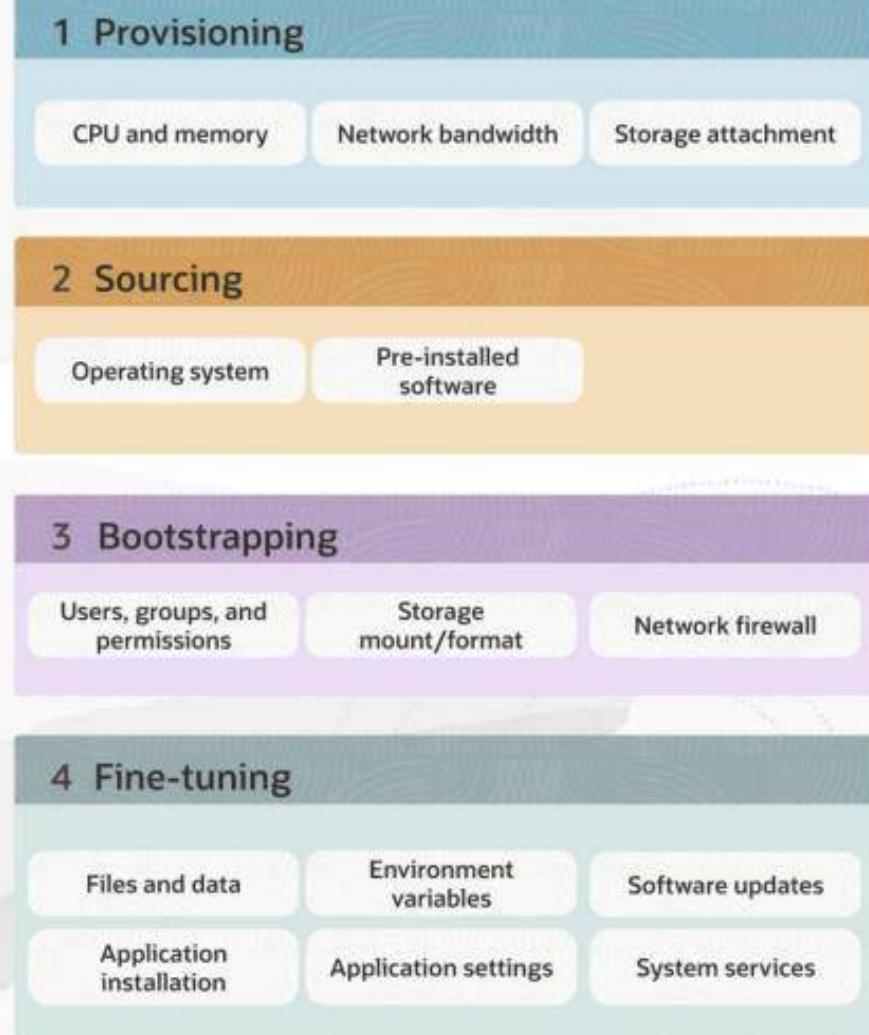
## Approach 2

Our example: Redis

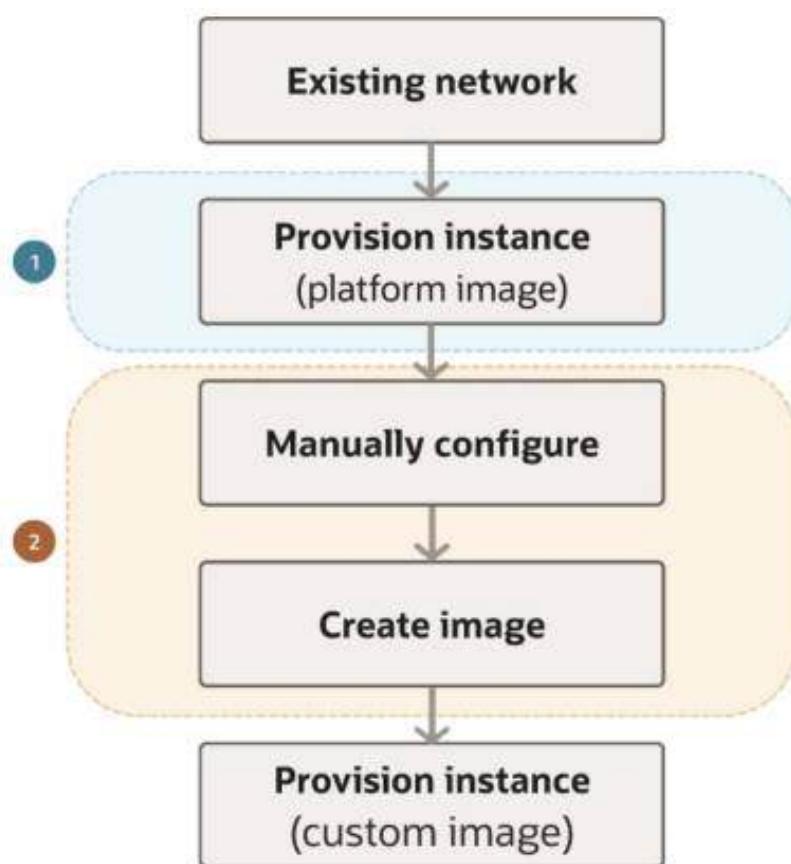


## Approach 3

Our example: Ruby on Rails



## Example Workflow 1



### 1 Provisioning

CPU and memory   Network bandwidth   Storage attachment

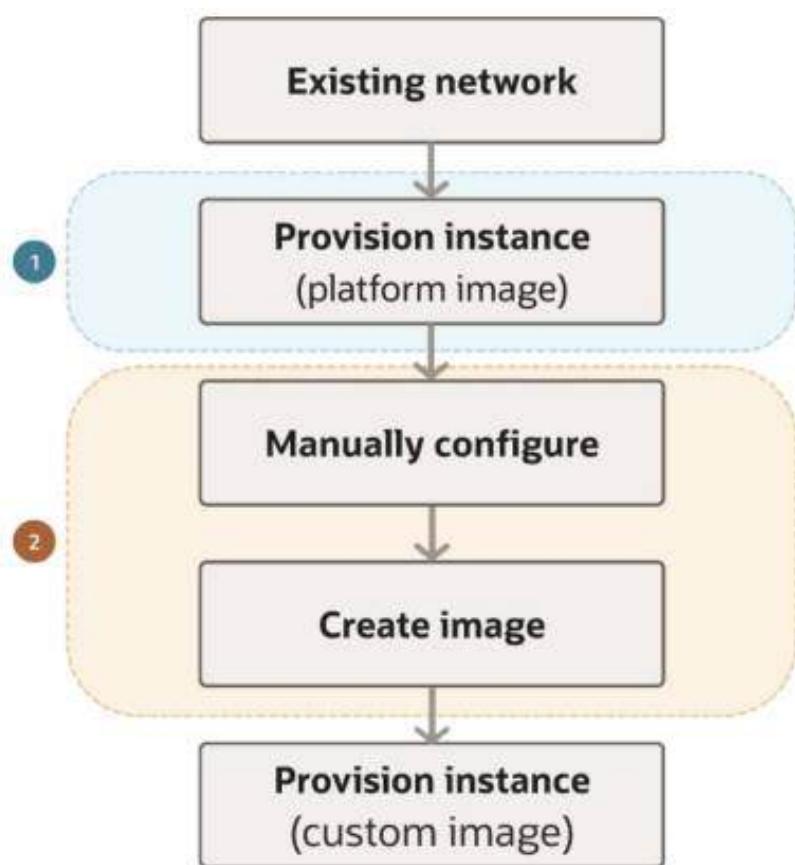
### 2 Sourcing

Operating system	Pre-installed software	Network firewall
Users, groups, and permissions	Storage mount/format	Software updates
Files and data	Environment variables	System services
Application installation	Application settings	

### 3 Bootstrapping

### 4 Fine-tuning

## Example Workflow 1



### 1 Provisioning

CPU and memory   Network bandwidth   Storage attachment

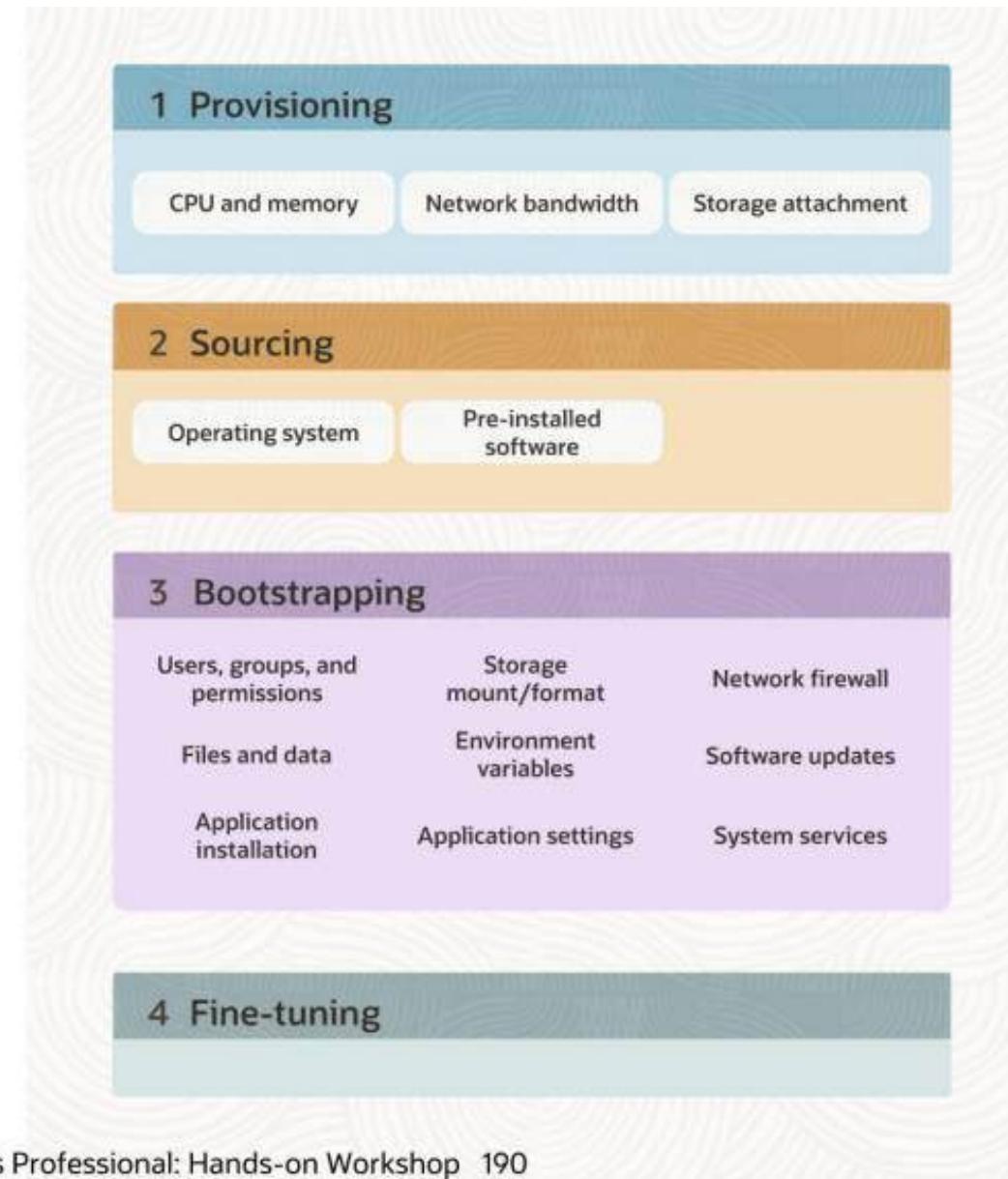
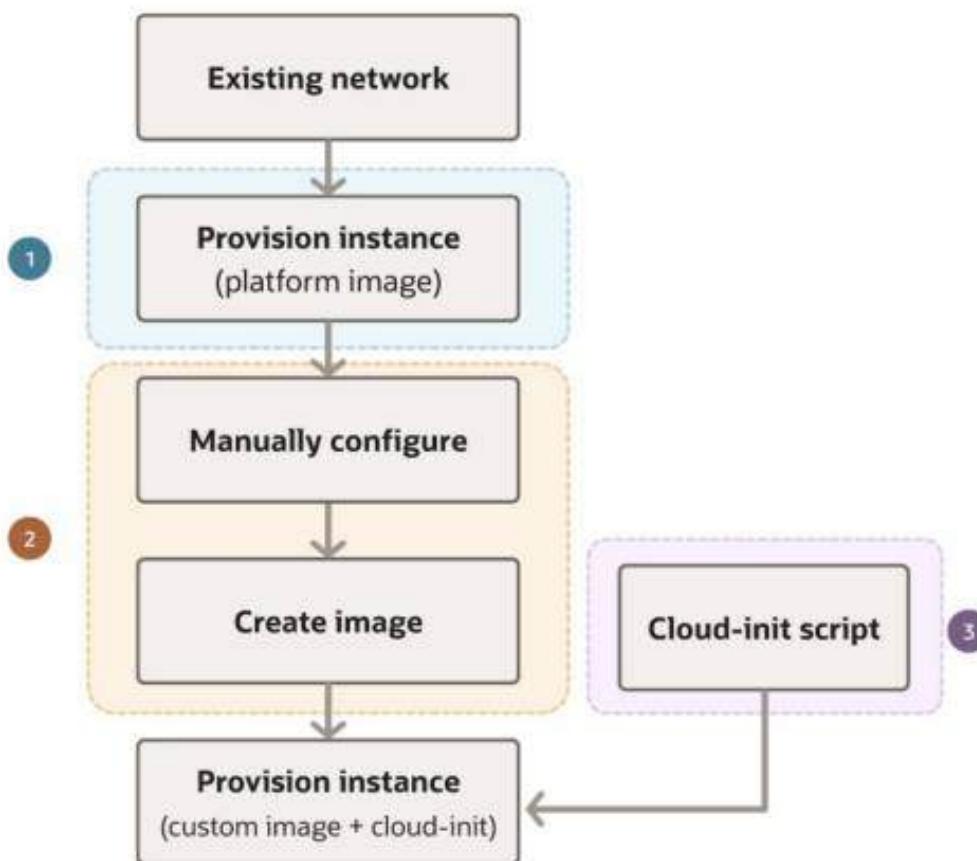
### 2 Sourcing

Operating system	Pre-installed software	Network firewall
Users, groups, and permissions	Storage mount/format	Software updates
Files and data	Environment variables	System services
Application installation	Application settings	

### 3 Bootstrapping

### 4 Fine-tuning

## Example Workflow 2



# Bootstrapping

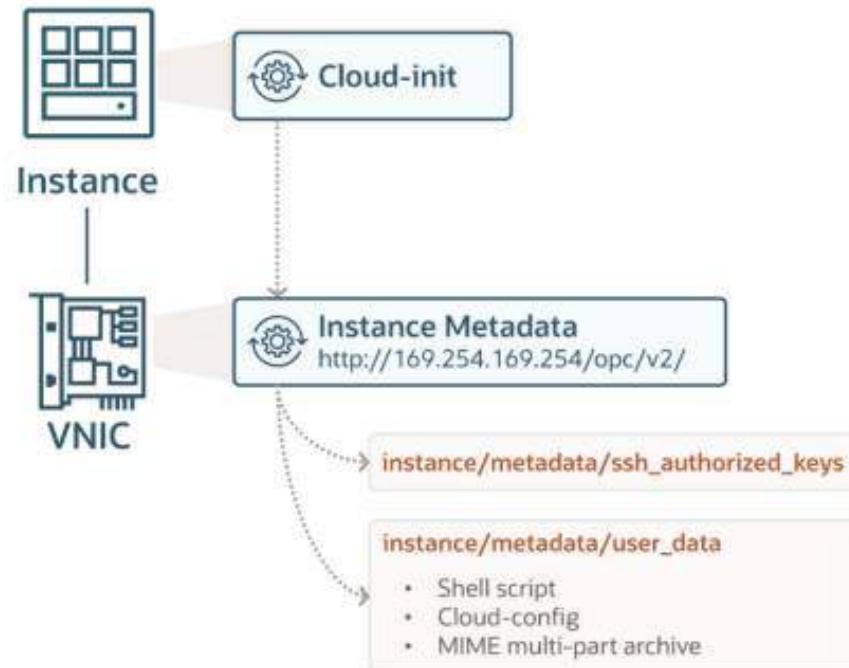
## Components

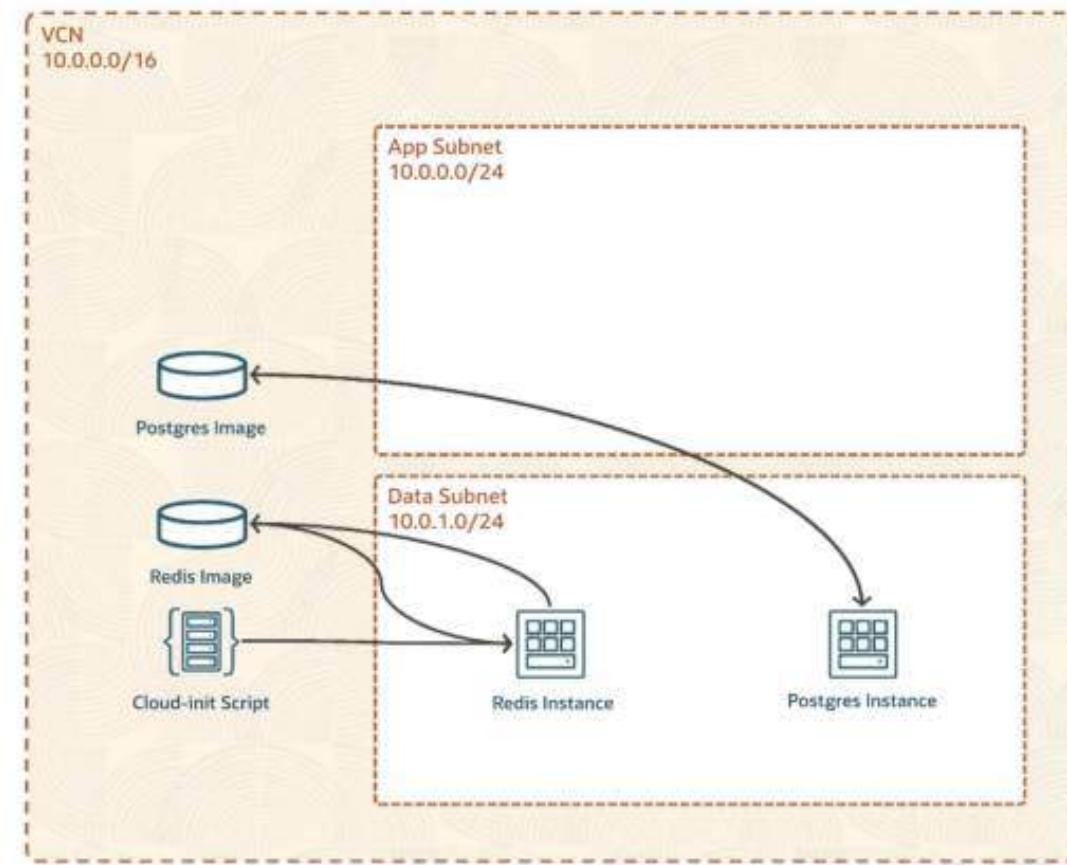
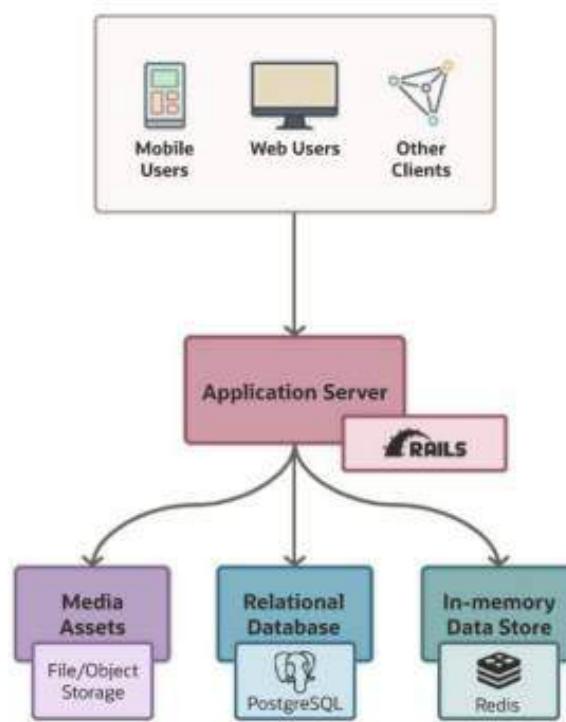
### Instance Metadata

- Arbitrary key/value pairs
- Pre-defined keys
  - ssh\_authorized\_keys
  - user\_data

### Cloud-Init

- Initialization agent
  - Adds SSH keys
  - Executes user data







# Compute Deep Dive: Fine-tuning with Ansible

---

**OCI Cloud Operations**

Recap...

## Instance Life Cycle

### 1 Provisioning

Allocate shape and network

### 2 Sourcing

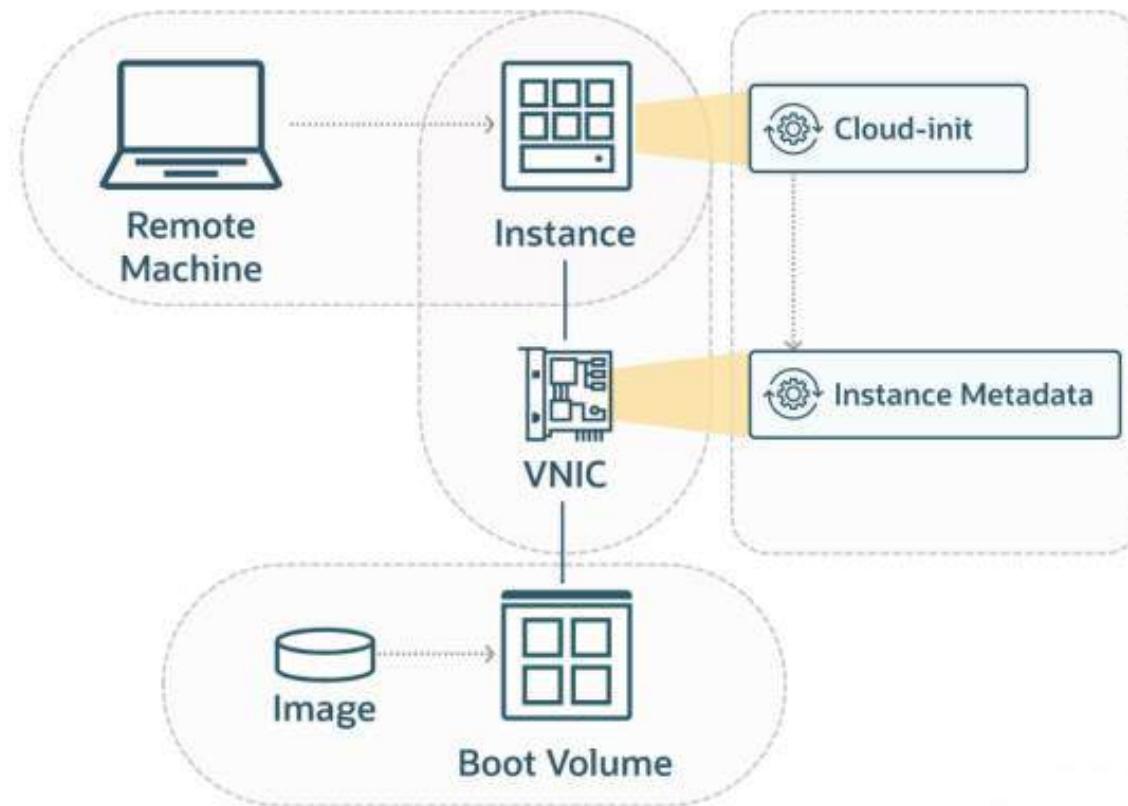
Instantiate from an image or connect existing boot volume

### 3 Bootstrapping

Run cloud-init with instance metadata

### 4 Fine-tuning

Remotely fine-tune manually or via configuration management tools



Recap...

## Instance Life Cycle

### 1 Provisioning

Allocate shape and network

### 2 Sourcing

Instantiate from an image or connect existing boot volume

### 3 Bootstrapping

Run cloud-init with instance metadata

### 4 Fine-tuning

Remotely fine-tune manually or via configuration management tools

### Infrastructure

CPU and memory

Network bandwidth

Storage attachment

### Base System

Operating system

Pre-installed software

### Access Control

Network firewall

Users, groups, and permissions

### System Maintenance

Storage mount/format

Software updates

### Data Management

Files and data

Environment variables

### Application Deployment

System services

Application installation

Application settings

## 1 Provisioning

### Infrastructure

CPU and memory

Network bandwidth

Storage attachment

## 2 Sourcing

### Base System

Operating system

Pre-installed software

## 3 Bootstrapping

### Access Control

Network firewall

Users, groups, and permissions

## 4 Fine-tuning

### System Maintenance

Storage mount/format

Software updates

### Data Management

Files and data

Environment variables

### Application Deployment

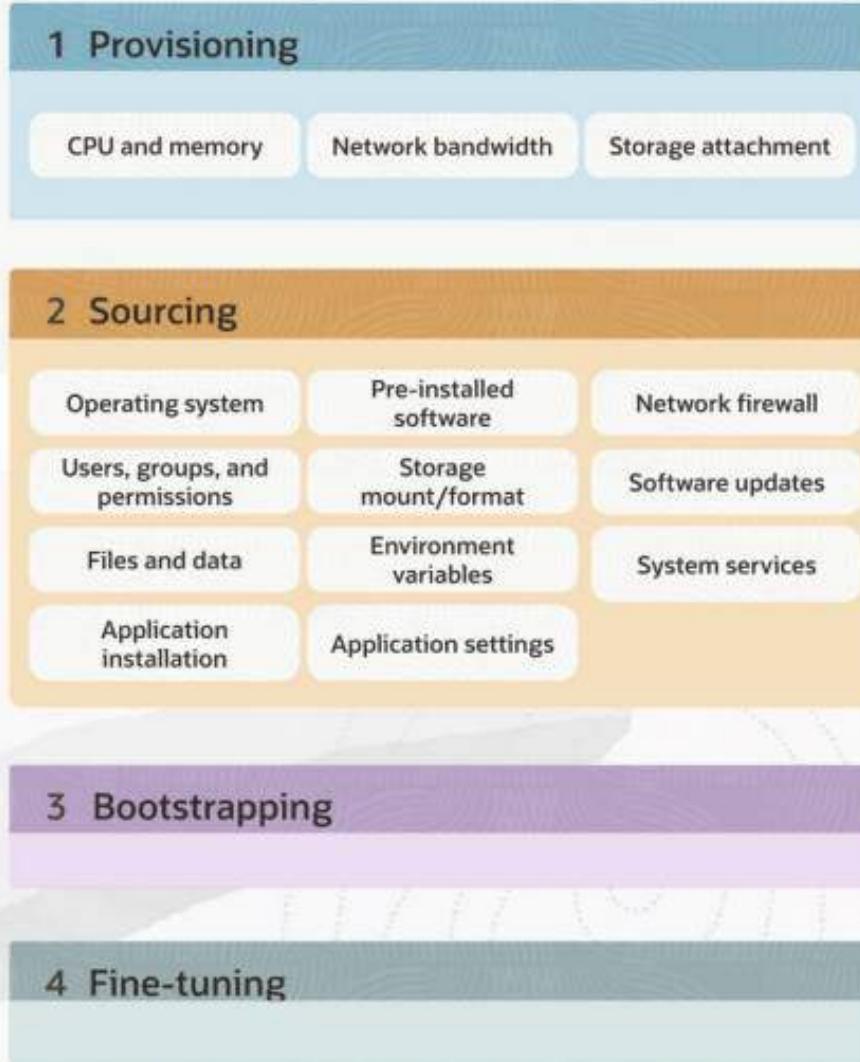
System services

Application installation

Application settings

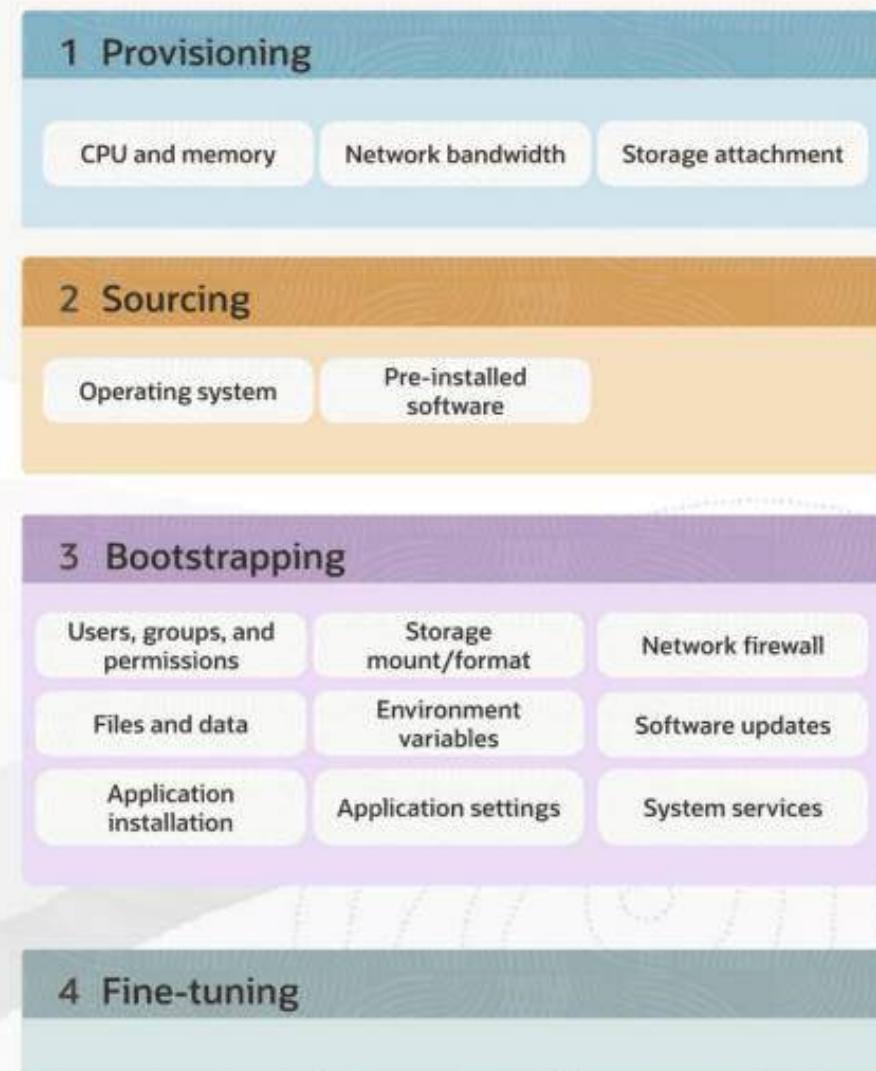
# Approach 1

Our example: PostgreSQL



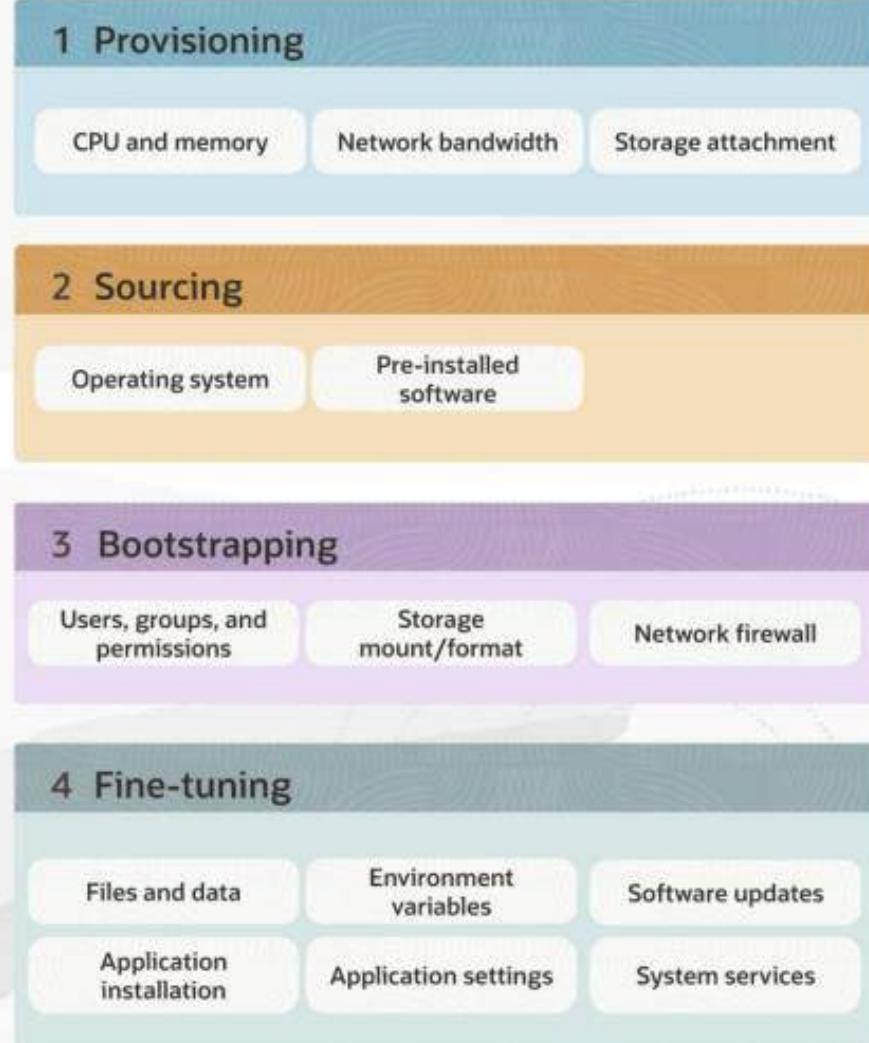
## Approach 2

Our example: Redis

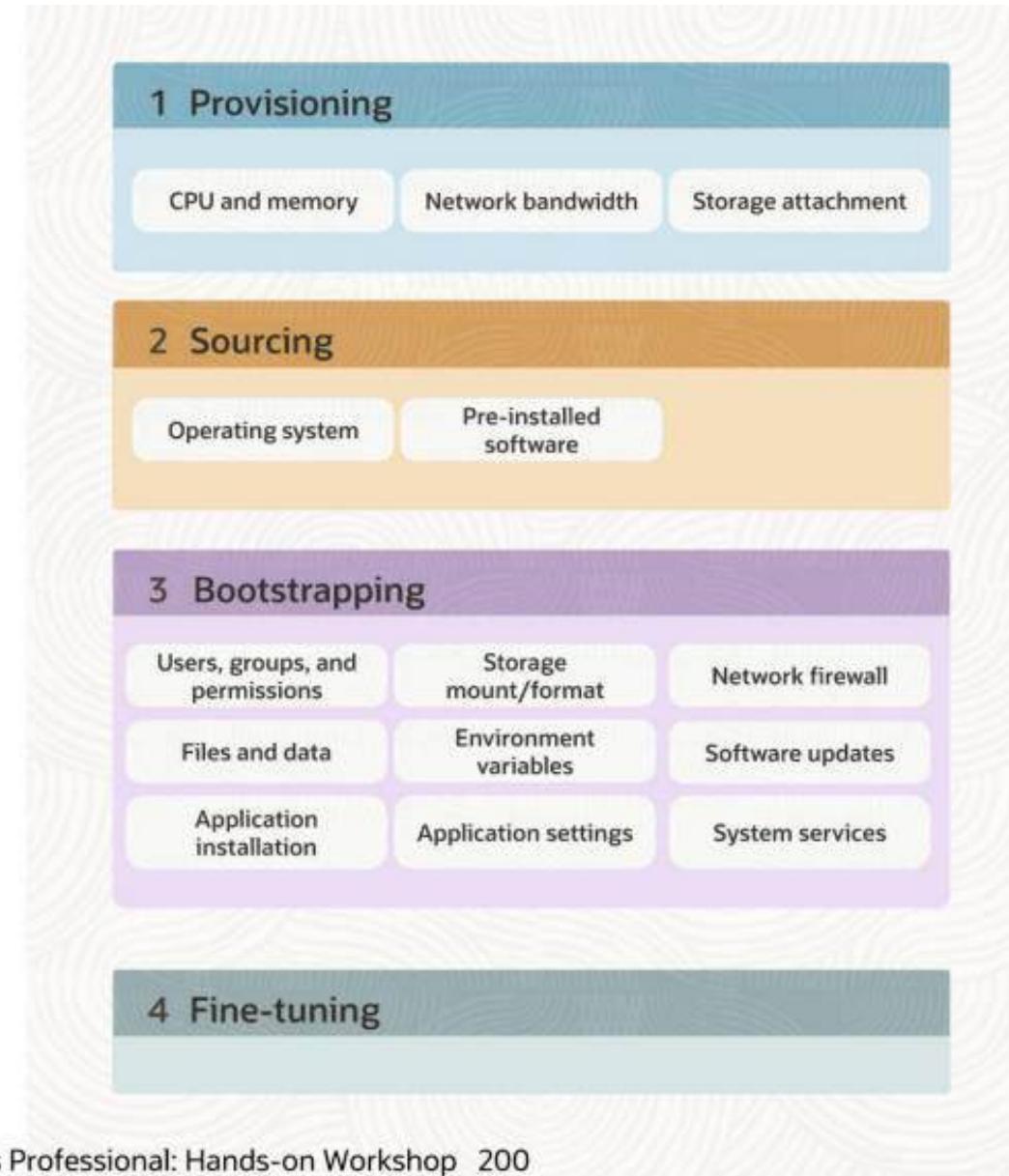
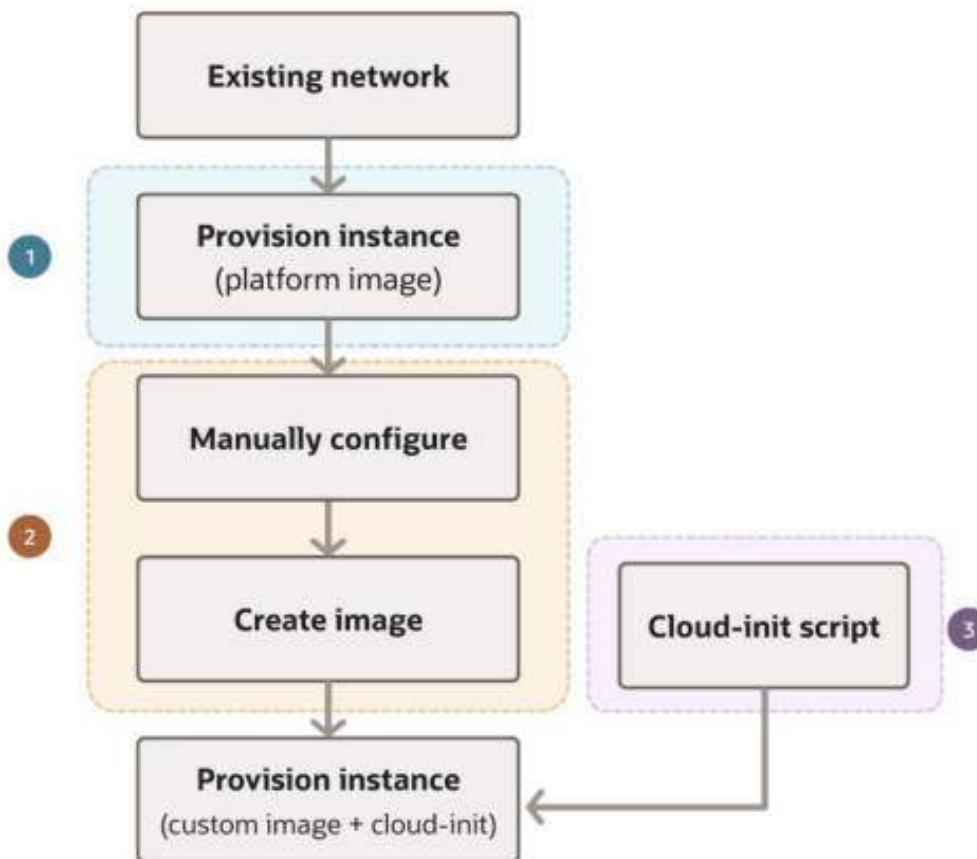


## Approach 3

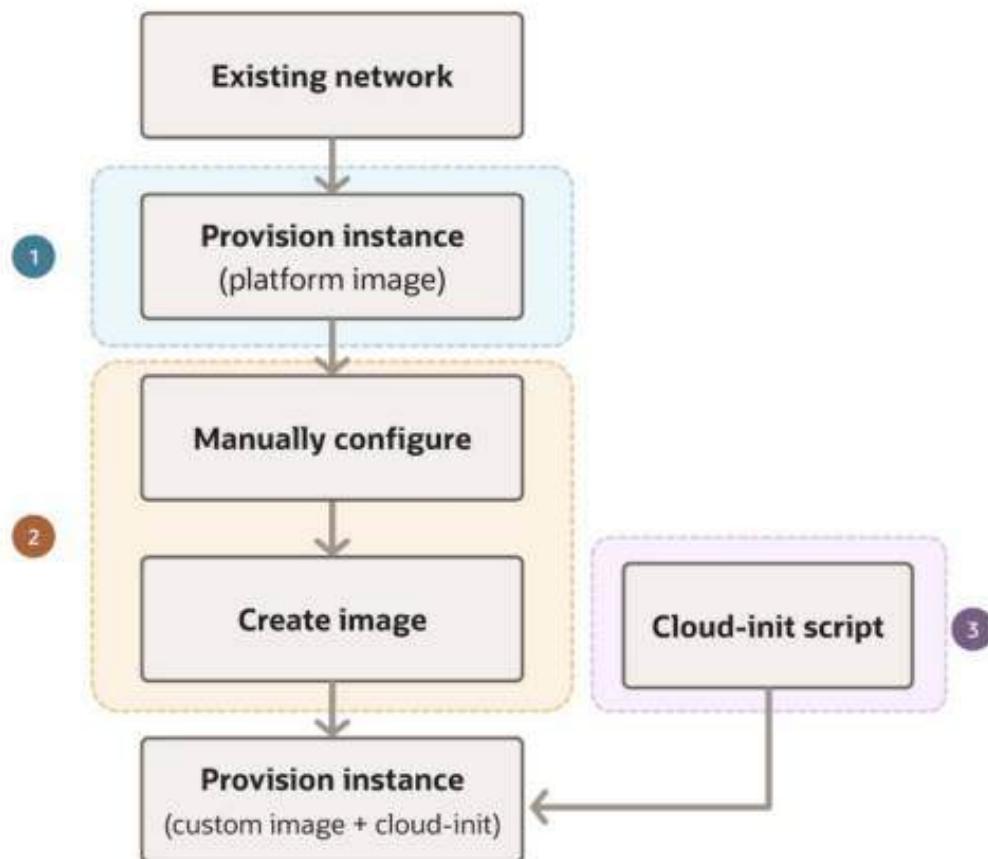
Our example: Ruby on Rails



## Example Workflow 2



## Example Workflow 2



### 1 Provisioning

CPU and memory

Network bandwidth

Storage attachment

### 2 Sourcing

Operating system

Pre-installed software

### 3 Bootstrapping

Users, groups, and permissions

Storage mount/format

Network firewall

Files and data

Environment variables

Software updates

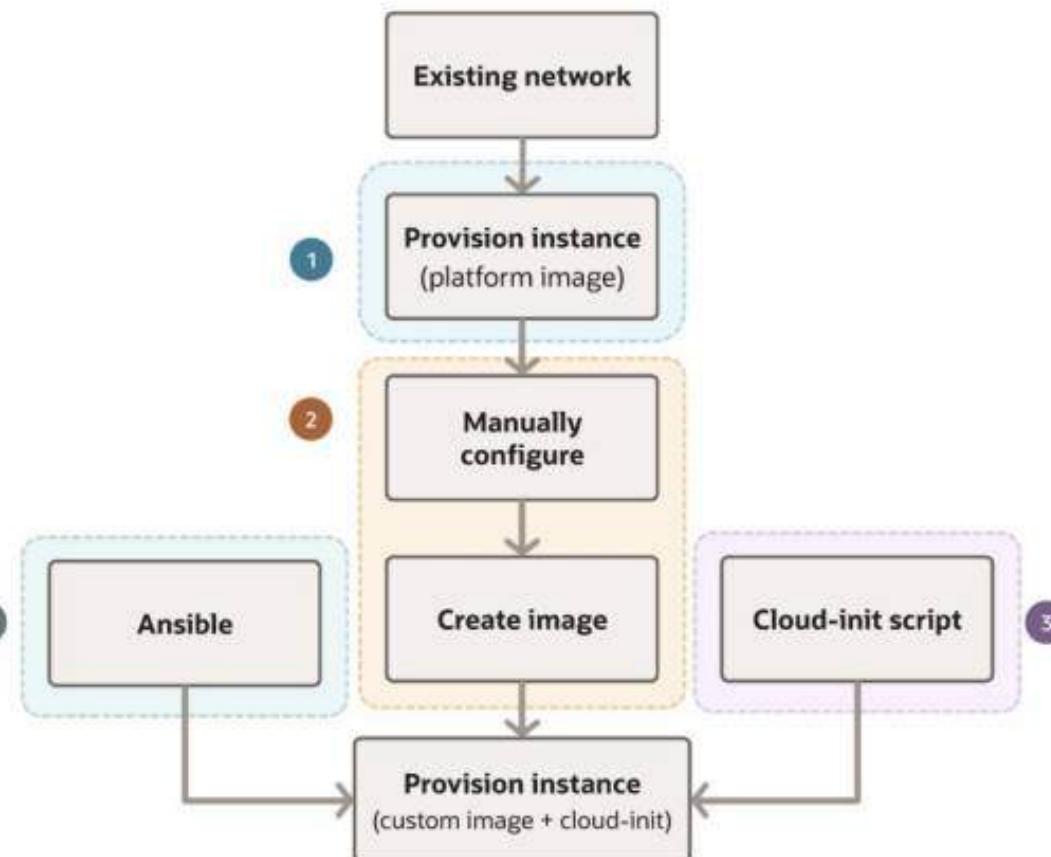
Application installation

Application settings

System services

### 4 Fine-tuning

## Example Workflow 3



### 1 Provisioning

CPU and memory   Network bandwidth   Storage attachment

### 2 Sourcing

Operating system   Pre-installed software

### 3 Bootstrapping

Users, groups, and permissions   Storage mount/format   Network firewall

### 4 Fine-tuning

Files and data   Environment variables   Software updates  
Application installation   Application settings   System services

## Fine-tuning



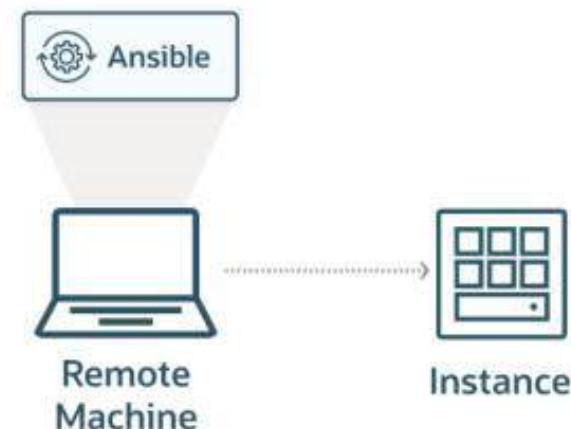
## Fine-tuning (Configuration Management)

### Method 1: Manual

- Connect remotely (SSH)
- Manually run commands

### Method 2: Ansible

- Connect remotely (SSH)
- Run ad hoc commands
- Run declarative modules



## Fine-tuning (Configuration Management)

### Method 1: Manual

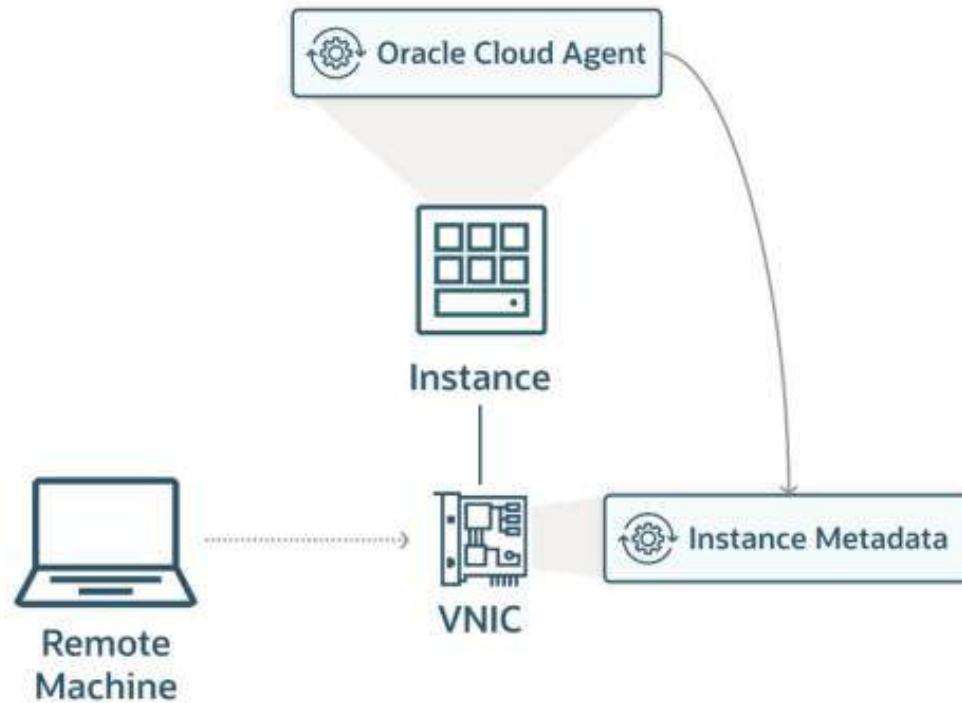
- Connect remotely (SSH)
- Manually run commands

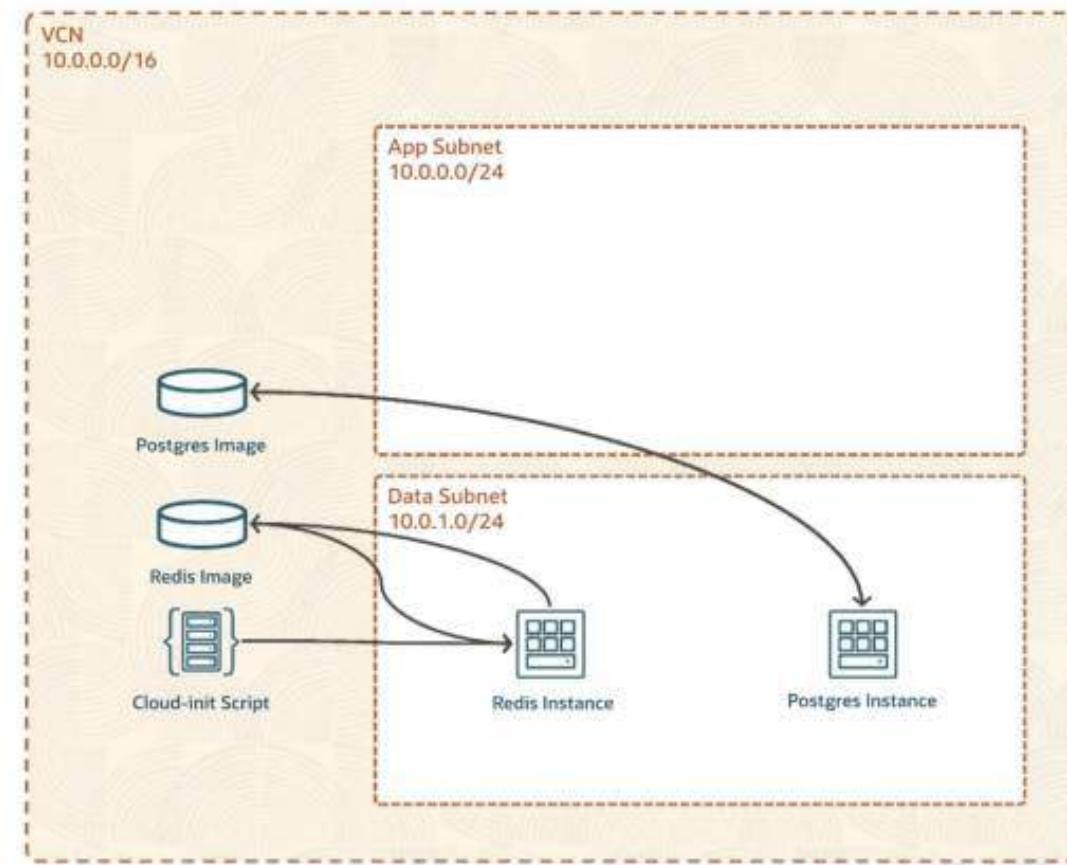
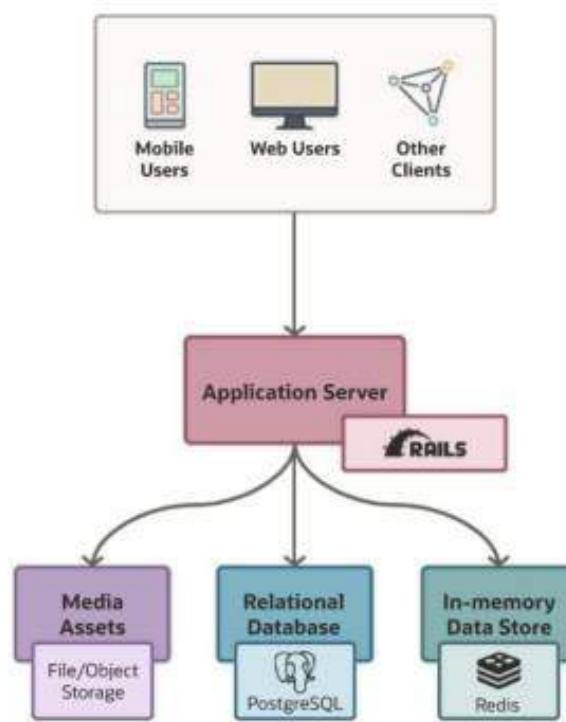
### Method 2: Ansible

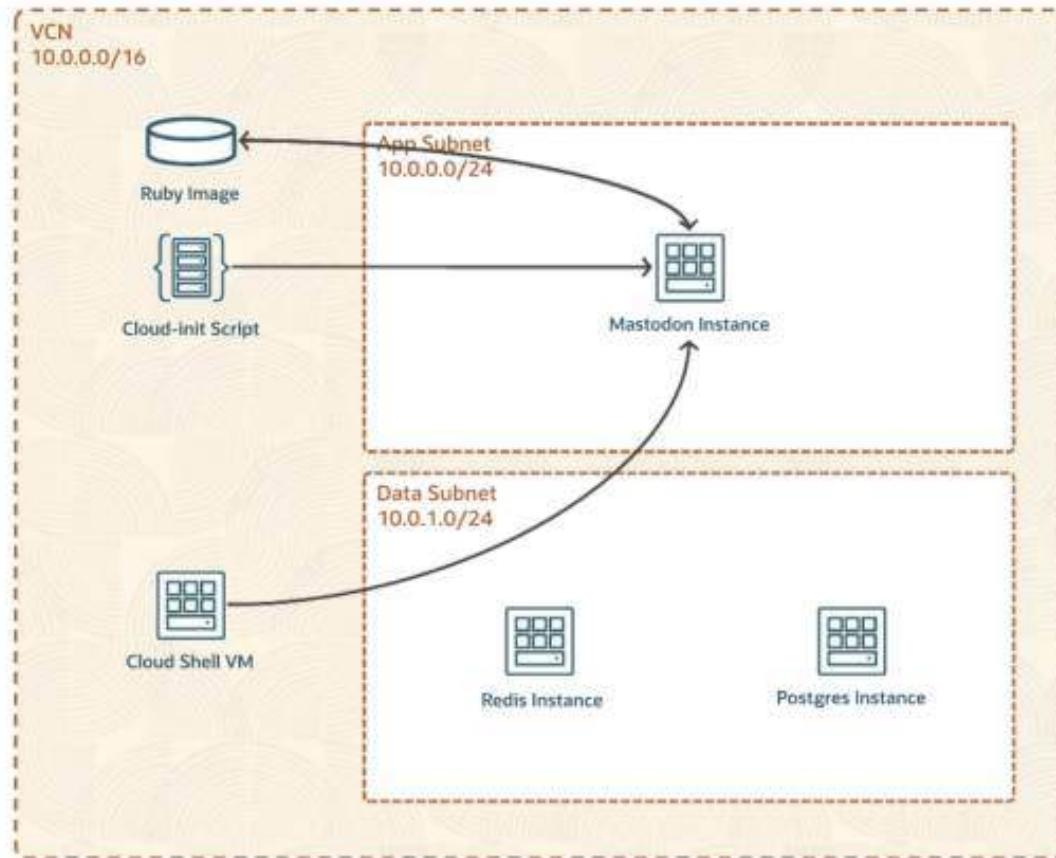
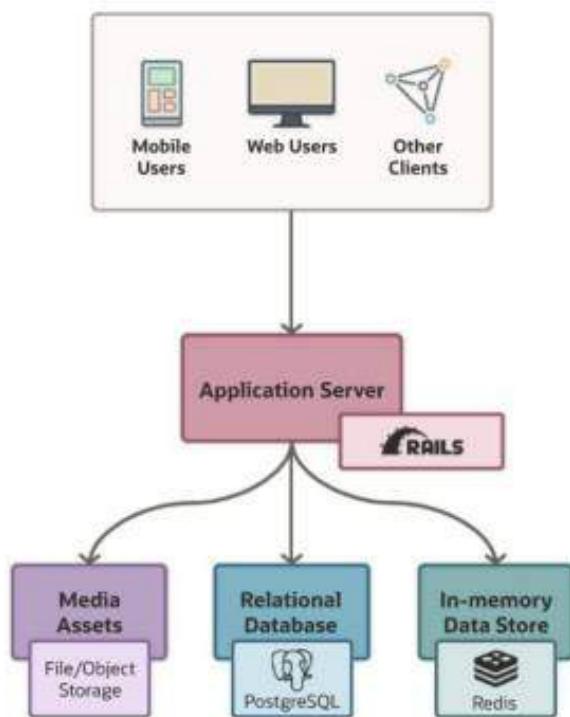
- Connect remotely (SSH)
- Run ad hoc commands
- Run declarative modules

### Method 3: Oracle Cloud Agent

- No connection needed
- Run ad hoc commands
- Update packages
- Scan for vulnerabilities
- OCI-specific configuration



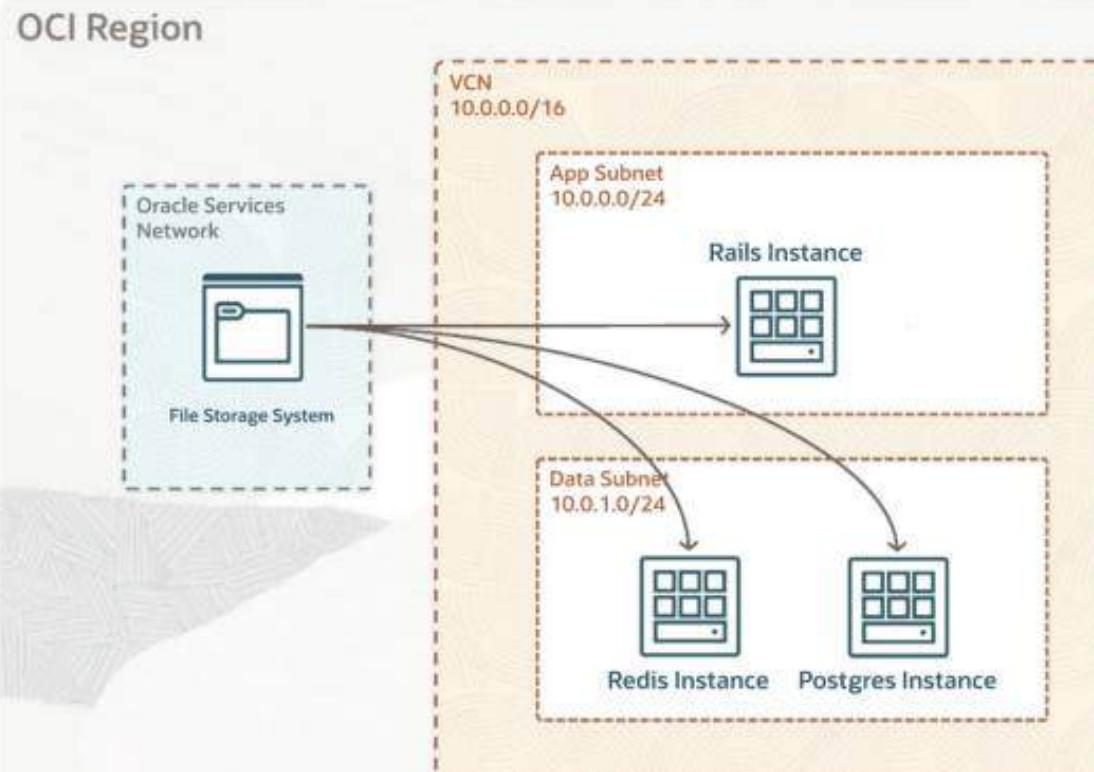


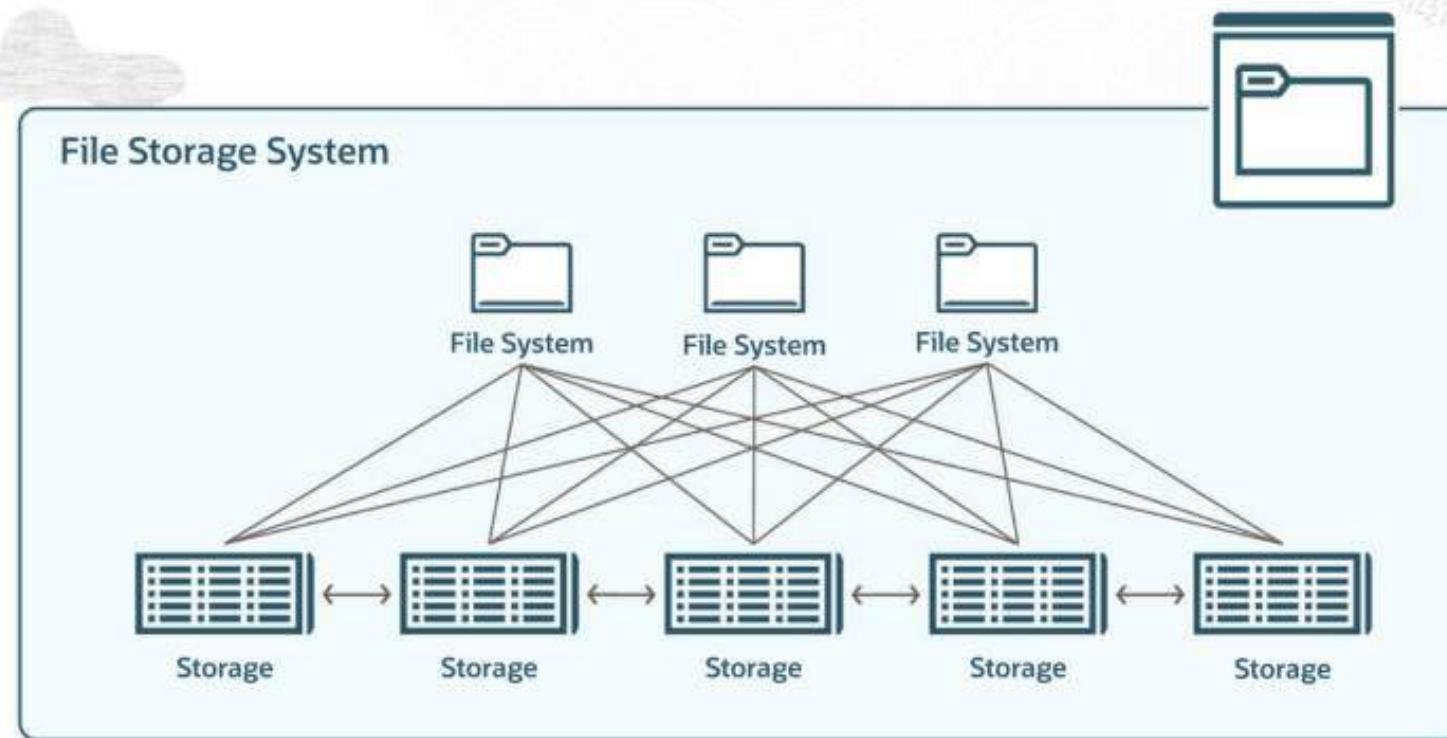


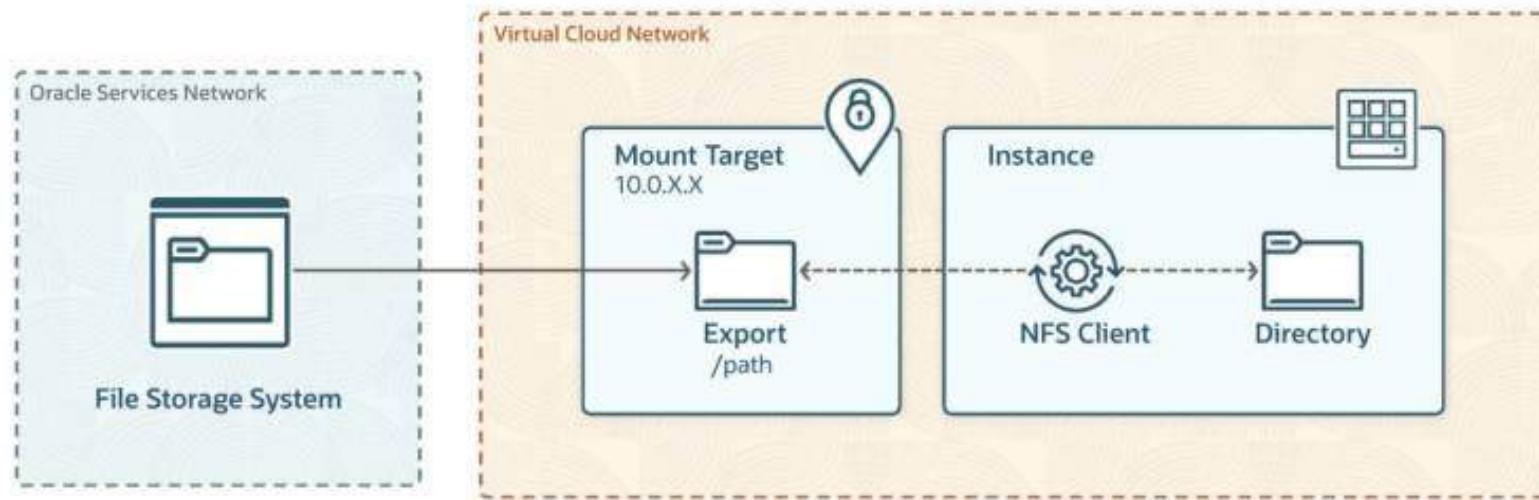


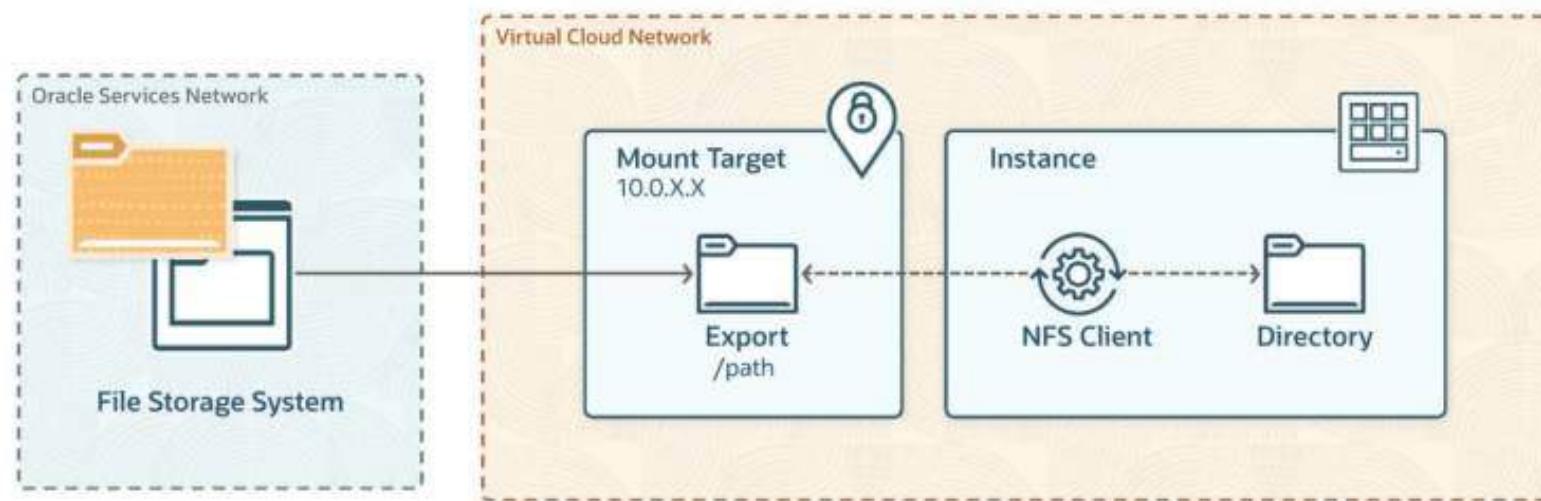
# File Storage Deep Dive

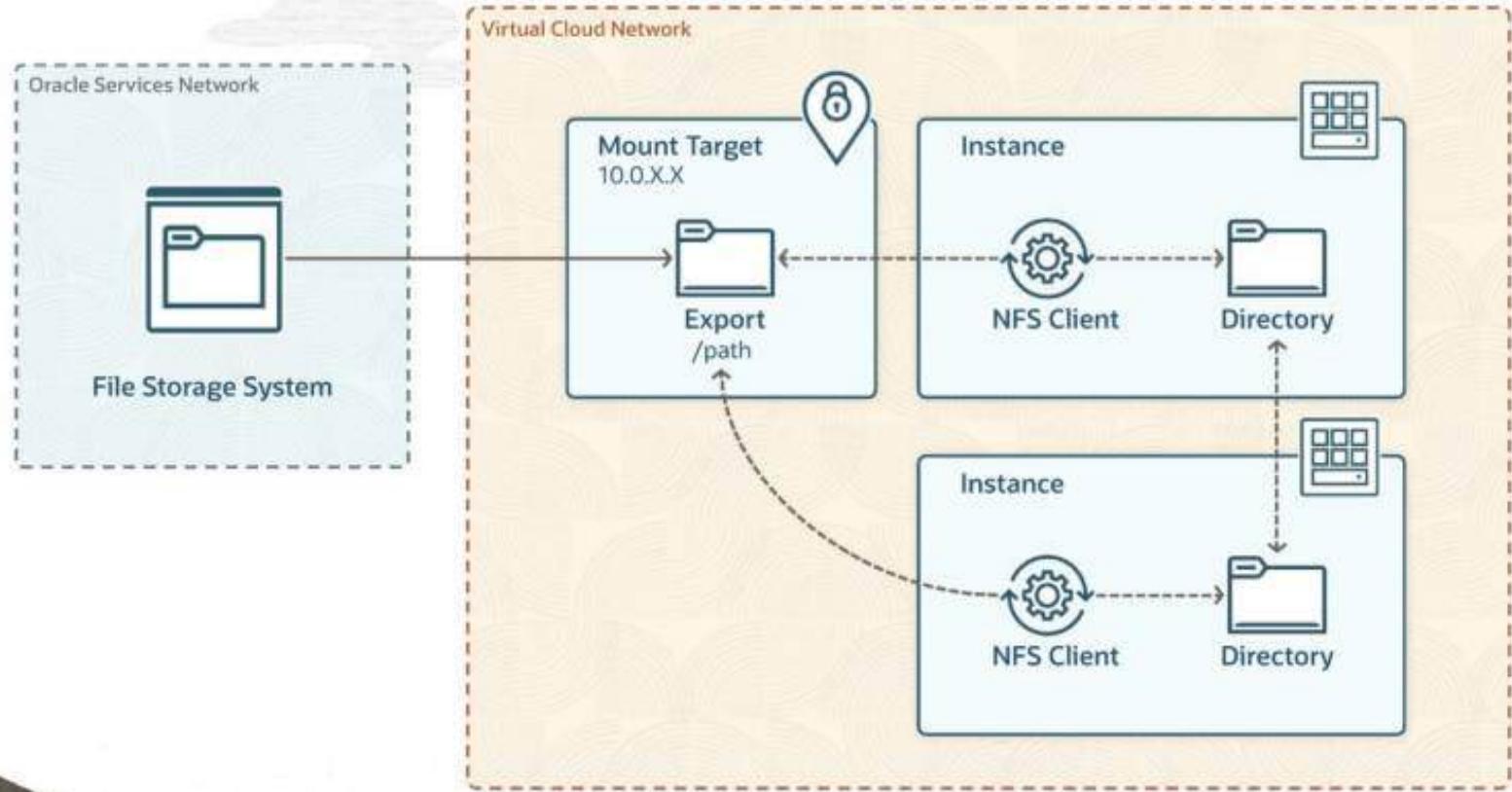
## OCI Cloud Operations

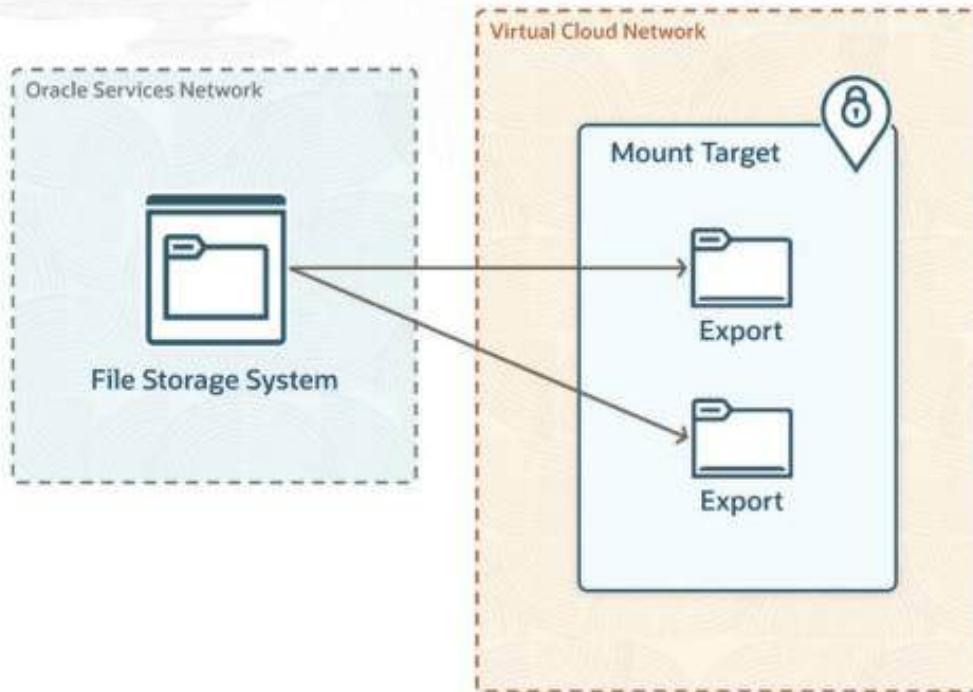


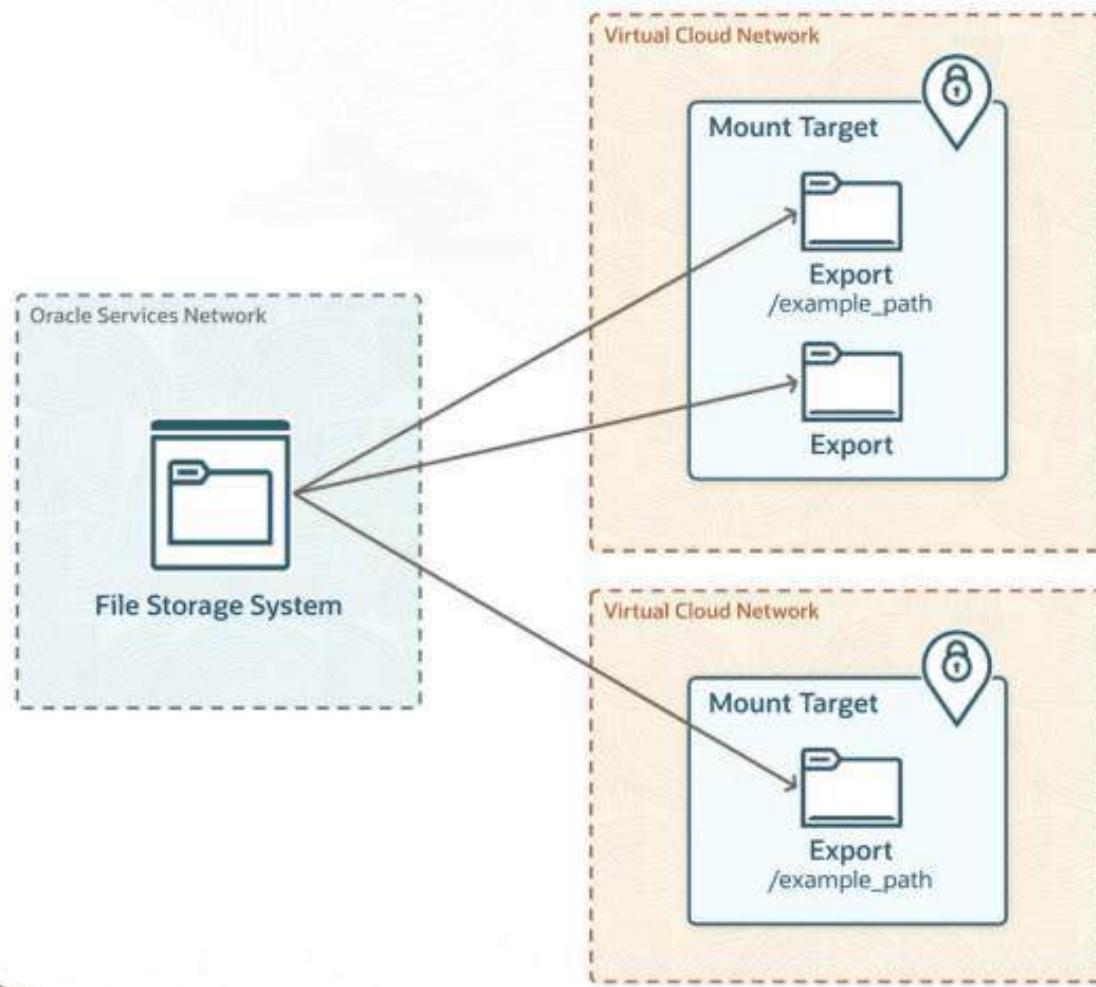


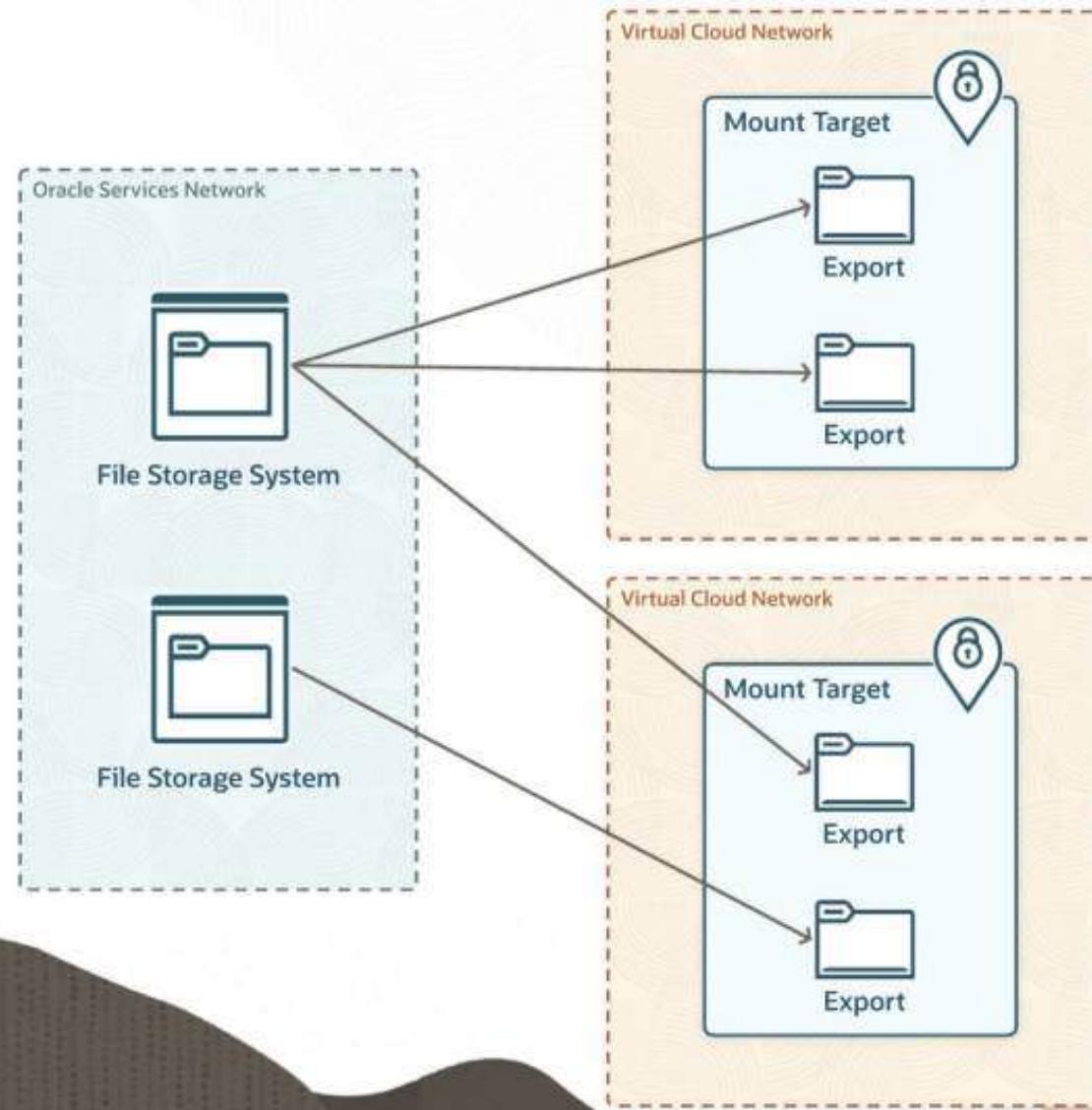




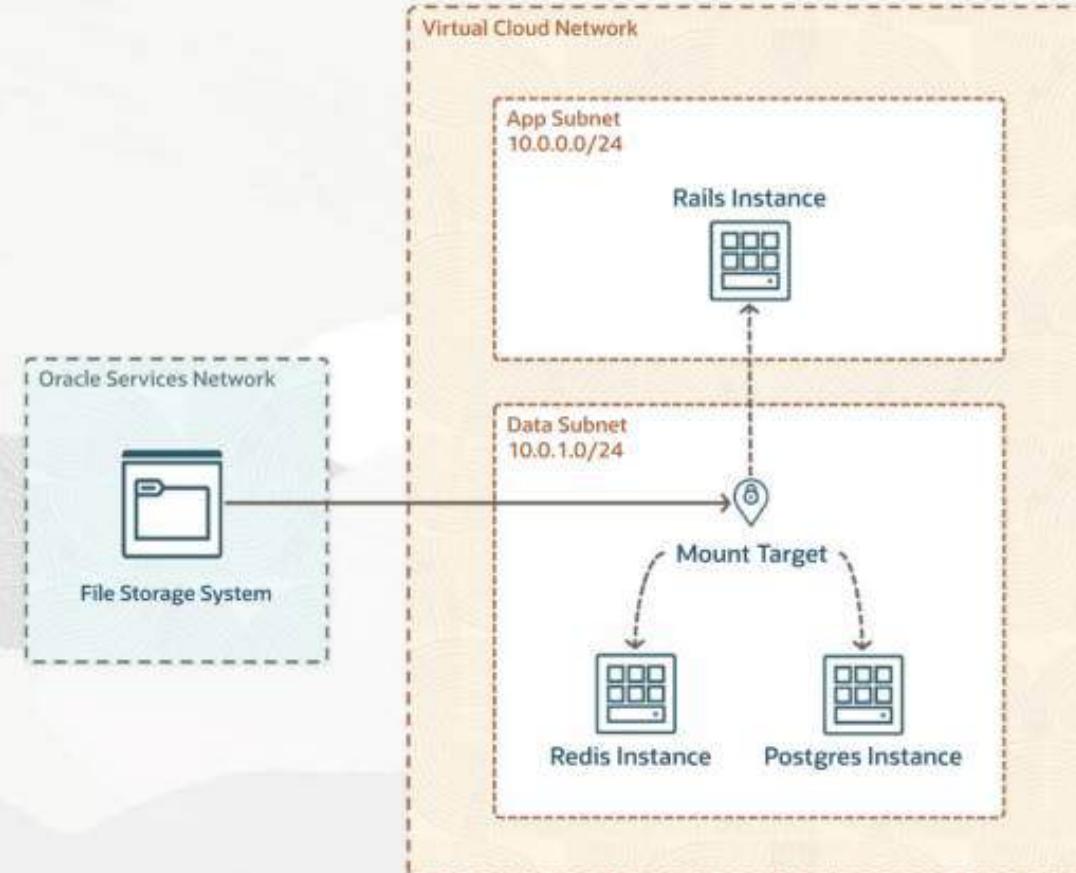








## OCI Region





# Object Storage Deep Dive

## OCI Cloud Operations

## OCI Region

VCN  
10.0.0.0/16

App Subnet  
10.0.0.0/24

Rails Instance

Data Subnet  
10.0.1.0/24

Mount Target



Redis Instance

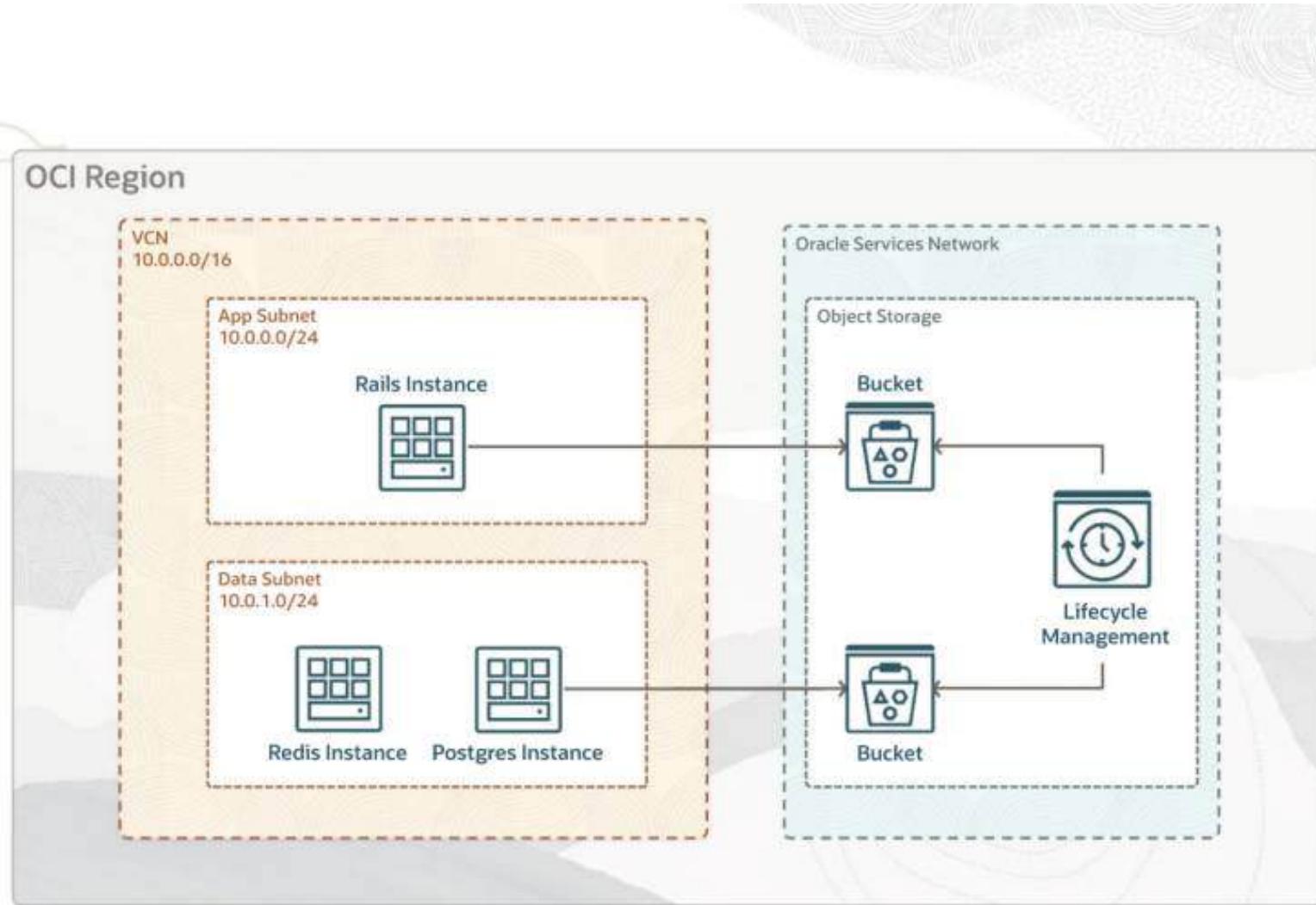


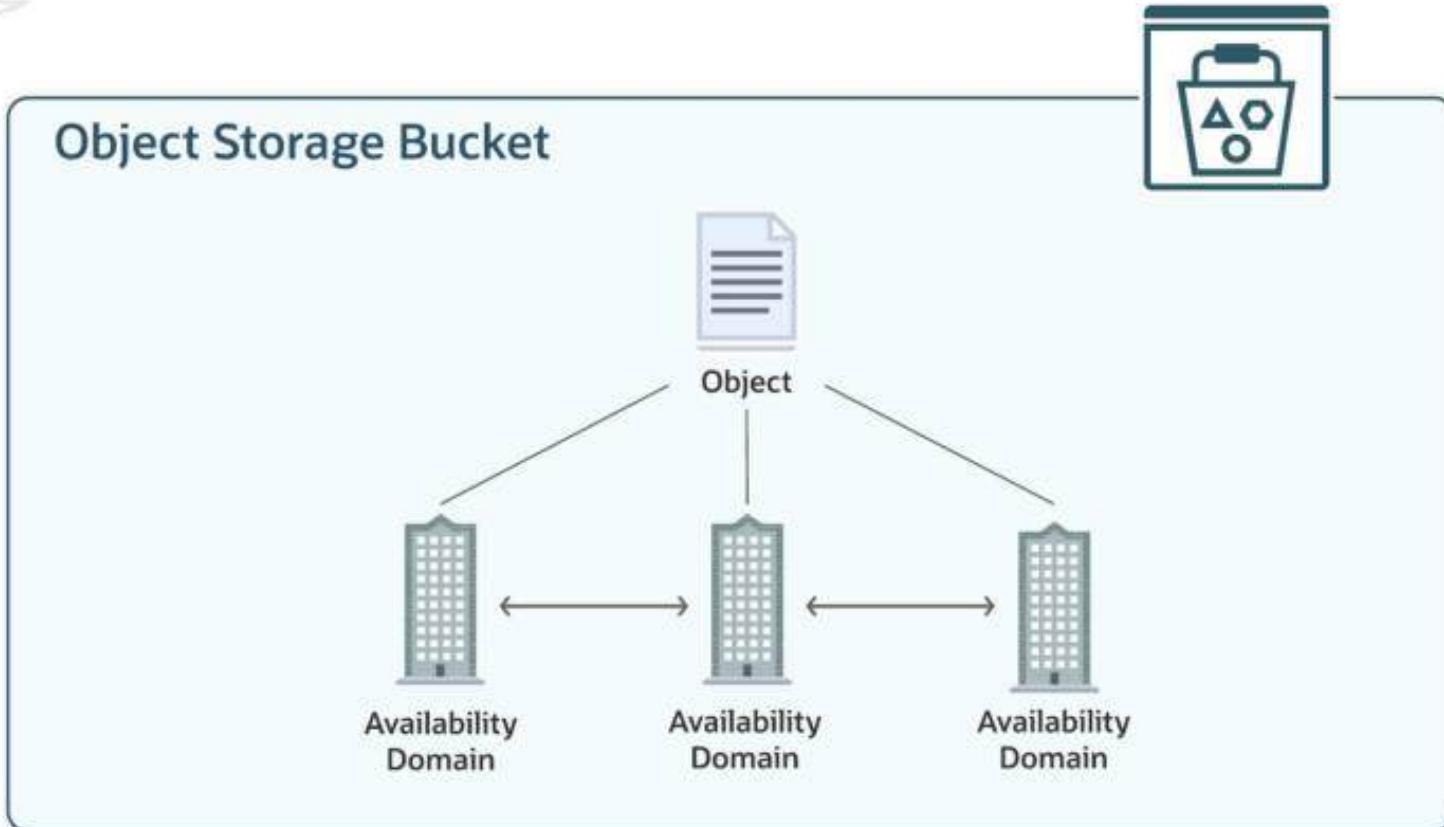
Postgres Instance

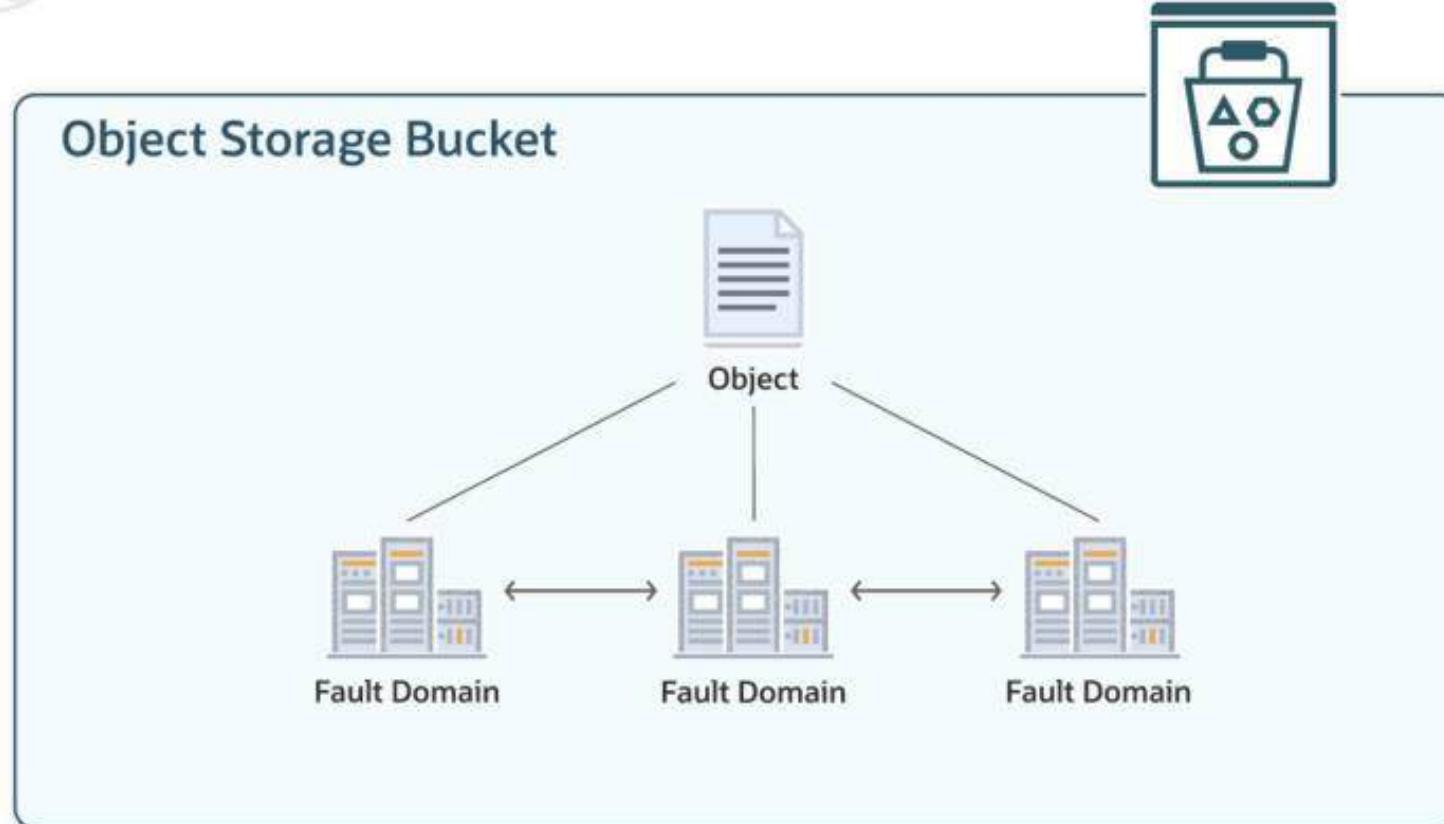
Oracle Services Network

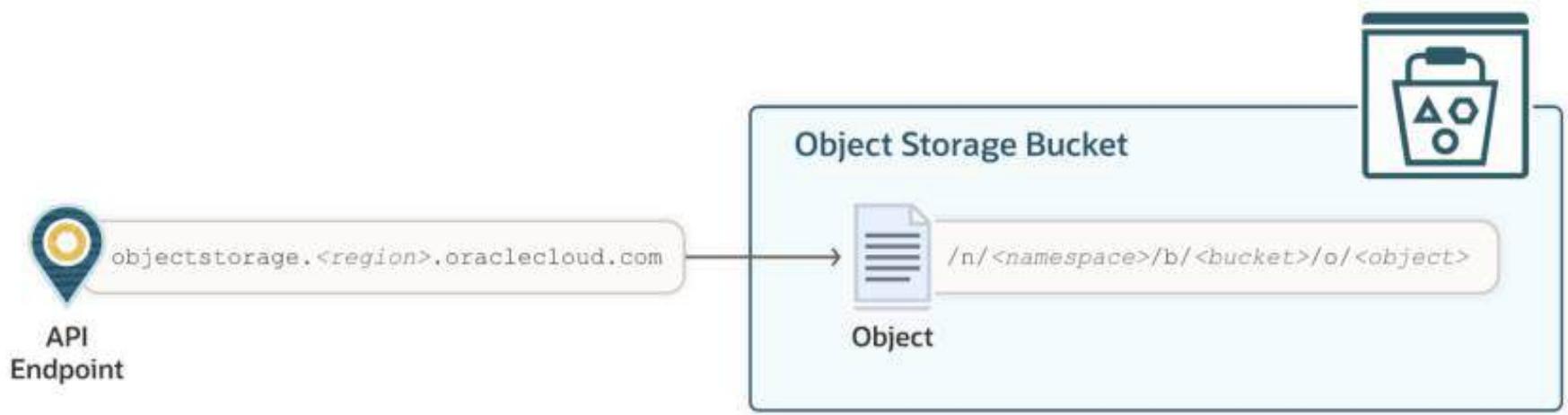


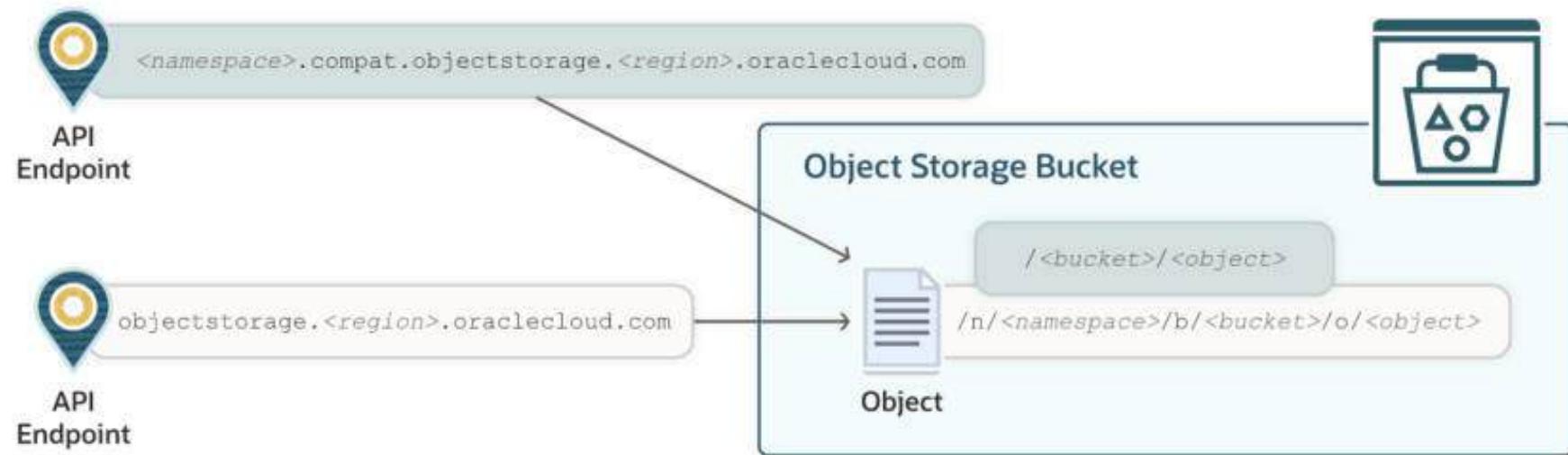
File Storage System













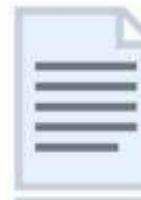
# Versioning

---

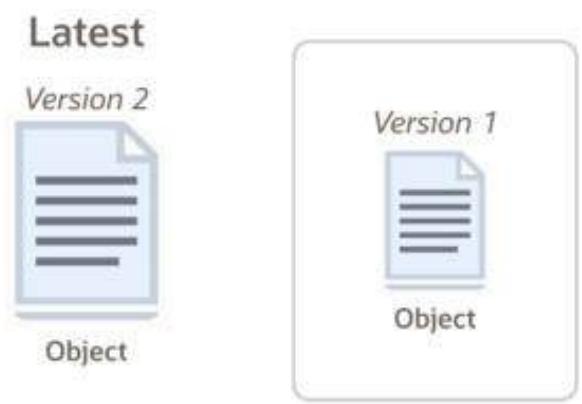


## Latest

*Version 1*



Object







"Object Deletion"

## Latest

Version 5



(Deleted)

Version 4



Version 3



Version 2



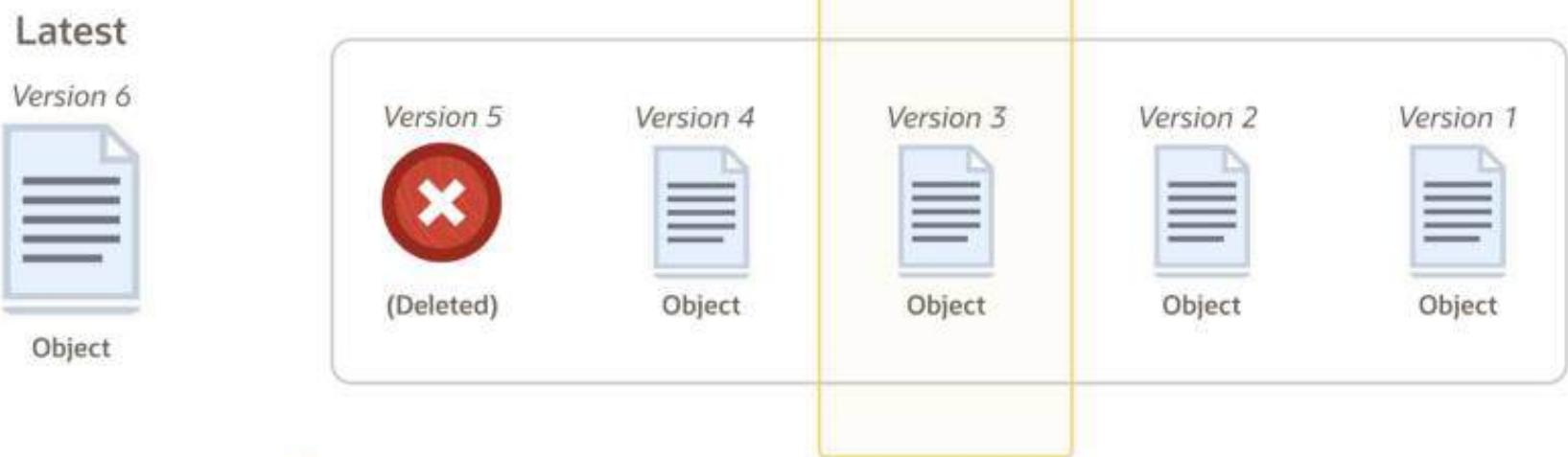
Version 1



### "Object Deletion"

Converts the latest version into a previous version.

Creates a "deleted" marker.











# Lifecycle Management

---



## Lifecycle Policy Rule

Action	Target	Time
Archive	Objects	60 days
Infrequent Access		1 year
Delete		Custom time...

Versioning Disabled





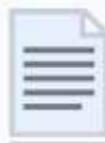
## Lifecycle Policy Rule

Action	Target	Time
Archive	Objects	60 days
Infrequent Access		1 year
Delete		Custom time...



Latest

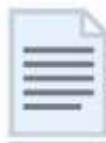
Version 2



Object

Previous Versions

Version 1



Object



Time since last update

Infrequent Access

Archive

Delete

Change tier of all versions

Leave delete marker as latest version



## Lifecycle Policy Rule

Action	Target	Time
Archive	Objects	60 days
Infrequent Access	Previous Object Versions	1 year
Delete		Custom time...



Latest

Version 2



Object



Time since it became a  
"previous version"

Infrequent Access

Archive

Delete

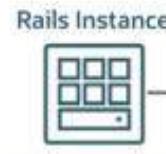
Change tier of specific version

Physically delete version

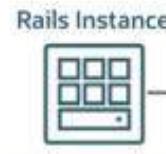
## OCI Region

VCN  
10.0.0.0/16

App Subnet  
10.0.0.0/24

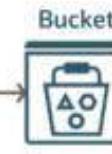


Data Subnet  
10.0.1.0/24



Oracle Services Network

Object Storage



Lifecycle Management

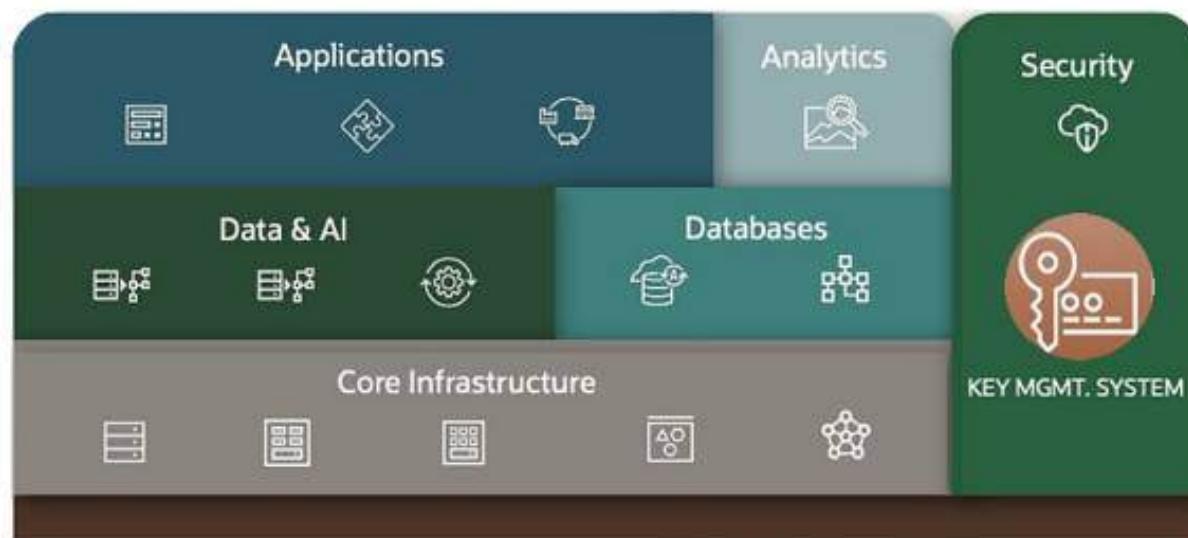


# Secrets and Encryption

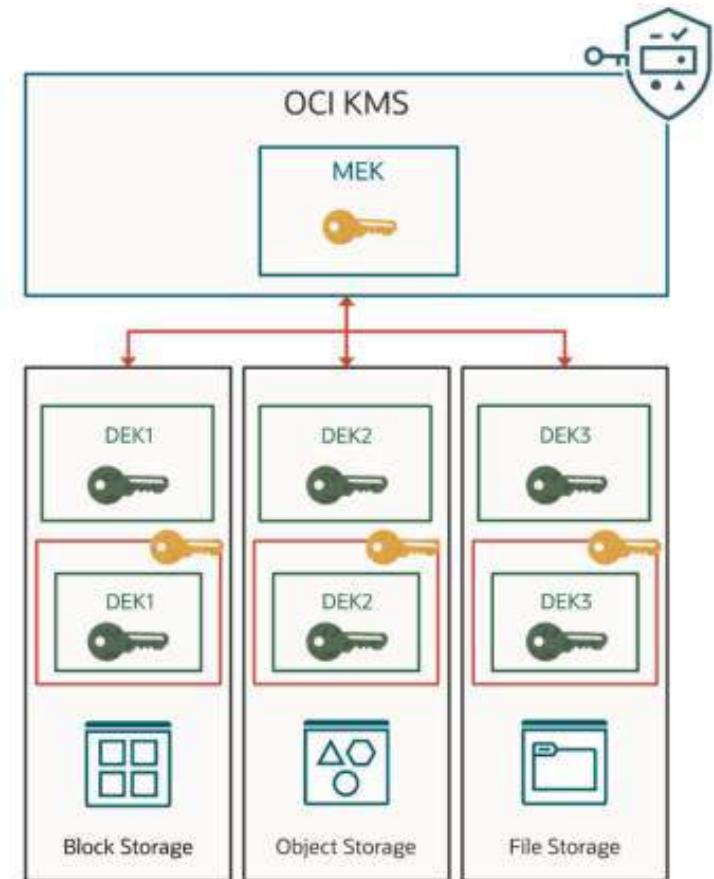
Oracle Cloud Infrastructure

# OCI Key Management Service (KMS)

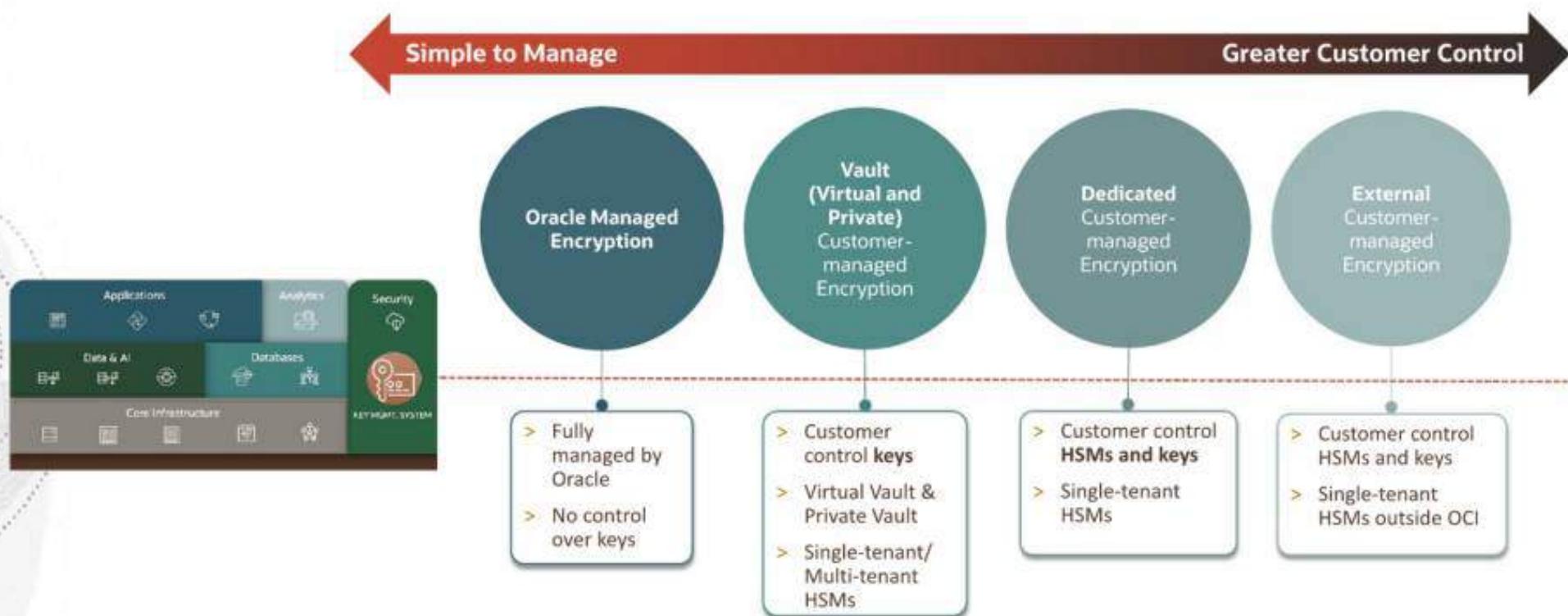
# OCI Encryption Options



# OCI Encryption Options



# OCI Encryption Options

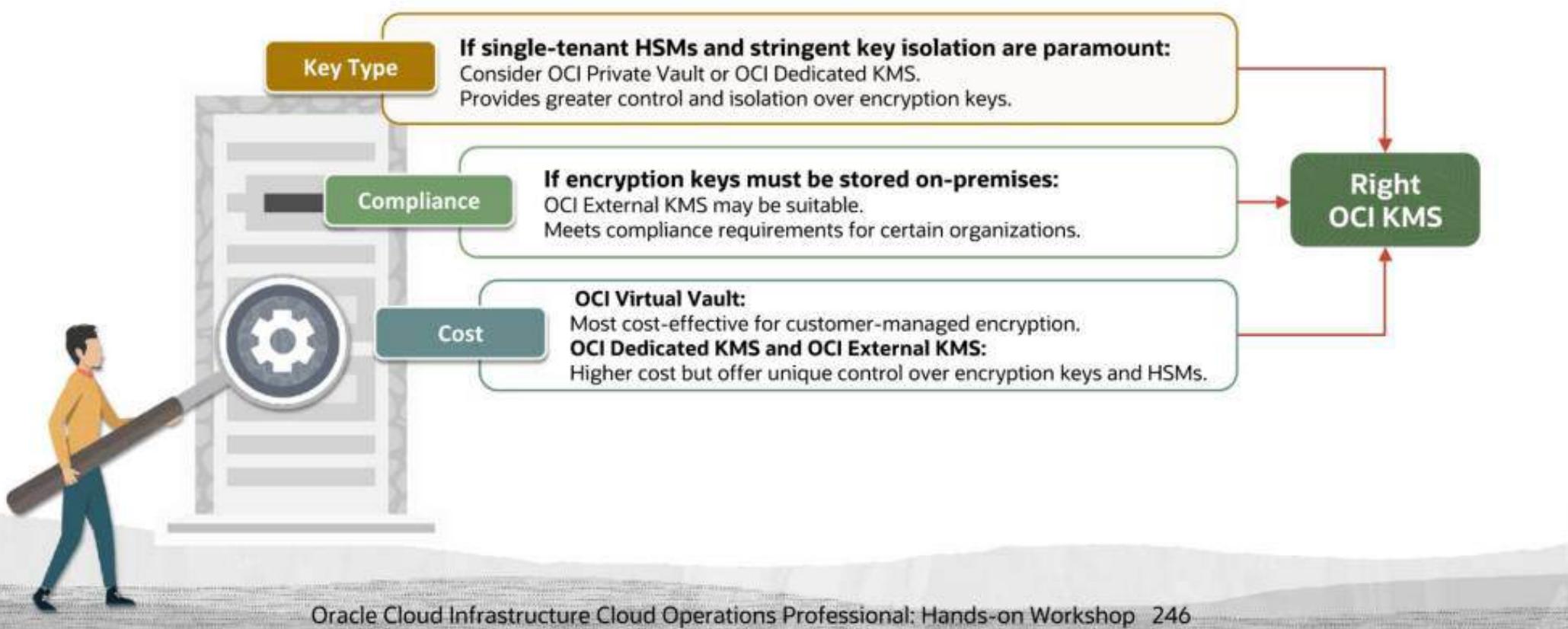


# OCI KMS encryption portfolio

Features	Virtual Vault	Private Vault	Dedicated KMS	External KMS
Tenant isolation	Multitenant	Single tenant	Single tenant	Single tenant
Customer Control	Keys	Keys	Keys and HSMs	Keys and HSMs
Supported key types	HSM and Software	HSM and Software	HSM	External
OCI Services Integration	Yes	Yes	No	Yes
Use cases	Protect data in OCI services using keys in multi-tenant HSMs	Protect data in OCI services using keys in single-tenant HSMs	Standard PKCS#11 interfaces to use keys in single-tenant HSMs	Compliance and Security regulations to use keys outside OCI
Limits	<ul style="list-style-type: none"><li>&gt; Free Tier offering</li><li>&gt; Limits: 10 Vaults and 100 key versions (HSM or SW) per Vault</li></ul>	<ul style="list-style-type: none"><li>&gt; Require explicit approval</li><li>&gt; Max of 3,000 key versions per private vault</li></ul>	<ul style="list-style-type: none"><li>&gt; Require explicit approval</li><li>&gt; Minimum of three HSM partitions</li><li>&gt; Max of 3,000 key versions per HSM partition.</li></ul>	<ul style="list-style-type: none"><li>&gt; Limits: 10 Vaults and 100 key versions per vault.</li></ul>

# Choosing the right OCI KMS offering

- To ensure the best fit for your organization, consider control, security, and other requirements when selecting your OCI KMS offering.



## OCI KMS offers

### Simplifies Key Management

- > Centralizes storage and management of encryption keys



### Enhanced Data Protection

- > Protects data at rest and in transit
- > Supports various encryption key types



### Compliance & Control

- > Allows Bring Your Own Keys, Create in OCI, or Hold Your Own Keys
- > Utilizes FIPS 140-2 Level 3-certified HSMs



### Seamless Integration with OCI Services

- > Enables encryption usage with storage, database, Fusion Applications, and other OCI services



# Oracle Cloud Infrastructure Encryption Basics

# Encryption Basics

**Encryption** is used to transform plain text data into cipher text.

**Decryption** is used to transform cipher text into plain text.

A **key** is a piece of information, usually a string of numbers or letters that are stored in a file, which, when processed through a cryptographic **algorithm**, can **encrypt** or **decrypt** data.

Encryption **key/key pair** is generated for a specific algorithm that can be used for encryption or digital signing.

**ENCRYPTION**



**DECRYPTION**

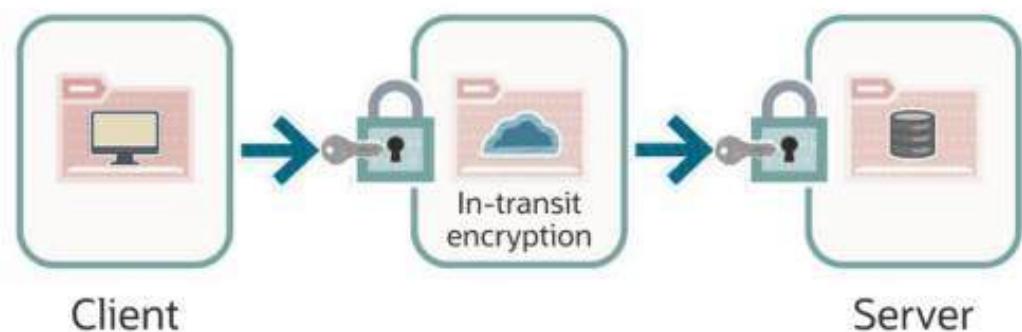


# Encryption at rest and in-transit

Encryption at rest

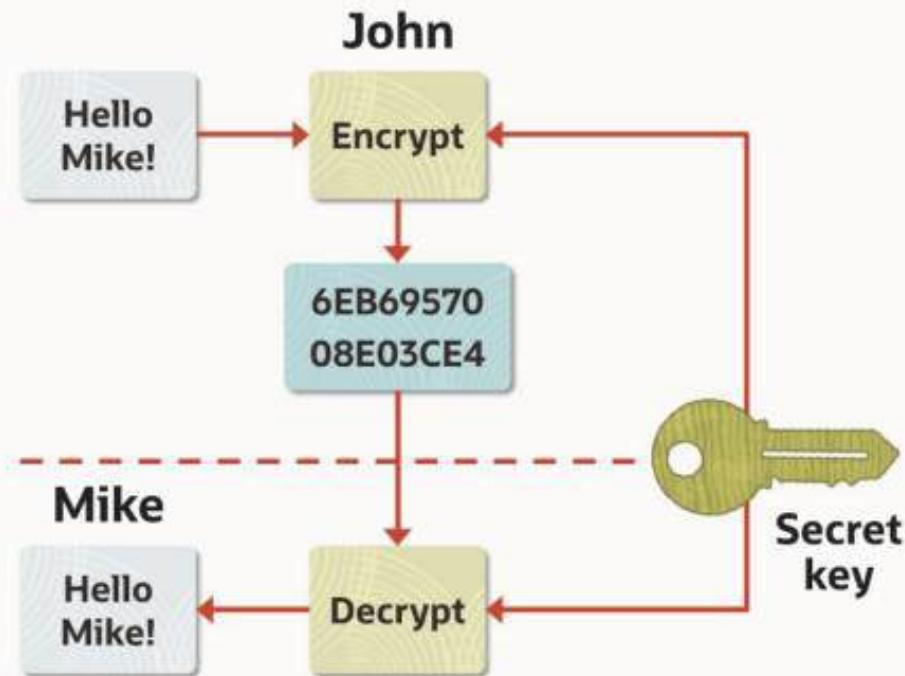


Encryption in-transit



# Symmetric Encryption

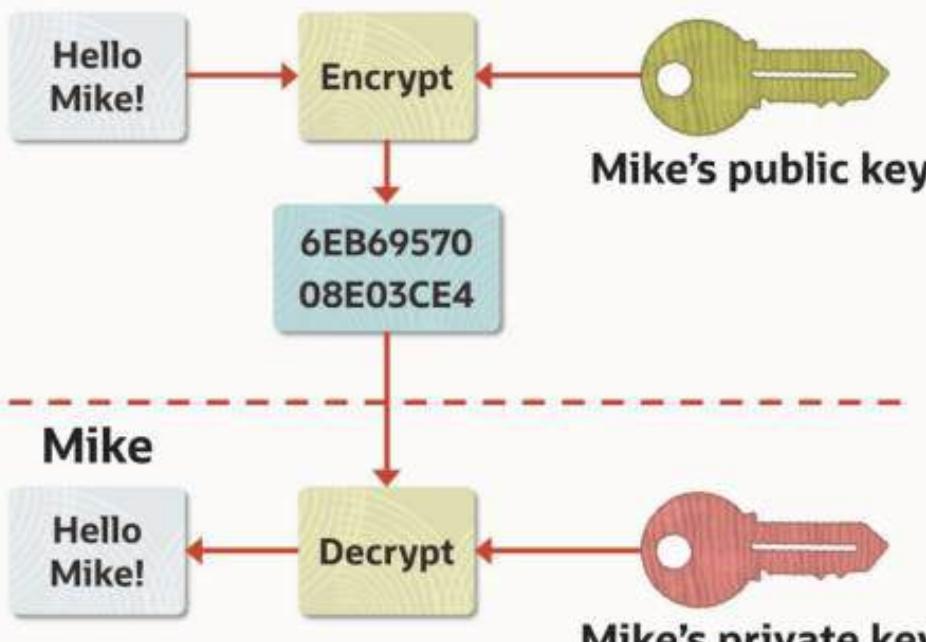
Symmetric-key cryptography is where a single key is used for encryption and decryption



# Asymmetric Encryption

Asymmetric encryption is where different keys are used for encryption and decryption.

**John**



# Encryption Concepts

Encryption is used to transform plain text data into cipher text.

Decryption is used to transform cipher text into plain text.

An encryption key/key pair is generated for a specific algorithm to be used for encryption or digital signing.

AES symmetric keys:

The same key encrypts and decrypts data; it cannot be used for digital signing.

RSA asymmetric keys:

A public key encrypts and private key decrypts data; it can be used for digital signing.

ECDSA keys:

Can be used only for digital signing, not for encryption and decryption of data

# Hardware Security Module (HSM)

HSM is a physical computing device.

Tamper-evident hardware

Used to manage digital keys

Performs cryptographic functions

OCI Vault service uses HSMs that meet the Federal Information Processing Standards (FIPS) 140-2 Security Level 3 security certification.

Tamper-resistant

Requires identity-based authentication

Deletes keys from the device when it detects tampering

# Oracle Cloud Infrastructure Vault Introduction

## OCI Vault



- Supports AES, RSA, and ECDSA algorithms.
- Several services integrate with OCI Vault.
- Vaults, Keys, Secrets

**Centrally manage encryption keys and secret credentials**



# Vaults

## Create Vault

[Help](#)

Vaults provide your growing data and application encryption with scalable key storage. You can start small, with as little as a single key, and grow to thousands of keys to support your growing cloud deployment.

Create in Compartment

rohit\_c

intraderohit (root)/rohit\_c

Name

Make it a virtual private vault

Creates the vault as a dedicated partition on the HSM, sets pricing based on the maximum usage against key limits, and accommodates greater performance needs. [Learn more](#)

**Vaults are logical entities where the Vault service creates and durably stores keys and secrets.**

### Virtual private vault:

Is a dedicated isolated partition in a hardware security module (HSM)

Can store up to 1000 key versions by default

Provides better isolation of your keys / secrets

Can back up to object storage

### Vault in a shared partition:

Shares the same partition with multiple tenants

Charges only for the number of keys/ secrets stored

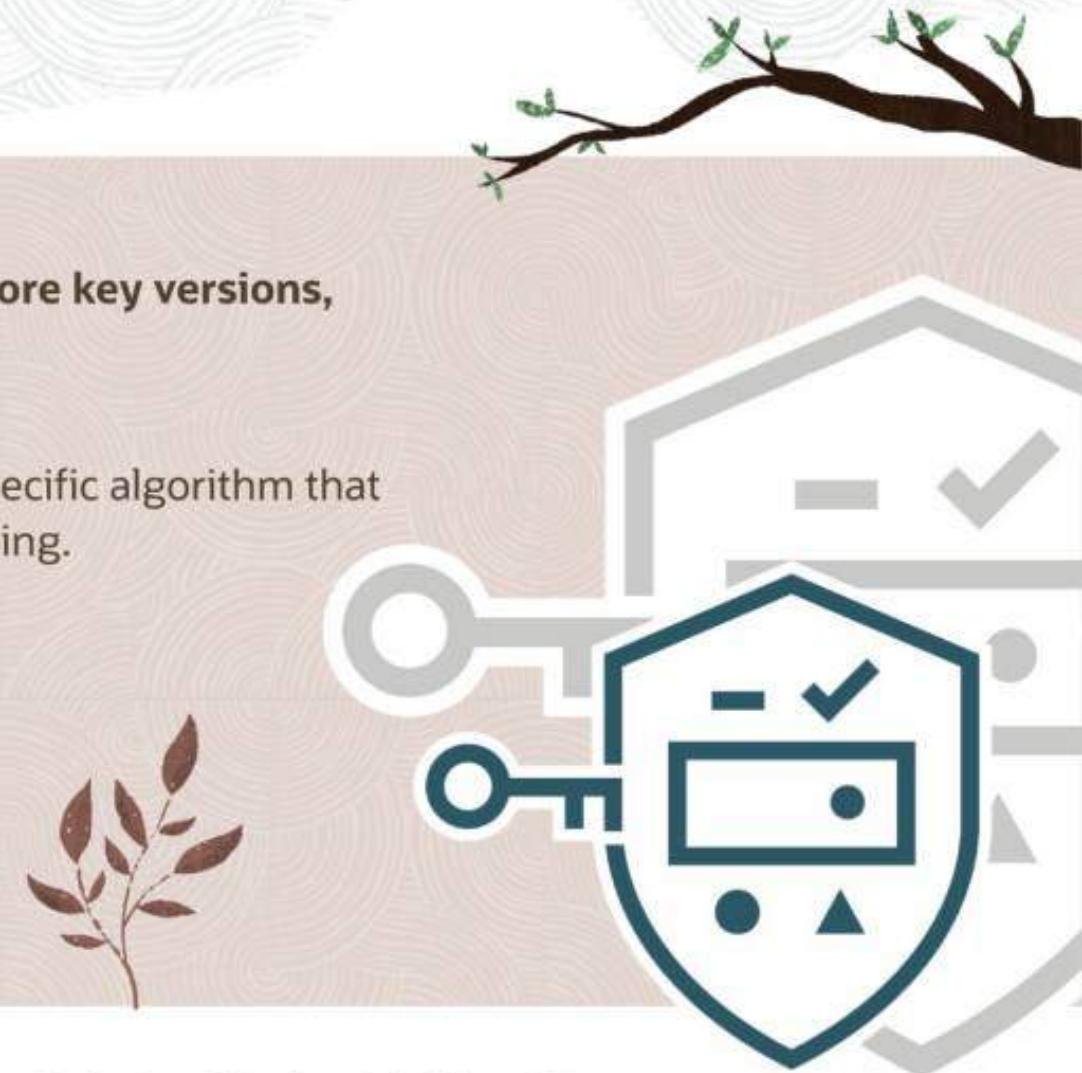
# Keys

**Keys are logical entities that represent one or more key versions, each of which contains cryptographic material.**

A key's cryptographic material is generated for a specific algorithm that lets you use the key for encryption or in digital signing.

Vault service recognizes **three** types of keys:

- Master encryption keys
- Data encryption keys
- Wrapping keys





## Master and Data Encryption Keys

### Master encryption keys (MEK)

You can create or import MEKs into Vault.

MEKs are used to generate data encryption keys.

MEK are always created in a vault.

Protection mode indicates how MEK persists and where cryptographic operations are performed.

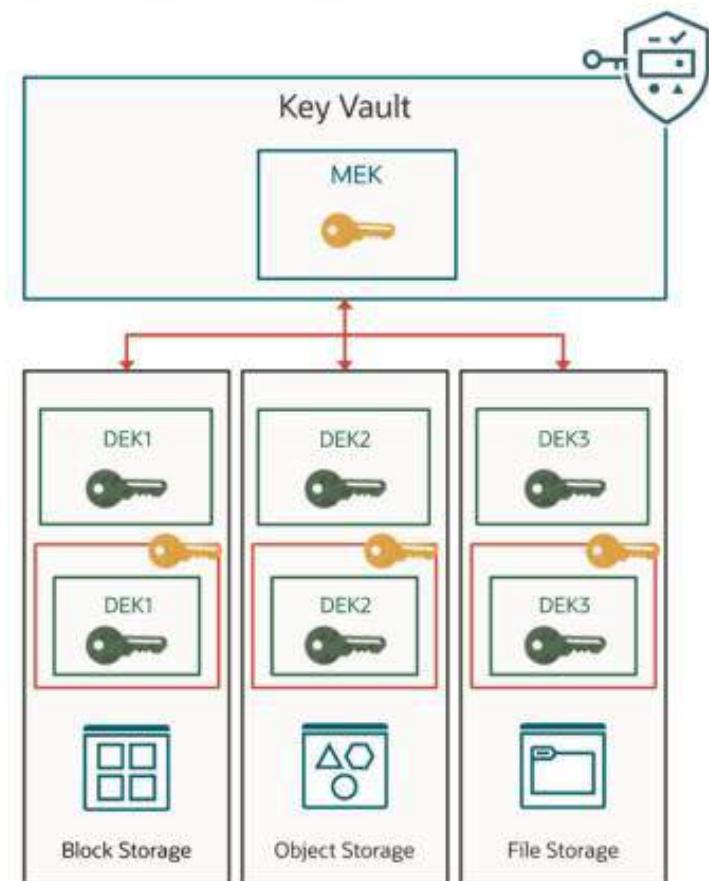
### Data encryption keys (DEK)

Generated by the master encryption key, used to encrypt data

DEK is encrypted with MEK, known as envelope encryption.

OCI services don't have access to plaintext DEK.

### Envelope Encryption



# Master Encryption Keys: Protection Modes

Master encryption keys can have one of two protection modes:

## HSM

Stored in an HSM

Cannot be exported from HSM

All cryptographic operations happen inside HSM.

## Software

Stored on a server

Can be exported to perform cryptographic operations

Software protected while at rest

Encrypted by a root key on HSM

## Create Key

Create in Compartment:

rohit\_c

oracledevclif (root)/rohit\_c

Protection Mode:

HSM

Name:

Key Shape: Algorithm:

AES (Symmetric key used fo...)

Key Shape: Length:

256 bits

Import external key

Create a new key by importing a wrapped file containing key data that matches the specified key shape. For more information, see [Importing Keys](#)

[Show Advanced Options](#)

# Wrapping Keys

A wrapping key is included with each vault by default.

Use the public wrapping key when you need to wrap key material for import into the vault service.

You cannot create, delete, or rotate wrapping keys.

The screenshot shows the Oracle Cloud Infrastructure Vault service interface. It includes tabs for 'Vault Information' and 'Tags'. Under 'General Information', details are provided for a compartment named '99523923-CD1'. The compartment has a Virtual Private IP of 'No'. It lists a 'Cryptographic Endpoint' at 'https://bsocg2yaasvq-crypto.kms.us-ashburn-1.oraclecloud.com' and a 'Management Endpoint' at 'https://bsocg2yaasvq-management.kms.us-ashburn-1.oraclecloud.com'. Under 'Wrapping Key Information', it shows a 'Public Wrapping Key' labeled 'PUBLIC KEY', an OCID of 'ykhna', and a status of 'Enabled'. The key length is listed as 4096 bits. A note indicates that the key was modified on 'Wed, Feb 10, 2021, 05:03:48 UTC'.

The screenshot shows the 'Create Key' dialog box. It is set to 'CREATE IN COMPARTMENT' '99523923-CD1'. The 'PROTECTION MODE' is set to 'HSM'. The 'NAME' field contains 'ImportedMEK'. The 'KEY SHAPE ALGORITHM' is 'AES' and the 'KEY SHAPE LENGTH' is '256 bits'. The 'IMPORT EXTERNAL KEY' checkbox is checked. Below it, instructions say 'Create a new key by importing a wrapped file containing key data that matches the specified key shape. For more information, see [Importing Keys](#)'. The 'PUBLIC WRAPPING KEY' field contains 'PUBLIC KEY'. The 'WRAPPING ALGORITHM' is 'RSA\_OAEP\_AES\_SHA256'. At the bottom, there is an 'EXTERNAL KEY DATA SOURCE' section with a placeholder 'Drop a file or select one...' and a 'Show Advanced Options' link. The 'Create Key' button is highlighted in blue.



## Rotating Keys

- Each MEK is automatically assigned a key version.
- When you rotate a MEK, a new key version is generated.
- Periodically rotating keys limits the amount of data encrypted or signed by one key version.
- Key rotation reduces the risk if a key is ever compromised.
- A key's OCID remains the same across rotations.
- Older versions cannot be used for encryption, but can be used to decrypt data previously encrypted with it.

### MEK

[Edit Name](#) [Disable](#) [Add Tags](#) [Move Resource](#) [Delete Key](#)

[Key Information](#) [Tags](#)

OCID: ...b7izla [Show](#) [Copy](#)  
Created: Tue, Oct 19, 2021, 08:01:13 UTC  
Compartment: Intoraclderohit (root)/rohit\_c  
Protection Mode: Software

Vault: Vault1-us-san  
Key Version: ...vdyu  
Algorithm: AES  
Length: 256 bits

### Versions

[Rotate Key](#)

OCID	State	Source
...vdyuoq <a href="#">Show</a> <a href="#">Copy</a>	Enabled	Internal

Oracle Cloud Infrastructure

# Demo: Vault Basics Part 1

## Oracle Cloud Infrastructure Demo: Vault Basics Part 2

## Oracle Cloud Infrastructure Import and Export Keys

# Cryptographic and Management Endpoints



- **Data plane URL/ Cryptographic endpoint**  
Unique service endpoint for cryptographic operations
- **Control plane URL/ Management endpoint**  
Unique service endpoint for management operations
- **Needed when using the CLI for key operations**  
Needed when using the CLI for key operations

Vault1-us-sanjose-1

Edit Name Add Tags Move Resource Delete Vault

Vault Information Tags

General Information

Compartment: intoraciero (root)/rohit\_c  
OCID: ...edmza [Show](#) [Copy](#)  
Created: Tue, Oct 19, 2021, 07:48:45 UTC  
HSM Key Version Usage: 2 [i](#)  
Software Key Version Usage: 3 [i](#)

Virtual Private: No

Cryptographic Endpoint: <https://ebqw46c5aabmk-crypto.kms.us-sanjose-1.oci.oraclecloud.com> [i](#)

Management Endpoint: <https://ebqw46c5aabmk-management.kms.us-sanjose-1.oci.oraclecloud.com> [i](#)



# Cryptographic and Management Endpoints

Unique service endpoint to perform cryptographic operations against.

Cryptographic operations include 'Encrypt,' 'Decrypt,' and 'GenerateDataEncryptionKey', 'Sign,' and 'Verify' operations.

Virtual Private: No

Cryptographic Endpoint: <https://ebqw46c5aabmk-crypto.kms.us-sanjose-1.oci.oraclecloud.com> ⓘ

Management Endpoint: <https://ebqw46c5aabmk-management.kms.us-sanjose-1.oci.oraclecloud.com> ⓘ

Unique service endpoint to perform management operations against.

Management operations include 'Create,' 'Update,' 'List,' 'Get,' and 'Delete' operations for keys.

# Crypto Operations

**Encrypt**

**Decrypt**

**GenerateDEK (Data Encryption Key)**

```
oci kms crypto encrypt --plaintext foobar --key-id  
ocid1.key.oc1.iad.bbowsfp2aaeuk.abuwcljt4p4jxjlvez53falyg5awtcvfyi6ckmlwkzjfj  
puirudpau5kifwoa --endpoint https://ebqw46c5aabmk-crypto.kms.us-sanjose-  
1.oci.oraclecloud.com  
{  
  "data": {  
    "ciphertext":  
    "IexQjs3Rq8VAABKPcP8T+scr/tBWAAJV6SWOCerm+No+b5TO0nHB9Ni5AAAAAA=="  
  }  
}
```



You can import your own AES symmetric and RSA asymmetric keys into OCI Vault service.

After being imported, the function will be the same as if the keys were generated by OCI Vault service.

Keys to be imported must be wrapped with the public wrapping key provided with each vault.

Vault's wrapping key pair makes it possible for the HSM to unwrap and store the key securely.



You can export a software-protected Master Encryption Key or key version if you want to use it to perform cryptographic operations in an app running on a client.

Exporting a key requires you to generate your own RSA key pair to wrap and unwrap the key material.

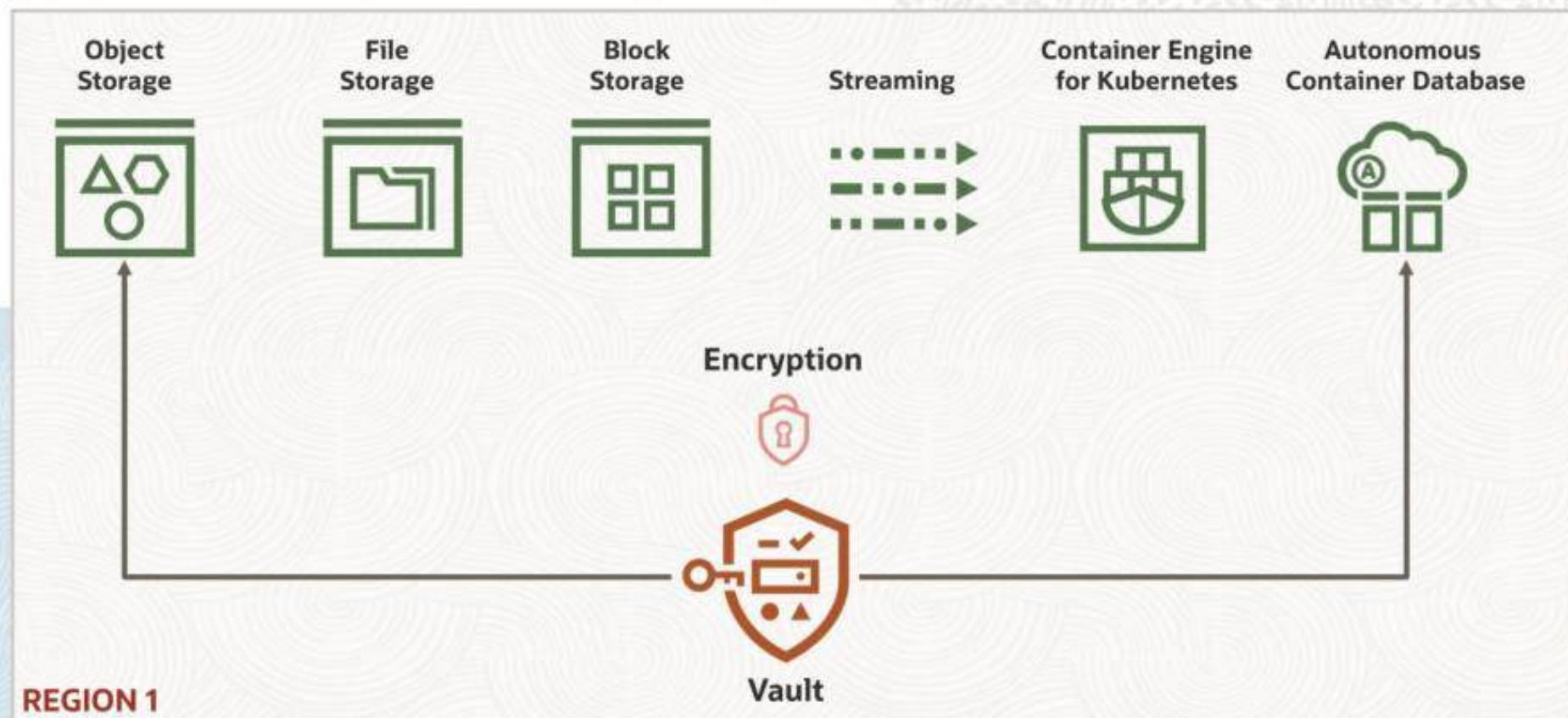
You can use the key locally, and then discard the key from local memory to protect the key contents.

## Exporting Keys or Key Versions

Oracle Cloud Infrastructure

# OCI Services Integration with Vault

# OCI Services Integration with Vault



# Encryption Using Oracle-Managed Keys

## Encrypt data

Block and boot volumes, file systems, Object Storage buckets, OCI Container Engine for Kubernetes secrets, Autonomous Container Databases (on dedicated Autonomous Exadata Infrastructure and Exadata Databases) and OCI Streaming stream pools are by default encrypted using Oracle-managed keys.

An Oracle-managed vault has a master encryption key, which provides a data encryption key to the respective service to encrypt data.

Thus, any data in these services is encrypted by default.

### Encryption

- Encrypt using Oracle managed keys  
Leaves all encryption-related matters to Oracle.
- Encrypt using customer-managed keys

Requires a valid key from a vault that you have access to. [Learn more](#)



## Encryption Using Customer-Managed Keys

### Encrypt data with your own key.

You can use your own master encryption keys in your vault.

Your master encryption key provides the data encryption key to the service.

#### Encryption

- Encrypt using Oracle managed keys

Leaves all encryption-related matters to Oracle.

- Encrypt using customer-managed keys

Requires a valid key from a vault that you have access to. [Learn more](#)

Vault in `rohit_c` ([Change Compartment](#))

Vault1-us-sanjose-1

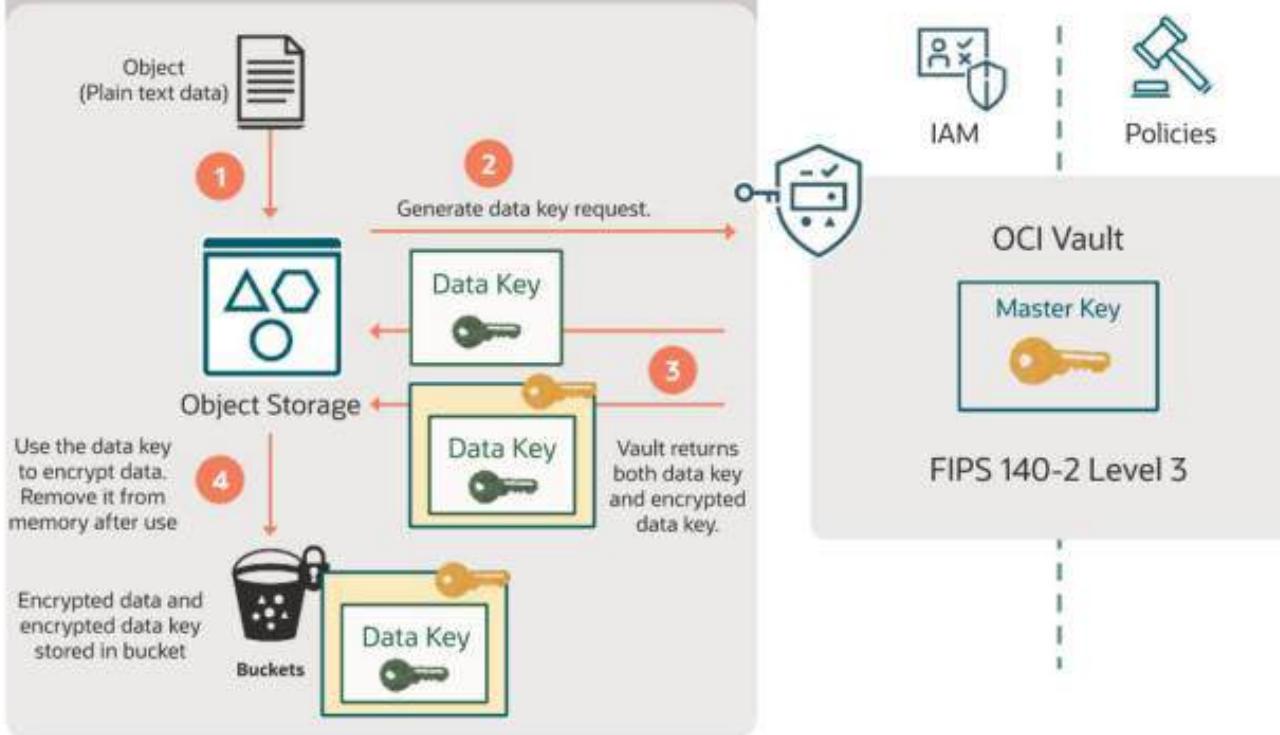
Master Encryption Key in `rohit_c` ([Change Compartment](#))

MEK

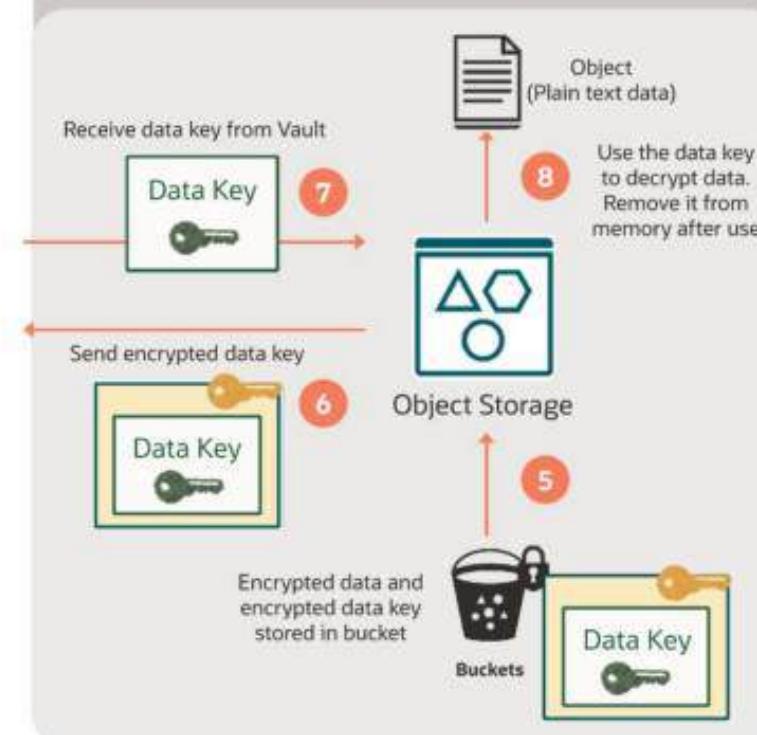
# OCI Object Storage Integration with Vault



## Encrypt process

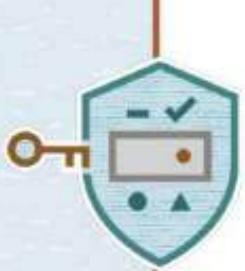


## Decrypt process



## Oracle Cloud Infrastructure

# Back up and Replicate Vaults and Keys



## Backing Up Vaults and Keys

- Back up and keep the resources before deleting the vault or key.
- Only virtual private vaults are supported for backups.
- Only a Master Encryption Key (MEK) of HSM protection type can be part of backups.
- MEKs are always associated with a vault. This relationship persists even as the key is backed up and restored.
- Backups are kept in existing or new object storage buckets.
- You can copy the backups to object storage buckets in another region.
- Backups are useful for disaster recovery scenarios.



## Backing Up Vaults and Keys

- Back up exports identifying information about the vault or key.
- Vault service encrypts the backups, and only the service can restore them.
- Backups can optionally include keys (assuming the vault has keys in a supported lifecycle state when you perform the backup).
- You can back up only one vault or one key at a time.
- Backup operations require you to specify where to download the backup.

# Restoring Vaults and Keys

A key must always be associated with a vault before they can be restored.

First restore the vault, and then restore the key.

## Restore Vault

Restore a vault from a backup you created earlier. You can restore a backup of a vault to its original compartment and tenancy in the same region or a different region. Restoring a vault generates a work request that you can view to track the progress of the operation.

Choose a source

### Object Storage Bucket

Import a backup from a bucket.

### Object Storage URL

Import a backup from a unique URL that you generated earlier to store the backup.

### Upload a File

Import a backup from an uploaded file.

Select a bucket in rohit\_c ([Change Compartment](#))

Select a bucket

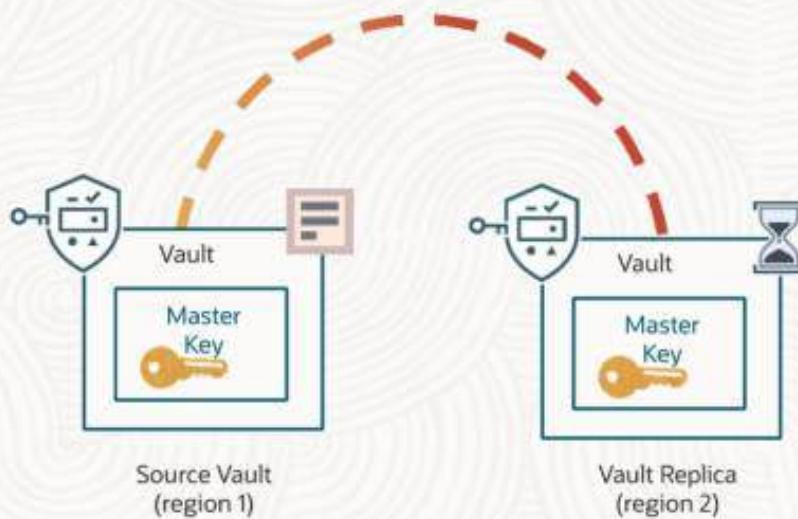
Select a file

Select a bucket first





# Cross-Region Replication



Cross-region replication helps in disaster recovery scenarios.

Only virtual private vaults are supported for replication.

When replication is configured, Vault service automatically synchronizes creation, deletion, update, or moving of any keys between the source and replica vaults.

Only one destination vault can exist for a given source vault at any time.

You cannot create keys directly in the vault replica, nor back up a vault replica.

You can support cryptographic operations against the vault replica and keys.

You can delete the vault replica to stop replication.

## Oracle Cloud Infrastructure

# Demo: OCI services integration with Vault

# Oracle Cloud Infrastructure Secrets



# What's a Secret?

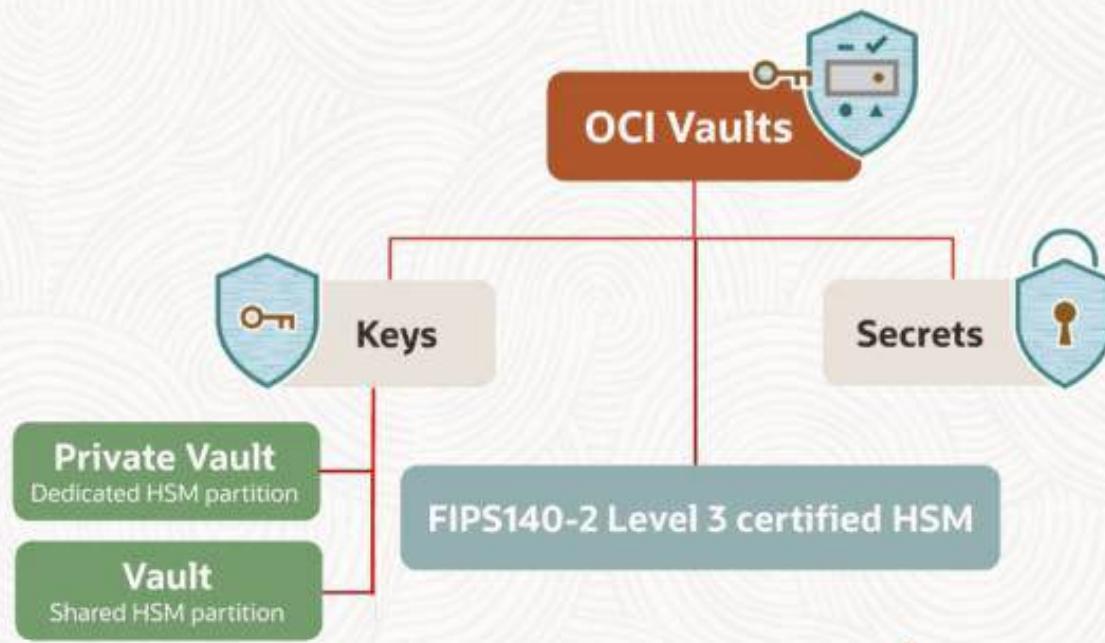
Using a Secret Manager is the foremost security best practice.

The screenshot shows a news article titled "Trello Scrambles To Rescue Users Who Foolishly Used Its Service To Store Passwords". The article discusses how Trello users stored their passwords in plain text, which was then encrypted. The screenshot includes the header, sidebar with recent posts, and the main content area with the article title and a small image of a keyboard.

The screenshot shows a Wireshark capture of network traffic. A red arrow points from the Gizmodo article to a specific frame in the packet list. A blue callout box highlights the "Credentials Transmitted in Cleartext" section, which details an "HTML Form URL Encoded" packet containing "username" and "password" fields. The packet bytes show the transmitted data, including the clear text credentials.

Credentials Transmitted in Cleartext

# Secrets



# Secrets



You can store other information like public keys or passwords as secrets in the vault.

This is a more secure way to store and retrieve them. You can create secrets by using the console, CLI, or API.

Secrets can be rotated to reduce impact in case the secret is exposed or compromised.

## Create Secret

[Help](#)

Create in Compartment

rohit\_c

infracrashrohit (root)/rohit\_c

Name

DBPassword

Description

Password for the database

Encryption Key in rohit\_c ⓘ [\(Change Compartment\)](#)

MEK

Secret Type Template

Plain-Text

Secret Contents

Passw@rd1

Show Base64 conversion

Show Advanced Options

[Create Secret](#)

[Cancel](#)



# Secrets Rules



## Secret Reuse Rule

Prevents the reuse of secret contents across different versions of a secret.

## Secret Expiry Rule

Restricts how long the secret contents of a particular secret version can remain in use.

This rule can also block the retrieval of secret contents for a secret or secret version past the configured expiration date.

Secret rules govern the use and management of secrets. For more information about secret rules, see [Rules for Secrets](#)

### Rule Type

Secret Reuse Rule

### Configuration

Enforce on deleted secret versions

Secret Expiry Rule

Version expiry interval: 10 days

Block content retrieval on expiry

Secret absolute UTC time of expiry

# Oracle Cloud Infrastructure Demo: Secrets

# Disaster Recovery



## Oracle Cloud Infrastructure

# High Availability

---

# High Availability Concepts



Computing environments configured to provide nearly full-time availability are known as high availability systems.

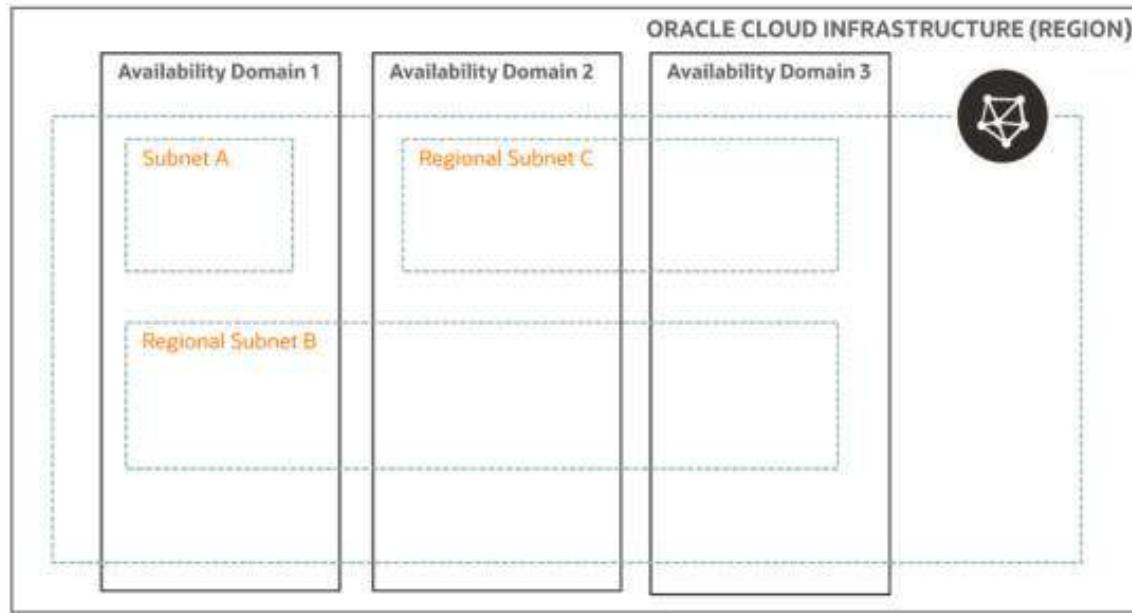
Such systems typically have redundant hardware and software that makes the system available despite failures.

Well-designed high availability systems avoid having single points of failure.

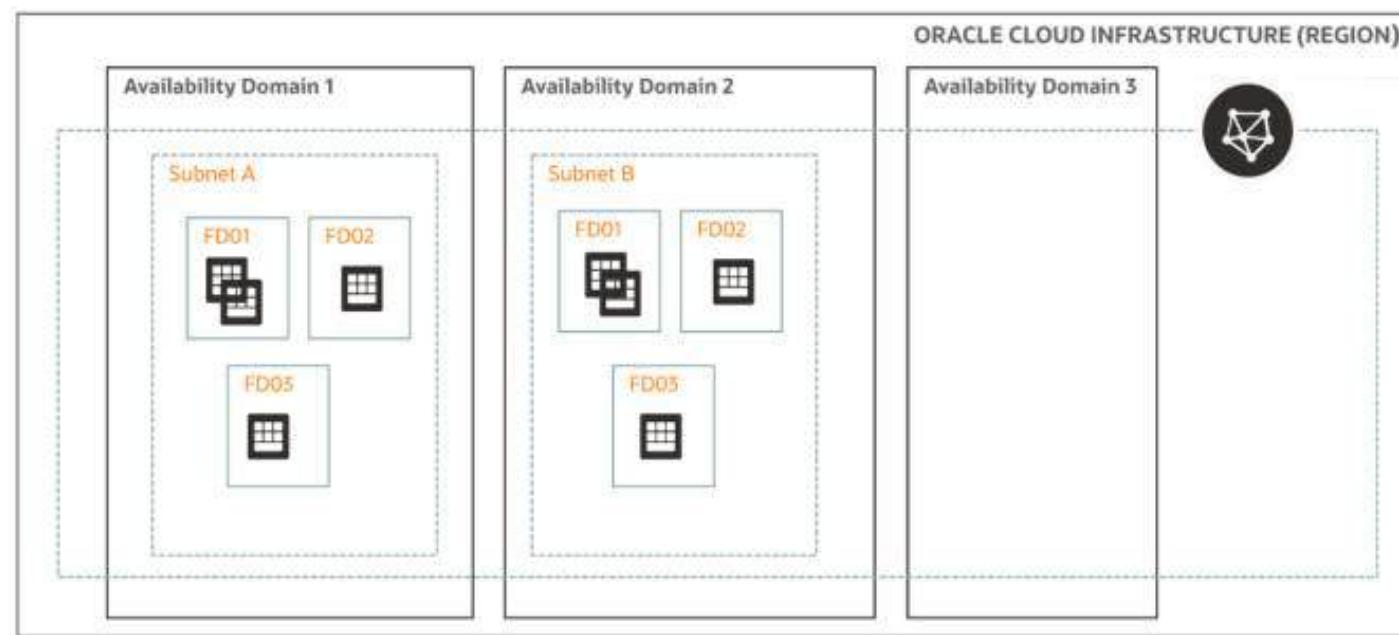
When failures occur, the failover process moves processing performed by the failed component to the backup component

The more transparent that failover is to users, the higher the availability of the system.

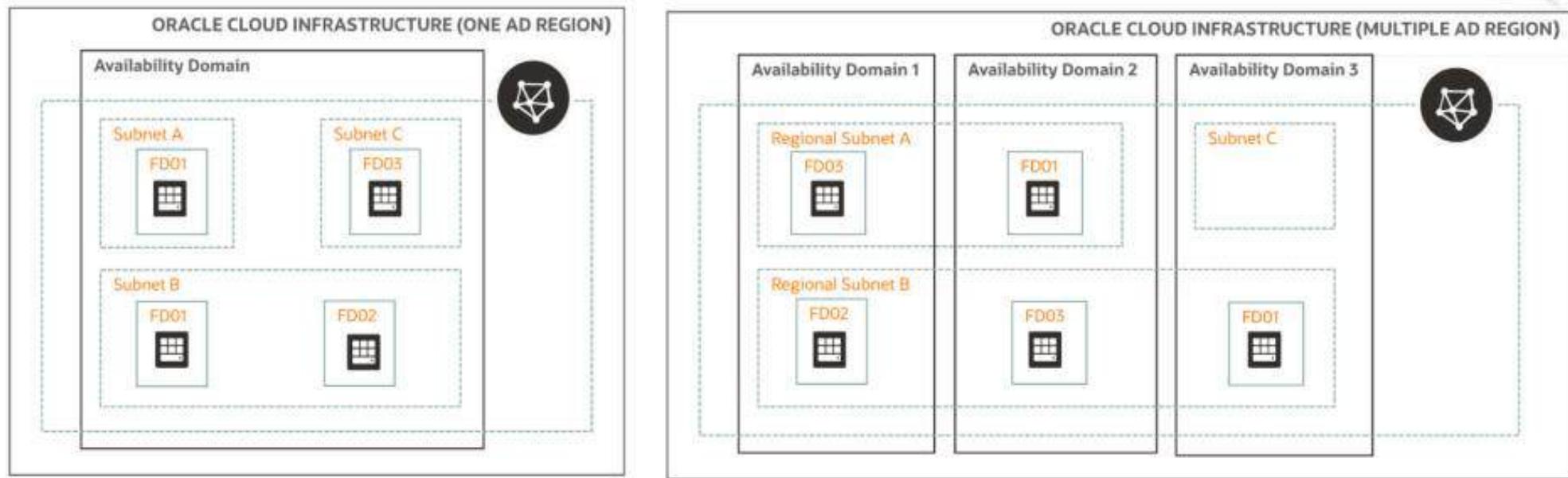
# Availability Domains



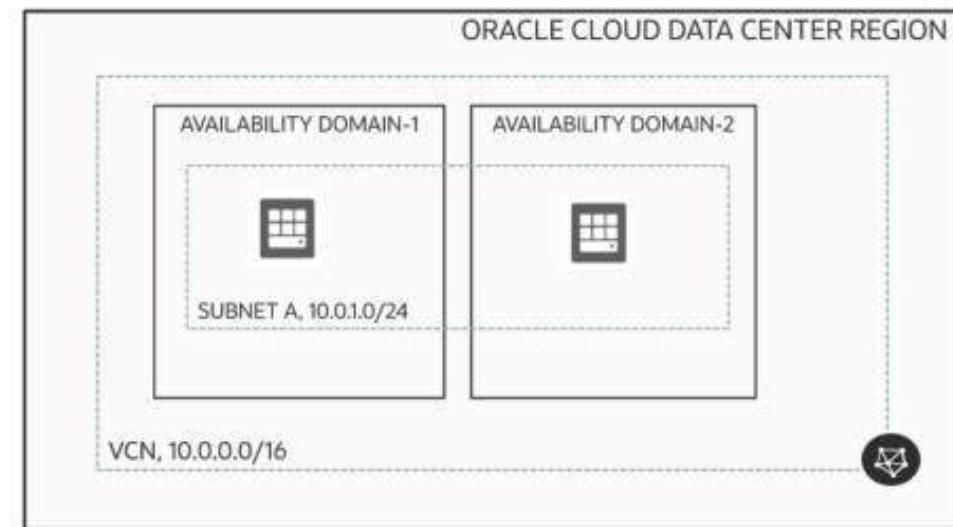
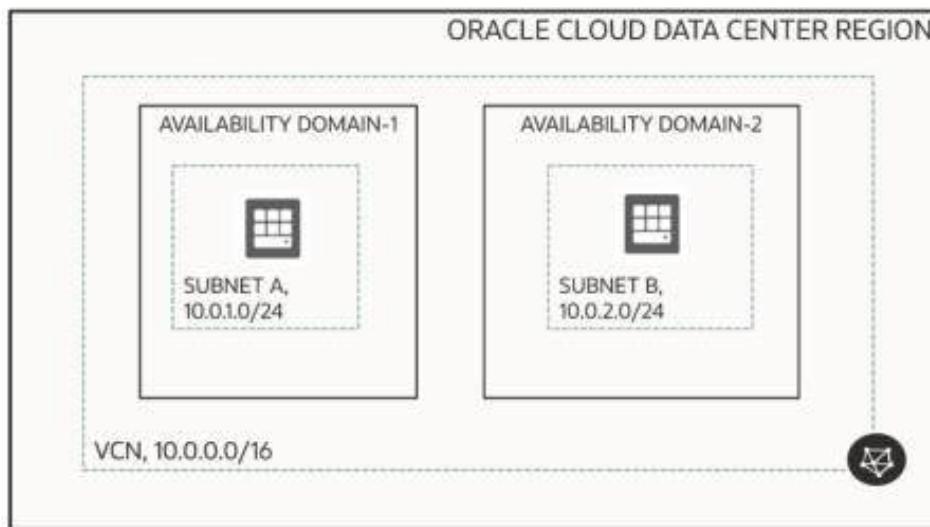
# Fault Domains



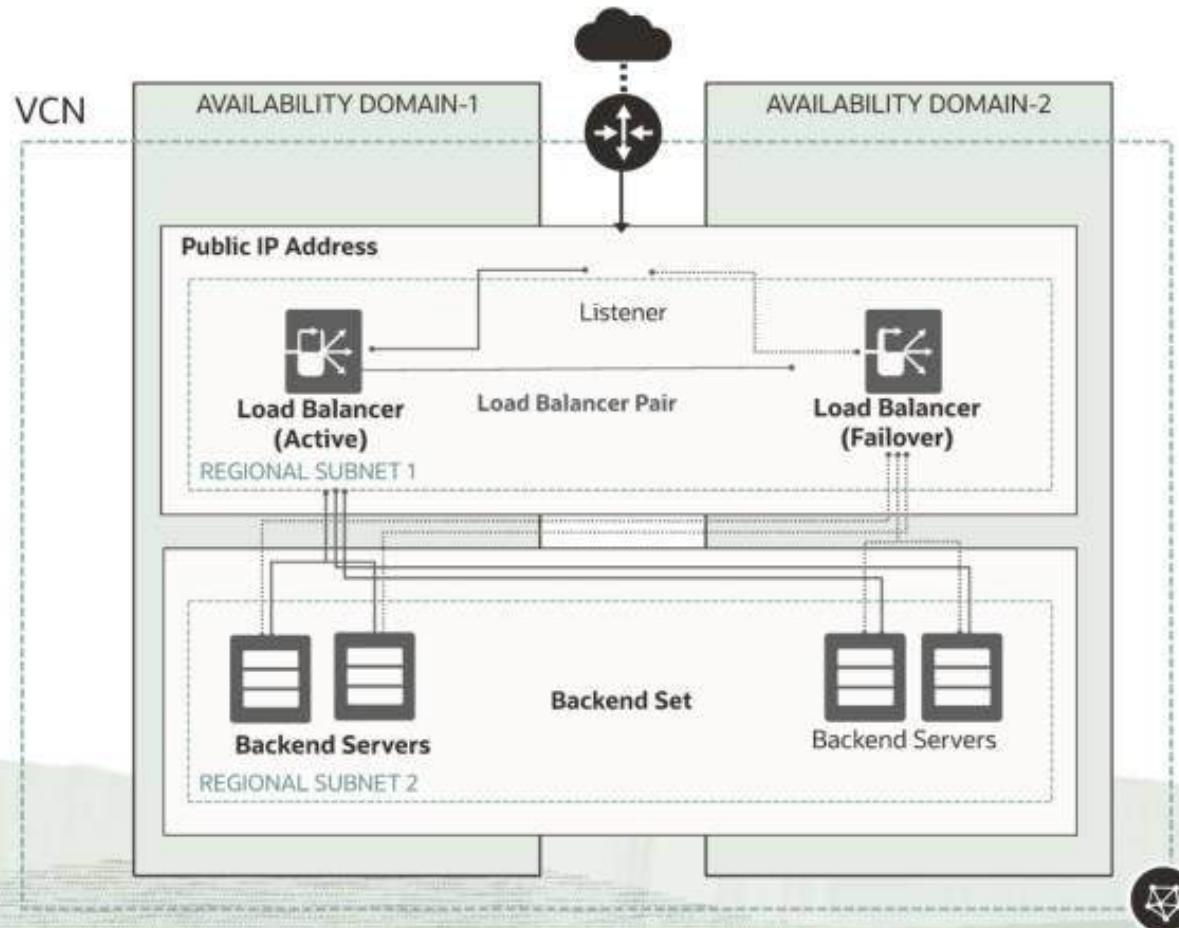
# Avoiding Single Points of Failure



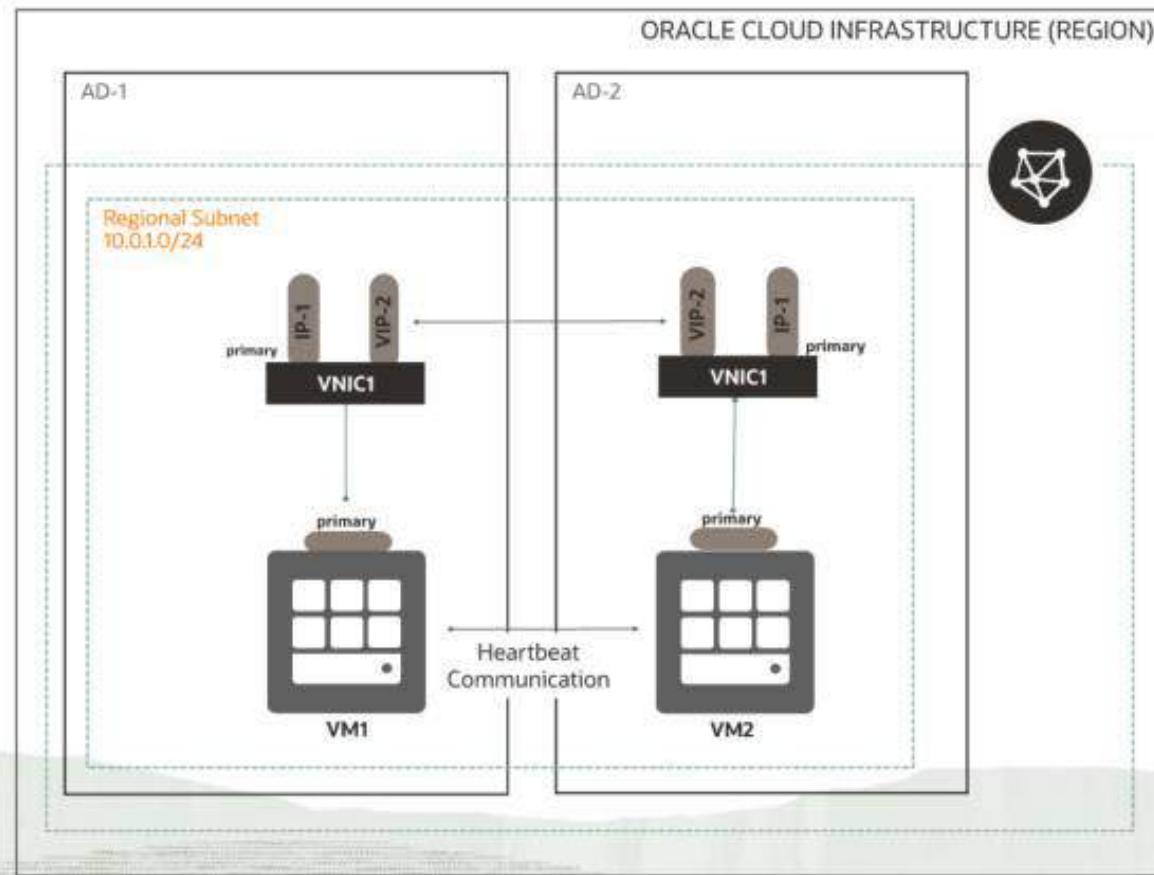
# Regional and AD-Specific Subnets



# Load Balancer



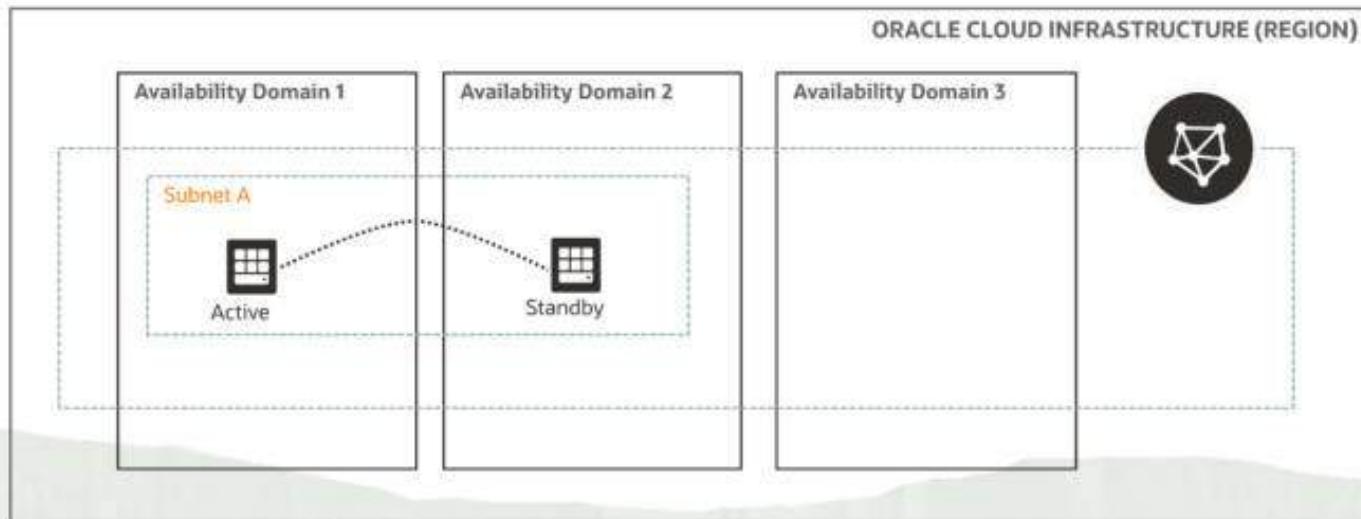
# Virtual IP



# Compute

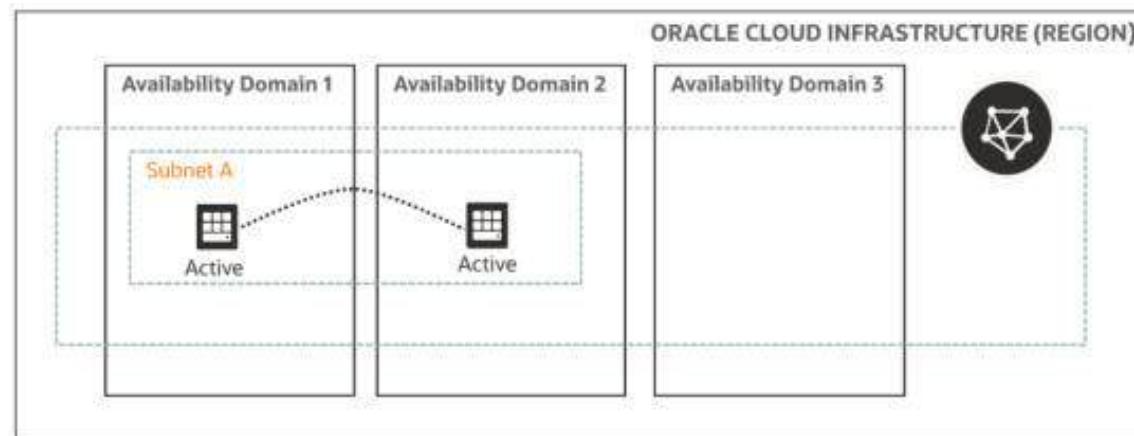
Depending on your system or application requirements, you can implement this architecture redundancy in either standby or active mode:

- **Standby mode:**

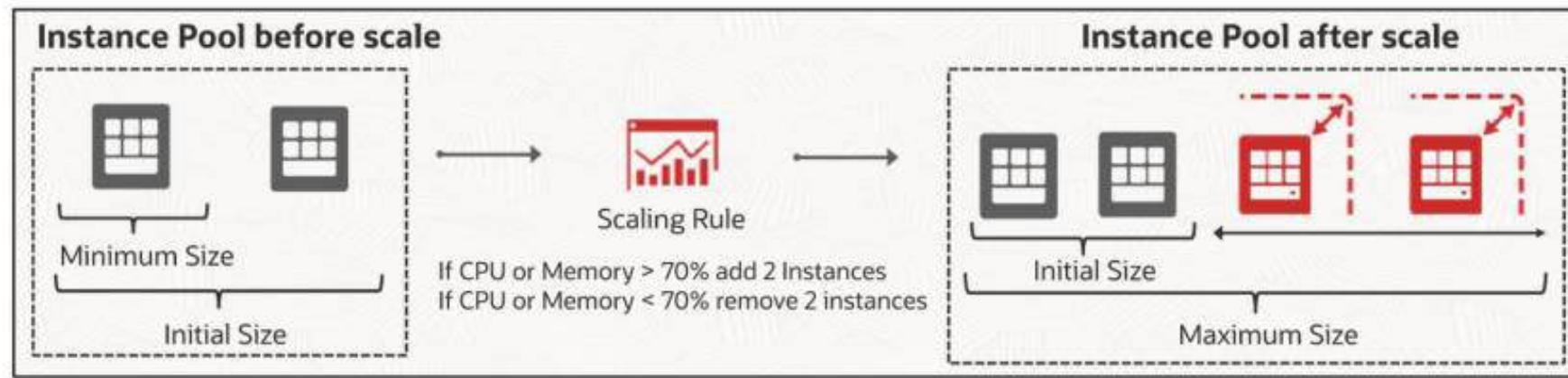


# Compute

- **Active/Active mode:**

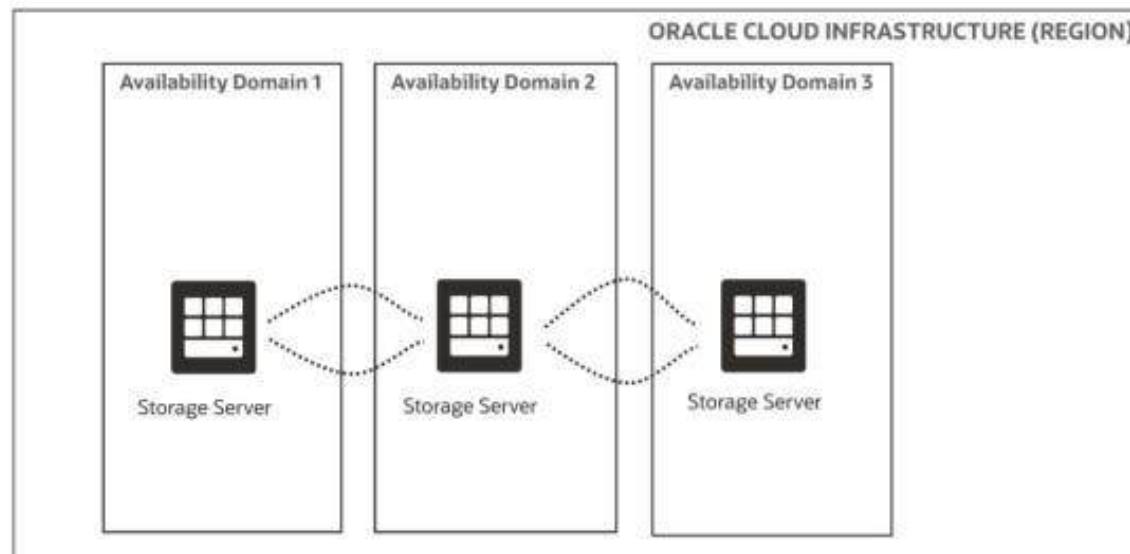


# Compute: Autoscaling



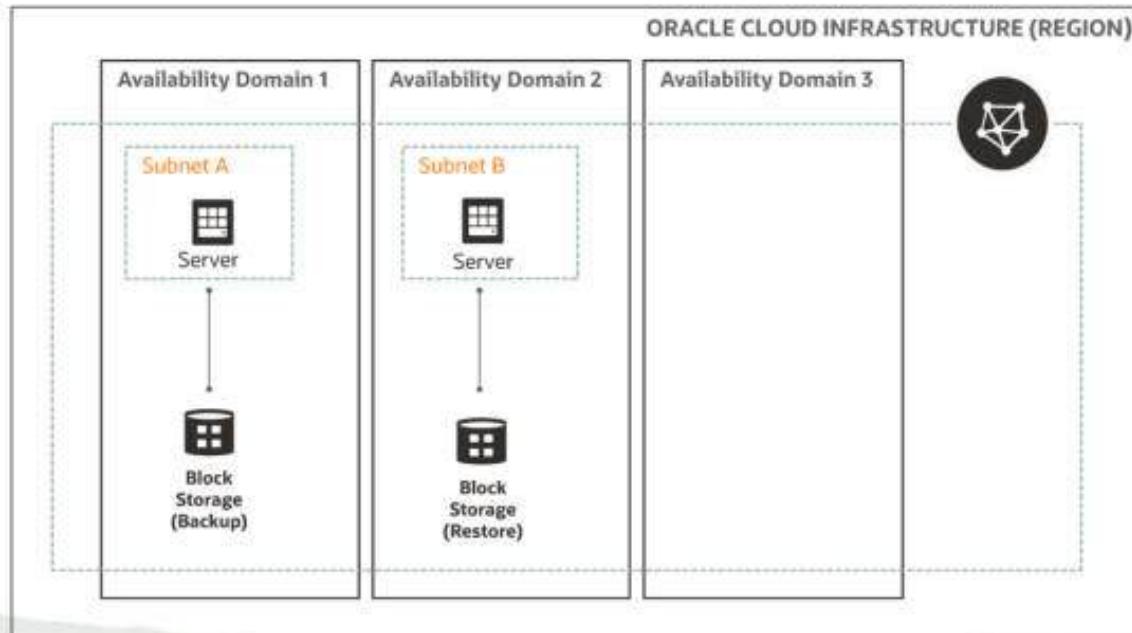
# Storage

## Object Storage:



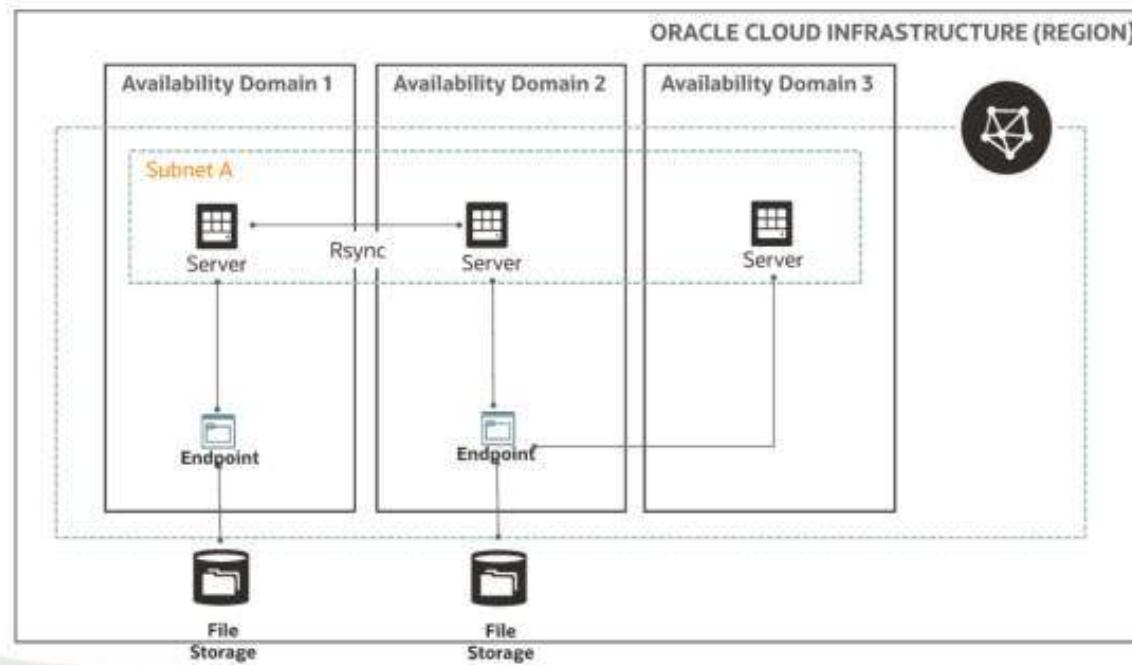
# Storage

## Block Volume:



# Storage

## File Storage:



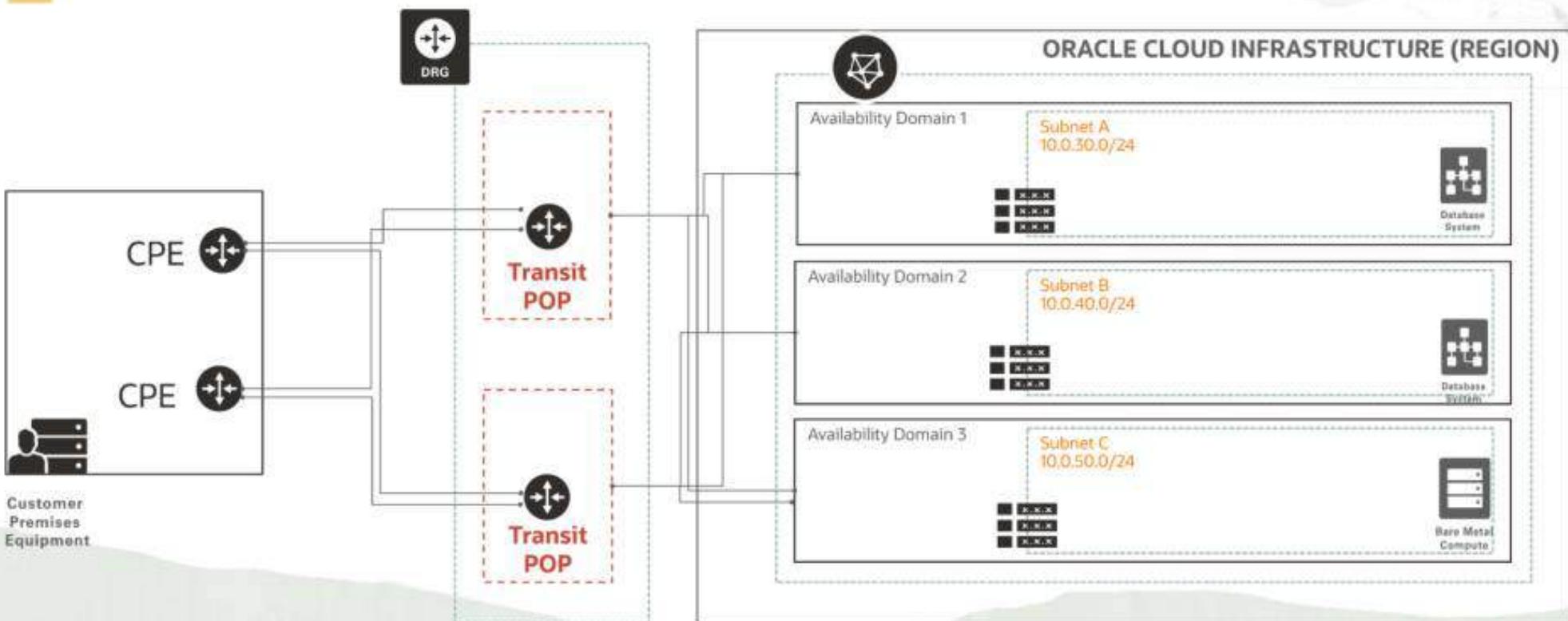


## High Availability for OCI: Connectivity

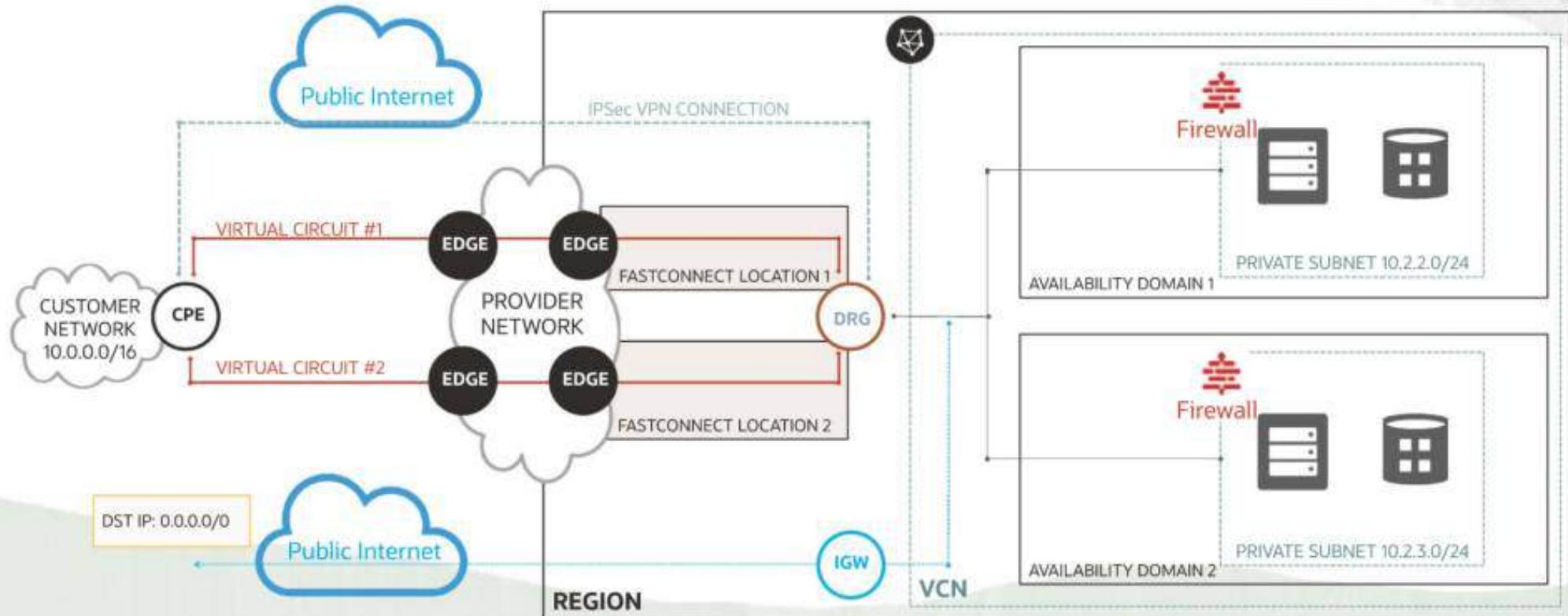


Highly available, fault-tolerant network connections are key to a well-architected system. You can choose to implement IPSec VPN connections to connect your data center to OCI or FastConnect, which provide higher bandwidth options and a more reliable and consistent networking experience compared to internet-based connections.

# IPSec VPN Redundancy Models (Multiple CPE)



# Redundant FastConnect



# Oracle Cloud Infrastructure Demo: Secrets

Oracle Cloud Infrastructure

# Demo: High Availability Workshop Part 02

## Oracle Cloud Infrastructure

# Disaster Recovery

# Disaster Recovery Terminology

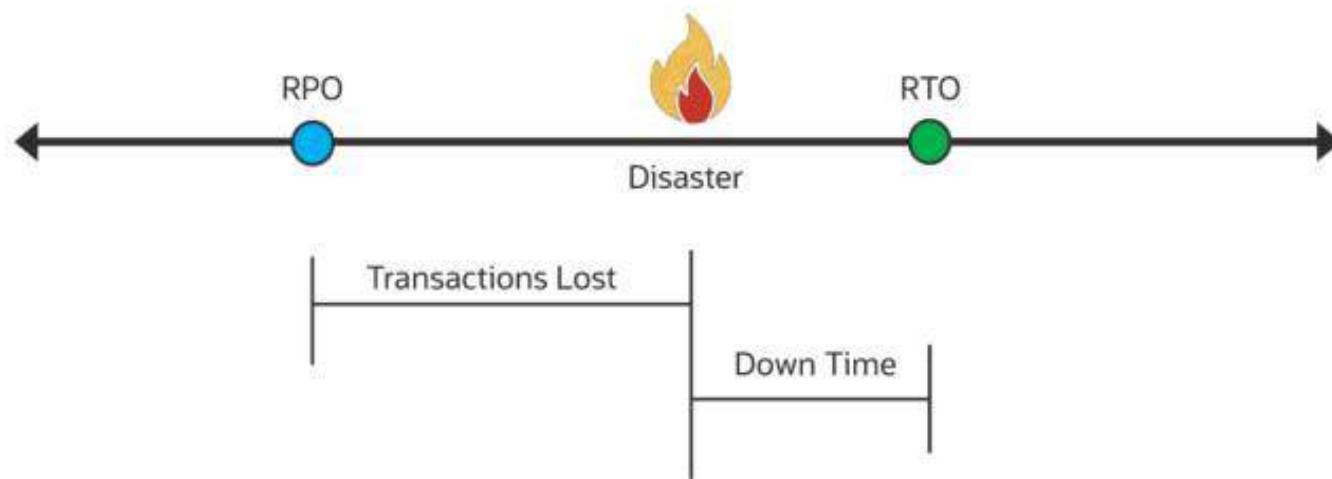


**Disaster recovery (DR)** involves a set of policies, tools, and procedures to enable the recovery or continuation of vital technology infrastructure and systems.

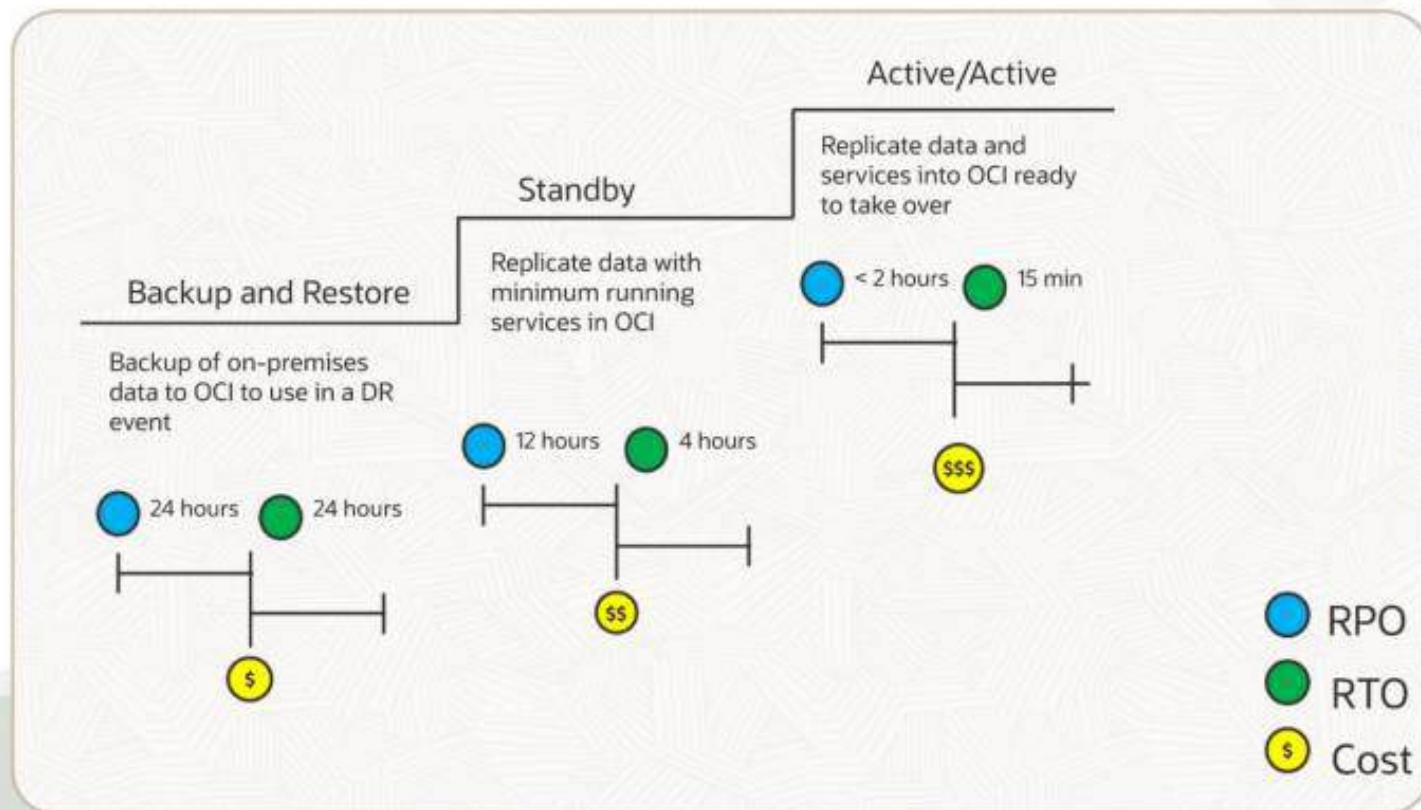
Disaster recovery should indicate the key metrics of recovery point objective (RPO) and recovery time objective (RTO).

In many cases, an organization may elect to use an outsourced disaster recovery provider to provide a stand-by site and systems rather than use their own remote facilities.

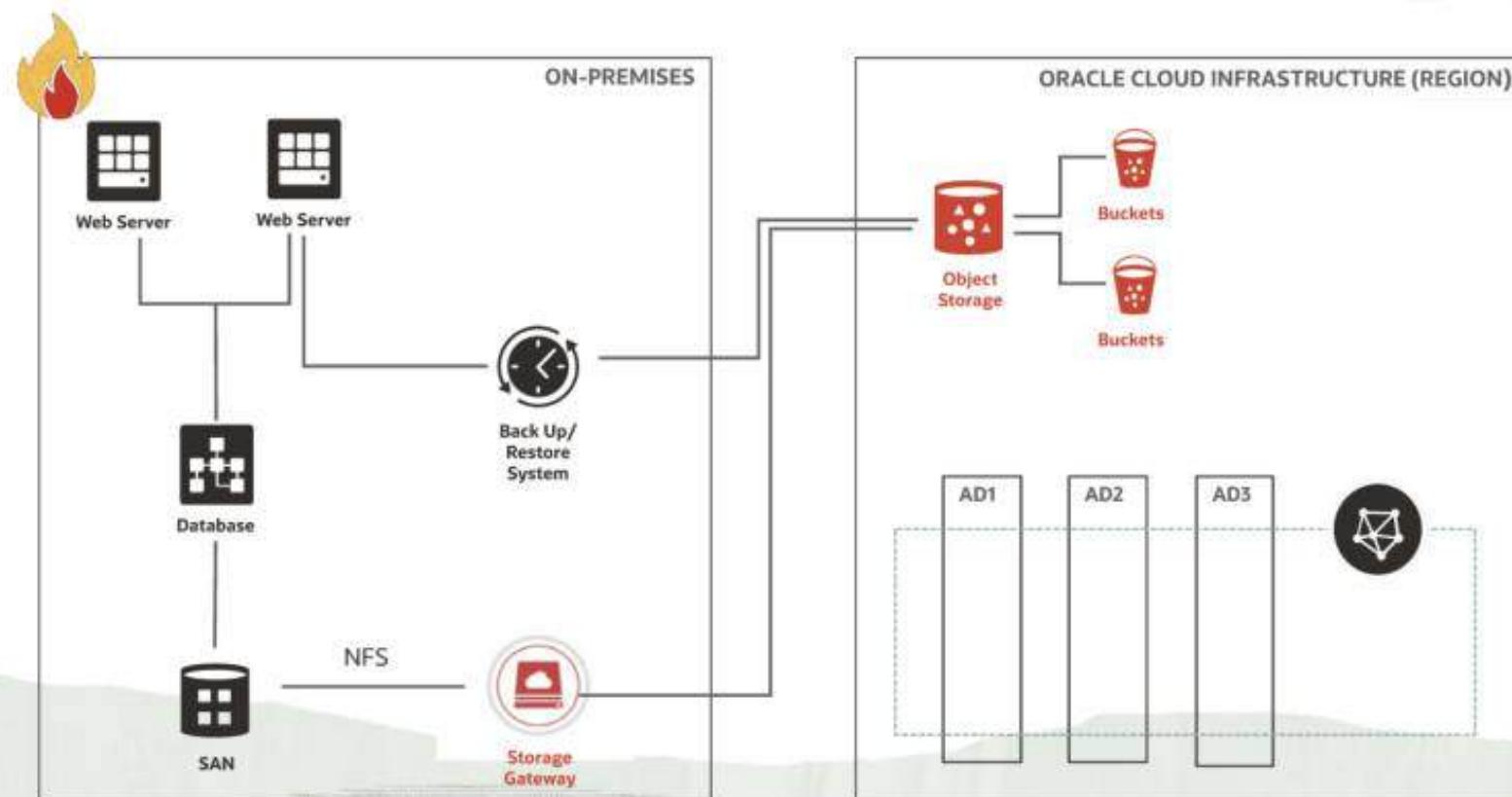
# Disaster Recovery RTO and RPO



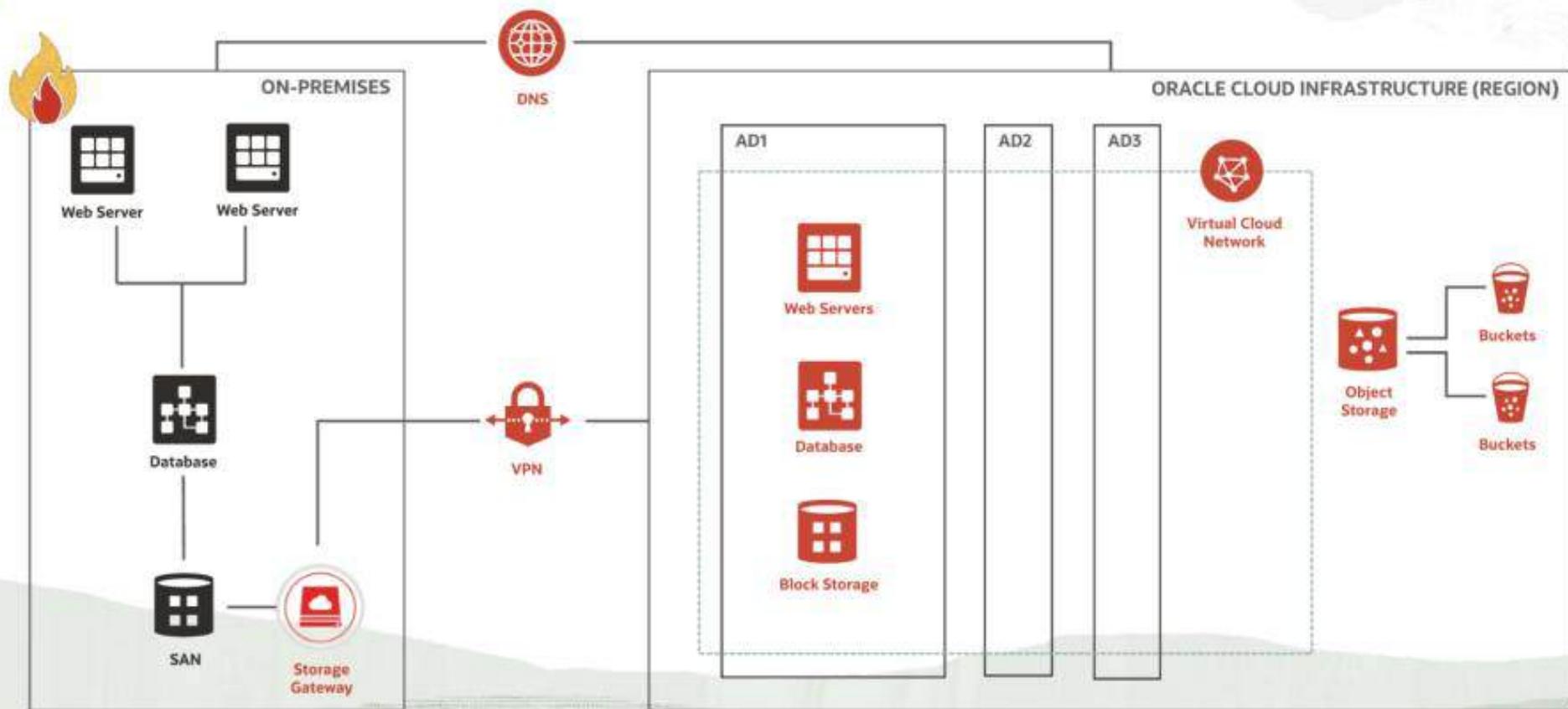
# Disaster Recovery Options



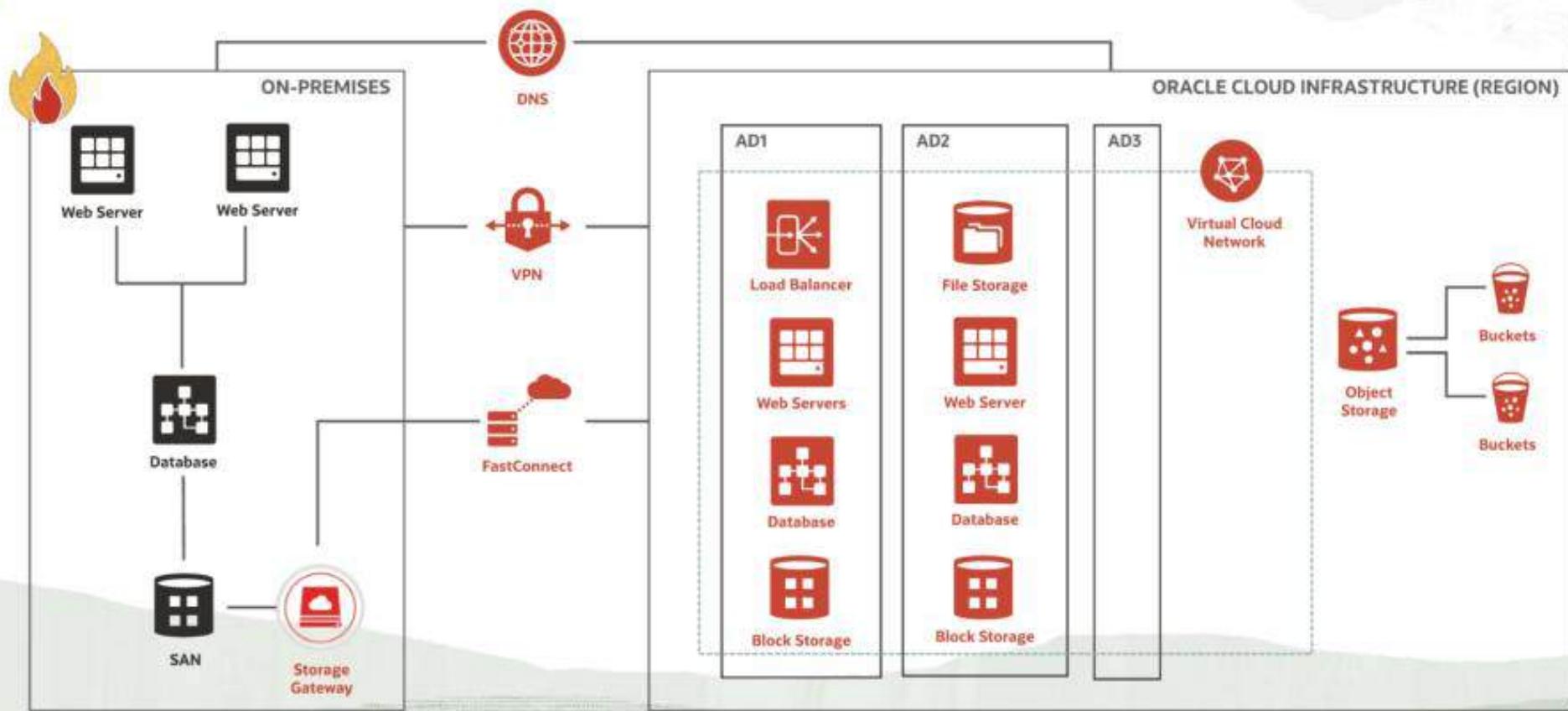
# Backup and Restore Architecture



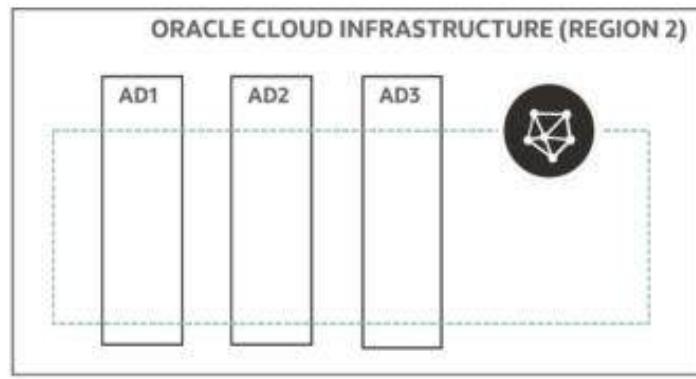
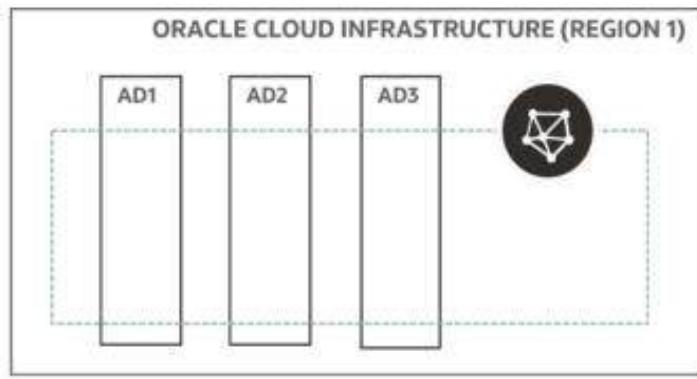
# Standby Architecture



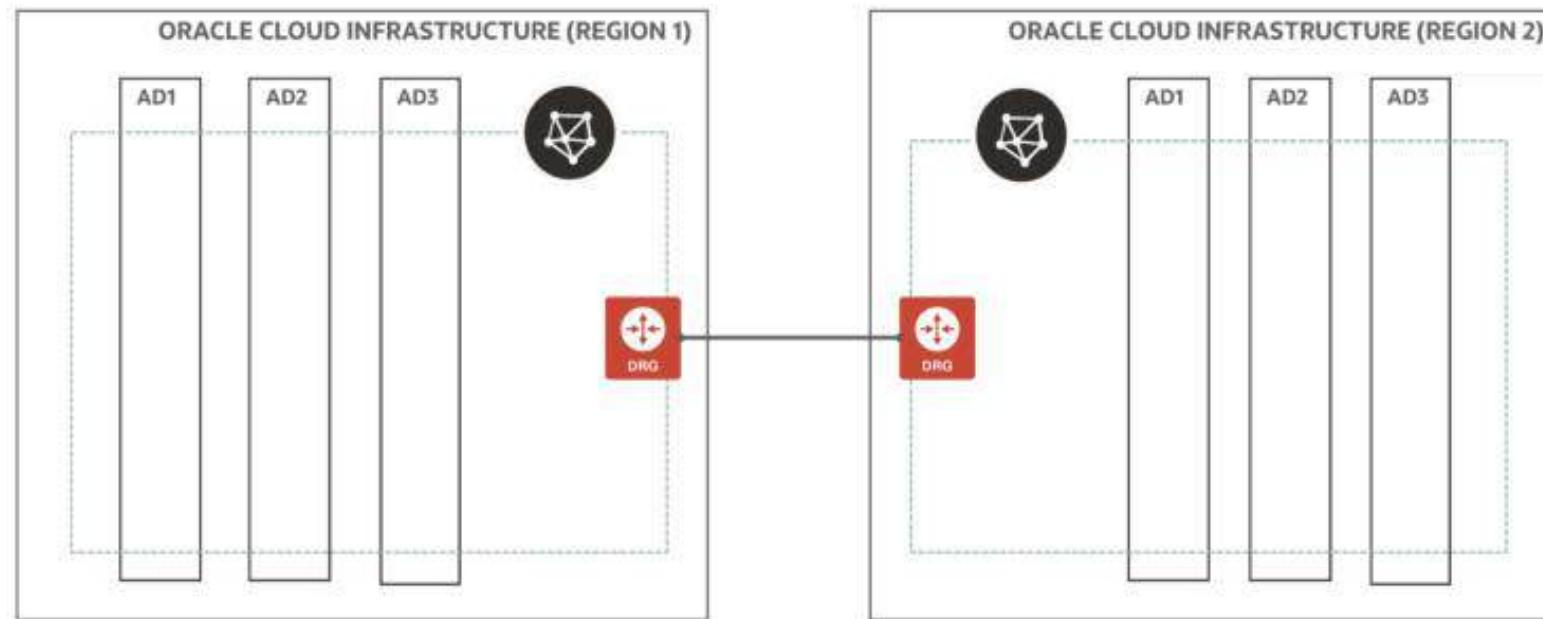
# Active/Active Architecture



# Disaster Recovery for OCI



# Disaster Recovery Using Multiple Regions





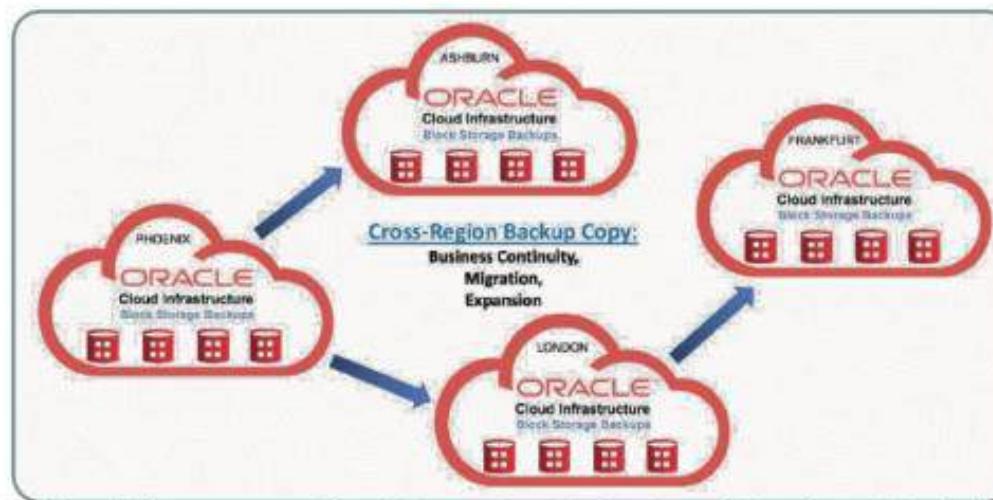
## Disaster Recovery Using Multiple Regions

### Cross-region block volume backup copy:

- By copying block volume backups to another region at regular intervals, it makes it easier to rebuild applications and data in the destination region if a region-wide disaster occurs in the source region.

### Migration and expansion:

- To easily migrate and expand your applications to another region





## Database Strategies for DR

### Active Data Guard

- Provides data protection and availability for Oracle Database in a simple and economical manner by maintaining an exact physical replica of the production copy at a remote location that is open read-only while replication is active

### GoldenGate

- Enables advanced logical replication that supports multi-master replication, hub and spoke deployment, and data transformation
- Provides customers flexible options to address the complete range of replication requirements, including heterogeneous hardware platforms

**Oracle Cloud Infrastructure**

# OCI Full Stack Disaster Recovery

## Overview



## Disaster Recovery Operational Challenges in OCI



### Complexity:

Recovering complex systems is tough; all parts must align to meet RTO and RPO targets.



### Testing:

Testing a DR plan is challenging as it must avoid production impact while ensuring smooth execution.



### Skills and Expertise:

Maintaining DR artifacts requires cloud expertise, which makes it difficult to find and retain skilled staff.



### Compliance:

Documenting DR activities across regions is vital for audits, requiring orchestration tools to provide detailed logs for compliance.



## How a typical DR runbook

Owner Team	Owner User	Plan details	Estimated Duration	Status
PM	Suraj/Greg	Start Runbook	5	Completed
PM	Suraj/Greg	<b>Preparation</b>	5	Completed
PM	Suraj/Greg	Engage all teams in a zoom call/slack	10	Completed
PM	Suraj/Greg	Prechecks for all teams	15	Completed
App Team	Rama	Check Application status	5	Completed
OS Team	Mahesh	Check App VM's replication status and healthchecks	10	Completed
Networking team	Santhosh	Check load balancer,DNS healthchecks	10	Completed
DBA	Sekar	Verify Data Guard status	10	Completed
PM	Suraj/Greg	Go/No.Go decision	5	Completed
		<b>Execution Switchover from Ashburn to Phoenix</b>		Completed
App Team	Rama	Shutdown Application servers	10	Completed
DBA	Sekar	Switchover DB	20	Completed
Networking team	Santhosh	Load Balancer changes	10	Completed
OS Team	Mahesh	Activate Block volume replicas,create VM and mount the volume	15	Completed
OS Team	Mahesh	Setup Reverse replication for the volumes	10	Completed
App Owner	Rama	Run app scripts	10	Completed
Networking team	Santhosh	Load Balancer and DNS changes	10	Completed
PM	All teams	Verify application status in new region	10	Completed
<b>Total Time</b>			<b>170</b>	



Customer Quote..."When a disaster happens,  
I do not have enough people to recover all  
the critical environments we support today."

# Full Stack DR orchestrates recovery with a single click

Tie unique recovery processes from many services & apps into a single workflow



## Build DR runbooks in minutes

Define which OCI compute, storage, and databases belong to an application stack, then build fully functional DR Plans in minutes

## Customize DR runbooks

Tailor DR Plans to recover Oracle & non-Oracle applications along with anything else unique to your environment

## Validate DR before its needed

Fully automated non-intrusive, non-disruptive DR drills using a single button are built into the service

## DR at Scale

Operators can recover many critical business systems at the same time without knowing anything about the steps needed to recover

## Capitalize on existing effort

Automate the recovery steps for business systems already deployed for DR without redesigning or reinstalling your application stack

## Use any DR topology

Full Stack DR does not limit or make you conform to a certain way of deploying or recovering your business system

\*Coming soon - H1 CY25

# Recovery made easy for many business systems

## Single pane of glass



Full Stack DR normalizes the way DR operations are executed and monitored for vastly different business systems using a single pane of glass

## DR at Scale



Full Stack DR is designed to handle DR workflows at scale without involving a cadre of technical experts when the time comes to recover many systems at the same time

## Simple execution



Any authorized user can execute and monitor recoveries without needing to understand anything about the complex processes each business system requires

## Validate DR before its needed



Validate DR readiness of business systems before a recovery or transition to another OCI region needs to happen using built-in pre-checks  
  
Use DR drills to perform a complete dry run of a failover with a single click without impacting production

# Capitalize on your existing effort

Keep current DR processes



No need to change anything about the way a business system is already deployed across two OCI regions for DR

Keep existing DR automation



No need to discard any existing scripted automation or Oracle functions that are used right now to recover systems

Design new DR processes



Design DR for a new business system using any DR deployment architecture that fits the need

# Flexible, highly scalable, highly extensible and customizable

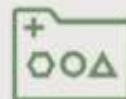
## Use any DR topology



Use OCI to deploy business systems using any deployment architecture:

- Cold
- VM failover
- Pilot Light
- Warm standby
- Hot standby
- Active/active

## Built-in modules reduce effort



Built-in intelligent modules generate custom DR plans prepopulated with the right steps to recover in the right order

Easily add, change and reshape DR plans to fit your unique requirements

## Fast deployment using APIs



Quickly create DR protection groups and DR plans and deploy resources using REST APIs, CLI, SDK, Terraform, and Resource Manager

## Serverless architecture



Full Stack DR does not require specialized snapshot storage, conversion servers, image servers, snapshot servers, or management servers of any kind

# Recovery point and recovery time objectives



## Recovery Point (RPO)

This goal is determined by you and dictates the frequency of backup and/or replication approaches you configure in OCI

Weeks Days Hours Minutes Seconds Zero/Near Zero Seconds Minutes Hours Days Weeks



- RPO for Oracle databases is dictated by Data Guard
- RPO for compute is dictated by OCI cross-region replication



## Recovery Time (RTO)

This goal is determined by you and depends entirely on the way you design your DR strategy and deploy your application stack for DR

Weeks Days Hours Minutes Seconds Zero/Near Zero Seconds Minutes Hours Days Weeks

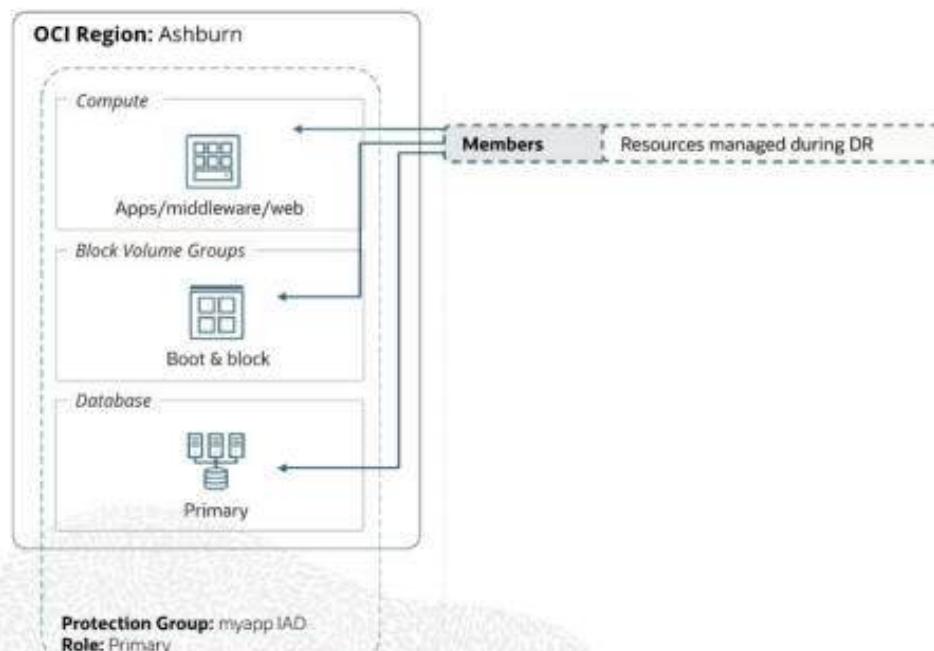
- RTO for Oracle databases is predicated on how long Data Guard takes to recover the databases
- RTO for compute is predicated on how long it takes OCI to start virtual machines
- RTO for applications is predicated on how long it takes your application to start

Recovery point and time are a function of standard OCI services

- Full Stack DR does not have any settings for recovery point
- Full Stack DR does not have any settings for recovery time

# Core Concepts

**DR Protection Groups:** OCI resources that define an application are organized into a group to ensure they are recovered together.



**Members:** OCI resources that can be added to the DR protection groups. Supported OCI resources are

1. Oracle Database PaaS
  - Oracle Autonomous Database Serverless (ADB-S)
  - Oracle Autonomous Database Dedicated Infrastructure(ADB-D)
  - Oracle Autonomous Database on Exadata Cloud@Customer (ADB-C@C)
  - Oracle Base Database Service (BaseDB)
  - Oracle Exadata Database Service on Dedicated Infrastructure (ExaDB-D)
  - Oracle Exadata Database Service on Cloud@Customer (ExaC@C)
  - Oracle Exadata Database Service on Exascale Infrastructure (ExaDB-XS)
2. Compute IaaS
  - Virtual machine
  - Dedicated Virtual Host (DVH)
3. Storage IaaS
  - Block storage service (BSS)
  - File Storage Service (FSS)
  - Object Storage Bucket (OSS)
4. Load Balancer IaaS
  - Load balancer
  - Network load balancer
5. Oracle Kubernetes Engine (OKE)

# Core Concepts

**DR Plan types:** DR runbooks or workflows are managed using four different plan types.

Plan Type	Action		
Failover	Recover at standby	←	Catastrophic unplanned event Primary is inaccessible
Switchover	Shutdown at primary, then transition to standby	←	
Start DR drill	<b>Perform a complete dry run of a failover for validation</b>	←	Planned event Both primary and standby accessible
Stop DR drill	Tear down workload at standby	←	

# Core Concepts

**DR Plan types:** DR runbooks or workflows are managed using four different plan types.

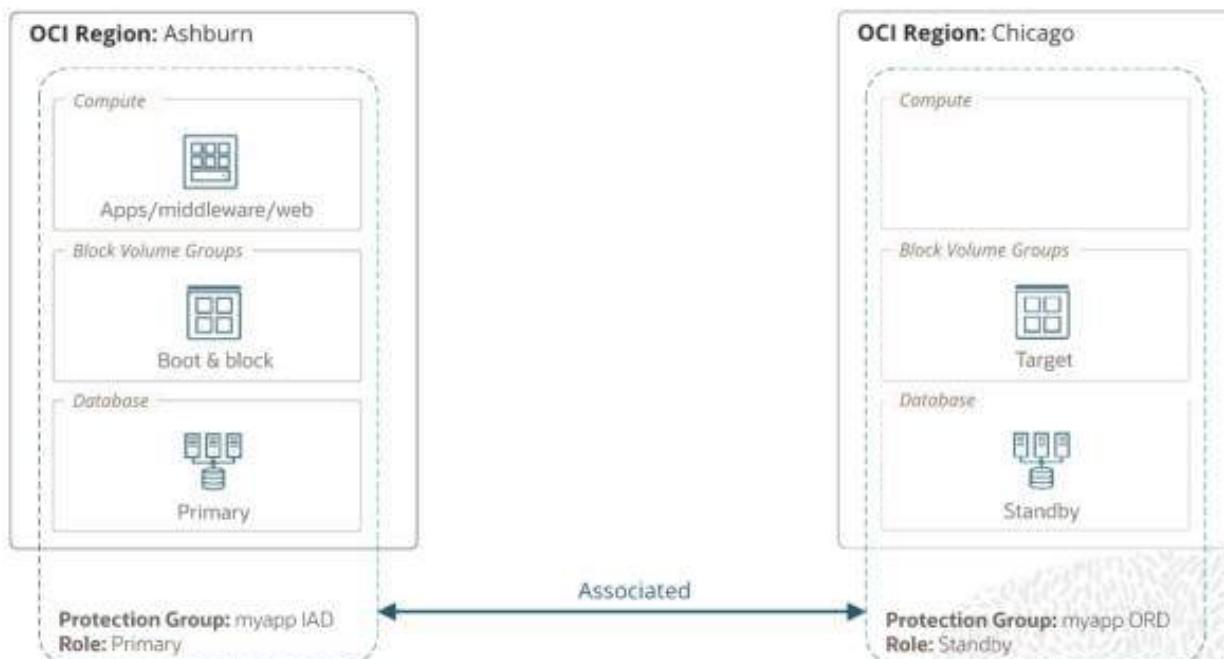
Plan Type	Action		
Failover	Recover at standby	Catastrophic unplanned event	Primary is Inaccessible
Switchover	Shutdown at primary, then transition to standby	Planned event	
Start DR drill	<b>Perform a complete dry run of a failover for validation</b>		Both primary and standby accessible
Stop DR drill	Tear down workload at standby		

**DR Plans:** Recovery steps to be executed by Full Stack Disaster Recovery on a protection group.

Name	Type	Enabled/Disabled		
prechecks	Built-in precheck	Enabled	Prechecks	Validate DR plan steps
stop compute	Built-in	Enabled		
launch compute	Built-in	Enabled		
switchover database	Built-in	Enabled	Built-in plan groups	Autogenerated tasks
launch application	User-Defined	Enabled	User-defined plan groups	Custom scripts & functions

# Core Concepts

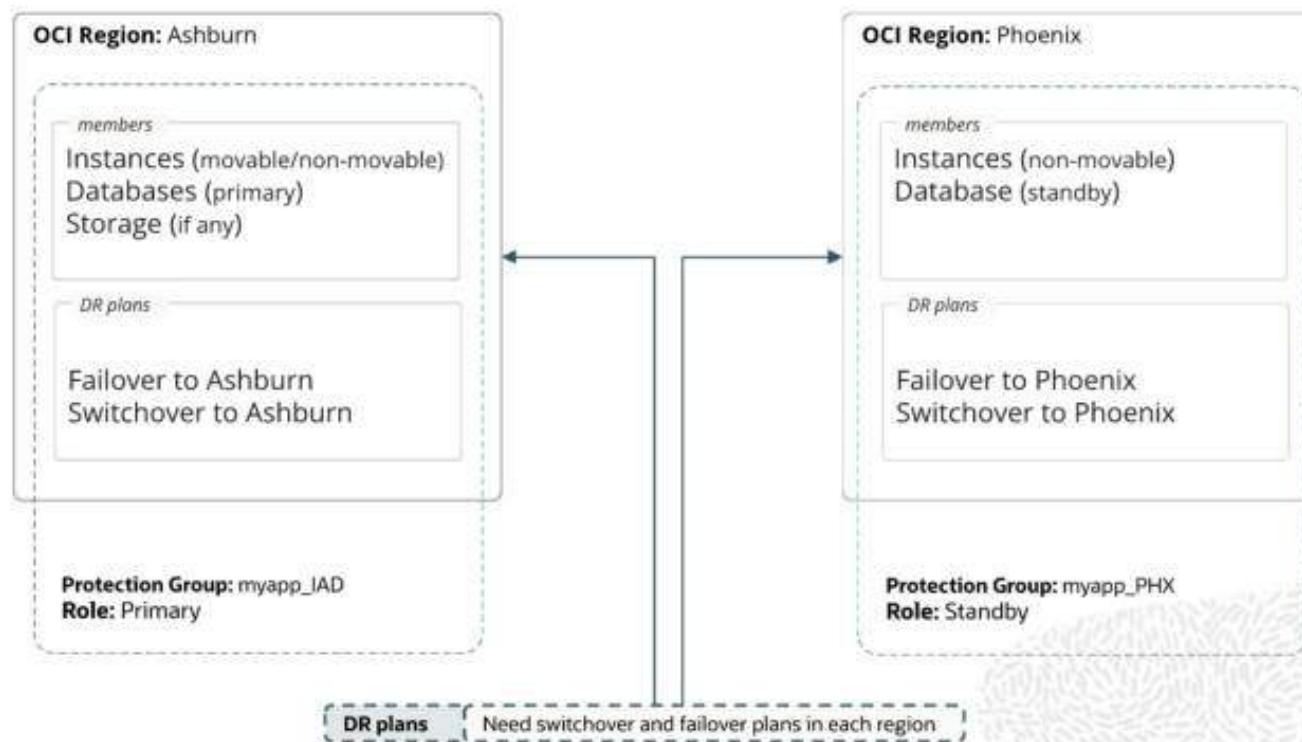
**Association:** DR Protection Groups in FSDR must be associated (paired) in a Primary and Standby relationship before they can be used to implement DR services.



- Associations simply define which two DR protection groups are peers of each other
- Associations also define the “primary” & “standby” peer relationship between two OCI regions

# Core Concepts

**DR Plans:** DR plans are the DR runbooks or workflows for Full Stack DR



DR Protection Groups can have many DR plans

- DR plans are always created, modified & executed from standby DR Protection Group
- DR plans are pre-populated with appropriate recovery tasks based on members
- A minimum of two DR plans are needed in each DR Protection Group

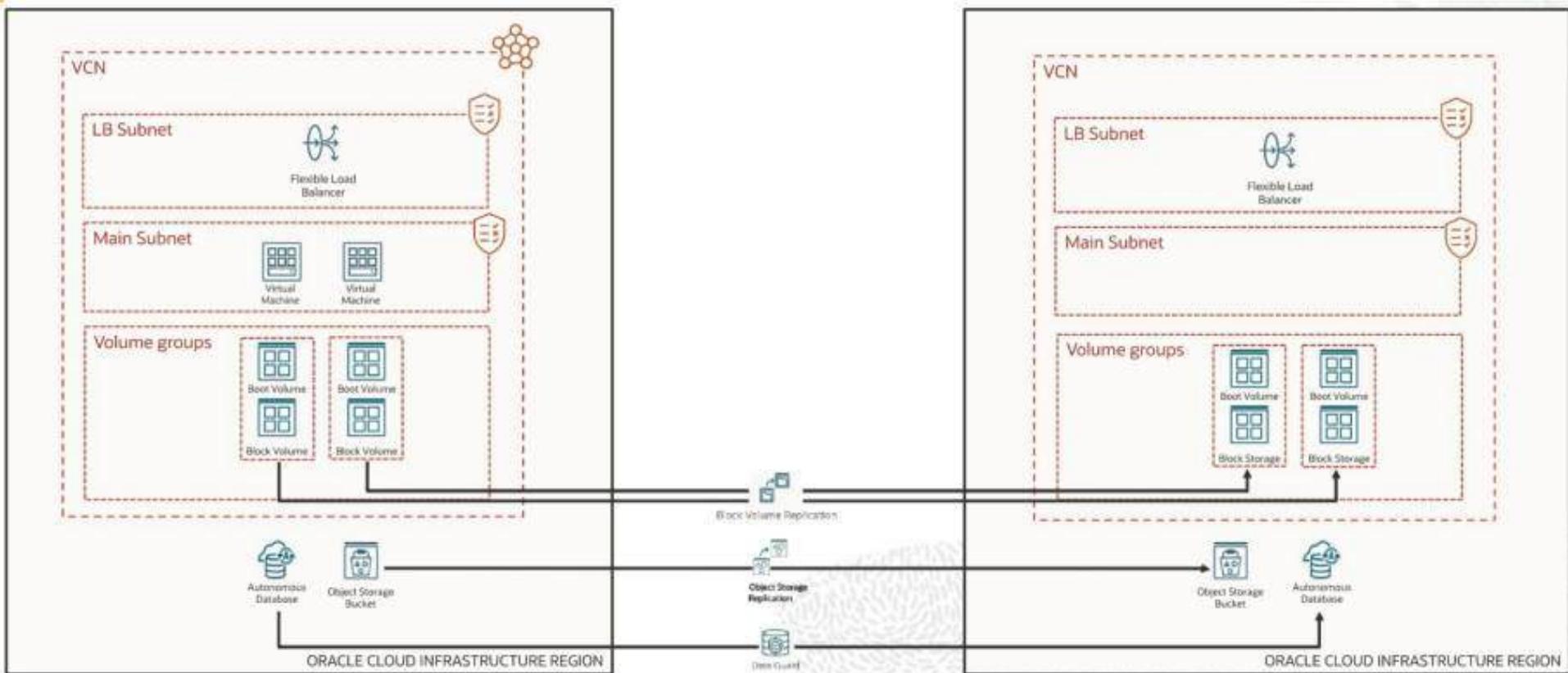
# Full Stack Disaster Recovery

## FSDR components and concepts

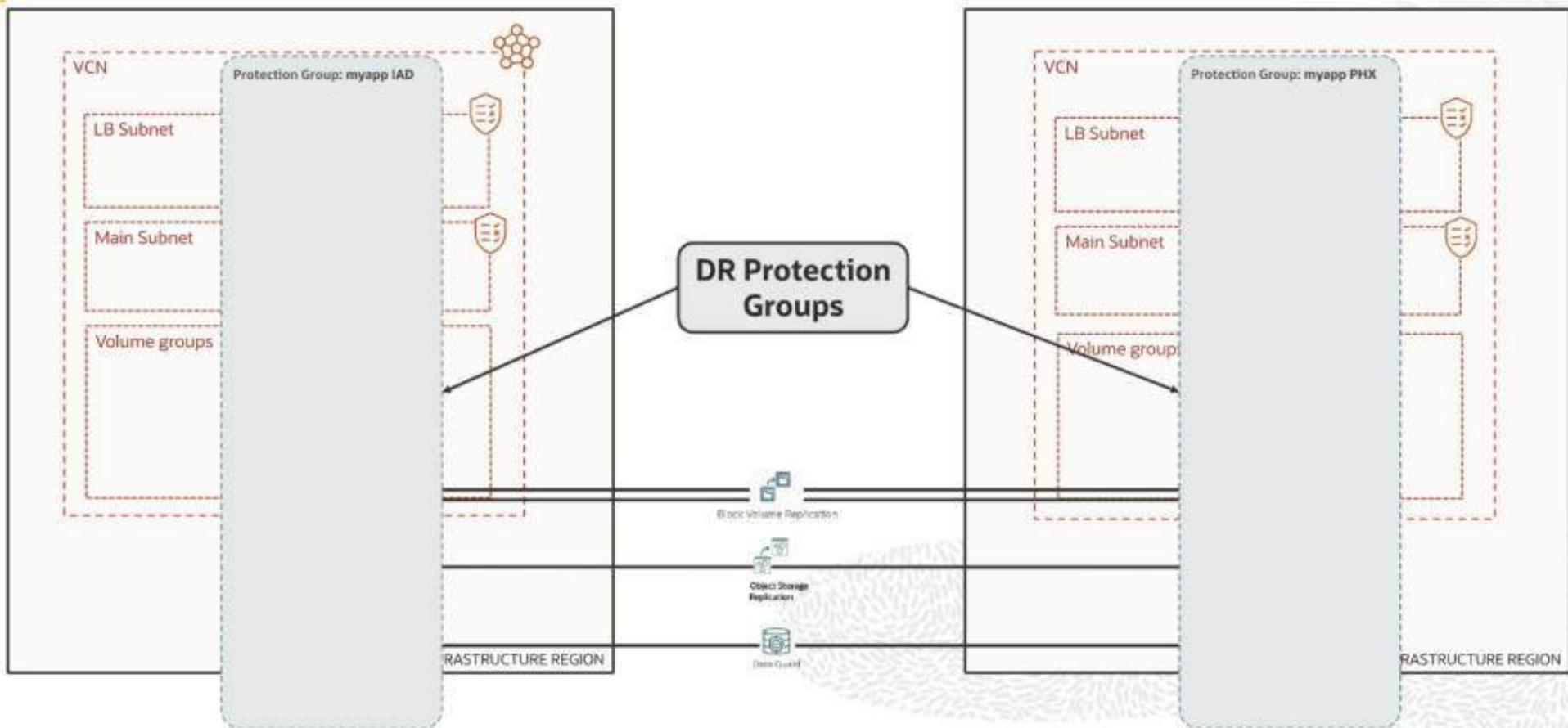


# Sample Scenario

Movable  
Compute

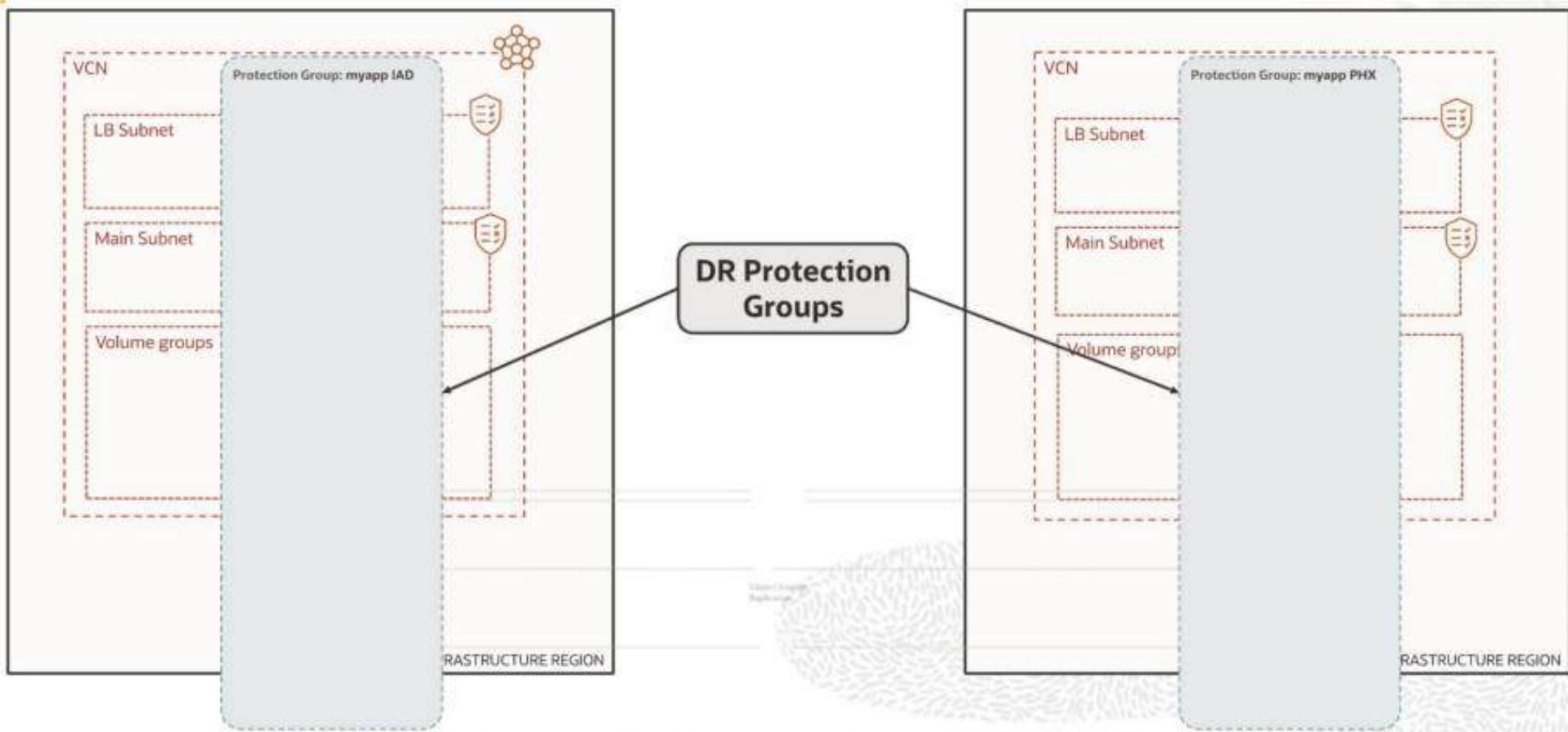


# DR Protection Group

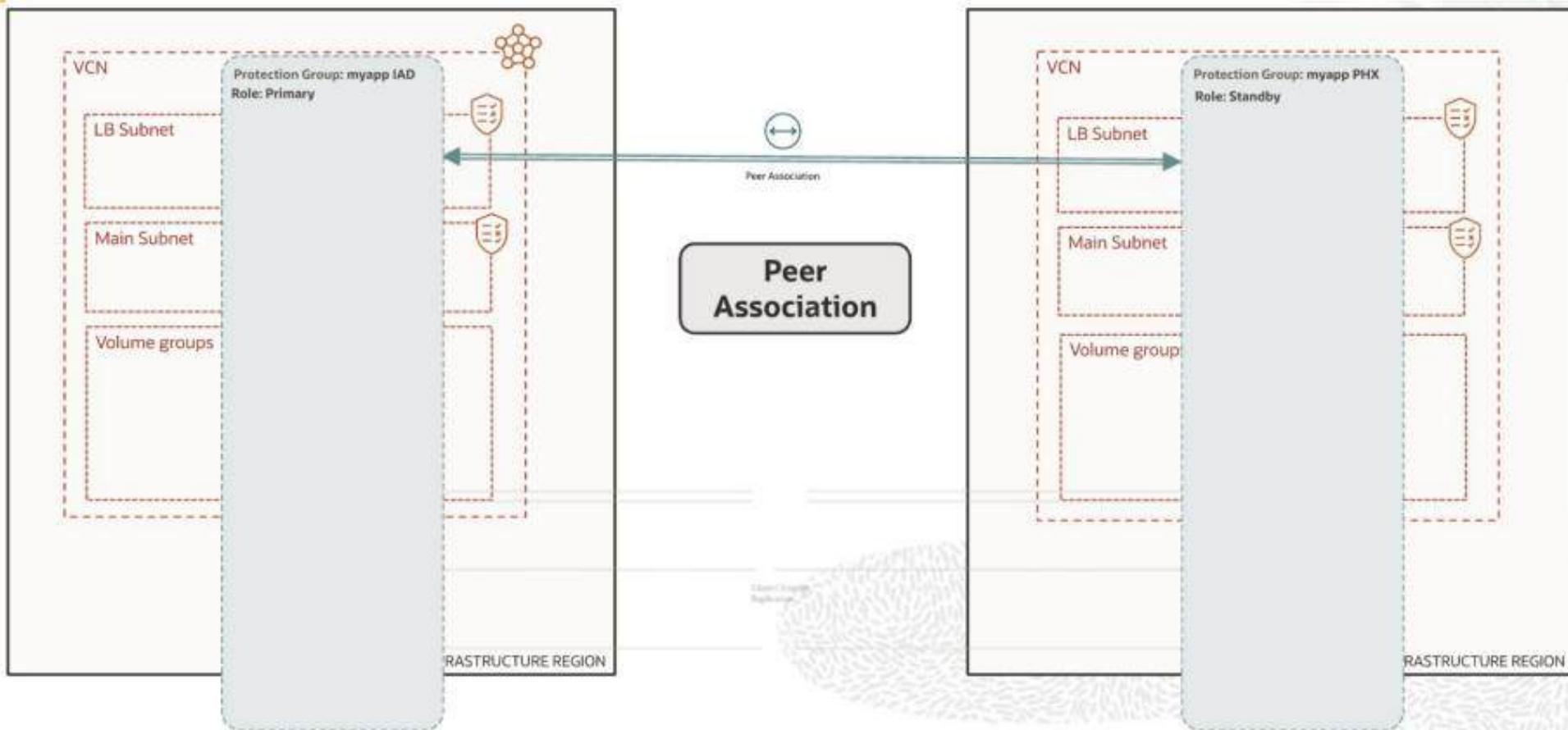


# DR Protection Group

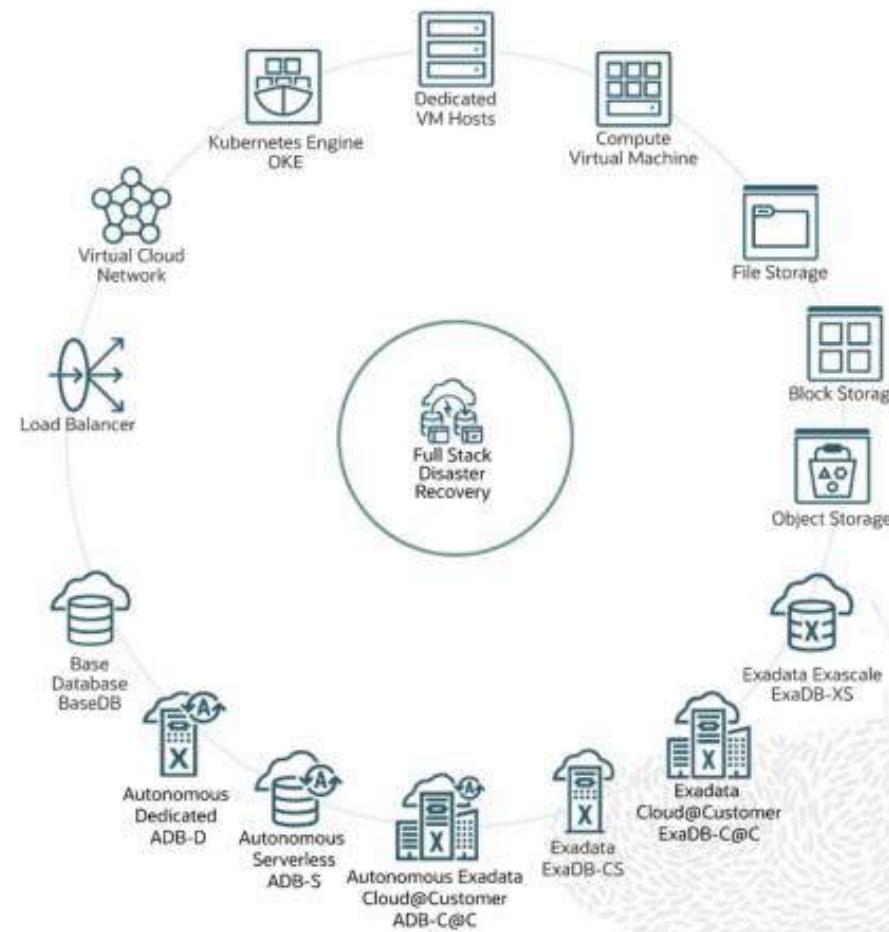
Movable  
Compute



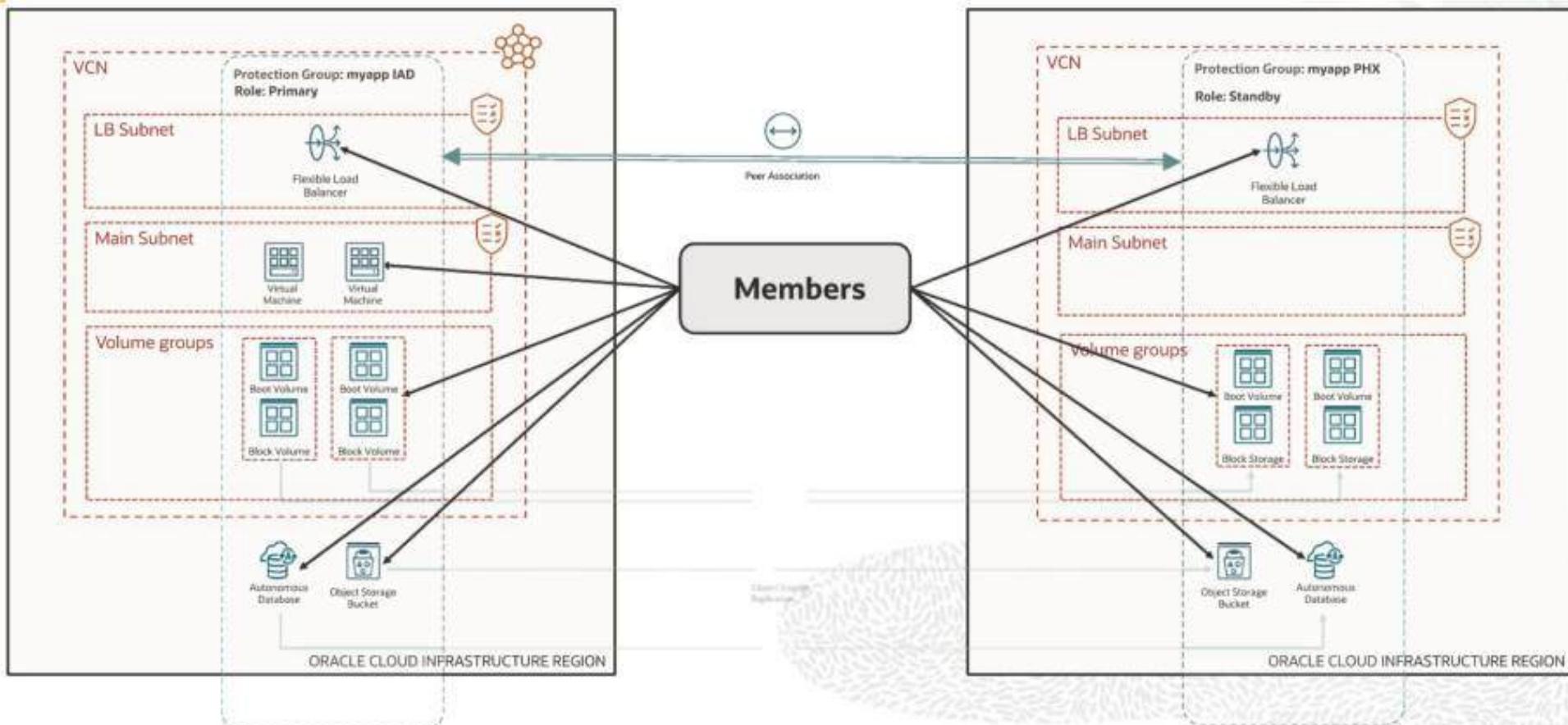
# Peer Association



# Members

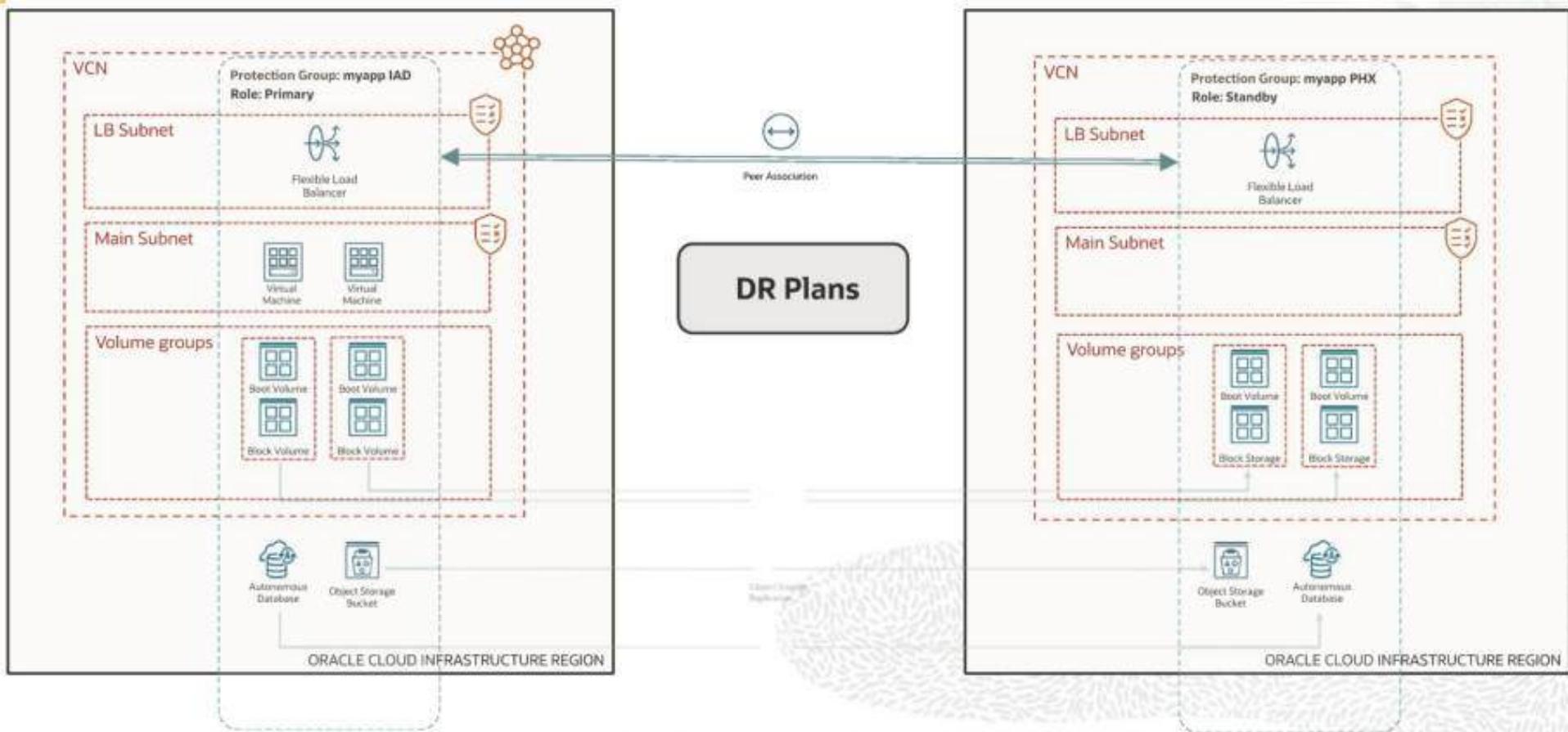


# Members

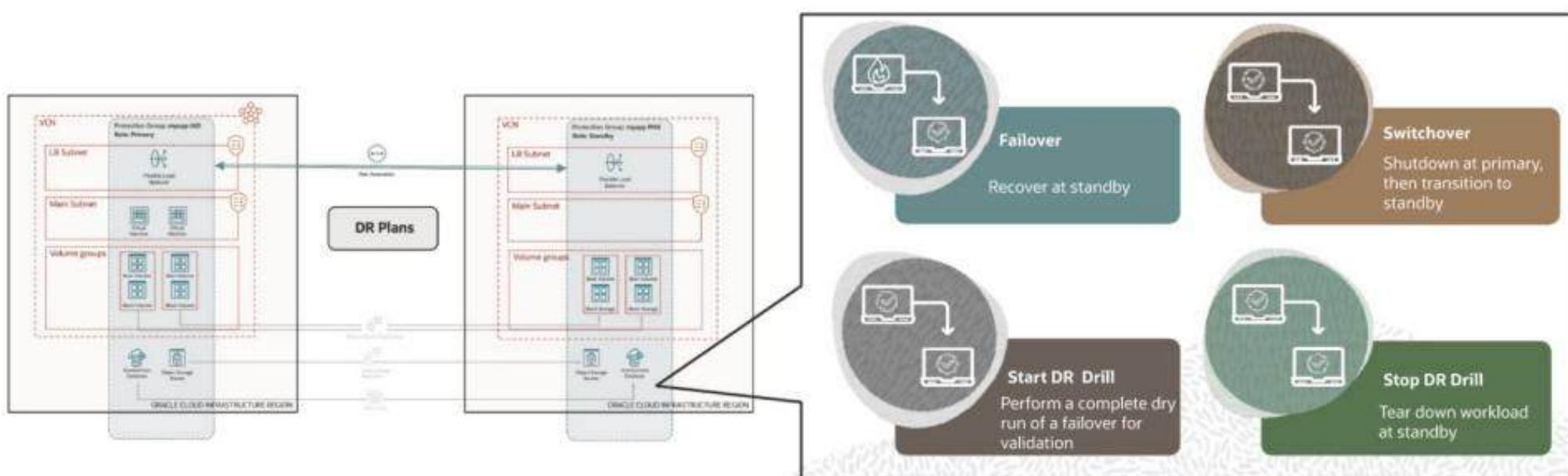


# DR Plans

Movable Compute



# DR Plans



# DR Groups

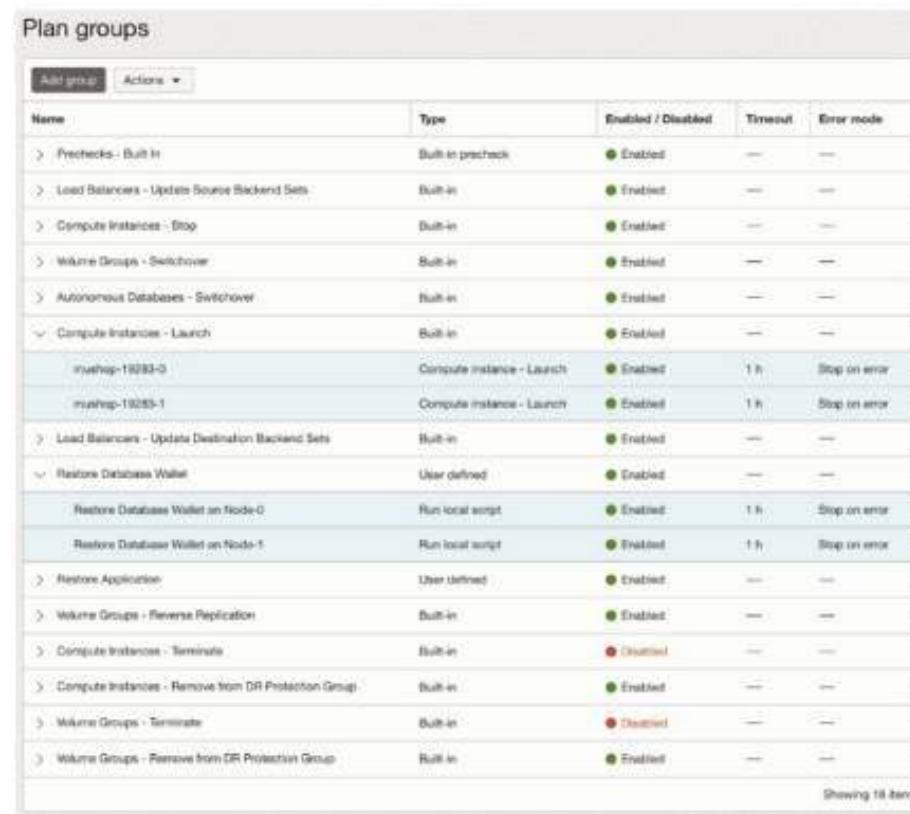
## Typical DR runbook

Owner Team	Owner User	Plan details	Estimated Duration	Status
PM	Suraj/Greg	Start Runbook	5	Completed
PM	Suraj/Greg	Preparation	5	Completed
PM	Suraj/Greg	Engage all teams in a zoom call/slack	10	Completed
PM	Suraj/Greg	Prechecks for all teams	15	Completed
App Team	Rama	Check Application status	5	Completed
OS Team	Mahesh	Check App VM's replication status and healthchecks	10	Completed
Networking team	Santhosh	Check load balancer,DNS healthchecks	10	Completed
DBA	Sekar	Verify Data Guard status	10	Completed
PM	Suraj/Greg	Go/No.Go decision	5	Completed
		Execution Switchover from Ashburn to Phoenix		Completed
App Team	Rama	Shutdown Application servers	10	Completed
DBA	Sekar	Switchover DB	20	Completed
Networking team	Santhosh	Load Balancer changes	10	Completed
OS Team	Mahesh	Activate Block volume replicas,create VM and mount the volume	15	Completed
OS Team	Mahesh	Setup Reverse replication for the volumes	10	Completed
App Owner	Rama	Run app scripts	10	Completed
Networking team	Santhosh	Load Balancer and DNS changes	10	Completed
PM	All teams	Verify application status in new region	10	Completed
Total Time			170	

# DR Groups

Name	Type	Enabled/Disabled		
prechecks	Built-in precheck	Enabled	Prechecks	Validate DR plan steps
stop compute	Built-in	Enabled		
launch compute	Built-in	Enabled	Built-in plan groups	Autogenerated tasks
switchover database	Built-in	Enabled		
launch application	User-Defined	Enabled	User-defined plan groups	Custom scripts & functions

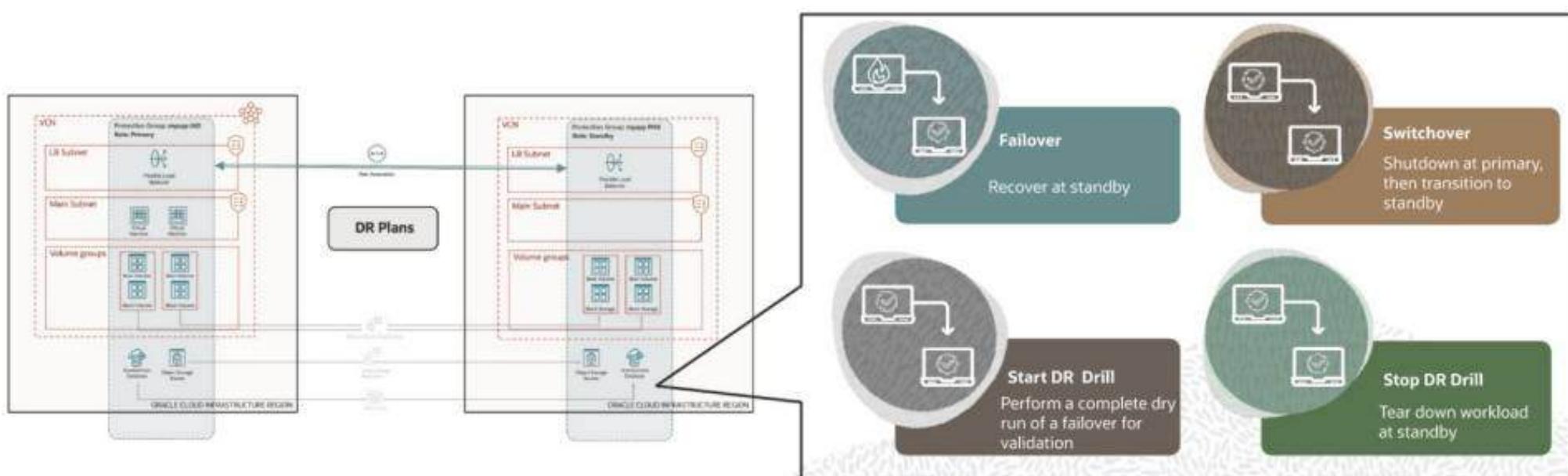
# DR Plan Groups



The screenshot shows a list of DR Plan Groups in a table format. The columns are: Name, Type, Enabled / Disabled, Timeout, and Error mode. The table contains 18 rows, each representing a different DR plan group. The groups include Prechecks, Load Balancers, Compute Instances, Volume Groups, Autonomous Databases, Compute Instances (Launch), Load Balancers (Update Destination Backend Sets), Restore Database Wallet, Run local script, Restore Application, Volume Groups (Reverse Replication), Compute Instances (Terminate), Compute Instances (Remove from DR Protection Group), Volume Groups (Terminate), and Volume Groups (Remove from DR Protection Group). Most groups are enabled, while some like Compute Instances (Terminate) and Volume Groups (Terminate) are disabled.

Name	Type	Enabled / Disabled	Timeout	Error mode
> Prechecks - Built-in	Built-in precheck	● Enabled	—	—
> Load Balancers - Update Source Backend Sets	Built-in	● Enabled	—	—
> Compute Instances - Stop	Built-in	● Enabled	—	—
> Volume Groups - Synchronization	Built-in	● Enabled	—	—
> Autonomous Databases - Synchronization	Built-in	● Enabled	—	—
✓ Compute Instances - Launch	Built-in	● Enabled	—	—
muship-19283-0	Compute instance - Launch	● Enabled	1 h	Stop on error
muship-19283-1	Compute instance - Launch	● Enabled	1 h	Stop on error
> Load Balancers - Update Destination Backend Sets	Built-in	● Enabled	—	—
✓ Restore Database Wallet	User defined	● Enabled	—	—
Restore Database Wallet on Node-0	Run local script	● Enabled	1 h	Stop on error
Restore Database Wallet on Node-1	Run local script	● Enabled	1 h	Stop on error
> Restore Application	User defined	● Enabled	—	—
> Volume Groups - Reverse Replication	Built-in	● Enabled	—	—
> Compute Instances - Terminate	Built-in	● Disabled	—	—
> Compute Instances - Remove from DR Protection Group	Built-in	● Enabled	—	—
> Volume Groups - Terminate	Built-in	● Disabled	—	—
> Volume Groups - Remove from DR Protection Group	Built-in	● Enabled	—	—

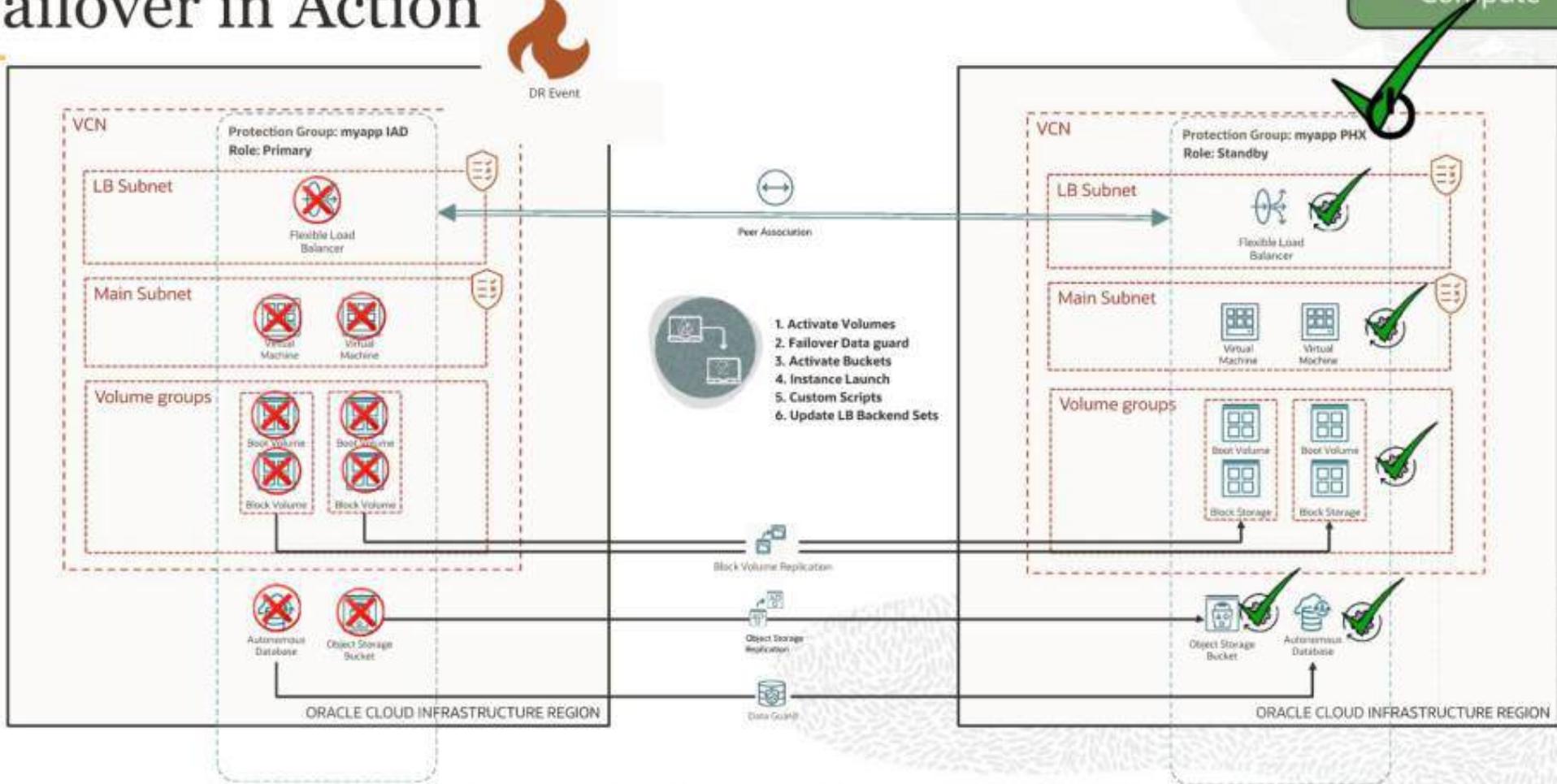
# DR Plans



# Failover in Action

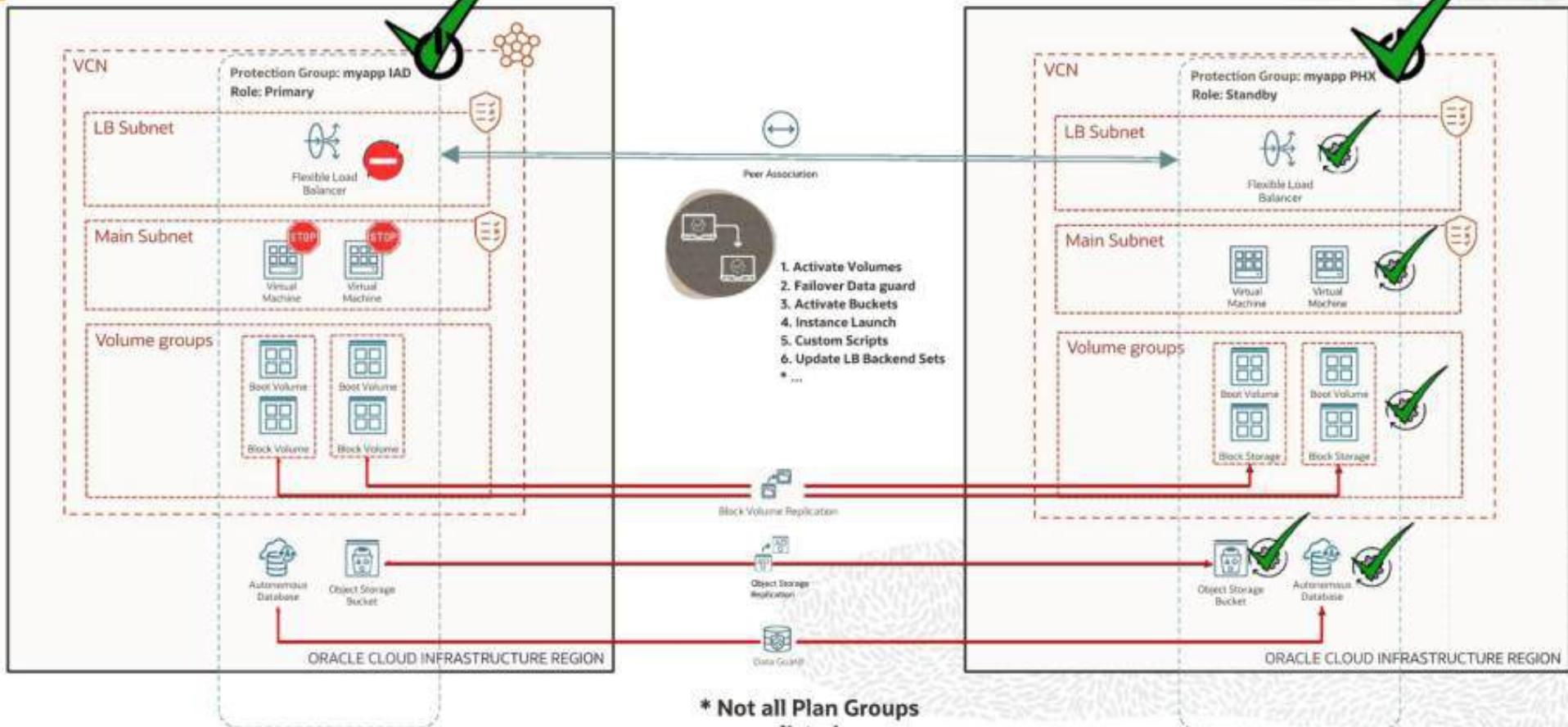


Movable Compute



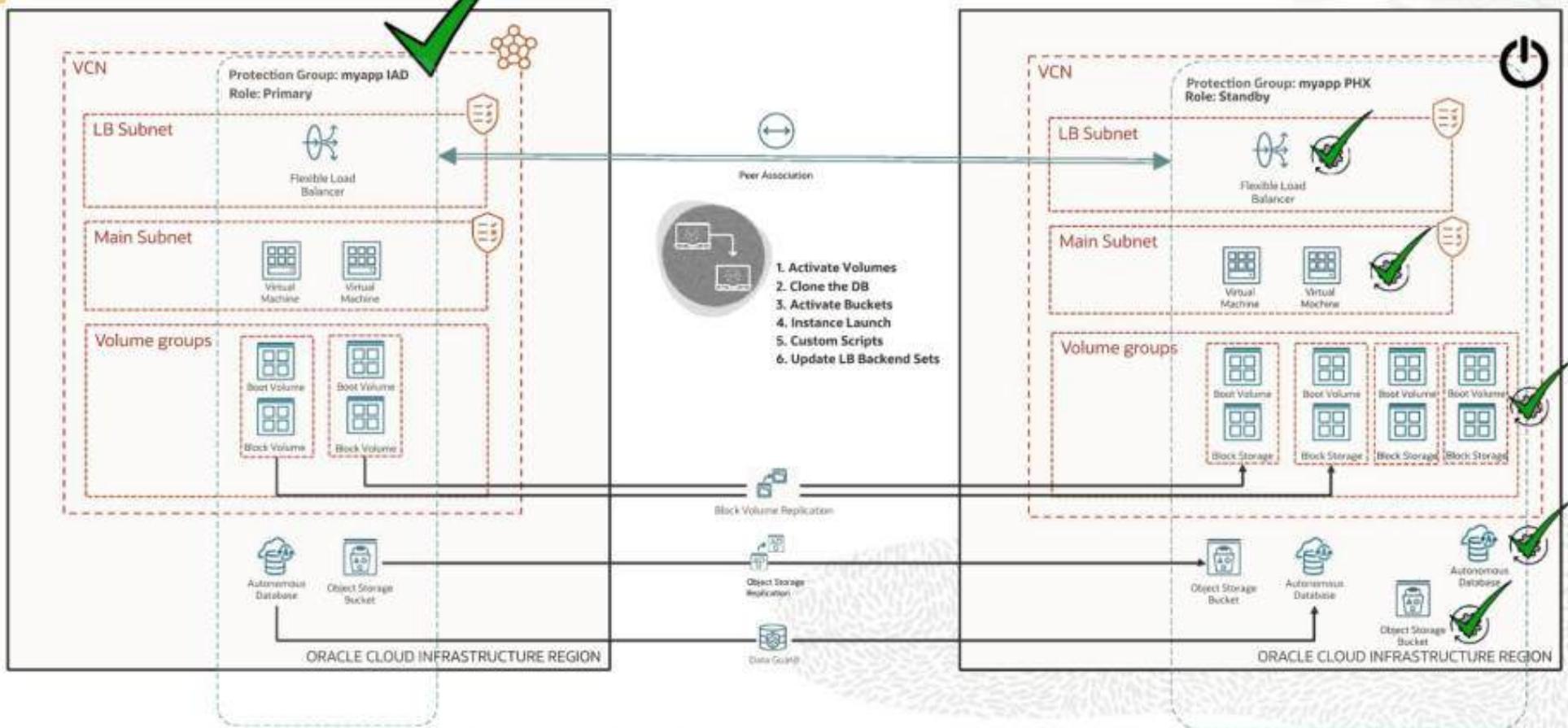
Movable  
Compute

# Switchover in Action

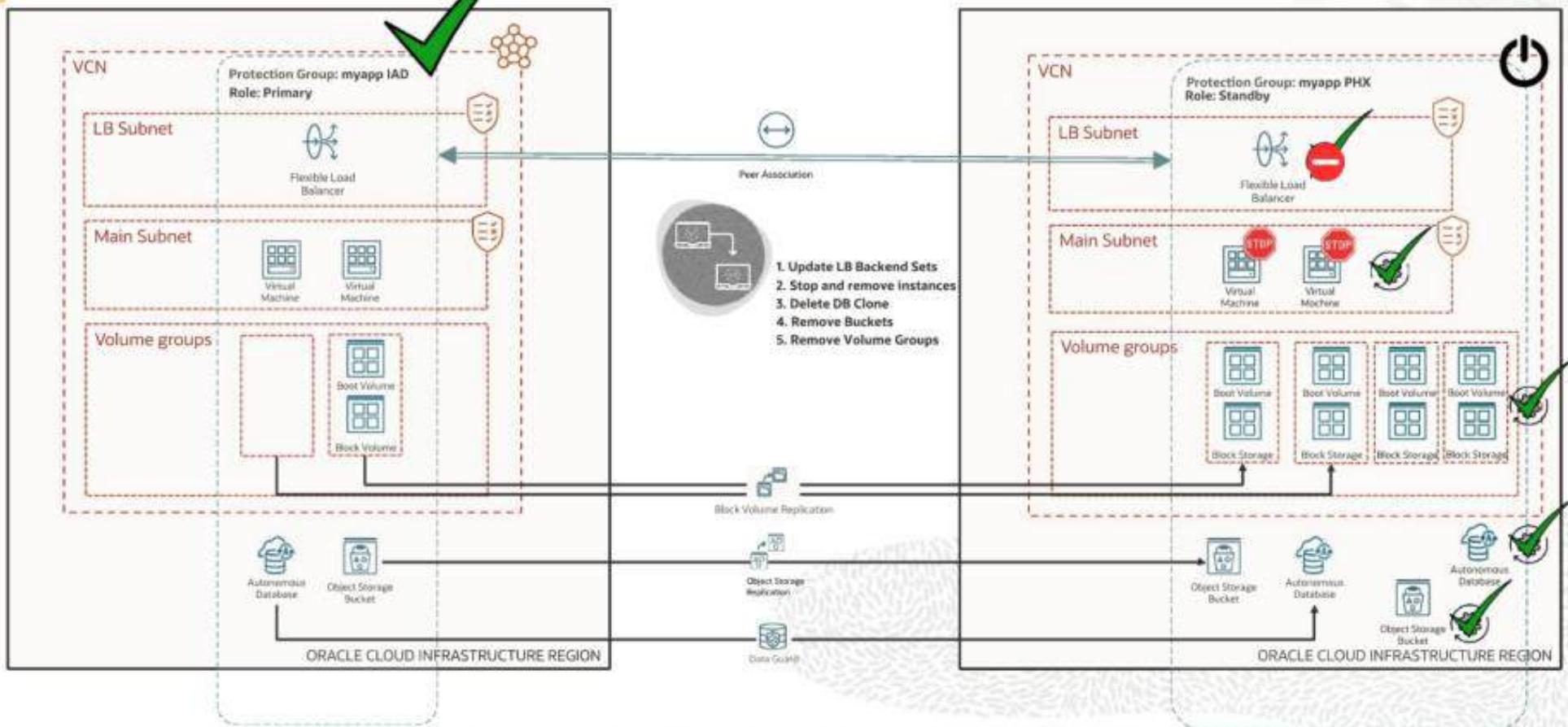


# Start Drill in Action

Movable  
Compute



# Stop Drill in Action

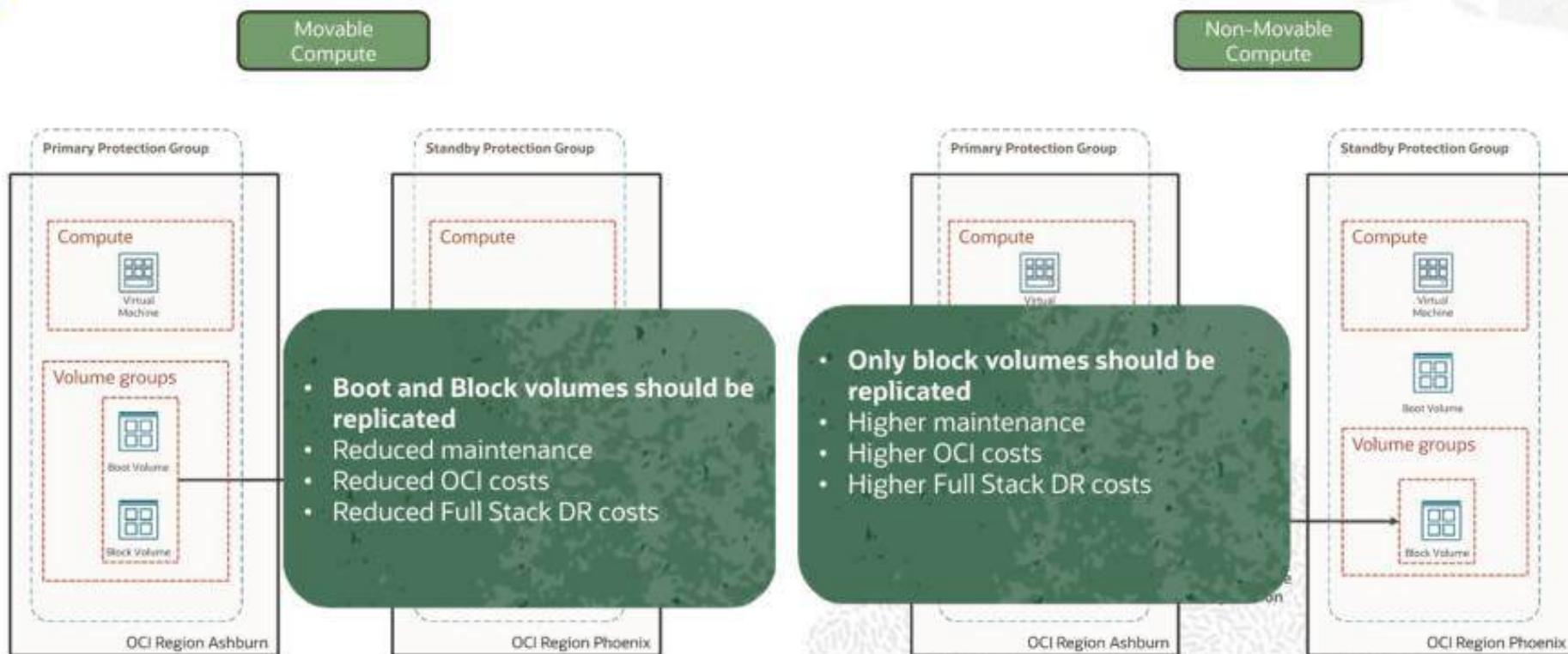


# Full Stack Disaster Recovery

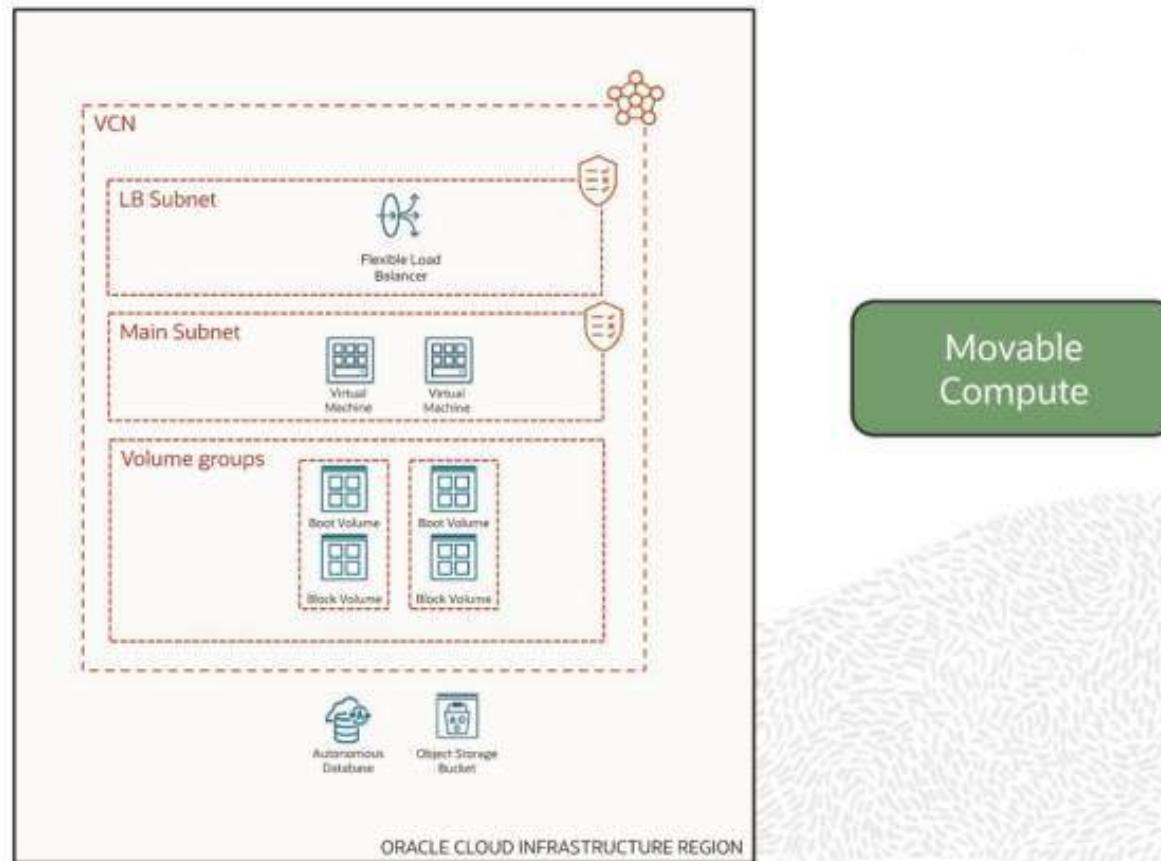
## Requirements



# Movable instance vs Non-movable instance

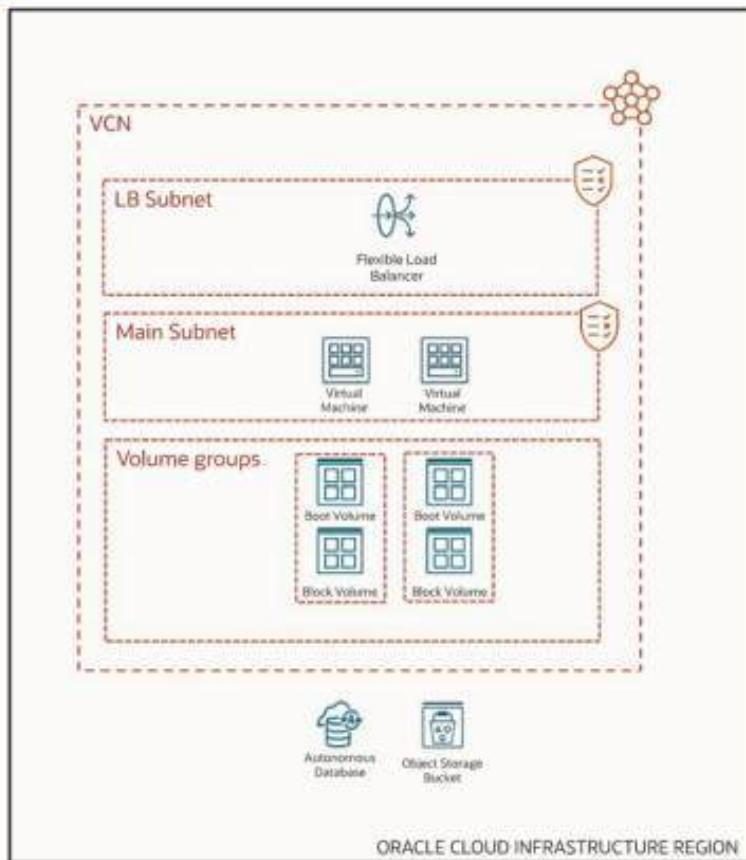


# Preparing for Full Stack Disaster Recovery



Movable  
Compute

# Preparing for Full Stack Disaster Recovery

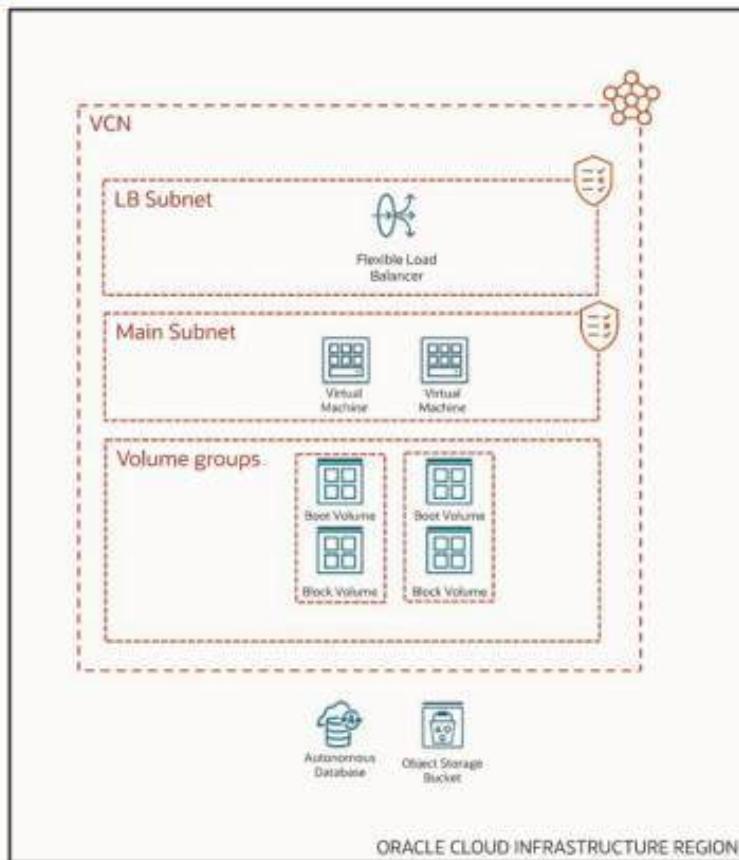


Choose  
Peer Region

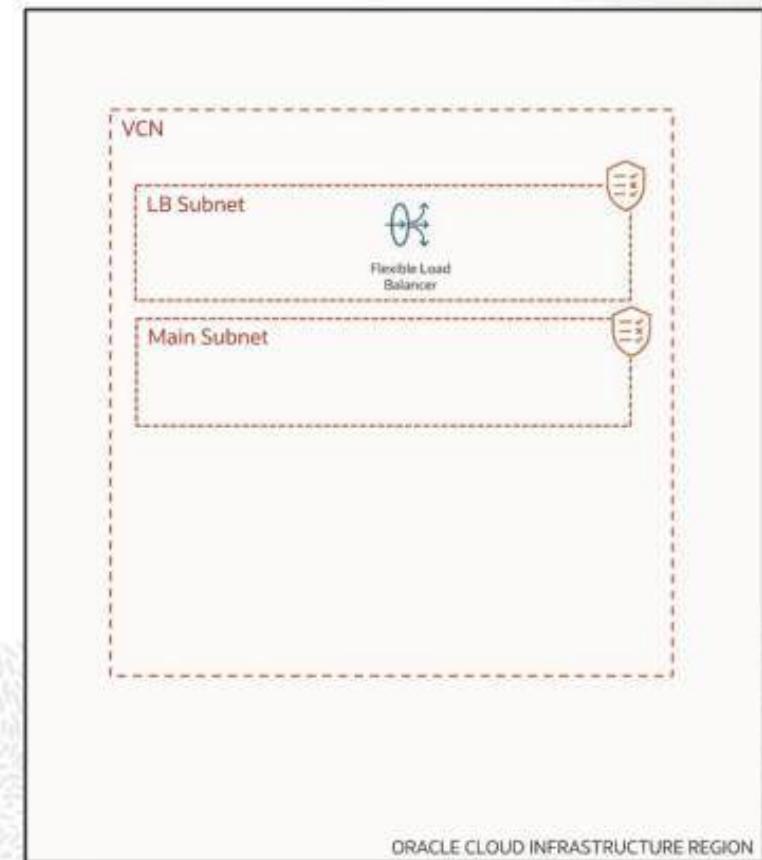
ORACLE CLOUD INFRASTRUCTURE REGION

# Preparing for Full Stack Disaster Recovery

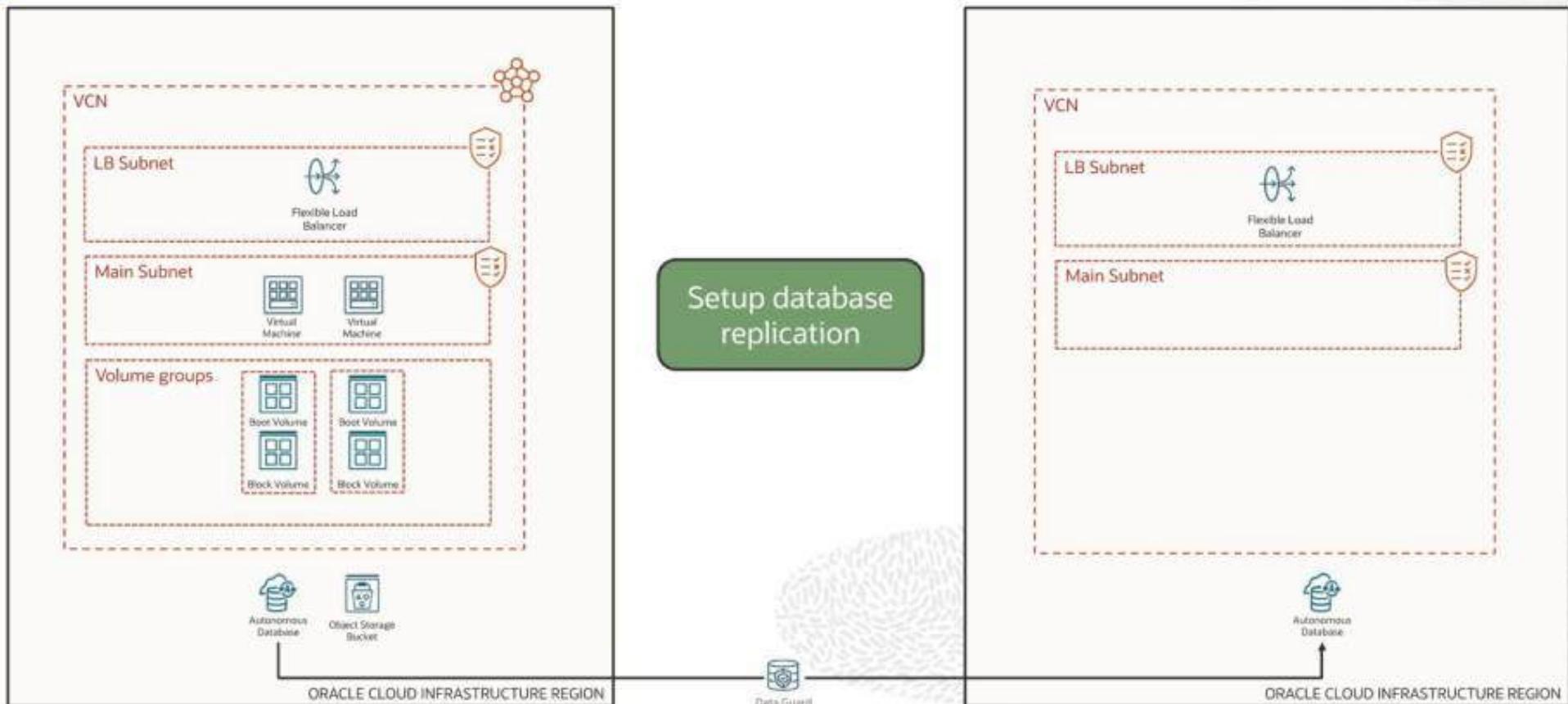
Movable  
Compute



Setup the  
network  
components

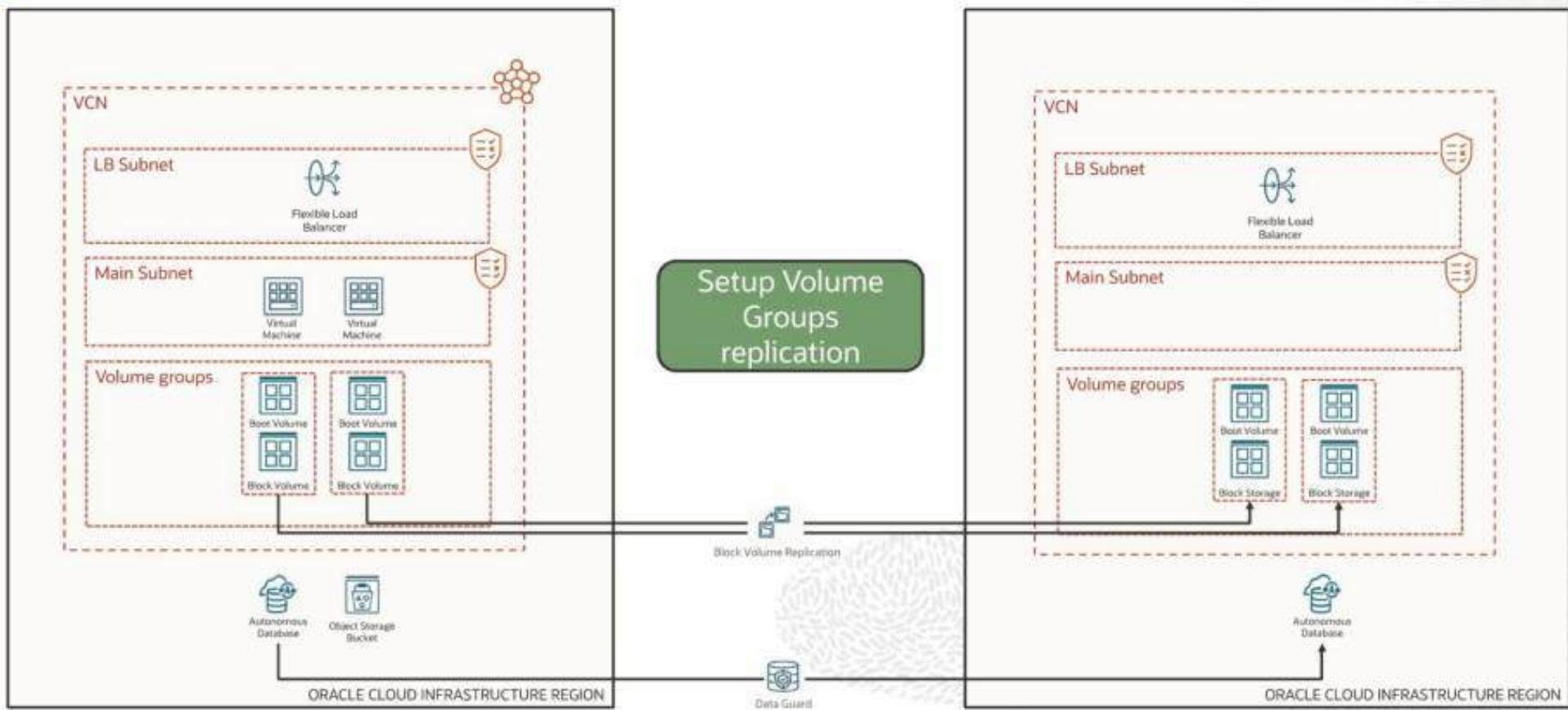


# Preparing for Full Stack Disaster Recovery

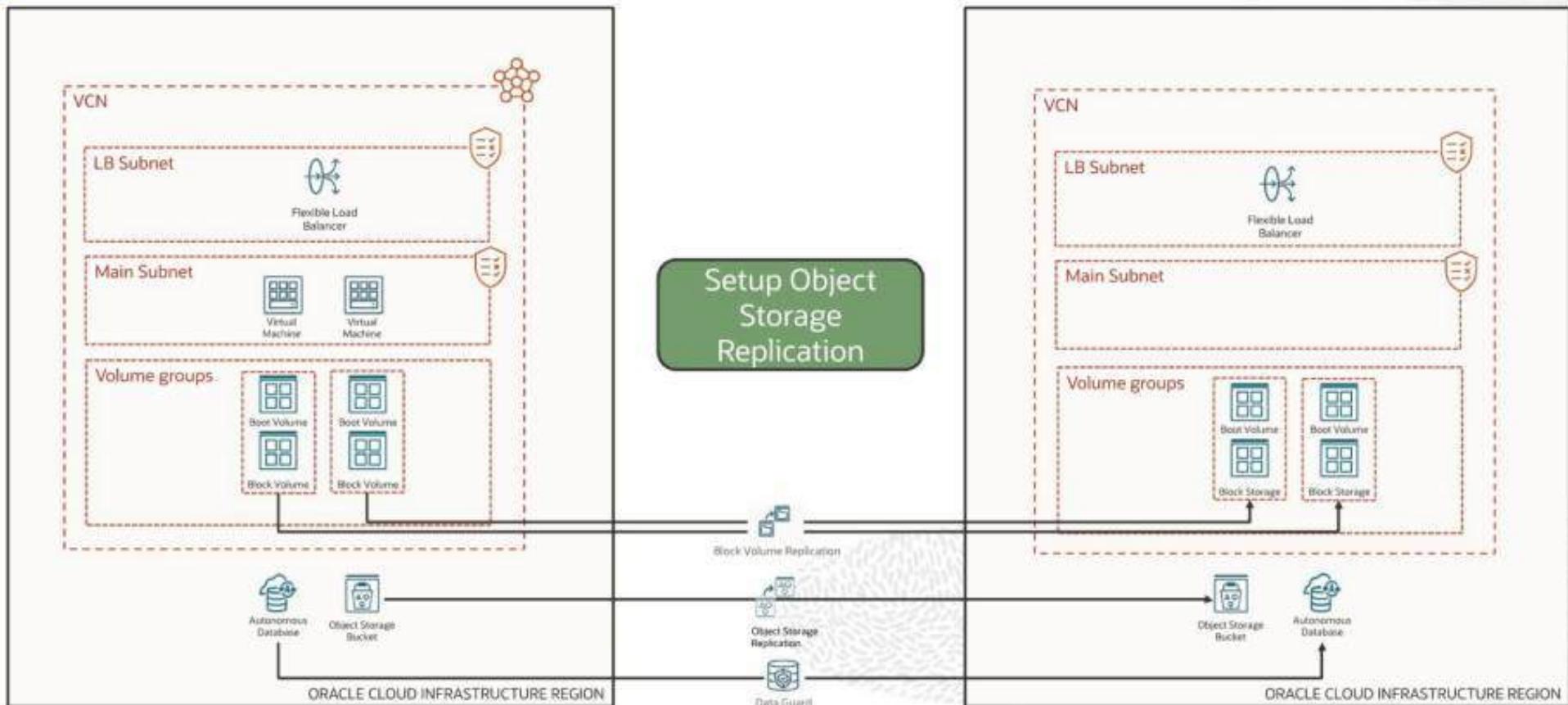


Movable  
Compute

# Preparing for Full Stack Disaster Recovery



# Preparing for Full Stack Disaster Recovery



# Preparing for Full Stack Disaster Recovery

## Movable Compute Scenario

- Peer region** Choose the region that will be used
- Network Infrastructure** Setup the network components
- Database replication** Setup the database replication
- Volume Groups Replication** Setup Block Storage replication
- Object Storage Replication** Setup Object Storage replication

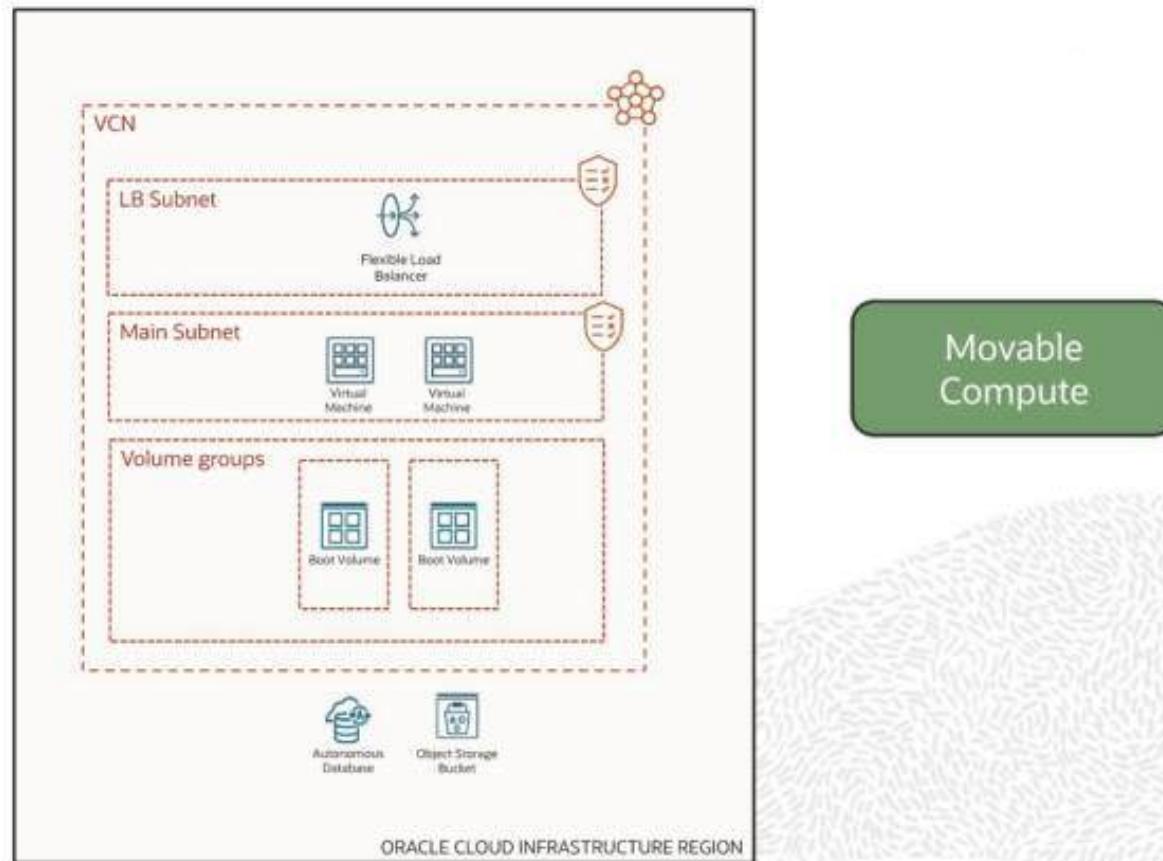


# Full Stack Disaster Recovery

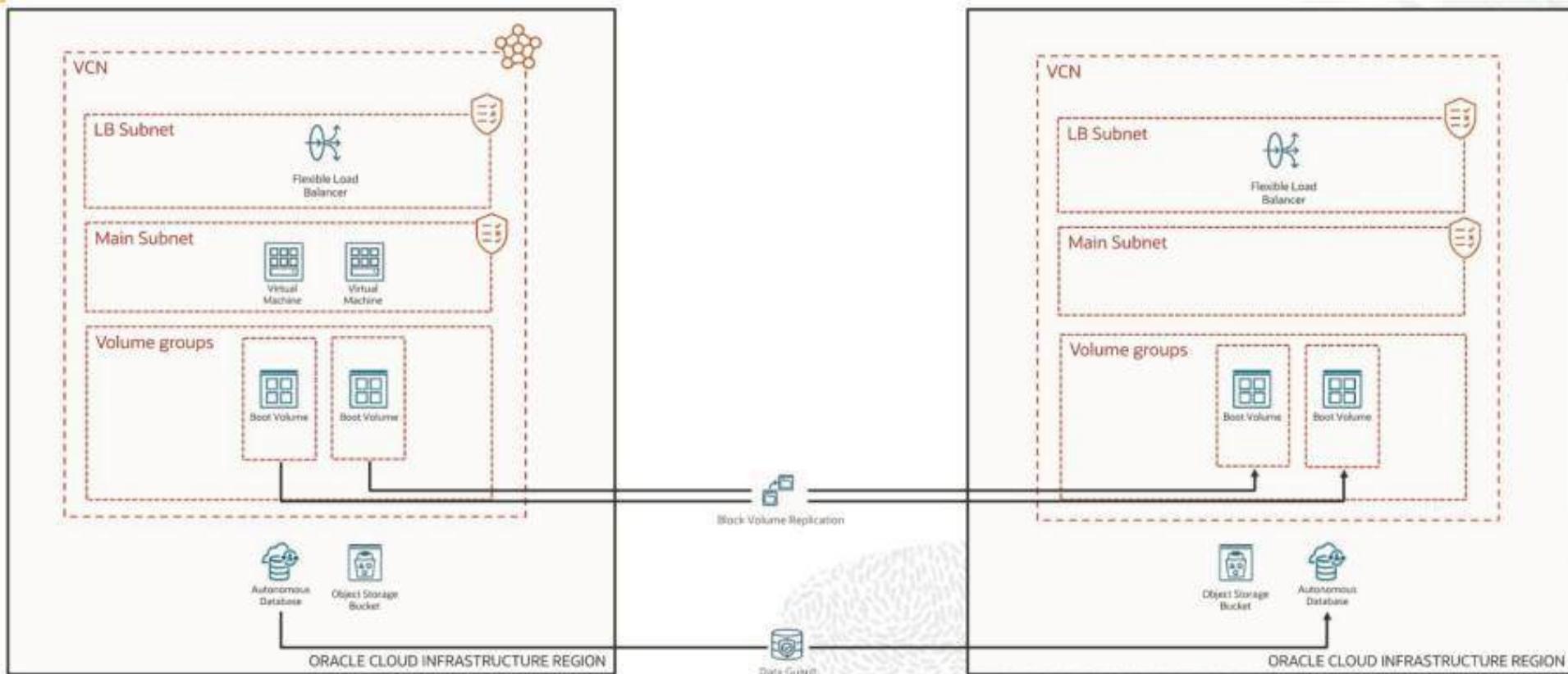
## Demo – Setup



# Preparing Mushop for Full Stack DR



# MuShop Scenario after deployment

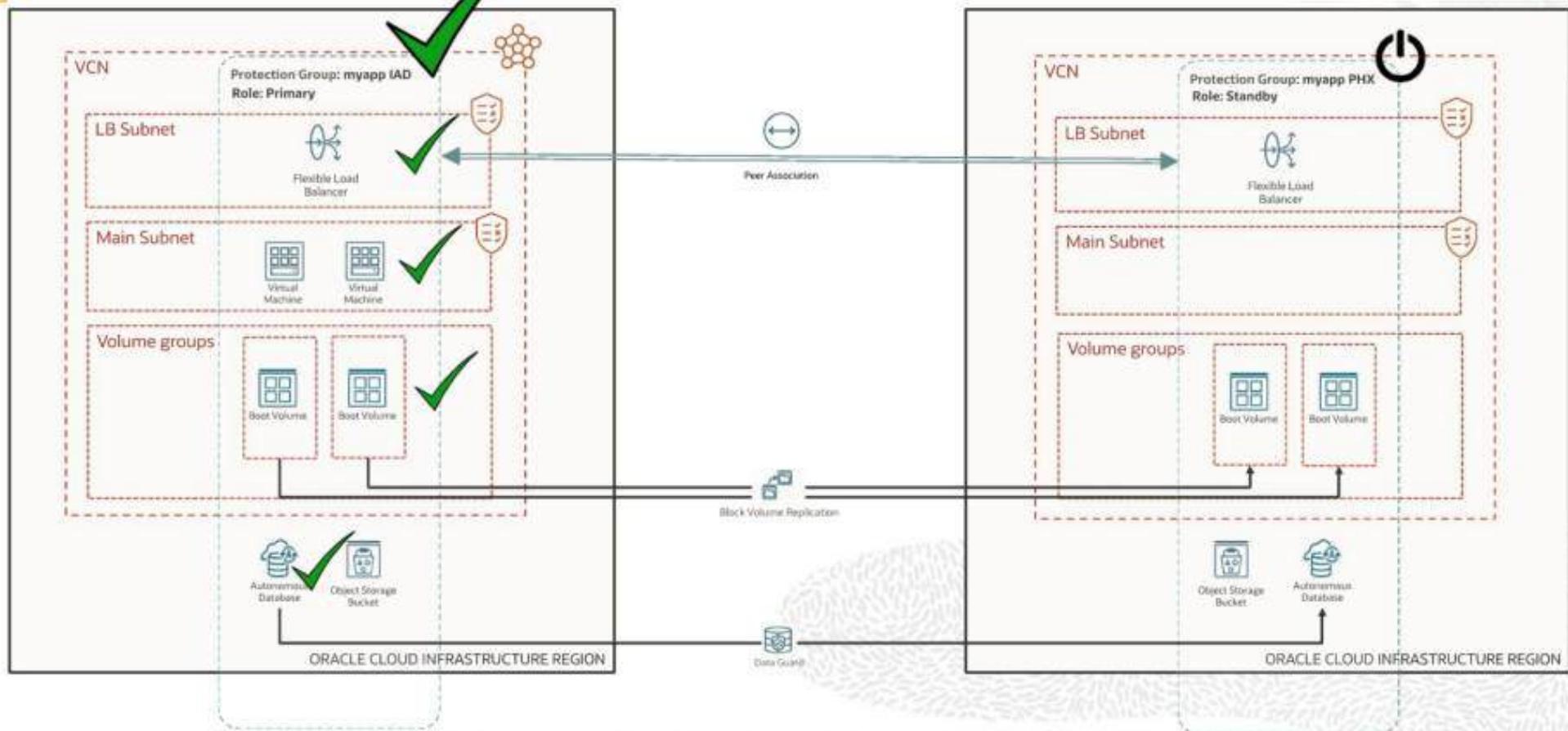


# Full Stack Disaster Recovery

**Demo – DR plan Pre-Check and execution**

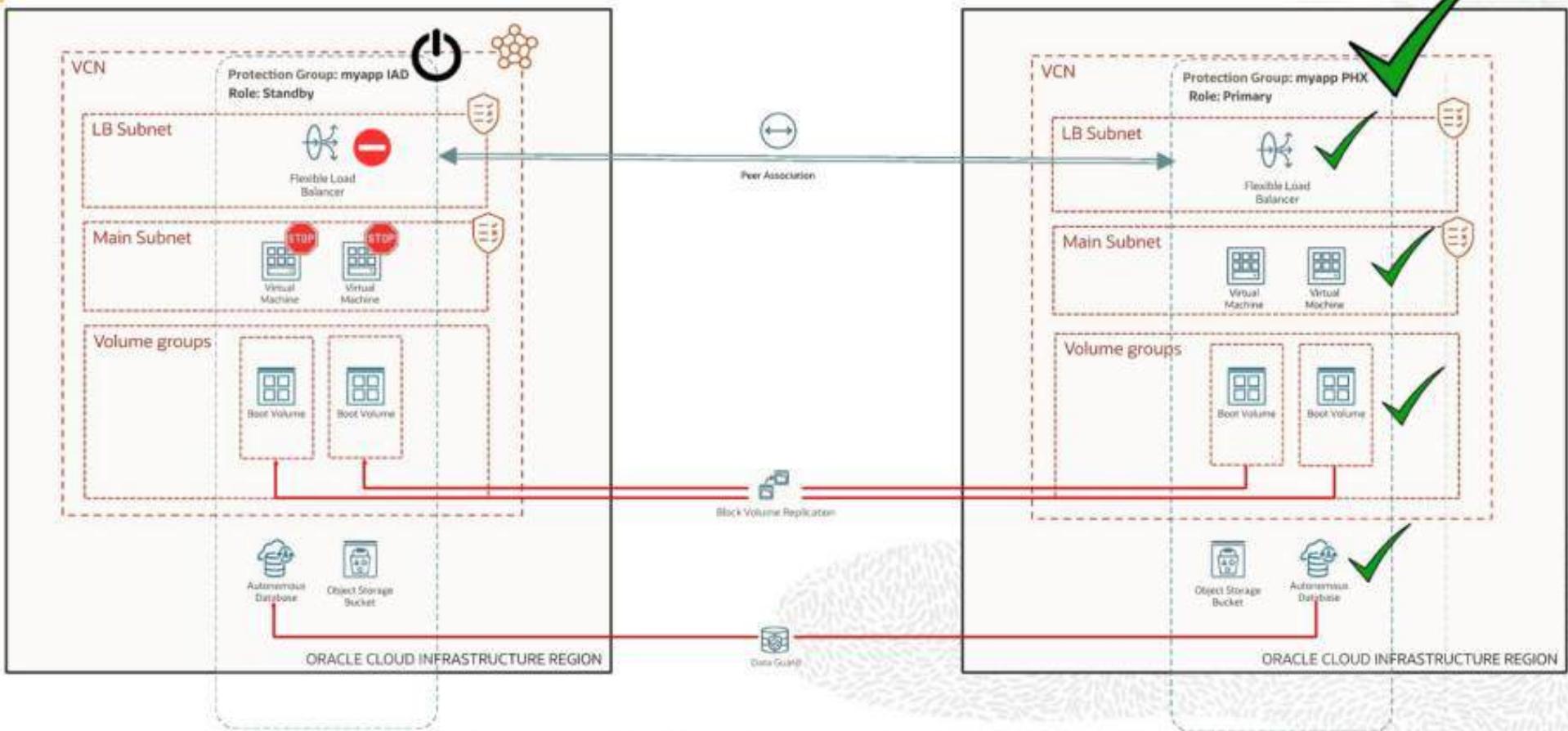


# MuShop Scenario after plan setup



# MuShop Scenario after Switchover

Movable Compute





# Troubleshooting



# Oracle Cloud Infrastructure Troubleshooting

## Oracle Cloud Infrastructure

# Objectives

---

After completing this lesson, you should be able to describe troubleshooting OCI Compute Services.



# SSH Connection

- The SSH key pair establishes trust between the client and server, thereby removing the need for a password during authentication.
- Client Side
  - Key Error
  - Permission Error
  - Firewall Issue
  - Security list issue
  - Different mechanism for Windows – Use Putty
- Server Side
  - Public key file should be available in ".ssh" directory under a user's home directory and should be readable by all (chmod +r publickey)

# Instance Console Connections

- Enables you to remotely troubleshoot instances, such as:
  - An imported or customized image that does not complete a successful boot
  - A previously working instance that stops responding
- You can perform tasks, such as:
  - Edit system configuration files
  - Add or reset the SSH keys for the opc user
- Two types of instance console connections:
  - Serial console connections
  - VNC console connections

# Troubleshooting Performance

- Check CPU, Memory, and Storage of the instances: Do they have capacity or fully utilized?
- Check system logs.
- If instance is not accessible, create an instance console connection and check console logs.
- Check application monitors and logs.
- If a database, check query performance.
- Validate Security Lists.
- Check network monitoring tools utilized for latency.
- If leveraging FastConnect or IPsec, validate health of on-premises routers and firewalls.



# Oracle Cloud Infrastructure Troubleshooting

## Oracle Cloud Infrastructure

# Objectives

---

After completing this lesson, you should be able to describe troubleshooting OCI Networking Services.



# IPSec connection testing

- Network Route Review:
  - Are there overlapping CIDRs
  - Are there multiple SPIs with Policy-Based tunnels?
  - Are both tunnels up?
- Ping Tests:
  - Send a continuous ping from On-Prem to the OCI Device
  - Send a continuous ping from the OCI Device to On-Prem
  - Inconsistent ping results
- Support Tools: iPerf for any iPerf test
- Tcpdump

# FastConnect Redundant Connections

- Two options for redundancy: IPSec connection or FastConnect connection
- Create a separate IPSec connection with less specific (or the same) static routes.
- Make sure only one tunnel is being used as your primary and that your CPE does not send traffic down the second tunnel while the primary is active.
- If you'd like to do separate FastConnects, you will need to have two separate VCs on two separate physical connections.

# Load Balancer Health Status

- A health check is misconfigured.
- A listener is misconfigured.
- A security rule is misconfigured.
- One or more of the backend servers reports as unhealthy.
- Other cases in which health status might prove helpful include:
  - VCN network security groups or security lists block traffic.
  - Compute instances have misconfigured route tables.

# Health Check

- STATUS: The status returned by the health check. Possible values include:
  - OK
  - INVALID\_STATUS\_CODE
  - TIMED\_OUT
  - REGEX\_MISMATCH
  - CONNECT\_FAILED
  - IO\_ERROR
  - OFFLINE
  - UNKNOWN



# Oracle Cloud Infrastructure Troubleshooting

## Oracle Cloud Infrastructure

# Objectives

---

After completing this lesson, you should be able to describe the following:

- Troubleshooting OCI Block Volume Service



# Block Storage Backup Copy – Common Errors

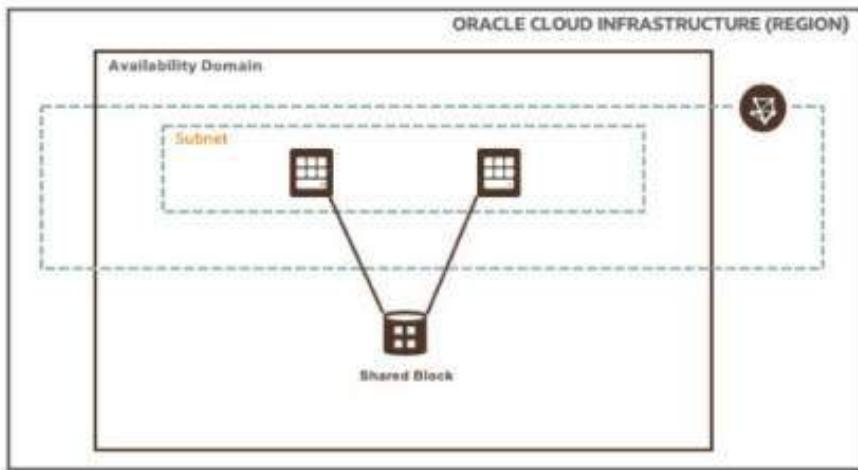
- Resource not found or action not authorized in destination region
- A copied backup already exists in the destination region %X% with a kmsKeyId which differs from %Y%"
- Backup quota exceeded in destination region
- Allowed limit of parallel cross-region copies reached.  
Maximum allowed no. of parallel cross region %X%

# Block Storage Recovery steps

1. Detach the block volumes from the instance
2. Back up the block volumes
3. Use the backups to spin up new volumes
4. Attach these new block volumes to a new instance
5. Verify your data on these new block volumes
6. After mount and run the iscsi commands
7. Mount it and access it

# Block Storage Multi-Attach

- Multi-Attach feature is to allow multiple instances to share 1 data volume (Boot Volume is NOT shareable).
- A user can config a block volume as shareable during attach call.
- Common Error message:
  - To attach a volume to more than one instance, all volume attachments must be configured as shareable.
  - The read-write attachment type conflict.
  - The volume shared by too many instances



# Block Storage Volume Resize

- Expand an existing volume in place with online resizing.
- Restore from a volume backup to a larger volume.
- Clone an existing volume to a new, larger volume.
- Expand an existing volume in place with offline resizing.

## Guidelines:

- Volume size may only be increased.
- Volumes may not be resized if there a prior resize / clone is still ongoing.
- Volumes may not be resized if there is a backup pending.
- Volumes may not have attachments added or removed during resize.

## Oracle Cloud Infrastructure Local NVMe Device Failures

# Local NVMe Device

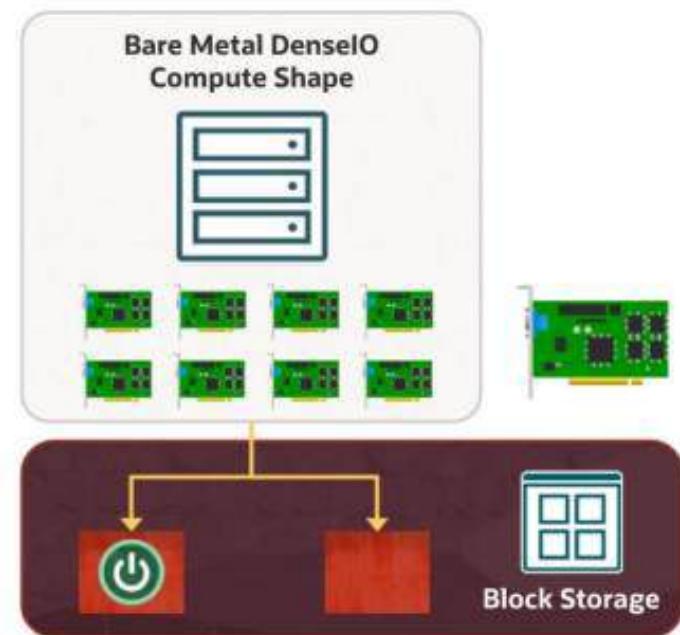
Found in some compute instance shapes like DenseIO

Best I/O performance

Boot volume for all instances is in Block Storage

Block Storage/File Storage have redundancy built into the service

Protect the data in the local devices by implementing RAID

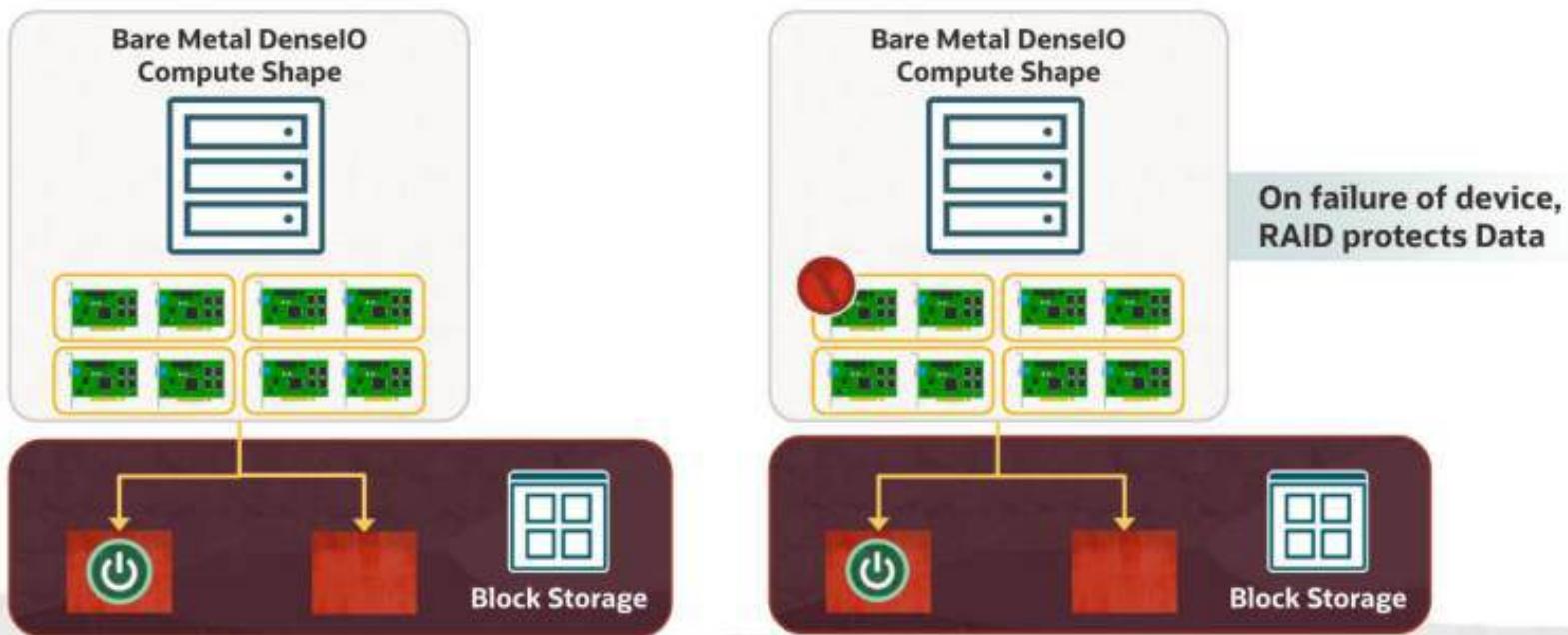


# RAID with Local NVMe Device

Local devices are accessible to the OS of the compute instance

Data is written to two or more local devices simultaneously

In the event of an NVMe device failure, your data is still accessible/available



## When a Device Fails

Detect Local Device failures

Spin a new Compute Instance of Similar Shape

Implement RAID with local devices

Copy data from the Old Instance to the New Instance using tools like rsync

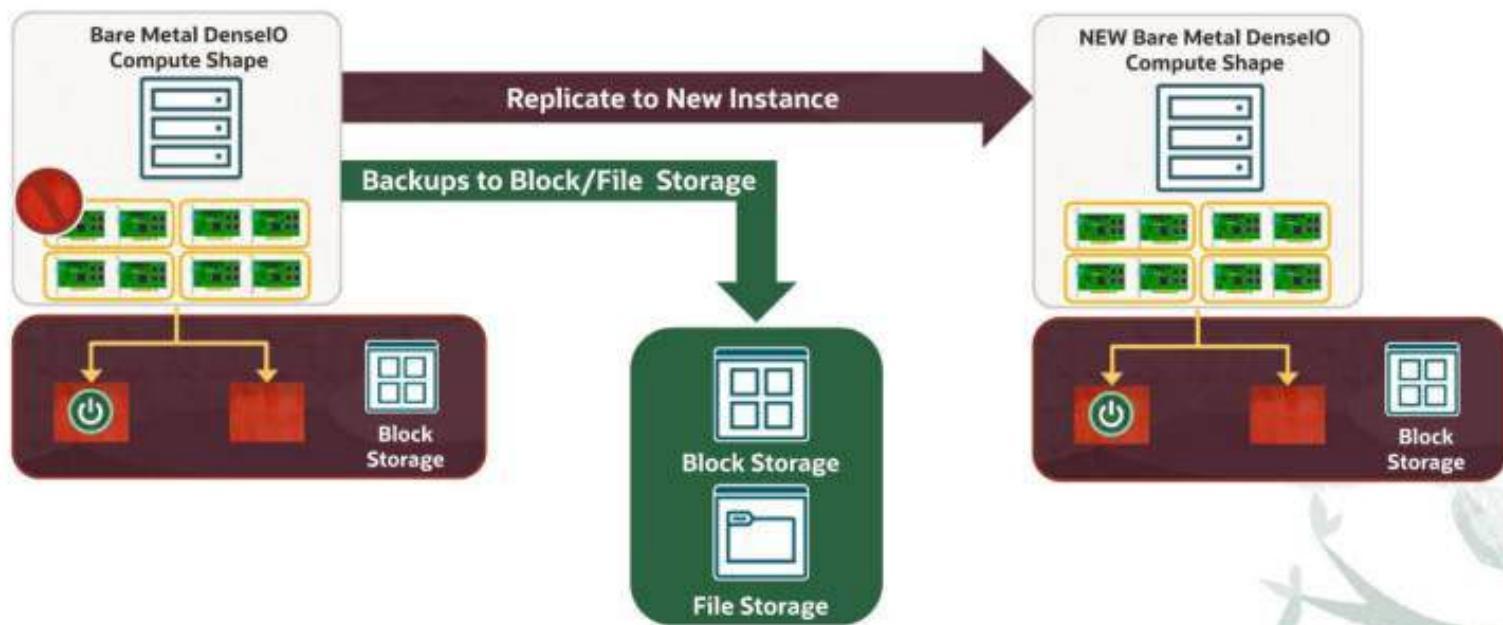
Terminate the old instance



# What if the Availability Domain fails?

Backups to Block Storage/File Storage

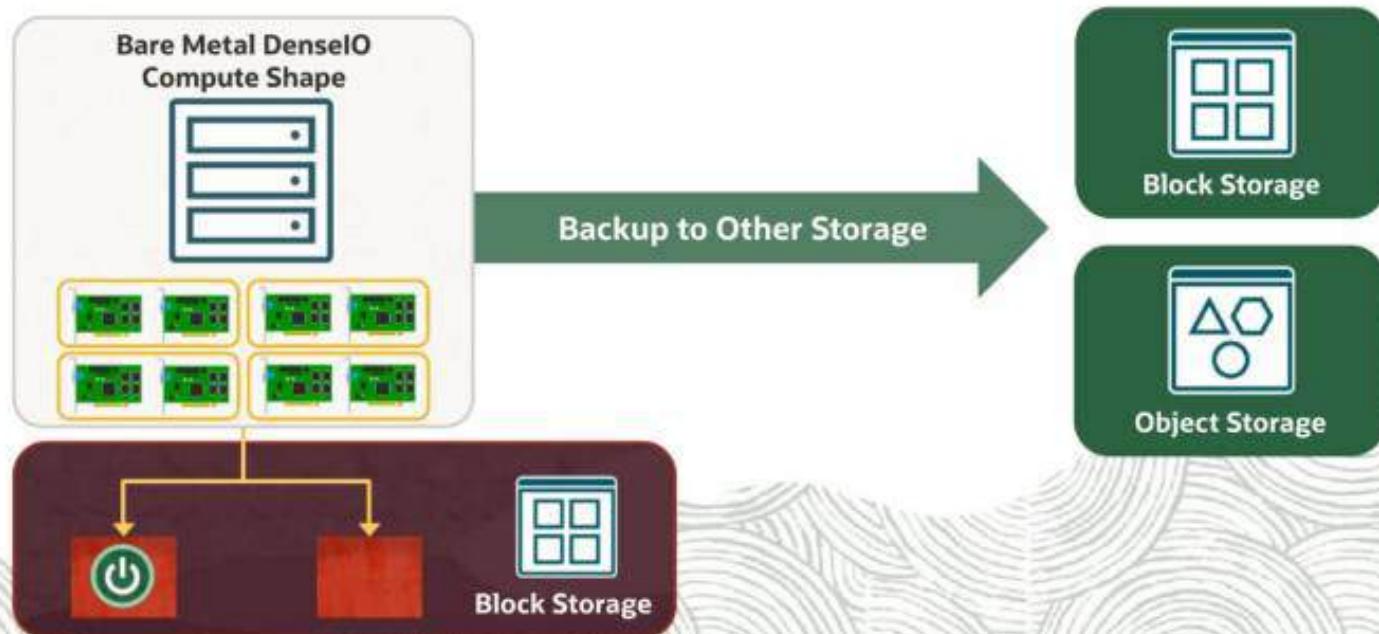
Replicate to another Compute Instance



# Backups to Block Storage/File Storage

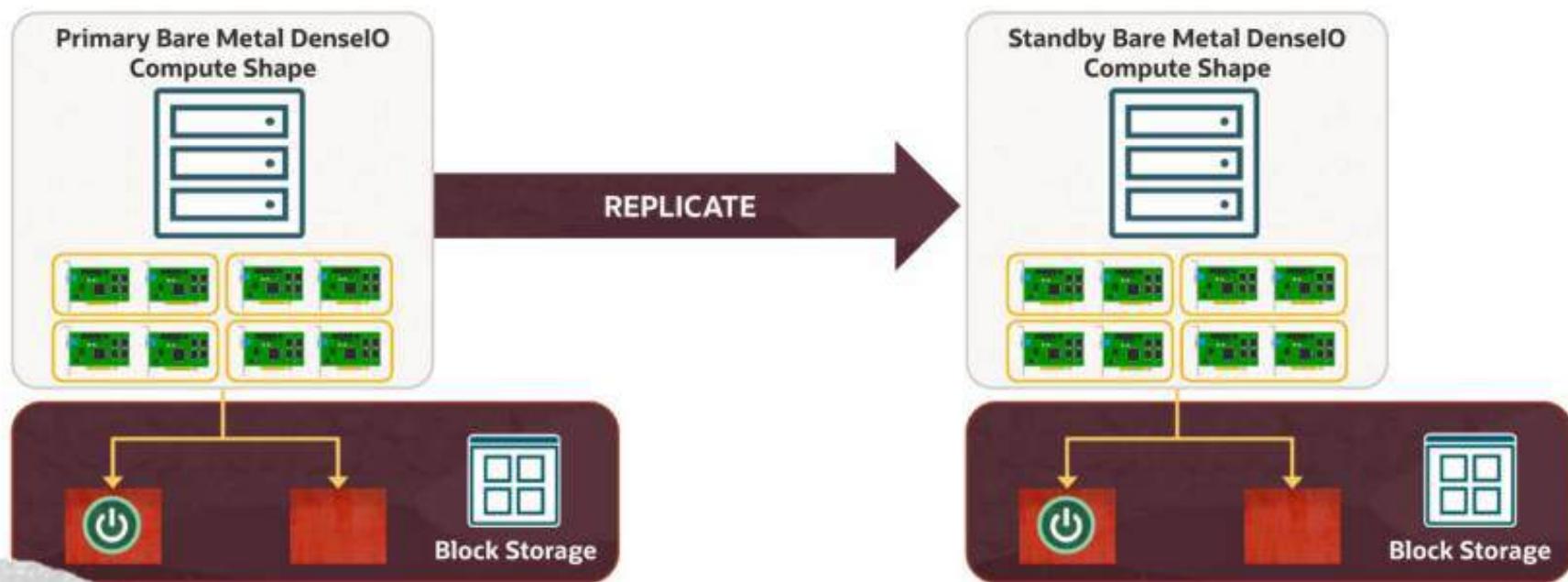
Backups to Block Storage/File Storage within same AD or to a different AD or another Region

Backups to Object Storage in same Region or another Region



# Replicate to another Compute Instance

If you want to protect from a regional failure (Disaster Recovery), create the Second Instance in another Region and replicate your data



## Oracle Cloud Infrastructure

# Troubleshoot and Attach Orphaned Mount Targets

# Mount Target and File Systems



File System available through Mount target

Mount Target is a VNIC in your VCN Subnet

Mount Target VNIC to be created in same AD as File System

The screenshot shows the "Create File System" wizard. At the top, it says "File Storage provides durable, scalable, and secure file systems." Below this is the "File System Information" section, which includes fields for Name (FS1), Availability Domain (yQU EU-FRANKFURT-1-AD-1), Compartment (greyed out), and Encryption Key (Oracle-managed key). The "Edit Details" link is located in the top right corner of this section. The next section is "Export Information", which includes fields for Export Path (/FS1) and a "Use Secure Export Options" toggle (disabled). The "Edit Details" link is located in the top right corner of this section. The final section is "Mount Target Information", which includes fields for Mount Target Name (mnt1) and Compartment (greyed out). The "Edit Details" link is located in the top right corner of this section. At the bottom of the screen are "Create" and "Cancel" buttons.

## Mount Target and File Systems

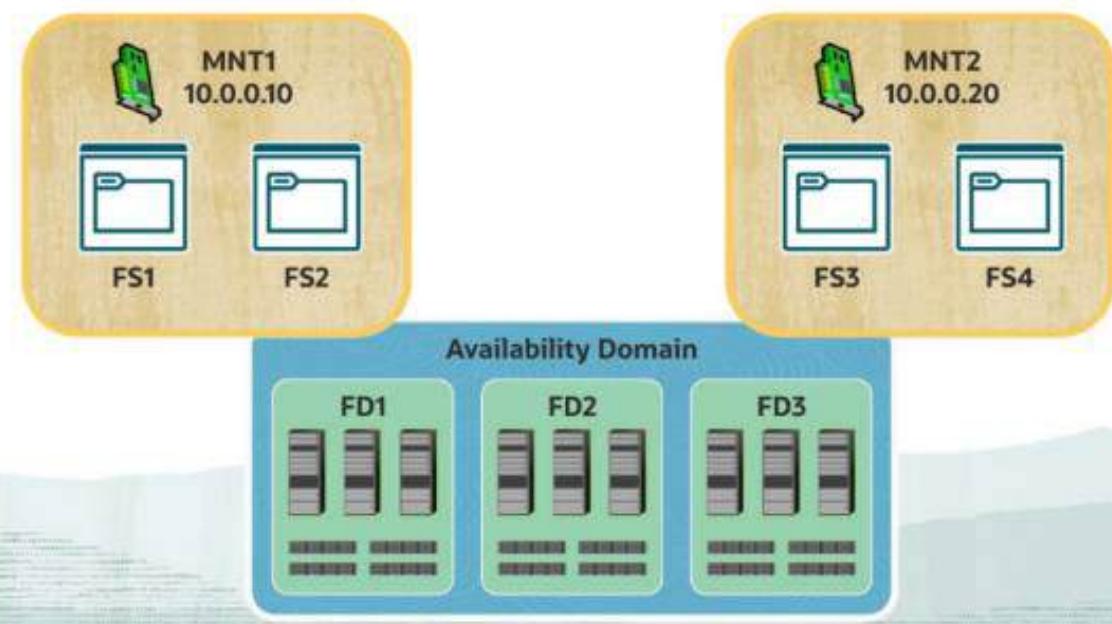


Create an Export path

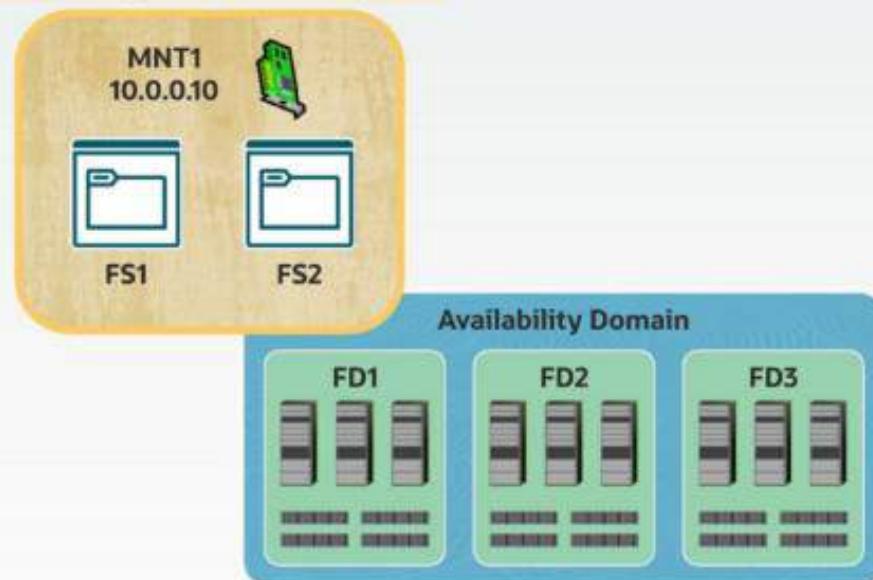
Up to 100 file systems  
behind a single Mount target

Two Mount targets in one AD

Both Mount Targets have two file systems



10.0.0.10:/FS1 10.0.0.10:/FS2



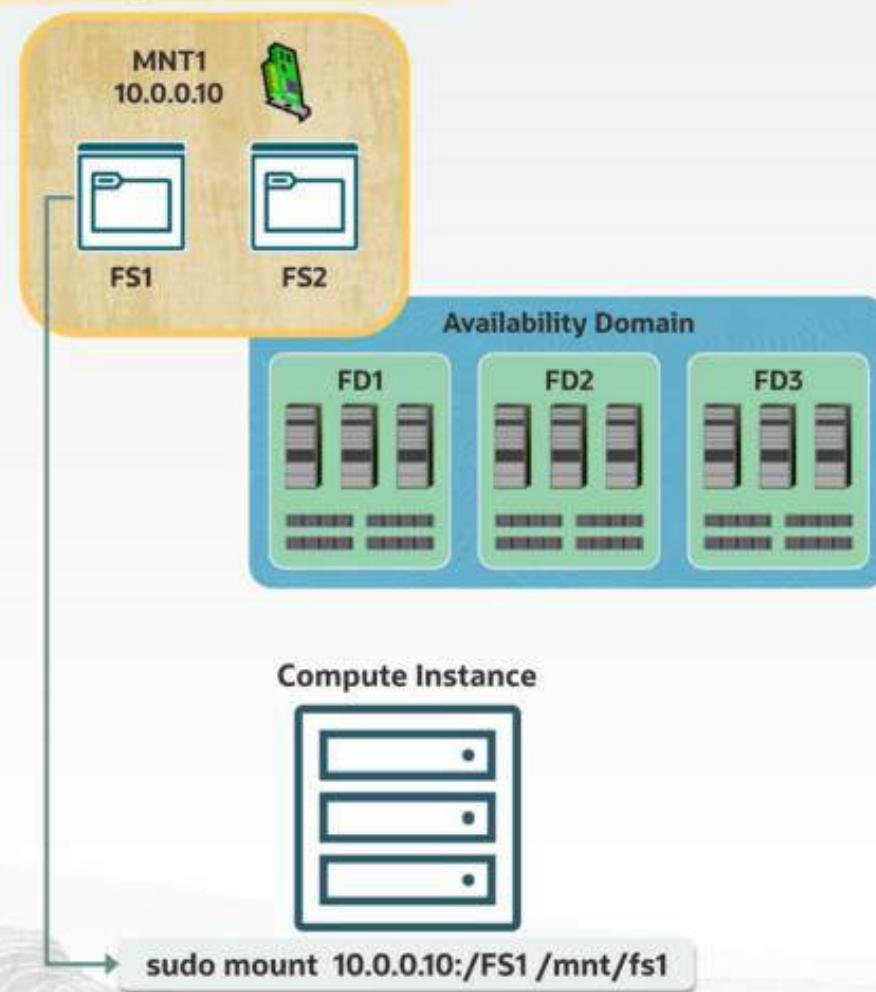
## Troubleshoot File Systems

Do not create Export Path for a File System using “root” / path in the Mount Target

If you create, it limits you to create only 1 Export path and only 1 File System can be behind the Mount Target

To attach multiple File Systems to a Mount Target ensure each File System has a unique Export Path

10.0.0.10:/FS1 10.0.0.10:/FS2



## Troubleshoot File Systems

Create a directory, called as the Mount path/Mount Point and mount the File System using the Export Path

Firewall rules on specific Ports on TCP and UDP

## Troubleshoot File Systems

Export path



File system



Detached  
and deleted

Mount target

Same Export path



New File  
system



Existing  
Mount target

Solution

# Observability & Management

## Oracle Cloud Infrastructure

# What is Observability



# Traditional Monitoring

*Monitor and understand the state of their systems with predefined sets of metrics or logs.*

## Purpose of monitoring

- Actively collect system metrics and logs
- Track errors as and when they occur
- Store metrics to later observe the system
- React to incidents through alerts



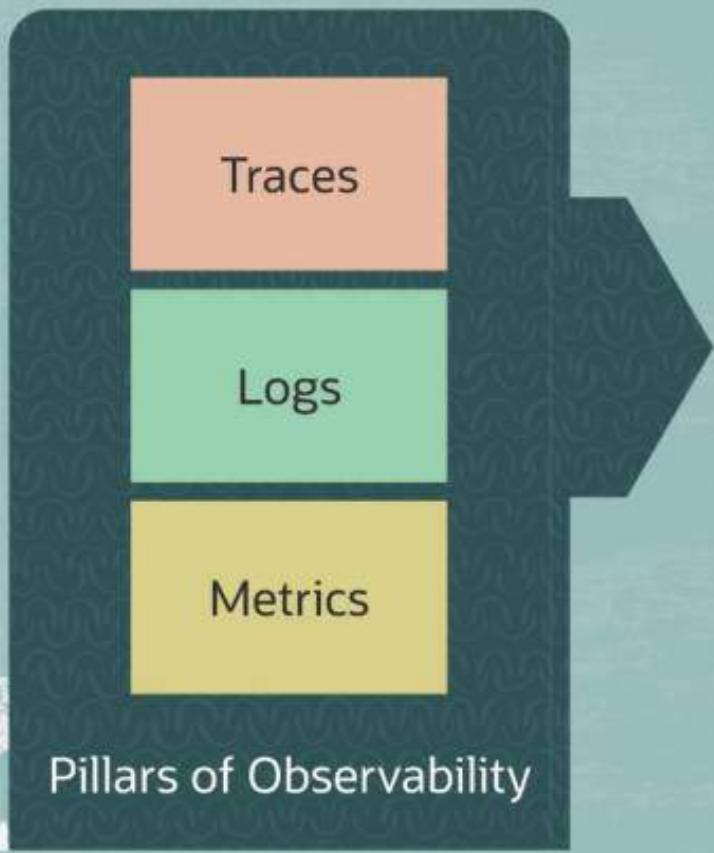


# Challenges with Traditional Monitoring

- What are the top applications or endpoints in our network?
- How many active users are there in a given time period?
- Why is my deployment taking so long?
- What Post/Put actions are run by the API Endpoint and why?
- What is the trend of data transfer between these points?
- Are there any unknown unknowns and why?
- Is there any unusual data transfer between them?
- How would I predict failures with distributed applications?



# Definition: Observability

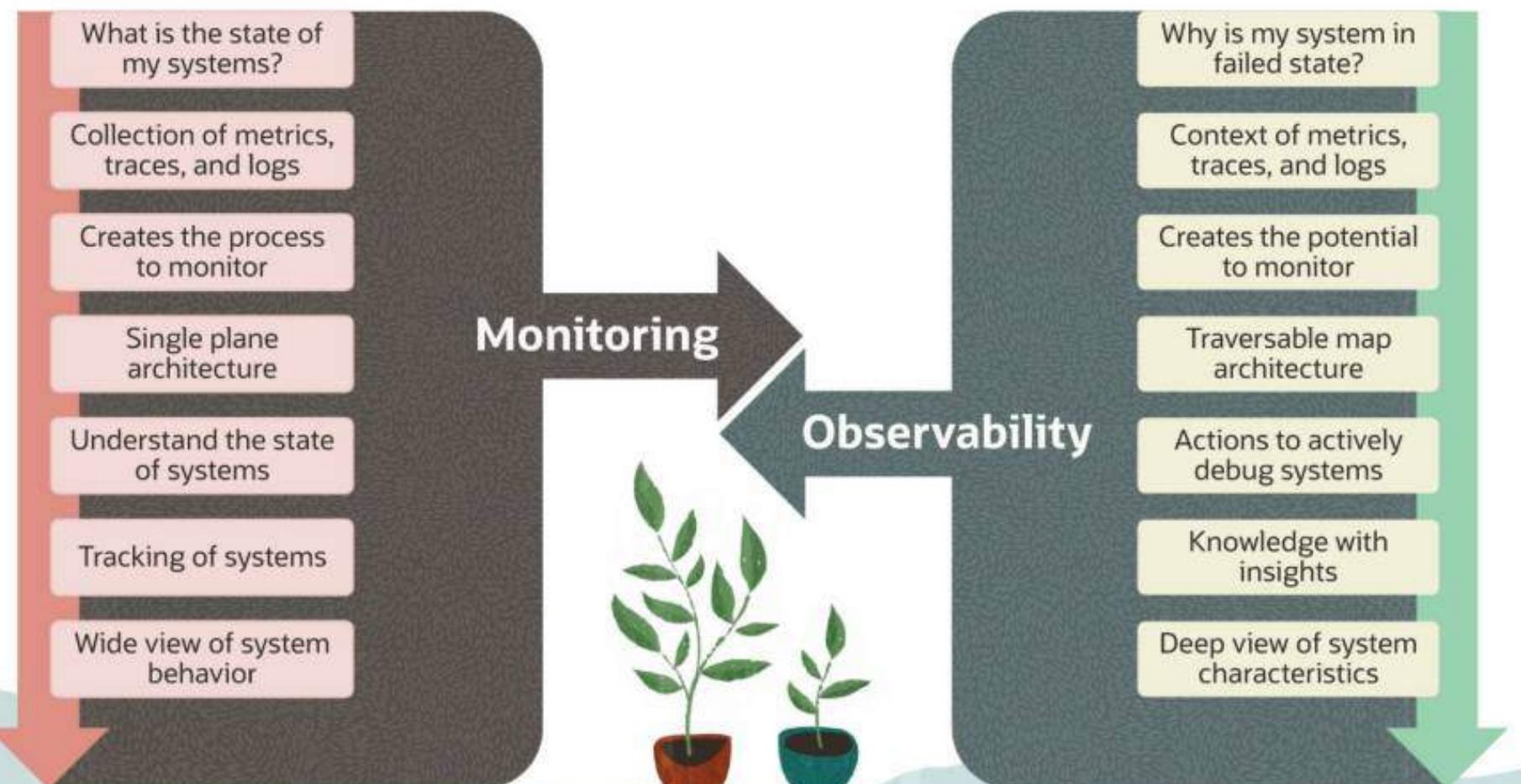


*Ability to understand a system's internal state by analyzing the data about what's happening in context across environments.*

## Purposes of Observability

- Enhance visibility in cloud environments
- Identifying trends and outliers
- Instrumentation, correlation, and computation
- Identifying potential bottlenecks and patterns
- Insights to take actions and prevent errors

# Comparing Monitoring and Observability



## Oracle Cloud Infrastructure

# Introducing Observability and Management Services

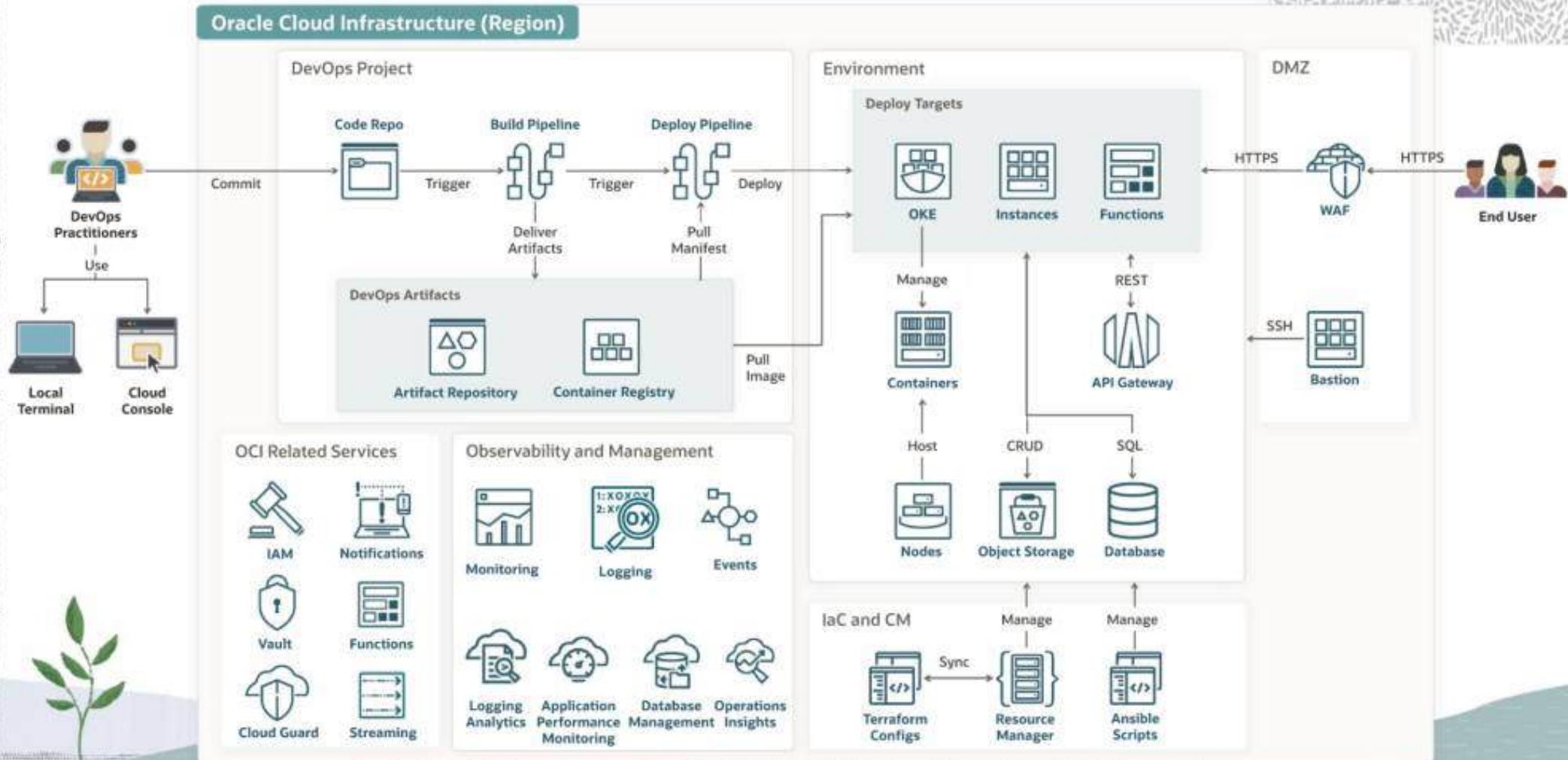


# Oracle Cloud Infrastructure

## Observability & Management Services



# Use Case: Observability and Management in DevOps



# Monitoring Service Overview

# OCI Monitoring Service: Getting Started



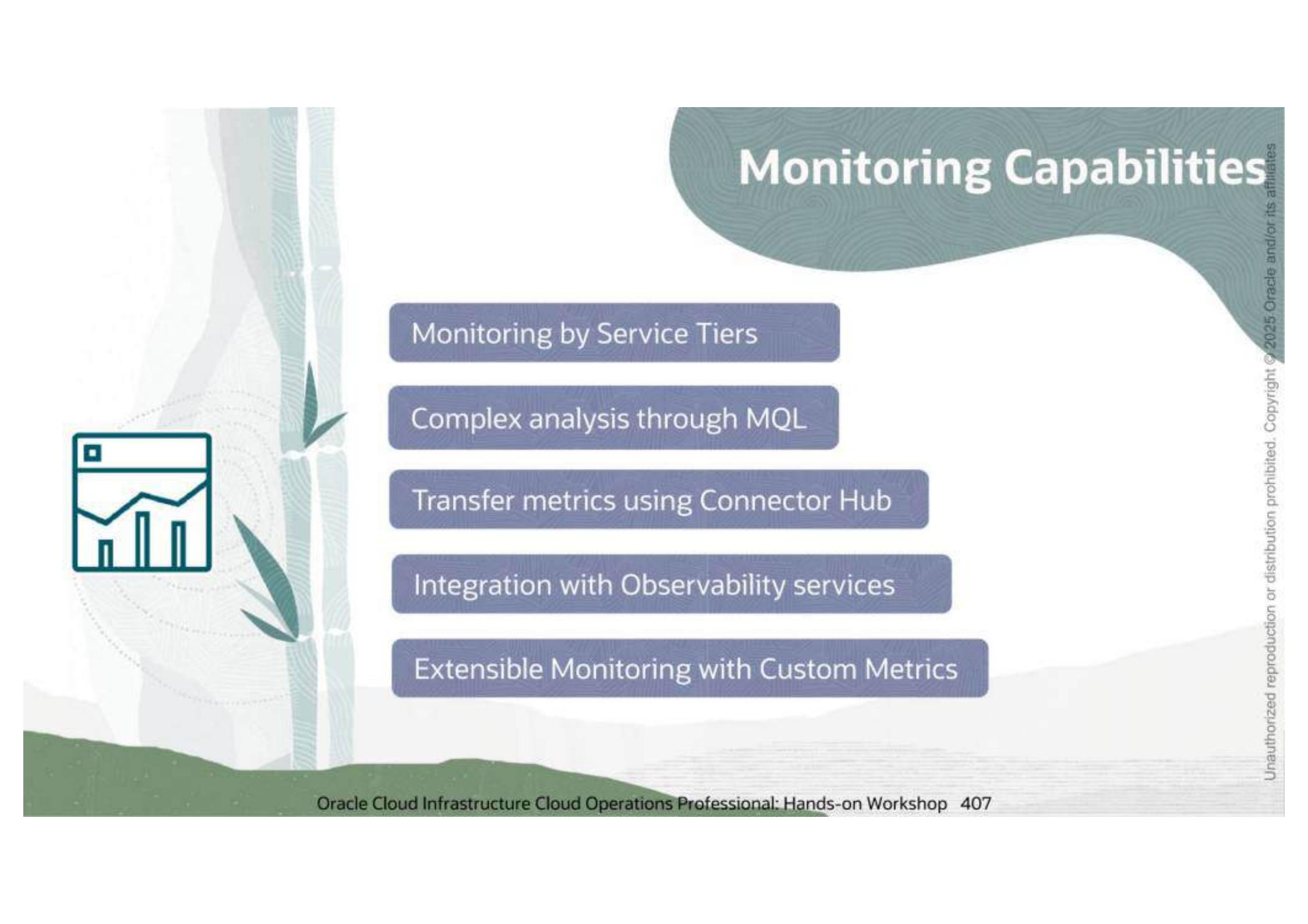
Actively and  
Passively  
Monitor Cloud  
Resources

Alarms publish  
messages to  
destinations  
managed by  
Notifications

Metrics to  
monitor cloud  
resources

Single Pane of glass  
with Dashboards

Access Metric and  
Alarm data via  
Console, CLI, API



# Monitoring Capabilities

Monitoring by Service Tiers

Complex analysis through MQL

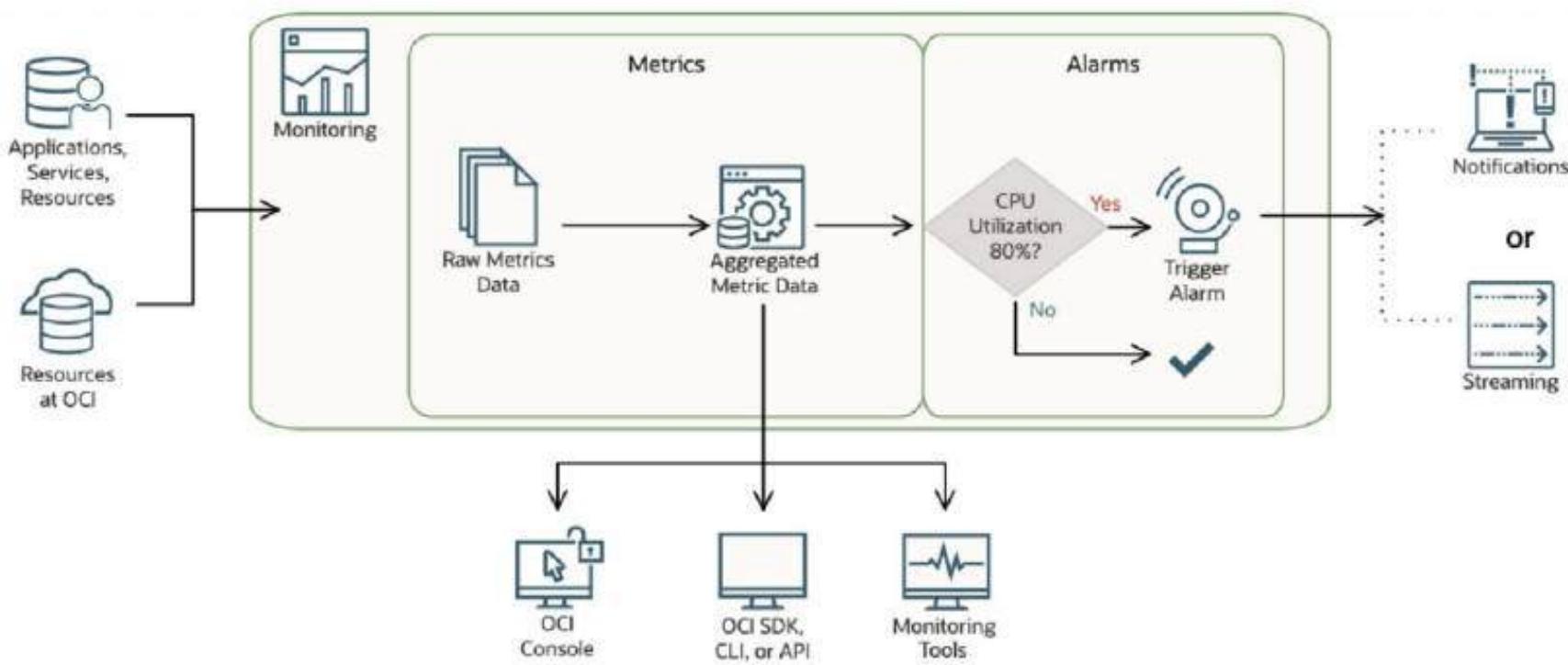
Transfer metrics using Connector Hub

Integration with Observability services

Extensible Monitoring with Custom Metrics



# Monitoring Service Workflow



Oracle Cloud Infrastructure

# Demo: Monitoring Concepts

# Oracle Cloud Infrastructure Monitoring Concepts

# Metrics

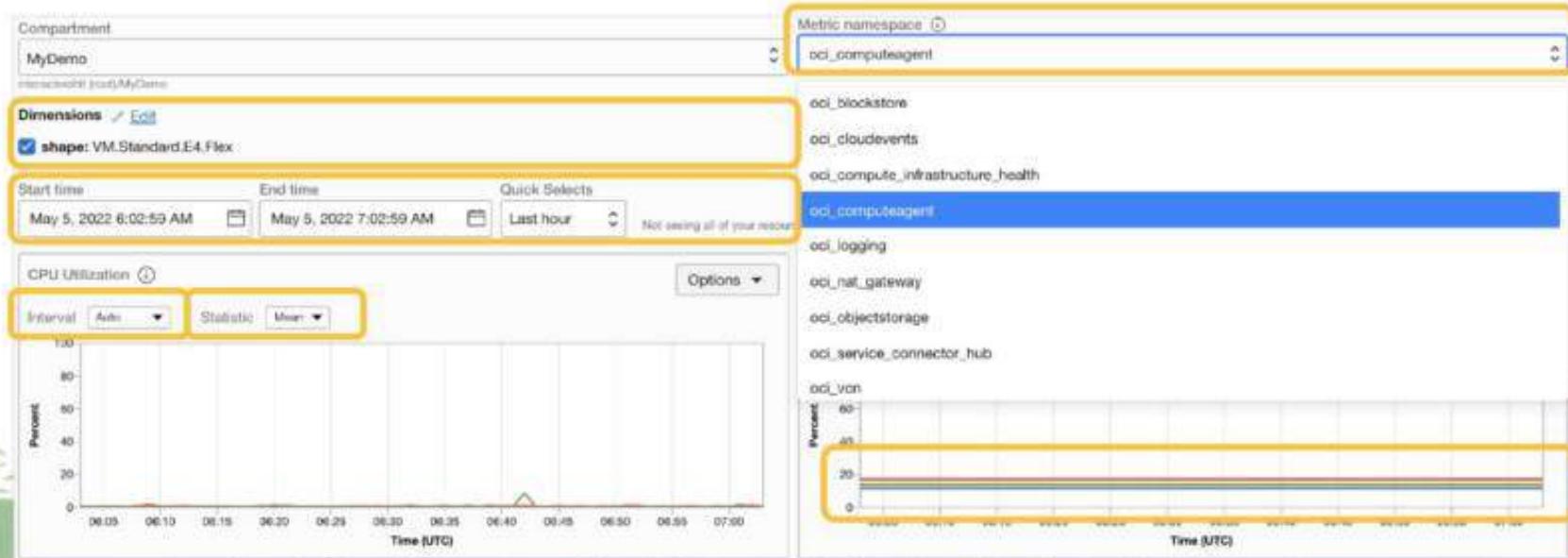
Metrics grouped within Metric Namespaces

Service Metrics | Custom Metrics (API: PostMetricData)

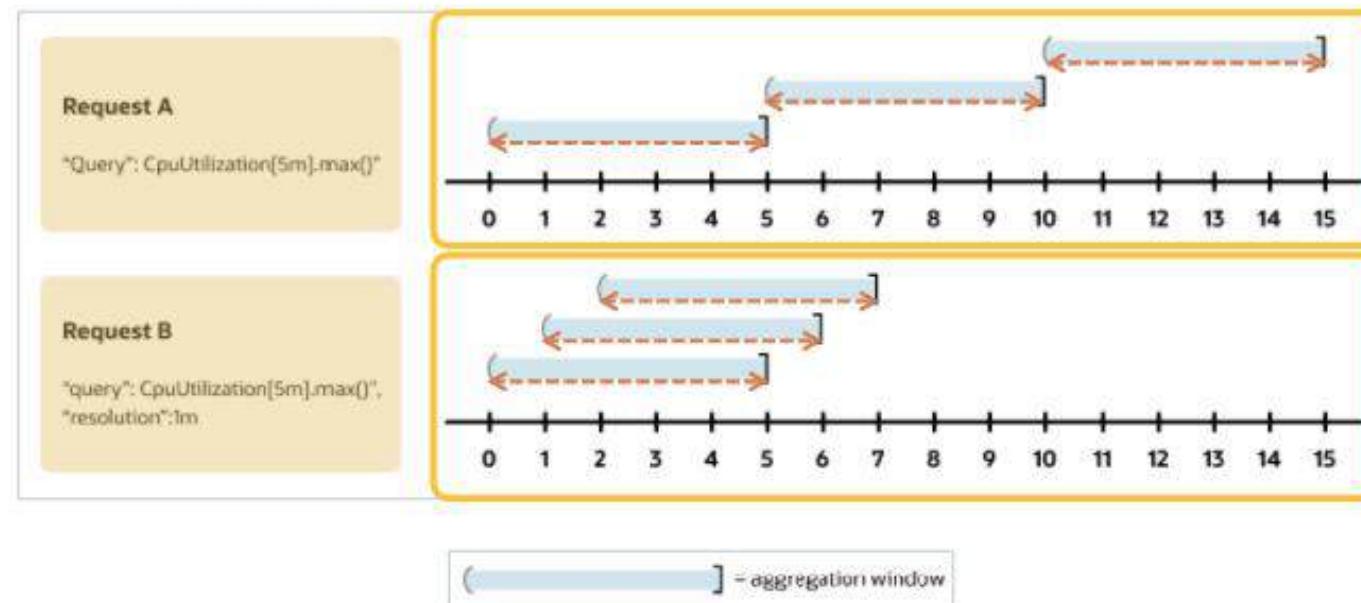
Filter by Dimensions and Time period

Data Aggregation with Interval and Statistic

Metric Streams and Grouping



# Intervals and Resolutions



# Statistics

**Sum** - Returns all values added together for the time interval

`BytesIngested[1d].sum()`

**Mean** - Returns the value of Sum divided by Count during the specified time period.

`CpuUtilization[1h].mean()`

**Count** - Returns the number of observations received in the specified time period.

`ServiceConnectorHubErrors[6h].count() > 1`

**Max** - Returns the highest value observed during the specified time period

`CpuUtilization[1m].max()`

**Percentile** - Returns the value of the given percentile during the specified time period

`CpuUtilization[1m].percentile(0.90)`

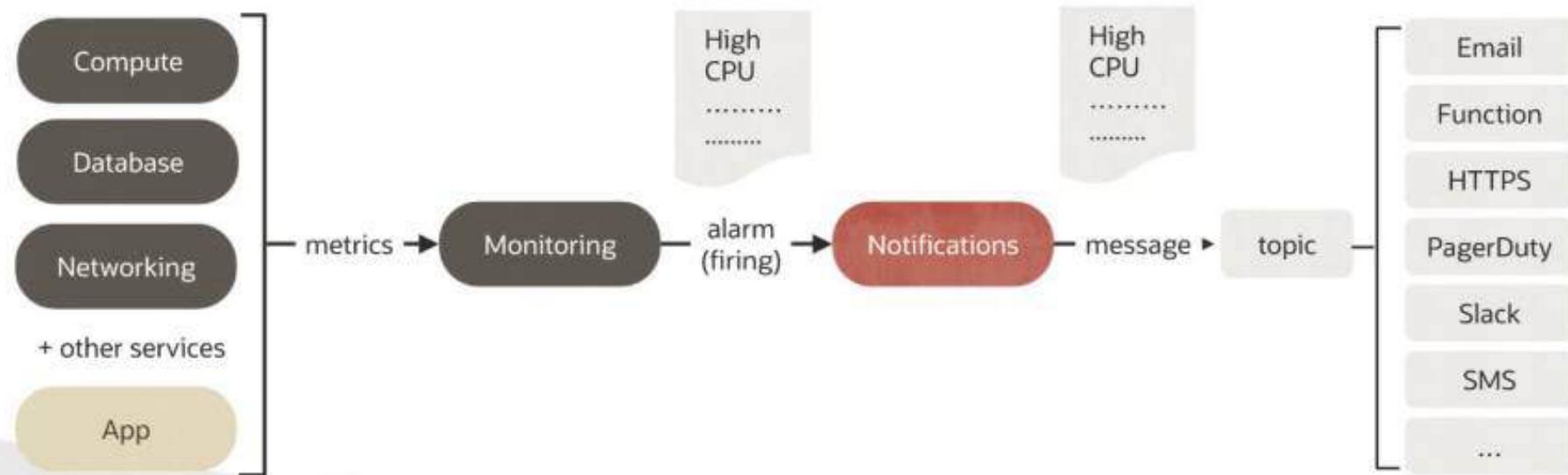
# Alarms



Alarm query must specify a metric, statistic, interval, and trigger rule (threshold or absence).

Alarm states: OK, Firing, Reset, Repeat

Suppression - Avoid Publishing Alarm Messages



# Metric Query Components



Query Component	Examples
<b>Metric</b>	<b>oci_computeagent:</b> CpuUtilization, MemoryUtilization, DiskBytesWritten, LoadAverage <b>oci_vcn:</b> VnicIngressDropsSecurityList, VnicEgressDropsSecurityList, VnicFromNetworkBytes, VnicToNetworkBytes <b>oci_blockstore:</b> VolumeReadOps, VolumeWriteOps, VolumeReadThroughput, VolumeWriteThroughput <b>oci_objectstorage:</b> ObjectCount, StoredBytes
<b>Interval</b>	1m, 5m, 1h (Basic mode) 1m-60m, 1h-24h, 1d (Advanced mode)
<b>Dimensions</b>	availabilityDomain, faultDomain, imageId, region, resourceDisplayName, resourceId, shape, projectId
<b>Grouping Function</b>	groupBy() grouping()
<b>Statistic</b>	absent(), avg(), count(), max(), min(), sum()
<b>Comparison operation (Trigger Rule)</b>	>, >=, ==, !=

# Oracle Cloud Infrastructure Notifications Service

# Notifications Service

## Overview

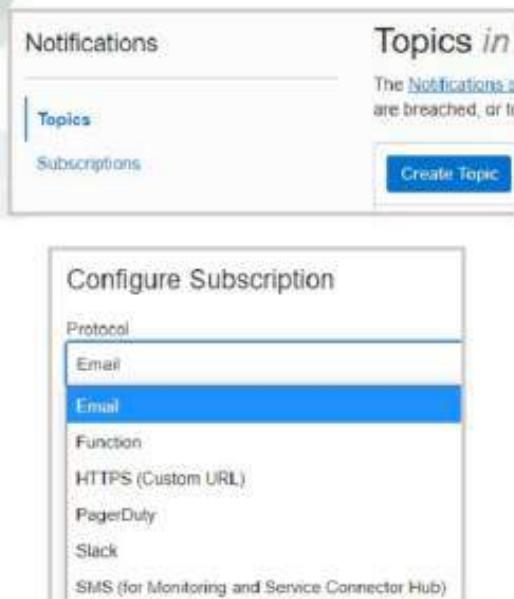
Uses a Publish-Subscribe pattern

Topics and subscriptions as channels

Configurable retry delivery duration

Integrated with OCI services





## Notifications Service: Creating a Topic

Use OCI Console, CLI, API,  
or SDKs

Configure Subscription:

Email

Function

HTTPS (Custom URL)

PagerDuty

Slack

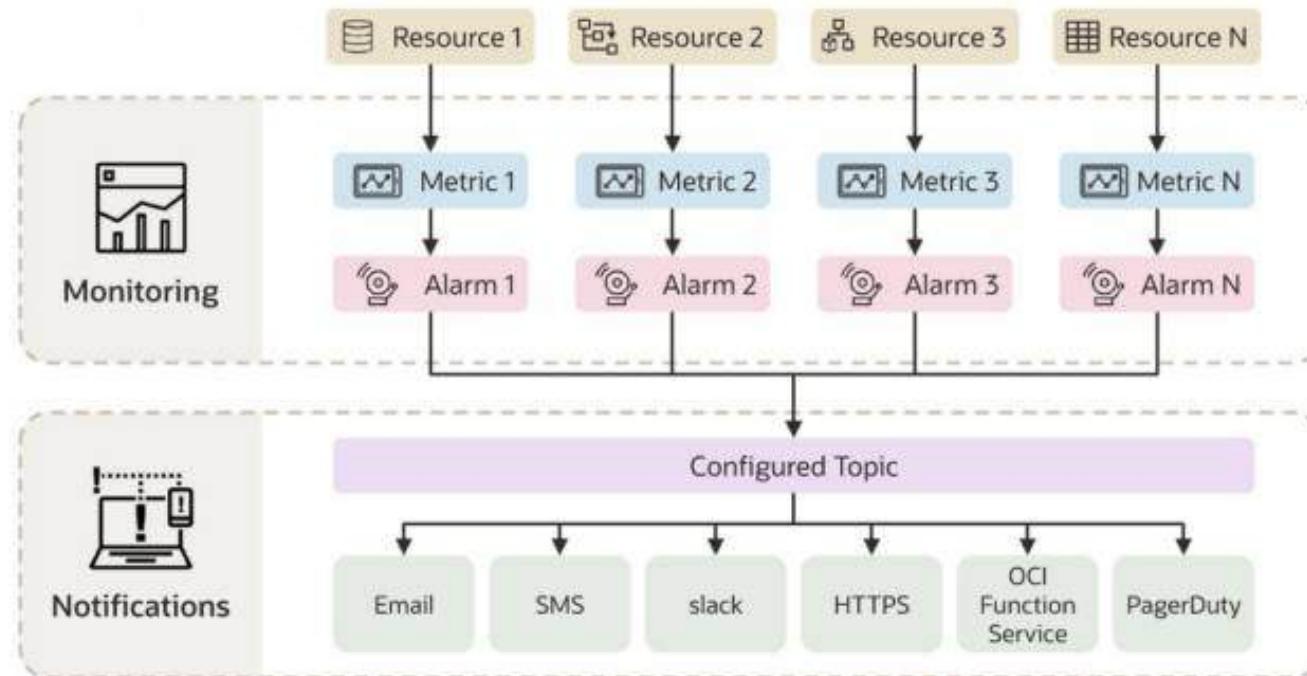
SMS

Oracle Cloud Infrastructure

# Demo: Notifications Service

# Oracle Cloud Infrastructure Alarms

# Alarms Workflow



# Best Practices



## Create a set of alarms for each metric

- ✓ At risk
- ✓ Non-optimal
- ✓ Resource is Up or Down

## Suppress alarms during mitigation

- ✓ Pause notification to avoid distractions
- ✓ Remove suppression after resolution

## Choose correct alarm interval for metrics

- ✓ Depends on frequency of emitted metric
- ✓ Determine valid intervals for each metric

## Routine tuning of alarms

- ✓ Review criticality of resource
- ✓ Review metric value fluctuations
- ✓ Assess notification methods



## Oracle Cloud Infrastructure Demo: Alarms

# Oracle Cloud Infrastructure Access and Limits

# Ways to Access Monitoring



Console, APIs, SDKs, and CLI

Metric Explorer: Basic and Advanced Mode

CLI Reference Examples

```
oci monitoring metric list --compartment-id <compartment ID>
oci monitoring alarm list --compartment-id <compartment ID>
oci monitoring metric-data post [OPTIONS]
oci monitoring alarm create [OPTIONS]
oci monitoring suppression remove --alarm-id <alarm-id>
```

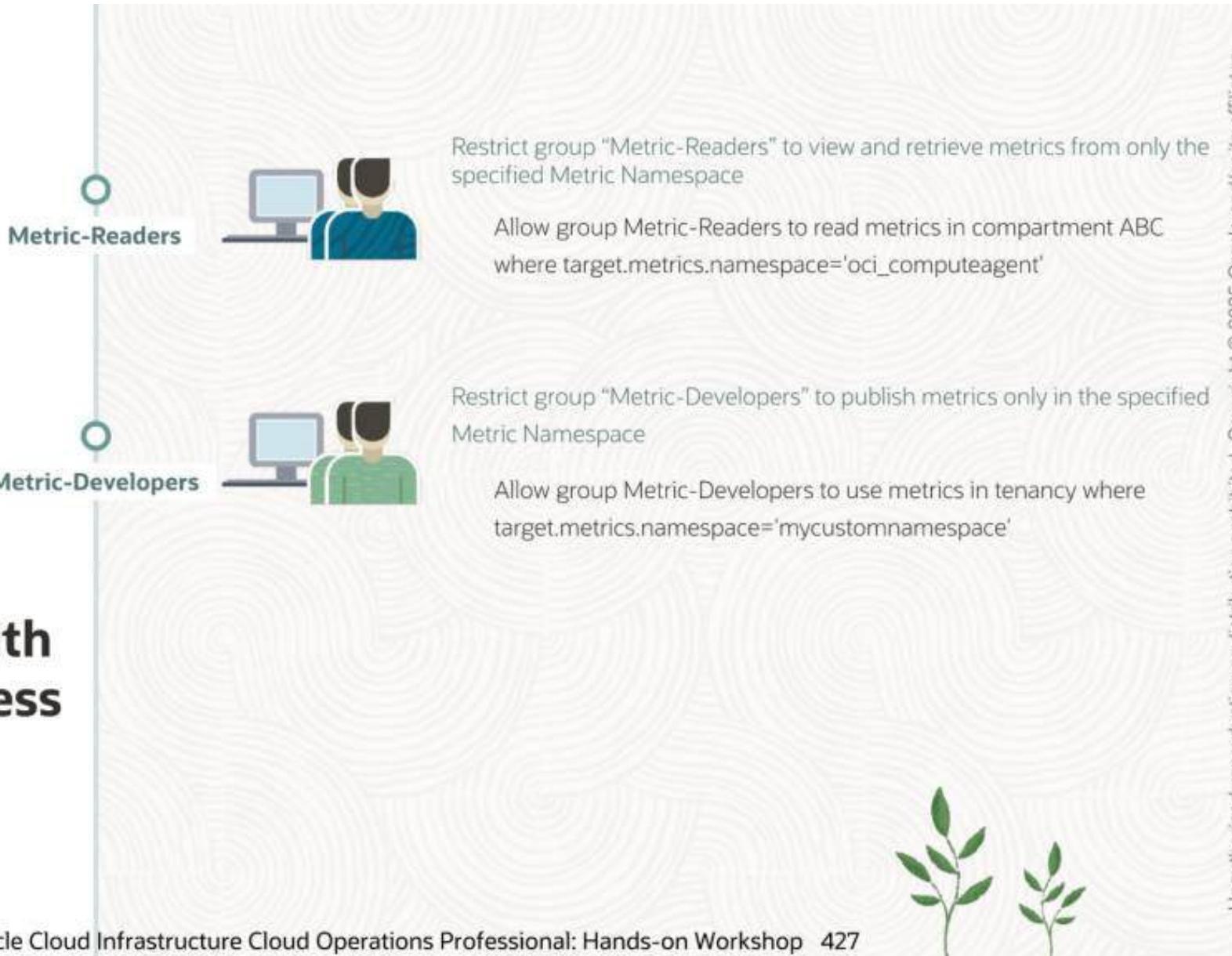
## IAM Policies for Access

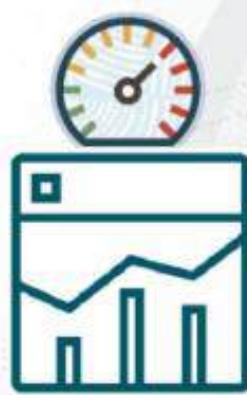


<b>Alarm-Viewers</b>		Allow group "Alarm-Viewers" to view alarms Allow group Alarm-Viewers to read alarms in tenancy Allow group Alarm-Viewers to read metrics in tenancy
<b>Alarm-Admins</b>		Allow group "Alarm-Admins" to create and manage alarms Allow group Alarm-Admins to manage alarms in tenancy Allow group Alarm-Admins to read metrics in tenancy Allow group Alarm-Admins to manage ons-topics in tenancy
<b>Metric-Readers</b>		Allow group "Metric-Readers" to view and retrieve metrics Allow group Metric-Readers to read metrics in compartment ABC
<b>Metric-Viewers</b>		Allow group "Metric-Viewers" to only view metrics Allow group Metric-Viewers to inspect metrics in compartment ABC
<b>Metric-Developers</b>		Allow group "Metric-Developers" to publish custom metrics Allow group Metric-Developers to use metrics in tenancy



## IAM Policies with Restricted Access





## Limits of Monitoring Service

Storage Limits: Metric Definitions stored up to 90 days

Storage Limits: Alarm history stored up to 90 days

Metrics posted by services are Unlimited

Limits to metric streams

# Oracle Cloud Infrastructure Metric Queries

# Building Metric Queries



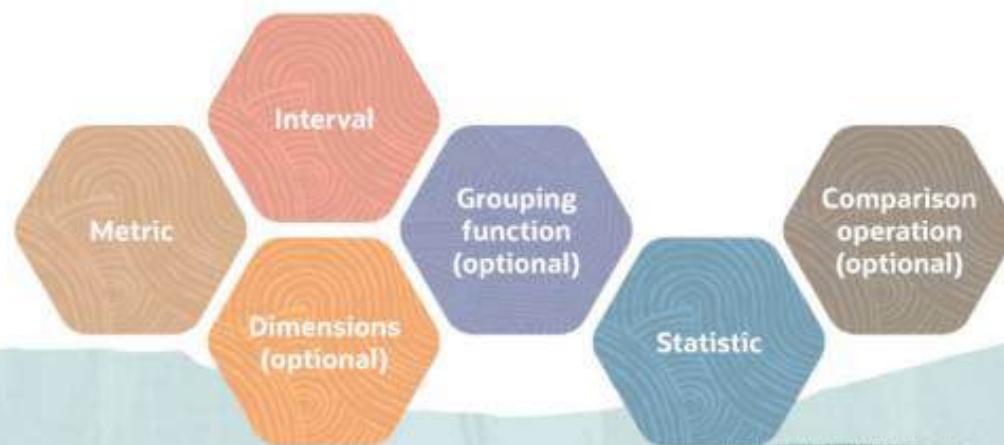
Use Basic or Advanced mode for creating queries

MQL expressions to evaluate for returning Aggregated Data

MQL syntax governs Expressions for Querying Metrics



An MQL expression (shown in Advanced Mode) includes the following components

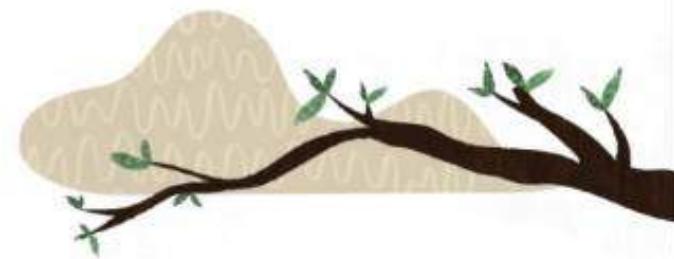


# Sample Queries



Query Syntax	Example
metric[interval].statistic	CpuUtilization[5m].max()
metric[interval]{dimensionname="dimensionvalue"}.statistic	CpuUtilization[5m]{availabilityDomain = "dKYS:AP-MELBOURNE-1-AD-1"}.max()
metric[interval].groupingfunction.statistic	CpuUtilization[5m].grouping().max()
metric[interval].groupingfunction.statistic	CpuUtilization[5m].groupBy(shape).max()

# Nested Queries



## Query Syntax

```
(metric[interval].statistic operator  
value).groupingfunction.statistic
```

## Example

```
(CpuUtilization[5m].max() > 80).grouping().sum()
```

```
(metric[interval].groupingfunction.statistic operator value).groupingfunction.statistic  
(SuccessRate[1m].groupBy(availabilityDomain).mean() < 0.99).grouping().sum()
```



## Oracle Cloud Infrastructure

# Demo: Metric Queries

## Oracle Cloud Infrastructure

# Logging Service: Overview

# OCI Logging Service

**Highly scalable and fully managed single pane of glass for all logs**

- Centralized and secure log management for all logs
- Built on open standards – leverages fluentd log ingestion, compliant with CNCF CloudEvents 1.0
- Logs accessed, searched, analyzed by using OCI Console, CLI, SDK, or REST API
- Rule-based actions on log events
- Log encryption – In Flight, Disk-level, Archived





## Types of Logs

### Service logs

- Emitted by OCI native services like VCN, Object Storage, Functions, DevOPS, etc.
- Pre-defined logging categories for all OCI Native services that can be Enabled or Disabled (One-Click)

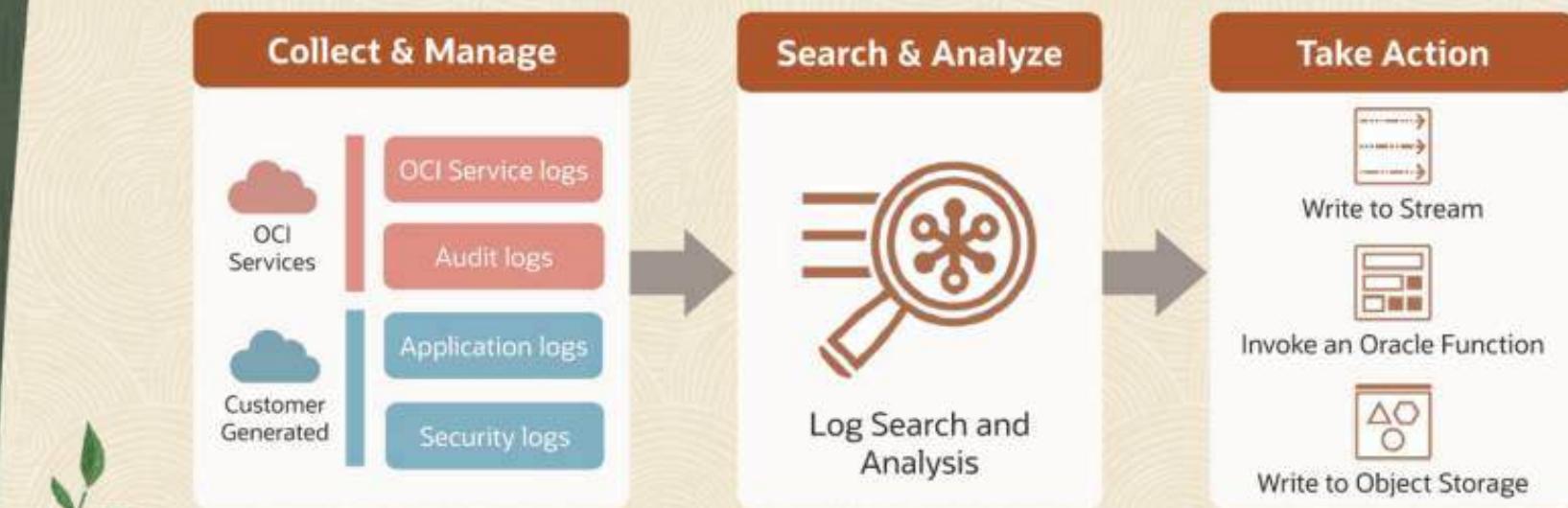
### Custom logs

- Emitted by Custom Applications running on OCI, other Cloud Providers, and On-premises Applications
- Custom Applications can ingest logs by using API or Unified Monitoring Agent

### Audit logs

- Related to Audit events emitted by OCI Audit service
- Accessed from the Console (Out-of-the-Box)

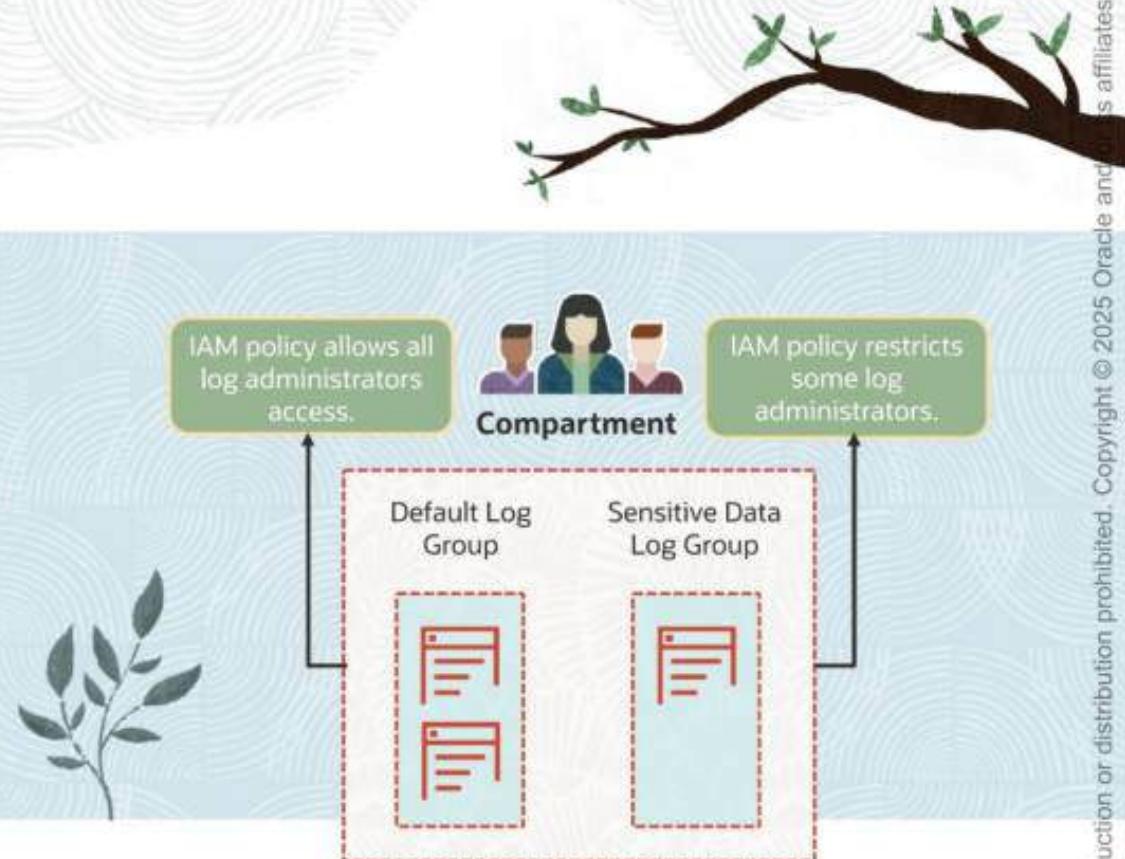
# Service Flow



# Oracle Cloud Infrastructure Logging Concepts

# Log Groups

- Collection of logs stored in a compartment
- Logical containers for logs to streamline log management
- Logs and Log groups are searchable, actionable and transportable.
- Can be used to limit access to sensitive logs by using an IAM policy



**Example:** If an application is logging sensitive information like PII, IAM policies can be used to restrict access to log data.



# Logging Concepts

## Service Log Category

Each service can have different log categories for different resources.

## Connector Hub

Helps move logging data to other services like archiving log data in object storage

## Unified Monitoring Agent

Fluentd-based agent that runs on OCI instances to help customers ingest custom logs

## Agent Configuration

Unified Monitoring Agent configuration that specifies how custom logs are ingested

## Audit

Records API calls to OCI Public API endpoints as Log Events

## Logging

[Search](#)

[Saved Searches](#)

[Logs](#)

[Log Groups](#)

[Agent Configurations](#)

[Service Connectors](#)

[Audit](#)

# Oracle Cloud Infrastructure Service Logs

## Service Log Format

```
{  
    "specversion": "1.0",  
    "type": "com.oraclecloud.devops.deployment",  
    "source": "Project name",  
    "subject": "ocid1.instance.oc1.<region_ID>.<unique_ID>",  
    "id": "e3002eaa-d717-472e-8474-d024943a0f27",  
    "time": "2020-10-18T21:02:40.58Z",  
  
    "oracle": {  
        "logid": "ocid1.log.oc1.<region_ID>.<unique_ID>",  
        "loggroupid": "ocid1.<loggroup>.ocl.<region_ID>.<unique_ID>",  
        "tenantid": "ocid1.<tenancy>.ocl..<unique_ID>",  
        "compartmentid": "ocid1.<compartment>.ocl..<unique_ID>",  
        "ingestedtime": "2020-10-18T21:02:40.58Z",  
    },  
  
    "data": {  
        "deploymentId": "ocid1.devopsdeployment.oc1.<region_ID>.<unique_ID>",  
        "deployPipelineId": "ocid1.devopsdeploypipeline.oc1.<region_ID>.<unique_ID>",  
        "deployStageId": "ocid1.devopsdeploystage.oc1.<region_ID>.<unique_ID>",  
        "message": "Manual Approval stage: Waiting for required approvals",  
        "producer": "DEVOPS_SERVICE"  
    }  
}
```

**DevOps Deployment Log: Example**

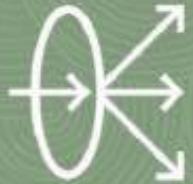


# Object Storage Logs

- Log Resource: Object Storage Buckets
- Log Categories: Write Access Events and Read Access Events
- Write event data.requestAction = PUT, POST, DELETE
- Read data.requestAction = GET, LIST, HEAD

```
    {"data": {  
        "additionalDetails": {...},  
        "apiType": "native",  
        "authenticationType": "user",  
        "bucketCreator": "ocid1.user.oc1..aaaaaaaaaaaaaa",  
        "bucketId": "ocid1.bucket.oc1.ap-melbourne-1",  
        "bucketName": "mybucket",  
        "clientIpAddress": "121.200.4.247",  
        "compartmentId": "ocid1.compartment.oc1..aaaaaaaaaaaaaa",  
        "compartmentName": "MyDemo",  
        "credentials": "ST$eyJraWQiOjIhc3dfb2Mx",  
        "eTag": "924b74ed-df34-4e91-b029-b8601b0",  
        "endTime": "2022-05-09T05:55:48.846Z",  
        "isPar": false,  
        "message": "Object uploaded.",  
        "namespaceName": "intoraclerohit",  
        "objectName": "picture100.jpeg",  
        "opcRequestId": "mel-1:STjX4rFc8rXIWE1K",  
        "principalId": "ocid1.user.oc1..aaaaaaaaaaaaaa",  
        "principalName": "tijo.thomas@oracle.com",  
        "region": "ap-melbourne-1",  
        "requestAction": "PUT",  
        "requestResourcePath": "/n/intoraclerohit",  
        "startTime": "2022-05-09T05:55:48.808Z",  
        "statusCode": 200,  
        "tenantId": "ocid1.tenancy.oc1..aaaaaaaaaaaaaa",  
        "tenantName": "intoraclerohit"}  
    }
```

**Write Access Log: Example**



## Load Balancer Logs

### Log Resource: Load Balancer

### Log Category: Access Logs

- ✓ Time when request was received
- ✓ Client and Intermediate IP addresses
- ✓ Time taken to process the request

### Log Category: Error Logs

- ✓ Time when request was received
- ✓ Error type
- ✓ Additional details related to error

### Sample Error Log Content

```
{  
  "timestamp": "2020-08-05T00:12:39+00:00",  
  "errorLog": {  
    "type": "healthChecker",  
    "errorDetails": {  
      "healthStatus": "Healthy to Unhealthy",  
      "backendSetName": "newtest",  
      "backend": "10.10.100.7:80",  
      "details": {  
        "date": 1596586352368,  
        "failures": 3,  
        "successes": 6,  
        "skips": 0,  
        "message": {  
          "msg": "connect timed out",  
          "elapsed": 3000  
        }  
      }  
    }  
  }  
}
```

### Error Log: Example



## VCN Flow Logs

- Log Resource: VCN subnet
- Log categories: ALL (Accept and Reject records)
- Troubleshoot traffic in and out of your VNICs
- Traffic accepted or rejected based on security rules
- Enable flow logs for a given subnet for all existing and future VNICs

```
{
  "datetime": 1652073462000
  "logContent": {
    "data": {
      "action": "REJECT"
      "bytesOut": 40
      "destinationAddress": "10.50.0.192"
      "destinationPort": 15911
      "endTime": 1652073462
      "flowid": "efd32869"
      "packets": 1
      "protocol": 6
      "protocolName": "TCP"
      "sourceAddress": "89.248.165.86"
      "sourcePort": 44340
      "startTime": 1652073462
      "status": "OK"
      "version": "2"
    }
    "id": "2aaf4a2"
  }
}
```

### VCN Flow Log: Example

# Oracle Cloud Infrastructure

## Demo: Service Logs

# Oracle Cloud Infrastructure Custom Logs



## Custom Log Ingestion

Logs from Custom Applications, On-Premises, External to OCI environment

Custom logs are ingested using:

- ✓ API *PutLogs*
- ✓ Unified Monitoring Agent



# Using Unified Monitoring Agent

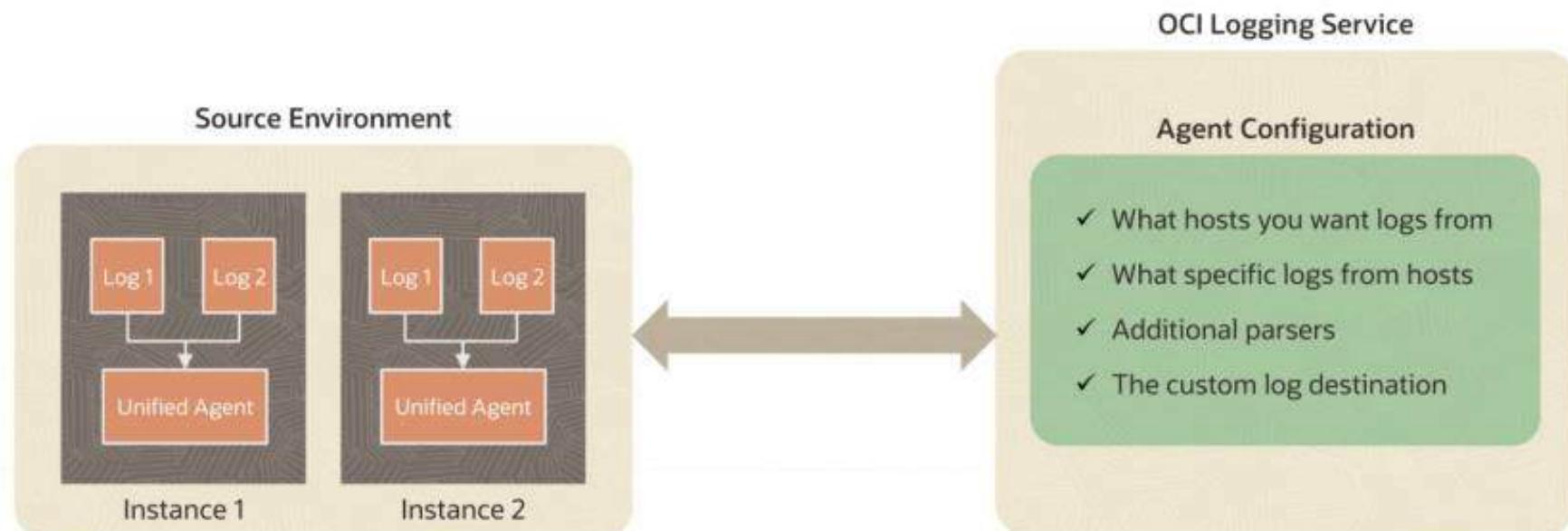
- Based on Fluentd (CNCF v1.0)
- Enabled through Oracle Cloud Agent Plugin
- “Custom Logs Monitoring” plugin is not supported on Arm-based Ampere A1 shapes
- Installed manually on external systems

## Supported Operating Systems:

- ✓ Oracle Linux 7, Oracle Linux 8
- ✓ CentOS 7, CentOS 8
- ✓ Ubuntu 16.04, Ubuntu 18.04, Ubuntu 20.04
- ✓ Windows Server 2012 R2, Windows Server 2016, Windows Server 2019

Instance information	Shielded instance	Oracle Cloud Agent
<p><a href="#">Oracle Cloud Agent</a> is a lightweight process that manages plugins running on the instance.</p>		
		<a href="#">Stop plugins</a>
	Plugin name	
	Vulnerability Scanning	(i)
	Oracle Autonomous Linux	(i)
	OS Management Service Agent	(i)
	Management Agent	(i)
	Custom Logs Monitoring	(i)
	Compute Instance Run Command	(i)
	Compute Instance Monitoring	(i)
	Block Volume Management	(i)

# Agent Communication Workflow





# Agent Configuration

- Host Groups with Dynamic Groups (OCI Instances)

IAM Policy for Dynamic Group

```
allow dynamic-group <dynamic_group_name> to use log-content in tenancy
```

Dynamic Group Rule (example)

```
ANY {instance.id = 'ocid1.instance.<region>.<location>.<unique_ID>',
      instance.compartment.id = 'ocid1.compartment.<region>..<unique_ID>')}
```



- Log Inputs with Windows Event Logs and Log Directory (Tail)
- Parsers – Auditd, JSON, CSV, Syslog, etc.
- Log Destination – Defines Compartment, Target Log Group, and Log Object

## Oracle Cloud Infrastructure

# Demo: Custom Logs

# Oracle Cloud Infrastructure Access & Explore Logs



# IAM Policies



## Allow to manage log groups or objects

Allow group Log-Admins-A to inspect log-groups in compartment XYZ

Allow group Log-Admins-B to read log-groups in compartment XYZ

Allow group Log-Admins-C to use log-groups in compartment XYZ

Allow group Log-Admins-D to manage log-groups in compartment XYZ

## Allow users to view logs

Allow group Log-Viewers-A to read log-content in compartment XYZ

## Allow to provision Agent Configuration

Allow group Log-Admins-E to use unified-configuration in compartment XYZ

## Allow instances to push logs

Allow dynamic-group production-fleet to use log-content in compartment XYZ



# Searching Logs

**Powerful tool to search indexed logs by using the Console or CLI or API**

- Search logs based on Custom, Basic, or Complex queries
- Filter values by Log Fields, Text Searches, Time Intervals
- Explore each log line with viewing raw JSON payload and before & after information.
- Export search results to a JSON file.
- Log retention up to 6 months (Service and Custom Logs)

Search

Show Advanced Mode

Custom filters

Enter search filters

Select logs to search

Filter by time

Start Date: Mar 1, 2021 3:01:11 AM

End Date: Mar 2, 2021 6:01:11 AM

Reset Search Save Search Create Service Connector Search

Explore Visualize Autorefresh OFF

Number of Log Events Per Hour

Hour	Events
03:00	100
04:00	90
05:00	85
06:00	75
07:00	45
08:00	60
09:00	55
10:00	40
11:00	50
12:00	110
13:00	100
14:00	80
15:00	90
16:00	40
17:00	110
18:00	80
19:00	90
20:00	70
21:00	110
22:00	60
23:00	50
00:00	80
01:00	100

Log Data

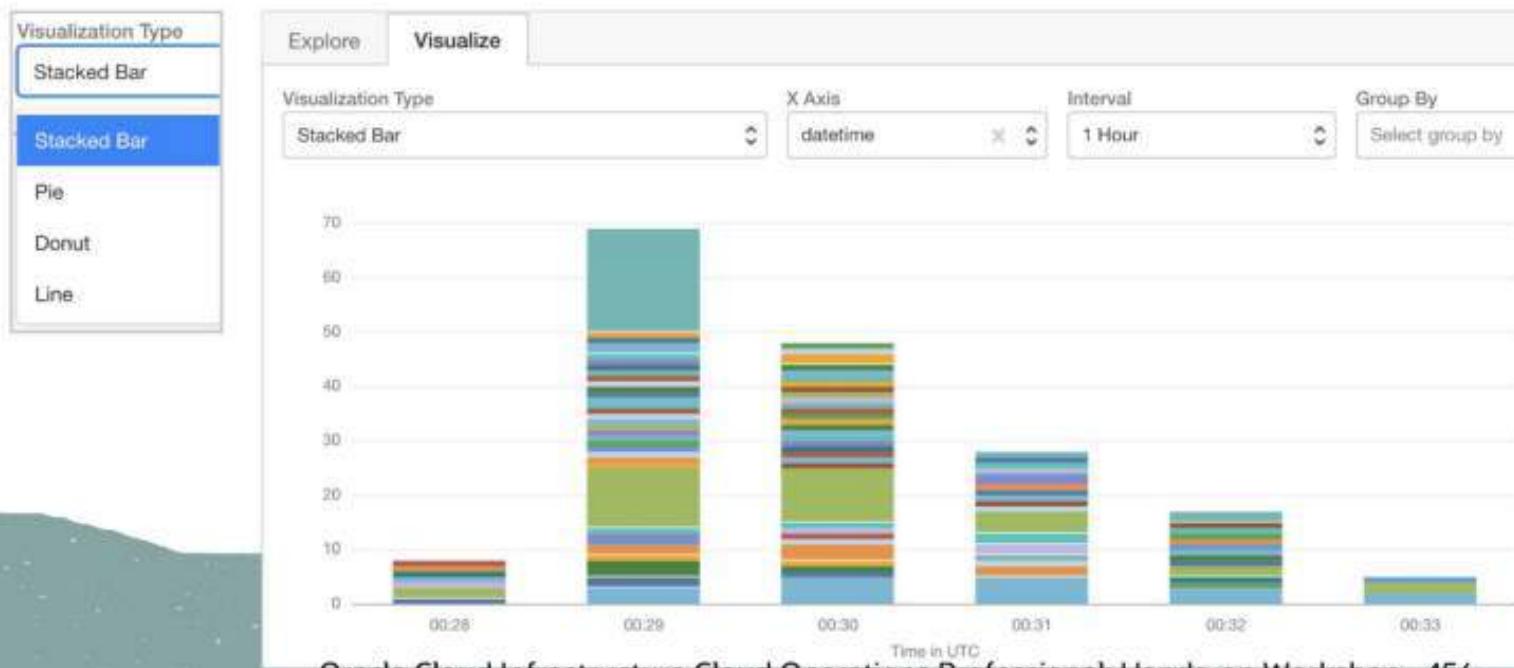
Tue, Mar 2, 2021, 06:00:06 UTC InstanceAgentService PollInstanceAgentCommands su...

Tue, Mar 2, 2021, 05:58:18 UTC InstanceAgentService PollInstanceAgentCommands su...

## Viewing Log Events

Visualize log data in various chart widgets like bar charts, pie charts, donut charts, and line charts.

Auto-refresh data to get the most recent logs in real time.



# Oracle Cloud Infrastructure Logging Queries

# Log Search



- Basic and Advanced Search using Console
- Search across multiple logs, data sources across multiple regions
- Filter using log fields, text search, and time intervals
- Search Logs using API **SearchLogs**
- Search Logs using Command Line Interface (CLI)

```
oci logging-search search-logs --search-query, --time-start --time-end
```

```
oci logging-search search-logs --search-query 'search  
"ocid1.tenancy.oc1..<unique_ID>"' --time-start 2021-06-22T00:30:00Z --time-end  
2021-06-22T00:35:00Z
```



# Logging Query Specification



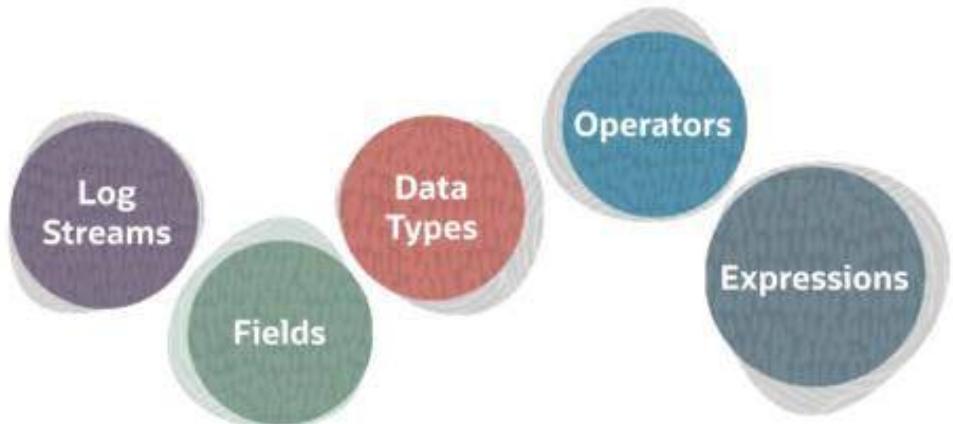
Queries processed using a Data Flow model



Searching, Filtering, and Aggregating Logs



Logging Query includes:



# Log Streams



Defines the set of logs to be searched

```
<log_stream> := "<compartment_ocid> ( /<log_group_ocid> ( /<log_object_ocid> )? )?"
```

## Examples

- search "compartmentOcid"
- search "compartmentOcid/logGroupNameOrOcid"
- search "compartmentOcid/logGroupNameOrOcid/logNameOrOcid"
- search "compartmentOcid/logGroupNameOrOcid/logNameOrOcid",  
"compartmentOcid\_2/logGroupNameOrOcid\_2"

# Fields



Represent the fields to be searched within log streams

Are case-sensitive

Fields: <field\_name> := <identifier> (DOT <identifier>)\*

## Examples

- type = 'com.oraclecloud.vcn.flowlogs.DataEvent'
- data.action != 'ACCEPT'
- data.message = 'Get Instance failed'
- data.resourceId = 'ocid1.instance.oc1...'

# Data Types



Search with strings, Numbers, Arrays, Booleans, Timestamps, and Intervals.

## Examples:

Type	Examples
string	'hello', 'world\'t'
wildcard pattern	"acc-*"
integer	-1, 0, +200

# Tabular Operators

- **search**
- **where**
- **top**
- **sort**
- **dedup**
- **select**

```
search "application" | where level = 'ERROR'
```

```
search "application" | top 3 by impact
```

```
search "application" | sort by impact desc
```

```
search "application"  
      | select level, host, impact+10 as impact, timestamp  
=>  
{"level":"ERROR", "host":"host1", "impact": 12, "timestamp": "2019-01-03T00:04:01"}  
{"level":"INFO", "host":" host1", "timestamp": "2019-01-03T00:04:05"}  
{"level":"WARNING", "host":"host2", "impact": 11, "timestamp": "2019-01-03T00:05:33"}  
{"level":"ERROR", "host":"host2", "impact": 14, "timestamp": "2019-01-03T00:06:39"}  
{"level":"ERROR", "host":"host2", "impact": 15, "timestamp": "2019-01-03T00:06:59"}  
{"level":"INFO", "host":" host2", "timestamp": "2019-01-03T00:06:59"}
```

# Scalar Operators



## Arithmetic Operators

- +
- -
- \*
- /

## Boolean Operators

- and
- or

## Comparison Operators

- <expr> > <expr>
- <expr> >= <expr>
- <expr> <= <expr>
- <expr> < <expr>
- <expr> = <expr>
- <expr> != <expr>

## Examples

```
search "ocid1.compartment.oc1..aaaaaaaaawqegmjifhni77bqm625cxioavoq775jckfn2syxqtmqliqydw5dq" | logContent='*10.50.0.220*' | sort by datetime desc
```

```
search "ocid1.compartment.oc1..aaaaaaaaawqegmjifhni77bqm625cxioavoq775jckfn2syxqtmqliqydw5dq" | (data.requestAction='DELETE' or data.requestAction='PUT') and data.bucketName='ProductionBucket' | sort by datetime asc
```

Oracle Cloud Infrastructure

# Demo: Logging Queries

# Oracle Cloud Infrastructure Connector Hub

# Overview and Key Concepts

Fully managed service that orchestrates data movement between source and destination OCI services

The screenshot shows the 'Create Service Connector' page in the Oracle Cloud console. It includes sections for 'Configure Service Connector' (Source: Logging, Target: Streaming), 'Configure source connection' (Log Group: Audit, Log: Audit Log), 'Configure Task' (Filter Type: Event type, Service Name: Database, Event Type: Autonomous Container Database - Maintenance Worker), and 'Configure target connection' (Stream ID: E2E1597780058251). A 'Query Syntax' section displays a complex SQL-like query for filtering logs.

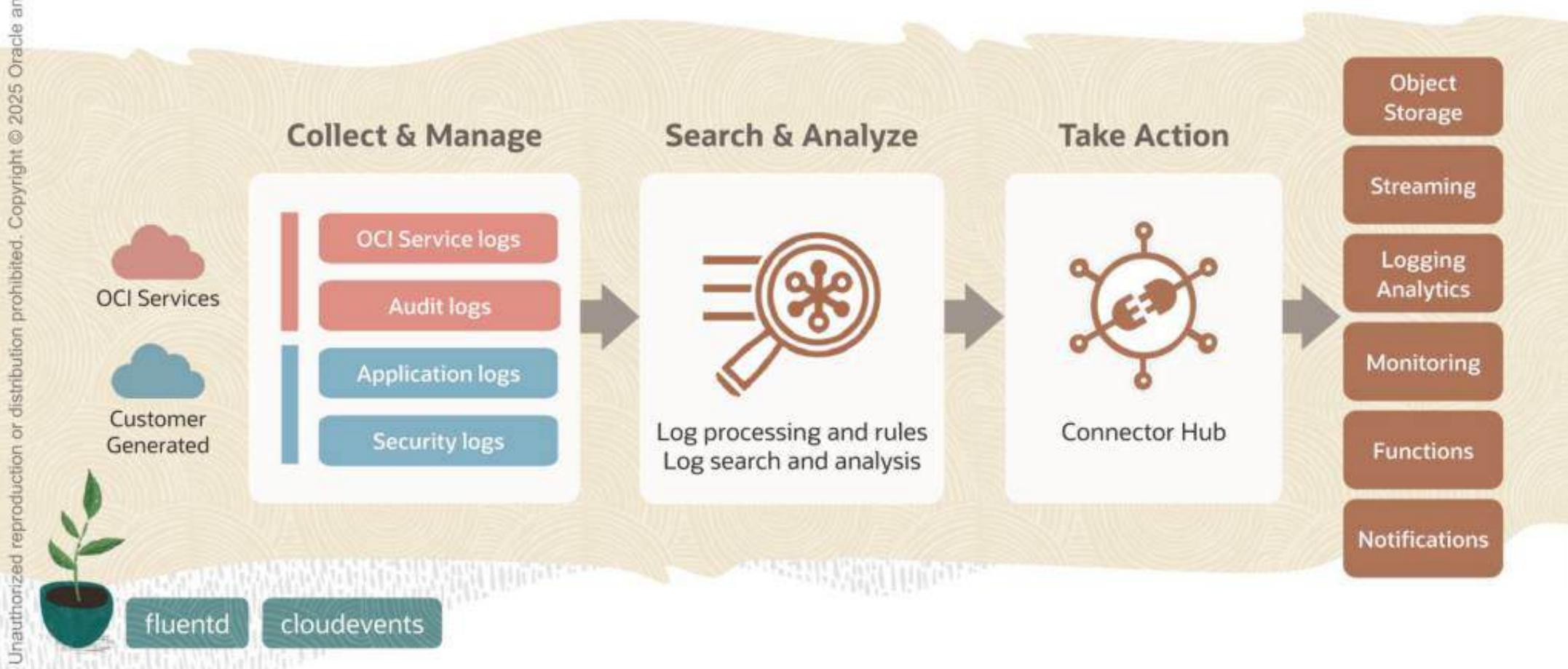
```
search * AND ( compartment_name = 'us-west-1' AND log_name = 'audit' ) WHERE type = 'com.oracle.database.logfilechange.event' AND timestamp > '2023-01-01T00:00:00Z'
```

**Source:** The service that contains the data to be moved

**Target:** The service that receives data from the source

**Task:** Optional filtering to data before moving to target

# Connectors Workflow



# Take Actions for Use Cases

Integrating with other Oracle services



Write to Stream



Archive to object storage



Invoke Oracle function



OCI Monitoring metric



Notification support

& more!



One click data archival to object storage

Easily emit log metrics to OCI Monitoring

Automated remediation and alerting using Oracle functions & notifications

Facilitate deeper data analysis by ingesting data into Logging Analytics

Move data to third-party services