# Cloud Native Architectures

**Workload Approach Workshop**

# Introduction

Do you consider…

❑ **Adopting a cloud native enterprise approach?**

❑ **Automating on-demand services?**

❑ **Utilizing cloud based tools? And think they change so much?**

❑ **Building and running apps that are highly available, resilient, fully secure, and compliant?**

❑ **Modernizing and simplifying your deployments and operations?**

The intention of this workshop is to facilitate and support you to adopt modern and cloud native principles. In a 3 stage approach we will lead from discovery to a verified and validated cloud native solution architecture.

# Objectives
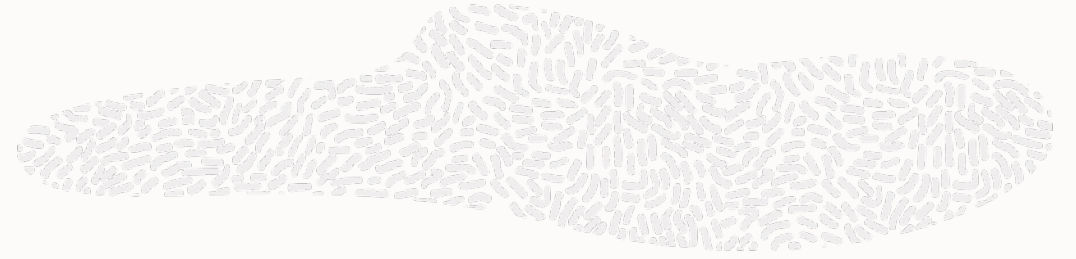
During this workshop we will discuss

- ❑ **Design principles** and **best practices** that govern application architectures
- ❑ **The core requirements** that apply to your application
- ❑ **Recommend technologies** for modern applications
- ❑ Showcase similar **customers use cases**
- ❑ Which **architecture pattern** fits your situation and the necessary steps to get there
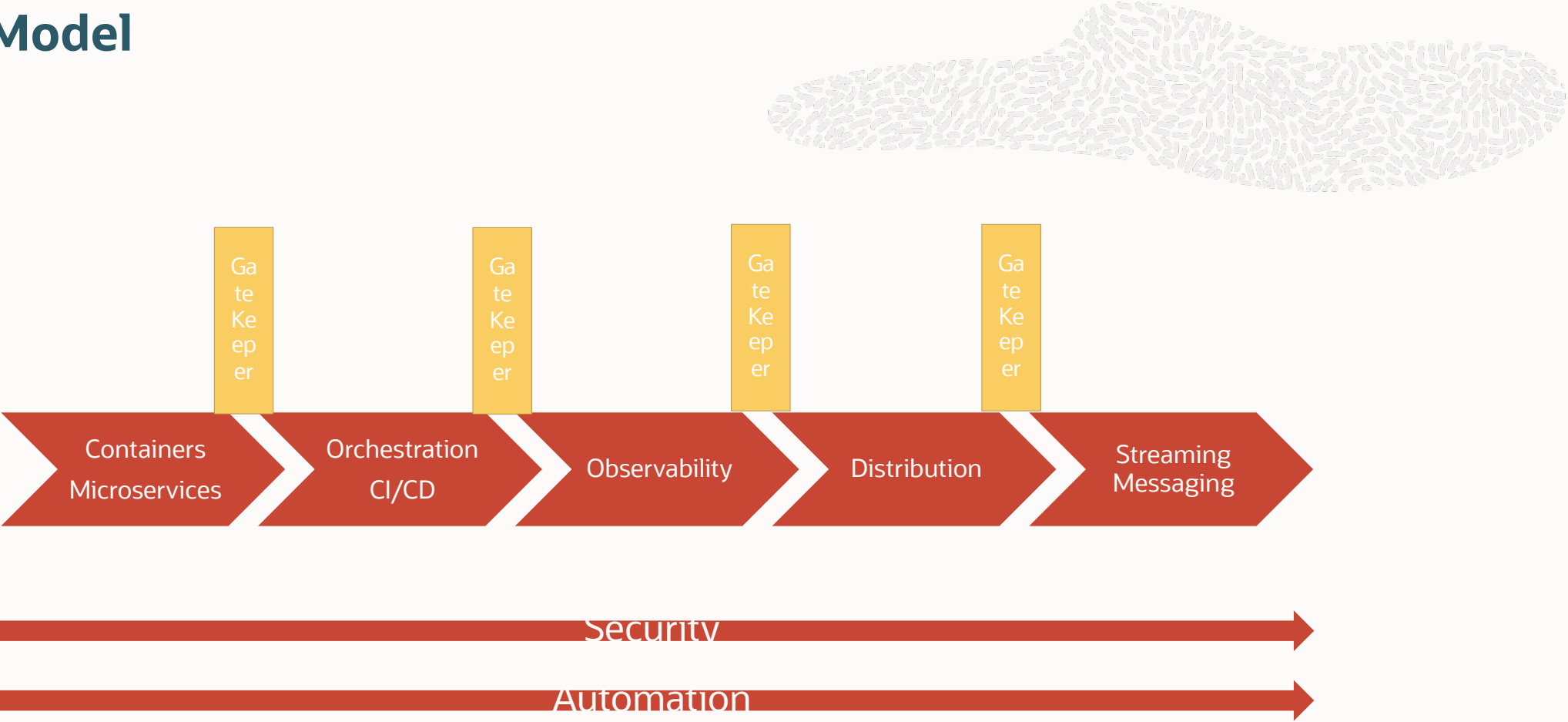
We would also like to discuss

- ❑ What are the **workloads You consider moving to cloud**
- ❑ How **can Oracle help** you with this journey with **Oracle's Lift Service**
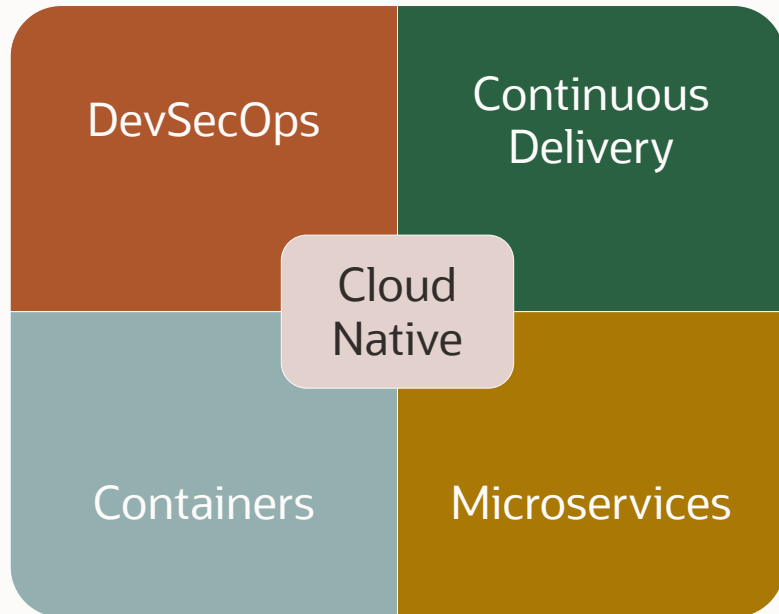
# Discovery

# Maturity Model

# The journey

# What is Cloud Native?



## CNCF Definition

Cloud native technologies empower organizations to build and run **scalable** applications in modern, dynamic environments such as public, private, and hybrid clouds. **Containers**, service meshes, microservices, immutable infrastructure, and **declarative APIs** exemplify this approach.

These techniques enable **loosely coupled systems** that are **resilient**, **manageable**, and **observable**. Combined with robust automation, they allow engineers to make high-impact changes **frequently** and **predictably** with minimal toil.

# Cloud Native Architecture
## Strategic Pillars

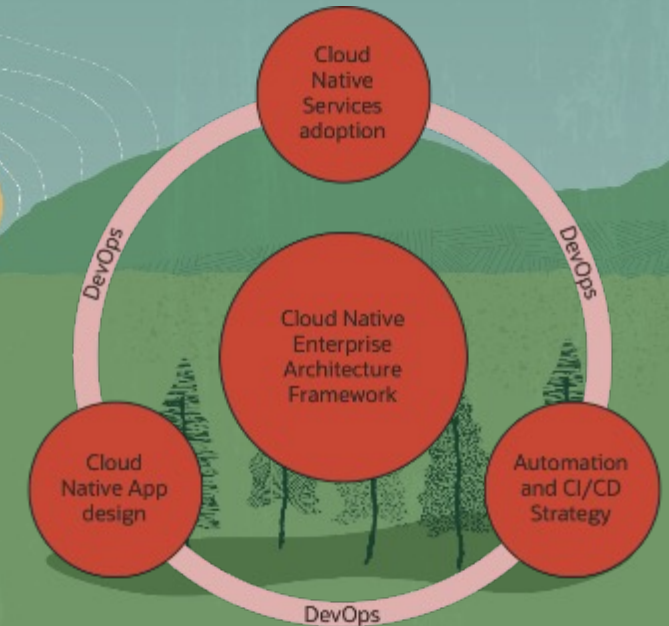**Cloud Native Enterprise Architecture Framework**
Aims at the ensuring a solid, accepted and implemented Enterprise Architecture aiming for a fully Cloud Native landscape and defining the target state for all the components in the IT landscape.

**Cloud Native Service Adoption**
Aims at ensuring the adoption of cloud native IaaS, PaaS and SaaS services over the adoption and inclusion of customer managed solutions to ensure optimal use and benefit of cloud based managed services.
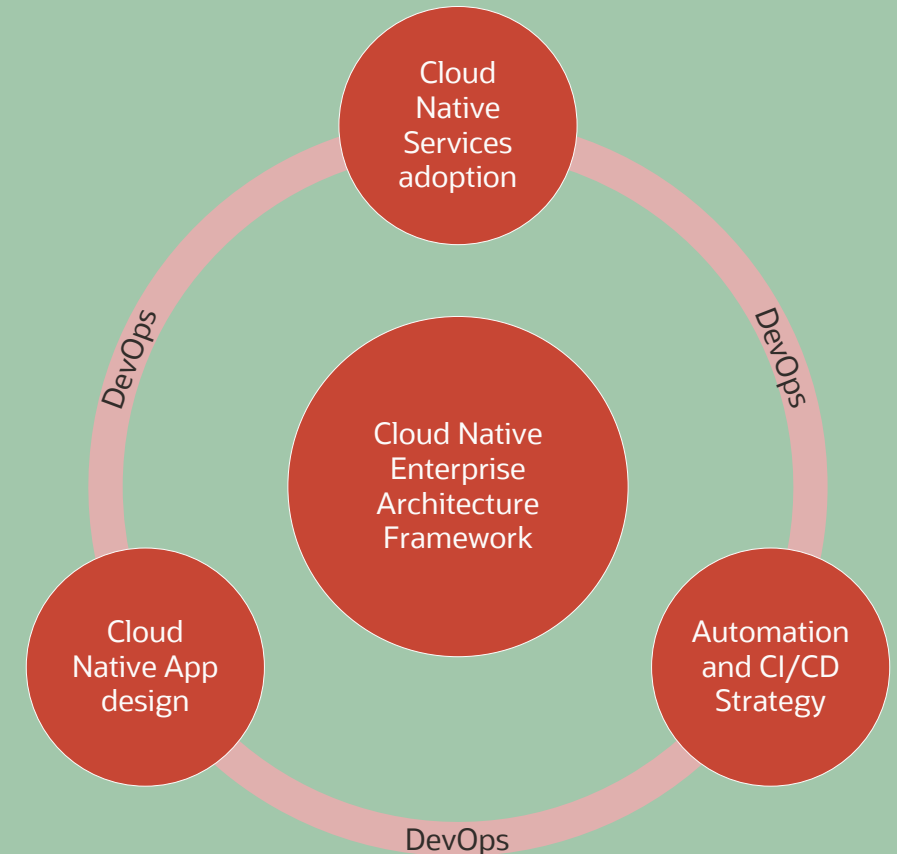
**Cloud Native Application Design**
Aims at ensuring the design and build of applications in a true cloud native manner by making use of cloud native design patterns and best practices while staying true to the Enterprise Architecture Framework.

**Automation and CI/CD Strategy**
Aims at ensuring to remove as many as possible human actions, embedding CI/CD strategies, automation solutions and self-managing system strategies to ensure more secure systems, remove toil and lower operational costs
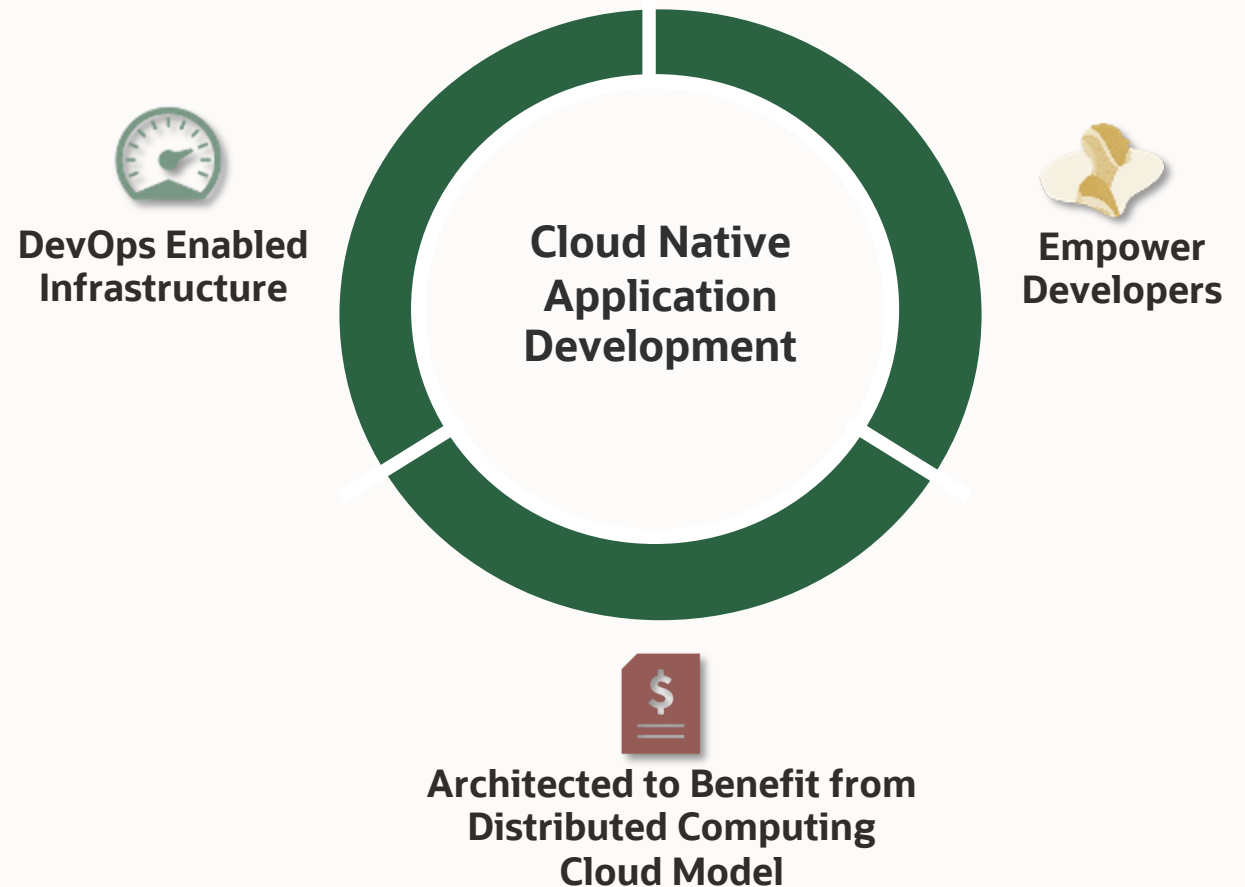
**Agile DevOps**
Aims at building, improving and supporting a DevOps / BusDevOps culture and way of working within all departments of the enterprise.

Cloud Native Services adoption

DevOps

DevOps

Cloud Native Enterprise Architecture Framework

Cloud Native App design

Automation and CI/CD Strategy

DevOps

**Cloud Native** - Oracle Architecture Services
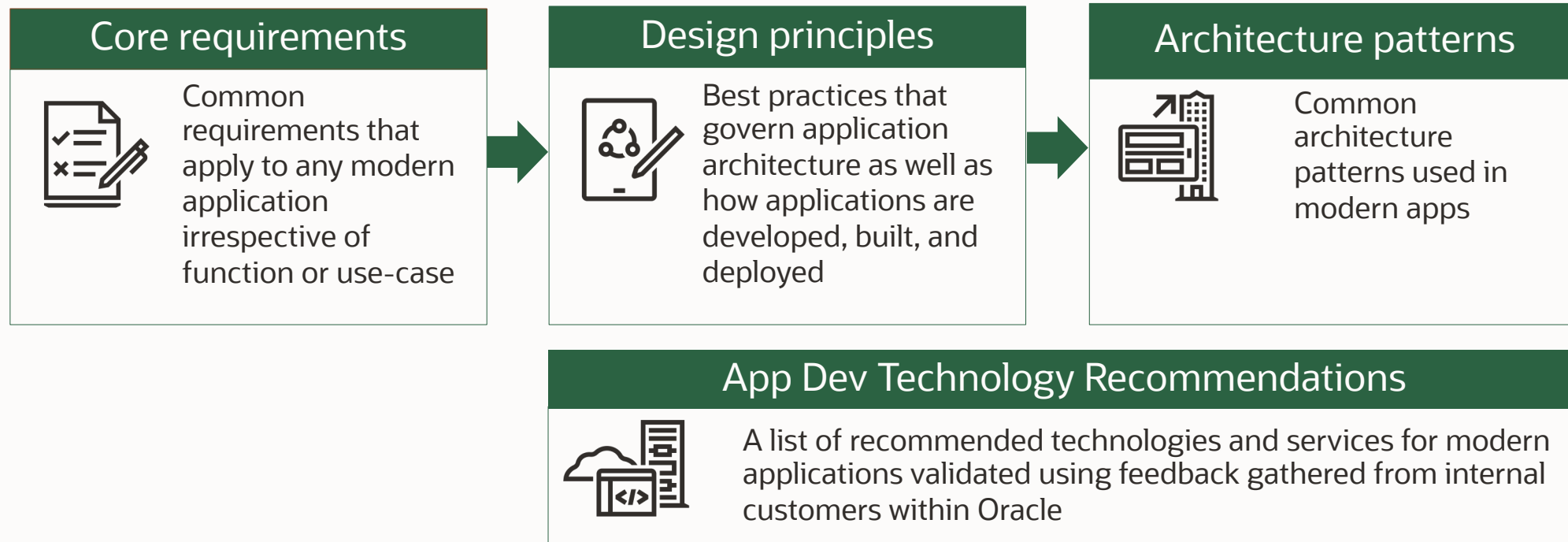
# What are Cloud Native apps?

- Agile CI/CD Development
- Standards based
- Choice for Developers
- Designed and developed to benefit from cloud model
- Often containerized, loosely coupled, event-driven

**DevOps Enabled Infrastructure**

**Cloud Native Application Development**

**Empower Developers**

**Architected to Benefit from Distributed Computing Cloud Model**

# Build New Cloud Native Apps
Simplify the architectural decision-making

## Oracle's Modern Application Development approach

| Core requirements |
|---|
| Common requirements that apply to any modern application irrespective of function or use-case |

| Design principles |
|---|
| Best practices that govern application architecture as well as how applications are developed, built, and deployed |

| Architecture patterns |
|---|
| Common architecture patterns used in modern apps |

| App Dev Technology Recommendations |
|---|
| A list of recommended technologies and services for modern applications validated using feedback gathered from internal customers within Oracle |

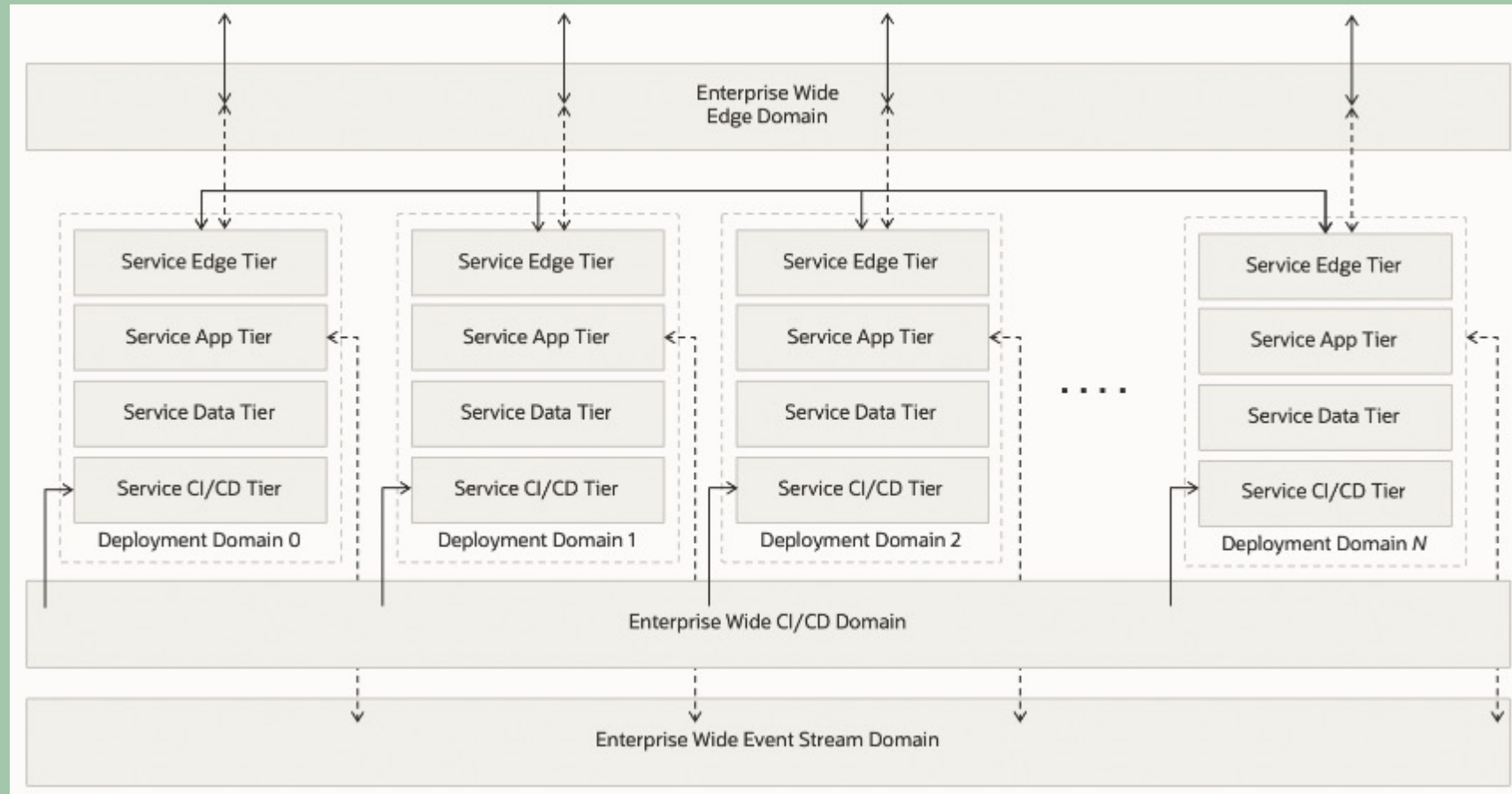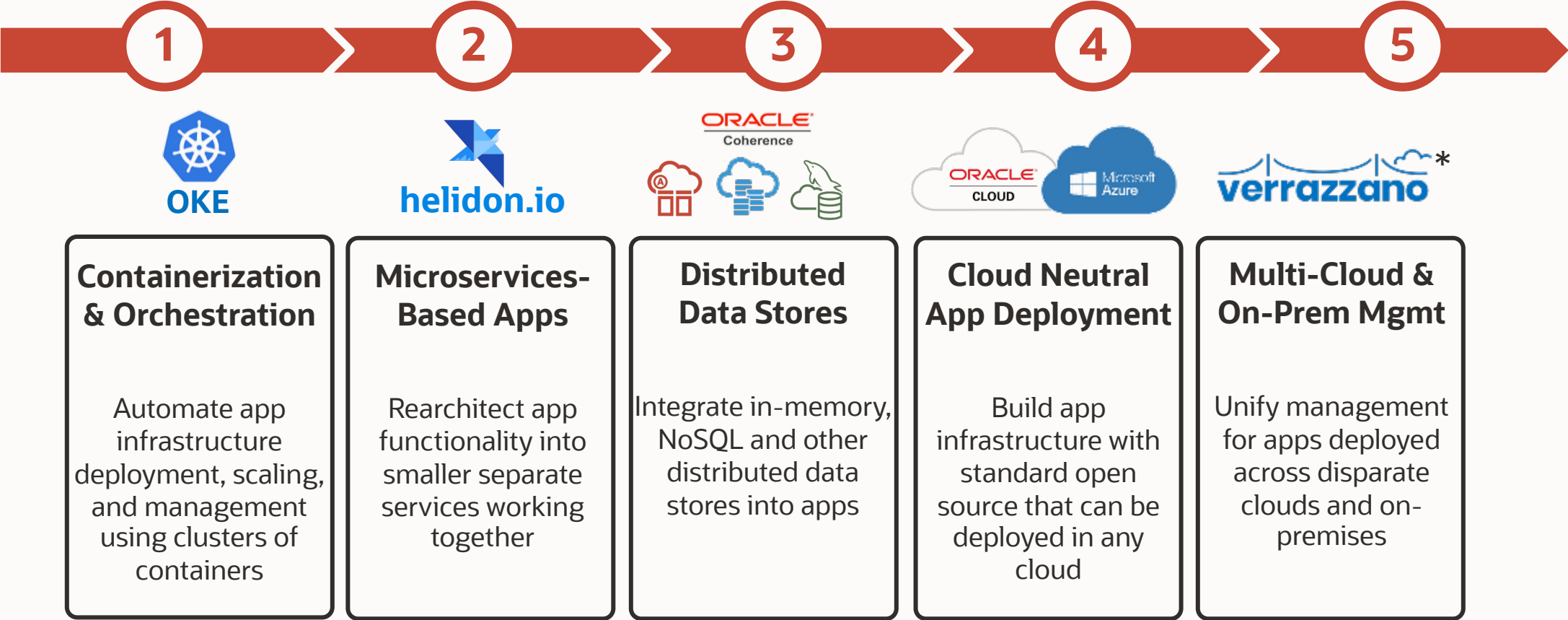https://www.oracle.com/cloud/architecture-center/modern-app-development/

# Oracle's Modern Application Design Principles

Use lightweight open-source frameworks and mature programming languages

Build apps as services that communicate through APIs

Package and ship apps as containers

Automate build, test, and deployment

Use fully managed services to eliminate complexity across application development, runtimes and data management

Keep application tier stateless

Use converged databases with full featured support across all data

Instrument end-to-end monitoring and tracing

Eliminate single points of failure through automated data replication and failure recovery

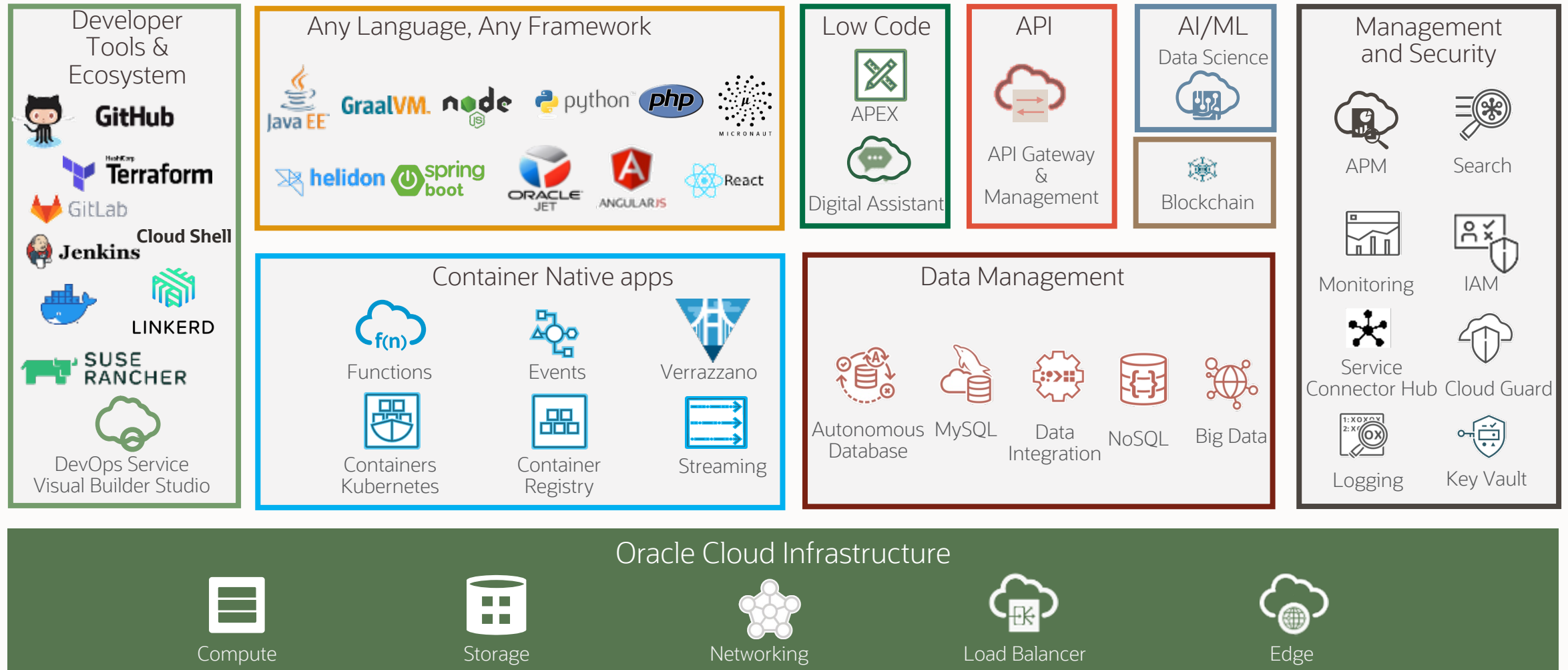Implement a defense-in-depth approach to secure the app lifecycle

# Design Pattern Implementation

# Key Technologies of the Modern Application



| | | | | |
|---|---|---|---|---|
| **1** OKE | **2** helidon.io | **3** ORACLE Coherence | **4** ORACLE CLOUD / Microsoft Azure | **5** verrazzano* |
| **Containerization & Orchestration** | **Microservices-Based Apps** | **Distributed Data Stores** | **Cloud Neutral App Deployment** | **Multi-Cloud & On-Prem Mgmt** |
| Automate app infrastructure deployment, scaling, and management using clusters of containers | Rearchitect app functionality into smaller separate services working together | Integrate in-memory, NoSQL and other distributed data stores into apps | Build app infrastructure with standard open source that can be deployed in any cloud | Unify management for apps deployed across disparate clouds and on-premises |

*Available today in Community Edition

# Application Development with Oracle Cloud

## Developer Tools & Ecosystem
GitHub
Terraform
GitLab
Cloud Shell
Jenkins
Docker
LINKERD
SUSE RANCHER
DevOps Service
Visual Builder Studio

## Any Language, Any Framework
Java EE
GraalVM
node JS
python
php
MICRONAUT
helidon
spring boot
ORACLE JET
ANGULARJS
React

## Low Code
APEX
Digital Assistant

## API
API Gateway & Management

## AI/ML
Data Science
Blockchain

## Management and Security
APM
Search
Monitoring
IAM
Service Connector Hub
Cloud Guard
Logging
Key Vault

## Container Native apps
Functions
Events
Verrazzano
Containers Kubernetes
Container Registry
Streaming

## Data Management
Autonomous Database
MySQL
Data Integration
NoSQL
Big Data

## Oracle Cloud Infrastructure
Compute
Storage
Networking
Load Balancer
Edge

# What's the best fit for you ?

|   | On-premises | OCI | Other Cloud |
|---|---|---|---|
| 1 |  | CI/CD · Applications Containers · Databases |  |
| 2 | CI/CD | Databases | Applications Containers |
| 3 | Databases | Databases · Applications Containers | Applications Containers · CI/CD |
| 4 | Applications Containers | Databases · CI/CD | Applications Containers |