



# Desarrollo de Aplicaciones Cloud Native

Innove e incrementa la velocidad de entrega y valor mediante aplicaciones Cloud Native usando DevOps

Francisco Moreno

Cloud Solution Engineer  
LAD Cloud Tech Knowledge  
23 Septiembre, 2022





# Agenda

1 h

Arquitecturas para Aplicaciones Modernas

Visión Oracle para DevOps y Cloud Native

## Herramientas

- CI/CD: DevOps Services
- IaC: Resource Manager (Terraform)

## Servicios

- Oracle Kubernetes Engine (OKE)
- Functions
- API: API Gateway
- DevSecOps: Herramientas variadas

**No hay nada tan estable como el cambio.**

- Bob Dylan -

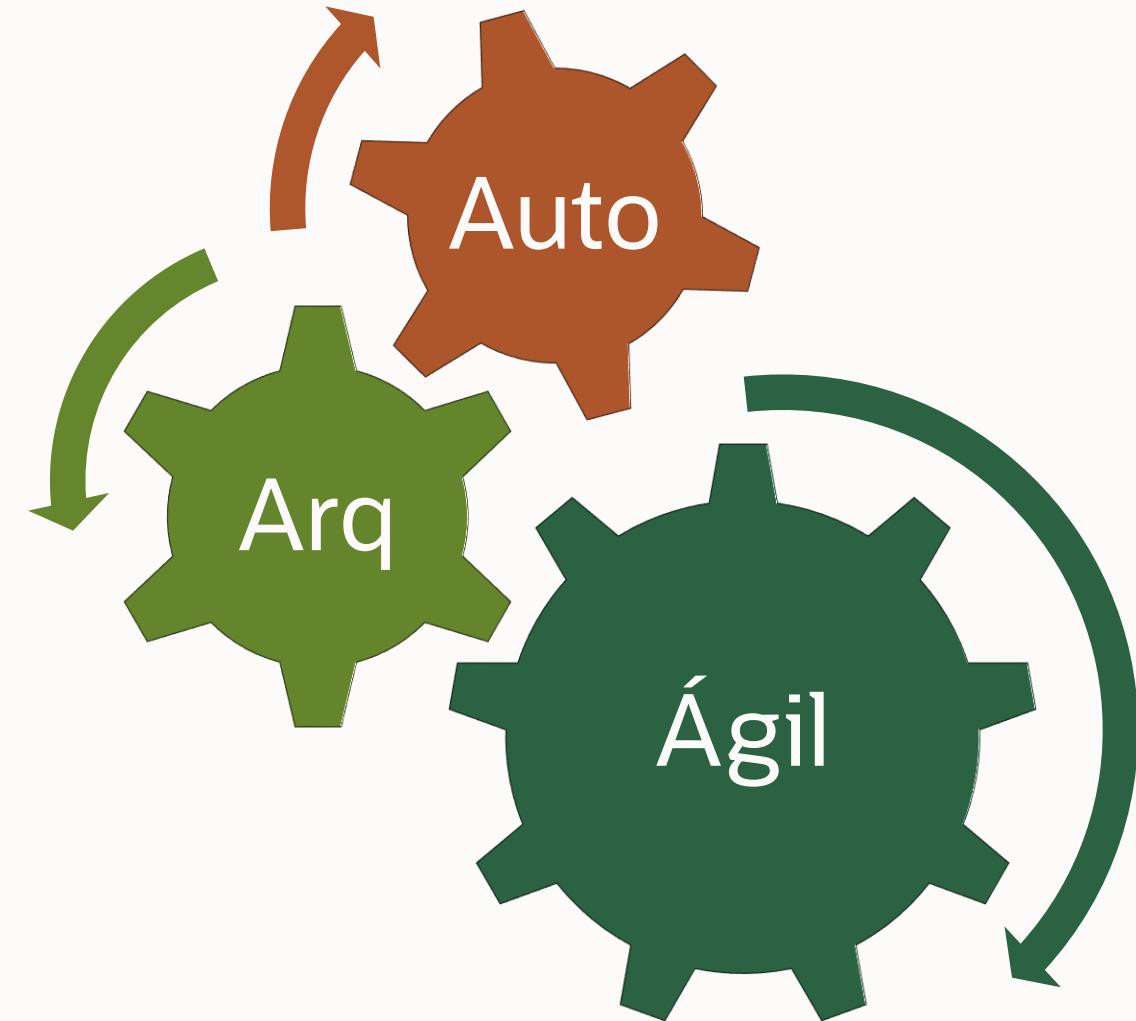
ORACLE

# Arquitecturas de Aplicaciones Modernas

## Beneficios, Características y Patrones

Francisco Moreno

# Pilares fundacionales para desarrollo Cloud Native



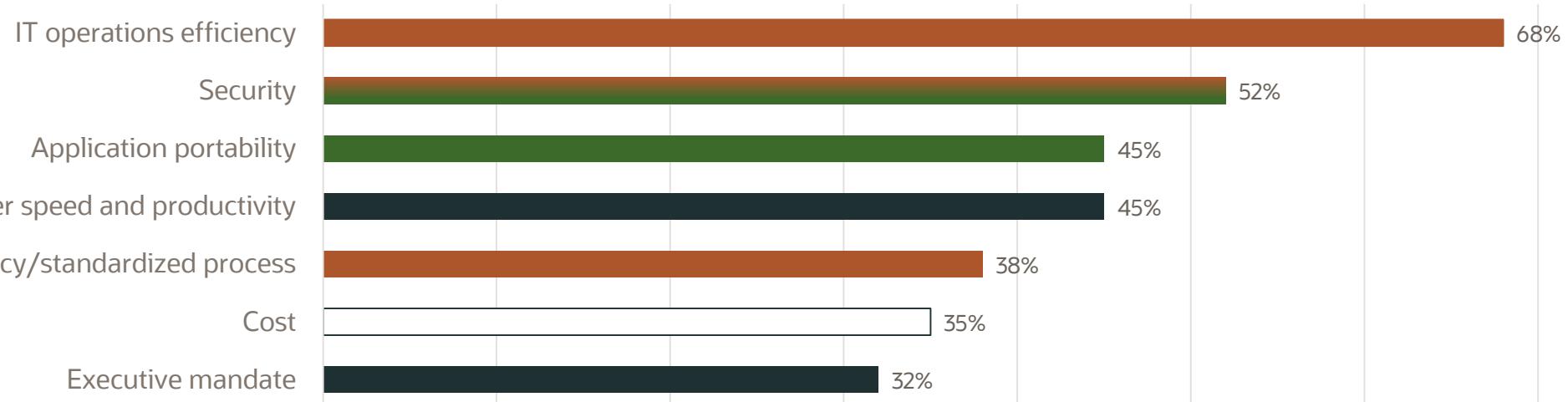
- |             |  |
|-------------|--|
| <b>Auto</b> | <b>Herramientas y Servicios: IaC, Contenedores y Orquestadores</b>                           |
| <b>Arq</b>  | <b>Arquitectura y Patrones: Microservicios y Eventos &gt;&gt; Orquestación y Coreografía</b> |
| <b>Agil</b> | <b>Metodologías: Scrum, SAFe</b>   |

# Porqué desarrollar aplicación Cloud-Native

451  
Research

S&P Global  
Market Intelligence

Top benefits of containers, serverless, and other cloud-native technologies



Tomado de: [451 Research's Voice of the Enterprise: DevOps, Workloads and Key Projects 2020](#)

# Que son las aplicaciones nativas?

---

Aplicaciones modernas están diseñadas:

- Basadas en **microservicios** con **baja cohesión** **Arq**
- Se comunica por **APIs** **Arq**
- Usualmente desplegadas **Contenedores** **Auto**
- Desarrolladas y desplegadas usando patrones y herramientas **DevOps Modernas** **Ágil**

Pero -- más



Arquitecturas con baja cohesión son el **predictor #1 del éxito de DevOps**

Cada servicio es desarrollados, desplegado y operado independientemente – Por eso puede moverse **rápidamente!**



# State of DevOps: Importance of Automation & Security to Business

**973x**

More deployments -  
increasing speed

**6570x**

Faster time to recover  
from incidents

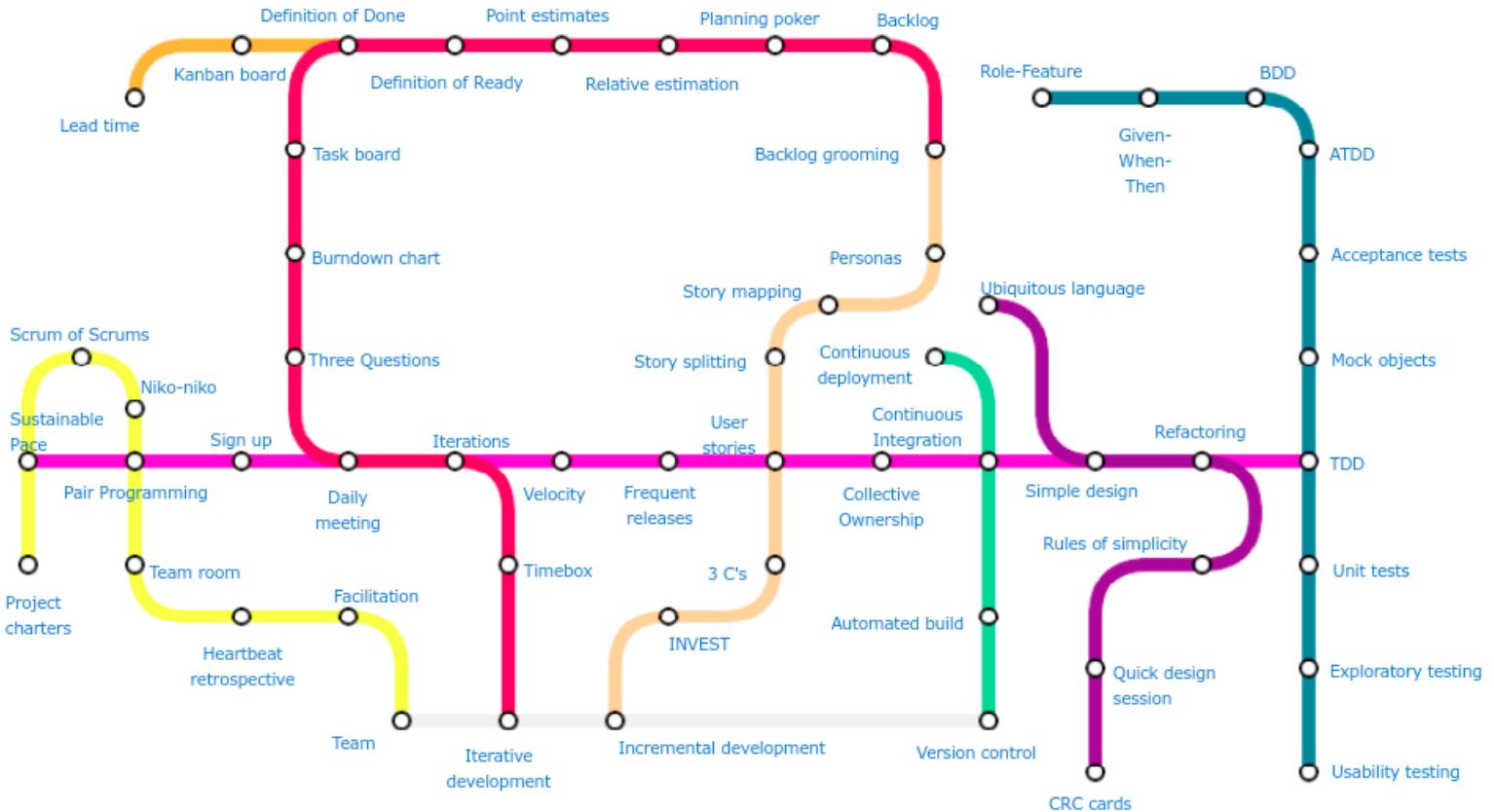
## Characteristics of Elite Organizations

Software delivery performance metric	Elite	High	Medium	Low
<b>Deployment frequency</b> For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per week and once per month	Between once per month and once every 6 months	Fewer than once per six months
<b>Lead time for changes</b> For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one hour	Between one day and one week	Between one month and six months	More than six months
<b>Time to restore service</b> For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day	Between one day and one week	More than six months
<b>Change failure rate</b> For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0%-15%	16%-30%	16%-30%	16%-30%

**Source:** DORA - State of DevOps 2021  
32,000 professionals worldwide surveyed over 7 years



# Mapa de Practicas Agiles

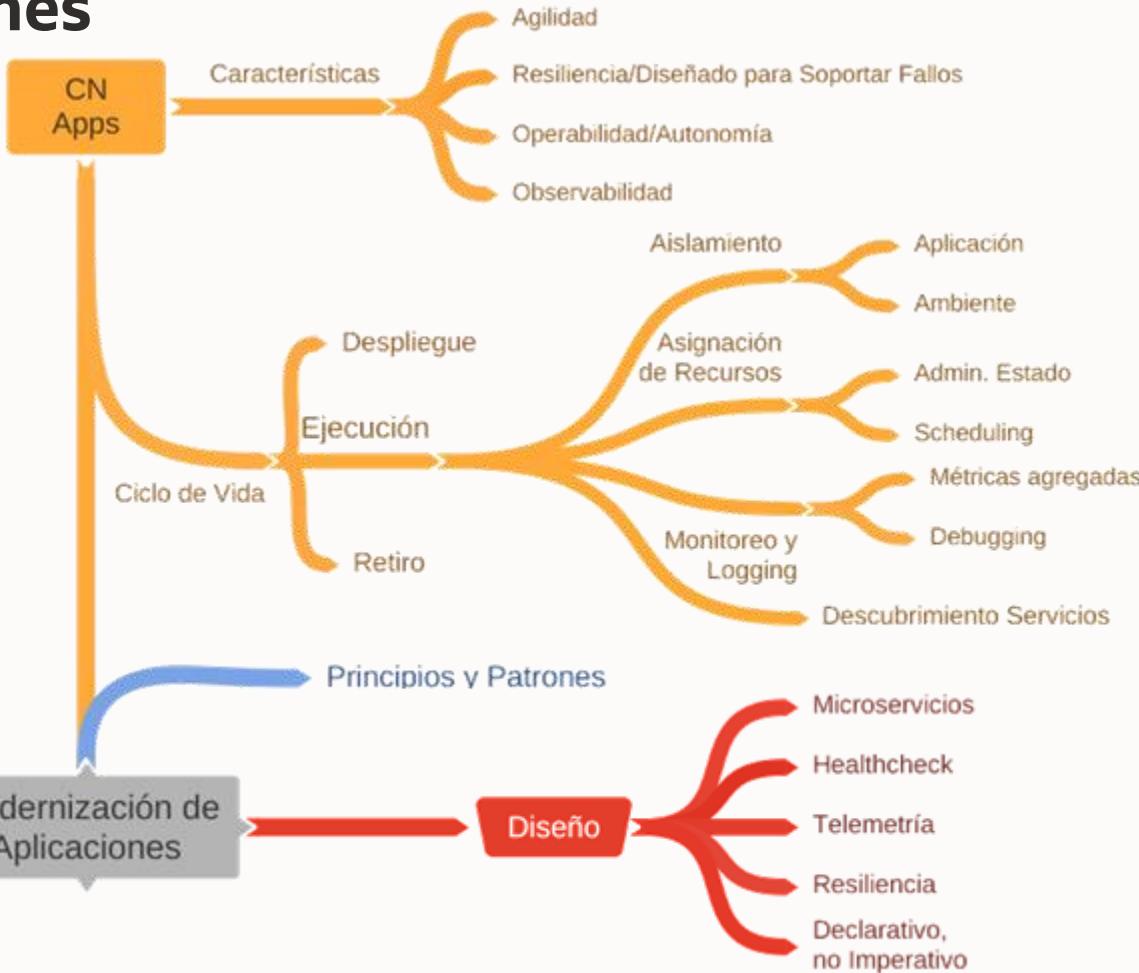
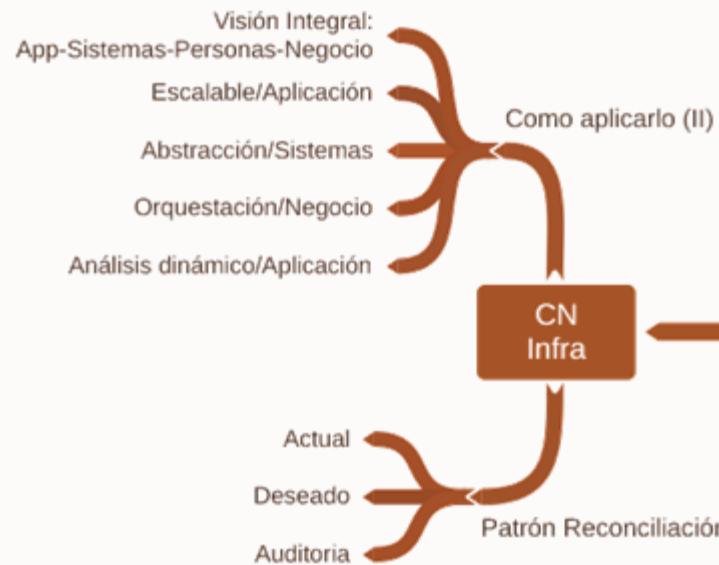


Lines represent practices from the various Agile "tribes" or areas of concern:

Extreme Programming	Scrum	Design
Teams	Product management	Testing
Lean	Devops	Fundamentals

Subway

# Modernizar Aplicaciones



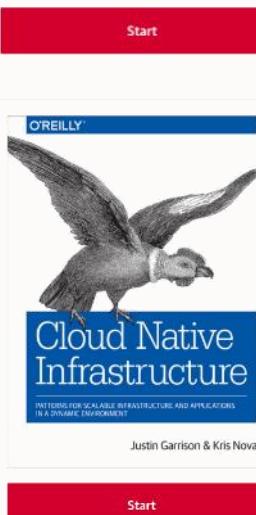
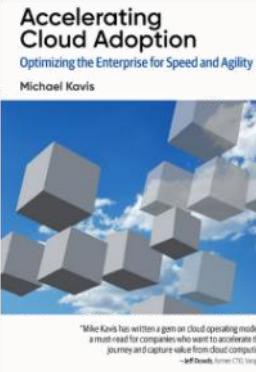
"(..) empower organizations to build and run **scalable** applications in(..) clouds.

Containers, service meshes, microservices, immutable infrastructure, and declarative APIs exemplify this approach.

(..) 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."

CNCF Definition



TOPICS:  
[Cloud Computing](#)

PUBLISHED BY:  
[O'Reilly Media, Inc.](#)

PUBLICATION DATE:  
November 2020

PRINT LENGTH:  
190 pages

TIME TO COMPLETE:  
4h 13m

TOPICS:  
[Cloud Computing](#)

PUBLISHED BY:  
[O'Reilly Media, Inc.](#)

PUBLICATION DATE:  
November 2017

PRINT LENGTH:  
160 pages

TIME TO COMPLETE:  
6h 27m

TOPICS:  
[Hybrid Cloud](#)

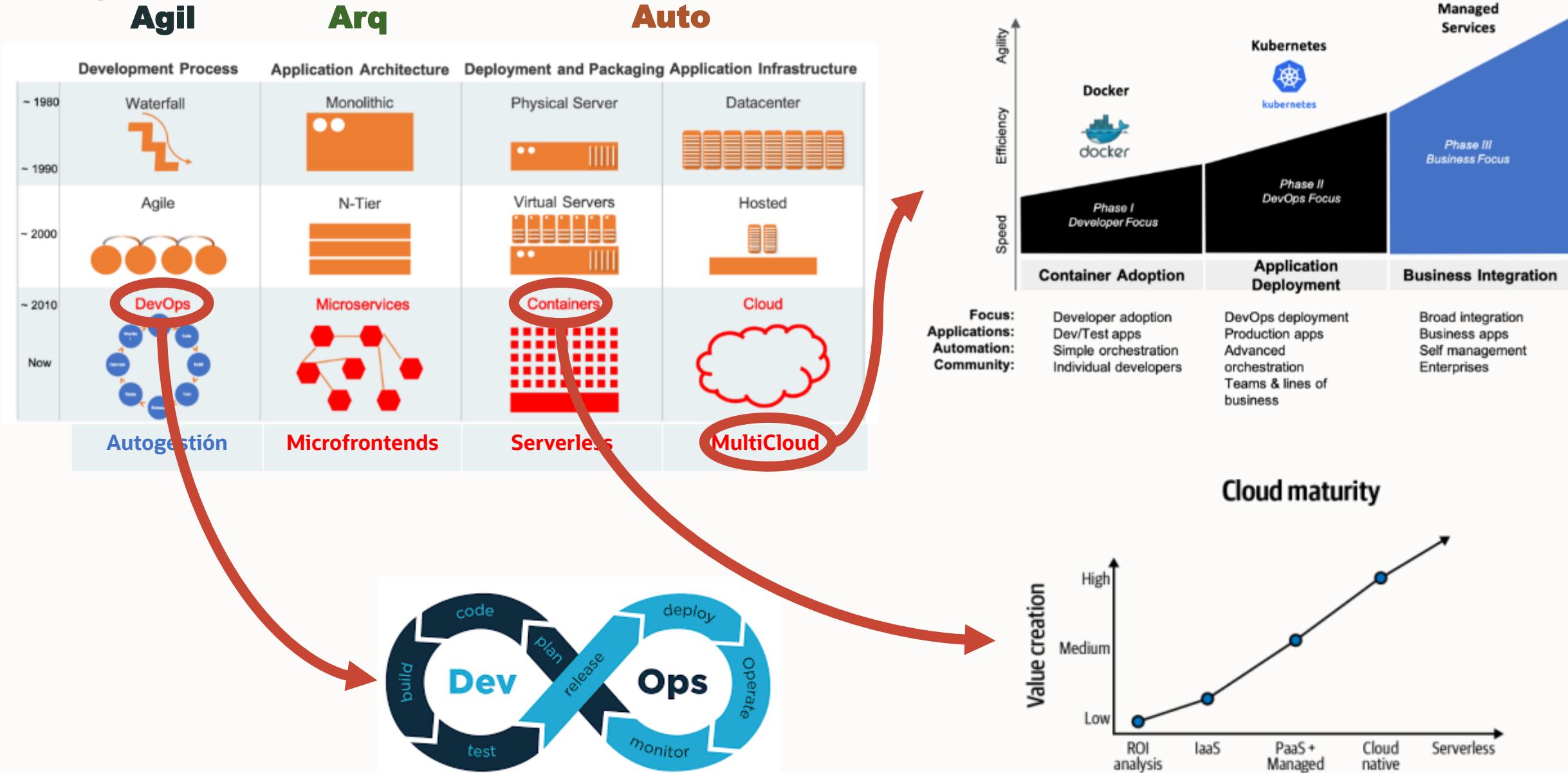
PUBLISHED BY:  
[Packt Publishing](#)

PUBLICATION DATE:  
October 2021

PRINT LENGTH:  
336 pages



# Del Agilismo a los DevOps



# Principios de Diseño de Aplicaciones Modernas

- Uso de frameworks ligeros y de código abierto así como lenguajes de programación maduros
- Construcción de aplicaciones como servicios que se comunican a través de APIs
- Empaquetado y despliegue de apps como contenedores
- Construcción, prueba y despliegue automático
- Servicios completamente administrados para eliminar complejidad a través de desarrollo de aplicaciones, runtimes y datos
- Mantener capa de aplicación sin estado
- Uso de bases de datos convergente con soporte completo a través de cualquier tipo de dato
- Recolectar métricas end-to-end (E2E) y monitoreo
- Eliminar puntos de falla (SPOF) a través de datos replicados y recuperación de fallas
- Implementar una aproximación de defensa en profundidad para asegurar el ciclo de vida de las aplicaciones

Principios y Patrones

Modernización de  
Aplicaciones



# Las Cloud Native Apps son maravillosas



## Características

**Agilidad**

**Time-to-market reducido**

**Escalable**

**Agnóstico a  
plataforma y lenguaje**

**Seguridad**

## Beneficio

Acelerara la entrega de servicios de alta calidad usando CI/CD y automatización

Construcción, iteración y despliegue de aplicaciones mas rápido

Construcción de aplicaciones low-code

Escalar automáticamente

Adopción de Estándares de código abiertos y portables para evitar el *vendor lock-in*

Selección correcta de lenguajes de programación y frameworks basado en la necesidad de la aplicación

Mejores practicas por defecto

Facilidad para ejecutar apps en una gestión segura (CSP)

# Pero las nuevas aplicaciones son complejas, un aliado simplifica el camino



## Características

**Agilidad**

**Time-to-market reducido**

**Escalable**

**Agnóstico a plataforma y lenguaje**

**Seguridad**

## Beneficio

Acelerara la entrega de servicios de alta calidad usando CI/CD y automatización

Construcción, iteración y despliegue de aplicaciones mas rápido

Construcción de aplicaciones low-code

Escalar automáticamente

Adopción de Estándares de código abiertos y portables para evitar el *vendor lock-in*

Selección correcta de lenguajes de programación y frameworks basado en la necesidad de la aplicación

Mejores practicas por defecto

Facilidad para ejecutar apps en una gestión segura (CSP)

## Diferenciador OCI

Blueprint para el Desarrollo de Aplicaciones

Plataforma completa para migrar, optimizar y modernizar aplicaciones

Infraestructura flexible con auto escalamiento (y redimensionamiento): Horizontal y Vertical

Soporte los estándares de código abierto, servicios y frameworks así como su extensión  
DevOps Toolchain para automatización

Seguridad Sencilla, Prescriptible e Integrada a través de IaaS, PaaS y SaaS



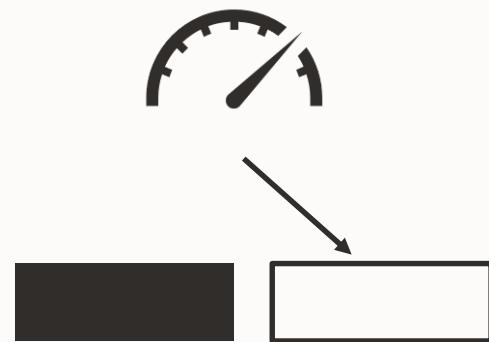
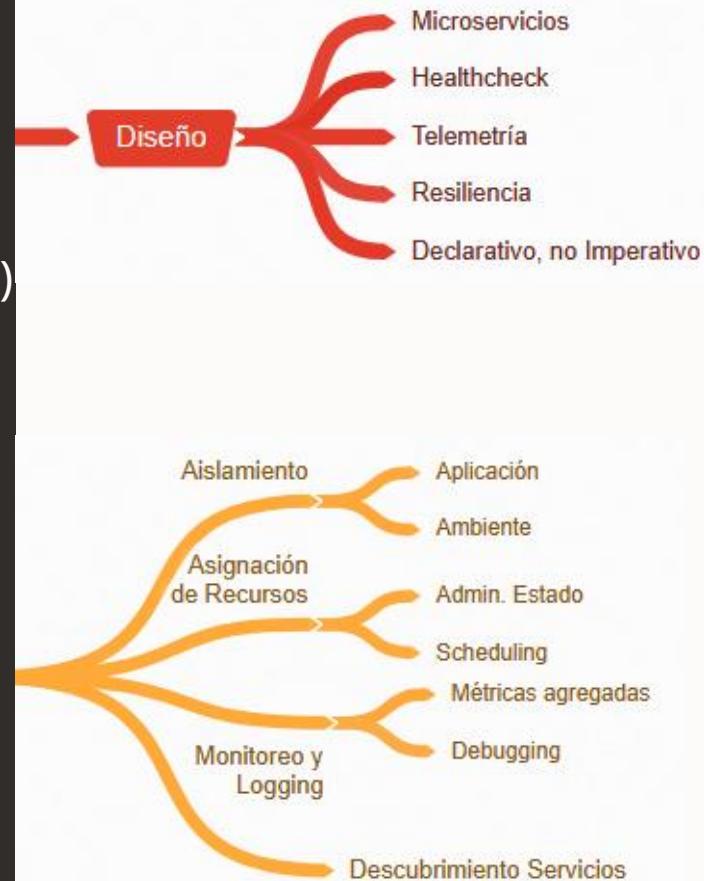
## Como lo hace y como se ejecuta?

### • Diseño:

- Microservicios: Contenedores y Funciones (Serverless)
- Healthcheck: Sondas y Agentes
- Telemetría: Métricas sobre funcionamiento
- Declarativo: Características de la aplicación

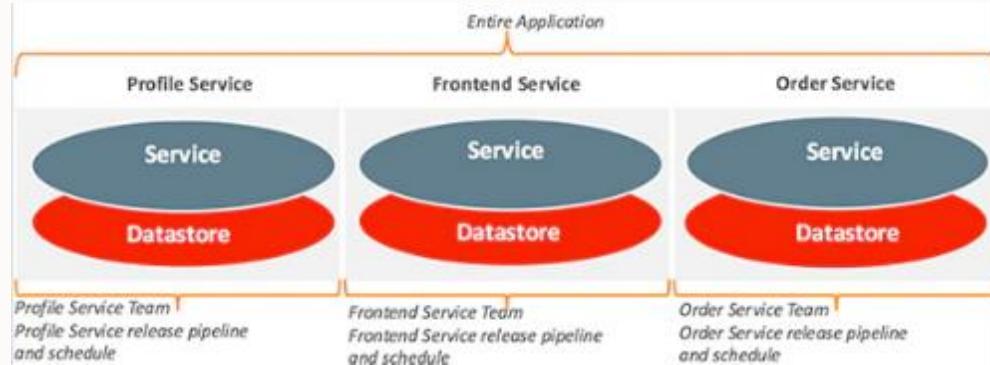
### • Ejecución:

- Aislamiento: Dividir funcionalmente.
- Asignación de Recursos: Disponible y Adecuado.
- Monitoreo y Logging: Comportamiento y Mejora.
- Asignación de Recursos/Monitoreo y Logging: Escalamiento.

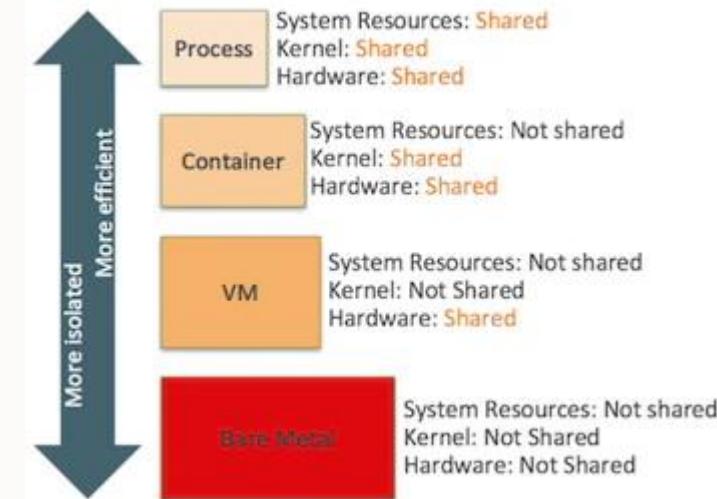


# Retos Diseño I

## Arq

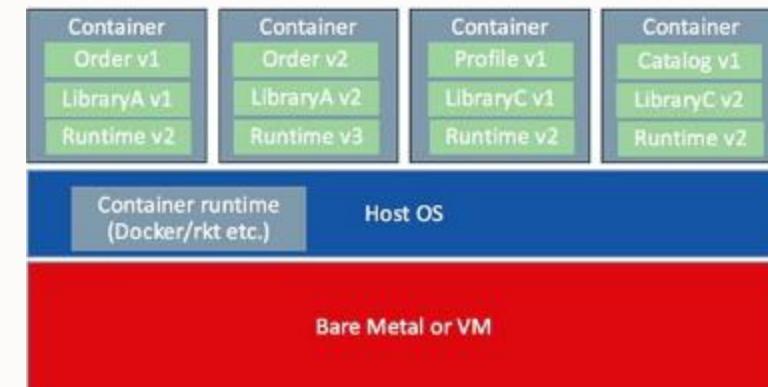


## Density & Isolation levels



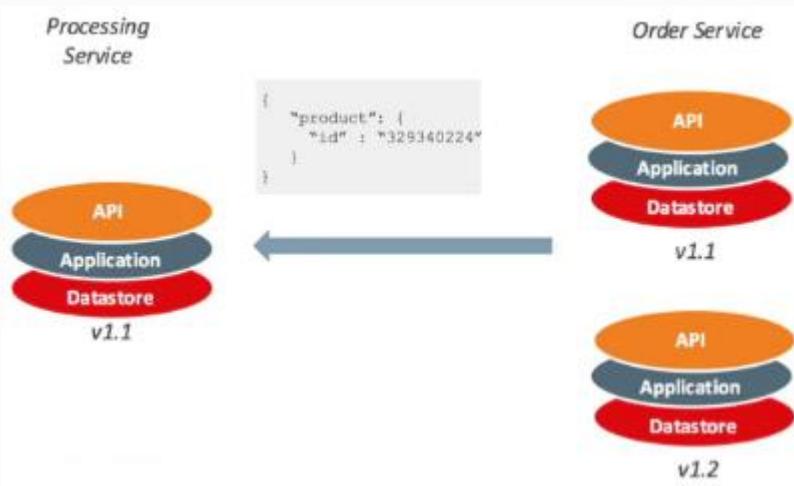
Aislamiento: Funcional (Microservicios), Aplicación (Contenedores), Ambiente (IaC).

Optimización: Uso Hardware (Contenedores), Costo (Serverless)



# Retos Diseño II

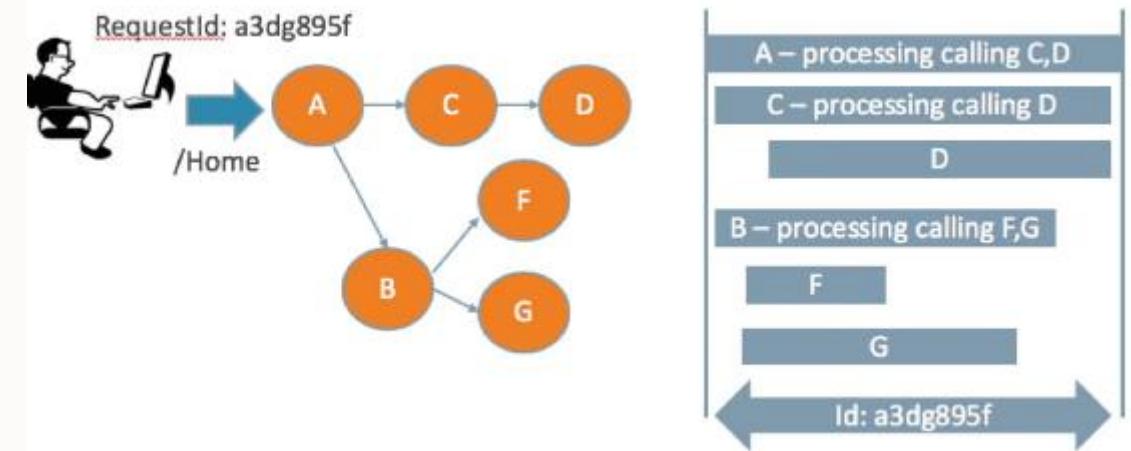
Arq



Necesidad Versiones y Compatibilidad hacia atrás

```
{  
  "credit" : {  
    "forAccount" : "12345"  
  },  
  "amount" : "100",  
}  
}  
{  
  "credit" : {  
    "creditID" : "124e456-e89b-12d3-a456-426655",  
    "forAccount" : "12345"  
  },  
  "amount" : "100",  
}  
}
```

Necesidad Idempotencia en Trx.



Trazabilidad Distribuida

## Patrones de Diseño:

Queries: Request/Response, Materialized View, Request/Reply (Queue Directional/2).

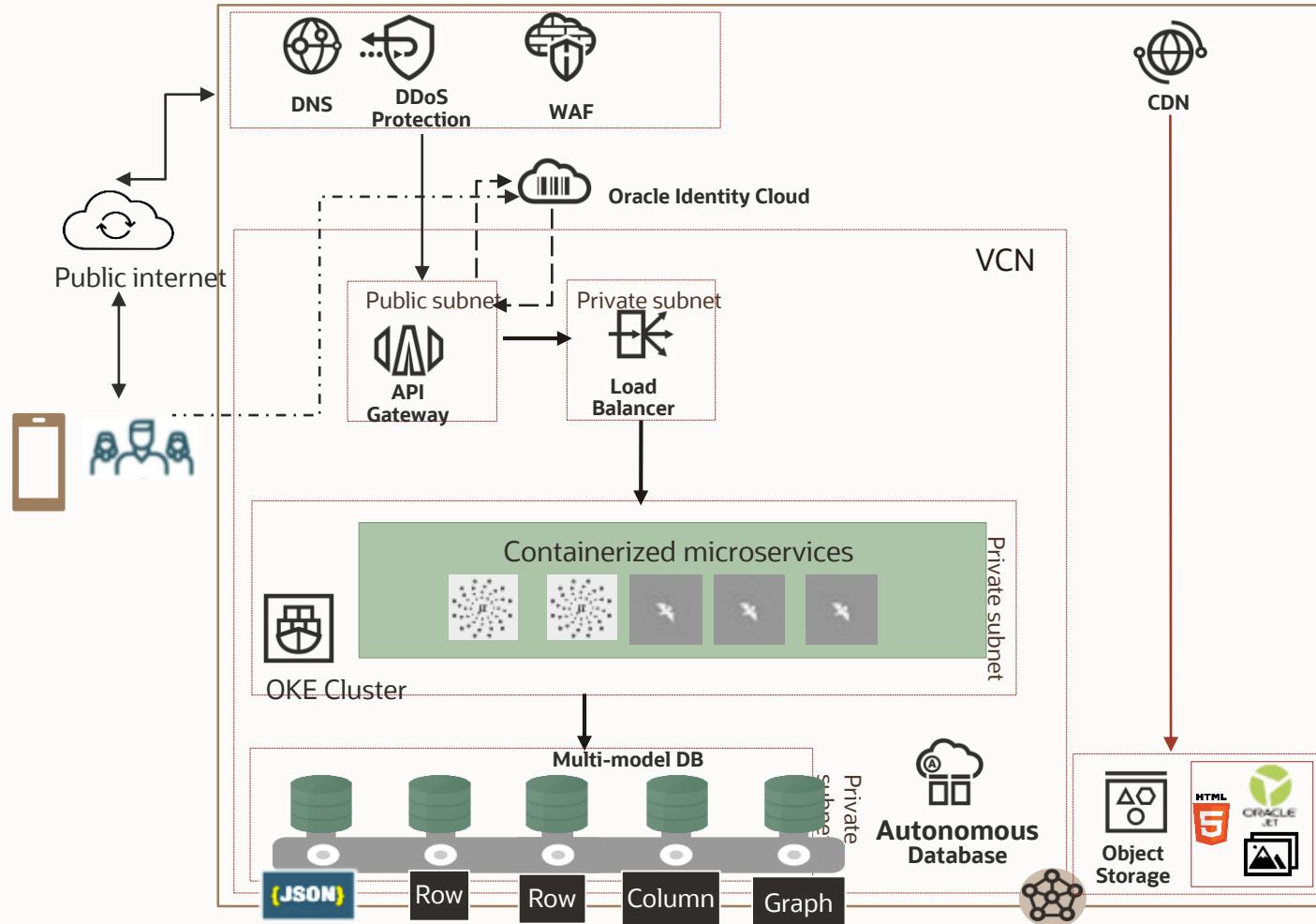
Command: Una Cola. Otro canal para replicar (i.e. PSE)

Events: Choreography

Políticas: Reintento, Circuit Breaker, Bulkhead (Limitar alcance servicios), Cache, Fallback.

# Web/ Mobile Microservices

Patrón de arquitectura



## Visión Oracle

- Framework Ligeros y abiertos
- DB convergentes y multimodelo
- Poliglota es adecuado, use según necesidad
- Seguridad en cada capa
- Automatización DevOps

## Beneficios

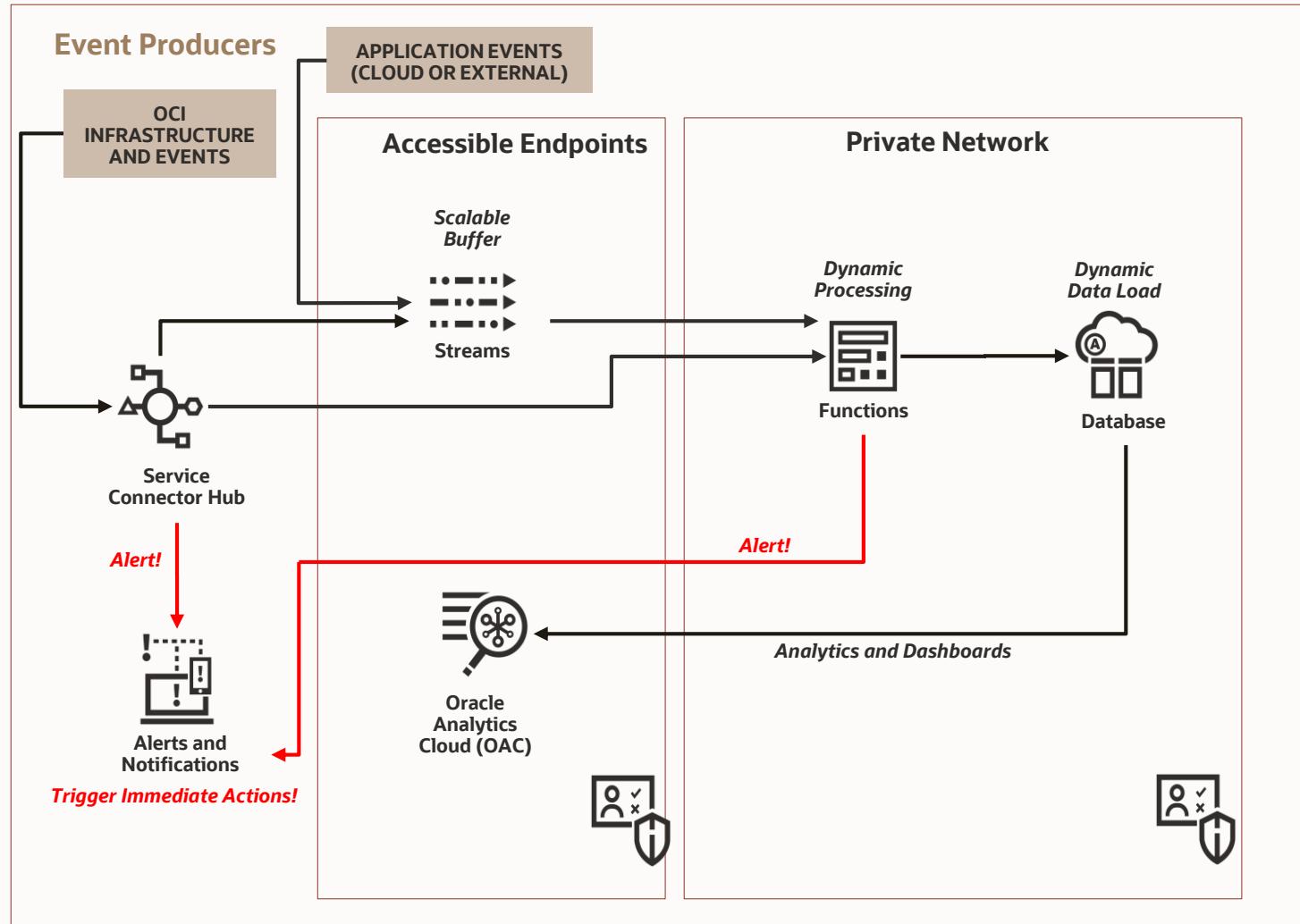
- Use su lenguaje y framework favorito
- Contenedores manejados reducen tiempo de despliegue
- DB convergente reduce administración, independencia de ciclos de desarrollo, etc.

## Casos de Uso

- e-commerce
- Online banking
- Apps de Entregas
- Sistemas de Recomendación

# Event-driven Architecture

## Patrón de Arquitectura



### Visión Oracle

- Permite arquitectura serverless
- Componentes bajo acoplamiento
- Entrega de mensajes idempotentes
- Automatización DevOps

### Casos de Uso

- IoT streaming
- Manejo de alertas
- Aplicaciones de Entrega
- Comunicación entre aplicaciones distribuidas

### Recomendaciones

- OCI Streaming (Kafka)
- OCI Functions
- OCI Notifications
- Service Connector Hub
- Oracle Autonomous Database
- Oracle Analytics Cloud



# Visión Oracle para DevOps y Cloud Native

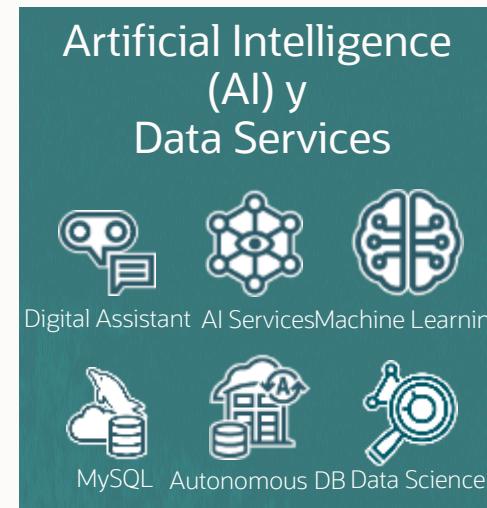
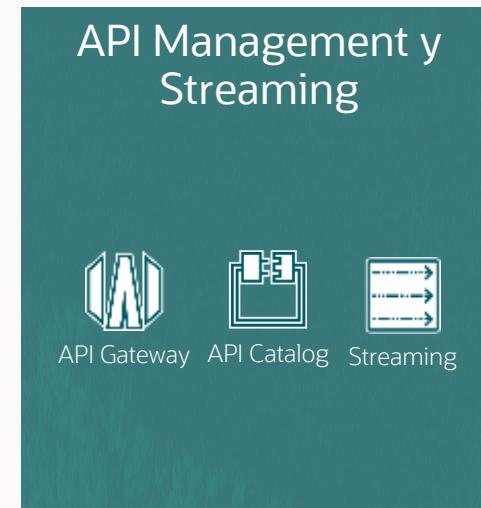
DevOps Services, Resource Manager, Oracle Kubernetes Engine,  
Functions, Monitoreo y Observabilidad, y Multicloud

**Francisco Moreno**

# Desarrollo Cloud Native en OCI



Productividad  
en Desarrollo



OCI DevOps + Libertad de escoger un Ecosistema Integrado



Confiabilidad  
en  
Operaciones

Operación Autónoma

Servicios Datos Compartidos

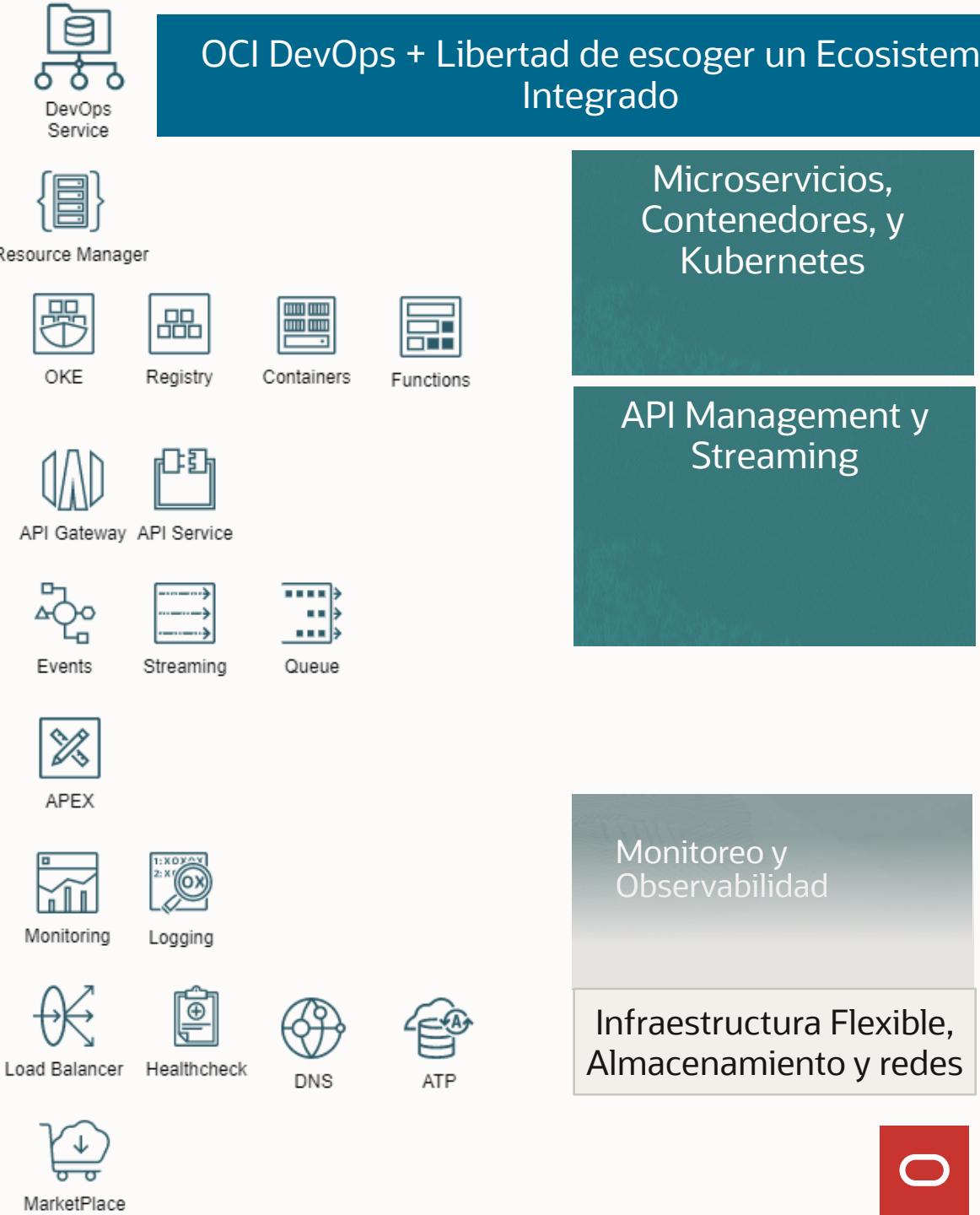
Infraestructura Flexible, Almacenamiento y redes: cualquier tamaño, carga en cualquier lugar

IAM, Seguridad y gobierno a través de un ciclo de desarrollo

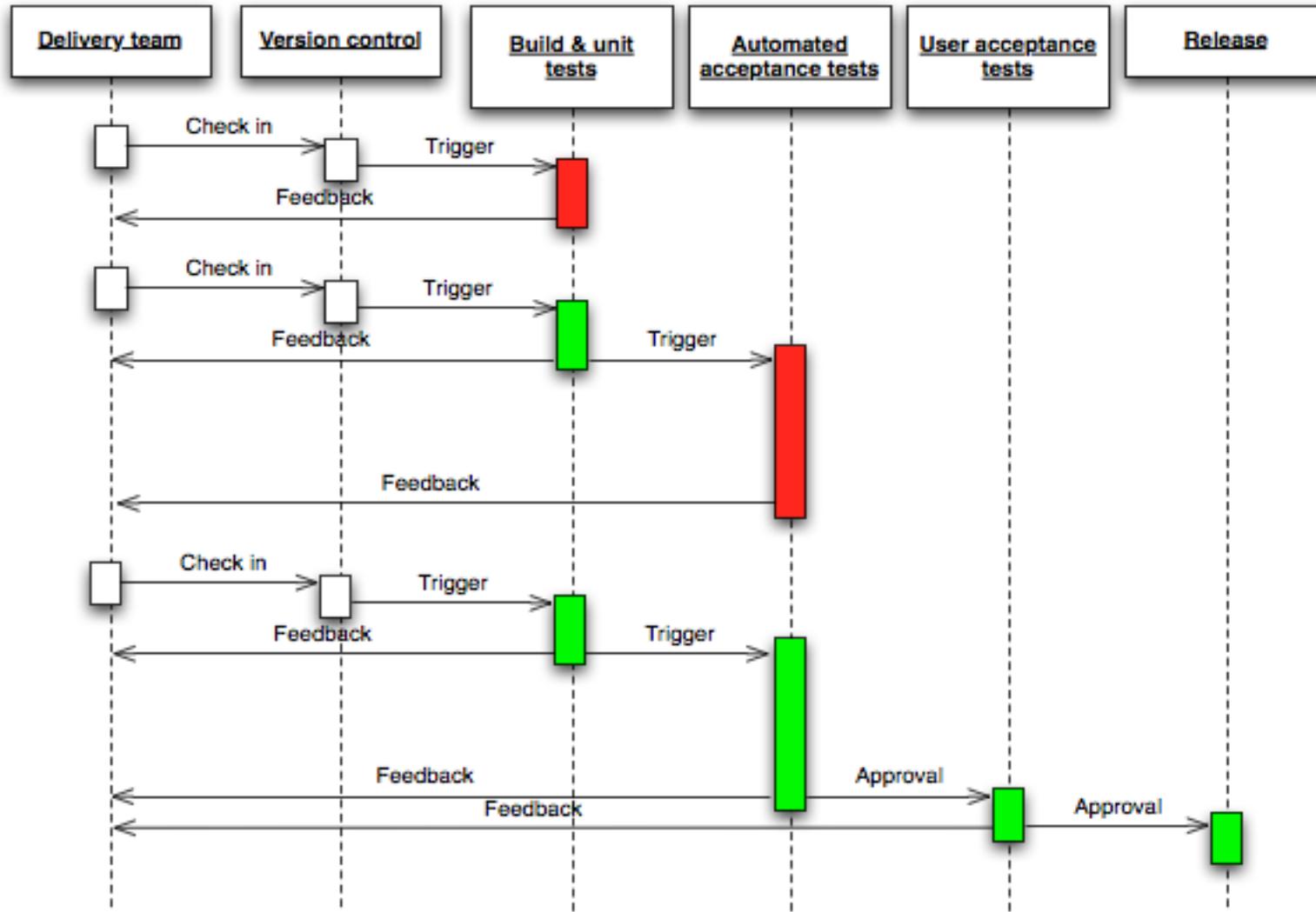
Desde capa gratuita hasta hiper-escala – en una misma plataforma

# Administración e Infraestructura Cloud Native

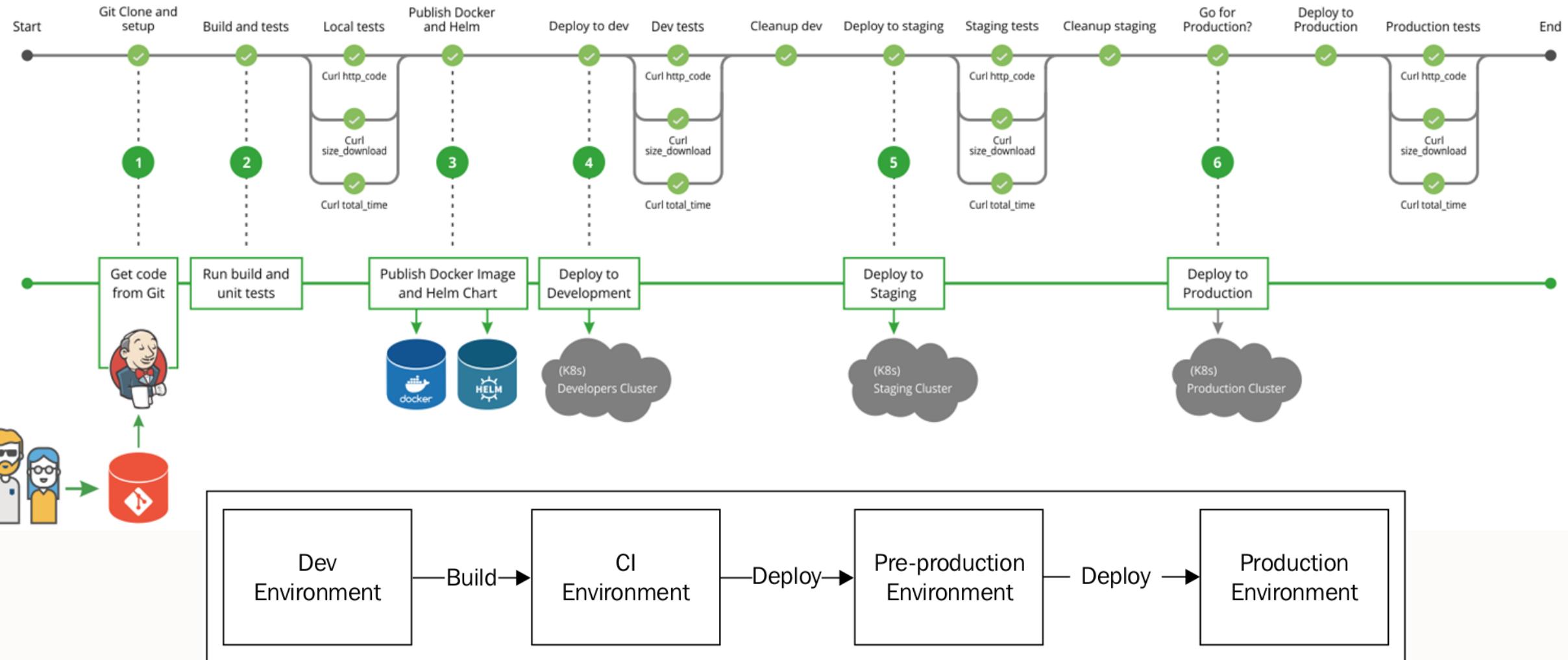
---

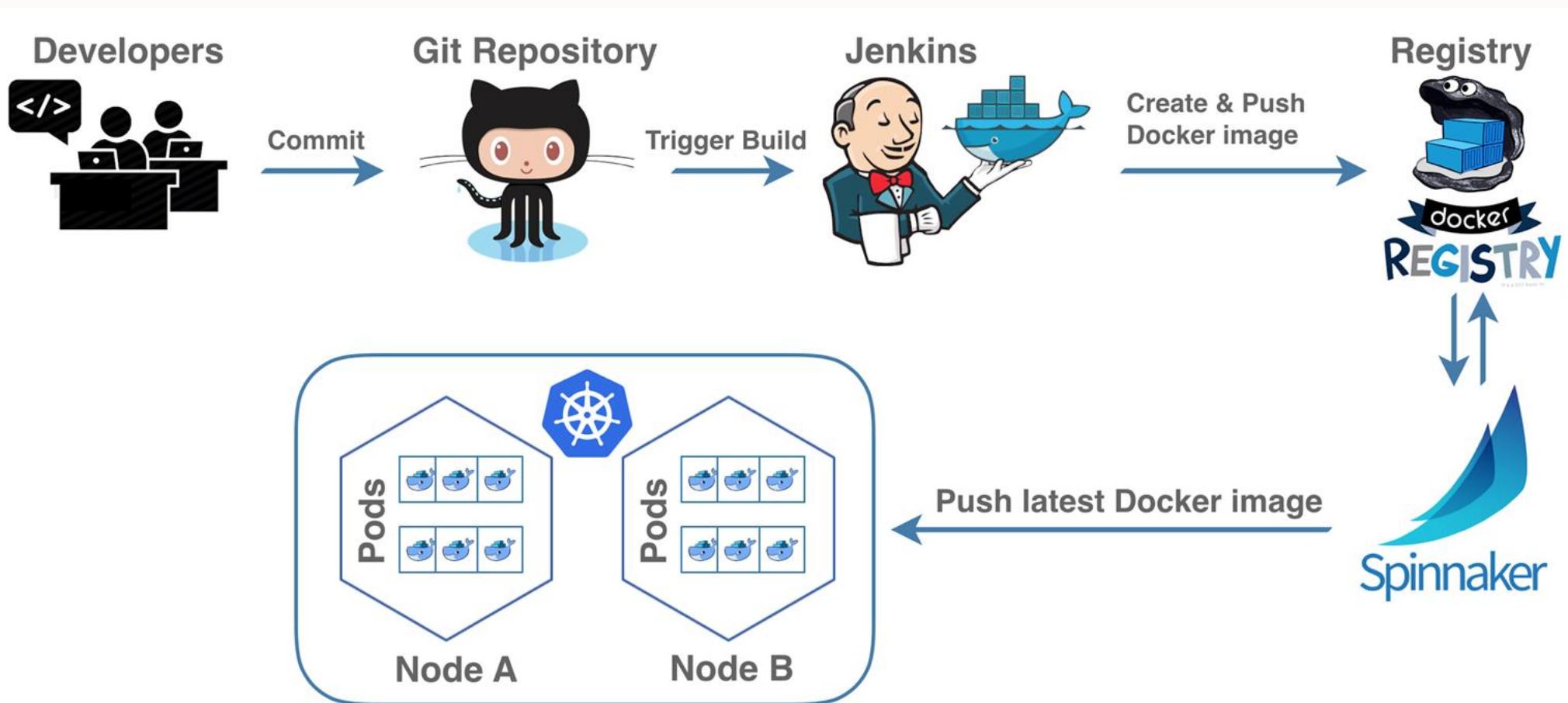


# Continuous Integration/Continuous Delivery (CI/CD)



# Continuous Integration/Continuous Delivery (CI/CD)

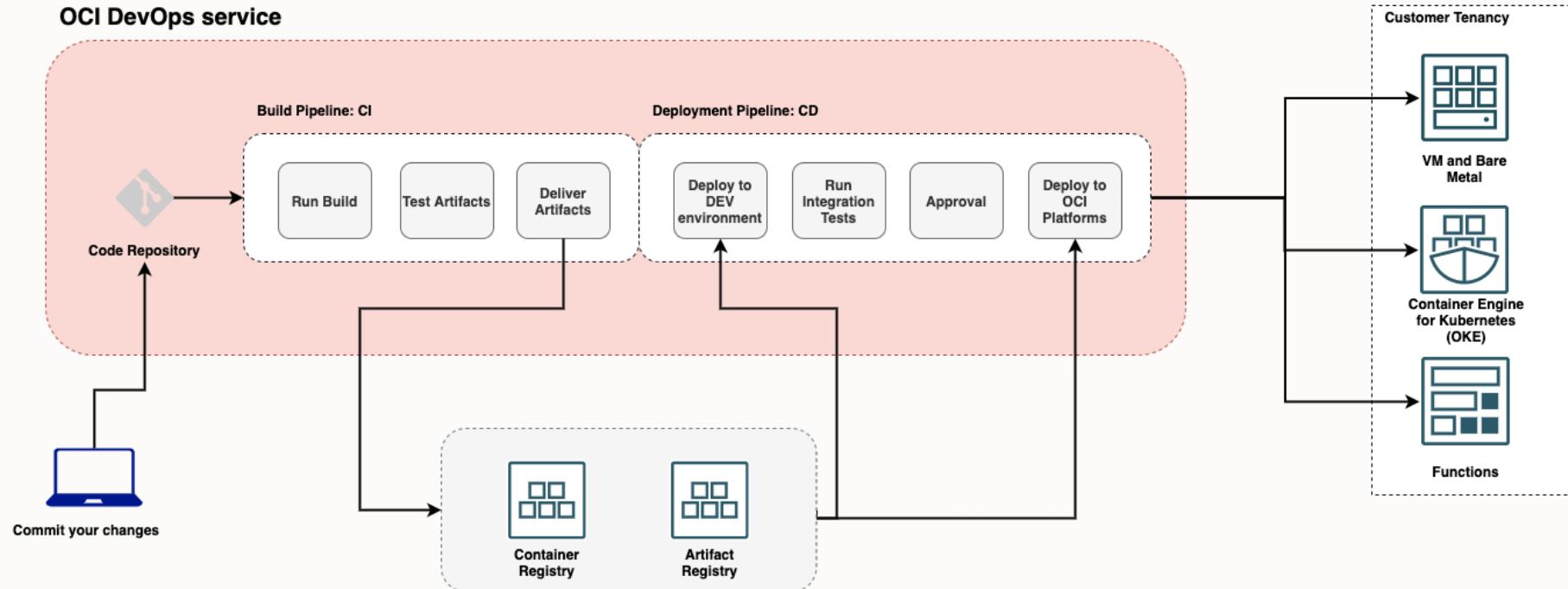




## CI/CD for Kubernetes Cluster with Jenkins and Spinnaker

# Construcción: DevOps Service

## OCI DevOps service



Automatización:  
Velocidad/Confiabilidad



Seguridad: Usuarios IAM



Gobierno: Observabilidad,  
Monitoreo y Gobierno



Serverless: PAY2G, Acciones Concurrentes.



Estrategias: Globales y otros.

**CI**



Integración: GitHub, GitLab, OCI.



Plataforma completa:  
Oci Svcs: Artifact, Vault, etc.

**CD**



Integración: Plugin Jenkins, Multicloud.

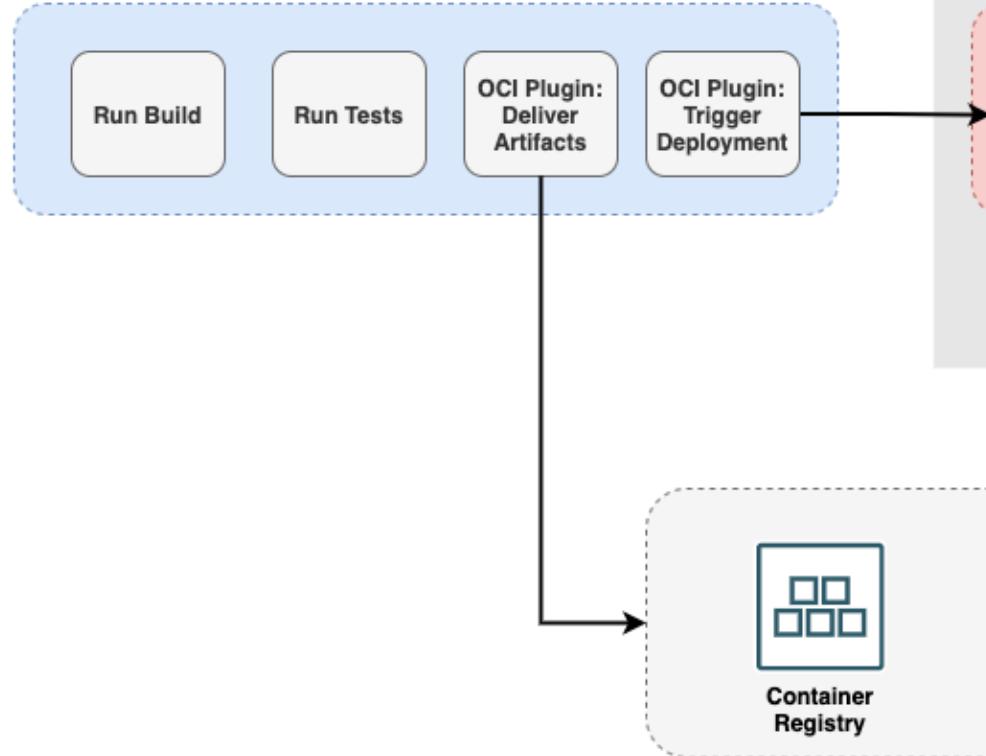


Rollback

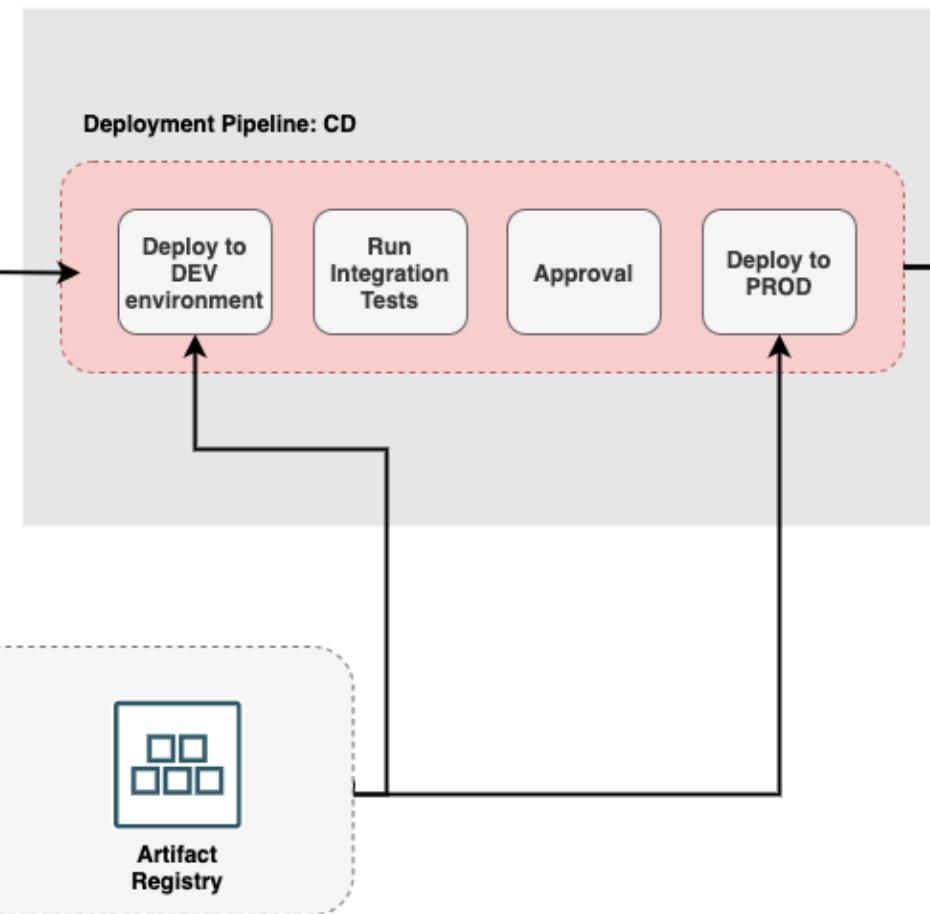
# DevOps Service: Pantallazos



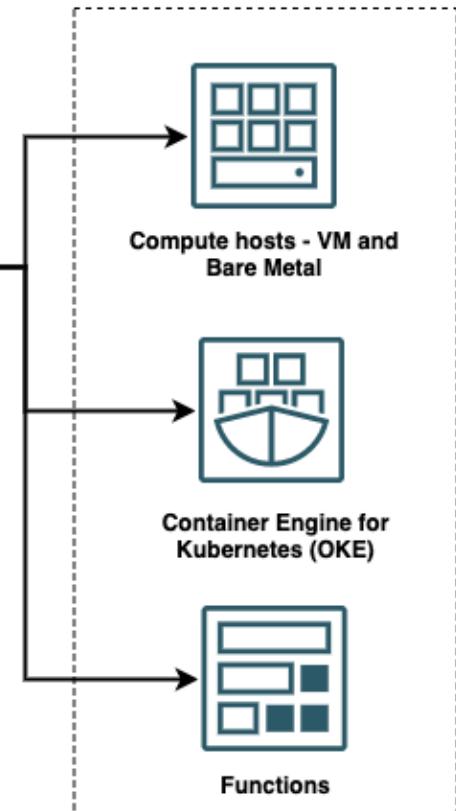
## Jenkins Pipeline



## OCI DevOps service



## Customer Tenancy



## OCI Artifact Repositories

Step	Status	Duration
NPM install	Completed	0min 29s
Run Tests	Completed	0min 2s
Build container image	Completed	0min 16s
SAVE_OUTPUT_ARTIFACTS	Completed	0min 5s
Run Deployment to OKE	Pending	-
Trigger Deployment	Pending	-

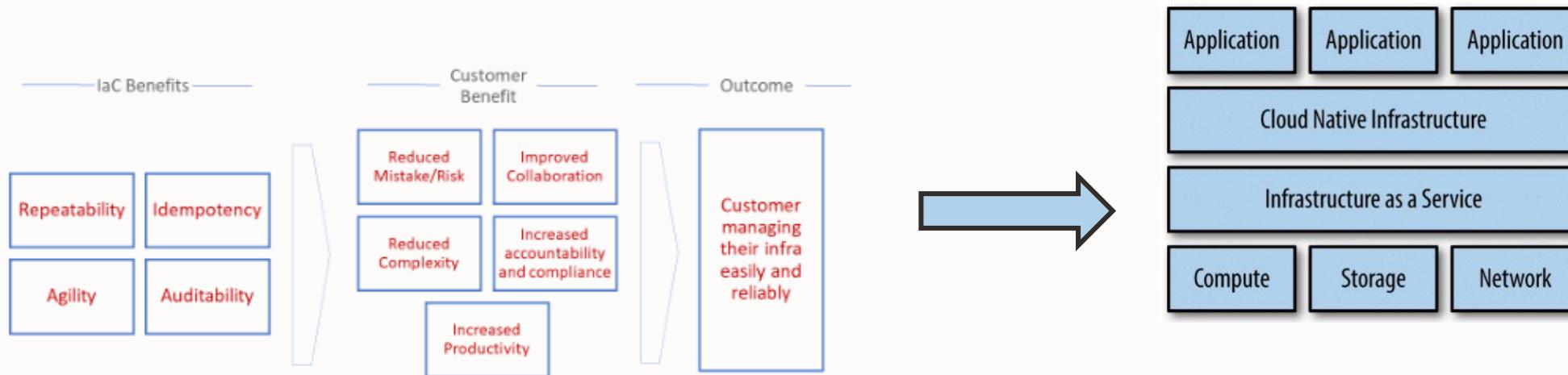
Logs:

```

2021-09-24T16:51:23.839Z Completed PARSE_BUILD_SPEC
2021-09-24T16:51:23.848Z Starting BUILD_SPEC_EXECUTION
2021-09-24T16:51:23.852Z Starting DOWNLOAD_INPUT_ARTIFACTS
2021-09-24T16:51:23.853Z Completed DOWNLOAD_INPUT_ARTIFACTS successful
2021-09-24T16:51:23.867Z Starting EXECUTE_BUILD_SPEC_STEPS
2021-09-24T16:51:22.009Z Executing BUILD_SPEC_STEP : Export variables
2021-09-24T16:51:22.014Z Executing STEP : Export variables with shell
2021-09-24T16:51:22.093Z EXEC : BUILDRUN_ASASH: /sys/fs/cgroup

```

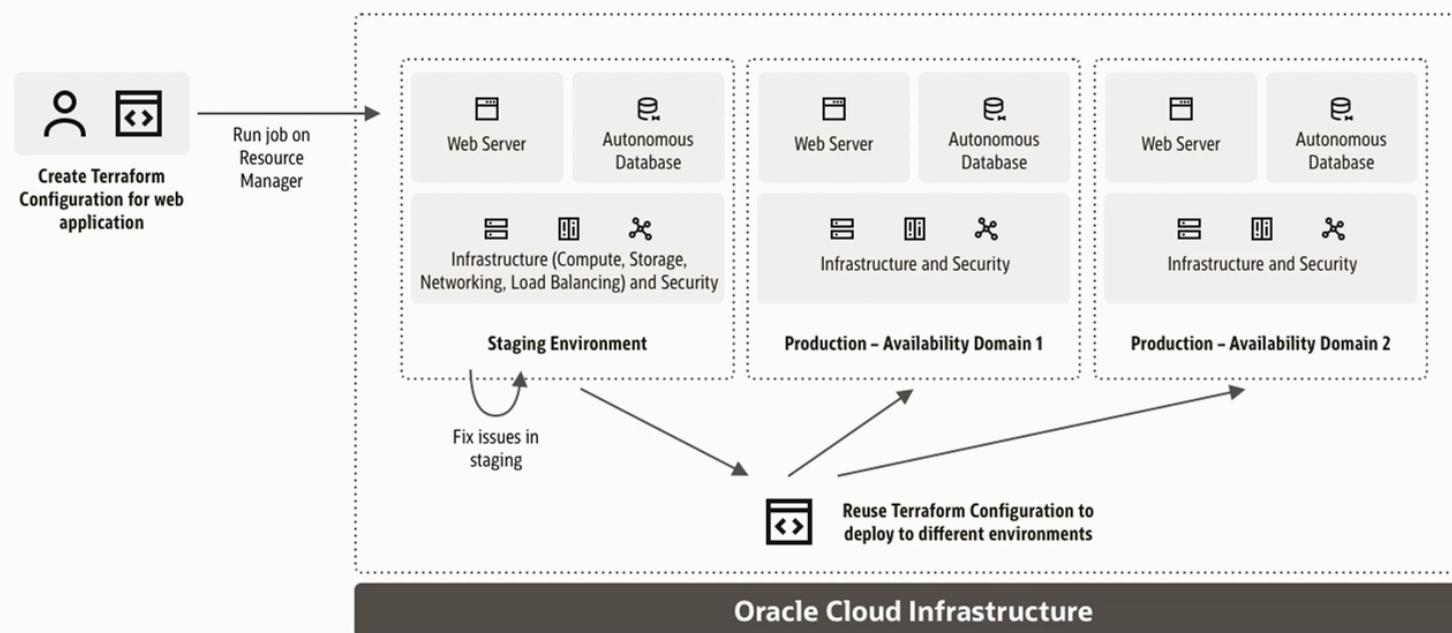
# Despliegue: Resource Manager



Terraform	
Visión Integral	Puede ser estricta
Escalable	Stack/Anidado
Abstracción	Compartiment (Solo OCI) o Modular (TF)
Orquestación	Grafos Internos.
Análisis Dinámico	N/A
Patrón Reconciliación	Completo.

# Resource Manager: Terraform

## Deploy apps to staging and production



A screenshot of the Oracle Cloud Resource Manager interface. The top navigation bar shows 'ORACLE Cloud' and 'Applications'. The main area is titled 'Resource Manager' and displays a list of 'Stacks'. A message indicates 'Sample solutions are now available for creating stacks. Use a sample resources from a provided Terraform configuration.' Below this, there are tabs for 'Stacks', 'Jobs', and 'Configuration Source Providers'. Under 'Stacks', there is a table with columns: Name, Description, State, and Created. The table lists several stacks:

Name	Description	State	Created
Subnet Stack		Active	Thu, Mar 12
Compute Instance		Active	Tue, Mar 10
Oracle Data Science		Active	Mon, Mar 11
Resource Discovery from compartment		Active	Wed, Mar 12
GitLab Stack-20200506174336		Active	Thu, Mar 12
Hyperion		Active	Tue, Mar 10

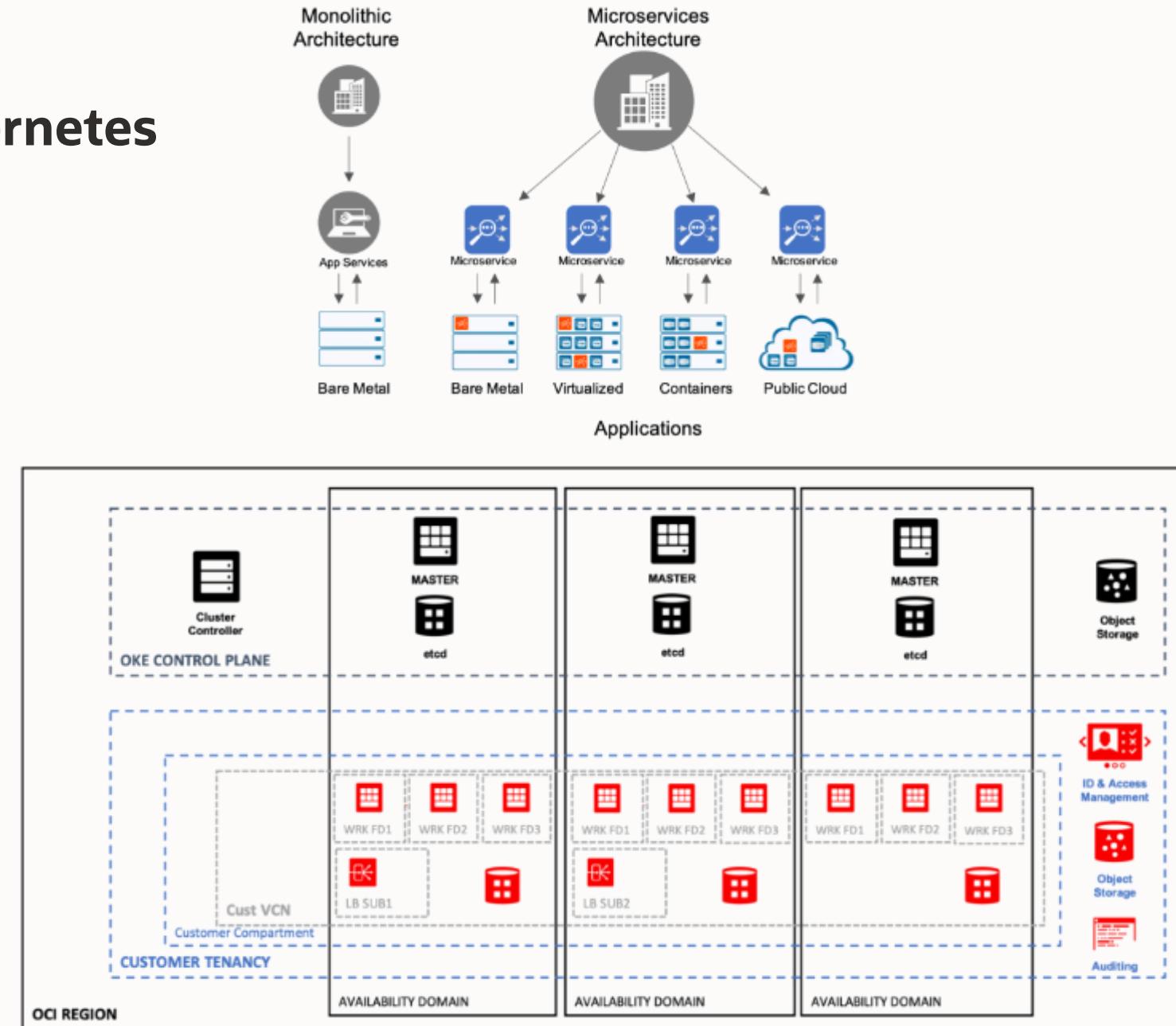
At the bottom of the interface, there are links for 'Terms of Use and Privacy' and 'Cookie Preferences', and a copyright notice 'Copyright © 2020'.

- Resource Manager: Alta Integración con Terraform y como fuente GitHub, GitLab.
- Promoción de entornos: Aislamiento

# Despliegue y Ejecución: Kubernetes

	<b>Oracle Kubernetes Engine: OKE</b>
Visión Integral	<b>Cluster/Clusters (Verazzano)</b>
Escalable	Replicaset (Adecuado) /DaemonSet (1 Agente)
Abstracción	Deployment
Orquestación	Por Diseño
Análisis Dinámico	Servicio
Patrón Reconciliación	Completo

Integración y Opciones usando Recursos OCI: Compute, Load Balancers, Storage.



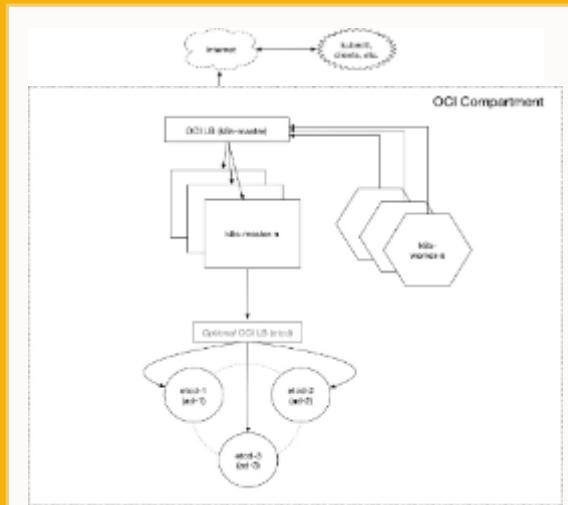
# Maneras de Ejecutar Kubernetes en OCI

Oracle Cloud Infrastructure



Roll-Your-Own  
Container Management

Quickstart Experience  
(OSS Terraform Installer on GitHub)



Pre-Built Kubernetes  
Installer

Container Engine for Kubernetes  
(OKE)



Enterprise Class Managed  
Kubernetes Service

# Container Engine for Kubernetes - OKE y Registry

## Cloud Native

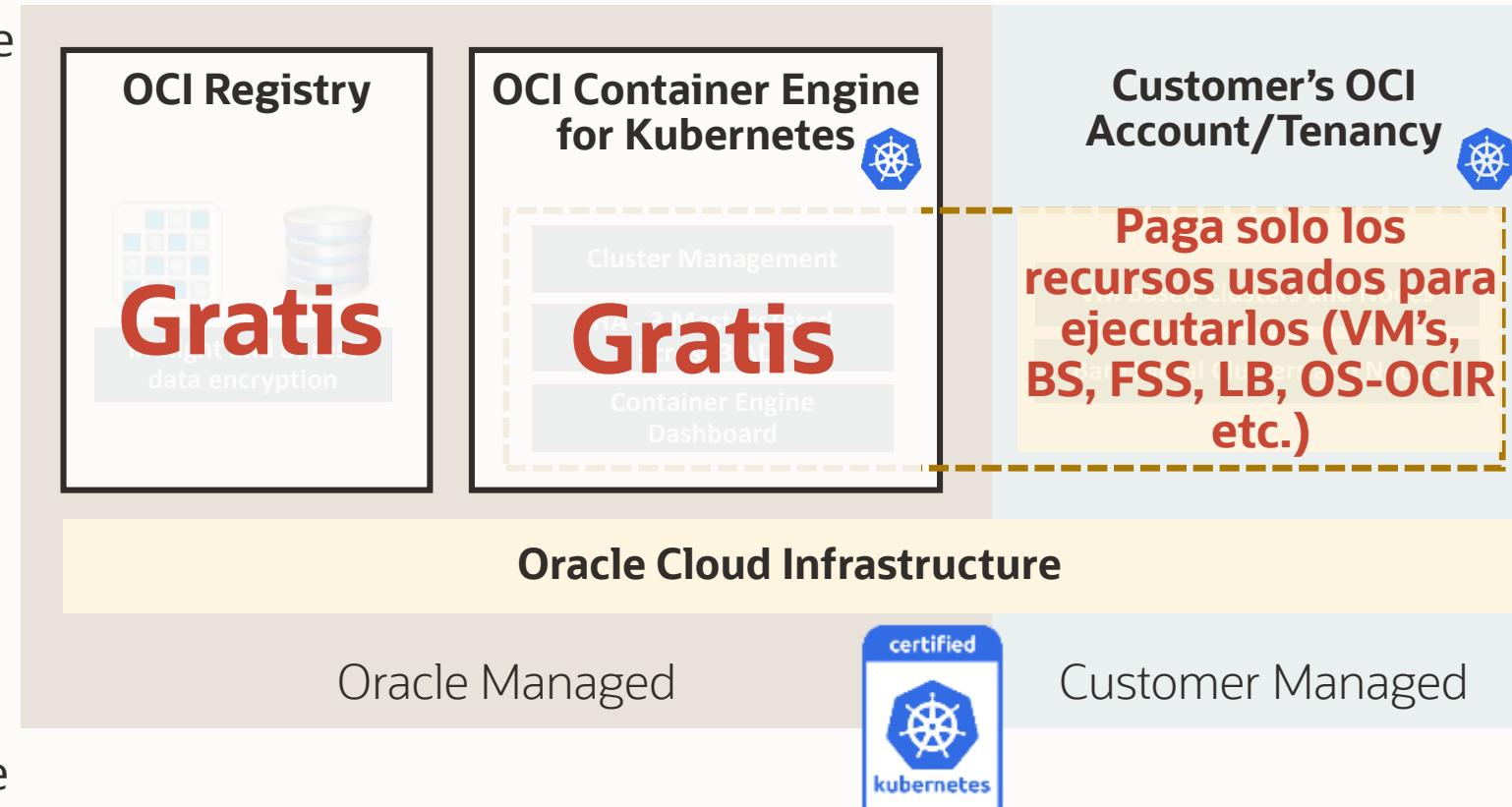
- Standard Docker y Kubernetes
- Integración con Registry, Red y Storage

## Integración con Desarrollo

- Streamlined workflow
- Full REST API
- Add-ons de Cluster preconstruidos
- Estándares Abiertos – Kubernetes sin modificar.

## Enfoque Empresarial

- Operadores Simples para Clúster
- Rendimiento para Full BM y HA IaaS (GPU, ARM)
- Clústeres Autónomos (Zero Downtime para Updates)



## Costos

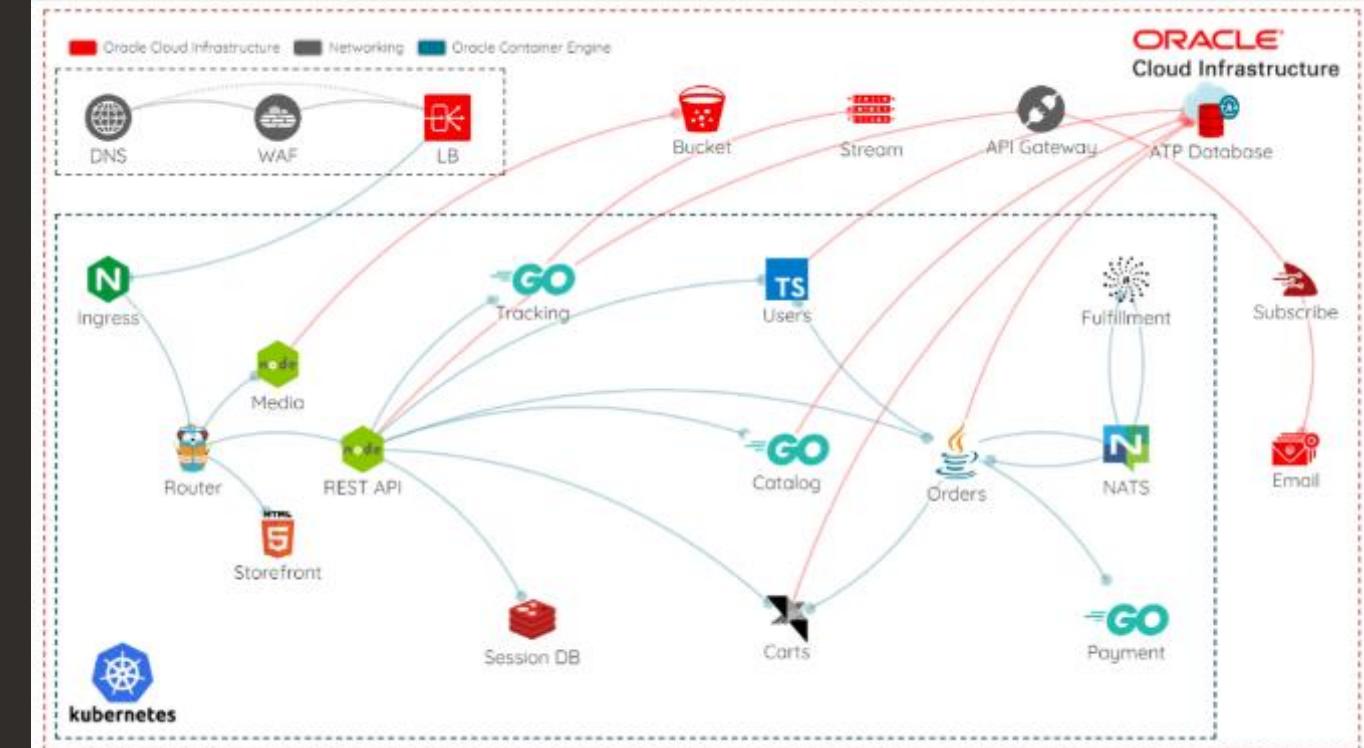
Sin costo para el Control Plane, Trafico entre AD, Zonas de Seguridad y Cloud Guard.

# Ejemplo del Cloud Native (OKE y Functions)

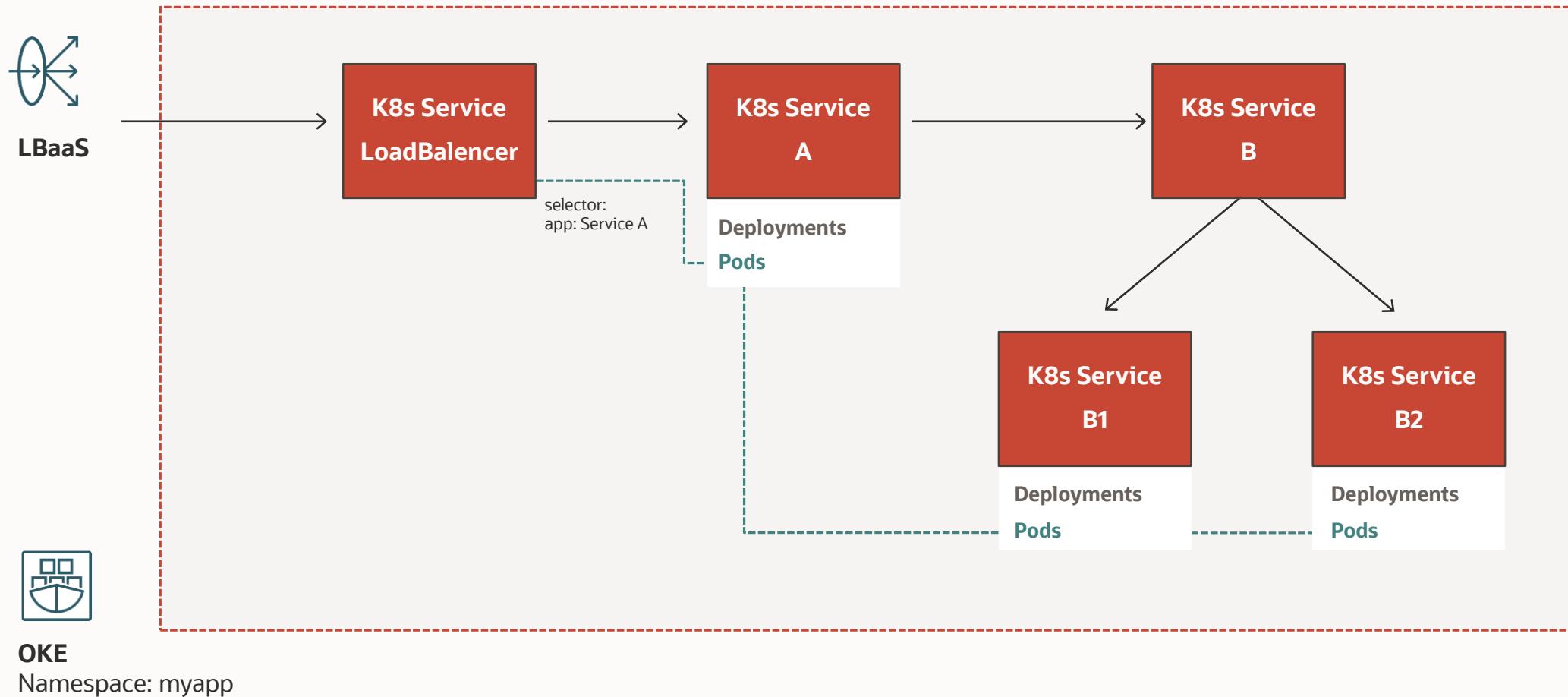
<https://github.com/oracle-quickstart/oci-cloudnative/>

Lanzamiento del Resource Manager  
Variables, Recursos, Actualizaciones, Salidas

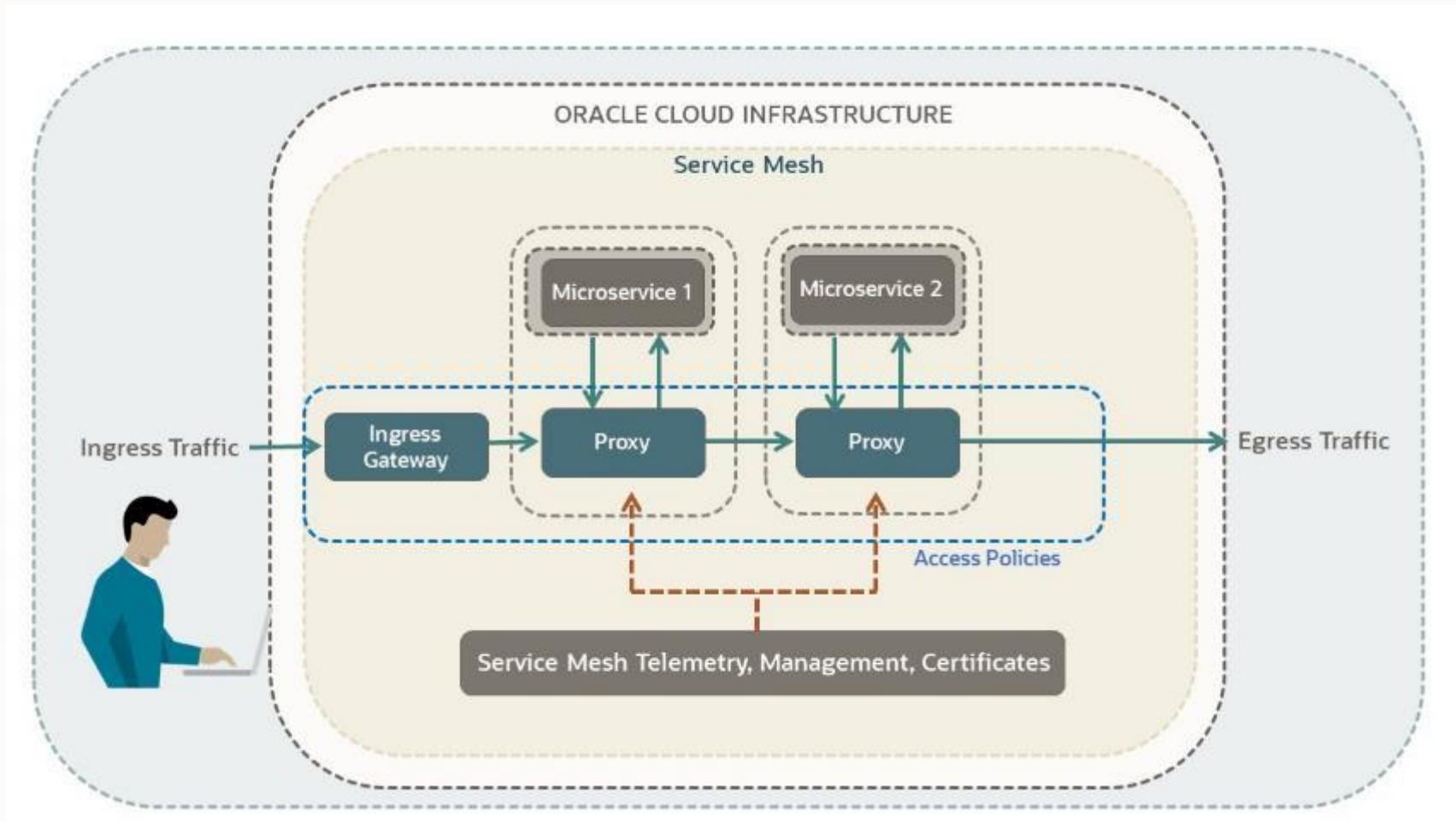
Métricas Prometheus en Grafana.  
Acceso al OKE desde CLI Shell: Pods, Services, Deployments



# Caso de Uso: Myapp



# Visión de Oracle para el Service Mesh



# OCI Service Mesh: Asegure, Observe y conecte sus microservicios fácilmente

Un servicio gratuito y gestionado para simplificar el desarrollo y las operaciones de aplicaciones nativas de la nube

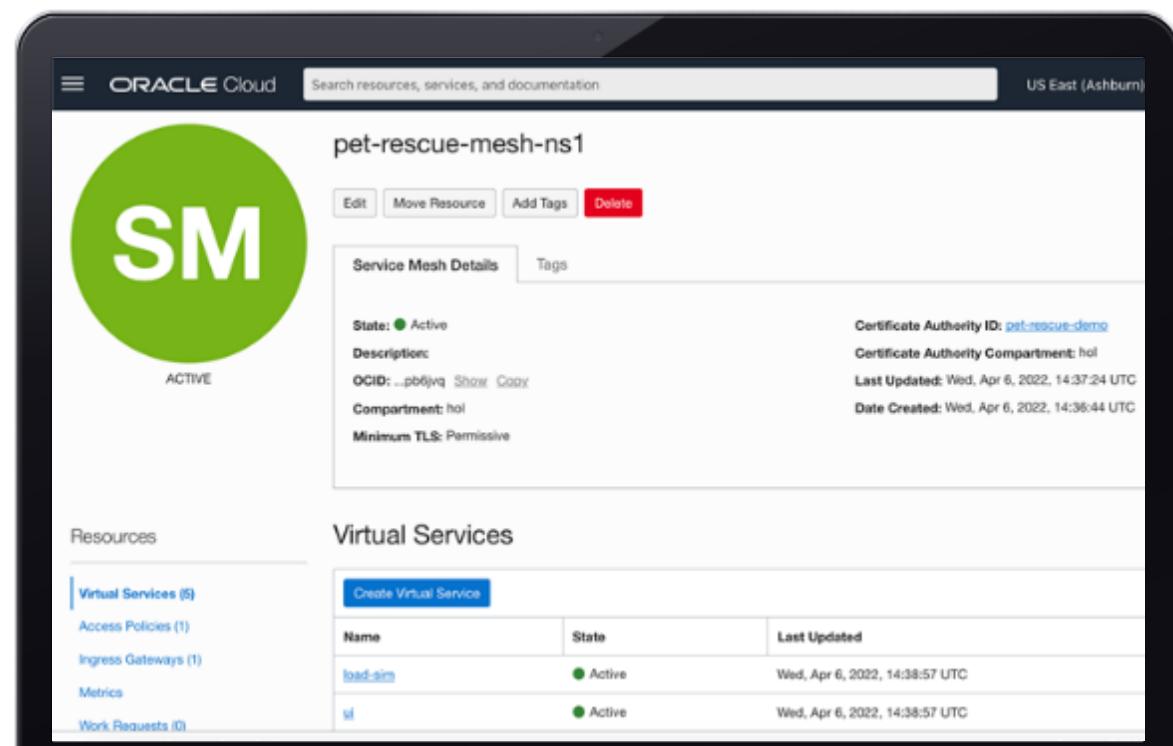
Aumente la productividad del desarrollador

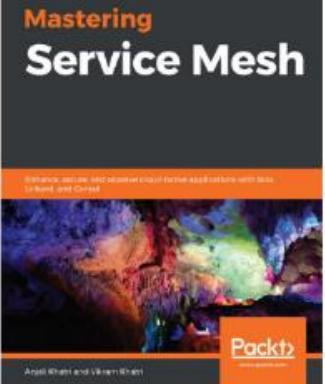
Mejore la eficiencia operativa

Asegure la carga de trabajo

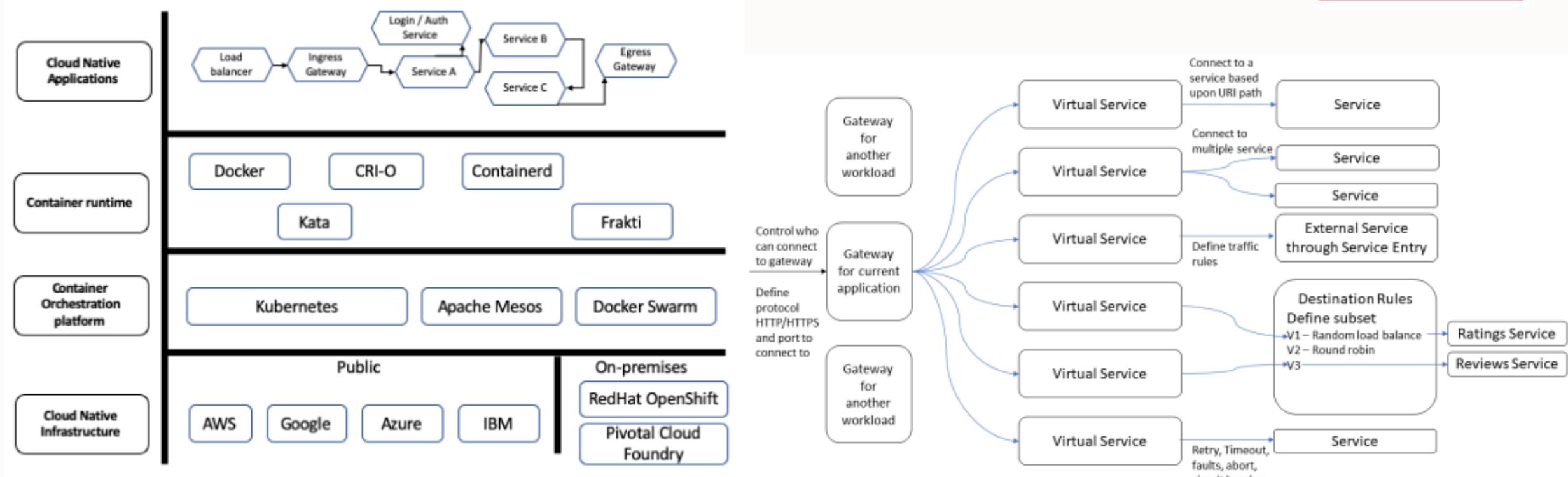
Habilite la visibilidad punta-a-punta

Conecte microservicios dinámicamente





Continue



# La selección importa y nuestro énfasis en estándares abiertos y flexibles

Soporte completo de *Open Source*, tecnología de 3eros, y herramientas de ecosistemas que permite la innovación, portabilidad y permitir un optimo TCO para cualquier recurso ejecutándose en OCI

Servicios gestionados  
basados en código abierto



Implemente lo que desee, asegurando  
ops sin interrupciones y un menor TCO



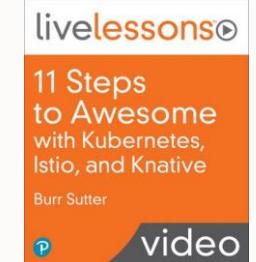
Comunidades a las que contribuimos



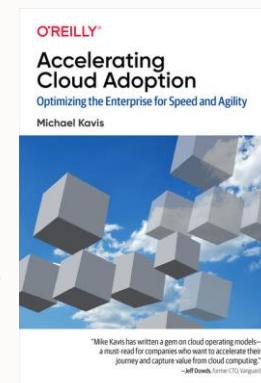
# No siga la moda...

Conversación interna y sincera

FaaS Kubernetes Players



Cloud maturity

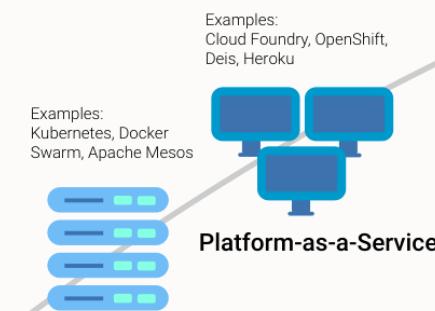


"Mike Kavis has written a gem on cloud operating models—a must-read for companies who want to accelerate their journey and capture value from cloud computing." —Jeff Bezos, former CEO, Amazon

↑

Increasing focus on business logic

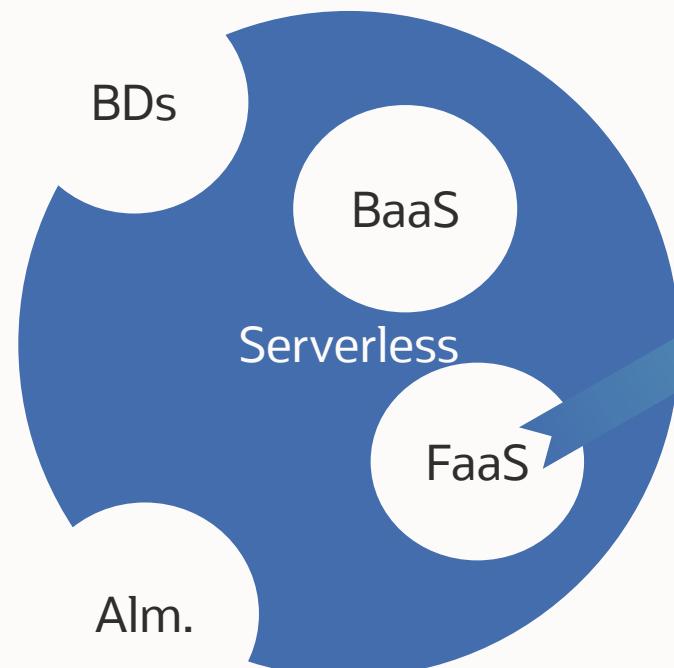
Decreasing concern (and control) over stack implementation



CNCF WG-Serverless Whitepaper v1.0

Cuestionario para seleccionar una opción preferida

# Que es Serverless....



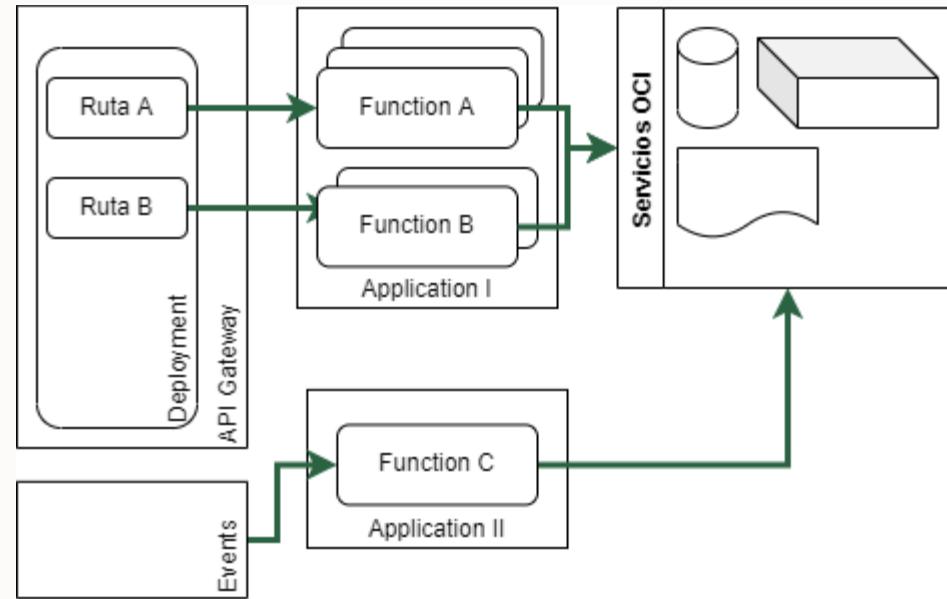
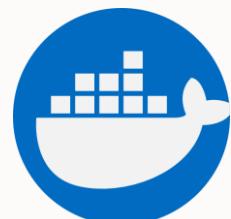
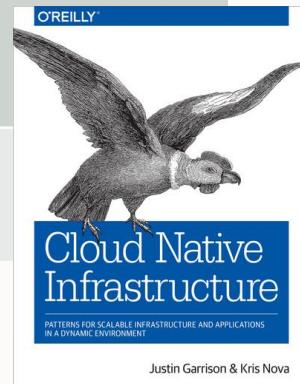
# Que es FaaS....



- Menor control
- Menor performance (*Cold Start*)
- Costos impredecibles precisos
- Función Limitada (Timeout, Memoria > Complejidad)
- Almacenamiento DB externa

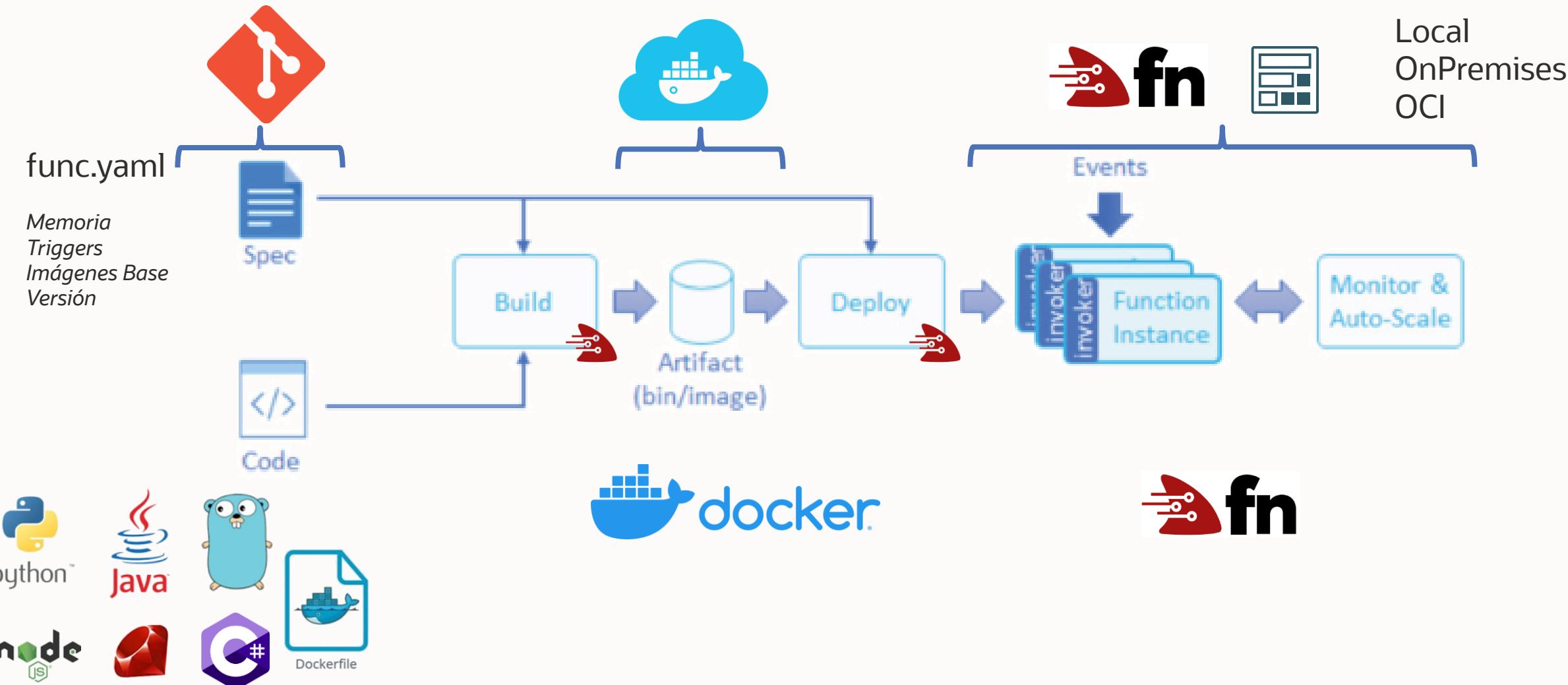
# Despliegue y Ejecución: Serverless

Function Fn: Serverless	
Visión Integral	Dentro de su App.
Escalable	Nativo
Abstracción	2 niveles: App y Function
Orquestación	Externa.
Descubrimiento Dinámico	N/A
Patrón Reconciliación	N/A

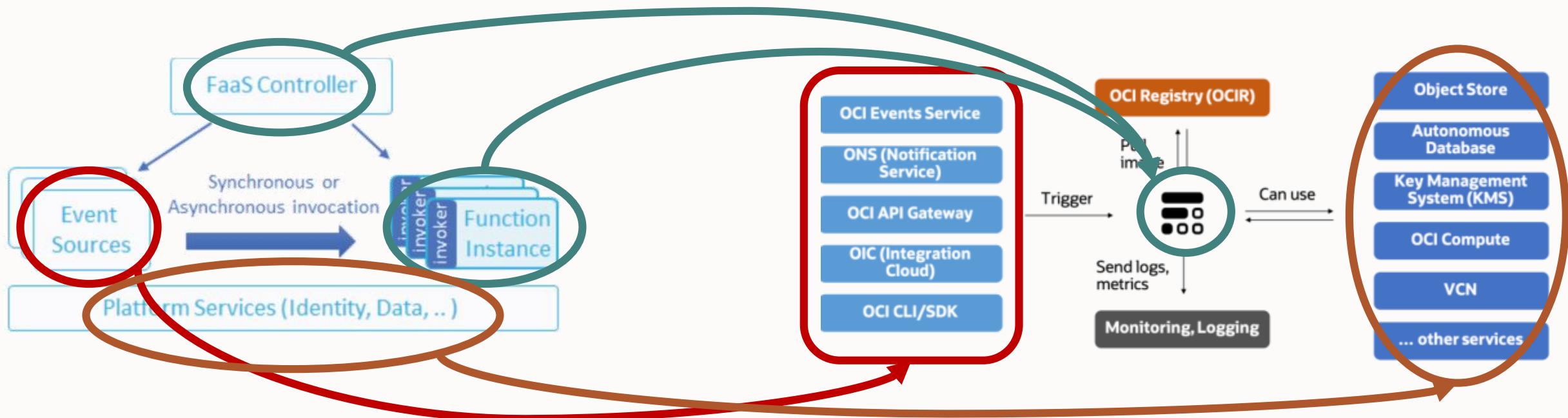


1. Open Source – Apache 2.0 License
2. Multisite - Multicloud
3. Mejora la experiencia Dev-QA y el comentario “En mi maquina funciona”
4. Container Native - Autoescalable
5. Modelo de Programación – Diseño, Costo, **TODOS** los lenguajes (Hotwrap)
6. Agnóstico al orquestador

# Proceso de Desarrollo y Despliegue



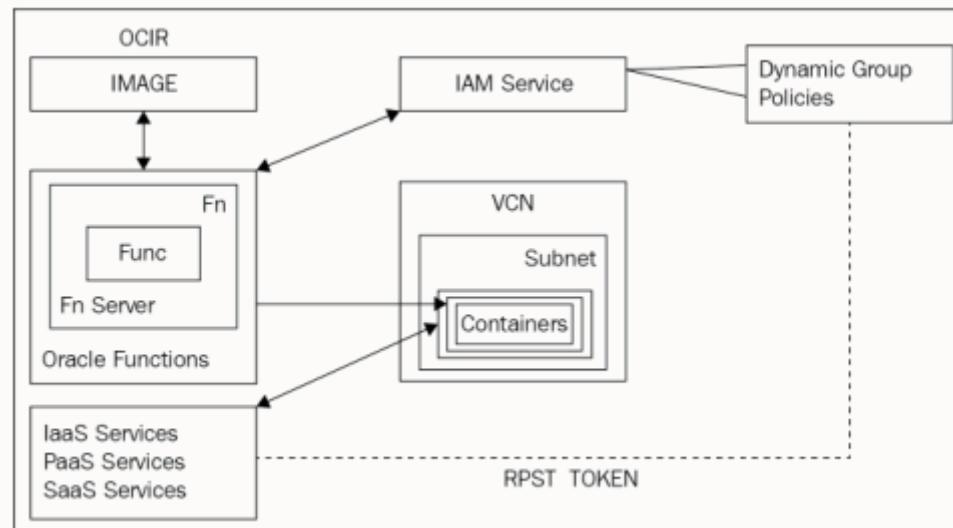
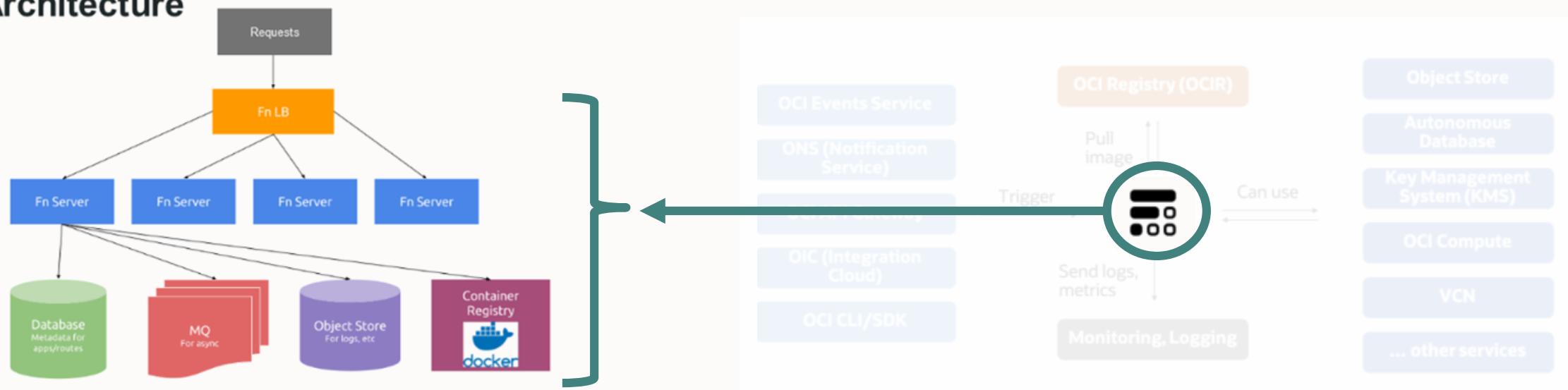
# Modelo de Procesamiento



Se configuran los **Eventos** para hacer las llamadas a las Apps/Functions.  
El **FaaS Controller** (Oracle Functions) se encarga de balancear e instanciar las Functions  
La integración es nativa desde las Functions hacia los **Servicios de la Nube**.

# Construcción Local y ambientes separados

## Fn Architecture



## Seguridad y Configuración:

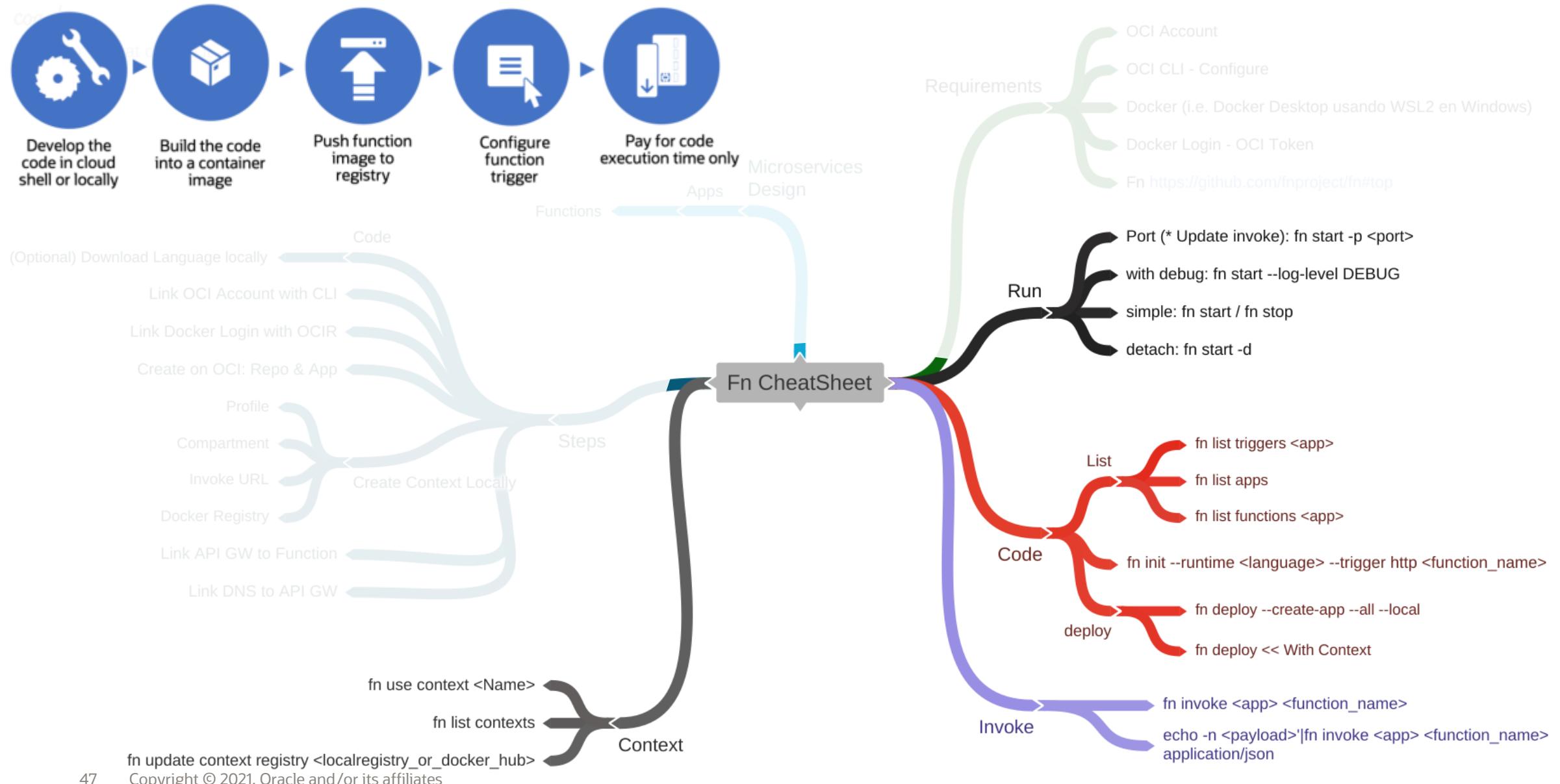
- Subred
- Firewall de entrada y salida
- Repositorio de Imágenes Docker
- Permisos de ejecución



# Conceptos de Functions

- **Application**
  - Colección de funciones
  - Conectadas a la VCN/Subredes usadas para tráfico de salida
  - Unidad de aislamiento de carga de trabajo
  - Habilitar o deshabilitar registros, y se establecen parámetros de configuración comunes
- **Function**
  - Imagen Docker (Almacenada en el Container registry)
  - Metadata: Localización de la imagen, memoria, timeout, parámetros de configuración
- **Fn CLI**
  - Comandos para compilar y empaquetar el código de la función como una imagen de Docker e insertar en el registro
  - Genera código Functions sencillo

# Mindmap – Functions CLI



# Desarrollo en Functions

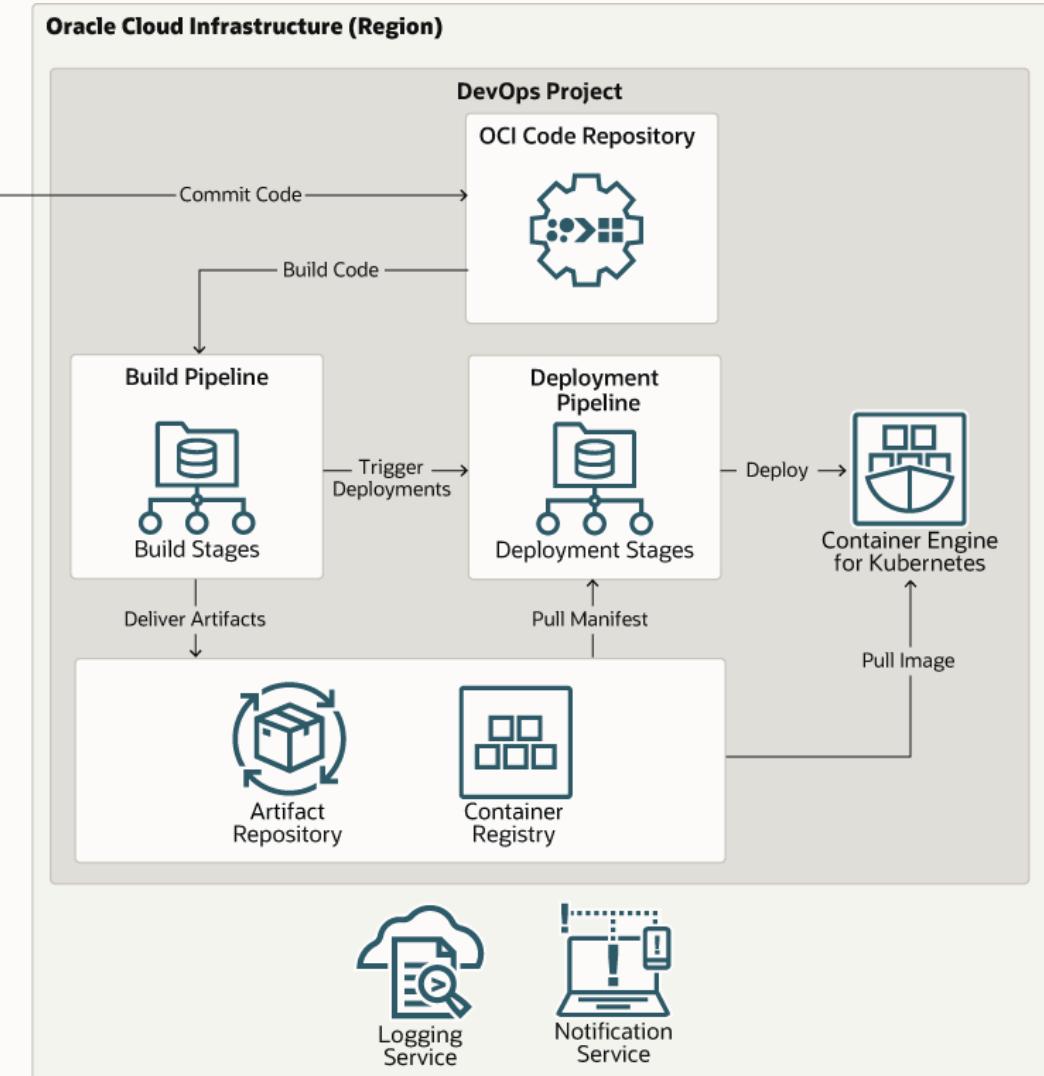
Ambiente de Desarrollo	Frameworks & Lenguajes	Empaquetado	Herramientas en Ciclo de Vida
<ul style="list-style-type: none"><li>✓ OCI Cloud Shell</li><li>✓ Maquina Local</li><li>✓ Instancia en OCI</li></ul>	<ul style="list-style-type: none"><li>✓ Fn Java FDK</li><li>✓ Fn Python FDK</li><li>✓ Fn Node FDK</li><li>✓ Fn Go FDK</li><li>✓ Fn Ruby FDK</li><li>✓ Micronaut*</li></ul>	<ul style="list-style-type: none"><li>✓ Fn Dockerfile</li><li>✓ BYO Dockerfile</li><li>✓ Graal Native*</li></ul>	<ul style="list-style-type: none"><li>✓ Fn Server</li><li>✓ Fn CLI</li><li>✓ OCI Console</li><li>✓ OCI CLI</li><li>✓ Docker CLI</li><li>✓ Terraform</li><li>✓ Visual Builder Studio CI-CD</li></ul>

\* -- 3<sup>rd</sup> Party/Oracle contribution

# Ejemplo del DevOps

<https://docs.oracle.com/en/solutions/ci-cd-pipe-oci-devops/index.html>

Lanzamiento del Resource Manager  
Generacion del Auth Token y usuario (Prueba  
desde Docker Login)



# Usos de casos comunes

## *Código ejecutado en respuesta a Eventos*



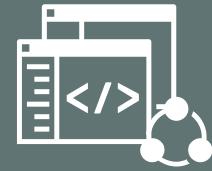
Gobierno basado  
en Eventos



Web, Mobile API  
Backends

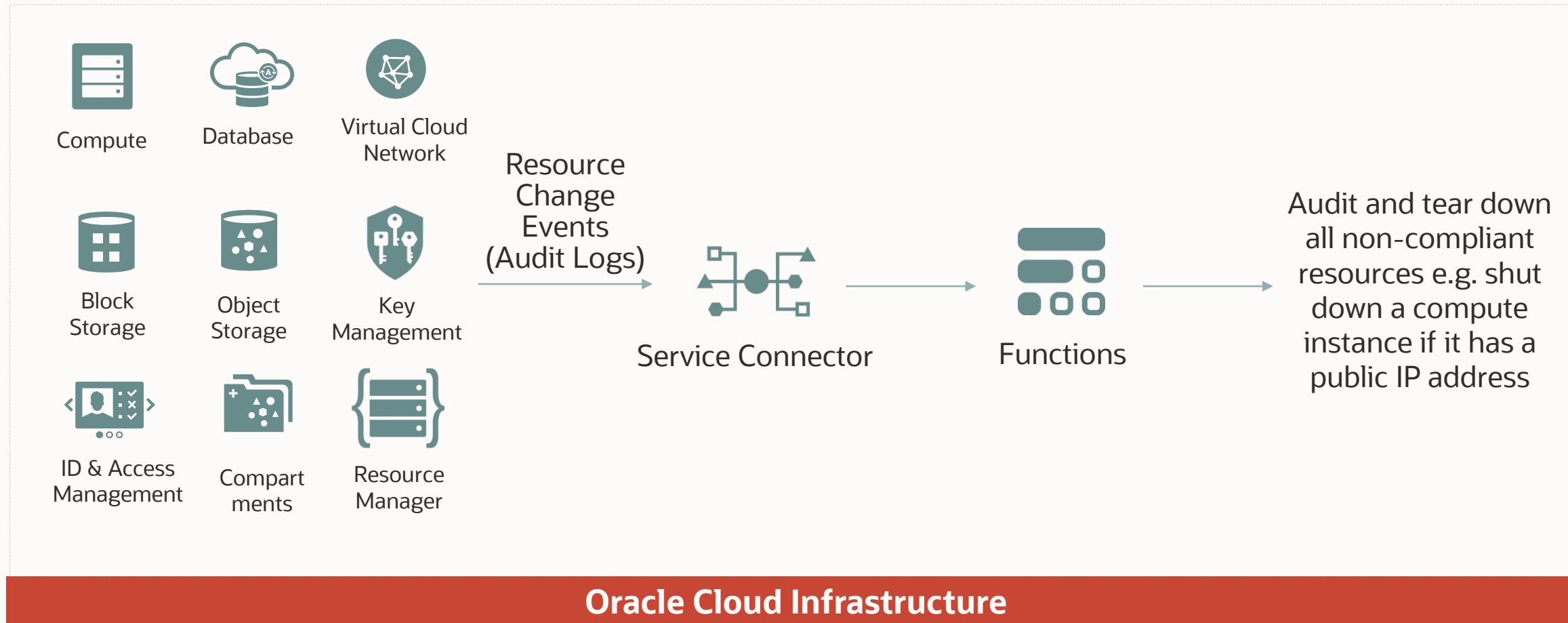


Proc. en tiempo  
real de Archivos,  
Streaming

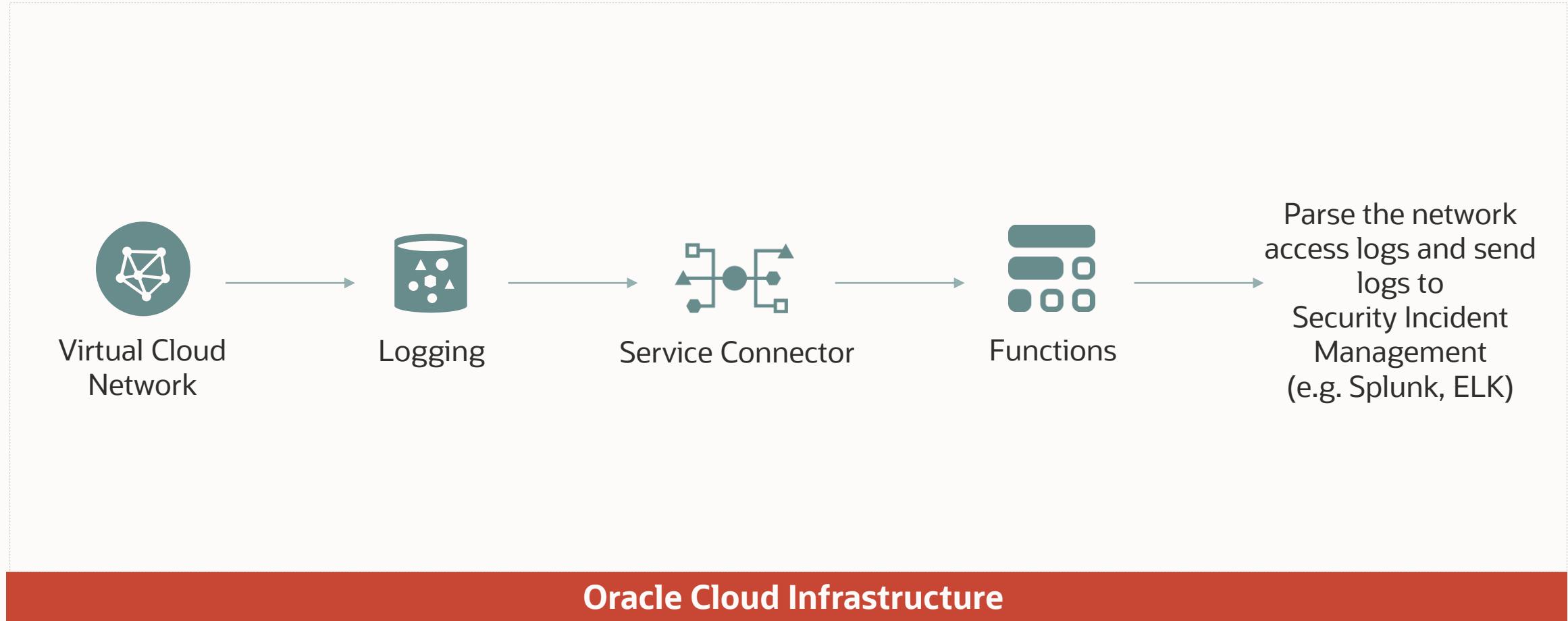


Machine Learning,  
DevOps

# Use Case 1: Enforce Corporate Security Policies and Governance Rules

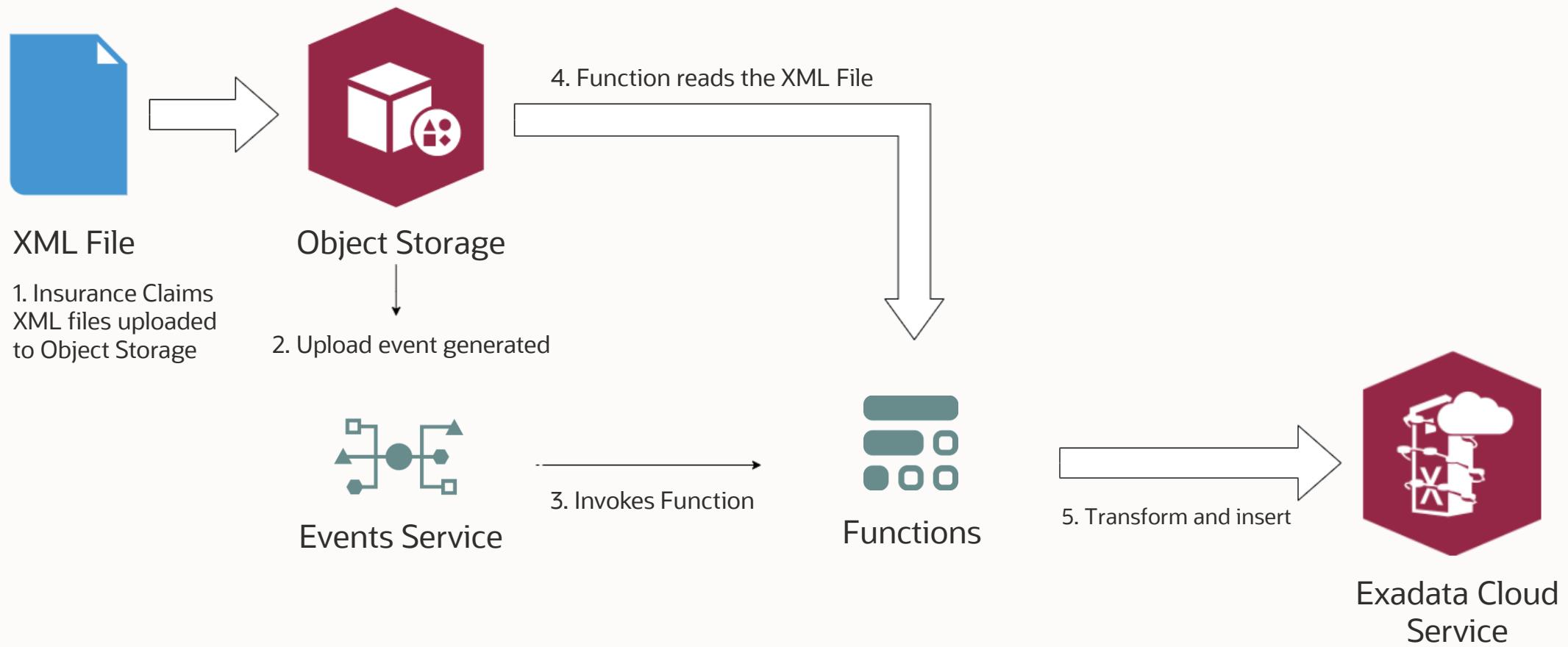


## Use Case 2: Ingest Access Logs in Security Incident Management

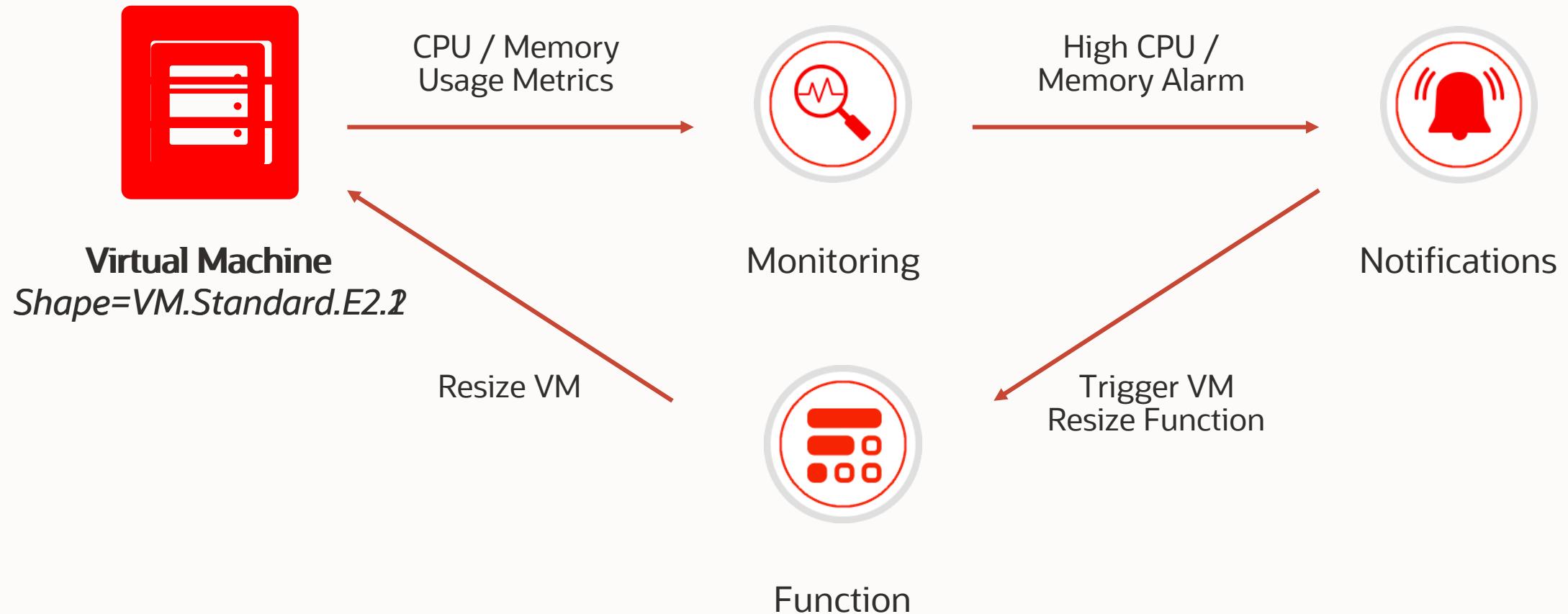


# Use Case 3: High Volume ETL Solution

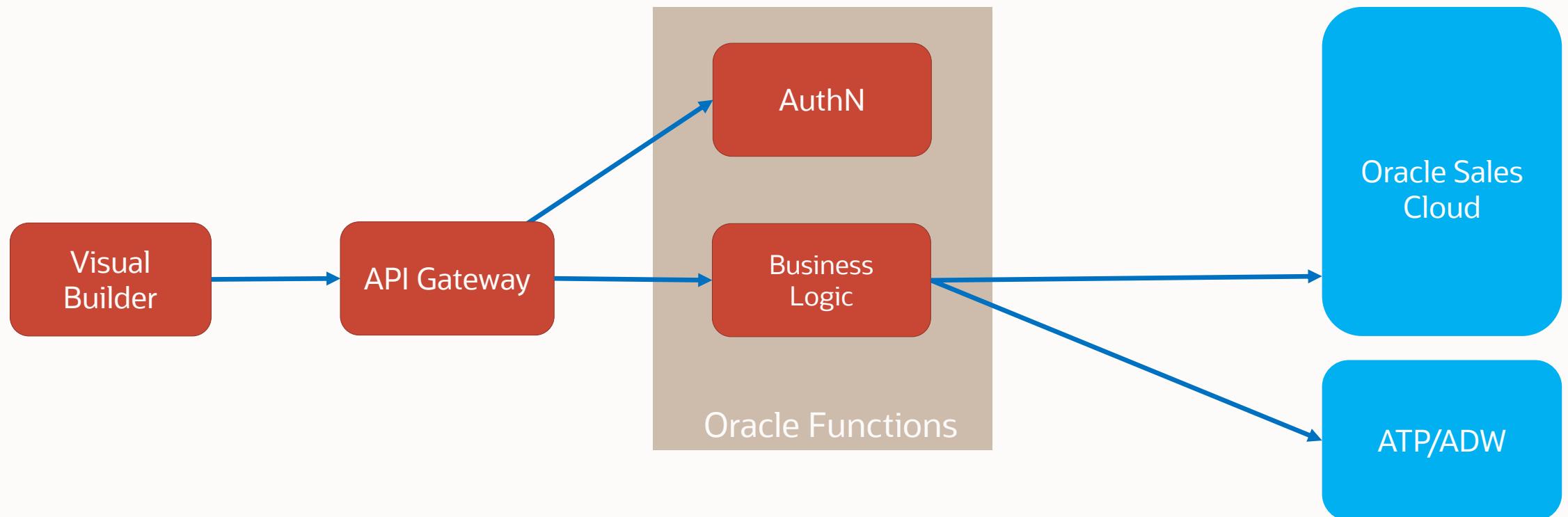
Processes ~1-1.5 million files per day



## Use Case 4: Resize Virtual Machine



# Use Case 5: Functions as API Backends to Extend Oracle Sales Cloud



# Functions Onboarding



- [Quick Start Guides](#)
  - [OCI Live Labs](#)
  - [OCI Tutorials](#)
- 
- [Ejemplos en GitHub](#)
  - [Service Documentation](#)
  - [Blogs](#)
- 
- [Reference Architectures](#)
  - [Solution Playbooks](#)

# Tutorials Fn y Sample Code

## Explore Fn

- [Create Apps with Fn](#) - Learn how to group your functions into an application/API and deploy them together.
- [Create a Function with a Docker Container](#) - This tutorial provides a simple example of how to define an Fn function using a custom built Docker container.
- [Create a Function with a Linux Command and HotWrap](#) - This tutorial provides an example of how to define an Fn function using Linux commands, HotWrap, and a custom Docker container.
- [Create a Function from a Docker image that contains a Node.js app with Oracle DB Support](#)
- [Fn Development with Multiple Contexts](#) - Shows how to setup and use multiple Fn configuration contexts for development.
- [Using Fn RuntimeContext with a Function](#) - See how to set Fn Application and Function variables and access them in your function using the Runtime Context.

## Test and Monitor Functions

- [Monitor Fn metrics with Grafana and Prometheus](#) - Learn how to view Fn server metrics with Prometheus and Grafana.
- [Troubleshoot and Log functions](#) - Resolve issues at both development and deployment time.

## Orchestrate with Fn Flow

Fn Flow provides a way to orchestrate functions to build sophisticated applications, initially using Java, and soon with other programming languages.

- [Flow 101](#)
- [Flow 102](#)
- [Fn Flow Saga](#) - In this tutorial you will use Fn Flow to implement a Java travel booking system that leverages functions written in a variety of languages.

<https://fnproject.io/tutorials/>

# Costos – Functions y API Gateway

## OCI Functions

Invocation Pricing		
	Pay as You Go	Metric
First 2 million per month	Free	Function invocation
Over 2 million per month	US\$0.0000002	Function invocation

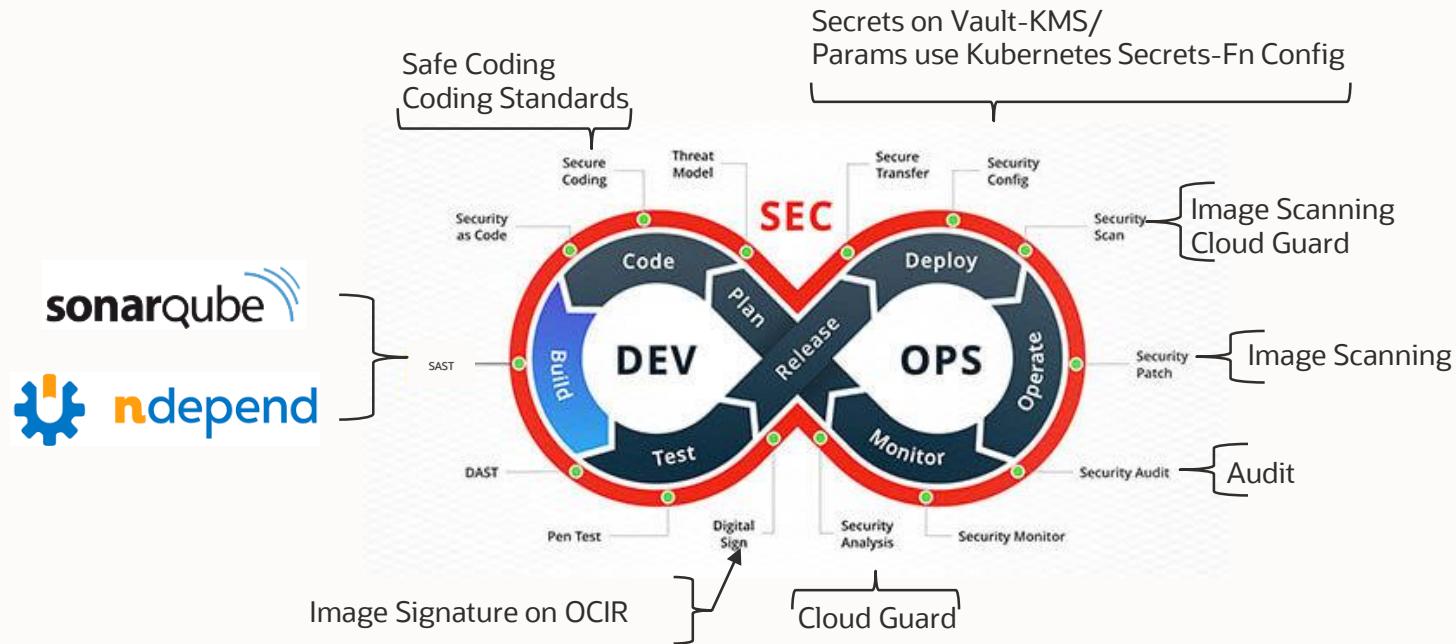
Execution Time Pricing		
	Pay as You Go	Metric
First 400,000 per month	Free	Gigabyte memory-seconds
Over 400,000 per month	US\$0.00001417	Gigabyte memory-seconds

## OCI API Gateway

API Management			
Service	Pay as You Go	Metric	
API Gateway	\$3.00	1 Million API Calls Per Month	

\* Tomado de <https://www.oracle.com/cloud-native/functions/> y <https://www.oracle.com/cloud-native/api-management/>

# Aspectos de Seguridad dentro del ciclo de vida de Desarrollo



Foco en desarrollo o arquitectura, en vez de equipos especializados (Perimetral – WAF, DDoS, LB).

# Ejemplos Image Scanning & Logging

```
import json
import pymssql
import logging

from fdk import response

def handler(ctx, data: io.BytesIO=None):
    logging.getLogger().info("Inside Event-Count function - 455 pm")
    conn = pymssql.connect(server="129.213.154.39",port=1444, user = "sa", pas
    cursor = conn.cursor()
    #cursor.execute('SELECT count(*) FROM Events')
    cursor.execute('SELECT * FROM Events order by Date desc')
    logging.getLogger().info('Getting information from SQL Command: ' + str(cu
    row = cursor.fetchone()
    data = {
        "eventCount": row[0]
    }
    conn.close()

    return response.Response(
        ctx, response_data=json.dumps(data),
        headers={"Content-Type": "application/json"}  
    )
```

Container Registry in CO-Francisco-Moreno

Create repository

vulnerability

conve

Scan details

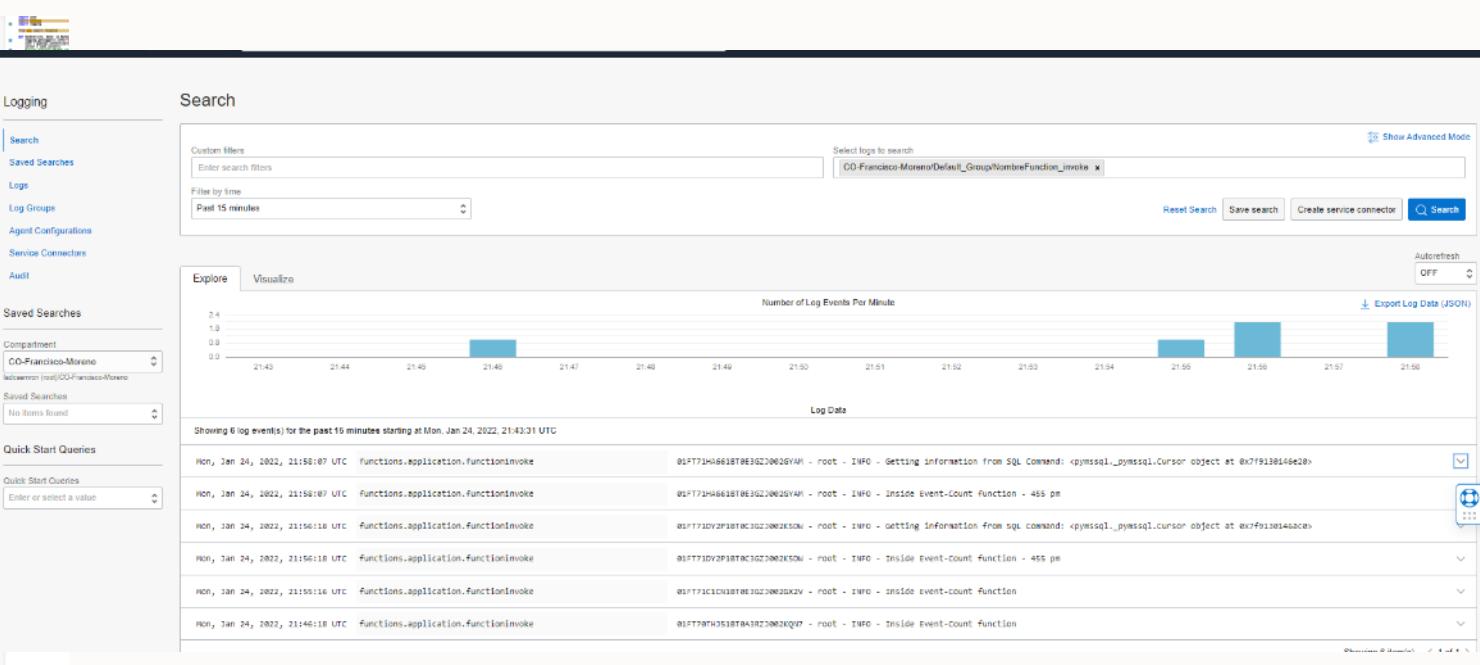
Repository: vulnerability

Image: conve

Risk level: Critical

Issues found

Issue	Risk level	Description
<a href="#">CVE-2021-45046</a>	Critical	It was found that the fix to address CVE-2021-44228 in Apache Log4j Thread Context Map (MDC) input data when the logging configuration Context Map pattern (%X, %mdc, or %MDC) to craft malicious input disable JNDI functionality by default.
<a href="#">CVE-2021-44228</a>	Critical	Apache Log4j 2.0-beta9 through 2.15.0 (excluding security releases) protect against attacker controlled LDAP and other JNDI related endpoints loaded from LDAP servers when message lookup substitution is enabled (2.12.3, and 2.3.1), this functionality has been completely removed. No Logging Services projects.
<a href="#">CVE-2021-43045</a>	High	A vulnerability in the .NET SDK of Apache Avro allows an attacker to a applications using Apache Avro version 1.10.2 and prior versions. Use
<a href="#">CVE-2014-3643</a>	High	jersey: XXE via parameter entities not disabled by the jersey SAX pars
<a href="#">CVE-2017-18640</a>	High	The Alias feature in SnakeYAML 1.18 allows entity expansion during a



# Functions: Additional Features

- [Micronaut Functions and API example](#). Has Java and Java Native support. Has Gradle support.
- [Managing Functions Lifecycle with Visual Builder Studio \(CI-CD\)](#)
- [LiveLab: Develop/Deploy a Modern Application with Node.js Functions & MySQL Database Service](#)
- Functions with Oracle NoSQL DB
- [Blog: Node/ADB functions](#) using a custom Dockerfile e.g., install Oracle Client for Node
- Service Connector Hub-triggered functions e.g., [Send OCI Logs to Datadog](#)
- Function chaining with Integration Cloud

# Por donde iniciar?

Contenedores y Functions	<b>Documentation &amp; tutorials</b> <ul style="list-style-type: none"><li><a href="#">Container Engine for Kubernetes</a></li><li><a href="#">Container Registry</a></li><li><a href="#">Functions</a></li></ul>	<b>Solutions Playbooks and Reference Architectures</b> <ul style="list-style-type: none"><li><a href="#">Set up a Kubernetes cluster for deploying containerized applications</a></li><li><a href="#">Deploy Cloud Native Apps with MySQL</a></li><li><a href="#">OCI for AWS Professionals</a></li><li><a href="#">OCI for Azure Professionals</a></li></ul>	<b>Labs</b> <ul style="list-style-type: none"><li><a href="#">Build Cloud Native Java Apps with Micronaut and Graal VM</a> <a href="#">OCI Certification</a></li><li><a href="#">Automate the Deployment of Java Apps to OCI/DevOps Pipeline</a> <a href="#">Developer Community</a></li></ul>
API y Streaming	<b>Training:</b> <ul style="list-style-type: none"><li><a href="#">Coursera Course: API Gateway</a></li></ul>	<ul style="list-style-type: none"><li><a href="#">API For Developers: Learn how to design, develop and manage APIs</a></li></ul>	
AI y Data Services	<b>AI Workshops</b> <ul style="list-style-type: none"><li><a href="#">Digital Assistant</a></li><li><a href="#">Language</a></li><li><a href="#">Speech</a></li></ul>	<ul style="list-style-type: none"><li><a href="#">Vision</a></li><li><a href="#">Anomaly Detection</a></li><li><a href="#">Forecasting</a></li></ul>	<b>Blogs</b> <ul style="list-style-type: none"><li><a href="#">Develop XR With Oracle Ep 3: Computer Vision AI, ML, and Metaverse</a></li><li><a href="#">ML and AI blog</a></li></ul>
DevOps y Application Development	<ul style="list-style-type: none"><li><a href="#">DevOps service example workflows and terraform automation</a></li></ul>	<ul style="list-style-type: none"><li><a href="#">OCI DevOps Documentation</a></li></ul>	<ul style="list-style-type: none"><li><a href="#">DevOps CI/CD Reference Architecture</a></li><li><a href="#">Deploy a Jenkins CI/CD pipeline</a></li><li><a href="#">OCI DevOps Professional Certification</a></li><li><a href="#">DB Operations with DevOps</a></li></ul>





# Preguntas?

## Libro De Referencia

KHATRI, Anjali y otros. Mastering Service MeshPackt Publishing, May 2020.

<https://learning.oreilly.com/library/view/mastering-service-mesh/9781789615791/>

## Example DevOps Service

<https://docs.oracle.com/en/solutions/ci-cd-pipe-oci-devops/index.html>

## OCI Cloud Native

<https://www.oracle.com/cloud-native/>

## Ejemplo Cloud Native de los Demos

<https://github.com/oracle-quickstart/oci-cloudnative/>

## Oracle Kubernetes Engine

<https://www.oracle.com/cloud-native/container-engine-kubernetes/>

## Oracle Functions

<https://www.oracle.com/cloud-native/functions/>

## Fn Project

<https://fnproject.io/>

<https://app.pluralsight.com/library/courses/serverless-programming-fn-project>

<https://hackernoon.com/playing-with-the-fn-project-8c6939cf5cc>

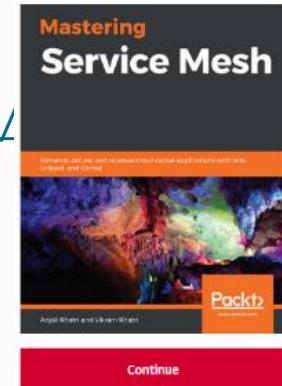
<https://dev.to/kh40sz3r0/serverless-functions-with-fn-project-64e>

## Events Service (usando CNCF Events)

<https://www.oracle.com/cloud-native/events-service/>

## API Management

<https://www.oracle.com/cloud-native/api-management/>





# Preguntas?

## Service Mesh

<https://docs.oracle.com/en-us/iaas/Content/service-mesh/home.htm>  
<https://www.oracle.com/cloud/cloud-native/service-mesh/faq/>

## Nuevas VCN para Kubernetes

<https://blogs.oracle.com/cloud-infrastructure/post/announcing-vcn-native-pod-networking-for-kubernetes-in-oci> (15-Julio-22)

## Enable Service Mesh on OKE

<https://docs.oracle.com/en/solutions/oci-service-mesh-oke/index.html#GUID-12216A44-2E62-48D8-8C89-864335BE9EF6>

## Mas Información

<https://oracle.com/application-development>

## Verazzano

<https://www.oracle.com/java/verrazzano/>  
<https://github.com/verrazzano/verrazzano>

## Hands-on Labs

<https://go.oracle.com/hols>

<https://apexapps.oracle.com/pls/apex/dbpm/r/livelabs/home>

## Architecture Center - Reference Architecture

<https://docs.oracle.com/solutions/?lang=en>



# Preguntas?

## Example DevOps Service

<https://docs.oracle.com/en/solutions/ci-cd-pipe-oci-devops/index.html>

## OCI Cloud Native

<https://www.oracle.com/cloud-native/>

## Ejemplo Cloud Native de los Demos

<https://github.com/oracle-quickstart/oci-cloudnative/>

## Oracle Kubernetes Engine

<https://www.oracle.com/cloud-native/container-engine-kubernetes/>

## Oracle Functions

<https://www.oracle.com/cloud-native/functions/>

## Fn Project

<https://fnproject.io/>

<https://app.pluralsight.com/library/courses/serverless-programming-fn-project>

<https://hackernoon.com/playing-with-the-fn-project-8c6939cf5cc>

<https://dev.to/kh40sz3r0/serverless-functions-with-fn-project-64e>

## Events Service (usando CNCF Events)

<https://www.oracle.com/cloud-native/events-service/>

## API Management

<https://www.oracle.com/cloud-native/api-management/>

## Verazzano

<https://www.oracle.com/java/verrazzano/>

<https://github.com/verrazzano/verrazzano>

## Mas Información

<https://oracle.com/application-development>

## Hands-on Labs

<https://go.oracle.com/hols>

<https://apexapps.oracle.com/pls/apex/dbpm/r/livelabs/home>

## Architecture Center - Reference Architecture

<https://docs.oracle.com/solutions/?lang=en>



# Preguntas?



## Example DevOps Service

[https://learning.oreilly.com/videos/11-steps-to/9780135945346/9780135945346-esak\\_01\\_10\\_01\\_00/](https://learning.oreilly.com/videos/11-steps-to/9780135945346/9780135945346-esak_01_10_01_00/)  
<https://www.bmc.com/blogs/serverless-faas/#:~:text=This%20offers%20users%20more%20flexibility,not%20limited%20to%20creating%20functions.>  
<https://kubesphere.io/blogs/serverless-vs-faas/>