IBM **Developer**

IBM

Build Smart on Kubernetes

World Tour

Orquestación de contenedores con Red Hat OpenShift

27 de Julio, 2020

Kubernetes with OpenShift

World Tour

Luis Reyes Oliva, IBM Developer Advocate

IBM

Red Hat

Kubernetes with OpenShift

World Tour

Mikel Díez Parra, Head of Innovation, IBM Spain, Portugal, Greece & Israel

Kubernetes with OpenShift

World Tour

Agenda

10:00 – Bienvenida

10:10 – Keynote Innovation Tech Talk

Ponente: Mikel Díez Parra, Head of Innovation, IBM Spain, Portugal, Greece & Israel

10:30 – Introducción a Contenedores, K8s y registro en IBM

11:15 – Taller para principiantes: Introducción a Red Hat OpenShift en IBM Cloud 4.3

12:00 – Taller avanzado: Operadores en OpenShift 4.3. Desplegando un servicio Cloudant

Ponente: Luis Reyes Oliva, IBM Developer Advocate

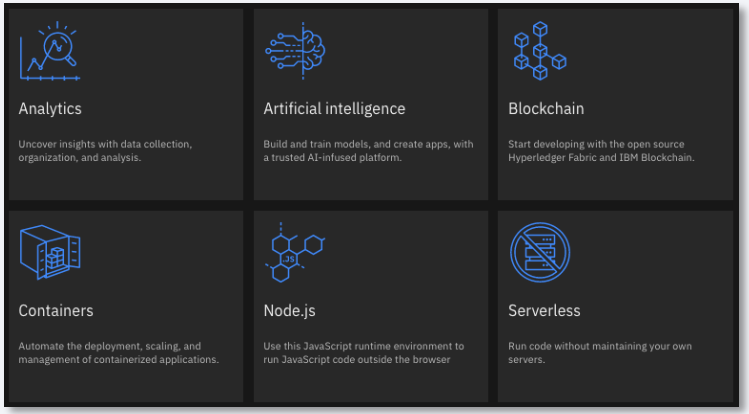
12:30 - Preguntas y respuestas

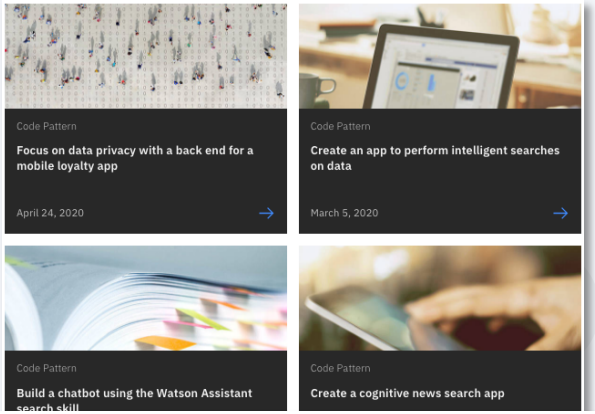
IBM

https://developer.ibm.com/

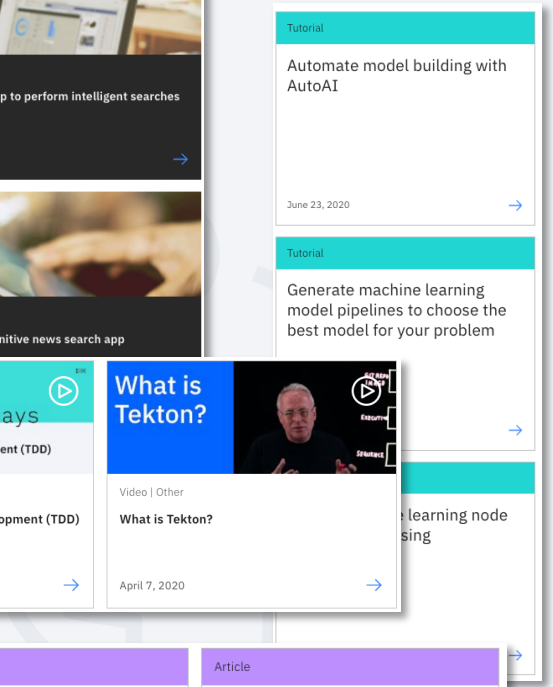
Kubernetes with OpenShift

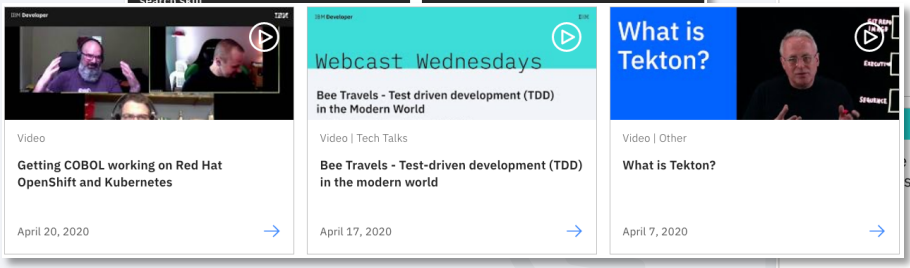
World Tour

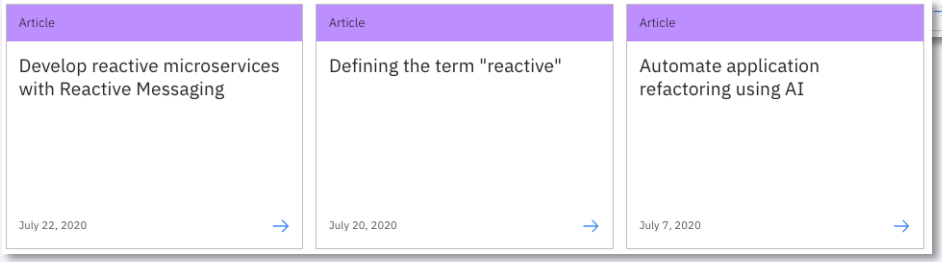




Red Hat



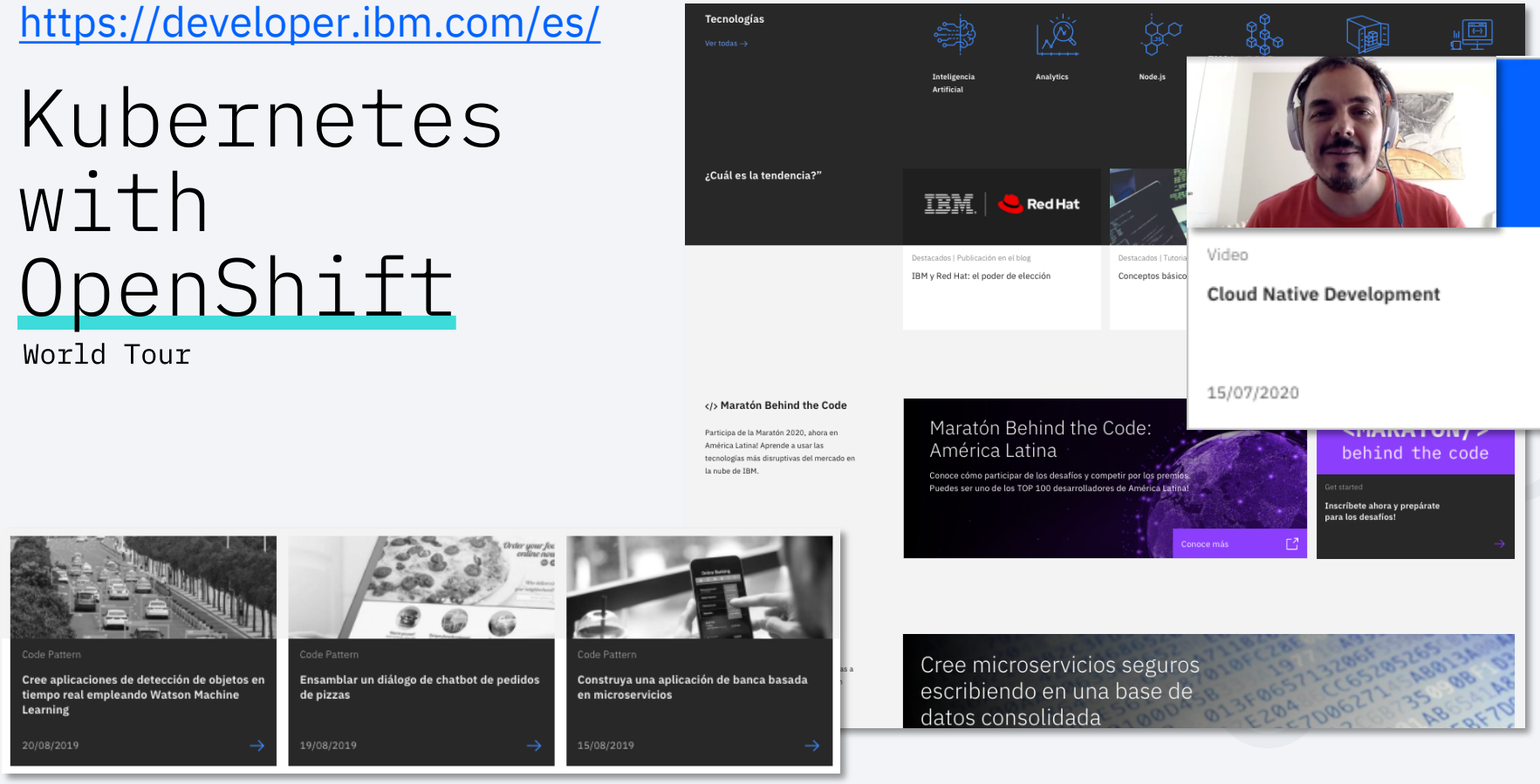




https://developer.ibm.com/es/

Kubernetes with OpenShift

World Tour



Kubernetes with OpenShift

World Tour

En esta sesión, revisaremos los elementos esenciales de la tecnología de contenedores y orquestación, en particular **Docker** y **Kubernetes**, y cómo trabajan juntos para habilitar nuevos modelos de desarrollo y operación de aplicaciones

10:30

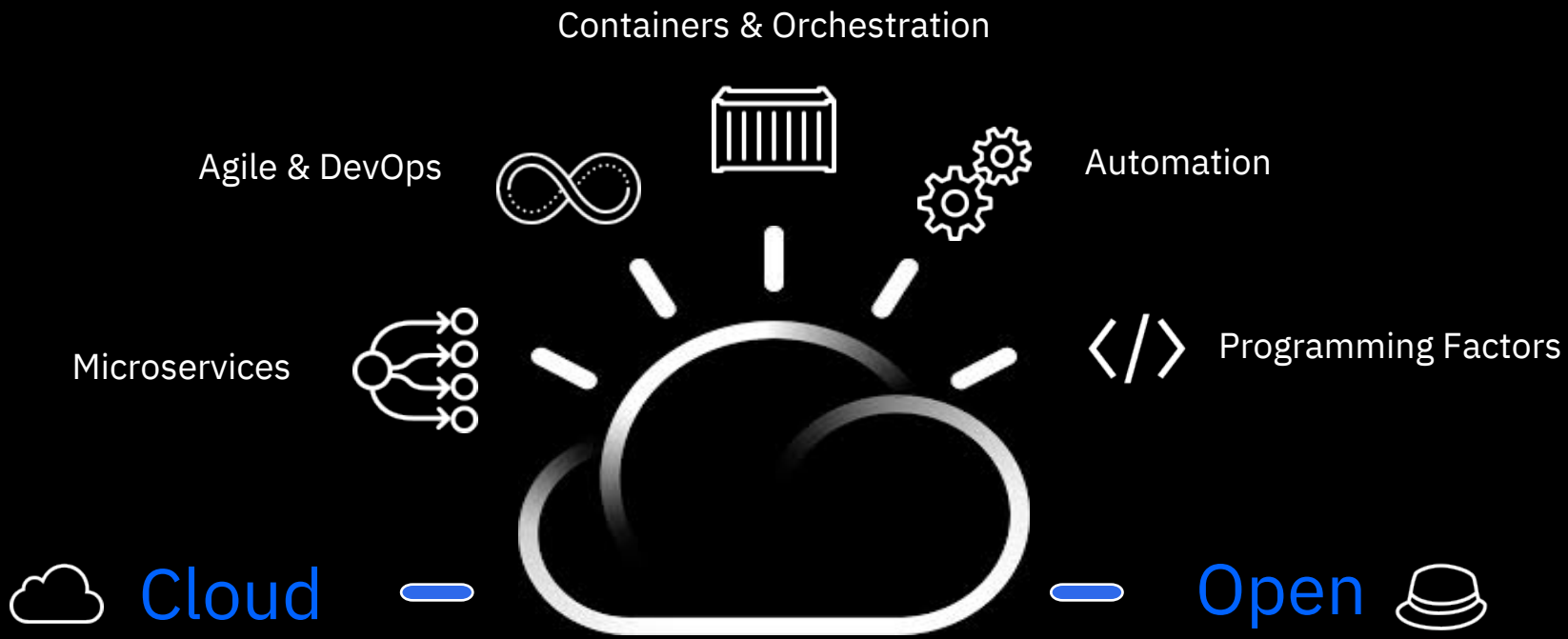
Introducción a Contenedores, K8s y registro en IBM

‘Cloud Native Development’ y la tormenta perfecta

Cloud-native development is an approach to building and running applications that exploits the advantages of the cloud computing delivery model. The Cloud Native Computing Foundation (CNFC) talks about using "… an open source software stack to deploy applications as microservices, packaging each part into its own container, and dynamically orchestrating those containers to optimize resource utilization. Cloud native technologies enable software developers to build great products faster." (CNCF, 2019)

CLOUD NATIVE
COMPUTING FOUNDATION

Cloud Native | La tormenta (perfecta)



CNCF Cloud Native Landscape

2020-05-05T22:49:53Z 8a2f76a

Overwhelmed? Please see the CNCF Trail Map. That and the interactive landscape are at l.cncf.io

Greyed logos are not open source

kubernetes

App Definition and Development

Database | Streaming & Messaging | Application Definition & Image Build | Continuous Integration & Delivery

Orchestration & Management

Scheduling & Orchestration | Coordination & Service Discovery | Remote Procedure Call

Runtime

Cloud Native Storage

Provisioning

Automation & Configuration | Container Registry | Security & Compliance | Key Management

Platform

PaaS/Container Service

Observability and Analysis

Monitoring | Logging | Tracing | Chaos Engineering

Special

Kubernetes Certified Service Provider | Kubernetes Training Partner

CLOUD NATIVE Landscape

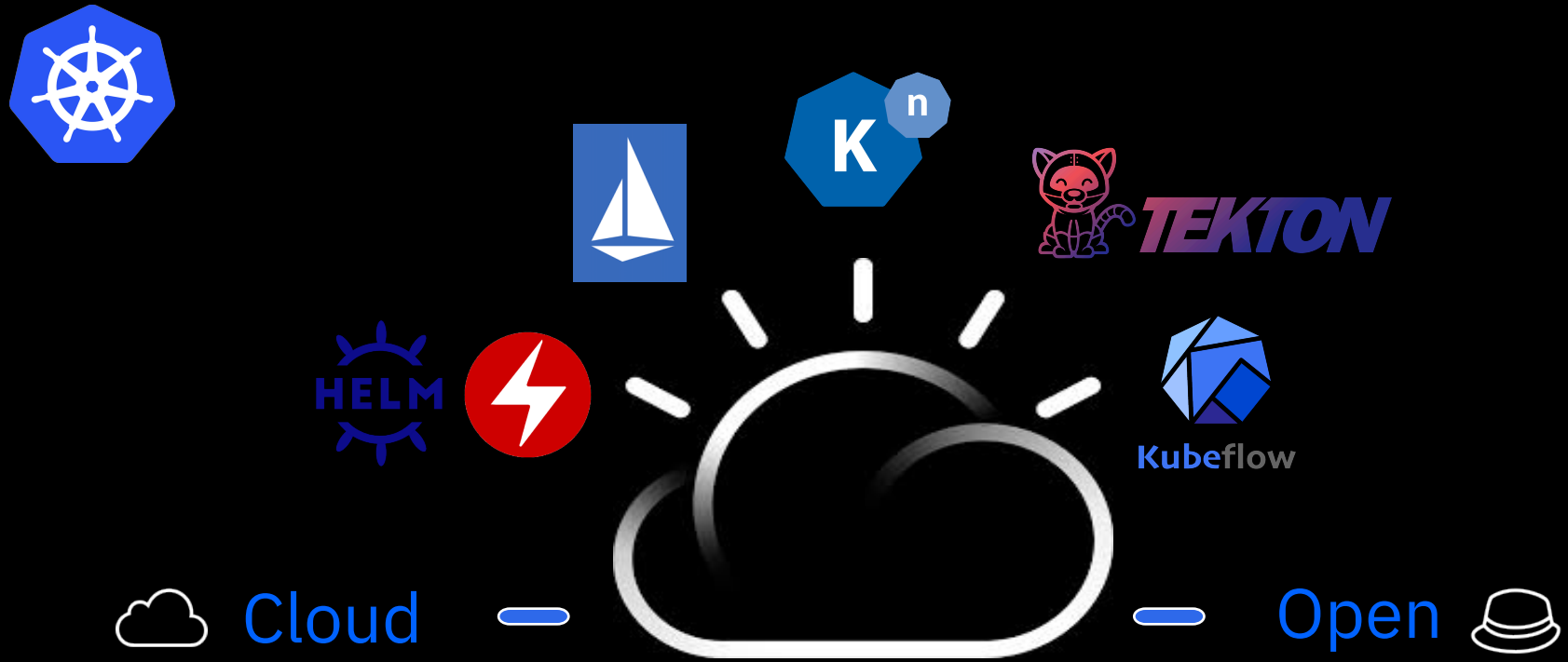
CLOUD NATIVE COMPUTING FOUNDATION

Redpoint Amplify

l.cncf.io

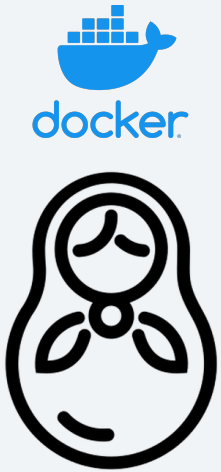
This landscape is intended as a map through the previously uncharted terrain of cloud native technologies. There are many Projects representing a particularly well-traveled path.

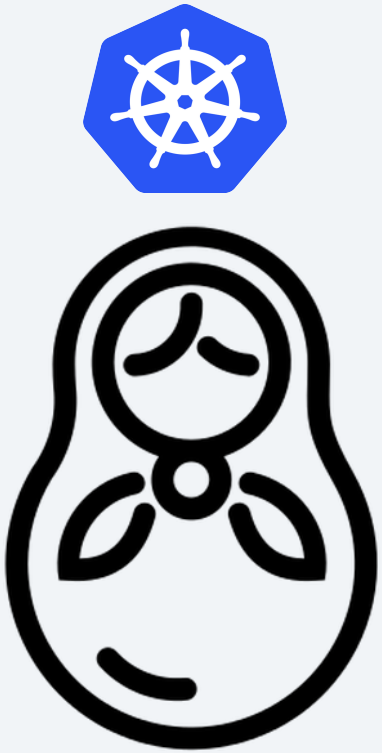
Y la tormenta arrecia…



Las frikyoskas

^^ Funcionalidad





>> Enterprise ready

Contenedores

¿Por qué surgen los contenedores?

Deb
The Developer

App.war

Guide

Deploy

App.war

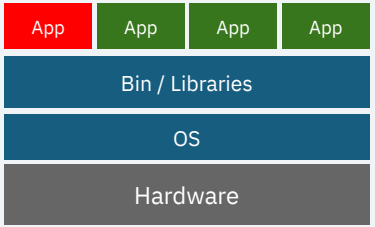
Guide

Mike
The Ops Guy

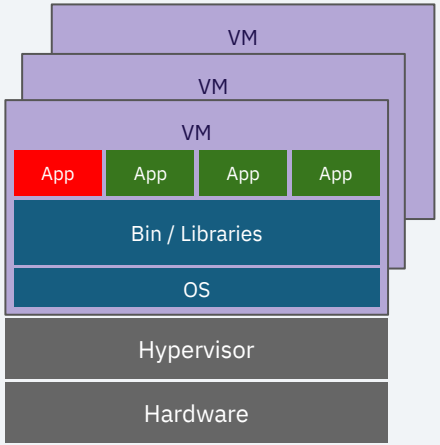
Compatibility, dependency, versions...

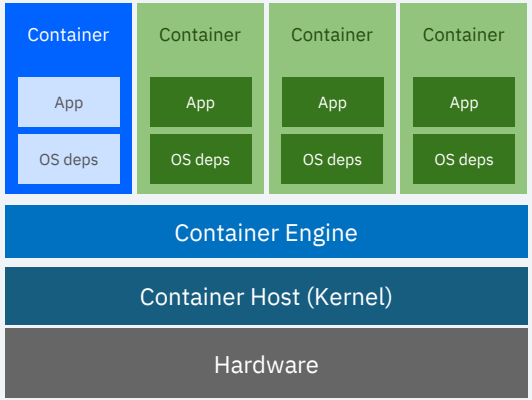
Long time setup

Different dev/test/prod environments



VMs vs Containers





- VMs are a better choice for running apps that require all of the operating system's resources and functionality when you need to run multiple applications on servers or have a wide variety of operating systems to manage.
- Containers are a better choice when your biggest priority is maximizing the number of applications running on a minimal number of servers.

VMs	Containers
Heavyweight	Lightweight
Limited performance	Native performance
Each VM runs in its own OS	All containers share the host OS
Hardware-level virtualization	OS virtualization
Startup time in minutes	Startup time in milliseconds
Allocates required memory	Requires less memory space
Fully isolated and hence more secure	Process-level isolation, possibly less secure

¿Qué es un contenedor?

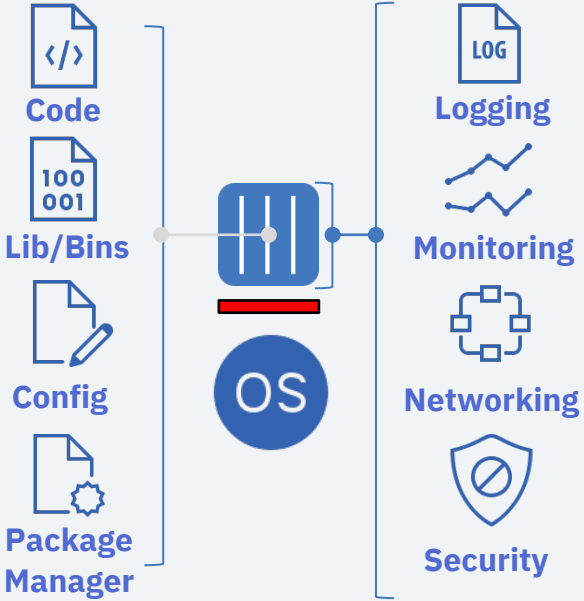
A standard way to **package** an application and all its dependencies so that it can be moved between environments and **run** without changes.

Containers work by **isolating** the differences between applications **inside** the container so that everything **outside** the container can be standardized.

Deb
The Developer

- Package apps with all dependencies
- Deploy to any environment in seconds
- Easily accessed and shared

Worries about what's « **inside** » the container



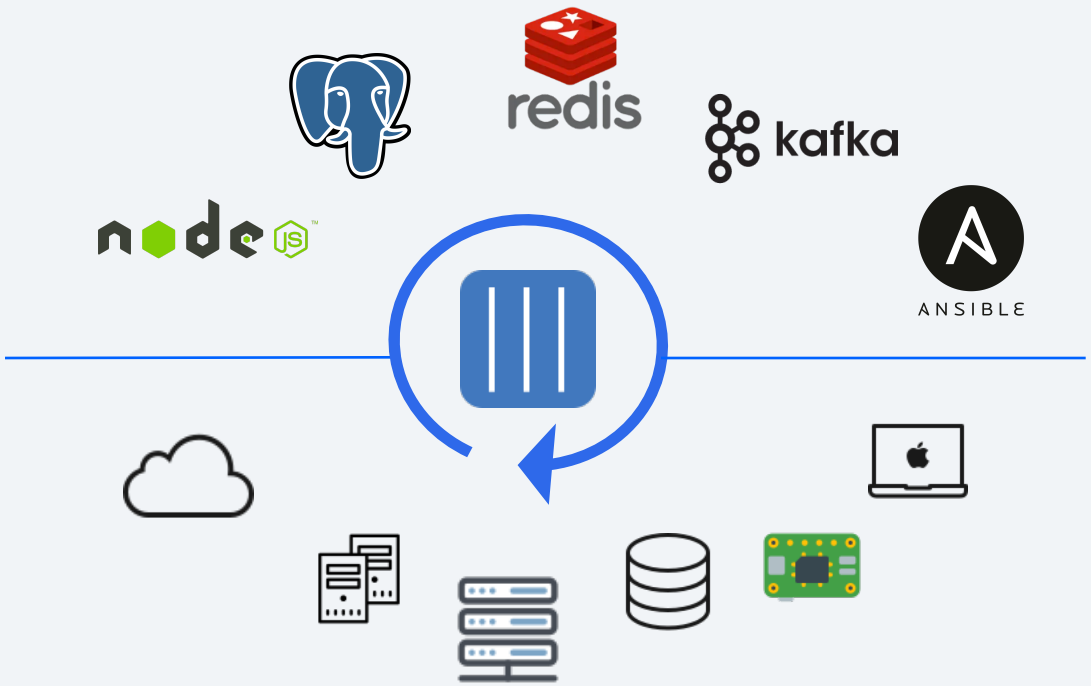
Mike
The Ops Guy

- Application processes on a shared kernel
- Simpler, lighter, and denser than VMs
- Portable across different environments

Worries about what's « **outside** » the container

Ventajas de los contenedores

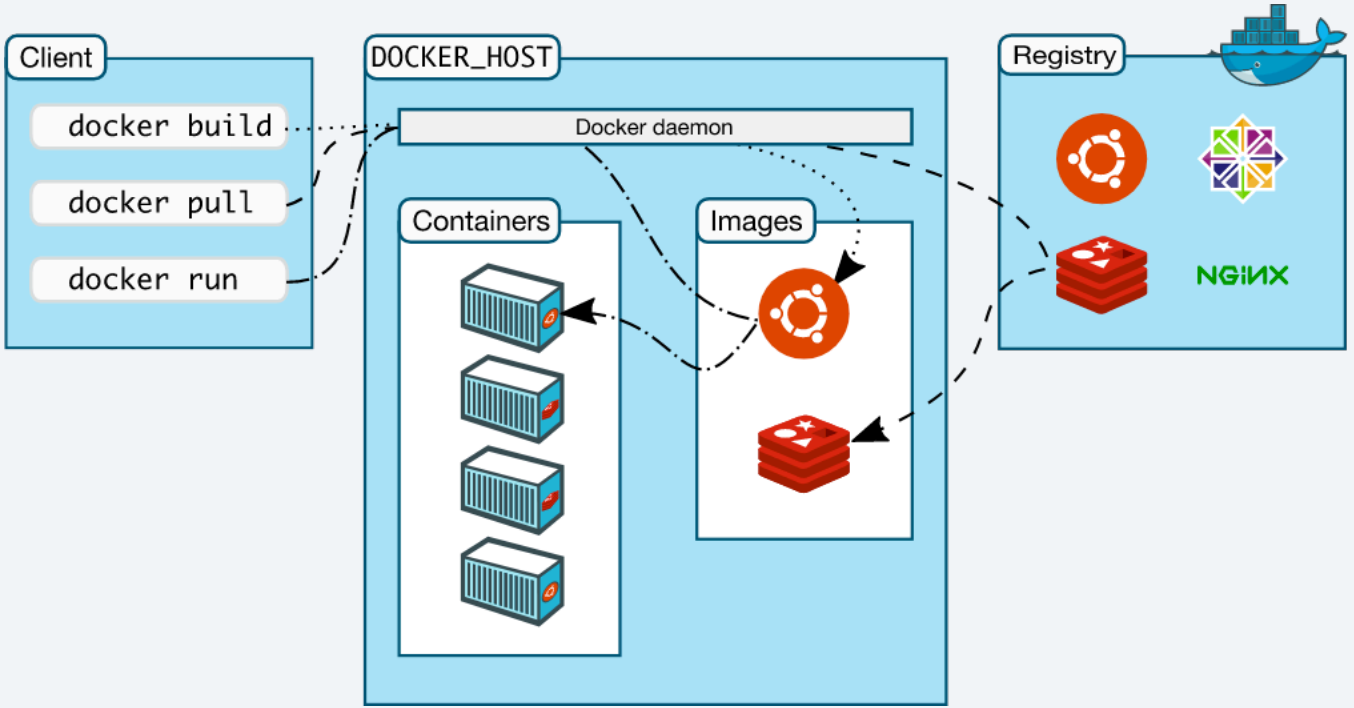
Multiplicidad del stack



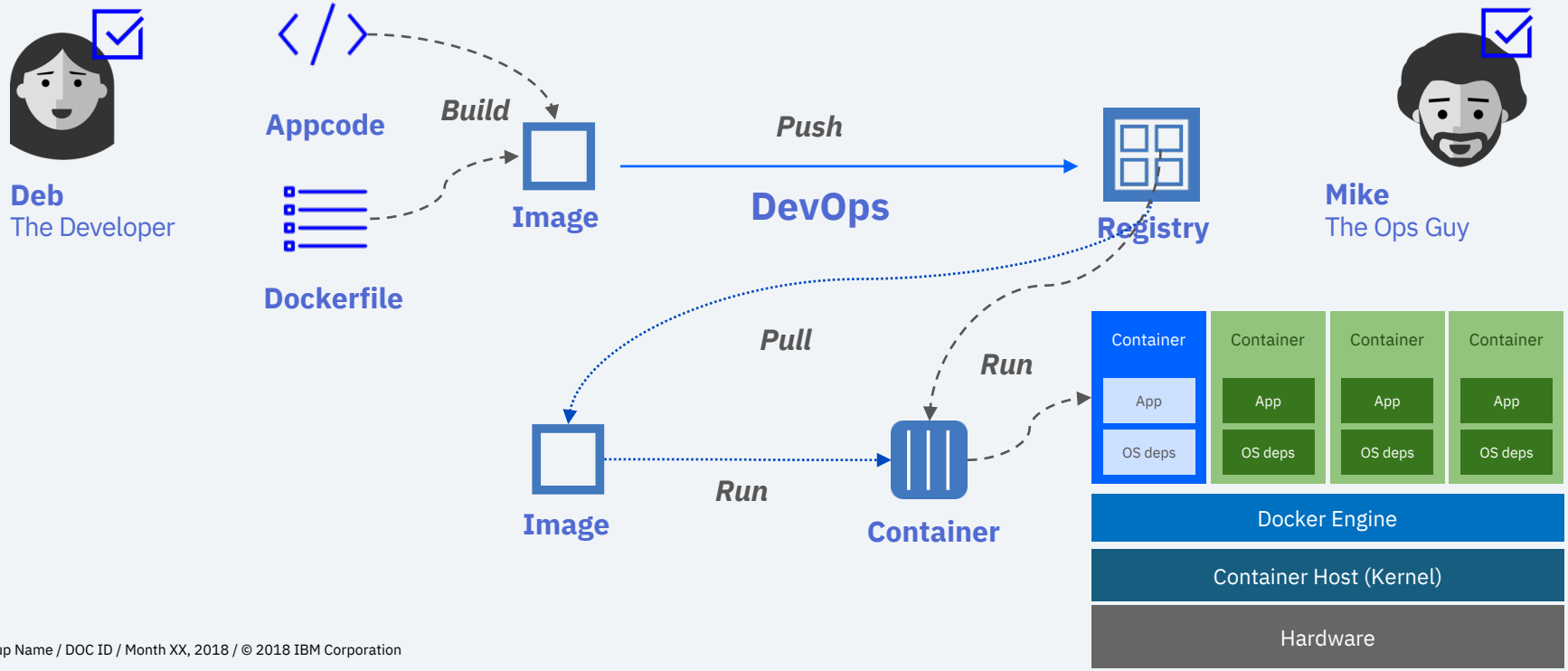
Multiplicidad de plataformas

- Consistencia
- Agilidad
- Portabilidad
- Eficiencia
- DevOps
- Microservicios
- Ecosistema

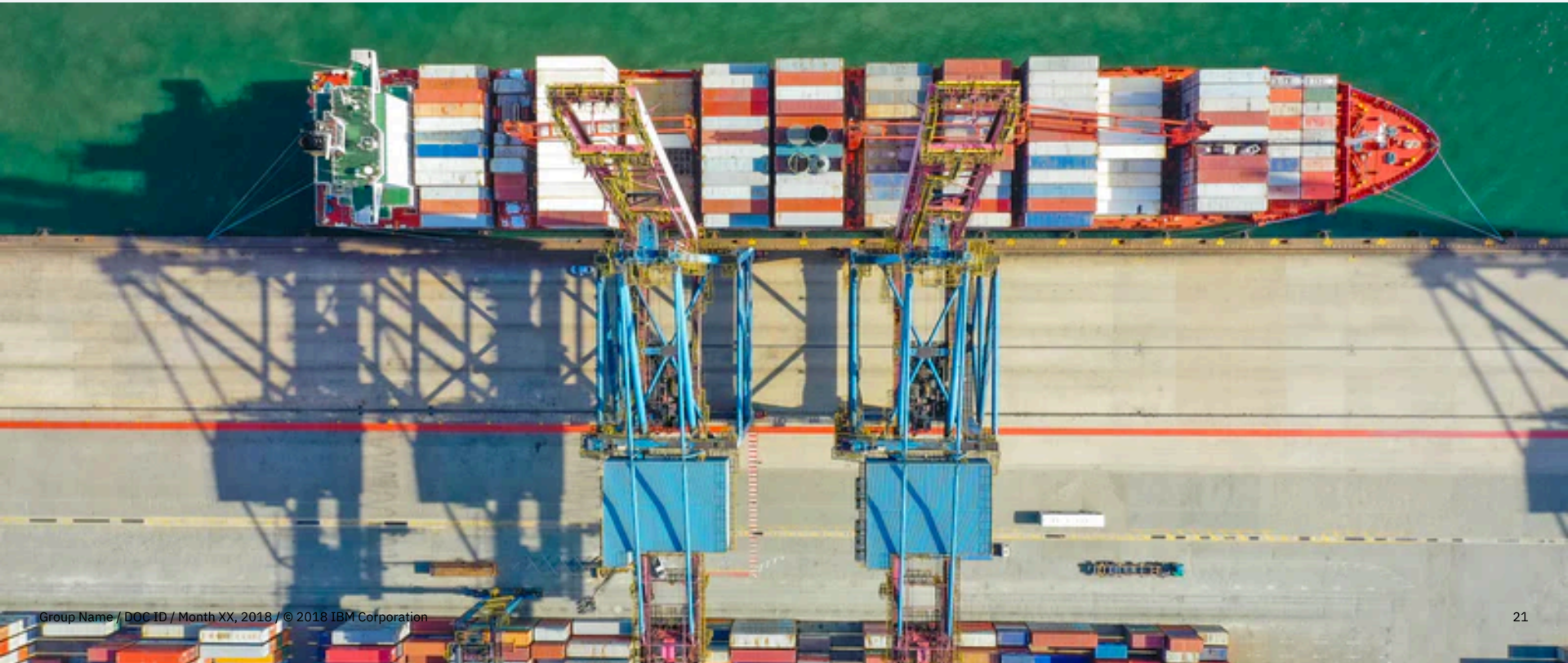
Docker architecture



Proceso básico de despliegue

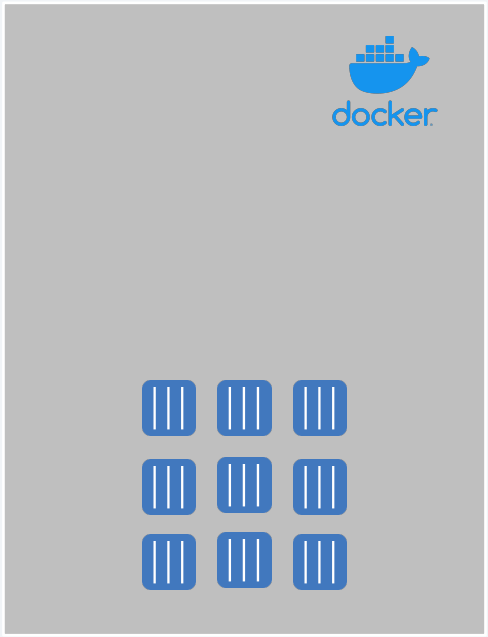


Orquestación



Unos pocos contenedores son manejables pero…

> docker run myApp



Cuando la cosa pasa a mayores…

Intelligent Scheduling

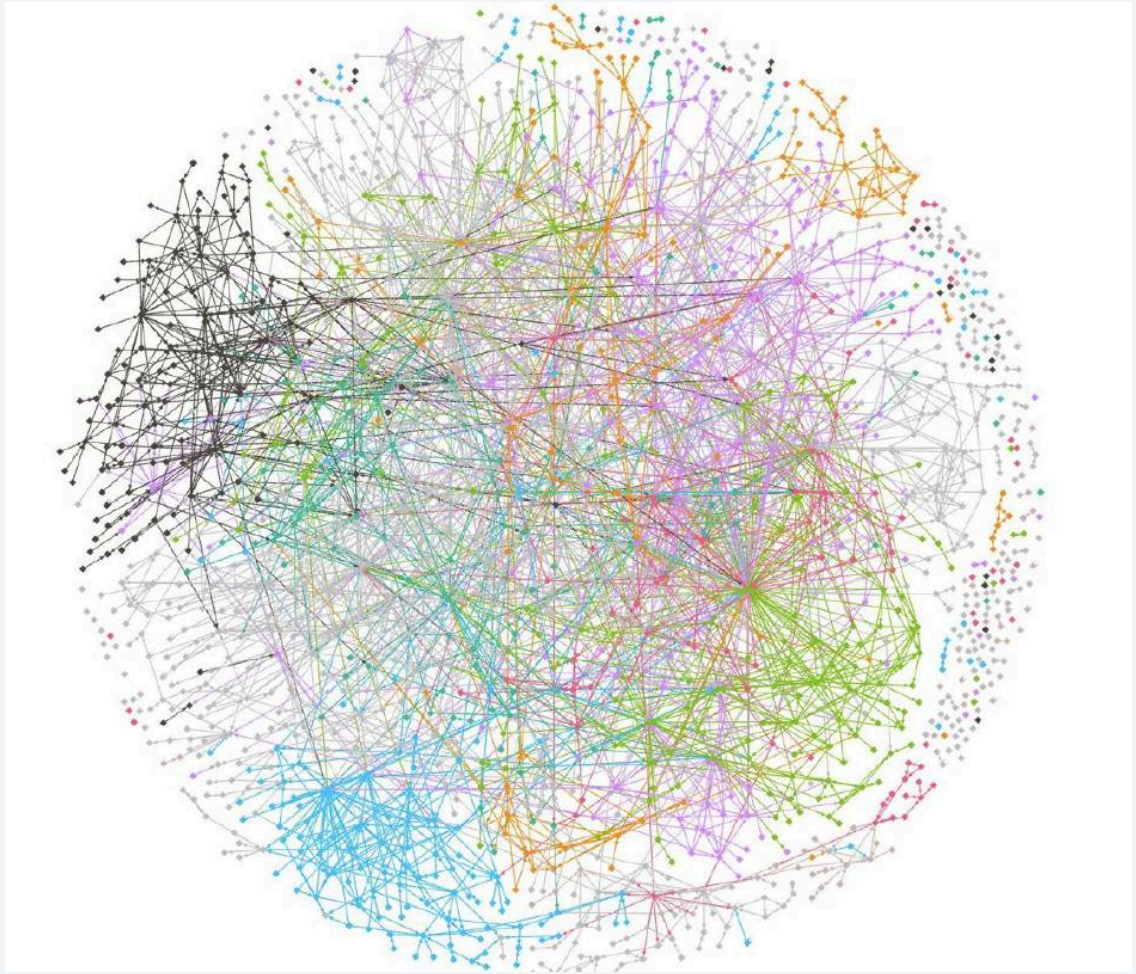
Self-healing

Horizontal scaling

Service discovery & load balancing

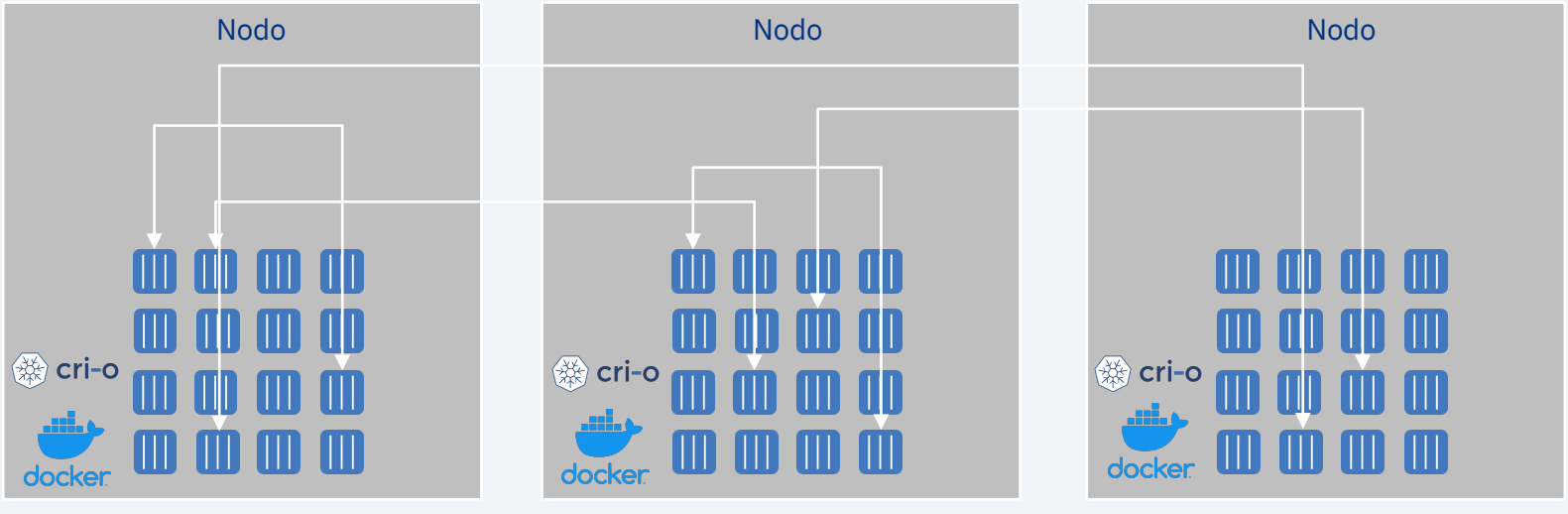
Automated rollouts and rollbacks

Secret and configuration management



https://www.theregister.com/2020/03/09/monzo_microservices/

Bienvenidos a Kubernetes



Despliegue inteligente

Auto reparación (*Self-healing*)

Escalado horizontal

Descubrimiento de servicios y balanceo de carga

Automatización despliegues y rollback

Gestión de claves (*Secret*) y configuración

Otros orquestadores

Apache
MESOS™

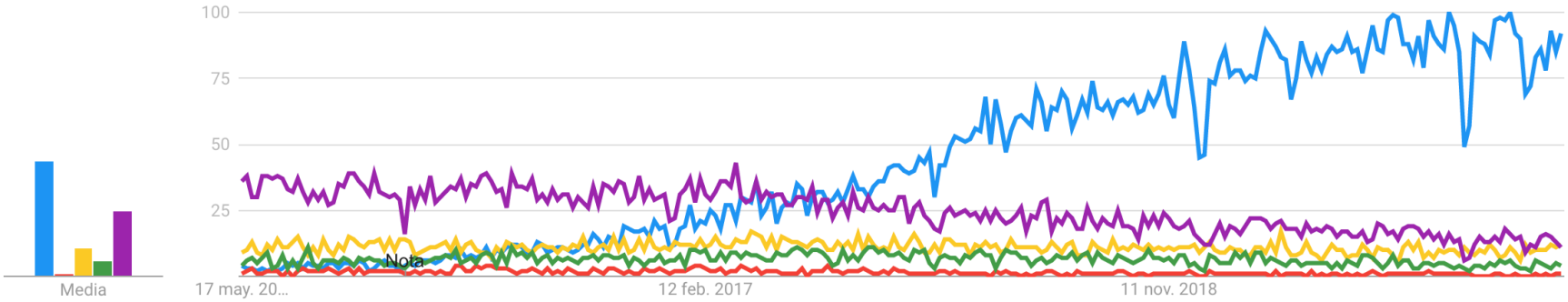
kubernetes
Término de búsqueda

Mesos
Término de búsqueda

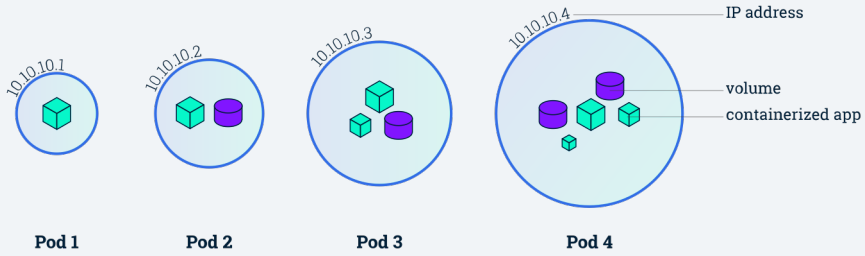
Swarm
Término de búsqueda

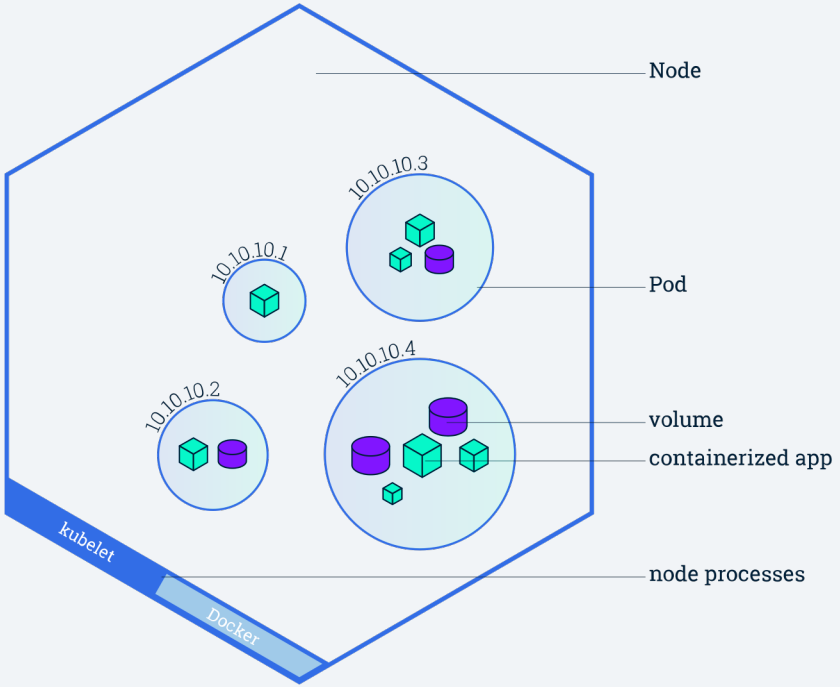
Cloud Foundry
Término de búsqueda

Openstack
Término de búsqueda



Una aplicación en Kubernetes

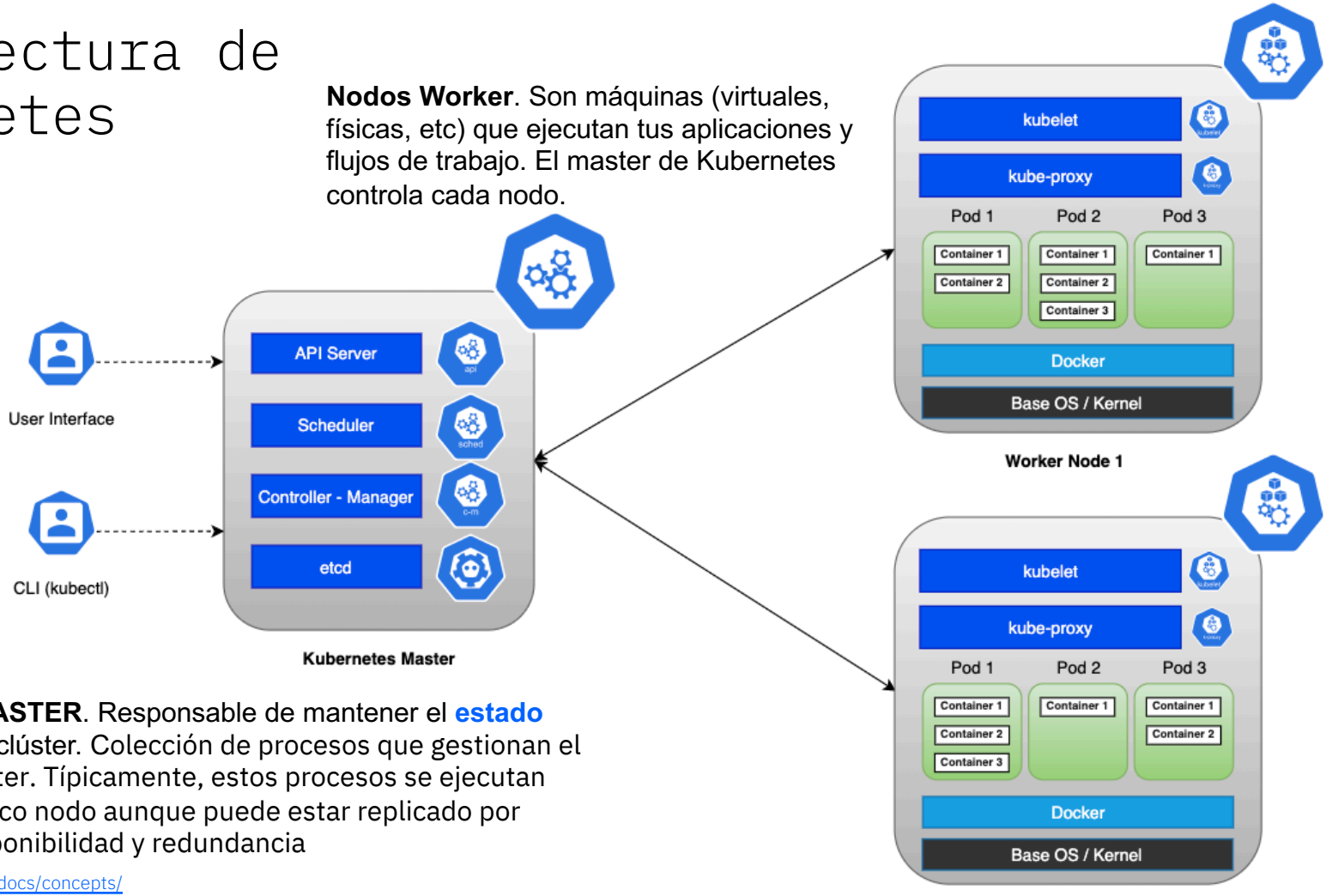




https://kubernetes.io/docs/tutorials/kubernetes-basics/explore/explore-intro/

Arquitectura de Kubernetes

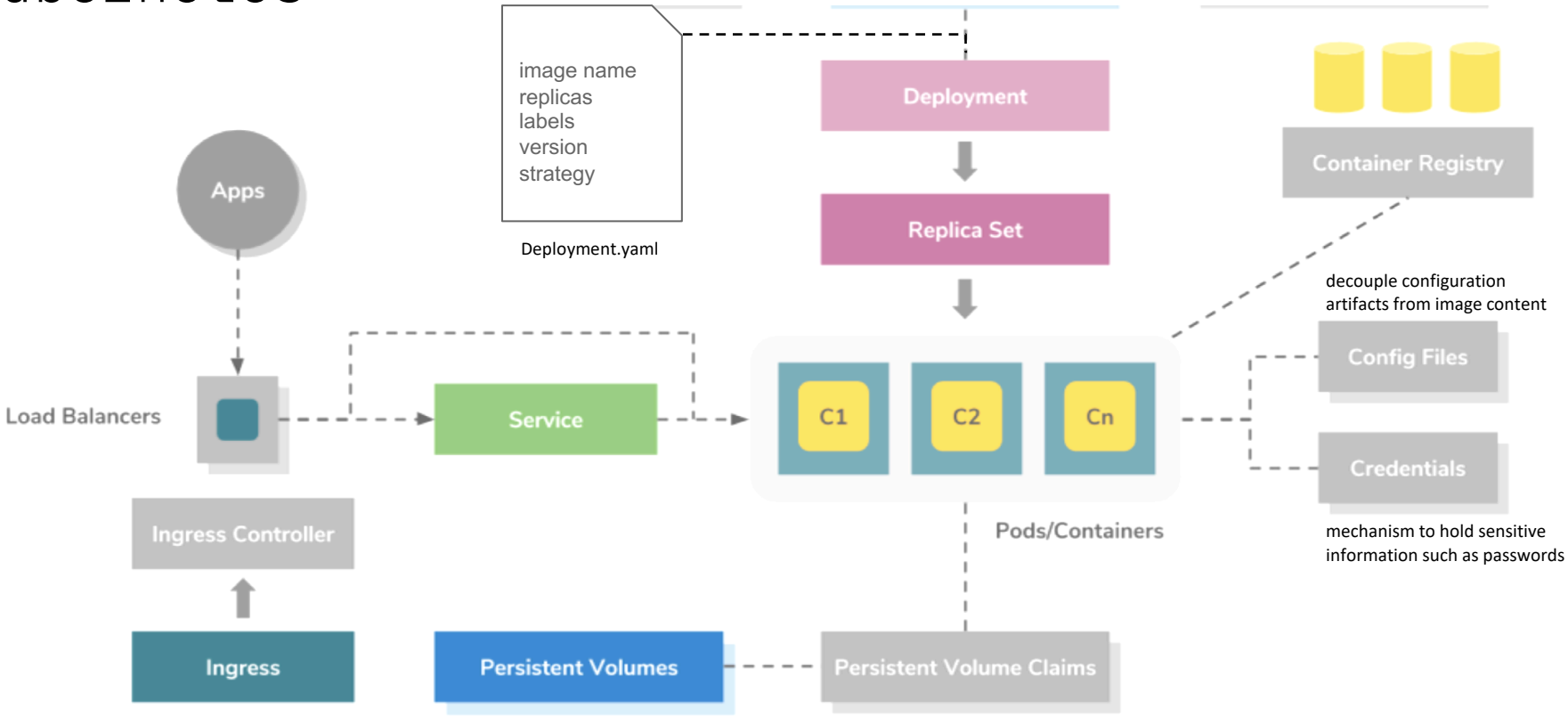
Nodos Worker. Son máquinas (virtuales, físicas, etc) que ejecutan tus aplicaciones y flujos de trabajo. El master de Kubernetes controla cada nodo.



Kubernetes MASTER. Responsable de mantener el **estado deseado** de tu clúster. Colección de procesos que gestionan el estado del clúster. Típicamente, estos procesos se ejecutan todos en un único nodo aunque puede estar replicado por motivos de disponibilidad y redundancia

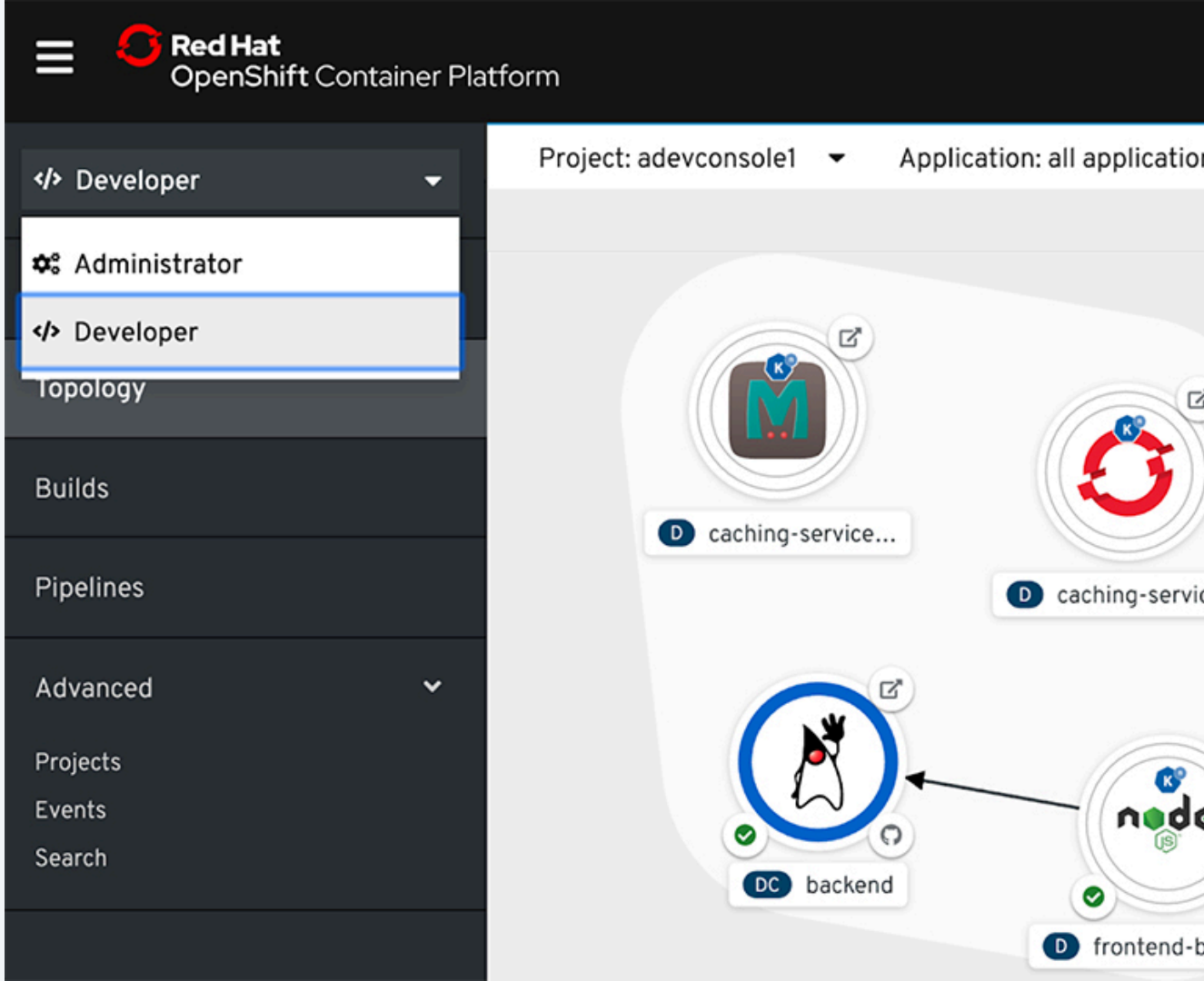
https://kubernetes.io/es/docs/concepts/

Componentes de Kubernetes



https://medium.com/containermind/a-beginners-guide-to-kubernetes-7e8ca56420b6

Red Hat OpenShift

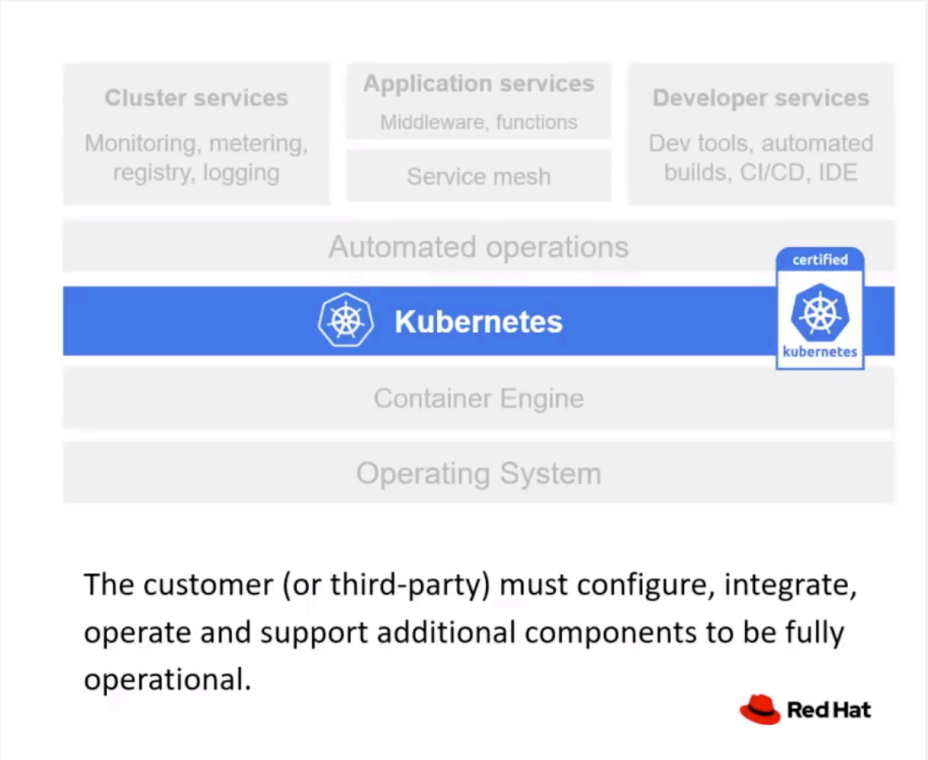


OPENSHIFT

Kubernetes vs Openshift

¿Qué necesita Kubernetes para ponerlo en producción?

- Sistema Operativo
- Runtime de Contenedores
- Registro de imágenes
- Networking avanzado
- Gestión de Logs
- Metricas y monitorización
- …

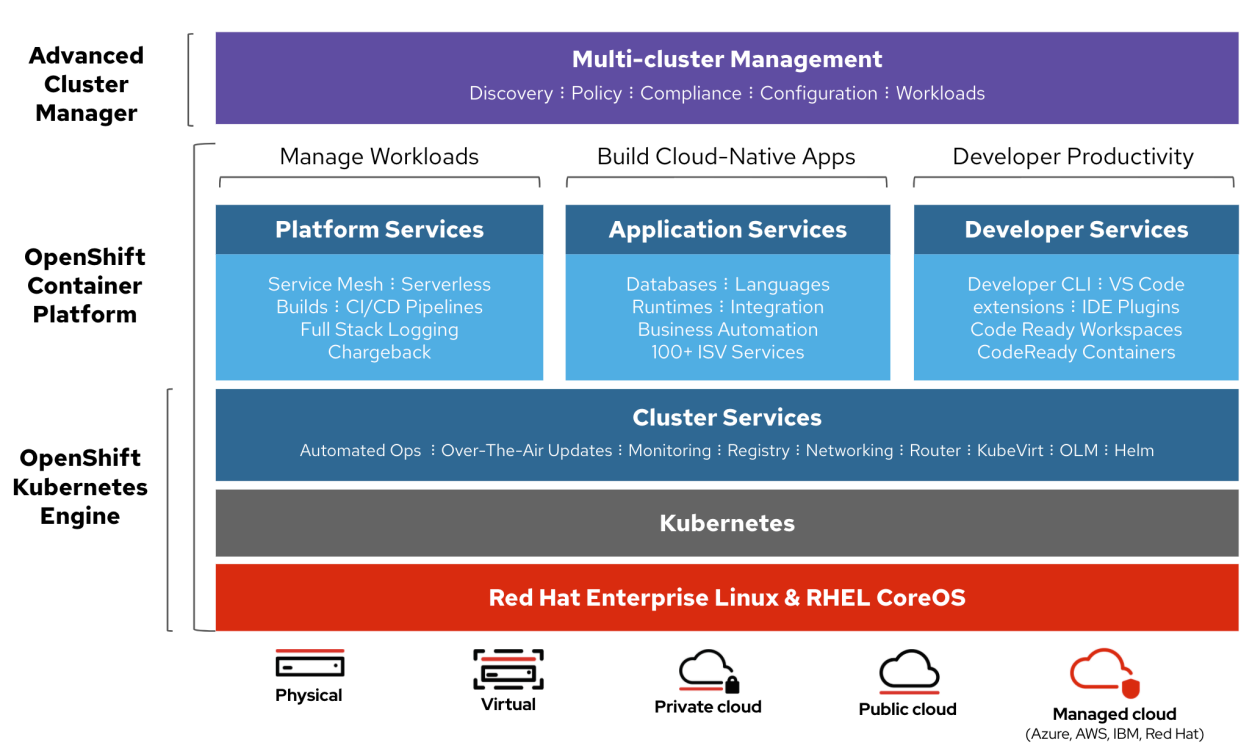


¿Qué es Red Hat OpenShift?

Red Hat OpenShift es Kubernetes para empresas, en modalidad Hybrid Cloud

Valor de OpenShift

- Plataforma certificada
- Soporte & SLA
- Seguridad
- Instalación Full Stack
- Automatización
- Ciclo de vida del producto
- Colaboración
- Catálogo
- Hybrid, Multi-Cluster Management
- Operator Framework
- OpenShift Service Mesh (Istio)
- OpenShift Serverless (Knative)
- CodeReady Workspaces (Che)

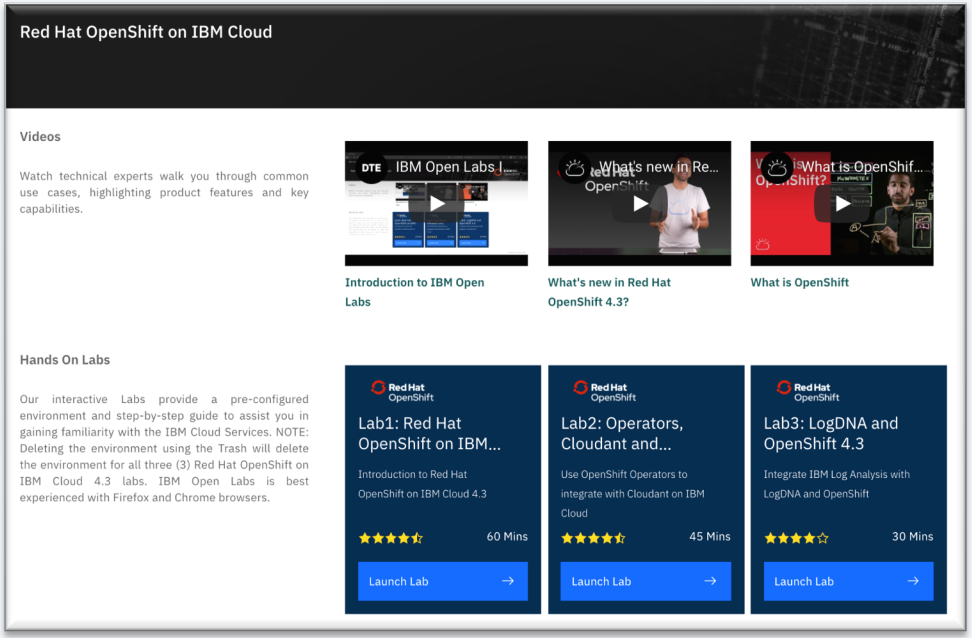


Registro en IBM Cloud y OpenShift Openlabs

Manos a la obra!

1. Regístrate en IBM Cloud https://ibm.biz/devopenshift

2. Accede a los laboratorios (cluster de OpenShift gratuito –sandbox–)

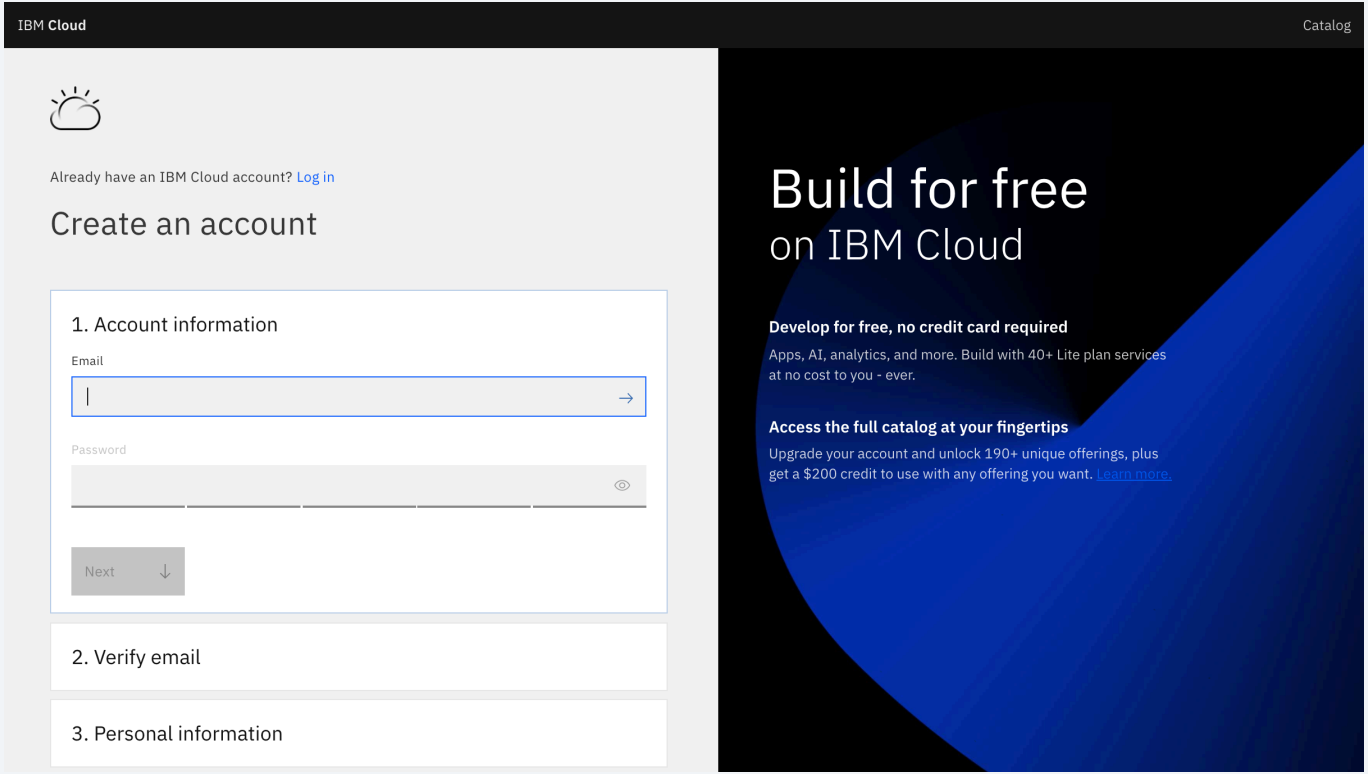


https://developer.ibm.com/openlabs/openshift

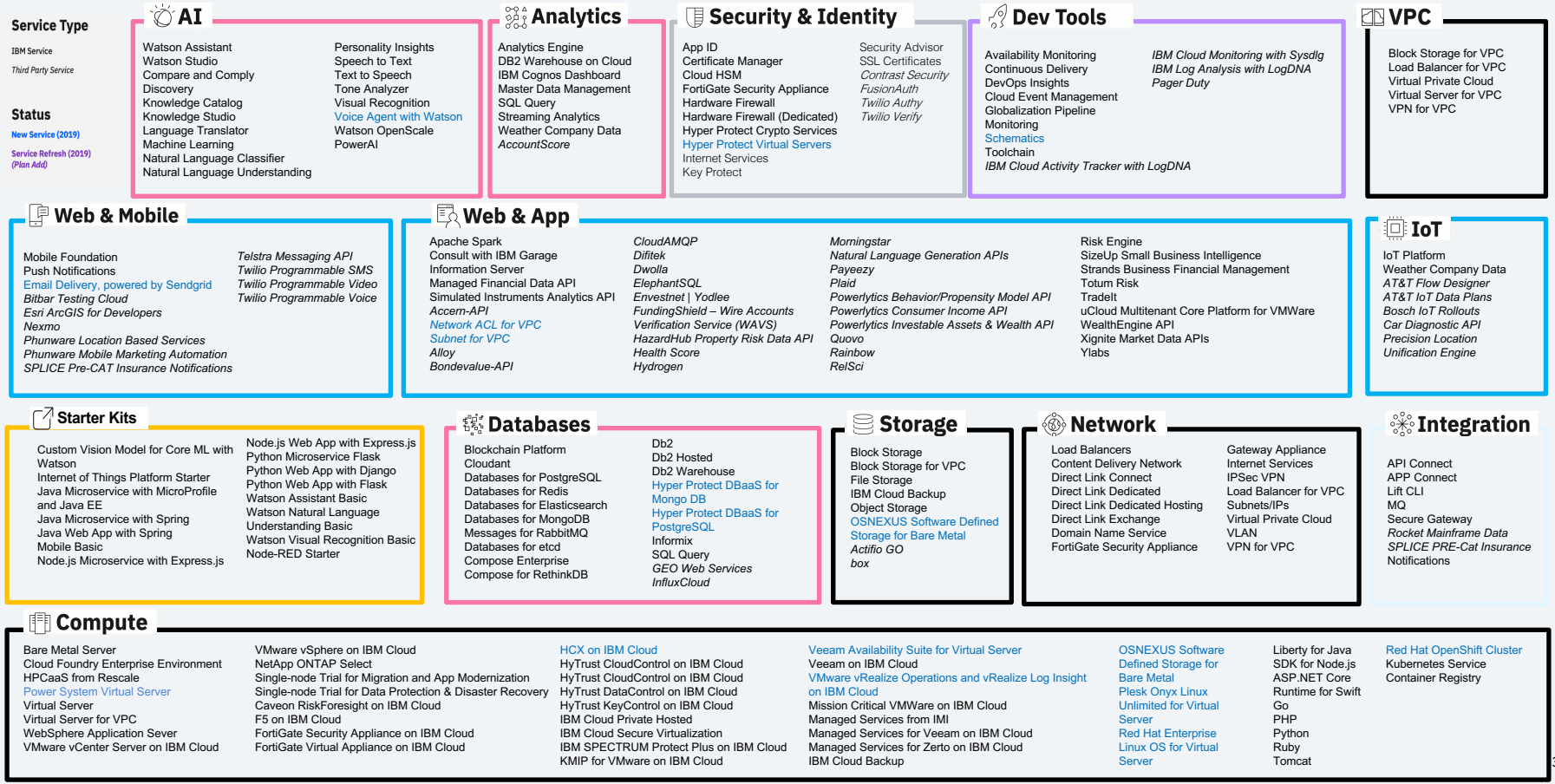
OPENSHIFT

Crea tu cuenta IBM Cloud…

https://ibm.biz/devopenshift



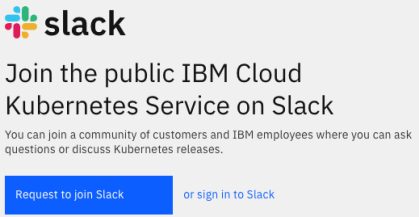
…y accede a un amplio catálogo de servicios

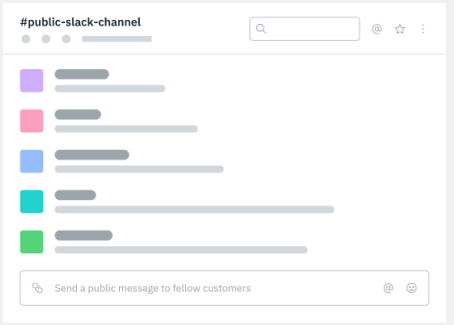


https://ibm.biz/devopenshift

IBM **Cloud**
Kubernetes Service

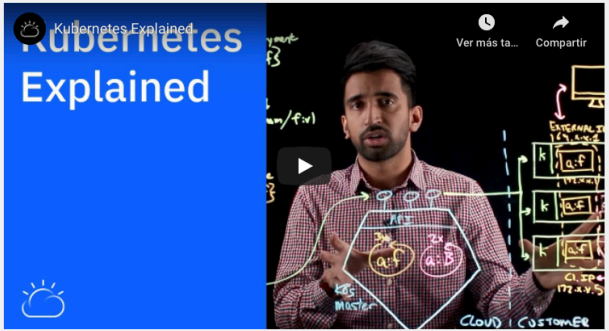
Join the public Slack channel

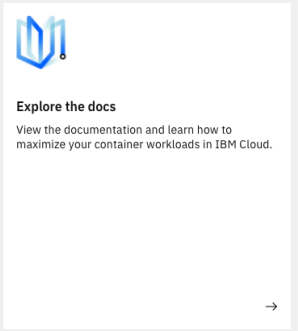




OPENSHIFT

View the documentation





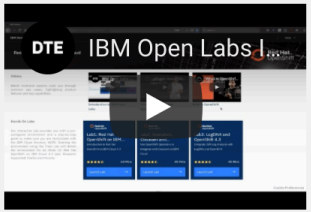
https://cloud.ibm.com/kubernetes/overview

https://developer.ibm.com/openlabs/openshift

Red Hat OpenShift on IBM Cloud

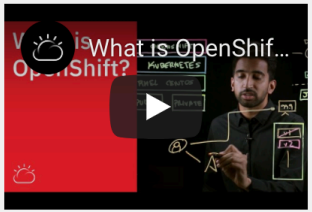
Videos

Watch technical experts walk you through common use cases, highlighting product features and key capabilities.



Introduction to IBM Open Labs

What's new in Red Hat OpenShift 4.3?



What is OpenShift

Hands On Labs

Our interactive Labs provide a pre-configured environment and step-by-step guide to assist you in gaining familiarity with the IBM Cloud Services. NOTE: Deleting the environment using the Trash will delete the environment for all three (3) Red Hat OpenShift on IBM Cloud 4.3 labs. IBM Open Labs is best experienced with Firefox and Chrome browsers.

Red Hat OpenShift

Lab1: Red Hat OpenShift on IBM...

Introduction to Red Hat OpenShift on IBM Cloud 4.3

★★★★½ 60 Mins

Launch Lab →

Red Hat OpenShift

Lab2: Operators, Cloudant and...

Use OpenShift Operators to integrate with Cloudant on IBM Cloud

★★★★½ 45 Mins

Launch Lab →

Red Hat OpenShift

Lab3: LogDNA and OpenShift 4.3

Integrate IBM Log Analysis with LogDNA and OpenShift

★★★★½ 30 Mins

Launch Lab →

Kubernetes with OpenShift

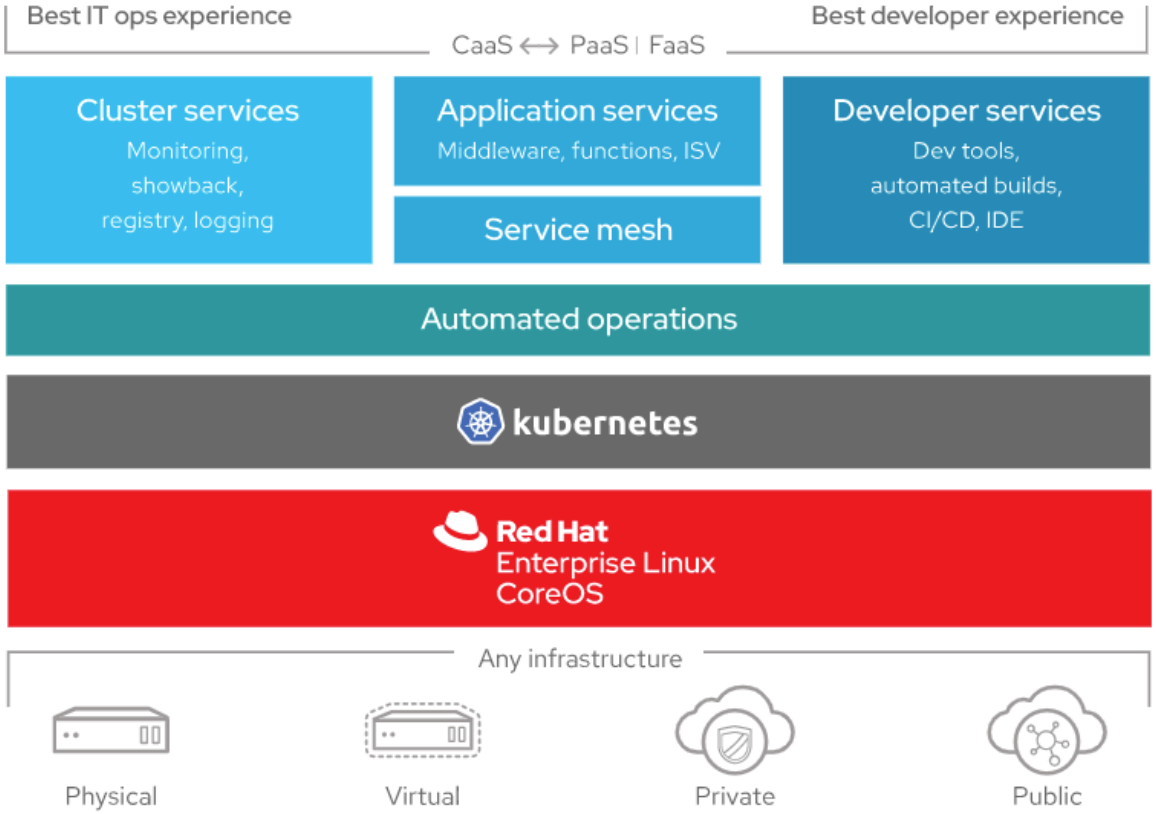
World Tour

En esta sesión aprenderemos los elementos principales en los que se basa la tecnología de **OpenShift** sobre **IBM Cloud**, y realizaremos un laboratorio para explicar escenarios típicos, por ejemplo, cómo desplegar aplicaciones, implementar un entorno de CI/CD, usar la linea de comandos (CLI), etc.

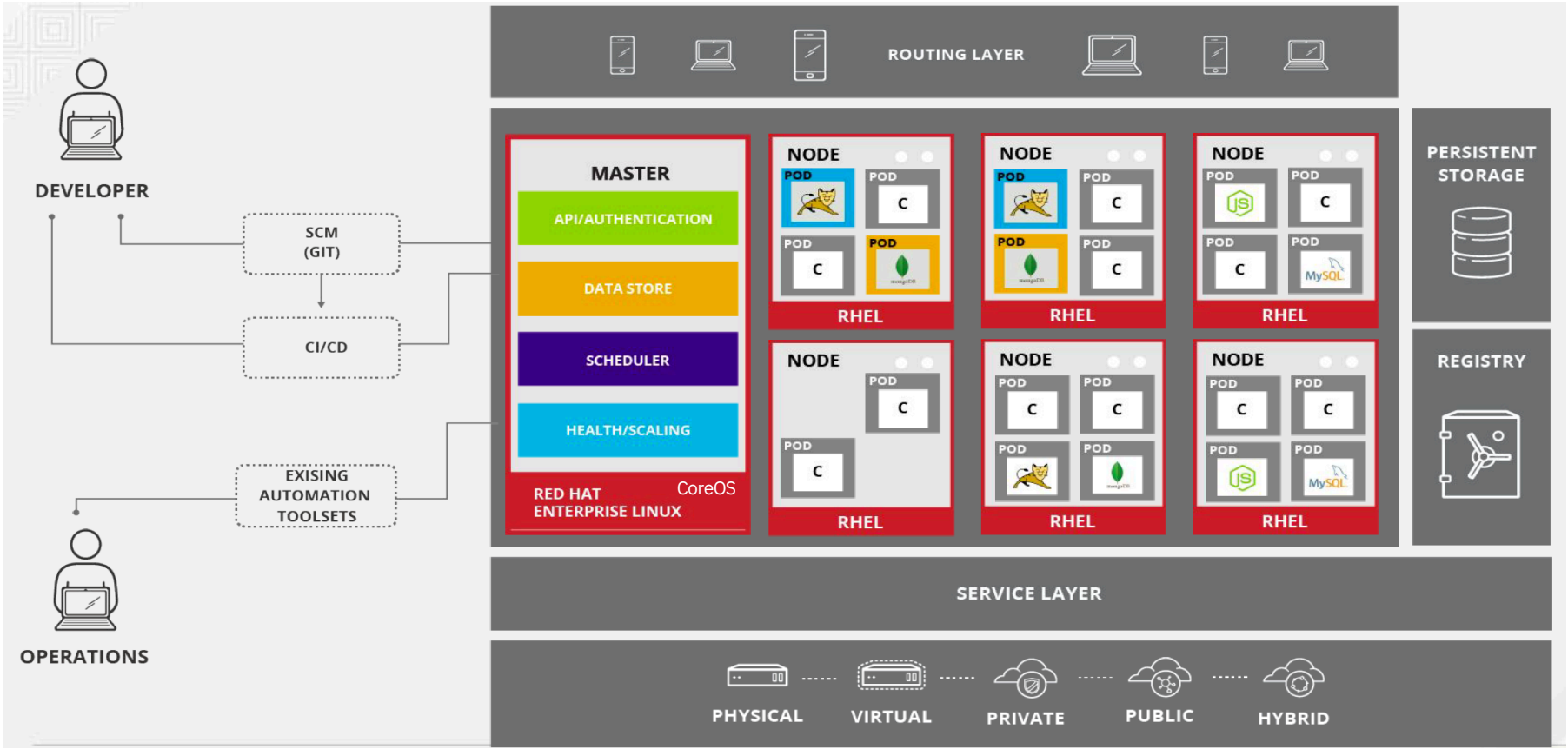
Taller para principiantes:
Introducción a Red Hat OpenShift en IBM Cloud 4.3

¿Qué es Red Hat OpenShift?

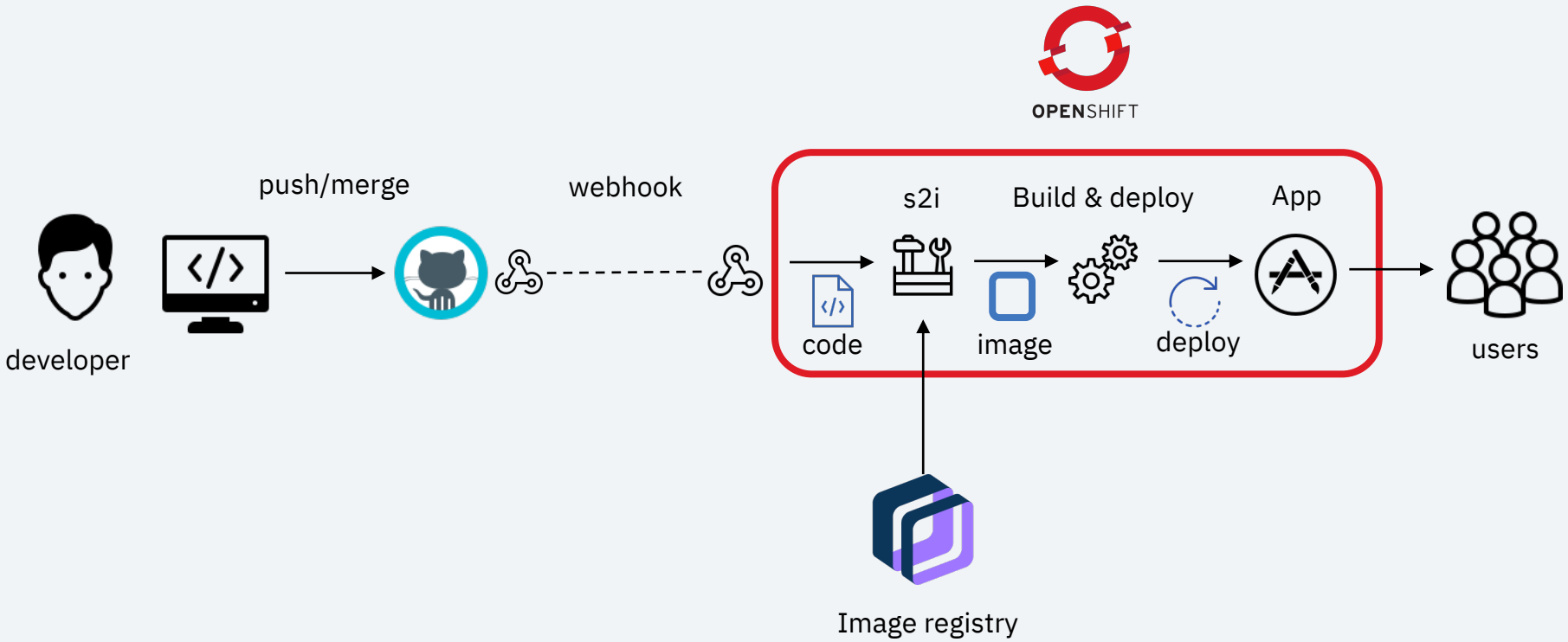
Red Hat OpenShift es Kubernetes para empresas, en modalidad Hybrid Cloud



OpenShift Architecture



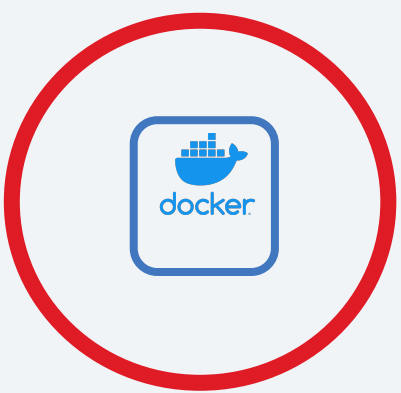
Ejemplo de CI/CD en OpenShift (s2i)



Modalidades de build & deploy en OpenShift

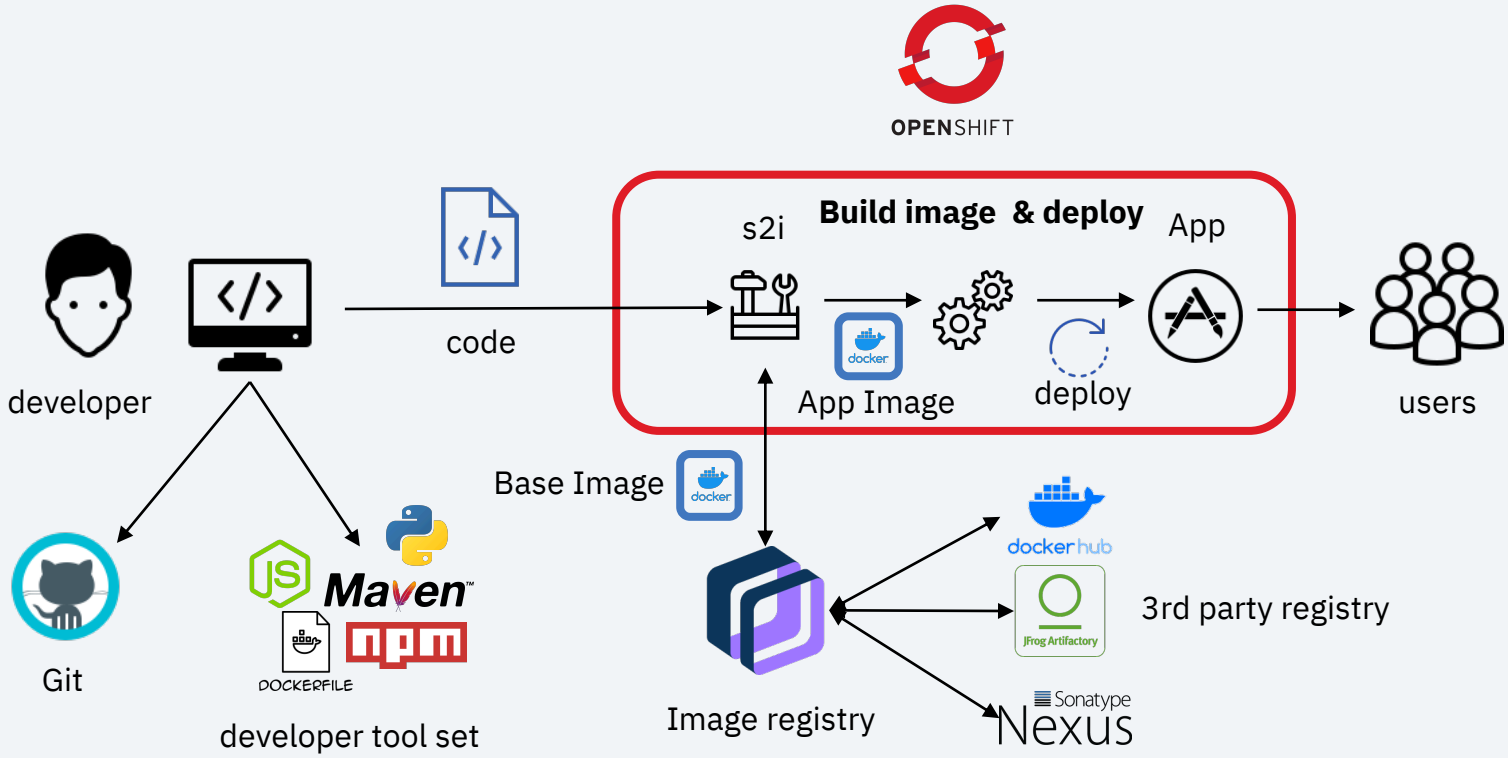
DEPLOY YOUR SOURCE CODE WITH SOURCE-TO-IMAGE (S2I)

DEPLOY YOUR APP BINARY WITH SOURCE-TO-IMAGE (S2I)

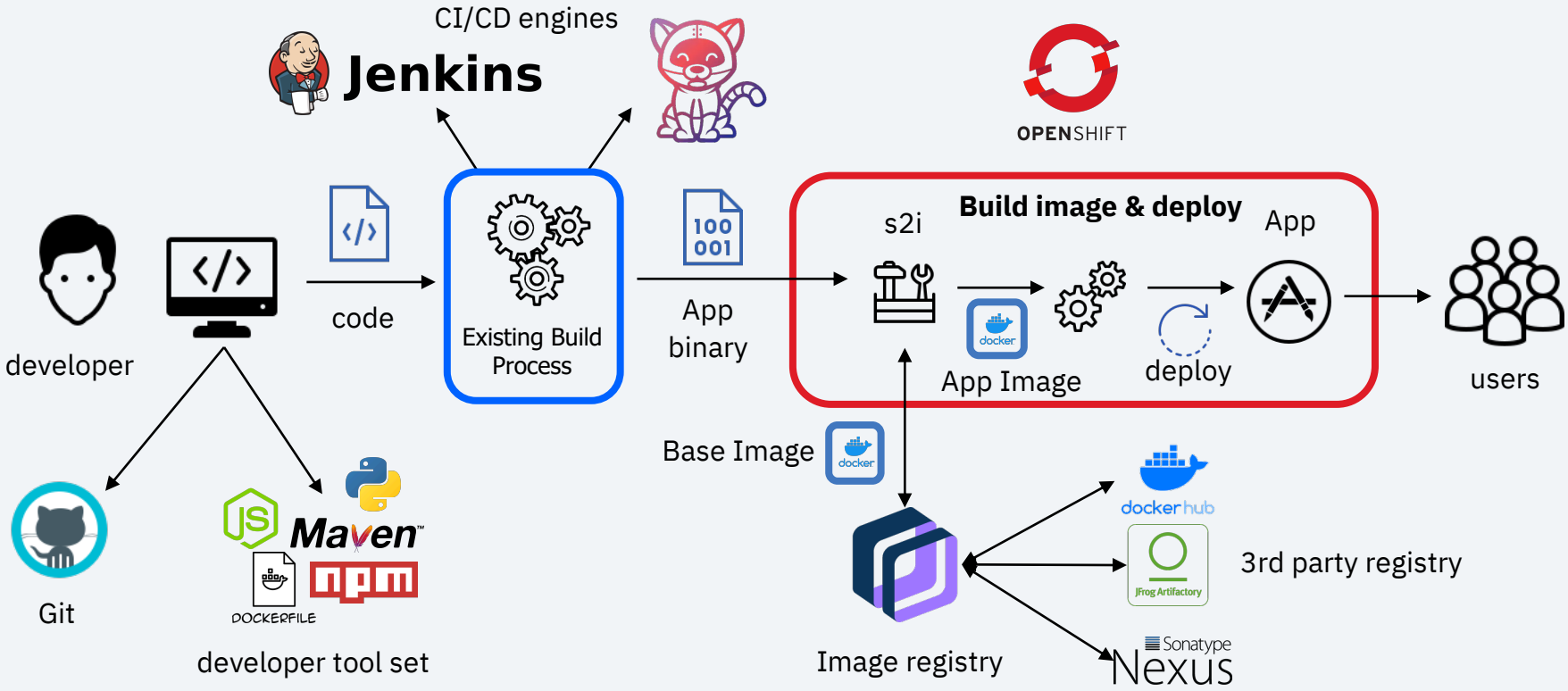


DEPLOY YOUR DOCKER IMAGE

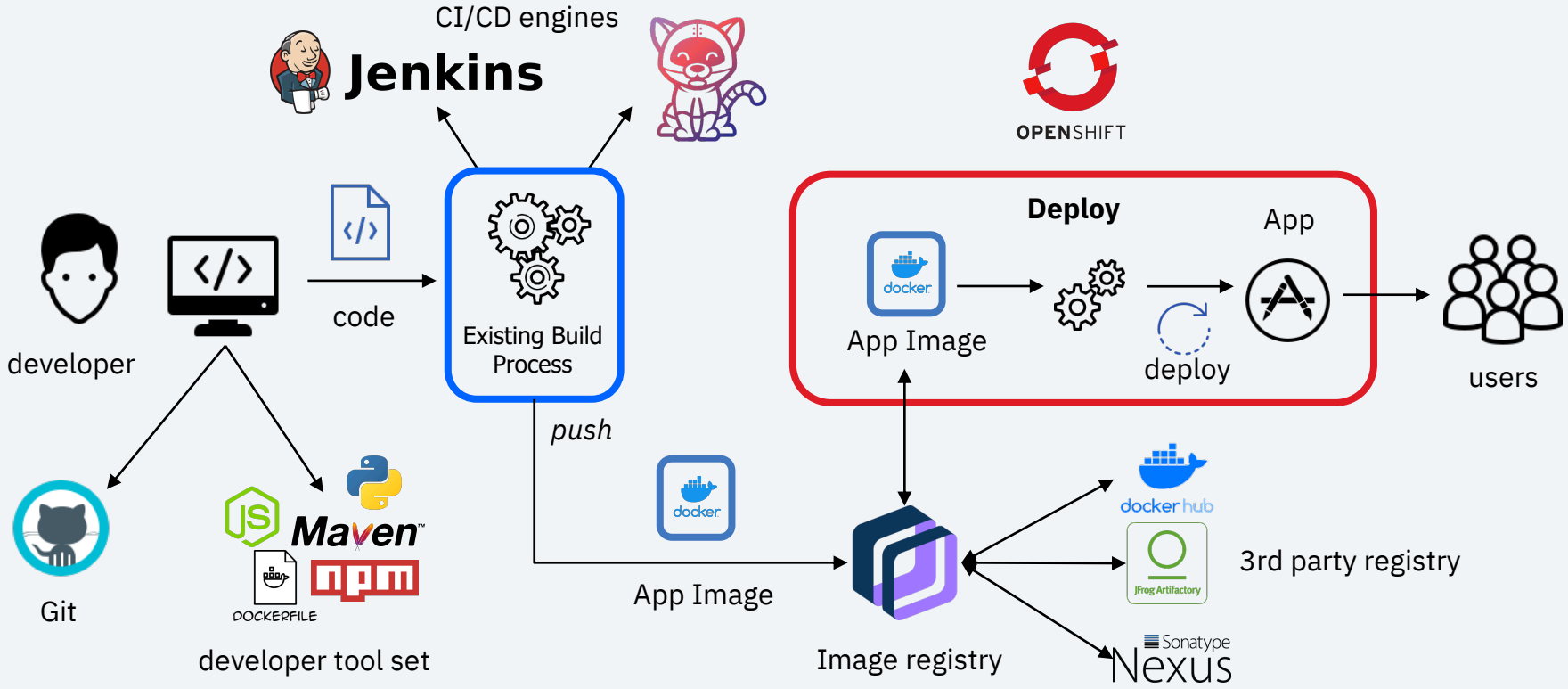
Deploy source code with S2I



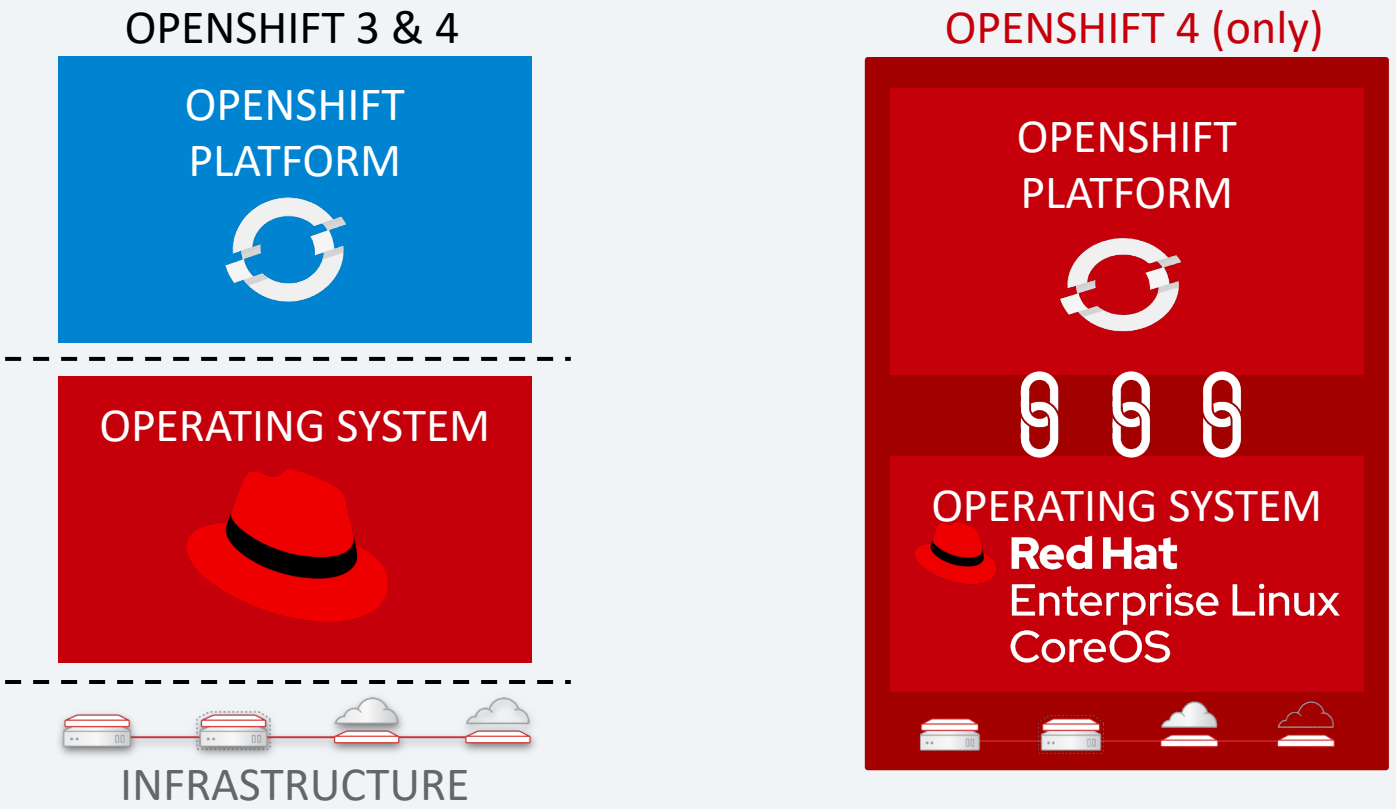
Deploy app binary with S2I



Deploy Docker Image



OpenShift Full-stack automated install



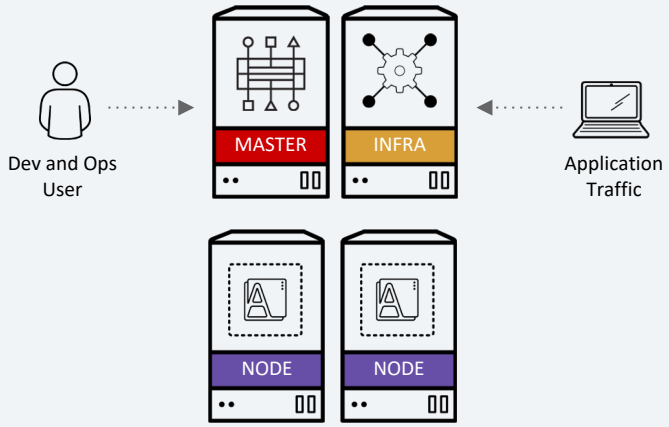
cri-o

A lightweight, OCI-compliant container runtime

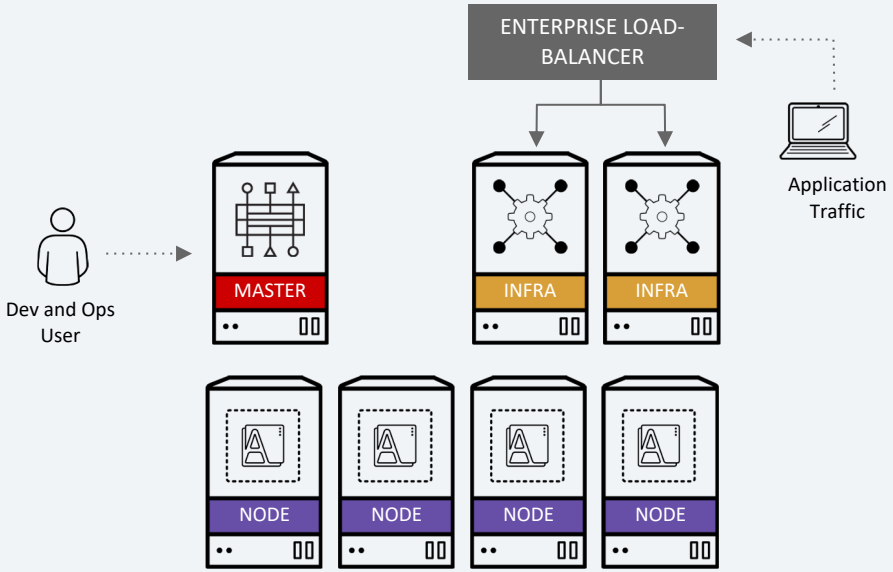
Minimal and Secure Architecture	Optimized for Kubernetes	Runs any OCI-compliant image (including docker)

PROOF-OF-CONCEPT ARCHITECTURE

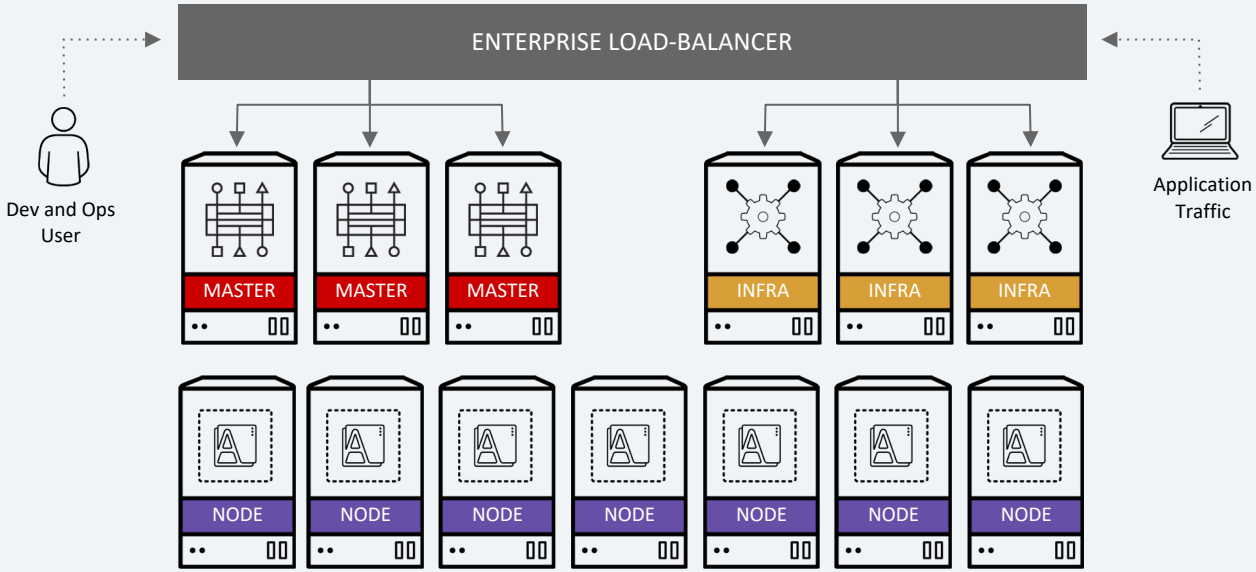
An infrastructure node is a node that is dedicated to infrastructure pods such as router, image registry, metrics, and logs



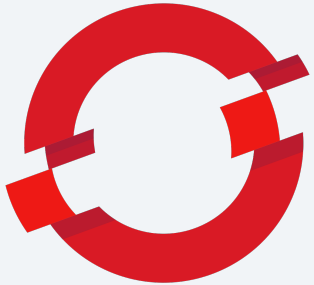
APP HIGH-AVAILABILITY ARCHITECTURE



FULL HIGH-AVAILABILITY ARCHITECTURE



Resumen del lab



CI/CD

CLI

Logging

Monitoring

AutoScale

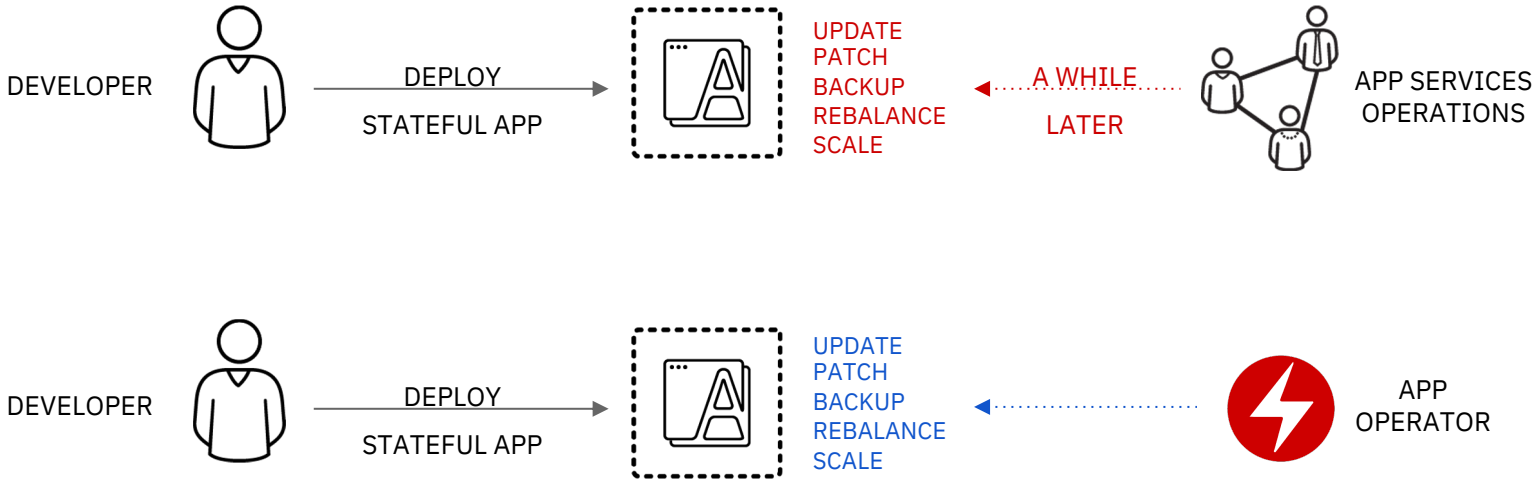
Kubernetes with OpenShift

World Tour

En esta práctica de laboratorio, veremos aspectos adicionales de administración de aplicaciones en **OpenShift** como por ejemplo, usar el concepto de Operador para integrar nuestra aplicación en OpenShift con una base de datos **Cloudant** desplegada y configurada como un servicio de **IBM Cloud** a partir de un Custom Resource Definition (CRD)

Taller avanzado:
Operadores en OpenShift 4.3. Desplegando un servicio Cloudant

Why Operators?



Operators codify operational knowledge and workflows to automate life-cycle management of containerized applications with Kubernetes

What is an Operator ?

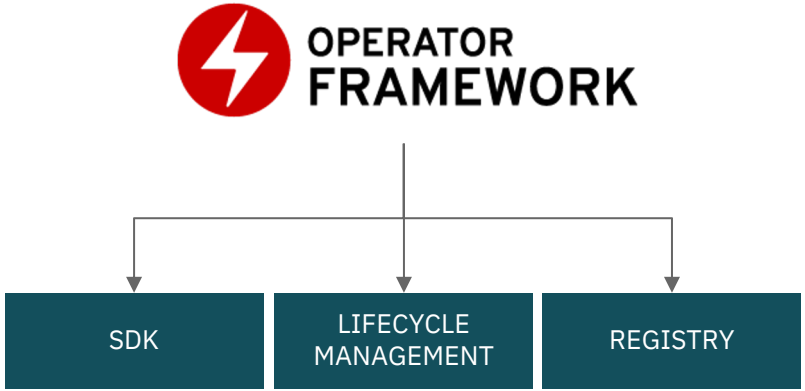
'Operators are software extensions to Kubernetes that make use of custom resources to manage applications and their components. Operators follow Kubernetes principles, notably the control loop'.

—*Kubernetes Documentation*

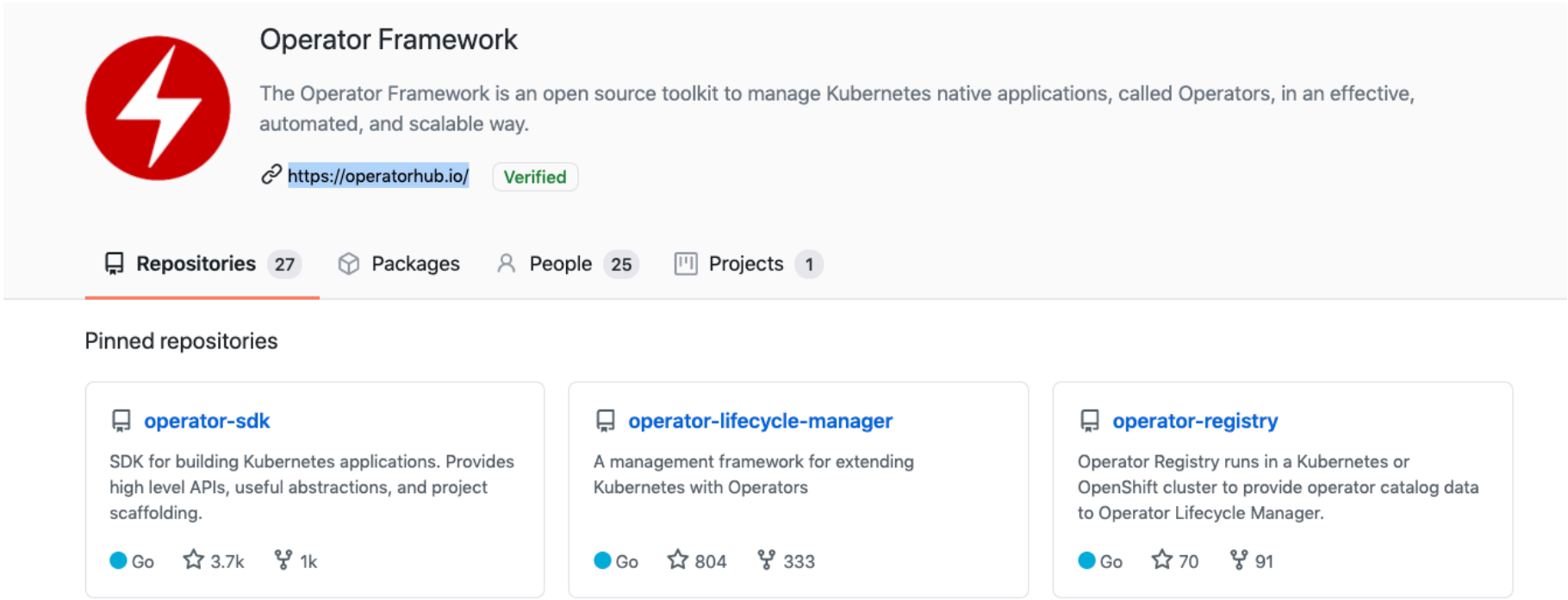
An Operator is an automation application that deals with the Kubernetes API and Custom Resources to create/operate new Resources.

It's an intelligent piece of software that embed the templating to deploy your resources.

The Operator watch events on the Kubernetes API and react (ex : re-create a pod, change Labels, update a Secret, Remove a Service, backup DB ...)

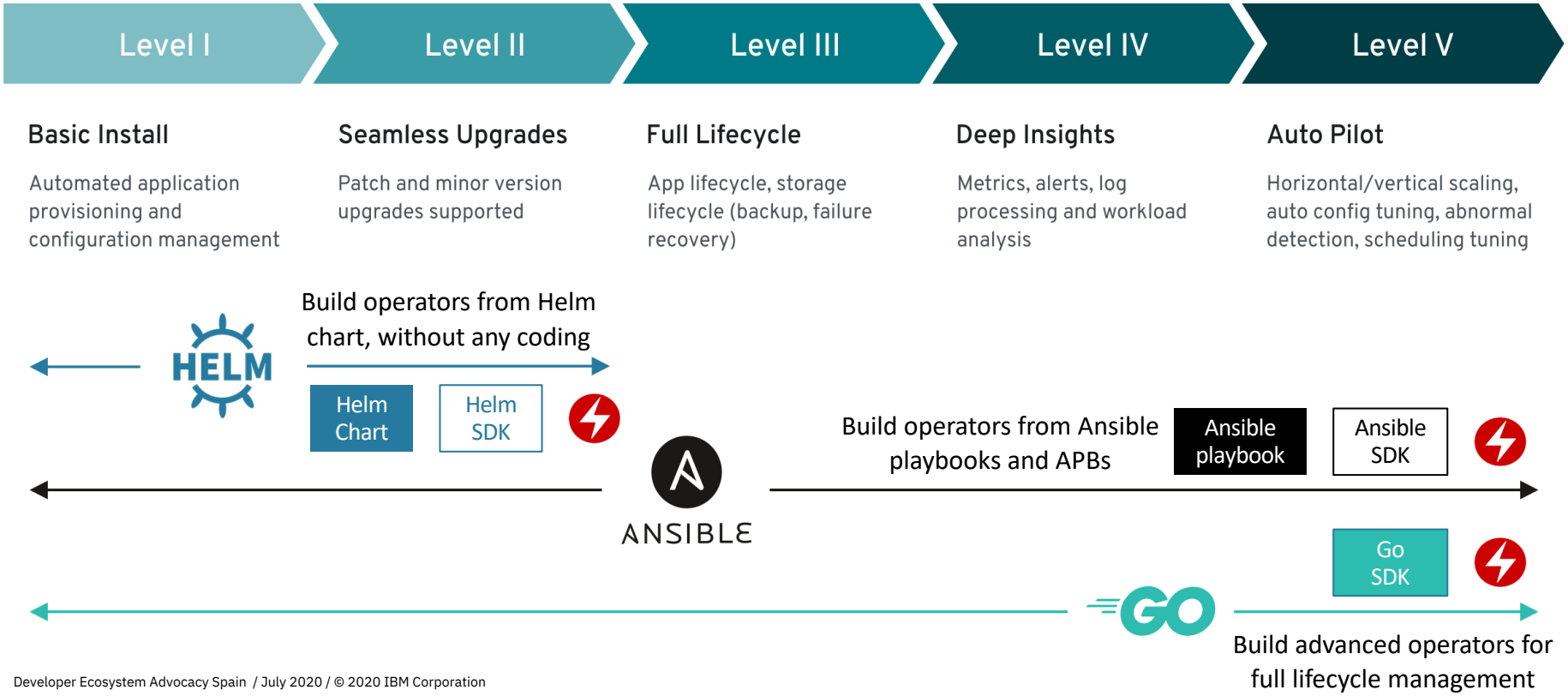


Operator Framework



https://github.com/operator-framework

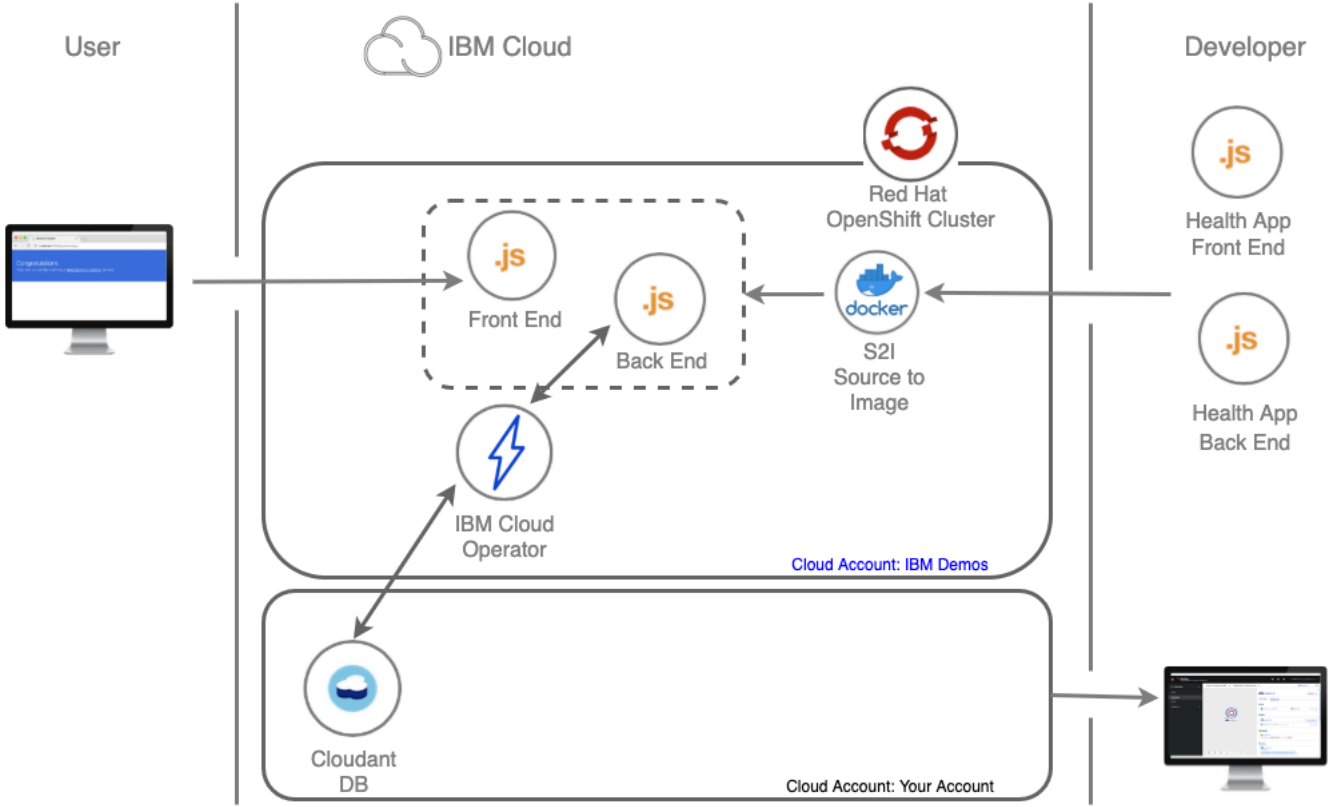
Type of Operators



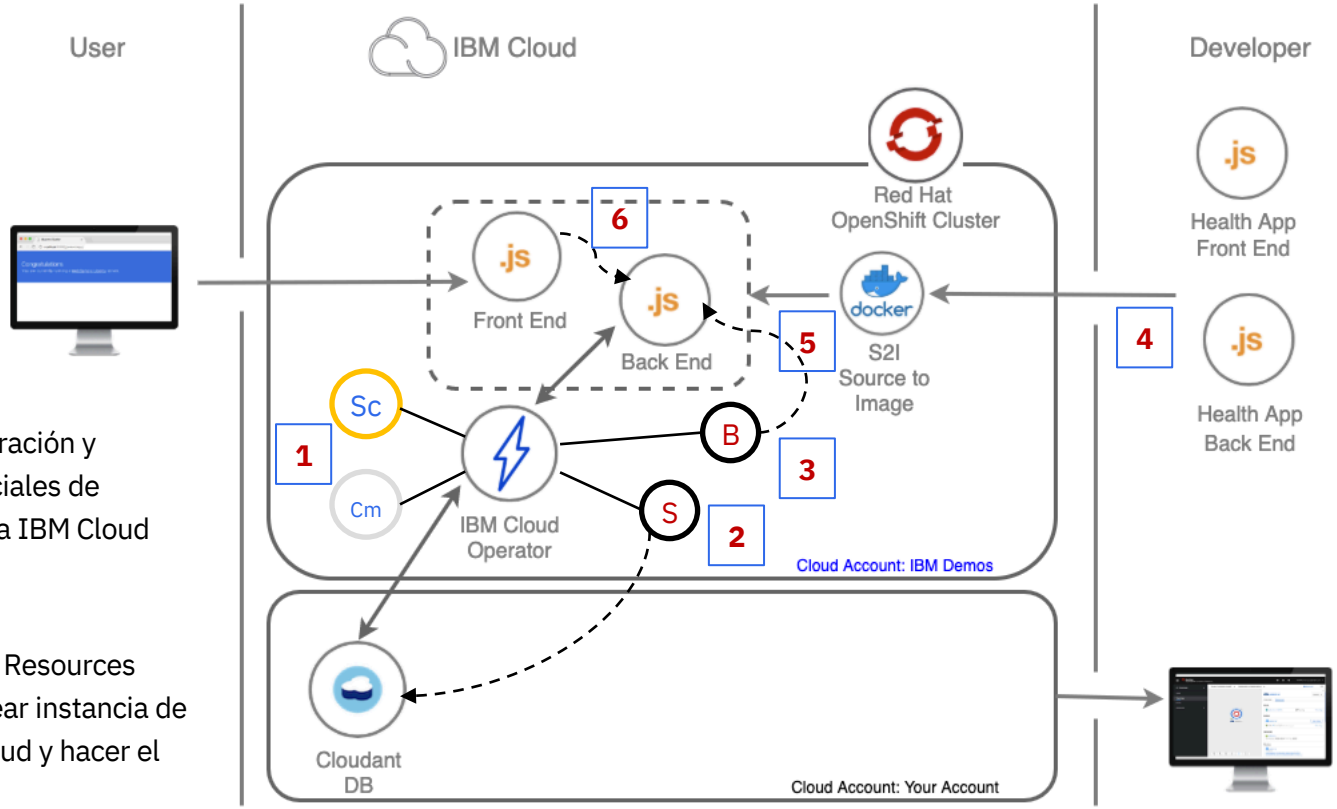
THE INDUSTRY IS ALIGNING BEHIND THE KUBERNETES OPERATOR FRAMEWORK

60+ Certified ISV Operators in Red Hat Early Access Program

Arquitectura de la Aplicación



Resumen del lab



Cm, Sc — Configuración y credenciales de acceso a IBM Cloud

B, S — Custom Resources para crear instancia de IBM Cloud y hacer el binding

1. Crear Operador
2. Crear Service
3. Crear Binding
4. Desplegar app
5. Configurar app
6. Integrar ui-bd