CI CD with Docker

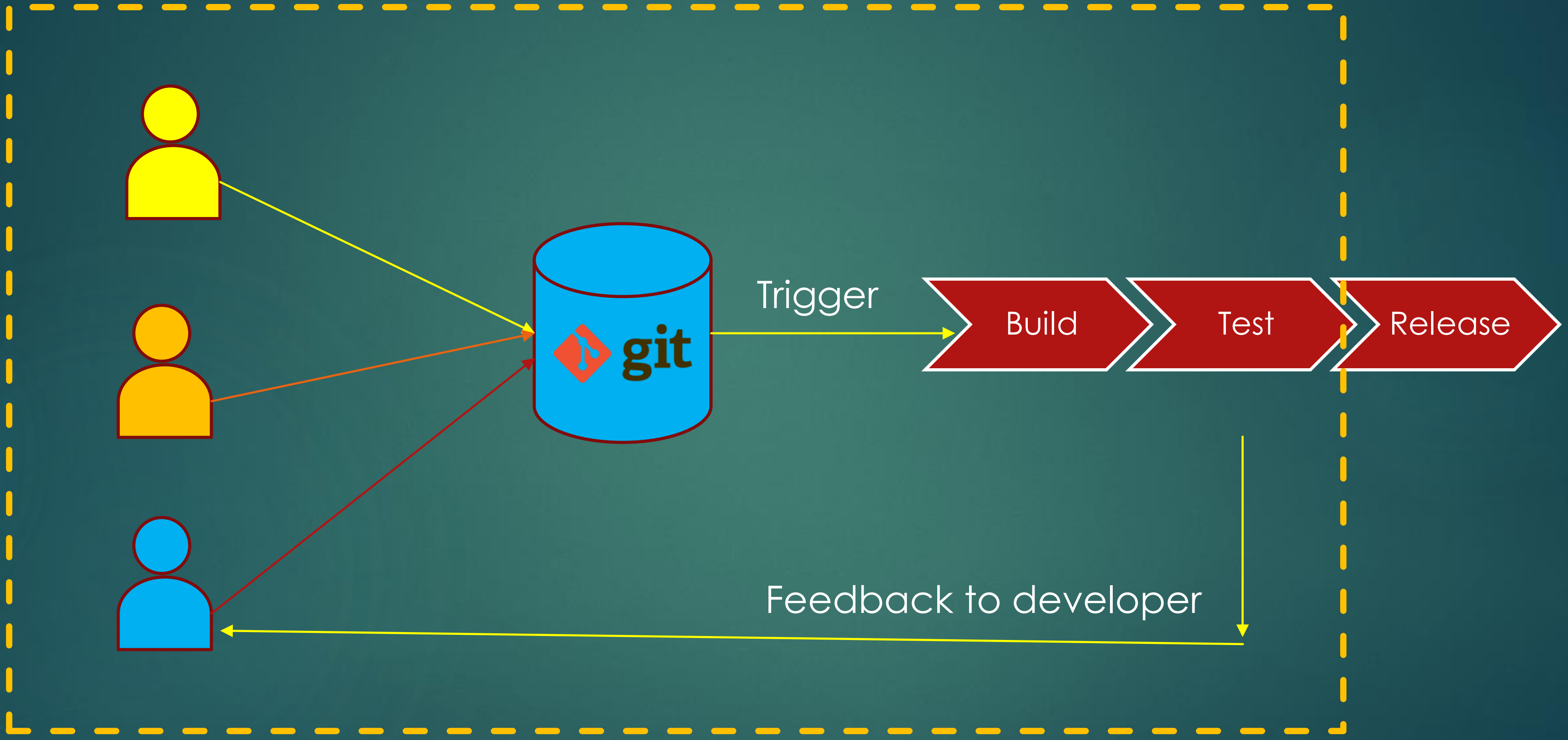
Agenda

- What is Continuous Integration
- What is Continuous Delivery/Deployment
- Tools
- Jenkins
- Docker
- Hands on Demo

What is Continuous Integration

- **Continuous Integration** (CI) is a development practice where developers **integrate** code into a shared repository frequently, preferably several times a day.
- Each **integration** can then be verified by an automated build and automated tests.
- By integrating regularly, you can detect errors quickly, and locate them more easily.

What is Continuous Integration



Trigger

Build

Test

Release

Feedback to developer

Continuous Delivery

- Continuous delivery (CD) is a software engineering approach in which teams produce software in short cycles, ensuring that the software can be reliably released at any time and, when releasing the software, doing so manually.

- It aims at building, testing, and releasing software with greater speed and frequency.

- The approach helps reduce the cost, time, and risk of delivering changes by allowing for more incremental updates to applications in production.

Continuous Delivery

- Continuous delivery (CD) is a software engineering approach in which teams produce software in short cycles, ensuring that the software can be reliably released at any time and, when releasing the software, doing so manually.

- It aims at building, testing, and releasing software with greater speed and frequency.

- The approach helps reduce the cost, time, and risk of delivering changes by allowing for more incremental updates to applications in production.

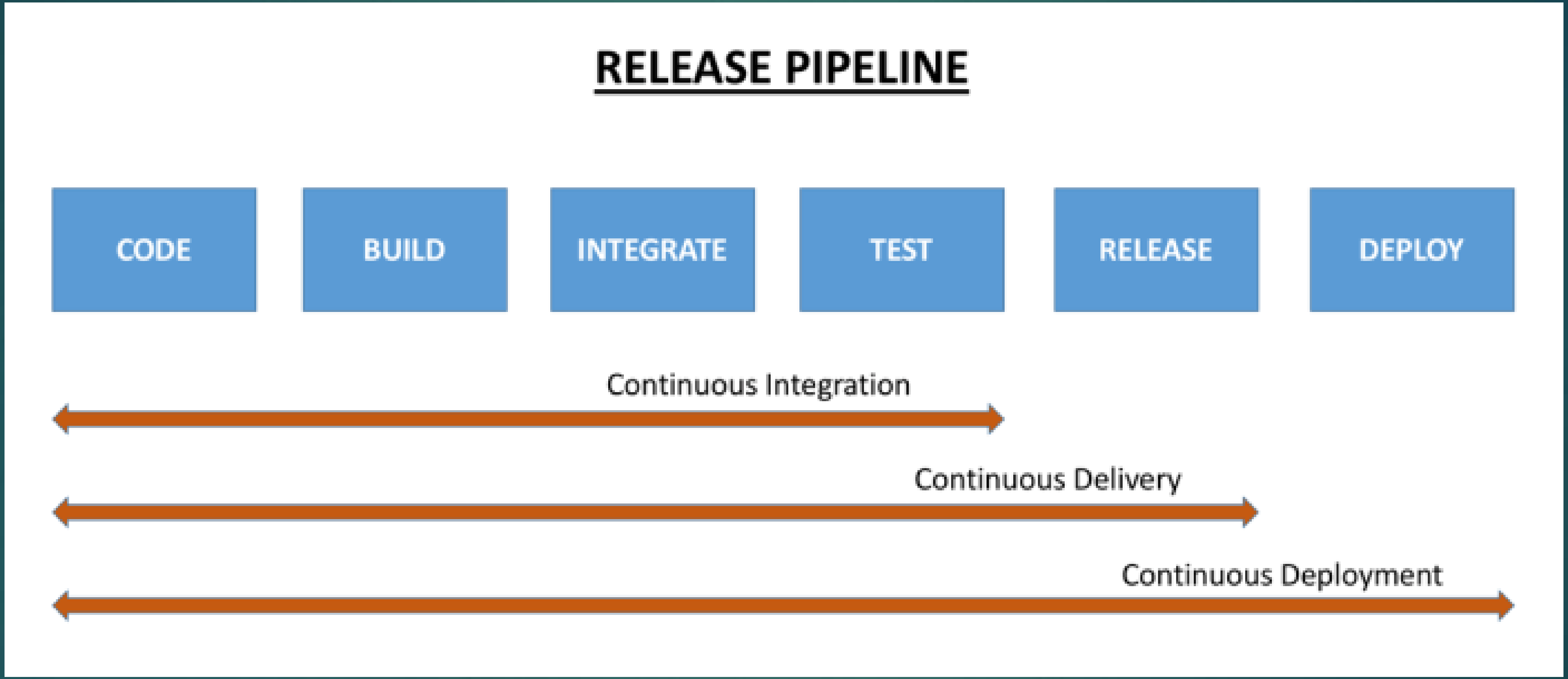
Continuous Delivery

- ***Continuous Delivery** goes one step further than Continuous Integration to automate the software release process, where code is deployed to environments and is prepared for Production release.*

Continuous Deployment

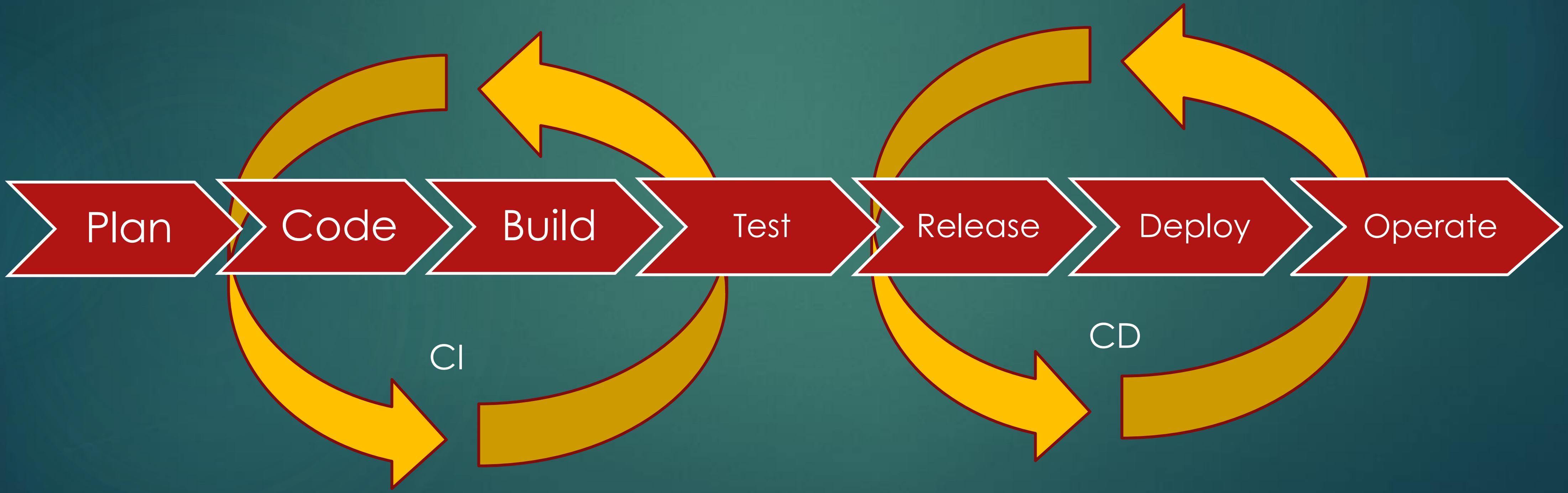
- ***Continuous Deployment*** *is the next step of Continuous Delivery, where every code commit that passes the Automated Tests and other Acceptance tests, is automatically deployed to Production without any manual intervention.*

The Release Pipeline



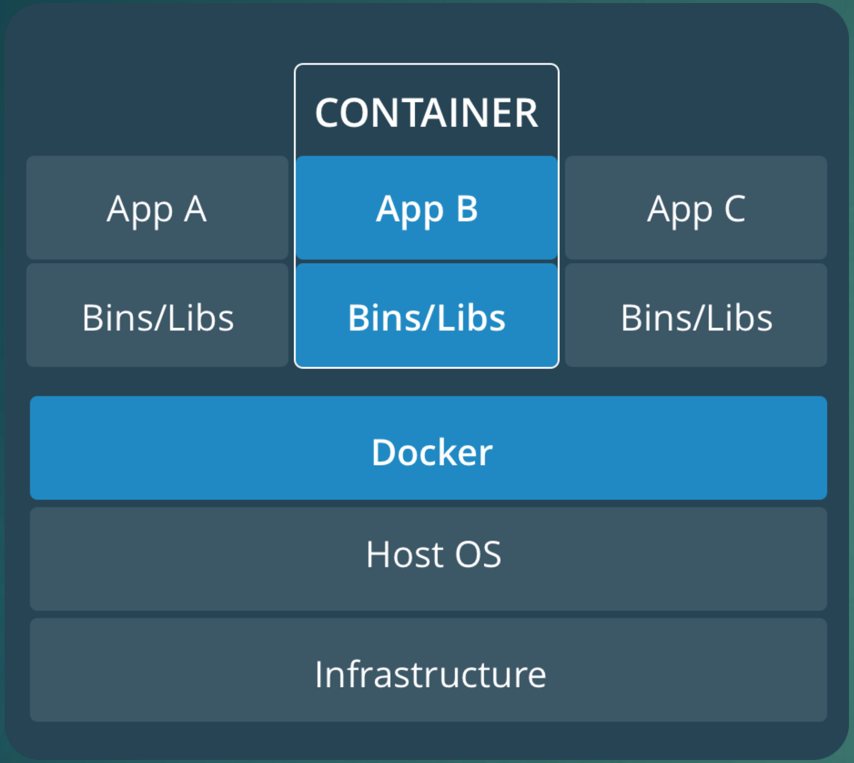
What is CI/CD Pipeline

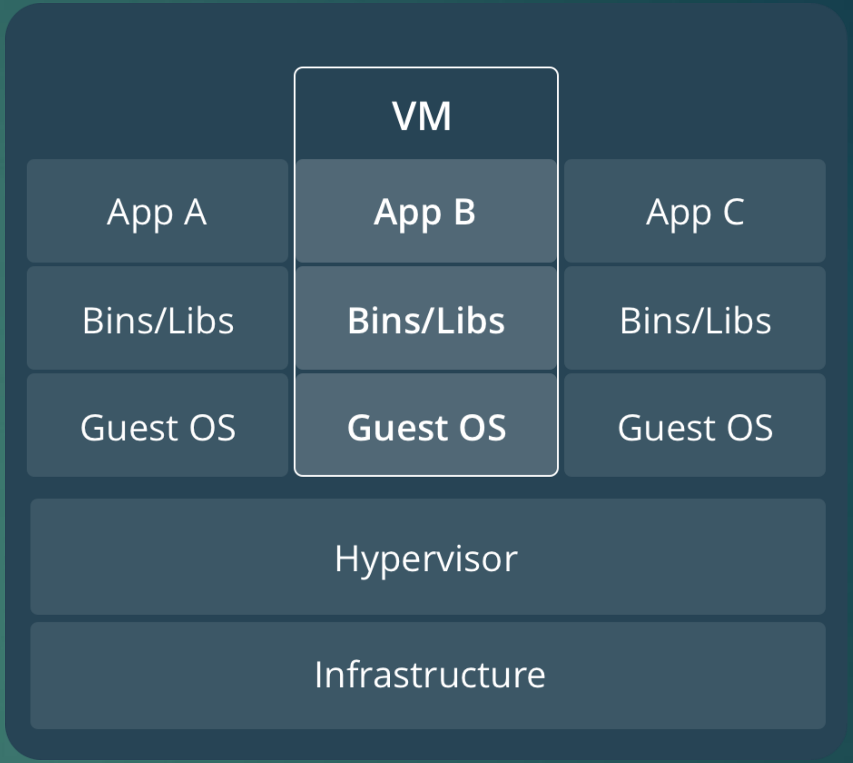
- The stages which constitute various activities from development to final production deployment of a software is the CI/CD pipeline



Docker CI/CD

Comparing Containers and VMs

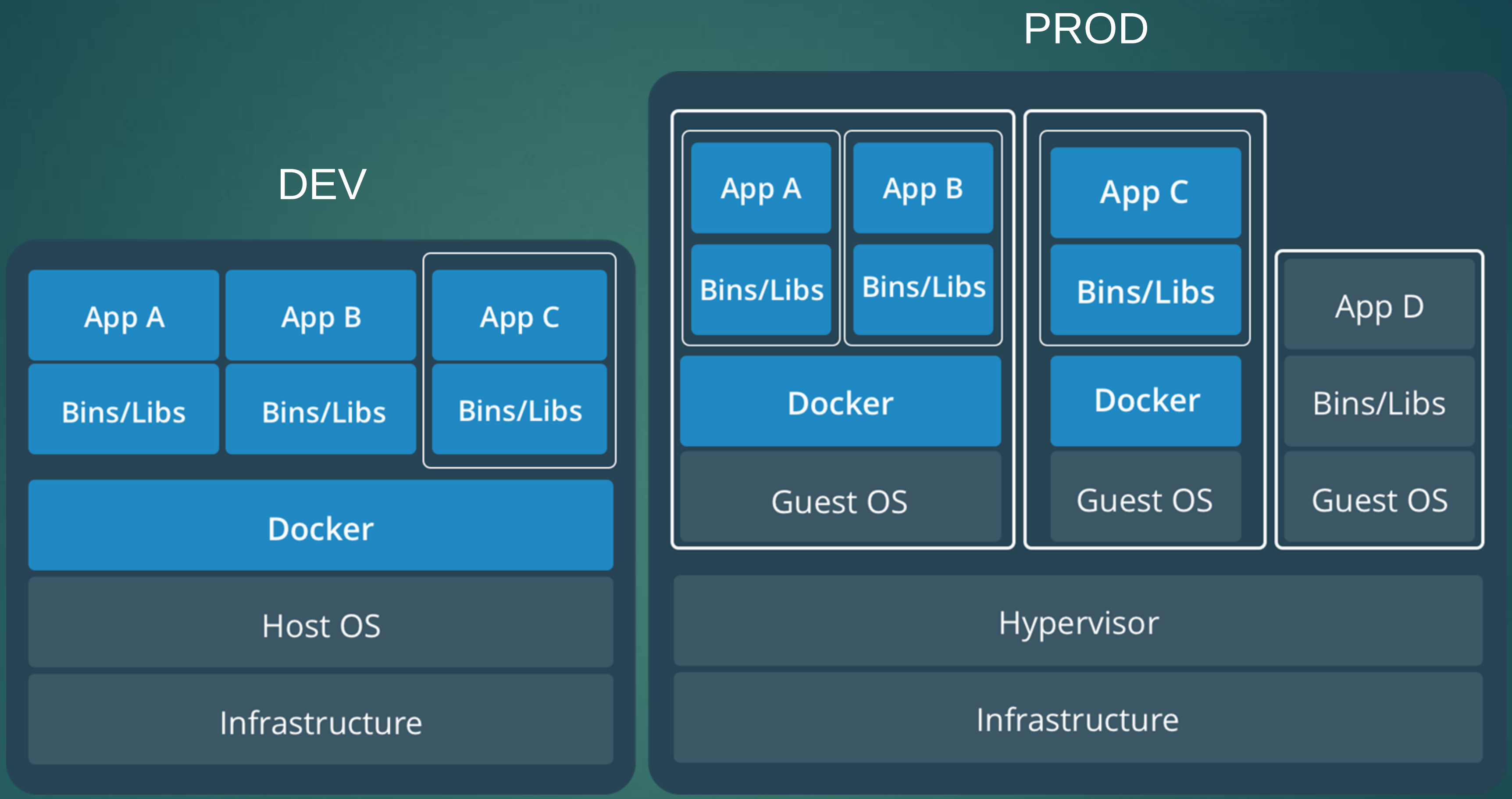




Containers are an app level construct

VMs are an infrastructure level construct to turn one machine into many servers

Containers and VMs together



Containers and VMs together provide a tremendous amount of flexibility for IT to optimally deploy and manage apps.

Key Benefits of Docker Containers

Speed

- No OS to boot = applications online in seconds

Portability

- Less dependencies between process layers = ability to move between infrastructure

Efficiency

- Less OS overhead
- Improved VM density

Container Solutions & Landscape

Docker Basics

Image

The basis of a Docker container. The content at rest.

Container

The image when it is ‘running.’ The standard unit for app service

Engine

The software that executes commands for containers. Networking and volumes are part of Engine. Can be clustered together.

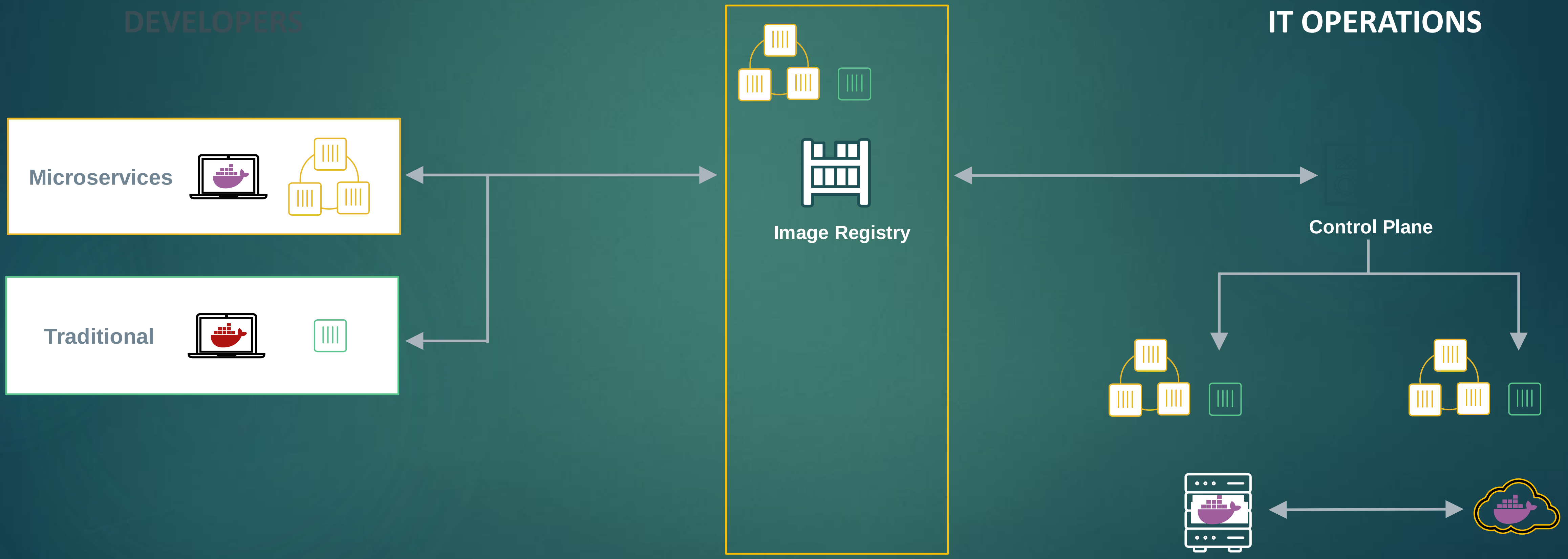
Registry

Stores, distributes and manages Docker images

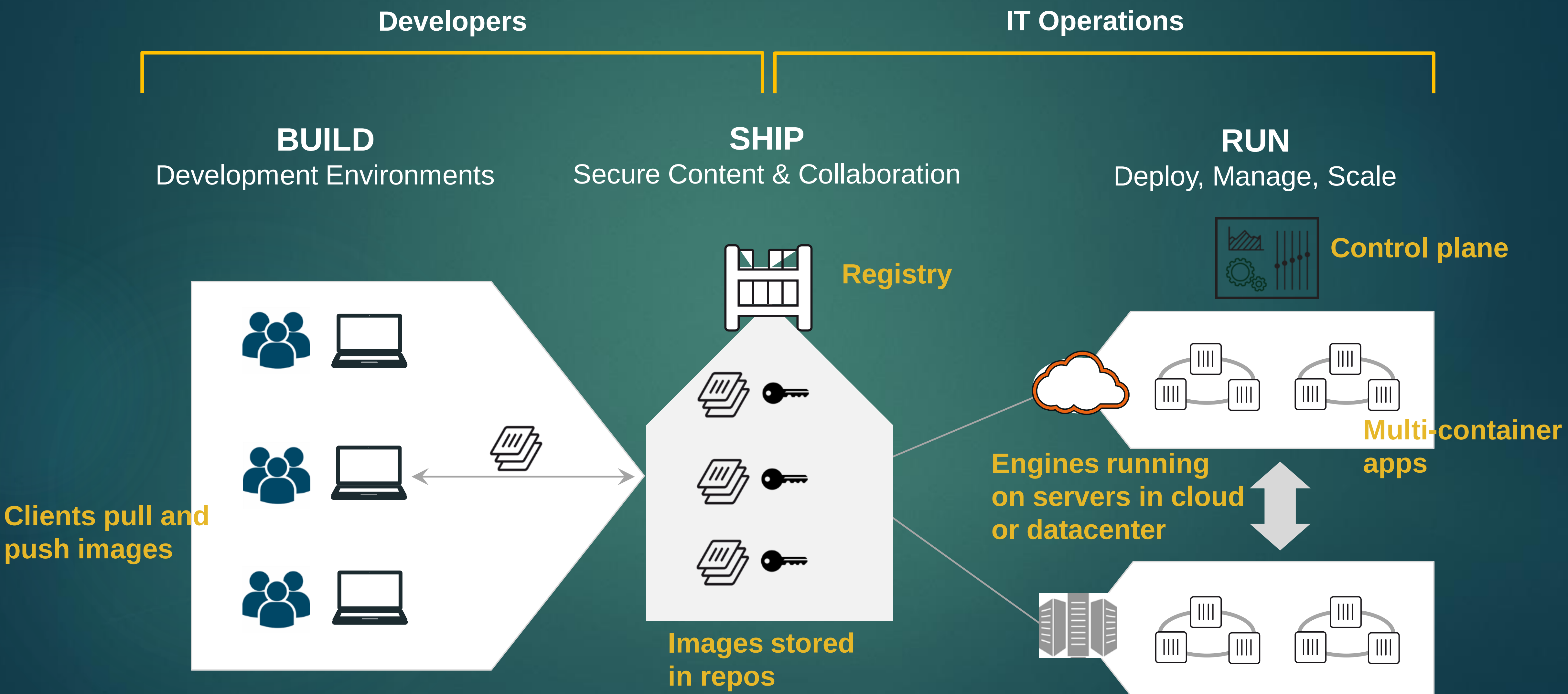
Control Plane

Management plane for container and cluster orchestration

Building a Software Supply Chain



Containers as a Service



Developers

IT Operations

BUILD
Development Environments

SHIP
Secure Content & Collaboration

RUN
Deploy, Manage, Scale

Control plane

Registry

Multi-container apps

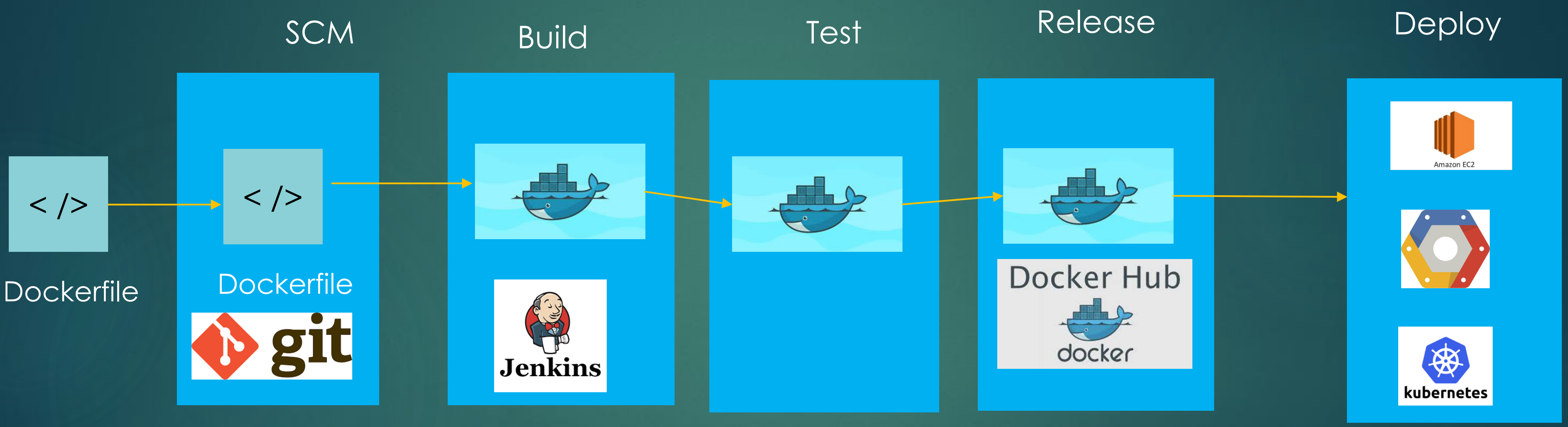
Engines running on servers in cloud or datacenter

Clients pull and push images

Images stored in repos

CI/CD with Docker and Jenkins

The CI/CD Landscape



SCM

Build

Test

Release

Deploy

< />

Dockerfile

< />

Dockerfile

git

Jenkins

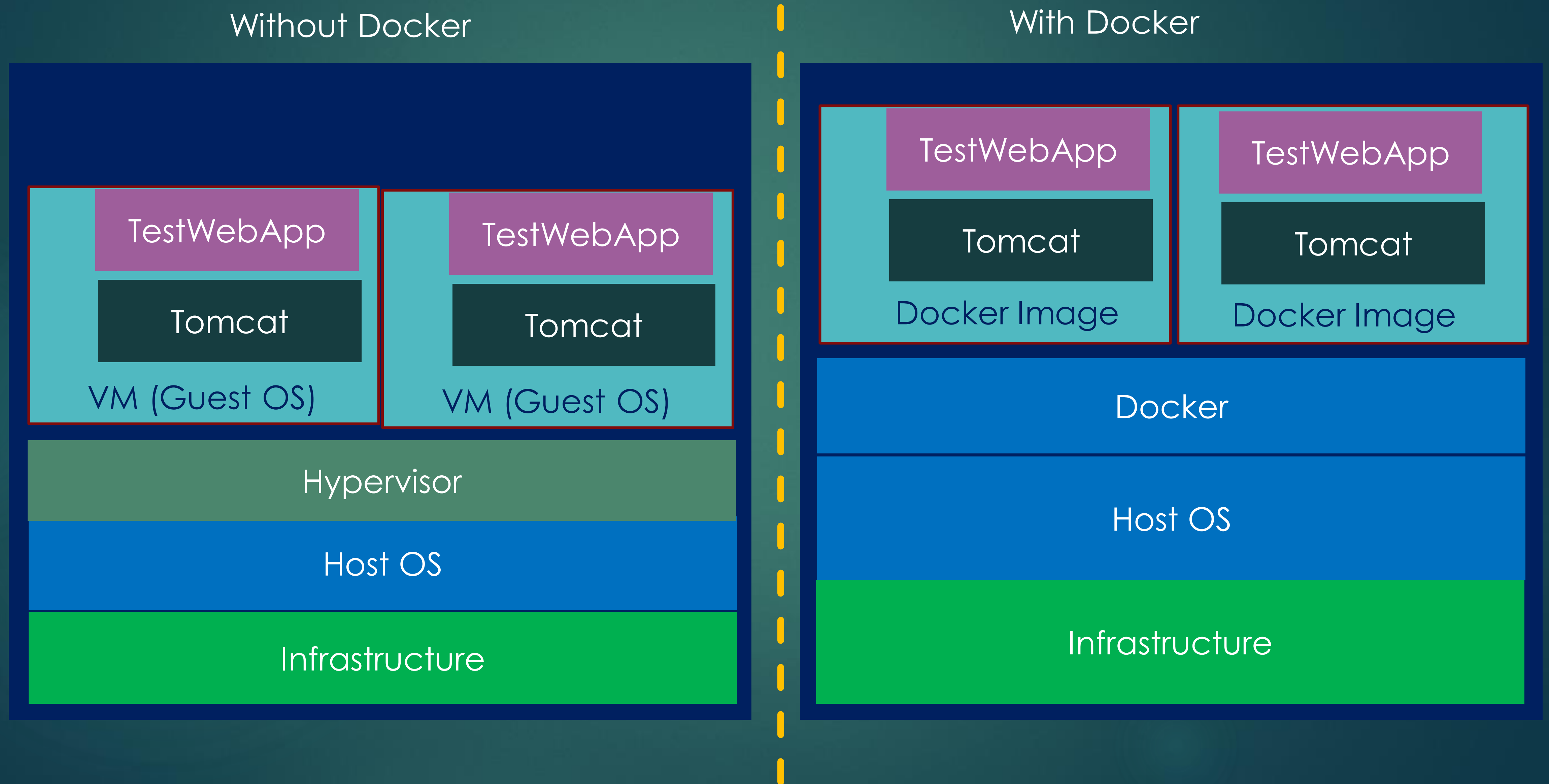
Docker Hub

docker

Amazon EC2

kubernetes

The Demo Application



Without Docker

TestWebApp
Tomcat
VM (Guest OS)

TestWebApp
Tomcat
VM (Guest OS)

Hypervisor

Host OS

Infrastructure

With Docker

TestWebApp
Tomcat
Docker Image

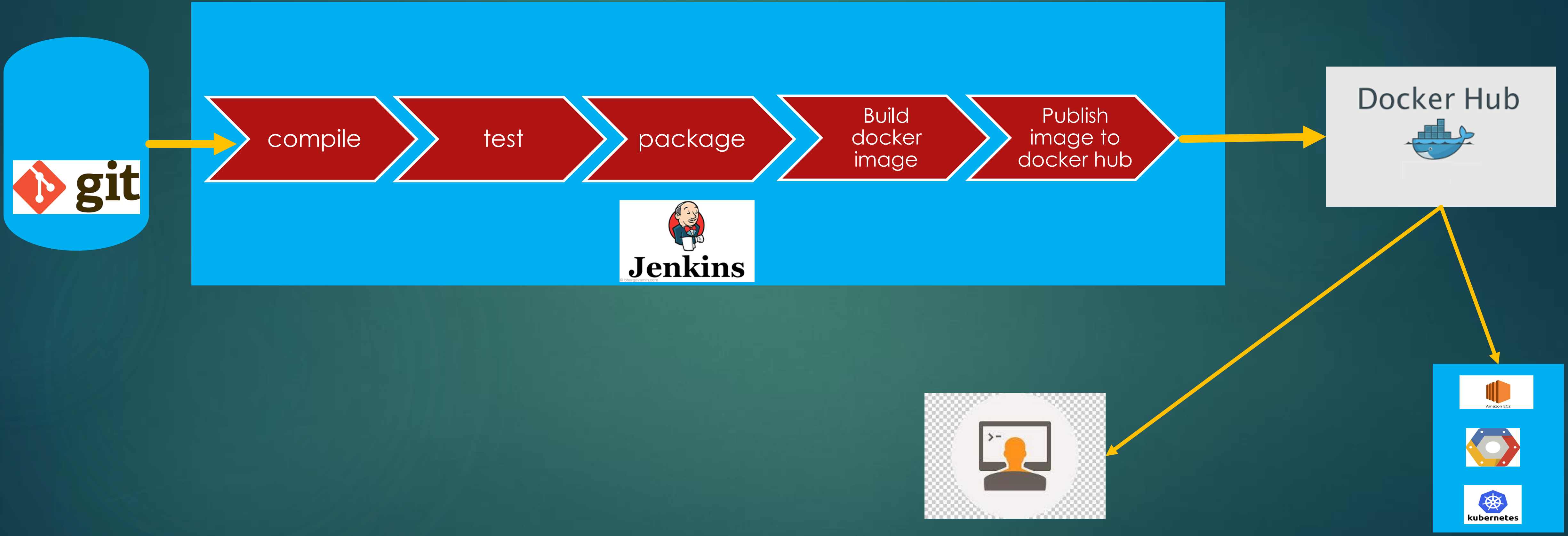
TestWebApp
Tomcat
Docker Image

Docker

Host OS

Infrastructure

Our Course of Actions



compile

test

package

Build docker image

Publish image to docker hub

Docker Hub

Test

Deploy