

No thanks!

We are  
too busy

**MANUAL BUILDS**

PAINFUL

HIGH COST

AVOID CHANGES

LOSS OF BUSINESS

DEMORALIZING

FRAGILE

~90 APPLICATIONS!

NIGHTMARE

NO PROCESS

MONOLITHIC MONSTER

HIGH

MAINTENANCE

LOSS OF REPUTATION

HIGH T2M

NO LOCAL DEV

TANGLED DEPENDENCIES

NO

TESTABILITY

CD/CI

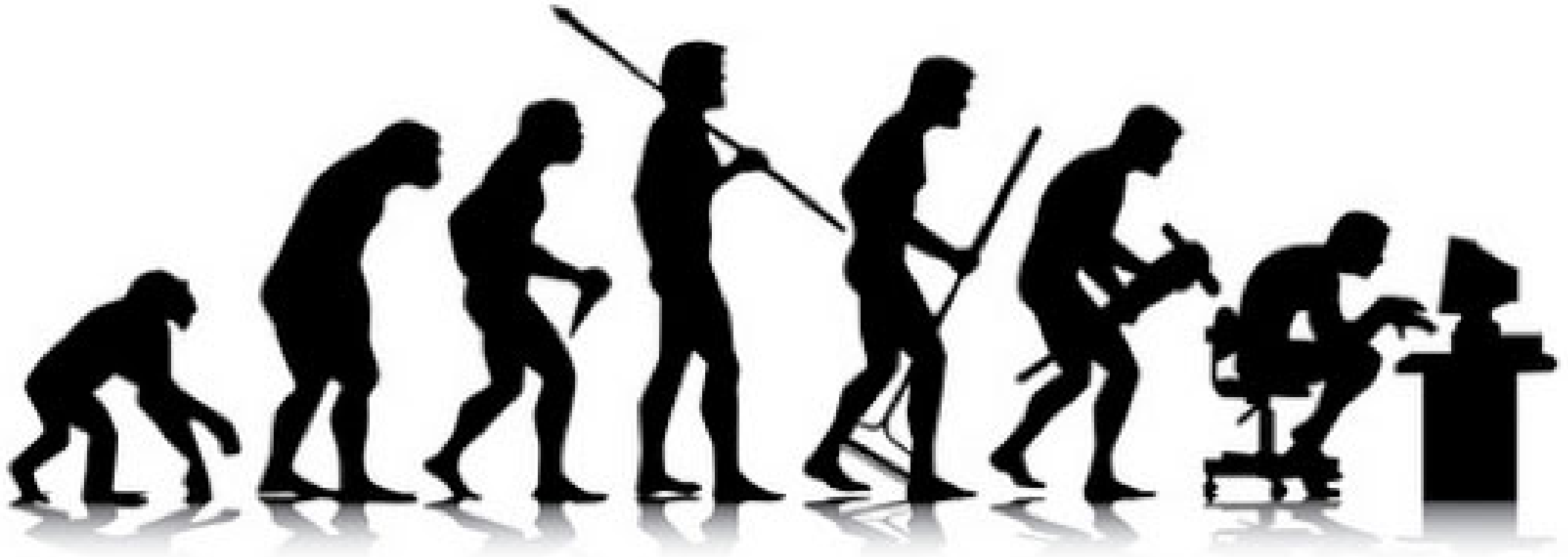
LEGACY TECH STACK

JAVA 1.5 TO 1.7!!!

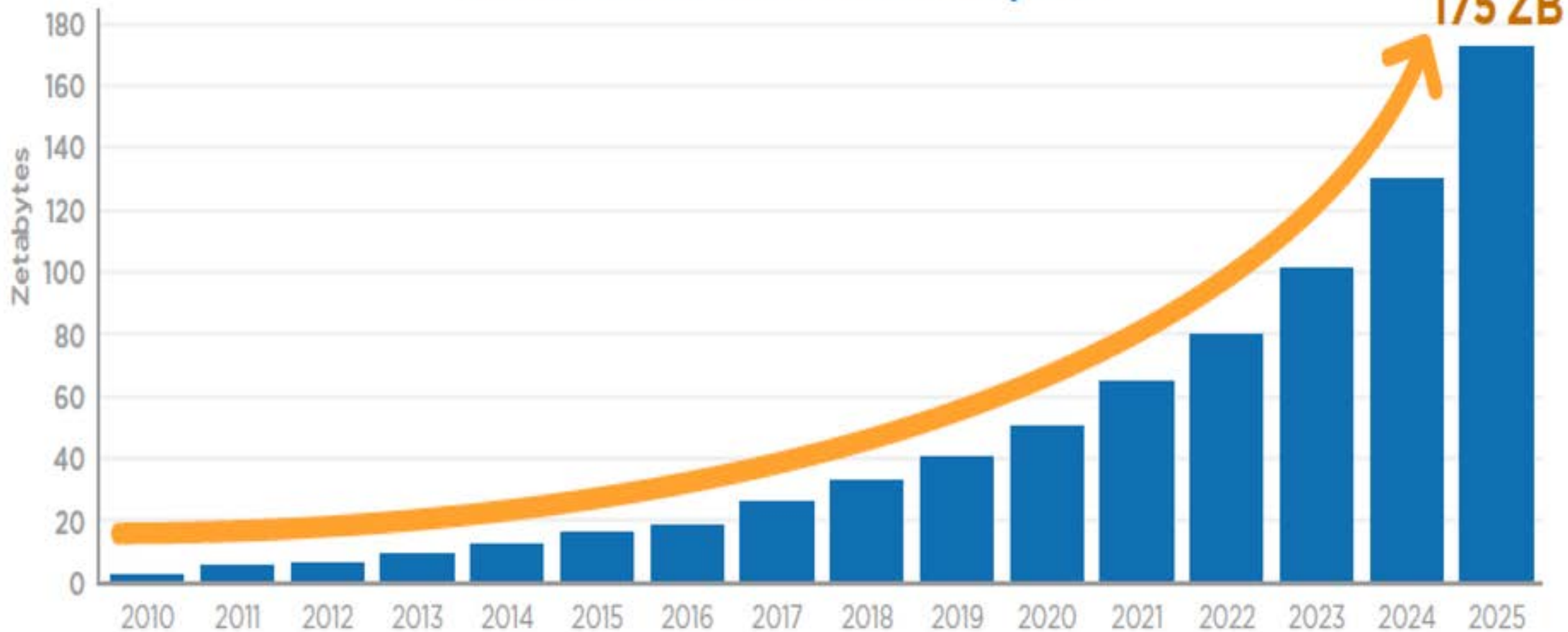


# Exhilarating Journey of Transformation into Digital Edge

# LEAN, AGILE & DEVOPS ADOPTION



## Annual Size of the Global Datasphere



Source: Data Age 2025, sponsored by Seagate with data from IDC Global DataSphere, Nov 2018



## Digitization

The process of making information available and accessible in a digital format.

## Digitalization

The process of considering how best to apply digitized information to simplify specific operations.



## Digital Transformation

The process of devising new business applications that integrate all the digitized data and digitalized applications.

## Management



## Development



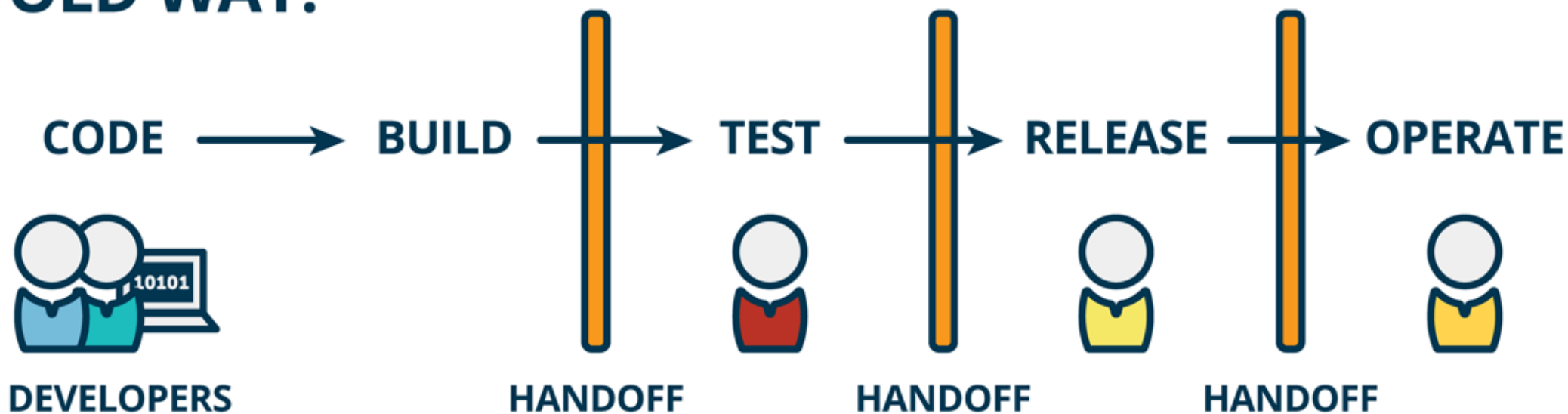
## QA



## IT Operations

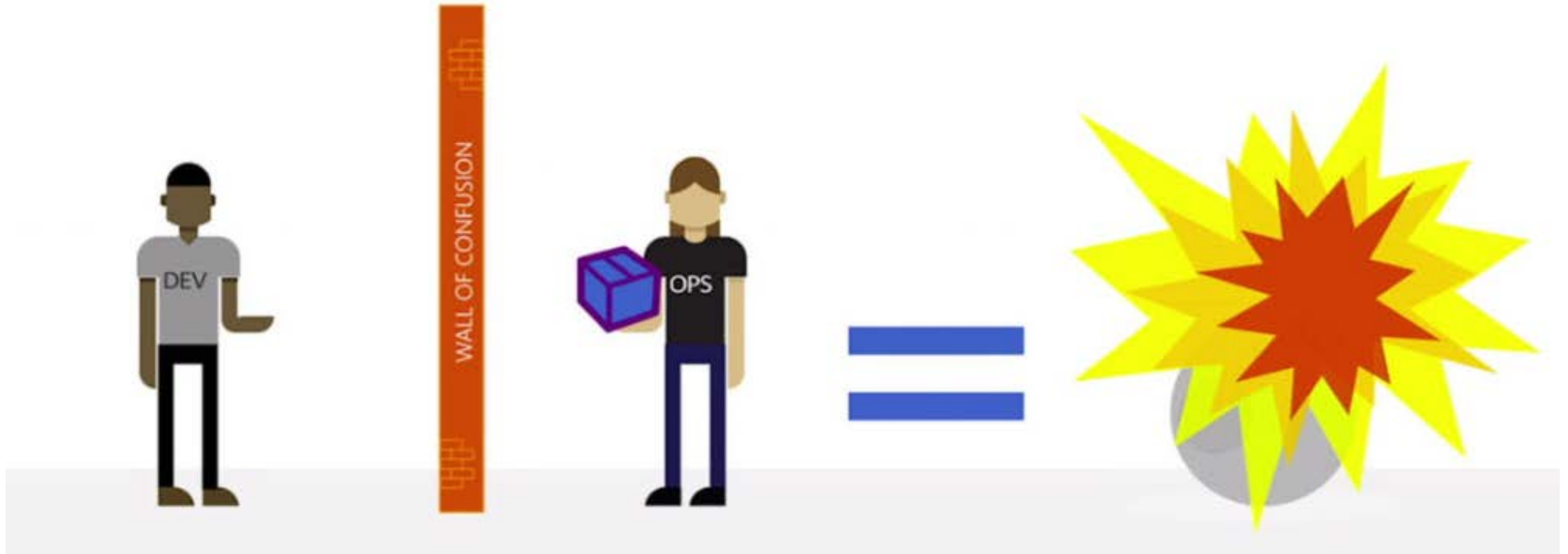


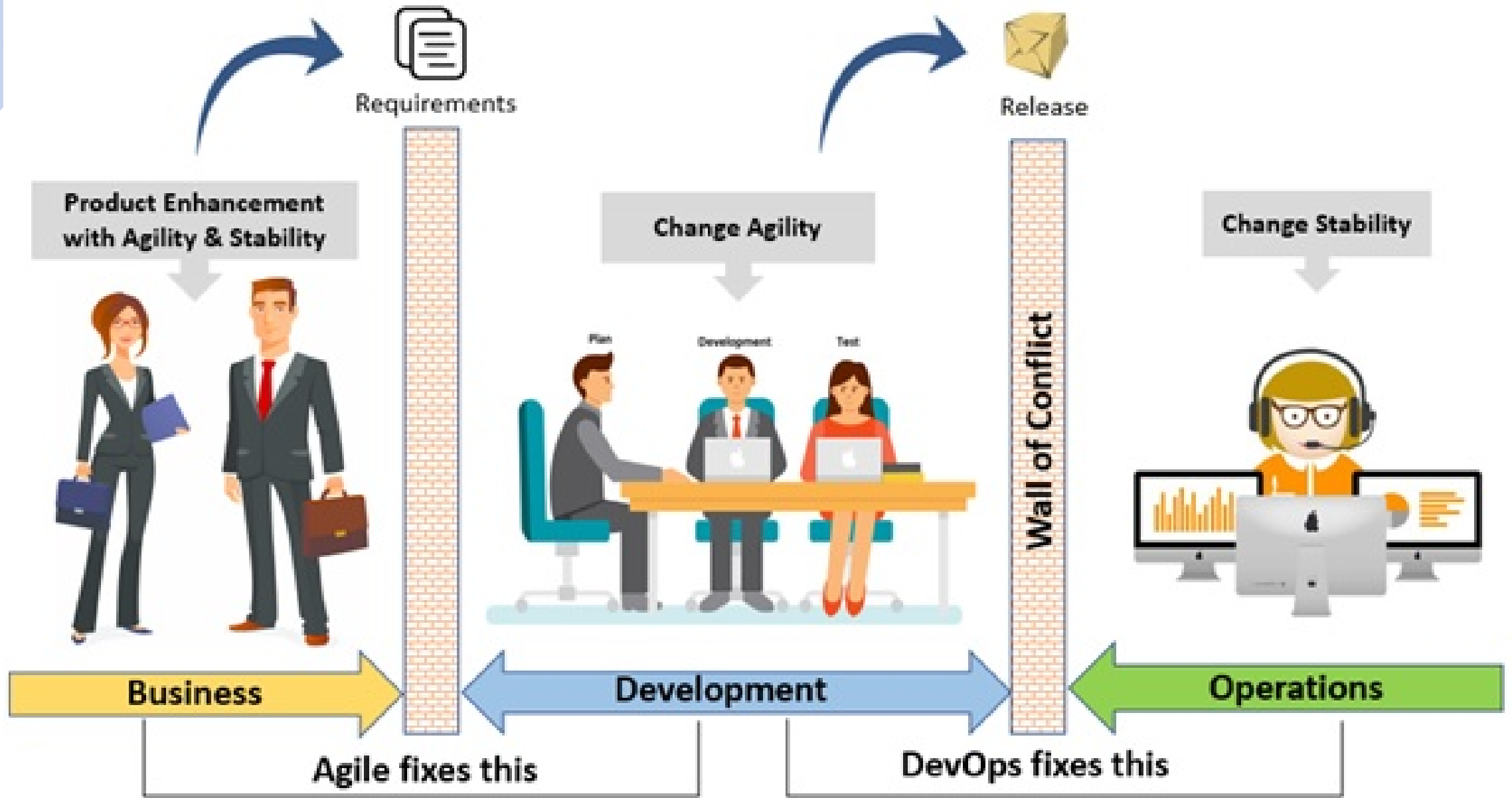
## OLD WAY:



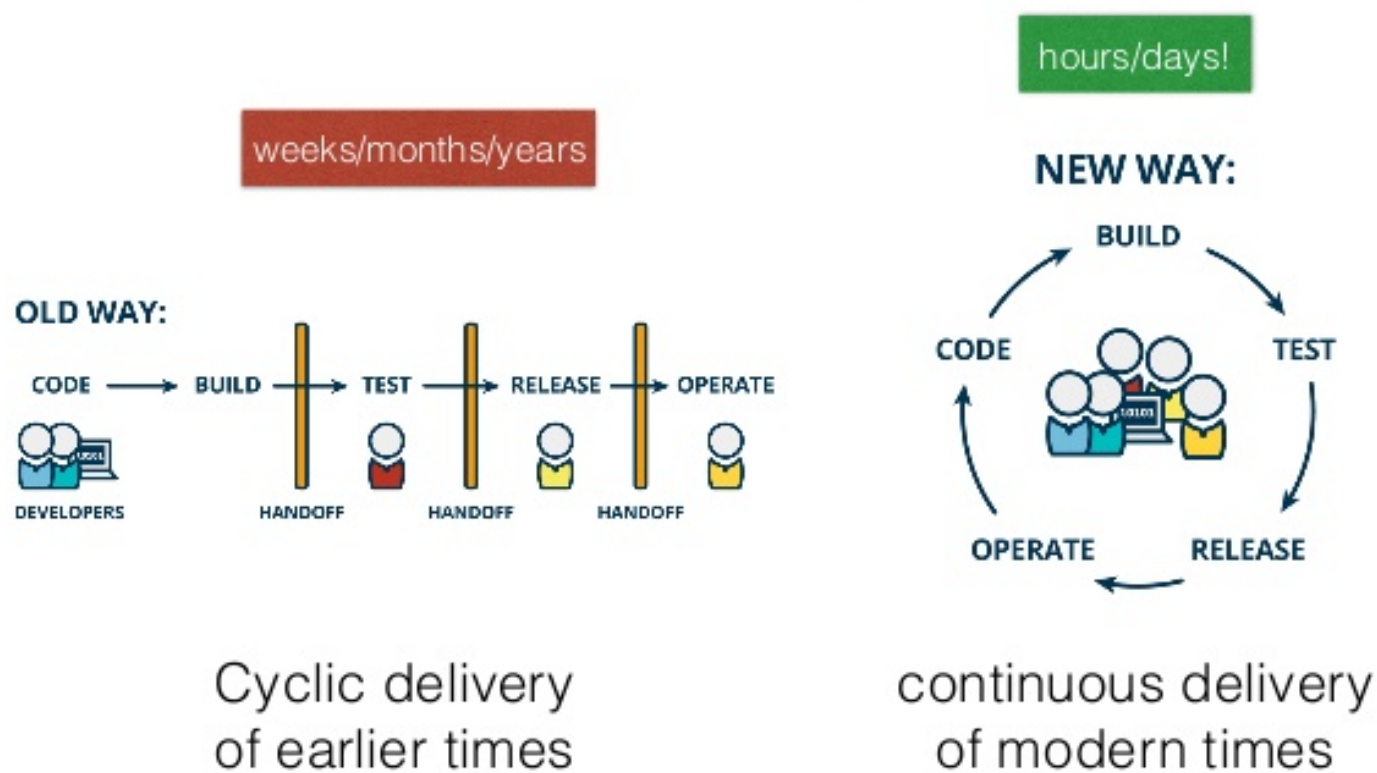


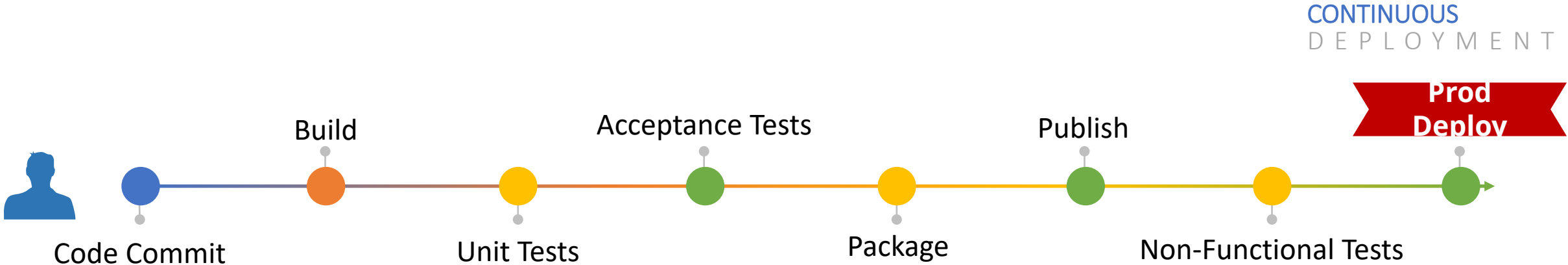
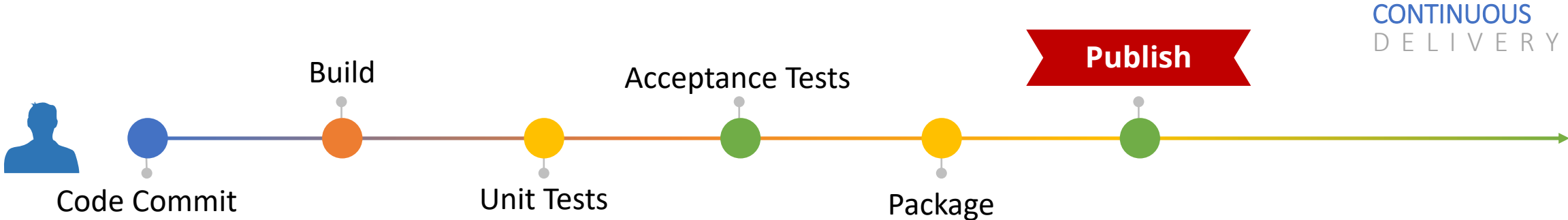
# Traditional Development and Operations

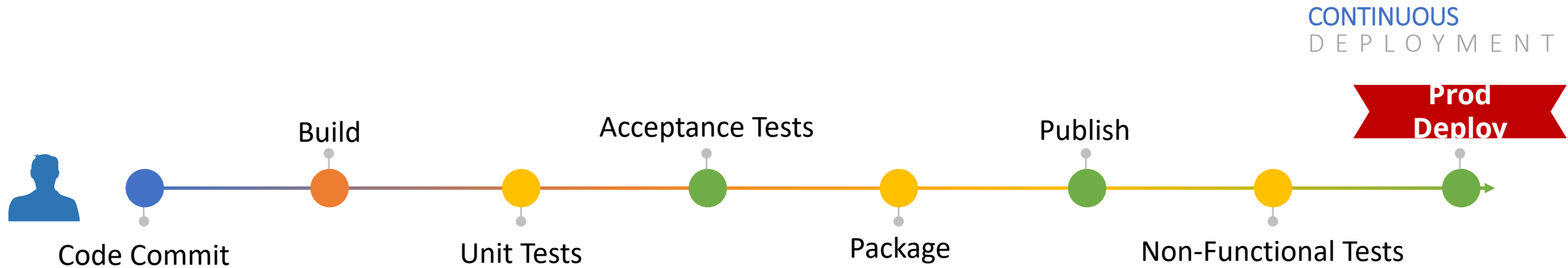
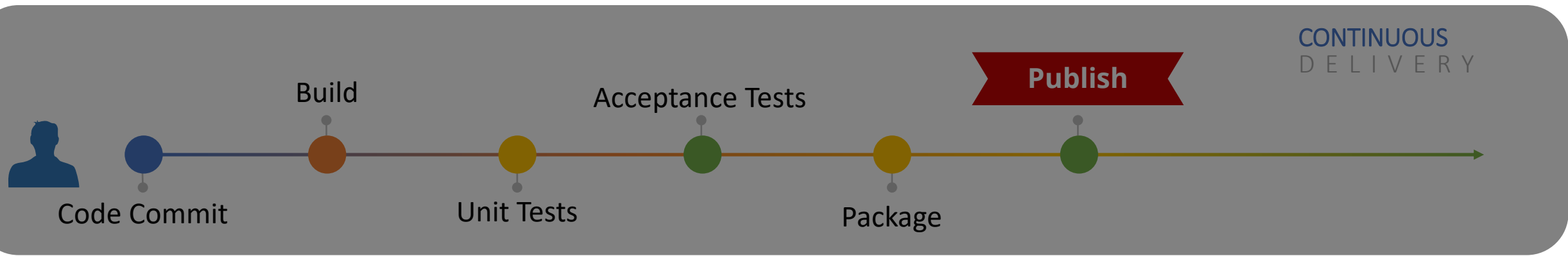
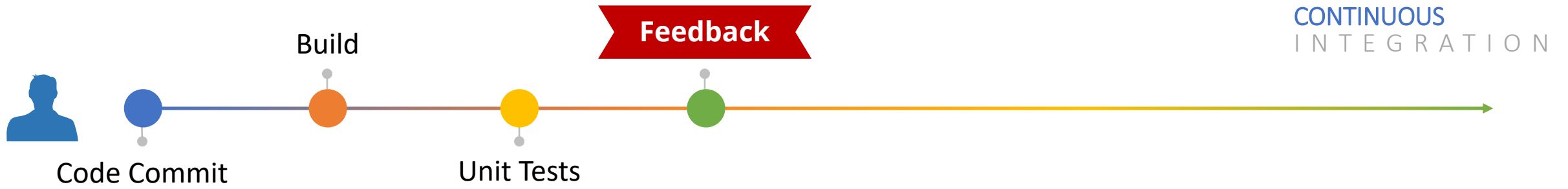


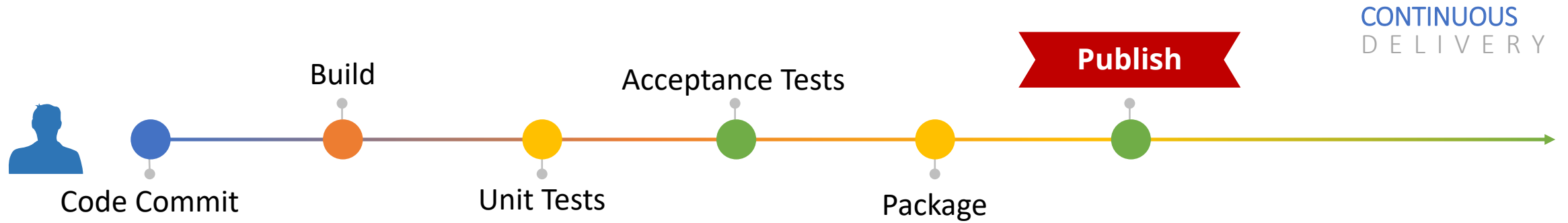
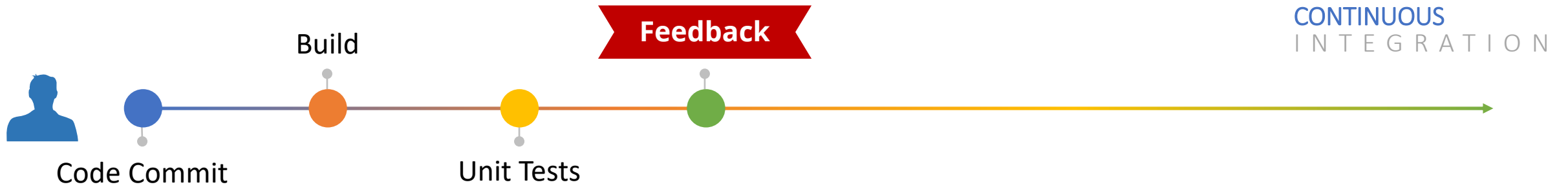


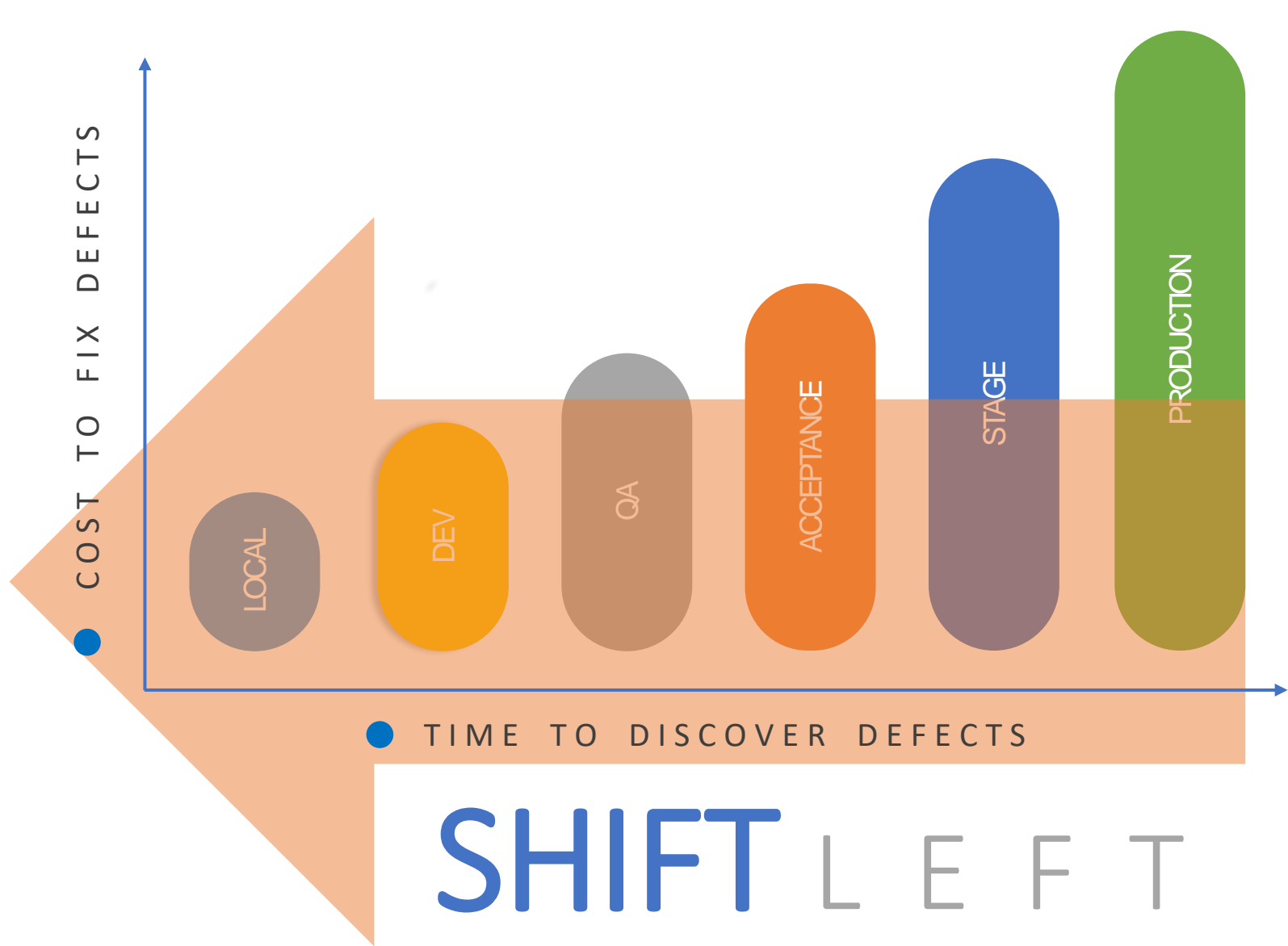
# Cyclic vs. continuous delivery











# SHIFT LEFT



EARLY DETECTION



TIME SAVING

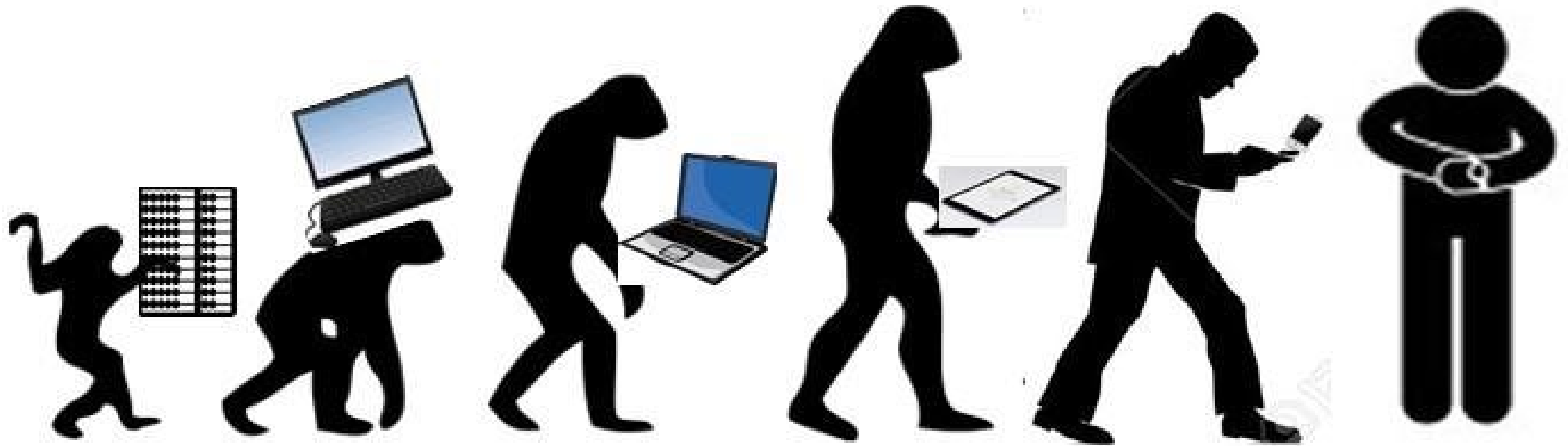


COST SAVING



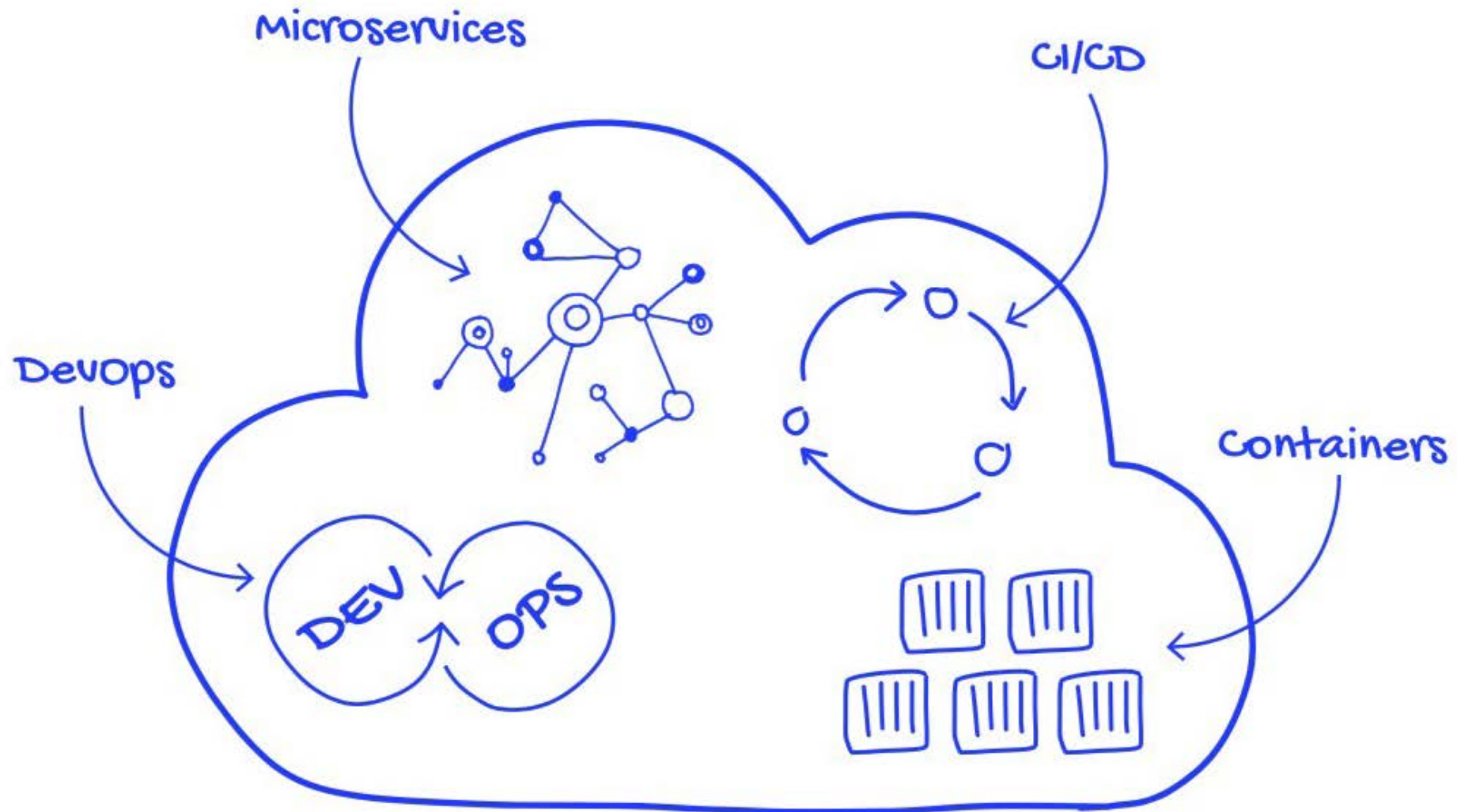
SMOOTH RELEASES

# CLOUD NATIVE ADOPTION

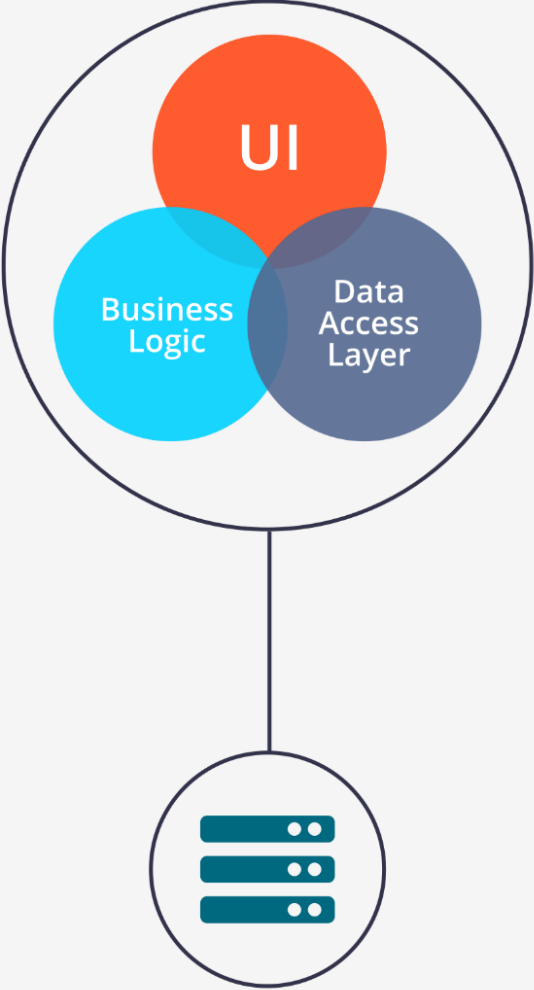




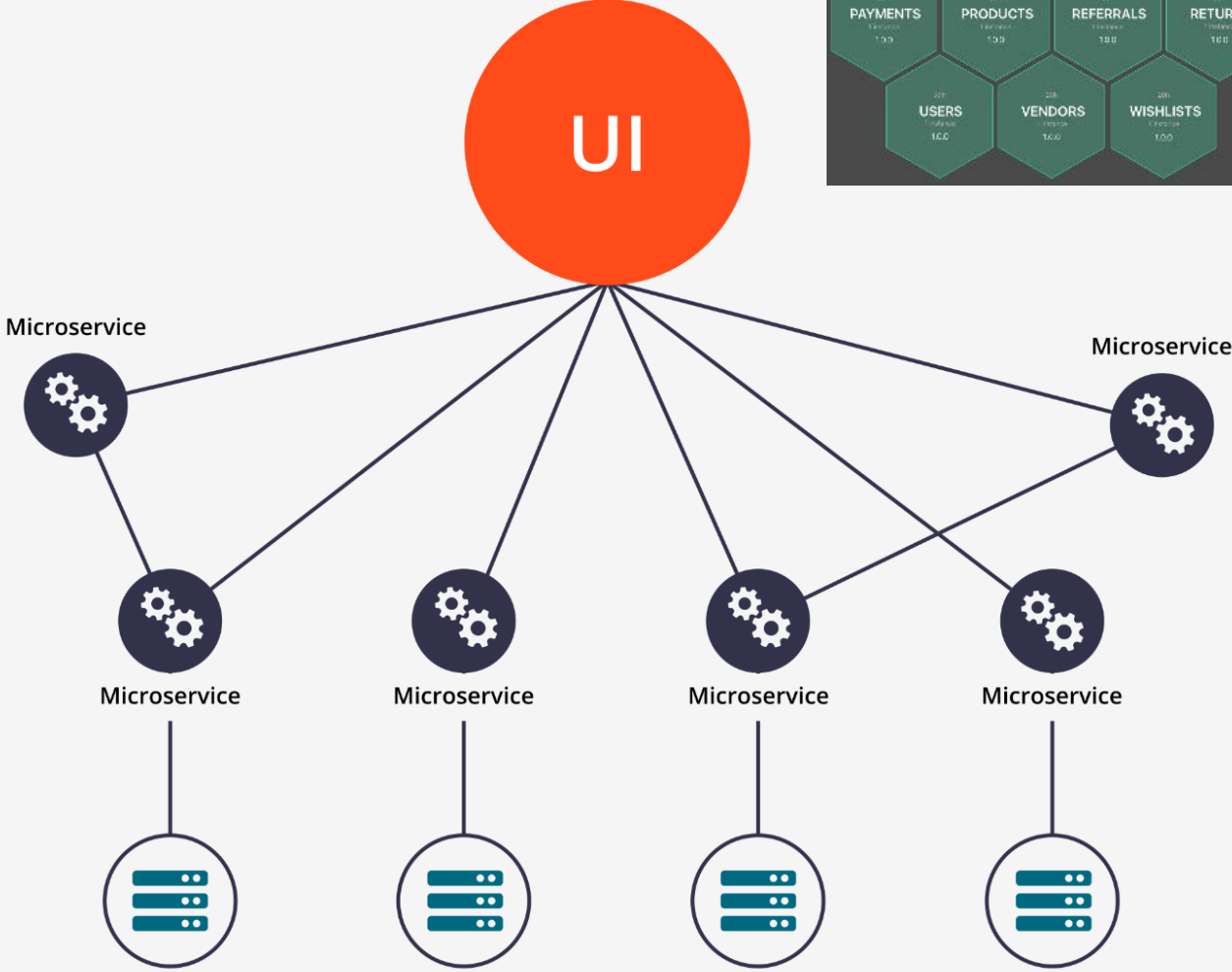
# Cloud Native



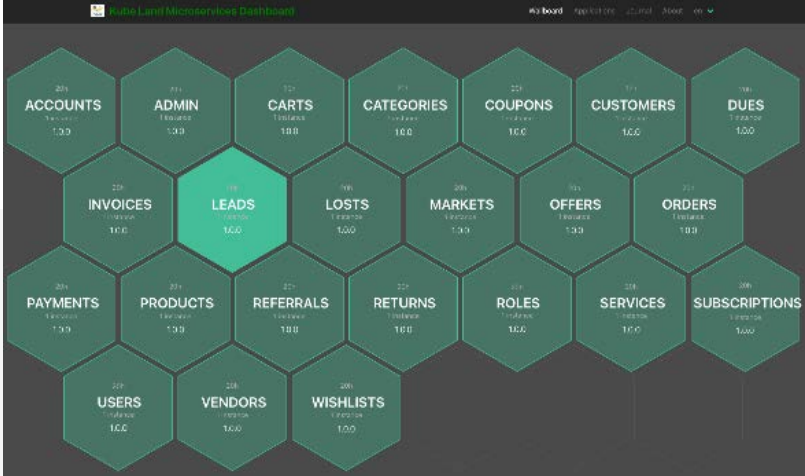
# Monolith vs. Microservices



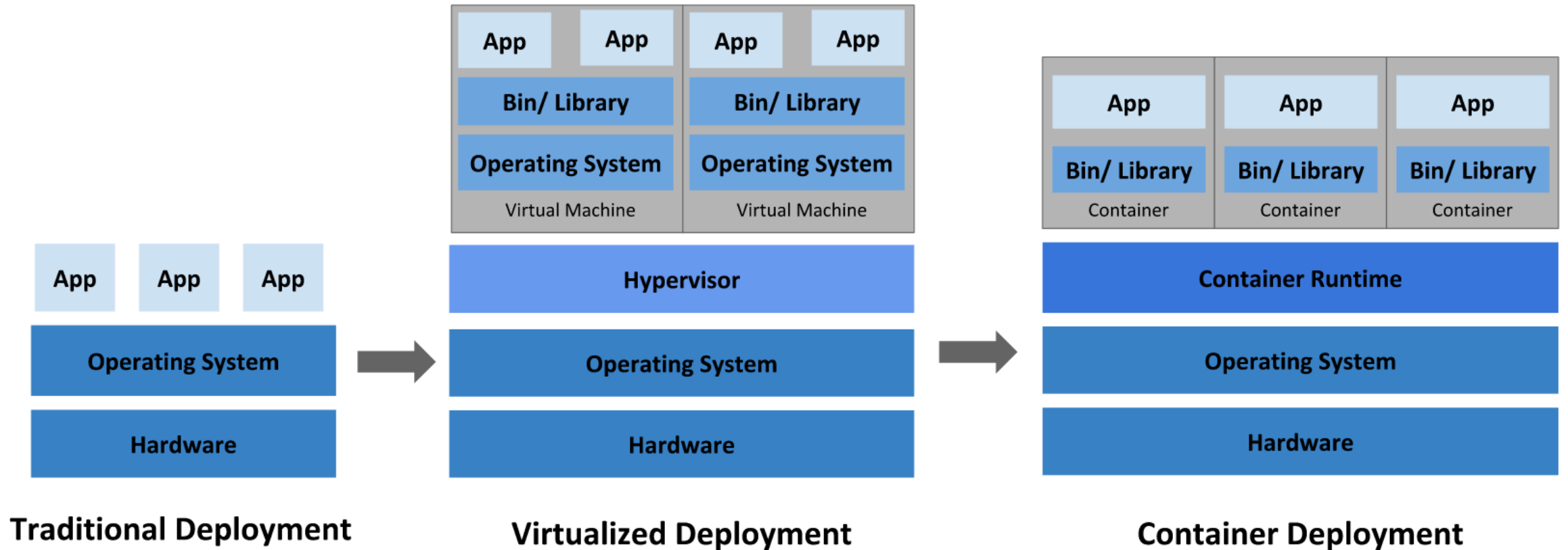
Monolithic Architecture

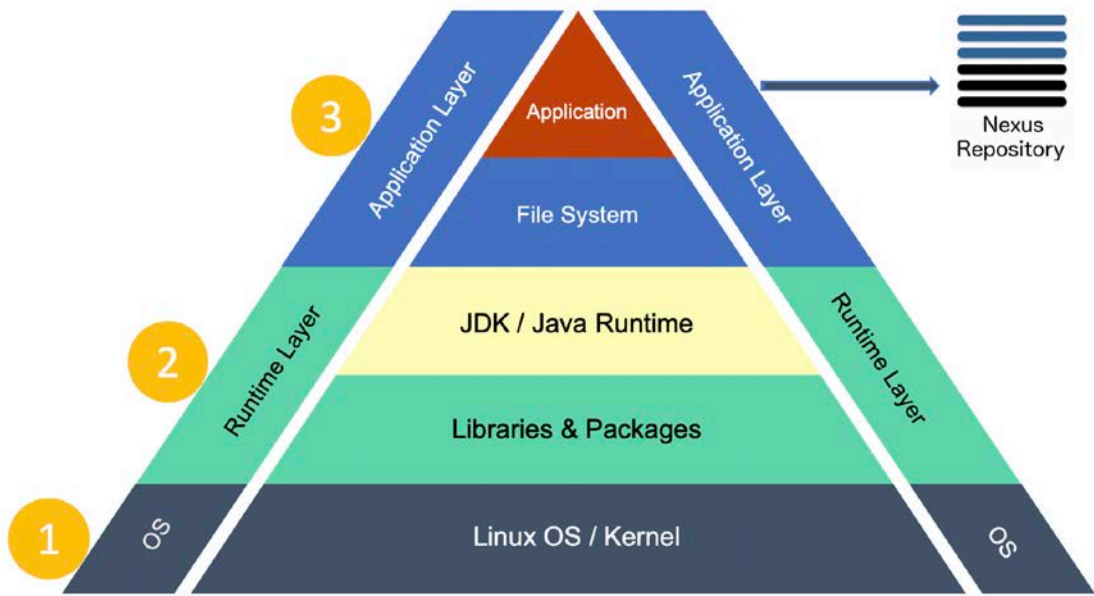


Microservice Architecture

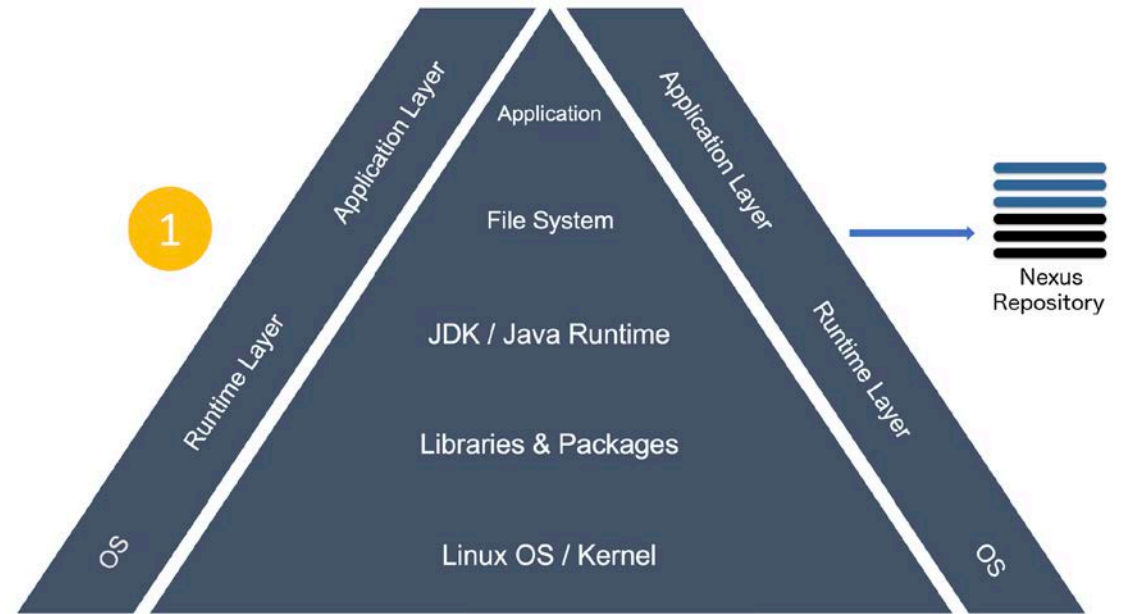


# EVOLVING DEPLOYMENTS





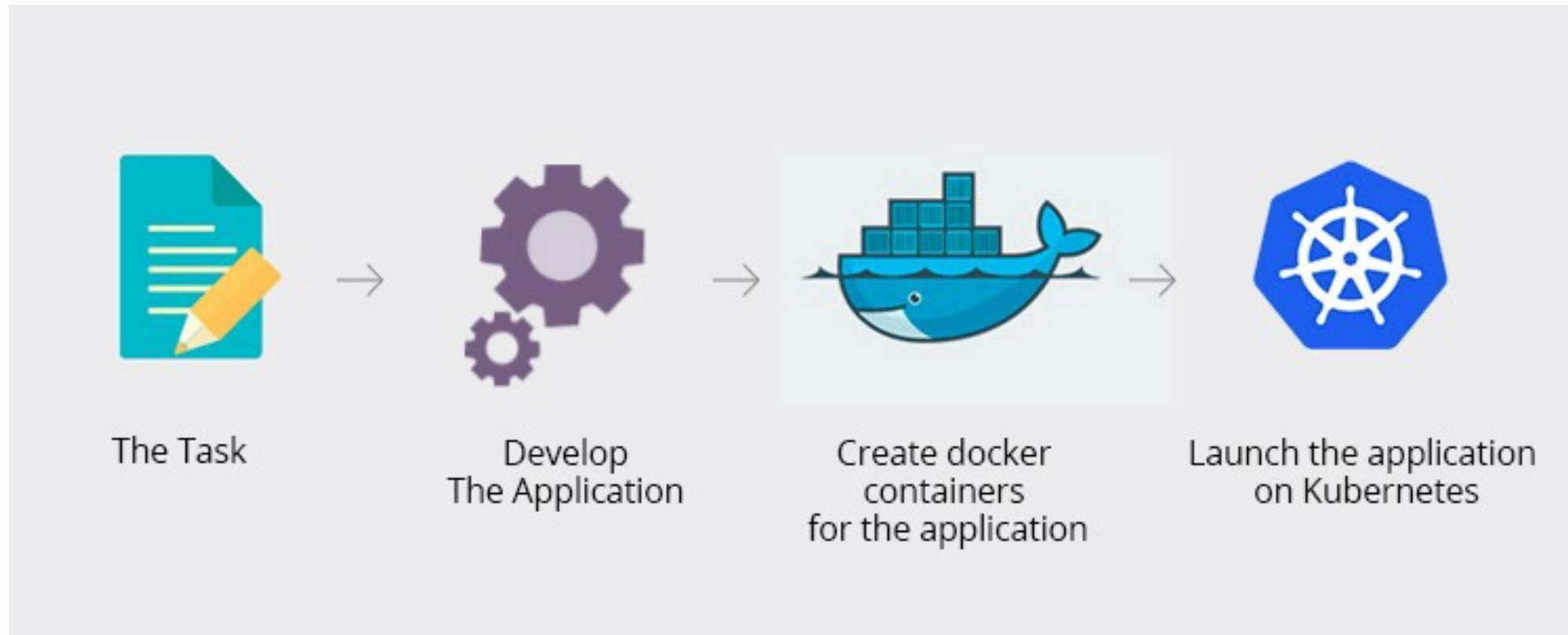
TRADITIONAL DEPLOYMENT



CONTAINERIZED DEPLOYMENT

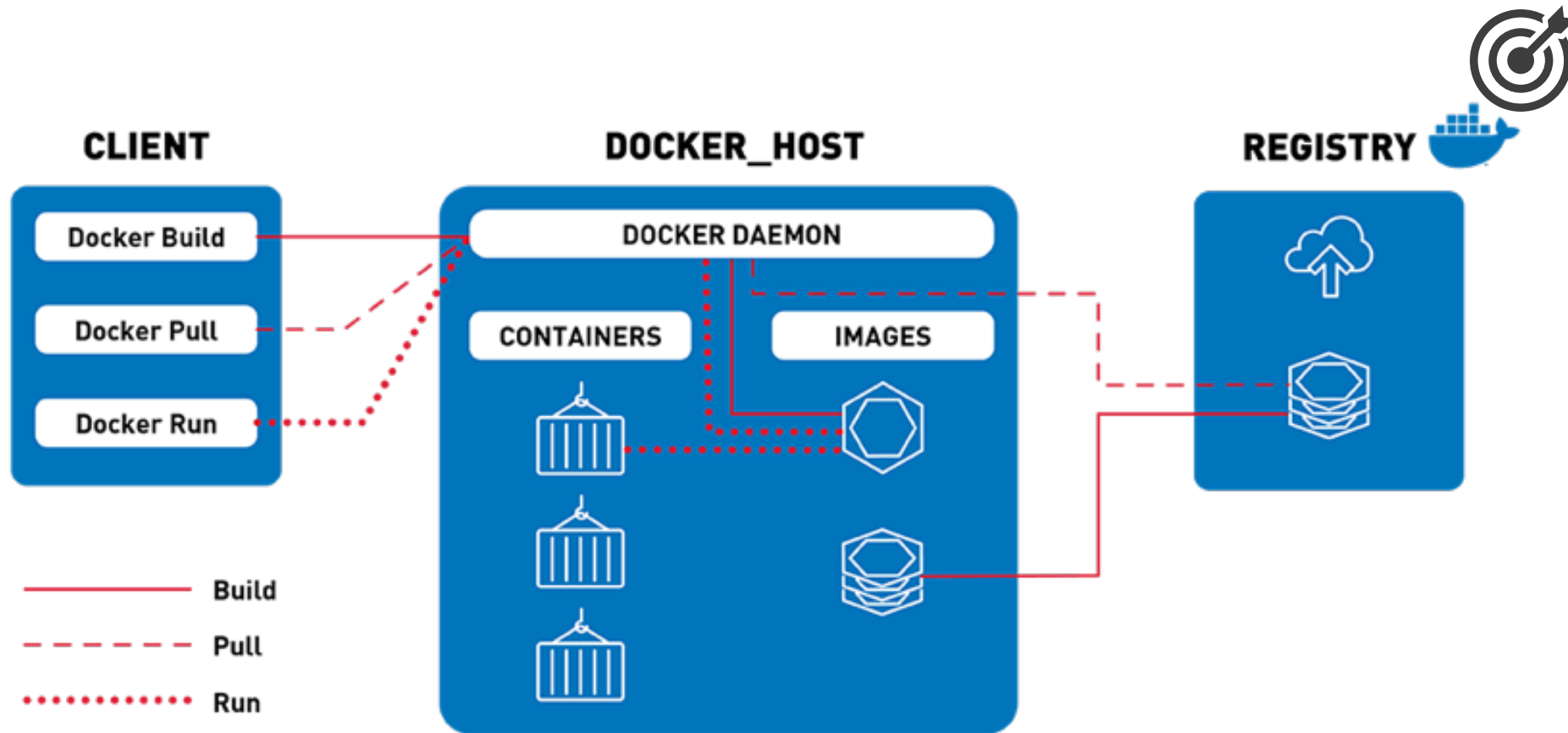
# Docker “AND” Kubernetes

Adapting Containerization & Orchestration



# Containerization – In Action

## Docker Lifecycle

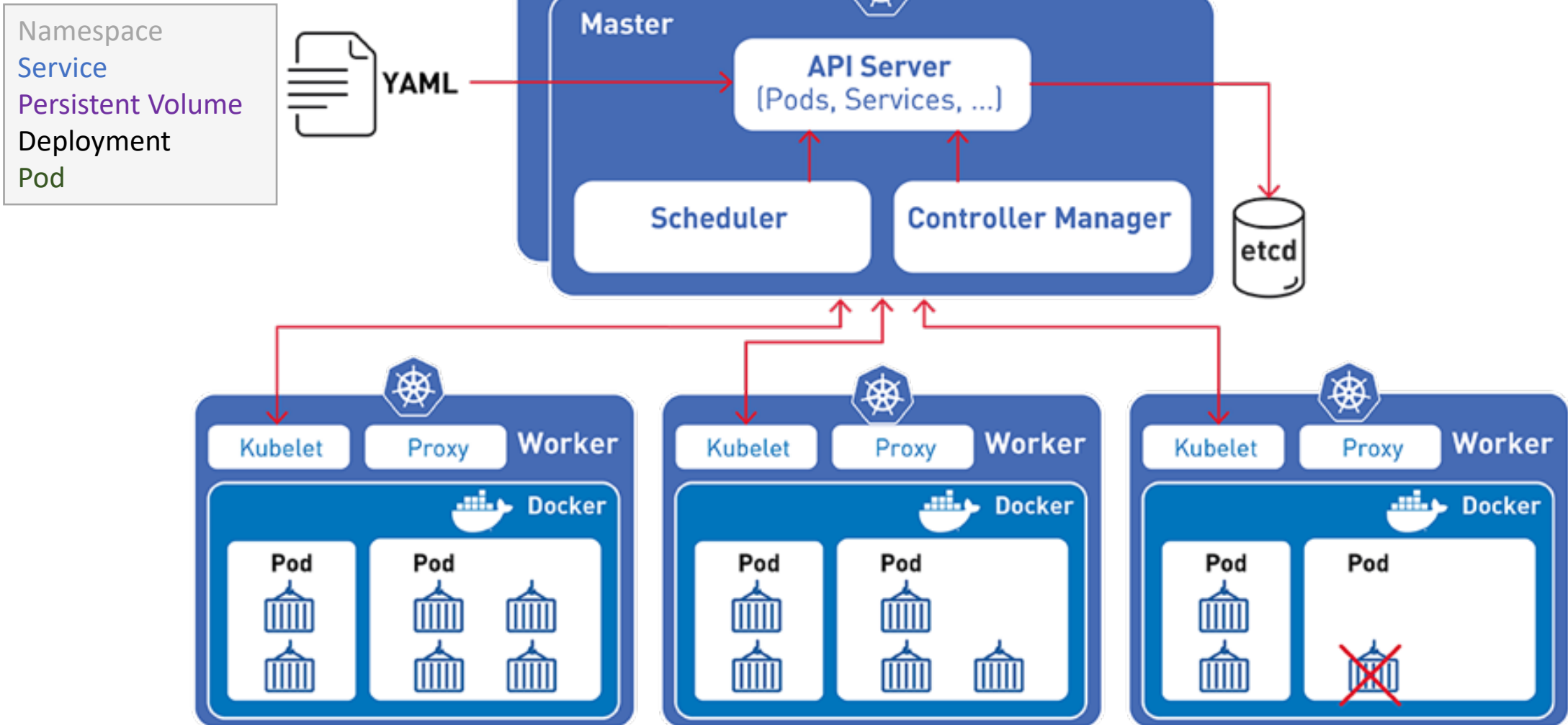


## Containerization Concepts

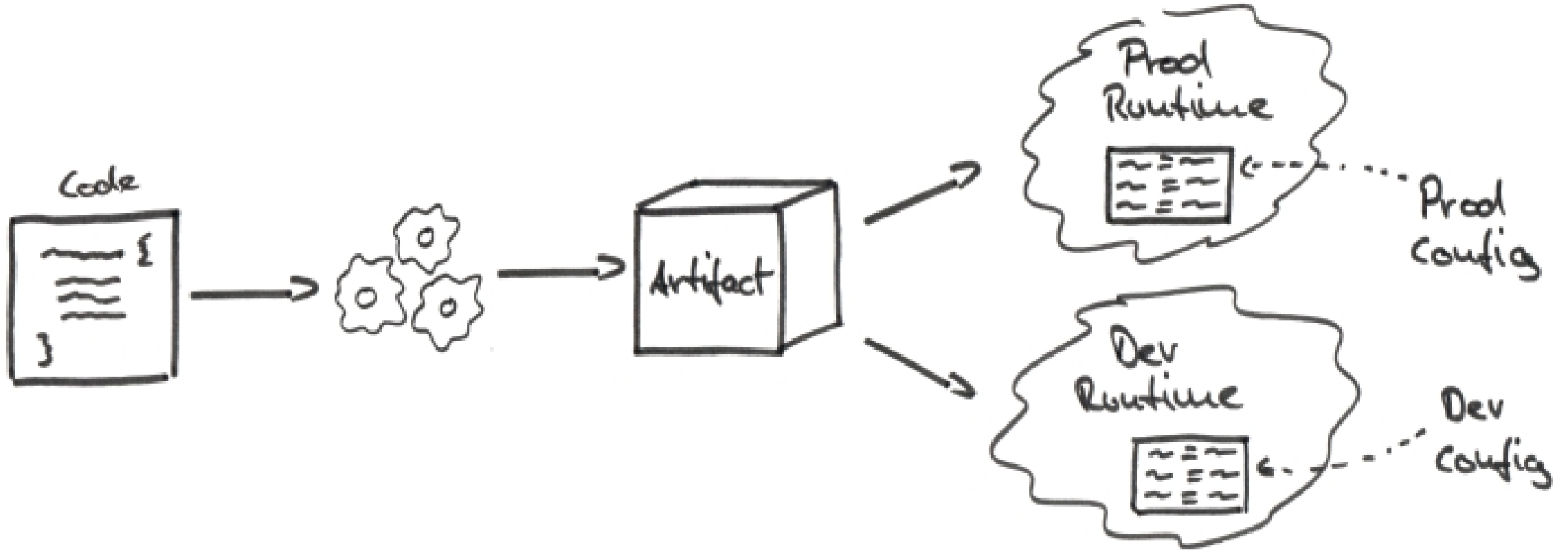
- Docker Daemon
- Docker Registry
- Docker Host
- Container
- Docker Image
  - Build
  - Pull/Push
  - Run

# Orchestration

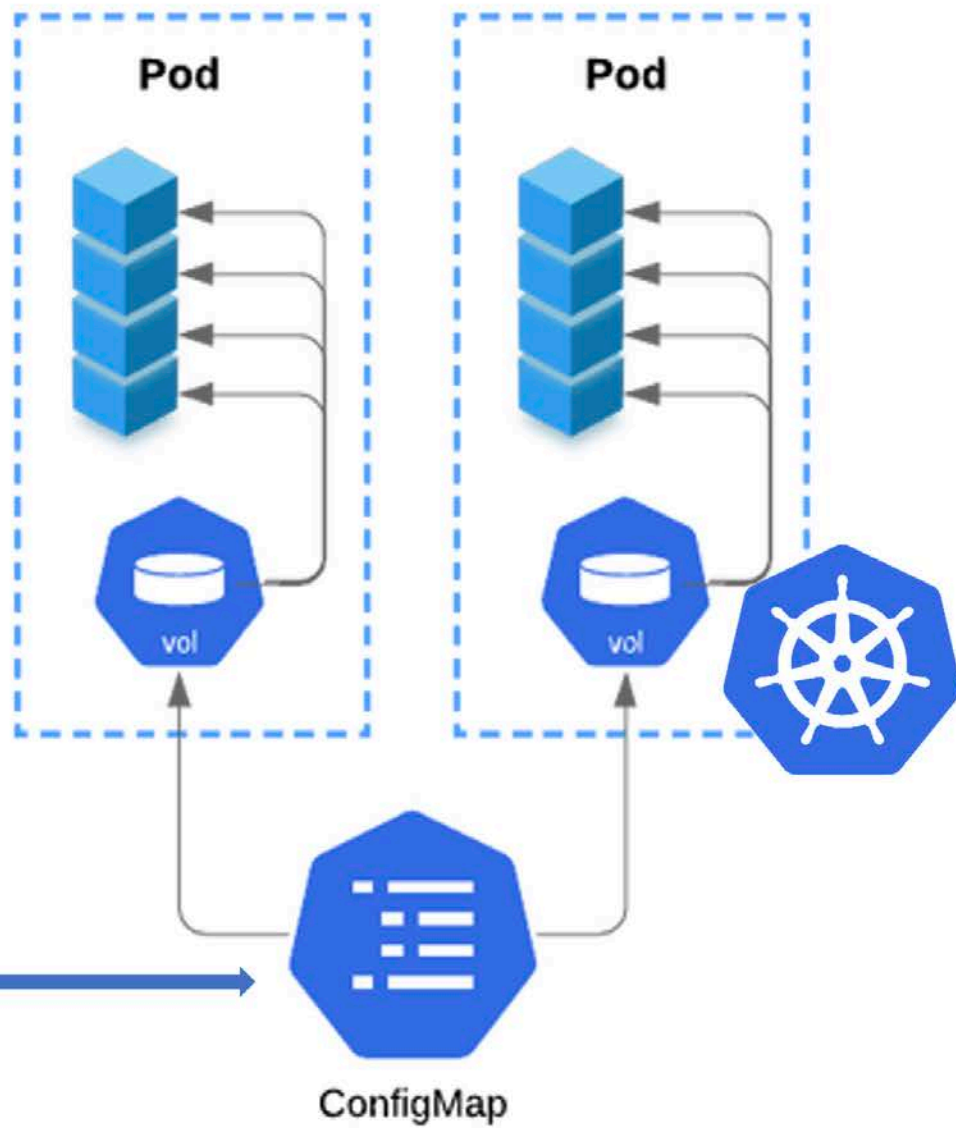
## Kubernetes Architecture

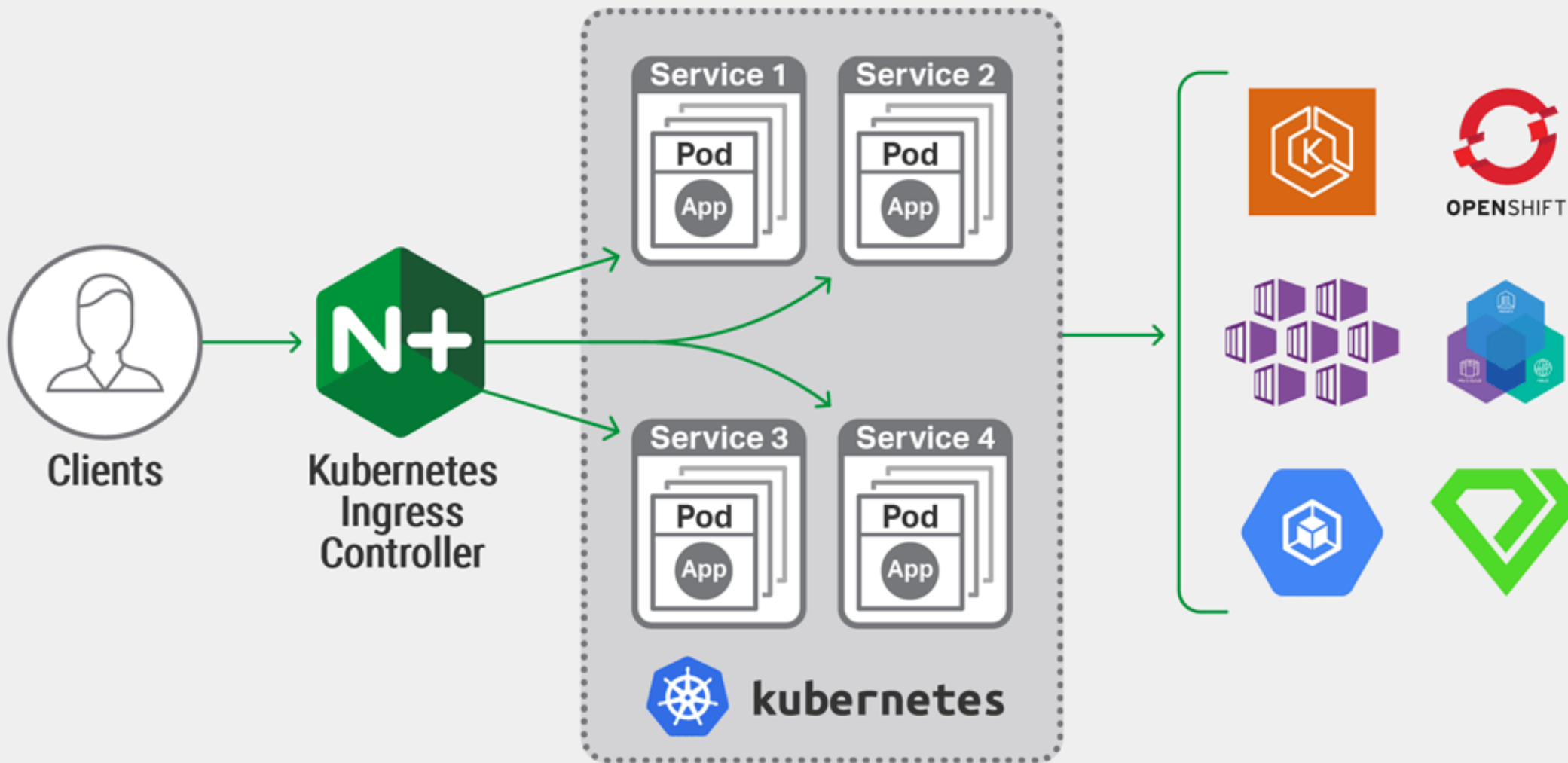


# BUILD ONCE, RUN ANYWHERE





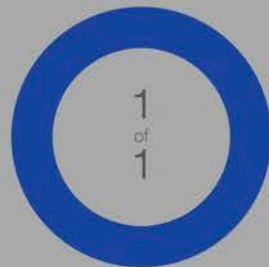






# Deployment • ewe-green • ewe-customers-deployment

SCALE EDIT DELETE



REPLICAS

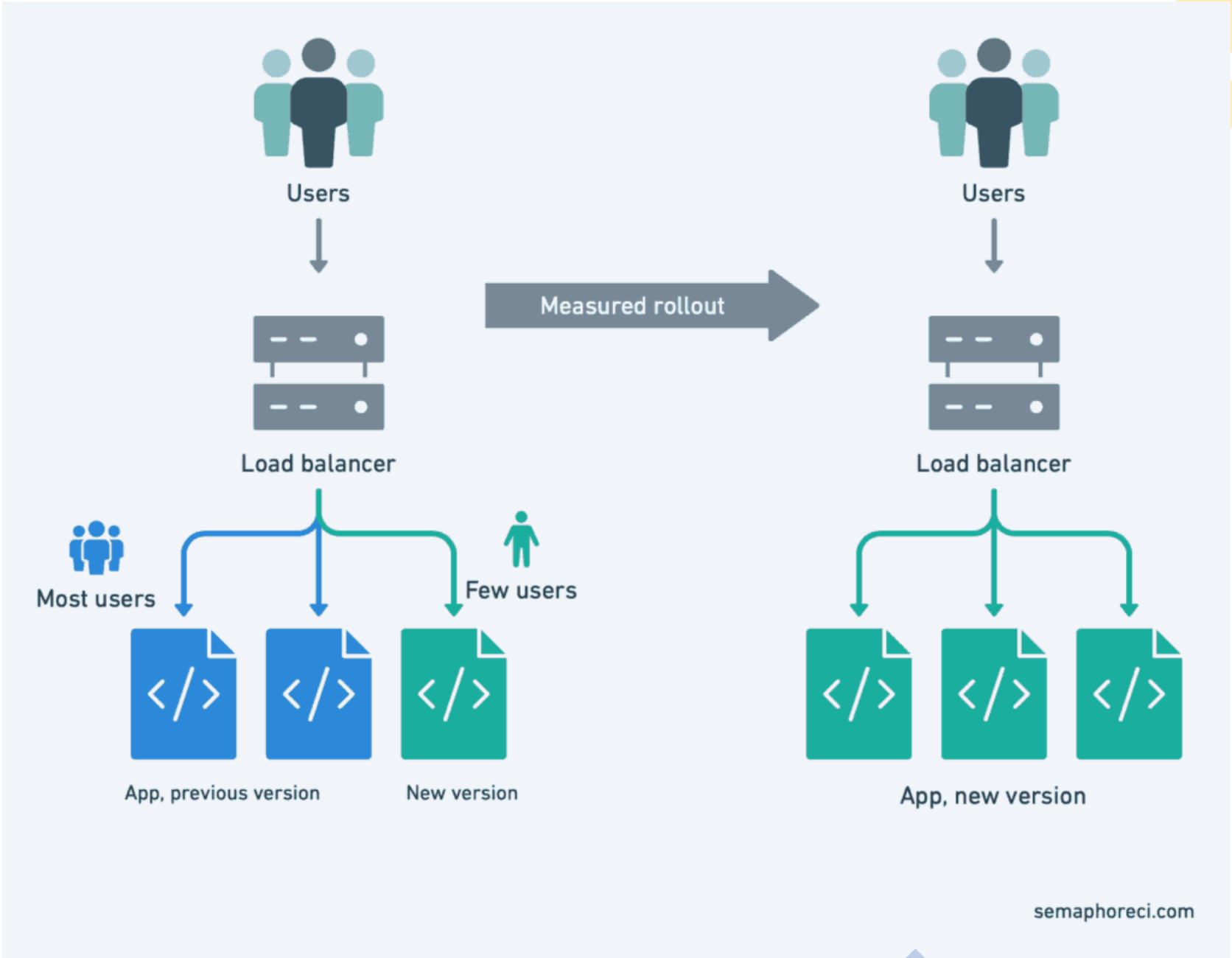


POD RAM USE  
ACTUAL VS RESERVED

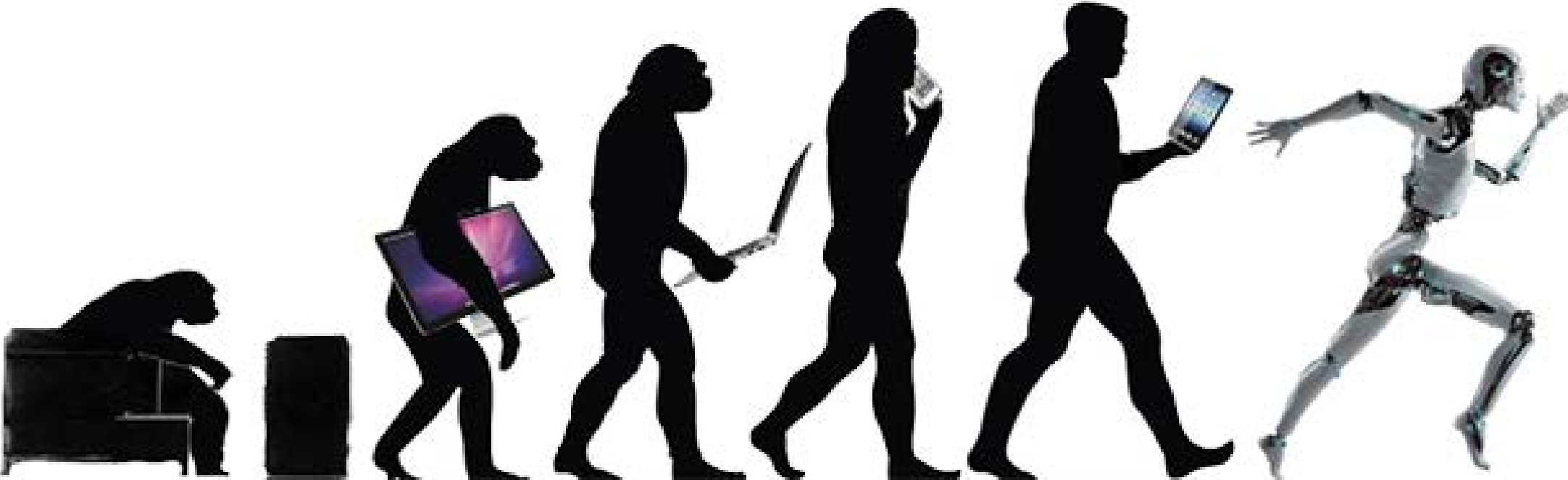
Desired Count

Scale Cancel

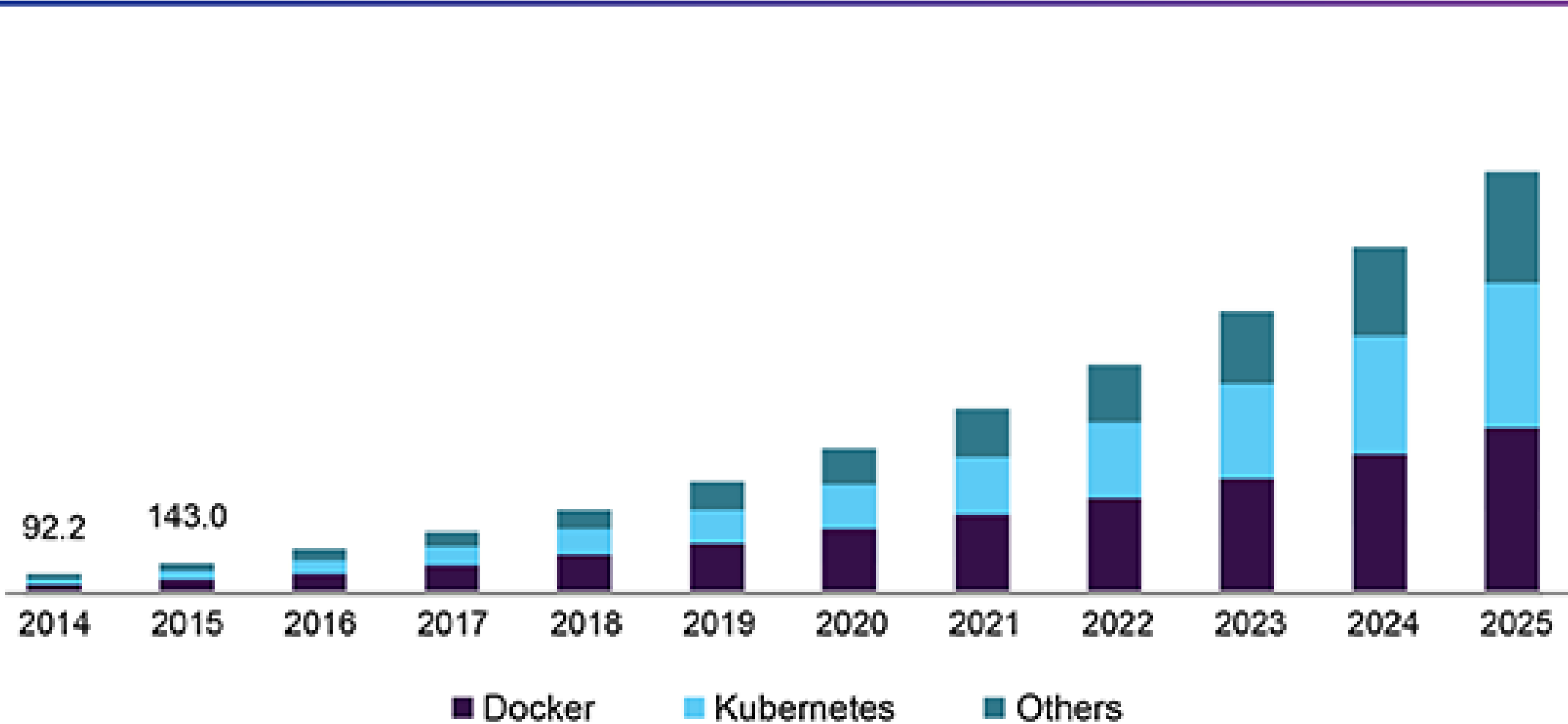
<b>Name</b>	ewe-customers-deployment
<b>Kind</b>	Deployment
<b>Namespace</b>	ewe-green
<b>Created</b>	4/17/2020, 2:17:13 PM
<b>Labels</b>	app • ewe-customers purpose • ewe-green-containers
<b>Annotations</b>	deployment.kubernetes.io/revision • 1



# Hyper-Risk ( Current Time )

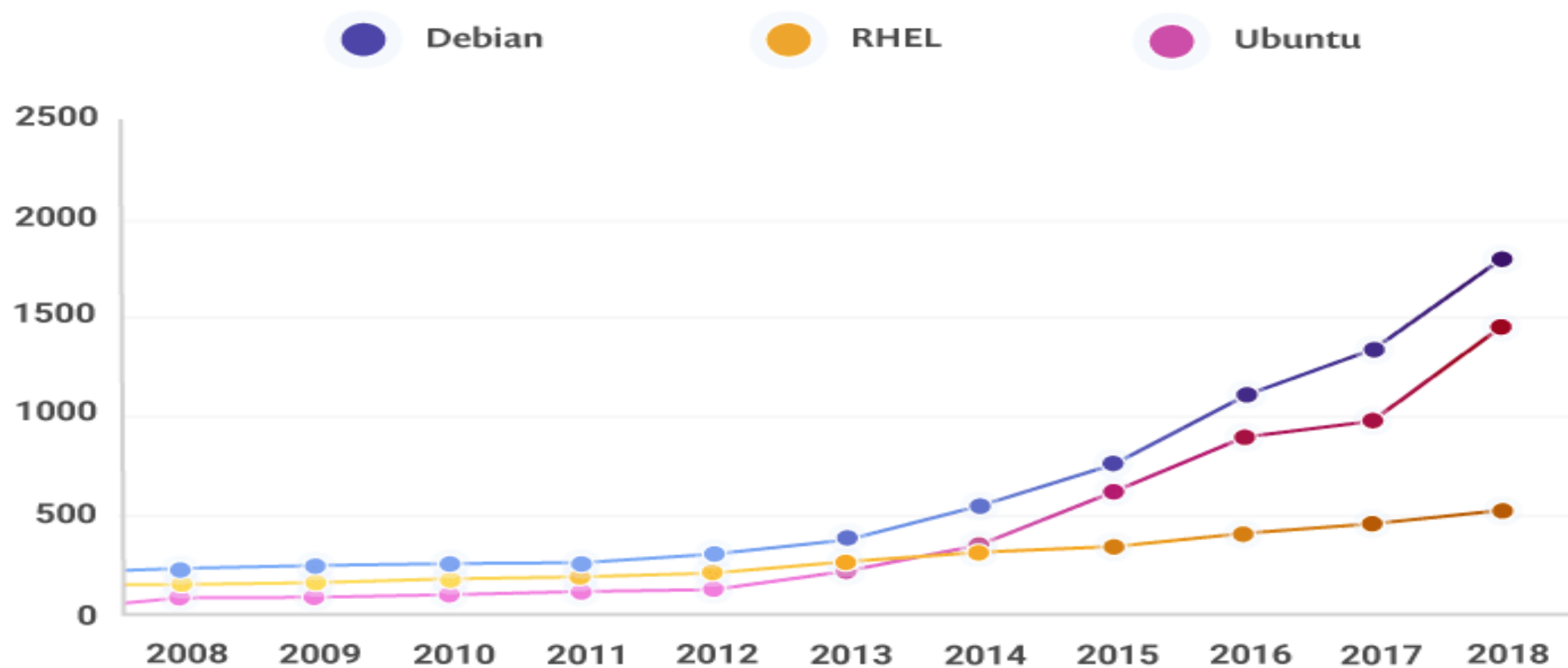


# U.S. application container market size, by platform, 2014 - 2025 (USD Million)

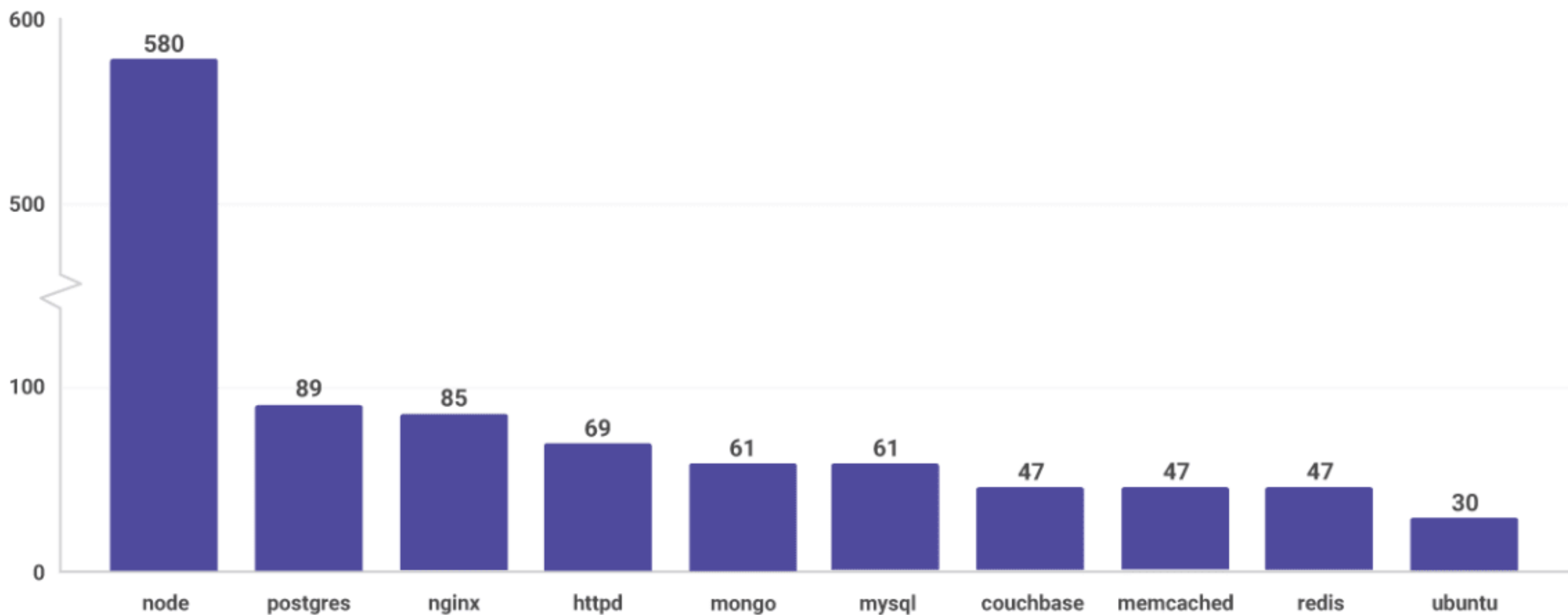


Source: [www.grandviewresearch.com](http://www.grandviewresearch.com)

# Linux OS vulnerabilities steadily increasing

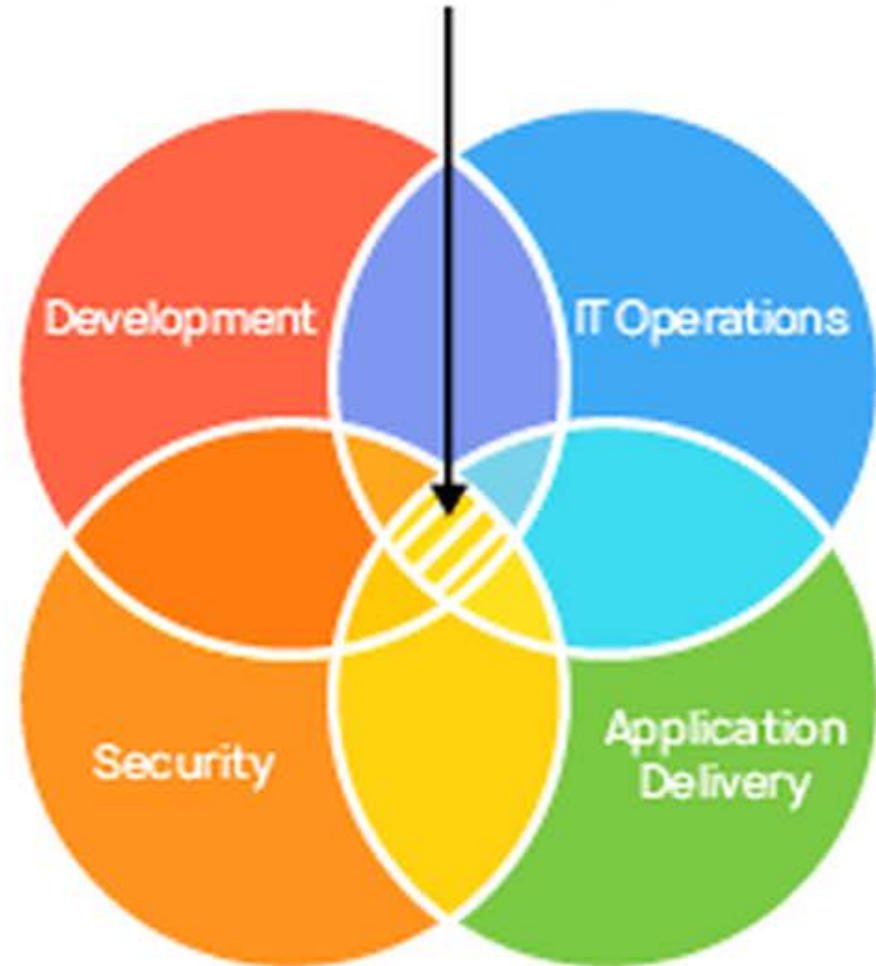
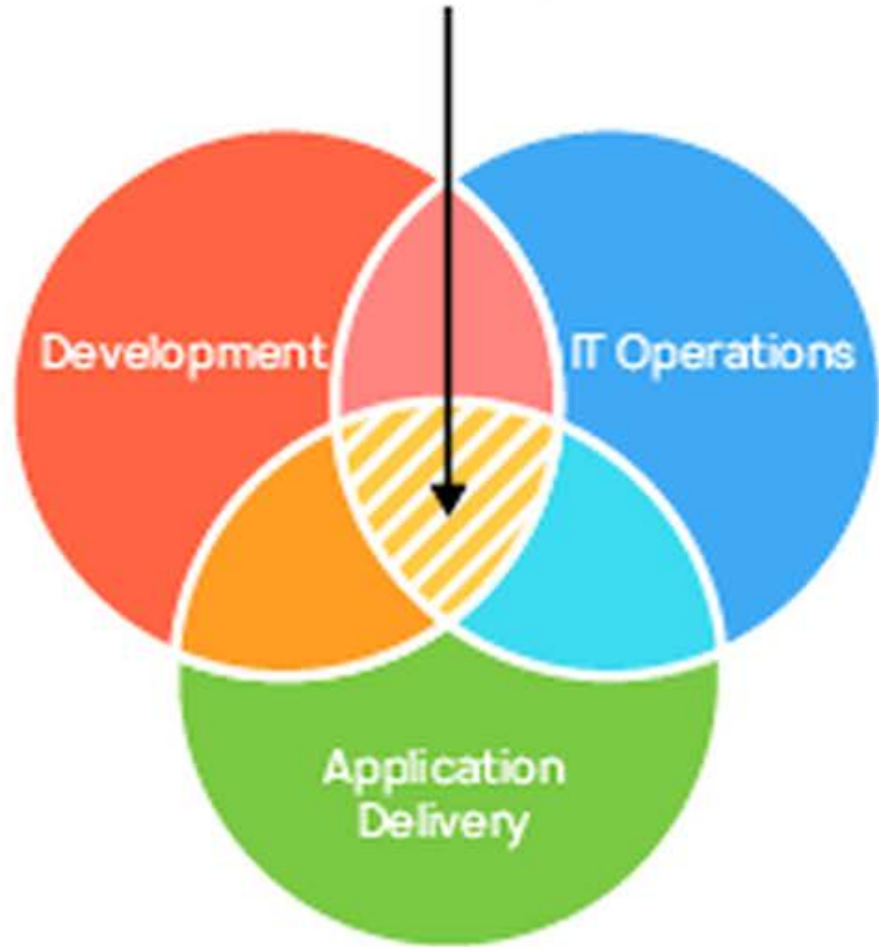


# Number of OS vulnerabilities by docker image





# DevOps **VS** DevSecOps



# ROOTLESS CONTAINERS

- LIGHTWEIGHT

- HIGH PERFORMANCE
- REDUCED RISK
- COST EFFECTIVENESS

- STATELESS

- CONSISTENCY
- QUICK STARTUP
- HIGH PERFORMANCE

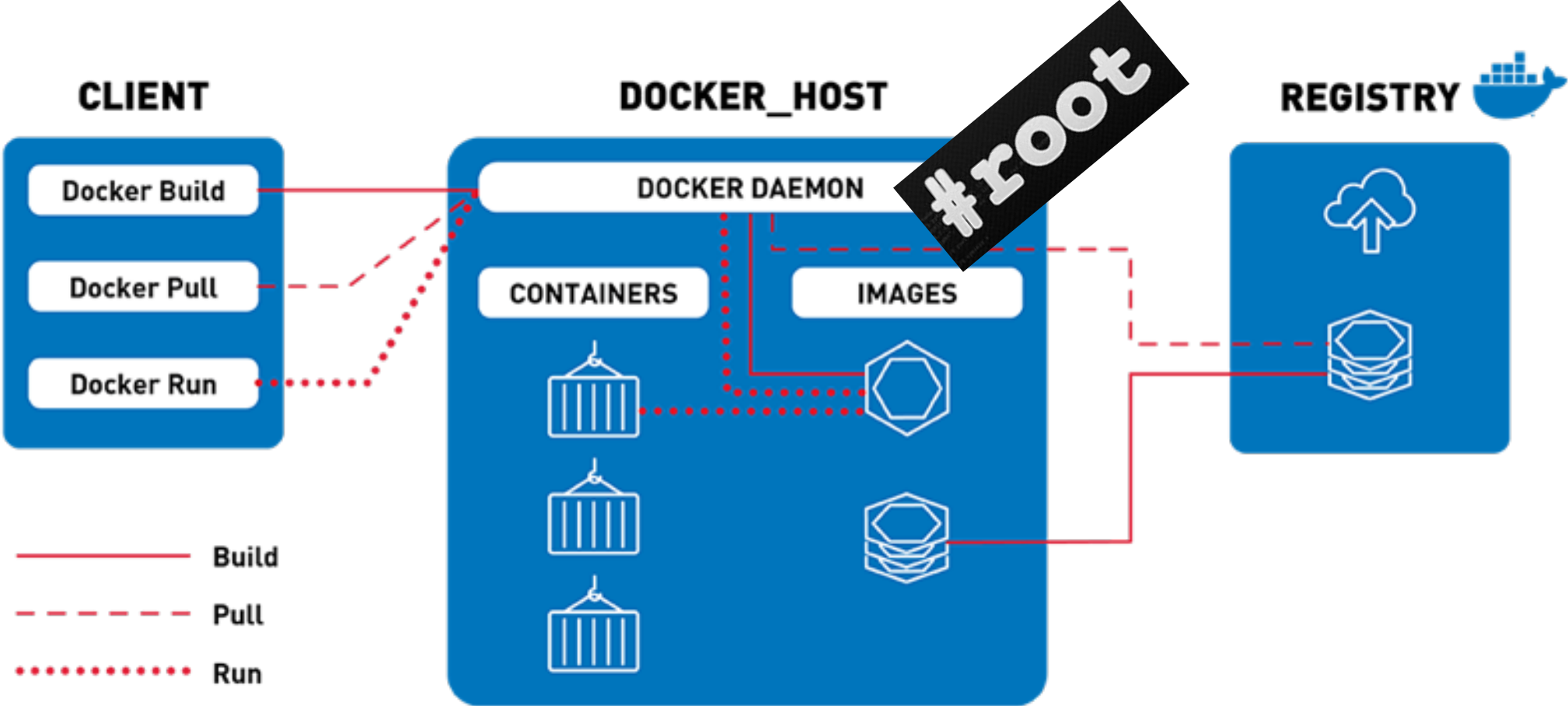
- IMMUTABLE

- STABILITY
- FEARLESS
- REDUCED RISK

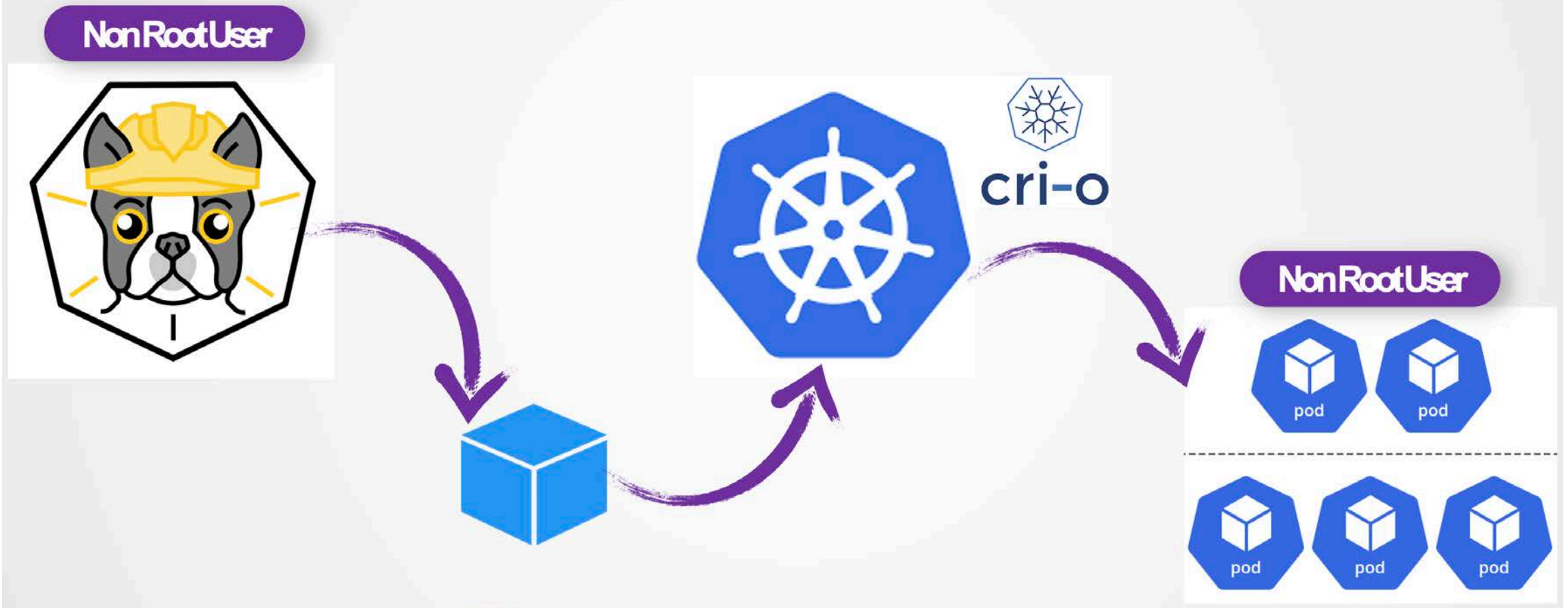
- ROOTLESS

- REDUCED RISK
- HIGH MAINTAINABILITY
- ASSETS TO ORG

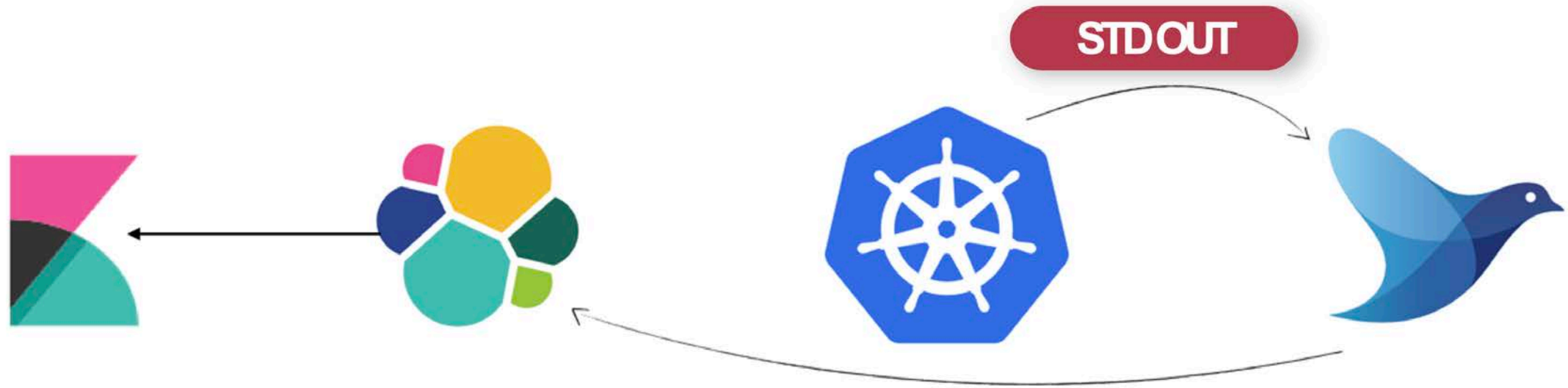
# Building Rootless Containers – Dockerless



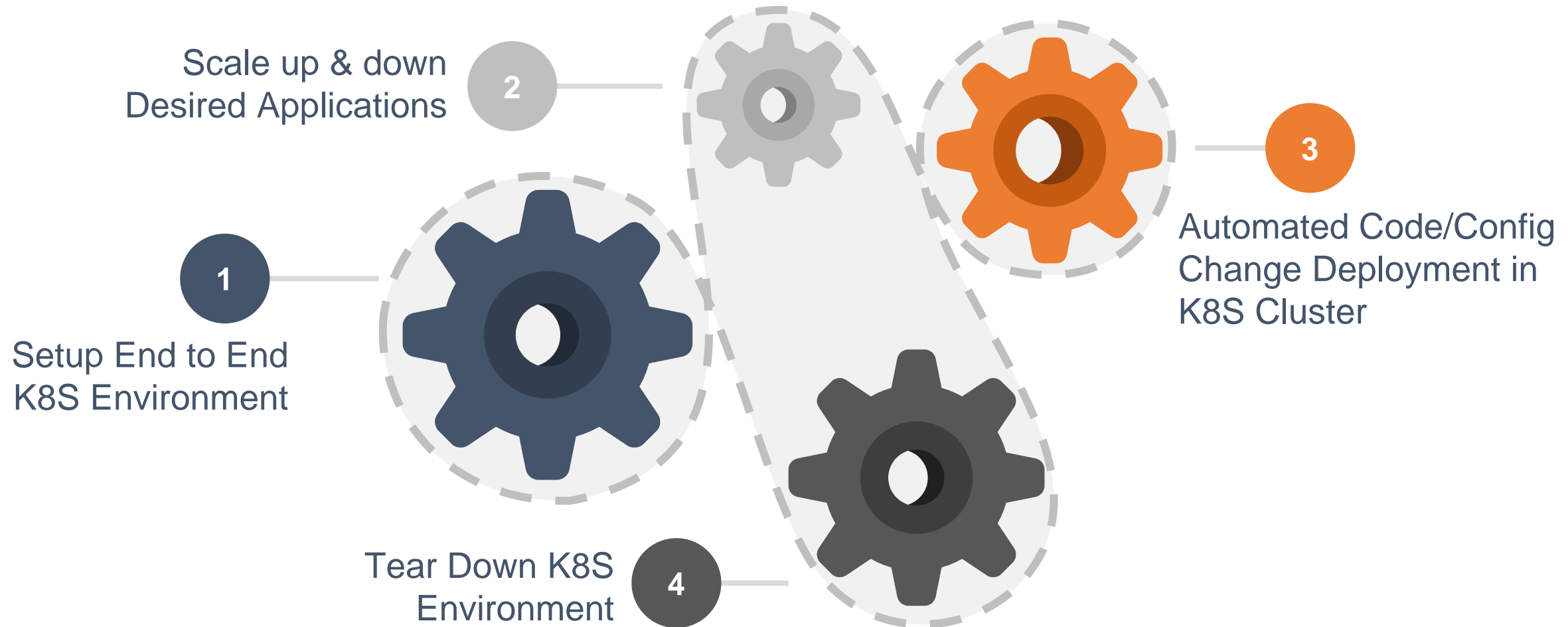
# Building Rootless Containers – Buildah & CRI-O



# EFK (ELASTIC SEARCH, FLUENTD, KIBANA)



# DEVOPS CI/CD AUTOMATION



# TEKTON & ARGO CD

The screenshot displays the Argo CD web interface. At the top left, there is a sidebar with a version indicator 'v1.6.1+1' and several navigation icons. The main header area includes the title 'Applications' and two buttons: '+ NEW APP' and 'SYNC APPS'. Below the header, there is a search bar and a filter sidebar on the left. The filter sidebar contains sections for 'FILTER BY:', 'SYNC', 'HEALTH', and 'LABELS'. The 'SYNC' section has three items: 'Synced' (2), 'Unknown' (0), and 'OutOfSync' (0). The 'HEALTH' section has six items: 'Healthy' (2), 'Unknown' (0), 'Progressing' (0), 'Suspended' (0), 'Degraded' (0), and 'Missing' (0). The 'LABELS' and 'PROJECTS' sections are currently empty. The main content area shows two application cards. The first card is for the 'blue' project, located in the 'ews-blue' namespace. It lists the following details: Project: ews-blue, Labels: (empty), Status: Healthy Synced (indicated by a green heart and checkmark), Repository: ssh://git@bitbucket.dal.securustech.net:7999/mid/..., Target Revis...: MW\_2.8.0, Path: cluster/, Destination: https://kubernetes.default.svc, and Namespace: ews-blue. Below the details are three buttons: SYNC, REFRESH, and DELETE. The second card is for the 'green' project, located in the 'ews-green' namespace. It lists the following details: Project: ews-green, Labels: (empty), Status: Healthy Synced (indicated by a green heart and checkmark), Repository: ssh://git@bitbucket.dal.securustech.net:7999/mid/..., Target Revis...: MW\_2.8.0, Path: cluster/, Destination: https://kubernetes.default.svc, and Namespace: ews-green. Below the details are three buttons: SYNC, REFRESH, and DELETE.

Applications

+ NEW APP SYNC APPS

SEARCH

FILTER BY:

SYNC

- Synced 2
- Unknown 0
- OutOfSync 0

HEALTH

- Healthy 2
- Unknown 0
- Progressing 0
- Suspended 0
- Degraded 0
- Missing 0

LABELS

PROJECTS

**blue**

Project: ews-blue

Labels:

Status: ♥ Healthy ✔ Synced

Repository: ssh://git@bitbucket.dal.securustech.net:7999/mid/...

Target Revis...: MW\_2.8.0

Path: cluster/

Destination: https://kubernetes.default.svc

Namespace: ews-blue

SYNC REFRESH DELETE

**green**

Project: ews-green

Labels:

Status: ♥ Healthy ✔ Synced

Repository: ssh://git@bitbucket.dal.securustech.net:7999/mid/...

Target Revis...: MW\_2.8.0

Path: cluster/

Destination: https://kubernetes.default.svc

Namespace: ews-green

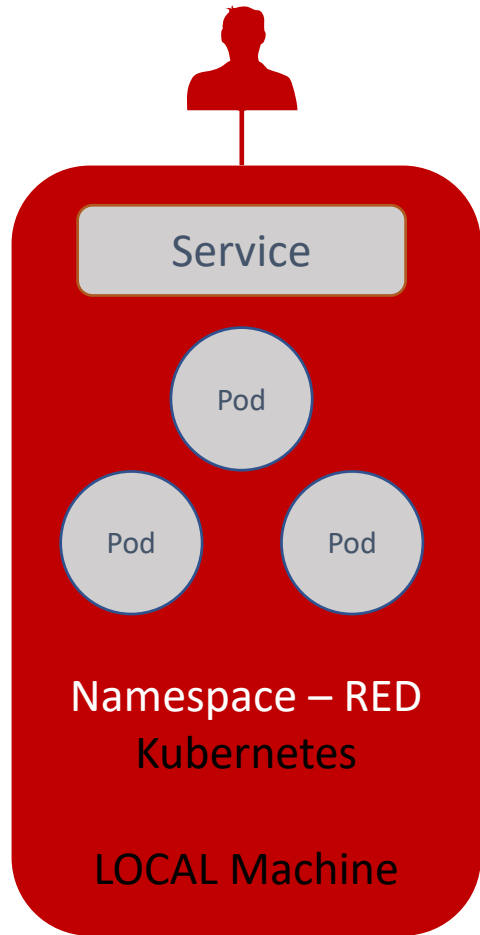
SYNC REFRESH DELETE

# ARGO CD – DETAILED VIEW

The screenshot displays the Argo CD interface for an application named 'blue'. The top navigation bar includes 'Applications / blue' and 'APPLICATION DETAILS'. Below this, there are several action buttons: 'APP DETAILS', 'APP DIFF', 'SYNC', 'SYNC STATUS', 'HISTORY AND ROLLBACK', 'DELETE', and 'REFRESH'. The application's overall status is 'Healthy', 'Synced', and 'Sync OK'. The 'Sync OK' status is detailed with a commit hash 'f26e832', a timestamp 'Succeeded 17 hours ago (Tue Jul 07 2020 16:45:41 GMT+0500)', and the author 'hpatel'. The main content area shows a hierarchical view of the application's resources. On the left, a cloud icon is connected to an IP address '10.6.247.195'. The resources are organized into three columns: 'esp-\*' services, 'esp-\*' deployments, and 'ews-\*' services and deployments. Each resource card shows its name, icon, status (green hearts), and a 'running 1/1' indicator. Red dashed lines connect the 'ews-ingress' service to the 'ews-admin', 'ews-customers', 'ews-inmateforms', 'ews-inmates', 'ews-notifications', 'ews-products', 'ews-rules', 'ews-timer', 'ews-visitors', and 'ews-visits' services. The interface also features a sidebar on the left with navigation icons and a top right corner with a 'Logout' button.



# SHIFT LEFT



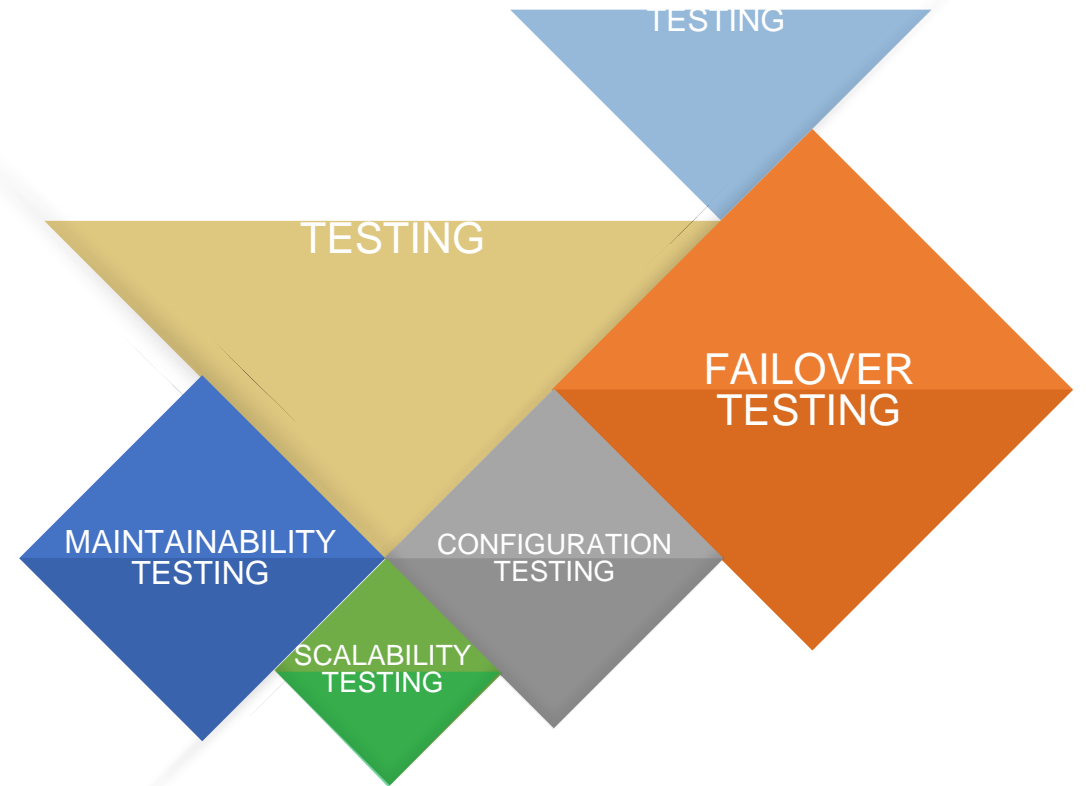
# kubernetes

MicroK8s

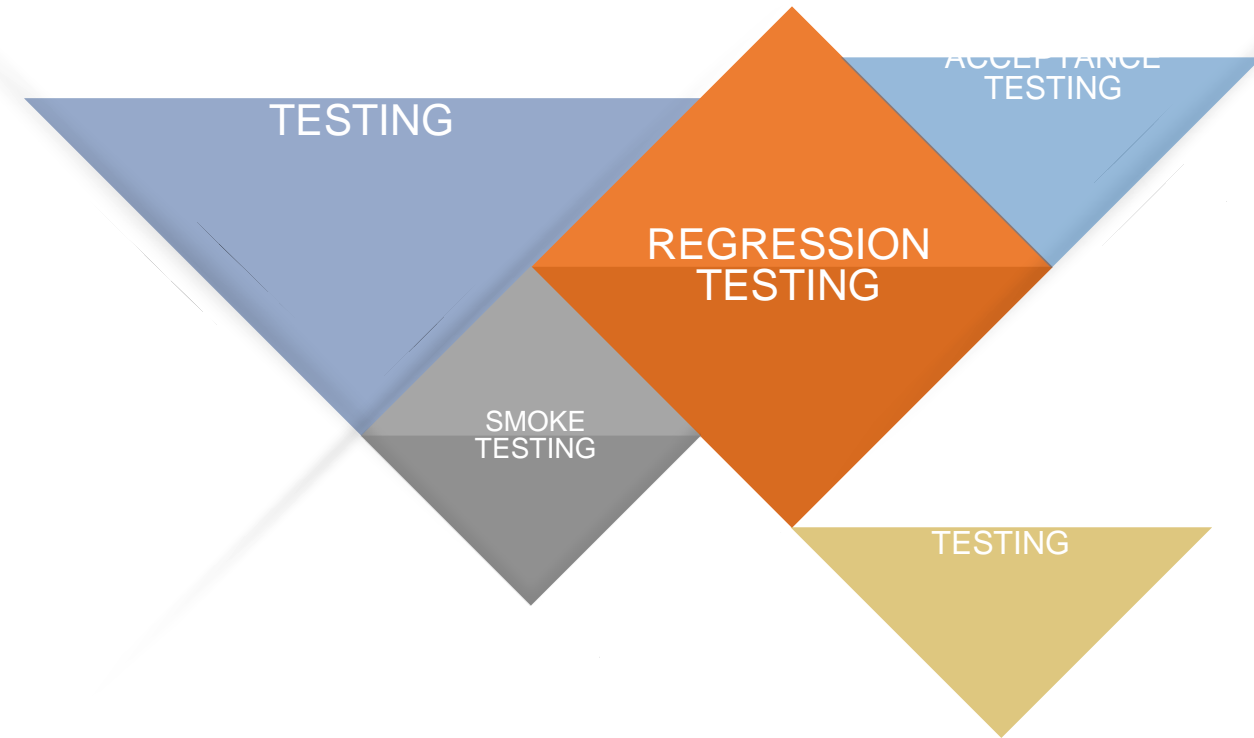


minikube

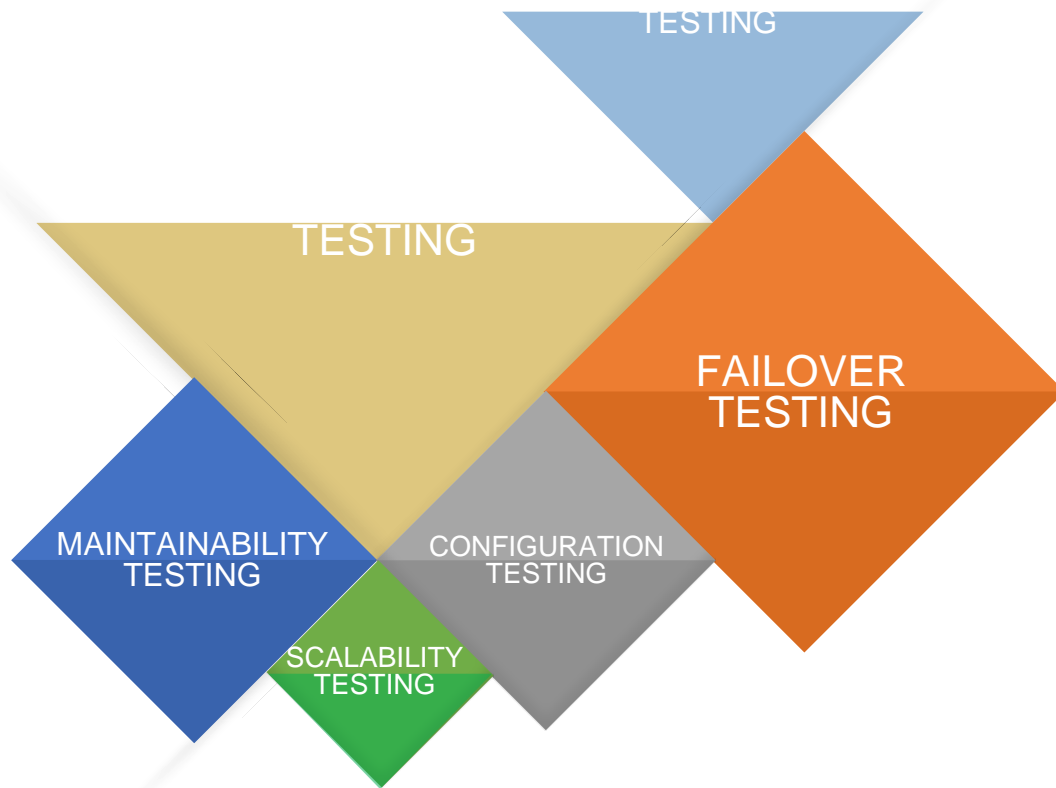
# NON-FUNCTIONAL TESTING



# FUNCTIONAL TESTING



# NON-FUNCTIONAL TESTING



Improved Security &  
Reduced Risk

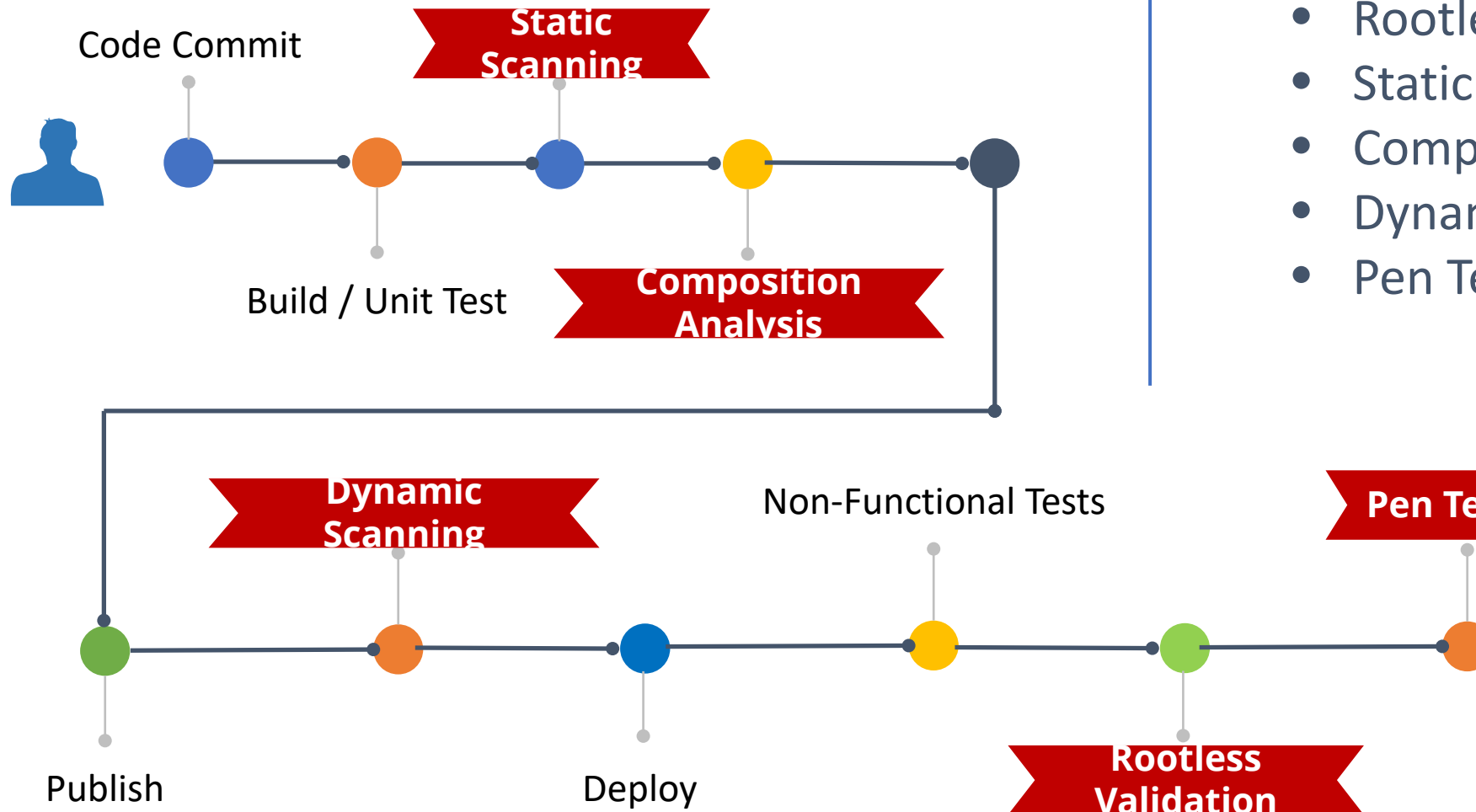


High Stability &  
Site Reliability



Cost & Time  
Savings

# SECURITY TESTING

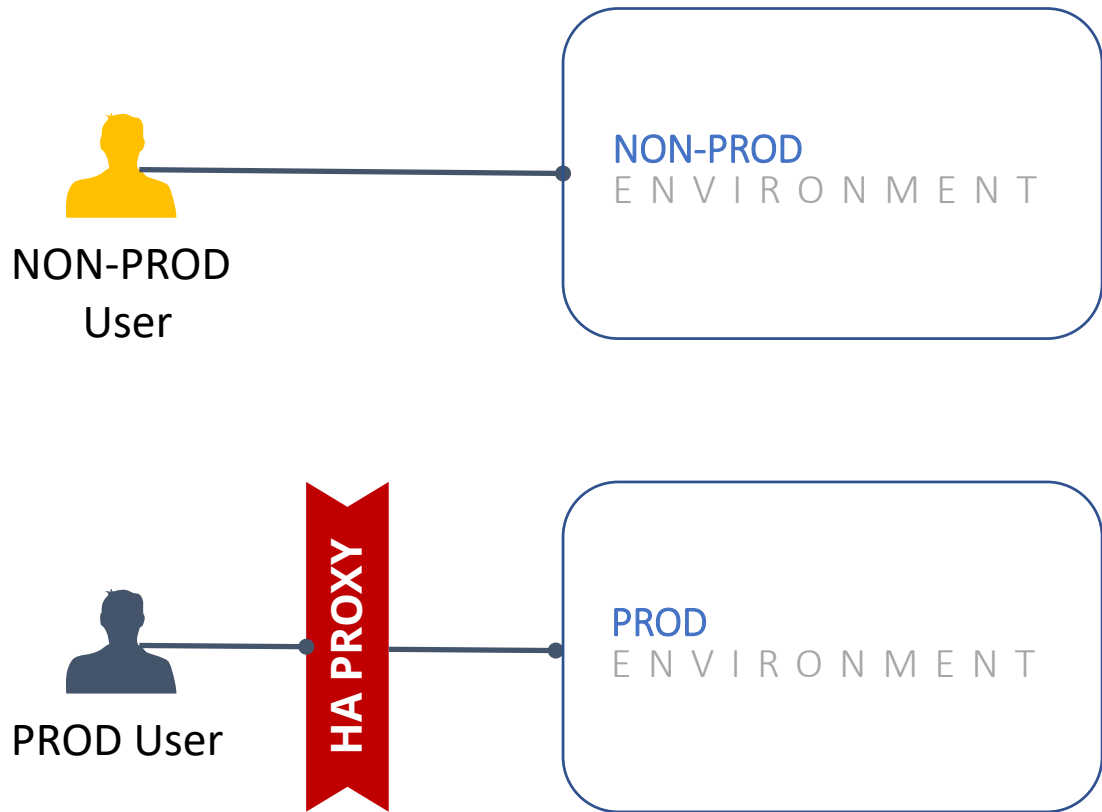


## ● Security Testing

- Rootless Containers
- Static Scanning
- Composition Analysis
- Dynamic Scanning
- Pen Testing

SECURITY  
AS PART OF  
CI/CD  
DELIVERY  
PIPELINE

# DEALING WITH FAILURES



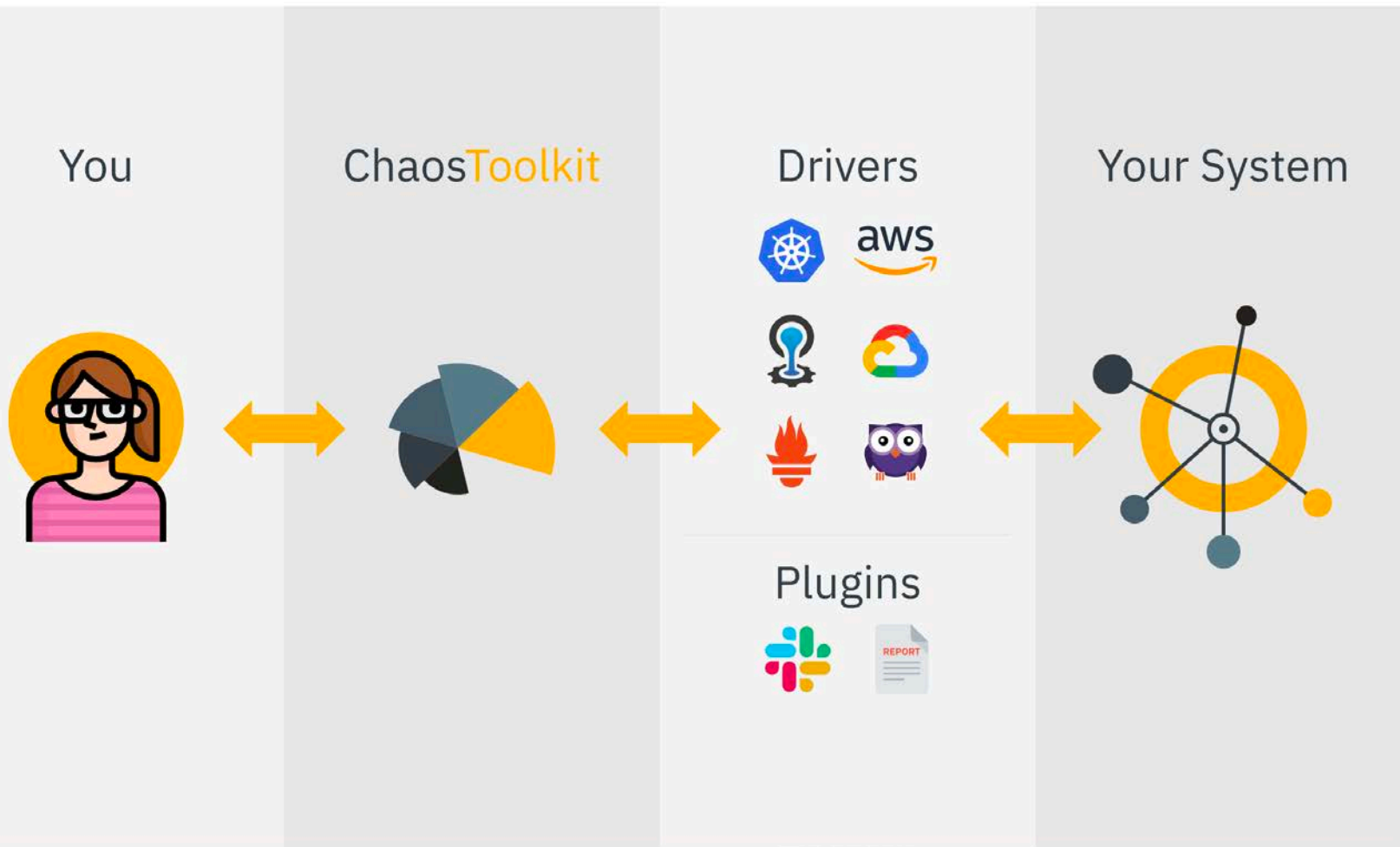
## ● Failover Testing

- Identification, Experimentation, Measurement & Remediation of Failures
- Identical Non-PROD & PROD Environments

## EPHEMERAL

ENVIRONMENTS  
BUILT WITH SAME  
PRODUCTION  
MANIFEST

# CHAOS ENGINEERING TOOLS



- Several Commercial & OSS Options
- Should be Declarative, Extensible & Automated
- Start Small & Build Confidence
- Experiment across Levels,
  - Application
  - Caching
  - Database
  - Network

**CHAOS ENGINEERING**  
AS PART OF  
CI/CD  
DELIVERY  
PIPELINE



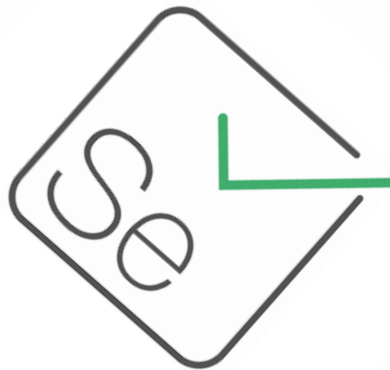
VERACODE



buildah



argo

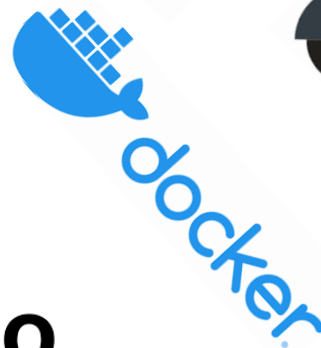


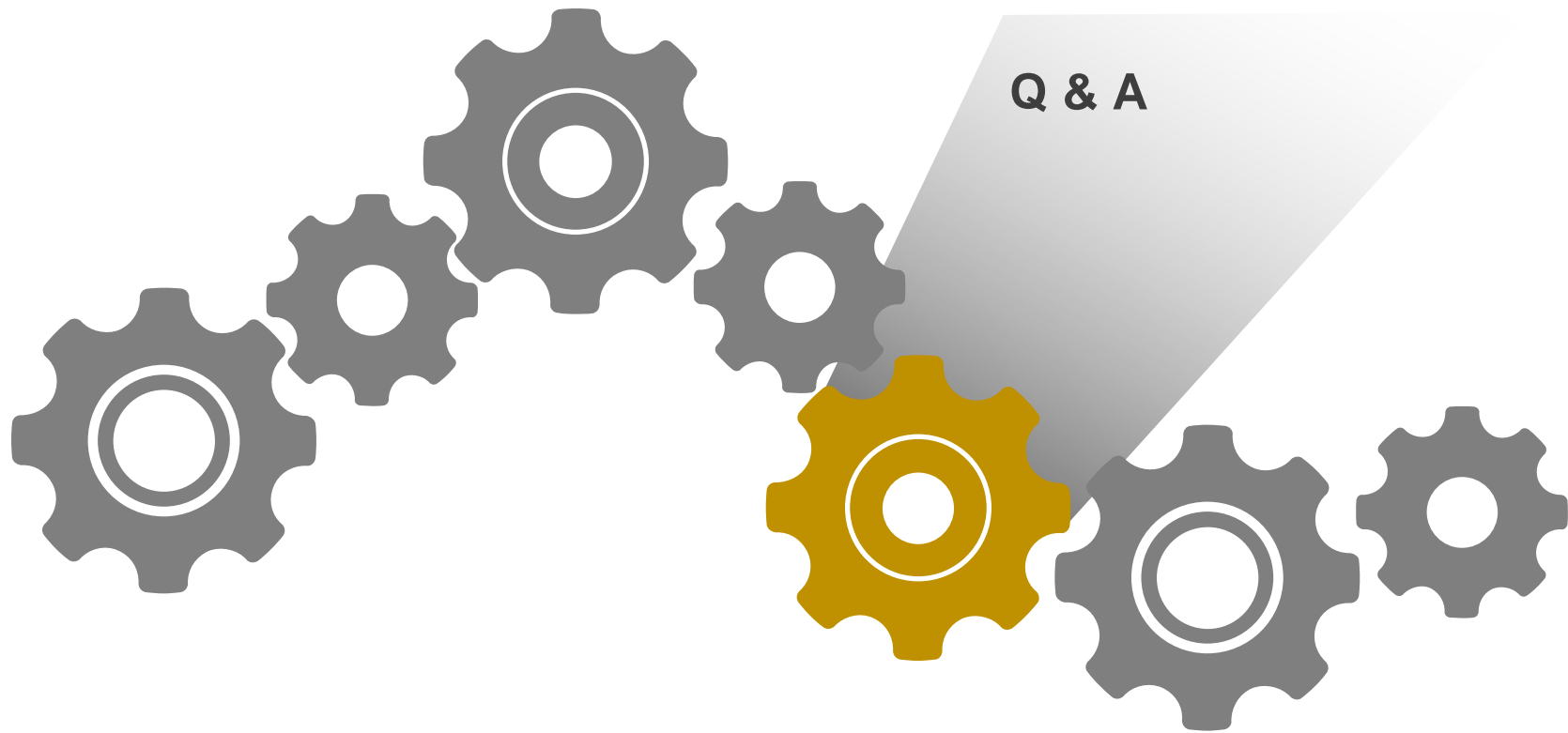
cri-o

NGINX



sonarqube





Q & A



# Thank You!



Himanshu Patel



HRPatel2000



NavikCo



@HubNavik