End-to-End Kubernetes

Moshe Zadka – https://cobordism.com

Acknowledgement of Country

Belmont (in San Francisco Bay Area Peninsula)
Ancestral homeland of the Ramaytush Ohlone

Examples as Caricatures

Realistic exaggeration

A Caricature

```
# e2e_k8s/demo.py
from pyramid import response, config, view
@view.view_config(route_name='add')
def add(request):
    x, y = (int(request.matchdict[c]) for c in "xy"
    return response.Response(str(x+y))
with config.Configurator() as _cfg:
    _cfg.add_route("add", "/add/{x}/{y}")
    _cfg.scan("e2e_k8s")
    application = _cfg.make_wsgi_app()
```

Software as a Pipeline

Code → Customer value

Leaky Pipeline

Leak:

Leaky Pipeline

Leak:
Problem happening at one stage but not the previous one

Pipeline Caricature

Code

Pipeline Caricature

Code
Unit tests

Pipeline Caricature

Code
Unit tests
Running locally

Pipeline Caricature

Code
Unit tests
Running locally
"Simulation" environments

Pipeline Caricature

Code
Unit tests
Running locally
"Simulation" environments
Production

Pipeline Caricature

Code
Unit tests
Running locally
"Simulation" environments
Production
Customer

Fixing Leaks

Less differences – less leaks

Model, Not Solutions

Hard problem to solve

Model, Not Solutions

Hard problem to solve
Better mental model for local solutions

Kubernetes

Orchestration: Running containers on compute resources

Why Kubernetes?

- Builds on containers
- Documented REST API
- Extensible

What is DevOps (Not?

an engineering speciality

What is DevOps (Not?

an engineering speciality

"developers should run their code"

What is DevOps (Not?

an engineering speciality

"developers should run their code"

"no need for operations"

What is DevOps (Not?

an engineering speciality

"developers should run their code"

"no need for operations"

"developers working for ops"

What is DevOps (Not?

an engineering speciality

”developers should run their code”

”no need for operations”

”developers working for ops”

”ops working for developers”

What is DevOps (Not?

an engineering speciality

”developers should run their code”

”no need for operations”

”developers working for ops”

”ops working for developers”

What is DevOps?

Developers, ops (and QA, and security, ...) collaborating on a pipeline

How and Why DevOps

Reduce friction

How and Why DevOps

Reduce friction
Not just technical solution

How and Why DevOps

Reduce friction
Not just technical solution
Part of technical part: reduce differences between developer environment and production environment

Environment

A collection of services which work together

Environment Cross-Talk

Environments (mostly) don’t cross-talk

Regional Environments

Jursidictional/Geographical

Maturity-Based Environments

Production

Maturity-Based Environments

Production
Stagingin, Testing....

Maturity-Based Environments

Production

Stagingin, Testing....

Ad-hoc Remote

Maturity-Based Environments

Production
Stagingin, Testing....
Ad-hoc Remote
Local

Software Development Lifecycle (Caricature)

Develop

Software Development Lifecycle (Caricature)

Develop
Review

Software Development Lifecycle (Caricature)

Develop
Review
Merge

Software Development Lifecycle (Caricature)

Develop
Review
Merge
Deploy

SDLC: Develop

Write code

SDLC: Review

Approve/Reject

SDLC: Merge

Integrate code into rest of product

SDLC: Deploy

Run in production

SDLC: No Test?

Every stage contains testing:

SDLC: No Test?

Every stage contains testing:
Develop: unit/ad-hoc/local

SDLC: No Test?

Every stage contains testing:
Develop: unit/ad-hoc/local
Review: Continuous Integration testing

SDLC: No Test?

Every stage contains testing:
Develop: unit/ad-hoc/local
Review: Continuous Integration testing
Merge: Continuous Integration testing

SDLC: No Test?

Every stage contains testing:
Develop: unit/ad-hoc/local
Review: Continuous Integration testing
Merge: Continuous Integration testing
Deploy: Monitoring and alerting

K8s across Environments

Environments are like k8s clusters

K8s across Environments

Environments are like k8s clusters
Can they be the same?

Clusters as Environments: Why

Production will be k8s probably

Clusters as Environments: Why

Production will be k8s probably
Less leaks!

Clusters as Environments: How

Different sizes

Clusters as Environments: How

Different sizes
Different platforms

Clusters as Environments: How

Different sizes
Different platforms
Different versions

Local environments: Lima

Run a VM running containers on Mac

Local environments: WSL2

Run a VM (that can runn containers) on Windows

Local environments: Minikube

Single-host kubernetes

Local environments: Minikube

Single-host kubernetes
Lima, WSL2, Local linux, Remote VM....

Remote environments: Cloud native

Part of cloud offerings!

Customizing Kubernetes

One size

Customizing Kubernetes

One size
literally

Customizing Kubernetes

One size
literally
does not fit all environments.

Customizing Kubernetes: Ad-hoc

Never underestimate a programmer with sed

Customizing Kubernetes: Template

Use a generic templating language (e.g., Jinja2)

Customizing Kubernetes: Kustomize

Part of kubectl

Customizing Kubernetes: Kustomize

Part of kubectl
YAML-based YAML-editing DSL

Customizing Kubernetes: Helm

Specialized templating system

Customizing Kubernetes: Server-Side Apply

Combine YAML from different sources

Customizing Kubernetes: Operators

Convert "abstract" description to "concrete" description

Kubernetes Architecture Caricature

(Say that three times fast!)

Containers

"Light weight VMs"

Containers

”Light weight VMs”
”Heavy weight processes”

Pods

Groups of containers

Pods

Groups of containers
Share network namespace

Pods

Groups of containers
Share network namespace
Can share process namespace

Pods

Groups of containers
Share network namespace
Can share process namespace
Can share ephemeral storage

Pods

Groups of containers
Share network namespace
Can share process namespace
Can share ephemeral storage
Can share durable storage

Deployment

Routable set of identity-less pods

StatefulSet

Set of identifiable pods

Kubernetes routing: Pods

Unique IP

Kubernetes routing: Pods

Unique IP
DNS depends on name

Kubernetes routing: Service

Select ”participating” pods

Kubernetes routing: Service

Select ”participating” pods
Regular: route (usually TCP) to Pods

Kubernetes routing: Service

Select "participating" pods
Regular: route (usually TCP) to Pods
Headless: DNS to Pods

Kubernetes routing: StatefulSet

With Service

Kubernetes routing: StatefulSet

With Service
Route to ”name-number”.suffix

Kubernetes routing: StatefulSet

With Service
Route to ”name-number”.suffix

Kubernetes-based Development: Basic

Build new container

Kubernetes-based Development: Basic

Build new container
Configure k8s with new contaier

Kubernetes-based Development: Basic

Build new container
Configure k8s with new contaier
Repeat

Kubernetes-based Development: Basic is Slow

Rebuild container (even with cache)

Kubernetes-based Development: Basic is Slow

Rebuild container (even with cache)
Redownload container

Kubernetes-based Development: Basic is Slow

Rebuild container (even with cache)
Redownload container
Restart container

Kubernetes-based Development: Update Containers In-place

Quicker

Kubernetes-based Development: Update Containers In-place

Quicker
less accurate

Kubernetes-based Development: Update Containers In-place

Quicker
less accurate
feedback

Kubernetes-based Development: SSH to Pod

With customization

Kubernetes-based Development: SSH to Pod

With customization
Add container to pod

Kubernetes-based Development: SSH to Pod

With customization
Add container to pod
Running ssh server

Kubernetes-based Development: Cross-Pod file-access

Share process namespace

Kubernetes-based Development: Cross-Pod file-access

Share process namespace
Use proc filesystem

Kubernetes-based Development: Cross-Pod file-access

Share process namespace
Use proc filesystem
SSH pod can modify files in pod-friend

Kubernetes-based Development: Sync files

Continuous sync

Kubernetes-based Development: Sync files

Continuous sync
Over SSH

Kubernetes-based Development: Sync files

Continuous sync
Over SSH
Over Pod

Kubernetes-based Development: Auto-restarting

Use customization

Kubernetes-based Development: Auto-restarting

Use customization
watchmedo and friends

Kubernetes Oriented Development: Build Images

Build container using rootless daemonless buildkit

Kubernetes Oriented Development: Build Images

Build container using rootless daemonless buildkit
Push to registry

Kubernetes Oriented Development: Local Integration

Unit tests, lint

Kubernetes Oriented Development: Local Integration

Unit tests, lint

Run in ”development” container

End-to-End Kubernetes?

Put together the pieces

End-to-End Kubernetes Recap

End-to-End Kubernetes Recap

Build environments

End-to-End Kubernetes Recap

Build environments
Implement customization

End-to-End Kubernetes Recap

Build environments
Implement customization
Build container images

End-to-End Kubernetes Recap

Build environments
Implement customization
Build container images
Set up dynamic container updates

End-to-End Kubernetes Recap

Build environments
Implement customization
Build container images
Set up dynamic container updates
Set up local testing

End-to-End Kubernetes Recap

Build environments
Implement customization
Build container images
Set up dynamic container updates
Set up local testing
Enjoy!

Kubernetes End-to-End: Benefits

Kubernetes End-to-End: Benefits

Develop,

Kubernetes End-to-End: Benefits

Develop, Review,

Kubernetes End-to-End: Benefits

Develop, Review, Merge,

Kubernetes End-to-End: Benefits

Develop, Review, Merge, and Deploy

Kubernetes End-to-End: Benefits

Develop, Review, Merge, and Deploy
with synchronized Kubernetes configuration!

Kubernetes End-to-End: Start to Finish

It can run everywhere!

Kubernetes End-to-End: Start to Finish

It can run everywhere!
Less leakage in the SDLC pipe.