Move your development environments to Kubernetes

RAMIRO BERRELLEZA | @RBERRELLEZA

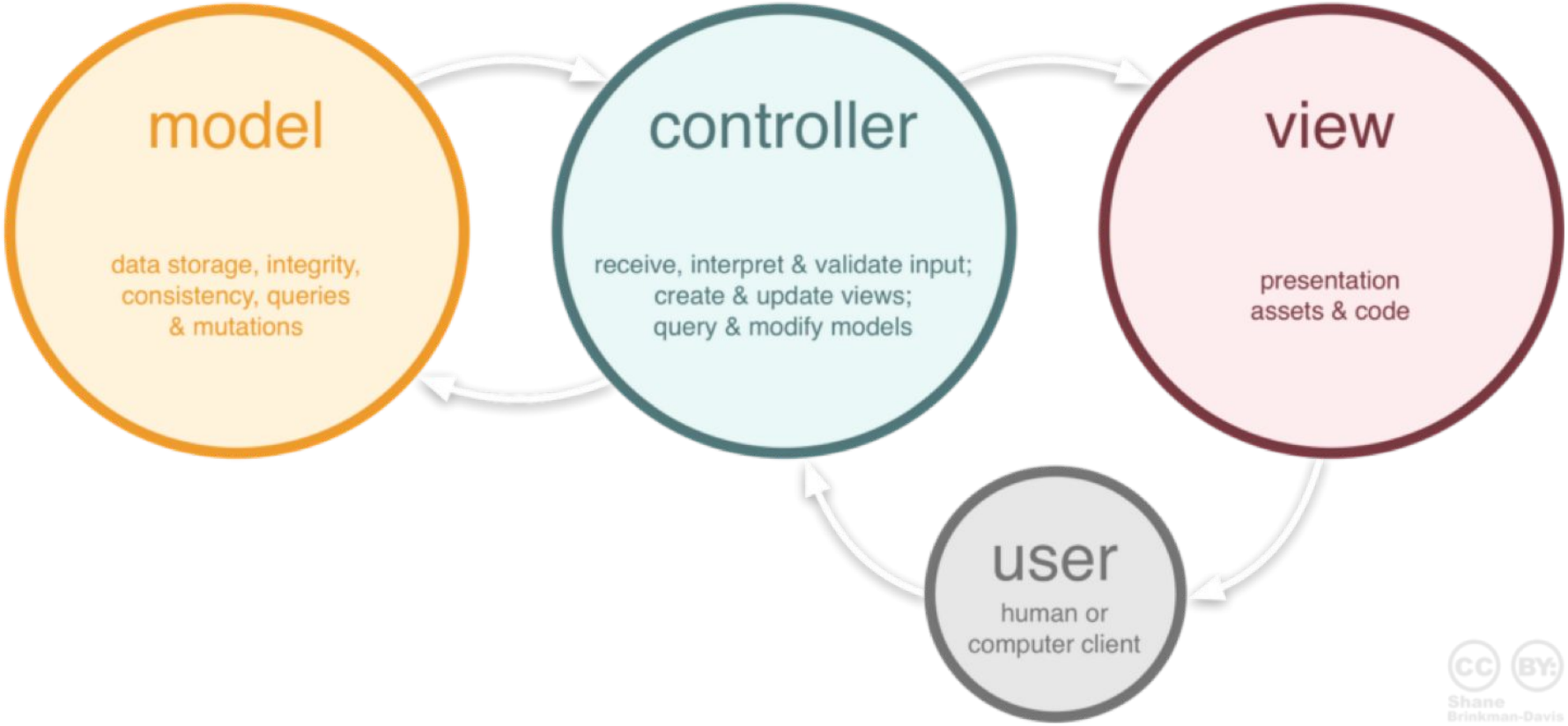
Hey everyone!

- Co-founder of Okteto.
- Former architect @ Atlassian, Software Engineer @ Azure.
- @rberrelleza in most places.

Application architecture has evolved a lot in the past few years...

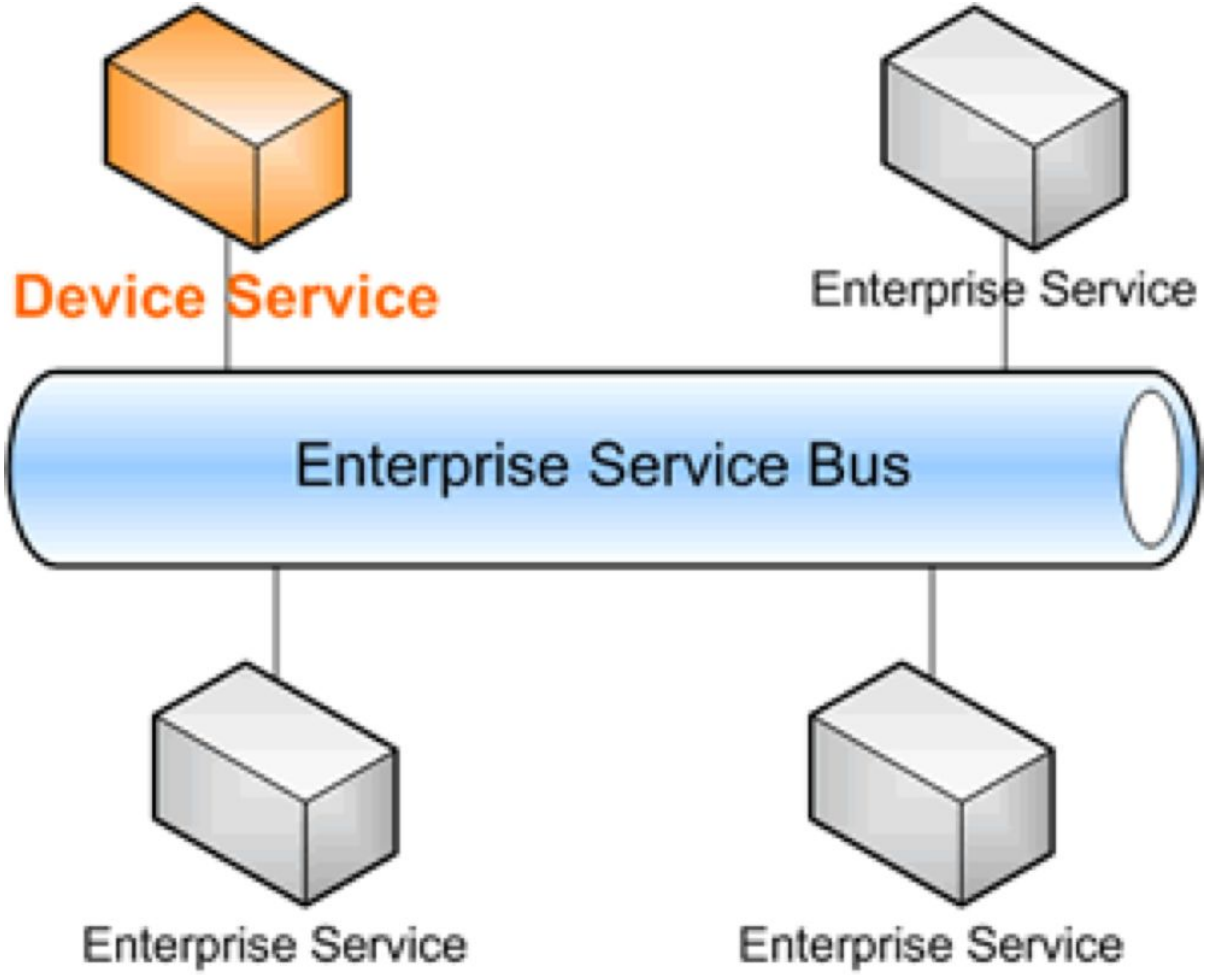
Monoliths

- Single process for everything.
- Runs locally.
- Easy to build.
- Hard to maintain.
- Hard to scale.



MVC applications

- Responsibility is split between layers.
- Easy to run locally.
- Easy to build.
- Hard to maintain.
- Hard to scale.


Device Service
Enterprise Service
Enterprise Service Bus
Enterprise Service
Enterprise Service

SOA architectures

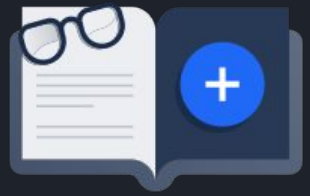
- Responsibility is split between components, typically with a Service Bus in between.
- Harder to run locally.
- Easy to build if properly abstracted.
- Easier to maintain.
- Easier to scale.


June

Microservices

- Responsibility is split between services, typically with a network in between.
- Hardest to run locally.
- Easiest to build if properly abstracted.
- Easiest to maintain.
- Easiest to scale.

Challenges



Dependency handling

Repeatable builds

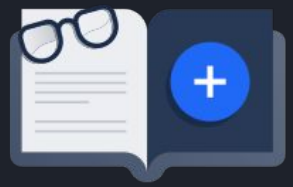
Automated deployments

Provisioning infrastructure


docker


by Google's
and internal systems
upports multiple cloud and bare-metal
nvironments
upports multiple container runtimes
00% Open source, written in Go
ge applications, not machines
Google Cloud

Challenges



Dependency handling

Repeatable builds

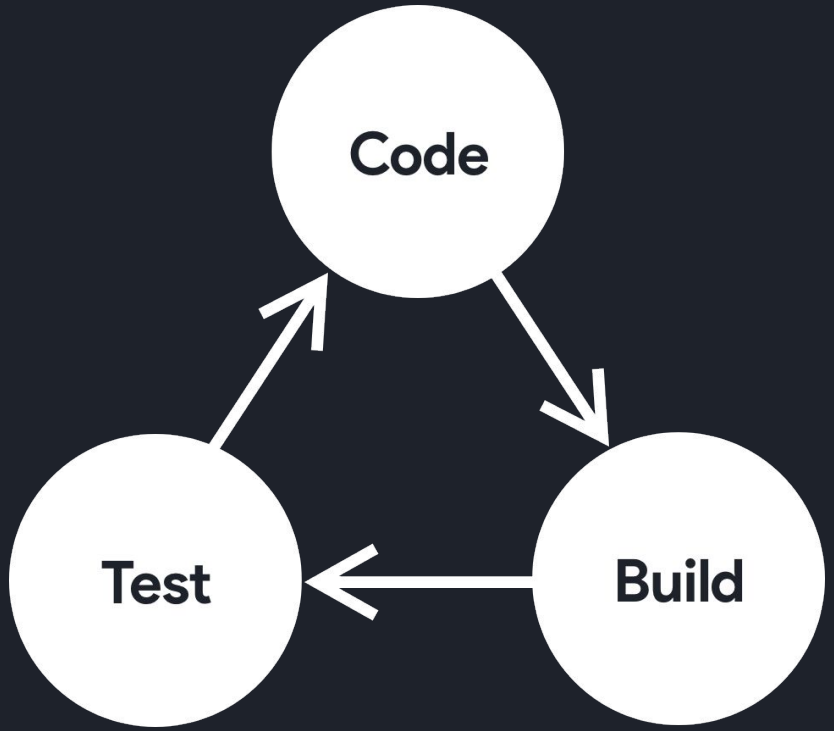
Automated deployments

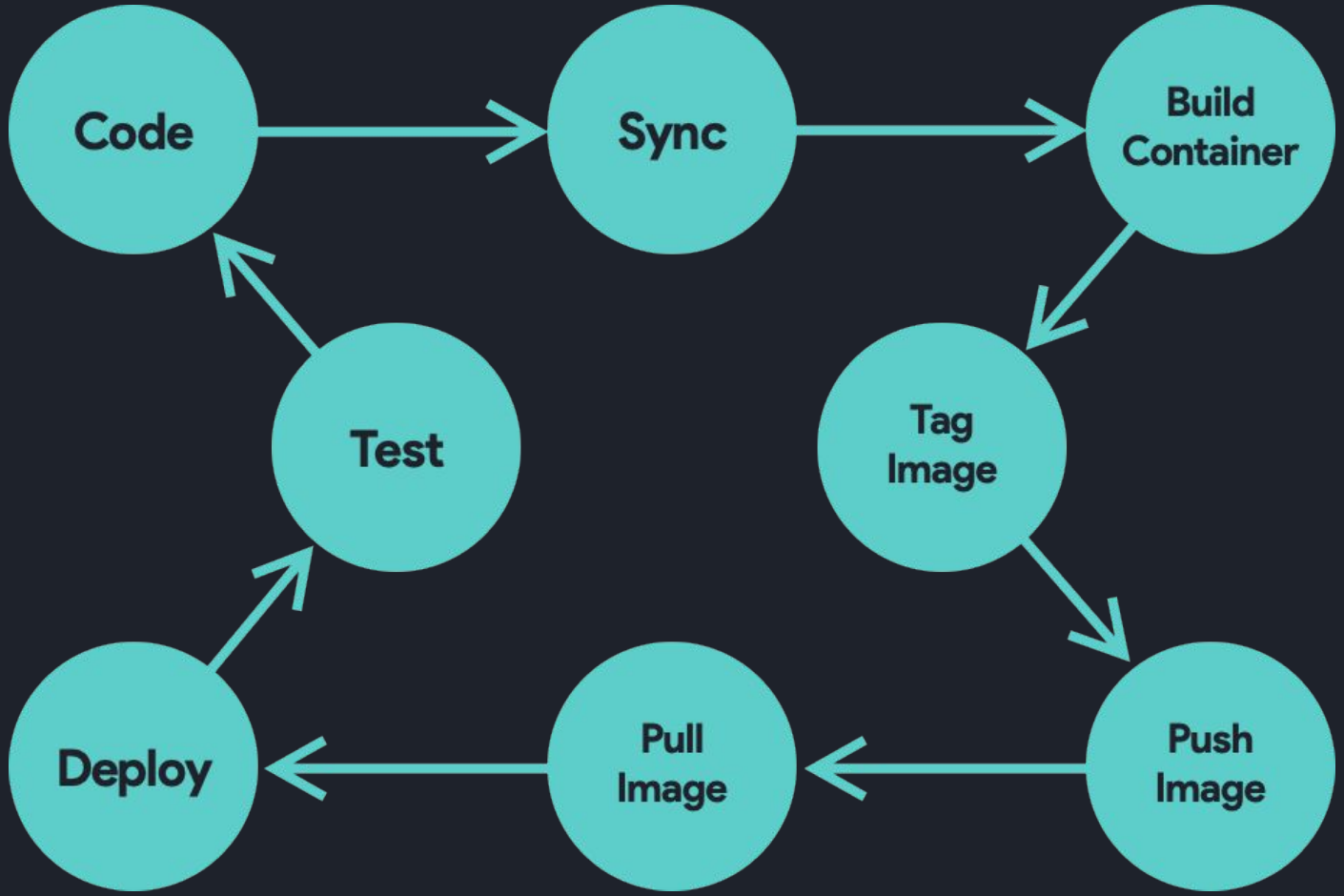
Provisioning infrastructure

We solved production*, but at the expense of the development experience.

Building Cloud Native Applications is hard.

An inner loop full of friction makes it harder to develop even the simpler features.


Code
Test
Build


Code
Sync
Build Container
Tag Image
Push Image
Pull Image
Deploy
Test

https://twitter.com/ellenkorbes/status/1143451907492655105

Community

Automate the inner loop

- Skaffold
- Draft
- Garden
- Tilt

Community

Dev environments on demand

- Visual Studio Code Online
- Code Ready workspaces (Eclipse Che)
- Jenkins X


yay!

But we're still waiting on builds and deploys.

We build Okteto to automate moving your development environment to Kubernetes.

Okteto: A Tool for Cloud Native Developers

Overview

Kubernetes has made it very easy to deploy applications to the cloud at a higher scale than ever, but the development practices have not evolved at the same speed as application deployment patterns.

Today, most developers try to either run parts of the infrastructure locally, or just test these integrations directly in the cluster via CI jobs or the "docker build, docker push, kubectl apply" cycle. It works, but this workflow is painful and incredibly slow.

Okteto makes this cycle a lot faster by launching your development environment directly in your Kubernetes cluster.

Features

Development environments on demand

Your development environment is defined in a simple yaml manifest.

- Run `okteto init` to inspect your project and generate your own config file.
- Run `okteto up` to launch your development environment in seconds.

Demo time!

Developing in the same environment as your applications are going to run lets you go way faster.

You are not waiting on builds and deploys.

And you’re fully integrated from the very beginning.

You can leverage the entire platform as well as your stack’s toolkit.

Incremental builds.

Hot reloaders.

Debuggers!

Demo time!

Q&A

Links!

- https://github.com/okteto/okteto
- https://cloud.okteto.com
- https://marketplace.visualstudio.com/items?itemName=okteto.remote-kubernetes
- https://twitter.com/rberrelleza

Thank you!

RAMIRO BERRELLEZA | @RBERRELLEZA