

Cloud Native Enterprise

September 2015

Agenda

- Why?
- What?
- How?

WHY?

Software is changing industries

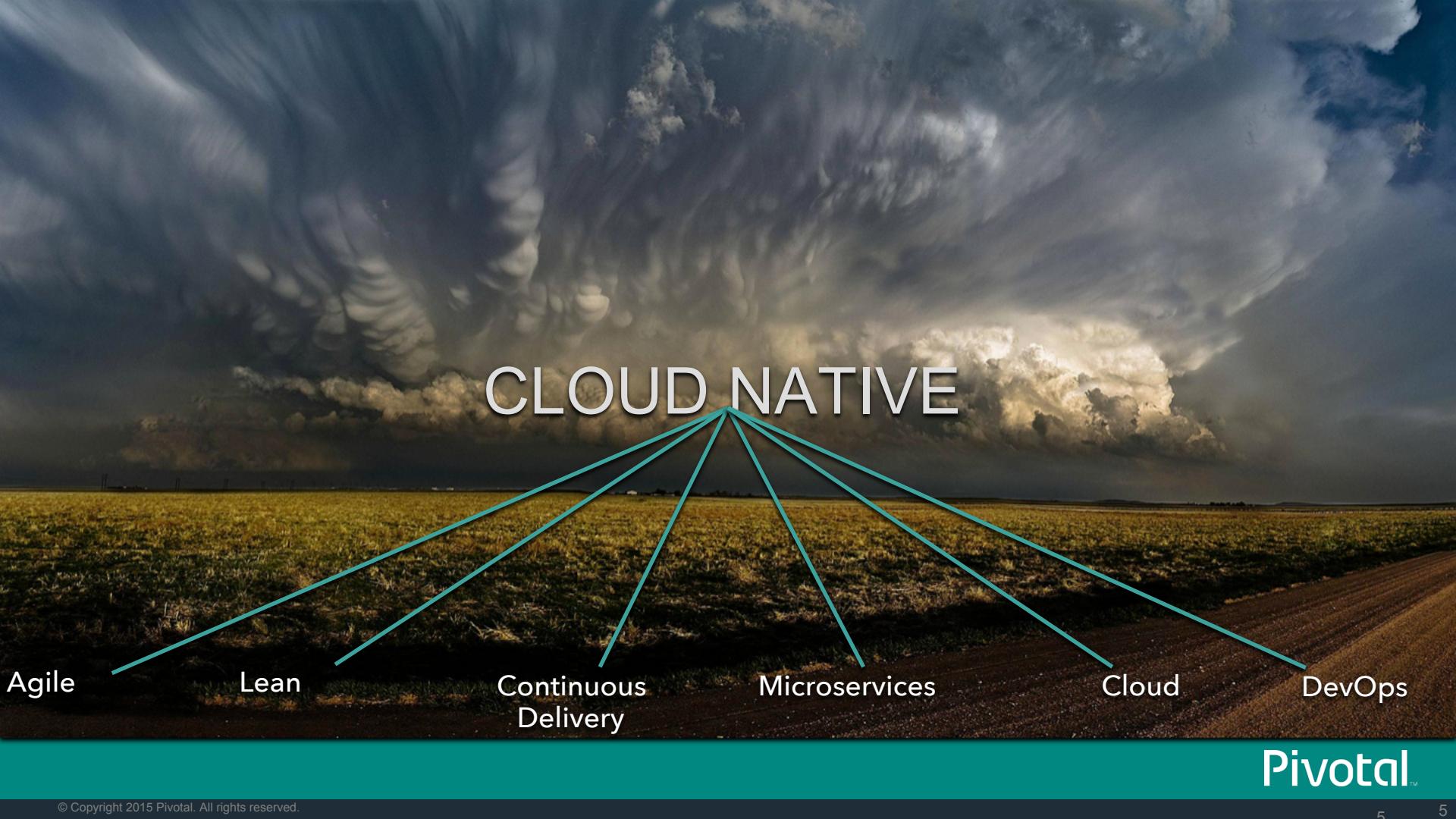


Square



U B E R





CLOUD NATIVE

Agile

Lean

Continuous
Delivery

Microservices

Cloud

DevOps

Pivotal™

The promise of Cloud Native

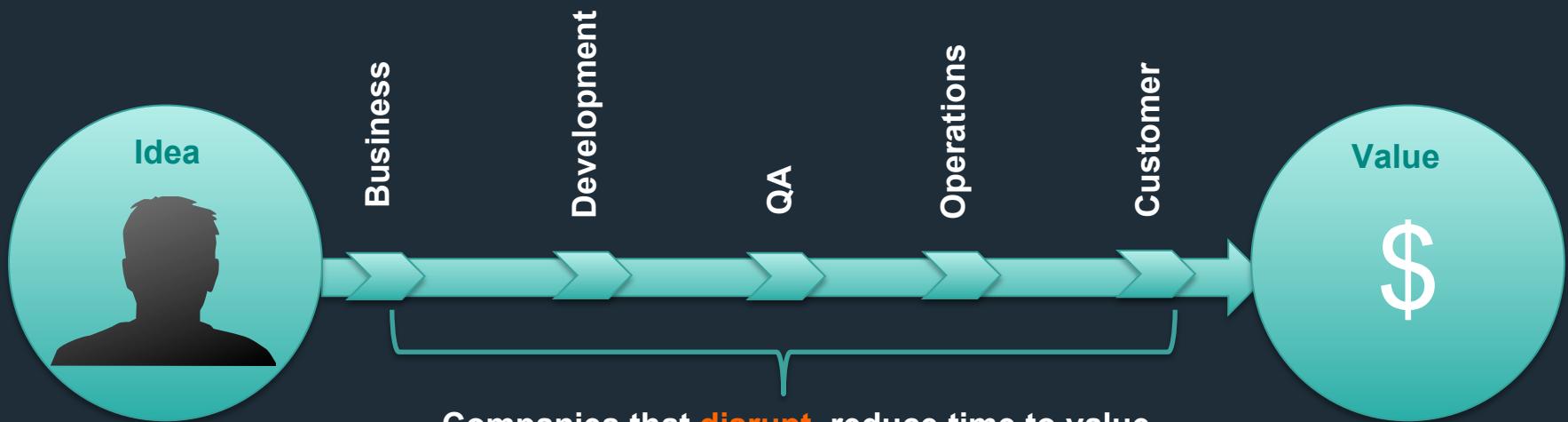
- Deploy new features daily to production
- Automate middleware
- Free/Cheap horizontal scaling
- Contract between App & Platform

WHAT?

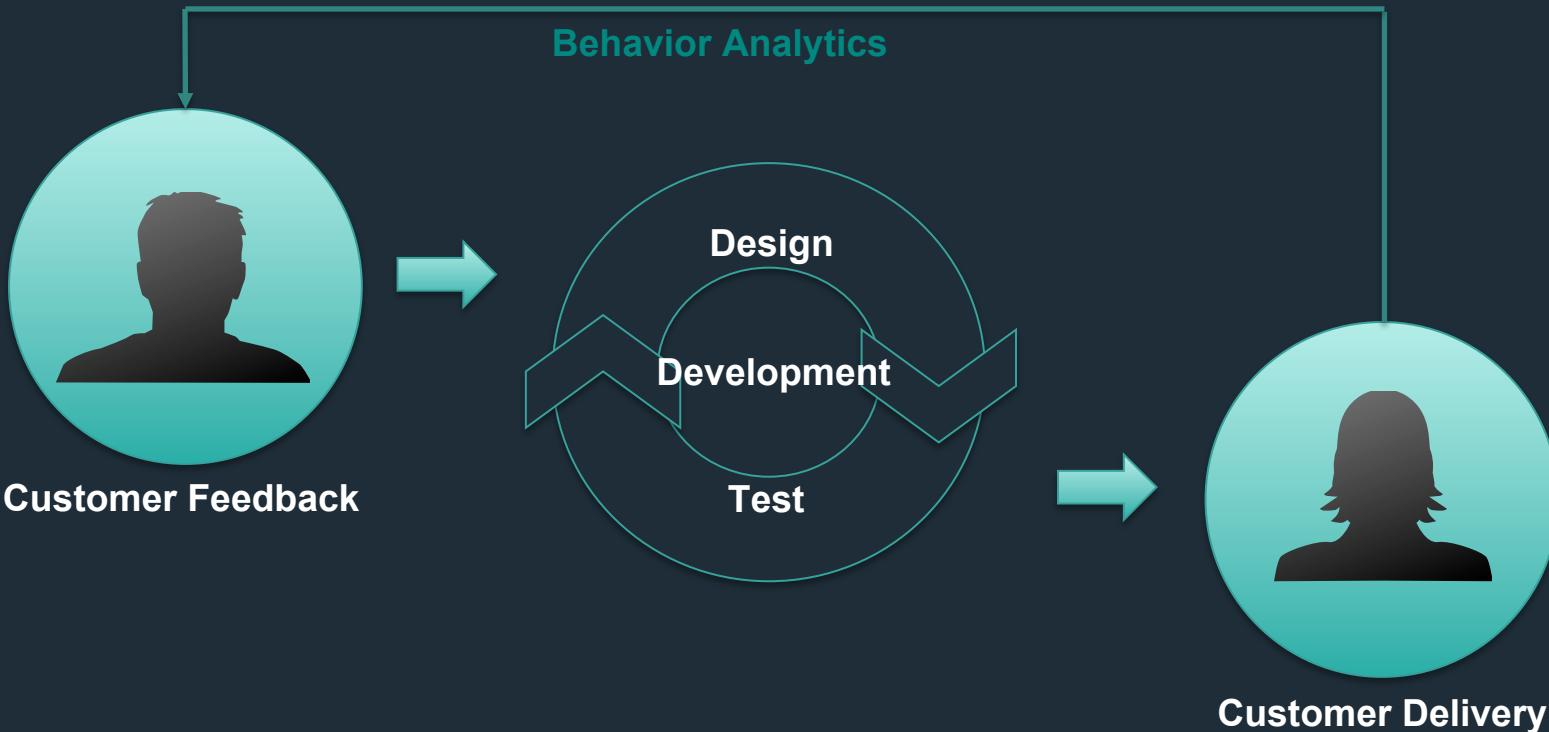
Characteristics of Cloud Native Architectures

- Continuously Delivered
- Twelve Factor Apps (<http://12factor.net/>)
- Microservices
- Self-Service agile infrastructure
- API-based collaboration
- Anti-fragility

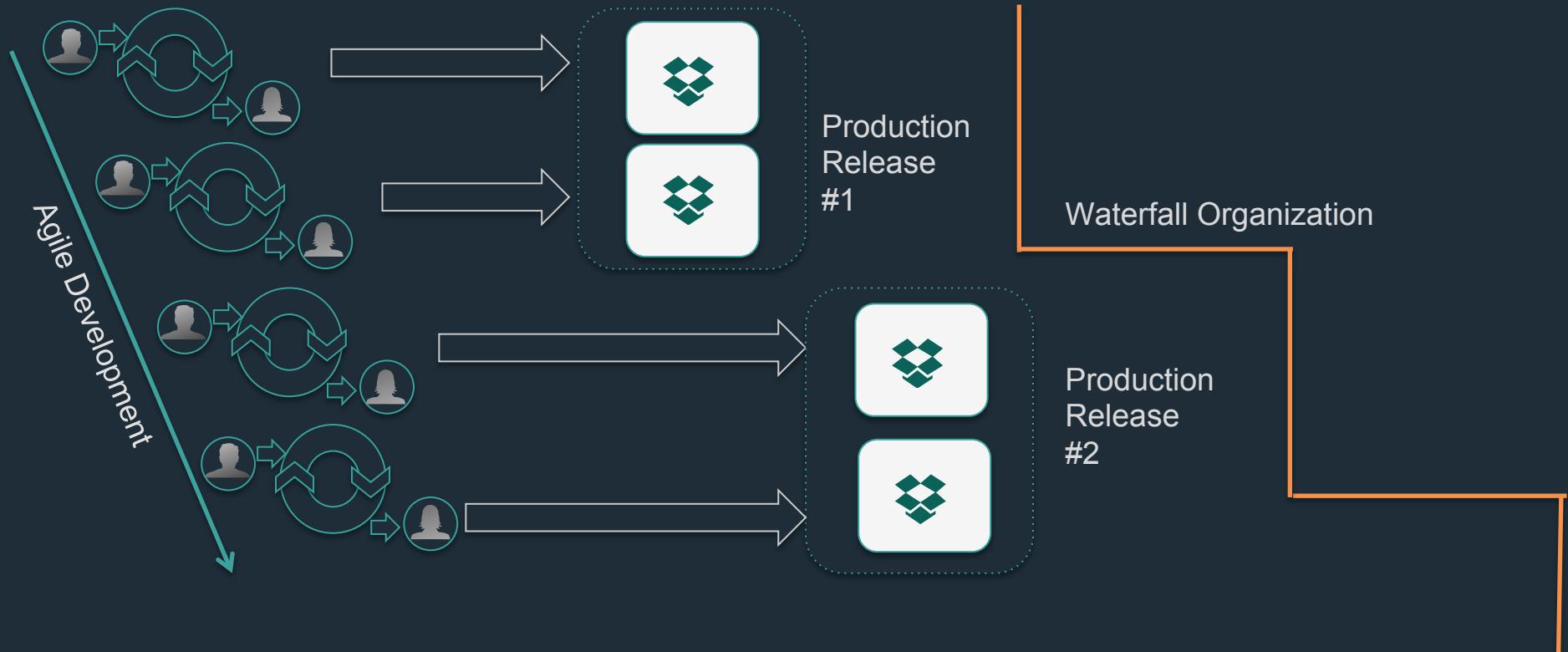
Enabling Continuous Delivery



Enabling Agile Delivery



WaterScrumFall



Conway's Law

“Any organization that designs a system (defined broadly) will produce a design whose structure is a copy of the organization's communication structure.”

- Melvin Conway, 1967

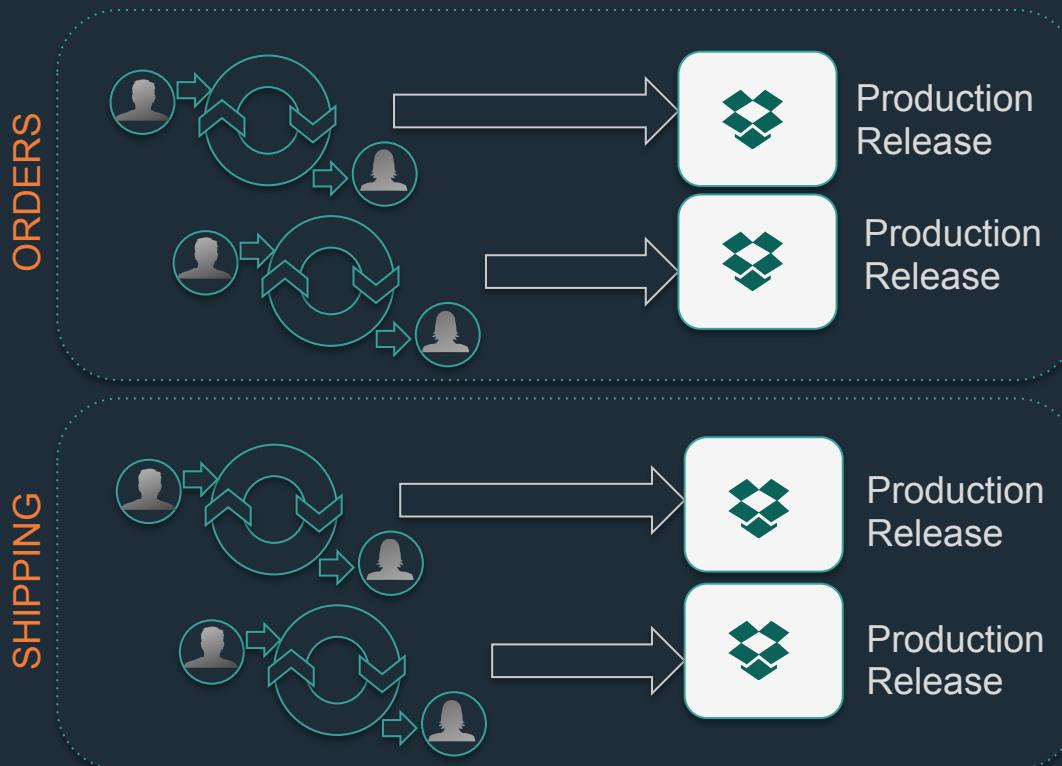
Inverse Conway Maneuver

Enable Continuous Delivery



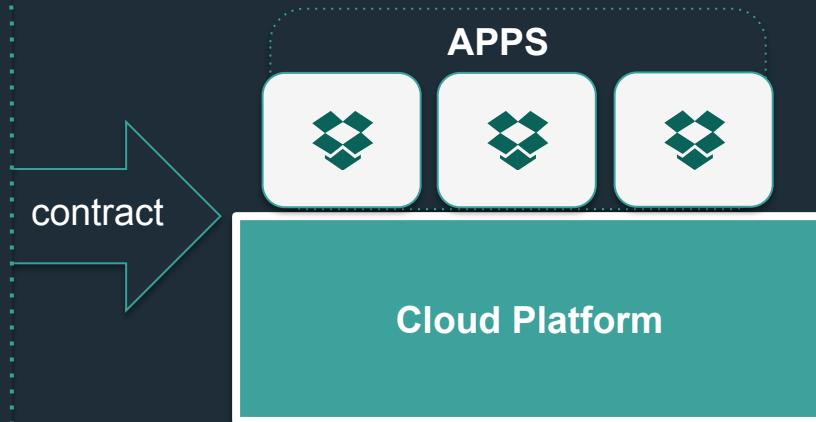
<http://jonnyleroy.com/2011/02/03/dealing-with-creaky-legacy-platforms/>
<http://www.slideshare.net/adriancockcroft/microservices-the-good-bad-and-the-ugly>

Enabling Continuous Delivery



Twelve-Factor Apps

- 1. Codebase
- 2. Dependencies
- 3. Configuration
- 4. Backing Services
- 5. Build, release, run
- 6. Processes
- 7. Port binding
- 8. Concurrency
- 9. Disposability
- 10. Dev/prod parity
- 11. Logs
- 12. Admin processes



Architectural and development practices – <http://12factor.net>

Microservice: Definition

If every service has to be updated in concert, it's not loosely coupled!

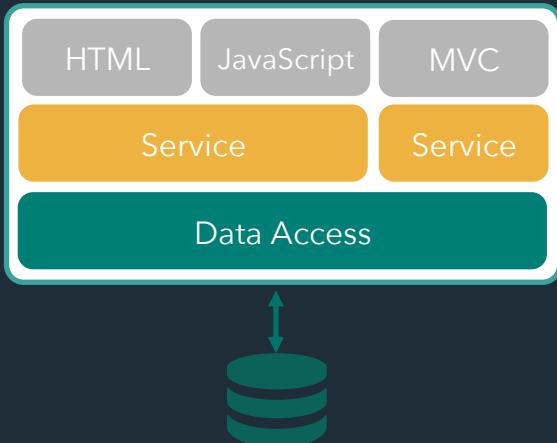
“Loosely coupled service oriented architecture with bounded contexts”

If you have to know about surrounding services you don't have a bounded context.

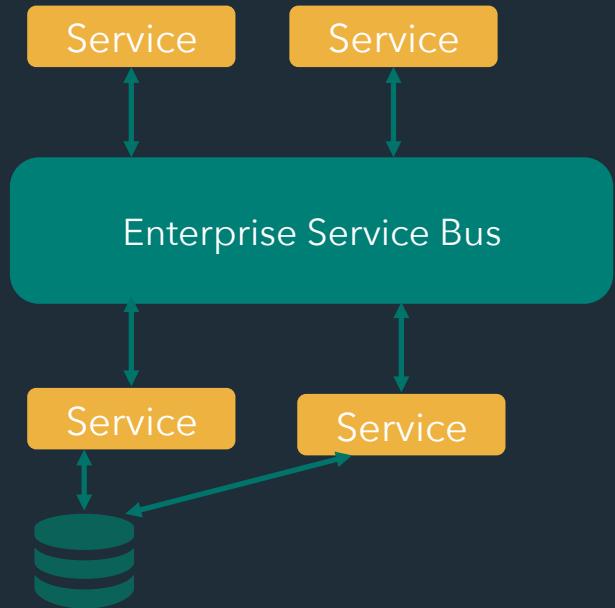
- Adrian Cockcroft

Microservices are NOT

Monolithic Application



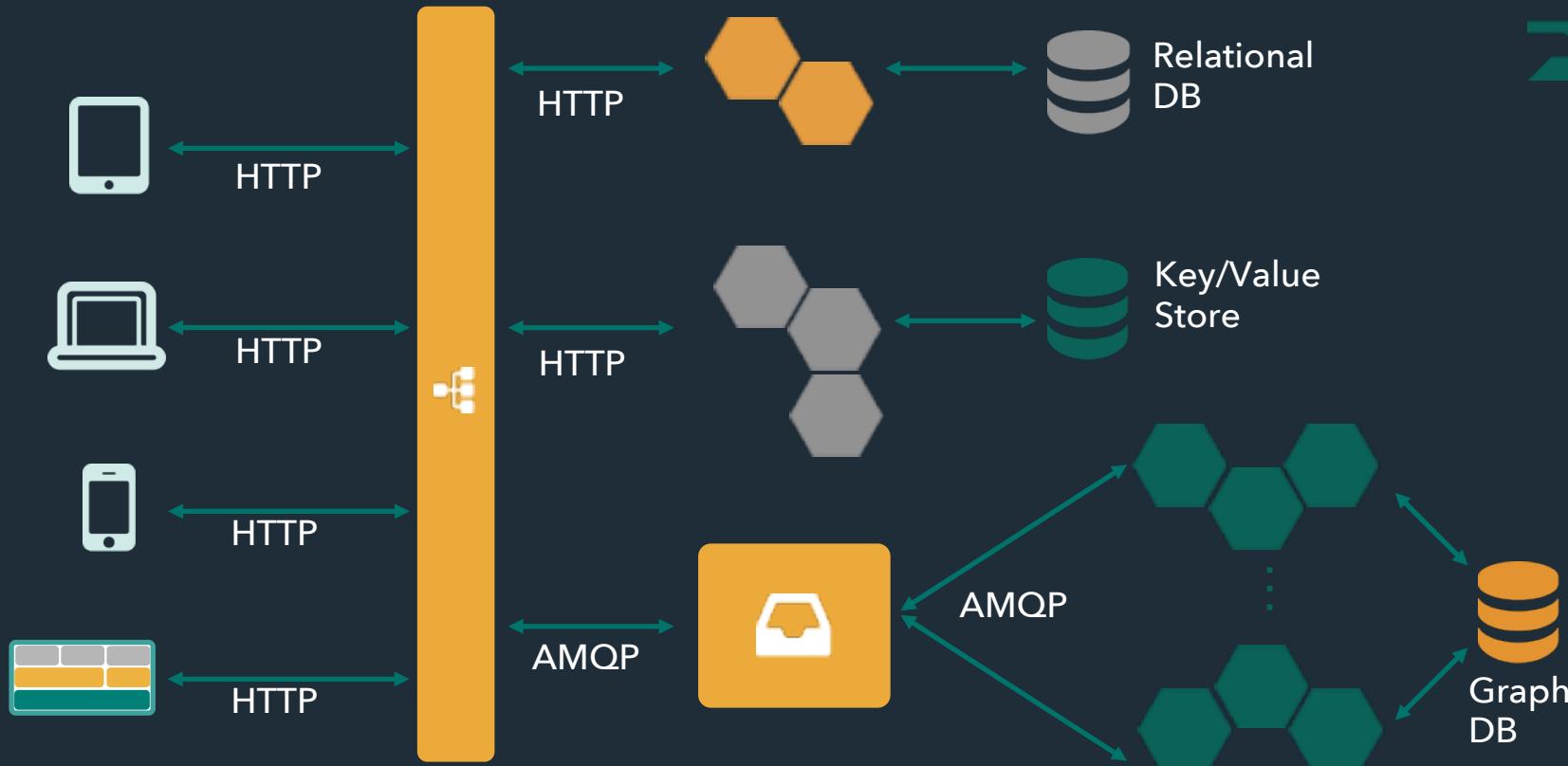
OR



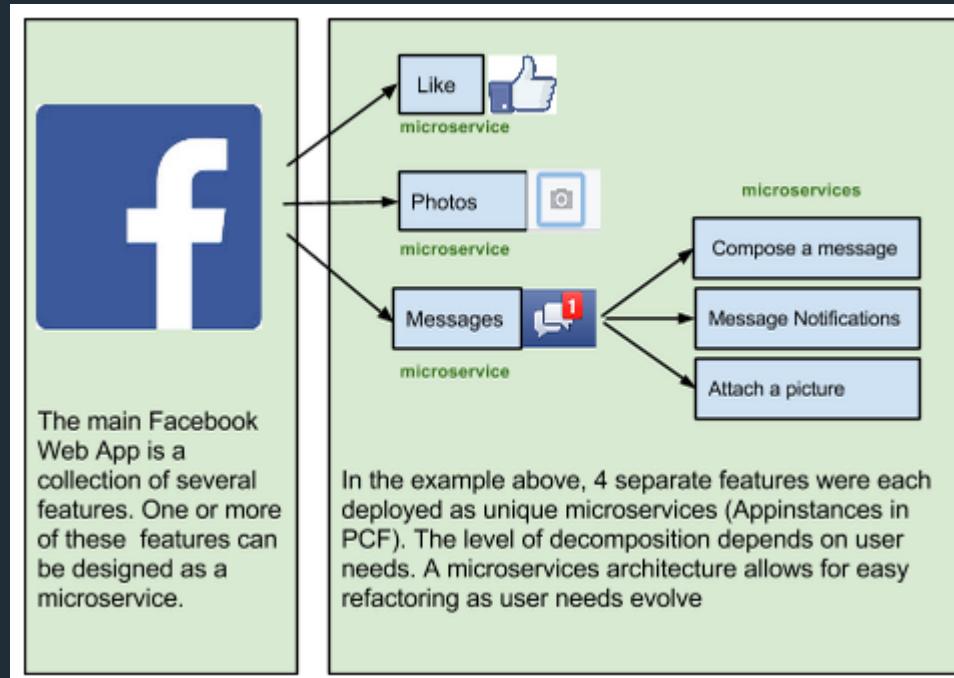
Tightly Coupled

Centralized

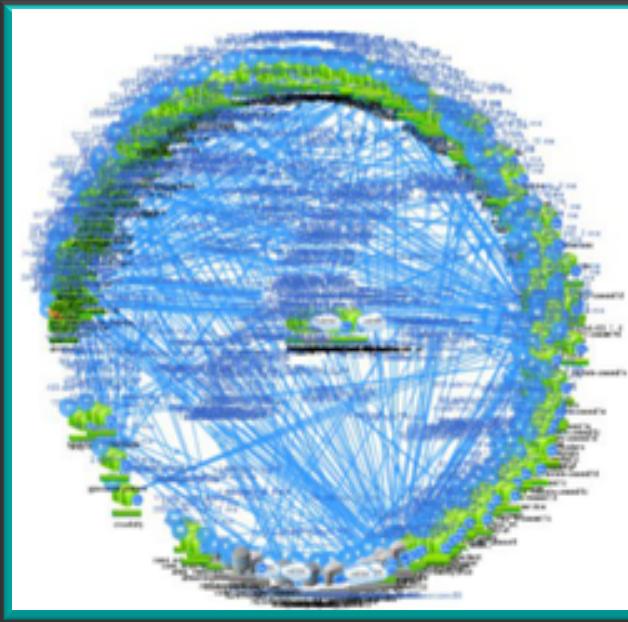
Microservice Architecture



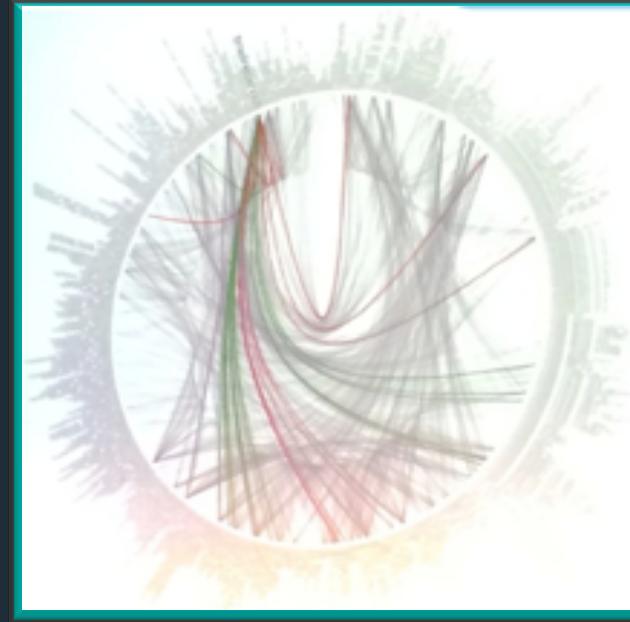
A familiar Microservice example



Microservices are hard



NETFLIX



twitter

Pivotal™

Challenges of distributed systems

- Configuration Management
- Service Registration & Discovery
- Routing & Load Balancing
- Fault Tolerance (Circuit Breakers)
- Monitoring
- Concurrent API Aggregation & Transformation

HOW?

It takes a platform

An end-to-end platform that
makes implementing
distributed application best
practices, a **turn-key** and **first**
practice

Anatomy of a Cloud Native Platform

Developer



1. Application Framework



Contract – 12 Factor App

Dev+Ops



2. Container Runtime

Contract – BOSH Release

IT Ops



3. Infrastructure Automation

Contract – Cloud Provider Interface

IT Ops



4. Infrastructure

“Contracts” between Applications, opinionated frameworks like Spring Boot and Spring Cloud and opinionated Cloud Native Platforms like Cloud Foundry help significantly accelerate the development of Cloud Native applications

Pivotal™

1. Application Framework



Spring

#1 Enterprise Java
App Framework

Netflix

Open Source

Spring Boot

Quick Start
Little to no config.
Opinionated View

Spring Cloud

Distributed Config
Service Registration
and Discovery
Routing/Load
Balancing
Service Integration
Fault Tolerance

Pivotal is the main custodian of Spring and we hire the main committers for Spring Boot and Spring Cloud. We can bring these resources to bear to help significantly accelerate a customer's journey to becoming a Cloud Native enterprise

NETFLIX | OSS



Spring Cloud

Companies want to be **fast** like Netflix

- Netflix needed to be faster to win / disrupt
- Pioneer & vocal proponent of microservices - the key to their speed and success
- Netflix OSS supplies parts, but it's not a solution
- Difficult for enterprises to build it themselves
- Pivotal offers the closest thing to “Netflix in a box” today



“

“Velocity on the JVM is the Killer App”

- Andy Glover (Netflix Eng) @ SpringOne2GX 2014 Keynote

<https://youtu.be/xU267-YHN5c?t=1938>

Spring Cloud Services Suite



Spring Cloud
Services



Config Server

- Git URL for Config Repo provided via Service Dashboard (post-provisioning)
- Single tenant, scoped to CF space



Service Registry

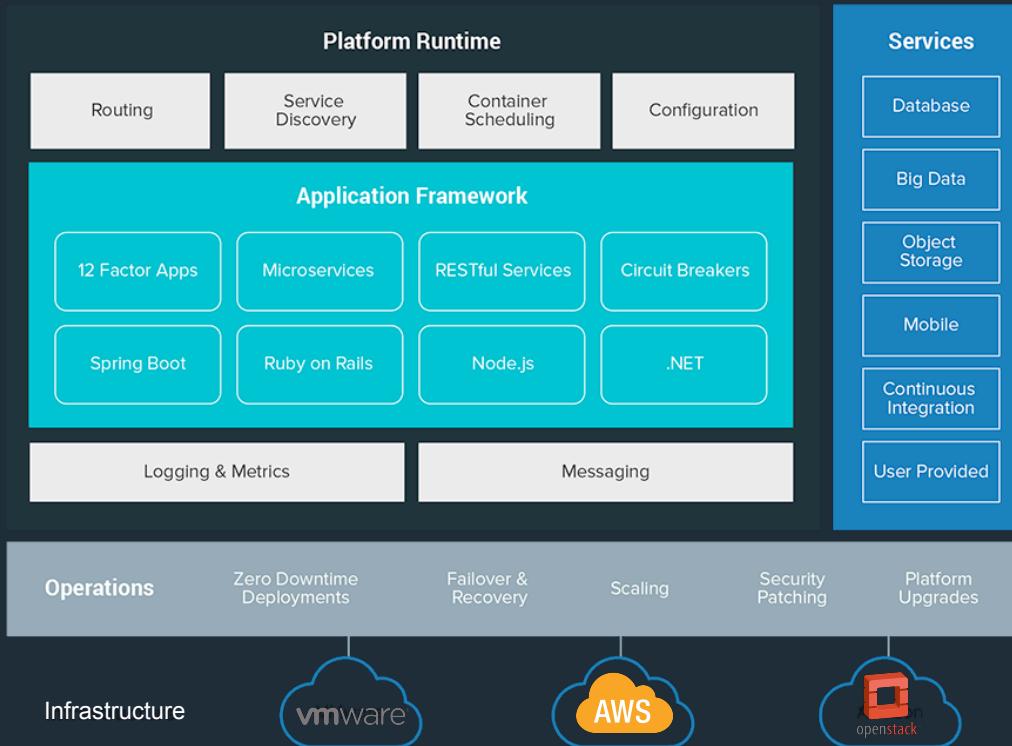
- Service Registration and Discovery via Netflix OSS Eureka
- Registration via CF Route



Circuit Breaker
Dashboard

- Netflix OSS Turbine + Hystrix Dashboard
- Aggregation via AMQP (RabbitMQ)

It Takes a Platform





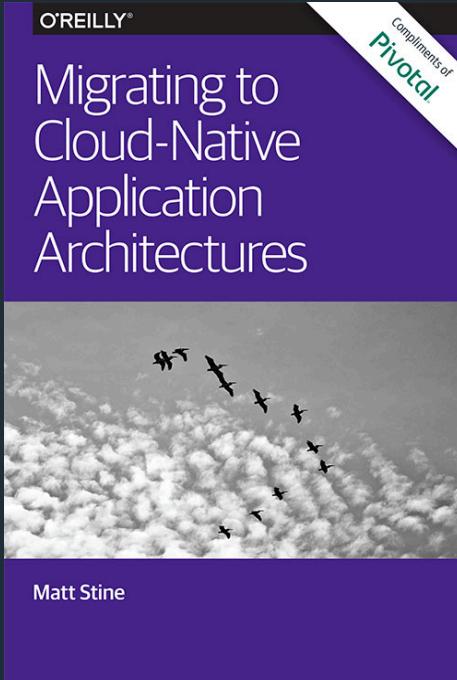
DEMO

CF

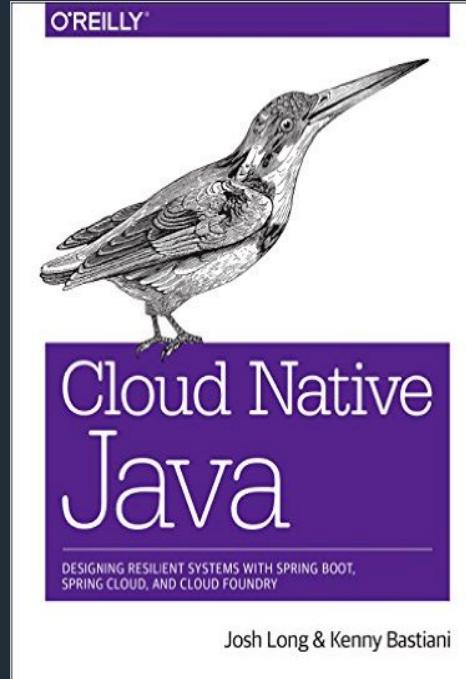
HOW?

Do we get started

Read



By Matt Stine (@mstine)

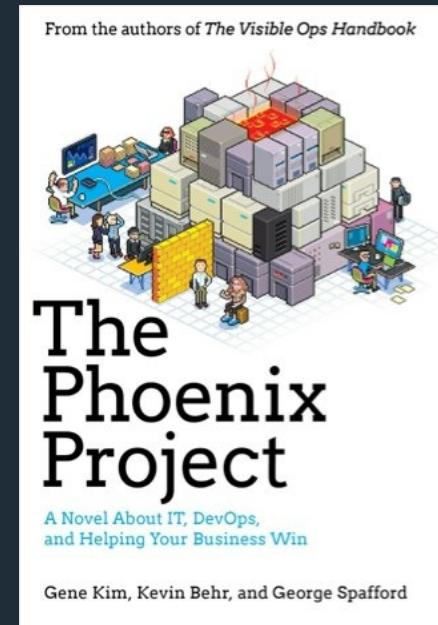
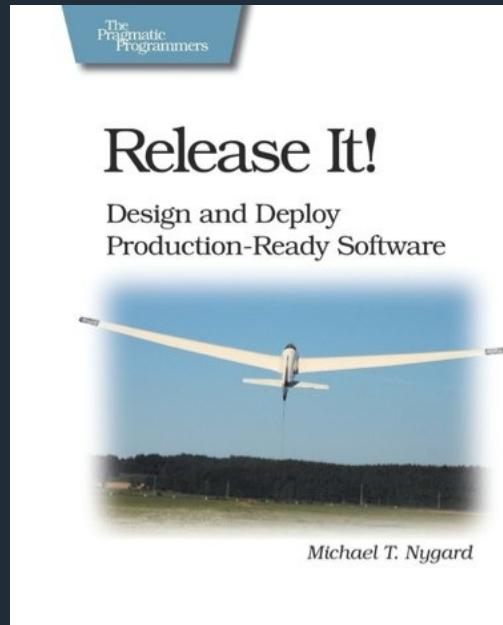
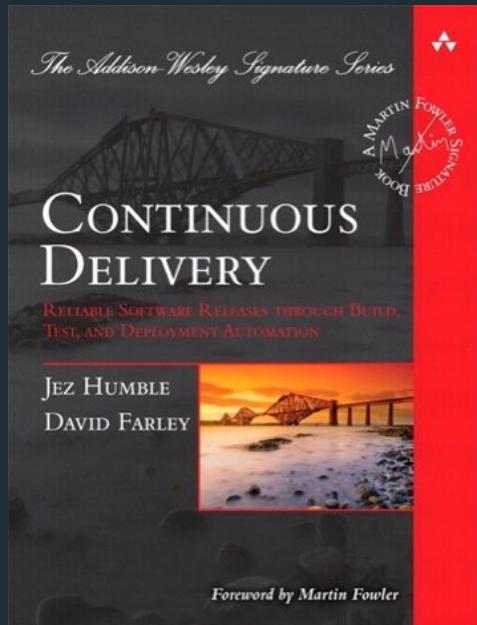


By Josh Long (@starbuxman)
and Kenny Bastani

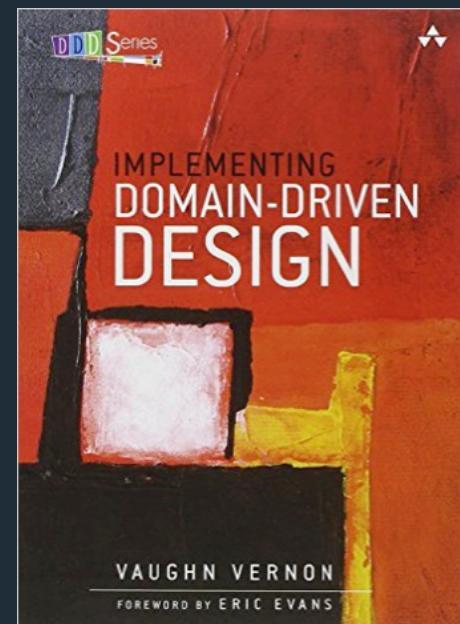
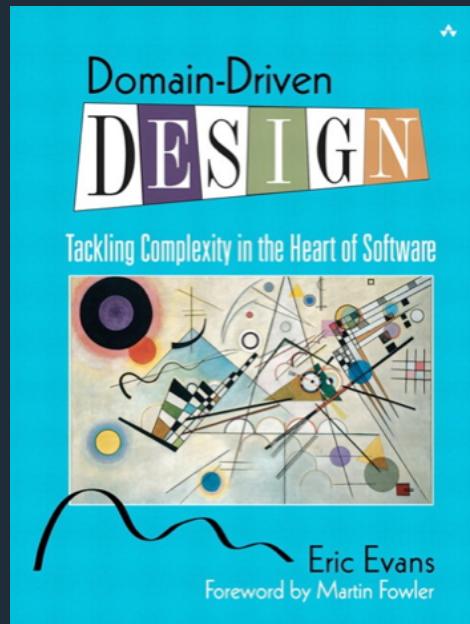
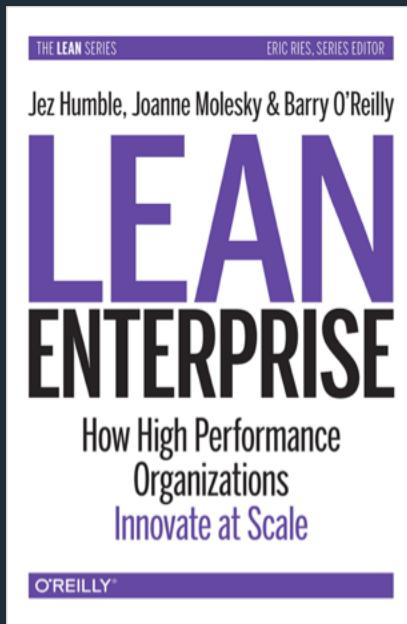
Build - <http://start.spring.io>



Foundations



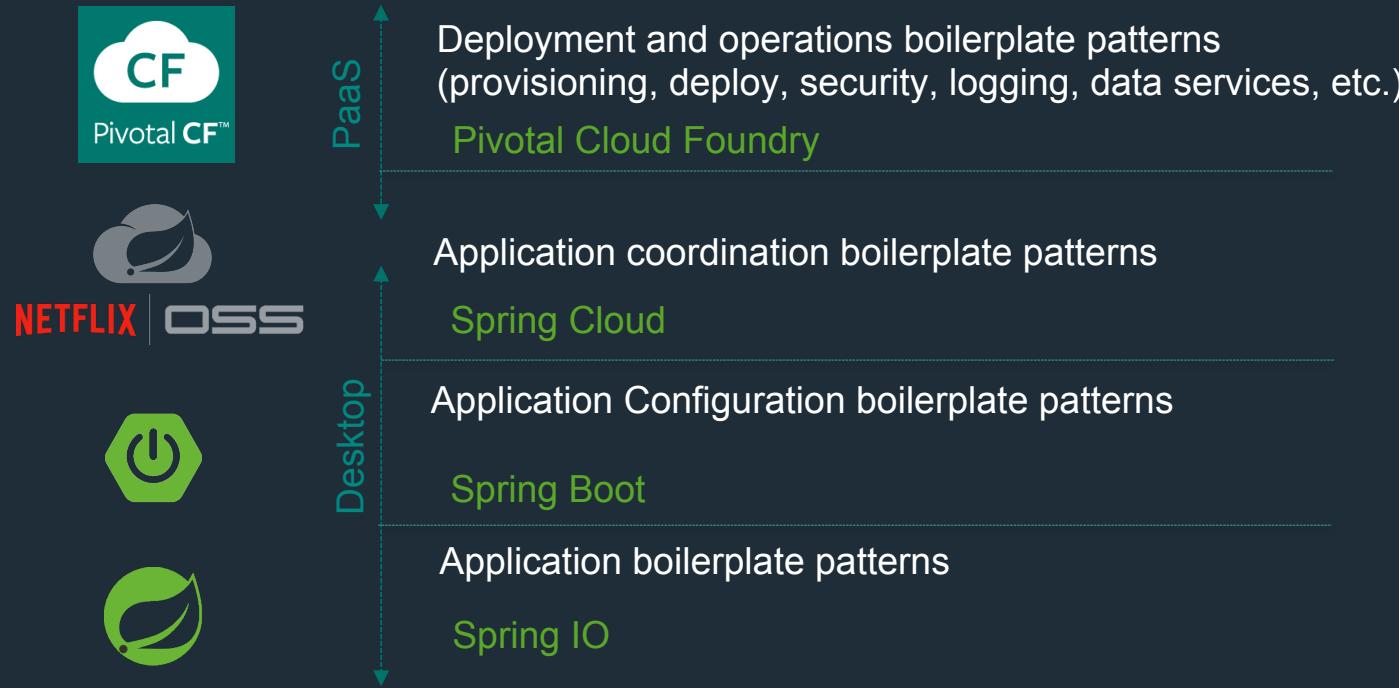
Foundations



Pivotal

A NEW PLATFORM FOR A NEW ERA

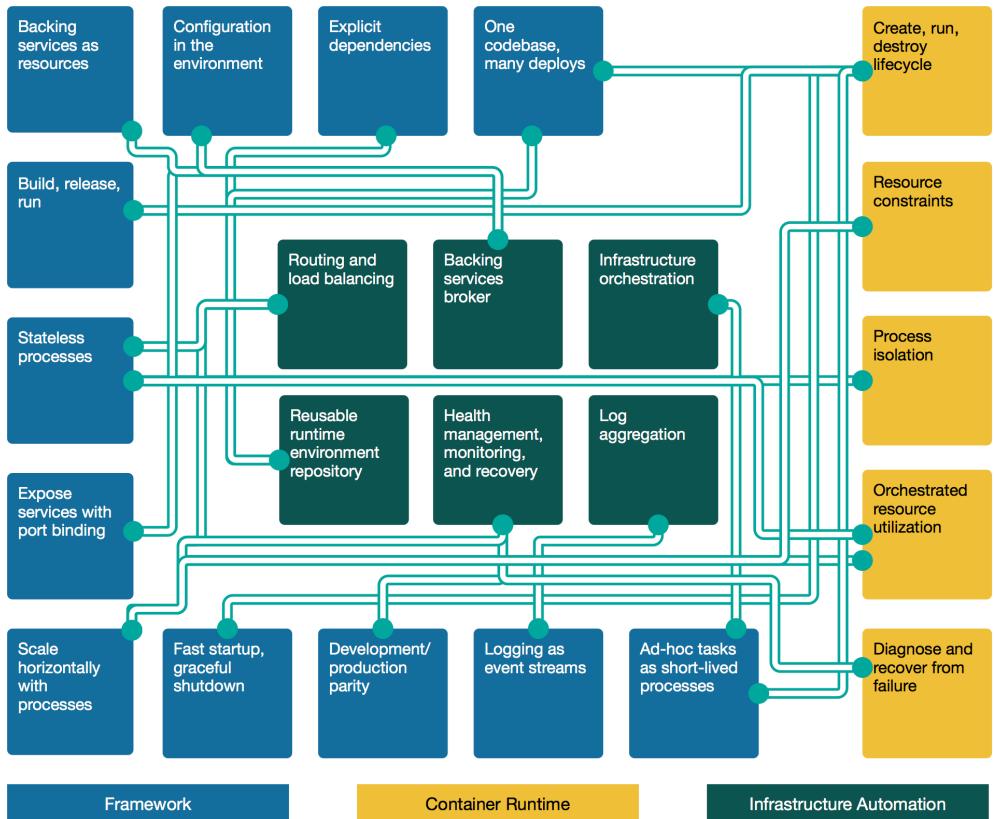
Anatomy of a Cloud Native Platform



The Cloud-Native Architecture

What you need to be cloud native.

A cloud platform enables continuous delivery speed by standardizing and automating as much as possible



by @caseywest of @pivotals @cloudfoundry

Pivotal™