



PCF Technical Overview

Anand Rao

Twitter : @honnuanand

Linkedin : <https://www.linkedin.com/in/honnuanand/>

March 8th 2018

Agenda

- Introductions + Level Set
- Value Proposition
- Pivotal Cloud Foundry Overview
- Competitive Landscape



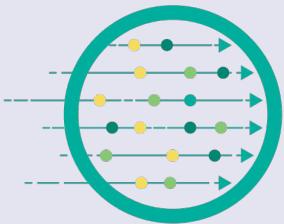
Level Set

1. What are you trying to **accomplish?**
2. Any noteworthy **milestones** and **dates?**
3. What **decisions** have already been made?



Value Proposition

Typical Customer Outcomes with Pivotal Cloud Foundry



Speed

Deploy new code
thousands of times a month

HIGHLIGHTS

- Best platform for Spring Boot
- Container-ready
- Native Windows + .NET
- Full integrated with CI/CD

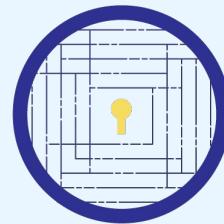


Stability + Scalability

Deliver enterprise SLAs and
breakthrough operational efficiency

HIGHLIGHTS

- Four layers of HA
- Zero downtime deployments
- Logging, metrics, and scaling
- Linux + Windows Server



Security

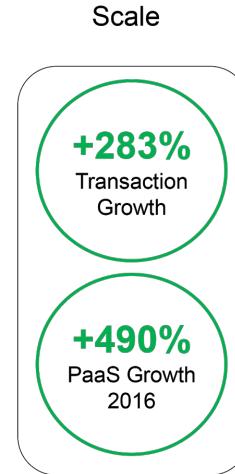
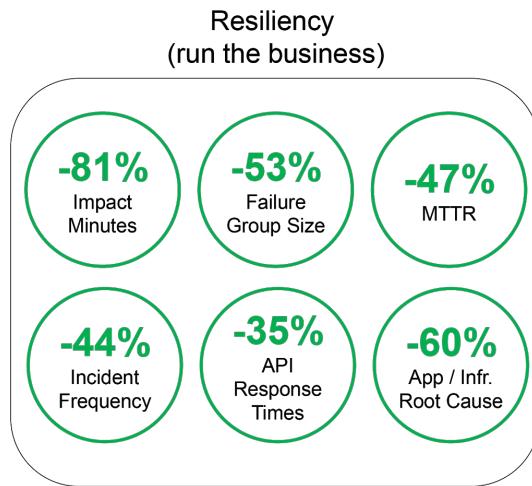
Improve your security posture
with built-in capabilities

HIGHLIGHTS

- “Secure by default” containers
- Full-stack support
- Rapid fixes to CVEs
- “Repair, repave, rotate”

Example Customer Outcomes

Our Results

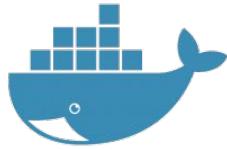


Pivotal Cloud Foundry Overview

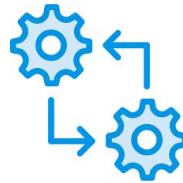
PCF Overview

Platform Evolution

There are many ways to package and run workloads in the cloud



Containers



Batches



Event-driven Functions



Microservices

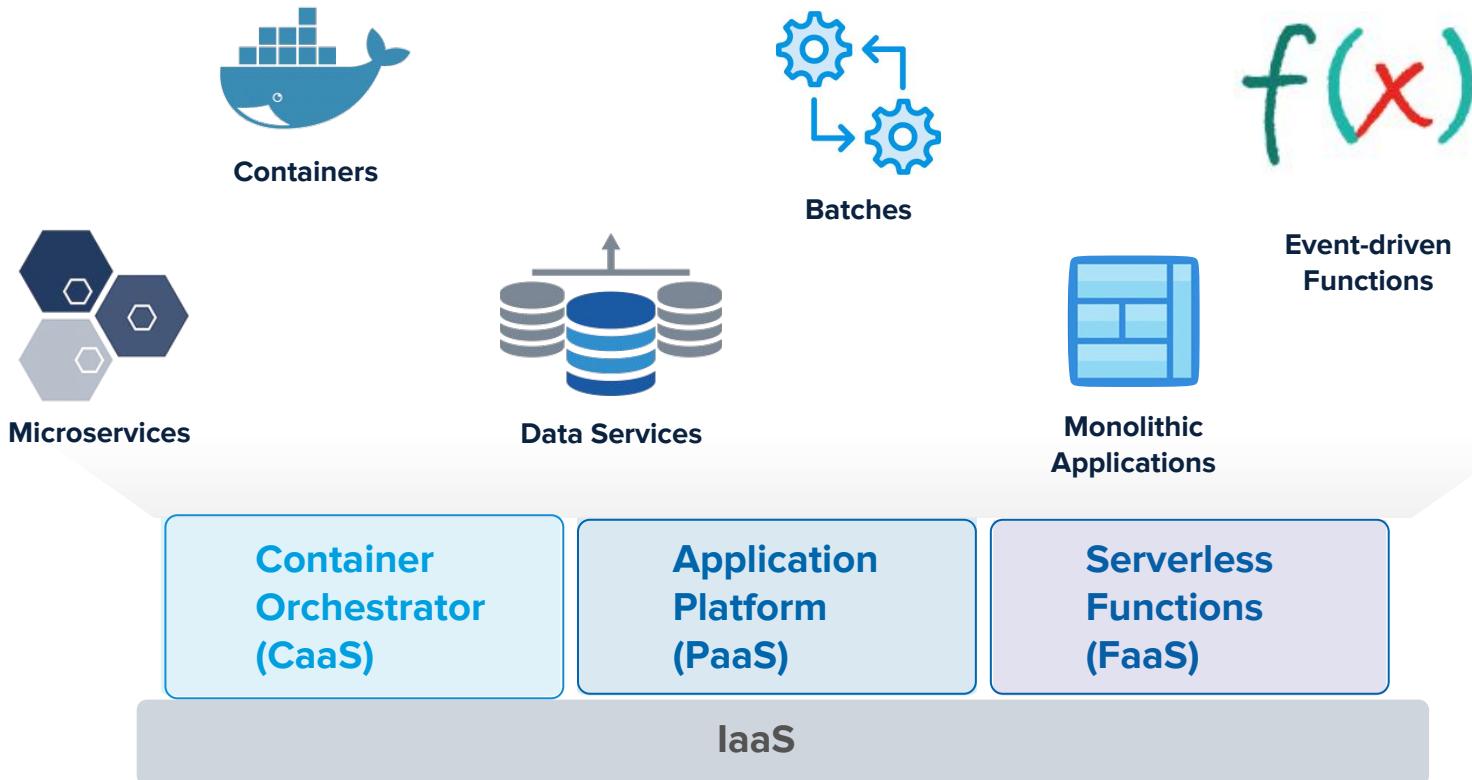


Data Services

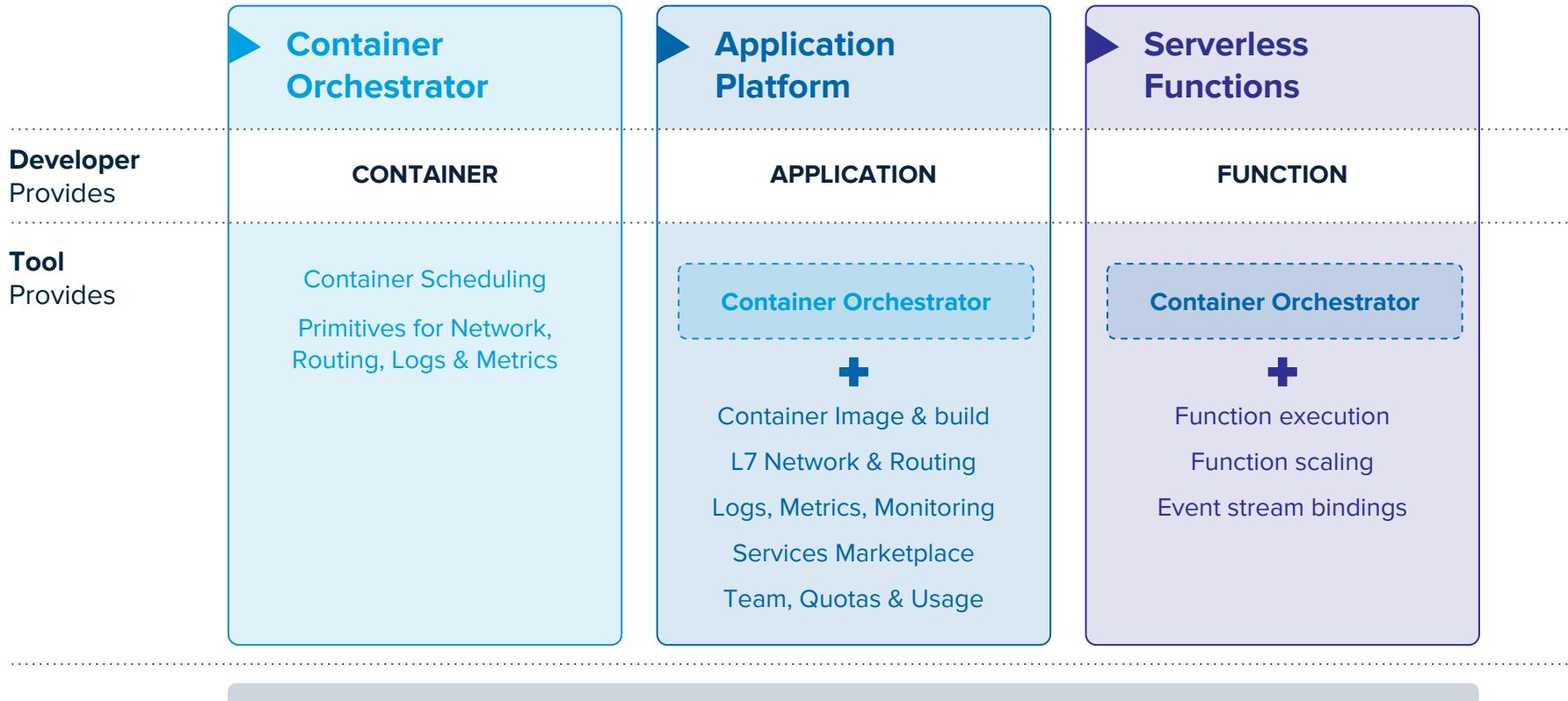


Monolithic Applications

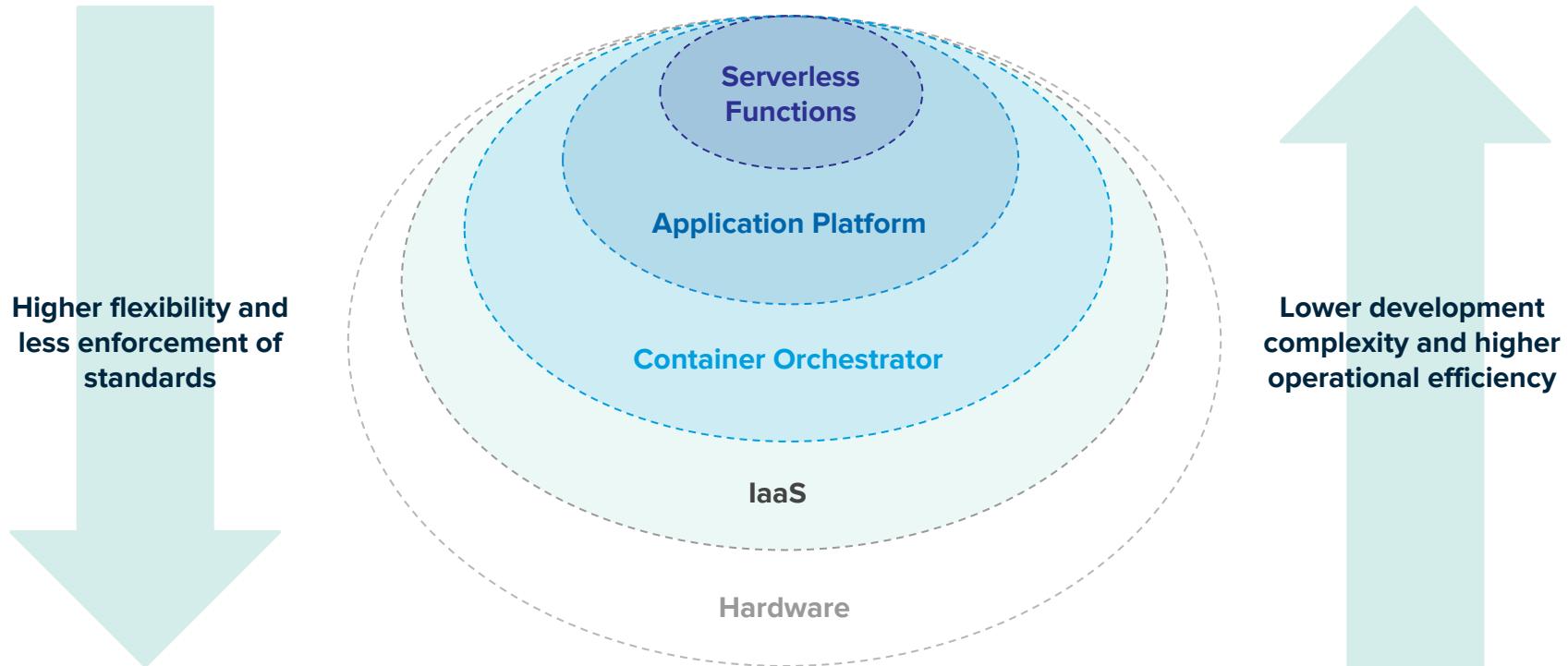
Your goal: choose the right runtime for each workload



Choose the right tool for the job



Use the highest abstraction possible



Your strategic goal: Push as many workloads as technically feasible to the top of the platform hierarchy

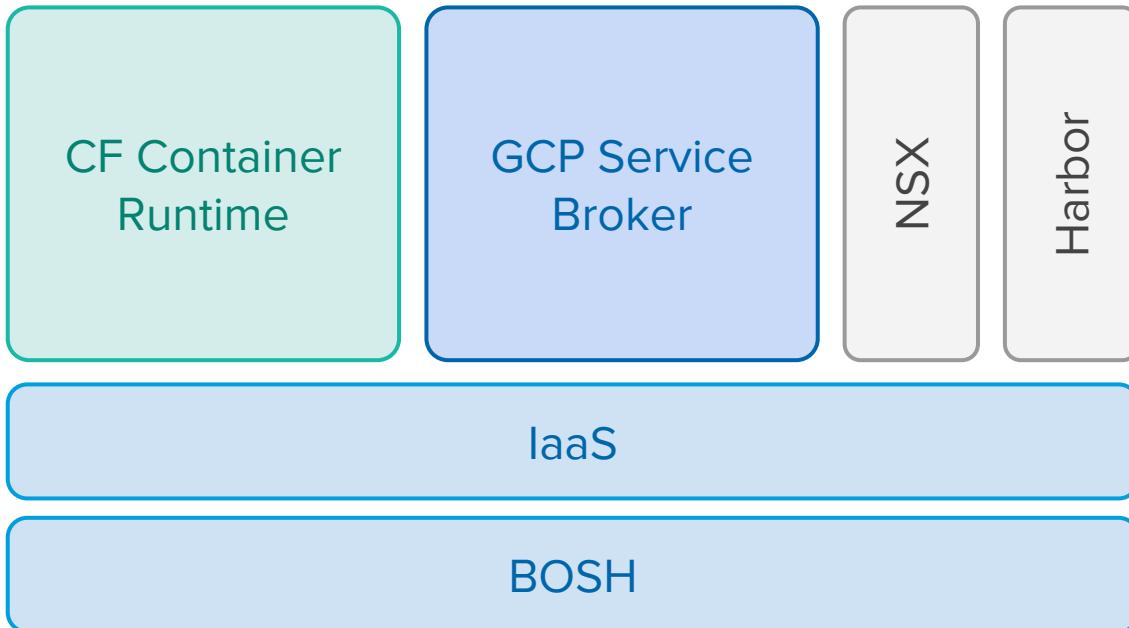
PCF (Originally)

ERT

Services

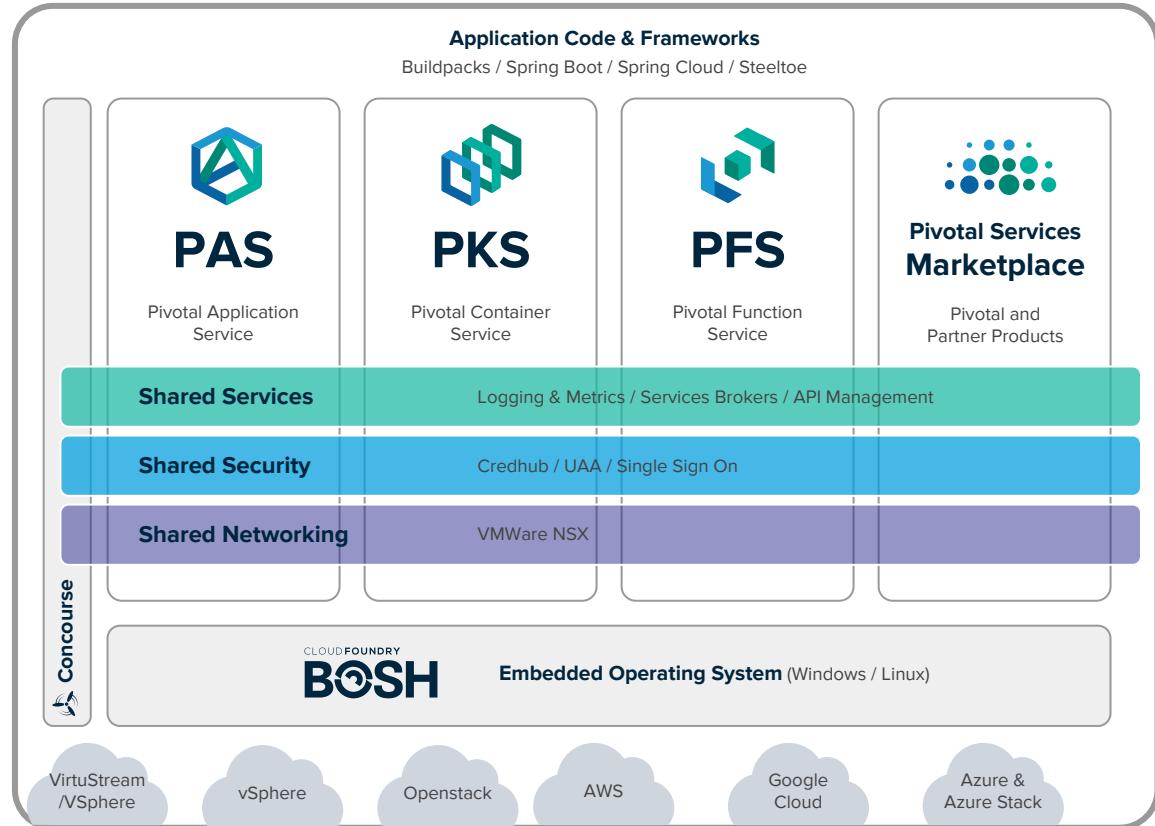
BOSH

PKS (Originally)

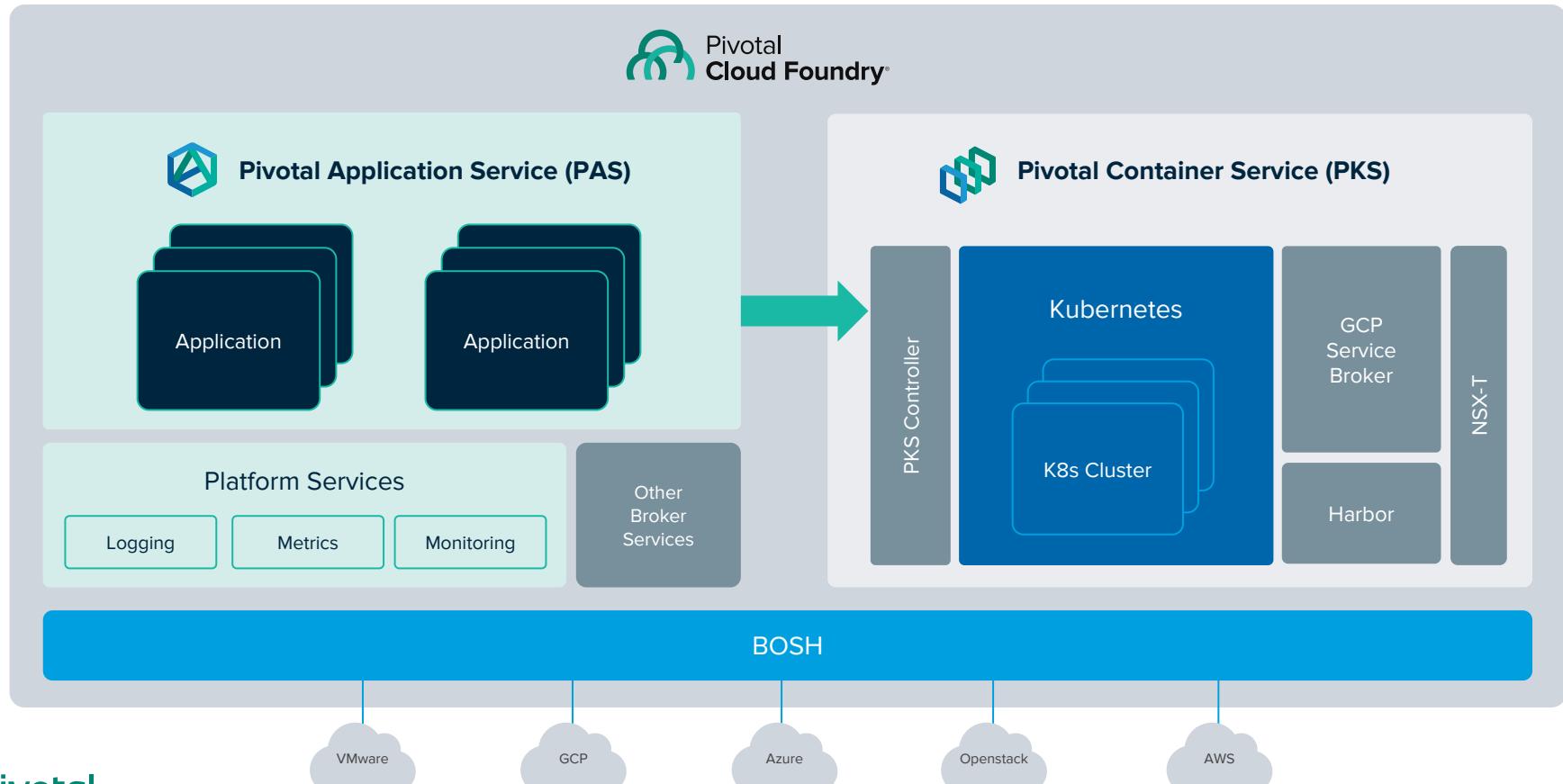


Any App Every Cloud One Platform

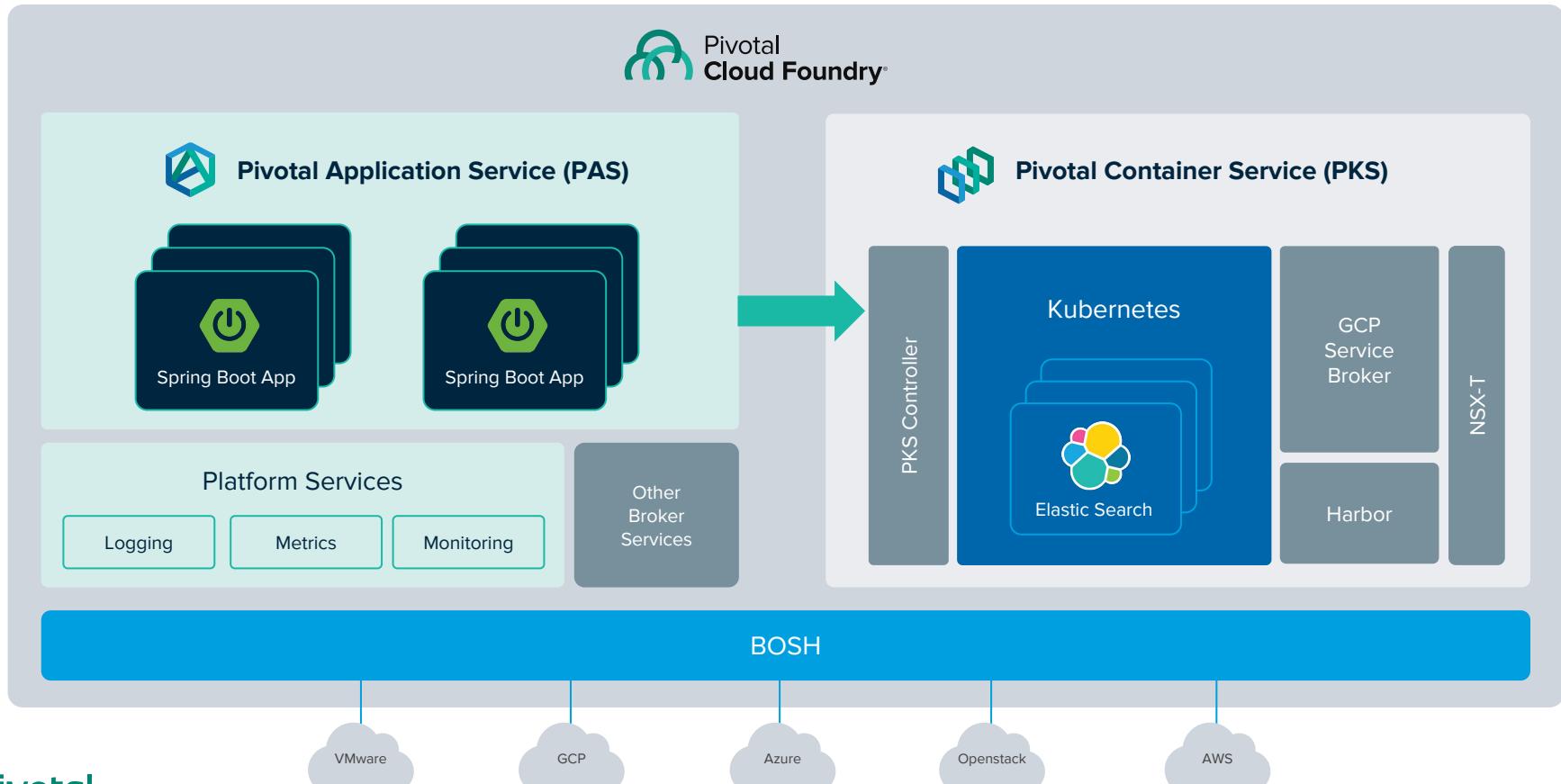
PCF 2.0 — for everything
that matters



Leveraging more than one abstraction



Sample Use Case



Containers

Pivotal Cloud Foundry Supports
Both Types of Containers

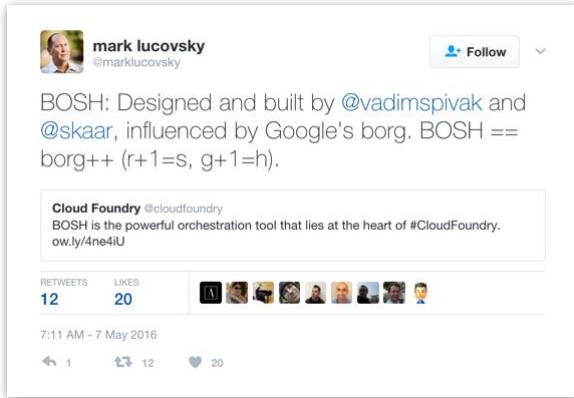
Developer-Built	Platform-Built
deploy apps as a container, often a Docker image	deploy apps as code
developer identifies middleware	platform handles middleware
bring your own language	language support dependent on existing buildpacks
developer is responsible for lifecycle management	platform handles much of the lifecycle management
Use for Elasticsearch, Apache Spark, ISV apps	Use for Java, .NET, Node

PCF Overview

The Platform Beneath All Abstractions

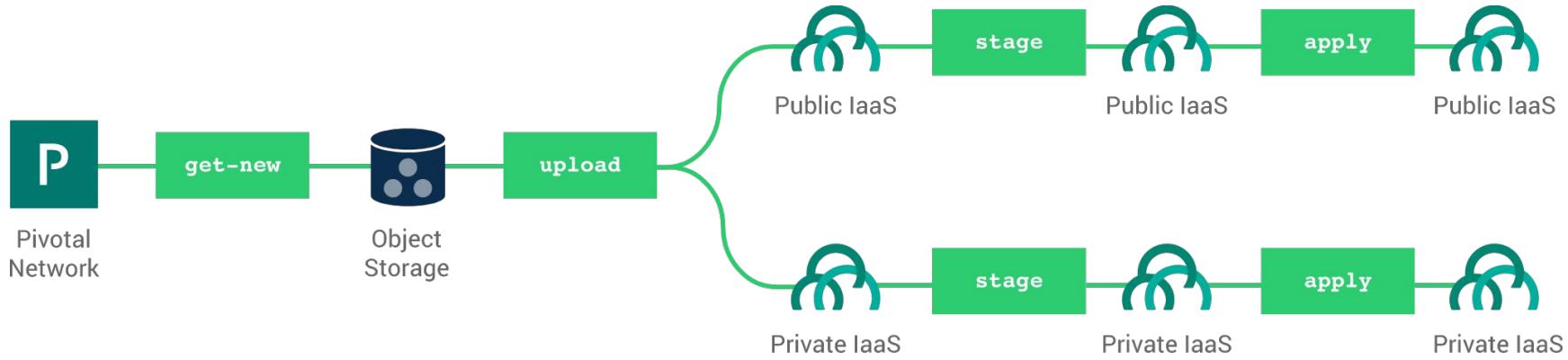
Cloud Foundry BOSH

An open source tool chain for release engineering, deployment, and lifecycle management of large scale distributed services.



- Packaging w/ embedded OS
- Server provisioning on any IaaS
- Software deployment across clusters
- Health monitoring (server AND processes)
- Service state monitoring
- Self-healing w/ Resurrector
- Storage management
- Rolling upgrades via canaries

Zero Downtime Updates for OS, CVEs, Patches with CF BOSH



CredHub Mitigates the Risk of Leaked Credentials

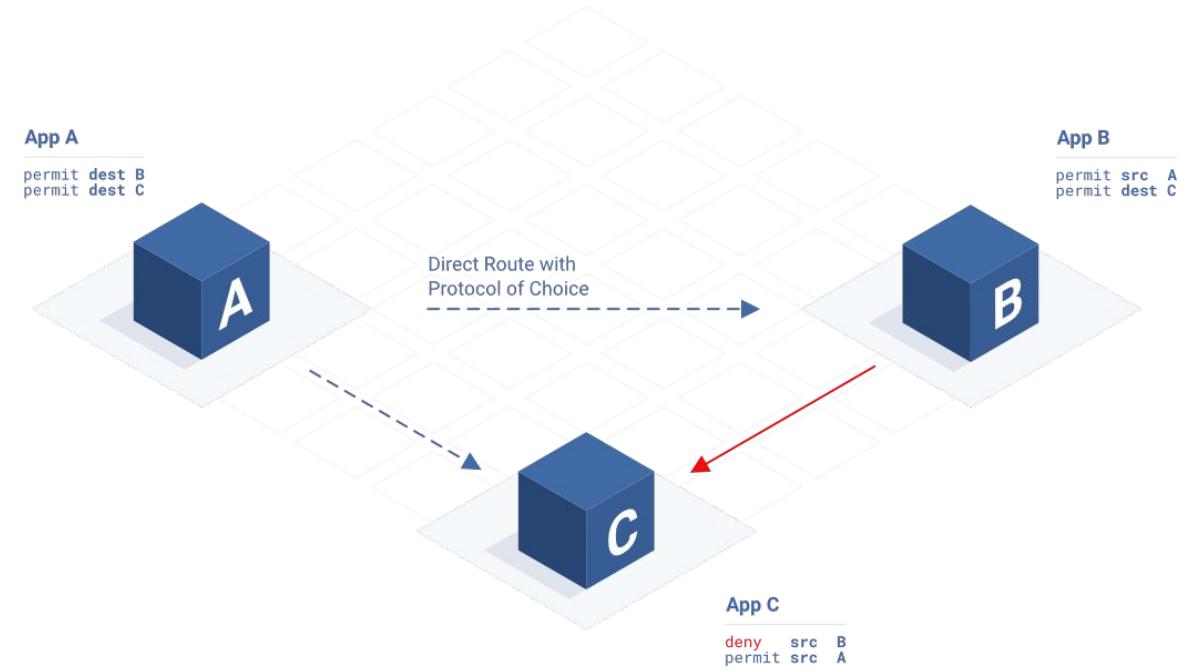


CredHub delivers centralized management of platform and application creds. (Rotation in the future.)

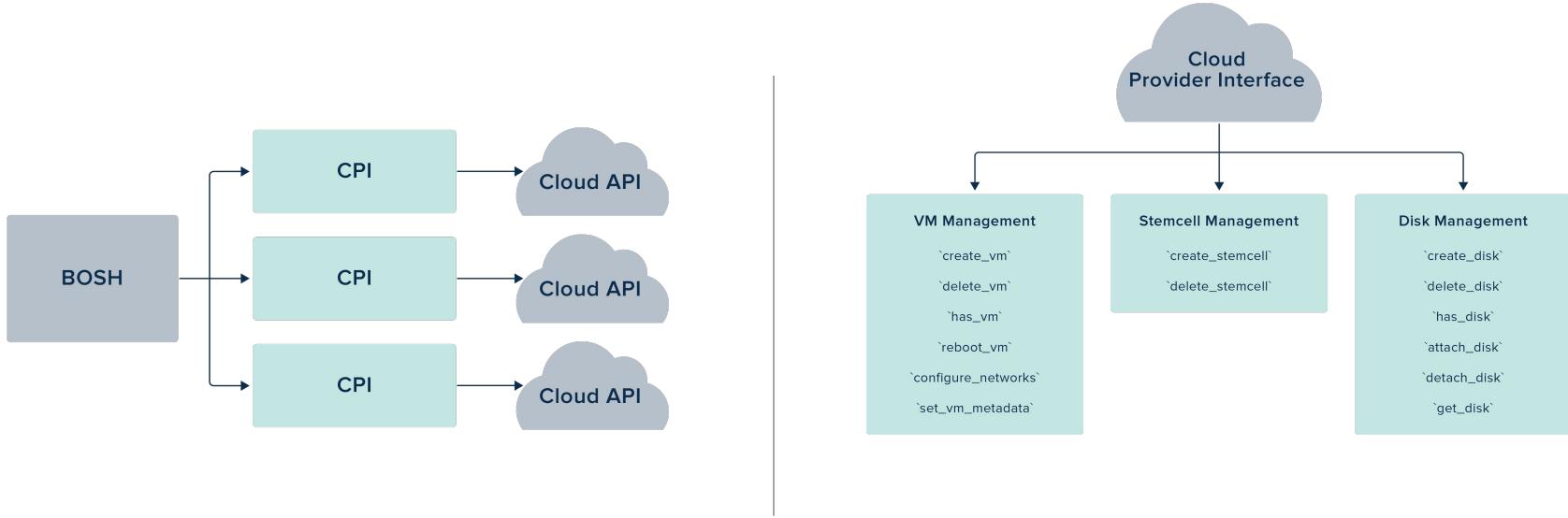
- Credentials are the bedrock for trust in the cloud.
- CredHub's goal: deliver cradle-to-grave management of credentials (create, access control, distribution, rotation, logging)
- Manages passwords, certificates, ssh keys, RSA keys, and arbitrary values (strings and JSON blobs).
- All credentials are encrypted w/a key that rotates (HSM support in OSS, coming soon to PCF)

PCF + NSX

App-defined network policies, with container networking



Multi-Cloud with BOSH + CPI



vmware®

openstack.

Azure

aws

Google Cloud Platform

Pivotal Application Service (PAS): A Runtime for Apps



Increase speed and deploy code to production thousands of times per month. Use PAS to run Java, .NET, and Node apps.

Best runtime for Spring and Spring Boot — Spring's microservice patterns—and Spring Boot's executable jars—are ready-made for PAS.

Turnkey microservices operations and security — Spring Cloud Services brings microservices best practices to PAS. It includes Config Server, Service Registry, and Circuit Breaker Dashboard.

A native Windows and .NET experience — Use PAS to run new apps built with .NET Core. Run your legacy .NET Framework apps on PAS too, using the .NET Hosted Web Core buildpack. Push applications to containers running on Windows Server 2016.

Built for apps — PAS has everything to need to run apps. Buildpacks manage runtime dependencies; metrics, logging, and scaling are done for you. Multitenancy, and blue/green deployment patterns are built-in. Extend apps with a rich service catalog.

Container-ready — PAS supports the OCI format for Docker images. Run platform-built and developer-built containers.

Pivotal Application Service 2.0

New features

- PCF Healthwatch: Dashboards + KPIs to monitor platform health
- Azure Stack [beta]
- Deeper NSX-T integration (container networking, security groups, isolation segments, vSphere CPI)
- Gorouter/HAProxy support multiple operator managed TLS certs + SNI
- PCF Scheduler: cron for Tasks
- Apps Manager: Greater parity with cf cli
- Windows 2016 containers

PAS for Windows

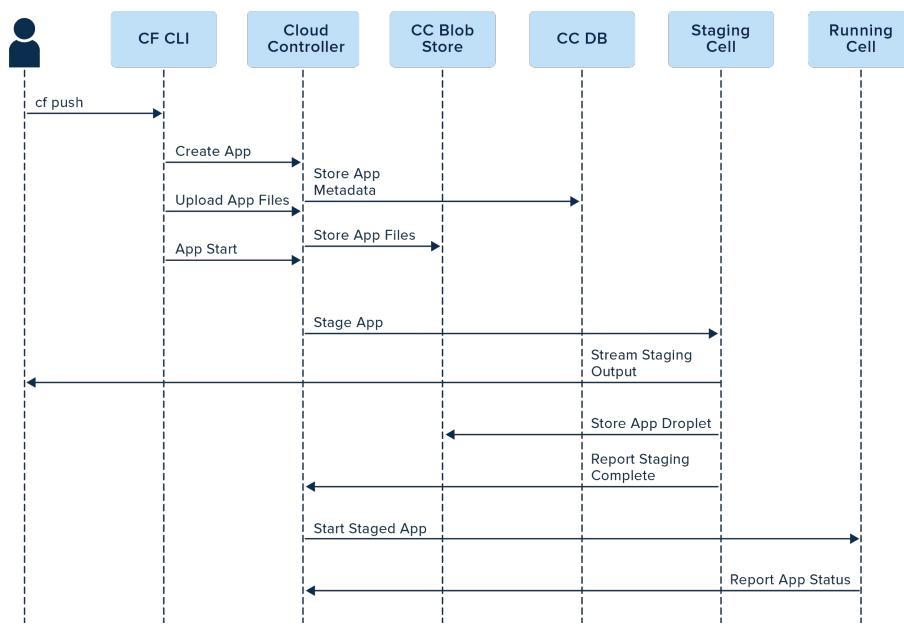
Includes Windows Server
Containers offered on Windows
Server 2016

- Windows 2016 Runtime Tile
- Support for Windows Server Core stemcells
- Unlocks the potential for the newest PCF platform features for an on-par experience with Linux applications: CPU + network limits, CPU-based autoscaling, CredHub encrypted service creds, Diego SSH (i.e. cf ssh)
- Made available via a new OpsManager tile for a smooth upgrade experience from 2012R2 to 2016

Pivotal Application Service

Key Concepts

cf push Automates Developer + Operator workflows



After you `cf push`, PAS:

- Uploads your code
- Detects and installs required runtime & middleware (“Buildpacks”)
- Sets up a route (or URL)
- Creates a load balancing entry
- Creates SSL termination
- Creates health monitoring & logging subsystems
- Starts your app in a healthy state, with the desired number of instances
- Binds specified backing services



Pivotal Application Service: Deployment Models



Zero Downtime

**Make Releases
Non-Events**

One Command



Blue-Green

**A/B Test Competing
Ideas**

Four Commands

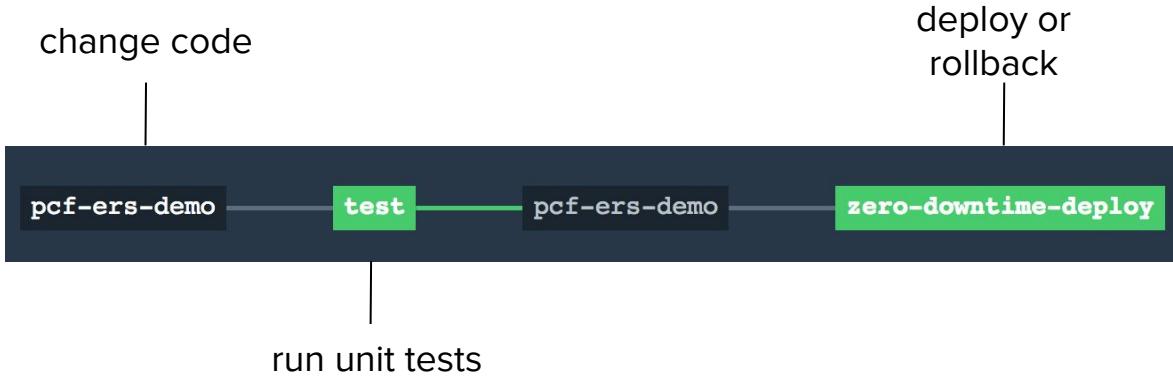


Canary

**Rolling Deployments
Minimize Risk**

Six Commands

Zero Downtime: One Command



1. **ARCHIVE V1**
2. **DEPLOY V2**
3. **URL ROUTE V2**
4. **URL UN-ROUTE V1**
5. **DELETE V1**

Blue | Green: Four Commands



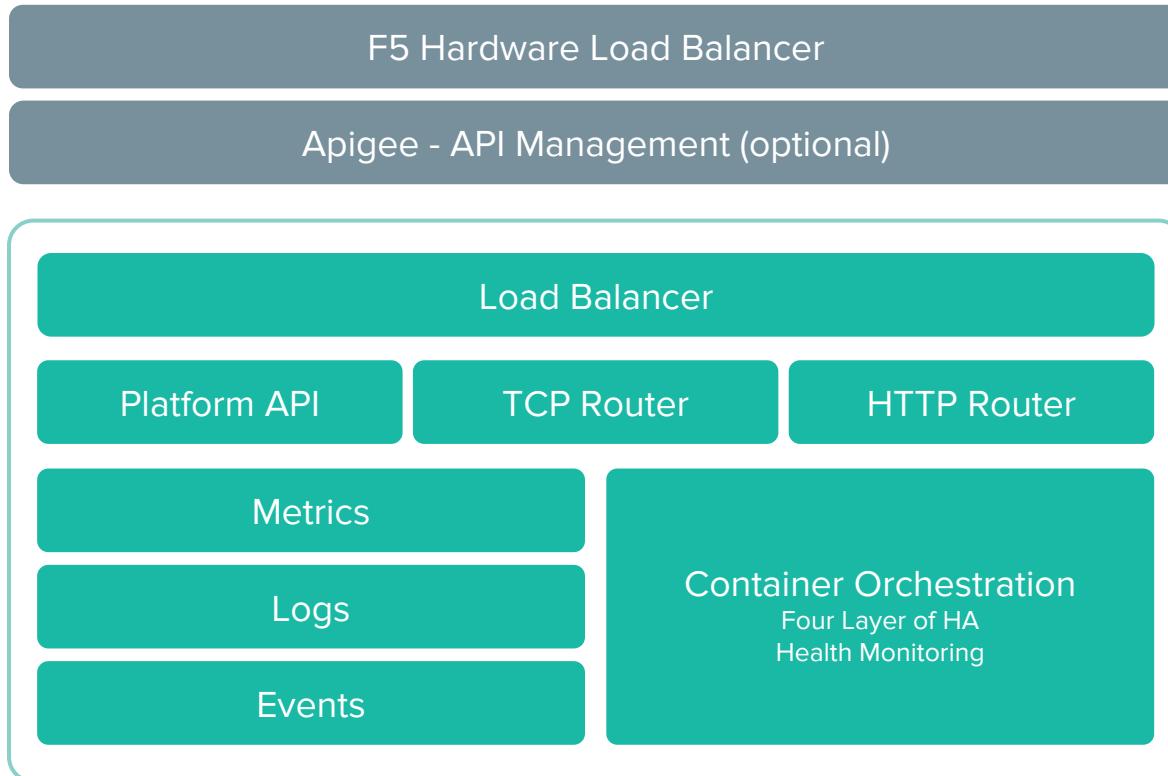
1. **DEPLOY A | DEPLOY B**
2. **URL ROUTE A | URL ROUTE B**
3. **ANALYZE METRICS**

Canary: Six Command

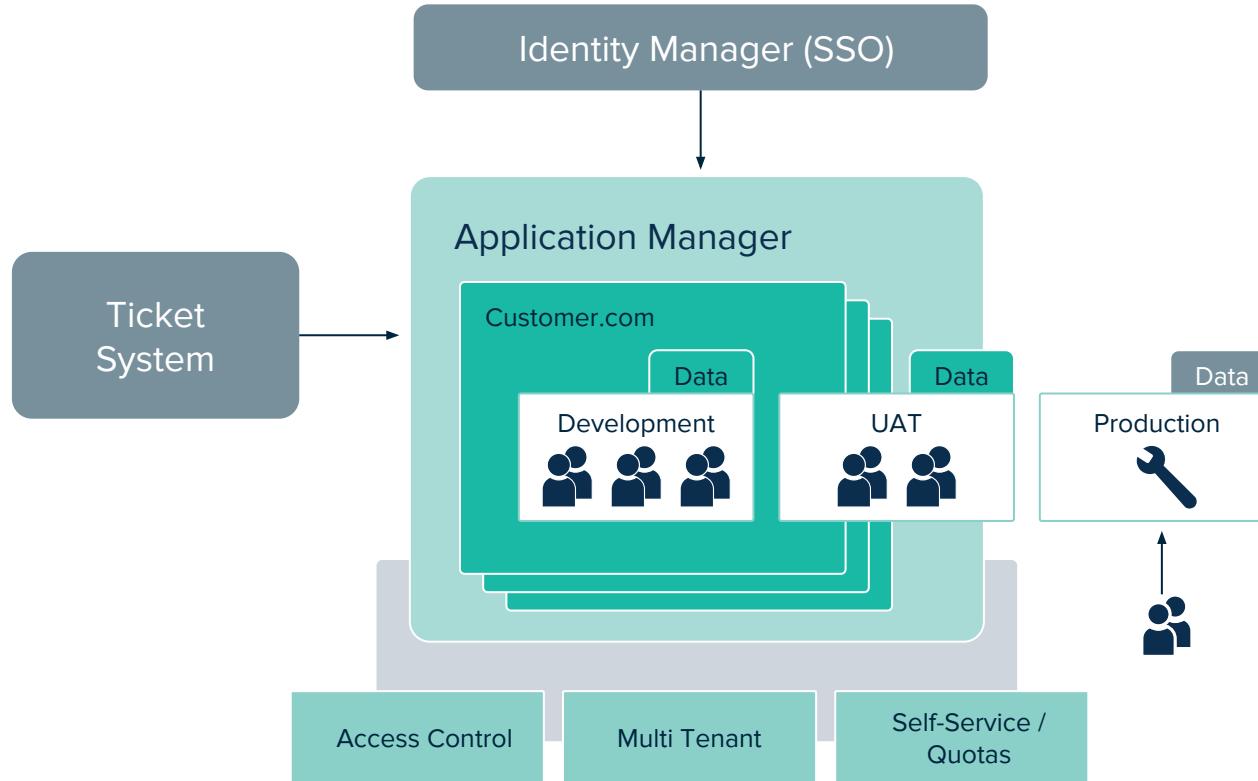


- 1 **DEPLOY V2**
- 2 **UAT V2**
- 3 **{ROLLOUT} SCALE V2 | {ROLLBACK} DEBUG V2**
- 4 **URL UN-ROUTE V1**
- 5 **DELETE V1**

Pivotal Application Service: Network Automation



Pivotal Application Service: Multi-tenancy



Pivotal Container Service (PKS): A Runtime for Containers

A turnkey solution to provision,
operate and manage enterprise
grade Kubernetes clusters

Pivotal[®]

+

vmware[®]

+

Google

Kubernetes Dial Tone:

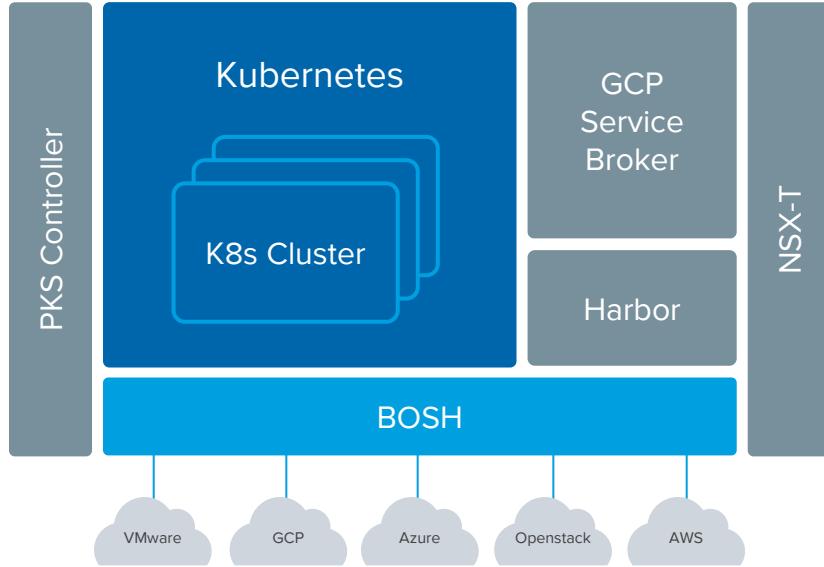
- Health management
- Aggregated Metrics and Logging
- Autoscaling
- Persistence interface

Control Plane:

- Provisioning Engine
- T-shirt sized clusters
- Self-service Clusters
- Software Update Automation
- Load balancing
- Networking
- Multi-tenancy



Pivotal Container Service™



Built with open-source Kubernetes — Constant compatibility with the current stable release of Kubernetes, operated by BOSH. No proprietary extensions.

Production-ready — Highly available from apps to infrastructure, no single points of failure. Built-in health checks, scaling, auto-healing and rolling upgrades.

Multicloud — BOSH provides a reliable and consistent operational experience. For any cloud.

Network management and security out-of-the-box with VMware NSX-T. Multi-cloud, multi-hypervisor.

GCP APIs access — The GCP Service Broker allows apps to transparently access Google Cloud APIs, from anywhere. Easily move workloads to/from Google Container Engine (GKE).

Fully automated Ops — Fully automated deploy, scale, patch, upgrade. No downtime. Use CD pipelines to deploy your platform, too.

Pivotal Function Service (PFS): A Runtime for Functions



Execute functions in response to events. Use PFS to handle web events, event-based integration, and large scale streaming data.

Trigger functions via HTTP/Message Broker — PFS is architected to support event stream processing, connecting to message topics via a language-neutral, function container interface.

Run functions anywhere — PFS lets you easily run functions on-premises and in the public cloud for maximum flexibility.

Use modern DevOps workflows — PFS allows you to use familiar, container-based workflows for serverless scenarios.

Pluggable event brokers — PFS can be connected easily with popular message brokers such as Kafka, RabbitMQ, Google Pub/Sub, and AWS Kinesis.

Polyglot — PFS supports the authoring of functions in your chosen framework - Node.js, Spring/Java, or Shell.

Kubernetes Native — PFS runs natively on top of Kubernetes, making it easy to trigger code or containers in response to events.

App Platforms + Functions

Comparing the experience for Devs + Ops

App Platform Abstraction	Functions Abstraction
runs apps	runs functions
push app to deploy server	register function and bind it to a trigger
server runs and waits for requests	function doesn't run until triggered
server listens to network	platform deploys and invokes functions
server handles lots of requests	functions handle events and then go away
scale out manually or by policy	auto-scale based on concurrent event load
pay per instance	pay per use – time & memory

Brokered Services from Pivotal

Web Events

- Website back-end services like form post handlers, authentication, tracking and logging.
- APIs to back-end data services for mobile and web apps e.g GraphQL
- Webhook handlers
- Chat integrations
- Digital assistant services e.g. Alexa skills

Event-based Integration

- Scheduled tasks, ETL
- File processing e.g. images and videos
- Security scanning
- Complex Event Processing and Change Data Capture
- Monitoring, notifications and alerting
- Custom auth e.g. via API Gateway

Stream Processing

- IoT streams
- Log ingestion
- Event streams e.g. with Kinesis
- ML pipelines

Extend Apps with Brokered Services from Pivotal



MySQL for PCF

- Enterprise-ready MySQL for your developers
- Automate database operations in developer workflows
- **NEW:** Leader-follower for multi-site HA



Pivotal Cloud Cache

- High performance, in-memory, data at scale for microservices
Look-aside caches & HTTP session state caching
- **NEW:** WAN replication



RabbitMQ for PCF

- Easily connect distributed applications with the most widely deployed open source message broker
- Enable connected scalable, distributed applications
- **NEW:** On-demand clusters



Redis for PCF

- In-Memory cache and datastore, configured for the enterprise
- Efficient provisioning matched to use cases

BOSH Managed

| On-Demand Provisioning

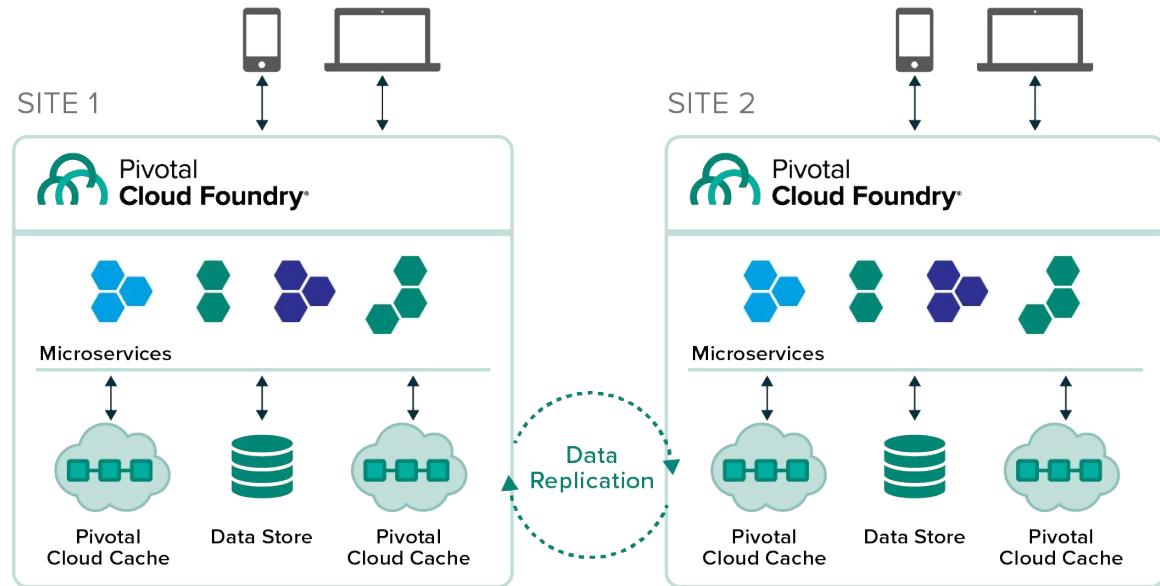
| Dedicated Instances

| Custom Service Plans

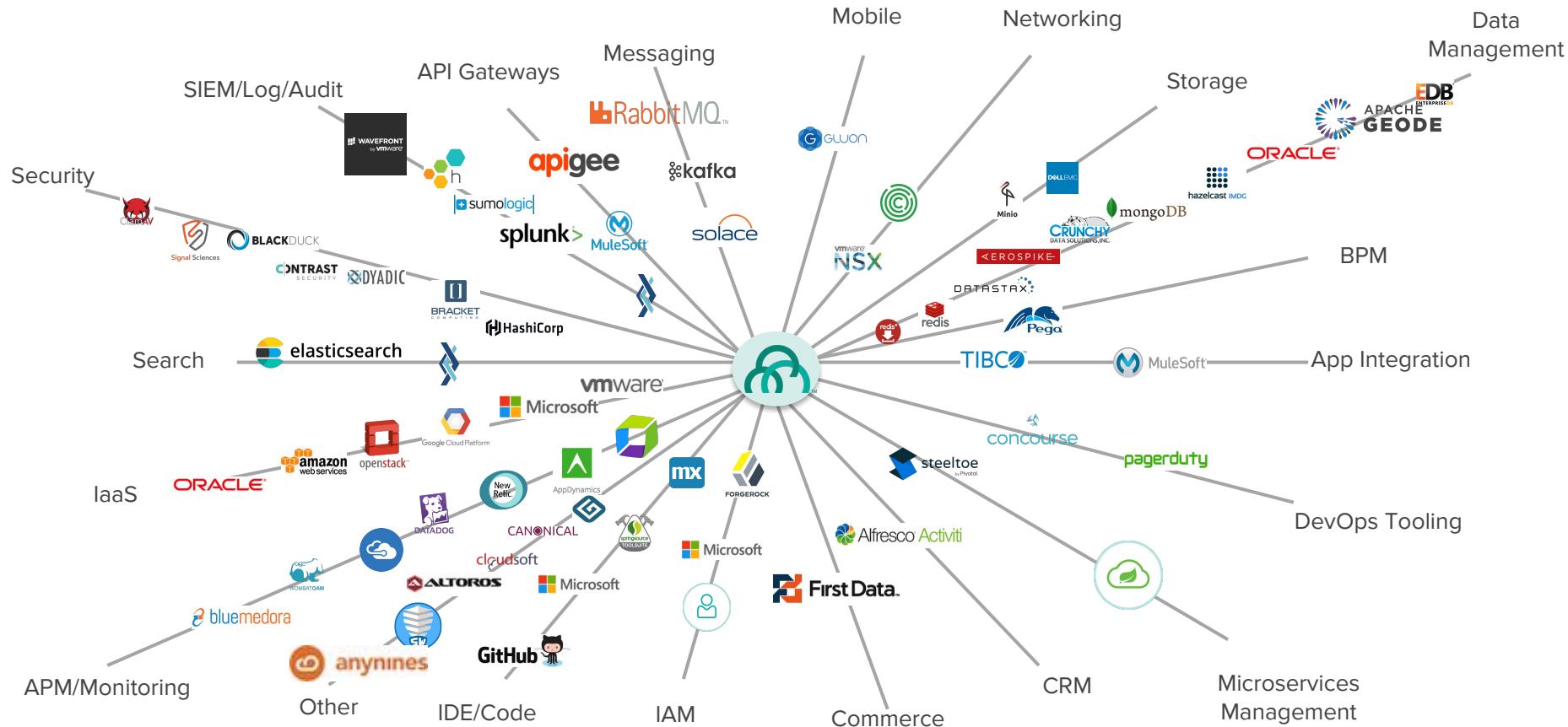
Pivotal Cloud Cache 1.3

WAN Replication

Release Target: Dec 2017



PCF Ecosystem

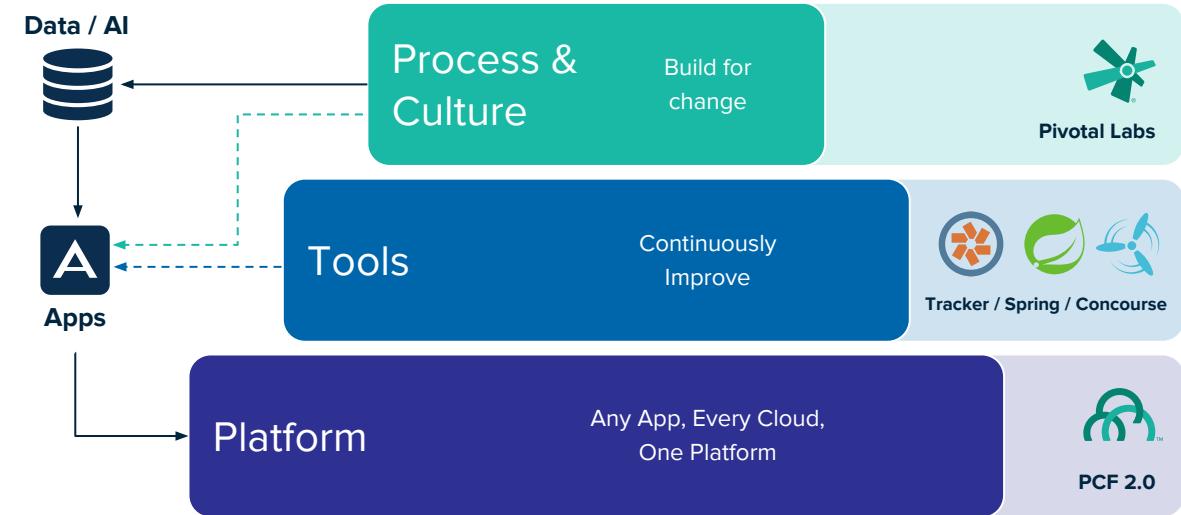




Competitive Landscape

Everything you need to transform

Culture, tools, and platform





Pivotal®

Transforming How The World Builds Software