

A dark, atmospheric photograph of the Golden Gate Bridge in San Francisco, viewed from a low angle looking down the length of the bridge towards the foggy horizon. The bridge's iconic towers and suspension cables are visible, and the water below is dark and calm.

Pivotal

PCF Operations Workshop - Introduction

<Presenter>

<Title>

Operations Workshop Agenda

- PCF Introduction
- Services Overview
- Platform Installation & Setup
- Role Based Access Control
- Platform & Application Scaling
- Platform & Application Health
- Patching & Updates
- Security
- Advanced BOSH

Labs Overview

- Lab 1 – Application Deployment
- Lab 2 – PCF Services
- Lab 3 – Role Based Access Control
- Lab 4 – Scaling
- Lab 5 – Application Monitoring & Health
- Lab 6 – Blue Green Deployment

Operations Workshop Agenda

- **PCF Introduction**
- Services Overview
- Platform Installation & Setup
- Role Based Access Control
- Platform & Application Scaling
- Platform & Application Health
- Patching & Updates
- Security
- Advanced BOSH

What is Pivotal Cloud Foundry?



Pivotal **Cloud Foundry**[®]

Pivotal Cloud Foundry is a **platform** for developing and running cloud applications

Cloud Foundry Principles



Developer Productivity



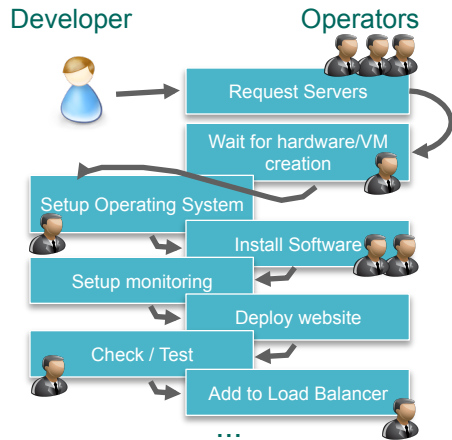
Operational Efficiency



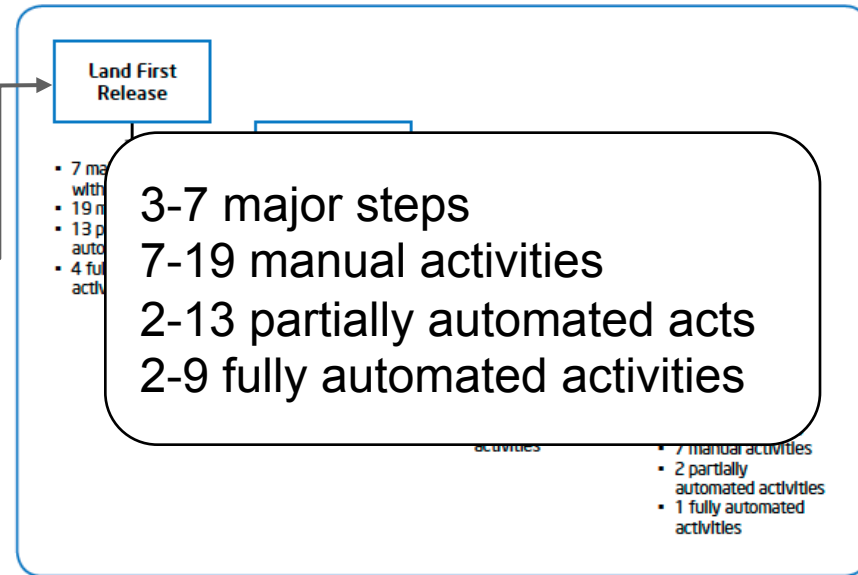
Enhanced Security Posture

Traditional App Lifecycle: Lengthy & Complex

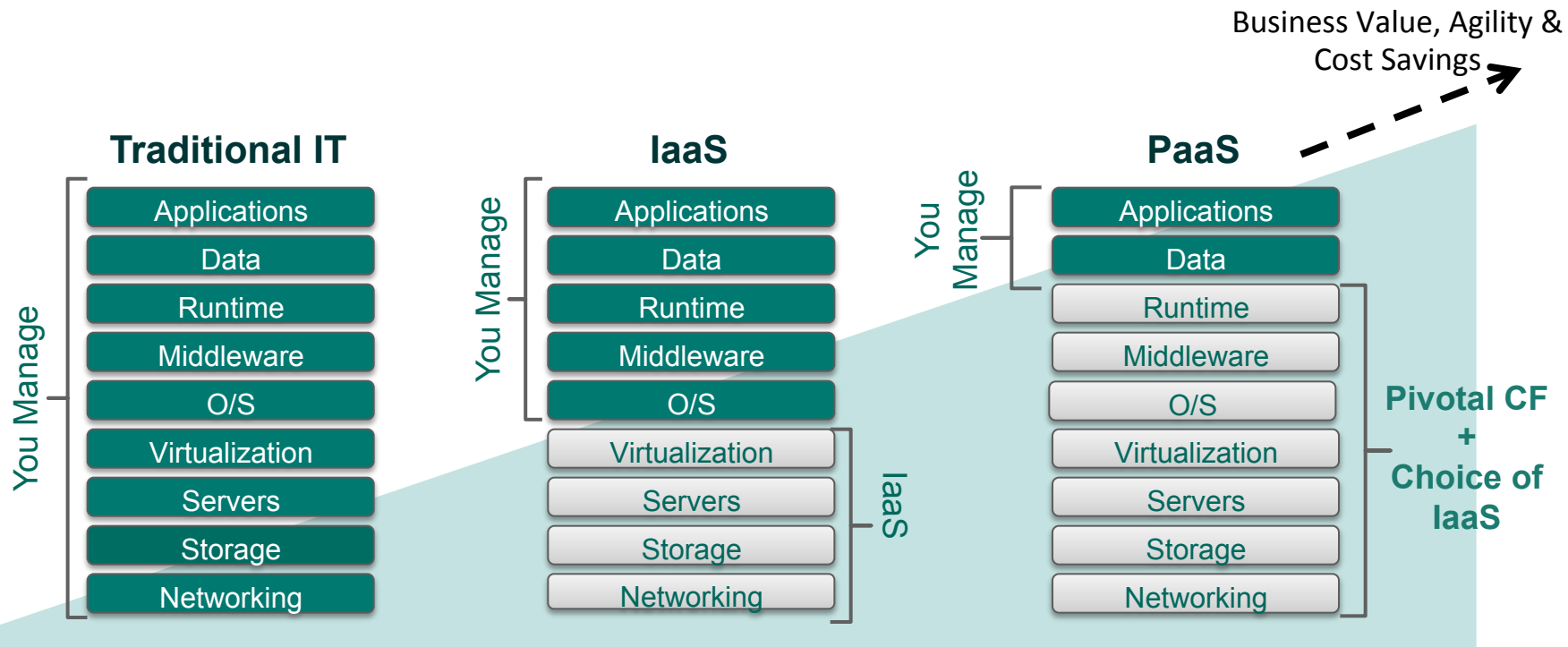
Traditional App Deployment:
8+ manual steps, developers
out of the workflow



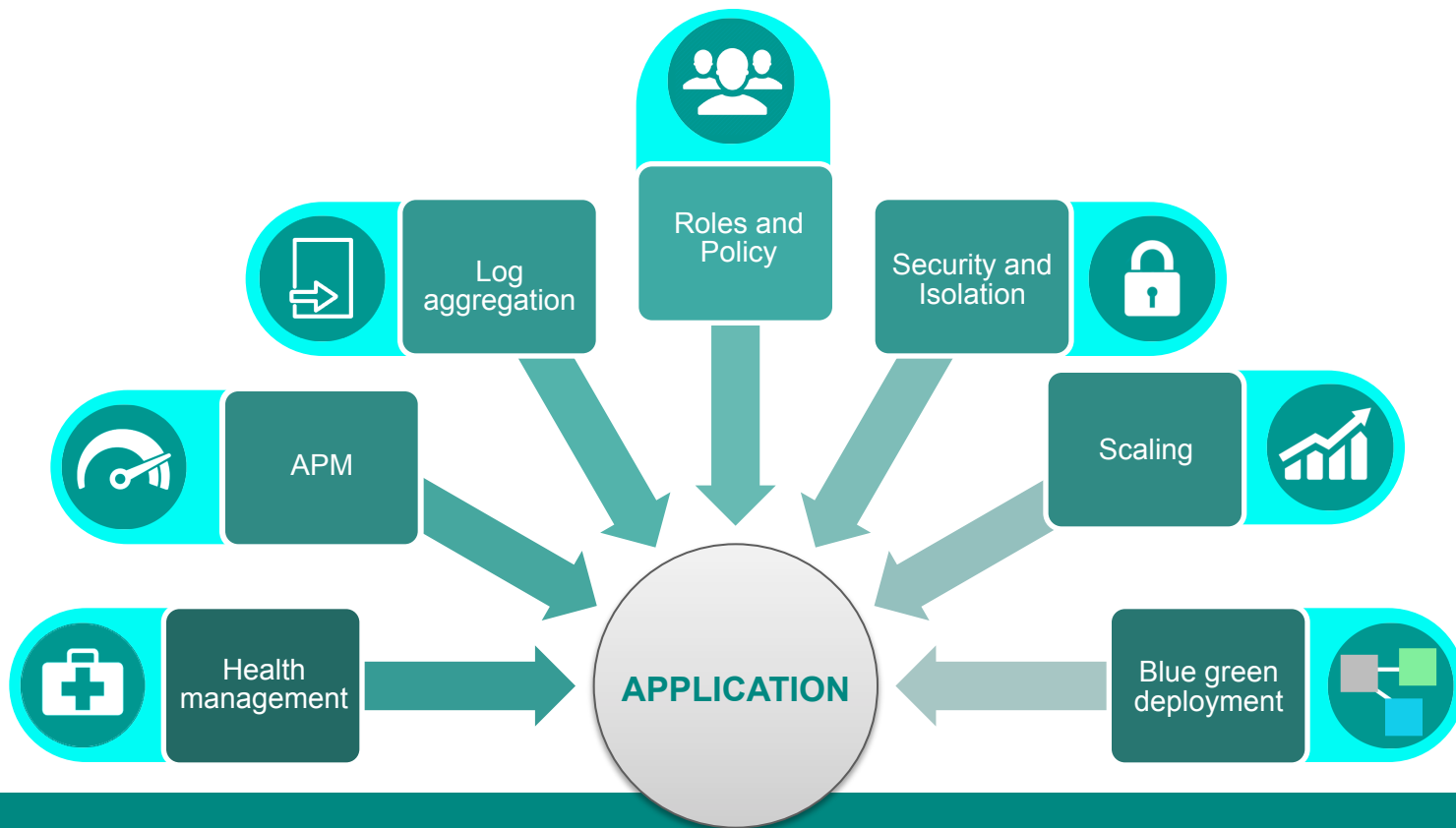
Traditional App Lifecycle:
Each phase compounds time and
complexity, **sacrificing agility**



The Power of PaaS



Everything to Deploy and Manage the App



Pivotal Cloud Foundry Architecture

Platform Runtime (Elastic Runtime)

Routing

Container Scheduling

Service Discovery

Configuration

Application Framework

12 Factor Apps

Microservices

RESTful
Services

Circuit Breakers

Spring Boot

Ruby on Rails

Node.js

.NET

Logging & Metrics

Messaging

Services

Database

Big Data

Object Storage

Mobile

Build CI

User Provided

Operations

Zero Downtime
Deployments

Failover &
Recovery

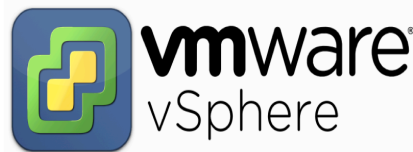
Scaling

Security
Patching

Platform
Upgrades

Pivotal

Deployment Flexibility



VMware vCloud® Air™



Google Cloud Platform



Extensive Support



Pivotal Cloud Foundry Includes

- Pivotal Cloud Foundry Elastic Runtime
- OS Entitlement
- OSS Support for Spring
- OSS Support for Apache Tomcat
- Session State Caching Powered by GemFire for PCF
- Spring Cloud Services for PCF
- Pivotal tc Server
- Single Sign-on for PCF
- Push Notification for PCF
- App Distribution for PCF
- Entitlement to Push App Instances to Pivotal Web Services

The Developer Dream Haiku

Here is my source code,

Run it in the Cloud for me,

I do not care how

Interacting with Cloud Foundry

```
1. bash

NAME:
  cf - A command line tool to interact with Cloud Foundry

USAGE:
  [environment variables] cf [global options] command [arguments...] [command options]

VERSION:
  6.15.0+fa1bfe2-2016-01-13

GETTING STARTED:
  help          Show help
  login         Log user in
  logout        Log user out
  passwd        Change user password
  target        Set or view the targeted org or space

  api           Set or view target api url
  auth          Authenticate user non-interactively

APPS:
  apps          List all apps in the target space
  app           Display health and status for app

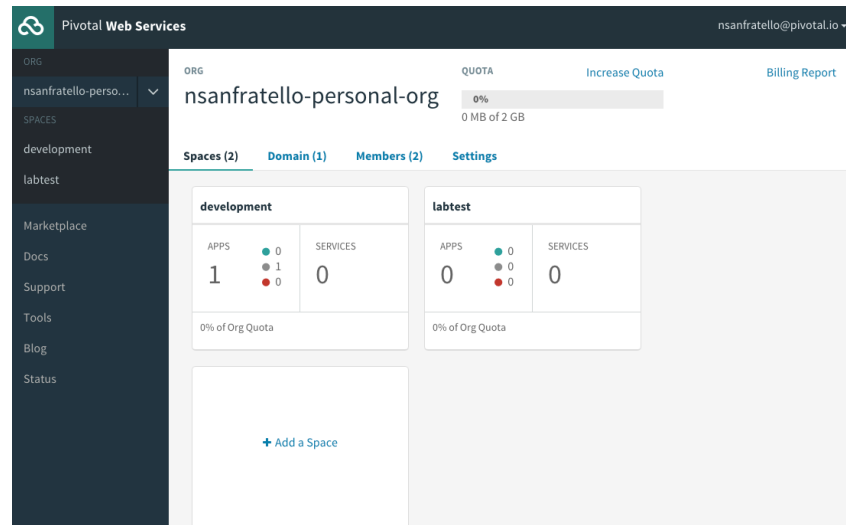
  push          Push a new app or sync changes to an existing app
  scale         Change or view the instance count, disk space limit, a
nd memory limit for an app
  delete        Delete an app
  rename        Rename an app

  start        Start an app
```

The screenshot shows the Pivotal Web Services dashboard. On the left is a dark sidebar with navigation links: ORG, SPACES, development, labtest, Marketplace, Docs, Support, Tools, Blog, and Status. The main content area is for the 'nsanfratello-personal-org' organization. It displays a 'QUOTA' section with a progress bar at 0% (0 MB of 2 GB) and a link to 'Increase Quota'. Below this are tabs for 'Spaces (2)', 'Domain (1)', 'Members (2)', and 'Settings'. Two space cards are visible: 'development' and 'labtest'. Each card shows 'APPS' and 'SERVICES' counts with status indicators (green, grey, red dots). The 'development' space has 1 app (green) and 0 services (grey). The 'labtest' space has 0 apps (green) and 0 services (grey). Both spaces show '0% of Org Quota'. At the bottom of the 'development' space card is a '+ Add a Space' button.

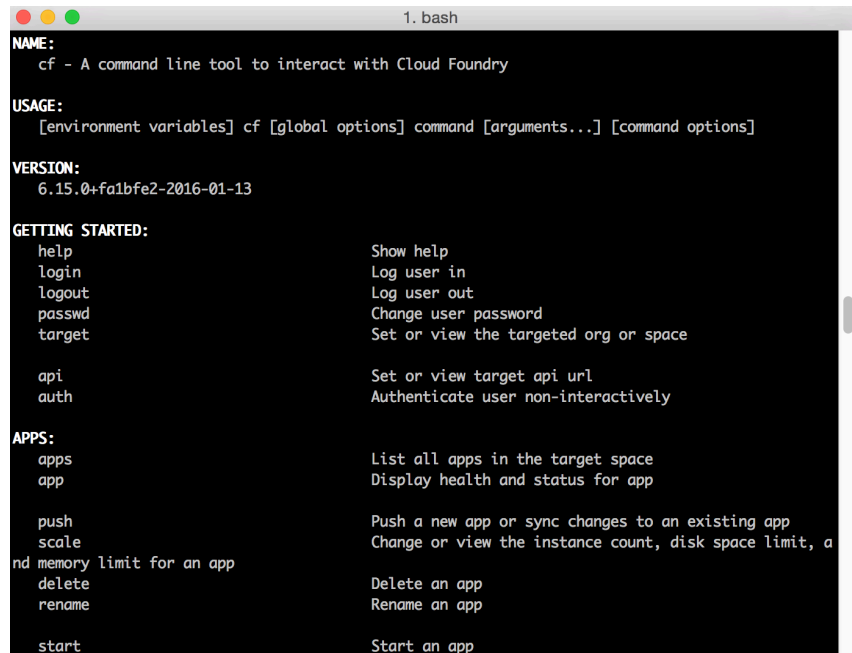
Platform Runtime – Apps Manager

- Manage organizations, users, applications and spaces
- Monitor applications logs, services, and routes
- Access Service Marketplace, create services and bind to applications



CLI

- Command line utility providing easy access to the Pivotal CF commands.
- Scriptable
- Fully documented via the “cf help” command



```
1. bash
NAME:
  cf - A command line tool to interact with Cloud Foundry

USAGE:
  [environment variables] cf [global options] command [arguments...] [command options]

VERSION:
  6.15.0+fa1bfe2-2016-01-13

GETTING STARTED:
  help          Show help
  login         Log user in
  logout        Log user out
  passwd        Change user password
  target        Set or view the targeted org or space

  api           Set or view target api url
  auth          Authenticate user non-interactively

APPS:
  apps          List all apps in the target space
  app           Display health and status for app

  push          Push a new app or sync changes to an existing app
  scale         Change or view the instance count, disk space limit, a
nd memory limit for an app
  delete        Delete an app
  rename        Rename an app

  start         Start an app
```

Cloud Controller API

- Cloud Controller component of Elastic Runtime manages all Cloud Foundry APIs
- CF CLI and other clients like Apps Manager directly call this API
- Before accessing the CC API, you must get an access token from the ERS component called User Account and Authentication (UAA) server
- <http://apidocs.cloudfoundry.org>

[Home](#) [Version 238 - CC API VERSION 2.57.0](#) [Version 3 - Experimental](#)

Cloud Foundry API

App Usage Events

- List all App Usage Events
- Purge and reseed App Usage Events
- Retrieve a Particular App Usage Event

Apps

- Associate Route with the App
- Copy the app bits for an App
- Creating a Docker App (experimental)
- Creating an App
- Delete a Particular App
- Downloads the bits for an App
- Downloads the staged droplet for an App
- Get App summary
- Get detailed stats for a STARTED App
- Get the env for an App
- Get the instance information for a STARTED App
- List all Apps
- List all Routes for the App
- List all Service Bindings for the App
- Remove Route from the App
- Remove Service Binding from the App
- Restage an App
- Retrieve a Particular App
- Terminate the running App Instance at the given index
- Updating an App
- Uploads the bits for an App

Blobstores

Deploying Apps Shouldn't Be Painful

Traditional

Create a VM

Install JDK

Install Tomcat

Deploy Application

Configure Load Balancer

Configure DNS

Configure Firewall

Configure Monitoring

Configure Logging

Pivotal Cloud Foundry

`cf push`

Deploying an App

```
my-php-app88 — bash — 90x26
greylag:my-php-app88 sbyrnes$ cf push "my-php-app88"
Updating app my-php-app88 in org sbyrnes-org / space development as sbyrnes@pivotal.io...
OK

Uploading my-php-app88...
Uploading app files from: /Users/sbyrnes/Documents/Pivotal Training/files2/my-php-app88
Uploading 335, 1 files
Done uploading
OK

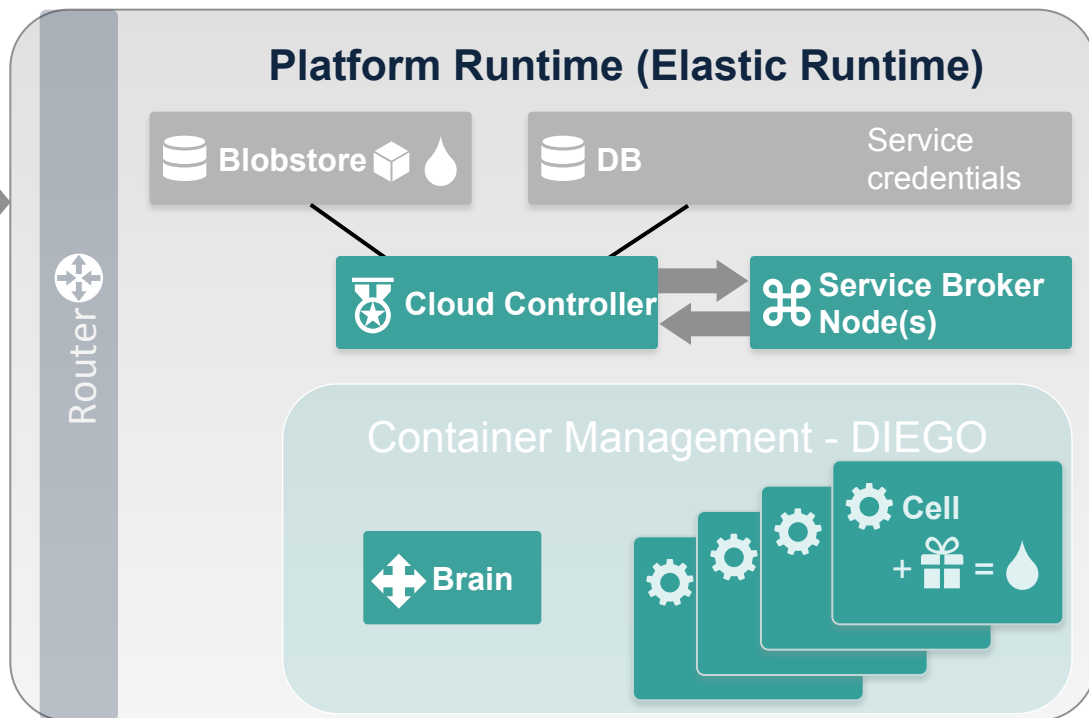
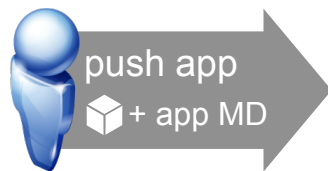
Stopping app my-php-app88 in org sbyrnes-org / space development as sbyrnes@pivotal.io...
OK

Starting app my-php-app88 in org sbyrnes-org / space development as sbyrnes@pivotal.io...
-----> Downloaded app package (4.0K)
-----> Downloaded app buildpack cache (4.0K)
-----> Buildpack version 3.2.1
Installing HTTPD
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 1534k  100 1534k    0     0  451M      0 --:--:-- --:--:-- --:--:--  451M
Downloaded [https://pivotal-buildpacks.s3.amazonaws.com/php/binaries/trusty/httpd/2.4.12/httpd-2.4.12.tar.gz] to [/tmp]
  % Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
                                 Dload  Upload   Total   Spent    Left   Speed
100 14682  100 14682    0     0  168M      0 --:--:-- --:--:-- --:--:--  168M
```

cf push “my-application-name”

Deploying an Application

- ① Upload app bits and metadata
- ② Create and bind services
- ③ Stage application
- ④ Deploy application
- ⑤ Manage application health



Application Manifest

- A client-side file that describes the application deployment options
 - Provides repeatability when pushing applications
 - Same options as `cf push` command
- `cf push` automatically detects “manifest.yml” file in current working directory
 - file name can be overridden with `-f` flag (`cf push -f mymanifest.yml`)

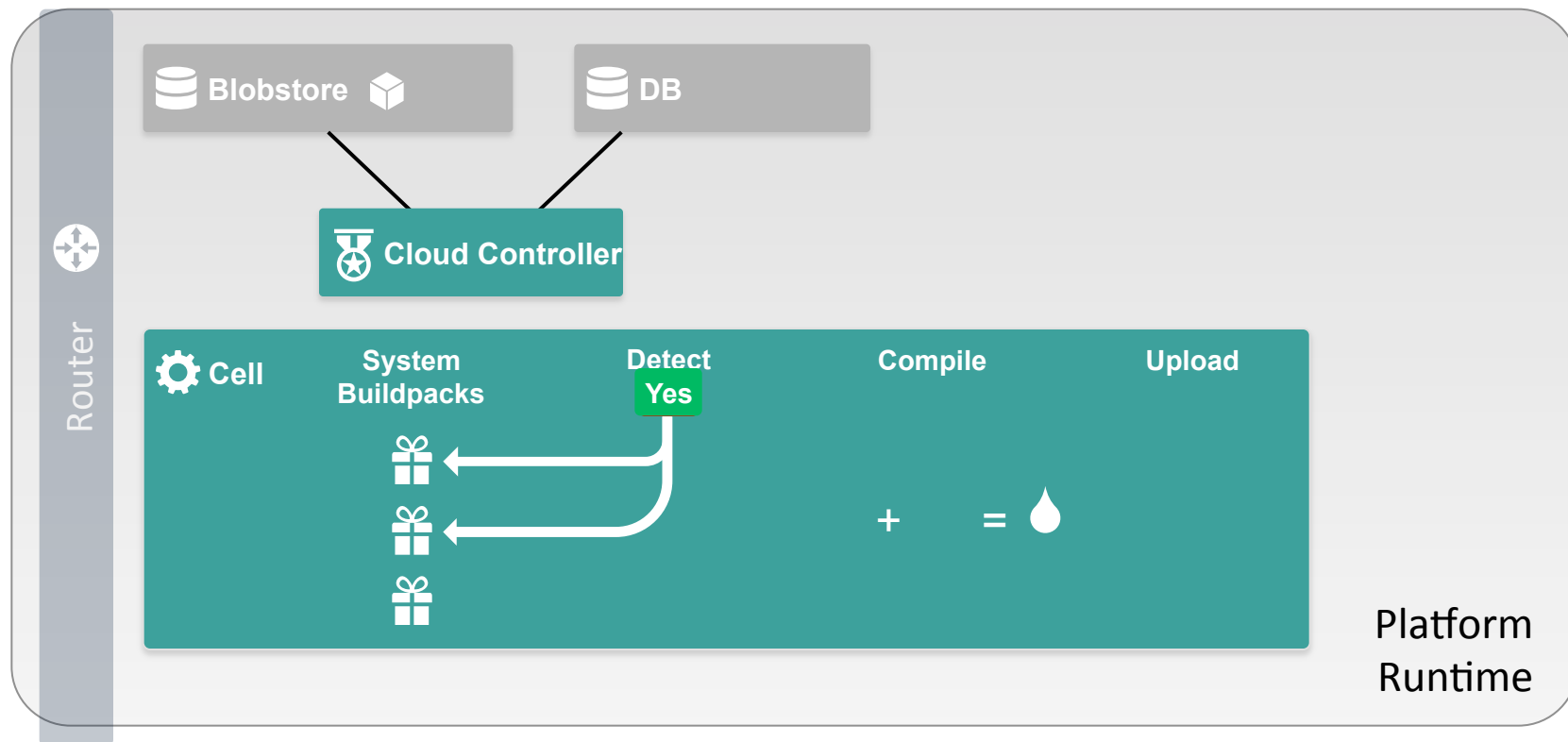
YAML Format

- Three dashes
 - Indicate start of document
- Indent with spaces, not tabs!
 - Determines hierarchy
 - Each indent 2 spaces
 - “-” defines “group”
- property-value pairs
- # starts a one line comment
- See <http://www.yaml.org/spec/1.2/spec.html>

```
---
applications:
- name: nodetestdh01
  memory: 64M
  instances: 2
  host: crn      # comment
  domain: cfapps.io
  # comment

- name: nextapp    # group 2
  memory: 256M
  ...
```

Staging an Application



Buildpacks

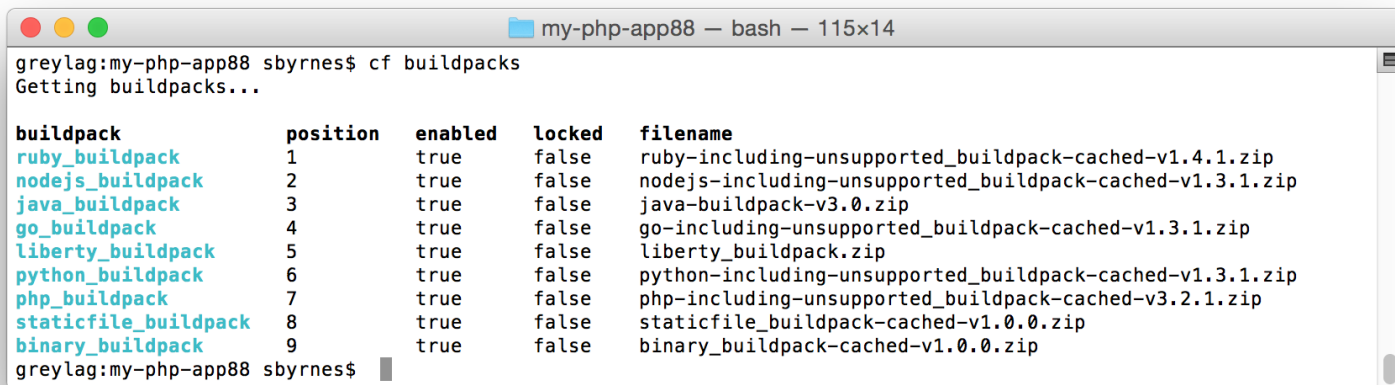
- Buildpacks are used to detect the type of application and set up the proper execution environment
- Buildpacks will download the necessary dependencies.
- Buildpacks are used at staging time (during a cf push or cf restage)
 - They are not involved during the actual execution of the app

Why Buildpacks

- Control what frameworks/runtimes are used on the platform
- Provides consistent deployments across environments
 - Stops deployments from piling up at operation's doorstep
 - Enables a self-service platform
- Eases ongoing operations burdens:
 - Security vulnerability is identified
 - Subsequently fixed with a new buildpack release
 - Restage applications

Viewing the Available Buildpacks

- Use cf buildpacks to view the system/installed buildpacks
- Buildpacks can be written for any language
- Cloud Foundry is language agnostic
- You can also specify a system or custom buildpack with cf push -b



```
my-php-app88 — bash — 115x14
greylag:my-php-app88 sbyrnes$ cf buildpacks
Getting buildpacks...

buildpack      position  enabled  locked  filename
ruby_buildpack 1         true    false   ruby-including-unsupported_buildpack-cached-v1.4.1.zip
nodejs_buildpack 2         true    false   nodejs-including-unsupported_buildpack-cached-v1.3.1.zip
java_buildpack 3         true    false   java-buildpack-v3.0.zip
go_buildpack 4         true    false   go-including-unsupported_buildpack-cached-v1.3.1.zip
liberty_buildpack 5         true    false   liberty_buildpack.zip
python_buildpack 6         true    false   python-including-unsupported_buildpack-cached-v1.3.1.zip
php_buildpack 7         true    false   php-including-unsupported_buildpack-cached-v3.2.1.zip
staticfile_buildpack 8         true    false   staticfile_buildpack-cached-v1.0.0.zip
binary_buildpack 9         true    false   binary_buildpack-cached-v1.0.0.zip
greylag:my-php-app88 sbyrnes$
```