



# Cloud Foundry

Deploying Applications to Cloud Foundry

# RX-M Cloud Native Training

## ▪ Microservice Oriented

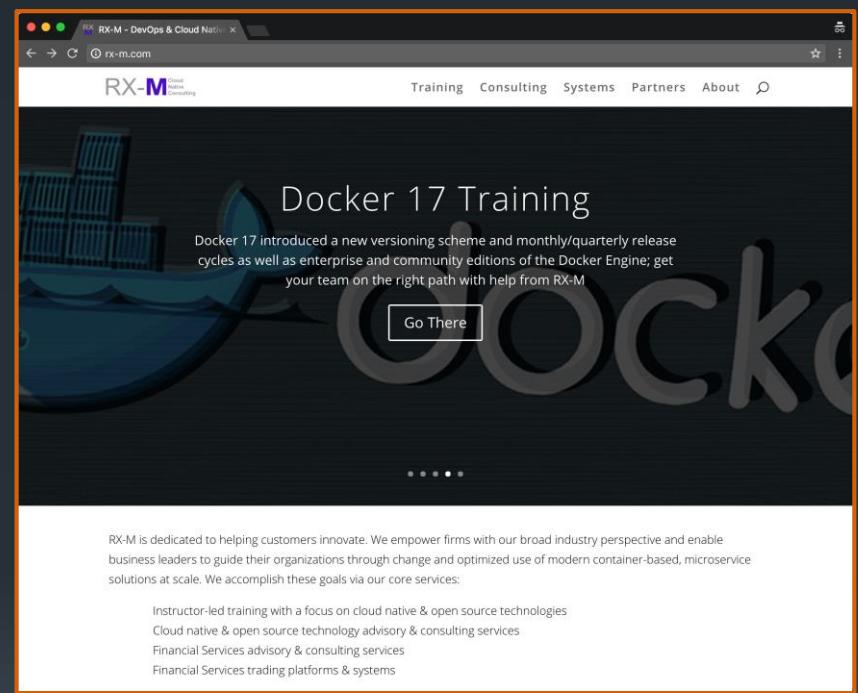
- Microservices Foundation [3 Day]
- Microservices on AWS [3 Day]
- Microservices on Azure [3 Day]
- Microservices on GCP [3 Day]
- Microservices on Bluemix [3 Day]
- Microservices on Oracle Cloud [3 Day]
- Building Microservices with Go [3 Day]
- Building Microservices with Apache Thrift [3 Day]
- Building Microservices with gRPC [3 Day]

## ▪ Container Packaged

- Docker Foundation [3 Day]
- Docker Advanced [2 Day]
- OCI [2 Day]
- Container Technology [2 Day]
- CRI-O [2 Day]

## ▪ Dynamically Managed

- Docker Orch. (Compose/Swarm) [2 Day]
- Kubernetes Foundation [2 Day]
- Kubernetes Advanced [3 Day]
- Kubernetes for Developers [3 Day]
- Mesos Foundation [2 Day]
- Nomad [2 Day]



**RX-M** Cloud  
Native  
Consulting

# Overview

1. Cloud Foundry Container Runtime
2. Cloud Foundry Application Runtime
3. Cloud Foundry Application Runtime Architecture

# Administrative Info

- Format: Lecture/Labs/Discussion
- Schedule: 9:00AM – 5:00PM  
15 minute break, AM & PM  
1 hour lunch at noon  
Lab work after each module
- Location: Fire exits, Restrooms, Security, other matters
- Attendees: Name/Role/Experience/Goals for the Course

# Lecture and Lab

5

Copyright 2013-2019, RX-M LLC

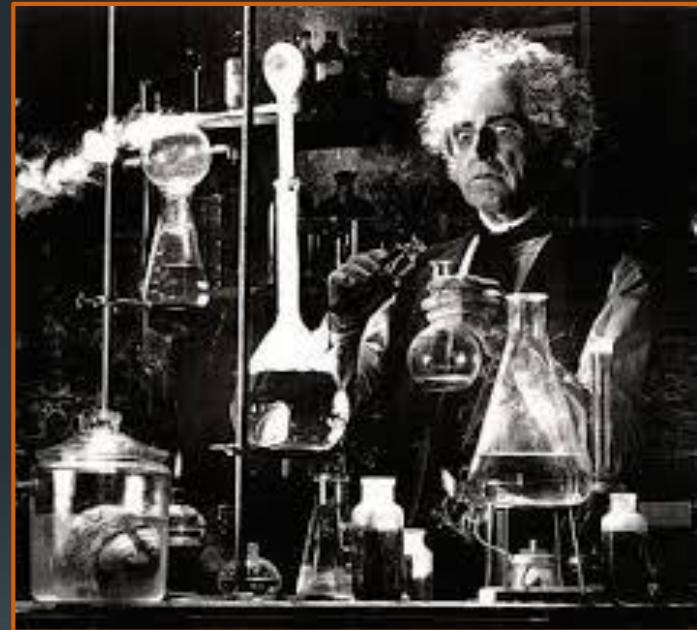
- Our Goals in this class are two fold:

## 1. Introduce concepts and ecosystems

- Covering concepts and where things fit in the world is the primary purpose of the lecture/discussion sessions
- The instructor will take you on a tour of the museum
  - Like a museum tour, you should interact with the instructor (tour guide), ask questions, discuss
  - Like a museum tour, you will not have time to read the slides during the tour, instead, the instructor will discuss and point out the highlights of the slides (exhibits) which will be waiting for you to read in depth later should you like to dig deeper

## 2. Impart practical experience

- This is the primary purpose of the labs
- Classes rarely have time for complete real world projects so think of the labs as thought experiments
  - Like hands on exhibits at the museum

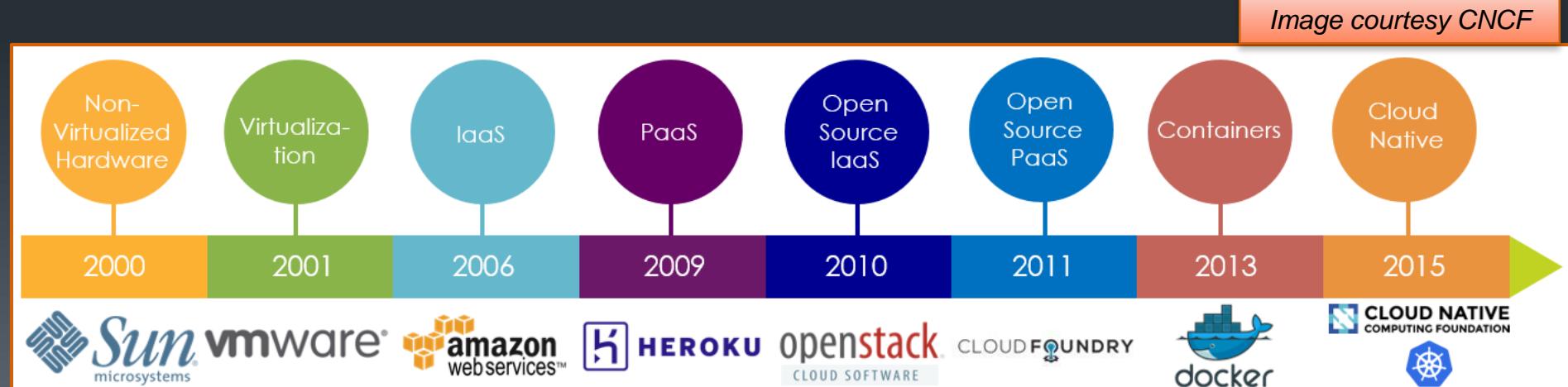


# 1: Cloud Foundry

# Platform Evolution

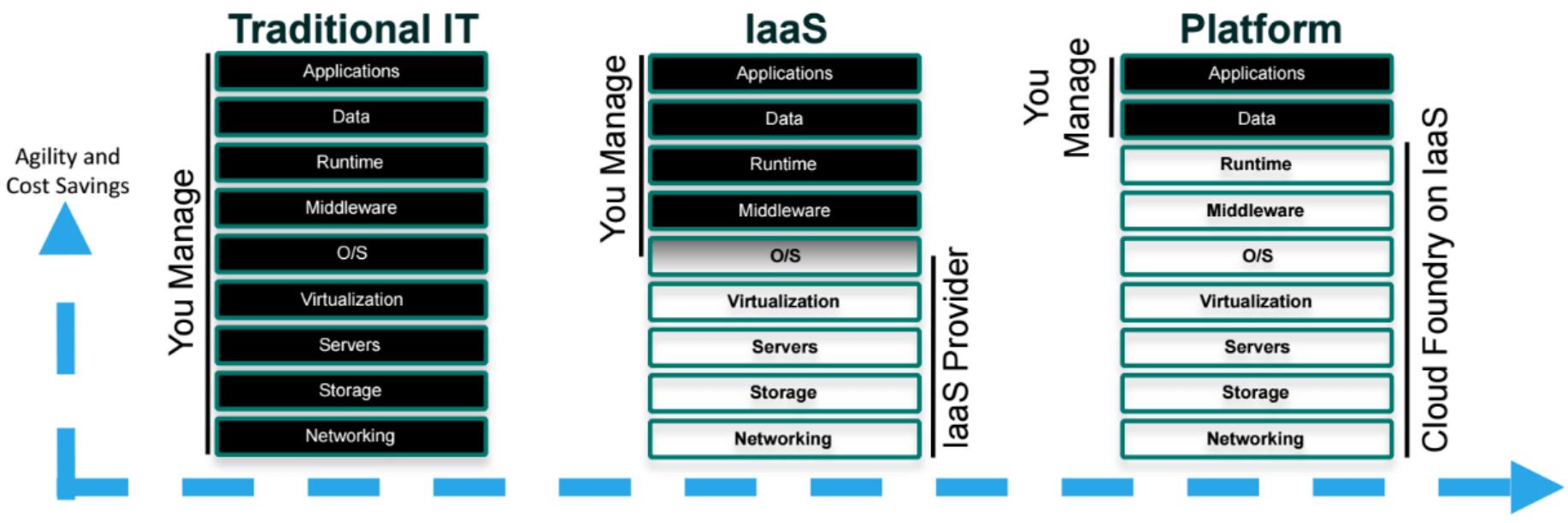
- The platform underlying rapidly growing Internet applications has changed significantly over the years
  - .com era – bare metal servers
  - Web 2.0 – IaaS
  - APIs – PaaS
  - Microservices – Cloud Native

Image courtesy CNCF



# Why PaaS?

- PaaS solution abstract away infrastructure concerns



# Important PaaS

9

Copyright 2017-2018, RX-M LLC

- Public Cloud based
  - AppEngine
    - Because it is Google, first major PaaS offering
  - Heroku/SalesForce
    - The most popular web tech PaaS (Ruby on Rails, Node.js Express, Python Django)
  - Elastic Beanstalk
    - Because it is Amazon and linked to the services of the largest cloud
  - Azure
    - Because it is Microsoft and the cornerstone of .net in the cloud
- Open Source PaaS
  - Cloud Foundry
    - The Java Spring community centric PaaS
  - OpenShift
    - The Java EE community centric PaaS



# The Cloud Foundry Foundation

10

Copyright 2017 - 2019, RX-M LLC

## PLATINUM



## GOLD



GE Digital



AKTIENGESELLSCHAFT

## SILVER



Inspire the Next



JPMORGAN CHASE & CO.



resilient scale



TOSHIBA

# PaaS - CFAR

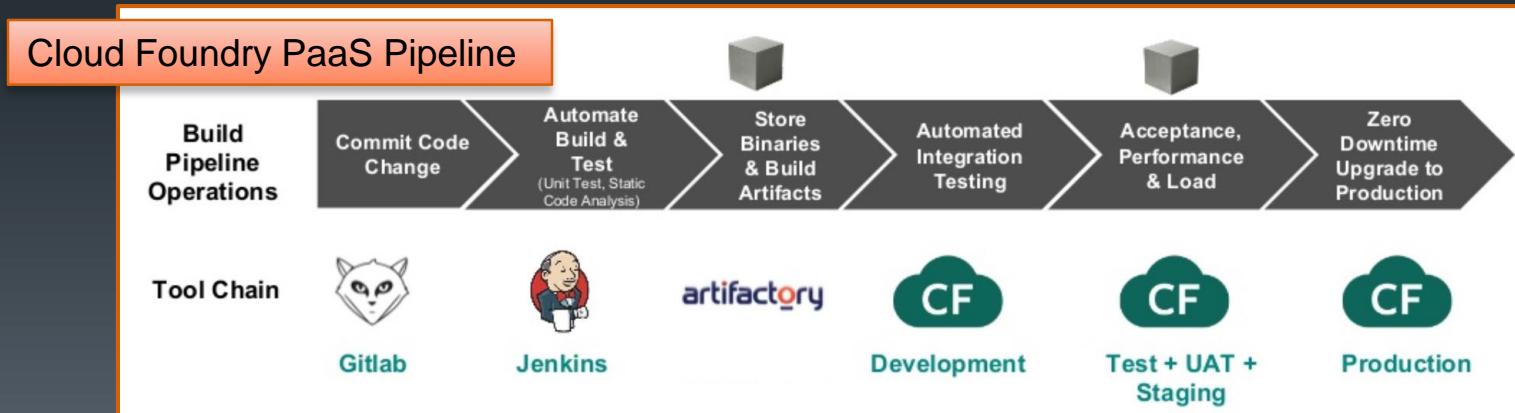
11

Copyright 2017 - 2019, RX-M LLC

- Platform as a Service

- Push code
- PaaS supports only applications fitting one of its buildpack models
- The PaaS builds, tests, packages and deploys
- Platform containerizes the artifacts
  - Perhaps JARs, a custom format, Docker Images or other

- Cloud Foundry Application Runtime [CFAR] is the Cloud Foundry PaaS



# CaaS - CFCR

12

Copyright 2017 - 2019, RX-M LLC

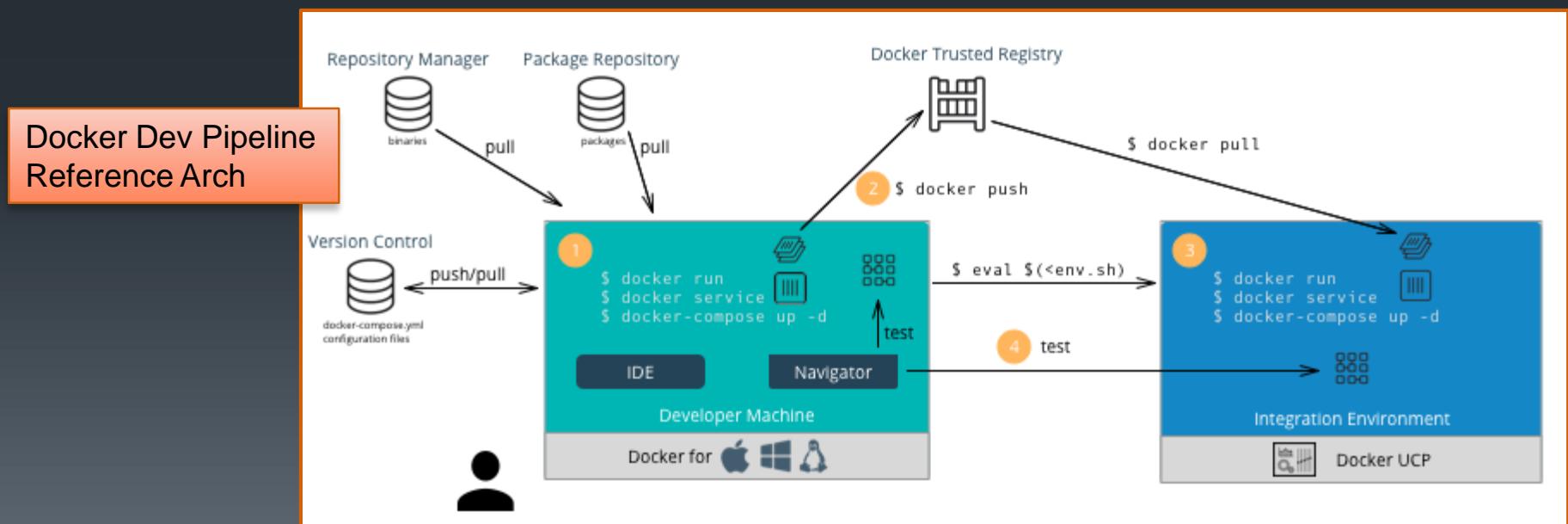
## ▪ Containers as a Service

- Push container Images
- The CaaS deploys and orchestrates
- Containers are created earlier in the pipeline reducing variability

## ▪ Hybrid

- Most popular PaaS today use container technology
  - Cloud Foundry uses containers and its own OCI runc based Guardian container manager (Greenhouse on Windows)
  - RedHat OpenShift uses Docker as the container manager and Kubernetes as the orchestration engine

## ▪ Cloud Foundry Container Runtime [CFCR] is Cloud Foundry's Kubernetes solution



# Pivotal Container Service [PKS]

13

Copyright 2017 - 2019, RX-M LLC

- **Kubernetes**

- Current stable release of Kubernetes, deployed and managed by BOSH
- No proprietary extensions

- **PKS Controller**

- Creates and scales Kubernetes clusters from the command line and API

- **GCP Service Broker**

- Allows apps to access Google Cloud APIs
- Provides Container Engine (GKE) portability

- **Harbor**

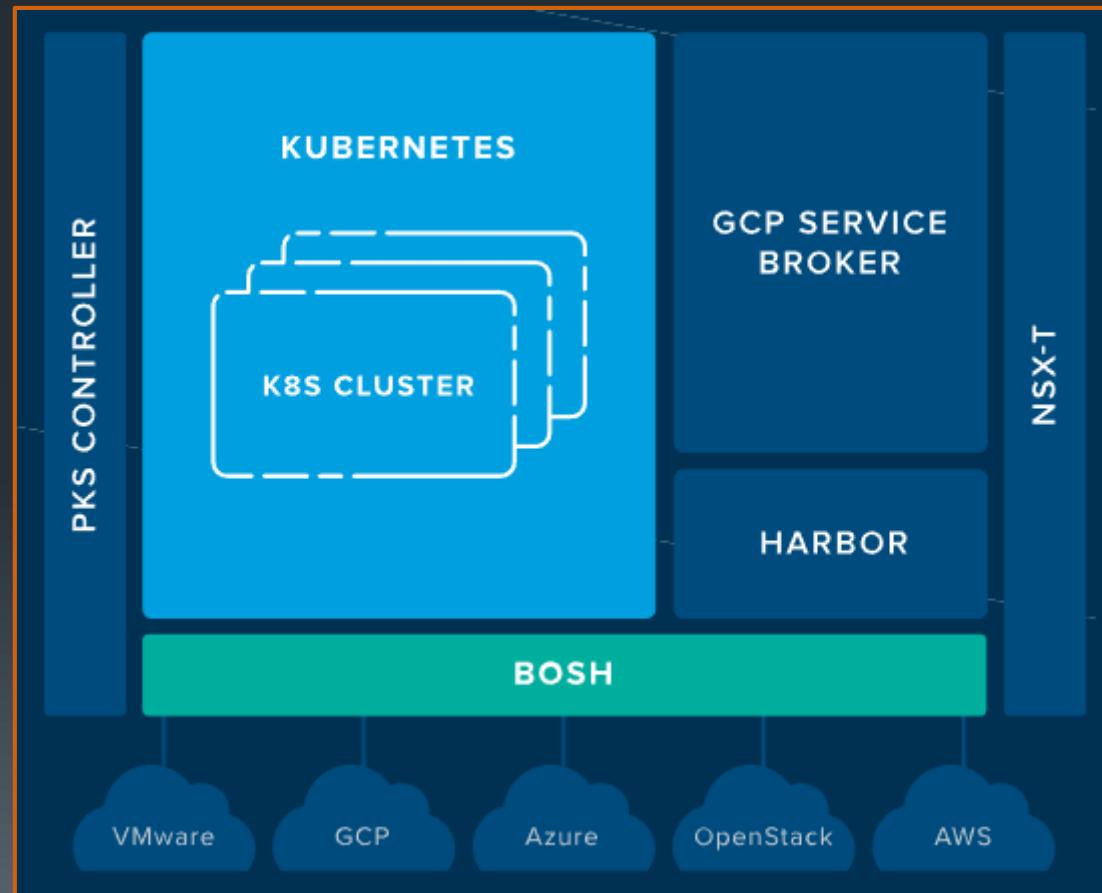
- A docker registry server with vulnerability scanning, identity management, and support for multiple registries

- **NSX-T**

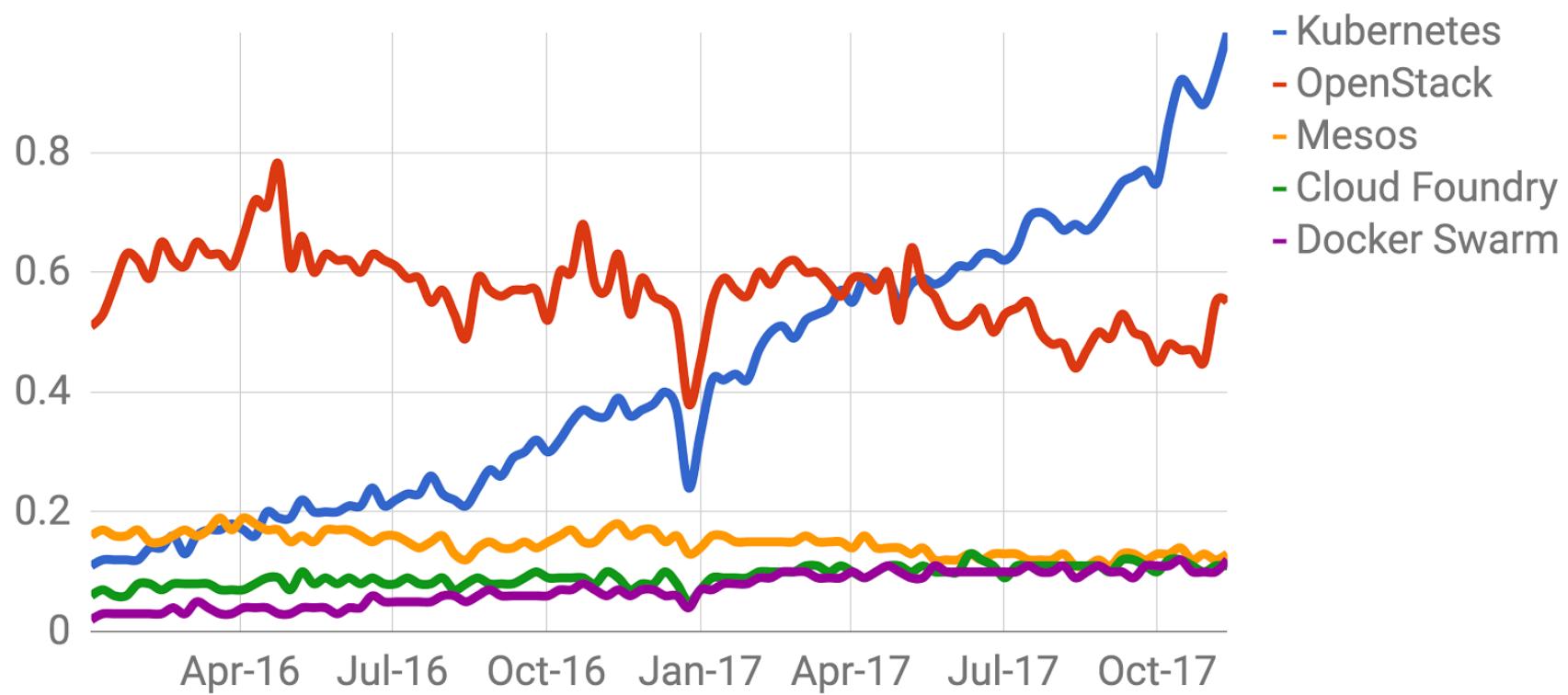
- VMware NSX-T secure SDN

- **BOSH**

- BOSH deploys, scales, monitors and heals Kubernetes clusters



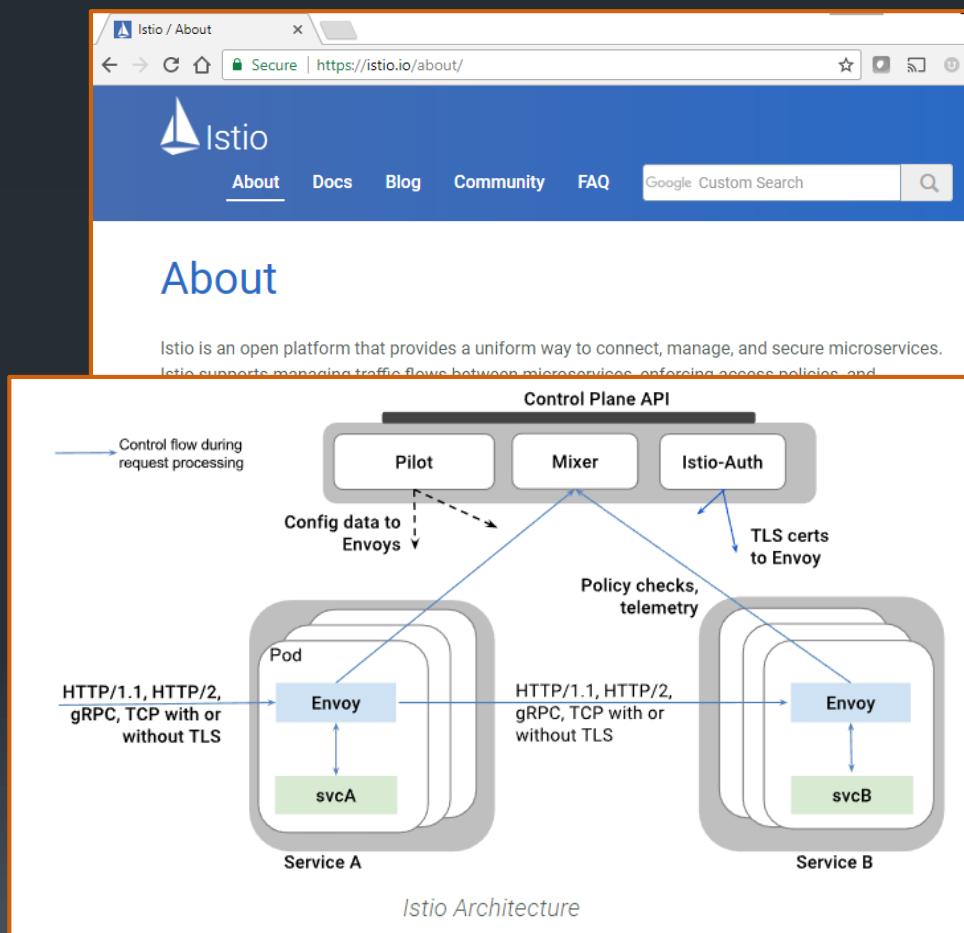
# Google Trends



# Istio

Copyright 2013-2019, RX-M LLC

- In a phrase: **Kubernetes integrated Envoy**
- Istio supports **policy based traffic management** between microservices
- Istio **generates telemetry data**
- Istio is **transparent to the actual microservices** it intermediates
- Features:
  - Automatic load balancing for HTTP, gRPC, and TCP traffic
  - Fine-grained control of traffic behavior with rich routing rules, retries, failovers, and fault injection
  - A pluggable policy layer and configuration API supporting access controls, rate limits and quotas
  - Automatic metrics, logs, and traces for all traffic within a cluster, including cluster ingress and egress
  - Secure service-to-service authentication with strong identity assertions between services in a cluster
- Only supports Kubernetes today
  - Support is planned for additional Cloud Foundry, Mesos, and bare metal
- Components:
  - **Envoy** Service Proxy
  - **Mixer** collects telemetry and enforces service mesh policy
  - **Pilot** is an interface for users to configure Istio via
  - **Istio-Auth** provides service-to-service and end-user authentication using mutual TLS



Welcome  
Concepts  
Tasks  
Samples  
Reference

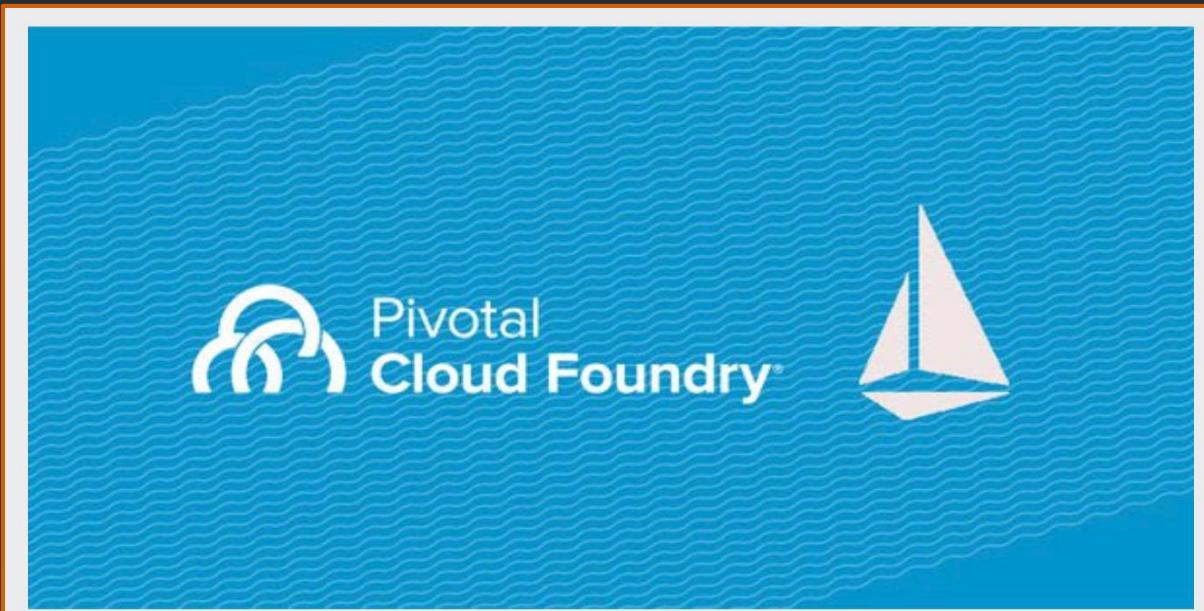
Frequently Asked Questions  
Troubleshooting Guide  
Report a Bug  
Report a Doc Issue  
Edit This Page on GitHub

User | Dev Mailing Lists  
Twitter  
GitHub

# Istio and CloudFoundry

Copyright 2013-2019, RX-M LLC

- Pivotal is making Istio core to
  - App platforms
    - CFAR/PCF
  - Container platforms
    - CFCR/PKS
  - Function platforms
    - Knative/PFS
      - Pivotal Function Service (PFS) is not yet released



## Istio 1.0 is Now GA!

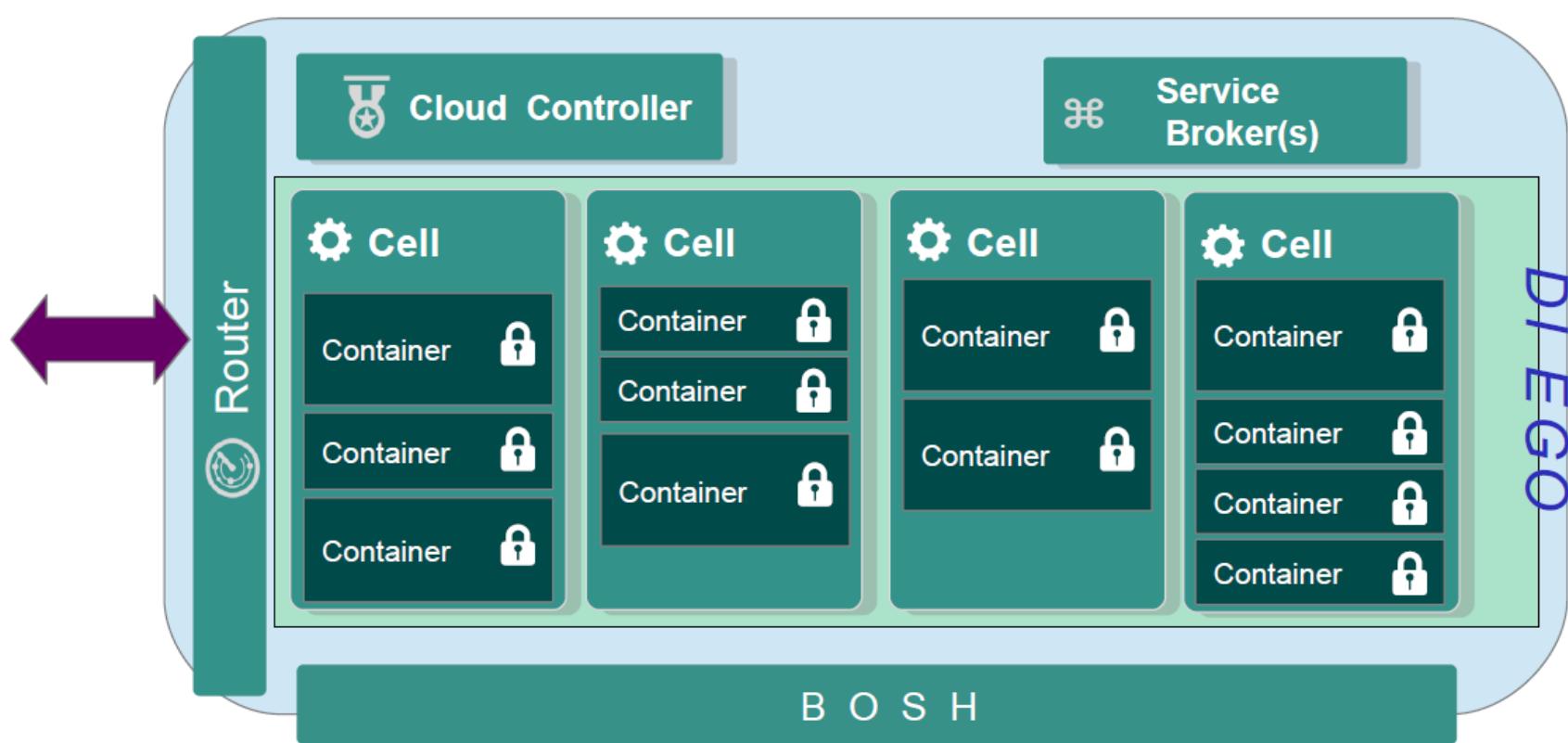
Istio 1.0 is now GA. Here's what it does, why it matters, and why we're adding it to Cloud Foundry and other projects to make your life better.

## 2: CFAR Overview

# Objectives

- Explore the features of Cloud Foundry Application Runtime
- Define
  - Organization
  - Domain
  - Space
  - Route
  - Buildpack
  - Service
- Understand application deployment
- Experiment with the cf CLI

# CFAR Architecture



**Cell** = Execution Agent  
= A VM where apps run



# CFAR App Deployment

20

Copyright 2017 - 2019, RX-M LLC

- User defined application

- Buildpack

- Platform



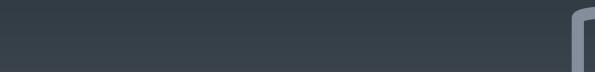
App



App Server (Tomcat)

Runtime (JVM)

Base OS rootfs



VM

Infrastructure

# Buildpacks

- Buildpacks define a specific **runtime environment**
  - Each service you deploy must run in a buildpack
  - **System Buildpacks** (built in to CloudFoundry):
    - Java, Python, Ruby, NodeJS, .Net Core, PHP, Go, Binaries, Static files
  - **Community Buildpacks** (developed by other CloudFoundry users):
    - Haskell, Django, Perl, Jetty, JBoss, Websphere Liberty, Swift, etc.
  - Many Heroku buildpacks work unmodified with CloudFoundry
- 
- Using a build pack:
    - `$ cf push <app-name> -p artifact -b https://github.com/cloudfoundry/java-buildpack.git`

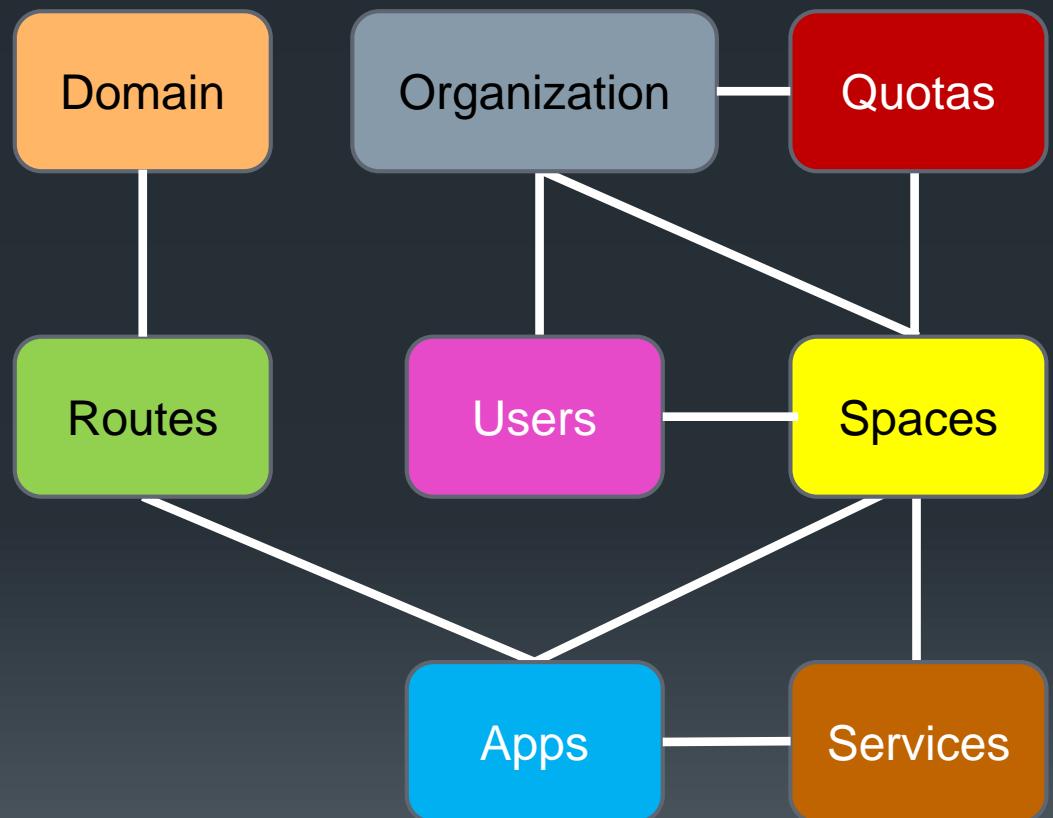
# Manifests

- Manifests allow you to provide application deployment instructions to Cloud Foundry
- Manifests eliminate manual/imperative tasks
  - Defining all aspects of the deployment in code
  - Repeatable
  - Auditable
  - Can be checked into source code control

```
---  
applications:  
- name: my-python-mongo-app  
  memory: 256M  
  instances: 2  
  path: .  
  host: mpma  
  domain: cfapps.io  
  command: python mongo-app.py  
  buildpack: https://github.com/rx-m/heroku-buildpack-python  
  services:  
    - my-mongo-db
```

# Organizations, Users and Quotas

- An **organization** is an independently billable entity that an individual or multiple collaborators can share
- Collaborators access an organization with **user** accounts
- Collaborators in an org share a resource **quota** plan
- Organizations have one or more **spaces**, each of which provides an isolated namespace for services and applications
- Access to organizations and spaces is controlled with role based permissions

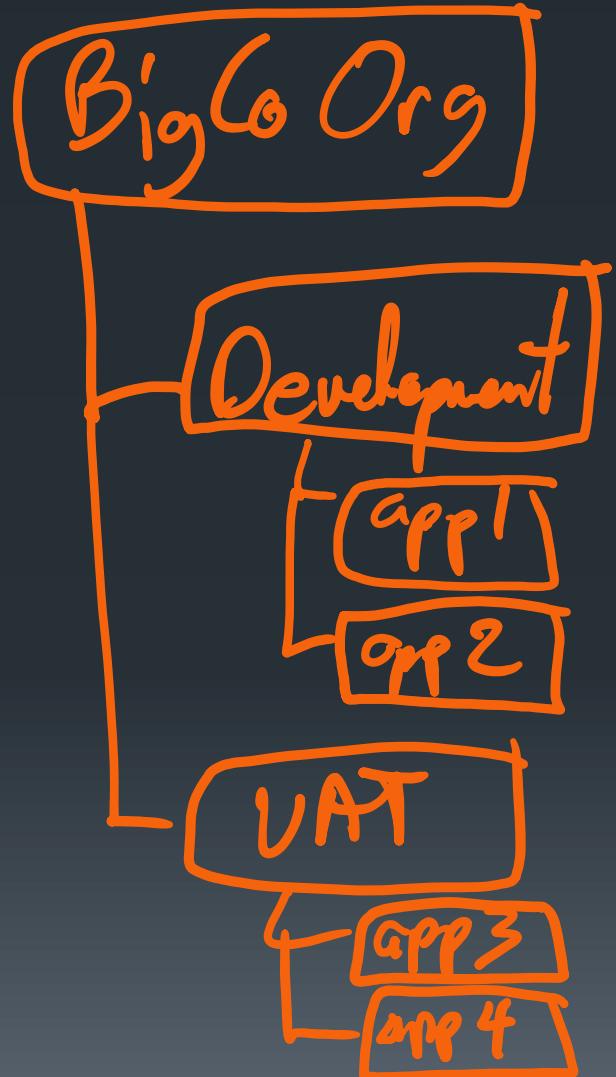


# Org Roles and Permissions

- Org Manager
  - Managers or other users that administer the org
- Org Auditor
  - Auditors can view but not edit user information or quota usage information
- Org User
  - Users can view the list of other org users and their roles

# Spaces

- Each Organization can container many spaces
  - The default PWS space is “development”
  - Administrators can create as many spaces as desired
- Users are given permissions on spaces
  - Developer privileges for creating and deploying applications
  - Administrative privileges for adding new users to the space
  - Auditing privileges for reading logs and examining configuration
- All applications and services live within some space



# Space Roles

- **Space Manager**
  - Can invite/manage users, enable features for a given space
- **Space Developer**
  - Can create, delete, manage applications and services, full access to all usage reports and logs
- **Space Auditor**
  - View only access to all space information, settings, reports, logs

# Domains

- Deployed applications are associated with a URL
  - All requests to that URL redirect to the application
- Each Cloud Foundry instance has a default app domain
  - PWS has cfapps.io
- Custom Domains
  - You can register your own domain or use a preexisting domain with Cloud Foundry
- Subdomains
  - Each application has a unique subdomain
    - App URLs are subdomain.domain
    - E.g.: http://testweb.cfapps.io

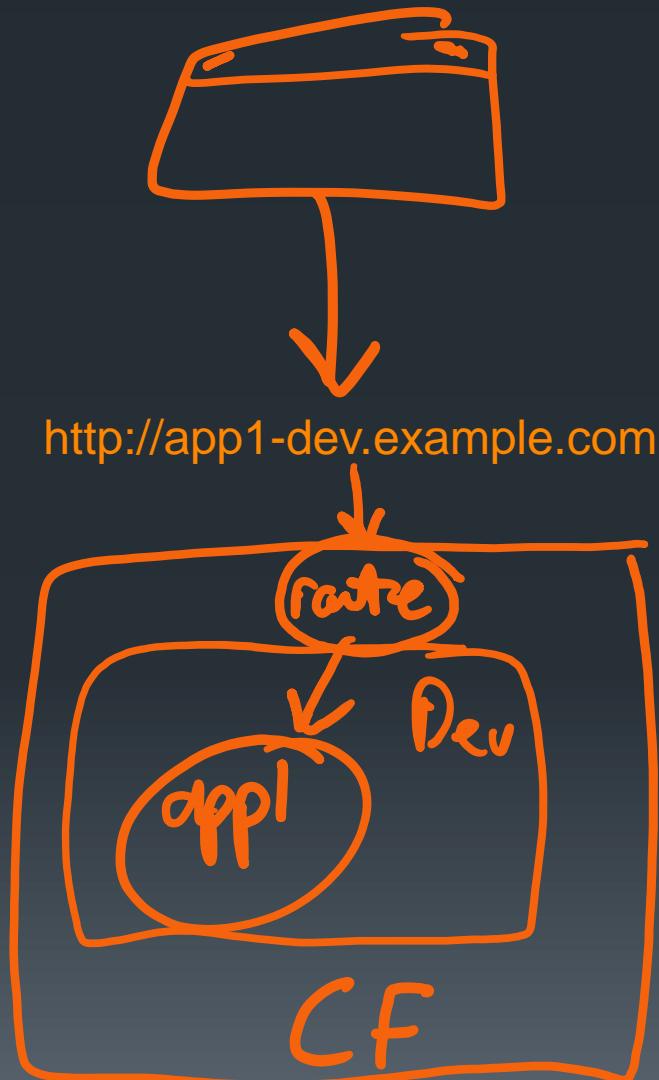
```
user@ubuntu:~/cf$ curl webtest-1234.cfapps.io/status
Server A
user@ubuntu:~/cf$ cf apps
Getting apps in org rx-m.com / space development as randy.abernethy@rx-m.com...
OK


| name    | requested state | state | instances | memory | disk | urls                   |
|---------|-----------------|-------|-----------|--------|------|------------------------|
| webtest | started         | 1/1   | 1/1       | 512M   | 1G   | webtest-1234.cfapps.io |


```

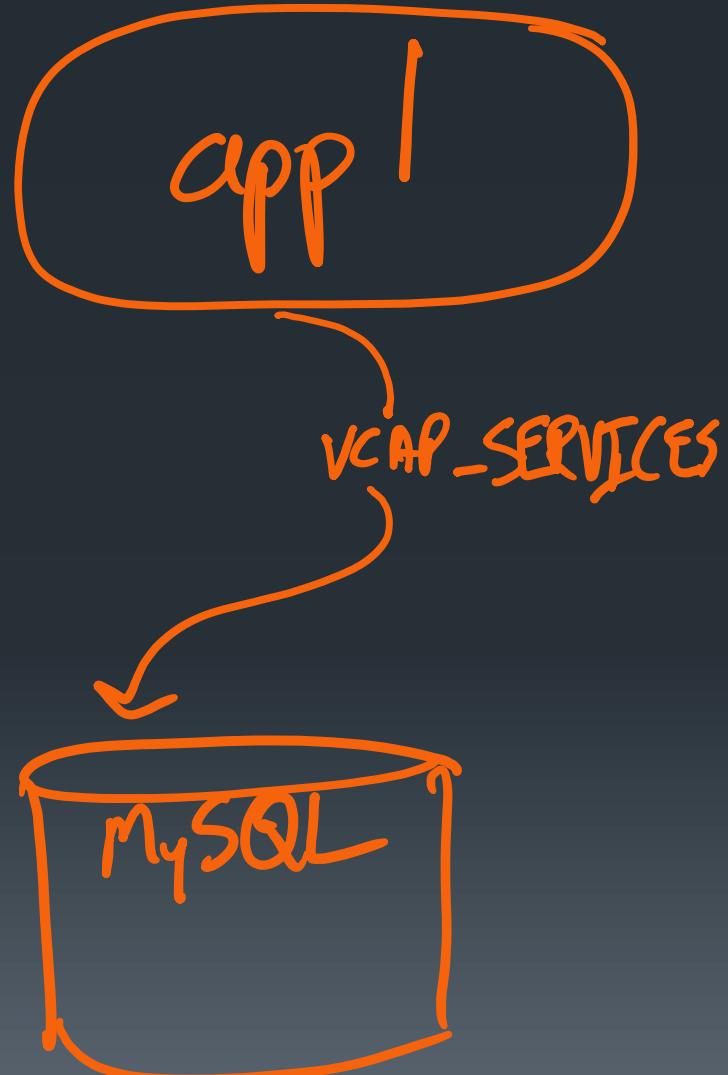
# Routes

- Routes provide a path from an Internet facing end point to a given application
  - A unique route exists to each application in each space
  - If multiple copies of a given application are running the route will load balance the instances as a group
- DNS domains can be shared by multiple spaces
  - App1 in the development space:
    - <http://app1-dev.example.com>
  - App1 in the UAT space:
    - <http://app1-uat.example.com>
  - You can use one or more of your own domain names in a given Cloud Foundry organization
  - PWS provides a DNS domain that you can use if you do not have your own:
    - cfapps.io
    - Shared by all users so prefixes must be unique



# Services

- In Cloud Foundry a service is a feature provided by the platform that your application can consume
  - A message queue
  - A database
  - Etc.
- Services can be consumed by multiple applications
  - Connecting a service to an application is called “binding” in Cloud Foundry
  - Applications discover connection information and login credentials through the VCAP\_SERVICES environment variable
  - All runtime specific application settings must be injected through environment variables
- Application state should generally be housed in platform services
  - Storage within an application container is ephemeral



- Pivotal Web Services (PWS)
  - Pivotal fully hosted and managed Cloud Foundry solution
  - <http://run.pivotal.io>
- Accounts can be created for free
  - Apps manager URL: [console.run.pivotal.io](https://console.run.pivotal.io)
  - API URL: [api.run.pivotal.io](https://api.run.pivotal.io)

The screenshot shows the Pivotal Web Services (PWS) console interface. The URL in the browser is <https://console.run.pivotal.io/organizations/aaa14634-a3f1-441e-9cae-7e237120db36/spaces/3ce1f344-0746-44c7-91...>. The interface includes a navigation bar with 'ORG' set to 'rx-m.com', a search bar, and a user account dropdown for 'randy.abernethy@rx-m.com'. The main area displays the 'development' space with the following details:

SPACE	RUNNING	STOPPED	CRASHED
development	1	0	0

Below this, there are tabs for 'App (1)', 'Services', 'Route (1)', 'Members (2)', and 'Settings'. The 'App (1)' tab is selected, showing a table for the 'Apps' section:

Status	Name	Instances	Memory	Last Push	Route
Running	webtest	1	512 MB	15 hours ago	<a href="https://webtest-1234.cfapps.io">https://webtest-1234.cfapps.io</a>

The left sidebar lists other spaces: 'development', 'UAT', 'Marketplace', 'Docs', 'Support', 'Tools', 'Blog', and 'Status'.

# Installing the CF CLI

31

Copyright 2017 - 2019, RX-M LLC

```
user@ubuntu:~$ wget -q -O - https://packages.cloudfoundry.org/debian/cli.cloudfoundry.org.key | sudo apt-key add -  
user@ubuntu:~$ echo "deb http://packages.cloudfoundry.org/debian stable main" | sudo tee /etc/apt/sources.list.d/cloudfoundry-cli.list  
deb http://packages.cloudfoundry.org/debian stable main  
user@ubuntu:~$ sudo apt-get update  
  
Hit:1 http://us.archive.ubuntu.com/ubuntu xenial InRelease  
Hit:2 http://us.archive.ubuntu.com/ubuntu xenial-updates InRelease  
Hit:3 http://us.archive.ubuntu.com/ubuntu xenial-backports InRelease  
Hit:5 http://security.ubuntu.com/ubuntu xenial-security InRelease  
Hit:6 https://download.docker.com/linux/ubuntu xenial InRelease  
Ign:4 https://cf-cli-debian-repo.s3.amazonaws.com stable InRelease  
Get:7 https://cf-cli-debian-repo.s3.amazonaws.com stable Release [1,797 B]  
Get:8 https://cf-cli-debian-repo.s3.amazonaws.com stable Release.gpg [819 B]  
Get:9 https://cf-cli-debian-repo.s3.amazonaws.com stable/main amd64 Packages [6,849 B]  
Get:10 https://cf-cli-debian-repo.s3.amazonaws.com stable/main i386 Packages [6,851 B]  
Fetched 16.3 kB in 1s (12.4 kB/s)  
Reading package lists... Done  
  
user@ubuntu:~$ sudo apt-get install cf-cli  
  
Reading package lists... Done  
Building dependency tree  
Reading state information... Done  
The following NEW packages will be installed:  
  cf-cli  
0 upgraded, 1 newly installed, 0 to remove and 230 not upgraded.  
Need to get 4,344 kB of archives.  
After this operation, 17.3 MB of additional disk space will be used.  
Get:1 http://packages.cloudfoundry.org/debian stable/main amd64 cf-cli amd64 6.34.1 [4,344 kB]  
Fetched 4,344 kB in 4s (1,024 kB/s)  
Selecting previously unselected package cf-cli.  
(Reading database ... 125146 files and directories currently installed.)  
Preparing to unpack .../cf-cli_6.34.1_amd64.deb ...  
Unpacking cf-cli (6.34.1) ...  
Setting up cf-cli (6.34.1) ...  
  
user@ubuntu:~$ cf -version  
cf version 6.34.1+bbdf81482.2018-01-17  
user@ubuntu:~$
```

# Help

- The cf help command provides general help:
  - cf help
- You can get help with specific commands using the form:
  - cf help <cmd>
- To get full help add the –a switch
  - cf help -a

```
user@ubuntu:~$ cf help push
NAME:
  push - Push a new app or sync changes to an existing app

USAGE:
  cf push APP_NAME [-b BUILDPACK_NAME] [-c COMMAND] [-f MANIFEST_PATH]
  [-i NUM_INSTANCES] [-k DISK] [-m MEMORY] [-p PATH] [-s STACK] [-t
  --no-route | --random-route | --hostname HOST | --no-hostname]
  cf push APP_NAME --docker-image [REGISTRY_HOST:PORT/]IMAGE[:TAG]
  [-c COMMAND] [-f MANIFEST_PATH | --no-manifest] [--no-start]
  [-i NUM_INSTANCES] [-k DISK] [-m MEMORY] [-t HEALTH_TIMEOUT] [-u
  --no-route | --random-route | --hostname HOST | --no-hostname]
  cf push -f MANIFEST_WITH_MULTIPLE_APPS_PATH [APP_NAME] [--no-start]

ALIAS:
  p

OPTIONS:
  -b          Custom buildpack by name (e.g. my-buildpack)
              or by URL (e.g. 'https://github.com/cloudfoundry/java-buildpack')
  -c          Startup command, set to null to rese...
```

```
user@ubuntu:~$ cf help
cf version 6.35.2+88a03e995.2018-03-15, Cloud Foundry command line tool
Usage: cf [global options] command [arguments...] [command options]
```

#### Before getting started:

```
config      login,l      target,t
help,h      logout,lo
```

#### Application lifecycle:

```
apps,a      run-task,rt    events
push,p      logs           set-env,se
start,st    ssh            create-app-manifest
stop,sp     app
restart,rs   env,e
restage,rg   scale
```

#### Services integration:

```
marketplace,m  create-user-provided-service,cups
services,s     update-user-provided-service,uups
create-service,cs  create-service-key,csk
update-service   delete-service-key,dsk
delete-service,ds service-keys,sk
service         service-key
bind-service,bs bind-route-service,btrs
unbind-service,us unbind-route-service,urs
```

#### Route and domain management:

```
routes,r      delete-route  create-domain
domains       map-route
create-route   unmap-route
```

#### Space management:

```
spaces        create-space  set-space-role
space-users   delete-space  unset-space-role
```

#### Org management:

```
orgs,o       set-org-role
org-users    unset-org-role
```

#### CLI plugin management:

```
plugins       add-plugin-repo repo-plugins
install-plugin list-plugin-repos
```

#### Commands offered by installed plugins:

#### Global options:

```
--help, -h          Show help
-v                  Print API request diagnostics to stdout
```

Use 'cf help -a' to see all commands.

```
user@ubuntu:~$
```

# Login

```
user@ubuntu:~$ cf login -a api.run.pivotal.io randy.abernethy@rx-m.com
API endpoint: api.run.pivotal.io
```

```
Email> randy.abernethy@rx-m.com
```

```
Password>
```

```
Authenticating...
```

```
OK
```

```
Targeted org rx-m.com
```

```
Targeted space development
```

```
API endpoint: https://api.run.pivotal.io (API version: 2.101.0)
```

```
User: randy.abernethy@rx-m.com
```

```
Org: rx-m.com
```

```
Space: development
```

```
user@ubuntu:~$ █
```

# CF URLs

- PWS URLs

- System domain: run.pivotal.io
- API Endpoint: api.run.pivotal.io
- Apps Manager: console.run.pivotal.io
- Apps domain: cfapps.io

- On Prem Cloud Foundry (PCF)

- System domain: <your configured CF system domain>
- API Endpoint: api.<your configured CF system domain>
- Apps Manager: login.<your configured CF system domain>
- Apps domain: <your configured CF app domain>

# CF state store

- Cloud Foundry login data, configuration and plugins are stored in the .cf directory
- The cf target command displays current config

```
user@ubuntu:~$ ls -la ~/.cf
total 12
drwx----- 2 user user 4096 Feb  1 19:25 .
drwxr-xr-x 16 user user 4096 Feb  1 19:24 ..
-rw----- 1 user user 3494 Feb  1 19:29 config.json
user@ubuntu:~$ cf target
api endpoint:      https://api.run.pivotal.io
api version:       2.101.0
user:              randy.abernethy@rx-m.com
org:               rx-m.com
space:             development
user@ubuntu:~$ █
```

# Displaying orgs and spaces

```
user@ubuntu:~$ cf orgs
```

```
Getting orgs as randy.abernethy@rx-m.com...
```

```
name
```

```
rx-m.com
```

```
user@ubuntu:~$ cf spaces
```

```
Getting spaces in org rx-m.com as randy.abernethy@rx-m.com...
```

```
name
```

```
development
```

```
user@ubuntu:~$
```

# Creating and setting (targeting) spaces

```
user@ubuntu:~/cf$ cf create-space UAT
Creating space UAT in org rx-m.com as randy.abernethy@rx-m.com...
OK
Assigning role RoleSpaceManager to user randy.abernethy@rx-m.com in org rx-m.com / space UAT as randy.abernethy@rx-m.com...
OK
Assigning role RoleSpaceDeveloper to user randy.abernethy@rx-m.com in org rx-m.com / space UAT as randy.abernethy@rx-m.com...
OK

TIP: Use 'cf target -o "rx-m.com" -s "UAT"' to target new space
user@ubuntu:~/cf$
```

# Deploying Applications

38

Copyright 2017 - 2019, RX-M LLC

- Cloud Foundry apps are deployed with the “cf push” command
    - \$ cf push myapp
    - “cf p” for short
  - Push options:
    - -i number of instances
    - -m Memory limit (256M, 1024M, 1G)
    - -n Host Name (app subdomain)
    - -p Local path to app directory (jar, war, \*.js, zip, etc.)
    - -b Custom buildpack URL (usually a github/bitbucket URL)
      - Not required for built-in buildpacks
    - --health-check-type, -u Health check type
      - Port (default)
      - None
      - Process
      - Http
      - Many others

# A Node JS App

```
user@ubuntu:~/cf$ cat test.js
var express = require('express');
var http = require('http');

var app = express();

app.get('/status', function(req, res) {
    return res.send('Server A\n');
});

http.createServer(app).listen(process.env.PORT || 9090, function() {
    console.log('Listening on port 9090');
});

user@ubuntu:~/cf$
```

# Dependencies

40

Copyright 2017 - 2019, RX-M LLC

```
user@ubuntu:~/cf$ ls -l
total 12
drwxrwxr-x 31 user user 4096 Feb  1 20:18 node_modules
-rw-rw-r--  1 user user   152 Feb  1 20:19 package.json
-rw-rw-r--  1 user user   282 Feb  1 20:09 test.js
user@ubuntu:~/cf$ cat package.json
{
  "name": "first",
  "version": "0.0.1",
  "author": "Demo",
  "dependencies": {
    "express": "3.4.8"
  },
  "engines": {
    "node": "4.8.6"
  }
}
user@ubuntu:~/cf$
```

# cf push [1]

```
user@ubuntu:~/cf$ cf push webtest -i 1 -m 512M -c "node test.js" -n webtest-1234
Pushing app webtest to org rx-m.com / space development as randy.abernethy@rx-m.com...
Getting app info...
Updating app with these attributes...
  name:                  webtest
  path:                  /home/user/cf
  command:               node test.js
  disk quota:            1G
  health check type:    port
  instances:             1
  memory:                512M
  stack:                 cflinuxfs2
  routes:
    webtest-1234.cfapps.io

Updating app webtest...
Mapping routes...
Comparing local files to remote cache...
Packaging files to upload...
Uploading files...
  387.34 KiB / 387.34 KiB [=====] 100.00% 1s

Waiting for API to complete processing files...

Staging app and tracing logs...
  Downloading binary_buildpack...
  Downloading nodejs_buildpack...
```

# cf push [2]

```
Uploading droplet, build artifacts cache...
Uploading build artifacts cache...
Uploading droplet...
Uploaded build artifacts cache (278B)
Uploaded droplet (13.6M)
Uploading complete
Stopping instance 44b806c4-cfa4-4a90-89b3-d3d01d539ad2
Destroying container
```

Waiting for app to start...

```
name:          webtest
requested state: started
instances:      1/1
usage:          512M x 1 instances
routes:         webtest-1234.cfapps.io
last uploaded:  Thu 01 Feb 20:19:58 PST 2018
stack:          cflinuxfs2
buildpack:      nodejs
start command:  node test.js
```

	state	since	cpu	memory	disk	details
#0	running	2018-02-02T04:20:28Z	0.0%	0 of 512M	0 of 1G	

user@ubuntu:~/cf\$

# Monitoring Apps

```
user@ubuntu:~/cf$ curl webtest-1234.cfapps.io/status
Server A
user@ubuntu:~/cf$ cf apps
Getting apps in org rx-m.com / space development as randy.abernethy@rx-m.com...
OK

name      requested state    instances   memory   disk     urls
webtest   started          1/1        512M     1G       webtest-1234.cfapps.io
user@ubuntu:~/cf$ cf logs webtest
Retrieving logs for app webtest in org rx-m.com / space development as randy.abernethy@rx-m.com...

2018-02-01T20:27:35.96-0800 [RTR/9] OUT webtest-1234.cfapps.io - [2018-02-02T04:27:35.957+0000] "GET /status HTTP/1.1" 200 0 9 "-" "curl/7.47.0" "10.10.2.158:52630" "10.10.149.157:61028" x_forwarded_for:"63.98.155.67, 10.10.2.158" x_forwarded_proto:"http" vcap_request_id:"26d48e0a-d488-4402-69bb-7e4e13fd6cf1" response_time:0.006506819 app_id:"8d01c6ce-1e9d-4280-9b01-0ee3f22f53db" app_index:"0" x_b3_traceid:"fe4727821b5bdcfe" x_b3_spanid:"fe4727821b5bdcfe" x_b3_parentspanid:"-"
2018-02-01T20:27:35.96-0800 [RTR/9] OUT
```

# Scaling Apps

- Changing the instance count (scaling horizontally):
  - \$ cf scale <app> -i <new-value>
  - To scale myapp to 5 instances:
    - \$ cf scale myapp -i 5
  - Will start or stop instances as necessary
- Changing instance memory allocation (scaling vertically)
  - \$ cf scale <app> -m <new-memory>
  - To scale myapp to 1 gig
    - \$ cf scale myapp -m 1G
  - The instance must be restarted before the new size will take effect

# Controlling Apps

- `cf stop` Sends SIGTERM, 10 seconds later sends SIGKILL
- `cf start` Starts app
- `cf restart` Starts and stops an app
- `cf restage` Redeploys the app and starts it  
(required to pick up config changes)

# CF Web

The screenshot shows the Pivotal Web Services (PWS) console interface. The URL in the browser is <https://console.run.pivotal.io/organizations/aaa14634-a3f1-441e-9cae-7e237120db36/spaces/3ce1f344-0746-44c7-9140-ec6baa4b8...>.

The left sidebar includes links for Org (rx-m.com), Spaces (development, Marketplace, Docs, Support, Tools, Blog, Status), and a search bar.

The main dashboard shows the following details:

- SPACE:** development
- RUNNING:** 1
- STOPPED:** 0
- CRASHED:** 0

The navigation tabs are **App (1)**, **Services**, **Route (1)**, **Members (2)**, and **Settings**. The **App (1)** tab is selected.

The **Apps** section displays one application instance:

Status	Name	Instances	Memory	Last Push	Route
Running	webtest	1	512 MB	10 minutes ago	<a href="https://webtest-1234.cfapps.io">https://webtest-1234.cfapps.io</a>

At the bottom, the footer reads "Pivotal © 2018 Pivotal Software Inc. All rights reserved. [Terms](#) | [Privacy](#)" and "GIVE FEEDBACK".



ORG

rx-m.com ▾

SPACES

development

Marketplace

Docs

Support

Tools

Blog

Status

APP

webtest



● Running

VIEW APP

Overview

Services

Route (1)

Logs

Tasks

Settings

Buildpack: N/A

Events

Last Push: 08:20 PM 02/01/18

Scaling

CANCEL

SCALE APP

- Started app  
randy.abernethy@rx-m.com 02/02/2018 at 04:20:06 AM UTC
- Stopped app  
randy.abernethy@rx-m.com 02/02/2018 at 04:19:57 AM UTC
- Started app  
randy.abernethy@rx-m.com 02/02/2018 at 04:18:41 AM UTC
- Stopped app  
randy.abernethy@rx-m.com 02/02/2018 at 04:18:32 AM UTC
- Stopped app  
randy.abernethy@rx-m.com 02/02/2018 at 04:16:14 AM UTC
- Stopped app  
randy.abernethy@rx-m.com 02/02/2018 at 04:14:39 AM UTC
- Mapped route to app  
randy.abernethy@rx-m.com 02/02/2018 at 04:13:42 AM UTC
- Created app  
randy.abernethy@rx-m.com 02/02/2018 at 04:13:42 AM UTC

Instances

Memory Limit

Disk Limit

1

512 MB

1 GB

 Autoscaling

Instances

View in PCF Metrics

#	CPU	Memory	Disk	Uptime
0	0%	18.69 MB	44.69 MB	10 min

# Lab 1

- Cloud Foundry Application Runtime

# 3: Garden

# Objectives

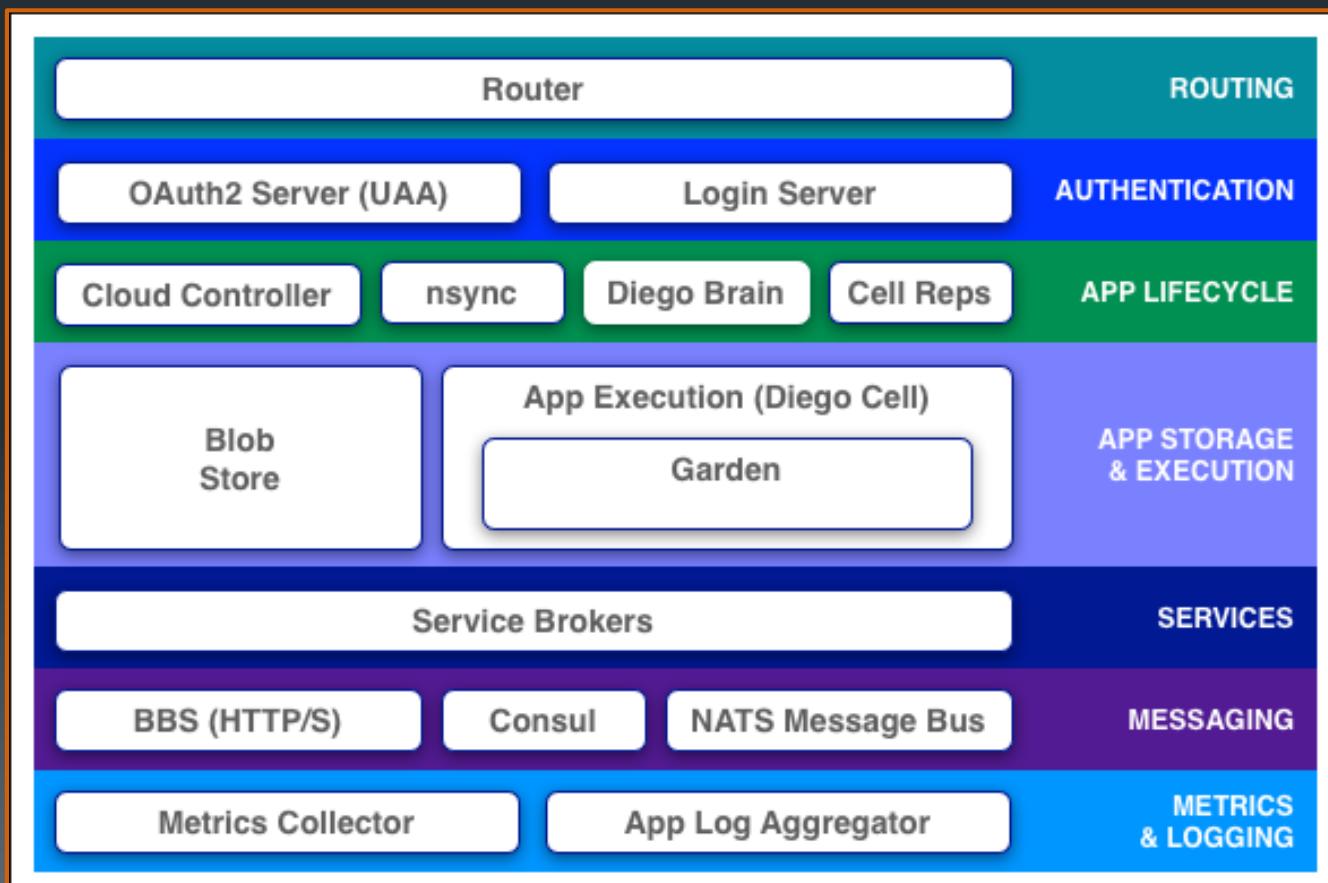
- Understand CF container tech and architecture
- Explain the role of Garden
- List the Linux kernel features used to isolate Garden containers
- Describe Garden images
- Map out the networking model of a garden container

# Garden Containers

51

Copyright 2017 - 2019, RX-M LLC

- Garden containers are the self-contained environments used to encapsulate an app deployed to CF
  - The Garden container manager is a Go language rewrite of the old Ruby based Warden container manager
- Containers isolates processes, memory, and the filesystem using operating system namespaces and cgroups
- Each container includes
  - Process ID (PID) namespace
  - Network namespace
  - Mount namespace
- Resource control is managed using Linux control groups
  - Limits the amount of memory that the container may use
  - Requires the container to use a fair share of CPU

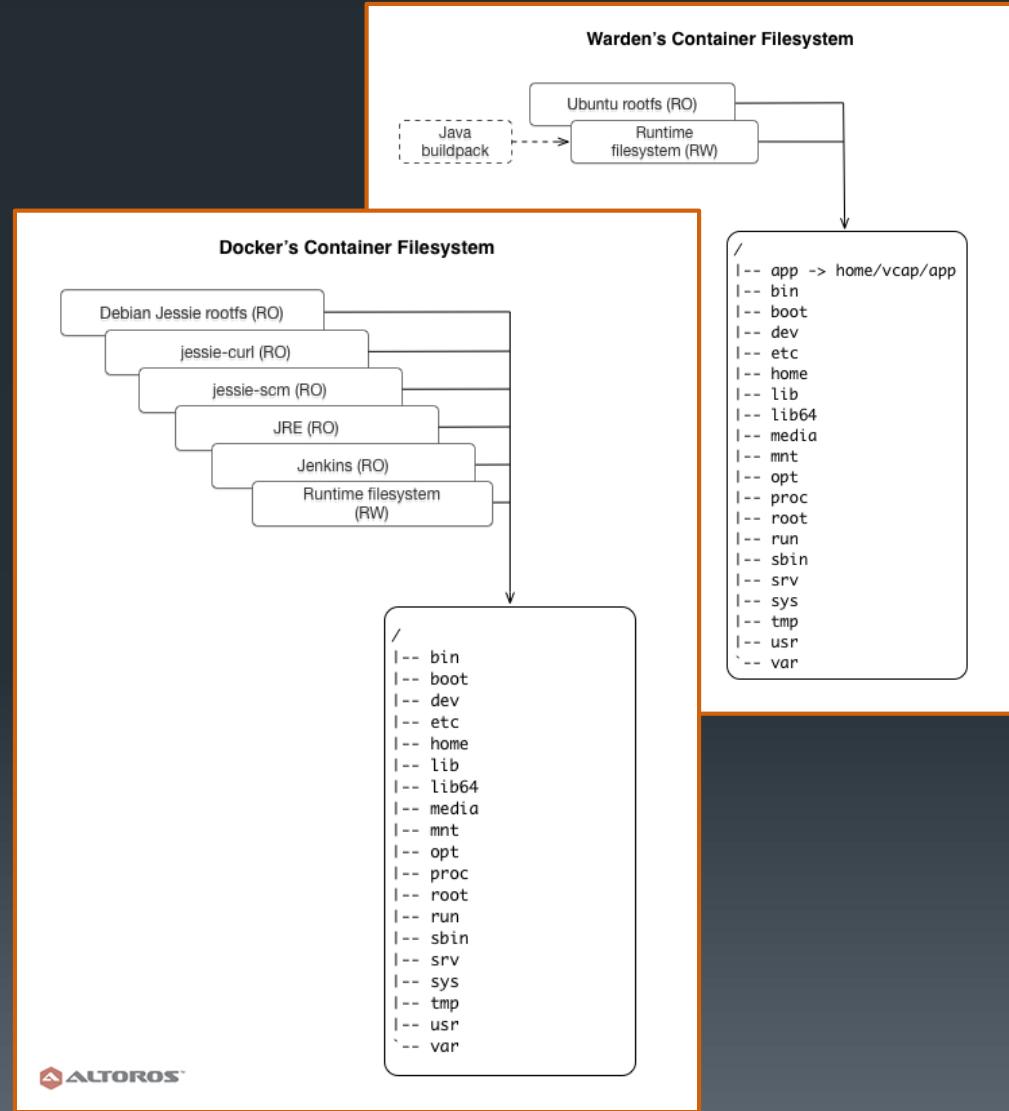


# Docker versus CF Containers

52

Copyright 2017 - 2019, RX-M LLC

- The main distinction between Docker containers and CF containers is the image structure used
- CF is designed to run applications that get all their dependencies from buildpacks
- CF containers have two layers:
  - a read-only layer with an OS root file system
    - e.g. Ubuntu 16.04
  - a nonpersistent, read/write layer for the application (based on a buildpack)
    - Dependencies and temporary data
    - Only this image can be run
- Docker containers can have 1 or more layers and each layer can be used to start a container

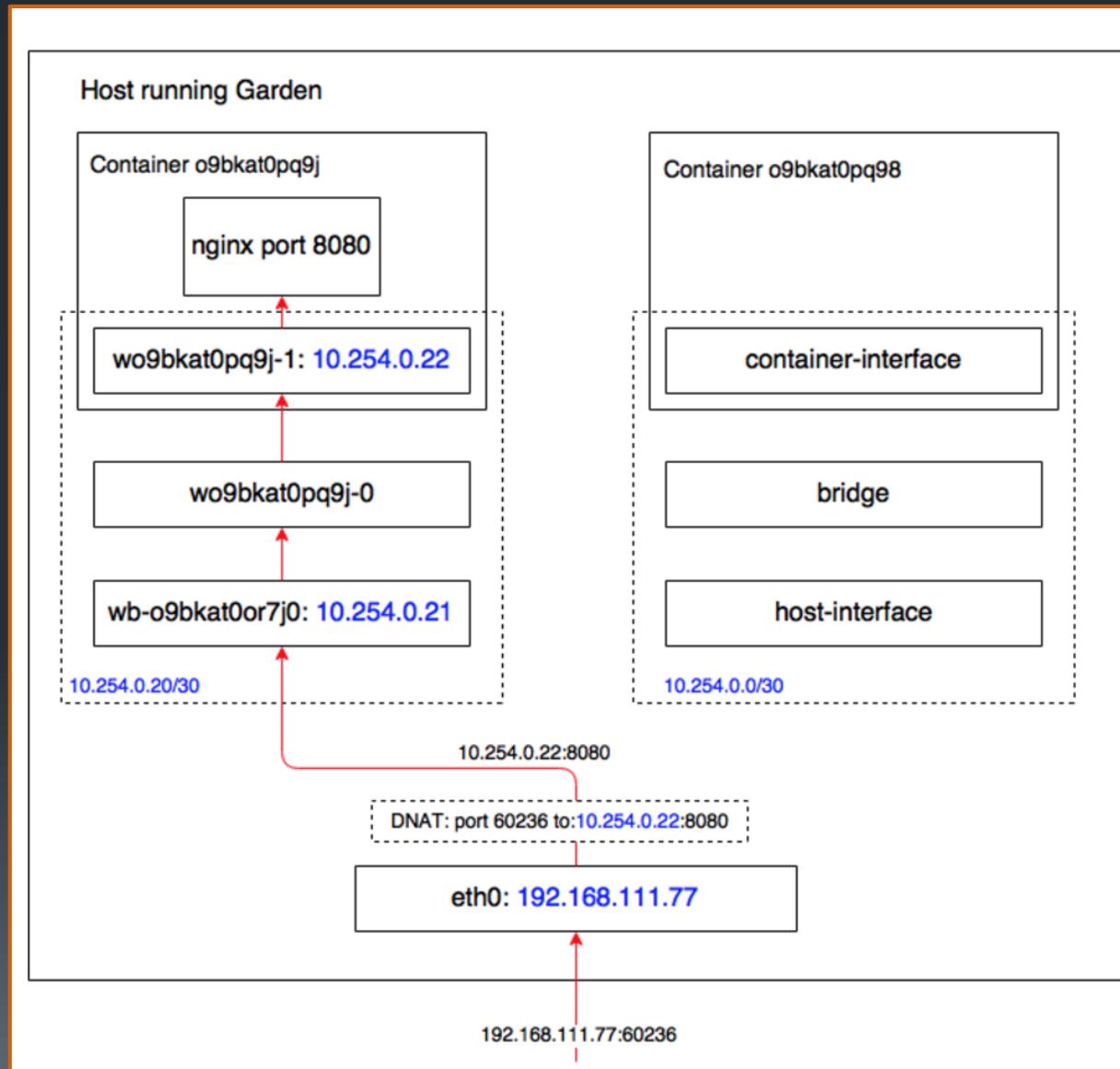


# Garden Networking

53

Copyright 2017 - 2019, RX-M LLC

- Networks in containers are isolated from a host with the **Linux network namespace**
- Garden creates a dedicated /30 network for each container with two IP addresses:
  - One for the host
  - One for the container
- Garden uses “network address translation” (NAT) based on a port to forward traffic from the host to the container
- Garden randomly picks an unused port and then adds a rule to iptables to forward traffic to the container



# Using Docker with CF

- Garden can run OCI compliant containers
  - Old versions of Garden used **Garden-Linux** (Diego specific) to run containers
  - New versions of Garden use **Garden-runc** (the OCI container launcher)
- OCI images must be properly prepared to run under Garden
- To enable Docker support on a CF deployment, an operator must enable the **diego\_docker** feature flag
- Registry servers are also commonly configured to allow CF to allow CI solutions to push images and to allow CF to pull images

```
$ cf enable-feature-flag diego_docker
```

# CF Docker Requirements

- To push apps with Docker, CF requires the following:
  - A Cloud Foundry (CF) deployment that has Docker support enabled
  - A Docker image that meets the following requirements:
    - Must contain an `/etc/passwd` file with an entry for the **root user**
    - The **home directory and shell for the root user** must be present in the image
    - **Image layer size must not exceed the disk quota for the app**
      - Max disk allocation for apps is set by the Cloud Controller (default is 2048 MB)
      - If the size of the image layers exceeds the disk quota app instances will not start
  - The location of the Docker image on Docker Hub or another Docker registry
    - The registry must support Docker Registry API V2
    - The registry must present a valid certificate
- **cf ssh requires a shell program** be installed in the container image

# Ports

- The Docker image exposed port will set the CF PORT
  - Use the Dockerfile EXPOSE directive to set the container and CF port
  - CF supports only one port per container

# Pushing a docker image

- To deploy a Docker image from a Docker Hub
  - \$ cf push APP-NAME --docker-image REG/ACCT/REPO:TAG
- APP-NAME: The name of the app being pushed
- REG: The host:port of the registry to pull from
- ACCT: The name of the account/org where the image is stored
- REPO: The name of the image repository in the account/org
- TAG: (Optional) The tag of the image (defaults to “latest”)
- For example, the following command pushes the my-image image from Docker Hub to a Cloud Foundry app:
  - \$ cf push my-app --docker-image cloudfoundry/my-image

# Lab 2

- Pushing container images to Cloud Foundry

# 4: Services and More

# Objectives

- Understand additional CF features
  - Services
  - Observability
  - Revisions
  - Droplet import/export

# Adding Services to a Space

Copyright 2013-2019, RX-M LLC

- Spaces offer services for apps running in the space

The screenshot shows the Pivotal Web Services interface. On the left, a sidebar menu includes options like Space, Apps, Services (which is selected), Member (1), Settings, Networking, Routes, and Marketplace. The main area shows a search bar and navigation links for Home > rx-m2, SPACE development, RUNNING 0, STOPPED 0, and CRASHED 0. Below this, a section titled 'Services' displays a message: 'No services in this app space. Add a service'. A central modal window is open, titled 'Create a new service' with a 'View in Marketplace' link. It features a section for 'Redis On-Demand' with the subtext 'Dedicated Redis for Pivotal Cloud Foundry instances'. Below this is a 'Service Plans' section with three options: '460MB free' (selected with a radio button), '920MB free', and '1840MB free'. Each plan has a 'Show Details' link. At the bottom of the modal are 'BACK' and 'SELECT PLAN' buttons. To the right of the modal, a list of available services is shown in a card format:

- App Autoscaler**: Scales bound applications in response to load.
- BlazeMeter**: Performance Testing Platform.
- Cedexis Openmix**: Openmix Global Cloud and Data Center Load Balancer.
- Circuit Breaker**: Circuit Breaker Dashboard for Spring Cloud Applications.
- ClearDB MySQL Database**: Highly available MySQL for your Apps.
- CloudAMQP**: Managed HA RabbitMQ servers in the cloud.
- Config Server**: Config Server for Spring Cloud Applications.
- ElephantSQL**: PostgreSQL as a Service.

# Services are bound to apps

Copyright 2013-2019, RX-M LLC

Pivotal Web Services

- Space
- Apps
- Service (1)
- Member (1)
- Settings
- Networking
- Routes
- Marketplace

Search apps, services, spaces, & orgs

Success: Service instance "rep-data" was successfully created

Home > rx-m2

SPACE	RUNNING	STOPPED	CRASHED
development	● 0	■ 0	▼ 0

Services

Service	Name	Bound Apps	Plan
redis®	Redis Cloud	rep-data	0 free - 30MB

Create a new service

View in Marketplace

Redis On-Demand  
cache-small - free

Instance Name: rep-data

Add to Space: development

Bind to App (Optional): [do not bind]

Add Parameters (Optional):  
key: appendonly, value: yes

Enter as JSON: OFF

CREATE

BACK

# VCAP\_SERVICES

- Bound services are discoverable by apps through the VCAP\_SERVICES environment variable
- Available in CLI and GUI
  - Application -> Settings

```
{  
  "staging_env_json": {},  
  "running_env_json": {},  
  "system_env_json": {  
    "VCAP_SERVICES": {  
      "scheduler-for-pcf": [  
        {  
          "label": "scheduler-for-pcf",  
          "provider": null,  
          "plan": "standard"  
        },  
        {  
          "label": "rediscloud",  
          "provider": null,  
          "plan": "30mb",  
          "name": "rep-data",  
          "tags": [  
            "Data Stores",  
            "Web-based",  
            "Data Store",  
            "Caching",  
            "Messaging and Queuing",  
            "Data Management",  
            "key-value",  
            "IT Management",  
            "caching",  
            "redis"  
          ],  
          "instance_name": "rep-data",  
          "binding_name": "",  
          "credentials": {  
            "hostname": "redis-11773.c93.us-east-1-3.ec2.cloud.redislabs.com",  
            "password": "ATcUOLrBX6G9K6c70qthxuDq5w5HYddN",  
            "port": "11773"  
          },  
          "syslog_drain_url": null,  
          "volume_mounts": []  
        }  
      ]  
    }  
  }  
}
```

# CF Observability

Copyright 2013-2019, RX-M LLC

- CF offers integrated observability:
  - system event logs
  - space metering
- Monitors can fire web hooks when thresholds are exceeded

The screenshot shows the PCF Metrics interface with a sidebar on the left and a main configuration page on the right.

**Left Sidebar:**

- PCF Metrics
- Dashboard
- Monitors BETA (highlighted)
- Docs

**Main Configuration Page:**

**Header:** cftest ORG: rx-m2 SPACE: development STATUS: ● Running

**TRIGGERED** (selected) **MANAGE**

**New Event Monitor**

**CRITERIA**: For  trigger when

**THRESHOLD**:

- Critical Threshold:  per minute
- Warning Threshold: (optional)  per minute

**Don't trigger this monitor more than once every**

**TITLE**: Spurious updates

**CRITICAL NOTIFICATION**: Webhook URL: <https://hooks.slack.com/services/T0000000/B0000000/XXXXXXXXXXXXXXXXXXXX>

**POST body (JSON)**: { "text": "This is an example alert. <https://metrics.run.pivotal.io/apps/09f5ea32-6a59-41e6-a971-7c6b7d714d84>" }

**TEST NOTIFICATION**: { }

The screenshot shows the PCF Metrics interface with a sidebar on the left and a log viewer on the right.

**Left Sidebar:**

- All selected (8) (highlighted)
- All selected (8)
- APP (Application)
- API (Cloud Controller)
- RTR (Router)
- CELL (Diego Cell)
- HEALTH (Health Check)

**Right Panel:**

LOGS

All selected (8)  Keyword

Date	Log Entry
Mon Oct 07 2019 [RTR/10]	1 cftest-12322.cfapps.io - [2019-10-07T12:58:16.847+0000] "OPTIONS /"
Mon Oct 07 2019 [RTR/10]	2 AppWebkit/537.36 (KHTML, like Gecko) Chrome/77.0.3865.98 Safari/537.36 x_forwarded_proto:"https" vcap_request_id:"c107c81b-5773-4629-7048-87b3_1b6580" x_b3_traceid:"1eb1e95440bb1b6850" x_b3_spand:"1eb1e95440bb1b6850" x_b3_parentspanid:"daef3686482be3ed7" x_b3_version:1.6.1-build.69
Mon Oct 07 2019 [RTR/7]	1 cftest-12322.cfapps.io - [2019-10-07T12:57:57.783+0000] "OPTIONS /"
Mon Oct 07 2019 [RTR/7]	2 AppWebkit/537.36 (KHTML, like Gecko) Chrome/77.0.3865.98 Safari/537.36 "10.10.66.62:40692" "10.10.148.98:61018" x_forwarded_for:"63.64.74.251, 10.10.66.62" x_forwarded_proto:"https" vcap_request_id:"8772424f-331d-4be6-7013-08ec08a7b065" response_time:0.009642428 app_id:1 https://hooks.slack.com/services/T0000000/B0000000/XXXXXXXXXXXXXXXXXXXX app_index:0 x_b3_traceid:"840b309f860814ea" x_b3_spand:"840b309f860814ea" x_b3_parentspanid:"" b3:"840b309f860814ea" x_b3_version:1.6.1-build.69
Mon Oct 07 2019 [RTR/1]	1 cftest-12322.cfapps.io - [2019-10-07T12:57:48.691+0000] "GET /status HTTP/1.1" 200 0 9 "-" "curl/7.47.0" "10.10.2.135:57310" "10.10.148.98:61018" x_forwarded_for:"18.189.192.159, 10.10.2.135" x_forwarded_proto:"http" vcap_request_id:"ad180f95-0030-4463-6248-8ff70bcab9" response_time:0.00977806 app_id:1 https://hooks.slack.com/services/T0000000/B0000000/XXXXXXXXXXXXXXXXXXXX app_index:0 x_b3_traceid:"534f782768168903" x_b3_spand:"534f782768168903" x_b3_parentspanid:"" b3:"534f782768168903" x_b3_version:1.6.1-build.69
Mon Oct 07 2019 [APP/0]	2 (node:13) [DEP0066] DeprecationWarning: OutgoingMessage.prototype._headers is deprecated
Mon Oct 07 2019 [RTR/10]	1 cftest-12322.cfapps.io - [2019-10-07T12:54:50.059+0000] "GET /status HTTP/1.1" 200 0 9 "-" "curl/7.47.0" "10.10.66.62:36194" "10.10.148.98:61018" x_forwarded_for:"10.100.107.160, 10.10.66.62" x_forwarded_proto:"http" vcap_request_id:"313c0000-3401-4370-8441-5411d4005311" response_time:0.009404204 app_id:1 https://metrics.run.pivotal.io/apps/09f5ea32-6a59-41e6-a971-7c6b7d714d84 app_index:0 x_b3_traceid:"534f782768168903" x_b3_spand:"534f782768168903" x_b3_parentspanid:"" b3:"534f782768168903" x_b3_version:1.6.1-build.69

- Revision: The code and configuration used by an app
- Cloud Foundry API (CAPI) object referencing
  - A droplet
  - A start command
  - Environment variables
- The most recent revision for a running app represents code and configuration currently running in Cloud Foundry
- Referencing a revision in a deployment deploys that revision
- Use cases for revisions:
  - Viewing app history
    - can help you understand how your app has changed over time
  - Roll back:
    - allows you to “un-deploy” a bad version of an app
    - no need to track previous state yourself
- Events that Trigger Creation of a Revision:
  - A new droplet is created for an app (restaging)
  - An app’s environment vars are changed
  - The start command for an app is changed
  - An app rolls back to a prior revision

Copyright 2013-2019, RX-M LLC

# App Revisions

Revision #	Status	Description	Time since last action
> 3	Deployed	New droplet deployed.	1 min
> 2	Not Deployed	New droplet deployed.	2 min
1	Not Deployed	Initial revision.	3 min

# CLI Revisions

Copyright 2013-2019, RX-M LLC

- Cloud Foundry retains 100 revisions per app by default and the 5 most recent droplets

```
ubuntu@ip-172-31-9-24:~/cfar$ cf apps
Getting apps in org rx-m2 / space development as randy.abernethy@rx-m.com...
OK

name      requested state    instances   memory   disk     urls
cf-test   started        1/1        512M     1G       cftest-12322.cfapps.io
my-app    started        1/1        1G       1G       nx5555.cfapps.io
ubuntu@ip-172-31-9-24:~/cfar$ cf app cftest --guid
09f5ea32-6a59-41e6-a971-7cb67d714d84
ubuntu@ip-172-31-9-24:~/cfar$ cf curl /v3/apps/09f5ea32-6a59-41e6-a971-7cb67d714d84/features/revisions -X PATCH -d '{ "enabled": true }'
{
  "name": "revisions",
  "description": "Enable versioning of an application",
  "enabled": true
}
ubuntu@ip-172-31-9-24:~/cfar$ cf curl /v3/apps/09f5ea32-6a59-41e6-a971-7cb67d714d84/revisions
{
  "pagination": {
    "total_results": 4,
    "total_pages": 1,
    "first": {
      "href": "https://api.run.pivotal.io/v3/apps/09f5ea32-6a59-41e6-a971-7cb67d714d84"
    },
    "last": {
      "href": "https://api.run.pivotal.io/v3/apps/09f5ea32-6a59-41e6-a971-7cb67d714d84"
    },
    "next": null,
    "previous": null
  },
  "resources": [
    {
      "guid": "0f38dccc-6f45-4a37-9b83-5d97b0c3e548",
      "version": 1,
      "droplet": {
        "guid": "2b4e09ab-414f-4143-a68f-1f5cdfab27b2"
      },
      "processes": {
        "web": {
          "command": "node app.js"
        }
      },
      "description": "Initial revision.",
      "relationships": {
        "app": {
          "data": {
            "guid": "09f5ea32-6a59-41e6-a971-7cb67d714d84"
          }
        }
      },
      "created_at": "2019-10-07T13:22:19Z",
      "updated_at": "2019-10-07T13:22:19Z",
      "links": {
        "self": {
          "href": "https://api.run.pivotal.io/v3/revisions/0f38dccc-6f45-4a37-9b83-5d97b0c3e548"
        }
      }
    }
  ]
}
```

Revision routes:

- /v3/apps/GUID/features/revisions
- /v3/apps/GUID/revisions
- /v3/apps/GUID/revisions/deployed
- /v3/revisions/GUID

To Rollback POST a deployment

```
cf curl v3/deployments \
-X POST \
-d '{
  "revision": {
    "guid": "REVISION-GUID"
  },
  "relationships": {
    "app": {
      "data": {
        "guid": "APP-GUID"
      }
    }
  }
}'
```

# Downloading and Uploading Droplets

Copyright 2013-2019, RX-M LLC

- cf app source-app --guid
  - Gets app guid
- cf curl /v2/apps/:source-guid/droplet/download --output /tmp/droplet.tgz
  - Downloads app's droplet to local machine
- cf target -s destination-space
  - Target space to push to
- cf push --droplet /tmp/droplet.tgz destination-app-name

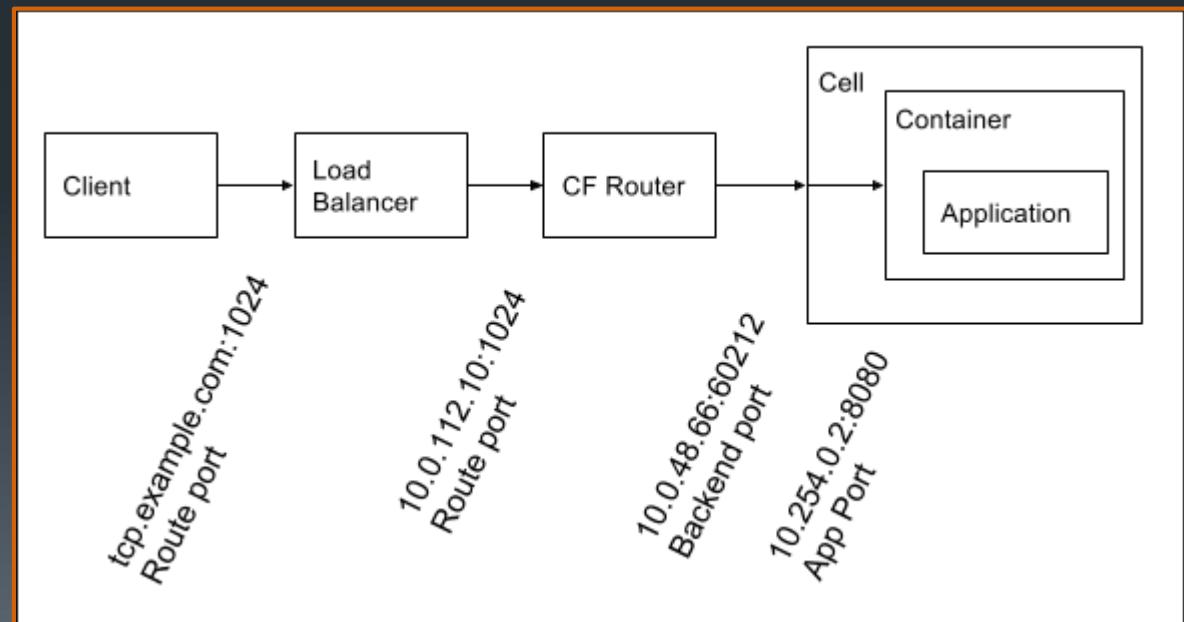
```
ubuntu@ip-172-31-9-24:~/cfar$ cf v3-droplets cftest
This command is in EXPERIMENTAL stage and may change without notice

Listing droplets of app cftest in org rx-m2 / space development as randy.abernethy@rx-m.com...
guid          state    created
2b4e09ab-414f-4143-a68f-1f5cdfab27b2  staged   Mon 07 Oct 12:53:12 UTC 2019
4bcb3e6d-51a4-46b4-8c7a-f4922c911ba9  staged   Mon 07 Oct 13:22:35 UTC 2019
6b0d3f61-9e26-4995-8d99-ecb6f063ee80  staged   Mon 07 Oct 13:23:44 UTC 2019
47d5d850-6799-4880-a66b-09515258273a  staged   Mon 07 Oct 13:24:09 UTC 2019
ubuntu@ip-172-31-9-24:~/cfar$
```

# TCP Routing

Copyright 2013-2019, RX-M LLC

- Create a TCP route for a domain on an arbitrary port with --random-port
  - \$ cf create-route my-space tcp-domain.example.com --random-port
  - Route tcp-domain.example.com:60034 has been created
- To request a specific port use --port
  - The port must not be reserved for another space
  - \$ cf create-route my-space tcp-domain.example.com --port 60035



# Internal Routes

- Apps can communicate without leaving the platform on the container network using internal routes
- Create an internal route with cf map-route
  - After an internal route is mapped to an app the IP addresses of the targets are visible in the app container
  - \$ cf map-route app apps.internal --hostname app
  - \$ cf ssh app
  - vcap@1234:~\$ host app.apps.internal
  - *app.apps.internal has address 10.255.169.200*
  - *app.apps.internal has address 10.255.49.7*
  - *app.apps.internal has address 10.255.49.77*
- To resolve individual instances, prepend the index to the internal route.
  - vcap@1234:~\$ host 1.app.apps.internal
- By default, apps cannot communicate with each other on the container network
  - To allow apps to communicate with each other you must create a network policy

# Network Policy

Copyright 2013-2019, RX-M LLC

- Container-to-Container Networking uses an overlay network
- Each Diego cell in the overlay network is allocated a /24 range that supports 254 containers
- Network policy defines the communications patterns allowed within the overlay network
- cf add-network-policy creates a policy to allow direct network traffic from one app to another
  - cf add-network-policy

```
SOURCE_APP
--destination-app DESTINATION_APP
[-s DESTINATION_SPACE_NAME
[-o DESTINATION_ORG_NAME]]
[--protocol (tcp | udp) --port RANGE]
```

## EXAMPLES

- cf add-network-policy frontend --destination-app backend --protocol tcp --port 8081
- cf add-network-policy frontend --destination-app backend -s backend-space
-o backend-org --protocol tcp --port 8080-8090

# Lab 3

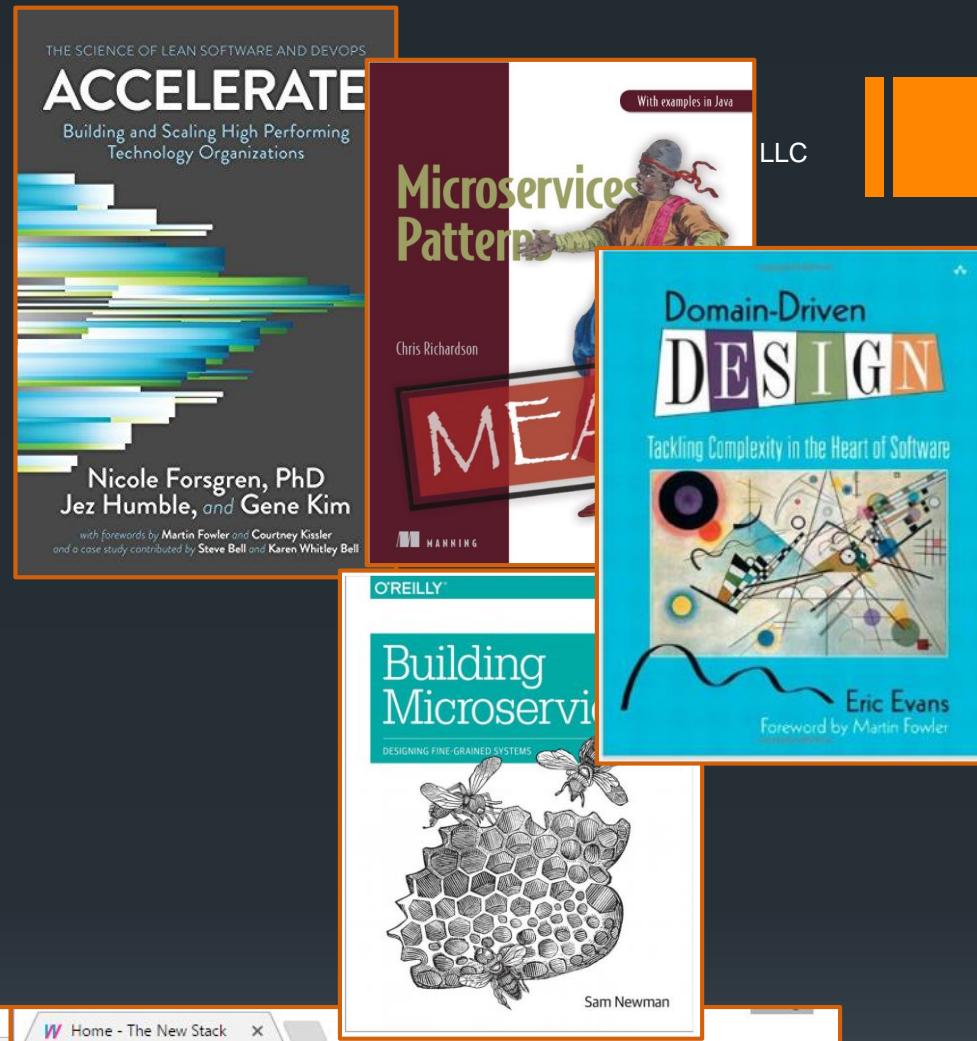
- Deploy microservice lab 6 to Cloud Foundry
  - Simulator nodejs buildpack push
    - => internal connection
  - Nats docker container push
    - => internal connection
  - Report Service nodejs buildpack
    - Route REST API for Internet availability

# The End

- Many thanks for attending!

# Books & Pubs

- Domain Driven Design
  - Eric J. Evans, Addison-Wesley Professional
- Building Microservices
  - Sam Newman, O'Reilly
- Accelerate: The Science of Lean Software and DevOps
  - Building and Scaling High Performing Technology Organizations
  - Nicole Forsgren
- Microservices Patterns
  - Chris Richardson
- The New Stack
  - <https://thenewstack.io>
- Microservices Weekly
  - <https://microserviceweekly.com>



**Microservice Weekly**

A free, weekly newsletter with the best news and articles on microservices. Hand-curated each week.

**THE NEW STACK**

MONITORING & MANAGEMENT WITH DOCKER CONTAINERS  
NEW TECHNOLOGIES FOR DOCKER & CONTAINERS  
FREE DOCKER & CONTAINER EBOOKS  
USE CASES FOR KUBERNETES  
FREE KUBERNETES EBOOK