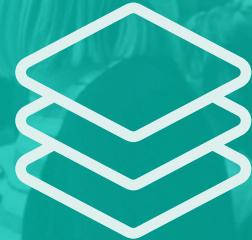


Pivotal®

Cloud Native Intro

The Cloud Native Platform

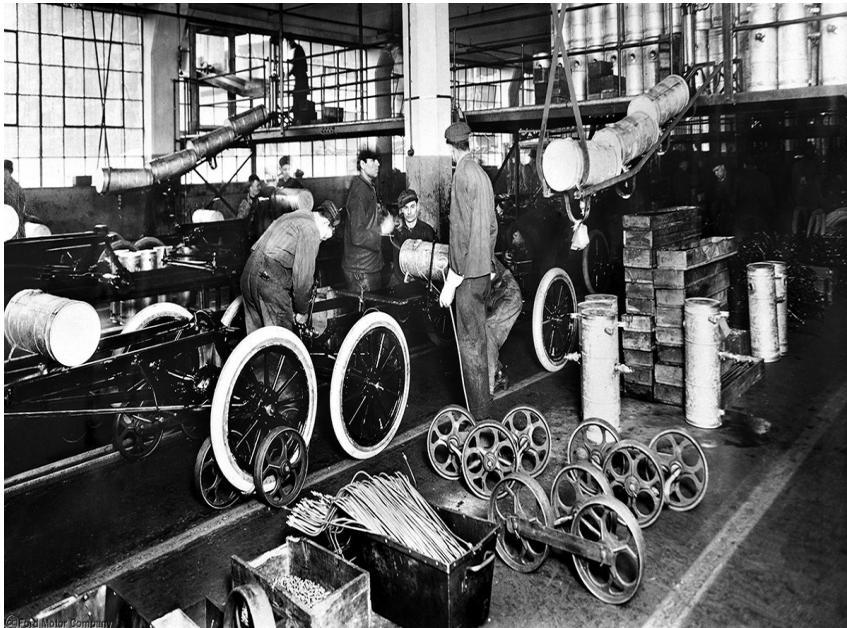
Platform Architecture Team



WE BELIEVE

Transforming how the world builds
Software will shape the future.

Then



- Manual assembly line
- Dozens of features

Now



- Fully automated assembly lines
- Thousands of features

Weeks/Months

Traditional Software Deployment Models

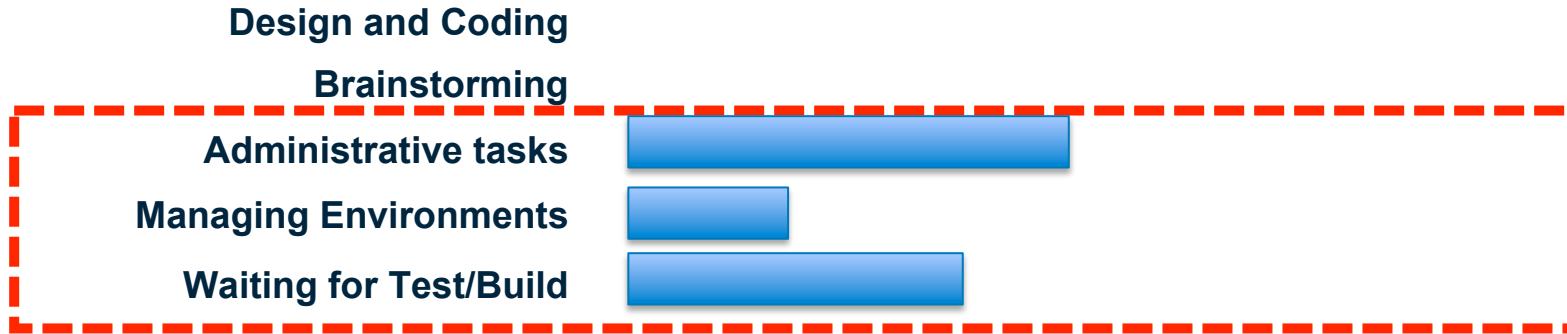
Are you able to do full continuous delivery for every major applications today?

- Dev team unable to iteratively build, deploy and test **early**
- Operations teams **overwhelmed** with requests & processes
- Resources in Dev & Ops **tied up** or they **multitask** between the projects
- Businesses seriously impacted by:

Difficulty 2 release quickly → Slower time to market → **Lost Opportunity**

Lack of on-demand scaling capabilities → Downtime → **Lost Revenue**

Software Developers Spend too Much Time NOT Writing Software



Average Hours/Week

17.5 hrs ~ 40% of full work week!

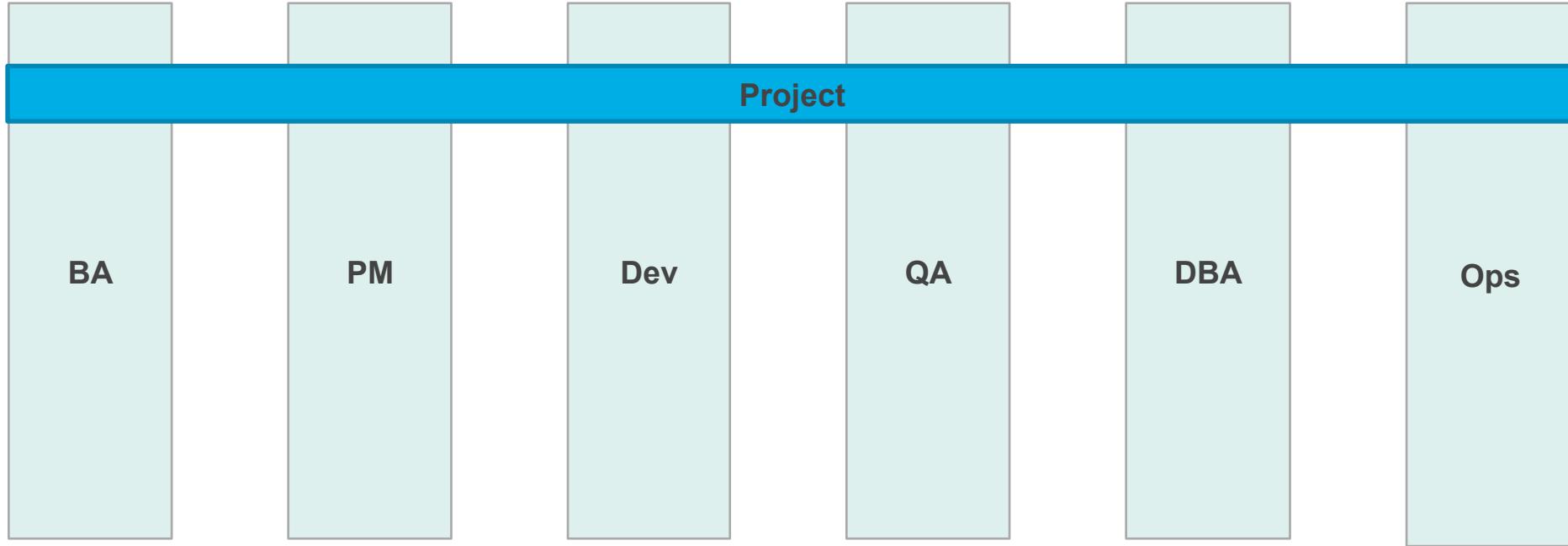
*Sources: “Electric Cloud LinkedIn Survey to software developers”



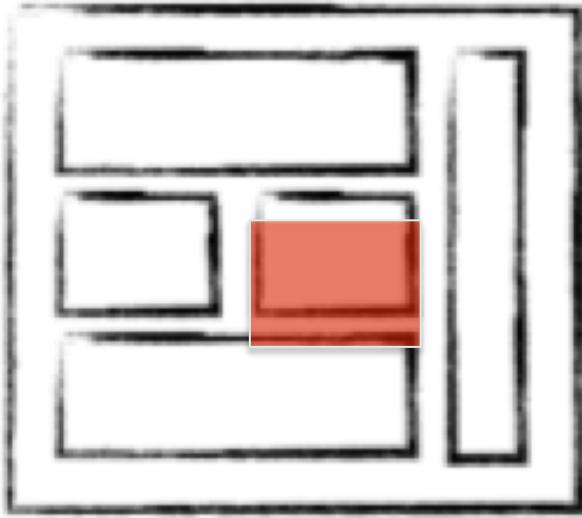
Wealthfront Engineering



Traditional Enterprise Silos



Monolith Applications



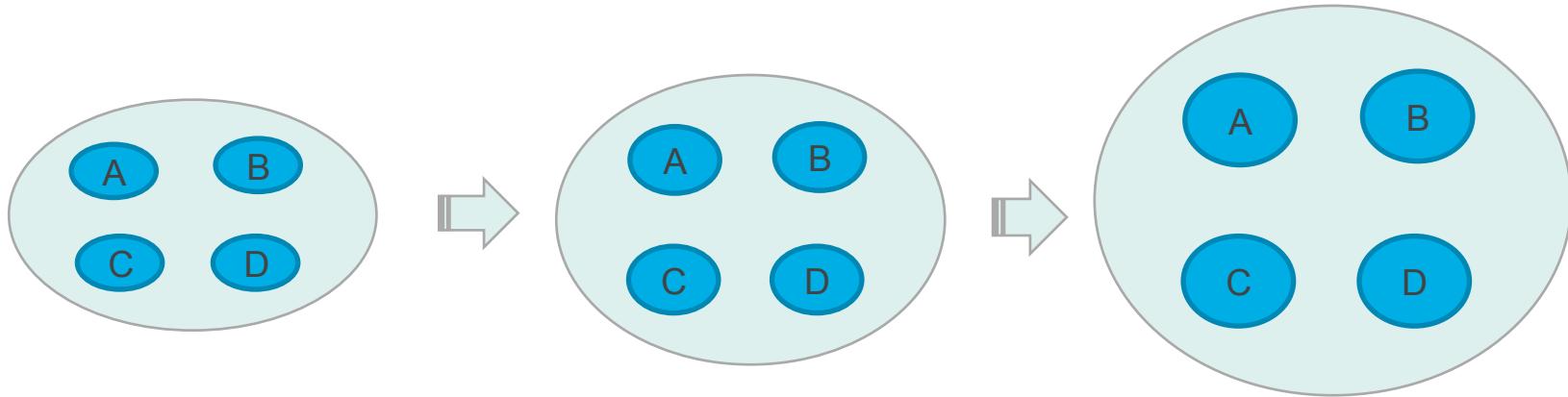
Great for the first deployment ...

What about the updates ?

- Tightly coupled - Holistic approach
- Reliability - Adding layers & change processes
- Backwards compatibility - Regression testing

MONOLITHIC/LAYERED

Monolith Technology Choice & Scaling

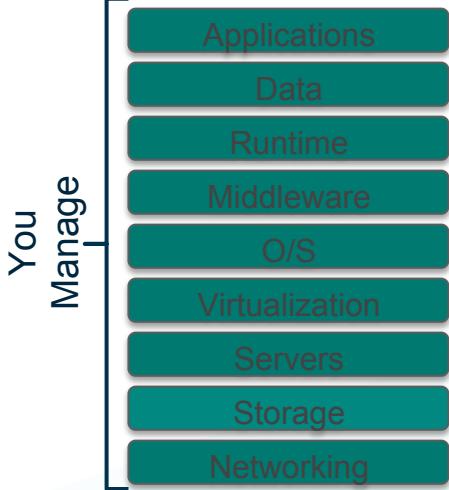


- A is mission critical used all the time with predictable load
- B is mission critical with spiky load
- C is mission critical with low usage
- D is not mission critical and is occasionally used

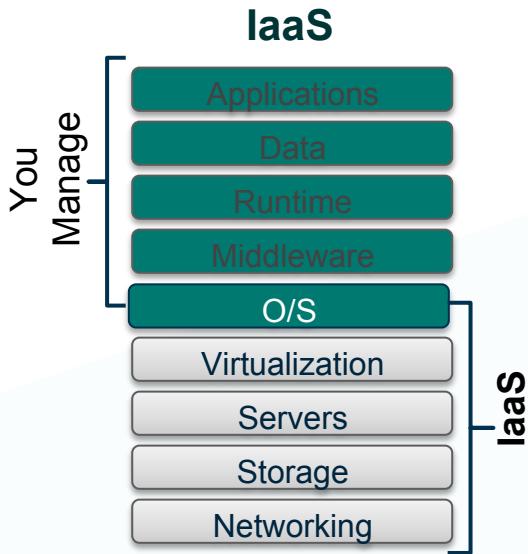
The Cloud Platform Evolution

Business Value, Agility &
Cost Savings

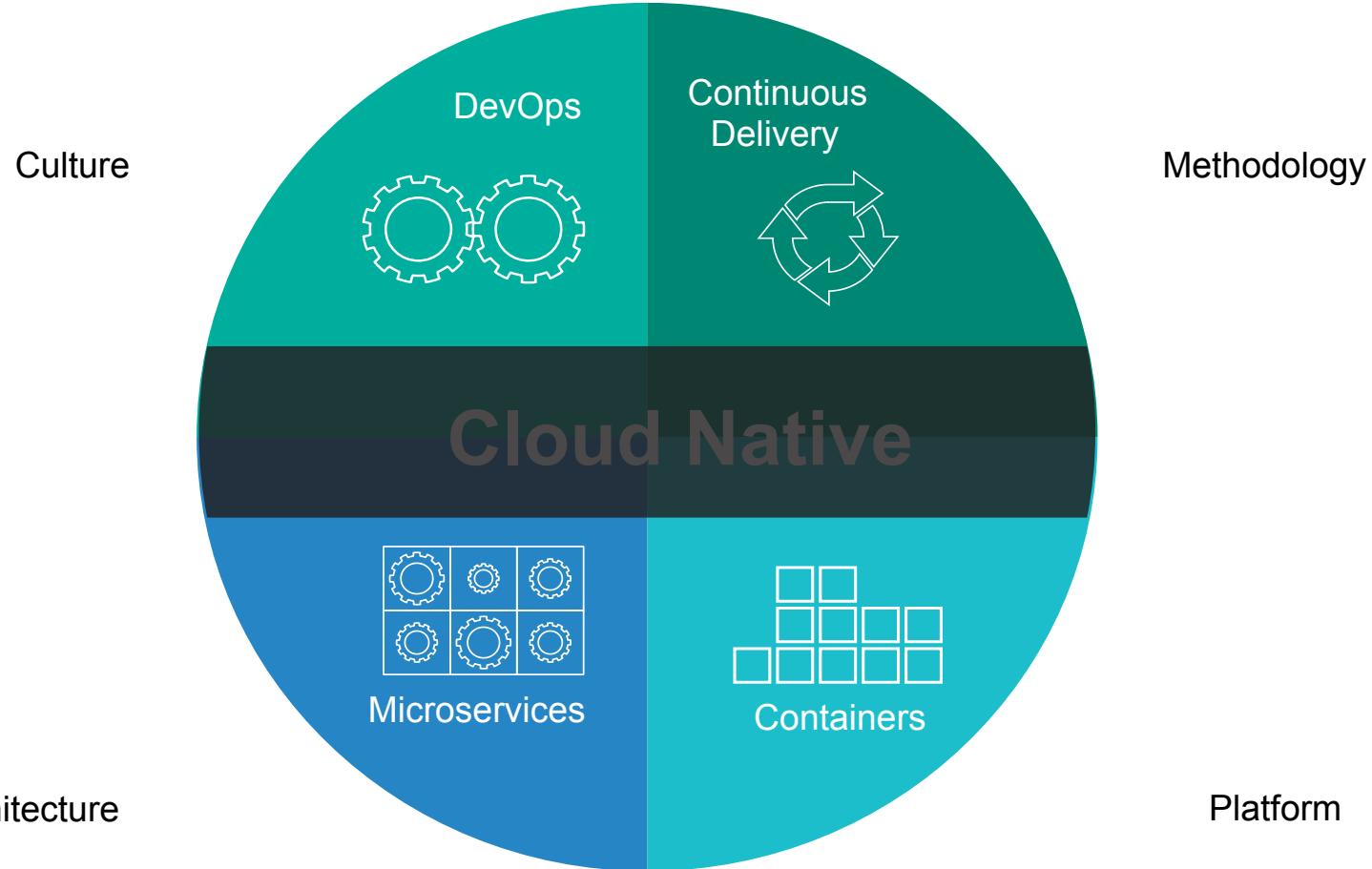
Traditional IT



IaaS



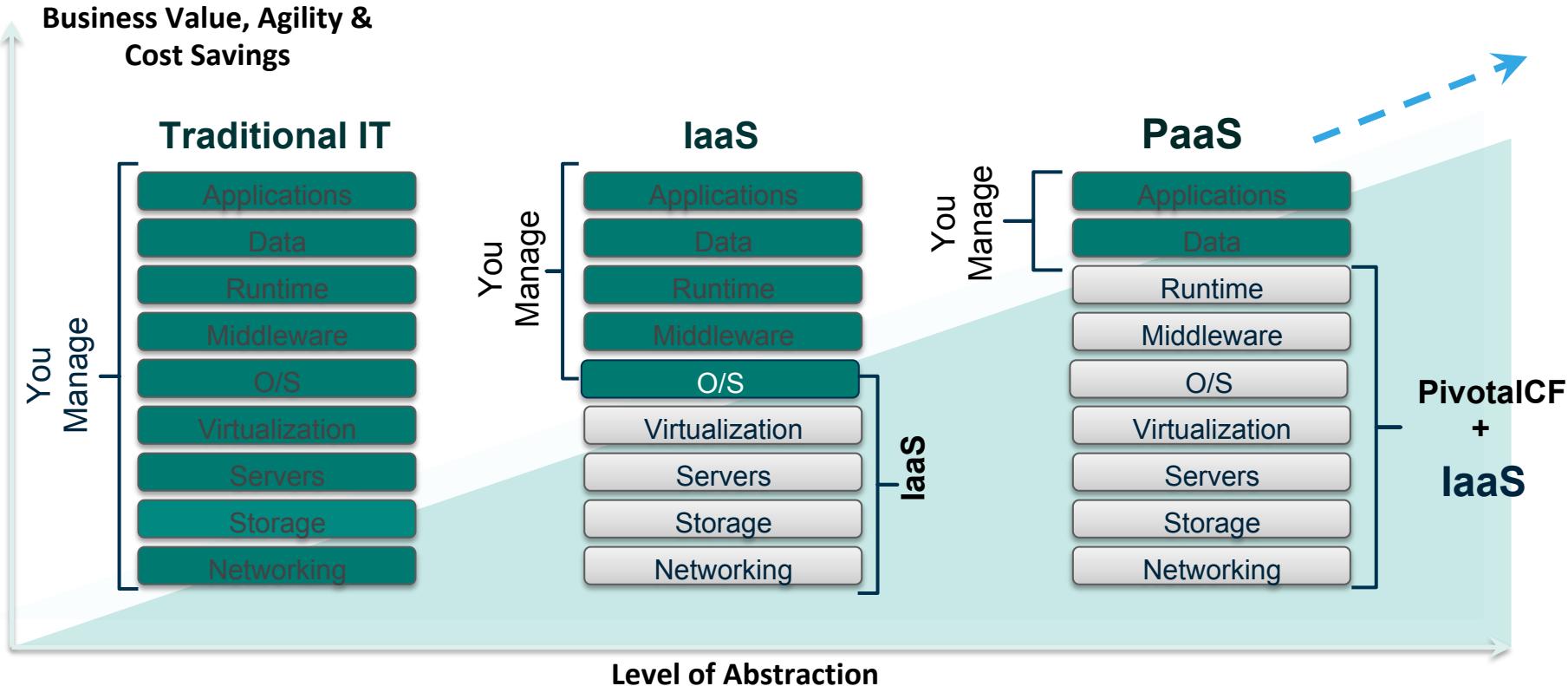
Level of Abstraction



Cloud Native - Resolving IT Gridlock



The Cloud Platform Evolution



Deploying Apps Shouldn't Be Painful

Traditional

- Provision a VM, IP, DNS
- Install Web Server
- Install Frameworks
- Deploy Application
- Configure Load Balancer
- Configure SSL Termination
- Configure Firewall
- Configure Monitoring
- Configure Logging

Pivotal Cloud Foundry

`cf push`

The developer dream haiku

Here is my source code,

Run it in the Cloud for me,

I do not care how

What Happened ?

```
D:\pcf\PCF-demo-1>cf push
Using manifest file D:\pcf\PCF-demo-1\manifest.yml
Updating app pcfdemo-1 in org PCF-Org-01 / space development as H141869...
OK
Using route pcfdemo.apps.pcf.dce [REDACTED]
Uploading app files from: D:\pcf\PCF-demo-1\target\pcfdemo.war
Uploading 615.4K, 66 files
Done uploading
OK
Binding service myrabbit to app pcfdemo-1 in org PCF-Org-01 / space development as H141869...
OK
Binding service mylogger to app pcfdemo-1 in org PCF-Org-01 / space development as H141869...
OK
Binding service myscaler to app pcfdemo-1 in org PCF-Org-01 / space development as H141869...
OK
Stopping app pcfdemo-1 in org PCF-Org-01 / space development as H141869...
OK
Starting app pcfdemo-1 in org PCF-Org-01 / space development as H141869...
----> Downloaded app package (8.5M)
----> Downloaded app buildpack cache (4.0K)
    Java Buildpack Version: v3.0 (offline) | https://github.com/cloudfoundry/java-buildpack.git#3bd15e1
    Downloading Open Jdk ORL 1.8.0_40 from https://download.run.pivotal.io/openjdk/trusty/x86_64/openjdk-1.8.0_40.tar.gz (found in cache)
    Expanding Open Jdk JRE to .java-buildpack/open_jdk_jre (0.9s)
    > Downloading Spring Auto Reconfiguration 1.7.0.RELEASE from https://download.run.pivotal.io/auto-reconfiguration/auto-reconfiguration-1.7.0.RELEASE.jar (found in cache)
    Modifying WEB-INF/web.xml for Auto Reconfiguration
    Downloading Tomcat instance 8.0.21 from https://download.run.pivotal.io/tomcat/tomcat-8.0.21.tar.gz (found in cache)
    Building Tomcat 8.0.21 (0.1s)
    > Downloading Tomcat Logging Support 2.4.0.RELEASE from https://download.run.pivotal.io/logging-support/tomcat-logging-support-2.4.0.RELEASE.jar (found in cache)
    > Downloading Tomcat Access Logging Support 2.4.0.RELEASE from https://download.run.pivotal.io/logging-support/tomcat-access-logging-support-2.4.0.RELEASE.jar (found in cache)
    > Uploading droplet (8.0M)
1 of 1 instances running
App started
OK

app pcfdemo-1 was started using this command "JAVA_HOME=$PWD/.java-buildpack/open_jdk_jre JAVA_OPTS="-Djava.io.tmpdir=$TMPDIR:OnOutOfMemoryError=-Xms382293K -Xmx382293K -XX:MaxMetaspaceSize=64M -XX:MetaspaceSize=64M -Xss595K -Djava.security.egd=file:///dev/urandom -Daccess.logging.enabled=false -Dhttp.port=$PORT" $PWD/.java-buildpack/tomcat/bin/catalina.sh start"
Starting health and status for app pcfdemo-1 in org PCF-Org-01 / space development as H141869...
OK

requested state: started
instances: 1/1
usage: 512M x 1 instances
urls: pcfdemo.apps.pcf.dce [REDACTED]
last uploaded: Mon Aug 3 00:19:12 UTC 2015
stack: cflinuxfs2

state  since          cpu   memory       disk      details
#0  running  2015-08-02 07:19:38 PM  0.0%  345.8M of 512M  138.3M of 1G
D:\pcf\PCF-demo-1>
```

Firewall & Routes

Service Bindings

App Lifecycle

Runtime Installation & Config

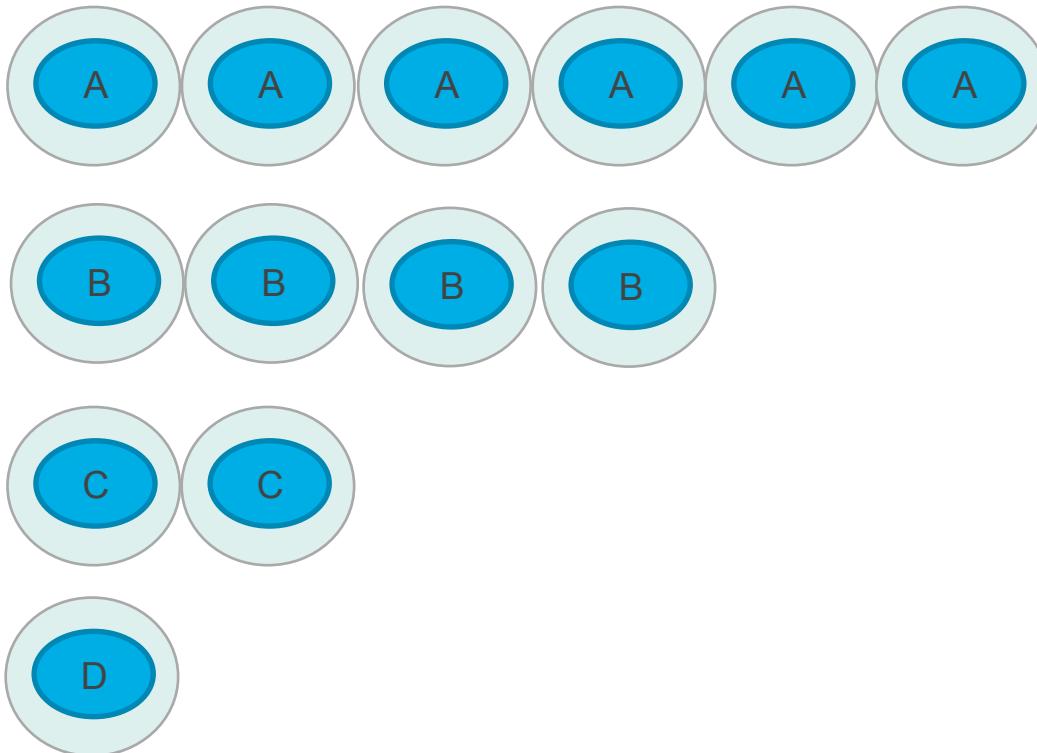
Middleware Installation & Config

Application Installation & Config

App Lifecycle

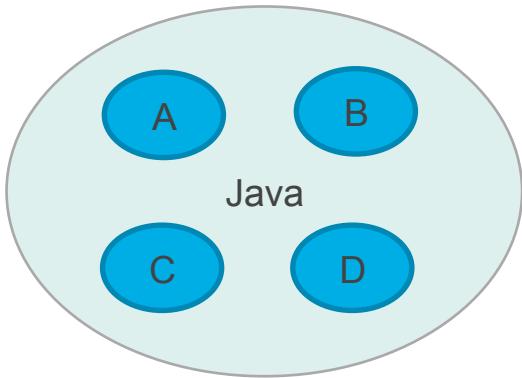
Logging, Health, Telemetry

Microservices Scaling Out



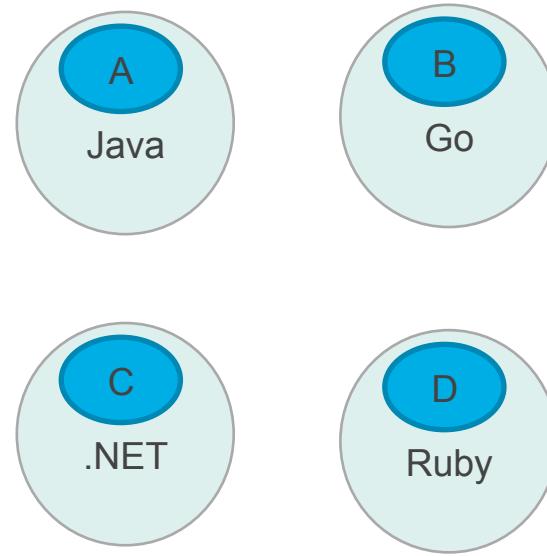
Technology Choices

Then



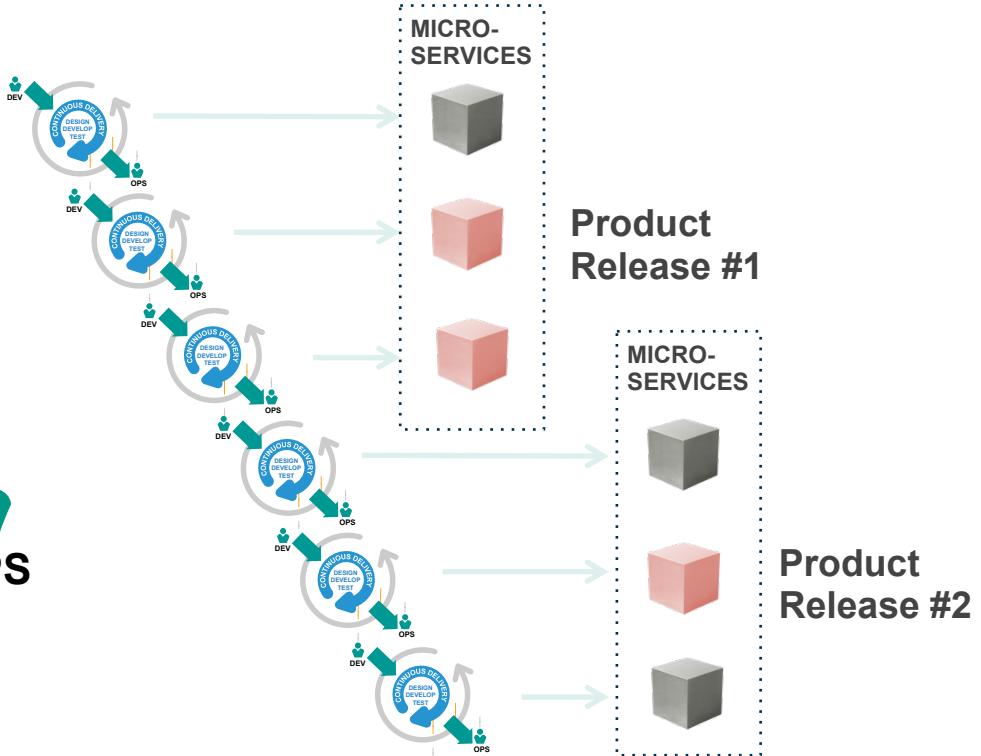
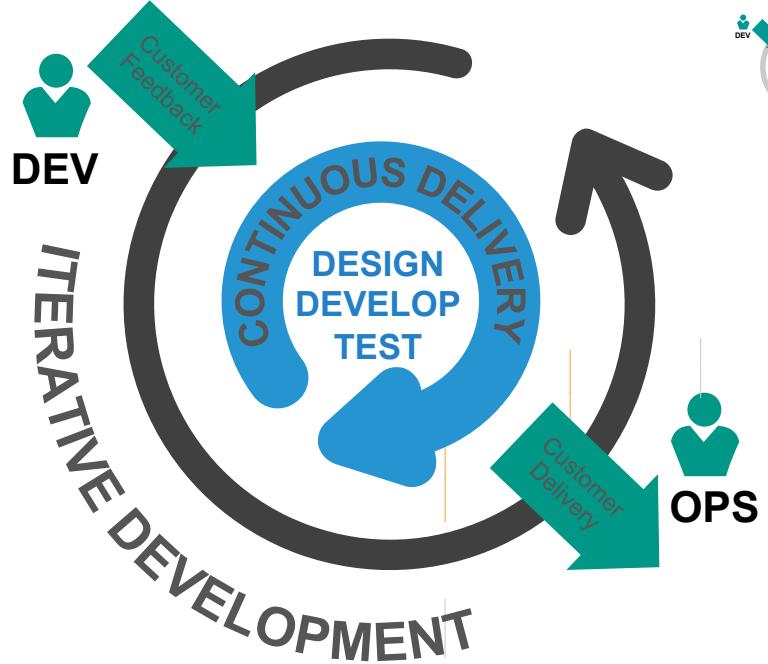
Monolithic Application
implemented using a
single technology

Now



Each microservice can be implemented using
the **best technology** for the feature

Continuous Delivery → Time To Market & Flexibility



DevOps – Separation of Concerns

DEVELOPER

target <my cloud>
push <my app>
bind <my services>
scale <my app> +1000

The screenshot shows the Pivotal CF interface. In the top navigation bar, 'development' is selected. Under 'APPLICATIONS', there is a table with columns: Status, App, Instances, and Memory. One row shows 'spring-music' with 100 instances and 128MB memory. Under 'SERVICES', there is a table with columns: Service Instance, Service Plan, and Bound Apps. It lists 'mysql staging' and 'rabbitdev' both using the 'Pivotal MySQL Dev 100mb' plan. On the right side, under 'TEAM', a list of users is shown: James.Jameson@gmail.com, admin, frank.taylor@gopivotal.com, jtim@hgmail.com, martin.lawles@yahoo.com, mospman@gopivotal.com, rboshman@gopivotal.com, and susan.branson@gmail.com. At the bottom, a footer bar includes links for 'PRIVACY' and '© 2014 Pivotal Software, Inc. All Rights Reserved. Version 1/26/2014 (ccb6f3)'.



App Deployment: 30-90 seconds

OPERATOR

provision cloud <Public/Private>
provision service <PaaS,Hadoop...>
upgrade/update <my cloud>
scale <my cloud>

The screenshot shows the Pivotal Ops Manager interface. On the left, a sidebar lists 'Available Products': Pivotal RabbitMQ (No upgrades available), Ops Manager Director for VMware vSphere (No upgrades available), Pivotal Elastic Runtime (No upgrades available), HD (No upgrades available), Metrics (No upgrades available), and Pivotal MySQL Dev (No upgrades available). Below this is a 'Import a Product' button and a note: 'Download Pivotal CF compatible products at Pivotal Network'. On the right, the 'Installation Dashboard' shows several cards: 'Ops Manager Director for vmware vSphere' (v1.1.0), 'Pivotal RabbitMQ' (v1.1.0, highlighted with a red dashed border), 'Pivotal Elastic Runtime' (v1.1.0, also highlighted with a red dashed border), 'HD' (v1.1.0), 'MySQL Dev' (v1.1.0), and 'Pivotal Ops Metrics' (v1.1.0).



Cloud Deployment: 2-4 hours



Pivotal®

Transforming How The World Builds Software