



RED HAT DEVELOPERS

AppDev State of the Industry

bit.ly/appdevstate

[@burrsutter](https://twitter.com/burrsutter)

<http://developers.redhat.com>

[@burrsutter](https://twitter.com/burrsutter)

Change History

0.1 - Created for Brno, CZ at DevConf.cz Jan 27 2017

Buzzword Bingo

Challenges of Digital Transformation

Cloud: Docker, Kubernetes & OpenShift

Embrace Automation: CI/CD, Testing, DevOps

Microservices: Go Fast or Die Trying!

Deployment Mastery: Blue/Green, Canary - Go Fast and Live Trying!

A/B and Hypothesis-Driven Development

Embrace the Full Stack

New JVM Runtimes: Dropwizard, Spring Boot, Wildfly Swarm, Vert.x

Reactive - Streams (IoT, Kafka, CQRS/Event-sourcing, even GET/POST looks like a stream a scale)

Deep/Machine Learning & AI

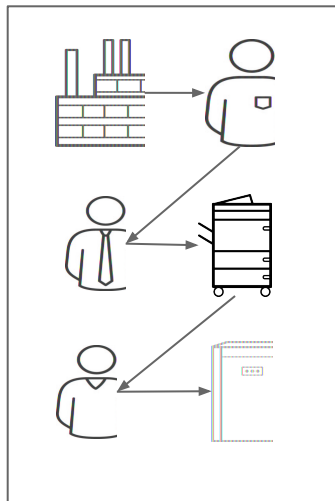
Digital Darwinism

As Deloitte's Shift Index shows, the average **life expectancy** of a **Fortune 500 company** has **declined** from around 75 years half a century ago to **less than 15 years today**.

Professor Richard Foster of Yale University estimates that “by 2020, more than three-quarters of the S&P 500 will be companies that we have not heard of yet.” (Lean Enterprise)

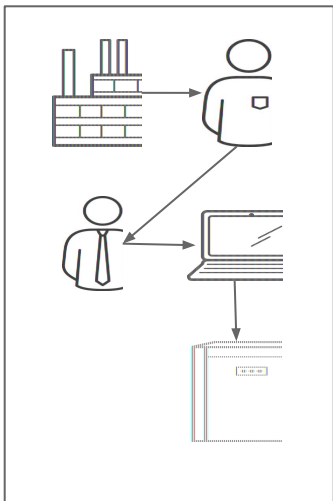
Business & IT: 80's to Future

ChUI



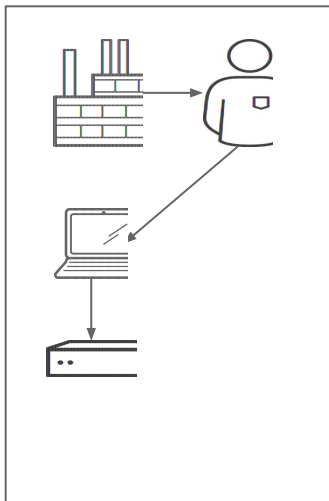
80's

GUI



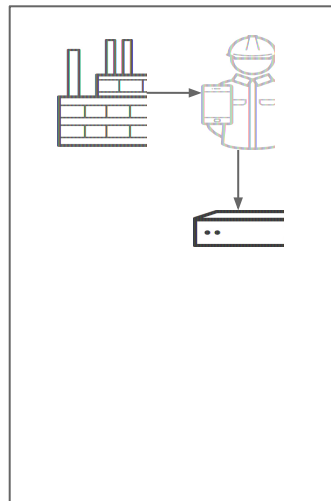
90's

Web



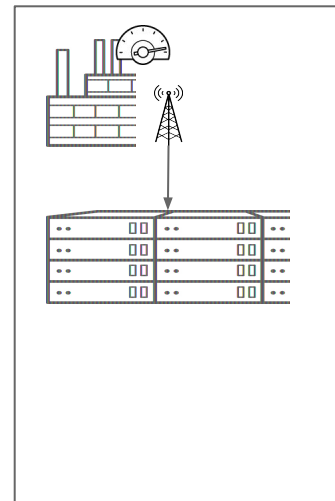
00's

Mobile



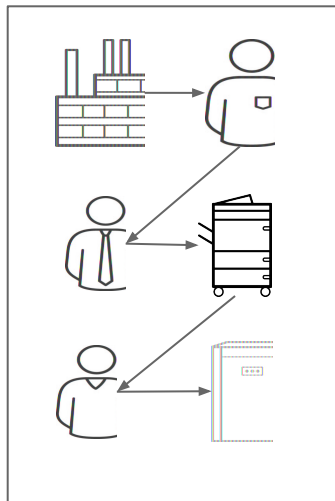
10's

Things



Business & IT: 80's to Future

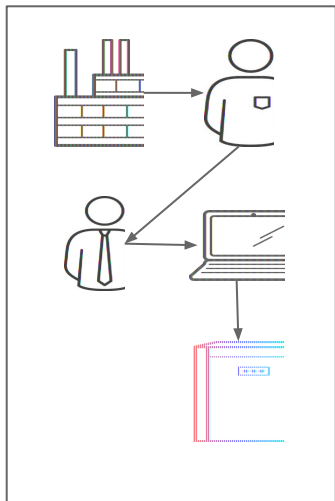
ChUI



80's



GUI

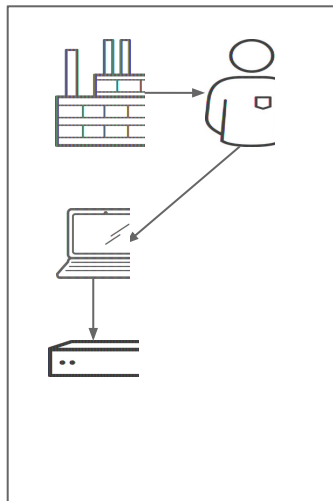


90's



Microsoft

Web

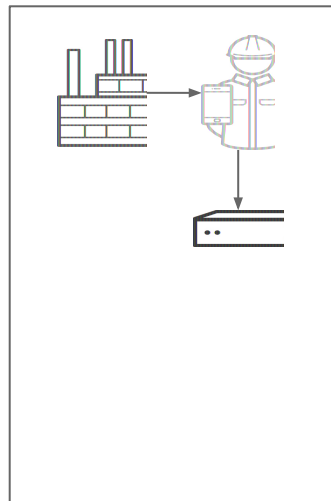


00's



vmware

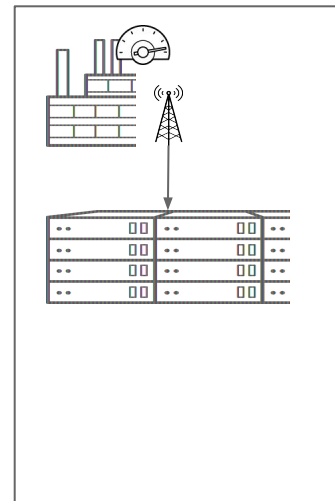
Mobile



10's



Things



amazon



NETFLIX

Are **We** Ready?

87% of surveyed executives believe digital technologies will disrupt their industries yet only 44% of surveyed executives indicated their organizations were taking appropriate measures to avert digital disruption.

Succinctly stated in a July 2016 MIT Sloan research report on digitization and digital transformation, “**Talent is leaving at the same time that it is in short supply**”—with a **mere 11%** of surveyed organizations across industries indicating that their **existing in-house talent has the competitive skills** necessary for success in the digital economy. For the most digitally-advanced companies, that number rose only to 20%.

* MIT Sloan Management Review, “Moving Digital Transformation Forward,” Infographic, 2016.

** G. C. Kane, D. Palmer, A. N. Phillips, D. Kiron and N. Buckley, “Aligning the Organization for Its Digital Future” *MIT Sloan Management Review* and Deloitte University Press, July 2016, p6-8.

Transformers

On Black Friday the Walmart servers didn't go over 1% CPU utilisation and the team did a deploy with 200,000,000 users online.



<http://www.nearform.com/nodecrunch/node-js-becoming-go-technology-enterprise/>

Key findings

1. **High-performing organizations are decisively outperforming their lower-performing peers in terms of throughput.** High performers deploy 200 times more frequently than low performers, with 2,555 times faster lead times. They also continue to significantly outperform low performers, with 24 times faster recovery times and three times lower change failure rates.
2. **High performers have better employee loyalty, as measured by employee Net Promoter Score (eNPS).** Employees in high-performing organizations were 2.2 times more likely to recommend their organization to a friend as a great place to work, and 1.8 times more likely to recommend their team to a friend as a great working environment. Other studies have shown that this is correlated with better business outcomes.

High-performing IT organizations
report experiencing:



200x more frequent
deployments



24x faster
recovery from failures



3x lower
change failure rate



2,555x shorter lead
times

We must open and free our minds--our hearts--our imaginations. In releasing the old--old habits, old assumptions, old ways--we become lighter and faster, sharper and more nimble, powerful and relevant. Wherever we connect, we are transformed into agents of change. authors of what matters. makers of the future.

Burr

Your Journey to Awesomeness :-)



Self-Service,
On-Demand,
Elastic,
Infrastructure
as
Code
(Cloud)



Re-Org to
DevOps



Automation
Puppet, Chef,
Ansible
and/or
Containers



CI & CD
Deployment
Pipeline



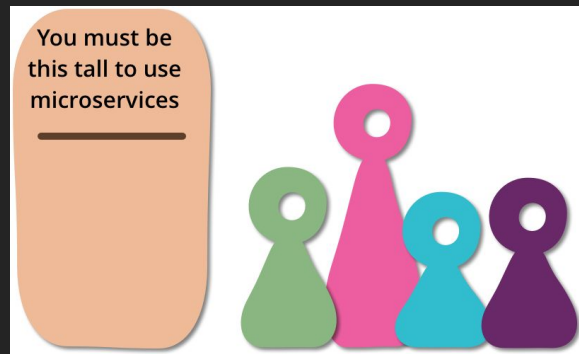
One
Microservice



Silicon
Valley
DotCom
Startup

You Must Be This Tall

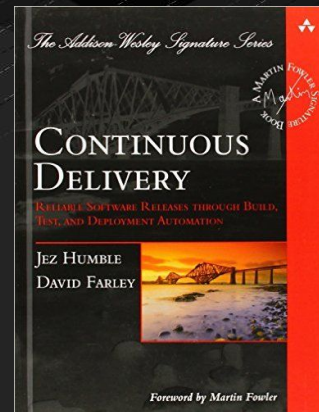
1. Self-Service, on-demand, elastic infrastructure as code
(how many days/weeks to provision a new VM?)
2. Dev vs Ops
(who is on the pager for production app outage?)
3. Automation
(phoenix vs snowflake?)
4. CI & CD
5. Deployment Pipeline



<http://martinfowler.com/bliki/MicroservicePrerequisites.html>

Jez Humble Continuous Integration

1. Software (trunk) is always deployable
2. Everyone is checking into trunk daily (at least), not feature branches
3. If the build breaks it is fixed in 10 minutes (all hands on deck)
4. A new engineer can be on-boarded in 1 day - with a production-like environment on the developer workstation
5. Deployment is a low-risk push button affair
6. NOT just about using Jenkins



Jez Humble Continuous Delivery

Continuous Delivery is the ability to get changes of all types—including new features, configuration changes, bug fixes and experiments—into production, or into the hands of users, *safely* and *quickly* in a *sustainable* way.



WALL OF CONFUSION







The illustration shows a server room with several people working. One person is at a desk with a computer, another is standing near a server rack, and others are interacting. A large stack of servers is in the center, with a red fire-like effect on top, suggesting a crisis or emergency situation.

The Phoenix Project

A Novel About IT, DevOps, and Helping Your Business Win

Gene Kim, Kevin Behr and George Spafford

The DevOps Handbook

HOW TO CREATE WORLD-CLASS AGILITY, RELIABILITY, & SECURITY IN TECHNOLOGY ORGANIZATIONS

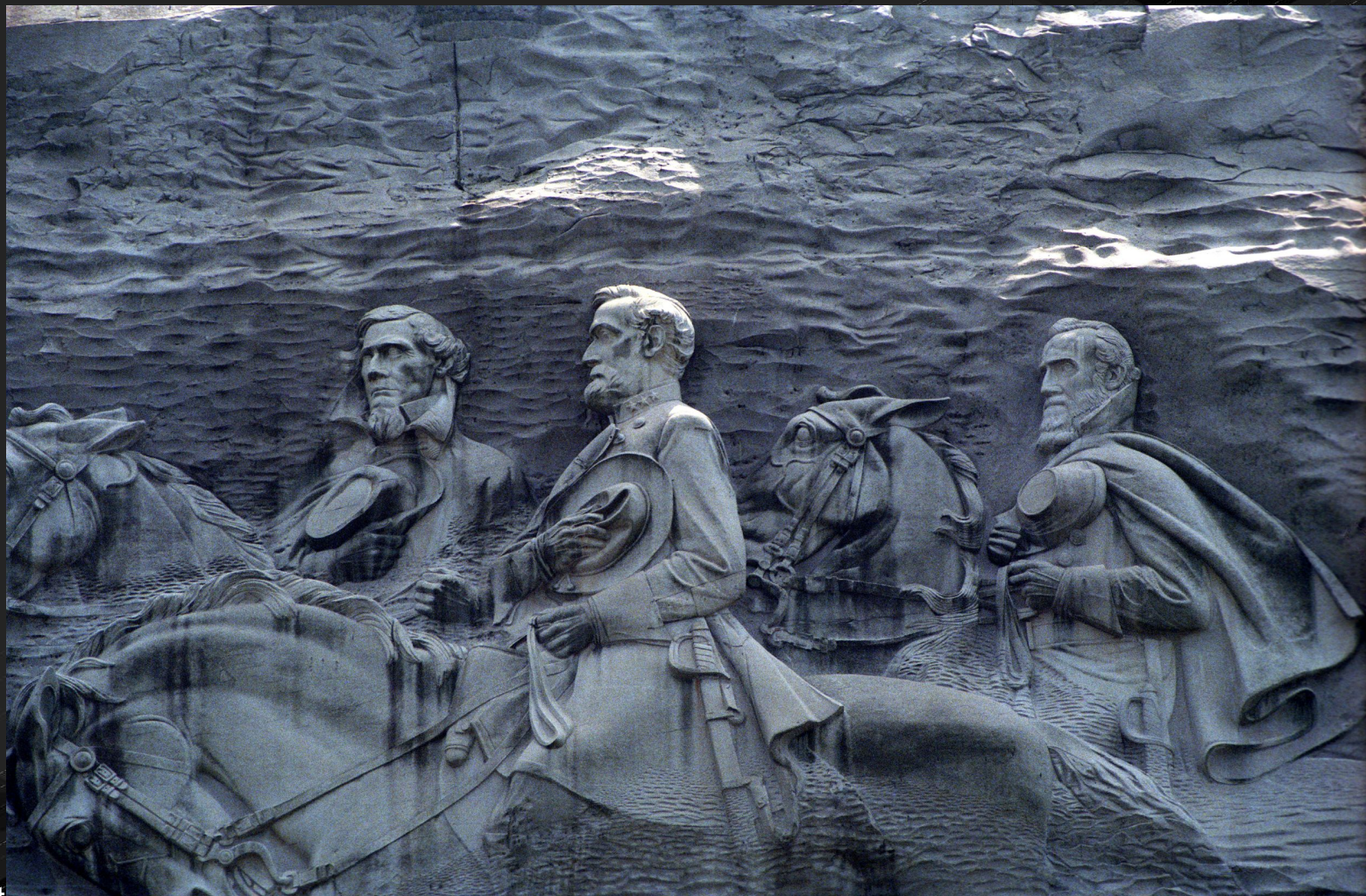


GENE KIM,
JEZ HUMBLE,
PATRICK DEBOIS,
& JOHN WILLIS

FOREWORD BY JOHN ALLSPAUGH

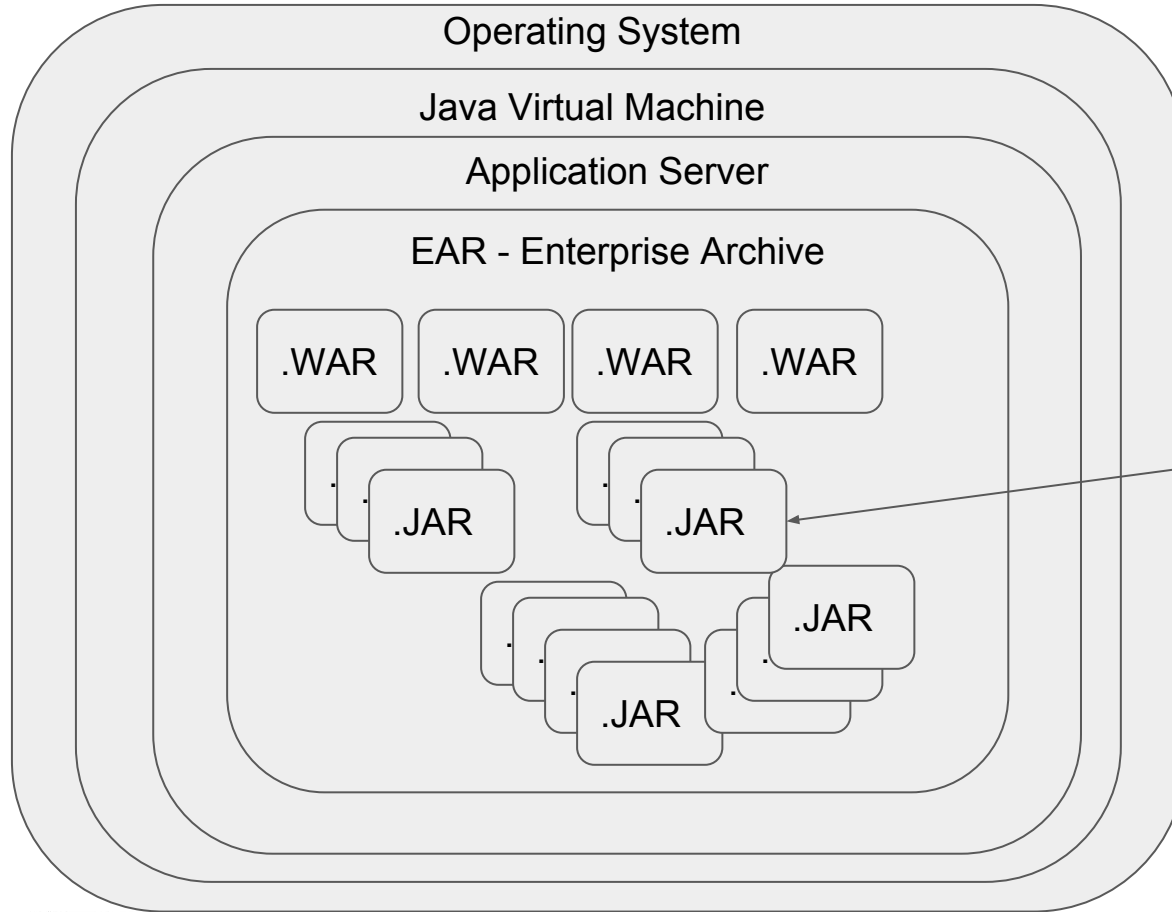
TAKE THE DORA DEVOPS X-RAY ASSESSMENT AND SEE WHERE YOU STAND.





Java EE
(formerly
known as
J2EE)

Monolith

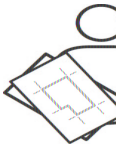




Programmers
(18)



Business
Analysts
(4)



Project
Managers
(2)



Quality
Assurance
(6)



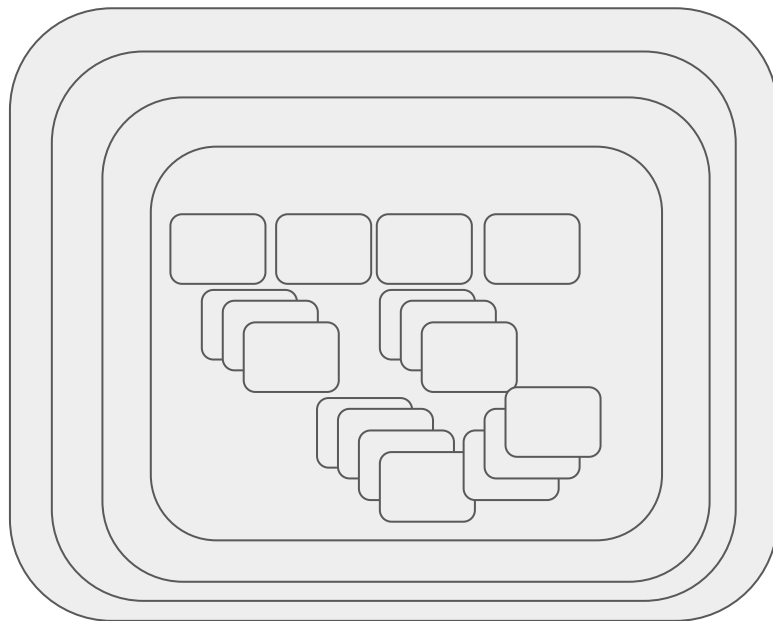
Compliance
(2)

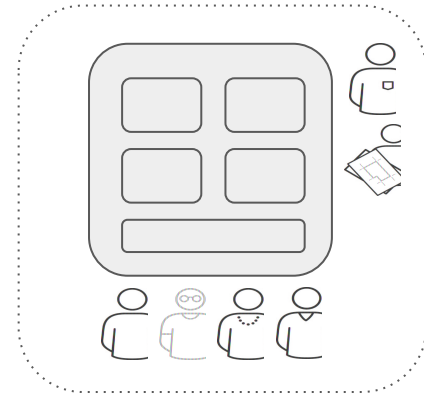
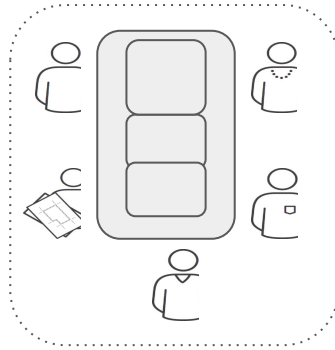
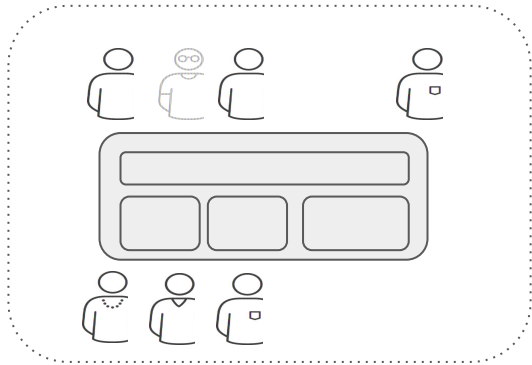


Operators
(6)

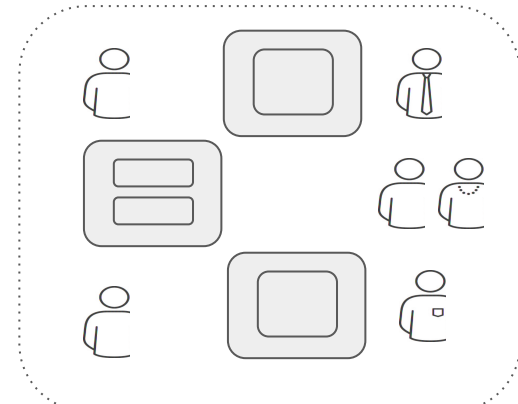
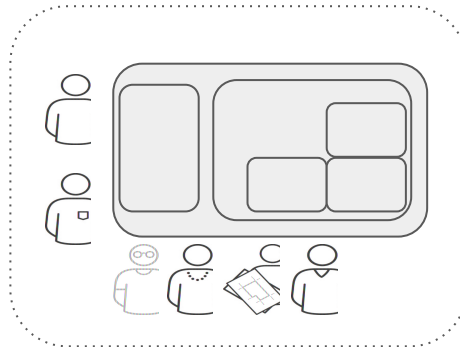
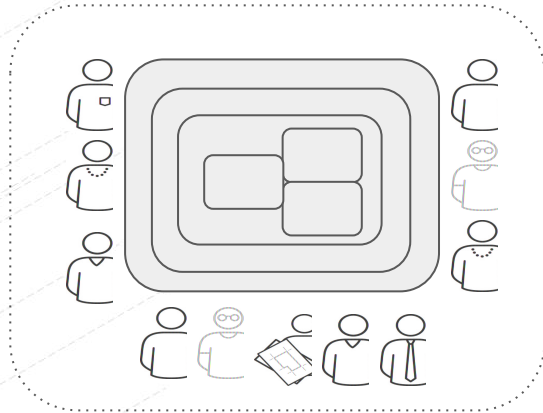


DBAs
(3)





2-Pizza Teams You Build It, You Own it



Microservice Principles/Characteristics

1. Deployment **Independence**: updates to an individual microservice have no negative impact to any other component of the system. Optimized for **Replacement**.
2. Organized around **business** capabilities
3. **Products** not Projects
4. **API-Focused**
5. **Smart** endpoints and dumb pipes
6. Decentralized Governance
7. Decentralized Data Management
8. Infrastructure Automation (infrastructure as code)
9. Design for failure
10. Evolutionary Design

“DOCKER AS PROCESS, PAAS AS MACHINE, MICROSERVICES ARCHITECTURE AS PROGRAMMING MODEL”

The microservices style of architecture highlights rising abstractions in the developer world because of containerization and the emphasis on low coupling, offering a high level of operational isolation. Developers can think of a container as a self-contained process and the PaaS as the common deployment target, using the microservices architecture as the common style. Decoupling the architecture allows the same for teams, cutting down on coordination cost among silos. Its attractiveness to both developers and DevOps has made this the de facto standard for new development in many organizations

“software development is still mostly a communication problem”

<https://assets.thoughtworks.com/assets/technology-radar-nov-2016-en.pdf>

bit.ly/msa-instructions

Friday - January 27

**12 Factors to cloud
success**

 Rafael Benevides

 Cloud

 1:00:00

Saturday - January 28

**Develop & Deploy
Java apps as resilient
Microservices
Architectures**

 Rafael Benevides

 JUDCon

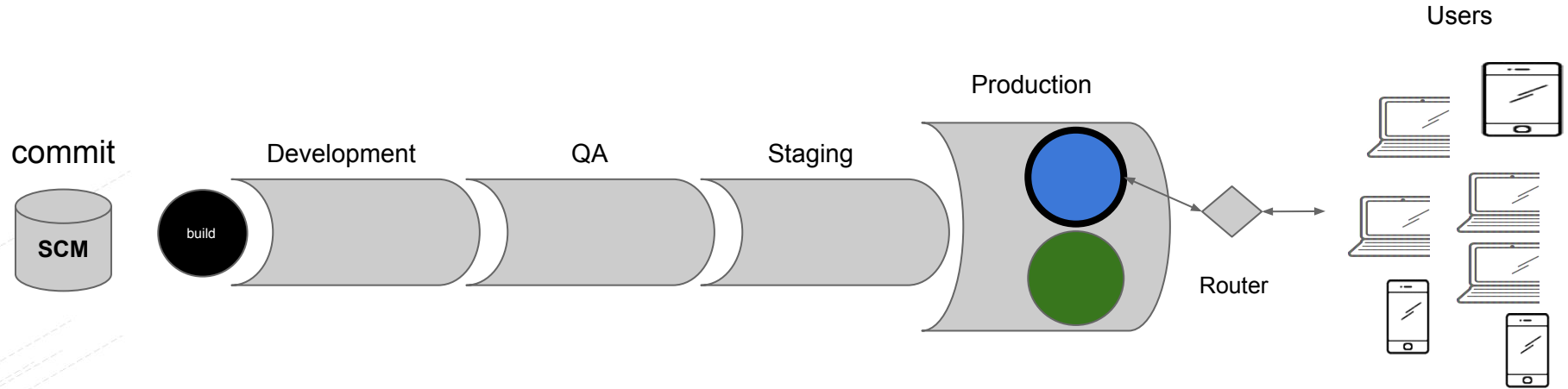
 1:00:00

Purpose Of Pipeline

Jez Humble

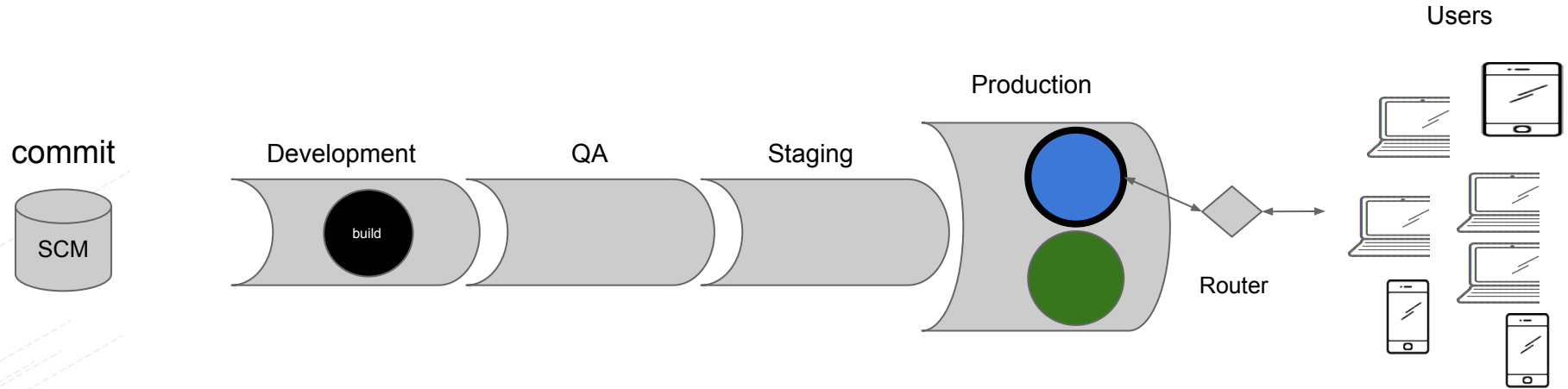
“Job of deployment pipeline is to prove that the release candidate is unreleasable”

Blue/Green Deployment 1

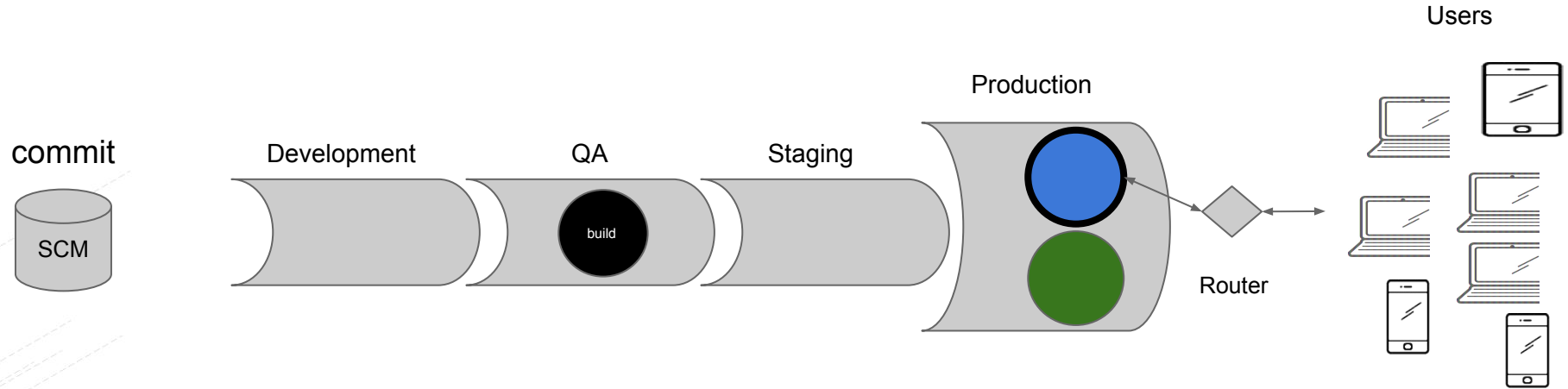


Starts with a “git commit and git push”

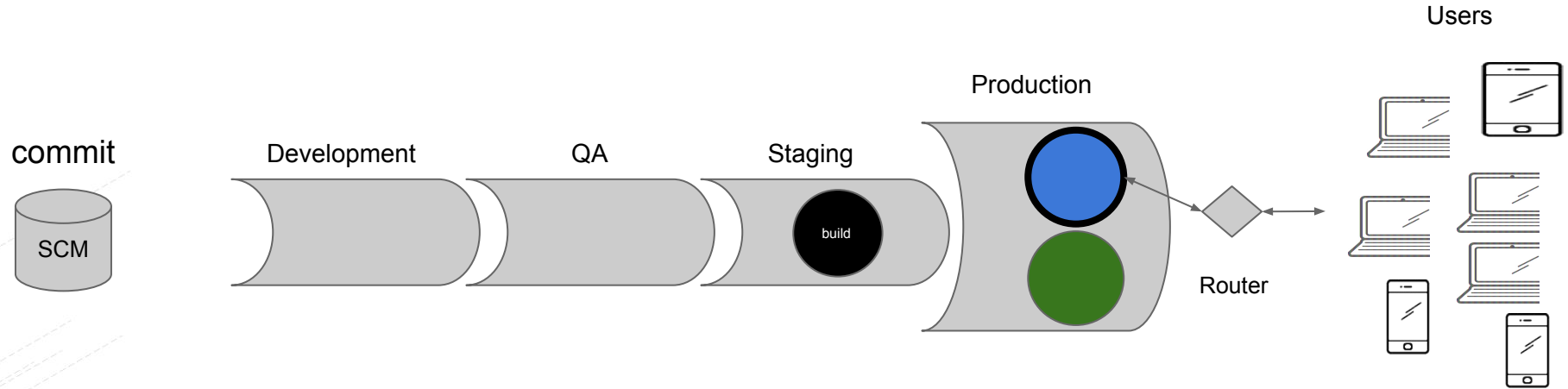
Blue/Green Deployment 2



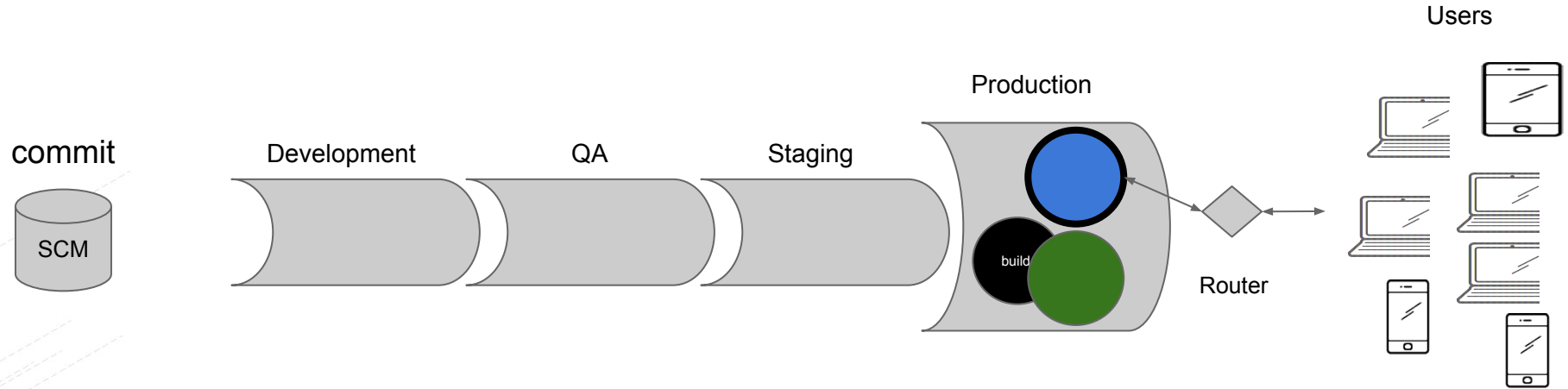
Blue/Green Deployment 3



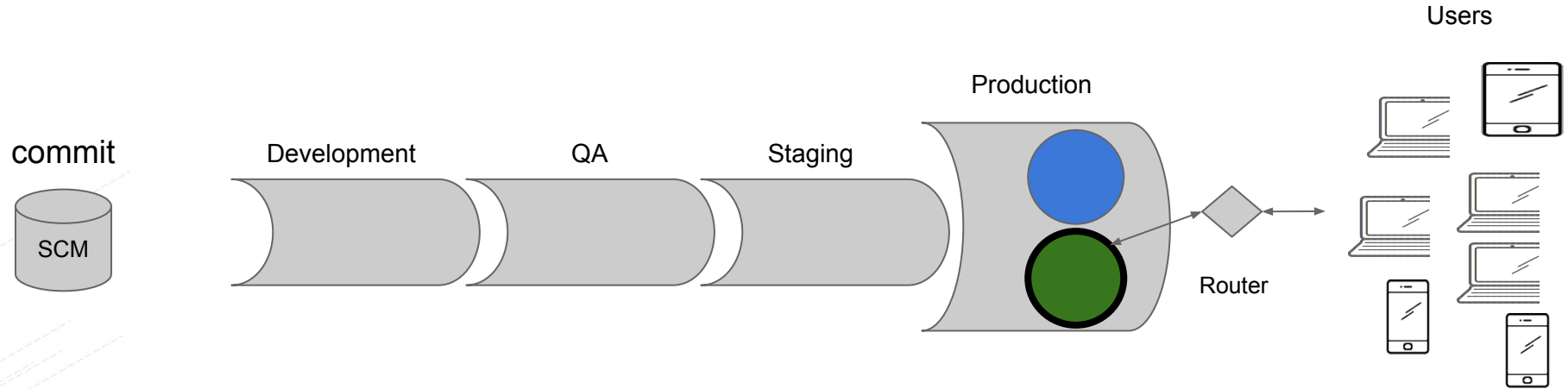
Blue/Green Deployment 4



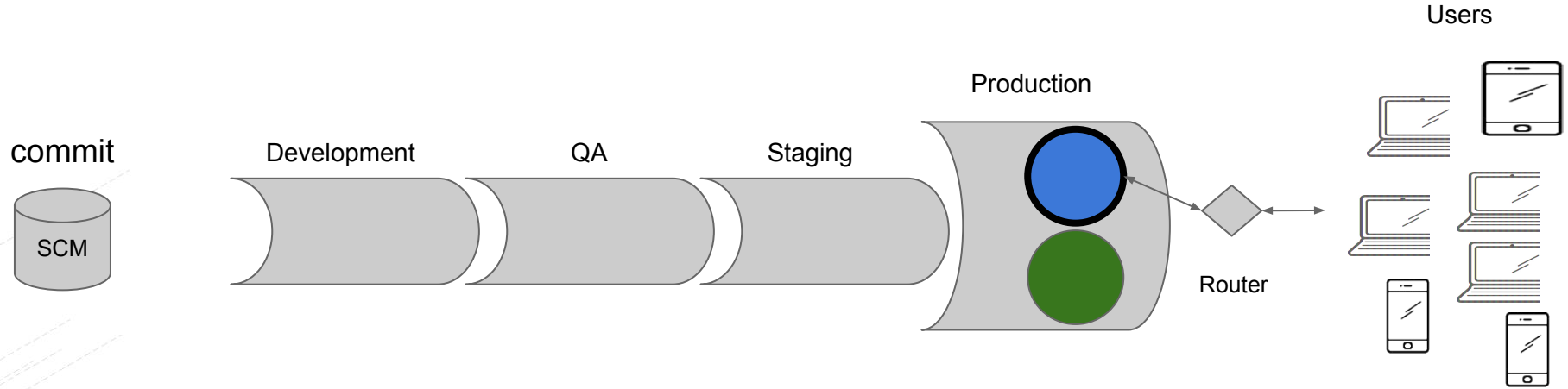
Blue/Green Deployment 4



Blue/Green Deployment 4



Blue/Green Deployment Fallback



$\frac{1}{3}$ Positive Value-Add

“Our experience at Microsoft is no different: only about $\frac{1}{3}$ of ideas improve the metrics they were designed to improve.”

Ronny Kohavi

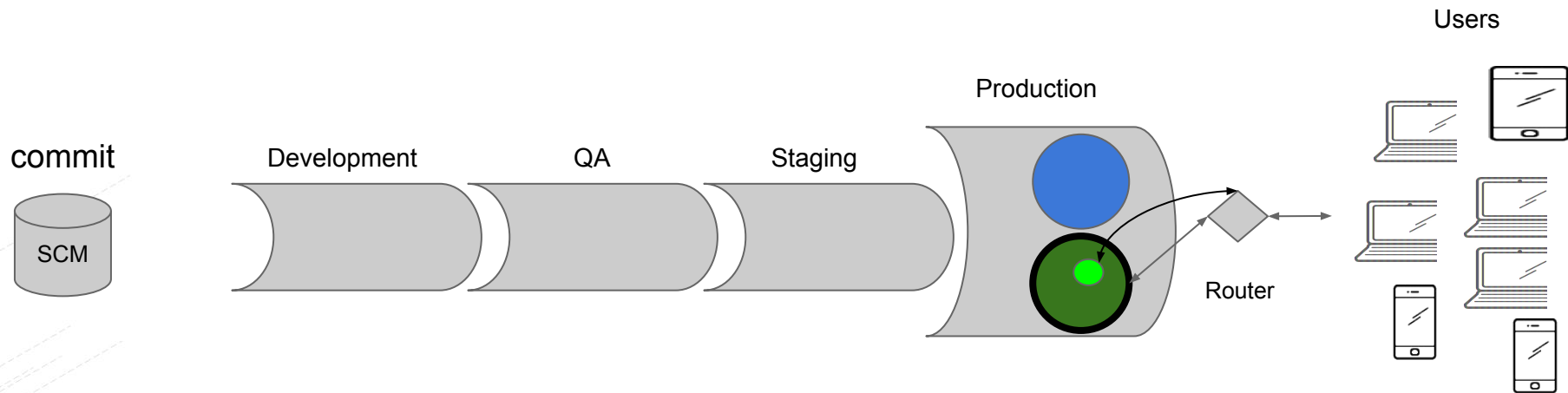
Microsoft (formerly Amazon)

<http://ai.stanford.edu/~ronnyk/ExPThinkWeek2009Public.pdf>

What is your

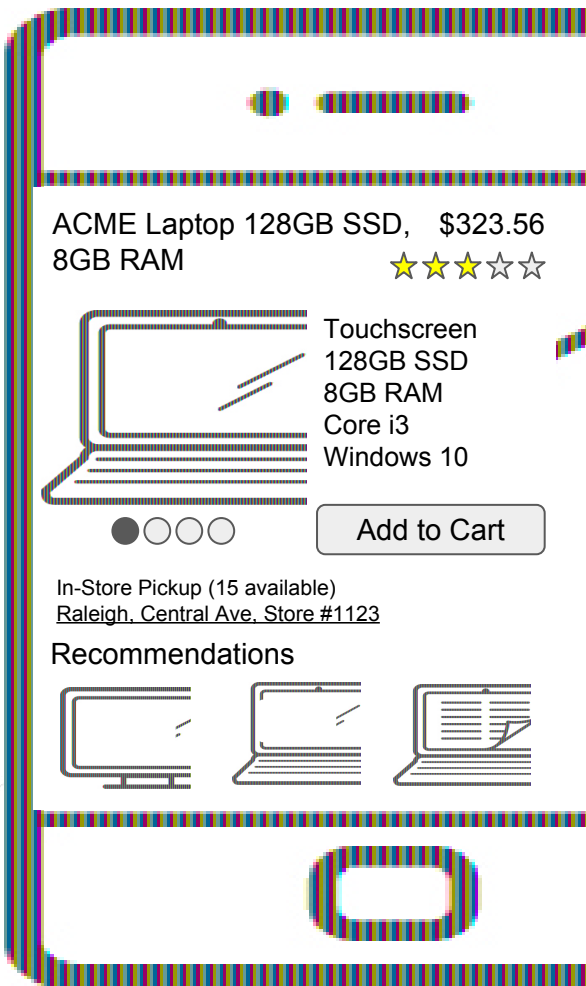
Hypothesis?

A/B Testing

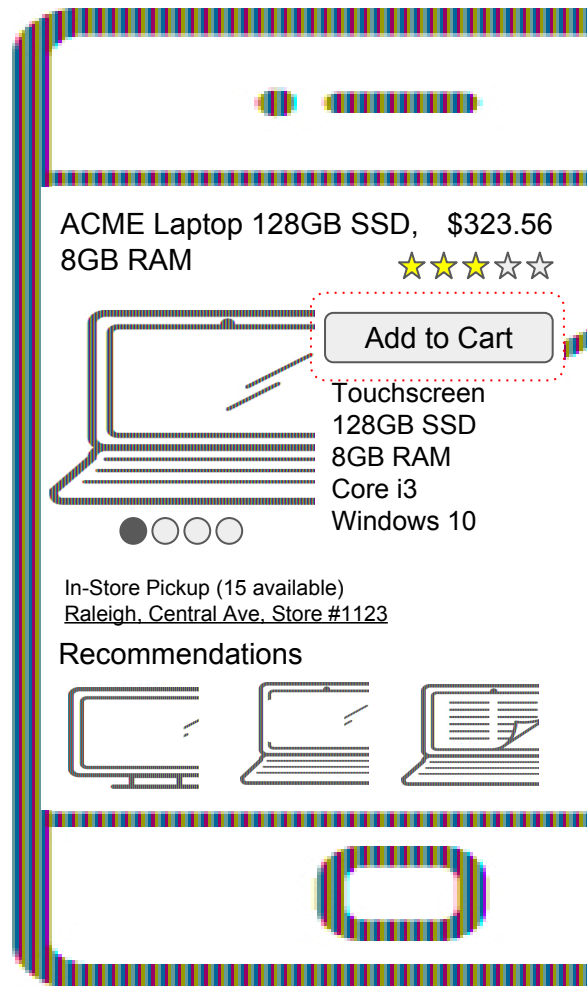


Hypothesis - Experiment - two variants: check-out button placement
increase sales conversions for iOS users?

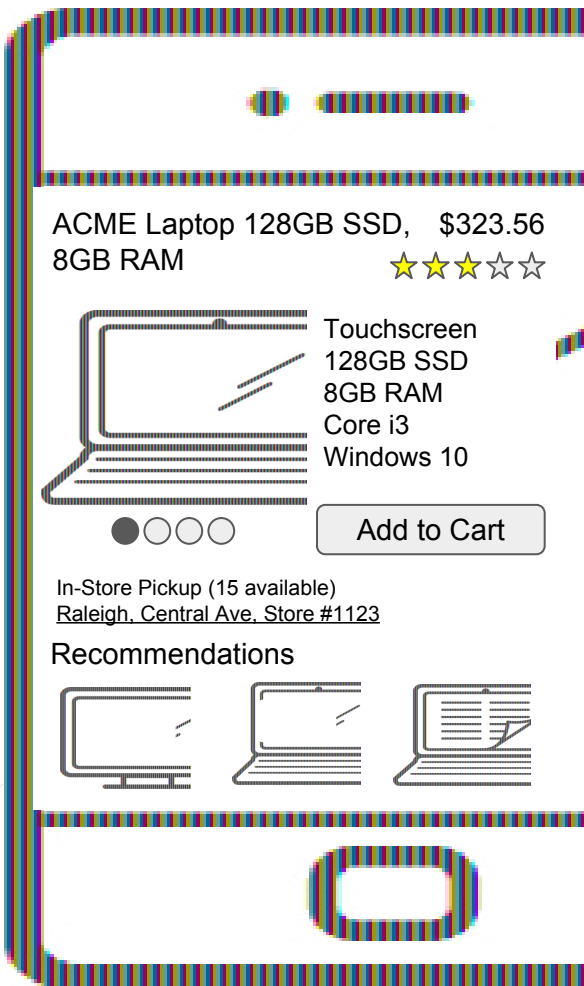
A



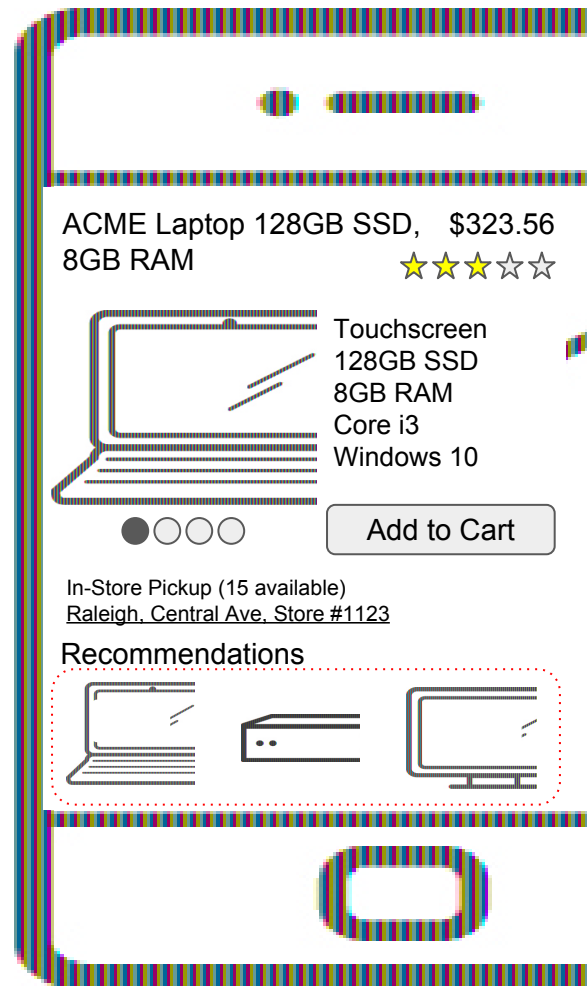
B



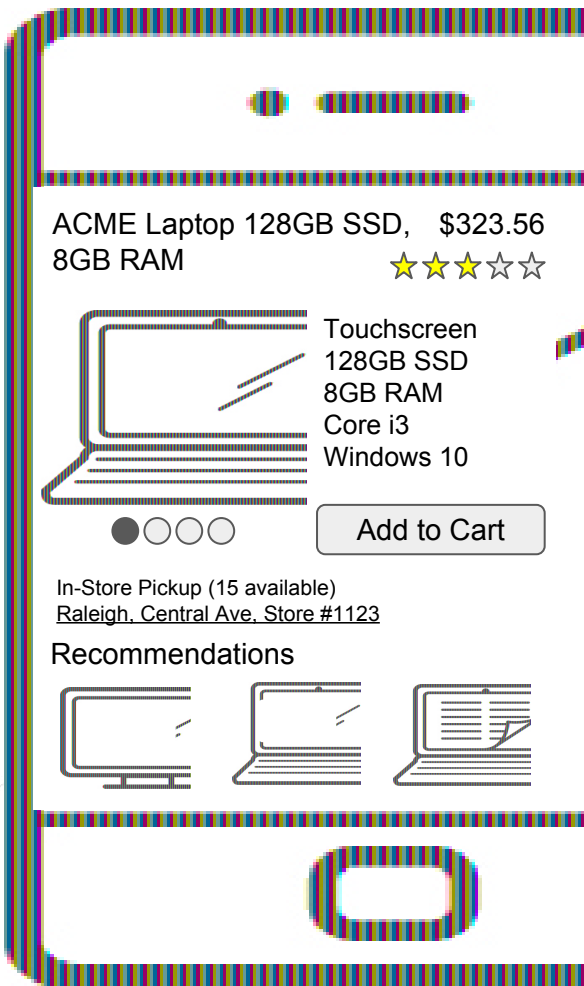
A



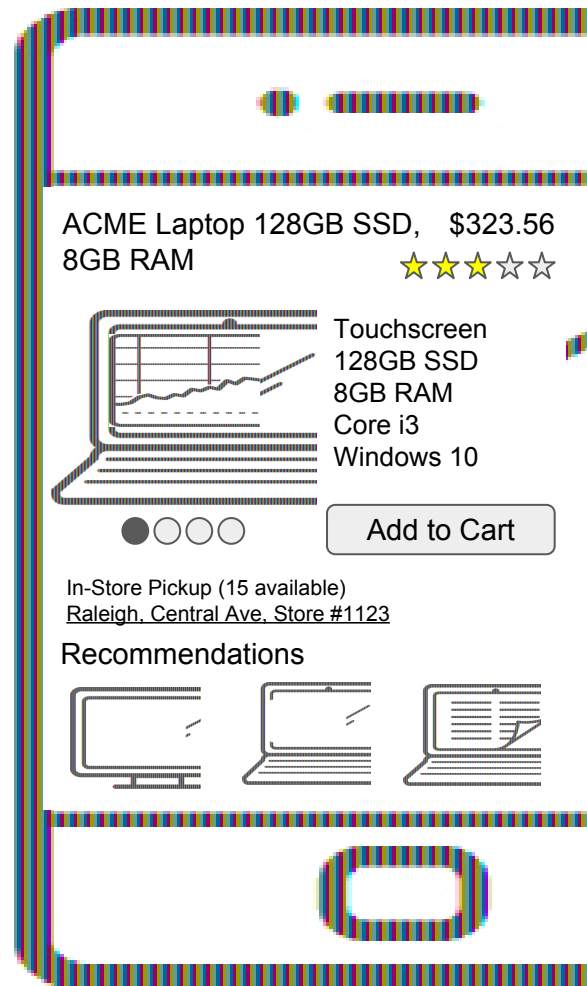
B



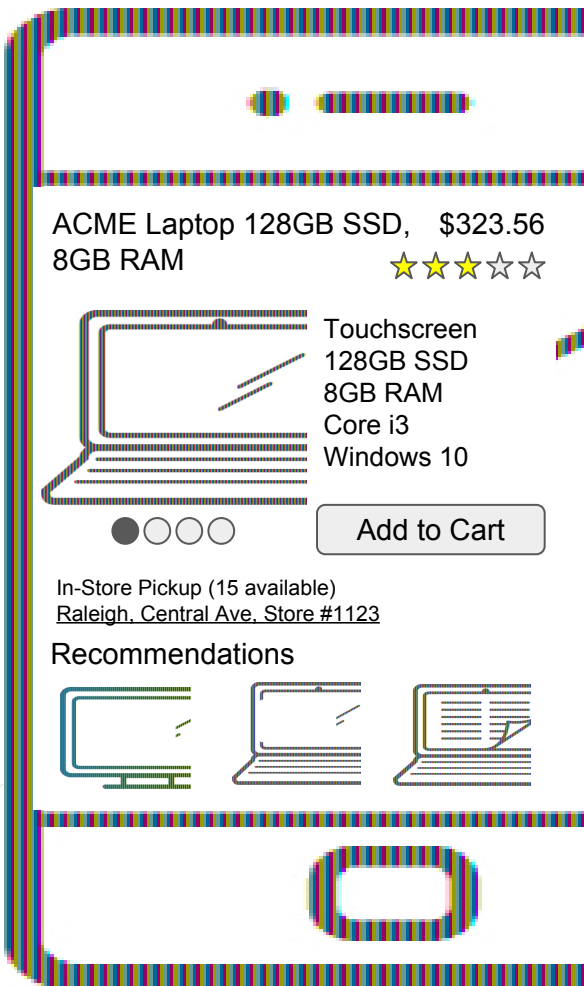
A



B

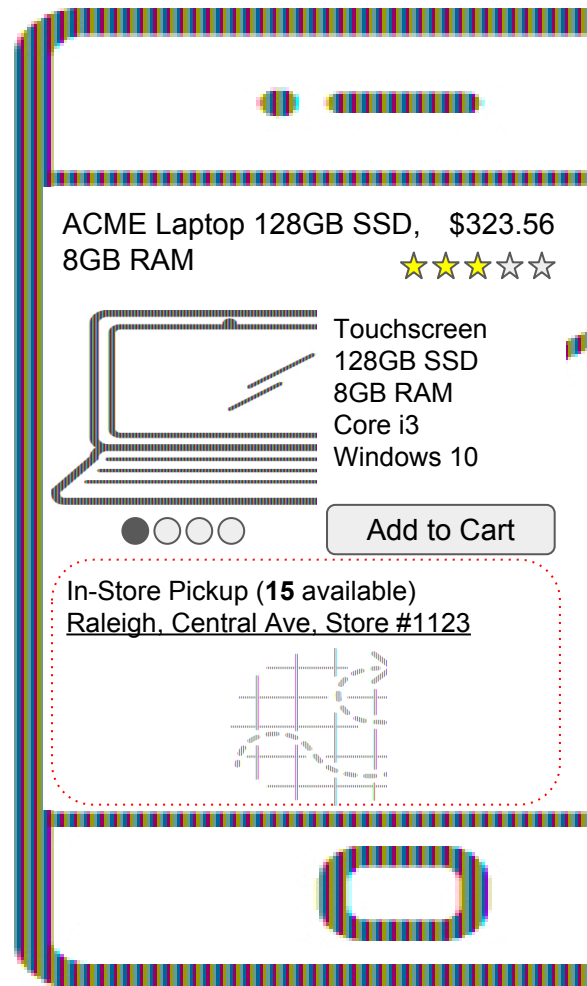


A



B

iOS Users
In North Carolina



bit.ly/cicdpipeline

Sunday - January 29 DevConf.cz 2017

The Deploy Master:
From Basic to Zero
Downtime,
Blue/Green, A/B, and
Canary

 Rafael Benevides

 DevTools

 1:00:00

Full Stack (Embrace the Front-End)

- 1) **Learn your language**: JavaScript dominates the client-side
ES6 & ES7 (aka ES2015 & ES2016)

<http://www.ecma-international.org/ecma-262/6.0/>

<https://www.ecma-international.org/ecma-262/7.0/>

<https://github.com/lukehoban/es6features>

- 2) **Service Workers**

<https://developers.google.com/web/fundamentals/getting-started/primers/service-workers>

<https://serviceworker.rs/>

- 3) Web Components (Polymer)

<http://webcomponents.org/>

<https://github.com/w3c/webcomponents>

<https://component.kitchen/tutorial>

- 4) Mobile Web Device APIs (light, battery, geolocation, vibration, etc)

- 5) Isomorphic (aka universal) apps

<http://isomorphic.net/>

- 6) PWA - Progressive Web Apps

<https://developers.google.com/web/progressive-web-apps/>

- 7) CI/CD - your client-side JS is also auto-buildable, auto-testable

npm and webpack for building

- 8) BFF: Backends for frontends

<http://samnewman.io/patterns/architectural/bff/>

- 9) Transpilers like Babel - brings new shiny ES6 or ES7 features to a browser near you

New JVM Runtimes

Opinionated & java -jar myapp.jar

DropWizard

www.dropwizard.io

JAX-RS API

First to market

DropWizard Metrics

Embeddable servers:
Jetty



Spring Boot

projects.spring.io/spring-boot

Spring API (@RestController)

'Starter' POMs:
start.spring.io

Embeddable servers:
Tomcat, Jetty, Undertow



WildFly Swarm

wildfly-swarm.io

Java EE 7 APIs

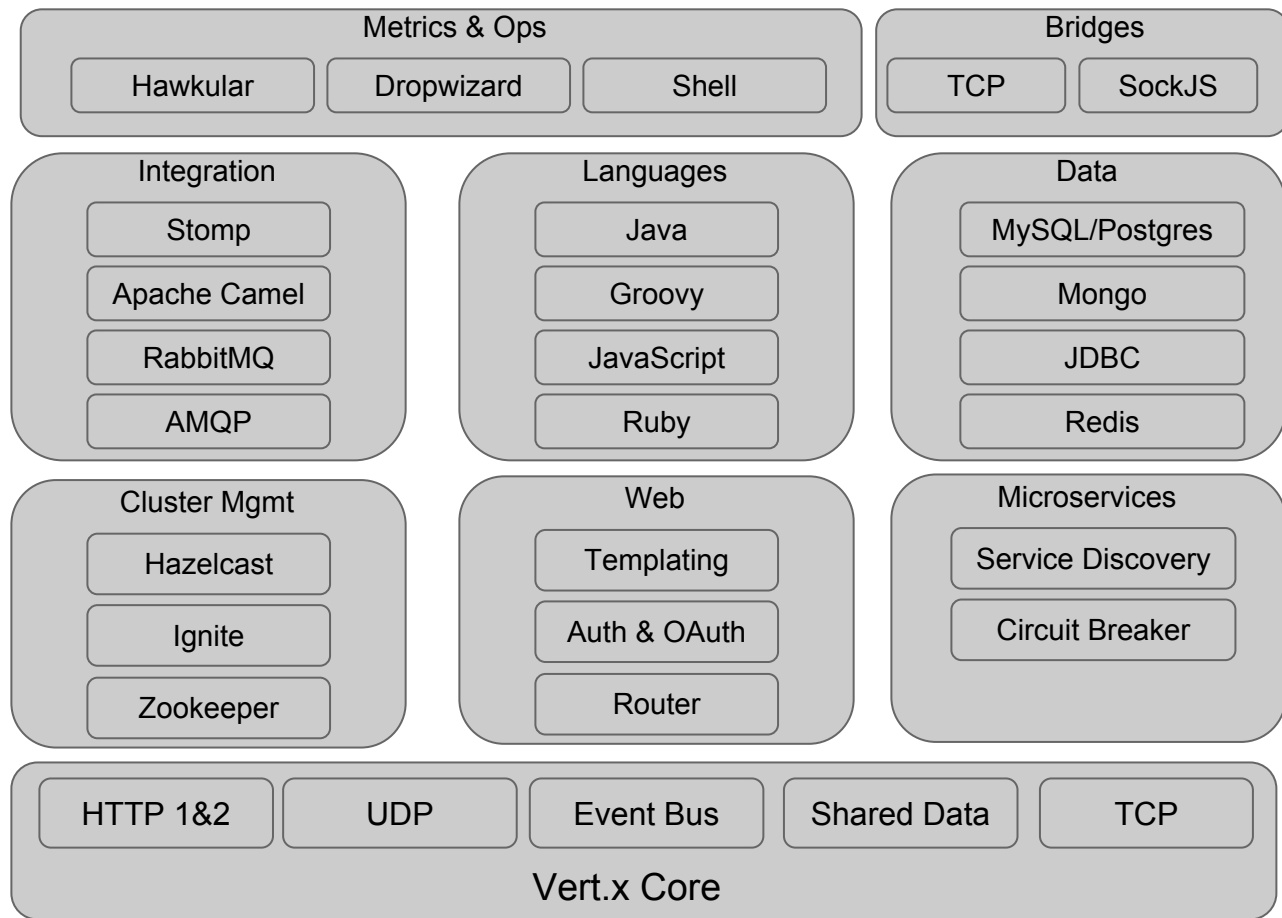
'Starter' POMs:
wildfly-swarm.io/generator

Embeddable servers:
WildFly (Undertow)





Vertx.io—an Eclipse project



<https://www.youtube.com/watch?v=7lbdWcdIYOI>

DevConf.cz 2017

January 27

Distributed and
Reactive applications
with vert.x

 Clement Escoffier

 JUDCon

 1:00:00

January 28

Modern app
programming with
RxJava and Vert.x

 Thomas
Segismont

 JUDCon

 1:00:00

The reactive
landscape

 Clement Escoffier

 JUDCon

 1:00:00

Reactive Systems
with Eclipse Vert.x
and Red Hat
OpenShift

 Clement Escoffier

 JUDCon

 1:00:00

AI

really deep learning, machine learning, advanced analytics

It is 2017, where are the Terminators?

Friday - January 27 DevConf.cz 2017

Building distributed
machine learning
pipeline

 Vojtěch Juránek

 JUDCon

 1:00:00



RED HAT DEVELOPERS

AppDev State of the Industry

bit.ly/appdevstate

[@burrsutter](https://twitter.com/burrsutter)

<http://developers.redhat.com>

[@burrsutter](https://twitter.com/burrsutter)