



DevOps for Defense

The 3 Ways: Summary

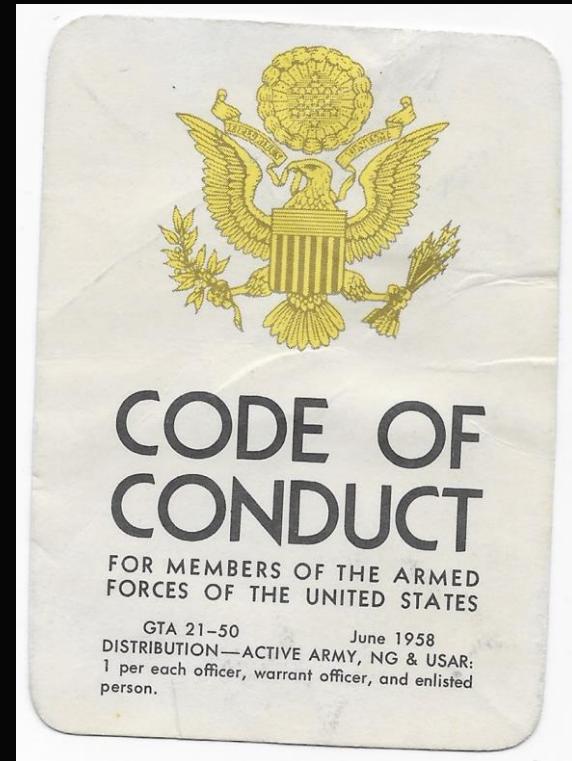
April 2018

JD Black
Justan Provence

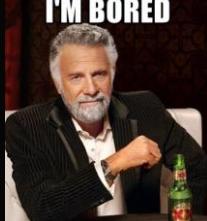
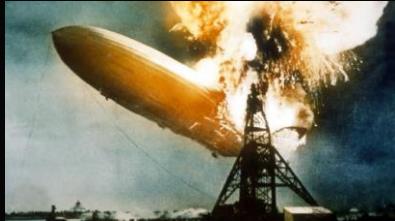
<https://www.meetup.com/DevOps-for-Defense/>
<https://github.com/jondavid-black/DevOpsForDefense>
devopsfordefense@gmail.com

DevOps for Defense Meetup: Code of Conduct

- UNCLASSIFIED ONLY!!!!
- Treat each other with respect and professionalism.
- Do not talk about private, sensitive, or proprietary work.
- Do talk about your experiences, needs, desires to improve work in our domain.
- Do share your thoughts.
- Do learn from others.
- Do respect & tip your bartenders!



Feedback from Last Month



1

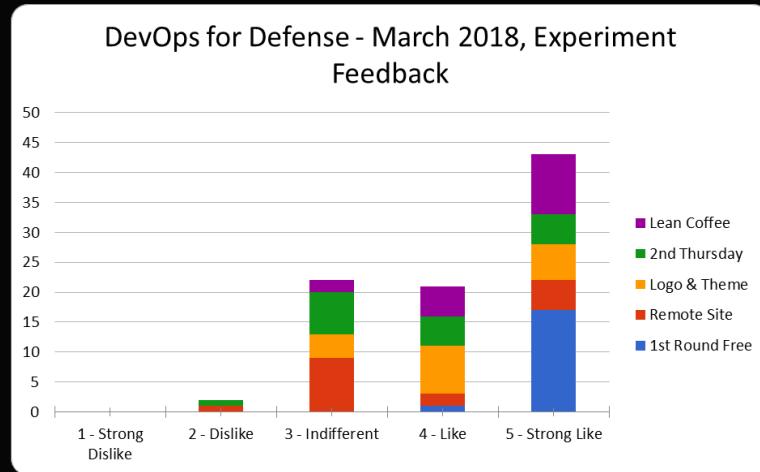
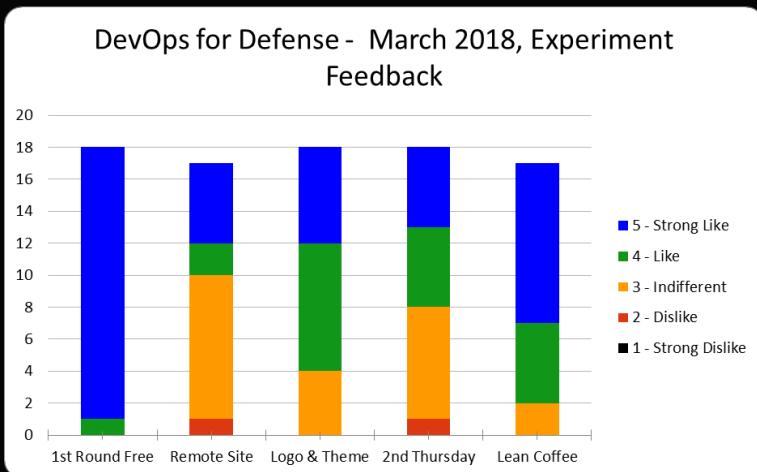
2

3

4

5

Never Again!

OK...or
Not...WhateverOMG,
Amazing!

Agenda: Start with Something New

- 
- DevOps Recap/Summary
 - Agile & DevOps
 - How to Screw Up DevOps

What's a Lightning Talk?

- 20 charts
- 15 sec auto advance
- 5 min total
- Danger = Extreme!

LIGHTNING TALKS

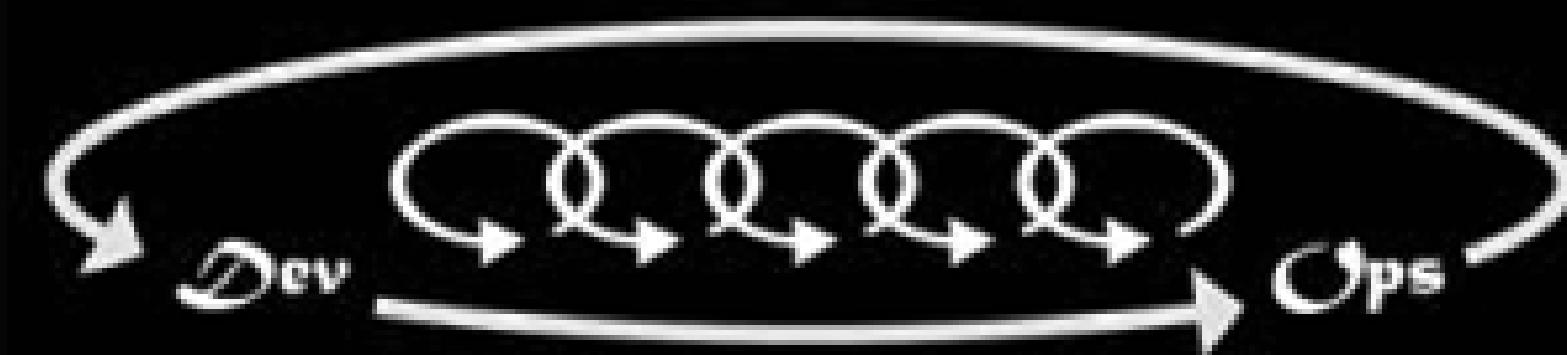
Agenda: Finish with a Lean Coffee

1. DevOps Obstacles
 - Facilitator: JD Black
2. DevOps Transition
 - Facilitator: Justan Provence
3. DevOps Leadership
 - Facilitator: Alana Gipson

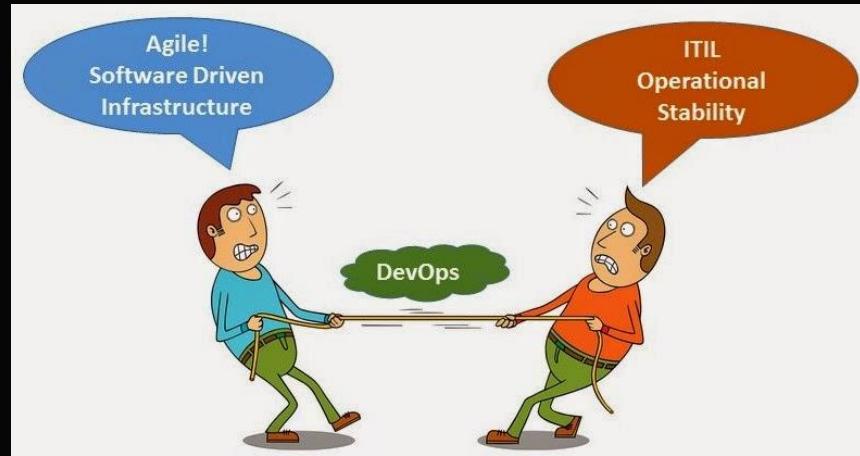


DevOps: The 3 Ways

Summary / Recap



Dev vs Ops



Why DevOps?

High Performers Are More Agile

46x

more frequent
deployments

440x

faster lead times
than their peers

High Performers Are More Reliable

5x

lower change
failure rate

96x

faster mean time
to recover (MTTR)

Why DevOps?

High Performers Are More Secure And Controlled

2x
less time spent
remediating
security issues

29%
more time spent
on new work

High Performers Win In The Marketplace

2x
more likely to
exceed profitability,
market share &
productivity goals

2x
more likely to achieve
organizational and
mission goals, customer
satisfaction, quantity &
quality goals

High Performers Win In The Marketplace

2.2x
higher employee
Net Promoter Score

50%
higher market
capitalization growth
over 3 years*

Key Capabilities that Drive High Performance

Technology and automation

- Version control
- Deployment automation
- Continuous integration
- Trunk-based development
- Test automation
- Test data management
- Shift left on security
- Continuous delivery
- Loosely-coupled architecture
- Architect for empowered teams

@nicolefv

Process

- Gather and implement customer feedback
- Work in small batches
- Lightweight change approval process
- Team experimentation

@nicolefv

Key Capabilities that Drive High Performance

Measurement and Monitoring

- Visual management
- Monitoring for business decisions
- Check system health proactively
- WIP limits
- Visualizations

@nicolefv

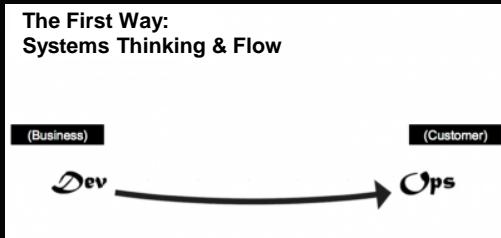
Culture

- Westrum organizational culture
- Climate for learning
- Collaboration among teams
- Make work meaningful
- Transformational leadership

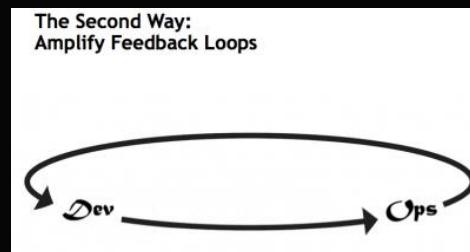
@nicolefv

Core of DevOps

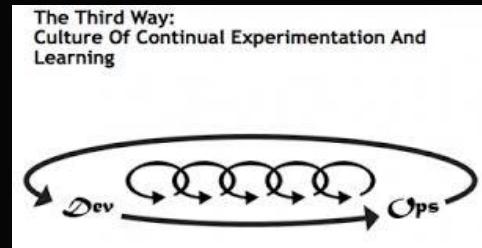
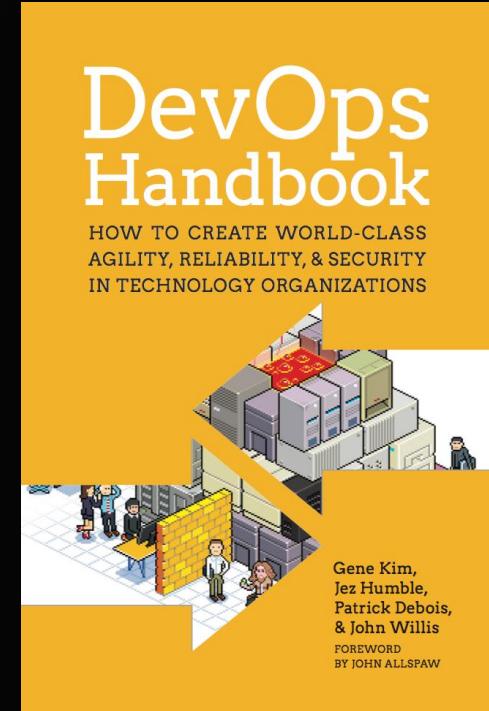
The 1st Way: Systems Thinking & Flow



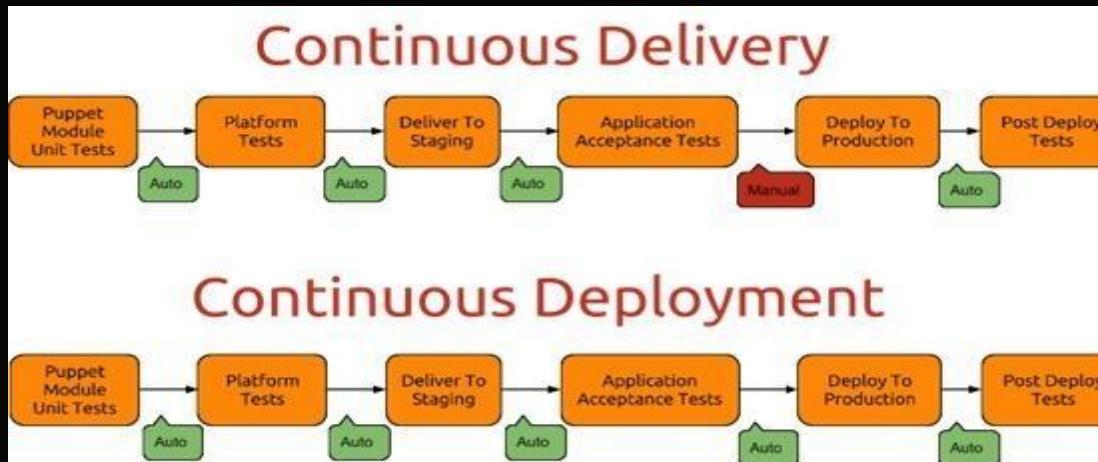
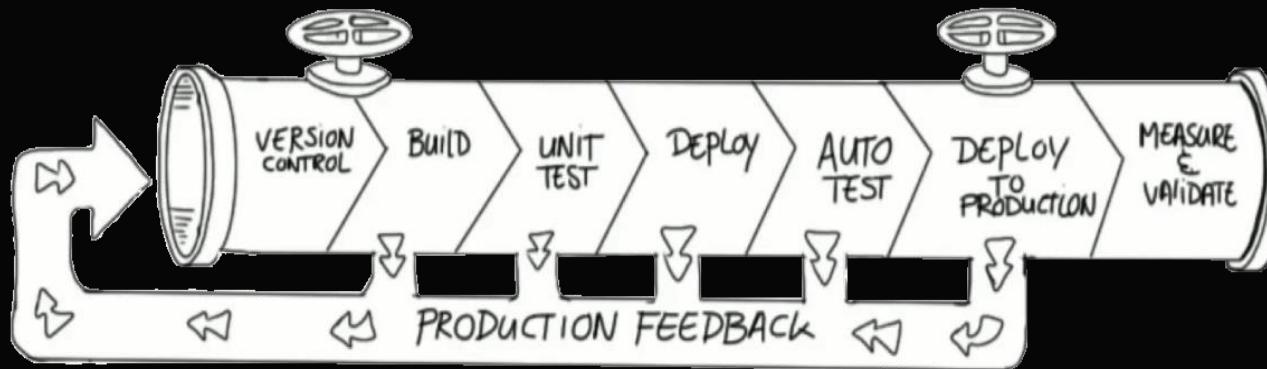
The 2nd Way: Feedback



The 3rd Way: Continual Experimentation & Learning

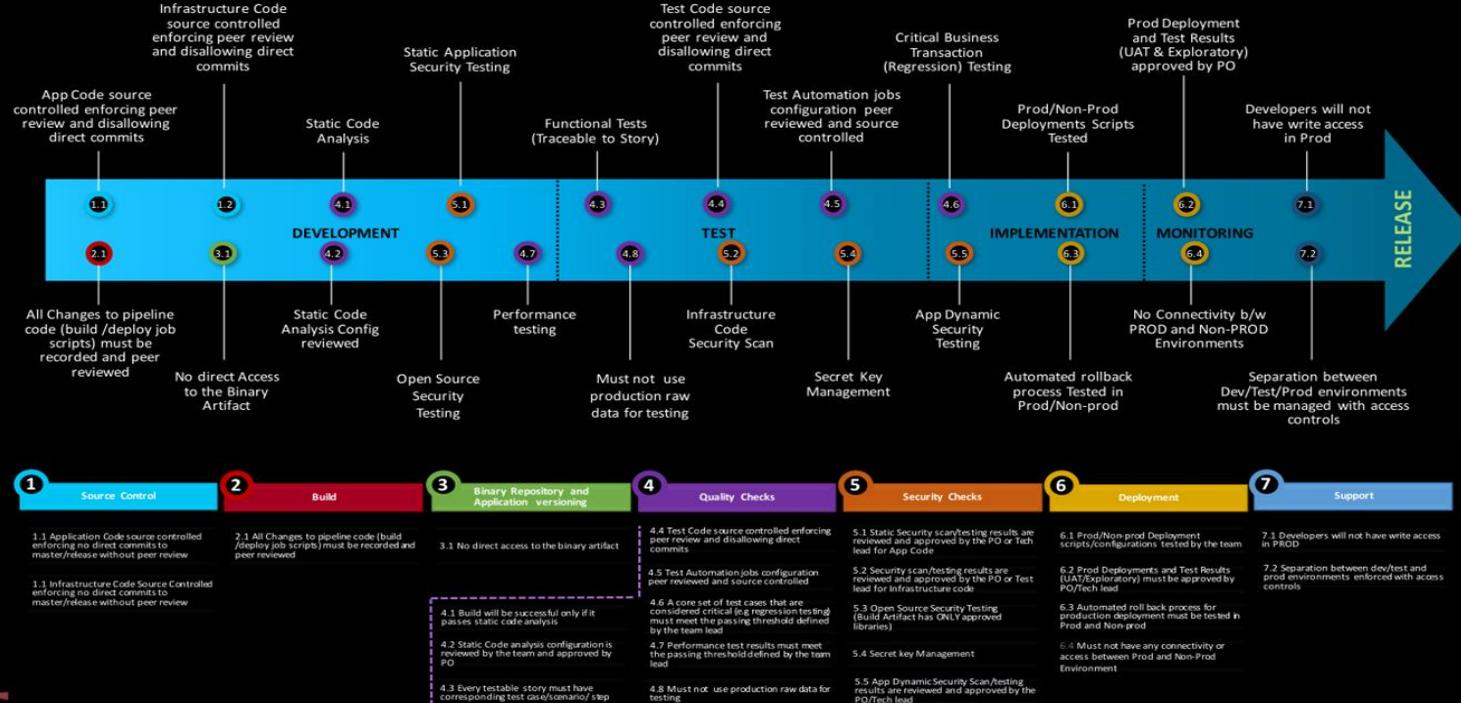


1st Way: What is a Continuous Delivery Pipeline?



1st Way: Continuous Delivery Pipelines (Capital One)

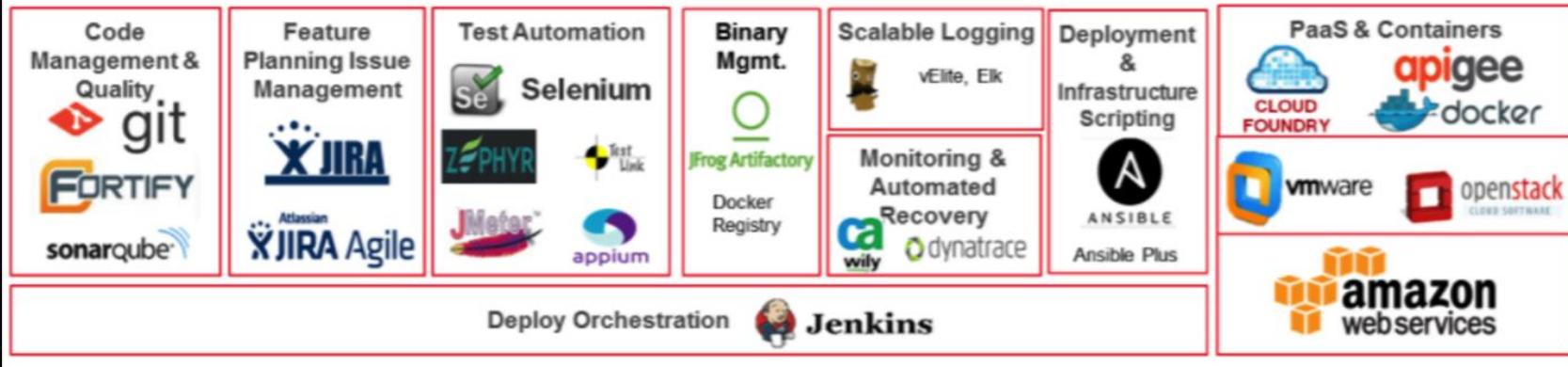
Software Delivery Clean Room



1st Way: Continuous Delivery Pipelines (Verizon)

A familiar toolchain,
operated at scale...

SDLC Automation Results	2017
Automated test runs	2M+
Automated code builds	1.2M+
Code check-ins	56K / month
Agile practicing apps in platform	3,297
Agile Jira tasks	1.9M
Agile sprints	20K+



1st Way: “Here at _____, we build _____ into our products!”

- Use your pipeline to build in quality, security, and the “ilities”.
- Quality:
 - Automated Unit Test - xUnit
 - Automated Acceptance Tests - Cucumber
 - Static / Dynamic Analysis - (sooo many)
 - Metrics (Sonar Qube, Jenkins, etc.)
- Security:
 - Klocwork, Fortify, Coverity, Arachni, etc.
- Deployability:
 - Puppet, Chef, Ansible
- Reliability:
 - Simian Army (Chaos Monkey, etc.)
- _____ility
 - Find/Build the right tool, Read Jez’s Book



2nd Way: Feedback - Let me Google that for you!

feed·back
'fēd,bak

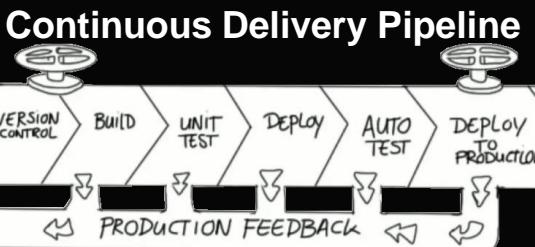
Noun



1. Information about reactions to a product, a person's performance of a task, etc., used as a basis for improvement.
2. The modification or control of a process or system by its results or effects, e.g., in a biochemical pathway or behavioral response.

synonyms: response, reaction, comments, criticism

2nd Way: Sources of Feedback



2nd Way: Make Your Feedback Meaningful...Pay Attention!

“Create a system of work where knowledge acquired downstream in Operations is integrated into the upstream work of Development and Product Management.”

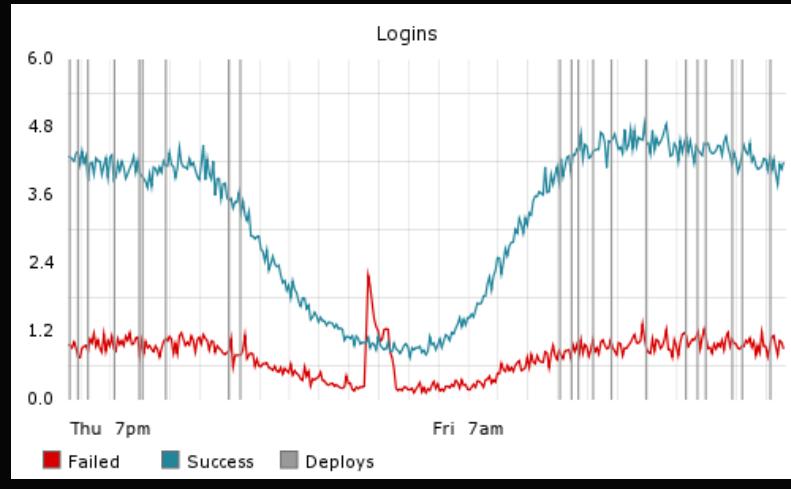
- The DevOps Handbook, 2016

“If Engineering at Etsy has a religion, it’s the Church of Graphs. If it moves, we track it. Sometimes we’ll draw a graph of something that isn’t moving yet, just in case it decides to make a run for it.”

- Ian Malpass, Etcy 2011

“By amplifying signals as part of our daily work, we make it possible to see and solve problems as they occur, and we grow safe systems of work that allow us to confidently make changes and run product experiments, knowing we can quickly detect and remediate failures.”

- The DevOps Handbook, 2016



2nd Way: Zen...and the Art of Feedback

**"If you can't *measure* it,
you can't *improve* it."**

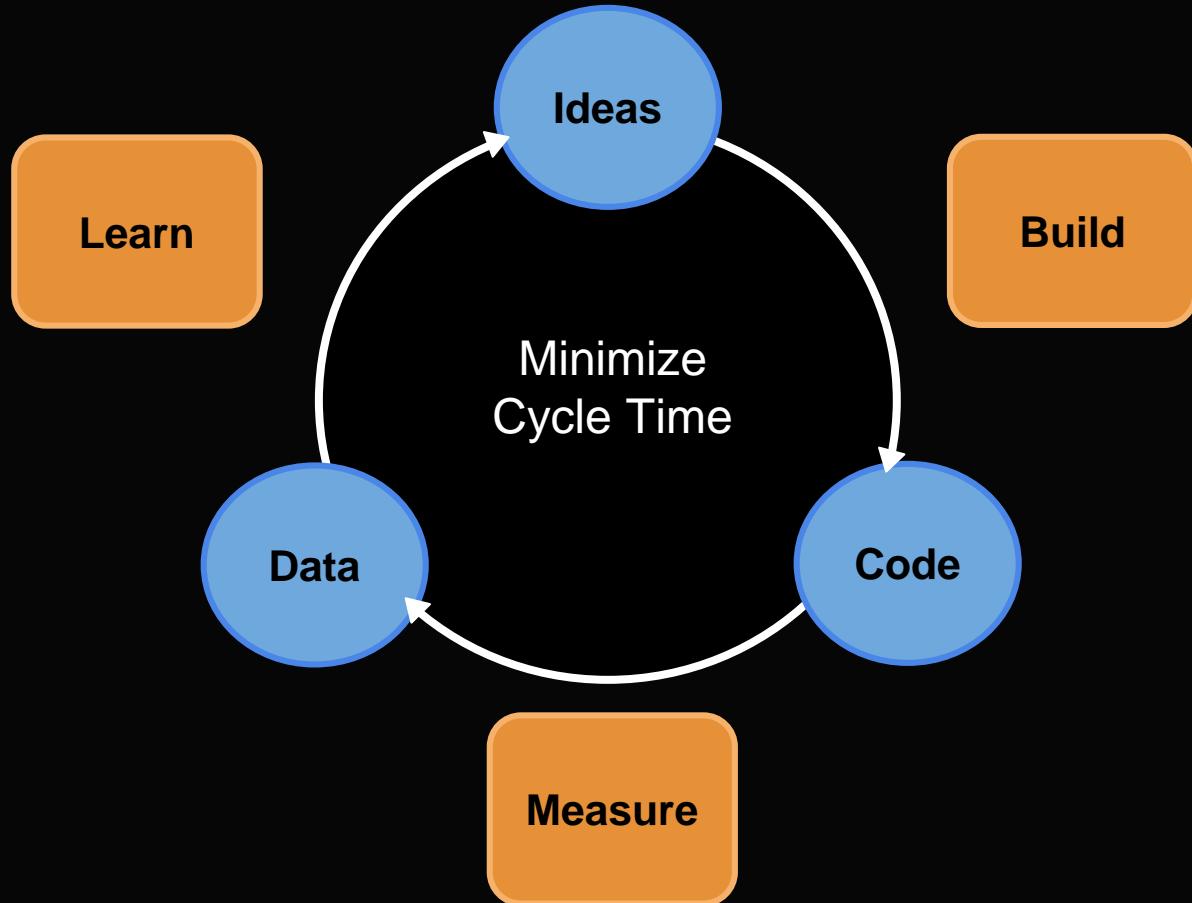
Peter Drucker

**"Invest *energy* in collecting
metrics that help you make
decisions."**

Eric Ries



3rd Way: Lean Start-Up (by Eric Ries)



Lean Start-up In Practice:

1. Define Hypothesis
2. Build the absolute smallest thing you can to test that hypothesis. This is your **Minimum Viable Product (MVP)**.
3. Collect and analyse the data to prove / refute your hypothesis.
4. If things go well, continue. If not, pivot.

In the defense domain we call this performing a **Decision Analysis Report** (DAR from CMMI) or a **Trade Study**.

Our problem is we define huge studies that take many months to complete.

3rd Way: Failure Is An Option - Pivot When Needed

“Evaluating well-designed and executed experiments that were designed to improve a key metric, only about $\frac{1}{3}$ were successful at improving the key metric!”

- Online Experimentation at Microsoft

Types of Pivots:

Zoom In

Customer Segment

Platform

Value Capture

Chanel

Zoom Out

Customer Need

Business Architecture

Engine for Growth

Technology

3rd Way: Define Your Experiment

Hypothesis Template

We believe that

[building this capability]

[for these people]

Will achieve [this outcome].

We know we are successful when we see

[this measure / metric / observation].

- adapted from Jeff Gothelf

The DevOps for Defense Meetup Hypothesis

creating a DevOps meetup

for defense industry professionals

a broad cultural transformation that improves DoD system development and mission value delivery by adapting commercial best practices to our domain

modern DevOps concepts and practices embraced by DoD and Industry to safely deliver greater value to our Warfighters

3rd Way: Limit the Blast Radius

Define in Your Backlog

Blameless Postmortems

Feature Toggles

Automate Deploy & Roll-back

Versioned Interfaces

Incremental Evolution

Canary Releases

A/B Deployments



3rd Way: Experiments Require a Safe Environment & Culture



“Don’t try this at home. We’re what you call experts.”

What is the Goal of DevOps?

**When we can safely, quickly,
reliably, securely achieve
all the needs, goals, dreams and
aspirations of everyone we serve...**

**...especially our citizens
and warfighters**

A photograph of the Eiffel Tower in Paris, France, at night. The tower is illuminated with blue and white lights, and a bright yellow lightning bolt is captured in mid-air, striking the top of the tower. The sky is dark and cloudy. In the foreground, the city skyline of Paris is visible with various buildings and lights.

Next Lightning Talk
...in 3, 2, 1

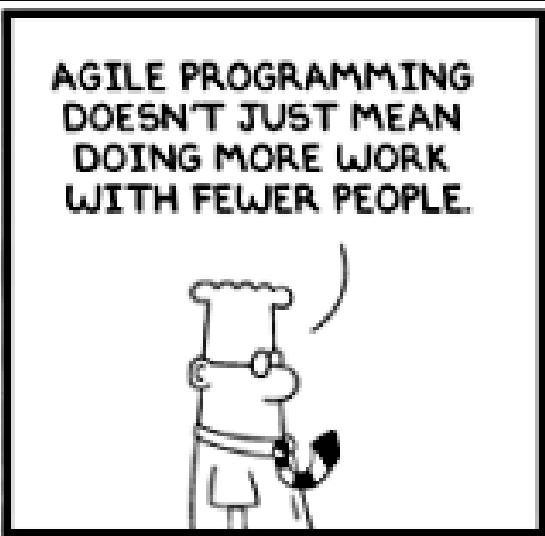
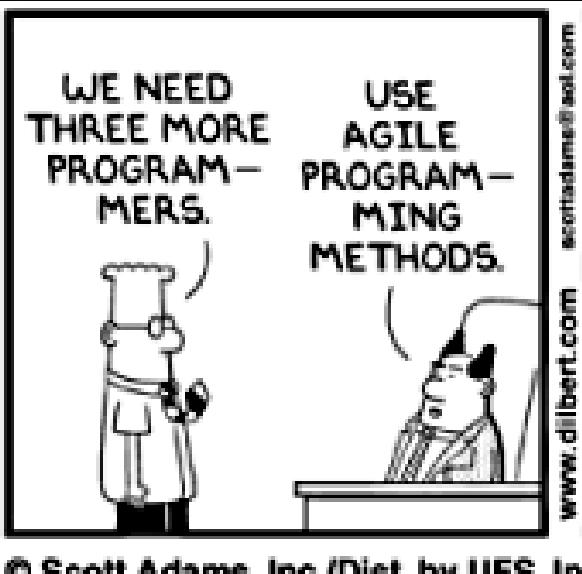


DevOps For Defense

Agile and DevOps Refresher Lightning Talk

April 2018 - Justan Provence

There are a lot of misconceptions about Agile and DevOps



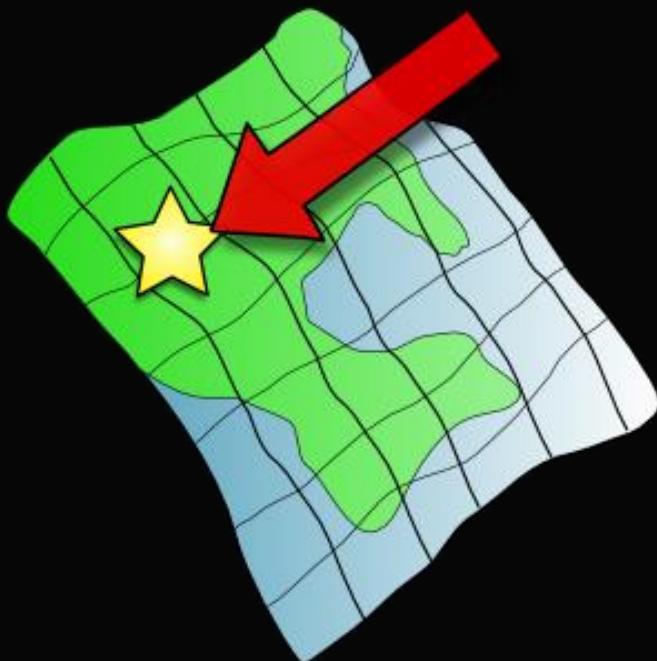
© Scott Adams, Inc./Dist by UFS, Inc.

History between the two?



Yes, starting back at the **Agile 2008 Conference** with Patrick Dubois presenting Agile Infrastructure and Operations

The DevOps philosophy is ingrained within the Agile Manifesto, and it could be said that DevOps is just an extension of Agile principles into operations



We can easily map DevOps principles to the Agile Manifesto

Agile Manifesto Values

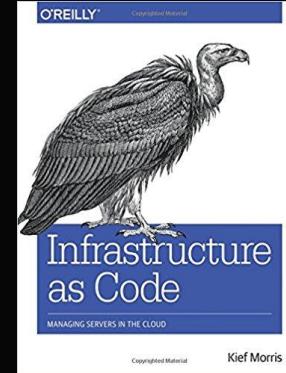
Individuals and interactions over processes and tools



DevOps is all about collaboration among
everyone in the software supply chain

Agile Manifesto Values

Working software over comprehensive documentation



DevOps takes this concept to the next level by automating environment provisioning, software installation, and automatic delivery of applications instead of document-centric step-by-step instructions

Agile Manifesto Values

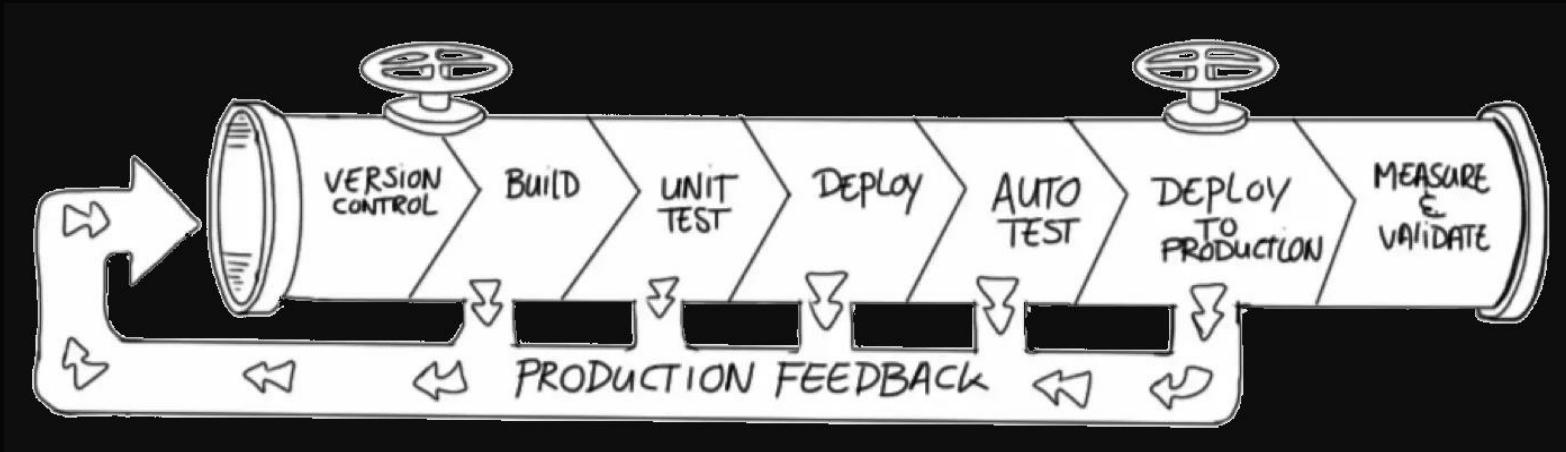
Customer collaboration over contract negotiation



When fixes to production issues can be automatically tested and redeployed,
it's certainly focusing on customer collaboration and value

Agile Manifesto Values

Responding to change over following a plan



Small batch sizes and continuous delivery support the ability to change as needed

How are they different?



It is typically how we tend to implement our Agile process

We declare victory too early



DevOps strives to focus on the overall service or software fully delivered to the customer instead of simply “working software”

Critical team members are not interconnected



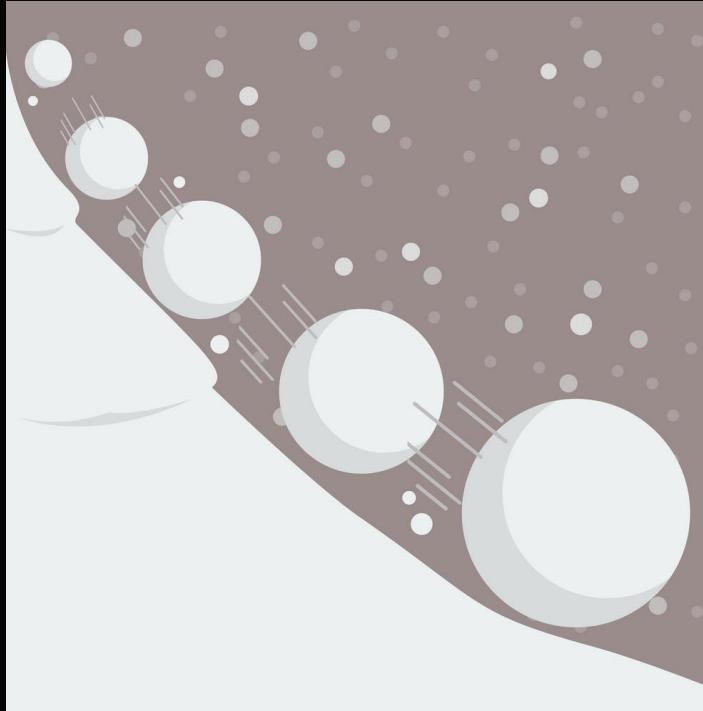
Unfortunately, not delivering releasable code frequently means that system administrators and operational personnel are not part of your cross-functional agile team. This causes communication challenges that agile was built to address!

DevOps can be implemented without Agile



It can be used to make sure that annual updates go smoothly

Agile or Bust!



You can implement DevOps a little at a time and gain real value.
This might not be worthwhile with Agile

What are the key similarities?



Agile and DevOps work better in combination
You should use both to unlock your potential

Agile nor DevOps are business goals in and of themselves



You probably won't get extra money
from your customer for doing it!

Agile and DevOps are cultural movements that can inspire your organization with better means for achieving your goals



Agile and DevOps both encourage

Small Batch Planning & Execution

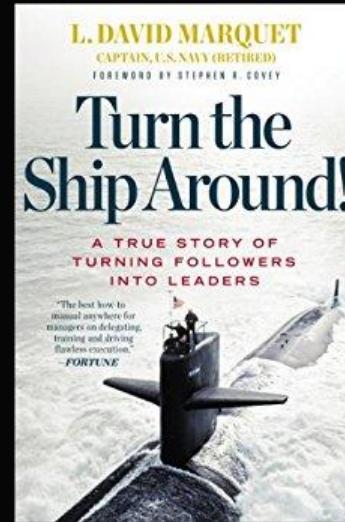
Regular cadence that delivers deployable products

Experimentation



Agile and DevOps both benefit from

A Learning Organization
Good Transformational Leadership



It's about communication!

Don't stop at Development and Operations!
Involve everyone in the development of the product

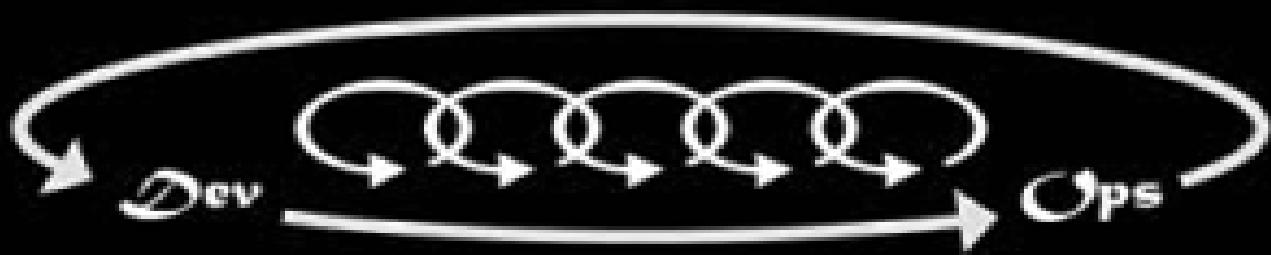


Attack ambiguity and uncertainty

Write good stories!



How to Screw Up DevOps



Leadership – Critical to DevOps Success

Decentralize decision making Unlock intrinsic motivation

Intellectual Stimulation (IS)

- Challenge the status quo
- Encourage followers to learn, be creative, explore new ways of doing things
- Decentralize decision making
- Expect relentless improvement
- Encourage innovative thinking
- **Adaptive leadership**

Inspire and align with mission

Inspirational Motivation (IM)

- Articulate a clear vision and intent
- Inspire passion and motivation to achieve goals
- Drive organizational alignment
- Encourage others
- **Charismatic leadership**

Transformational Leadership



Develop leaders

Individualized Consideration (IC)

- Offer personalized support, coaching, and encouragement
- Keep lines of communication open
- Offer direct recognition individual and team contributions
- Exhibit genuine care and concern
- **Empathetic, Servant leadership**

Lead the change Know the way

Idealized Influence (IIA/IIB)

- Be a role model; set the example
- Be a lifelong learner; gain the knowledge required for change
- Create an environment of trust and respect through transparency
- Act with integrity
- **Authentic, Charismatic leadership**

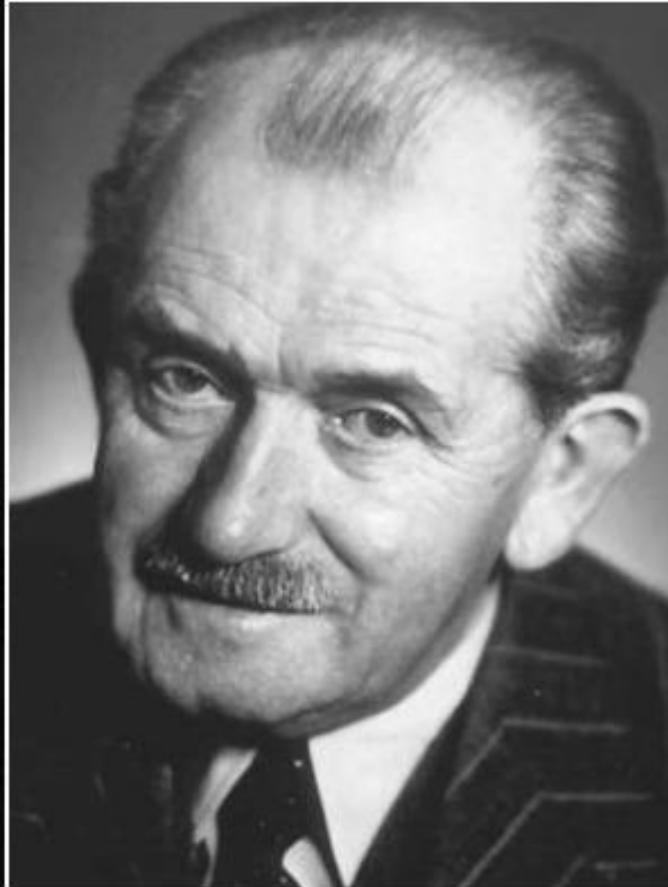
Leadership...Except – Knows We're Going Somewhere...Where?



Today I'm excited to lead
this team as we begin our
DevOps journey?

What on Earth
is this DevOps
thing?

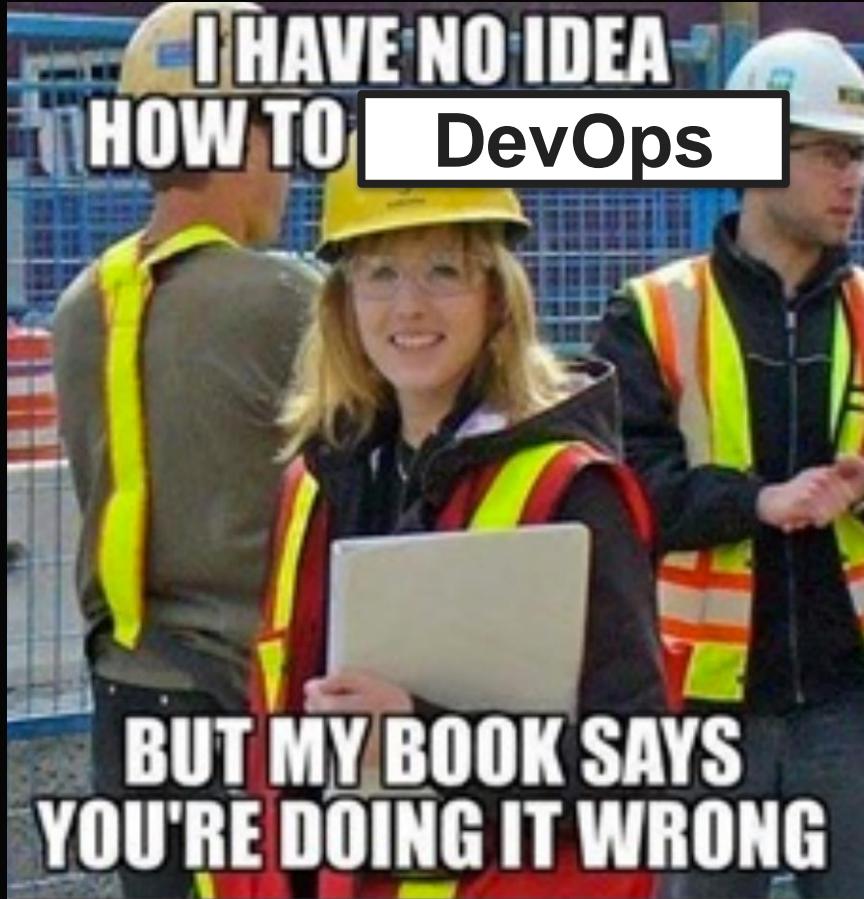
Leadership...Except – When Change == Success



Change is easy. Improvement is far more difficult

— *Ferdinand Porsche* —

Leadership...Except – When Just Checking the Boxes



Leadership...Except – When Authority is Centralized and Responsibility is Decentralized

I need to see
more
analysis to
assess risk.

Now is not a
good time.

Has this
been pre-
coordinated
with that
team that
has nothing
to do with
this?



**Change Control
Emperor**



New Feature?

Reduce Technical
Debt?



Process
Improvement?



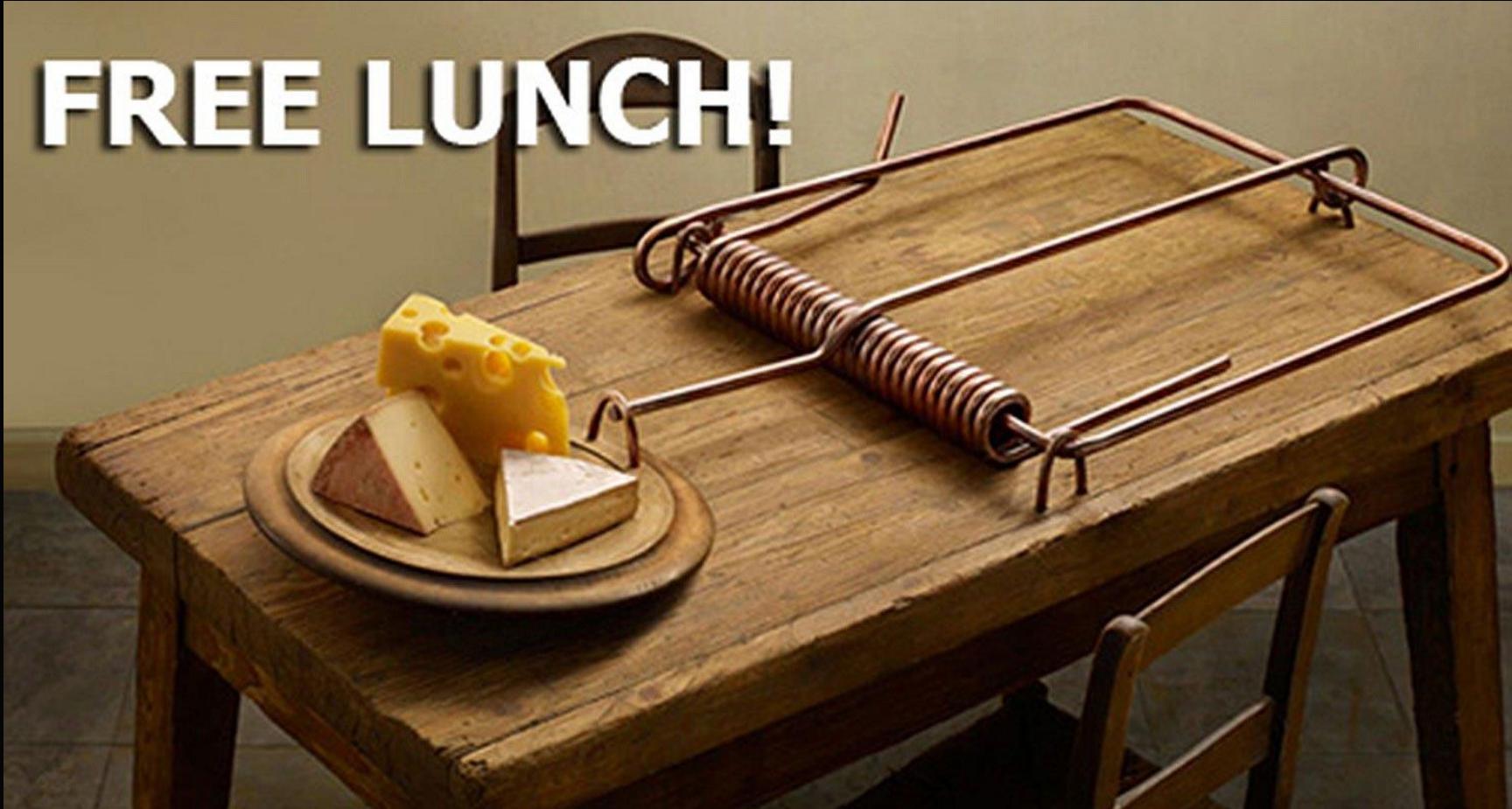
Expectations – Critical to DevOps Success



High achievement always takes place in the framework of high expectation.

— *Charles Kettering* —

Expectations...Except – Better, Faster, & Safer is Free Right?



Expectations...Except – Complete that Transition by Tomorrow!

Yesterday, I told you to drop 50 pounds...why aren't you there yet?



Can I be done with you now?

Expectations...Except – Jenkins is Installed, DevOps Achieved!



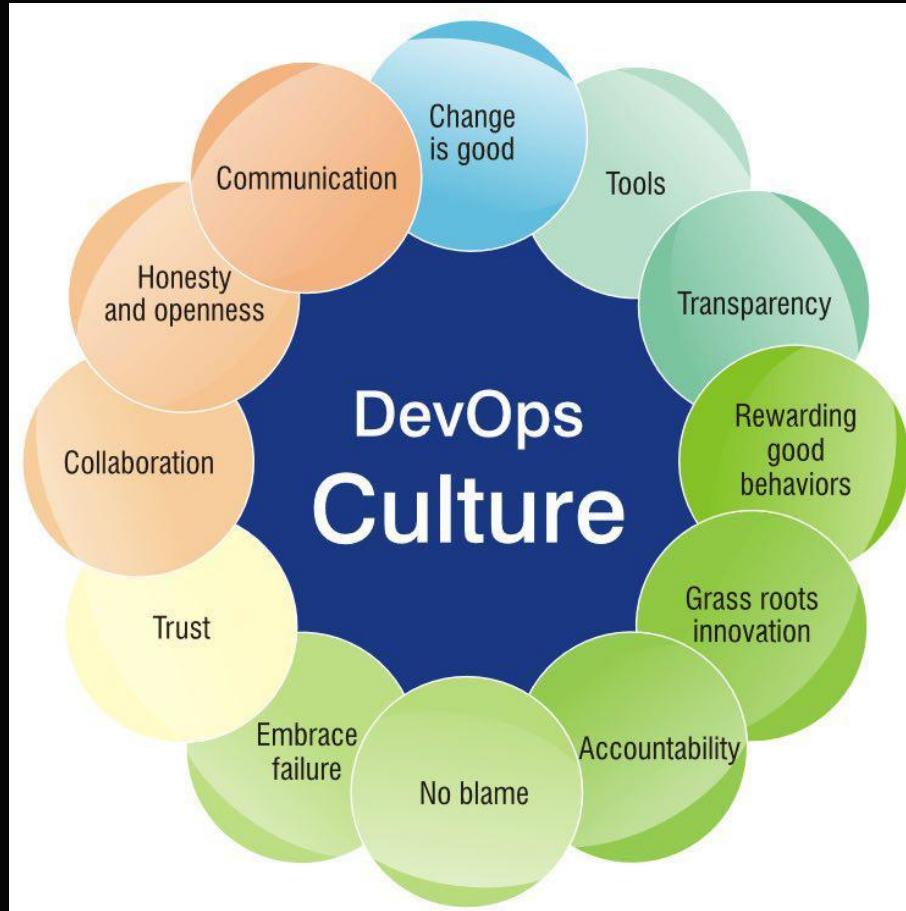
!=



Expectations...Except – It works for that team, it must work here.



Culture – Critical to DevOps Success



Culture...Except – “My Work is Too Important to Change.”

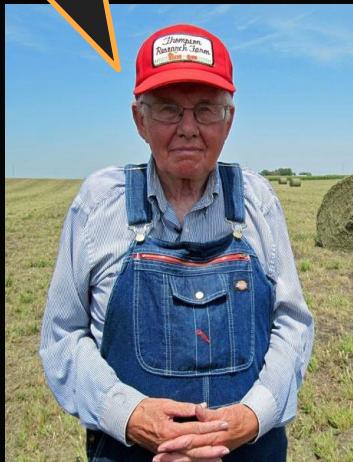


Culture...Except – DevOps is just a Layer on Top

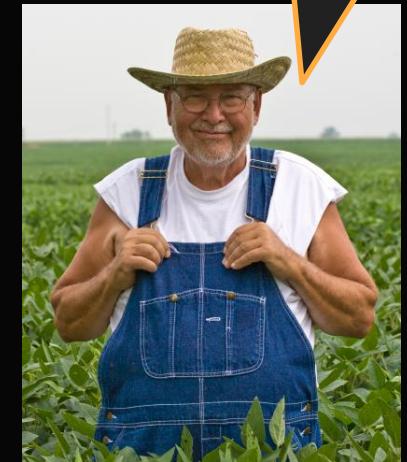


Culture...Except – When we Keep Our Silos.

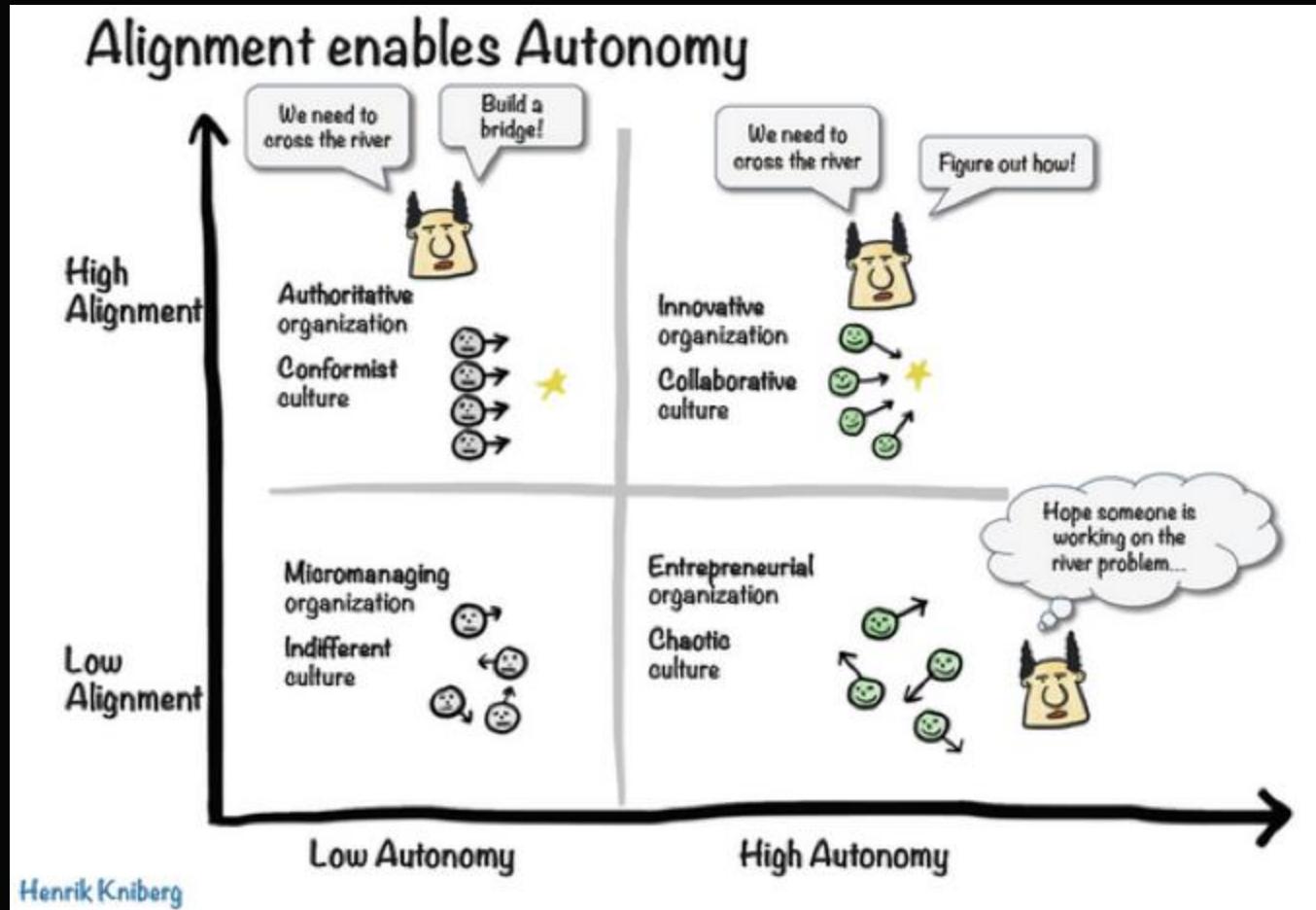
Yep...my silo is looking pretty good.



Yep...my silo is looks good too. This is going great!



Culture...Except – When Teams Aren't Aligned to be Autonomous



Architecture – Critical to DevOps Success



Architecture...Except – Can't Risk Changing the System



Architecture...Except – When We've Lost the 'Why'



The Architect put
that there for a
reason.

Really? What
reason?

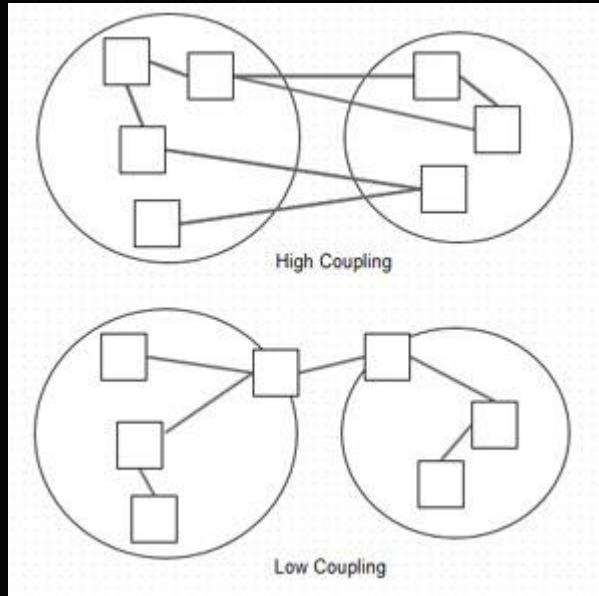
I have no
idea, so we
shouldn't
mess with it.



Architecture...Except – When It Only Applies to New



Architecture...Except – Doesn't Reduce Coupling or Support Separation of Concerns



Lightning Talks – Done!

A wide-angle photograph of a city skyline at night under a dark, turbulent sky. Numerous bright, branching lightning bolts strike across the sky, illuminating the clouds and casting a glow on the city lights below. The city's buildings are silhouetted against the bright lightning and the dark sky.

Group Exercise: Lean Coffee

1. Each table has a facilitator.
2. The facilitator has a short introduction to an aspect of continuous experimentation and improvement.
3. Everyone write down questions on the subject. Place them in the middle of the table.
4. The group votes on questions by placing a dot on the card. 3 votes per person.
5. Question with most dots goes first. Set a timer for 5 minutes and discuss.
6. After 5 minutes, either vote (thumbs up/down) to keep going or move on to the next question.



Agenda: Finish with a Lean Coffee

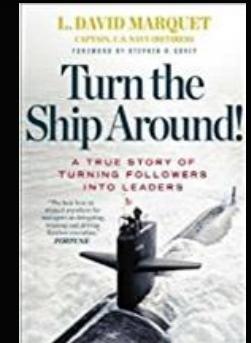
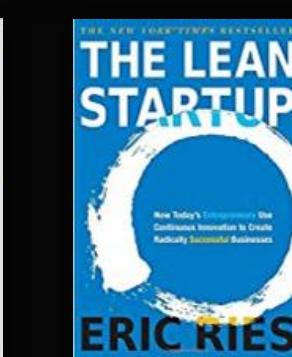
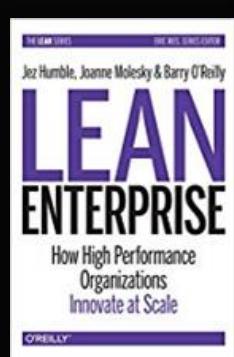
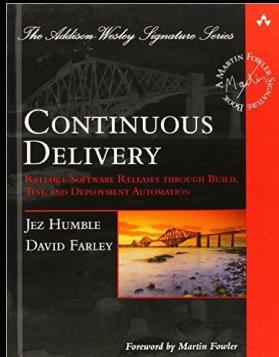
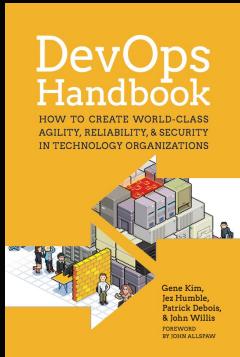
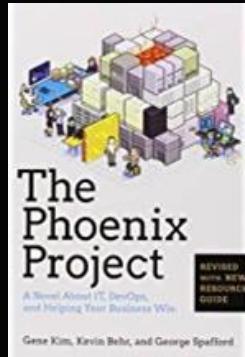
1. DevOps Transition & Obstacles
 - Facilitator: JD Black
2. DevOps Tools & Best Practices
 - Facilitator: Justan Provence
3. DevOps Leadership
 - Facilitator: Alana Gipson



DevOps Resources

Books / Publications:

- The Phoenix Project
- The DevOps Handbook
- Continuous Delivery
- Lean Enterprise
- Lean Startup
- The State of DevOps Report
- Turn This Ship Around!



<https://www.meetup.com/DevOps-for-Defense/>
<https://github.com/jondavid-black/DevOpsForDefense>
devopsfordefense@gmail.com

Conference Presentations (YouTube):

- DevOps Enterprise Summit (DOES)
- Velocity
- GoTo

