



Sponsored by:



# DevOps for Defense

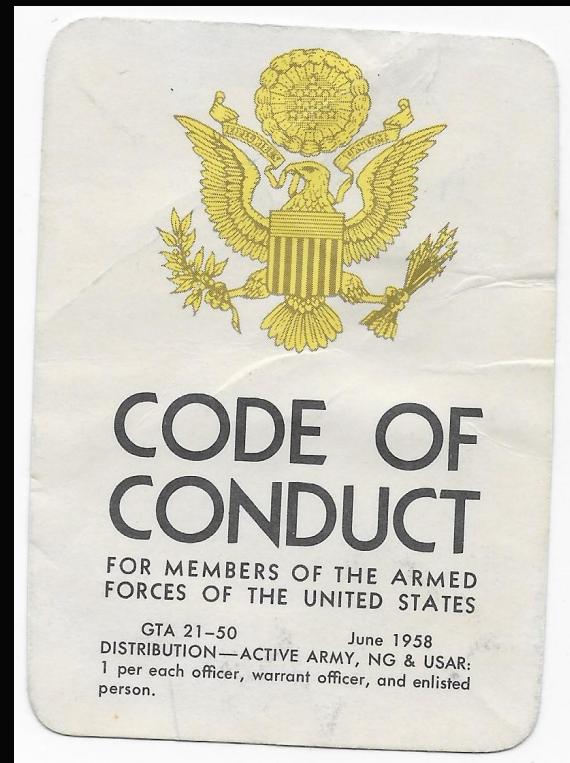
January 2020

DevOps Fundamentals  
JD Black

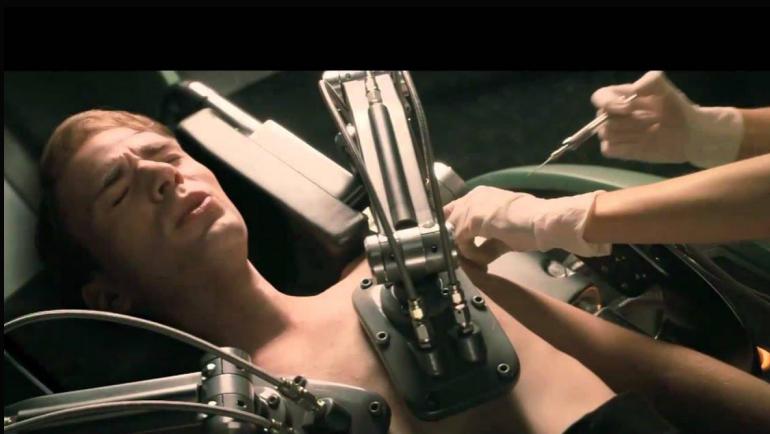
<https://devopsfordefense.org>  
<https://www.meetup.com/DevOps-for-Defense/>  
<https://github.com/jondavid-black/DevOpsForDefense>  
[devopsfordefense@gmail.com](mailto:devopsfordefense@gmail.com)  
<https://twitter.com/devops4defense>

# 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!

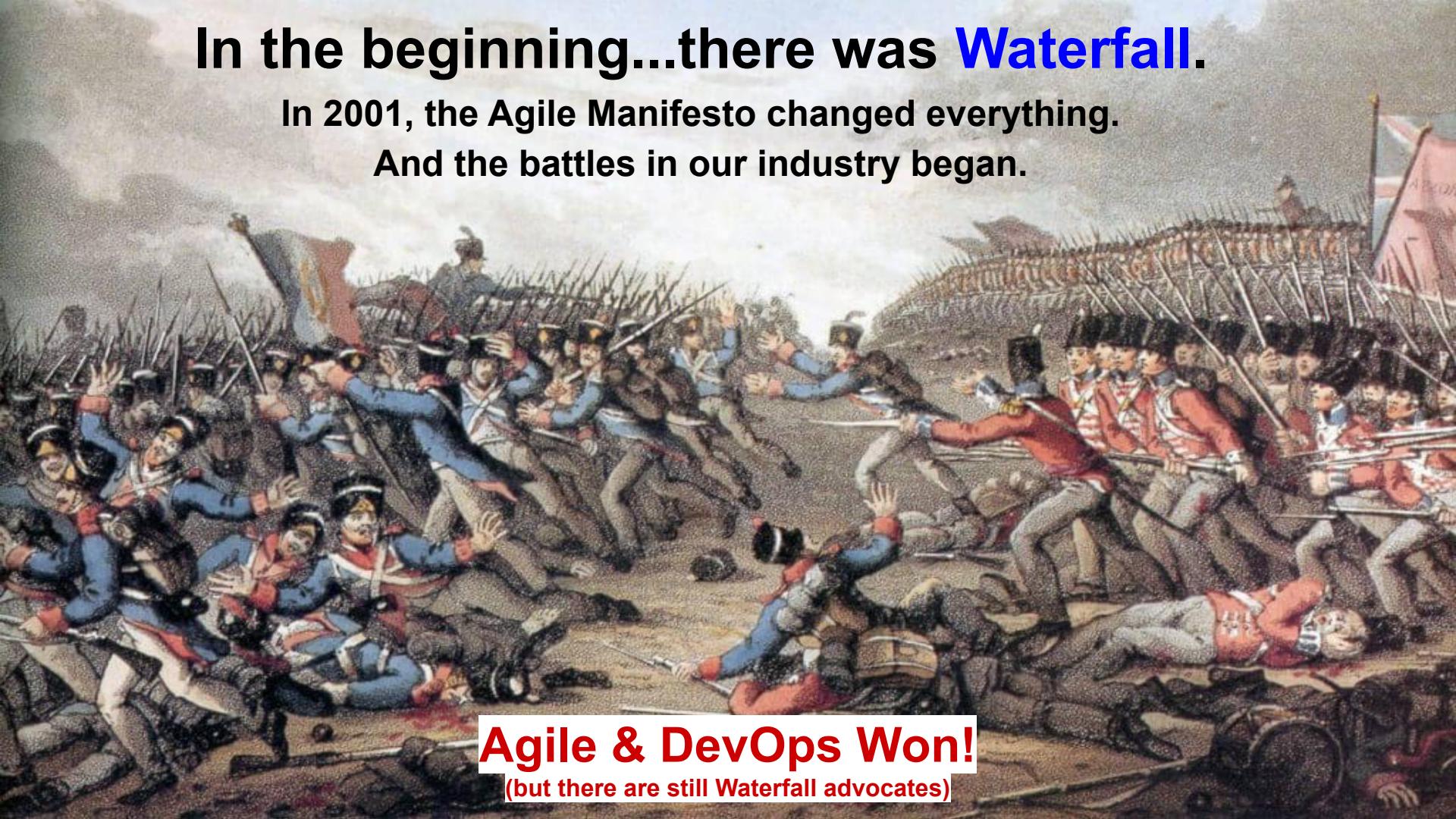


# Origin Story



# In the beginning...there was Waterfall.

In 2001, the Agile Manifesto changed everything.  
And the battles in our industry began.



**Agile & DevOps Won!**  
(but there are still Waterfall advocates)

# History of DevOps

There is definitely a historical connection between DevOps and Agile. Starting all the way back in 2008 with Patrick Dubois presenting Agile Infrastructure and Operations at the Agile 2008 conference.

See The History of DevOps by Damon Edwards for more information



10 deploys per day  
Dev & ops cooperation at Flickr

John Allspaw & Paul Hammond  
Velocity 2009

# Agile nor DevOps are Business Goals in and of Themselves

- Both are cultural movements that can inspire your organization with better means for achieving your goals
  - Encourage a Learning Organization
  - Experimentation, Fail Fast, Limit the Blast Radius
- Agile and DevOps work better in combination
  - Small Batch Planning & Execution
  - Regular cadence that delivers deployable products
- It is about communication and involving anyone in the development of the product
  - Attack ambiguity and uncertainty - write good stories!

# 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)

## 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

## 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



## Measurement and Monitoring

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

@nicolefv



## Culture

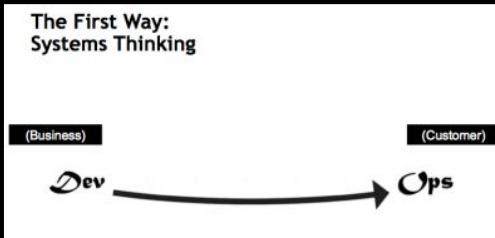
- Westrum organizational culture
- Climate for learning
- Collaboration among teams
- Make work meaningful
- Transformational leadership

@nicolefv

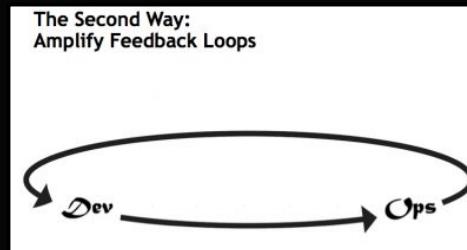


# Core of DevOps

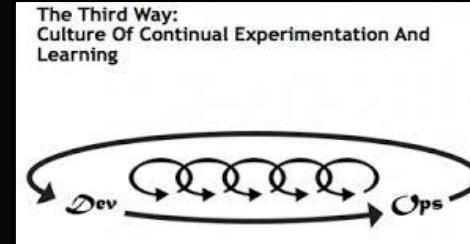
## The 1st Way: Systems Thinking & Flow



## The 2nd Way: Feedback



## The 3rd Way: Experimentation & Learning



Transform Your SW Lifecycle!

COP 3331

Exam 1

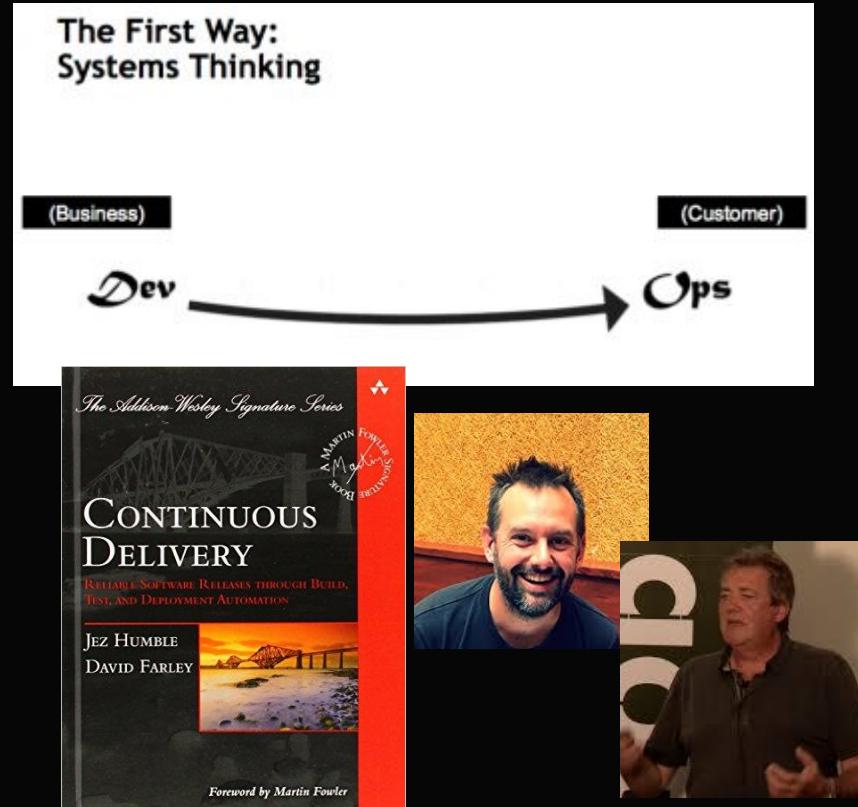
### 2 Short Answer Questions

11. [10 points] Name and describe the five key phases of software development.

- ~~1. denial  
2. bargaining  
3. Anger  
4. depression  
5. acceptance~~

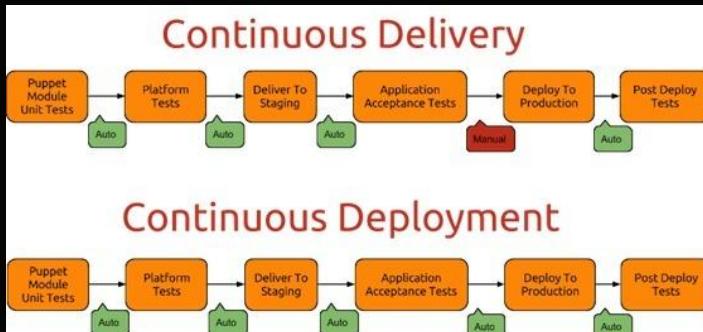
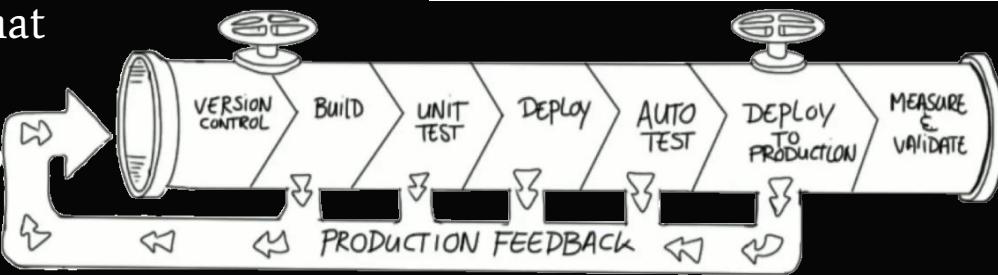
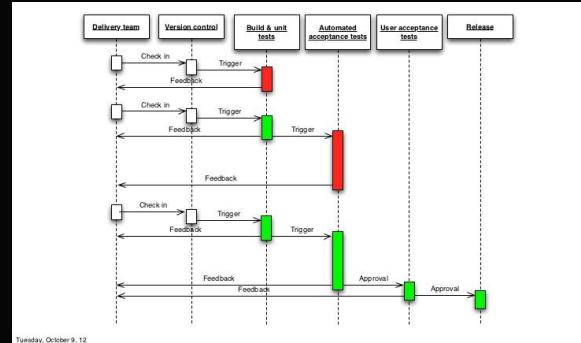
# DevOps: The First Way - Systems Thinking & Flow

- Systems Thinking: “Emphasize the performance of the entire system, as opposed to the performance of a specific silo of work or department” - Gene Kim (2012)
  - Interpretation: Optimize the delivery of value to your customer across the organization recognizing that your delivery process is a system itself
- Flow: “Accelerate the delivery of work from Development to Operations to our customers” - Gene Kim (2016)
  - Interpretation: Build a Continuous Delivery pipeline! Feed it constantly!



# What is a Continuous Delivery Pipeline?

1. Automated suite of tools, triggered by an update to your product's configuration managed baseline, that performs all the build, test, QA, packaging, (and deployment) of your system.
2. Rapid feedback to engineers of issues introduced by a change
3. Continuous Integration...on steroids!
4. Your product's immune system
5. Your source of confidence



**“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



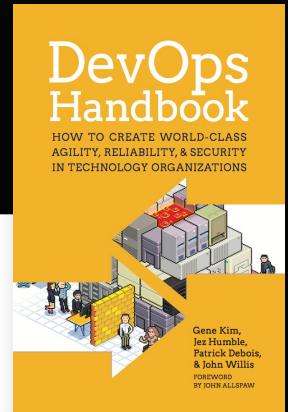
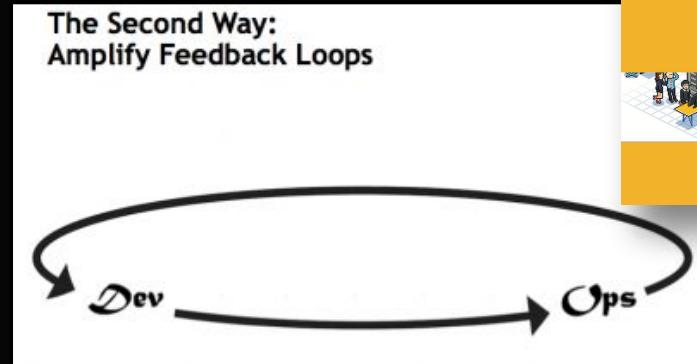
# DevOps: The Second Way - Feedback

“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

# Remember: State of DevOps Report 2017

Technology and automation

## Measurement and Monitoring

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

@nicolefv



Process

Culture

## High Performers Are More Agile

**46x**

more frequent deployments

Source: Puppet/DORA, 2017 State Of DevOps Report <https://puppet.com/resources/state-of-devops-report>

**440x**

faster lead times than their peers

## High Performers Are More Reliable

**5x**

lower change failure rate

**96x**

faster mean time to recover (MTTR)

## High Performers Win In The Marketplace

**2x**

more likely to exceed profitability, market share & productivity goals

Source: Puppet/DORA, 2017 State Of DevOps Report <https://puppet.com/resources/state-of-devops-report>

**2x**

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

Source: Puppet/DORA, 2017 State Of DevOps Report <https://puppet.com/resources/state-of-devops-report>

## High Performers Win In The Marketplace

**2.2x**

higher employee Net Promoter Score

**50%**

higher market capitalization growth over 3 years\*

## High Performers Are More Secure And Controlled

**2x**

less time spent remediating security issues

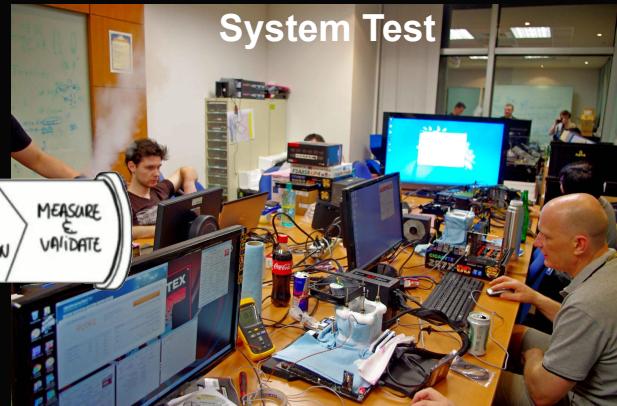
Source: Puppet Labs 2016 State Of DevOps Report <https://puppet.com/resources/state-of-devops-report>

**29%**

more time spent on new work

Source: Puppet Labs 2016 State Of DevOps Report <https://puppet.com/resources/state-of-devops-report>

# Tap All Your Sources of Feedback

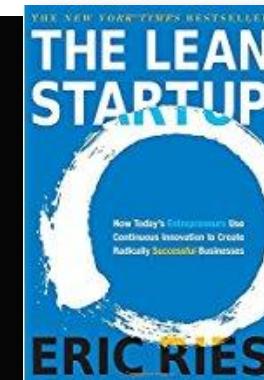
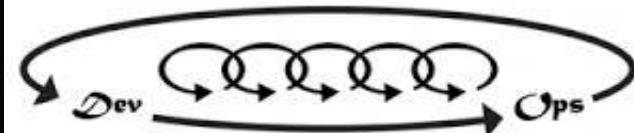


# DevOps: The Third Way - Experimentation & Learning

- Knowing how to get smarter, faster matters a lot.
- Smarter faster means finding fault early and often.
- Easier said than done.

- Create opportunities for learning as quickly, frequently, cheaply, and as soon as possible.
- Learn from accidents and failures, which are inevitable when working within complex systems.
- Institutionalize rituals that increase safety, continuous improvement, and learning
  - Establish a just culture to make safety possible
  - Inject ~~production~~ failures to create resilience
  - Convert local discoveries into global improvements
  - Reserve time to create organizational improvements and learning

Culture Of Continual Experimentation And Learning



- "Outlearn the competition"

# 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

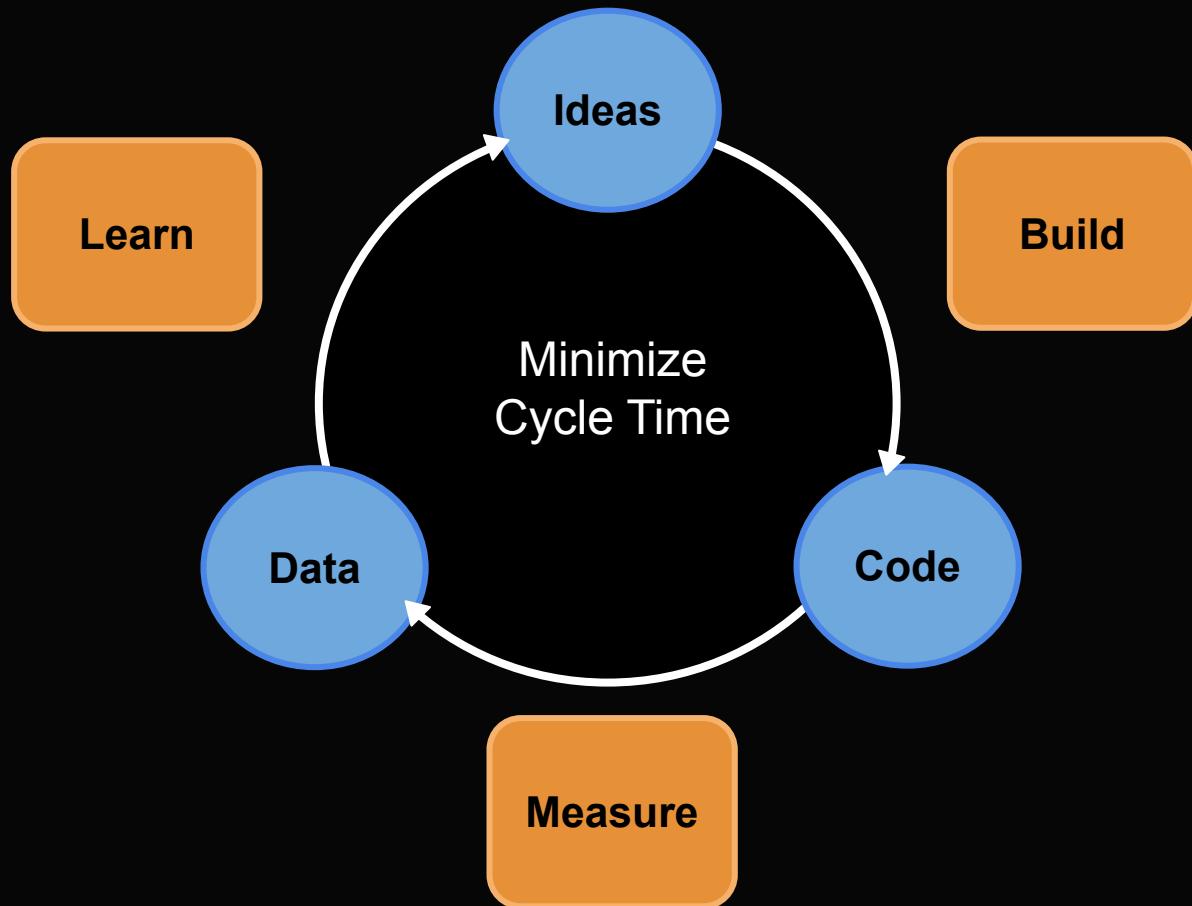
creating a DevOps meetup

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

# Lean Start-Up (by Eric Ries)



## Lean Startup 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.

# 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



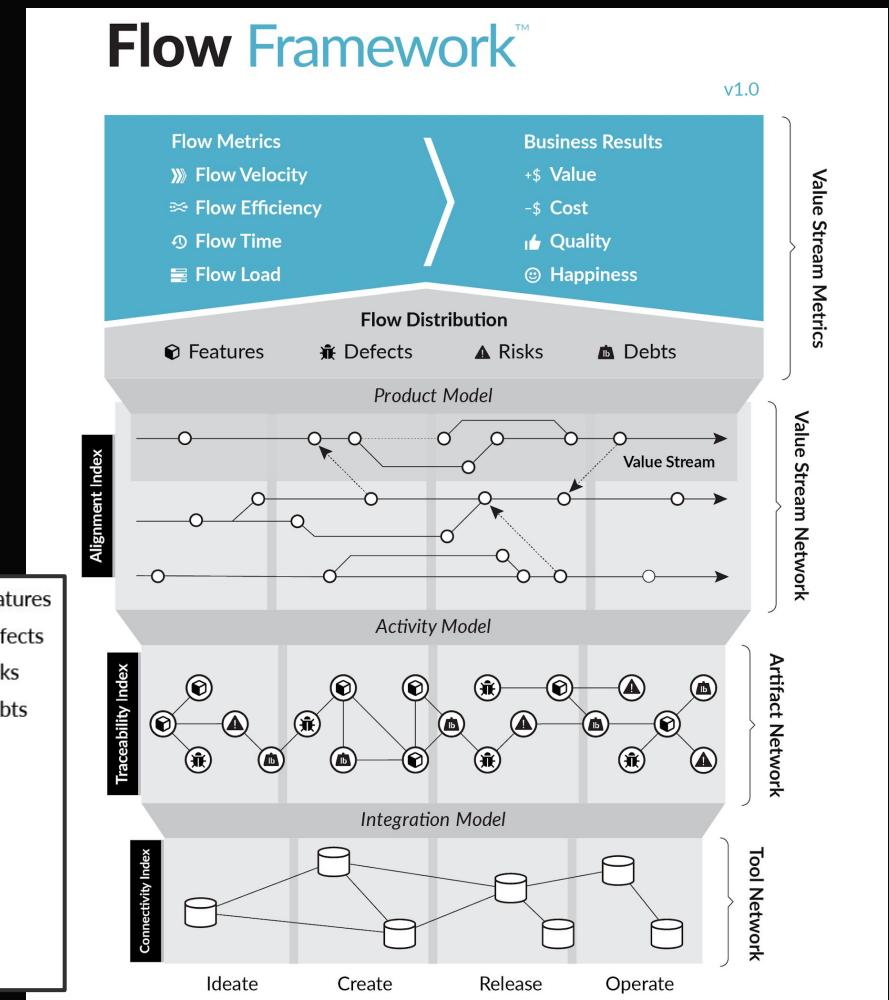
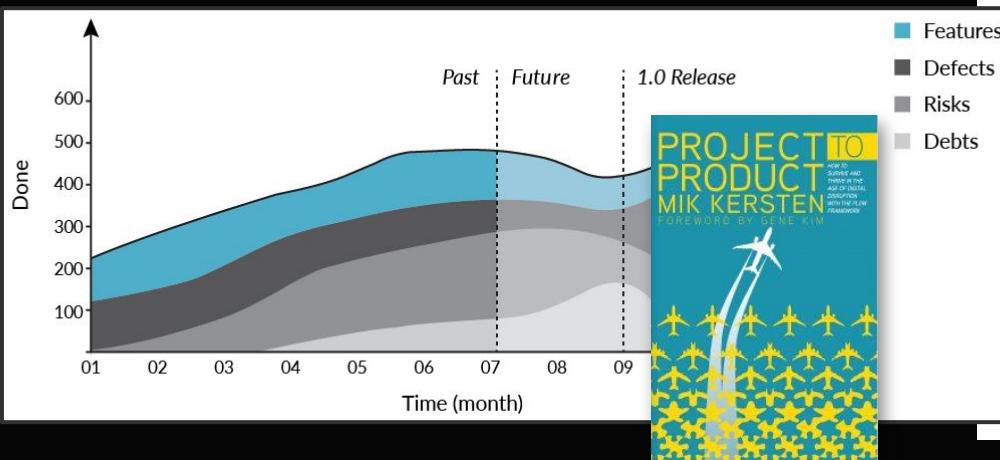
# 4 Types of Work

Features - Value Generation

Defects - Unplanned Mandatory Investment

Risks - Security, Reliability, Etc.

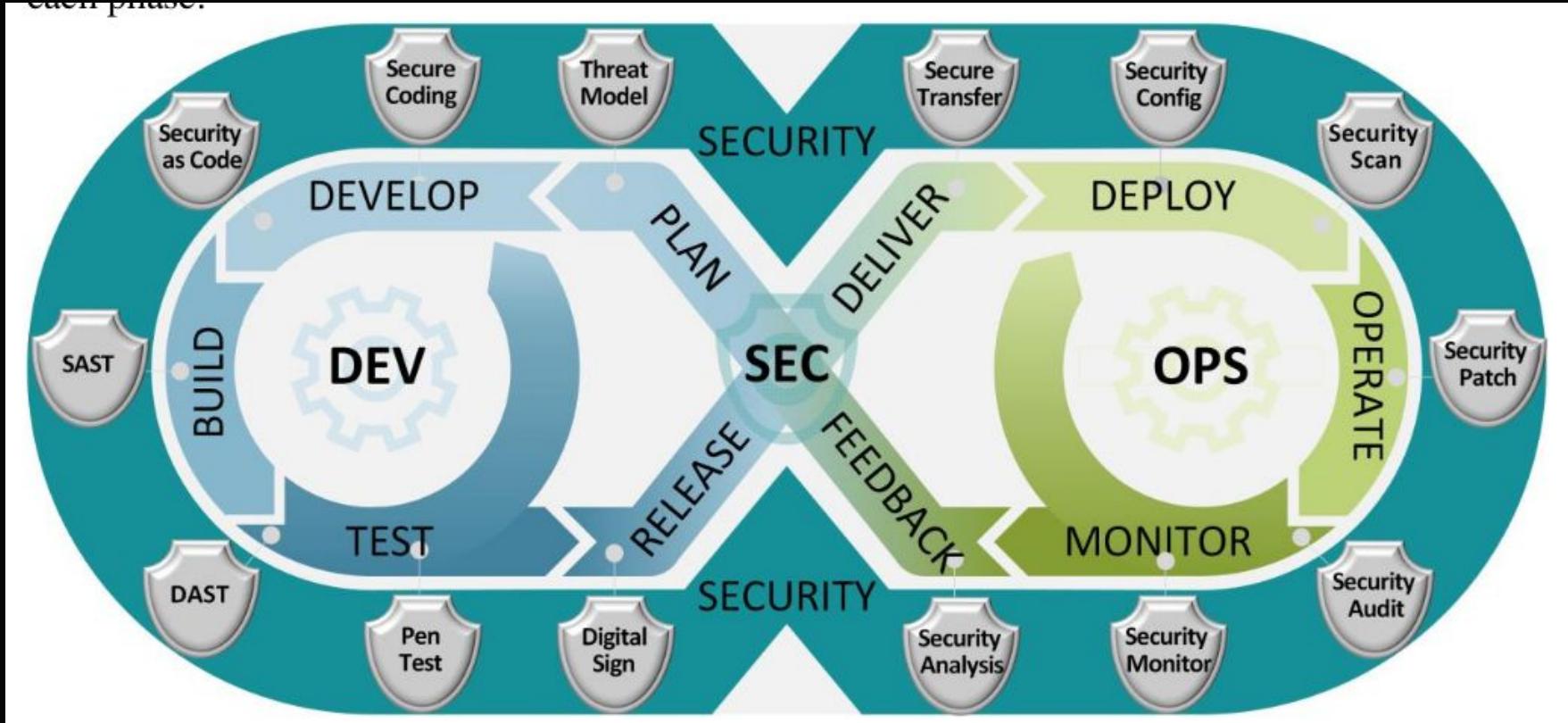
Debts - Impediments to Progress



Then commercial and defense industries emphasized Cyber Security within DevOps...

# DevSecOps

each phase.



UNCLASSIFIED



**DoD Enterprise DevSecOps  
Reference Design**

**Version 1.0  
12 August 2019**

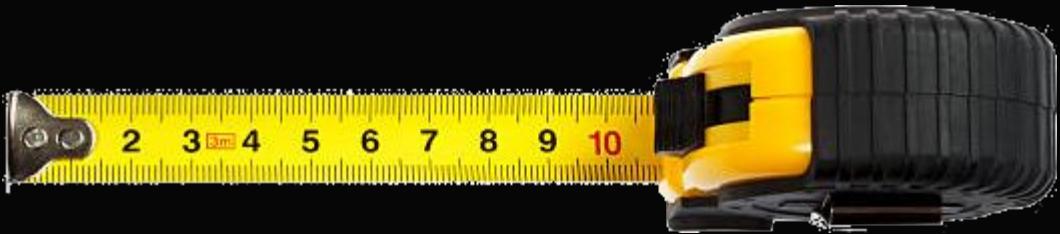
**Department of Defense (DoD)  
Chief Information Officer**

Distribution Statement C: Distribution authorized to U.S. Government Agencies and their contractors; Administrative or Operational Use; 29 November 2018. Other requests for this document shall be referred to the DoD Chief Information Officer.

# DevSecOps (as defined by DoD CIO)

DevSecOps is an organizational software engineering culture and practice that aims at **unifying software development (Dev), security (Sec) and operations (Ops)**. The main characteristic of DevSecOps is to **improve customer outcomes and mission value** by **automating, monitoring, and applying security at all phases of the software lifecycle**: plan, develop, build, test, release, deliver, deploy, operate, and monitor. Practicing DevSecOps provides **demonstrable quality and security improvements** over the traditional software lifecycle.

# Measures



**Mean-time to production**: the average time it takes from when new software features are required until they are running in production.

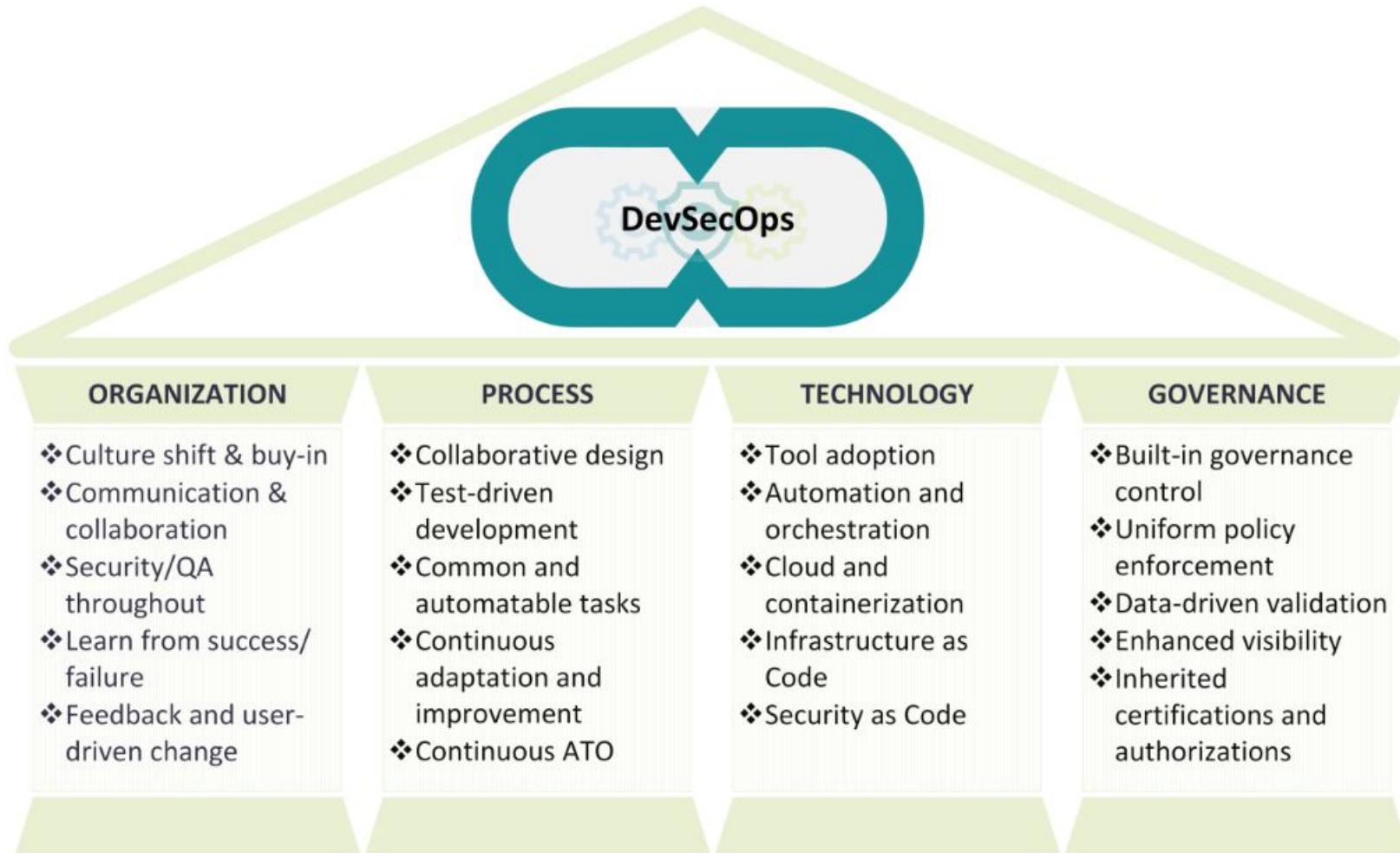
**Average lead-time**: how long it takes for a new requirement to be delivered and deployed.

**Deployment speed**: how fast a new version of the application can be deployed into the production environment.

**Deployment frequency**: how often a new release can be deployed into the production environment.

**Production failure rate**: how often software fails during production.

**Mean-time to recovery**: how long it takes applications in the production stage to recover from failure.



**Figure 4: DevSecOps Pillars**

# Software Factory

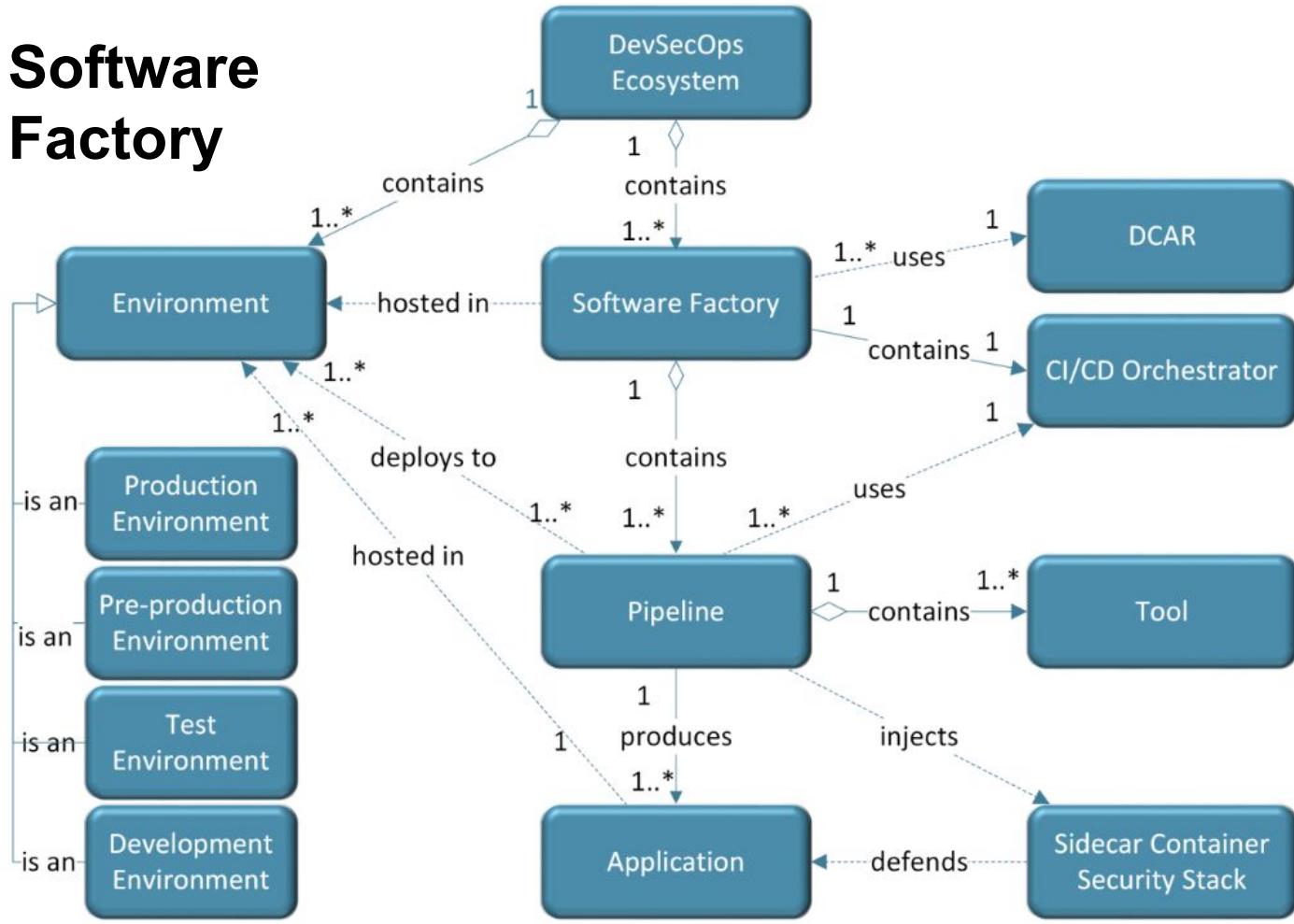


Figure 2: Conceptual Model

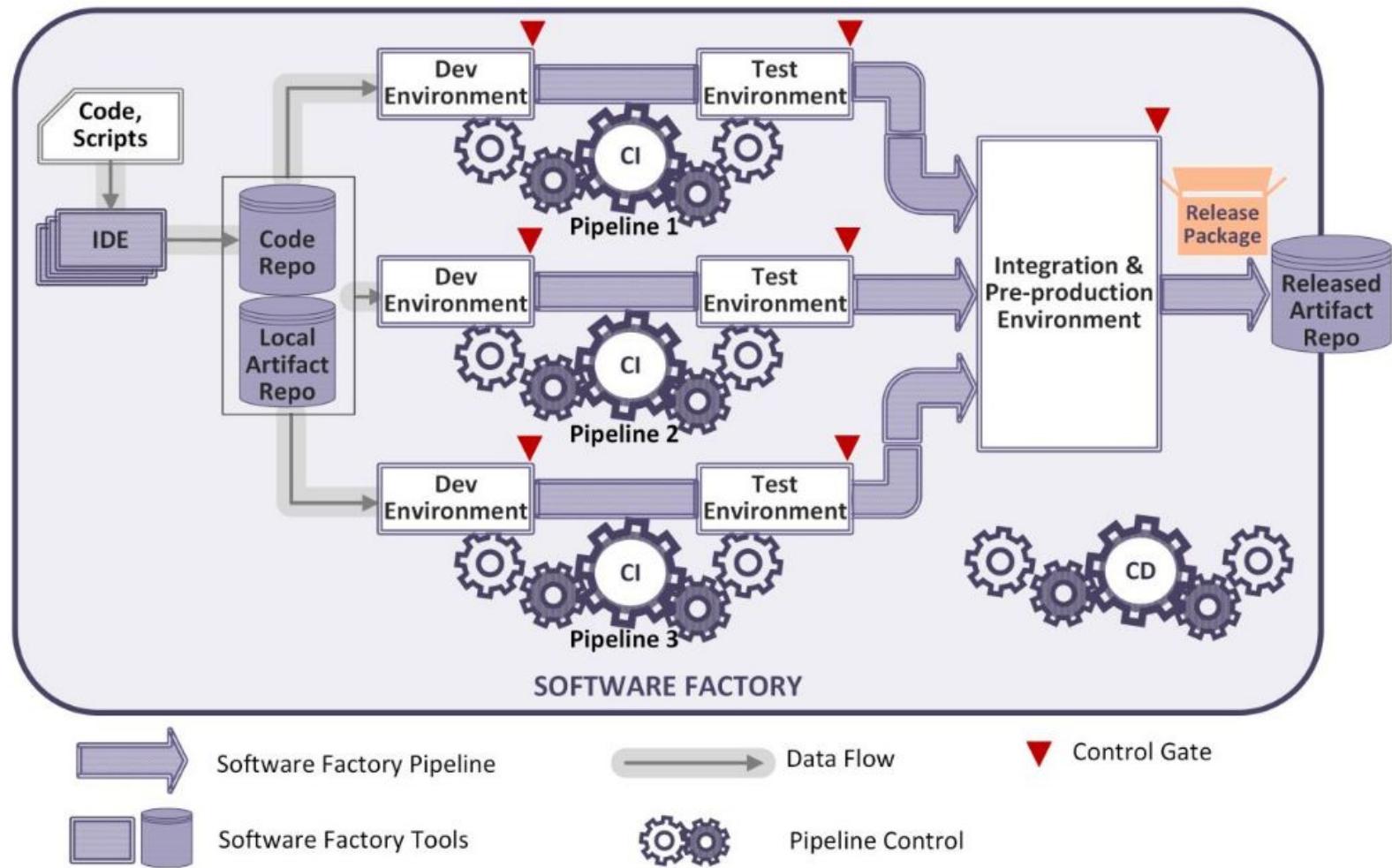


Figure 9: DevSecOps Software Factory

# DoD CIO DevSecOps MVP (...objective not shown)

## Planning

- Team Collaboration System
- Issue Tracking System
- Project Management System

## Develop

- Integrated Development Environment
- Source Code Repository

## Build

- Build Tool
- Container Builder
- Artifact Repository
- Static Application Security Test (SAST) tool

## Test

- Test Development Tool
- Test Tool Suite
- Test Coverage Tool
- Container Security Tool
- Container Policy Enforcement

## Release & Deliver

- Release Packaging Tool

## Software Factory

- CI/CD Orchestrator

## Production Operations

### Deploy

- Virtualization Manager
- CNCF-certified Kubernetes
- Configuration Automation Tools
- Service Mesh

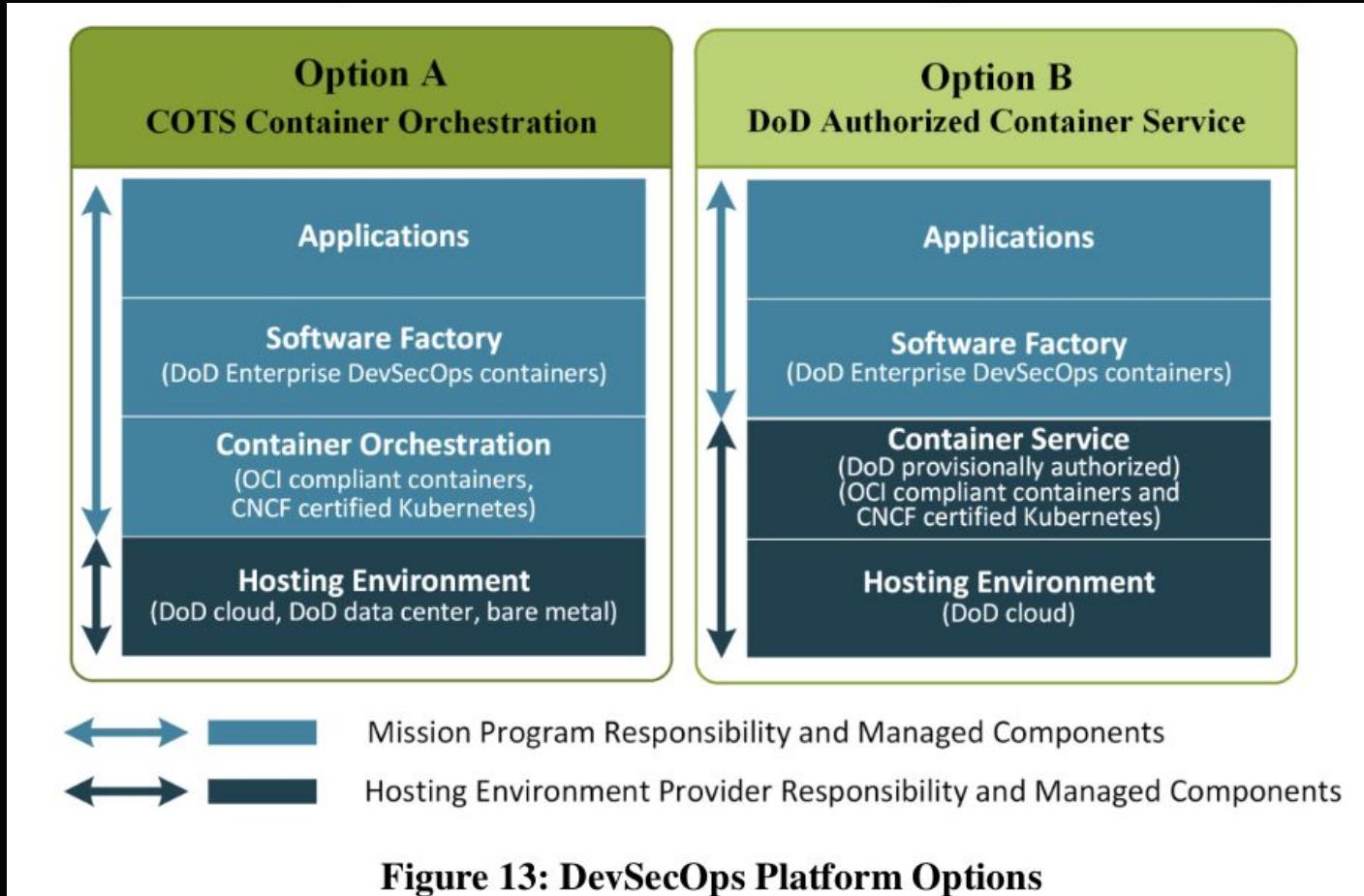
### Operate

- Backup Management

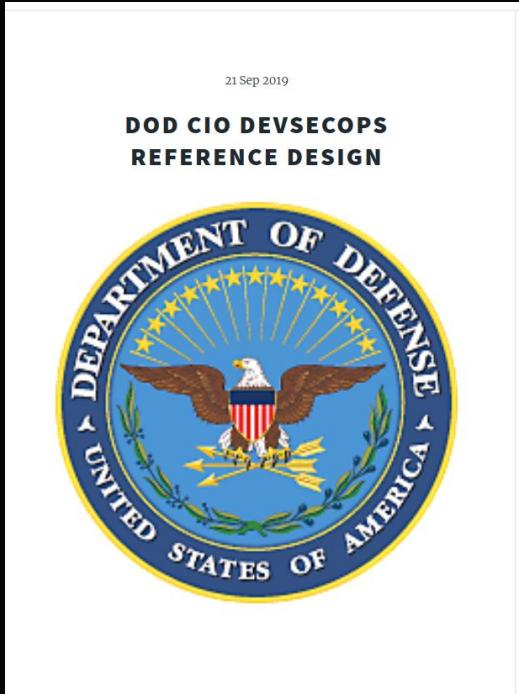
### Monitor

- Logging
  - Log Aggregator
  - Log Analysis & Monitoring
  - Operations Monitoring
- Information Security Continuous Monitoring (ISCM)
  - Alerting & Notification

# Containers Everywhere!



# See for yourself on DevOpsForDefense.org (in the Resources section)



Full document in web section.

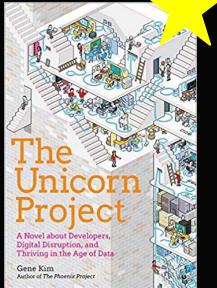
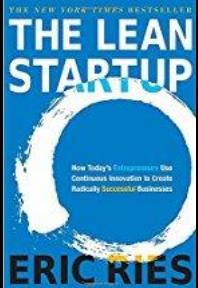
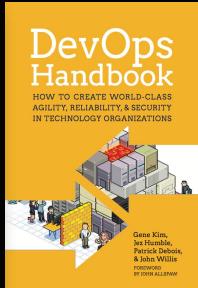
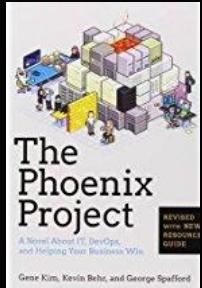
A screenshot of a presentation slide. At the top right is the date "14 Aug 2019". Below it is the title "DOD SOFTWARE FACTORY" in bold capital letters. To the left of the title is a small video thumbnail showing a man speaking at a podium. To the right of the title is a larger video thumbnail showing three people seated at a table with laptops. The main content area contains a slide titled "DevNation Federal 2019 - DoD Software : What Are We Doing?". The slide lists several bullet points about joint programs, software engineering, and security. At the bottom of the slide is the text "Integrating, Accelerating, Resilience".

Conference presentation in video section.

# DevOps Resources

<https://devopsfordefense.org/resources/>

Books / Publications:



The  
Phoenix  
Project

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

Gene Kim, Kevin Behr, and George Spafford

REVISED WITH NEW  
INTRODUCIONS

DevOps  
Handbook

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

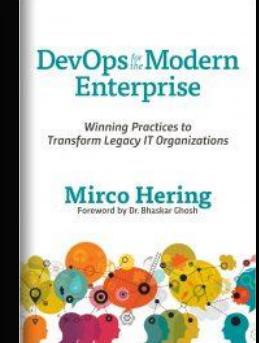
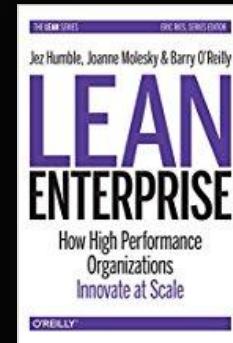
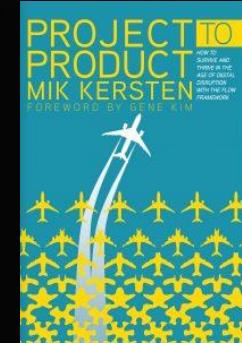
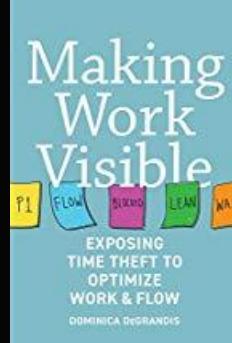
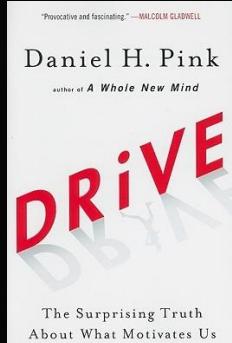
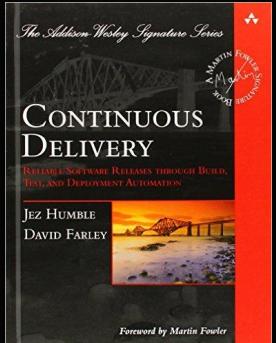
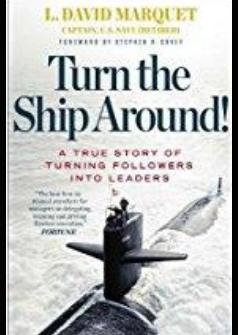
THE NEW YORK TIMES BESTSELLER  
**THE LEAN  
STARTUP**

Eric Ries

The  
Unicorn  
Project

A Novel about Developers,  
Digital Disruption, and  
Thriving in the Age of Data

Gene Kim  
Author of The Phoenix Project



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

Conference Presentations (YouTube):

- DevOps Enterprise Summit (DOES)
- IT Revolution
- Velocity
- GoTo

