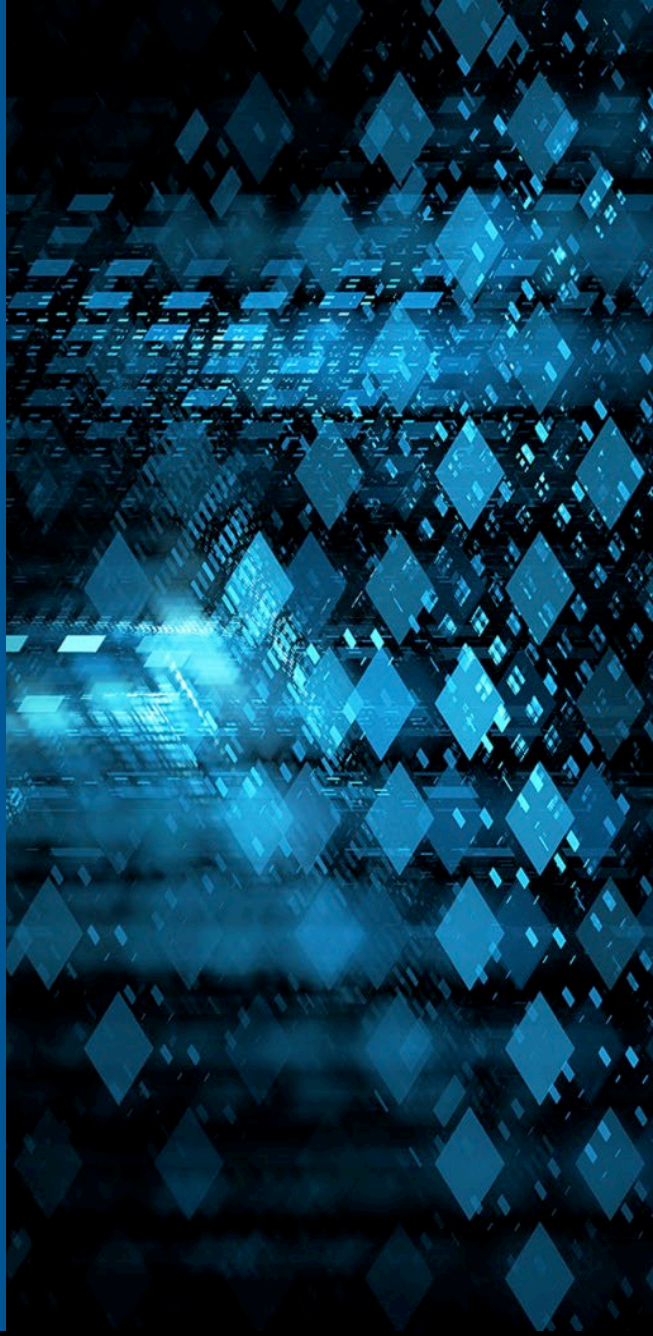
Security Practitioner Perspective on DevOps for Building Secure Solutions

Zane Lackey

Hasan Yasar

Software Engineering Institute
Carnegie Mellon University
Pittsburgh, PA 15213



This material is based upon work funded and supported by the Department of Defense under Contract No. FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM-0004111

This talk will cover the perspectives of security practitioners on building secure software using the DevOps development process and modern security approach.

Building Secure Solutions

DevOps Foundations

Picture (optional)

The DevOps Movement Began as a Reaction …

to years of disconnect between Development and Operations that began to manifest itself as conflict and inefficiency

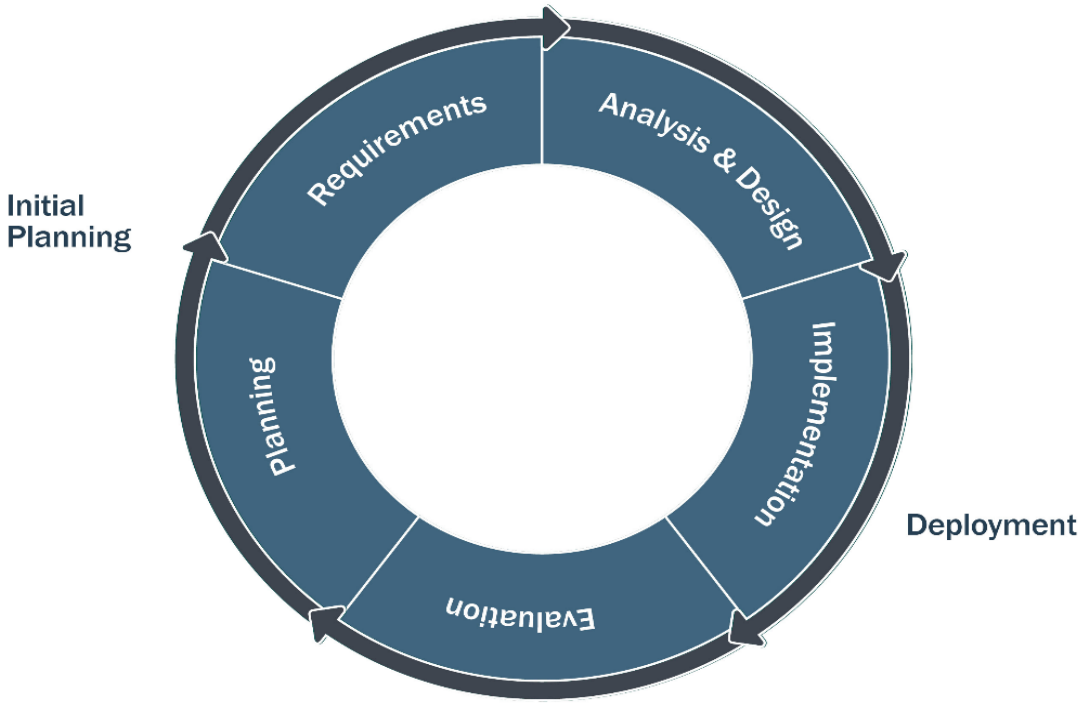
What is DevOps?

DevOps (a portmanteau of "development" and "operations") emphasizes *communication*, *collaboration*, and *integration* between software developers and information technology (IT) operations personnel. [1]

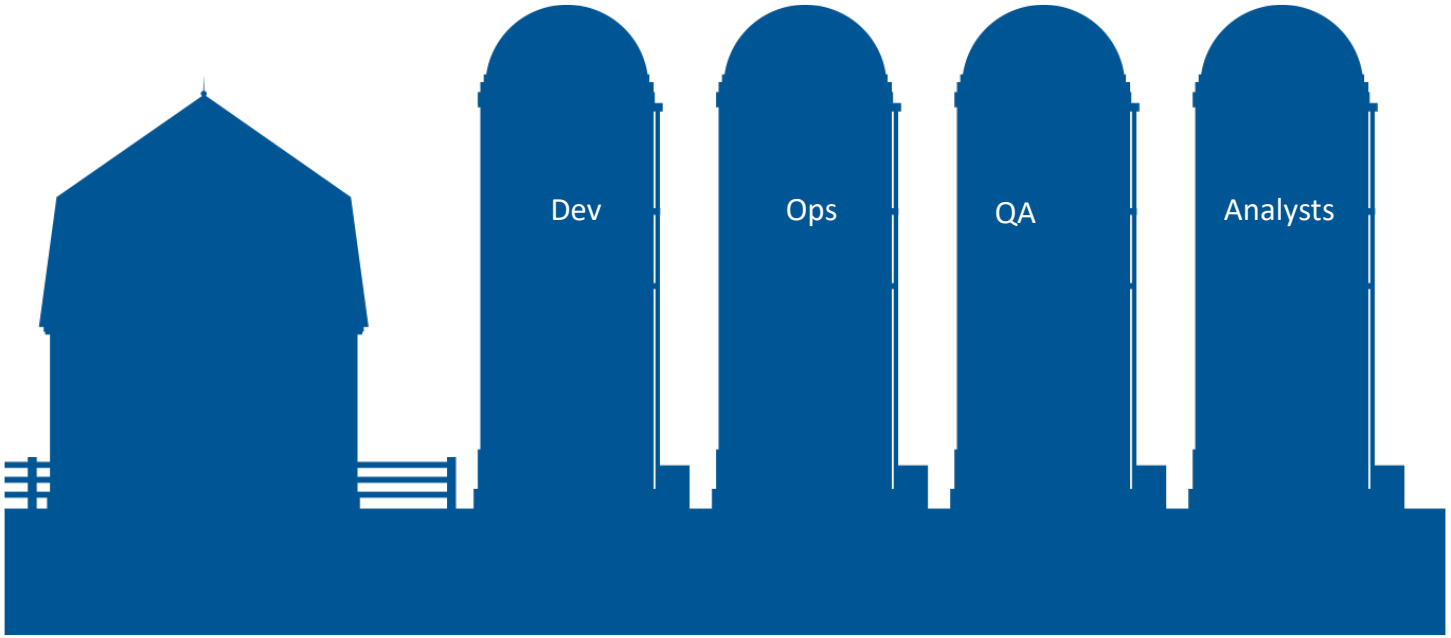
[1] http://en.wikipedia.org/wiki/DevOps

Picture (optional)

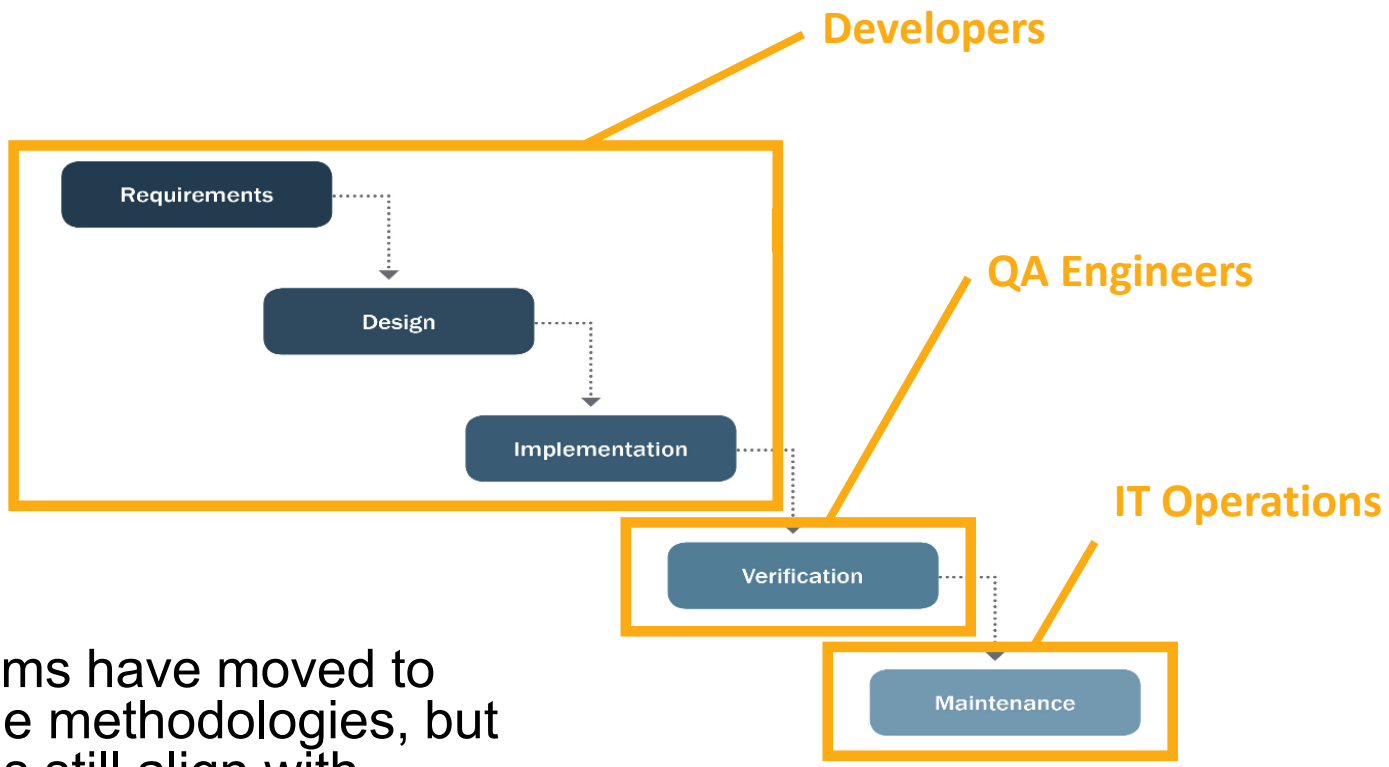
Agile Method



Silos Block Collaboration



Silos Reinforce Waterfall



Teams have moved to Agile methodologies, but roles still align with waterfall methods

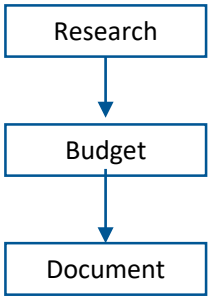
Picture (optional)

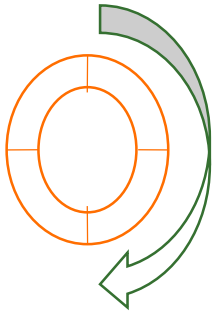
Water - Scrum - Fall

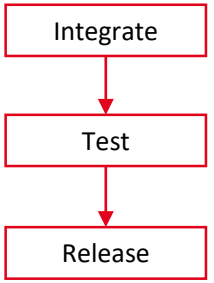
Business

Development

QA
Operations







Jez Humble, https://youtu.be/L1w2_AY82WY
Dave West, http://sdtimes.com/analyst-watch-water-scrum-fall-is-the-reality-of-agile/

DevOps is an Extension of Agile Thinking

Agile

Embrace constant change

Embed Customer in team to internalize expertise on requirements and domain

DevOps

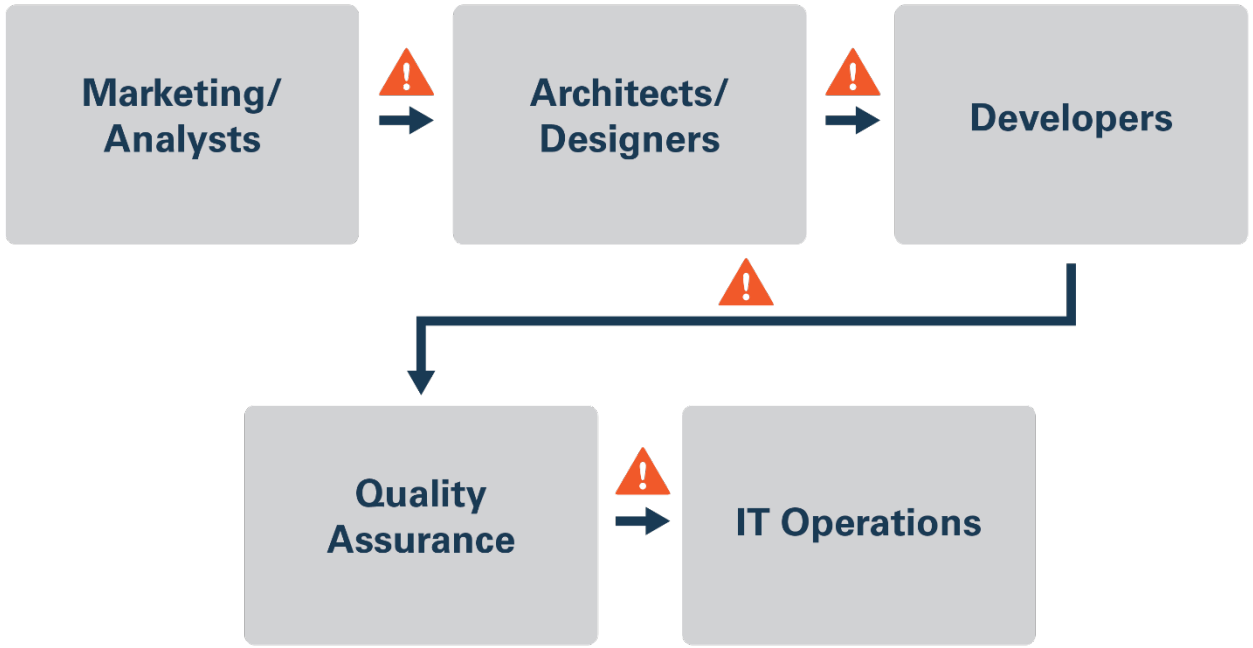
Embrace constant testing, delivery

Embed Operations in team to internalize expertise on deployment and maintenance

Polling ?

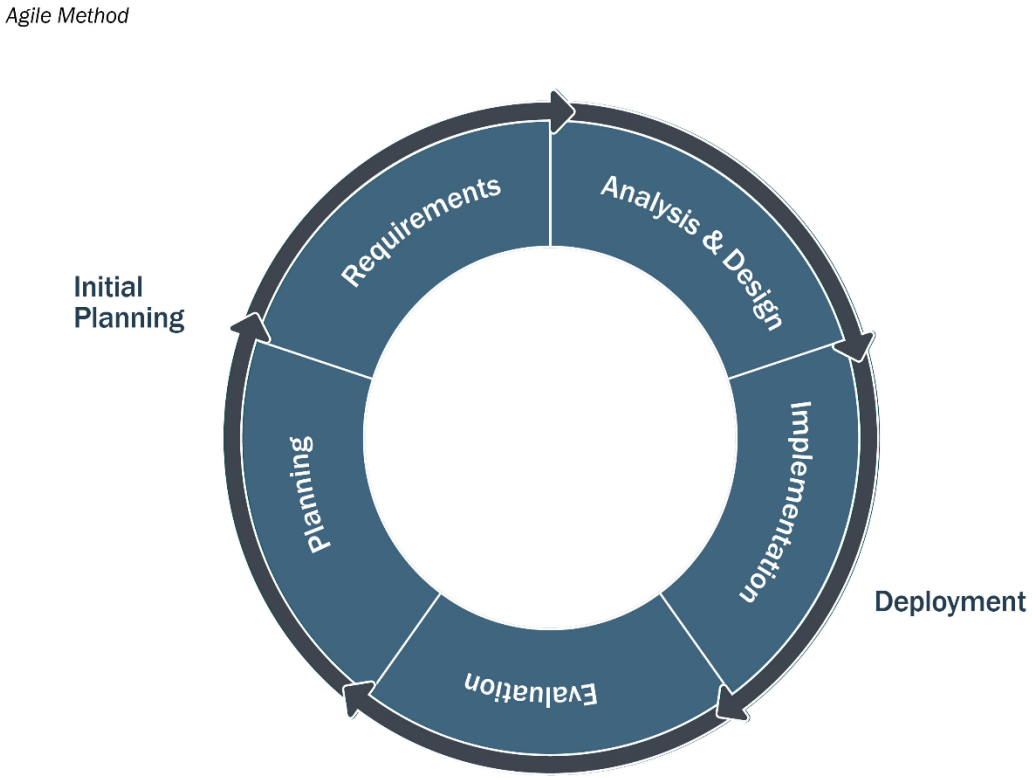
Does your organization follow DevOps process and methodologies?

Every Transition of the System is a Risk

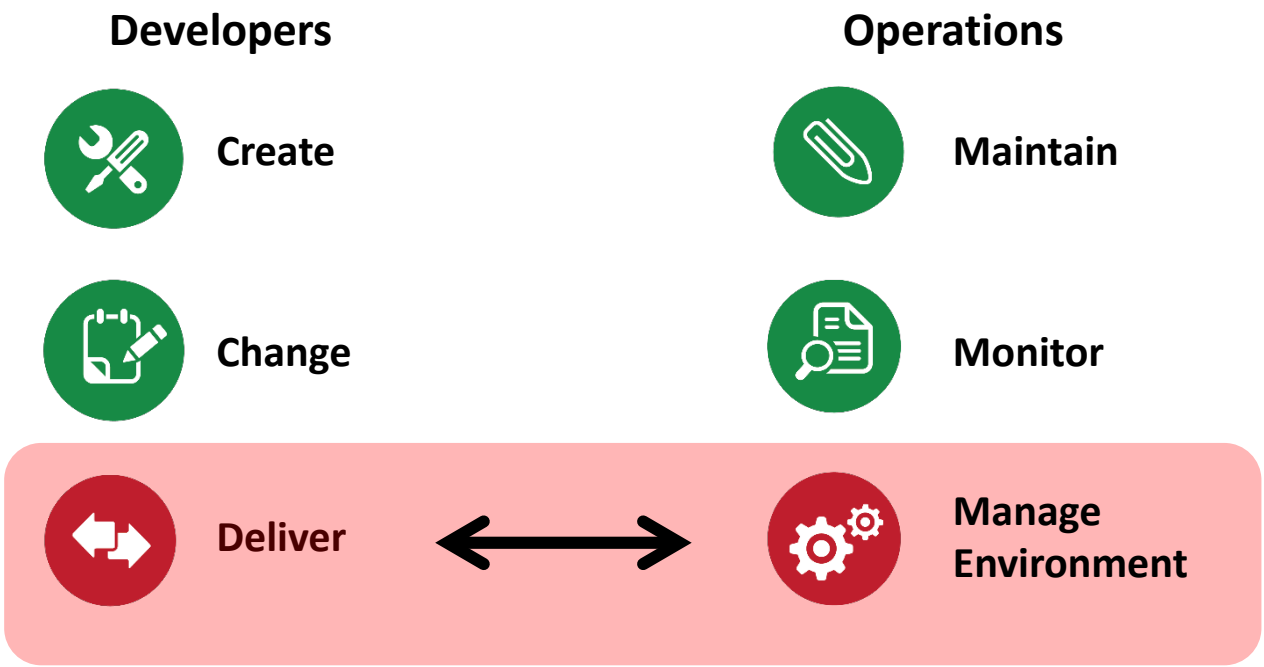


Agile Means Constant Transition

Picture (optional)



Significant Collaboration Is Needed Where Paths Intersect



To address these pain points, DevOps promotes Collaboration

Heavy collaboration between Dev and Ops on:

- Design / Architecture decisions
- Environment / Network configuration
- Deployment planning
- Code Review

Constantly available open communication channels:

- Dev and Ops together in all project meetings
- Chat/Email/Wiki services available to all team members
- Dev / Ops report together as one project team

Picture (optional)

An Engaged, Cross-Functional team is needed

Early involvement of experts

- Ops = experts in maintainability and deployability

Complete engagement

- Don’t bring Ops Engineers in as consultants – make them first-class team members with same success criteria as devs

Break down organizational silos

- Enable and require constant communication

DevOps Aims to Increase…

…the pace of **innovation**

…**responsiveness** to business needs

…**collaboration**

…software **quality**

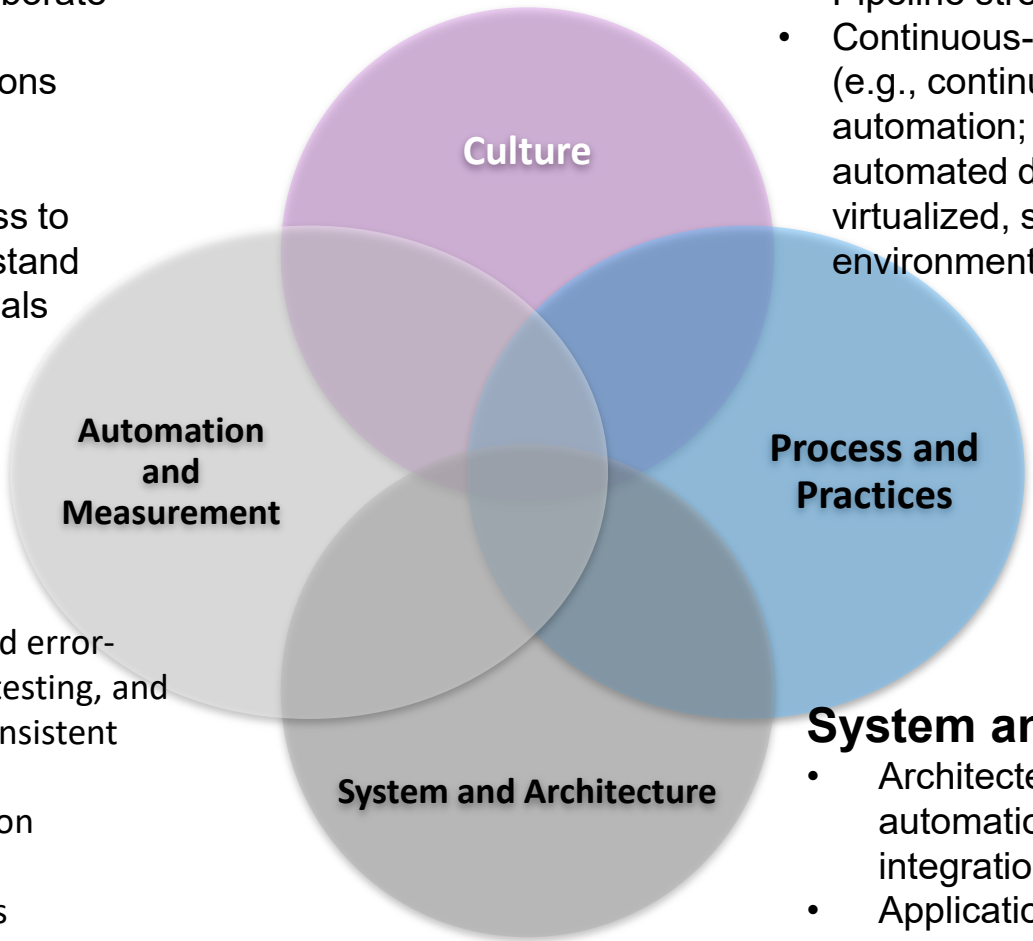
Multiple Dimensions of DevOps

Culture

- Developer and Ops collaborate (Ops includes security)
- *Developers* and Operations support releases beyond deployment
- Dev and Ops have access to stakeholders who understand business and mission goals

Process and Practices

- Pipeline streamlining
- Continuous-delivery practices (e.g., continuous integration; test automation; script-driven, automated deployment; virtualized, self-service environments)



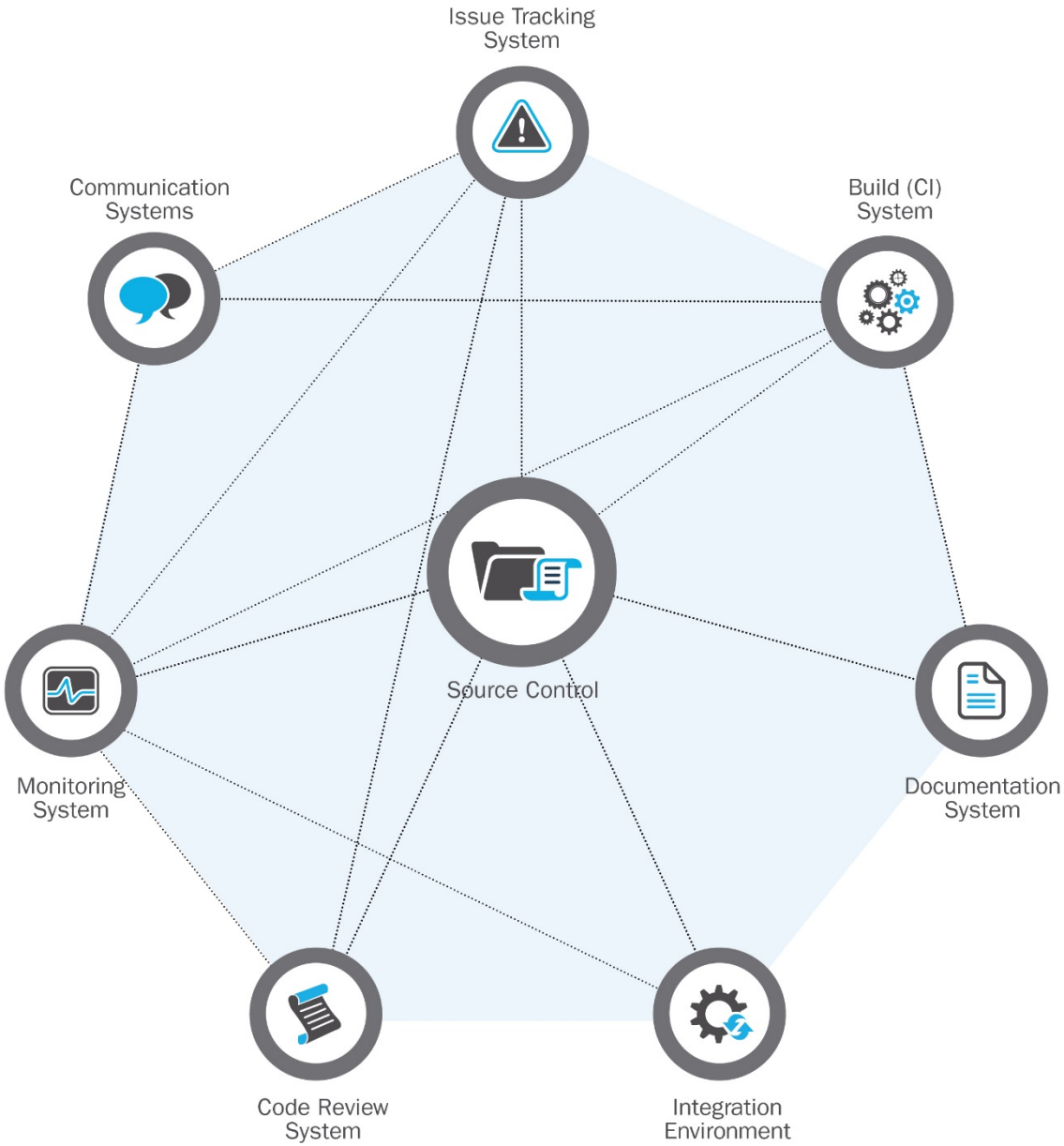
Automation/ Measurement

- Automate repetitive and error-prone tasks (e.g., build, testing, and deployment maintain consistent environments)
- Static analysis automation (architecture health)
- Performance dashboards

System and Architecture

- Architected to support test automation and continuous-integration goals
- Applications that support changes without release (e.g., late binding)
- Scalable, secure, reliable, etc.

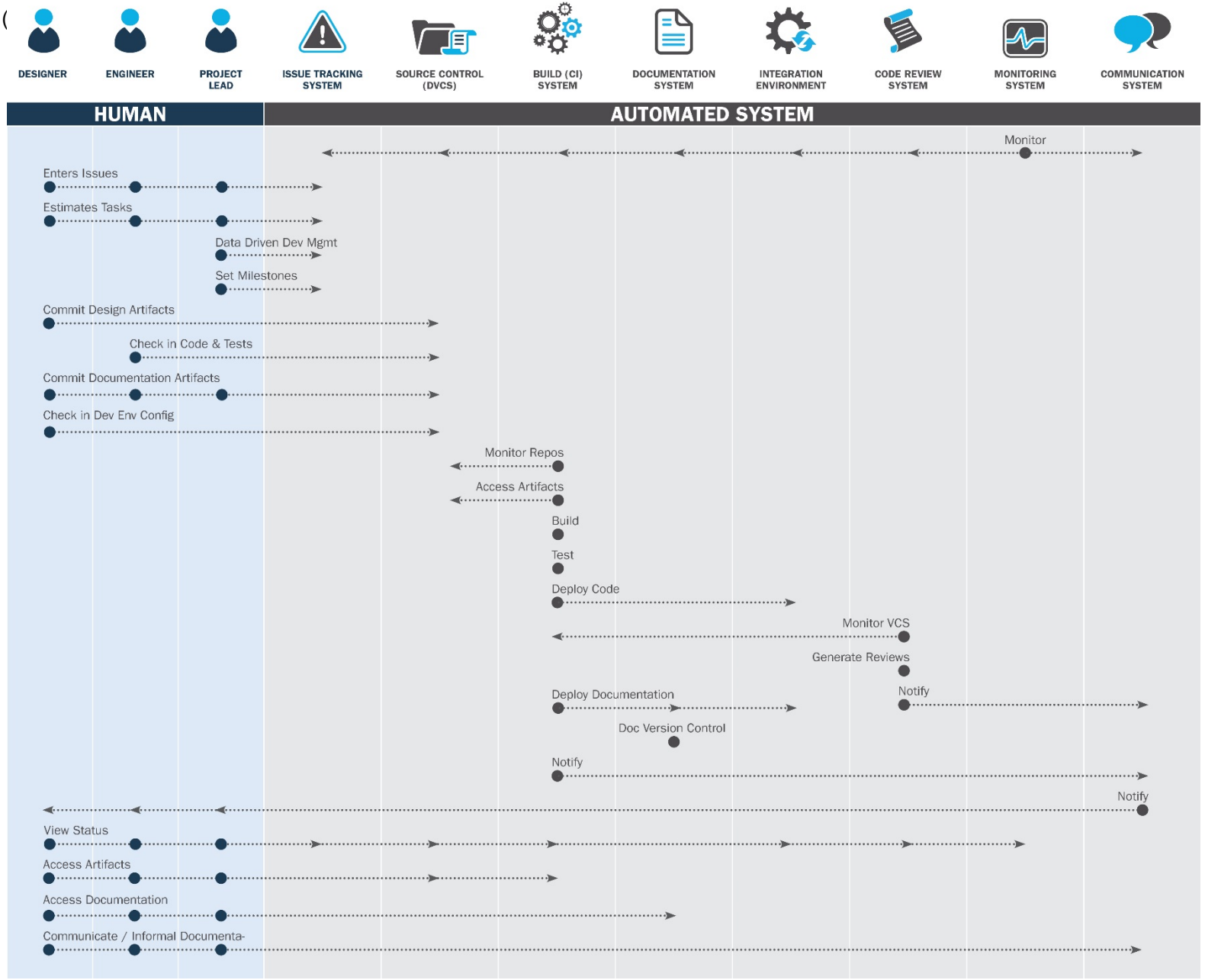
Picture
(optional)



Integration and communication, even among tools, is the key to integrate Security into Development Platform!

Section (

Picture (optional)



Building Secure Solutions

DevOps Lesson Learned

Polling ?

Do you have Security Ops Team as part of development activities?

Picture (optional)

For security teams, the world has changed in three fundamental ways:

- Agility means code deployment is trending to near-instantaneous
- Security is no longer the gatekeeper to deployment
- If security is a blocker, it will be routed around

Near-instantaneous deployment?

A simulation of deploying code in the waterfall model

What is this shifting to?

An agility example: Etsy pushes to production **50 times a day** on average

Constant iteration **in production** via feature flags, ramp ups, A/B testing

But doesn't the rapid rate of change mean things are less secure?!

Actually, the opposite is true

They key to realize is vulnerabilities occur in **all** development methodologies

They key to realize is vulnerabilities occur in **all** development methodologies

…But there's no such thing as an out-of-band patch in continuous deployment

Compared to:

“We’ll rush that security fix. It will go out … in about 6 weeks.”

- **Former** vendor at Etsy

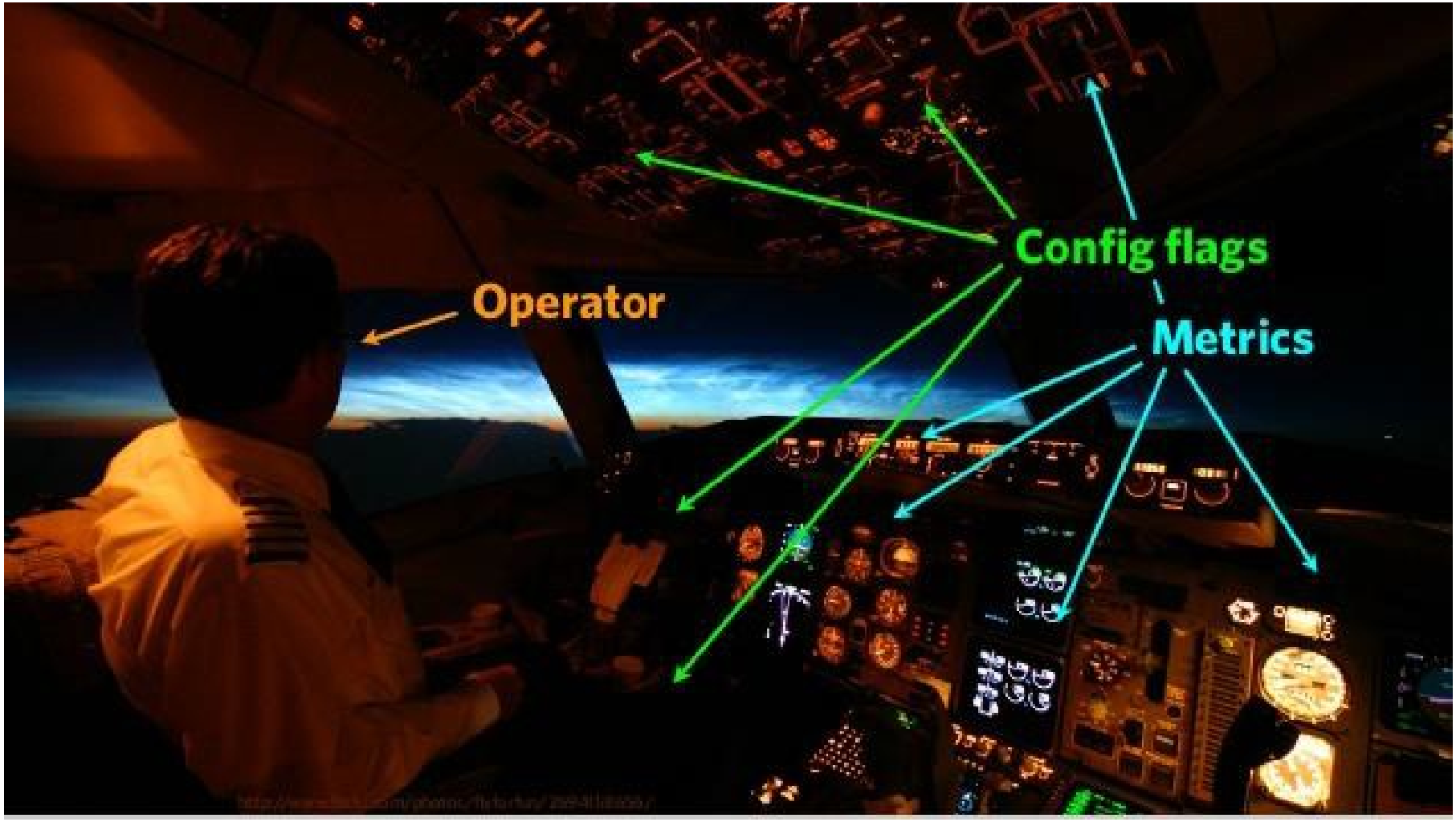
Polling ?

Do you believe that the DevOps process, mainly Continuous Delivery is a barrier for application security?

What makes continuous deployment safe?

What makes continuous deployment safe?

Visibility

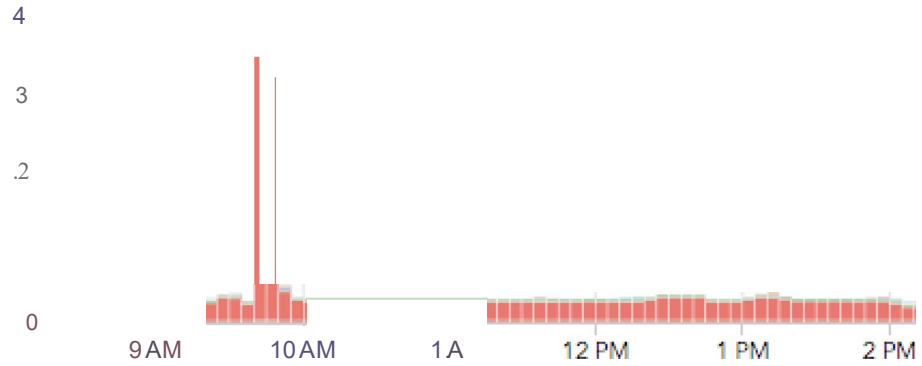


Source: http://www.slideshare.net/mikebrittain/advanced-topics-in-continuous-deployment

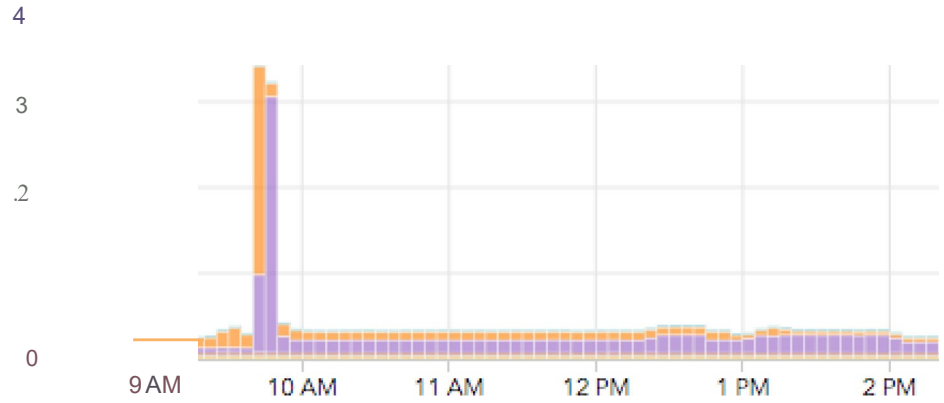
The same hard lessons are slowly shifting to security

Ex: Which of these is a quicker way to spot an attack?

Attacks >



Anomalies >



Increase agility by surfacing security visibility for **everyone**, not just the security team

Having to talk to security to get security awareness causes delays

Having to talk to security to get security awareness causes delays

Delays get routed around

To embrace agility, security has to decentralize

Picture (optional)

Lessons Learned:

- Embracing DevOps/Agile/Continuous Deployment helps not harms security
- Visibility is the key to moving quickly and safely
- You (in the general case) are never going to be able to hire enough staff, so steal everyone else's

More on SEI DevOps Blog

https://insights.sei.cmu.edu/devops

https://signalsciences.com/resources/

Picture (optional)

Thank you!

zane@signalsciences.com
hyasar@sei.cmu.edu

@zanelackey
@securelifecycle