



NAVAL
POSTGRADUATE
SCHOOL

Innovation in Software Acquisition: The Good, Bad, and Ugly

Author: Jeff Dunlap, CAPT USN (Ret.)

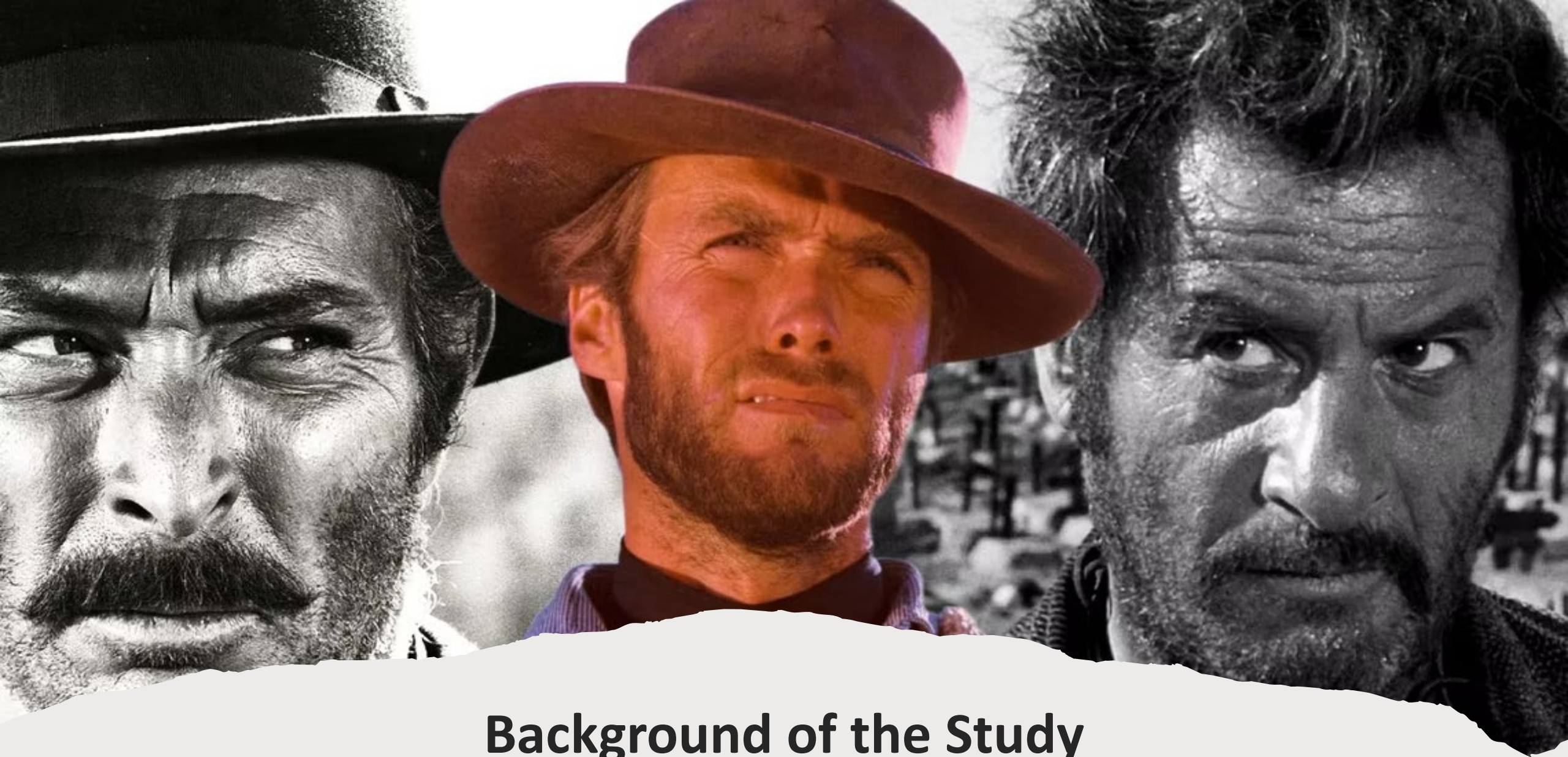
Date: May 9th, 2024



AGENDA



Background of the Study	01
Problem Statement	02
Framework	03
Methodology	04
Analysis	05
Contact Information	06



Background of the Study

Software done wrong for the right reasons is worse than continuing with the status quo
Not understanding the 'How and the Why' prior to execution can lead to the pit of despair

Problem Statement

Software can provide tactical advantages during modern conflicts *if designed for change*

SCOPE OF THE STUDY

Most of US weapons systems are not capable for rapid change both from cultural/process resistance and architecture entrenchment

RELEVANCE OF THE STUDY

The impact of smaller, less expensive weapon systems dictating the order of battle moves the software adaptability to the edge of conflict

RESEARCH QUESTIONS

- Why are legacy weapon systems slow to respond to modern software practices?
- What prevents software capability deliveries to the tactical edge
- How a faster development cadence exposes process mismatches
- What are some best practices and nuggets for Program Managers to consider?



Framework

OVERVIEW



The Department of Defense (DoD) has recognized that interconnected warfighting systems are vulnerable due to their inability to swiftly adapt to new technologies and effectively combat advanced cyber threats.

PROONENTS



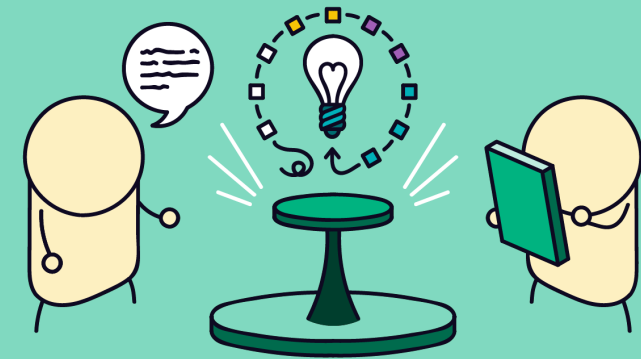
The commercial sector has developed methods to rapidly and securely implement new software capabilities without significant interference to current operations.



METHODOLOGY

QUALITATIVE METHOD

Conducting interviews with current thought leaders within the DoD and Industry to gain insights into challenges and key nuggets to consider before embarking on software modernization



Agile



Agile software development methods embrace an adaptive approach, breaking the project into smaller, manageable iterations or sprints to deliver working software of value to users sooner.

Frequent feedback loops and opportunities exist for course correction based on user input and evolving needs.

Retiring technical debt earlier and delivering working software sooner allows for the early identification and mitigation of issues.

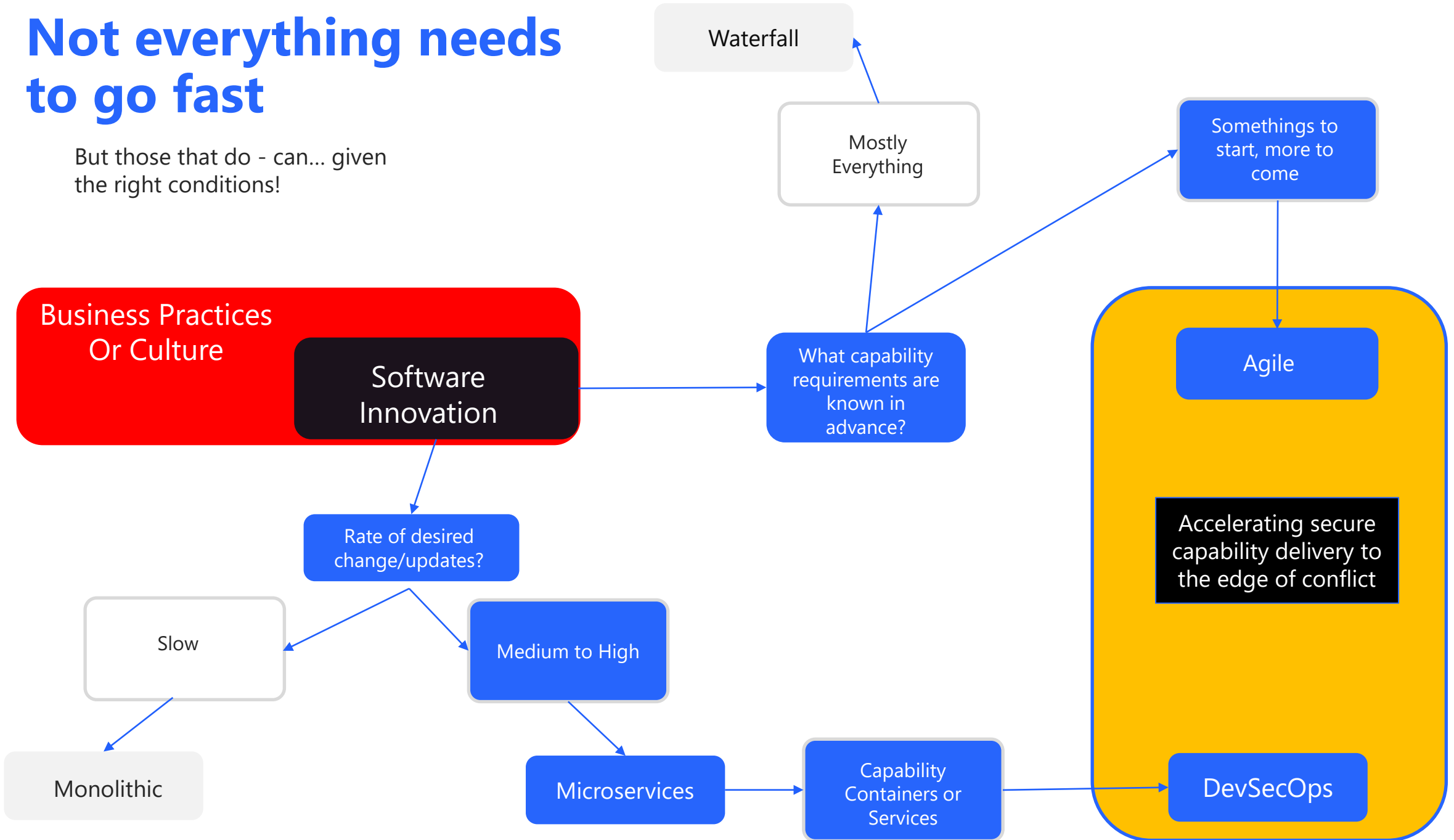
DevSecOps

DevSecOps (aka Software Factory) delivers faster and more frequent releases into the production environment and becomes available to operations based on their need.

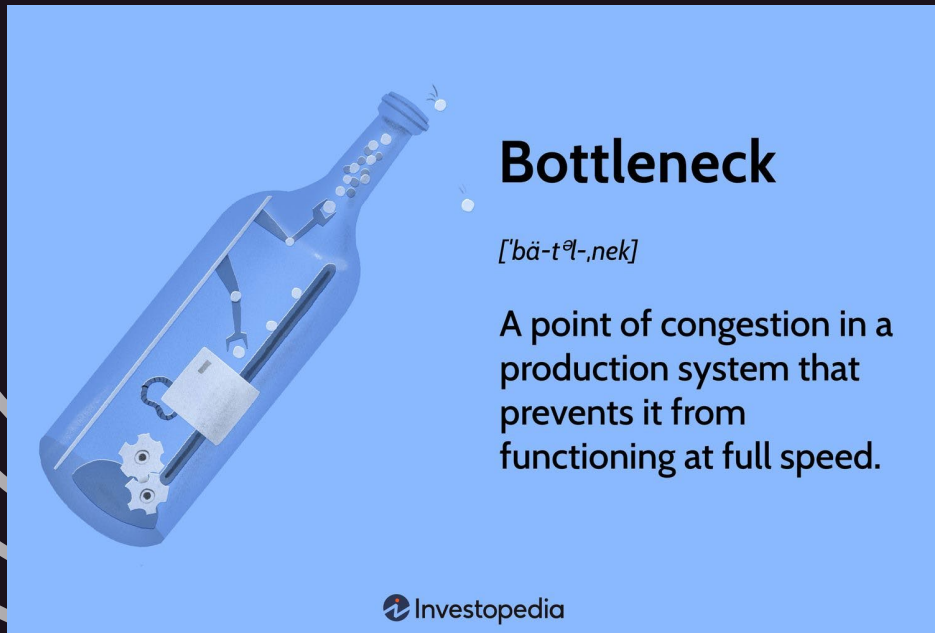
Continuous Integration and Continuous Delivery (CI/CD) accelerate DevSecOps, where automated builds and tests are run continuously and delivered to the production environment

Not everything needs to go fast

But those that do - can... given the right conditions!



Speed Bottlenecks



Applying legacy software certification processes to DevSecOps creates a Speed bottleneck

Manual testing and risk-averse siloed teams can decelerate the advantages gained in the DevSecOps software development lifecycle

Fixing the current speed bottleneck triggers the next

Be careful on what you ask for

The cadence of continuous software delivery (or deployment) into the operational environment dramatically impacts the DevSecOps architectural investments aligned with the need for agility.

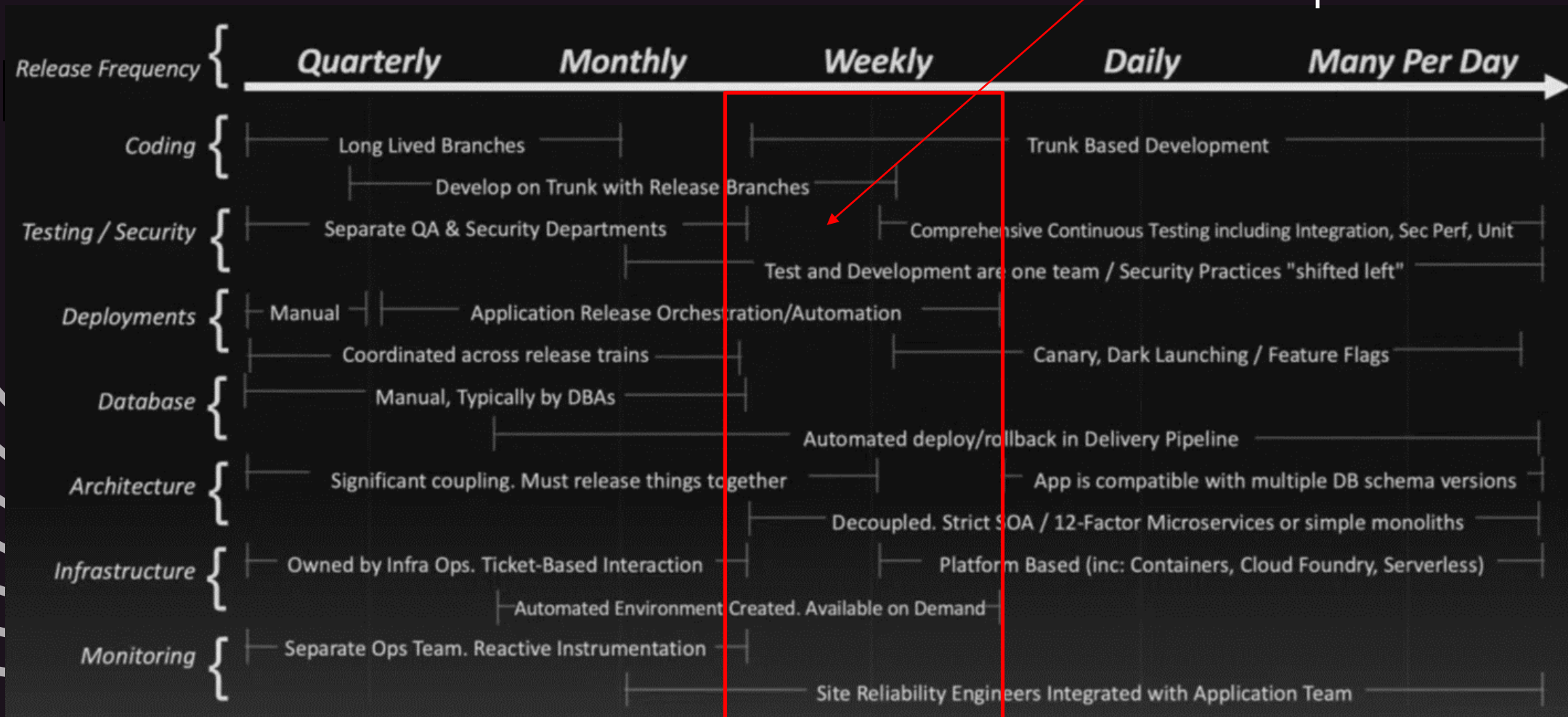
The operational 'user burden' becomes the driving factor for the release frequency

Be careful on what you ask for

Software Release	System Impact	User Burden	Cadence
Major new features	Software installation with Testing	Training / System downtime	Annually

Everything Changes with release frequency

Pit of Despair



Example: SW Deployments

Quarterly with coupled, distributed systems

- Zero-downtime unimportant
- Feature flags low priority
- Rollback is nice to have
- Releasing what was tested together - critical

Daily

- Zero-downtime with auto-rollback is critical
- Feature flags make sense
- Creating “release sets” - irrelevant

*World class at one
cadence is terrible
at another*



How to deliver faster?

To move from weekly or faster: change “everything” all at once

- New architecture
- New tooling
- New processes

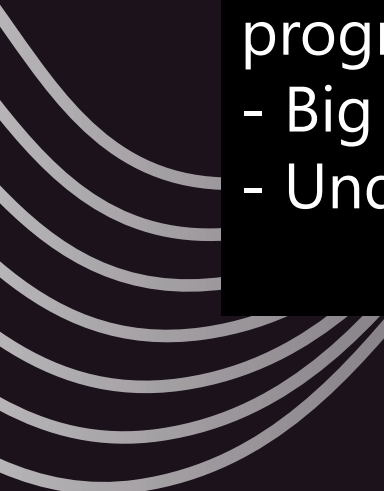
World class at one cadence is terrible at another

SW Program Management Observations

- Top Cover
- Clear and Well-Defined Requirements for the PM
- Effective Project Management
- Agile Development Methodology
- Skilled Development Team
- User-Centric Design
- Modular Contracting
- OTA (Other Transactional Authority) Contracts
- Defending the Budget using DevSecOps
- Ongoing Support and Maintenance



Nuggets from PEO IWS-X ICS PM

- The DoD is its own unique culture... but it is not that different
 - Deliberately build your culture
 - Know your Stakeholders and Constantly Communicate
 - Have a robust Communications Plan and Talking Points
 - Know your Market and Compete where you can (forward progress is better than none)
 - Big Bang almost always fails
 - Under promise and over-deliver
- 

Nuggets from PEO IWS-X ICS PM

- Hope that the competition underestimates you because that gives you an opening
- Change takes investment... but you will have to earn it...
- Humility – Your first idea or version is almost always bad
- The faster you learn, the better
- You need to educate the entire industry on what you are doing
- Slowly Force Change
- Both an Industrial and Digital Mindset is required to be a successful change agent in the future.

Good, Bad, the Ugly

DoD Software Acquisition has been a **system accelerating without a complete understanding of the bureaucratic resistance and business practices necessary to achieve velocity.**

Major considerations are needed for modular and flexible contracts, incorporating testing and evaluation throughout the software process and shifting left certifications and approvals to deploy at the speed of relevance.

The importance of a trained and skilled workforce with user interaction and senior leadership support cannot be understated.



Good, Bad, the Ugly

Focus is needed in understanding the significance of reasonable and prioritized requirements, advocating for a shift from compliance-based, overly prescriptive requirements to more iterative approaches like iterative/agile development approaches to reduce cost, risk, and time.

A comprehensive approach that includes effective project management, stakeholder engagement, risk management, and a focus on iterative development and continuous improvement.

Culture and behaviors take time to adjust to the applied force; they must be constant and consistent to ensure that the capability delivered is responsive to a changing threat.

Contact for inquiries

Email Jeffrey.dunlap@nps.edu

Social Media www.linkedin.com/in/captjeffdunlap

Call or Text (619)-370-6647



NAVAL
POSTGRADUATE
SCHOOL

