SEI Virtual Learning Package 11:

DevSecOps for All!

SEI Continuous Deployment of Capability Directorate

May 2020

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. FA8702-15-D-0002 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center.

DM20-0439

Meeting Conventions for Today

Please stay on mute for the lecture portion of the course module. If all goes according to plan, you will be muted automatically when you come into the Skype meeting (both web and dial-in participants). The lecture portion will be recorded for future use.

If you are “in” the Skype meeting via web or app, please ask questions via the Chat window.

- A facilitator will collect the questions and either pass them to the facilitator if something immediate, or organize them for the Q&A portion of the course module

Those on dial in will enter questions by sending email to your site POC (David Walbeck)

Instructor will call for participation and discussion at break points. Please remember to come off mute before talking. Discussions will NOT be recorded.

When you are done talking, before going back on mute, please say “Over” so others know you are finished

Topics the SEI will address in this module include:

- BLUF (Bottom Line Up Front)
- What is DevSecOps?
- Typical Program Challenges and Contributions to Solutions from DSO
- Call to Action

BLUF

Your program is adopting new practices and tools to support improving your value delivery to stakeholders, especially the field! Today's discussion is about DevSecOps.

Biggest impact is on those who

- Develop software (contractor and govt/partners)
- Integrate and Test software (contractor and govt/partners
- Engineer systems that contain software (govt/partners and contractor)
- Integrate and Test systems that contain software (govt/partners and contractor)

BUT there are impacts to all government (at least) and partner roles supporting your program

- Contracting/Finance personnel
- Logisticians
- Training
- Configuration and other Board members
- End users/MajCom requirements providers
- Security and other Certification Entities

WE NEED EVERYONE'S SUPPORT TO SUCCEED!

What is DevOps?

redhat.

DevSecOps Defintion SEI Uses

SW-Centric Definition:

"A set of ***principles*** and ***practices*** emphasizing collaboration and communication between software development teams and IT operations staff along with acquirers, supplies and other stakeholders the life cycle of a software system"

Complex Cyberphysical System-Centric Definition (per SuZ):

“A set of principles and practices emphasizing collaboration and communication among staff from engineering, hw/software design, development, integration and test, acquisition, security, services, end users and any other stakeholders key to delivery of a software-intensive HW/SW system.”

DevOps Is an Extension of Agile Thinking

Agile

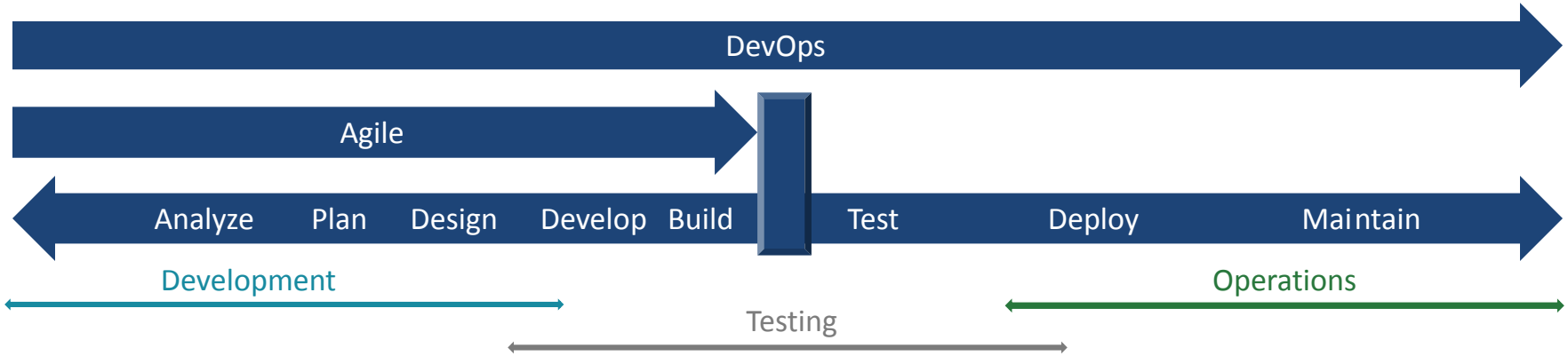
Embrace constant change

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



Four Principles of DevSecOps for Complex Cyberphysical Systems

Collaboration

- Among program team roles (government and contractors/suppliers)

Infrastructure as Code:

- The management of infrastructure (networks, virtual machines, load balancers, connection topology, and various environments) using the same versioning as source code

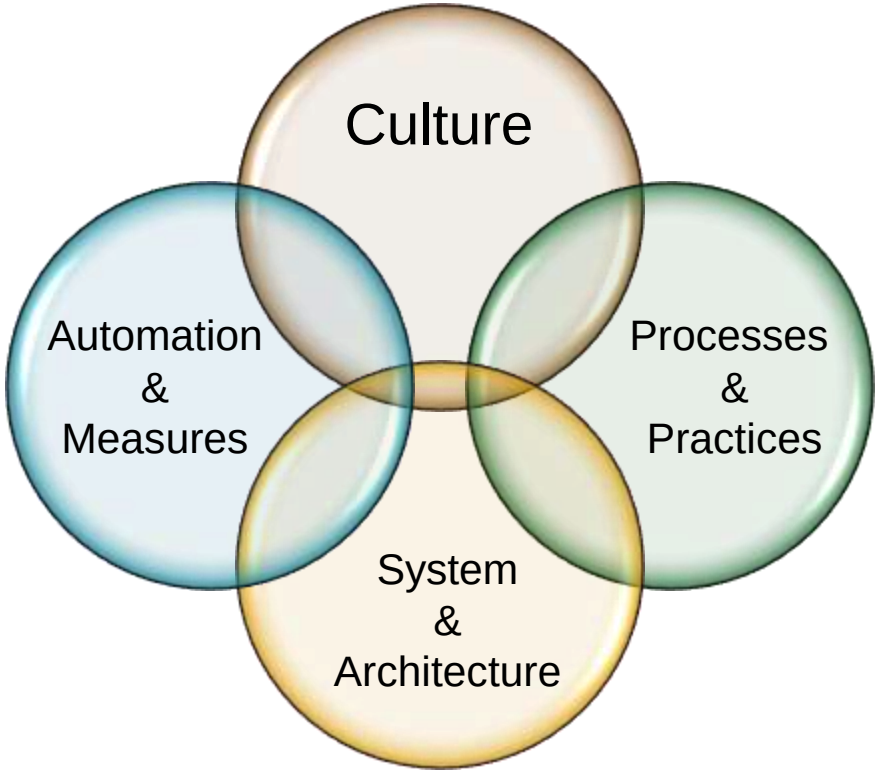
Automation:

- All manual, especially human-error-prone processes are automated

Monitoring:

- Any measurement in the engineering, development, deployment, or operational space that can inform priorities, direction, and policy is pulled from the value stream and used appropriately

Four Dimensions to Pay Attention To Implied by the Four Principles



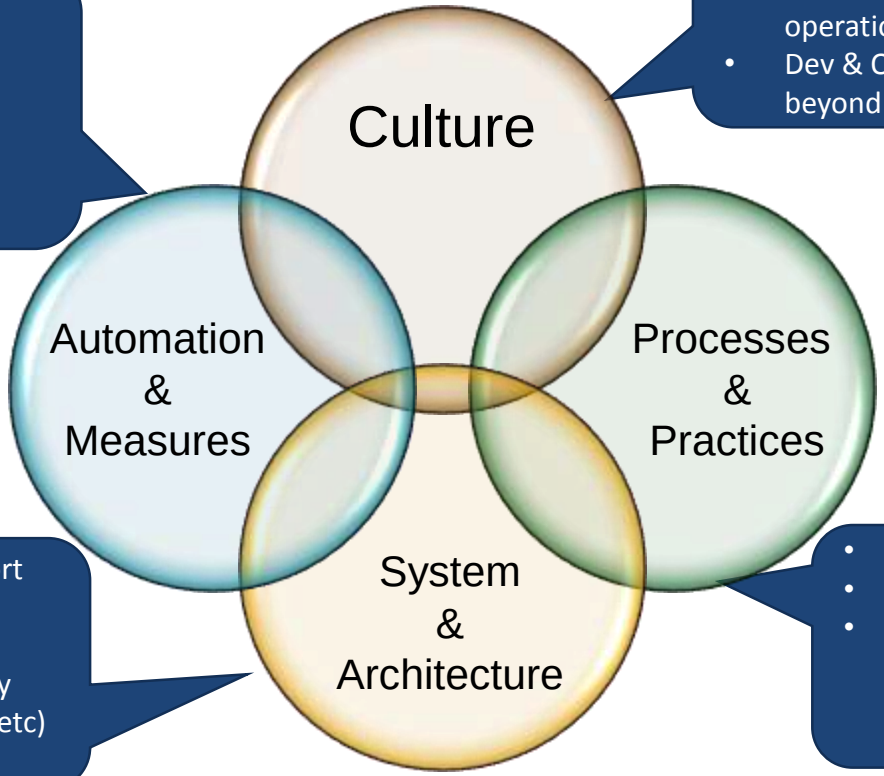
Discussion:

1) Which of these is most commonly associated with DevSecOps?
2) Which of these do you think is the most difficult to address in an organization like yours?

Might Seem Simple, but not EASY!

- What Some People Think Boundaries of DevSecOps is!
- Automate repetitive, error-prone tasks
- Static & Dynamic Systems Analysis
- Performance dashboards

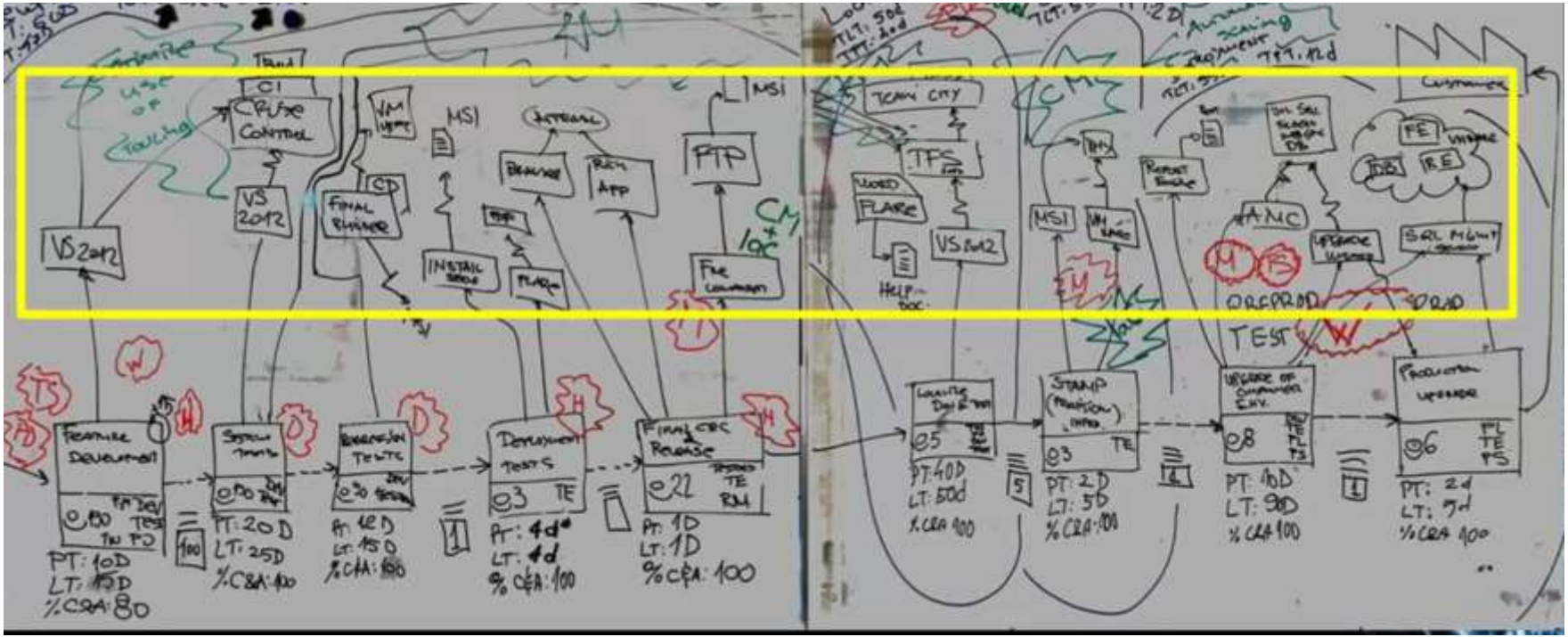
- All roles collaborate
- Dev, Ops, Sustainment have stakeholders that understand operational drivers
- Dev & Ops support products beyond delivery



- System architected to support integration and automation goals
- Represents important quality attributes (scalable, secure, etc)

- Value stream understanding
- Whole pipeline accounted for
- Continuous integration, automated test, virtualization, self-serve, scripting, automated deployment…

Understanding Your Value Stream is Key to Enabling Coherent DevSecOps Decisions



Source: https://www.youtube.com/watch?v=C9tfAE7ug8A

Value Stream Mapping is More than a Jira Workflow Diagram

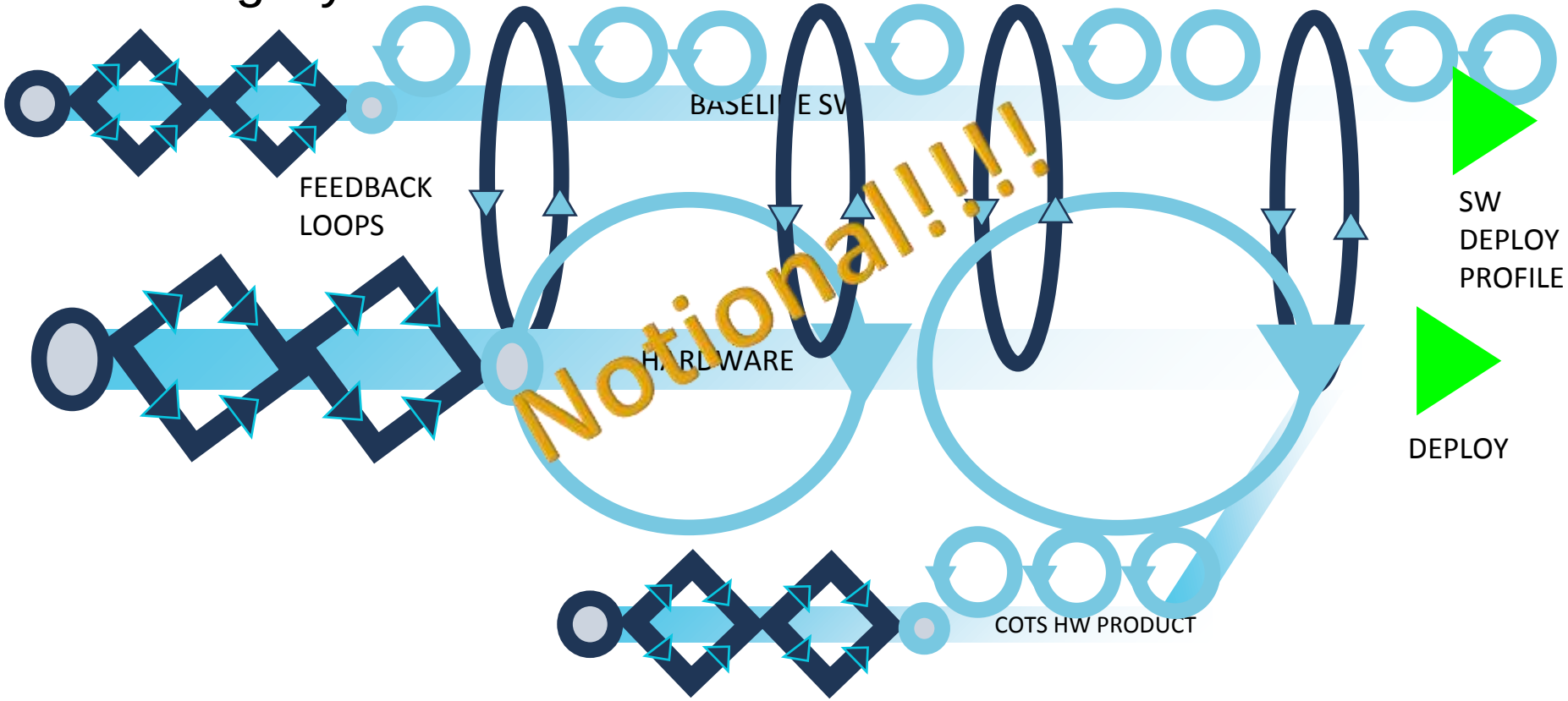
Webinars are popping up all over advertising “value stream mapping” to support DevOps transformation

- Many are focused only on identification of the workflow paths and integrations among tools (important, but NOT the defining aspect of VSM)
- Some are veiled advertisements for a particular workflow management tool

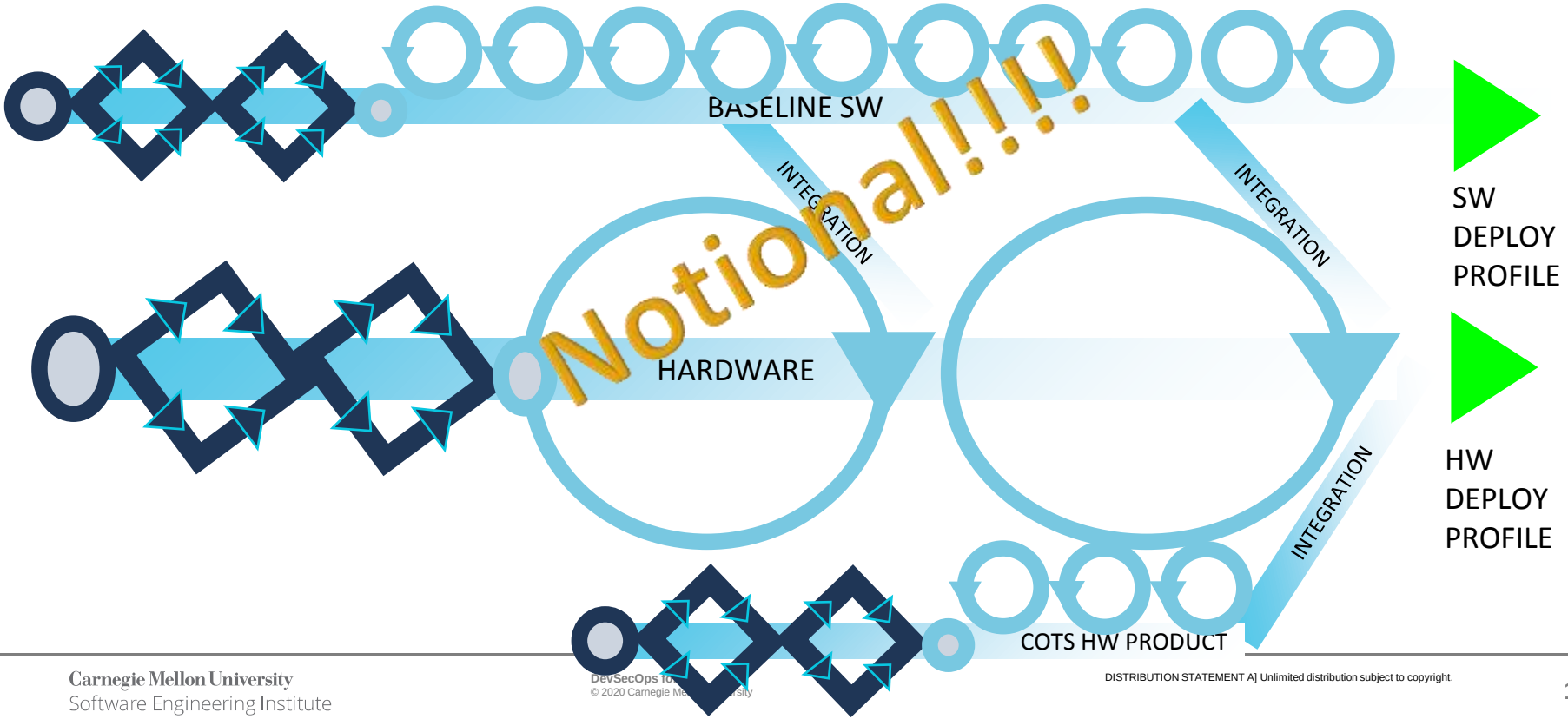
When done well, value stream mapping goes beyond workflow identification

- Identifies both “touch time” (the value added time that work towards the goal is being done) and “wait time” (the non-value added time taken up by the work waiting in various queues to be worked, reviewed, approved, sent on)
- Highlights the people and governance aspects of the journey from concept to capability, not just the tooling aspects
- Provides the basis for conversations about governance, mindset, and measurement that are difficult without data about “how work gets done”

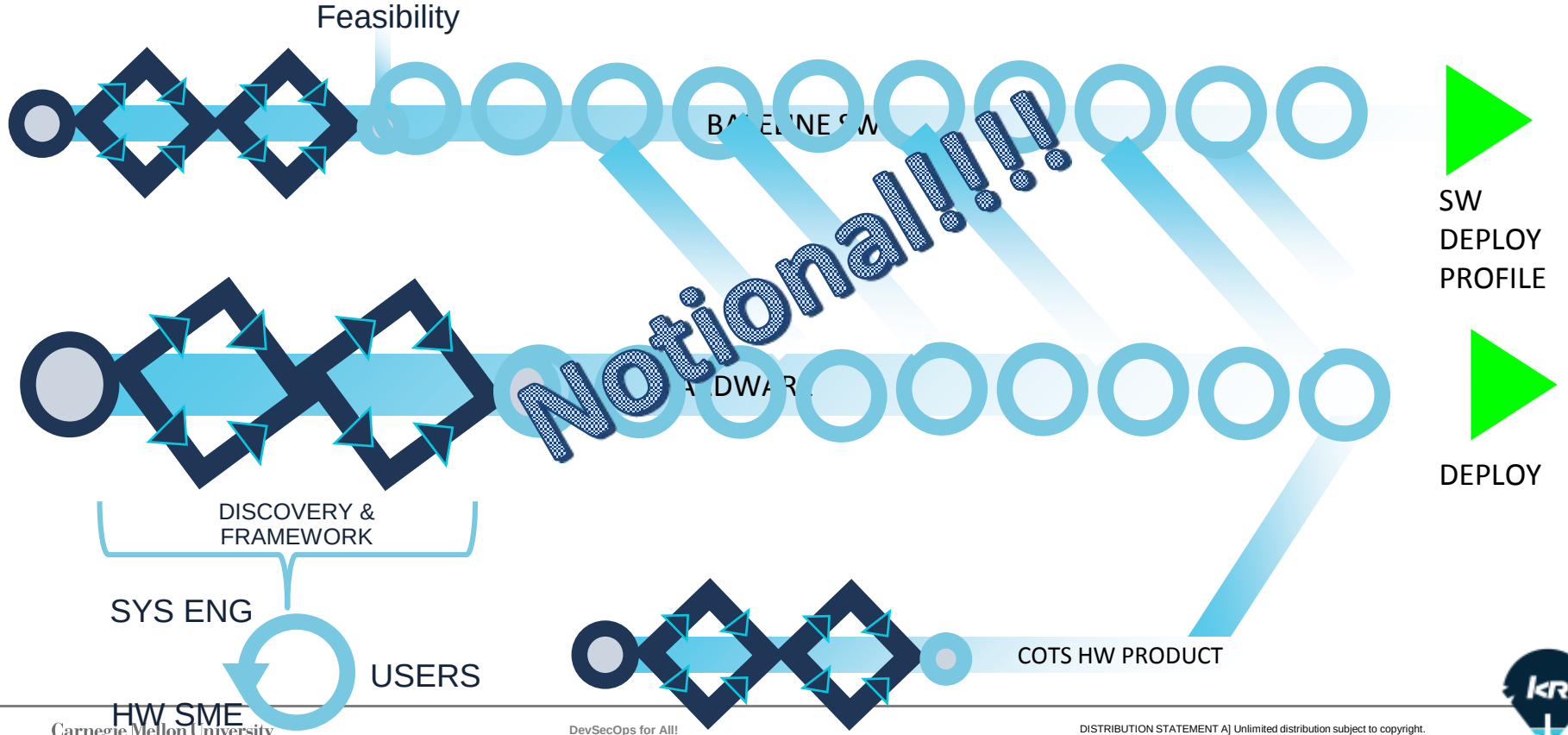
Foundation: Understand Value Stream to Delivery AND Learning Cycles Needed



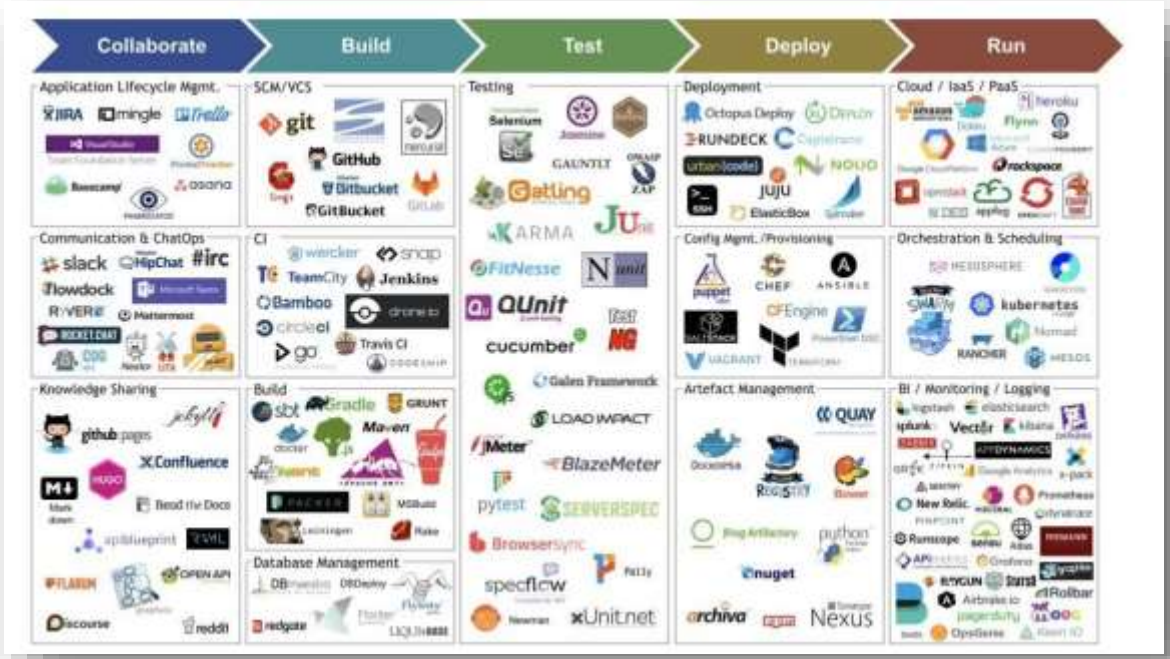
Learning Cycles are Driven by INTEGRATION – Lots and Frequent, Hardware and Software!



Implementation Execution Depends on Enough Systems Engineering to Evolve (Discovery & Framework)



Value Stream + *Architecture* + *Process*+ *Culture* = **Requirements** for an *Automated* DevSecOps Pipeline



The DevSecOps technology stack is *NOT* where we start

- Automating processes that don't add to or lead to value is more waste

Typical Program Challenges and Contributions to Solutions from DSO

(Some of the) Problems We Hear About on Large, Complex Programs

Lack of alignment among stakeholders on practices used to engineer, develop, integrate, test, certify

Lack of alignment among stakeholders on tools used to engineer, develop, integrate test, certify

Lack of transparency – data, measures, decisions – among stakeholders

“Nothing is done until everything is done”—large batch processes and mindset

Delays due to governance cadence are routine

DevSecOps (DSO) Contribution to Solving Above Problems

Lack of alignment among stakeholders on practices used to engineer, develop, integrate, test, certify	DSO makes practices explicit for moving through value stream to delivery
Lack of alignment among stakeholders on tools used to engineer, develop, integrate test, certify	DSO uses a defined and agreed upon (by all stakeholders) set of tools to automate various aspects of value stream processes
Lack of transparency – data, measures, decisions – among stakeholders	DSO tools have the capability of enabling transparency, where participants choose
“Nothing is done until everything is done”—large batch processes and mindset	DSO automation enables small batches to flow through the value stream efficiently
Delays due to governance cadence are routine	DSO allows defined governance decisions to be automated based on explicit criteria

Some Terms May Not Be Familiar to You

Software delivery profile:

- A description of the characteristics of development, tooling, and deployment that determines the applicable practices and tooling for that software type

Progress metrics:

- In this context, we aren't just talking about programmatic progress metrics, we are focusing on *progress toward end-user-focused outcomes*

Continuous Integration/Continuous Deployment (CI/CD) Pipeline

- The practices, governance points, and tools that describe the automation of the value stream, including software design, development, integration, test, certification, and deployment for a particular context

Governance is Key to Successful DevSecOps

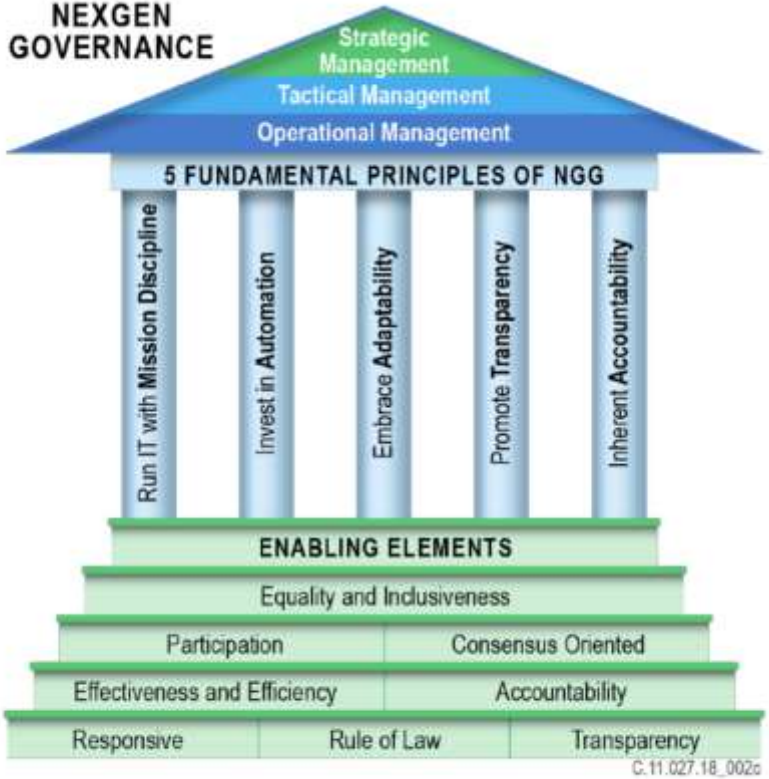
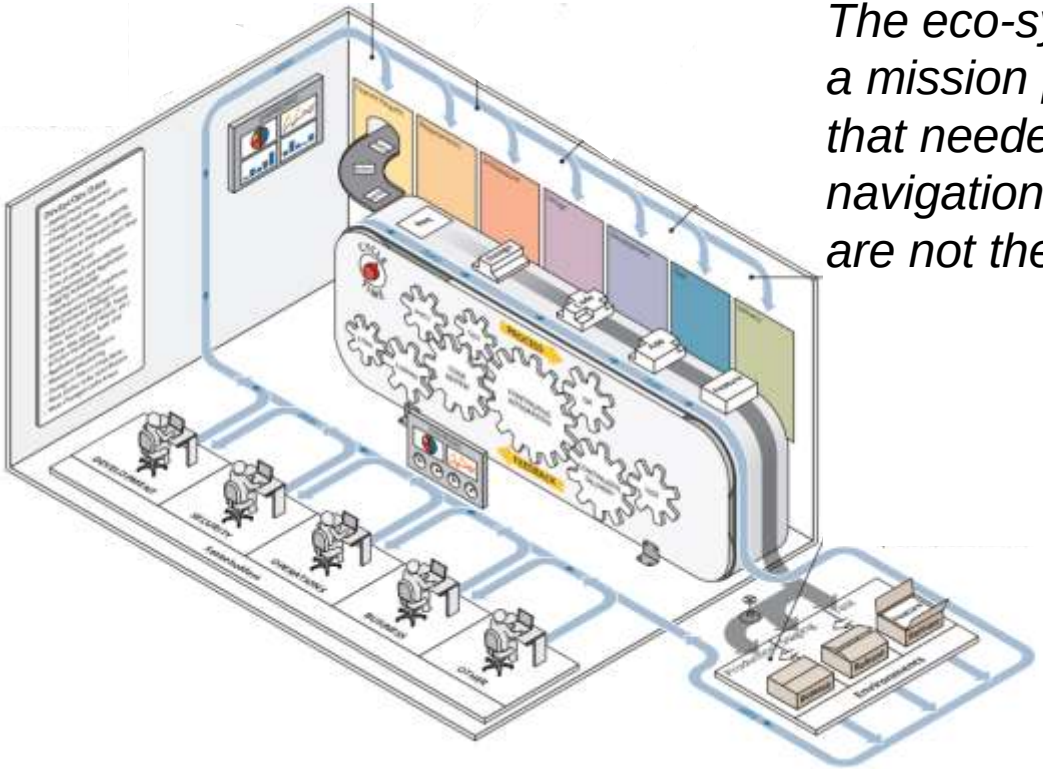


Figure 6: Five Principles of Next Generation Governance

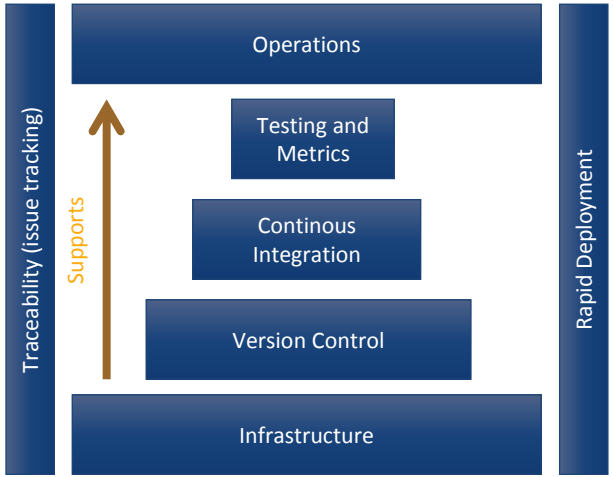
https://dodcio.defense.gov/Portals/0/Documents/DoD%20Enterprise%20DevSecOps%20Reference%20Design%20v1.0_Public%20Release.pdf?ver=2019-09-26-115824-583

Different SW Delivery Profiles Will Require Different (but related) DevSecOps Eco-System Elements



The eco-system needed for a mission planning system and that needed for an on-board navigation component are not the same

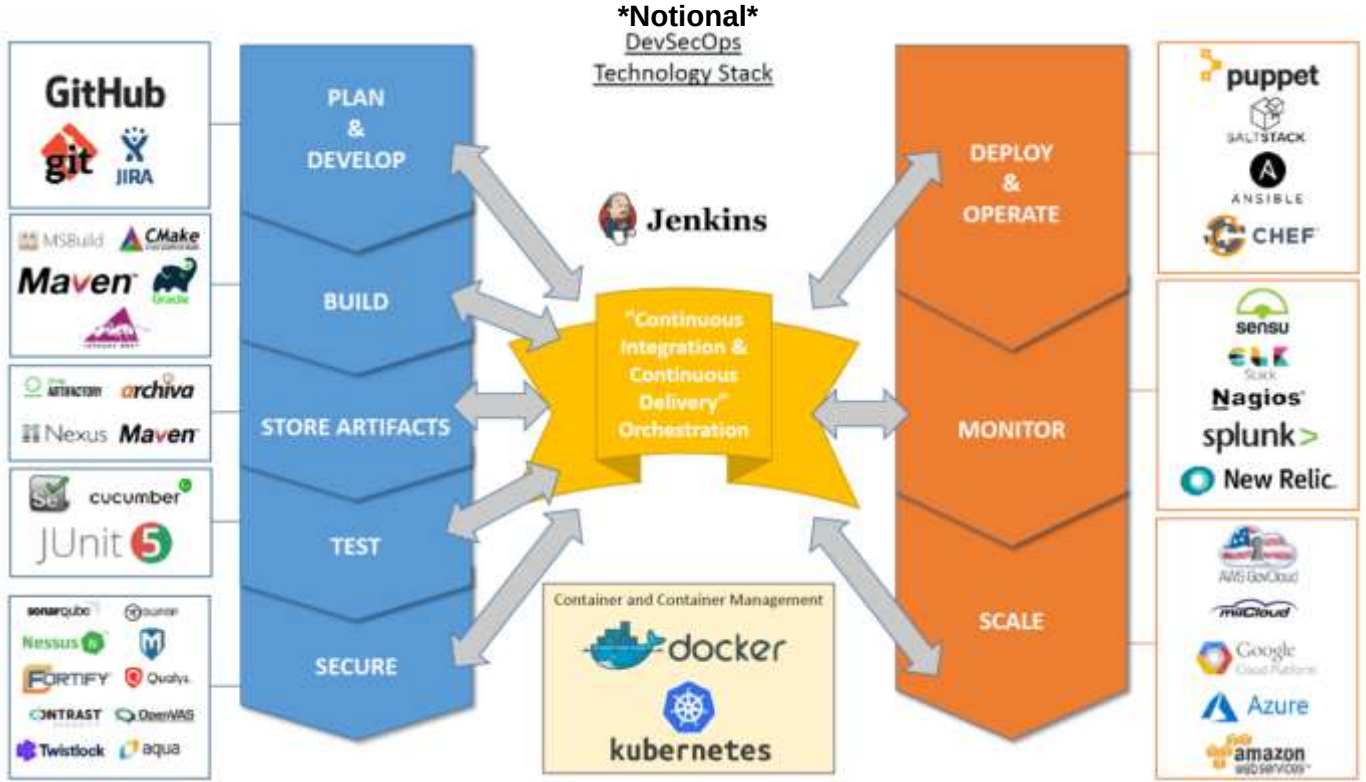
Pipeline Tool Landscape Complexity (~180 tools)



- Release configuration and release software (e.g., Puppet, Chef)
- Scripts and code used to release software (e.g., Python scripts)
- Servers, network or other infrastructure that support release tools
- Software and tools to support developer self-service operations
- External test frameworks (e.g., Jersey Test Framework)
- External operational monitoring and log mining tools (e.g., Splunk)
- Source code repositories (e.g., Git)
- Issue tracking systems (e.g., JIRA)
- Container driven tools (e.g., Docker)
- Rqmts mgmt. (Doors, Blueprint)
- Infrastructure and cloud providers
- IDEs integrated DevOps process

Modern software factory tooling makes it easier to move to a continuous development, test, and delivery model

The USAF CSO Office is Working with Programs to Identify the Subset of Useful Tools for Different Aspects of the Value Stream

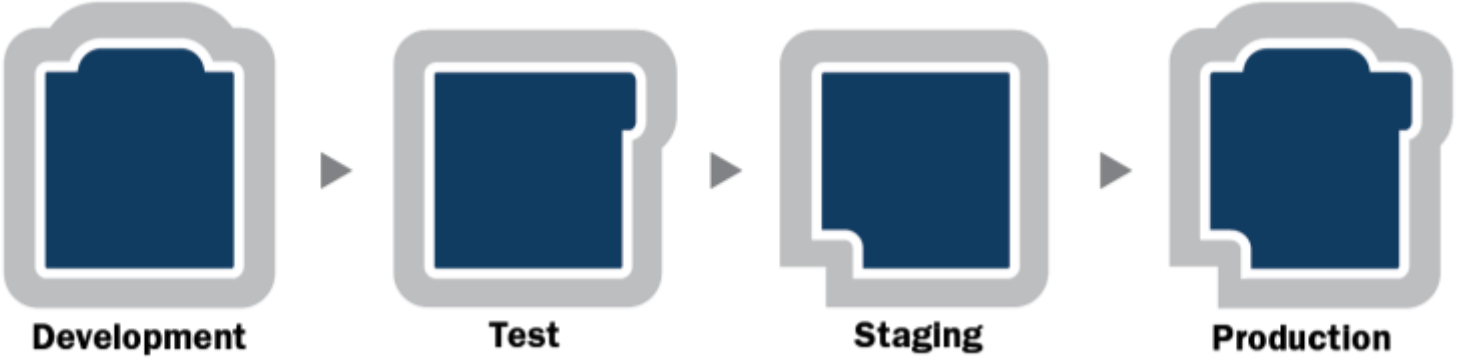


Source: https://www.cloudbees.com/blog/uniting-devops-and-security-devsecops-government

Engineering the Deployment Pipeline is a *challenge*

- If pipeline is not engineered, it may require extensive effort integrate tools and share data across the pipeline
- Key questions related to designing the integrated pipeline include:
  - Who owns the integrated deployment pipeline?
  - How/what to measure/monitor to assess pipeline health?
  - What are the key qualities attributes teams should look for as they select tools for pipeline integration?
- Whether designing or buying it is important to understand the end-to-end requirements (e.g., workflow visibility)

Divergence of Environments is Typical Starting Point



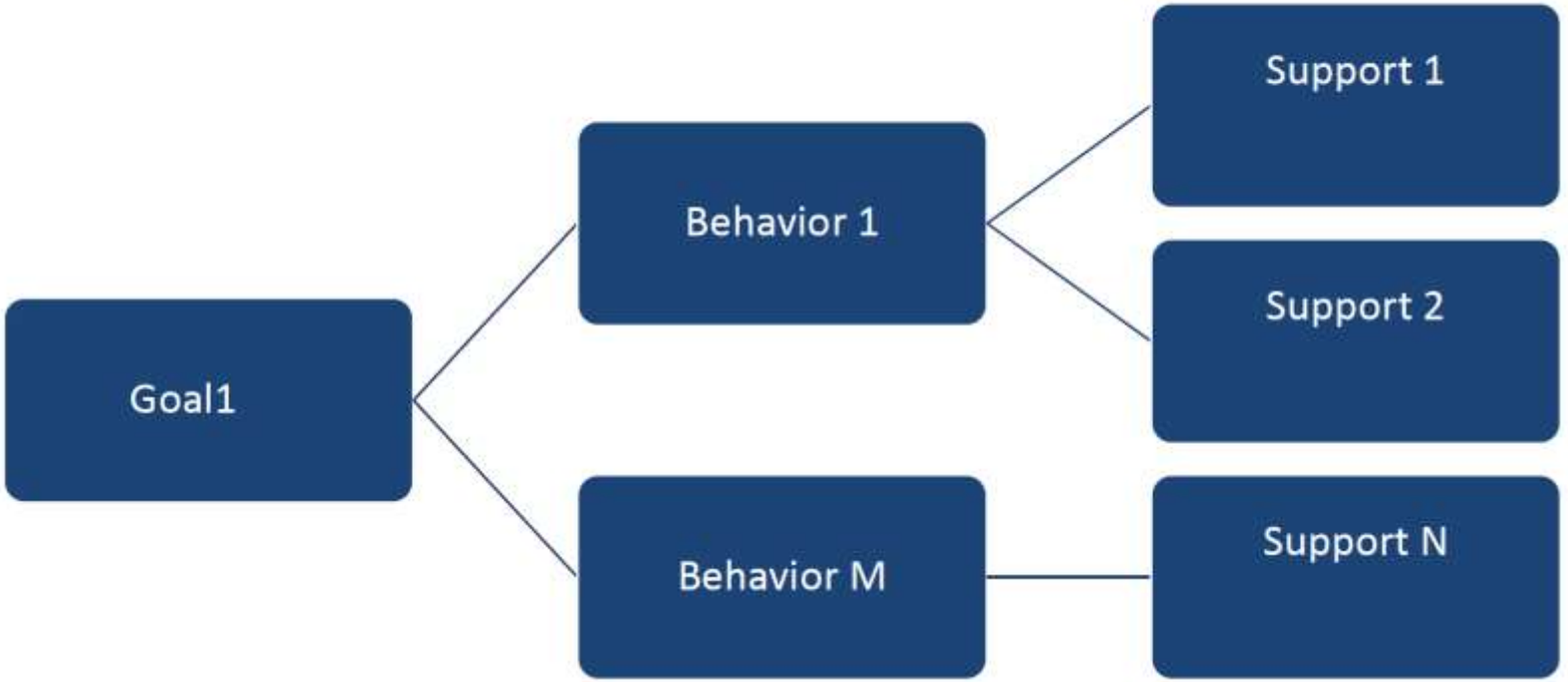
Environments are independent, volatile, and easily manipulated.

Without care, they will diverge.

This is one of the reasons that so much attention is focused on the technical stack.

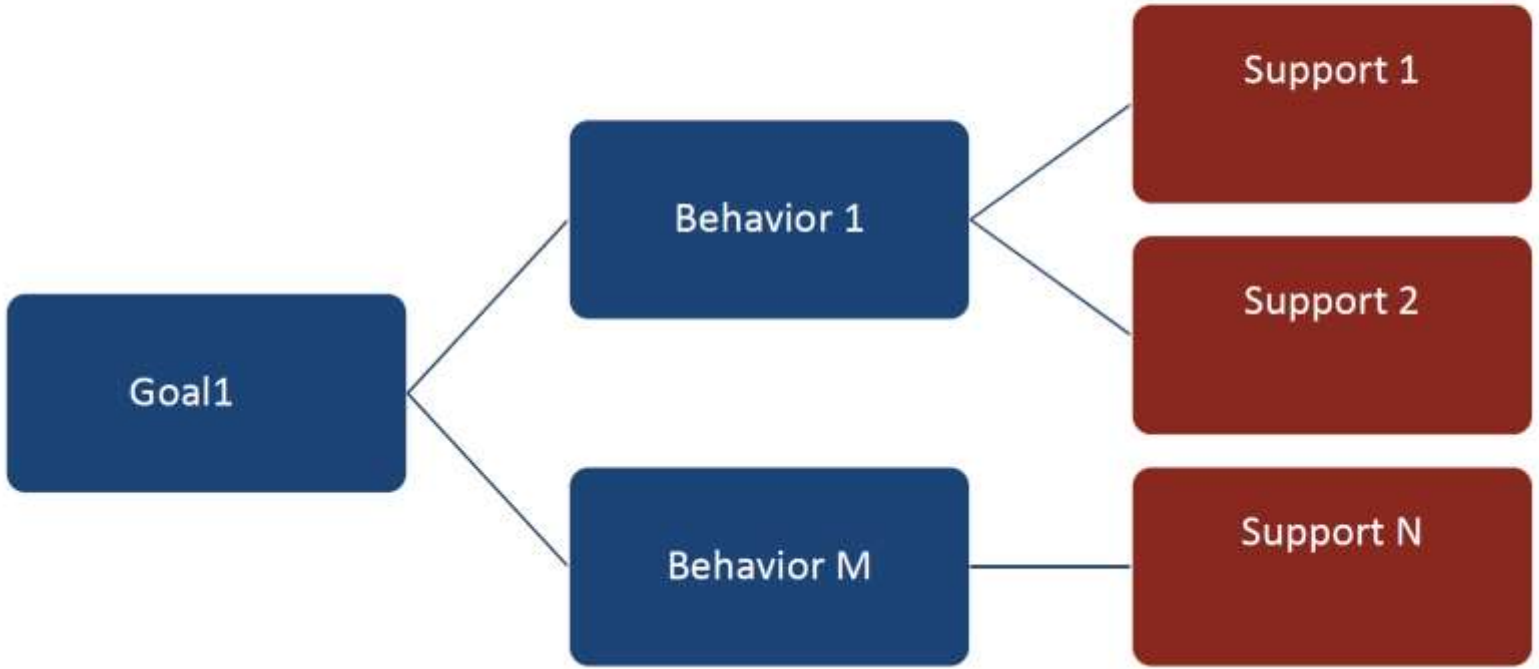
Getting from Divergence to DevSecOps

How Will You Get There From Here? Ladder Model of Behavior Change



Adapted from Ben Tiggelaar, Ladder Behavior Change Model

The “Support” Piece is What’s Often Left Out of Other Behavioral Change Models



Adapted from Ben Tiggelaar, Ladder Behavior Change Model

What Behaviors Reflect a DevSecOps Mindset Outside of Software Development?

Even if you never program a line of code, there are ways you can express a DSO mindset

- Identify your local value stream and how it contributes to overall operational delivery value
- Look for places to move from error-prone human-centric processes to streamlined workflows
  - Kanban, supported by DI2E Jira, is a common way for Program Office staff to make their work flow more effectively and transparently
  - Confluence wiki (also via DI2E) enables pull-based reporting (stakeholder ‘pulls’ info they need rather than things like PPT briefings being ‘pushed’ to them in meetings)

Look for places where governance can be simplified (either through automation or reworking of processes and stakeholders

What Supports Program Offices Typically Have to Help You Adopt New DSO-aware Behaviors?

Access to a Program DI2E site

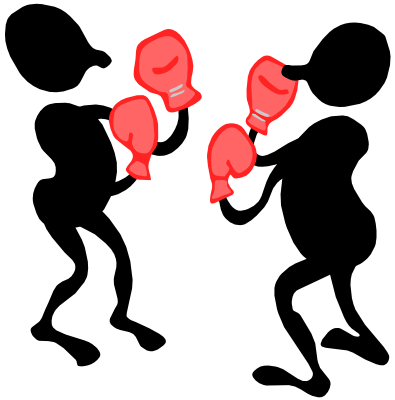
- Confluence Wiki and Jira workflow management are early opportunities to start collaborating
- Other tools on DI2E may be of use to particular stakeholders

Training (self-serve as well as in person) on mechanics of using DI2E tools

Workshops to help your group identify its Value Stream(s) and adopt appropriate team work practices (typically Kanban or Scrum)

Workshops to acquaint you with Program Office decisions related to DSO automation and opportunities for defining your own DSO pipeline

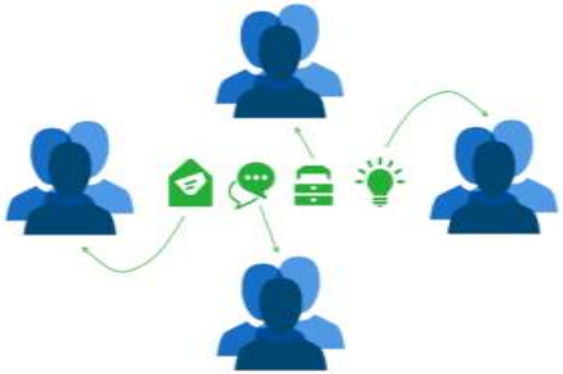
Some Other Supports Found to be Useful



- Blame-Free Culture
  - No Hiding of Problems
  - Culture of shared responsibility
  - Collective decision and continuous learning
- Cross-Silo Goals
  - Incentivize Collaboration
  - Reduce “Not My Job”
  - Increase Sense of Purpose
- Optimize Ease-of-Use
  - Tools: Chat, ChatOps, Wiki
  - Integrated Pipelines

Summary

DevSecOps: People



Heavy collaboration between all stakeholders

- Continuous secure design / architecture decisions
- Agreed-on environment / network configuration
- Continuous secure deployment planning
- Continuous secure code review

Constantly available, open communication channels:

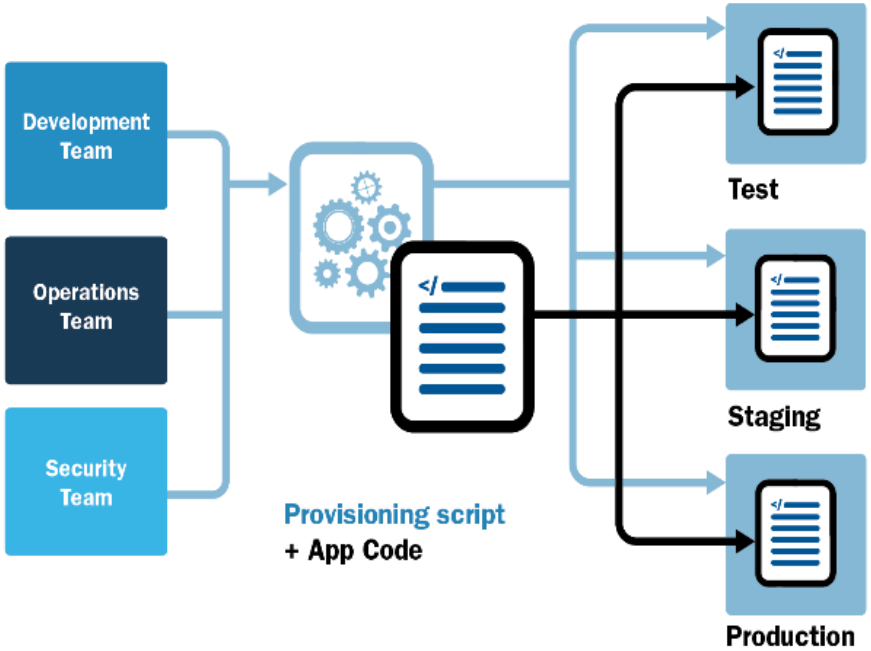
- Dev, Ops and Security together in all project decision meetings, virtually or physical but sharing a common collaboration environment
- Chat/email/Wiki services available to all team members

DevSecOps: Process

Establish a *process* to enable *people* to succeed using the *platform* to develop secure applications such that

- communication is constant and visible to all
- tasks are testable and repeatable
- human experts are free to do challenging, creative work
- tasks can be performed with minimal effort or cost
- teams have confidence in task success after past repetitions
- deployment is faster, and quality releases are more frequent

DevSecOps: Platform



Where **people** use **process** to build software

- Automated secure environment creation and provisioning
- Automated secure infrastructure testing
- Parity between development, QA, staging, and production environments
- Sharing and versioning of environmental configurations
- Collaborative environment between all stakeholders

DevSecOps is Just One Piece of the Agile Acquisition Transformation Challenge

DevSecOps is probably the least familiar set of concepts and tools for stakeholders who don’t develop software

- Don’t try and go it alone!

Work with your program office to adapt DSO to your context and to adapt your context to DSO!

Contact Information

Stephen Beck
SEI LRSO Technical Lead
SSD/TSAPP
Email: srbeck@sei.cmu.edu

David Walbeck
SEI LRSO Team
SSD/TSAPP
Email: dtwalbeck@sei.cmu.edu

Suzanne Miller
Principal Researcher
SSD/CDC
Email: smg@sei.cmu.edu

U.S. Mail
Software Engineering Institute
Customer Relations
4500 Fifth Avenue
Pittsburgh, PA 15213-2612
USA

Customer Relations
Email: info@sei.cmu.edu
Telephone: +1 412-268-5800
Fax: +1 412-268-6257