CISQ

Consortium for Information & Software Quality ™

9th Annual Cyber Resilience Summit

Hosted by:
Dr. Bill Curtis
Mr. Luke McCormack

9TH CISQ CYBER RESILIENCE SUMMIT

CISQ

Consortium for Information & Software Quality ™

OCTOBER 12, 2021

HOSTS:

BILL CURTIS
EXECUTIVE DIRECTOR
CONSORTIUM FOR INFORMATION & SOFTWARE QUALITY

LUKE MCCORMACK
FORMER CIO
DEPARTMENT OF HOMELAND SECURITY

What Is CISQ ?

CISQ is chartered to specify measures of software size and quality that can be automated from source code, and promote them through OMG and other international standards organizations

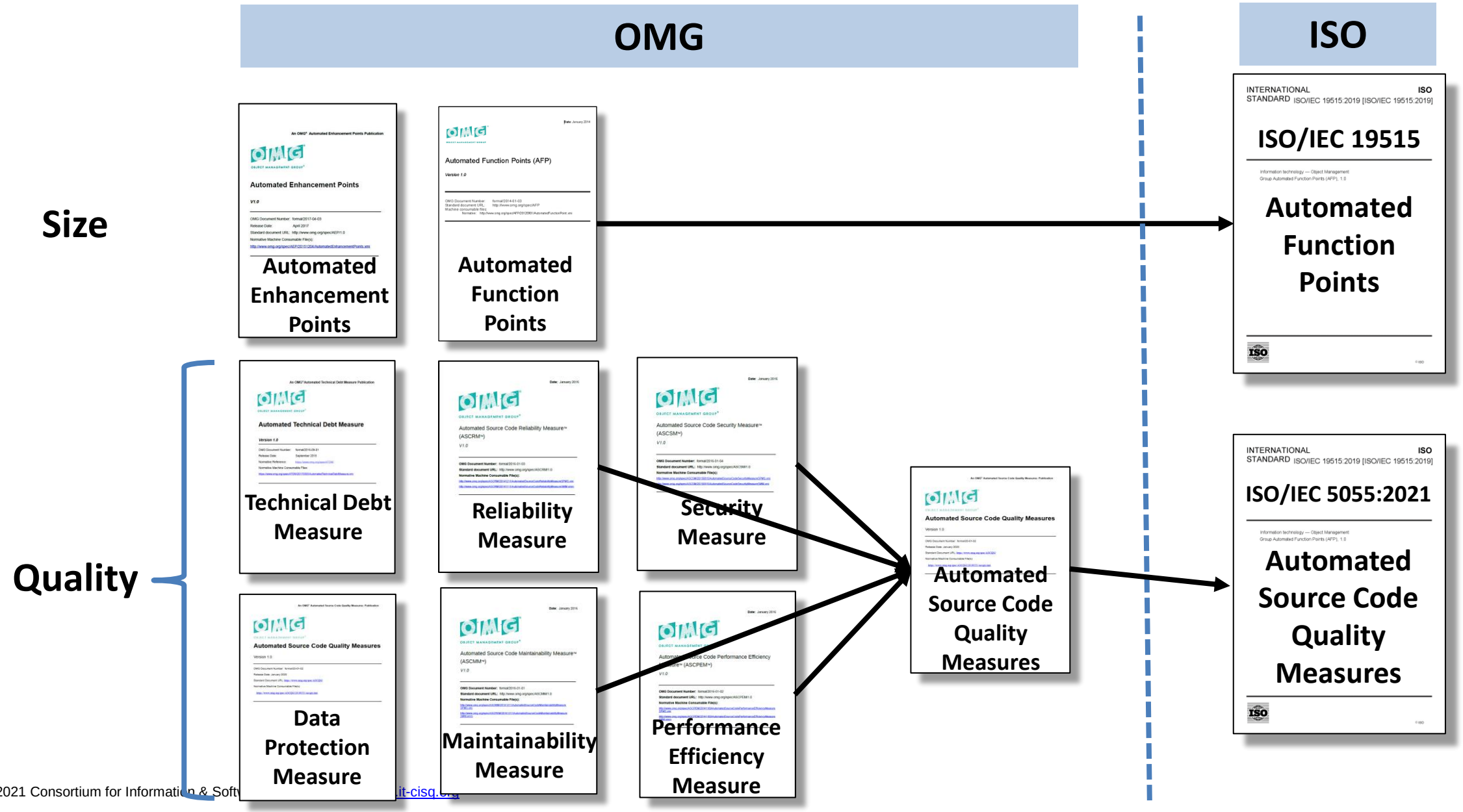
CISQ Sponsors

Tech Mahindra
SYNOPSYS®
7N
CAST
USC CSSE
SHPI
NORTHROP GRUMMAN
CGI

CISQ Partners

Gartner
QuEST FORUM
FORRESTER®
IAOP
USCC
TECHWELL
ITAAC
MAKING SOFTWARE BETTER
WCEA WASHINGTON, DC CHAPTER
ISG
QA

CISQ's History of Software Standards



CISQ's Future Standards

- Software Bill of Materials (SBOM)
- Automated Source Code Data Protection Measure
- Flow and Modernization Measures for Agile/DevOps Environments
- Updated Automated Technical Debt Measure
- Process Maturity Metamodel
- Dependable Programmer Certification

www.it-cisq.org — Membership Is Free

- **Almost 4000 individual members from Fortune 1000 organizations**
- **Contents:**
  - **Approved standards**
  - **Contract language**
  - **Trustworthy Systems Manifesto**
  - **Presentations**
  - **Webinars**
  - **Tutorials**
  - **Whitepapers**
  - **Use Cases**
  - **Blogs**
  - **News**
  - **Current standards projects**
  - **Process Maturity Metamodel**
  - **Upcoming events**
- **Cyber Resilience Summits**

KEYNOTE:
IS TECHNOLOGY THE SOLUTION OR PART OF THE PROBLEM? TECHNICAL DECISION POINTS ON THE JOURNEY TO RESPONSIBLE COMPUTING

CISQ

Consortium for Information & Software Quality™

PRESENTED BY:

MARC PETERS
DISTINGUISHED ENGINEER, CTO FOR ENERGY, ENVIRONMENT & UTILITIES EMEA, IBM

GAINING INSIGHT INTO CYBERSECURITY MATURITY

CISQ

Consortium for Information & Software Quality™

PRESENTED BY:

RON ZAHAVI
CHIEF STRATEGIST FOR IOT STANDARDS, MICROSOFT

MATTHEW JAMES BUTKOVIC
TECHNICAL DIRECTOR, SEI

SAMMY MIGUES
PRINCIPAL SCIENTIST, THE SYNOPSYS SOFTWARE INTEGRITY GROUP

CMMC AND SMM

PRESENTED BY: Ron Zahavi, Chief Strategist for IoT standards, Microsoft Azure IoT, SMM co-author
October 2021

CMMC and SMM

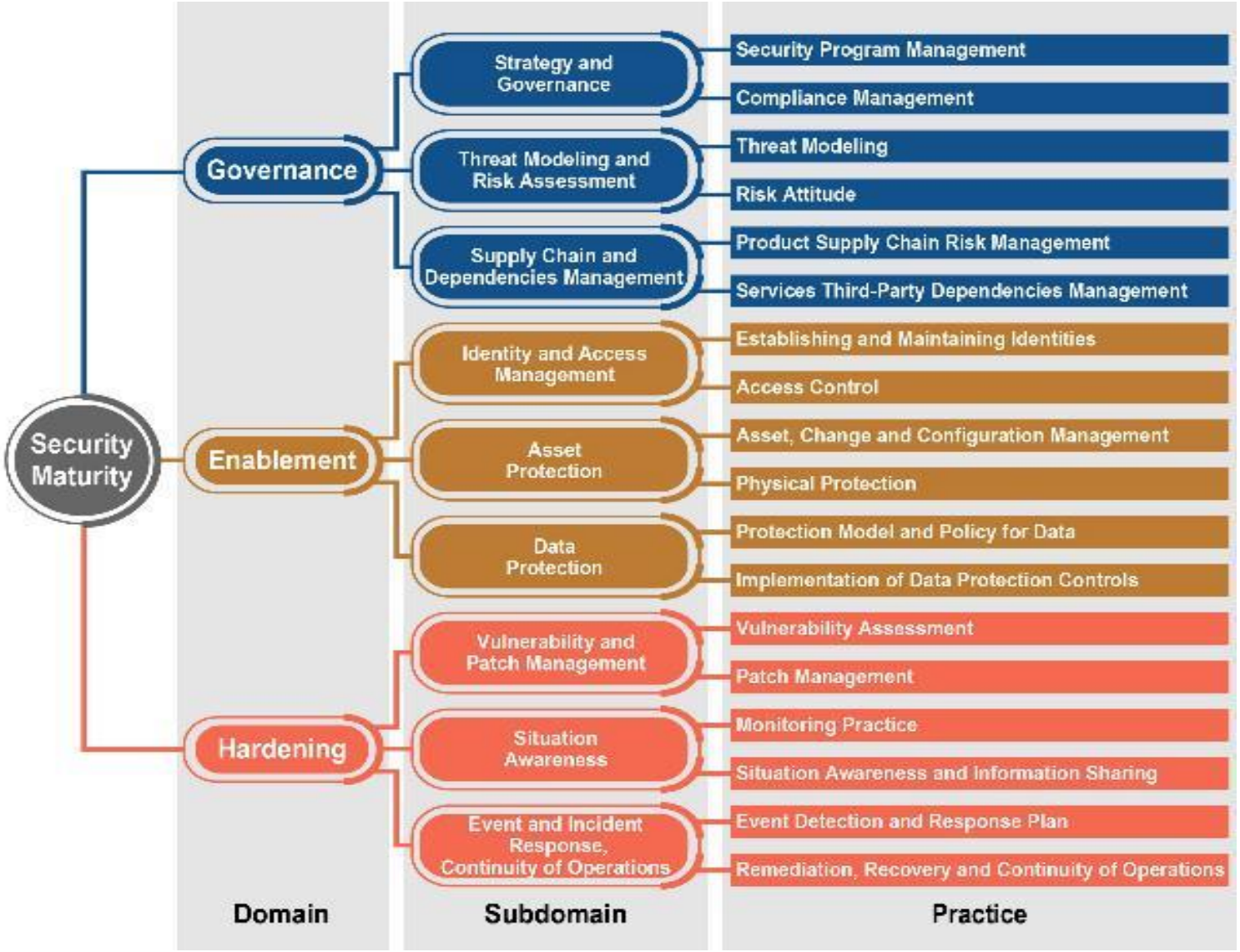
- What are they?
- What's similar?
- What's different?
- Complementary use

CMMC and SMM purpose

- CMMC: The **Cybersecurity Maturity Model Certification (CMMC)** is a new cybersecurity framework and accompanying certification by the US Department of Defense (DoD). The goal of the new CMMC compliance requirement is to protect Federal Contract Information (FCI) and Controlled Unclassified Information (CUI).

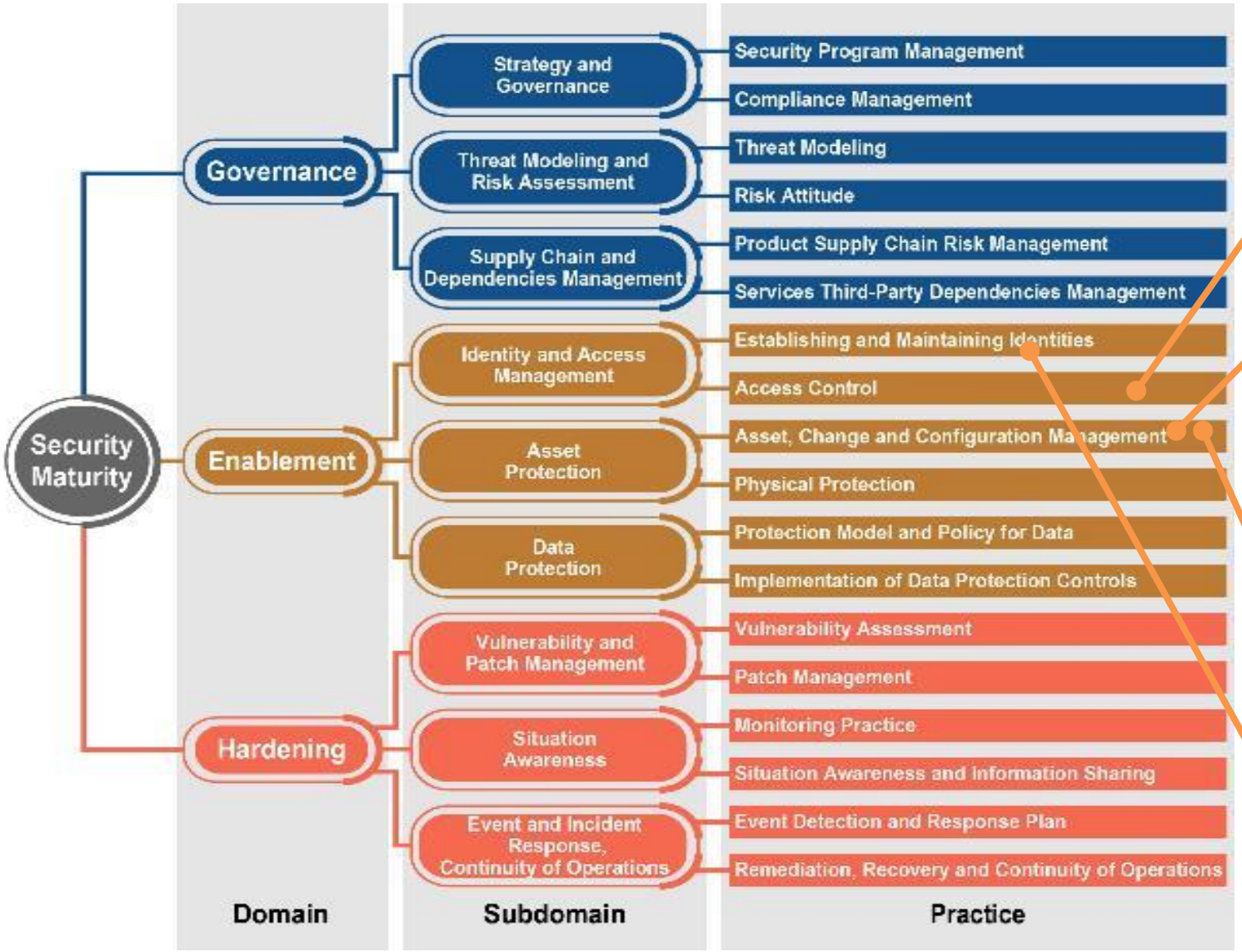
- SMM: The Internet of Things (IoT) **Security Maturity Model (SMM)** builds on the concepts identified in the Industrial Internet Security Framework (IISF) and provides a path for IoT providers to understand where they need to be, make intelligent choices about which mechanisms to use and how to invest in the mechanisms to meet their needs

Structure - CMMC Domains and SMM Practices



Access Control (AC)	Incident Response (IR)	Risk Management (RM)
Asset Management (AM)	Maintenance (MA)	Security Assessment (CA)
Awareness and Training (AT)	Media Protection (MP)	Situational Awareness (SA)
Audit and Accountability (AU)	Personnel Security (PS)	System and Communications Protection (SC)
Configuration Management (CM)	Physical Protection (PE)	System and Information Integrity (SI)
Identification and Authentication (IA)	Recovery (RE)	

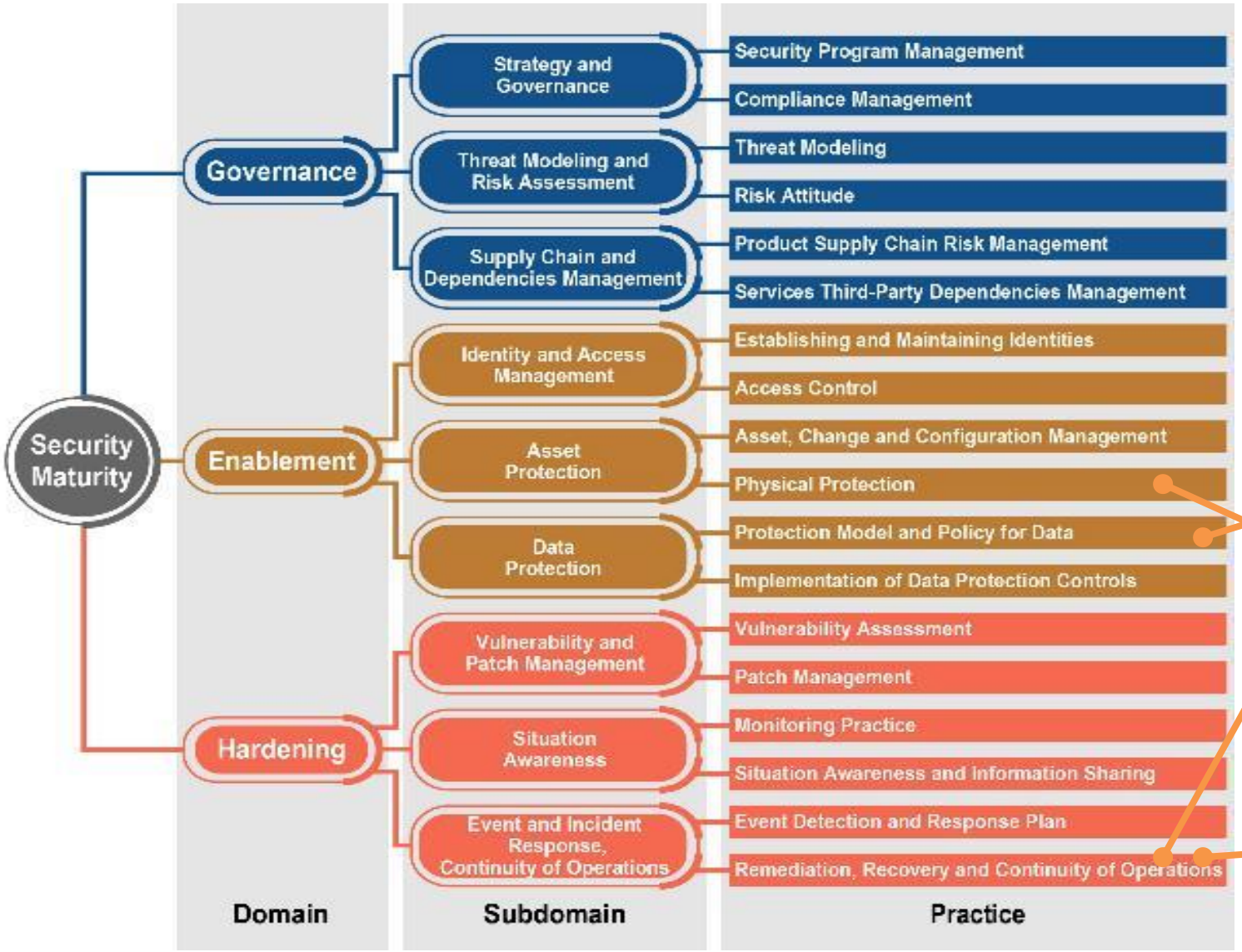
Structure - CMMC Domains and SMM Practices

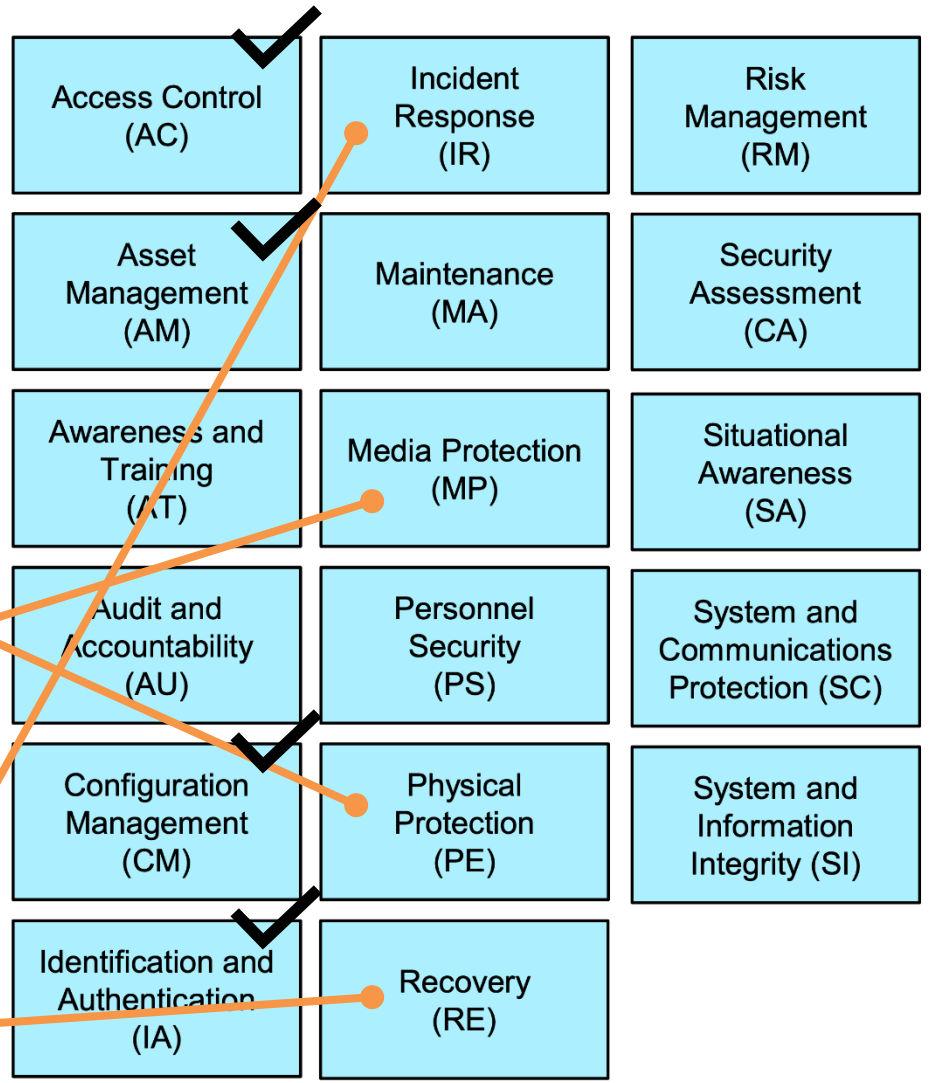


Access Control (AC)	Incident Response (IR)	Risk Management (RM)
Asset Management (AM)	Maintenance (MA)	Security Assessment (CA)
Awareness and Training (AT)	Media Protection (MP)	Situational Awareness (SA)
Audit and Accountability (AU)	Personnel Security (PS)	System and Communications Protection (SC)
Configuration Management (CM)	Physical Protection (PE)	System and Information Integrity (SI)
Identification and Authentication (IA)	Recovery (RE)	

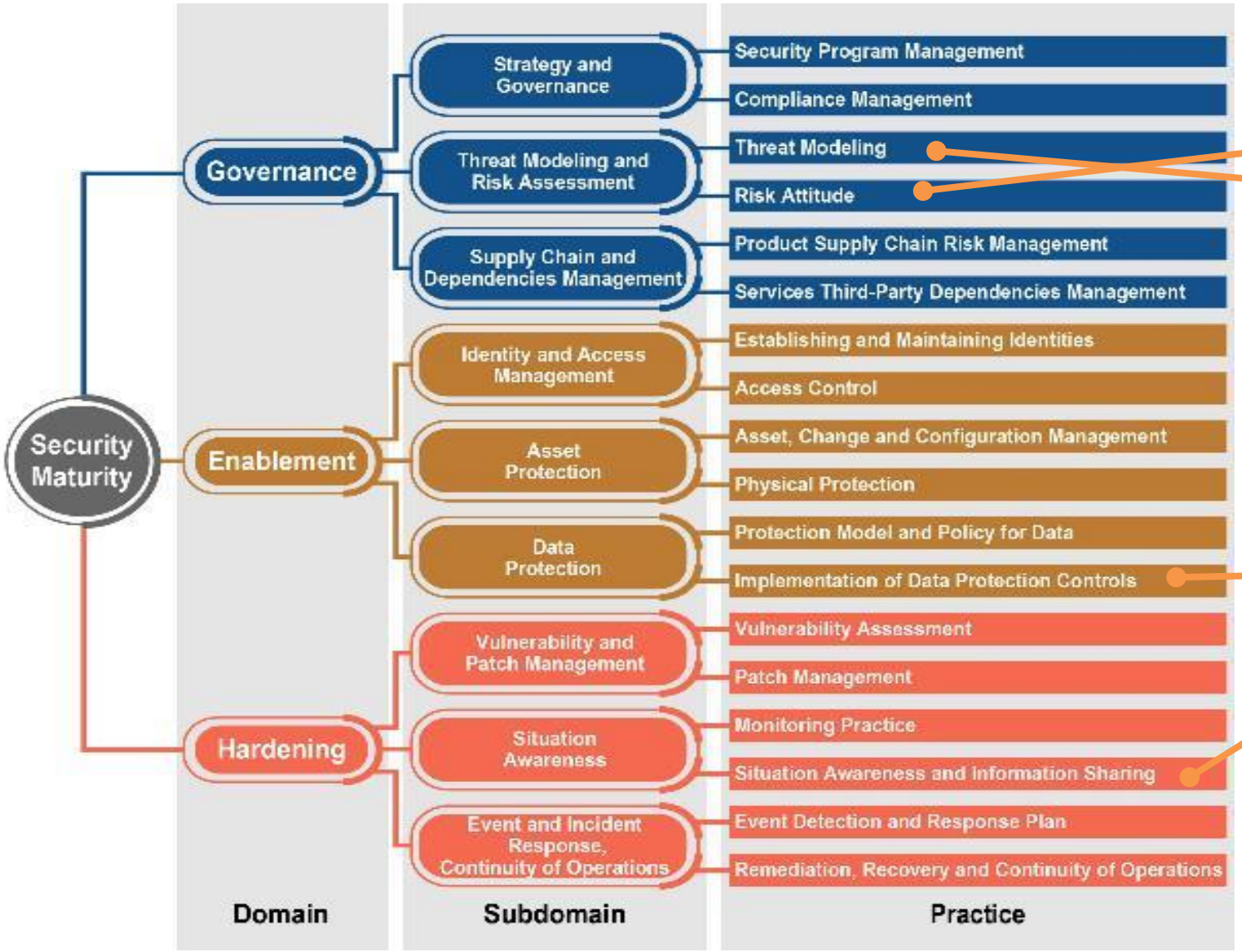
CISQ
Consortium for Information & Software Quality™

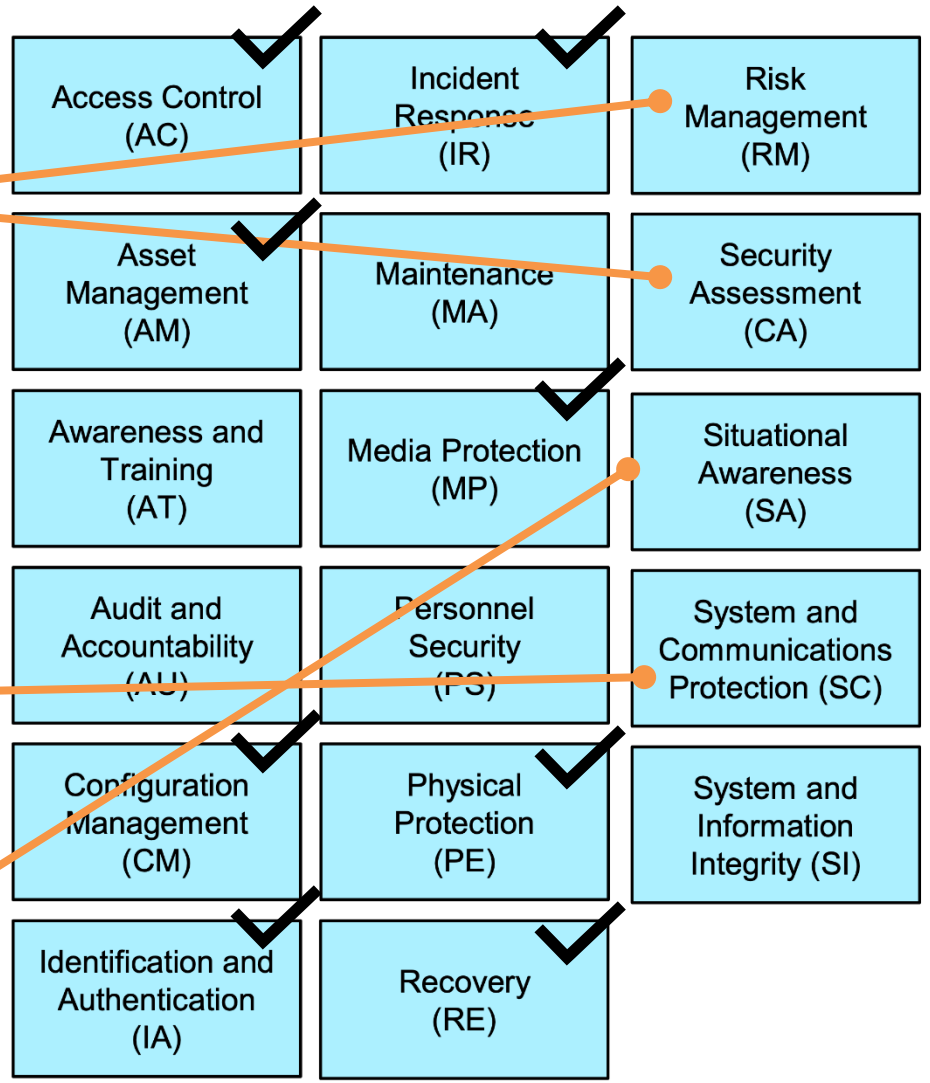
Structure - CMMC Domains and SMM Practices



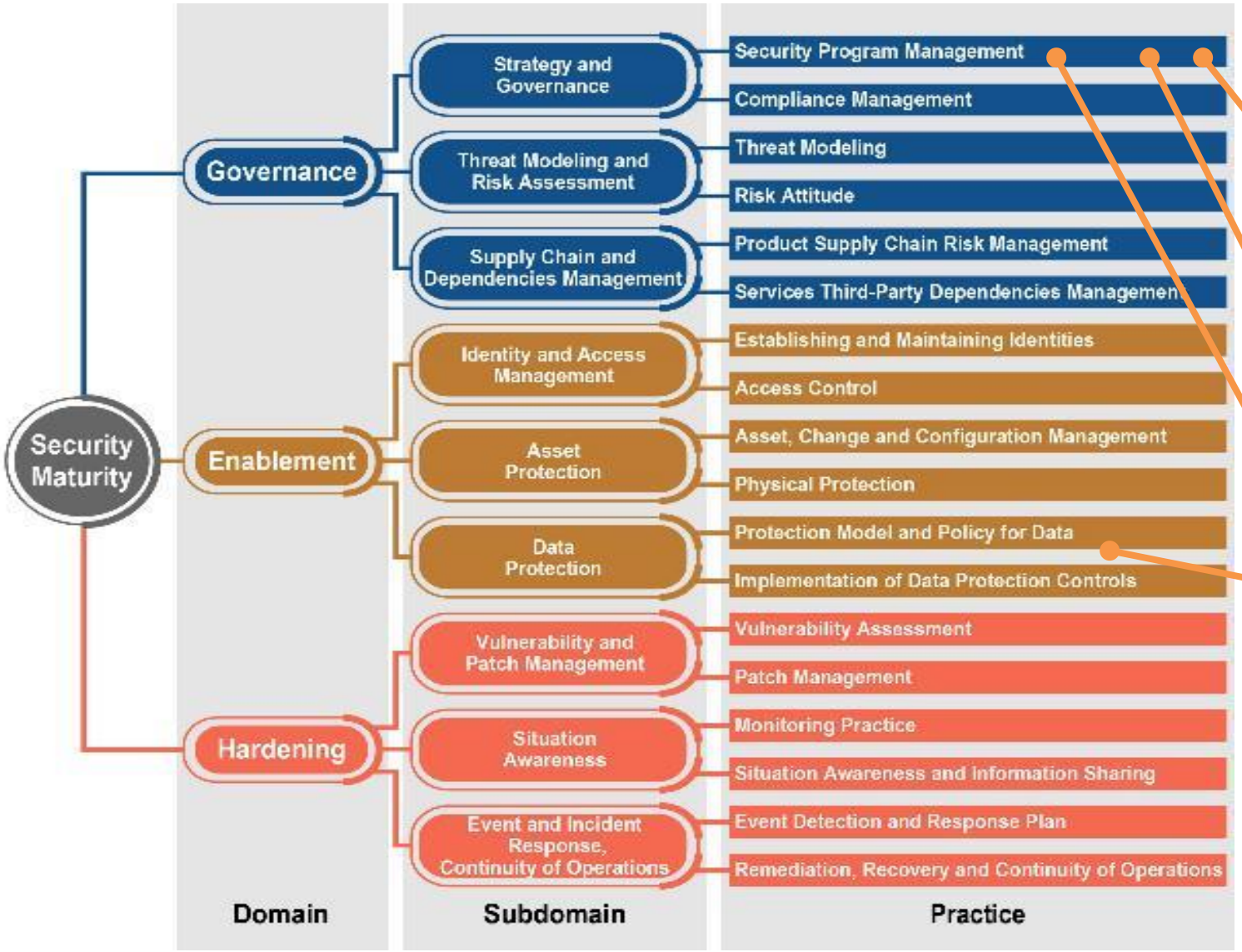


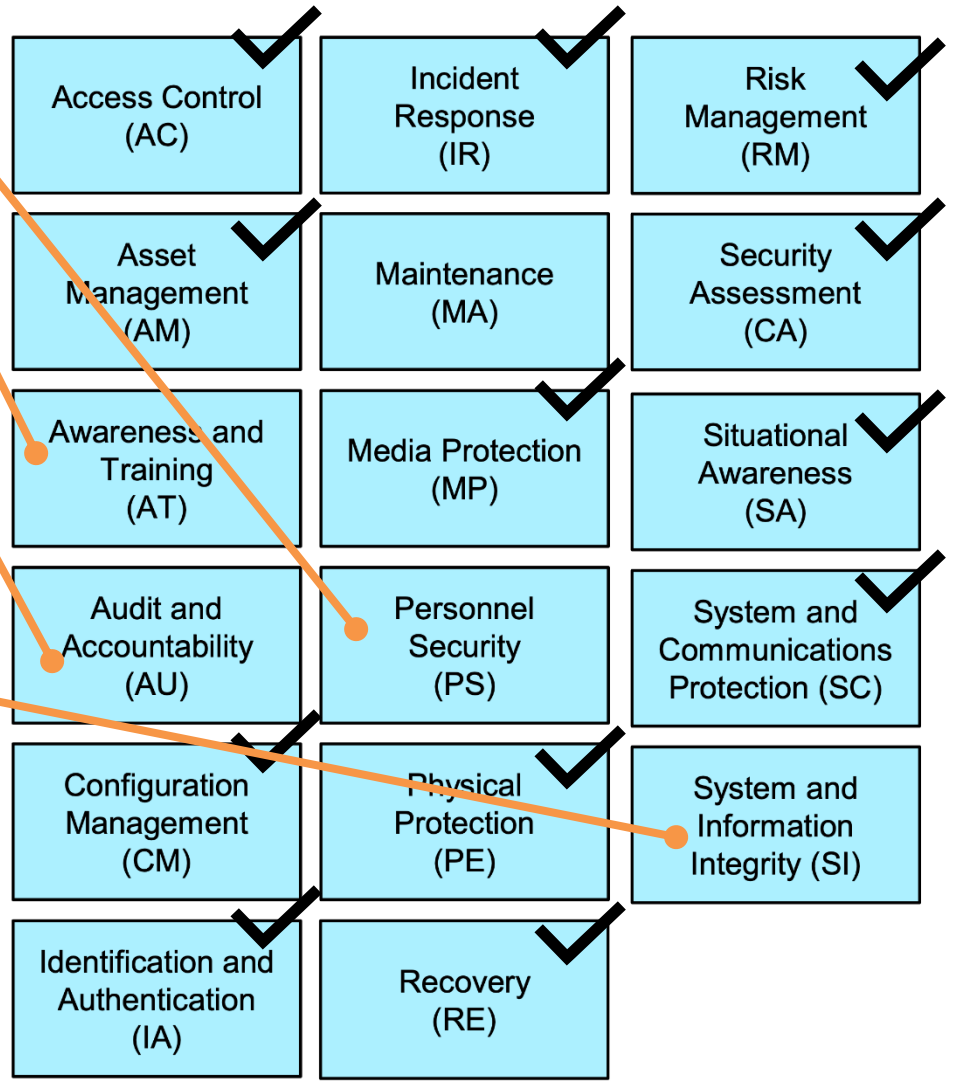
Similar topic areas



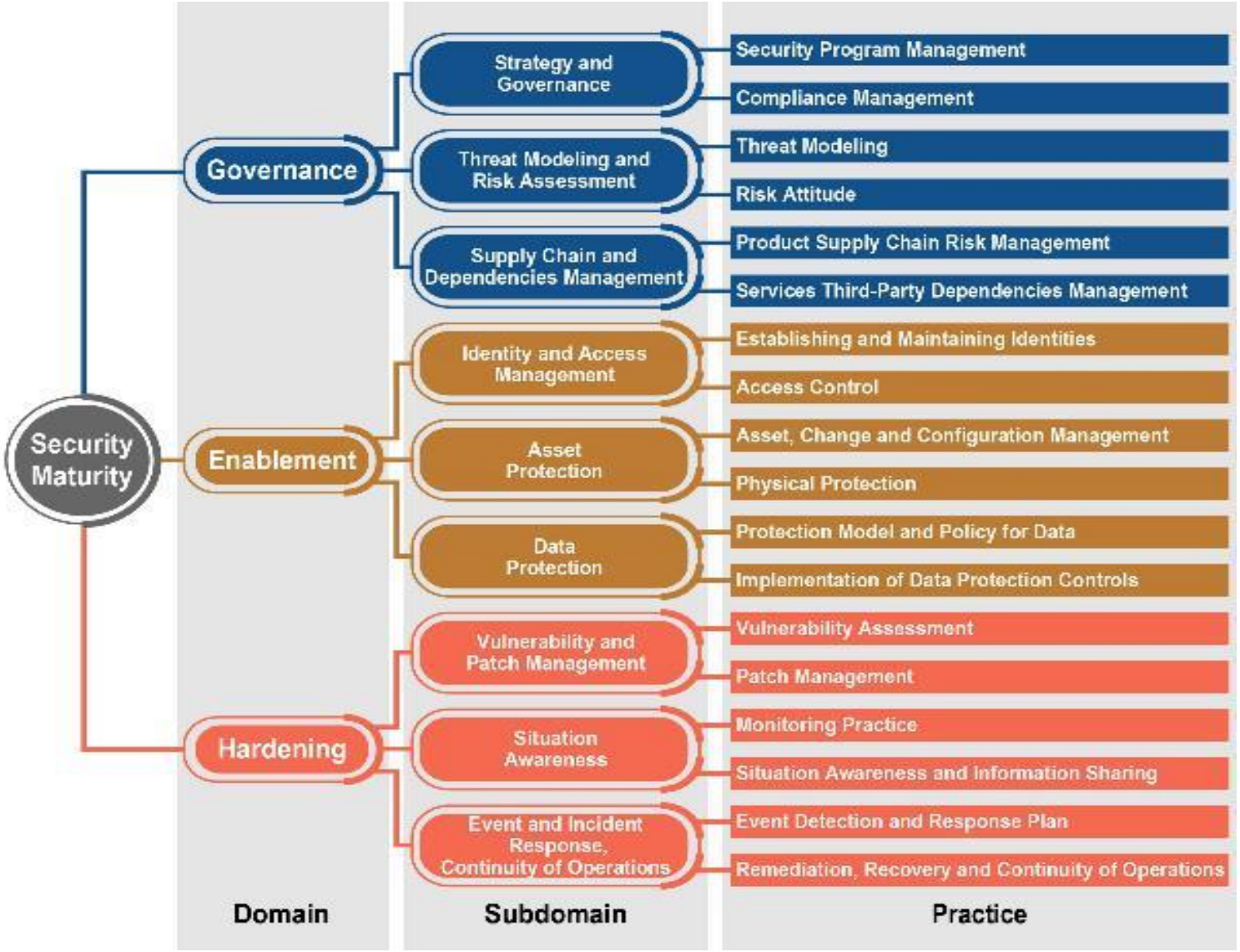


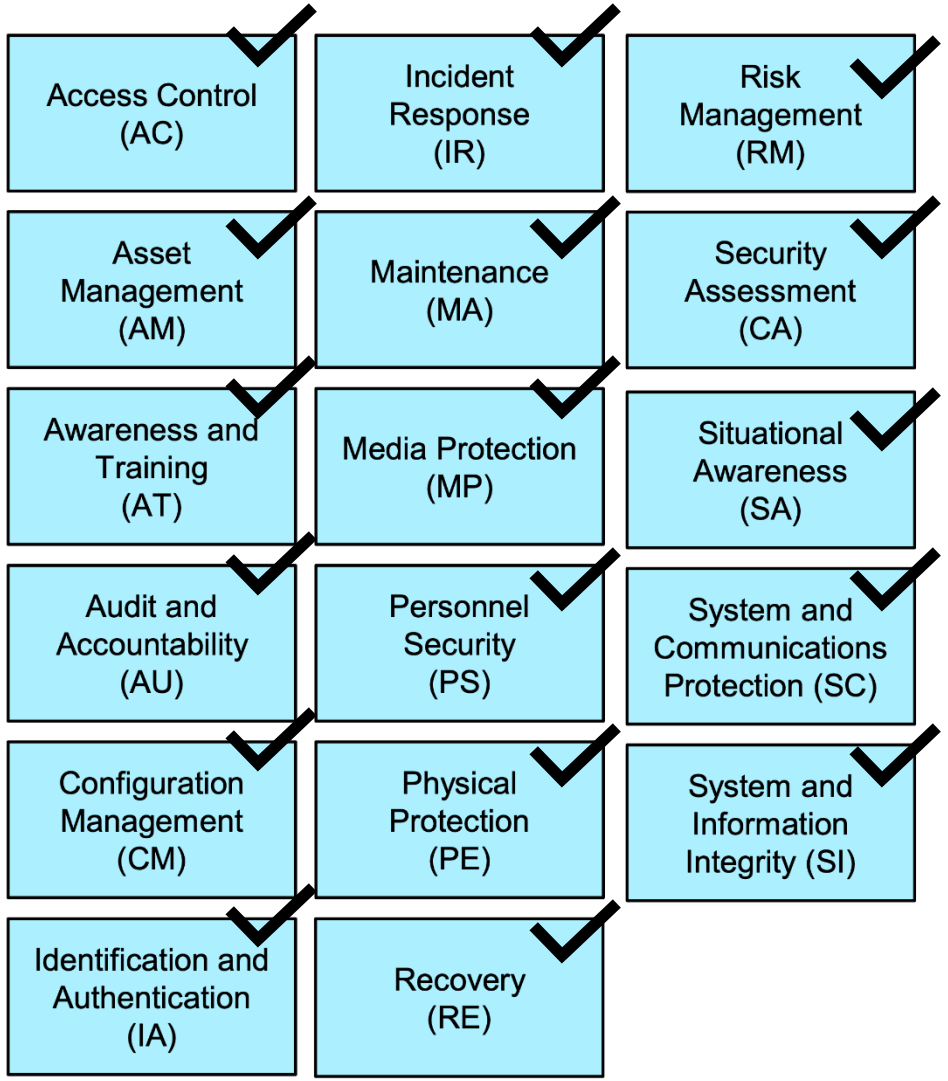
Similar topic areas





Similar topic areas



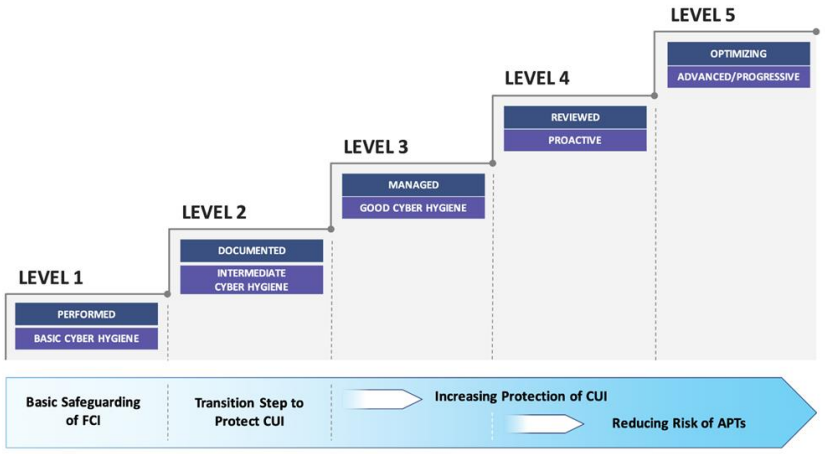


But they are different and have different context

CMMC

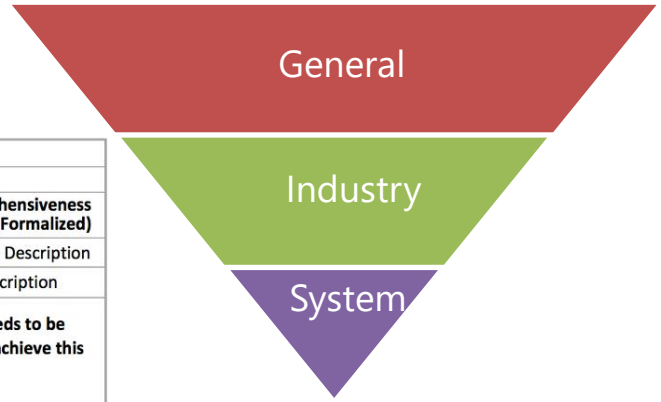
- US Government based
- Guidelines for protection of government IP by vendors
- Certification of compliance
- Improvement between levels, reaching a higher level

CMMC Model



SMM

- International
- For evaluating security of IoT solutions (sensors to the cloud including IT/OT/IoT)
- Certification of assessment companies that want to evaluate solutions
- Levels match need and investment goals, identify the right level
- Profiles and mappings



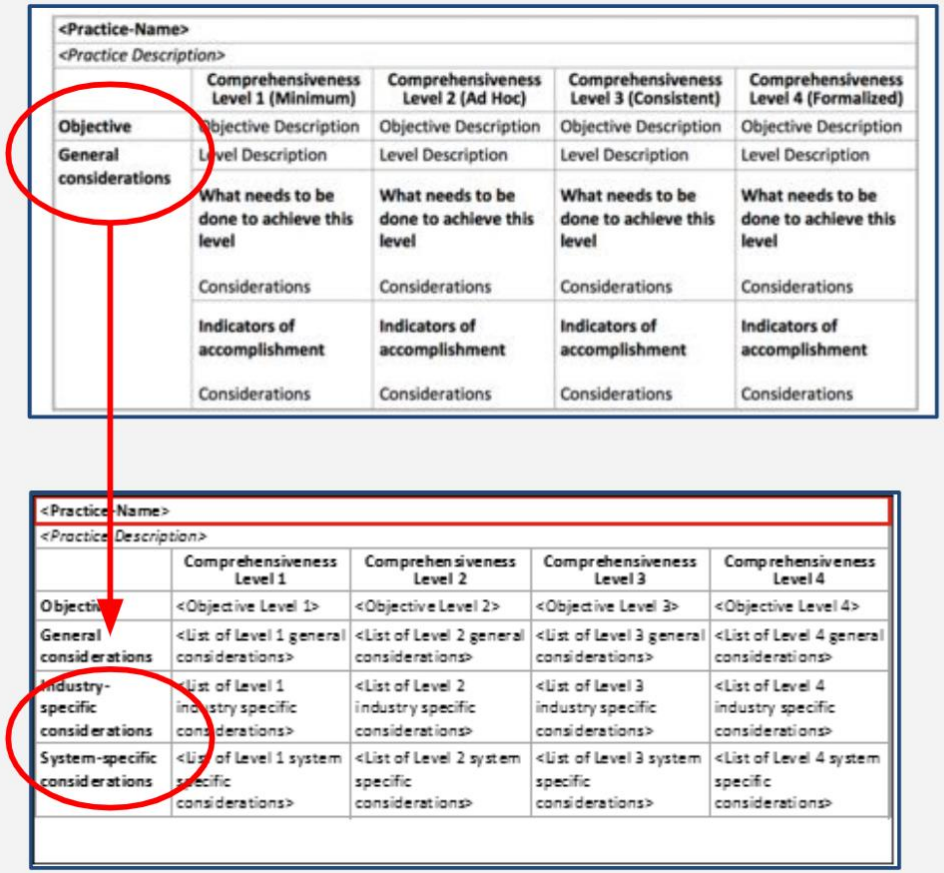
<Practice-Name>				
<Practice Description>				
	Comprehensiveness Level 1 (Minimum)	**Comprehensiveness Level 2 (Ad Hoc)**	**Comprehensiveness Level 3 (Consistent)**	**Comprehensiveness Level 4 (Formalized)**
Objective	Objective Description	Objective Description	Objective Description	Objective Description
General considerations	Level Description	Level Description	Level Description	Level Description
	What needs to be done to achieve this level Considerations	**What needs to be done to achieve this level** Considerations	**What needs to be done to achieve this level** Considerations	**What needs to be done to achieve this level** Considerations
	Indicators of accomplishment Considerations	**Indicators of accomplishment** Considerations	**Indicators of accomplishment** Considerations	**Indicators of accomplishment** Considerations

Profiles

- Extending the scope to create an industry profile

  - Make general considerations more specific
  - Add more detail
  - Provide guidance on “what needs to be done”
  - Provide industry specific detail on “indicators of accomplishment”

- System-specific guidance can also be added for a system or a device



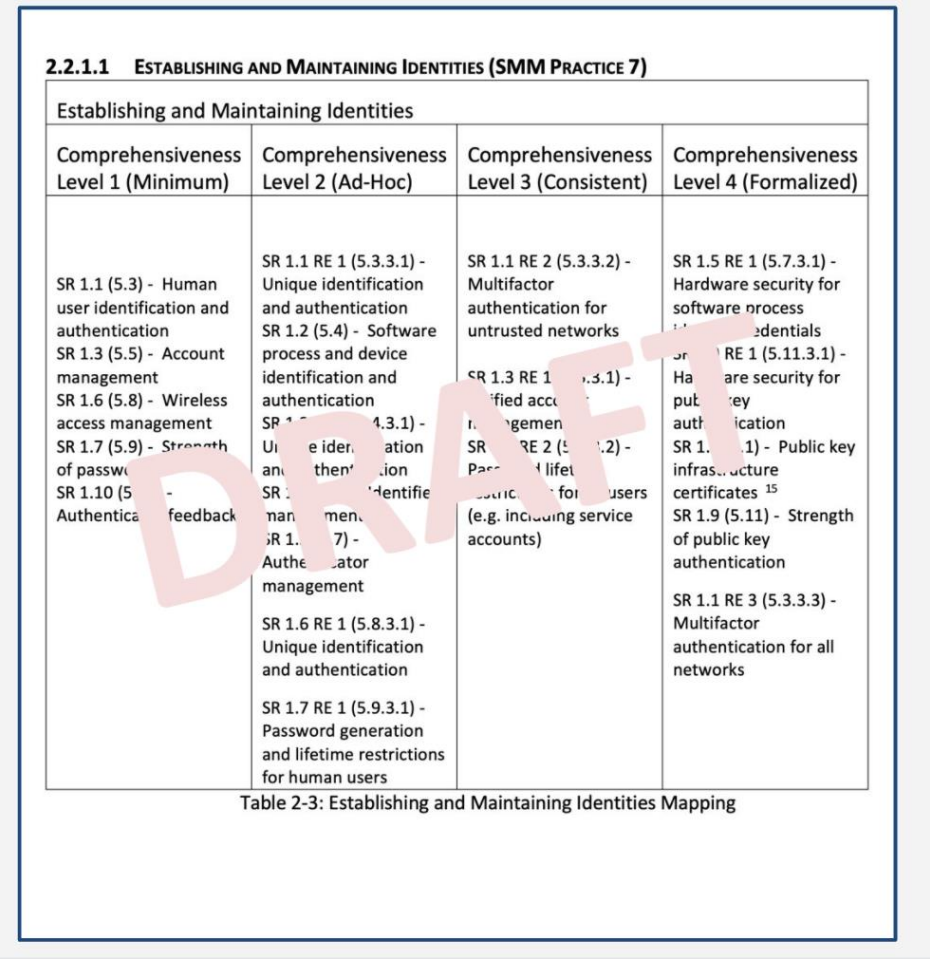
<Practice-Name>				
<Practice Description>				
	Comprehensiveness Level 1 (Minimum)	**Comprehensiveness Level 2 (Ad Hoc)**	**Comprehensiveness Level 3 (Consistent)**	**Comprehensiveness Level 4 (Formalized)**
Objective	Objective Description	Objective Description	Objective Description	Objective Description
General considerations	Level Description	Level Description	Level Description	Level Description
	What needs to be done to achieve this level Considerations	**What needs to be done to achieve this level** Considerations	**What needs to be done to achieve this level** Considerations	**What needs to be done to achieve this level** Considerations
	Indicators of accomplishment Considerations	**Indicators of accomplishment** Considerations	**Indicators of accomplishment** Considerations	**Indicators of accomplishment** Considerations

<Practice-Name>				
<Practice Description>				
	Comprehensiveness Level 1	**Comprehensiveness Level 2**	**Comprehensiveness Level 3**	**Comprehensiveness Level 4**
Objective	<Objective Level 1>	<Objective Level 2>	<Objective Level 3>	<Objective Level 4>
General considerations	<List of Level 1 general considerations>	<List of Level 2 general considerations>	<List of Level 3 general considerations>	<List of Level 4 general considerations>
Industry-specific considerations	<List of Level 1 industry specific considerations>	<List of Level 2 industry specific considerations>	<List of Level 3 industry specific considerations>	<List of Level 4 industry specific considerations>
System-specific considerations	<List of Level 1 system specific considerations>	<List of Level 2 system specific considerations>	<List of Level 3 system specific considerations>	<List of Level 4 system specific considerations>

Mappings

Relating SMM to familiar and accepted work

- SMM is a maturity model, so it does not include specific security controls
- Mappings take SMM "actions to be taken" and "indicators of accomplishment" for the maturity levels and relate them to control frameworks and best practices
- You can identify your desired maturity level for a given practice and appropriate controls that are relevant for it



2.2.1.1 ESTABLISHING AND MAINTAINING IDENTITIES (SMM PRACTICE 7)

Establishing and Maintaining Identities			
Comprehensiveness Level 1 (Minimum)	Comprehensiveness Level 2 (Ad-Hoc)	Comprehensiveness Level 3 (Consistent)	Comprehensiveness Level 4 (Formalized)
SR 1.1 (5.3) - Human user identification and authentication SR 1.3 (5.5) - Account management SR 1.6 (5.8) - Wireless access management SR 1.7 (5.9) - Strength of passw SR 1.10 (5 - Authentica feedback	SR 1.1 RE 1 (5.3.3.1) - Unique identification and authentication SR 1.2 (5.4) - Software process and device identification and authentication SR 1. 4.3.1) - U e iden ation an then ion SR dentifie man men R 1. 7) - Authe ator management SR 1.6 RE 1 (5.8.3.1) - Unique identification and authentication SR 1.7 RE 1 (5.9.3.1) - Password generation and lifetime restrictions for human users	SR 1.1 RE 2 (5.3.3.2) - Multifactor authentication for untrusted networks SR 1.3 RE 1 .3.1) - ified acc gemen SR RE 2 (.2) - Pas l lifet for users (e.g. including service accounts)	SR 1.5 RE 1 (5.7.3.1) - Hardware security for software process edentials RE 1 (5.11.3.1) - Ha are security for pub ey auth ication SR 1. .1) - Public key infras ucture certificates [15] SR 1.9 (5.11) - Strength of public key authentication SR 1.1 RE 3 (5.3.3.3) - Multifactor authentication for all networks

Table 2-3: Establishing and Maintaining Identities Mapping

Better together

- SMM Profiles
  - Extensions for different industries and purposes, for example
    - Retail
    - Digital Twins
- Mappings, for example
  - To other control security frameworks
  - ISA 62443 for different roles
- Could create SMM profile for CMMC to leverage SMM assessment approaches and mappings (such as for 62443) to identify gaps and possible controls for achieving CMMC levels
- Could combine SMM and CMMS certification review as part of an assessment, creates more consistency
- Call to action: let's create an SMM CMMC profile

SMM reference material

- SMM Main page: Security Maturity Model Practitioners' Guide | Industry IoT Consortium (iiconsortium.org)
- SMM White Paper: IoT SMM: Description and Intended Use (iiconsortium.org)
- SMM Practitioner's Guide: IoT SMM Practitioner's Guide (iiconsortium.org)
- SMM Retail Profile: IoT SMM: Retail Profile for Point-of-Sale Devices (iiconsortium.org)

CMMC AND BSIMM

PRESENTED BY: Sammy Migues, Principal Scientist, Synopsys; BSIMM co-author and analyst

October 2021

CMMC and BSIMM

- What are they?
- What's similar?
- What's different?
- Complementary use
- Data

CMMC and BSIMM purpose

- The **Cybersecurity Maturity Model Certification (CMMC)** is a new cybersecurity framework and accompanying certification by the US Department of Defense (DoD). The goal of the new CMMC compliance requirement is to protect Federal Contract Information (FCI) and Controlled Unclassified Information (CUI).

- The **Building Security In Maturity Model (BSIMM)** is a descriptive, data-driven model resulting from an ongoing study—since 2008—of *actual practices* in application security programs across many organization sizes, industry verticals, and geographies. It's both a yardstick for measuring programs and a guide for creating them.

Model domains

BSIMM Domains and Practices

DOMAINS			
GOVERNANCE	**INTELLIGENCE**	**SSDL TOUCHPOINTS**	**DEPLOYMENT**
Practices that help organize, manage, and measure a software security initiative. Staff development is also a central governance practice.	Practices that result in collections of corporate knowledge used in carrying out software security activities throughout the organization. Collections include both proactive security guidance and organizational threat modeling.	Practices associated with analysis and assurance of particular software development artifacts and processes. All software security methodologies include these practices.	Practices that interface with traditional network security and software maintenance organizations. Software configuration, maintenance, and other environment issues have direct impact on software security.
PRACTICES			
GOVERNANCE	**INTELLIGENCE**	**SSDL TOUCHPOINTS**	**DEPLOYMENT**
1. Strategy & Metrics (SM) 2. Compliance & Policy (CP) 3. Training (T)	4. Attack Models (AM) 5. Security Features & Design (SFD) 6. Standards & Requirements (SR)	7. Architecture Analysis (AA) 8. Code Review (CR) 9. Security Testing (ST)	10. Penetration Testing (PT) 11. Software Environment (SE) 12. Configuration Management & Vulnerability Management (CMVM)

CMMC Domains

Access Control (AC)	Incident Response (IR)	Risk Management (RM)
Asset Management (AM)	Maintenance (MA)	Security Assessment (CA)
Awareness and Training (AT)	Media Protection (MP)	Situational Awareness (SA)
Audit and Accountability (AU)	Personnel Security (PS)	System and Communications Protection (SC)
Configuration Management (CM)	Physical Protection (PE)	System and Information Integrity (SI)
Identification and Authentication (IA)	Recovery (RE)	

Similar topic areas

BSIMM Domains and Practices

DOMAINS			
GOVERNANCE	**INTELLIGENCE**	**SSDL TOUCHPOINTS**	**DEPLOYMENT**
Practices that help organize, manage, and measure a software security initiative. Staff development is also a central governance practice.	Practices that result in collections of corporate knowledge used in carrying out software security activities throughout the organization. Collections include both proactive security guidance and organizational threat modeling.	Practices associated with analysis and assurance of particular software development artifacts and processes. All software security methodologies include these practices.	Practices that interface with traditional network security and software maintenance organizations. Software configuration, maintenance, and other environment issues have direct impact on software security.
PRACTICES			
GOVERNANCE	**INTELLIGENCE**	**SSDL TOUCHPOINTS**	**DEPLOYMENT**
1. Strategy & Metrics (SM) 2. Compliance & Policy (CP) 3. Training (T)	4. Attack Models (AM) 5. Security Features & Design (SFD) 6. Standards & Requirements (SR)	7. Architecture Analysis (AA) 8. Code Review (CR) 9. Security Testing (ST)	10. Penetration Testing (PT) 11. Software Environment (SE) 12. Configuration Management & Vulnerability Management (CMVM)

CMMC Domains

- Access Control (AC)
- Incident Response (IR)
- Risk Management (RM)
- Asset Management (AM)
- Maintenance (MA)
- Security Assessment (CA)
- Awareness and Training (AT)
- Media Protection (MP)
- Situational Awareness (SA)
- Audit and Accountability (AU)
- Personnel Security (PS)
- System and Communications Protection (SC)
- Configuration Management (CM)
- Physical Protection (PE)
- System and Information Integrity (SI)
- Identification and Authentication (IA)
- Recovery (RE)

Similar topic areas

BSIMM Domains and Practices

DOMAINS			
GOVERNANCE	**INTELLIGENCE**	**SSDL TOUCHPOINTS**	**DEPLOYMENT**
Practices that help organize, manage, and measure a software security initiative. Staff development is also a central governance practice.	Practices that result in collections of corporate knowledge used in carrying out software security activities throughout the organization. Collections include both proactive security guidance and organizational threat modeling.	Practices associated with analysis and assurance of particular software development artifacts and processes. All software security methodologies include these practices.	Practices that interface with traditional network security and software maintenance organizations. Software configuration, maintenance, and other environment issues have direct impact on software security.
PRACTICES			
GOVERNANCE	**INTELLIGENCE**	**SSDL TOUCHPOINTS**	**DEPLOYMENT**
1. Strategy & Metrics (SM) 2. Compliance & Policy (CP) 3. Training (T)	4. Attack Models (AM) 5. Security Features & Design (SFD) 6. Standards & Requirements (SR)	7. Architecture Analysis (AA) 8. Code Review (CR) 9. Security Testing (ST)	10. Penetration Testing (PT) 11. Software Environment (SE) 12. Configuration Management & Vulnerability Management (CMVM)

CMMC Domains

- Access Control (AC)
- Incident Response (IR)
- Risk Management (RM)
- Asset Management (AM)
- Maintenance (MA)
- Security Assessment (CA)
- Awareness and Training (AT)
- Media Protection (MP)
- Situational Awareness (SA)
- Audit and Accountability (AU)
- Personnel Security (PS)
- System and Communications Protection (SC)
- Configuration Management (CM)
- Physical Protection (PE)
- System and Information Integrity (SI)
- Identification and Authentication (IA)
- Recovery (RE)

Similar topic areas

BSIMM Domains and Practices

DOMAINS			
GOVERNANCE	**INTELLIGENCE**	**SSDL TOUCHPOINTS**	**DEPLOYMENT**
Practices that help organize, manage, and measure a software security initiative. Staff development is also a central governance practice.	Practices that result in collections of corporate knowledge used in carrying out software security activities throughout the organization. Collections include both proactive security guidance and organizational threat modeling.	Practices associated with analysis and assurance of particular software development artifacts and processes. All software security methodologies include these practices.	Practices that interface with traditional network security and software maintenance organizations. Software configuration, maintenance, and other environment issues have direct impact on software security.
PRACTICES			
GOVERNANCE	**INTELLIGENCE**	**SSDL TOUCHPOINTS**	**DEPLOYMENT**
1. Strategy & Metrics (SM) 2. Compliance & Policy (CP) 3. Training (T)	4. Attack Models (AM) 5. Security Features & Design (SFD) 6. Standards & Requirements (SR)	7. Architecture Analysis (AA) 8. Code Review (CR) 9. Security Testing (ST)	10. Penetration Testing (PT) 11. Software Environment (SE) 12. Configuration Management & Vulnerability Management (CMVM)

CMMC Domains

- Access Control (AC)
- Incident Response (IR)
- Risk Management (RM)
- Asset Management (AM)
- Maintenance (MA)
- Security Assessment (CA)
- Awareness and Training (AT)
- Media Protection (MP)
- Situational Awareness (SA)
- Audit and Accountability (AU)
- Personnel Security (PS)
- System and Communications Protection (SC)
- Configuration Management (CM)
- Physical Protection (PE)
- System and Information Integrity (SI)
- Identification and Authentication (IA)
- Recovery (RE)

Similar topic areas

BSIMM Domains and Practices

DOMAINS			
GOVERNANCE	**INTELLIGENCE**	**SSDL TOUCHPOINTS**	**DEPLOYMENT**
Practices that help organize, manage, and measure a software security initiative. Staff development is also a central governance practice.	Practices that result in collections of corporate knowledge used in carrying out software security activities throughout the organization. Collections include both proactive security guidance and organizational threat modeling.	Practices associated with analysis and assurance of particular software development artifacts and processes. All software security methodologies include these practices.	Practices that interface with traditional network security and software maintenance organizations. Software configuration, maintenance, and other environment issues have direct impact on software security.
PRACTICES			
GOVERNANCE	**INTELLIGENCE**	**SSDL TOUCHPOINTS**	**DEPLOYMENT**
1. Strategy & Metrics (SM) 2. Compliance & Policy (CP) 3. Training (T)	4. Attack Models (AM) 5. Security Features & Design (SFD) 6. Standards & Requirements (SR)	7. Architecture Analysis (AA) 8. Code Review (CR) 9. Security Testing (ST)	10. Penetration Testing (PT) 11. Software Environment (SE) 12. Configuration Management & Vulnerability Management (CMVM)

CMMC Domains

- Access Control (AC)
- Incident Response (IR)
- Risk Management (RM)
- Asset Management (AM)
- Maintenance (MA)
- Security Assessment (CA)
- Awareness and Training (AT)
- Media Protection (MP)
- Situational Awareness (SA)
- Audit and Accountability (AU)
- Personnel Security (PS)
- System and Communications Protection (SC)
- Configuration Management (CM)
- Physical Protection (PE)
- System and Information Integrity (SI)
- Identification and Authentication (IA)
- Recovery (RE)

But they are different and have different context

BSIMM

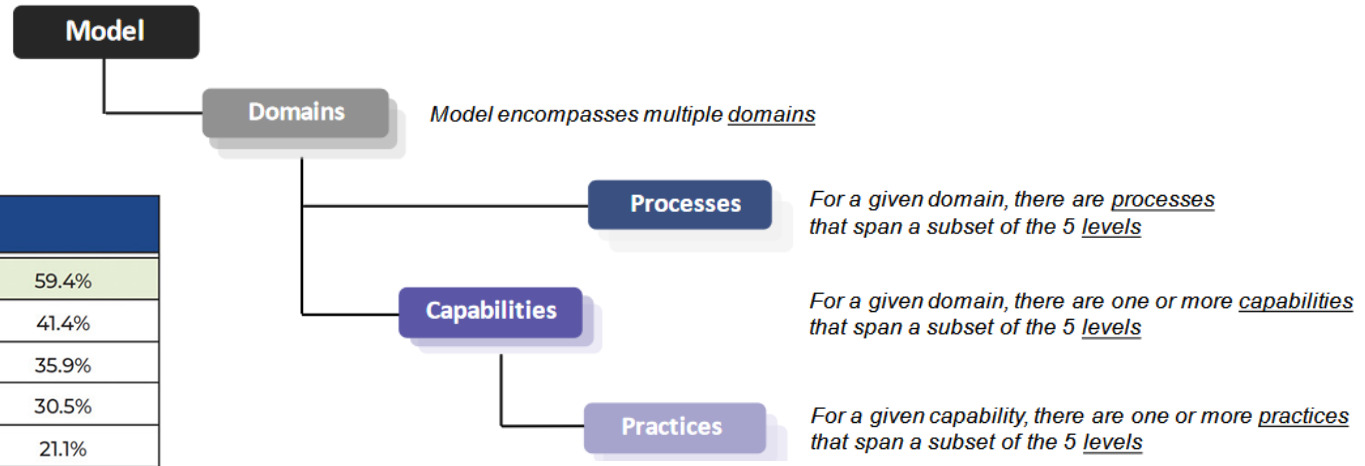
- Any organization, anywhere
- Actual practices in current app security programs
- Program scorecard; self-assessments encouraged
- 122 unique activities / controls

1. Strategy & Metrics (SM)
2. Compliance & Policy (CP)
3. Training (T)
4. Attack Models (AM)
5. Security Features & Design (SFD)
6. Standards & Requirements (SR)
7. Architecture Analysis (AA)
8. Code Review (CR)
9. Security Testing (ST)
10. Penetration Testing (PT)
11. Software Environment (SE)
12. Configuration Management & Vulnerability Management (CMVM)

TRAINING (T)			
Conduct software security awareness training.	[T1.1]	76	59.4%
Deliver on-demand individual training.	[T1.7]	53	41.4%
Include security resources in onboarding.	[T1.8]	46	35.9%
Enhance satellite through training and events.	[T2.5]	39	30.5%
Create and use material specific to company history.	[T2.8]	27	21.1%
Deliver role-specific advanced curriculum.	[T2.9]	35	27.3%
Reward progression through curriculum.	[T3.1]	6	4.7%
Provide training for vendors and outsourced workers.	[T3.2]	23	18.0%
Host software security events.	[T3.3]	23	18.0%
Require an annual refresher.	[T3.4]	24	18.8%
Establish SSG office hours.	[T3.5]	9	7.0%
Identify new satellite members through observation.	[T3.6]	4	3.1%

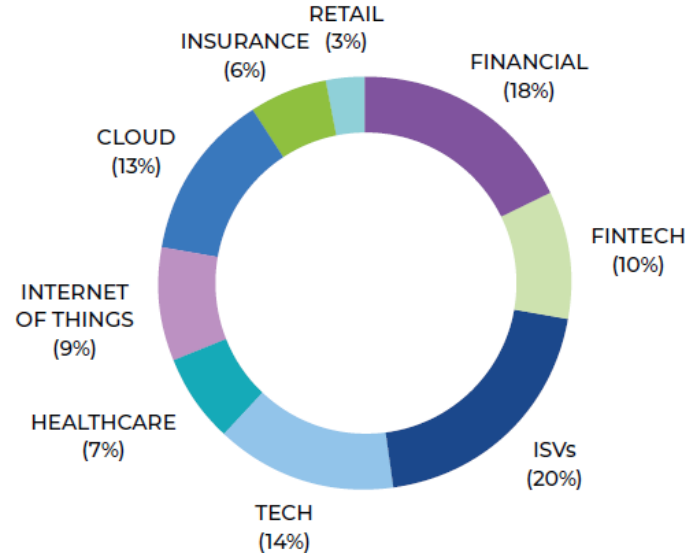
CMMC

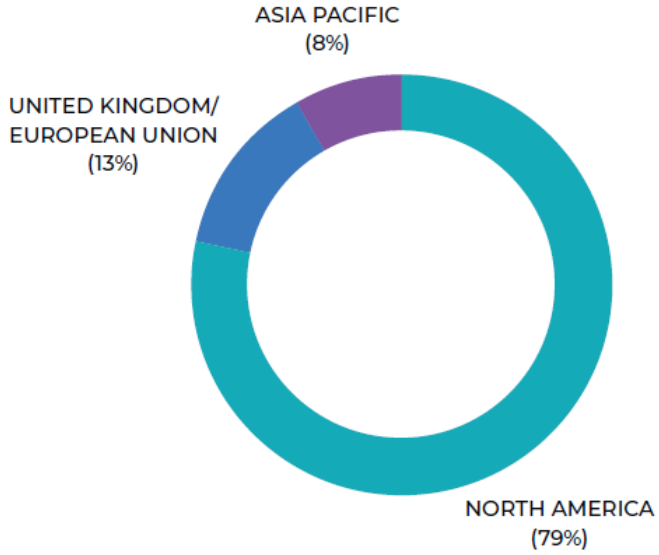
- DoD and defense contractors
- Cybersecurity best practices from various standards
- Certification element; self-assessments encouraged
- Level 3 includes 130 practices



BSIMM Participants

	BSIMM12	BSIMM11	BSIMM10	BSIMM9	BSIMM8	BSIMM7	B
FIRMS	128	130	122	120	109	95	
MEASUREMENTS	341	357	339	320	256	237	
2ND MEASURES	31	32	50	42	36	30	
3RD MEASURES	14	12	32	20	16	15	
4TH MEASURES	4	7	8	7	5	2	
SSG MEMBERS	2,837	1,801	1,596	1,600	1,268	1,111	
SATELLITE MEMBERS	6,448	6,656	6,298	6,291	3,501	3,595	
DEVELOPERS	398,544	490,167	468,500	415,598	290,582	272,782	2
APPLICATIONS	153,519	176,269	173,233	135,881	94,802	87,244	6
AVG. SSG AGE (YEARS)	4.41	4.32	4.53	4.13	3.88	3.94	
AVG. SSG RATIO	2.59/100	2.01/100	1.37/100	1.33/100	1.60/100	1.61/100	1.





550 assessments across 231 firms since 2008
122 activities (controls) in BSIMM12

Trends seen in recent BSIMM data

- Governance automation and governance-as-code
- Continuous defect discovery
- Security as part of resilience and quality
- Growth in software supply chain risk management

BSIMM reference material

- BSIMM Main page: Building Security In Maturity Model (bsimm.com)

PROGRAM BREAK

15-MINUTE BREAK

PROGRAM WILL RESUME AT 10:30AM EST

THANK YOU TO OUR SPONSORS FOR MAKING THIS EVENT POSSIBLE!

DEVOPS IMPLEMENTATION

PRESENTED BY:

Challenges in implementing and sustaining DevOps environment

Hasan Yasar
Director, Lifecycle Innovation and Automation
Software Engineering Institute | Carnegie Mellon University

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.

- DM19-0402

Challenges in Implementing and sustaining DevOps environment

- DevOps Foundation

DevOps ?

DevOps is a set of principles and practices which enable better communication and collaboration between relevant stakeholders for the purpose of specifying, developing, continuously improving, and operating software and systems products and services *

Four Fundamental Principles

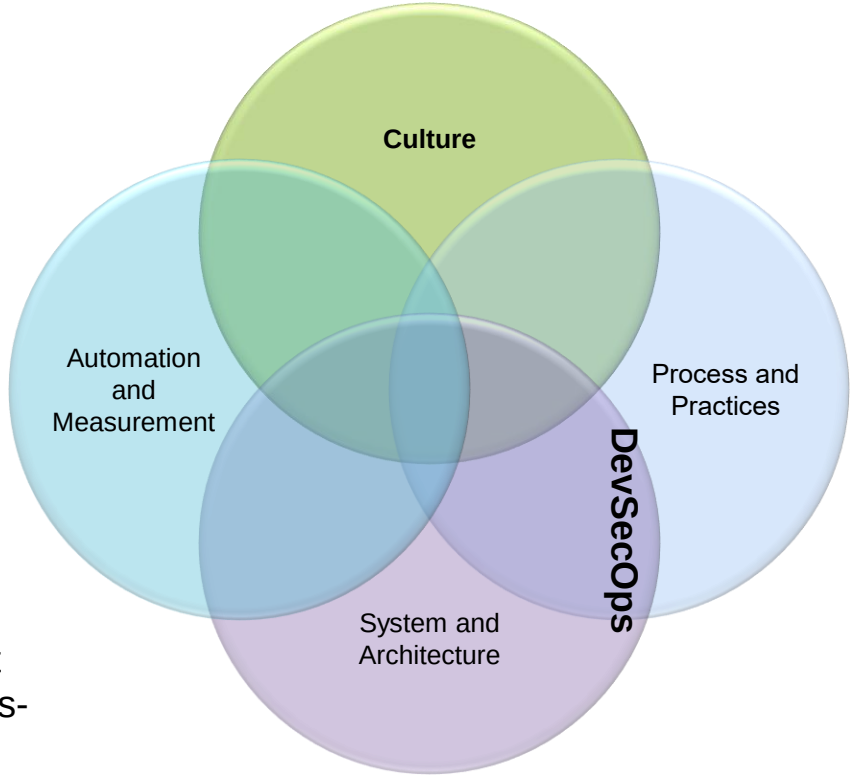
1. ***Collaboration****:* between all stakeholders
2. ***Infrastructure as code (IaC):*** assets are versioned, scripted, and shared
3. ***Automation***: deployment, testing, provisioning, any manual or human-error-prone process
4. ***Monitoring***: any metric in development or operation that can inform priorities, direction, and policy

* IEEE P2675 DevOps Standard for Building Reliable and Secure Systems Including Application Build, Package and Deployment

Dimension of DevOps

Automation/ Measurement

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



Culture

- All stakeholders collaborate
- *Developers* and Operations support releases beyond deployment
- Continuous learning
- Transparent and sharable
- Constant communication

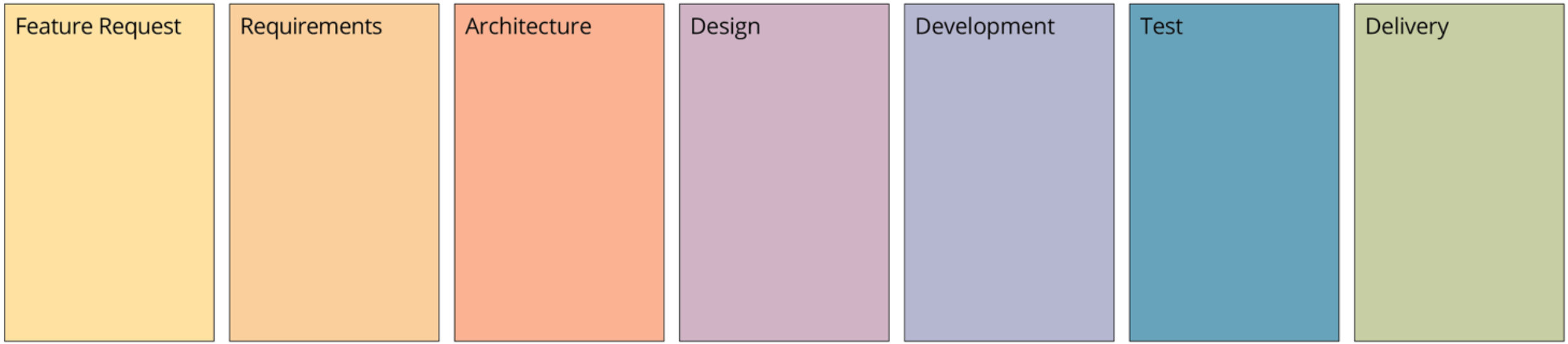
System Architecture

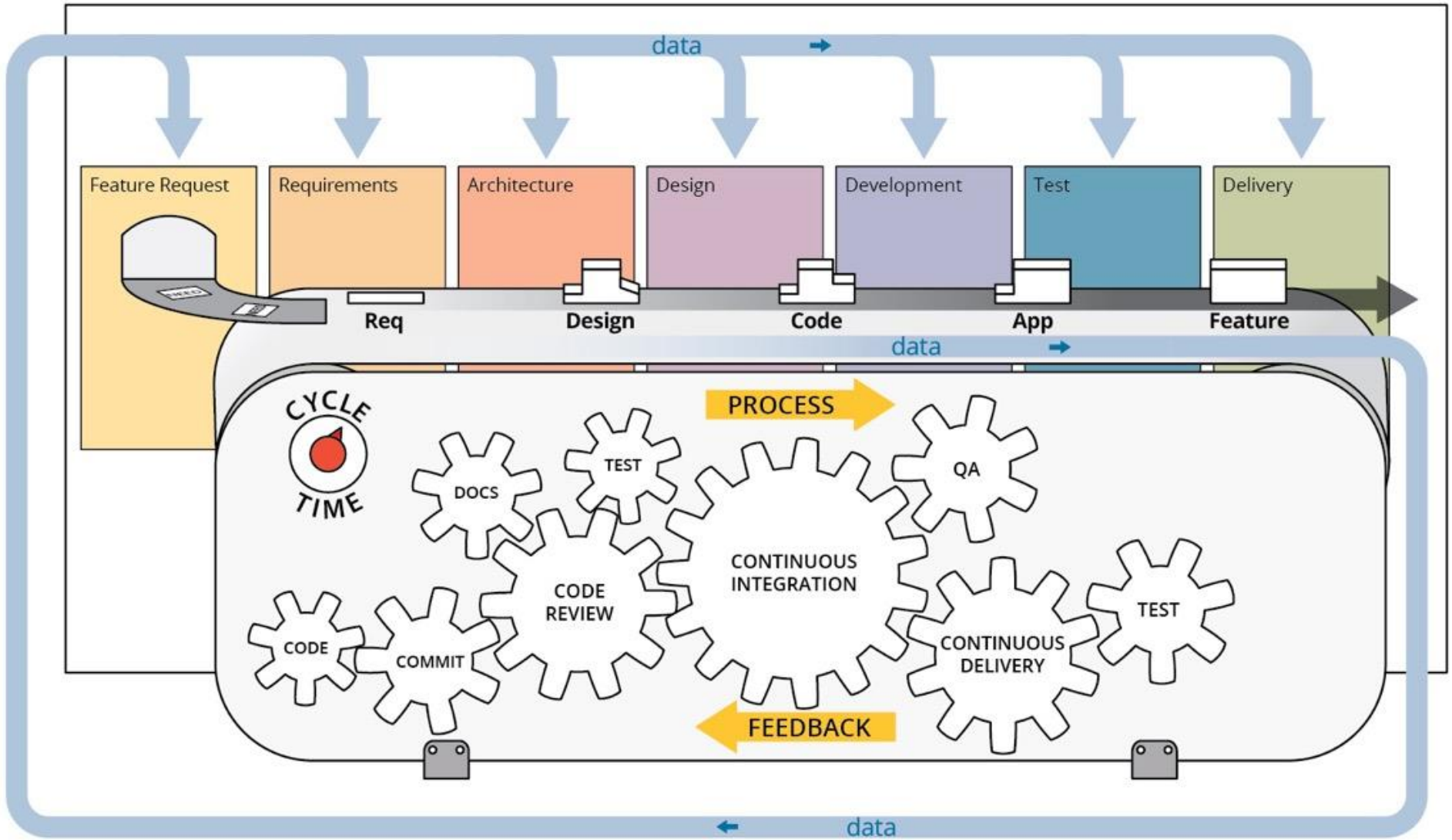
- Architected to support test automation and continuous-integration goals
- Applications that support changes without release (e.g., late binding)
- Scalable, secure, reliable, etc.

Process and Practices

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

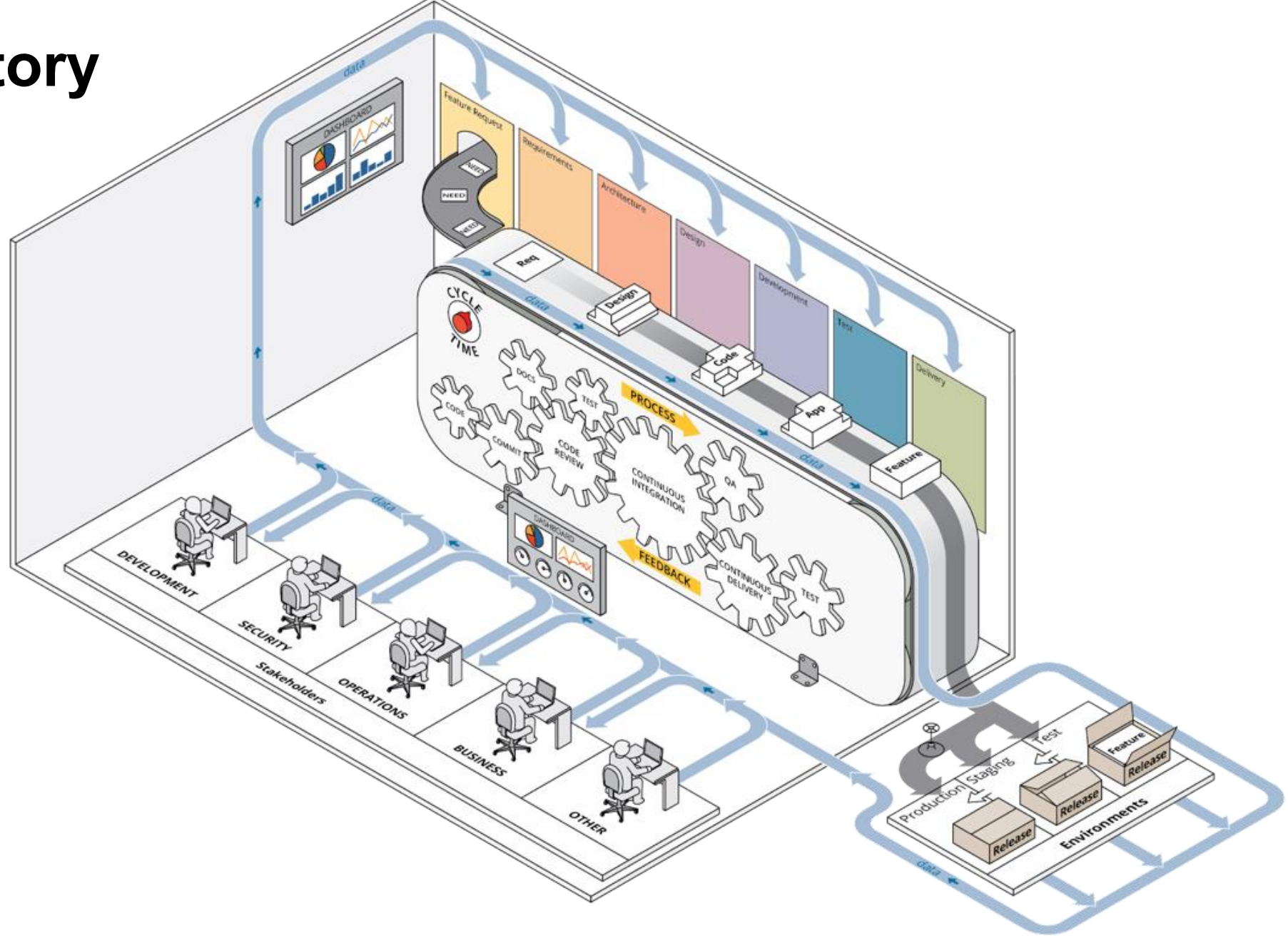
SW Development Phases – *on each iteration/sprint*



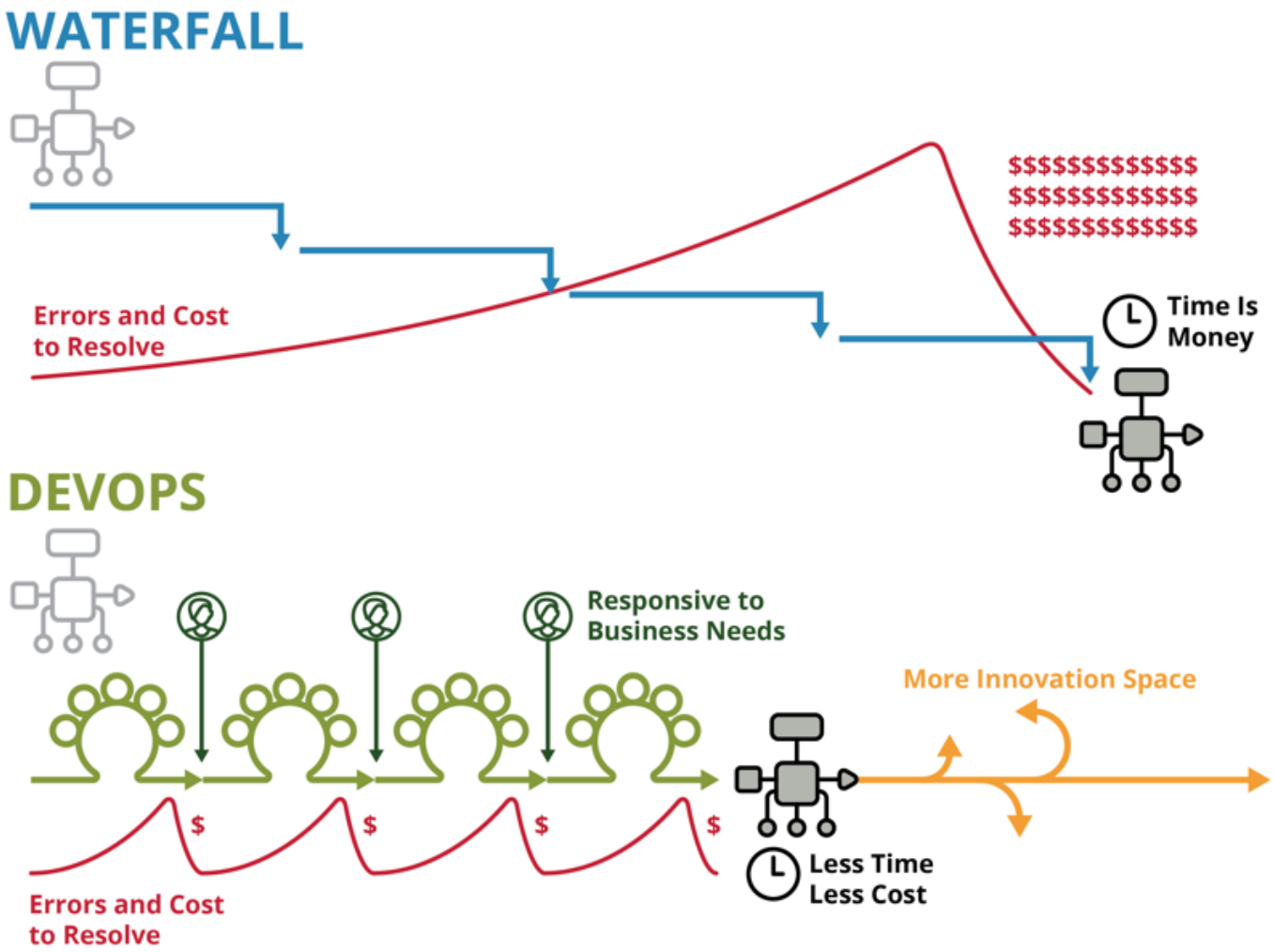

data
Feature Request
Requirements
Architecture
Design
Development
Test
Delivery
Req
Design
Code
App
Feature
data
CYCLE TIME
PROCESS
DOCS
TEST
QA
CODE REVIEW
CONTINUOUS INTEGRATION
CODE
COMMIT
CONTINUOUS DELIVERY
TEST
FEEDBACK
data

The DevOps Factory

- Feature to deployment
- Iterative and incremental development
- Automation in every phase of the SDLC
- Continuous feedback
- Metrics and measurement
- Complete engagement with all stakeholders
- Transparency and traceability across the lifecycle



Key Benefits of DevOps



- Reduced errors during deployment
- Reduced time to deploy and resolve discovered errors
- **Repeatable** steps
- **Continuous availability** of pipeline and application
- Increased innovation time
- **Responsiveness** to business needs
- **Traceability** throughout the application lifecycle
- Increased stability and quality
- **Continuous feedback**

DevOps ROI

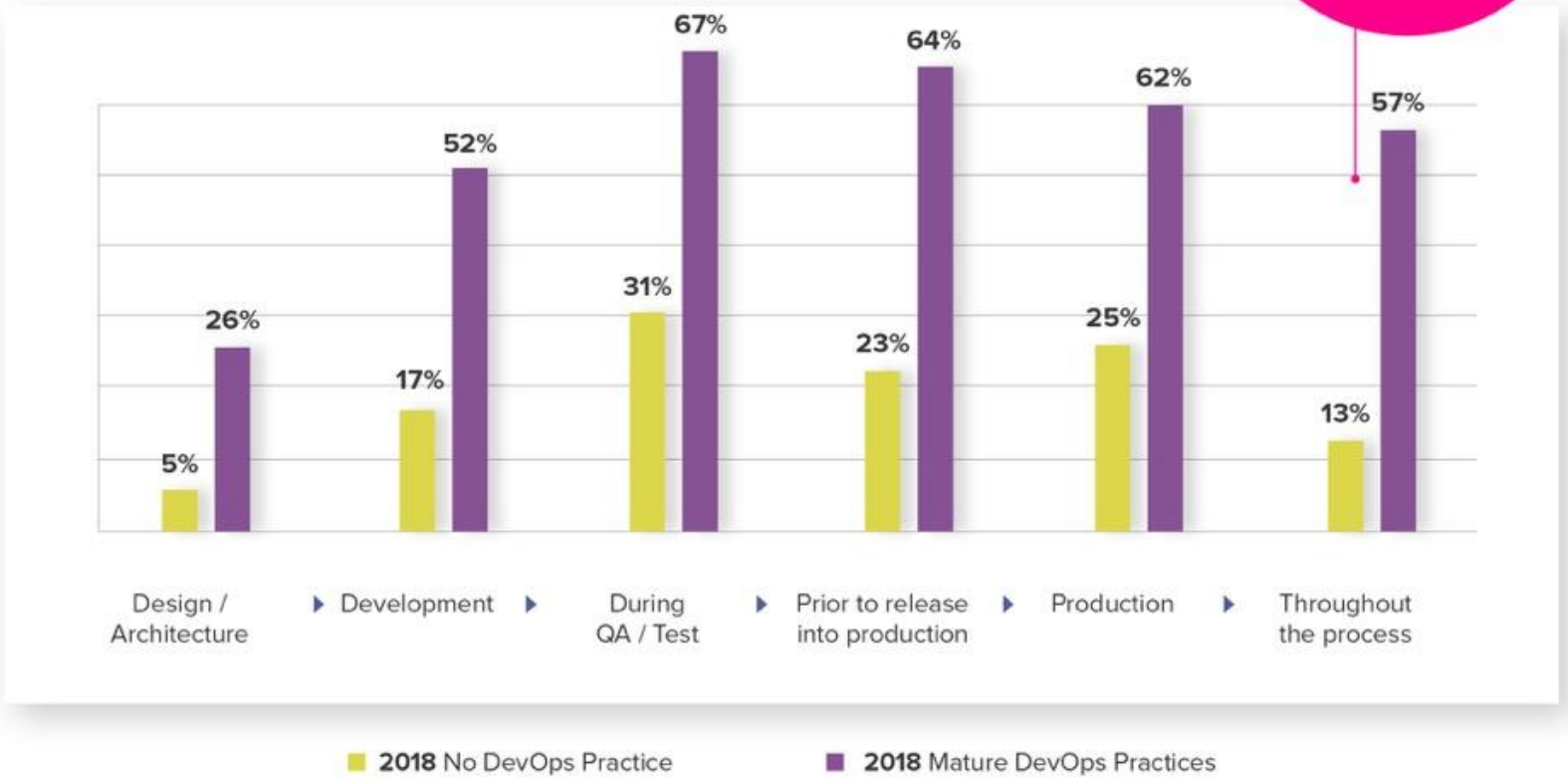
Yearly Returns Possible from Cost of Unnecessary Rework Avoided	High IT Performer	Medium IT Performer	Low IT Performer
LARGE ORGANIZATION that relies on in-house software (8,500 technical staff)	8,500 staff x $105,000 salary x 1.5 benefits x 1% rework **= $13.4M**	8,500 staff x $105,000 salary x 1.5 benefits x 12% rework **= $160.7M**	8,500 staff x $105,000 salary x 1.5 benefits x 7% rework **= $93.7M**
MEDIUM TO LARGE TECHNICAL ORGANIZATION (2,000 technical staff)	2,000 staff x $105,000 salary x 1.5 benefits x 1% rework **= $3.2M**	2,000 staff x $105,000 salary x 1.5 benefits x 12% rework **= $37.8M**	2,000 staff x $105,000 salary x 1.5 benefits x 7% rework **= $22.1M**
SMALL TO MEDIUM BUSINESSES AND NON-TECHNICAL ENTERPRISES (250 technical staff)	250 staff x $105,000 salary x 1.5 benefits x 1% rework **= $393.8K**	250 staff x $105,000 salary x 1.5 benefits x 12% rework **= $4.7M**	250 staff x $105,000 salary x 1.5 benefits x 7% rework **= $2.8M**

*DORA, DevOps ROI

At what point in the development process does your organization perform automated application security analysis?

Mature DevOps practices are 338% more likely to integrate automated security.

350% 2019 survey

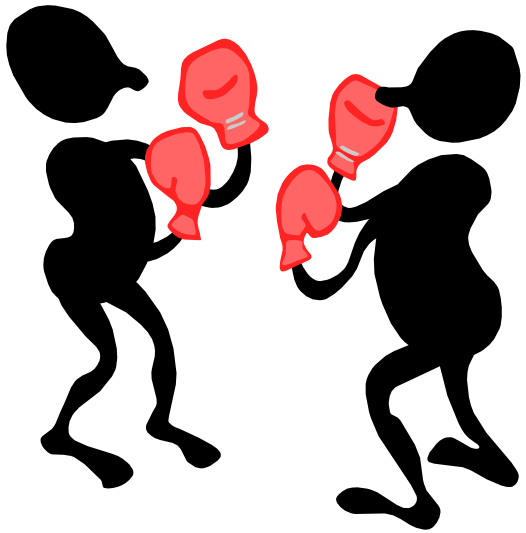


Challenges in Implementing and sustaining DevOps environment

- Obstacles & Recommendations

1. Culture

Incentivizing Behaviors



- 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

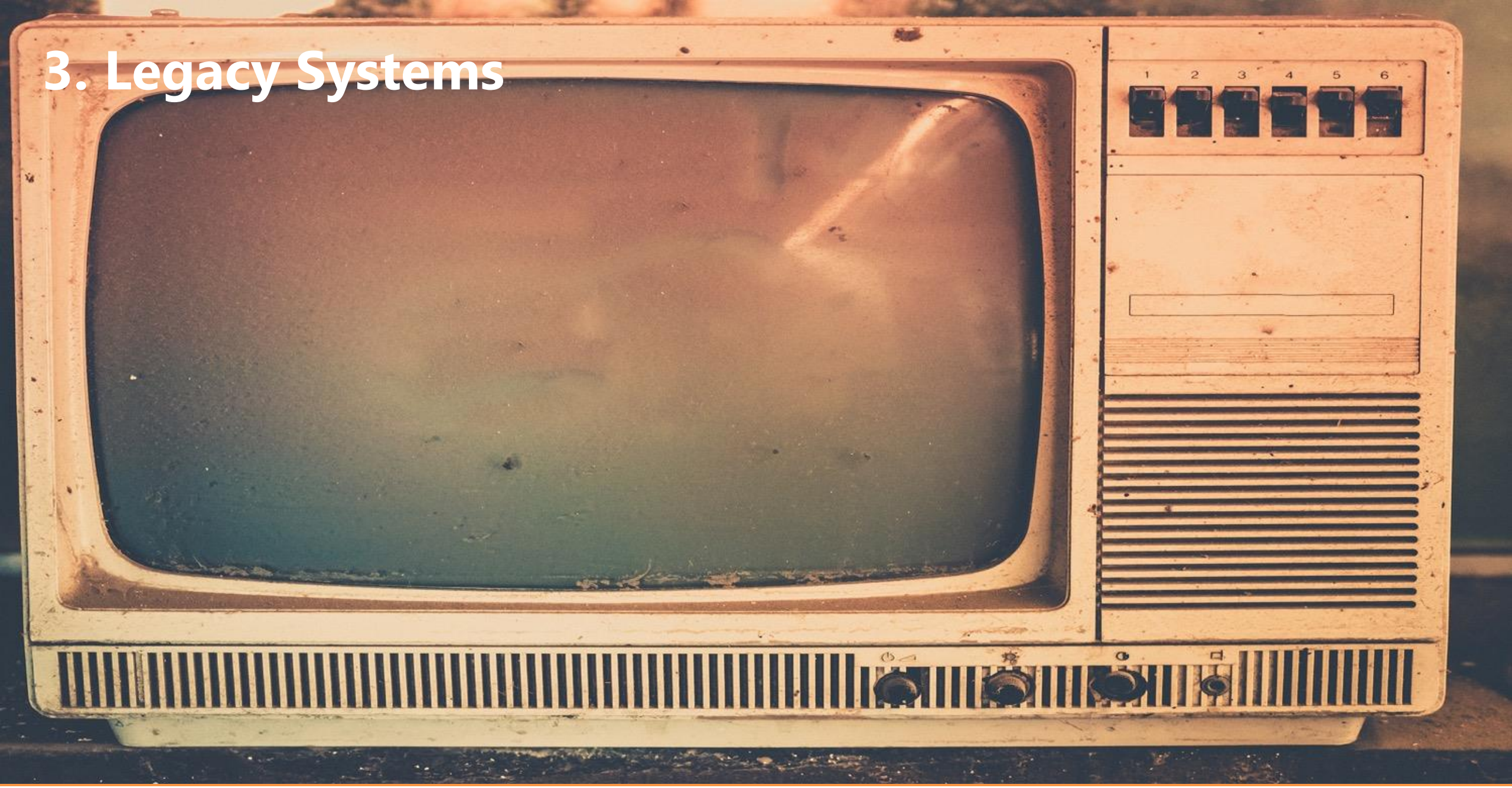
2. Organizational Structure

Conway's Law:

" How to organize our teams affects how we perform our work"

- Share common goals from top to bottom
- Enable business value oriented team
- Functional Team
- Share responsibilities (like Security is everyone's job)
- Keep team size small

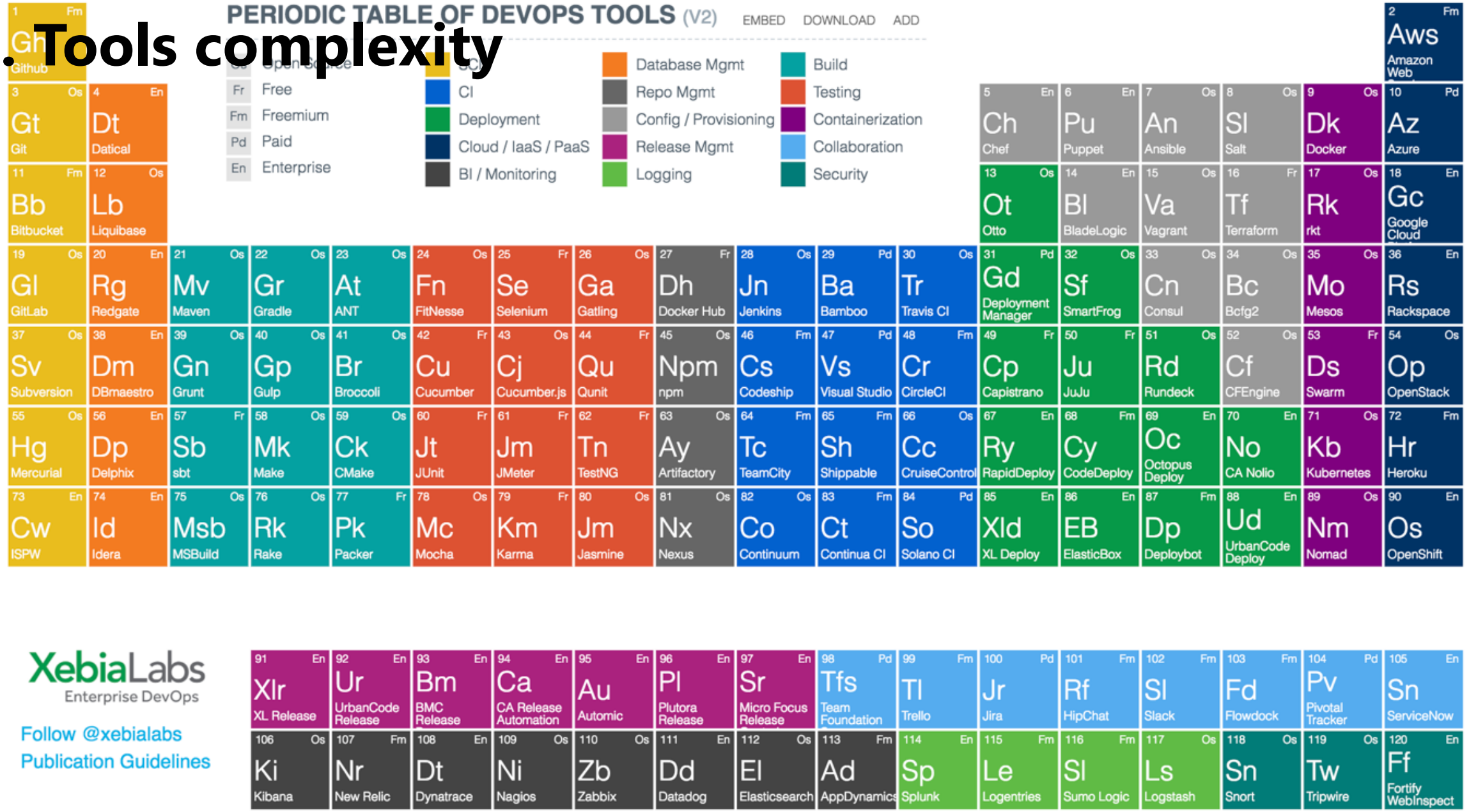
3. Legacy Systems



Apply DevOps to migrate Legacy Systems

- Ancient systems should be replaced.
- Installing new systems to fit in
- Build a new version instead of maintaining
- Re-architect to support incremental and iterative development
- Enable dynamic integration of systems

4. Tools complexity



XebiaLabs
Enterprise DevOps

Follow @xebialabs
Publication Guidelines

Platform as a Service for SW development

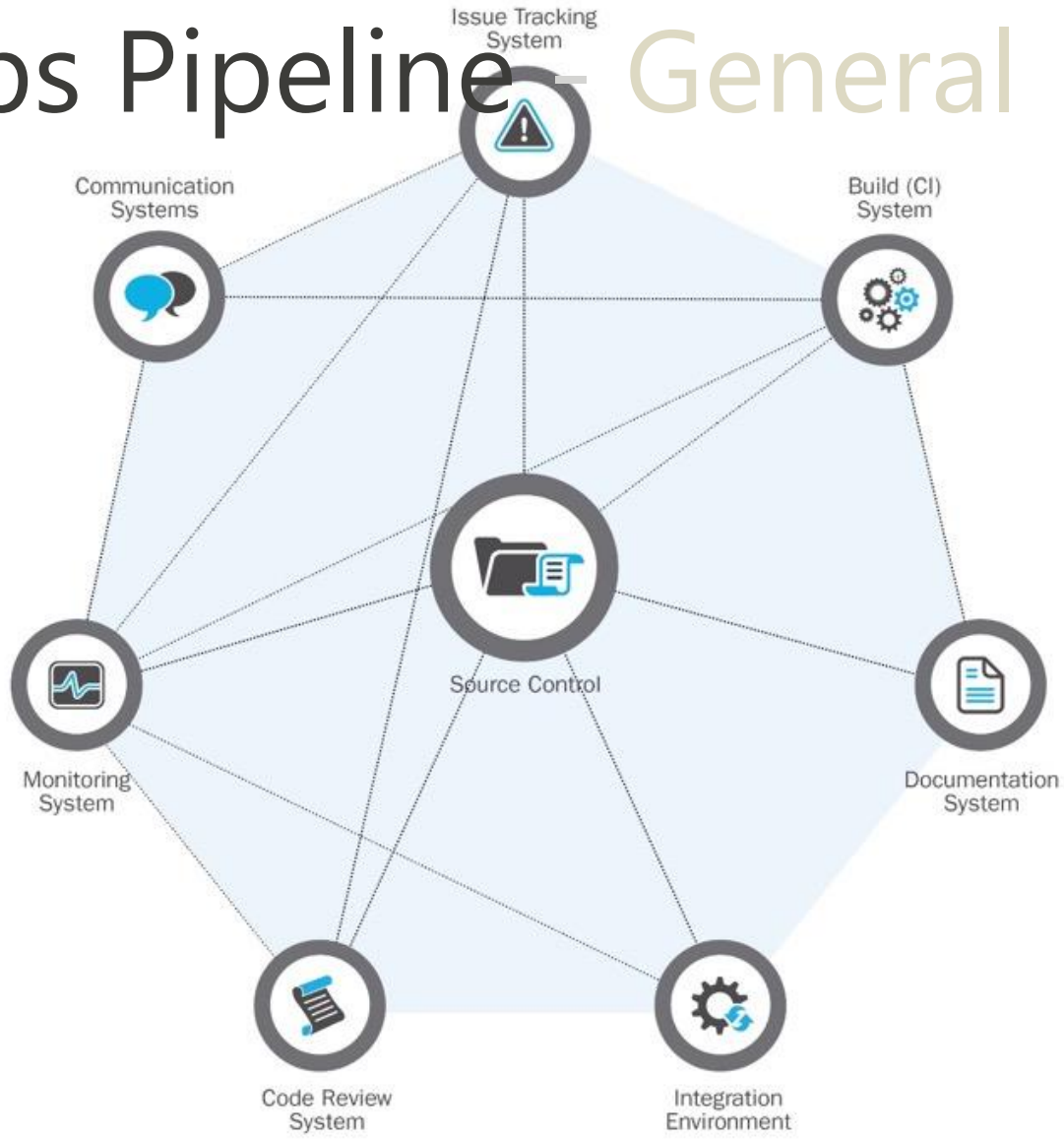
- 12Factor is a methodology for building software-as-a-service apps. https://12factor.net
- Heroko, https://www.heroku.com
- Cloud Foundry, https://www.cloudfoundry.org
- Pivotal, https://pivotal.io
- Amazon DevOps, https://aws.amazon.com/devops/
- Azure DevOps, https://azure.microsoft.com/en-us/services/devops/
- OpenStack – Open Shift, https://www.openshift.com
- Electric Cloud https://electric-cloud.com
- …….

Key Considerations

- Integrate-ability
- Interoperability
- Usability
- Portability
- Reliability
- Security/Permissions/ATO
- Availability
- Scalability

- Affordable
- Performance
- Modifiability
- Configurability
- “Automate-ability” (of manual tasks)
- “Approvability” (allows for manual approval)
- Measurability
- Adaptability
- Connectivity with other platform

DevOps Pipeline – General



DevOps Pipeline – With tooling

4. Lack of Metrics and Measurements

Decide what to measure

- Deployment frequency
- Change Lead time and Volume
- # of work items (tickets)
- Defect escape rate
- Mean time to detection (MTTD)
- Mean time to recovery (MTTR)
- Application performance
- Time to approval
- Time to patch vulnerabilities
- Operational Logs (IP, Stack Trace, Rate of Attack etc)
- Server/Services Usage (Disk, memory, CPU)

5. Process Challenges

DevOps Enabler..

Establish a process to enable people to succeed using the platform to develop Secure application

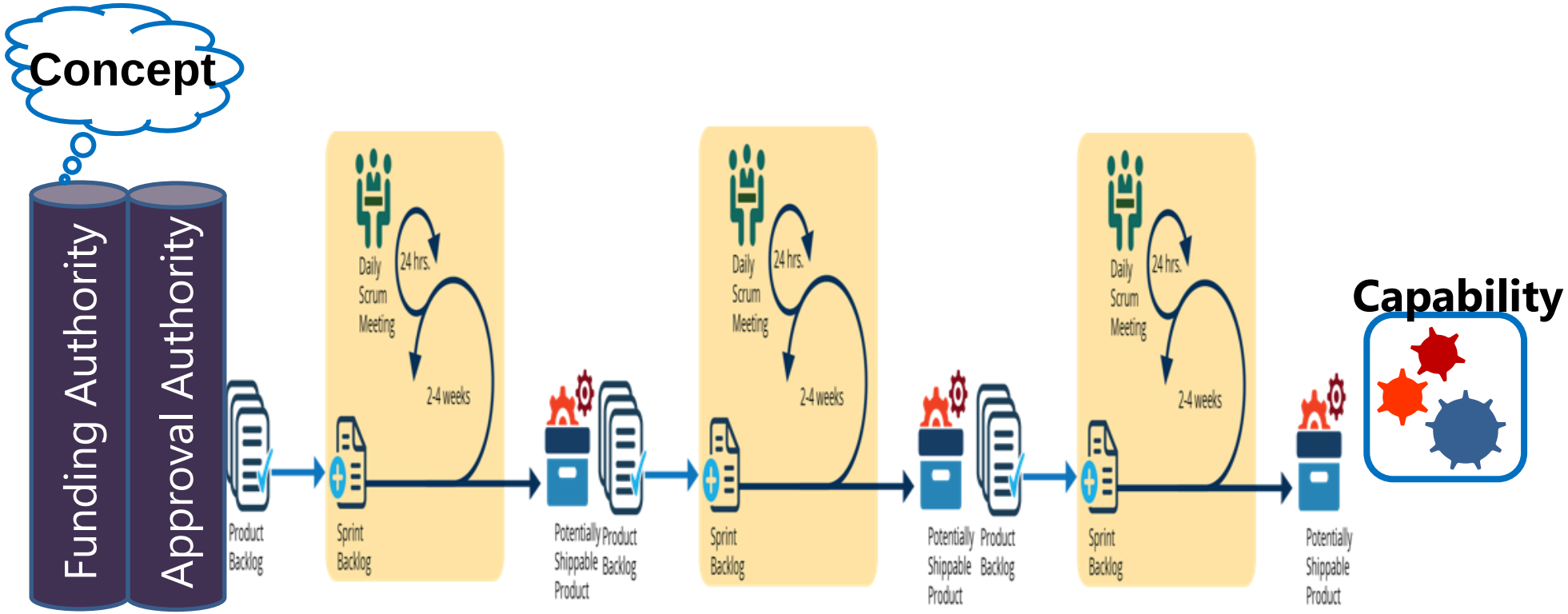
Such that;

- Constant communication and visible to all
- Ensures that tasks are testable and repeatable
- Frees up human experts to do challenging, creative work
- Allows tasks to be performed with minimal effort or cost
- Creates confidence in task success, after past repetitions
- Faster deployment , frequent quality release

6. DevOps and Acquisition

Apply DevOps Mindset

- Understand many portfolios of work as a continuous flow of smaller efforts



- Expand the collaboration, iteration, distributed (automated) governance constructs of Agile and DevOps to acquisition, needs analysis, certification, etc…

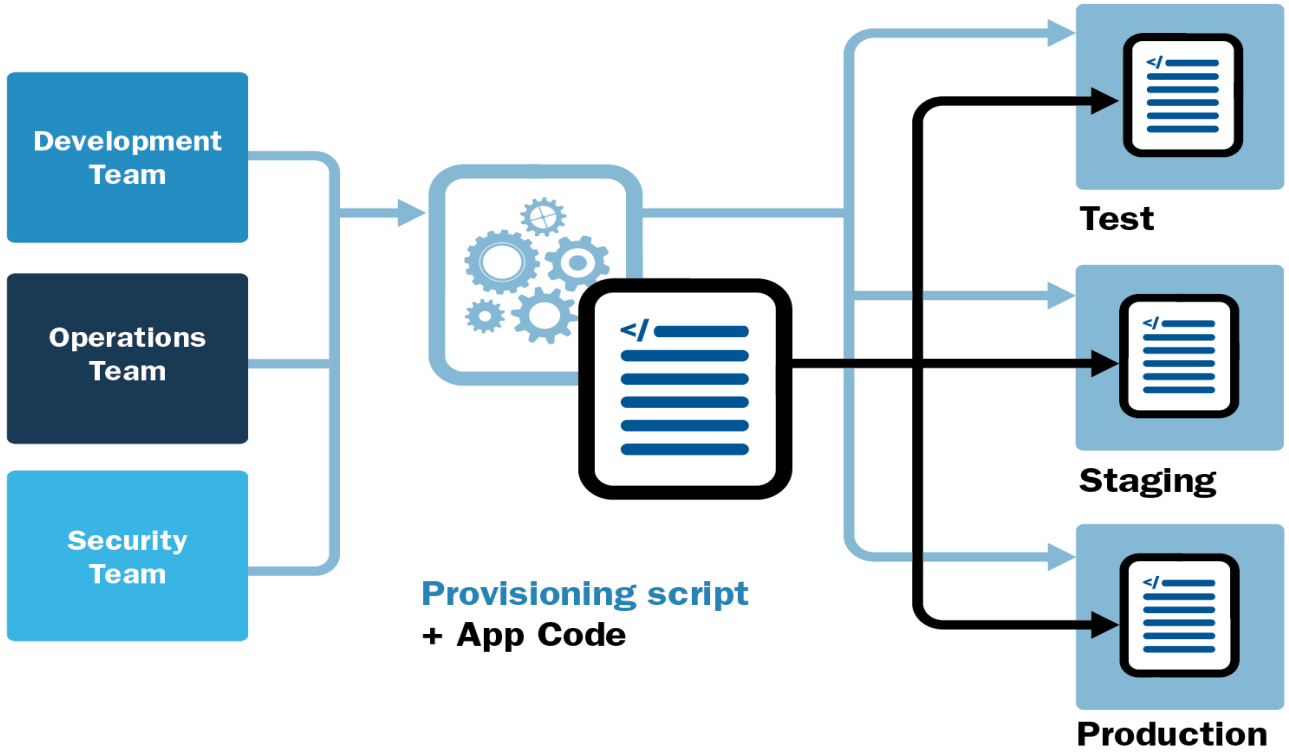
7. DevOps and Governance

Compliance as Code

- Plan from beginning and carry-out throughout the lifecycle
- Enable audit log
- Design DevOps pipeline to comply with governance
- Make policy available to all stakeholders
- Implement configuration management and keep track every changes.
- Check and verify any configuration items
- Enable base configuration/OS
- Centralized and automated compliance policy

8. Inconsistent environments

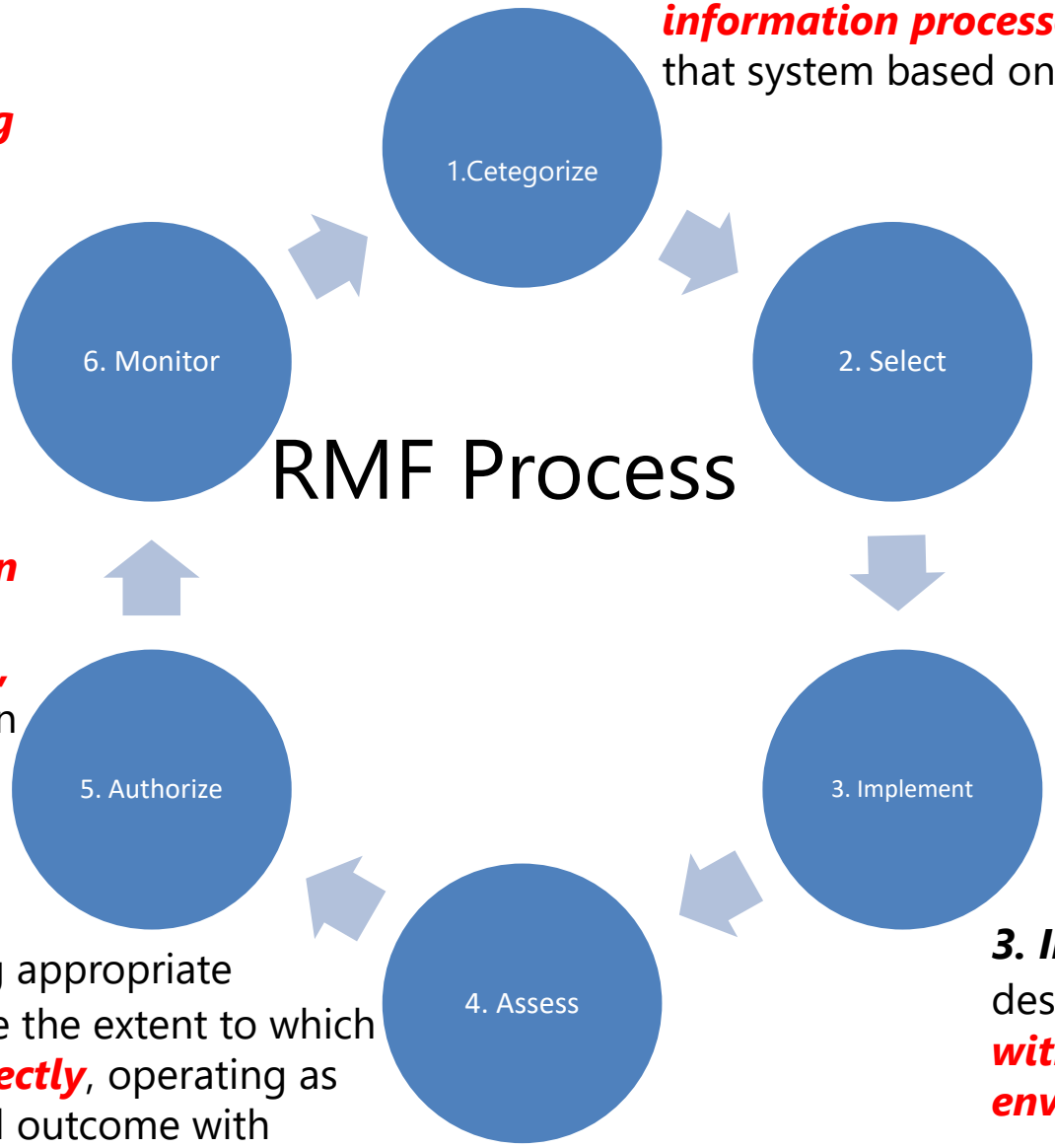
Use Infrastructure as Code (IaC)



- Environment parity throughout the development pipeline
- Develop and treat provisioning scripts as part of code repository
- Share IaC amongst the developer and IT operational teams

9. Security (RMF, ATO)

1. Categorize the information system and the ***information processed, stored, and transmitted*** by that system based on an impact analysis



2. Select an initial set of baseline security controls for the information system based on the security categorization; tailoring and supplementing the security control baseline as needed based on an ***organizational assessment of risk and local conditions***.

3. Implement the security controls and describe how ***the controls are employed within the information system and its environment of operation***.

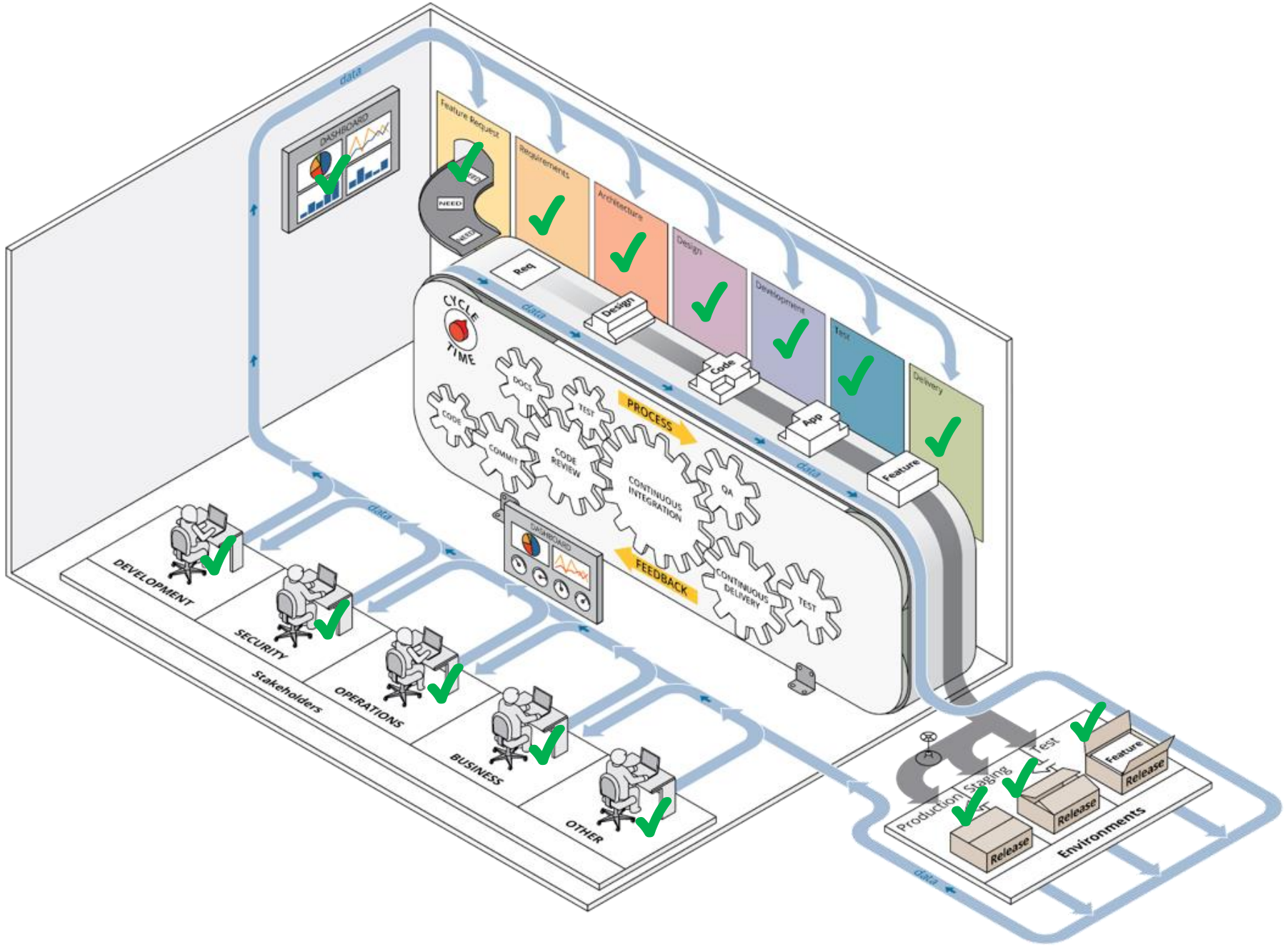
4. Assess the security controls using appropriate assessment procedures to determine the extent to which the ***controls are implemented correctly***, operating as intended, and producing the desired outcome with respect to meeting the security requirements for the system

5. Authorize information system operation based on a ***determination*** of ***the risk to organizational operations and assets, individuals, other organizations***, and the Nation resulting from the operation of the information system and the decision that this risk is acceptable.

6. Monitor the security controls in the information system on an ***ongoing basis including assessing control effectiveness, documenting changes to the system or its environment of operation***, conducting security impact analyses of the associated changes, and reporting the security state of the system to designated organizational officials.

- **DevSecOps** is a model on integrating the software development and operational process considering security activities: requirements, design, coding, testing , delivery , deployment and incident response.

Think Security from Inception to Deploy and improve on every delivery



10. Sustaining

Set up Cost(Effort Cost)

# Of Users	# of App	Key CI/CD	Common Tools	Dev+Ops	Estimated Cost
<100	2-3	IaC, Single Cloud & App Support Monitoring and Notification	AWS, GCP, Docker, Jenkins, Nagios, PagerDuty, Slack, WikiPages, Jira, owasp zap, Twistlock	1 Dev + 1 Ops	$350K
100-1000	>3, <100	IaC, Hybrid Cloud & App Support, Configuration Management, Env&Pipeline Management, Release Strategies, Monitoring, Deployment Verification.	AWS, GCP, Docker, Jenkins, Nagios, PagerDuty, Slack, WikiPages, Jira, owasp zap, Twistlock, Circle CI, Sumo Logic, Logz.io, Kubernetes	3 Dev + 1 Ops	$750K
>1000	>100	IaC, Hybrid Cloud & App Support, Configuration Management, Env&Pipeline Management, Release Strategies, Monitoring, Deployment Verification, Rollback, Secure Pipeline Management, Auditing and Compliance, Dashboard, Scalability, Organization/project segregation, multiple pipelines support.	AWS, GCP, Docker, Jenkins, Nagios, PagerDuty, Slack, WikiPages, Jira, owasp zap, Twistlock, Circle CI, Sumo Logic, Logz.io, Kubernetes, cloud fonduary, Vsphere, AppDyanmics, Dynatrace, Splunk, Chefm Ansible, HashiCorb, Xmatters, OpenShift	10 Dev + 4 Ops	$2.600K

- Dev: $200K, Ops:150K
- Platform depends on * on-prem or cloud

How to sustain DevOps environment

- **Effective Usage**
- Train Users and build DevOps skills
- All stakeholders access
- Playbook/Developers guidance
- Project startup guidance
- Project Architectural Guidance
  - Common Services,
  - Common Security approach
  - Architectural patterns
  - Test methods
- DevOps environment usage policy
  - Build and Deployment Strategies

- **Maintaining (cost/update)**
- Updating the environment (new version or security patches)
- Supporting new tools
- Adding/setting up new projects
- Operational Support
  - Base Image, OSS Support, Test harness, Temp Environment Creation
- Pipeline orchestration
- Securing pipeline
- Usage meter/billing support
- Auditability/log and data collection

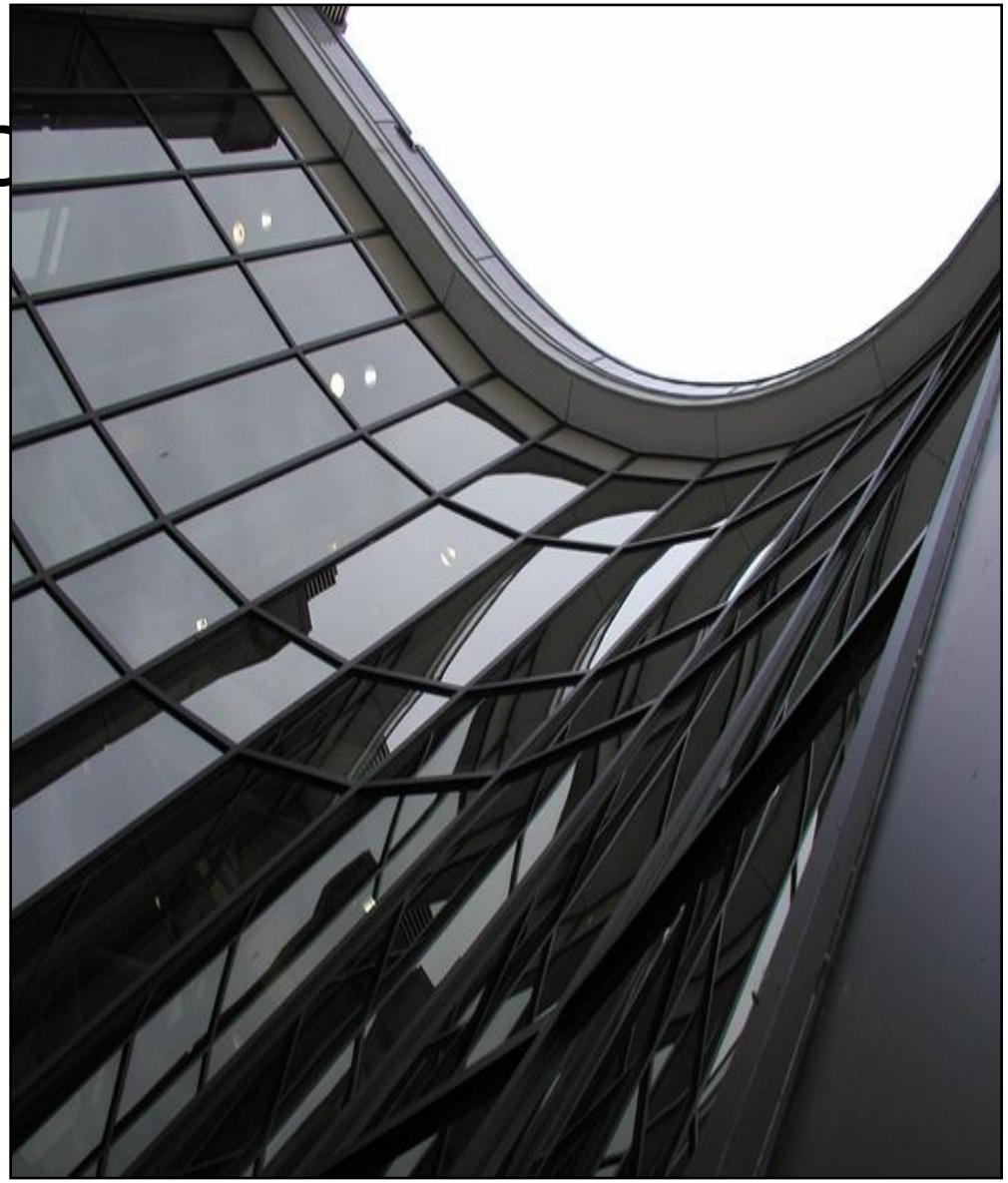
SEI DevOps GitHub Projects

- Once Click DevOps deployment
  https://github.com/SLS-ALL/devops-microcosm
- Sample app with DevOps Process
  https://github.com/SLS-ALL/flask_api_sample
  - Tagged checkpoints
    - v0.1.0: base Flask project
    - v0.2.0: Vagrant development configuration
    - v0.3.0: Test environment and Fabric deployment
    - v0.4.0: Upstart services, external configuration files
    - v0.5.0: Production environment
- On YouTube:
  https://www.youtube.com/watch?v=5nQlJ-FWA5A

For more information…

DevOps: https://www.sei.cmu.edu/go/devops
DevOps Blog: https://insights.sei.cmu.edu/devops
Webinar : https://www.sei.cmu.edu/publications/webinars/index.cfm
Podcast : https://www.sei.cmu.edu/publications/podcasts/index.cfm

Any Questio



Hasan Yasar

Interim Director,
Lifecycle Innovation and Automation

hyasar@sei.cmu.edu

@securelifecycle

SOFTWARE SUPPLY CHAIN TRANSPARENCY

CISQ

Consortium for Information & Software Quality™

PRESENTED BY:

Software Supply Chain Transparency

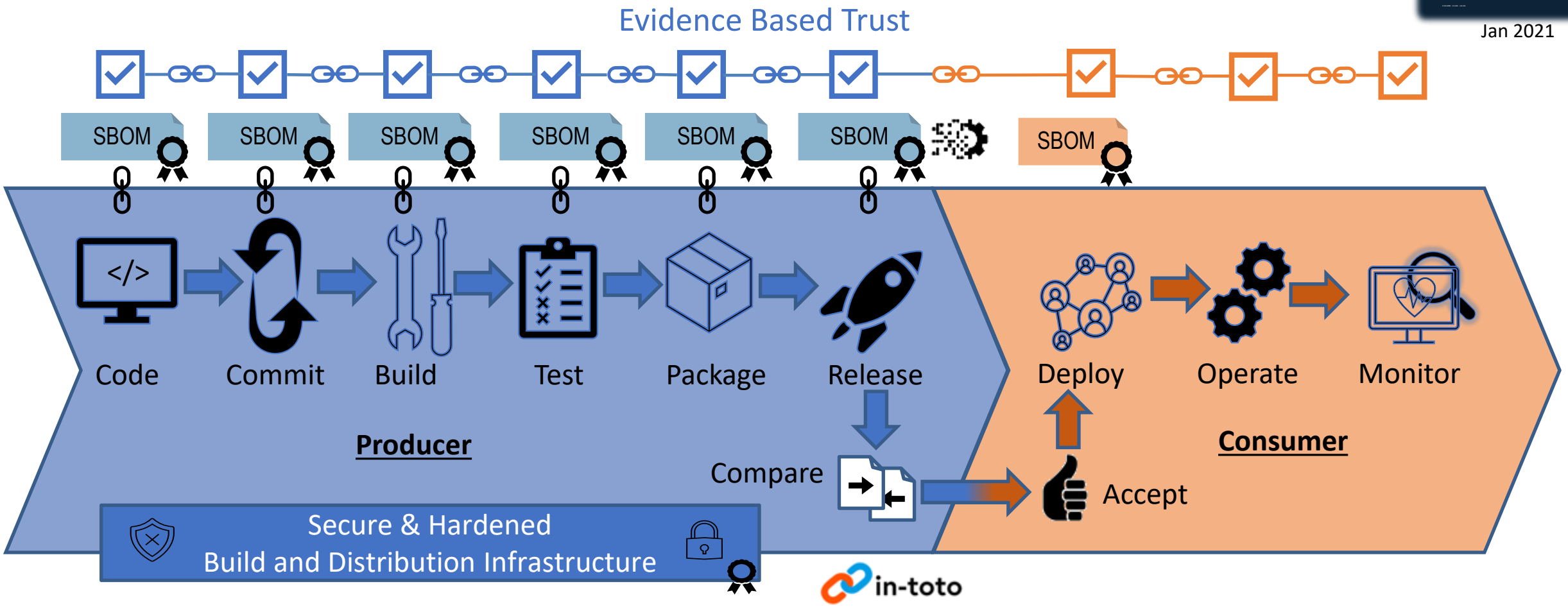
Sr. Software and Supply Chain Assurance Prin. Eng.
Cross Cutting Solutions and Innovation Dept.
Cyber Solutions Innovation Center
MITRE Labs

9TH ANNUAL CYBER RESILIENCE SUMMIT
Virtual
October 12th, 2021

CISQ
Consortium for Information & Software Quality™

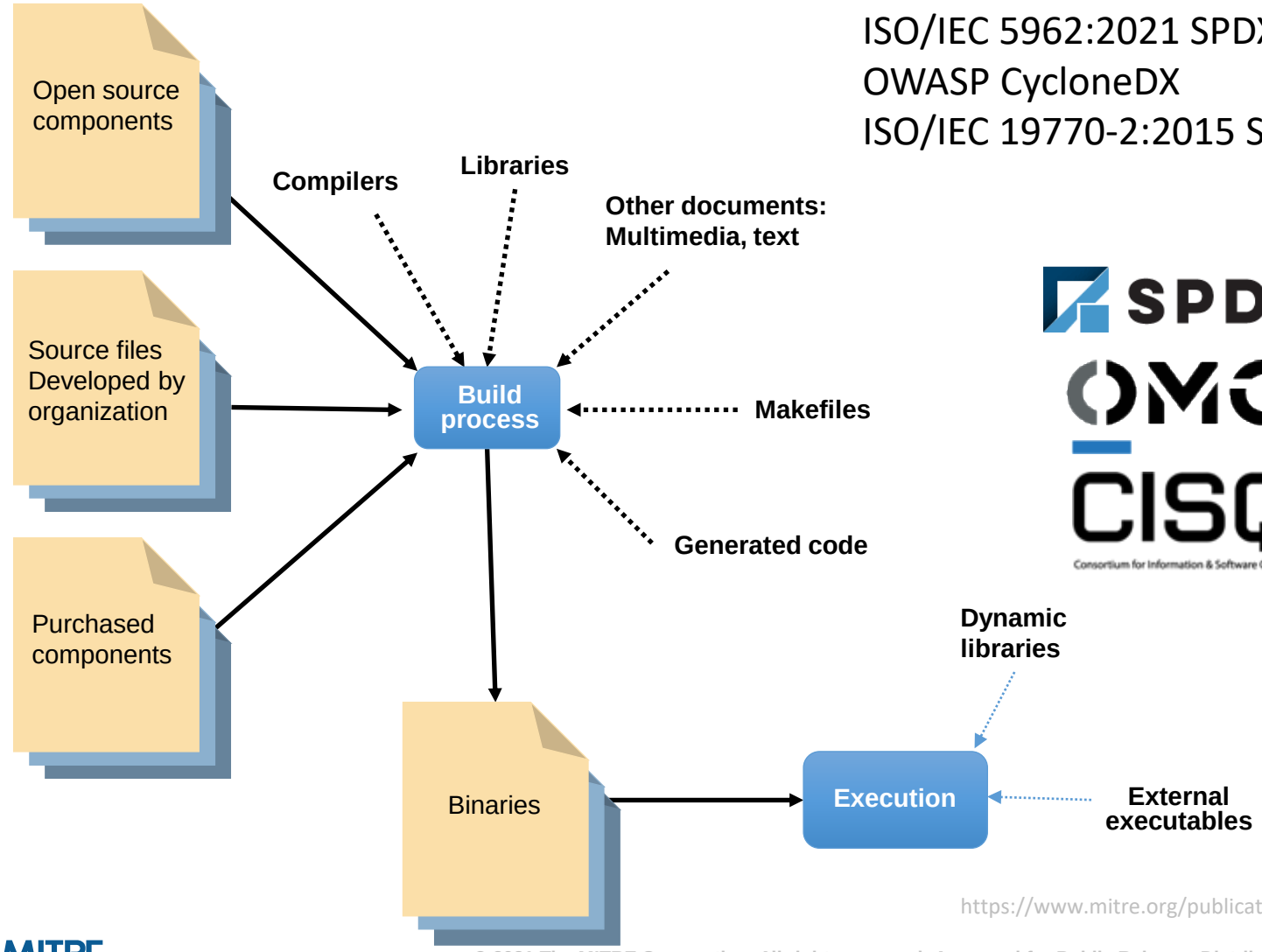
Software Supply Chain Integrity

Jan 2021



https://www.mitre.org/sites/default/files/publications/pr-21-0278-deliver-uncompromised-securing-critical-software-supply-chains.pdf

Software Bill of Materials Standardization



ISO/IEC 5962:2021 SPDX®
OWASP CycloneDX
ISO/IEC 19770-2:2015 SWID tags

SPDX
OMG
CISQ
Consortium for Information & Software Quality™

Usage Scenarios Around SBOMs

- Refer, Transfer or Purchase (definition of what it is)
- Pedigree (history of how it was produced)
- Provenance (chain of custody of it)
- Integrity (cryptographic basis of unalteredness)
- Proper and Legal (conditions about its use)
- Known Sw Vulns (known fixes are applied to it)
- Assurance (safe-secure-resilient)
- SBoM of a SW Service (SBoM of sw delivering service)
- Supply Chain Sequence Integrity

https://www.mitre.org/publications/technical-papers/standardizing-sbom-within-the-sw-development-tooling-ecosystem

MITRE

sigstore

https://sigstore.dev/what_is_sigstore/

A non-profit service to improve the open source software supply chain by easing the adoption of cryptographic software signing, backed by transparency log technologies

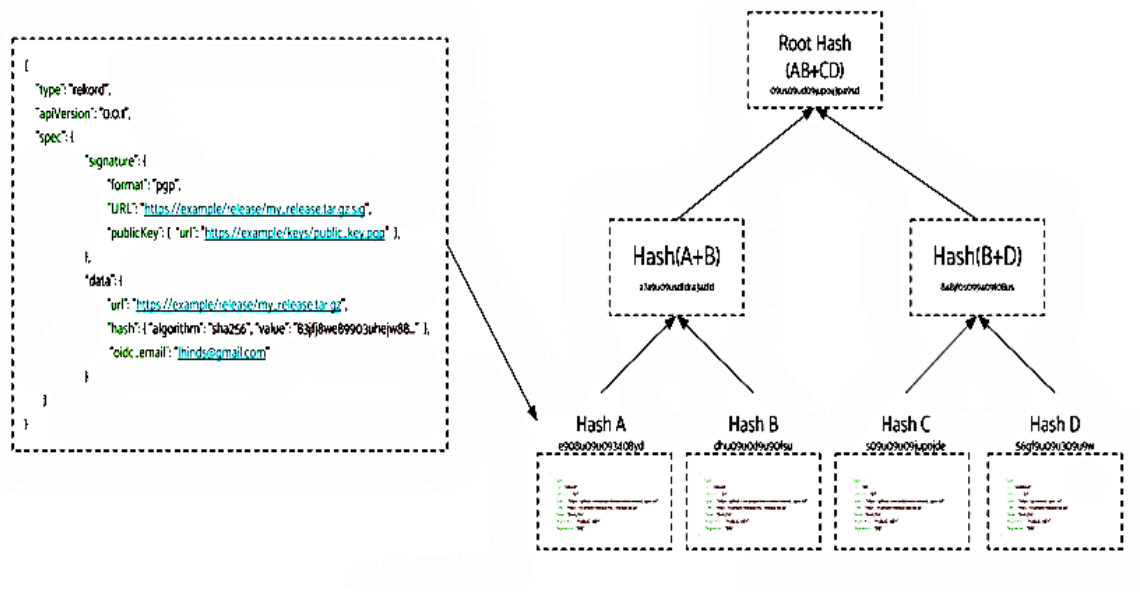
THE LINUX FOUNDATION

fulcio – free Root-CA for code signing certs

- issues certificates based on an OIDC email address.
- only signs short-lived certificates valid for under 20 minutes.

rekor – the binary transparency log project under sigstore

- client CLI (for adding an entry to a rekor transparency log)
- pluggable PKI and support present for: GPG, X.509, Minisign



sigstore manifests entry into the transparency log

cosign – Container Signing, Verification and Storage in an OCI registry.

- aims to make signatures **invisible infrastructure**.
- supports: Hardware and KMS signing, Bring-your-own PKI, OIDC PKI (Fulcio), Built-in binary transparency and timestamping service (Rekor)
- Tested/demonstrated with the following registries:
  1. AWS Elastic Container Registry
  2. GCP's Artifact Registry and Container Registry
  3. Docker Hub
  4. Azure Container Registry
  5. JFrog Artifactory Container Registry
  6. The CNCF distribution/distribution Registry
  7. Gitlab Container Registry
  8. GitHub Container Registry
  9. The CNCF Harbor Registry
  10. Digital Ocean Container Registry
  11. Sonatype Nexus Container Registry

MITRE

OCI Registry As Storage (ORAS)

https://github.com/oras-project

Tools and libraries to enable leveraging OCI registries for arbitrary artifacts

Open Container Initiative

https://github.com/opencontainers/

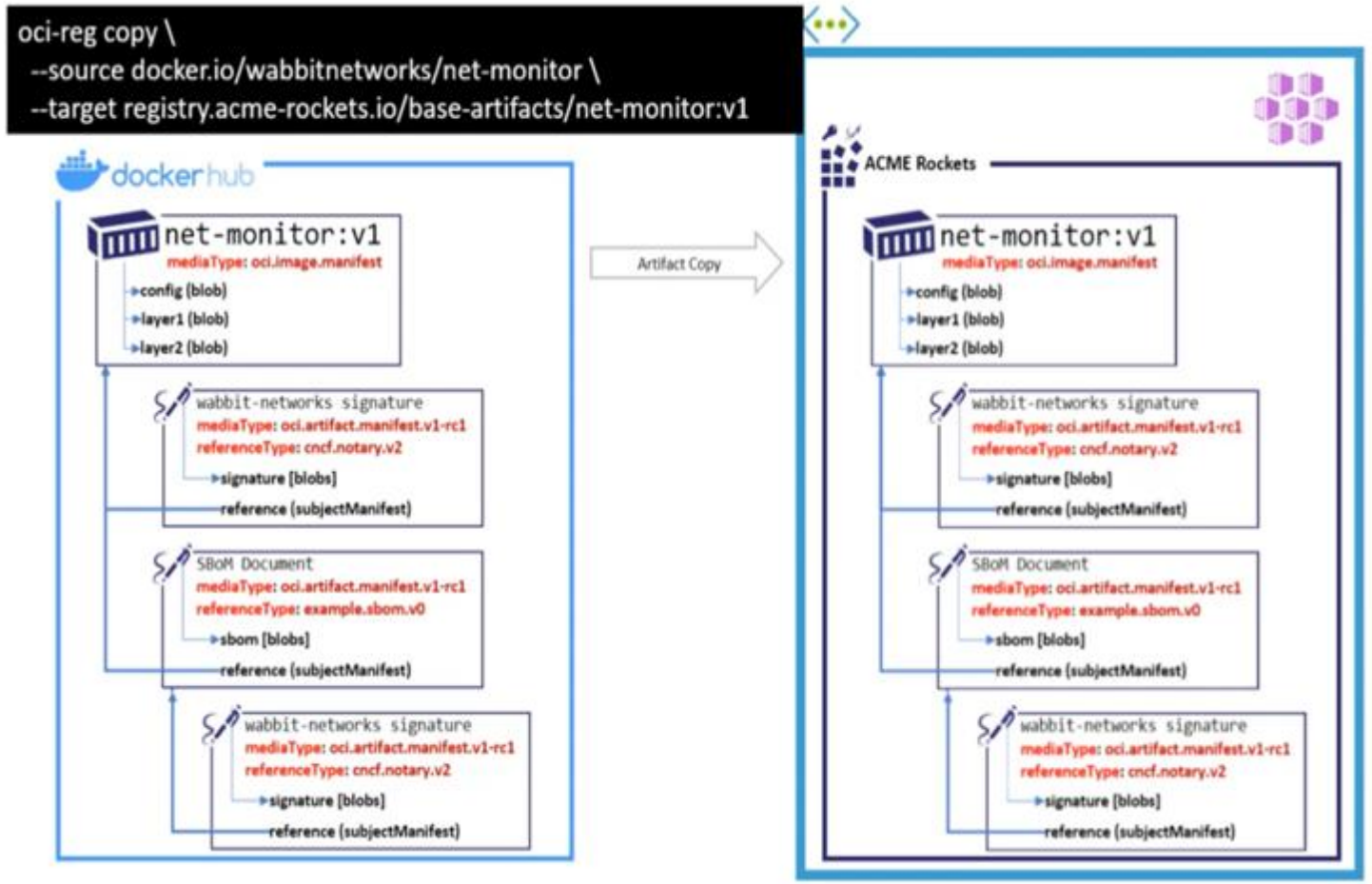
Creating open standards around container technology

OCI artifact manifest, Phase 1-Reference Types #29

The OCI artifact manifest generalizes the use of OCI image manifest, by reducing the constraints on all artifacts, enabling specific artifact-specs to set constraints for their type. Phase 1 adds support for artifacts to reference other artifacts through a subjectManifest property enabling reference graphs, as those required for secure supply chain efforts.

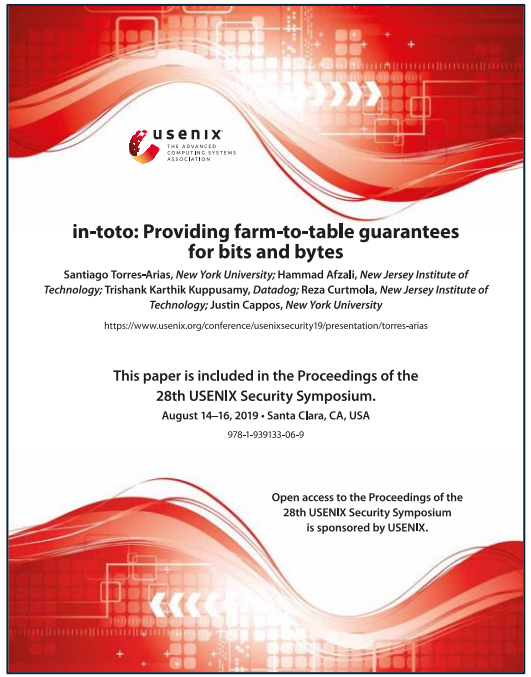
Phase 1: Reference Types

The PR focuses on Phase 1, enabling reference type support in 2021, supporting secure supply chain artifact types including signatures and SBoMs.



- OCI Artifacts Reference Types: github.com/opencontainers/artifacts/pull/29
- ORAS Reference Types: github.com/deislabs/oras/blob/reference-types/docs/artifact-manifest.md
- CNCF Distribution Reference Types: github.com/notaryproject/distribution/blob/prototype-2/docs/reference-types.md
- Notary v2: github.com/notaryproject/notaryproject

MITRE



Oct 2016+

in-toto

https://www.nist.gov/document/responses-enhancing-software-supply-chain-security-toto-team

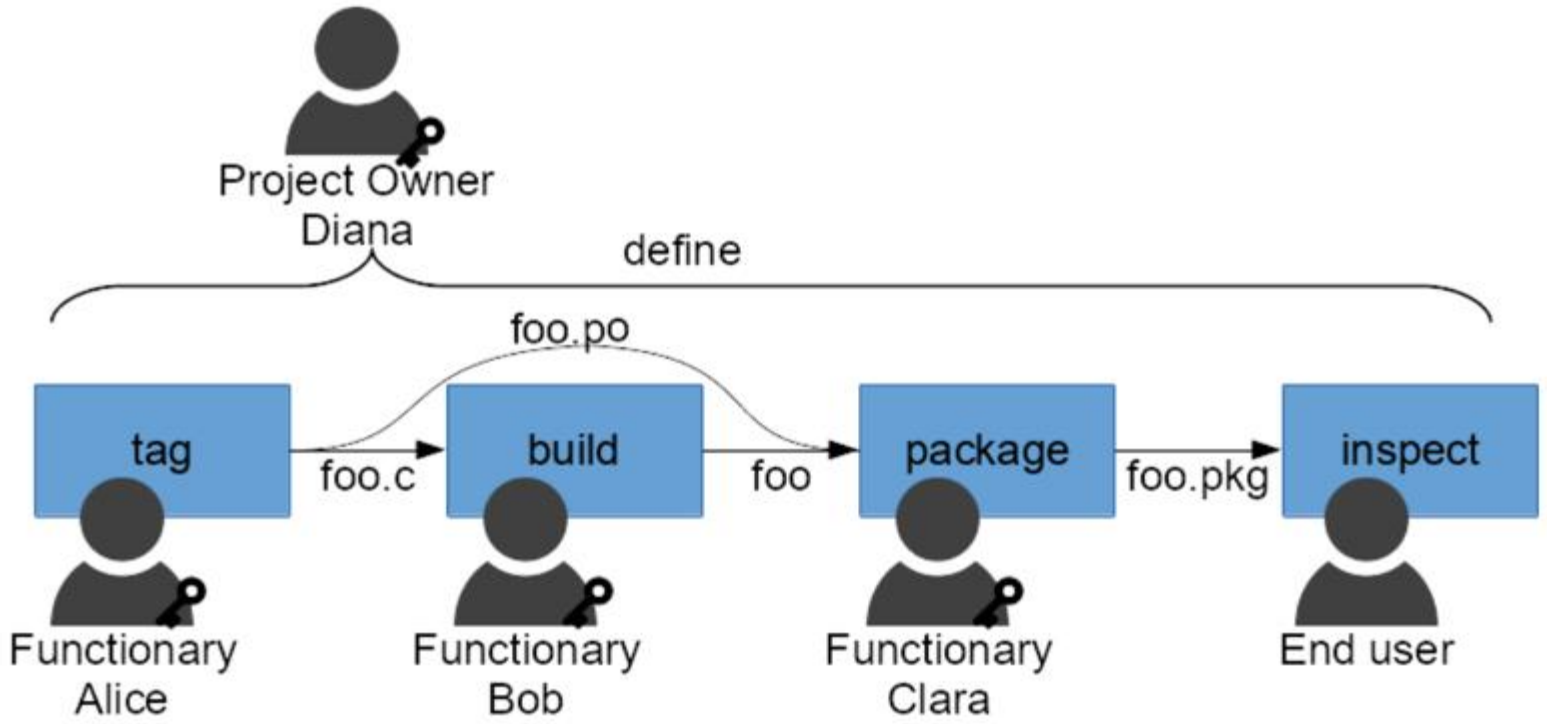


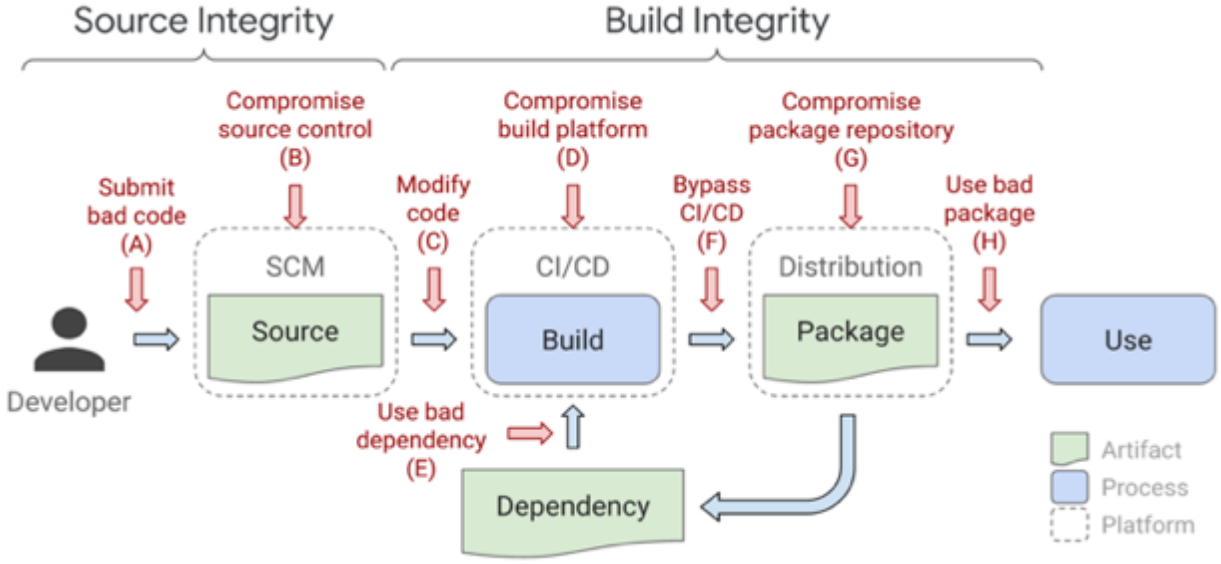
Figure 1: Graphical depiction of the software supply chain with `in-toto` elements added. The project owner creates a layout with three steps, each of which will be performed by a functionary. Notice how the tag step creates `foo.c` and a localization file `foo.po`, which are fed to different steps down the chain.

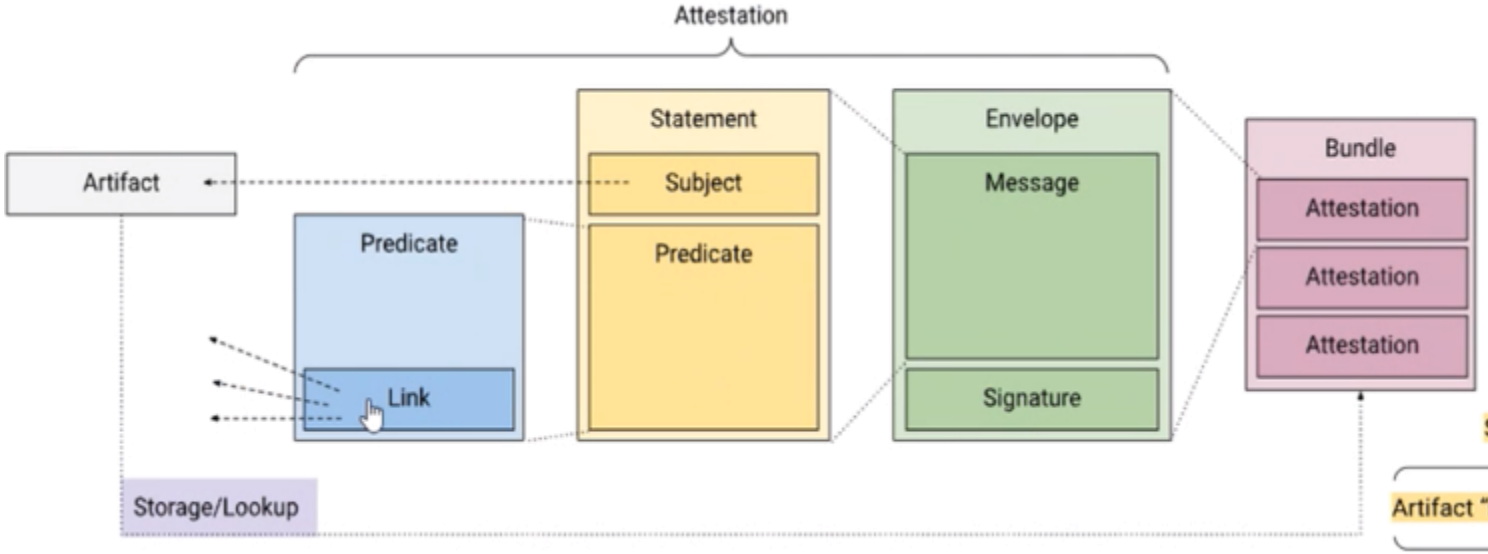
https://www.usenix.org/system/files/sec19-torres-arias.pdf

THE LINUX FOUNDATION

MITRE

Supply-chain Levels for Software Artifacts (SLSA)



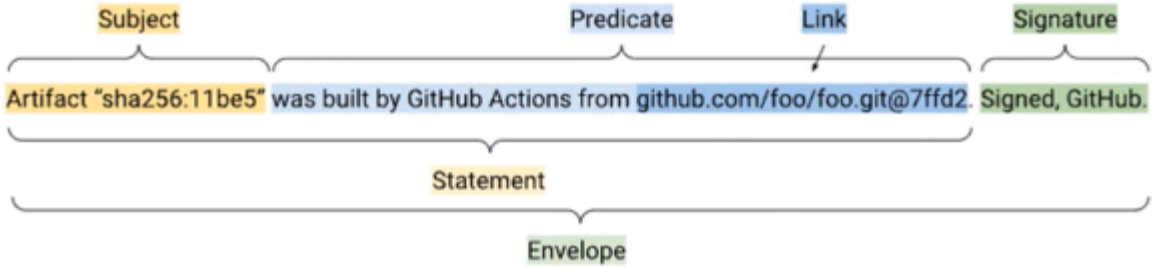


SLSA guidelines have 4 levels of incremental and actionable things that software producers can claim to do to protect against specific integrity attacks

https://github.com/slsa-framework/slsa

Requirement	SLSA 1	SLSA 2	SLSA 3	SLSA 4
Source - Version Controlled		✓	✓	✓
Source - Verified History			✓	✓
Source - Retained Indefinitely			18 mo.	✓
Source - Two-Person Reviewed				✓
Build - Scripted Build	✓	✓	✓	✓
Build - Build Service		✓	✓	✓
Build - Ephemeral Environment			✓	✓
Build - Isolated			✓	✓
Build - Parameterless				✓
Build - Hermetic				✓
Build - Reproducible				○
Provenance - Available	✓	✓	✓	✓
Provenance - Authenticated		✓	✓	✓
Provenance - Service Generated		✓	✓	✓
Provenance - Non-Falsifiable			✓	✓
Provenance - Dependencies Complete				✓
Common - Security				✓
Common - Access				✓
Common - Superusers				✓

○ = required unless there is a justification



Supply Chain Integrity Model (SCIM)

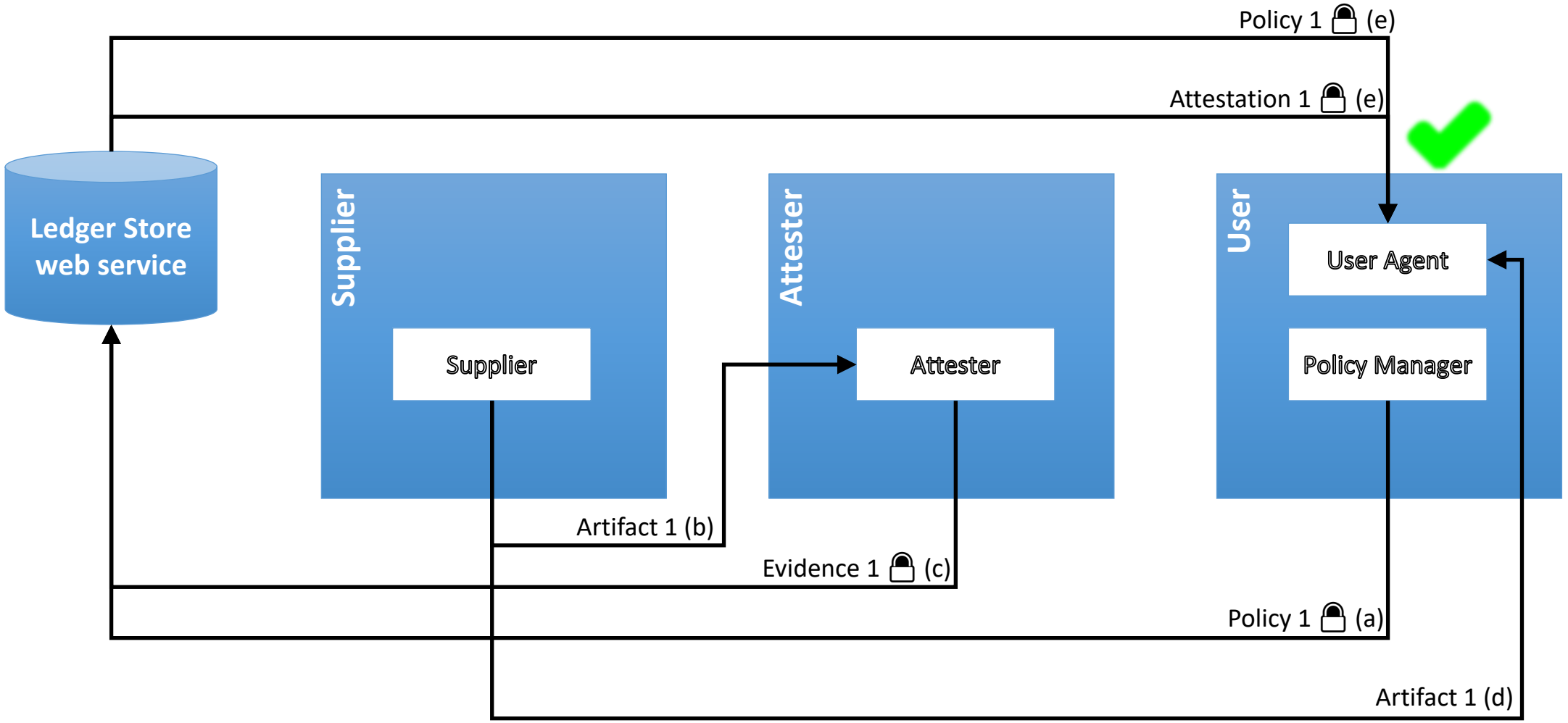
Technologies leveraged:

- **Attestations/Evidence, Confidential Ledgers, Hardware Roots of Trust, BOMs, CBOR (RFC 8949) and COSE (RFC 8152)**

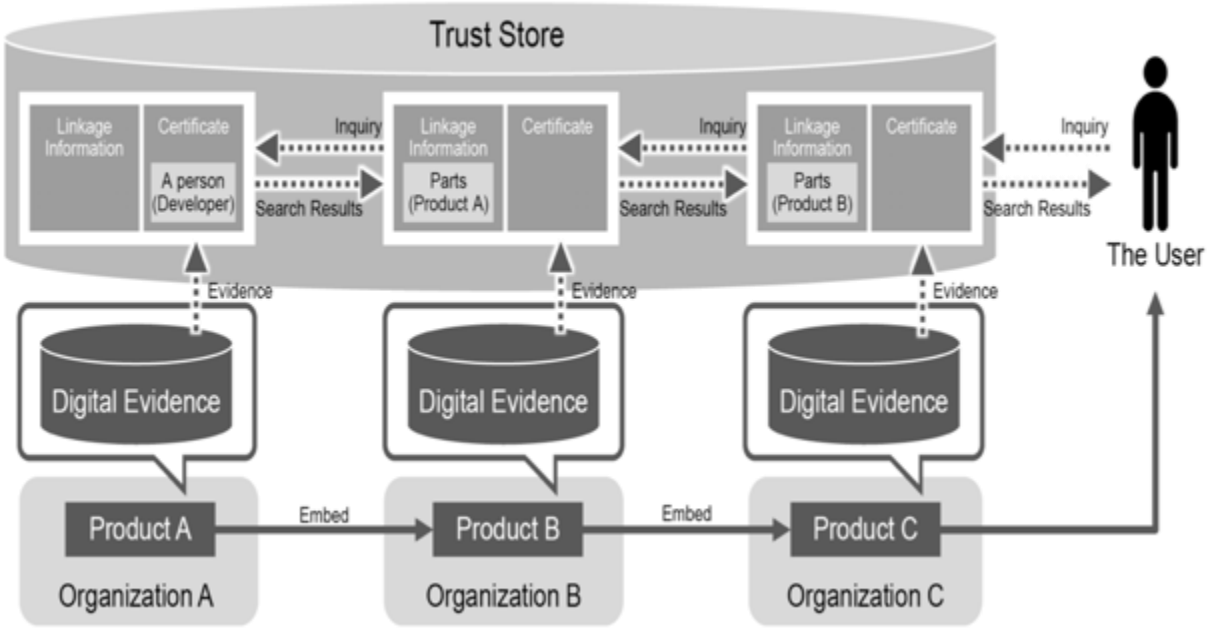
SCIM:

- **defines minimum standards around the:**
  - **preparation, storage, distribution, consumption, validation and evaluation of arbitrary attestations/evidence about artifacts that are critical to maintaining the integrity of supply chains**
- **specifies an end-to-end system for validating arbitrary artifacts in terms of supply chains whose integrity has been proven.**
- **is applicable to both hardware (objects in the physical world) and software (digital) artifacts.**
- **does not define how artifacts are produced or distributed, nor the methods by which attestations/evidence about artifacts are produced prior to preparation for inclusion in SCIM.**

SCIM Usage Scenario



Trust Systems for a Supply Chain



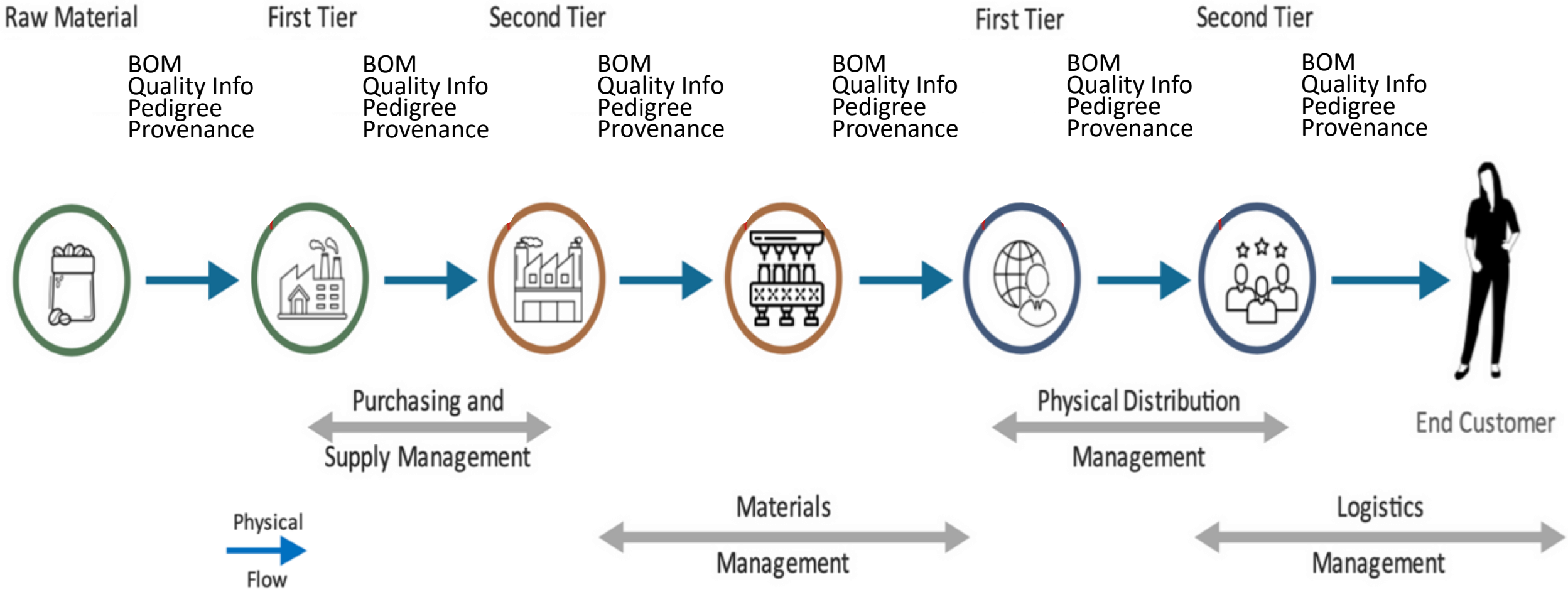
https://www.iiconsortium.org/pdf/Trustworthiness_Framework_Foundations.pdf

HITACHI

Conformance Validation	This validates whether production activities were performed correctly as stated in the corresponding regulations.
Digital Evidence	This evidence is used to explain to a third party that production activities are performed correctly.
Trust Store	This stores the Trust of companies and is connected via the supply chain.

https://www.hitachi.co.jp/products/it/security/activities/digitaltrust/english/index.html

Supply Chains – As multi-Stakeholder Network



https://www.iiconsortium.org/pdf/Trustworthiness_Framework_Foundations.pdf

SOFTWARE SUPPLY CHAIN TRANSPARENCY

Robert Martin

Sr. Software and Supply Chain Assurance Principal Eng., MITRE

Allan Friedman

Senior Advisor & Strategist, CISA

CISQ

Consortium for Information & Software Quality ™

KEYNOTE:

MODERNIZATION AND DEVOPS BEST PRACTICES AT AMAZON

PRESENTED BY:

LEO ZHADANOVSKY
CHIEF TECHNOLOGIST, US EDUCATION
AMAZON WEB SERVICES

CISQ
Consortium for Information & Software Quality™

Who am I?

- Worked for one of the biggest AWS customers in 2012
- 8 years at AWS
  - Ensure some of our biggest customer launches go smoothly
  - Help customers build modern applications
  - Work with customers and AWS service teams to meet our customer’s current and future needs

Today, we will cover

- Lessons learned at Amazon for developing and delivering software
- How to modernize your applications to take full advantage of the cloud
- How an AWS customer was able to scale successfully, despite extreme increases of demand caused by the pandemic
- Q&A

AWS in the public sector

7,500+

Government agencies

14,000+

Educational institutions

35,000+

Nonprofit organizations

Government, education, and nonprofit organizations are using AWS for digital transformation

We had three big ideas at Amazon that we have stuck with for 20+ years, and they are the reason we are successful: **put the customer first, invent on our customers behalf, and be patient.**

Jeffrey P. Bezos
Founder
Amazon.com, Inc.

Customer Case Study: Blackboard

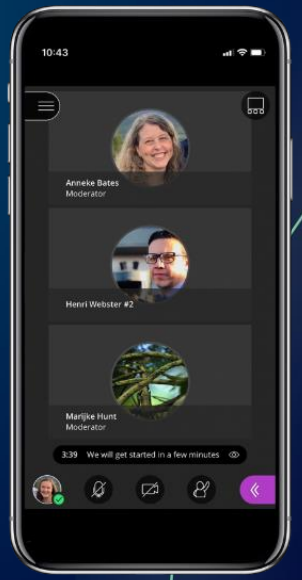
Blackboard is a leading EdTech company, serving higher education, K–12, business, and government clients in every region of the world

Blackboard connects a deep understanding of education with the power of technology to continuously push the boundaries of learning

aws

Blackboard virtual classrooms

Collaborate



Adapting to unpredictable spikes in traffic

- March – Start of school year in southern hemisphere
- March 7th ,2020
  - “We must stop, contain, control, delay and reduce the impact of this virus at every opportunity.” – World Heath Organization
- **4800%** increase in usage on Collaborate compared to pre-pandemic

"We had entire countries shifting to online learning overnight … not only did we have to accommodate the increased usage, but we also had to support the institutions as they shifted their entire paradigm from onsite to online learning."

Kris Stokking

VP of Software Engineering

Blackboard

How did Blackboard scale?

- At first, overprovision
- Next – autoscaling
- Diversify compute
- Let AWS handle undifferentiated heavy lifting
- Partner with AWS
  - Support
  - Solutions Architecture
- Learn more here: https://aws.amazon.com/solutions/case-studies/blackboard-ec2-case-study

"AWS provides value in a way that empowers Blackboard to really focus on its core value proposition … Blackboard is more agile and better equipped to deal with change because we're on AWS."

Kris Stokking

VP of Software Engineering

Blackboard

Amazon’s Modernization and DevOps Journey

One area where I think we are especially distinctive is failure. I believe we are the best place in the world to fail (we have plenty of practice!), and **failure and invention are inseparable twins**. To invent you have to experiment, and if you know in advance that it's going to work, it's not an experiment.

Jeffrey P. Bezos
Founder
Amazon.com, Inc.

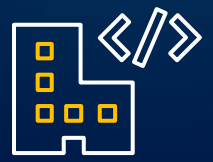
Lessons learned at Amazon for developing and delivering software

Decompose for agility
(microservices, 2-pizza teams)

Automate everything

Standardized tools

Belts and suspenders
(governance, templates)

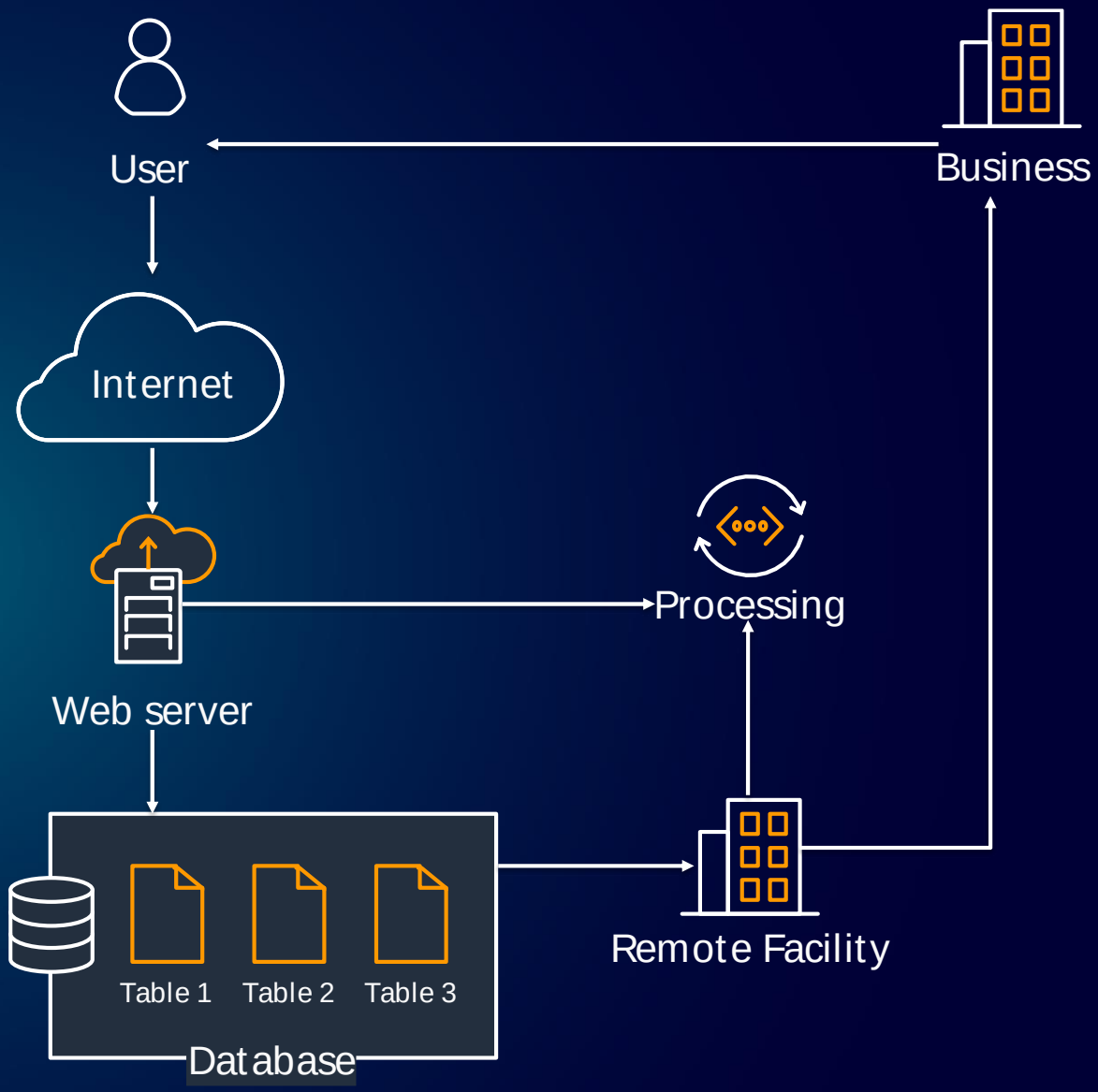


Infrastructure as code

Just starting out

This is how many web architectures started out, and it's how Amazon started too…

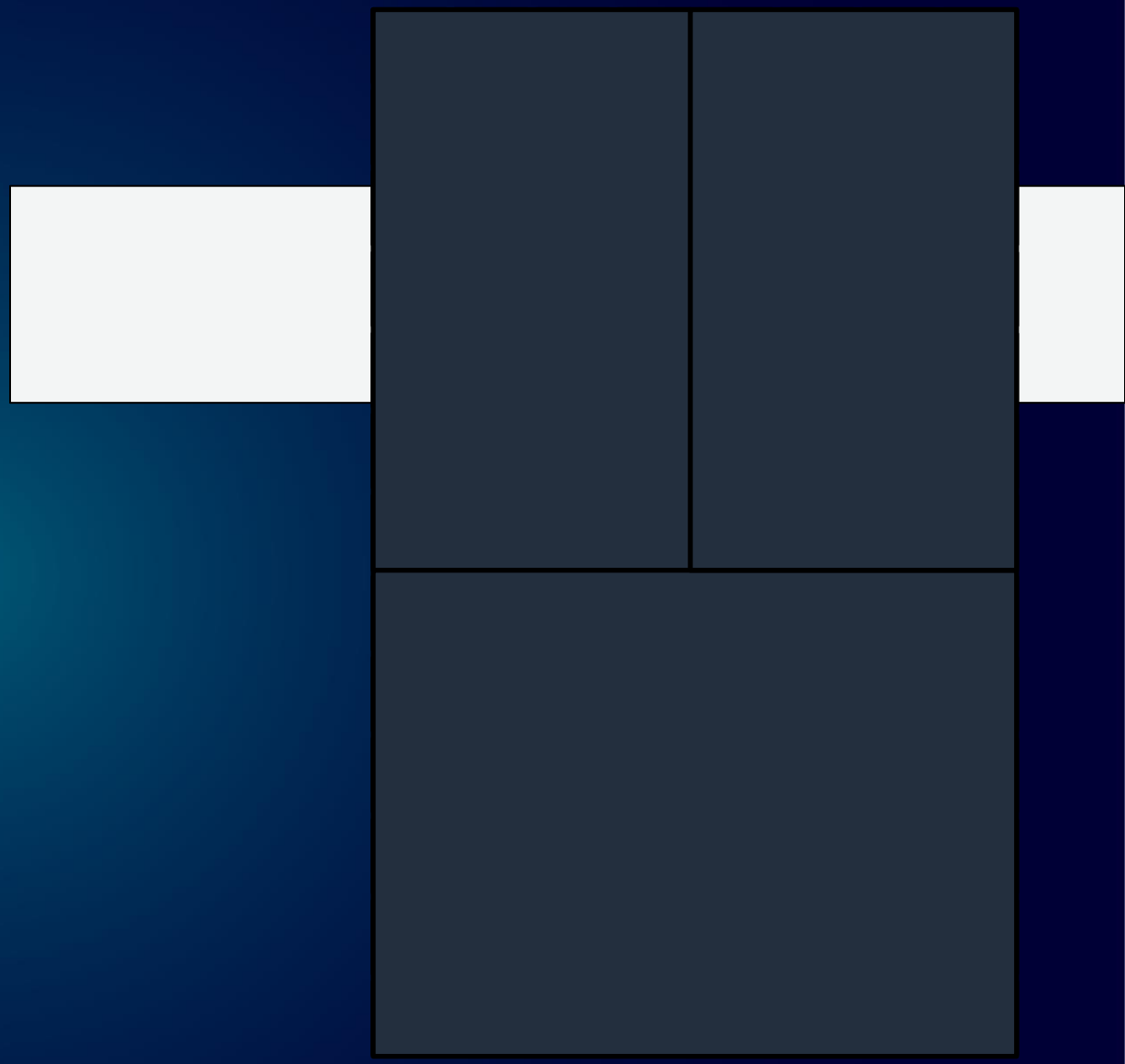
There any many bottlenecks, and scaling of the web server was an immediate factor



Going further

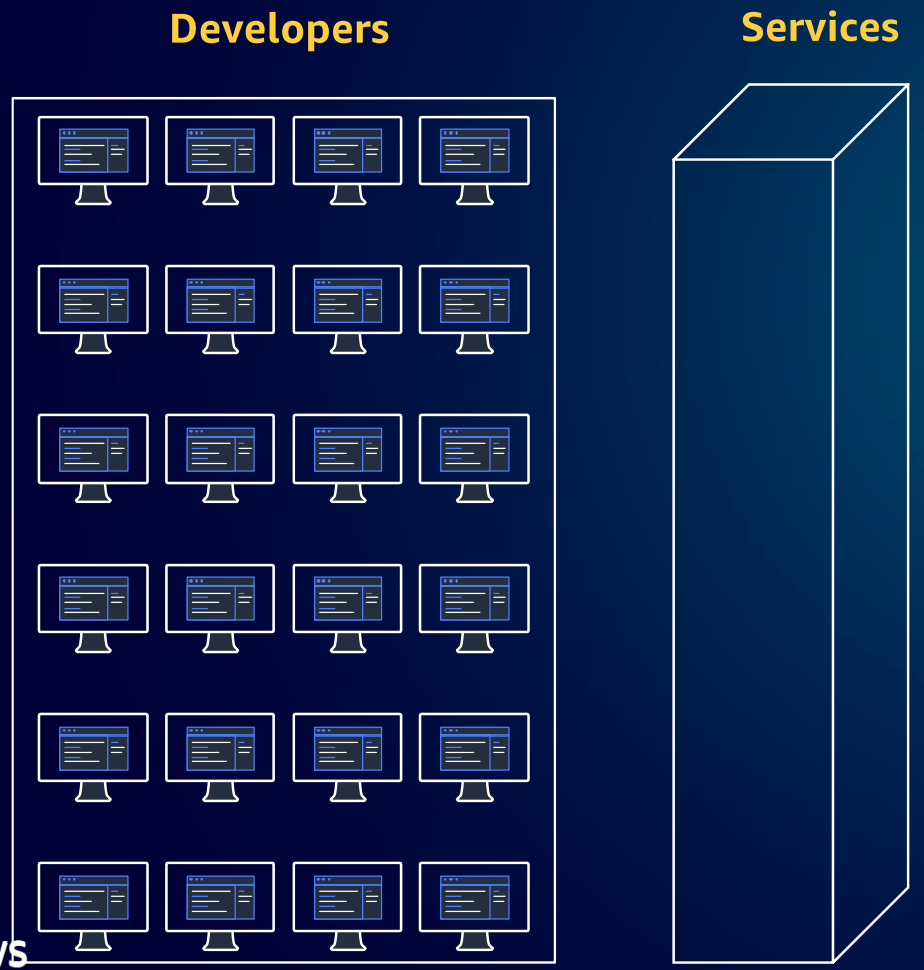
Principles

- Make units a small as possible (Primitives)
- De-couple based on scaling factors, not functions
- Each service operates independently
  "Communication is terrible!" —Jeff Bezos
- APIs (contracts) between services



Impact to our development

Monolith development lifecycle



Delivery pipelines

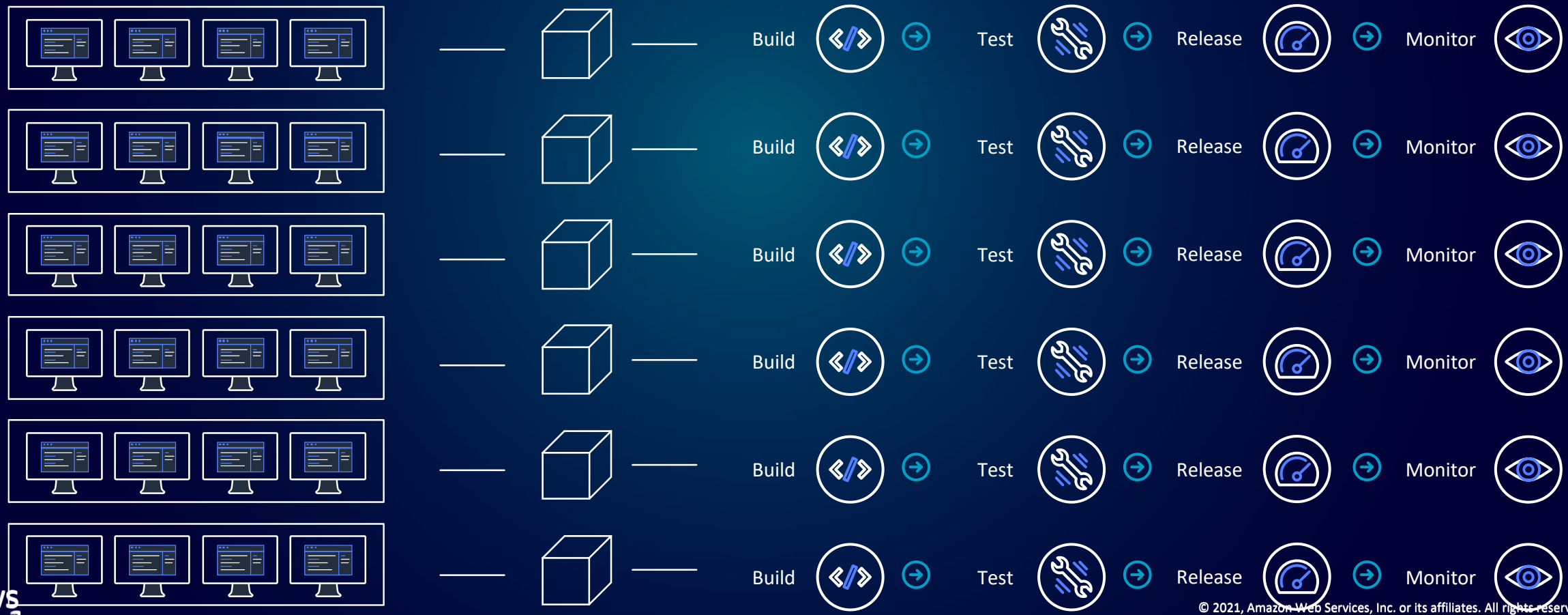
Build → Test → Release → Monitor

Monolith development lifecycle

This led to changes in organization

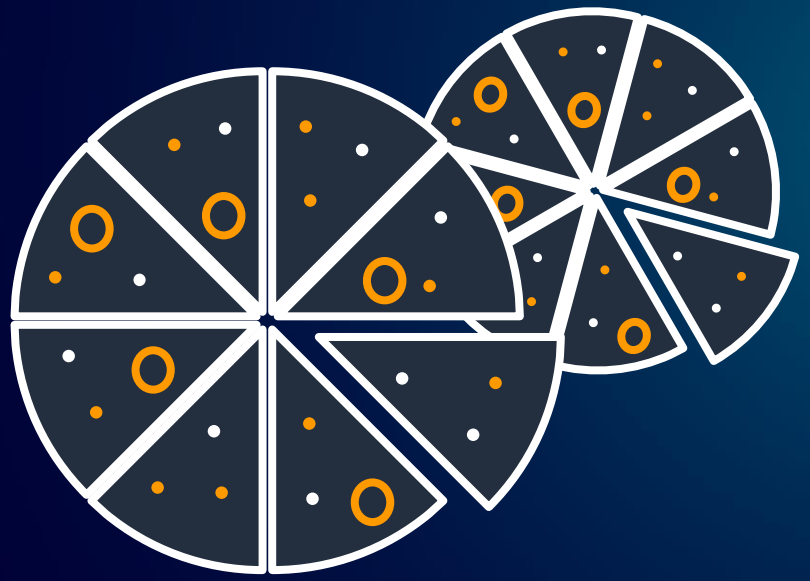
Developers

Services



Impact to our organization

Getting (re)organized

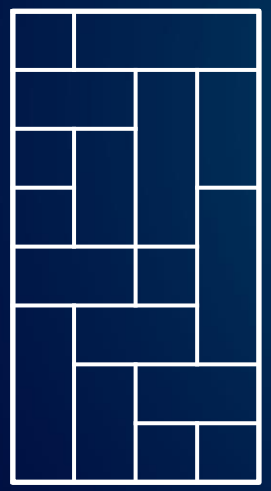


“Two-pizza” teams

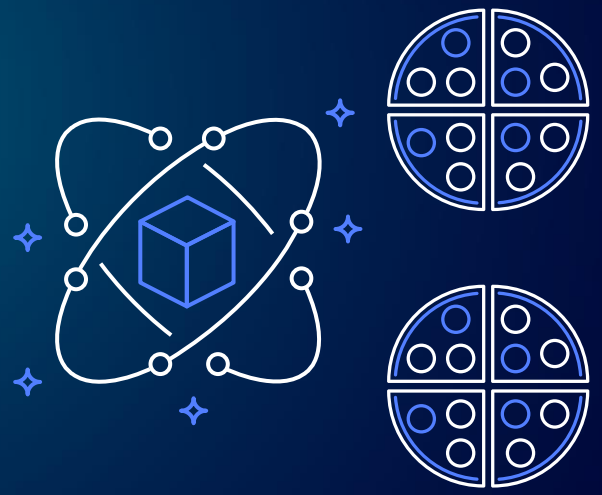
- Own a service
- Minimizes social constraints (Conway’s law)
- Autonomy to make decisions

Transformation timeline

2001 → **2002**



Monolithic application + teams



Microservices + 2-pizza teams

Teams Own Everything

- Planning
- Security
- Performance
- Scalability
- Deployment

- Operation
- Bugs
- Documentation
- Testing…

Now we have…

Modern applications

Today we have modern applications

- Use independently scalable microservices (serverless, containers...)
- Connect through APIs
- Deliver updates continuously
- Adapt quickly to change
- Scale globally
- Are fault tolerant
- Carefully manage state and persistence
- Have security built-in

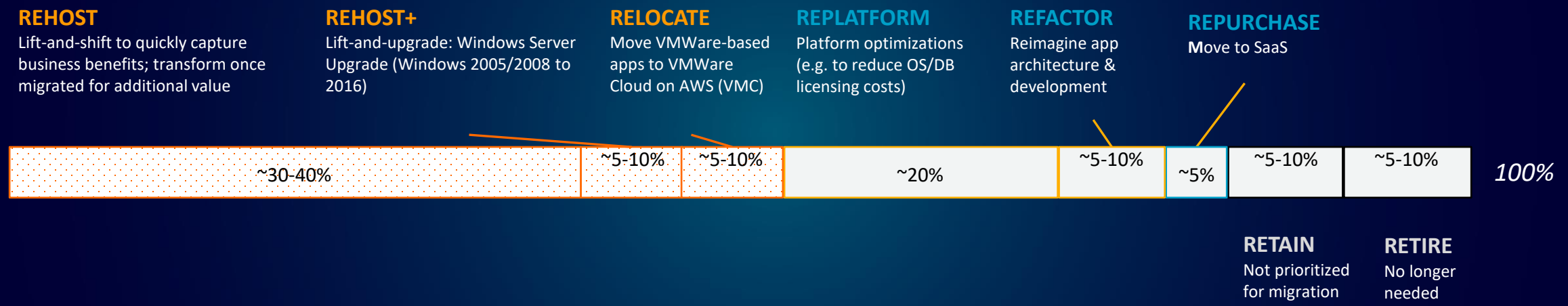
Modernization

"How do I get to modern applications?"

Migration and Modernization Patterns

AVERAGE CUSTOMER ENVIRONMENT, BY MIGRATION PATTERN

(based on AWS experience)



Full spectrum of patterns is important for transformation – but up to ~60% of typical environment can be rapidly migrated at a predictable price, freeing time & budget to focus on modernization

Modernization Pathways

Move to Cloud Native Architecture

Agile, scalable apps built on containers, serverless and microservices

Move to Managed Cloud Services

Deploy applications rapidly and operate reliably at scale with managed services

Move to Managed Databases

Open source, fit for purpose, highly scalable databases

Move to Open Source

Freedom from proprietary licensed software with open source technology

Martin Fowler's Strangler Pattern

"…gradually create a new system around the edges of the old, letting it grow slowly over several years until the old system is strangled."

Martin Fowler
June 29, 2004

Success Stories

Built an image processing solution in just days using AWS serverless. The solution processed 50 million images in 8 days for $6,000 and is now saving more than $100,000 per year.

Re-architected on-premises Hadoop cluster to AWS serverless in 3 months; increased cost efficiency by 2x while handling half a trillion stock trade validations a day, improving security and compliance

Modernized and built an entirely serverless website with half the team size normally required to build and operate at their scale. Freed up engineers to focus on building out new features and innovating, and realized 84% cost savings.

Improved system uptime from 85% to 99.99% after migrating its finance reporting system from on-premises to SAP S/4HANA on AWS.

Increased SAP ERP response times by 40% while lowering costs to support supply chain initiatives.

Q&A

Thank you!

Leo Zhadanovsky

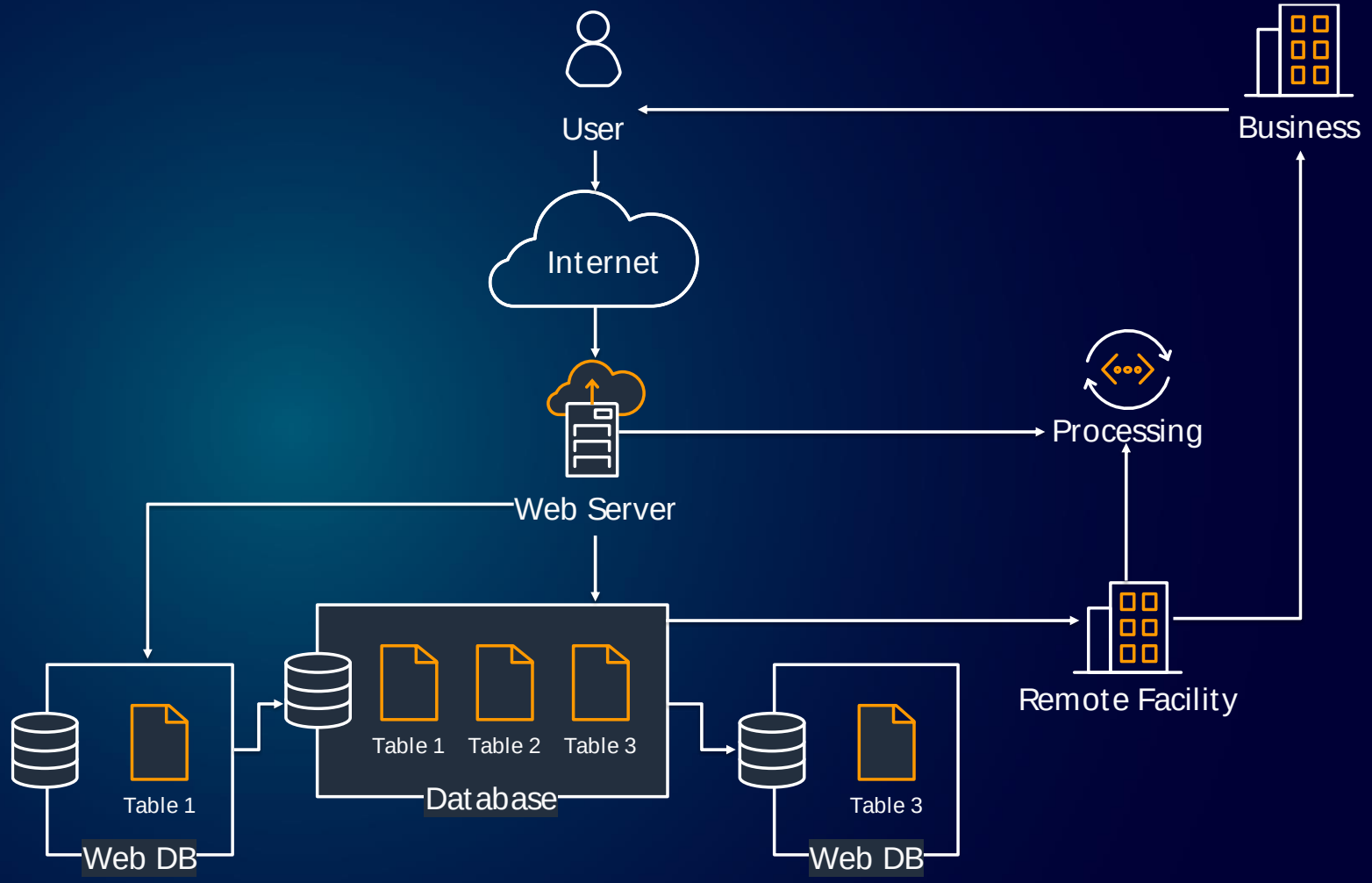
Our mission

- Our task was to improve:
  - Innovation
  - Speed
  - Agility
  - Safety
- What we did:
  - Decomposed for agility
  - Cultural and operational shift
  - Created tools for software delivery

Scaling v1

In 1998 the “Distributed Computing Manifesto” came out and we began breaking things down into separate components…

This was a bit better, still not very scalable



TESTING FOR DATA PRIVACY AND PROTECTION

PRESENTED BY:

Consortium for Information & Software Quality™

ENSURING SECURE & RESILIENT IT MODERNIZATION OUTCOMES

CISQ

Consortium for Information & Software Quality ™

PRESENTED BY:
DAVID POWNER, EXECUTIVE DIRECTOR
CENTER FOR DATA-DRIVEN POLICY,
MITRE

SETH CARMODY

Why IT Modernization is Important

- Strengthens security posture
- Reduces technical debt
- Advances mission outcomes and citizen services
- Positions us for future scalability and maintainability

Recent History Calling for Attention

2016-2017

- 2016: GAO releases report on modernizing IT legacy systems
- 2017: Report to the President on Modernization
- 2017: NDAA MGT Act

2018-2020

- 2018: President's Management Agenda (3 Priorities)
- 2019: GAO releases their second report
- 2020: Senator Hassan questions to 10 federal agencies

2021

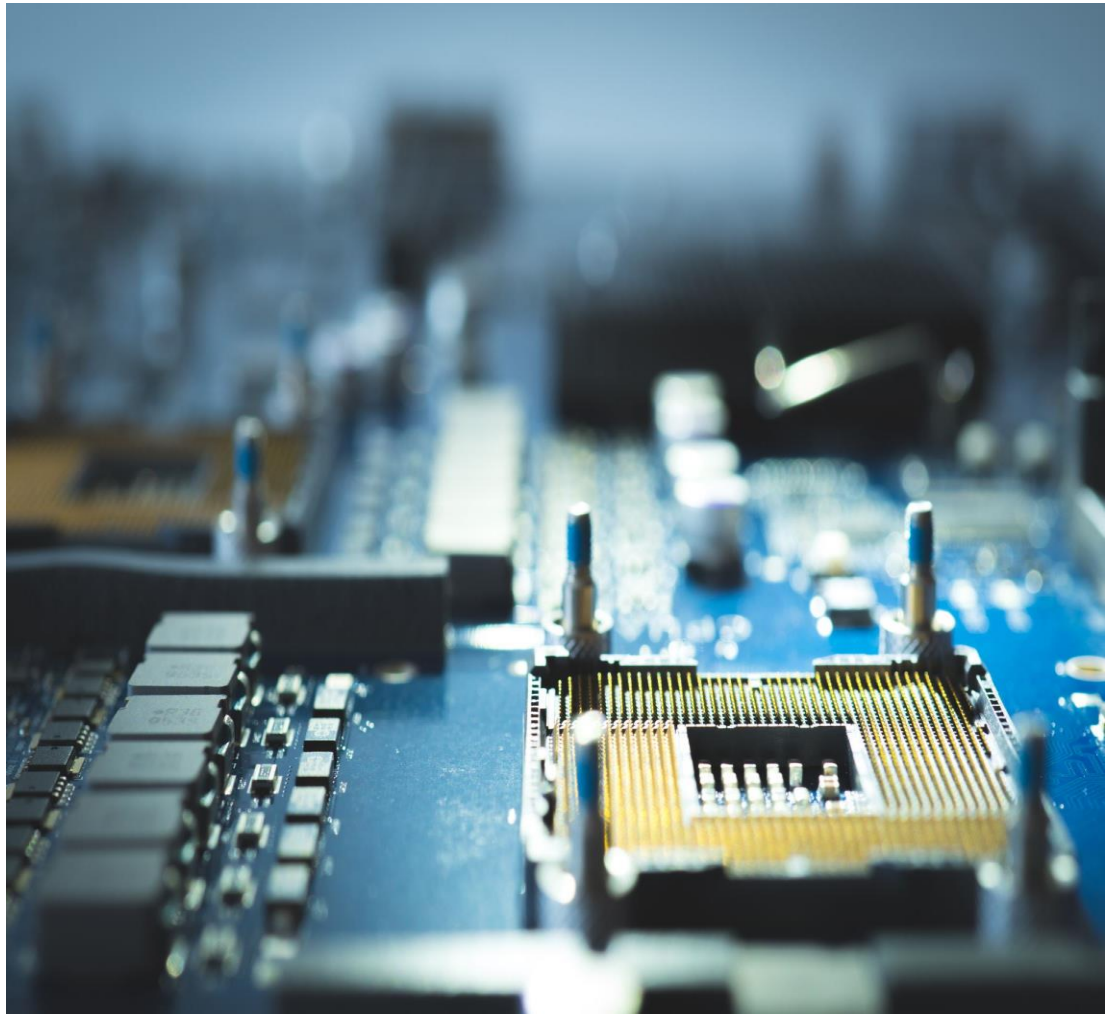
- Senate Hearings
- PMA?
- Legacy Reduction Act Legislation?

Focus and Results to Date

- Business System modernization
- Cloud Migration
- Limited Mission Critical Legacy System Modernization
  - Wartime readiness
  - Tax Processing
  - Benefit programs

Path Forward

- Move beyond just identifying Legacy Systems
- Prioritize – criteria/MITRE recent research
- Plans/transparency/accurate budgets
- Progress against plans
  - Business ownership/SW SCRM/ ISO 5055/CX
- OMB help

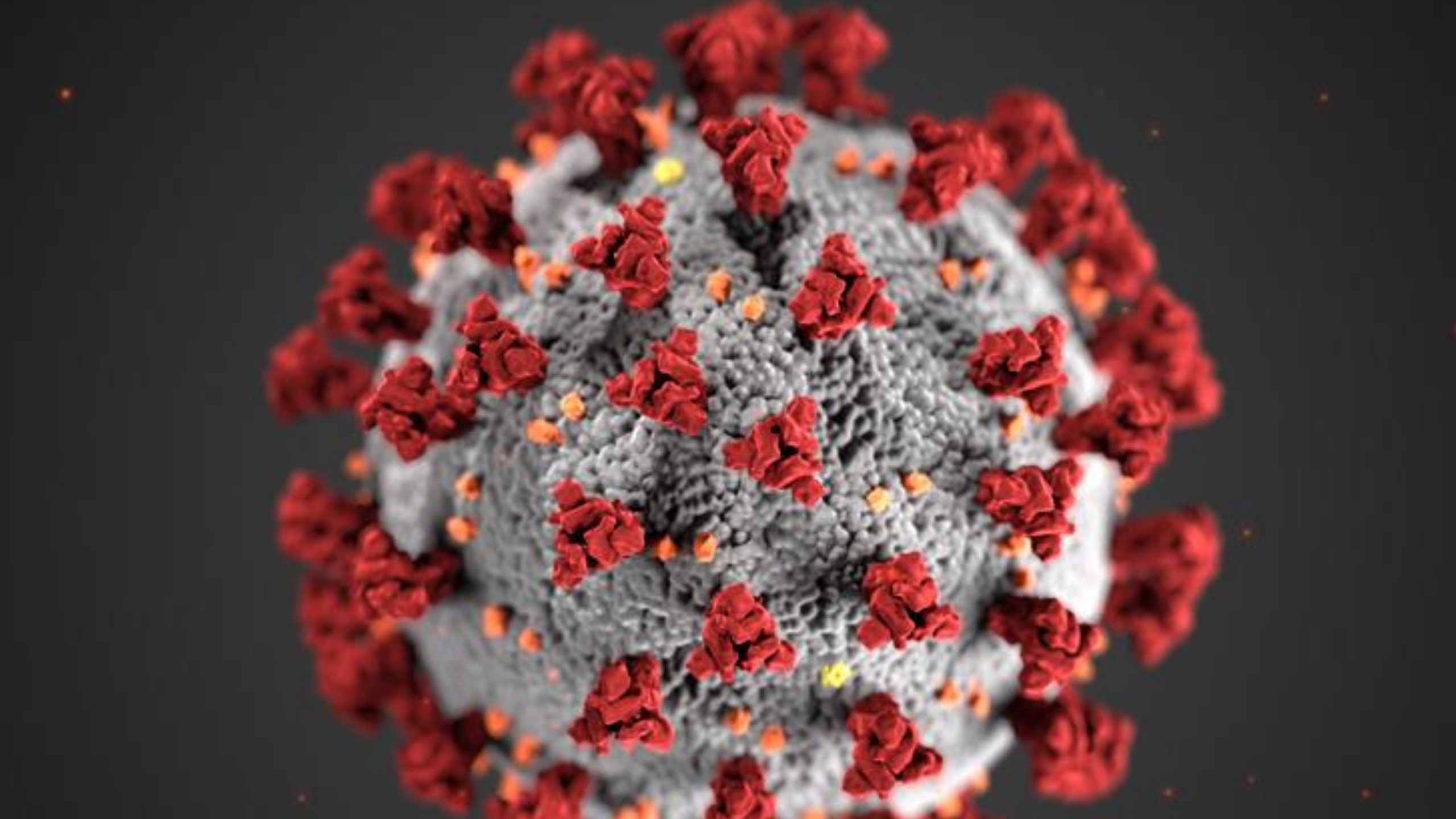


Why Healthcare Cybersecurity is Hard

medcrypt

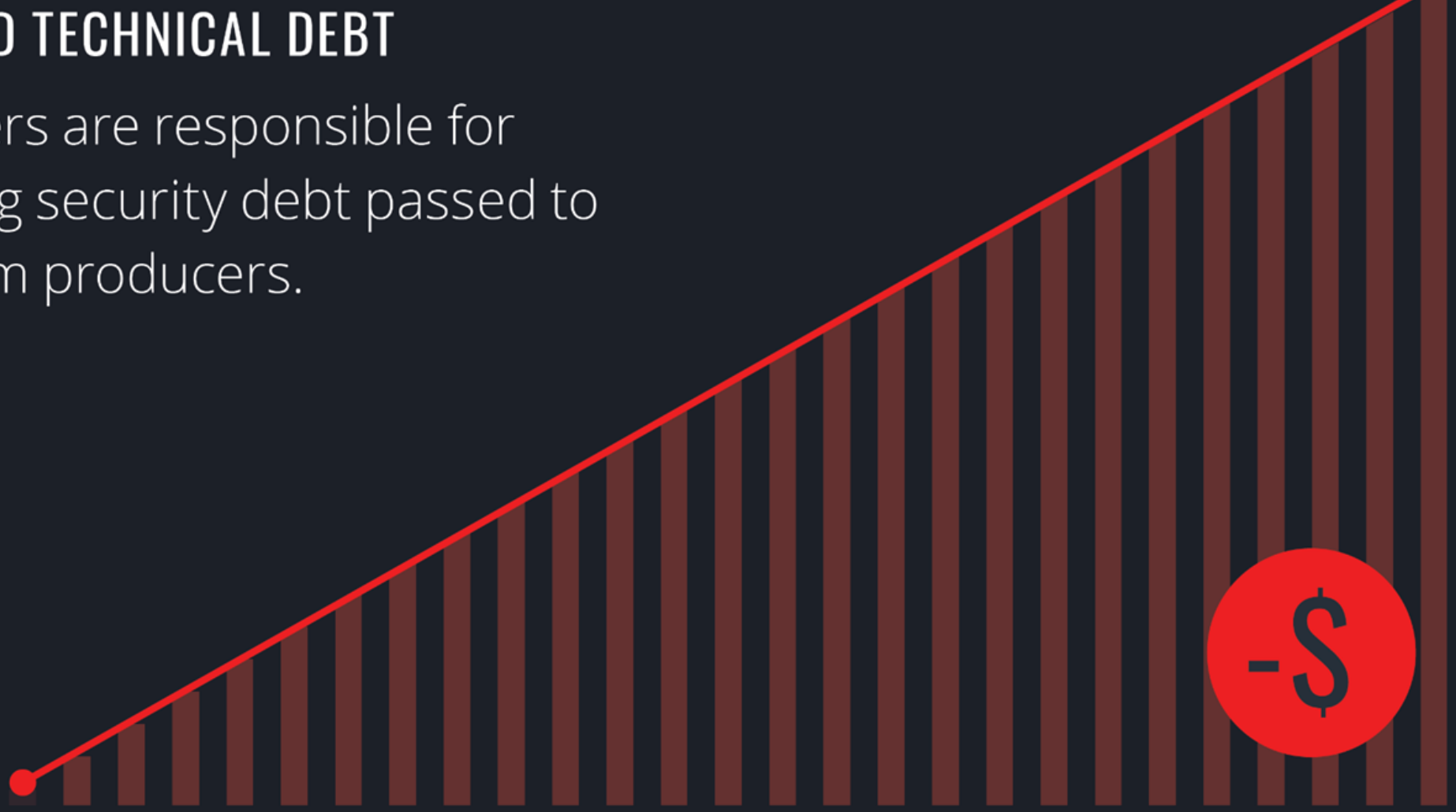
Seth Carmody, PhD
VP Regulatory Strategy

CISQ Cyber Resilience Summit
October 12, 2021



INCREASED TECHNICAL DEBT

Consumers are responsible for managing security debt passed to them from producers.



TECHNICAL

PRODUCERS

CONSUMERS

No investment | Tech | MDM | HDO | Clinicians | Patients

REGULATORS: FDA FDO & Congress

FIERCE Healthcare

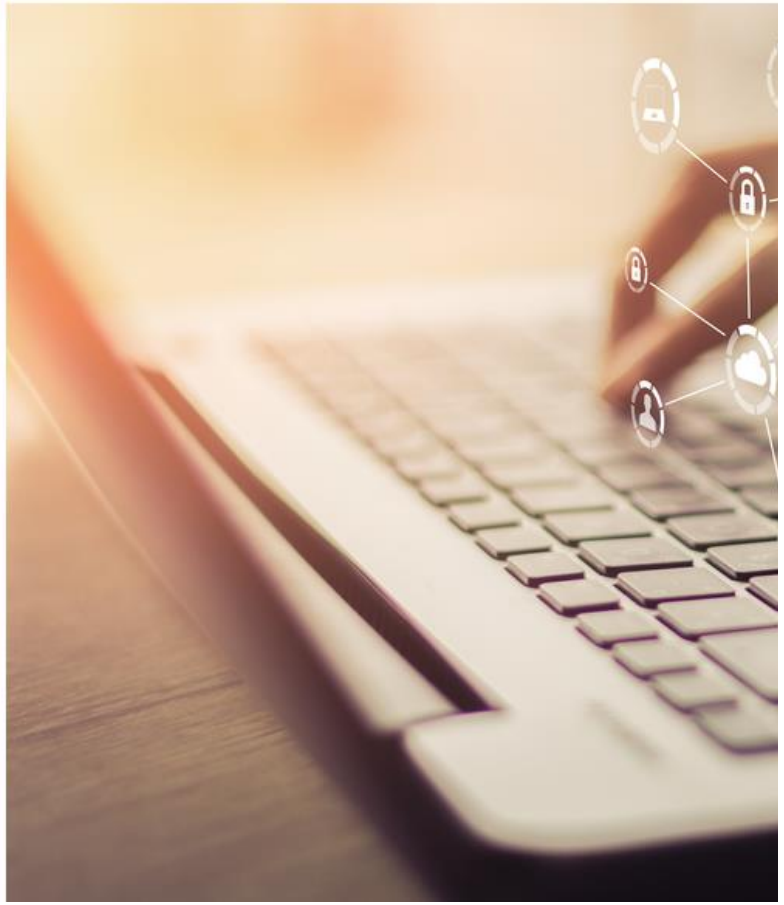
HOSPITALS TECH PAYER FINANCE PRACTICES DIGI

Virtual Events Jobs Resources

Hospitals

May cyberattack cost Scripps nearly $113M in lost revenue, more costs

by Robert King | Aug 11, 2021 3:55pm



RiskBased SECURITY

2021 Mid Year Report

Data Breach QuickView

Global Edition Privacy & Security

Nevada hospital ransomware attack could affect data of 1.3M patients

An Ohio-based law firm is investigating claims on behalf of the breach victims.

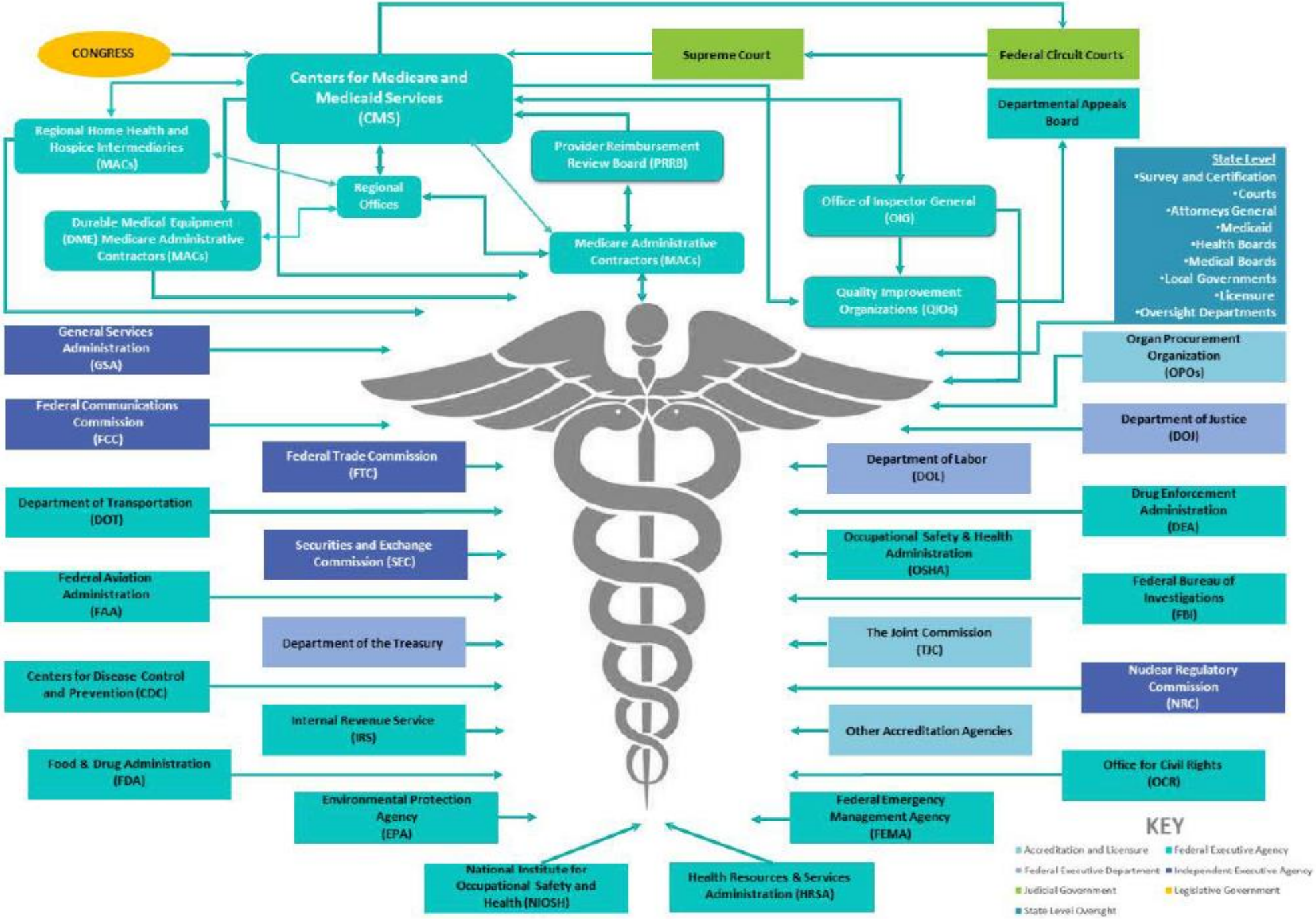
By Kat Jercich | August 23, 2021 | 05:02 PM

Photo: "Welcome to Nevada," James Cridland/Flickr, licensed under CC BY 2.0


UNITED STATES
ENVIRONMENTAL PROTECTION AGENCY

U.S. FOOD & DRUG ADMINISTRATION


CONGRATULATIONS
FLAMMABLE
KEEP FIRE AWAY


CONGRESS
Supreme Court
Federal Circuit Courts
Centers for Medicare and Medicaid Services (CMS)
Departmental Appeals Board
Regional Home Health and Hospice Intermediaries (MACs)
Provider Reimbursement Review Board (PRRB)
Regional Offices
Durable Medical Equipment (DME) Medicare Administrative Contractors (MACs)
Medicare Administrative Contractors (MACs)
Office of Inspector General (OIG)
Quality Improvement Organizations (QIOs)
State Level
•Survey and Certification
•Courts
•Attorneys General
•Medicaid
•Health Boards
•Medical Boards
•Local Governments
•Licensure
•Oversight Departments
General Services Administration (GSA)
Organ Procurement Organization (OPOs)
Federal Communications Commission (FCC)
Department of Justice (DOJ)
Federal Trade Commission (FTC)
Department of Labor (DOL)
Department of Transportation (DOT)
Drug Enforcement Administration (DEA)
Securities and Exchange Commission (SEC)
Occupational Safety & Health Administration (OSHA)
Federal Aviation Administration (FAA)
Federal Bureau of Investigations (FBI)
Department of the Treasury
The Joint Commission (TJC)
Centers for Disease Control and Prevention (CDC)
Nuclear Regulatory Commission (NRC)
Internal Revenue Service (IRS)
Other Accreditation Agencies
Food & Drug Administration (FDA)
Office for Civil Rights (OCR)
Environmental Protection Agency (EPA)
Federal Emergency Management Agency (FEMA)
National Institute for Occupational Safety and Health (NIOSH)
Health Resources & Services Administration (HRSA)
KEY
Accreditation and Licensure
Federal Executive Agency
Federal Executive Department
Independent Executive Agency
Judicial Government
Legislative Government
State Level Oversight

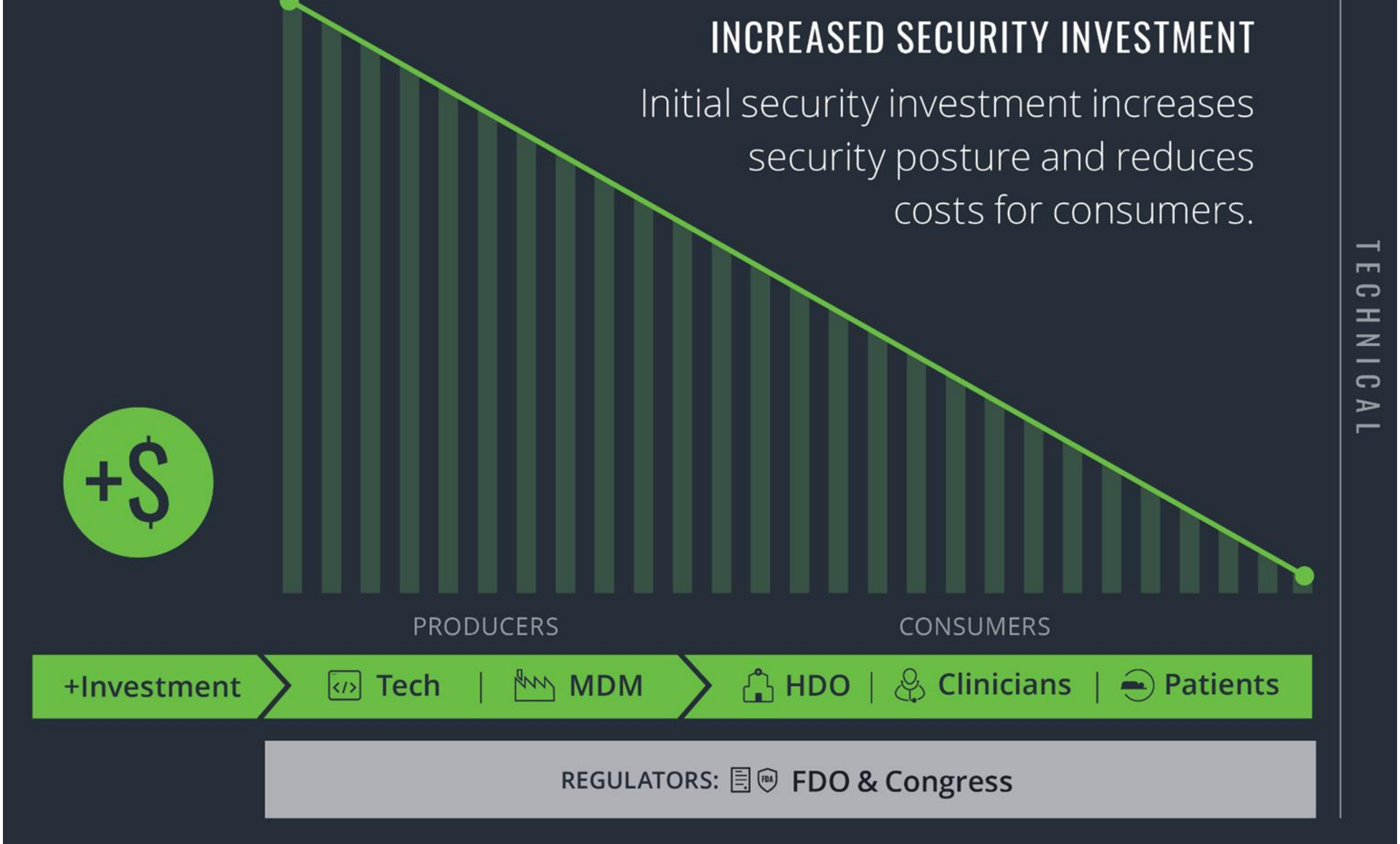
Computers Aren’t Pills

“We Need to Rethink the Whole Thing”

Jeff Shuren Will Aim High on MDUFA V-Linked Reforms

Computers == Pills

=> FALSE


INCREASED SECURITY INVESTMENT
Initial security investment increases security posture and reduces costs for consumers.
+$
PRODUCERS
CONSUMERS
+Investment
Tech
MDM
HDO
Clinicians
Patients
REGULATORS:
FDO & Congress
TECHNICAL

Questions?

https://medcrypt.com/whitepapers-medical-device-thoughtleadership.html

seth@medcrypt.co

SUMMARY AND CLOSING REMARKS

PRESENTED BY:
DR. BILL CURTIS
LUKE MCCORMACK

THANK YOU FOR ATTENDING!

Consortium for Information & Software Quality™