DevOps Metrics-Performance Playbook

ATARC DevOps Community of Practice: Metrics Working Group

April 2022

ATARC

Advanced Technology Academic Research Center

Acknowledgments

On behalf of the Advanced Technology Academic Research Center, ATARC is proud to announce the release of the 2022 Report titled **"DevOps Metrics-Performance Playbook"**, authored by the members of the **ATARC DevOps Community of Practice: Metrics Working Group**.

ATARC would like to take this opportunity to recognize the following contributors, in alphabetical order:

Claire Bailey	VP of Governmental Affairs, Veracode
Robert Buckstad	DevOps Acting Senior Manager, IRS
Chazara Clark-Smith	Senior Technical Advisor, IRS
Dennis Jerome	Management and Program Analyst, IRS
Stephen W. King	DevOps Process Improvement Lead, IRS
Nicole Mandes	Working Group Project Manager, ATARC
Annette Mitchell	DevOps Federal Interagency Council (DFIC) Lead, IRS
Sue Rundell	Business Development, KWR Strategies
David Wray	CTO Public Sector, Micro Focus Government Solutions

Lead Participants and Government Advisors, in alphabetical order:

Hassib Amiryar	MITRE
Lisa Folio	Science Applications International Corporation (SAIC)
Robert Gerner	BMC Software
Jennifer Kenney-Smith	GitLab
Ana Kreiensieck	U.S. Department of Defense (DoD)
Jason Mcknight	U.S. Department of the Treasury (USDT)
Trinette Tucker	Federal Aviation Administration (FAA)

Sincerely,

DevOps Performance Metrics Working Group

Tom Suder, Founder, Advanced Technology Academic Research Center (ATARC)

Disclaimer

This DevOps Metrics-Performance Playbook was developed and published through collaboration of various Subject Matter Experts (SME) throughout government and industry.

References in this paper to any specific commercial product, process, or service, or the use of any trade, firm or corporation name is for informational purposes only, and does not constitute endorsement, recommendation, or favoring by the Advanced Technology and Academic Research Center (ATARC). These materials within may also contain hypertext links, embedded documents, contact addresses and websites to information created and maintained by other public and private organizations. The opinions expressed in any of these materials do not necessarily reflect the positions or policies of ATARC and/or any contributors. ATARC does not control or guarantee the accuracy, relevance, timeliness, or completeness of any outside information included in these materials.

Table of Contents

Table of Figures

Executive Summary

This playbook will provide a framework that will serve as a guide for organizations and agencies seeking to make data driven decisions, drive desired performance outcomes, and transform organizational culture to advance along their respective DevOps journeys toward enterprise-wide adoption.

This paper provides:

- Approach to Scaling DevOps Based on Maturity Assessment
- Value Stream Mapping (VSM) and Utilization in DevOps Metrics and Measurement
- Relationship Between Successful DevOps Delivery and Org Strategy / IT Mission / Org Performance
- Opportunities to Accelerate Desired Strategic People, Process and Technology Outcomes
- Performance Measures and Metrics that Capture Impact and Progress

1 Approach to Scaling DevOps Based on Maturity Assessment

1.1 Introduction

The purpose of this playbook is to provide a framework that will serve as a guide for agencies seeking to make data driven decisions, drive desired performance outcomes, and transform organizational culture to advance along their respective DevOps journeys toward enterprise-wide adoption. This playbook will describe and define processes that can be used to characterize the current state and the desired future state for the DevOps Continuous Integration and Continuous Delivery (CI/CD) Pipeline Metrics.

Performance metrics are integral to an organization's success. It is important that organizations select their chief performance metrics and focus on these areas because these metrics help guide and gauge an organization's achievements as well as identify bottlenecks which drive organizational improvements. Operationally, these defined metrics or service level indicators can be traced to organizational objectives.

1.2 Establishing a Common Definition of DevOps

DevOps is a software engineering culture and practice that aims at unifying software development (Dev) and software operations (Ops). The main characteristic of the DevOps movement is to strongly advocate automation and monitoring at all steps of software construction, from integration, testing, releasing to deployment and infrastructure management. DevOps aims at shorter development cycles, increased deployment frequency, and more dependable releases, in close alignment with business objectives. The primary objectives of the DevOps methodology are to speed up the time to market, apply incremental improvements in response to the changing environment, and create a more streamlined development process.

A further delineation of DevOps is adding the security component into your DevOps software engineering culture. Organizations can begin the DevOps journey, and as the organization matures and security is embedded into the application life cycle, organizations establish a DevSecOps approach.

Organizational adoption of measuring overall DevOps begins with performance metrics which begins with an internal review of where your organization stands today with your journey to a value-based metrics and measurement-based approach to managing your Application and Development processes. This assessment begins with both a capability assessment and a maturity assessment.

To provide a guide to assess your agency's progress toward implementation of DevOps and Application Development Continuous Integration and Continuous Delivery (CI/CD) processes, please note the organizational maturity levels identified in the DevOps Maturity Matrix below:

DevOps Maturity Matrix[1]:

Initiative	Initial	Defined	Managed	Optimized	Continuous Improvement
Build, Deploy, & Release	Manual deployments	• Some automated deployment scripts • Environment tailoring required • Few enterprise-level tools	• Automated deployments into environments	• Orchestrated deployments	Zero-touch, zero downtime deployments
Testing	Mostly manual test scripts	• Test strategy defined • Automated unit tests • Separate test environment	• Automate integration testing and database testing for select projects • Static code analysis	• Fully automated acceptance tests • Automated performance tests	Tests run as a function of continuous monitoring
Security	Security bolted on at end of development lifecycle	• Ensure policies and tools developed	• Security moved up in development lifecycle for earlier weakness detection and mitigation	• Automate security requirements/ security control testing • Defects identified and immediately fixed	Security built-in from beginning
Infrastructure	Ad hoc environments, little consistency	• Establish enterprise code standards • Environments manually created	• Automated creation of environments • All follow common set of blueprints	• Enable Cloud processes and capabilities to improve infrastructure use	Utility-based computing that leverages auto-scaling
Delivery methodology	Poorly defined scope, ad hoc change requests	• Releases take a long time, delay "business" value	• Release cadenced defined • Requirements are stable • Use business metrics to assess progress and recommend improvements	• Release on demand, time-boxed to meet business need	Continuous deployments further innovation
Culture and Governance	Multiple layers, hierarchical decision-making	• Culture strategy developed and behavioral changes identified • Communities of Practice established	• Build buy-in, support, and executive confidence DevOps decision-making process	• Blame-free culture embedded in governance • Increase range of distributed decisions • Organize Cross/Up Skill training strategies to socialize best practices	Distributed decision-making

[1] From DevOps Primer: Case Studies and Best Practices from Across Government, American Council for Technology Industry Advisory Council (ACT-IAC) EMERGING TECHNOLOGY COMMUNITY OF INTEREST DevOps Working Group, 19 February 2020

Once your organizational maturity level is defined and recognized across your organization utilizing the DevOps Maturity Matrix, this provides the foundation for identifying and building your organization's approach to categorizing appropriate measures and metrics to continually improve your organization's Agile DevOps and CI/CD processes.

We will identify components that influence positive/negative outcomes of Agile/DevOps change. We will also identify the relative degree to which an "influencer" or "workflow process" that impact a positive/negative change could be a type of measure.

Organizations that identify an Agile Metrics Strategy approach for Development Solutions can utilize this data for methods to increase efficiency and reduce cost using cutting edge DevOps metrics solutions. This enables government, academia, and private industry thought leaders to communicate, utilizing the same measurements and metrics terminology.

2 Value Stream Mapping (VSM) and Utilization in DevOps Metrics and Measurement

2.1 Introduction

To understand Value Stream Mapping, we need to first understand what a "value stream" is. Simply put, a value stream is a series of steps that occur to provide the product or service that their customers want or need. To provide the product or service that the customers desire, every organization has a set of steps that are required. Value Stream Mapping enables us to better understand what these steps are, where the value is added, where it is not, and more importantly, how to improve upon the collective process. Examples include creating a product, fulfilling an order, admitting, and treating a medical patient, providing a loan, delivering a service, etc.

2.2 Background of VSM

Value Stream Mapping (VSM) is the initial step to determine the true value of your DevOps lifecycle.

By identifying steps with lead time, bench marking as well as fully describing the mechanisms needed to determine your current state, you can identify the impacts of improving your current DevOps Pipeline.

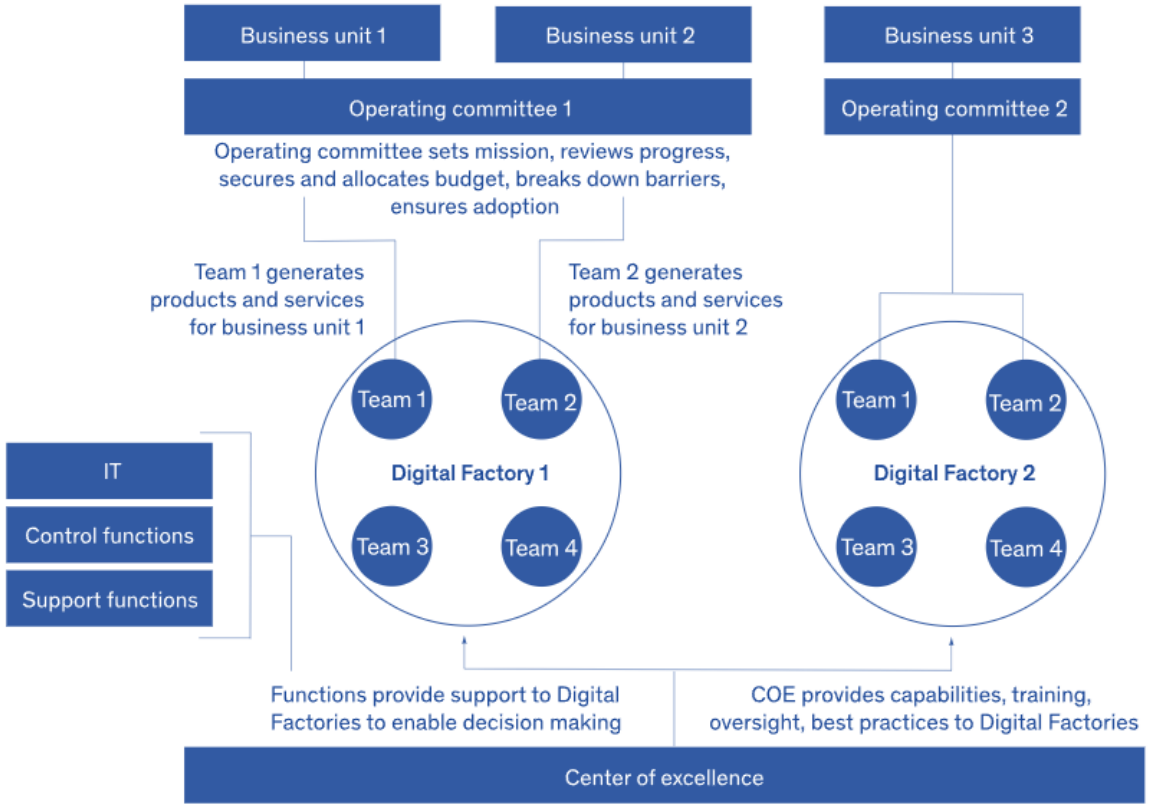

Figure 1: The Digital Factory Scales Digital Transformation

The VSM differentiator of a Process Mapping Process provides a foundation for identifying additional details of the process with a focus on improving the efficiencies by modifying the processes and possibly eliminating unnecessary steps.

From a DevOps and CI/CD perspective, there are several highly successful industry examples of how VSM, DevOps and CI/CD Automation can greatly help with digital transformation.

For example, McKinsey & Company, has documented success stories across industry organizations building a Digital Software Factory that leverages automation techniques to optimize IT Transformation.

As illustrated in **Figure 1** above, a Digital Software Factory can facilitate faster decision making and higher quality products by creating unique teams to lines of business underpinned by a center of excellence.

Digital Software Factories are evolving within industry and government as a technique to optimize DevOps by creating common services and processes. A Digital Software Factory leverages lean manufacturing and value stream management best practices and helps organizations build a common, dedicated capability and services that improve efficiencies and reduce risks.

One of the advantages of creating common services to support DevOps is the ability to capture higher quality metrics that can be leveraged by a community of practice and stakeholders to continuously improve their software delivery lifecycle.

Digital Software Factories can also be leveraged to align with recent best practices defined by National Institute of Standards and Technology (NIST) for "Mitigating the risks with software vulnerabilities by adopting a secure software development framework (SSDF)". Additionally, the Office of Management and Budget's (OMB) recent publication for Federal Zero Trust Strategy aligns with guidance from NIST and Cyber and Infrastructure Security Agency (CISA). These recommendations require agencies to have a dedicated application security process as part of their CI/CD Automation for application security.

2.3 Overview of VSM and Approaches to Build VSM for DevOps Measurement

Once you have assessed your level of maturity, you should identify the measurements and metrics you would like to measure to your DevOps and CI/CD processes. This will enable you to build a continuum of measurements and continually review your application development lifecycle based on your Value Stream Mapping results.

Sample steps to build your data-driven VSM:

- Identify your maturity level
- Identify your measures that tie to your organizational plan and Value Stream
- Identify metrics
- Identify the range of metrics that identify organizational, or process change is needed
- Implement the measurements
- Adjust your processes based upon the metrics

2.4 Building and Executing Value Streaming Mapping Processes

The concept of Value Stream Mapping (VSM) is certainly not new and was widely introduced to the manufacturing industry by Michael Porter in his 1985 book *Competitive Advantage.* Today, VSM is leveraged in many industries to improve efficiency, reduce risk, and ensure that value is being delivered and a critical for success with Agile and DevOps.

Value Stream Mapping (VSM) provides us with a structured visualization of the key steps and corresponding data needed to understand and intelligently make improvements that optimize the delivery of a product or service. It accomplishes this by removing process steps that exist, but these steps are not required to fulfill delivery, or are duplicative or wasteful. Each value stream represents the sequence of steps an organization uses to deliver value, highlight efficiencies, delays, rework, and impact to other meaningful measurements.

Value streams cut across departments, operational functions and should contain all the steps necessary to deliver value to the end consumer. They should define the people who perform each step and the systems they use to complete their tasks as well as the flow of information required to satisfy a request. At the highest level, the production of a product or service typically defines the major steps, metrics, and organizational responsibilities.

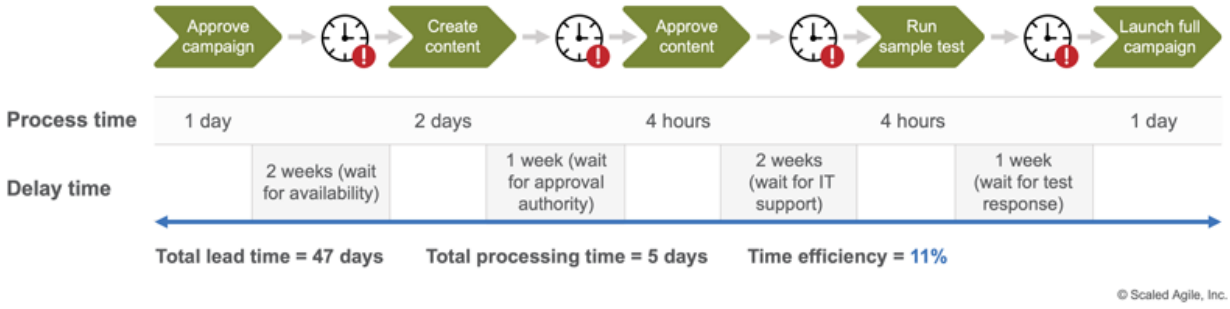

Figure 2: Example Value Stream Map for a Marketing Campaign that Supports a Product Launch[2]

"Value Stream Mapping is a lean manufacturing or lean enterprise technique used to document, analyze and improve the flow of information or materials required to produce a product or service for a customer." (iSixSigma.com)

Value Stream Mapping is a foundational element of lean thinking and the Scaled Agile Framework (SAFe) standard and is commonly adopted as the best method to define all the work items across development, security, and Development Security Operations (DevSecOps) to fulfill a product or service.

Below is a broader example of how Value Stream Mapping was applied to the IT Service Lifecycle to define value streams that can be used to measure efficiency and agility throughout the lifecycle of an IT

[2] Example Value Stream Map for a Marketing Campaign that Supports a Product Launch, provided by Scaled Agile, Inc.

service. This example is based upon The Open Group's IT Value Chain reference architecture, and illustrates how the entire lifecycle of an IT investment can be defined in four key value streams (Plan, Build, Deliver, Run).

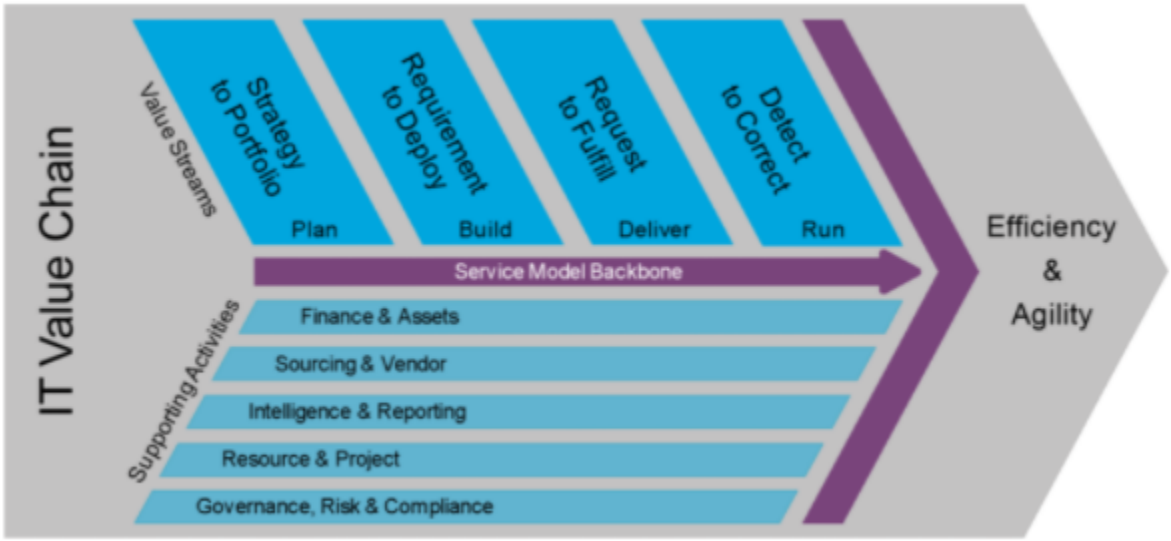

Figure 3: The IT4IT Value Chain Defines Four Value Streams that Can Be Used to Measure How Well IT Is Performing

The IT4IT Reference Architecture uses the value stream construct as a way of grouping the functional components and data objects together to provide context for where value is being created/delivered. This figure illustrates the high-level value streams (level one). The details of each value stream are also provided that define who is responsible for creating, refining, and tracking key data objects across the IT service lifecycle. Each value stream contains data objects as well as key measurement indicators that can be used to reduce risks and cost and continuously improve. The complete IT4IT Value Chain Reference Architecture is available from the open group, and has been leveraged by public and commercial entities as a framework for IT optimization and defining requirements for Digital Transformation, DevOps Optimization and defining policies and standards for IT Management that are vendor agnostic.

Across federal government information technology organizations, there are project-level examples of Agile DevOps and CI/CD processes. Teams have captured measurements and metrics based upon VSM. To continually gain agency-level support, the VSM utilization in this DevOps Metrics and Measurement Playbook showcases quantitative, measurable process improvements based on value to illustrate new techniques that are contributing to critical mission capability needs.

2.5 Defining Measurements and Metrics for your Performance Measures

Your value map will help mitigate excessive downtime, congestion, and other issues in the future. Naming the type of waste in your process is helpful to guide the type of improvements needed. Waste can occur within groups of people, specific processes, technology, or even in a combination.

Resist the urge to overlook poorly constructed processes, gaps, and weaknesses—or getting stuck in a "that's just the way we do it" mentality. Seek out your discrepancies in order to achieve long-term benefits and perceived customer value.

For example, there are performance metrics that identify waste within Lean methodologies [3]. There are also references that are available through Plutora.com[4]. Below is a simple list of how those types of waste can be translated into DevOps, using DOWNTIME as a helpful acronym:

Defects	Mistakes, bugs, and rework
Overproduction	Producing more than is necessary, such as extra features
Waiting	Delays and the amount of time the product spends in queue
Non-utilized talent	Not including all employees in the improvement process
Transport	Handoffs from developers, to testers, to deployment
Inventory	Partially done work
Motion	Individual task switching
Extra processing	Relearning or reworking, such as undocumented code
Value Added Time	Value added time is the amount of time that a team actually spends working on the project (as opposed to, for example, the time that a project or request sits in the queue). Whenever there is no change in the product, it is considered non-value-added time
Lead Time (LT)	Lead time represents the total time it takes a person or team to complete a task—it is the combination of value added and non-value added
% Complete/Accurate (%C/A)	This is the percentage of information-based work that is complete and accurate the first time and requires no re-work by downstream processes

For a complete table of Key Performance Metrics, please refer to **Appendix 1 – Sample Critical Success Factors and Key Performance Indicators (KPIs).**

2.6 How a VSM impacts the Software Development Life Cycle (SDLC)[5]

When we talk about value streams in relation to software development, we mean to include every activity in the process from the first idea to production. The actual value of the software being developed will be determined by customers. It seems so simple. If your web or mobile app delivers value

[3] https://www.lucidchart.com/blog/best-lean-tools-for-process-improvement
[4] https://www.plutora.com/blog/value-stream-mapping
[5] Experitest "How CICD and VSM planning Separately Work Together", accessed via https://experitest.com/mobiel-app-testing/

to your customers, that will come back to the enterprise in the guise of business value. Creating value for customers is part of the movement towards more customer-centric aspects of web and mobile app development. The issue is that in large organizations, the process of development and delivery are so complex that it is hard to map a value stream.

The first step in accomplishing this is by creating a current state Value Stream Map. With this map, we will be able to list each step of the SDLC in detail. The first step is to bring all stakeholders together to identify every step in the process of web and mobile app delivery.

Every government organization has a SDLC, but they have different patterns/methods of delivering software.

Building on the high-level value stream from the IT4IT framework example provided in **Figure 3** above, the lower-level value stream for “Requirement to Deploy” illustrates the application build process and provides a framework that can be leveraged to define both your current process as well as your desired state.

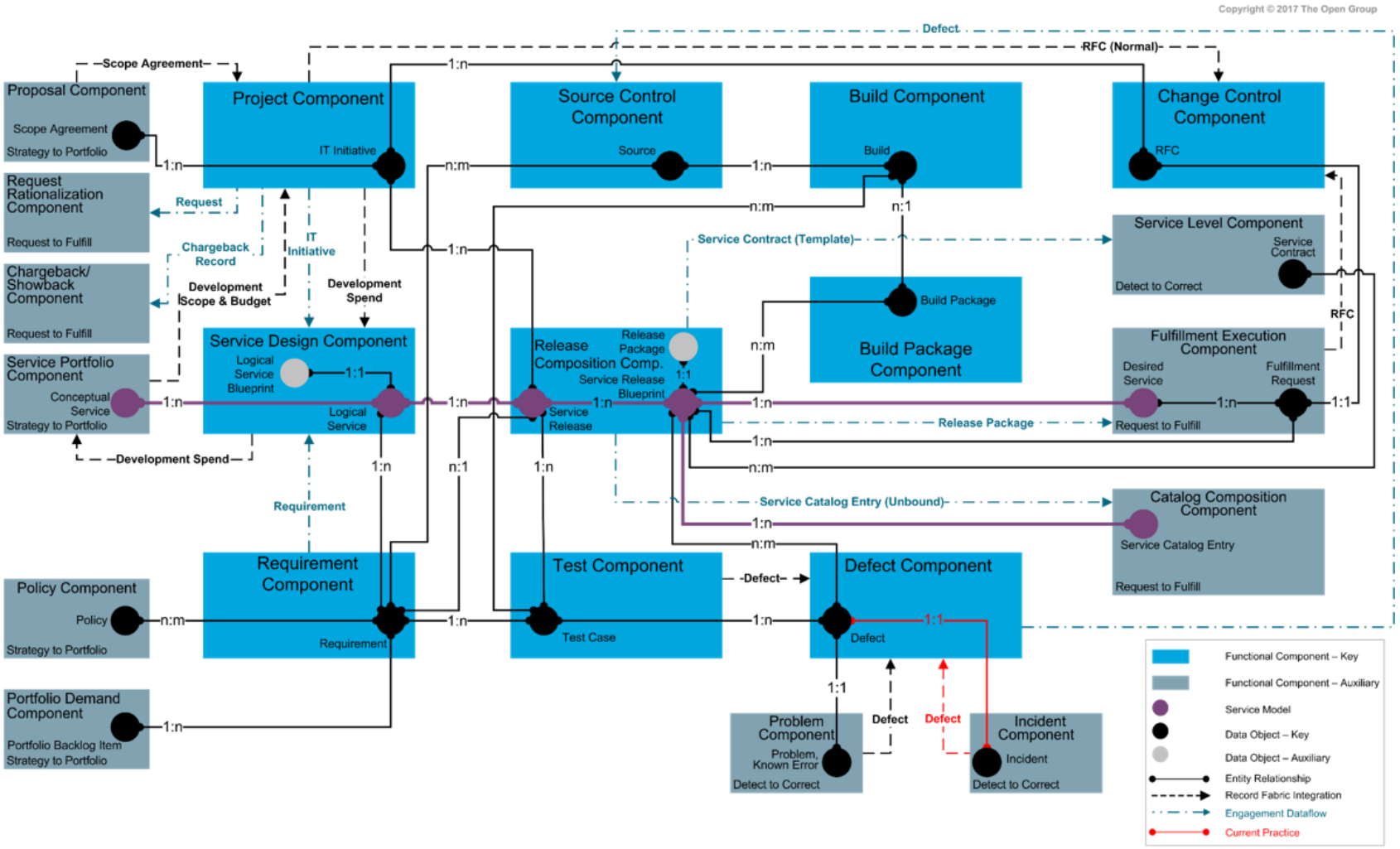

Figure 4: Illustration of the Functional Components and their Relationships Within a Software Development Life Cycle Using VSM[6]

[6] Illustration of the functional components and their relationships within a Software Development Life Cycle using VSM[6], provided by Open Group: https://www.opengroup.org/it4it

Based on this example, please refer to **Appendix 1 – Sample Critical Success Factors and Key Performance Indicators (KPIs).** These KPIs can be referenced within a value requirement to deploy a value stream.

Most government agencies have documented SDLC's, and a few have also defined details on the value stream with measurements for their CI/CD Pipeline.

Below is an example from the Department of Treasury/Internal Revenue Service (IRS).

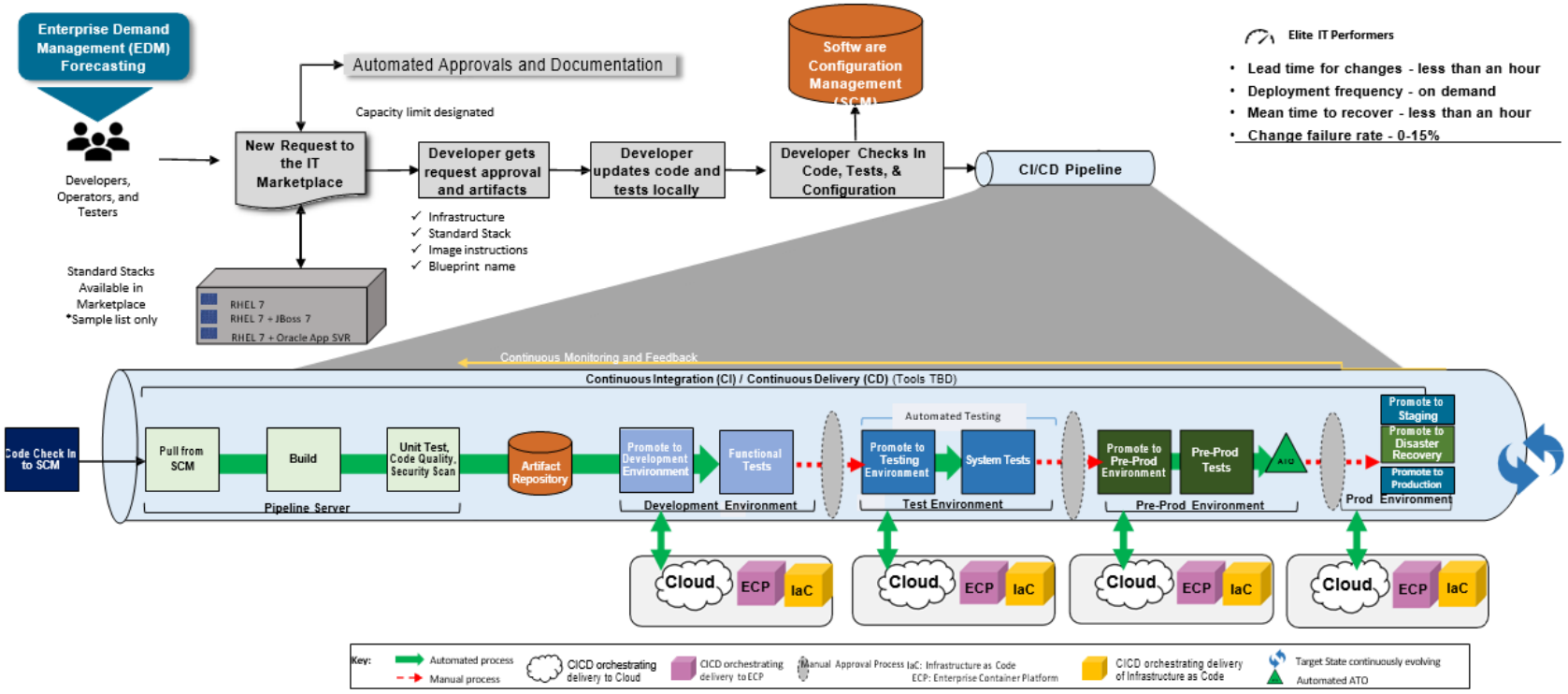

Figure 5: IRS Enterprise CI/CD Pipeline

2.7 Build On Your Current Workplans[7]

After sitting with stakeholders from across your organization, you can begin the work of optimizing your delivery workflow. This will require optimizing the Value Stream Map that you just made. Improvements can be made when you have measured where your company's goals reside, while simultaneously identifying opportunities to work toward.

With a Value Stream Map in place, you have every aspect of your delivery process at hand. This means that each role in the company has the tools they need to continuously problem solve and improve development.

When you perform VSM on current processes, you can identify a bottleneck in specific areas of your DevOps lifecycle. A VSM can identify gaps that can be leveraged to identify needs for additional automation, elimination of redundant tasks or tools, or you may see multiple tools being utilized that need consolidation.

2.8 Building and Improving DevOps based on your VSM

The outcomes and improvements identified by your VSM will indicate areas of modifications for your SLM, SDLC, and/or DevOps CI/CD Pipeline. The outcome will be process improvements and points of measurement to monitor overall improvement in velocity, quality, efficiency, and security.

The DevOps Metrics-Performance Playbook assumes that each agency has a documented SLM and SDLC. It may only be a document or paper with variable adoption throughout the enterprise. In this Playbook, the DevOps CI/CD Pipeline is the last mile of how the components work together to deliver a product.

A VSM can improve different areas like change management, your SDLC, your DevOps CI/CD Pipeline, and applications that are used by citizens, migration to cloud, or any other technology area.

As we walk the new path with Digital.ai, we are seeing the relationship between CI/CD and VSM is constantly evolving. DevOps, for instance, essentially combines development which is a value stream and Operations which is considered a non-value stream. These steps in the process both complement each other and add value in the way that they are implemented overall.

When mapping a value stream, you need to look at both material and information flows. This is true of DevOps as well. DevOps CI/CD Pipeline visualization gives you both the ability to analyze your material (artifacts) but also to communicate your findings and share the information across an organization.

The most important way that a VSM measures value is by determining which step of the process adds value, and which does not. A DevOps CI/CD Pipeline should do the same. The strategy is to automate as much of your web, application, and mobile app testing as possible, which will increase the speed and quality of your apps. The extra benefit, of course, is improving productivity and velocity, which in turn will reduce waste.

[7] Ibid, Experitest

3 Relationship Between Successful DevOps Delivery and Org Strategy / IT Mission / Org Performance

3.1 Introduction

Twenty years ago, Information Technology (IT) was not core to an organization's mission. IT was done in the background. People ran cables and hooked up computers to a network. IT took care of back-office functions like finance and human resources. Well, times have changed.

Today, technology is critical to everyone. Software is a critical part of technology solutions. For example, in the 1990s and early 2000s, Blockbuster Video was a physical store where people rented VHS tapes or DVDs to watch movies or play videogames at home. Blockbuster was at its peak in 2004 and filed for bankruptcy by 2010. Contrast that with today, where people access streaming services to watch movies and TV shows. Those streaming services are only possible because of cloud-based technologies and software. IT is now a part of the core mission of the streaming business.

3.2 Aligning Metrics and Measurements to your Strategic Missions

Most organizations have a strategic plan which includes a vision, mission statement, and goals and objectives. Because software is so critical to an organization's mission, it is imperative to reexamine the organization's strategy through new eyes to identify where software has become a critical aspect of a goal or objective. Most organizations today not only have a Strategic Plan, but also have a separate IT Strategic Plan. The IT Strategic Plan, sometimes referred to as the Information Resources Management Strategic Plan, is signed by the organization's Chief Information Officer (CIO). This Plan details how IT will achieve their goals and objectives identified in the overall Strategic Plan.

The organization's CIO is aware of software development methodologies, such as Agile, which began in 2000 to focus on faster software development. Realizing that Agile did not fix the entire process, DevOps emerged in the late 2000s to include the operations aspect of the software lifecycle. DevOps has become an industry standard to develop and deliver software capabilities.

Resources available to CIOs to make the connection to the IT Strategic Plan to use DevOps to achieve the goals and objectives are outlined below in the organization's Strategic Plan. These resources include:

1. Alignment of IT Strategy to Agency Mission - Government Performance and Results Act (GPRAMA)
2. Examples of mapping IT strategic goals and objectives in Fed Agencies to Goals & Objectives of DevOps
    a. DevOps Research and Assessment (DORA) Reports, 2019
    b. Booz Allen Playbook
    c. State of DevOps Report – 2019
    d. IT Projects/Program's map to DevOps practices to Agency Goals & Objectives
3. Research Gartner references and examples of how DevOps aligns/drives/accelerates achievement of strategic goals & objectives:
    a. Operations & Maintenance (O&M) costs

   b. People skills (Upskilling Report - DevOps Institute)
   c. Technology adoption/modernization
   d. Agile adoption/cultural transformation

4. Summarize how technological advances, legislative demands, security challenges require enterprise-wide agility (higher frequency and adaptability of SW deployments / updates / improvements), “baked in” cyber security, proactive technological adoption / awareness

4 Opportunities to Accelerate Desired Strategic People, Process and Technology Outcomes

4.1 Introduction

This section identifies how you analyze metrics and measurements to improve your DevOps environment. Through continual measurement and analysis, your agency is positioned to Accelerate Strategic People, Process and Technology Outcomes securely and efficiently.

4.2 Aligning your Strategic Vision to your Operational Execution Initiatives

One approach to aligning your strategic vision to your operational initiatives is to begin by looking at the single sources of truth and data and identifying how to remove technical debt. This ensures that the metrics and measurements data answer the right questions for the teams who are working to continually gain efficiency and improve processes. The overall goal is to build executive buy-in to ensure your Strategic Vision aligns to your Operational Execution Initiatives. Additional details provided in Section 3, **Relationship Between Successful DevOps Delivery and Org Strategy / IT Mission / Org Performance** of this document.

To begin your journey to align your strategies, it is recommended to begin determining your methodology. One framework example for Agile development is Scaled Agile Framework® (SAFe®). This link, Advanced Topic - Accelerating Flow with DevSecOps and the Software Factory - Scaled Agile Framework, provides additional detail for accelerating your DevSecOps approach and to build a software factory. As referenced in this TechBeacon article, DevOps at scale: How to build your software factory | TechBeacon, there are still several approaches to building a software factory.

The building blocks for getting started are the same. Agencies should focus on Processes, People, and Technology Outcomes. When identifying Processes, the Business Process Change Management focus should identify metrics that spotlight areas of improvement to ensure that you can positively impact the business organization and the business processes. It is recommended that you embrace Open Standards that align between your VSM to your organization.

When aligning personnel to improving metrics and measures, considerations for identifying key resources should be reviewed. For example, identifying the chief process officer, specific personnel for vendor management, or other shared resources with clearly identified roles leads to success. Also address internal processes, such as if a resource need is identified for your project, but the resource reports through a different area. How do you ensure your organization is Agile Enabled to quickly realign resources to improve the overall project flow and success?

To improve technology outcomes, highlight how you can gain efficiencies in the operations pipeline and reducing technical debt. By adopting DevOps, an outcome is more automation, which provides teams the ability to analyze critical components versus redundant components.

5 Performance Measures and Metrics that Capture Impact and Progress

5.1 Introduction

Performance Measures and Metrics are integral to an organization's success. It is important that organizations select their chief performance metrics and focus on these areas because these metrics help guide and gauge an organization's achievements as well as remove bottlenecks and drive improvements. These chief performance metrics are aligned to the Value Stream Management for your organization as well as your overall entity mission. In this section, we will further analyze how to identify metrics that can be utilized and aligned to your Value Stream Map (VSM). Additional information on VSM approaches is described in Section 2 of this document.

In addition to identifying an Agile Application Development Methodology, for example Scaled Agile Framework® (SAFe®) or Large Scale Scrum (LeSS), metrics strategy approach for development solutions, government, academia, and private industry leaders can utilize this approach for measurement methods to increase efficiency and reduce cost using cutting-edge DevOps metrics solutions.

5.2 Performance Measures and Metrics Defined

Metrics are used for the quantitative and periodic assessment of a process. They should be associated with targets that are set, based on specific business objectives. Metrics provide information related to the goals and objectives of a process and are used to take corrective action when desired results are not being achieved and can be used to drive continual improvement of process effectiveness and efficiency.

There are four types of Performance Measures and five core Agile Metrics that can be coupled together to evaluate and measure value. The primary measure of success for agile projects is delivery of value or capability to the end user continuously.

5.3 Performance

1. Progress: Milestones and deliverables in the capability of the process

2. Compliance: Compliance of processes required to meet governance requirements, regulatory requirements, and validate people are using the required processes

3. Effectiveness: The accuracy and correctness of the process and its ability to deliver the 'right result'

4. Efficiency: The productivity of the process, its speed, throughput, and resource utilization

Each of these Performance Metrics can be further defined and broken down into very specific areas. To illustrate how metrics are defined and captured in your CI/CD Pipeline that can be summarized and aggregated to show data-based progress toward meeting a specific high-level metric. For example, an Efficiency metric is a summary of CI/CD measurements.

5.4 DevOps or Agile Measurements

1. Product: Size, architecture, structure, quality, and complexity

2. Resource: Personnel, software and hardware, and performance

3. Process: Maturity, management, and life cycle

4. Project: Earned business value, cost, time, quality, risk, etc

5. Strategic: Net present value, earned business value, return on investment

Depending upon the DevOps Maturity Level utilized in your entity, step one is identifying how your group defines and aligns to a specific framework and your identified CI/CD Pipeline. These high-level metric categories are further defined in the **Glossary** and Acronyms.

5.5 Metrics Benchmarking

For developing and determining business value performance measures, an exercise in Benchmarking is required. Benchmarking creates the ability to be able to compare performance to industry standards and averages. It must be focused, relevant and business aligned to deliver value.

Benchmarking is a foundational piece of Performance management and continuous improvement. For success, Managers must:

- Know (map) the Current State. This establishes a clear and defined performance baseline.
- Determine the comparison environment. Establishing the baseline creates an environment to begin conducting comparative analysis against similar organizations and industry best practices.
- Engineer a clear Desired Future State. Identify the starting point and desired end which enables managers to identify and act on opportunities for savings and improvements.
- Track savings and learning and continue to assess and improve. As the process has been improved, a better benchmarking process can start with the next iteration.

5.6 Industry Standard

The benchmarked industry standards per the State of DevOps 2019 report are identified by Four Key Metric[8] areas: Deploy Frequency, Lead Time for Changes, Time to Restore Service, and Change Fail Rate. These are illustrated in **Figure 6**:

[8] https://www.thoughtworks.com/radar/techniques/four-key-metrics

Figure 6: Accelerate: State of DevOps 2019

Aspect of Software Delivery Performance*	Elite	High	Medium	Low
Deployment frequency For the primary application or service you work on, how often does your organization deploy code to production or release it to end users?	On-demand (multiple deploys per day)	Between once per day and once per week	Between once per week and once per month	Between once per month and once every six months
Lead time for changes For the primary application or service you work on, what is your lead time for changes (i.e., how long does it take to go from code committed to code successfully running in production)?	Less than one day	Between one day and one week	Between one week and one month	Between one month and six months
Time to restore service For the primary application or service you work on, how long does it generally take to restore service when a service incident or a defect that impacts users occurs (e.g., unplanned outage or service impairment)?	Less than one hour	Less than one day[a]	Less than one day[a]	Between one week and one month
Change failure rate For the primary application or service you work on, what percentage of changes to production or released to users result in degraded service (e.g., lead to service impairment or service outage) and subsequently require remediation (e.g., require a hotfix, rollback, fix forward, patch)?	0-15%[b,c]	0-15%[b,d]	0-15%[c,d]	46-60%

Figure 7: Aspect of Software Delivery Performance[9]

[9] Accelerate: State of DevOps 2019

5.7 Current State Metrics

The DevOps Current State can be described as a repeatable process enabled by automation and manual toolsets with supported asynchronous monitoring. Defined metrics must be collected and compared from projects/applications both within and outside development. This provides a true representation of the value of the pipeline while also giving areas for improvement.

Aligned to Industry Standards, the following metrics are currently being collected/managed within the DevOps processes:

1. **Lead Time:** How long it takes from code commitment to deployment to production, represented as a duration.
2. **Deployment Frequency:** How often/frequently application(s) are deployed to production, usually represented as a percentage. Also known as Throughput.
3. **Change Failure Rate:** The measure of the percentage of changes that result in a failure.
4. **Mean Time to Recover (MTTR):** The measure of the mean duration for restoration/fix of a build failure.
5. **Wall Time:** The 'real' time (clock on the wall) that it takes for the process to complete. It is primarily used internally for measuring time in-between functions and/or sub-processes. This may also be classified as Cycle Time for Value Stream Mapping purposes.
    - **Cycle time**: The frequency of units/features produced or the average time between the completed production of one unit/feature to the completed production of the next.

5.8 DevSecOps (Security)

DevSecOps is a set of extended and enhanced proactive security practices which integrate security considerations into DevOps throughout planning, deployment, and operational stages. Adopting DevSecOps provides value by helping to further automated security testing and compliance solutions, adding speed to deployment and reducing costs.

5.9 Metrics

1. **Vulnerabilities**[10]
    a. **High:** The loss of Confidentiality, Integrity, or Availability (CIA) could be expected to have a severe or catastrophic adverse effect on organizational operations, organizational assets, or individuals
    b. **Medium/Moderate:** The loss of Confidentiality, Integrity, or Availability (CIA) could be expected to have a serious adverse effect on organizational operations, organizational assets, or individuals
    c. **Low:** The loss of Confidentiality, Integrity, or Availability (CIA) could be expected to have a limited adverse effect on organizational operations, organizational assets, or individuals

[10] FIPS PUB 199: Standards for Security Categorization of Federal Information and Information Systems

2. **Maturity**
    a. **Level 1 Initial:** Little scanning, triage, and remediation capability
    b. **Level 2 Repeatable:** Applications inconsistently triage and remediate (process in place)
    c. **Level 3 Defined:** Applications continuously triage and remediated high findings (done consistently)
    d. **Level 4 Managed:** Applications continuously triage and remediated H/M findings (self-sufficient)

5.10 Desired Future State Metrics

Future State Metrics includes Current State collection(s) and the following:

Security Plan of Action and Milestones (POA&M): FISMA requirement to effectively manage security program risk and mitigate program- and system-level weaknesses.

Percentage (%) of reduction in registered/actionable POA&Ms

Incident Management: The process responsible for tracking and recording incidents throughout the incident's lifecycle. There should exist an allocated appropriate prioritization code for how the incident is to be mitigated by administrators. The prioritization calculation is normally produced by accounting for both the incident urgency and the assigned level of impact. Although industry formats for prioritization vary, however, a set standard calculation matrix should be used to ensure best compliance and management.

For example:

Priority 1 – Severe Critical Work Stoppage – Any issue causing severe mission critical work stoppage or any IT issue impacting safety or health, i.e. fire, shock from equipment etc. Impact may be on multiple internal or external customers and service to users. Immediate action required.

Priority 2 – Potential Critical Work Stoppage – Any issue that could have a direct impact on the service to users or if its scope is multi-user and there is no work-around. Could lead to severe mission critical work stoppage if actions are not taken to resolve problem.

Priority 3 – Any issue causing work stoppage for one customer with no work around. Examples include password resets or unlocks; service disruption for single customer resulting in work stoppage; alert notification of authorized outage or scheduled maintenance resulting in a work stoppage.

Priority 4 – Any issue that is non-Critical or non-software problems where it is not a work stoppage and there is a workaround.

For performance reporting, there can be a calculated percentage reduction in registered/actionable incidents per priority levels given by the impact/urgency calculation and priority code matrix.

Release and Deployment Management (RDM): According to ITIL® v3, Release and Deployment Management process is defined as:

To plan, schedule, and control the build, test, and deployment of releases, and to deliver new functionality required by the business while protecting the integrity of existing services.

It is important to note here that ITIL® v4 has decoupled Release Management and Deployment Management to provide for a more Value Centric approach with emphasis on the user experience.

Agencies should model and adapt their RDM processes according to industry best business practices, such as ITIL®, but do not have to be constrained by them. IRS IT Enterprise Architecture, for example, defines RDM as "Focuses on the end-to-end process for planning and implementing systems or other solutions through an incremental approach."

Key Performance Indicators (KPIs) for RDM can include, but not limited to:

Percentage (%) of reduction person hours between a set time period performing deliveries/deployments

Percentage (%) of faster releases/deliveries to Production, including S/W deliveries and Authority to Operate (ATO)

Percentage (%) of Release Success Rate

Release downtime

Incidents/Outages caused by a release

Automated Virtual Machine Management (AVMM): Provides for proper Virtual Machine management through automation given approved workflows, automated provisioning, etc.

In use, IRS IT defines AVMM as "The ability to automate management of Virtual Machines (i.e., OS hypervisors and their guest operating systems) and virtual containers (i.e., container hypervisors and their guest containers), with the given KPI: Percentage (%) faster stand up of VM's/Containers.

Agencies should build their AVMM processes while utilizing various VM toolsets and Virtual Management software, such as Microsoft Azure or Amazon Web Services (AWS) suite of tools and products.

5.11 Reporting: Dashboarding and Tools

Dashboarding

Dashboarding seeks to aiding in providing real-time or near-real-time proactive monitoring, root cause analysis, and business intelligence. Proper implementation and operational use of dashboards will help to integrate domain-specific tools and information, helping to focus monitoring across the different personas and domain silos.

Availability and performance monitoring dashboards have four delivery goals[11]:

[11] Gartner, "How to Build an Ideal Performance Monitoring Dashboard", 15 Sept 2017

1. Depiction of current conditions across the organization (IT and beyond)
2. The ability to perform root-cause analysis during outages and degradations
3. Contextual access to historical data for trending and planning purposes
4. Actionable insight and predictions to prevent future issues

An ideal performance-monitoring dashboard framework must[12]:

- Provide top-level views and drilldown, intuitive navigation, and raw data access
- Bridge the domain silos that exist across IT by providing information from a variety of tools and sources
- Provide for the rapid resolution of performance issues by allowing access to a broad spectrum of users

Tools

Business Intelligence Tools are available to provide better data business intelligence analysis and near-real time reporting with drill-down leadership views.

A variety of tools exist in today's marketplace that provide flexibility and interoperability to dynamically report an agency's movement and progress in their DevOps environment. Examples include Microsoft Power BI, Tableau, as well as other industry leaders.

[12] Ibid. Gartner

Glossary and Acronyms

Advanced Technology and Academic Research Center (ATARC) – A non-profit organization that provides professional development and collaborative forums for government, academia, and industry to identify, discuss and resolve emerging technology challenges.

Agile Frameworks – An Agile framework is a specific approach to planning, managing, and executing work. Agile frameworks typically fall into two categories: Frameworks designed for teams, and frameworks designed to help organizations practice Agile at scale, across many teams. Example: SAFe. Please reference SAFe in Glossary.

American Council for Technology Industry Advisory Council (ACT-IAC) – A 501(c)(3) non-profit educational organization established to improve government through the effective and innovative application of technology.

Asset - A major application, general support system, high impact program, physical plant, mission critical system, personnel, equipment, or a logically related group of systems.

Authorization to Operate (ATO) – The official management decision given by a senior organizational official to authorize operation of an information system and to explicitly accept the risk to organizational operations (including mission, functions, image, or reputation), organizational assets, individuals, other organizations, and the Nation based on the implementation of an agreed-upon set of security controls.

Availability – Ensuring timely and reliable access to and use of information.

Awareness (Information Security) – Activities which seek to focus an individual's attention on an (information security) issue or set of issues.

Capability Assessment – For organizations seeking to improve their process or knowledge management efforts, an important first step is to thoroughly and objectively assess current capabilities and performance.

Chief Information Officer (CIO) – In government, the CIO can serve as the designated authorized individual, in charge of an agency's or department's information technology and computer systems. Their duties include assessing current processes, recommending software upgrades, and directing the executive team on the best processes. Also known as an Information Technology (IT) Director.

For example, the Designated Approving Authority (DAA), in the United States Department of Defense, is the official with the authority to formally assume responsibility for operating a system at an acceptable level of risk.

Chief Information Security Officer (CISO) – A senior-level executive responsible for developing and implementing an information security program, which includes procedures and policies designed to protect enterprise communications, systems, and assets from both internal and external threats.

Chief Technology Officer (CTO) – An agency official responsible for:

1. Providing advice and other assistance to the head of the executive agency and other senior management personnel of the agency to ensure that IT is acquired, and information resources

are managed in a manner that is consistent with laws, Executive Order, directives, policies, regulations, and priorities established by the head of the agency

2. Developing, maintaining, and facilitating the implementation of a sound and integrated IT architecture for the agency
3. Promoting the effective and efficient design and operation of all major information management processes for the agency, including to work processes of the agency

Cloud Computing – A model for enabling on-demand network access to a shared pool of configurable IT capabilities/ resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. It allows users to access technology-based services from the network cloud without knowledge of, expertise with, or control over the technology infrastructure that supports them. This cloud model is composed of five essential characteristics (on-demand self-service, ubiquitous network access, location independent resource pooling, rapid elasticity, and measured service); three cloud service delivery models (Software as a Service [SaaS], Platform as a Service [PaaS], and Infrastructure as a Service [IaaS]); and four models for enterprise access (Private, Community, Public, and Hybrid).

Common Service Account – Service accounts that are common across software platforms (i.e., IWAM, IUSER, SQL).

Common Vulnerabilities and Exposures (CVE) – A dictionary of common names (i.e., CVE Identifiers) for publicly known cybersecurity vulnerabilities. CVE's common identifiers make it easier to share data across separate network security databases and tools and provide a baseline for evaluating the coverage of an organization's security tools. If a report from a security tool incorporates CVE Identifiers, an individual then quickly and accurately access fix information in one or more separate CVE-compatible database to remediate the problem. http://cve.mitre.org/about/index.html.

Common Vulnerability Scoring System (CVSS) – Organizations can reference CVSS in order to work on the action(s) that have the highest priority or present the greatest amount of risk. CVSS can measure how serious a given vulnerability is compared to other vulnerabilities so remediation efforts can be prioritized. The Base metrics produce a score ranging from 0 to 10. http://www.first.org/cvss/

Compliance – Compliance of processes required to meet governance requirements, regulatory requirements, and validate people are using the required processes.

Confidentiality – Preserving authorized restrictions on information access and disclosure, including means for protecting personal privacy and proprietary information.

Continuous Integration / Continuous Delivery (CI/CD) Pipeline Implementation – the backbone of the modern DevOps environment. It bridges the gap between development and operations teams by automating the building, testing, and deployment of applications. The mission of the CI/CD Pipeline is to orchestrate the delivery of all project software artifacts, computing infrastructure and patches, from source code check-in through production deployment for all IT customers, fully automated with requisite security and quality in place.

Continuous Integration (CI) is a software development practice where members of a team integrate their work frequently, usually each person integrates at least daily - leading to multiple

integrations per day. Each integration is verified by an automated build (including test) to detect integration errors as quickly as possible. Many teams find that this approach leads to significantly reduced integration problems and allows a team to develop cohesive software more rapidly, in short, in CI Code is regularly delivered to a code repository and builds and tests are automatically performed to find any issues. Alerts are sent if issues arise during the automated build process.

Continuous Delivery (CD) is a software development discipline and is continuously delivered to development and upper-level environments with phase-gates and approvals in place. Automated testing can drive the acceptance of software.

Cybersecurity and Infrastructure Security Agency (CISA) – A United States federal agency, an operational component under Department of Homeland Security (DHS) oversight. Its activities are a continuation of the National Protection and Programs Directorate (NPPD).

DevOps – Set of practices that combines software development (Dev) and IT operations (Ops). It aims to shorten the systems development life cycle and provide continuous delivery with high software quality.

DevOps Maturity Model – A model that determines an organization's standing in DevOps journey along with deciding what more to be accomplished to achieve the desired results. The DevOps maturity model determines growth through continuous learning from both teams and organizational perspectives.

DevOps Research and Assessment (DORA) – Helps industries achieve the DevOps philosophy of speed and stability by identifying four key traits and capabilities of elite teams.

DevSecOps – Tactical trifecta that connects three different disciplines: development, security, and operations. The goal is to seamlessly integrate security into your Continuous Integration and Continuous Delivery Pipeline in both pre-production (dev) and production (ops) environments.

DevSecOps is a further adoption of principles and metrics of DevOps into your security operations. DevSecOps is a set of extended and enhanced proactive security practices which integrate security considerations into DevOps throughout planning, deployment, and operational stages. Adopting DevSecOps provides value by helping to further automate security testing and compliance solutions, adding speed to deployment, and reducing costs.

Effectively – In a way that accomplishes a purpose or produces the intended or expected results

Efficiency – The ability to achieve an end goal with little to no waste, effort, or energy. Being efficient means you can achieve your results by putting the resources you have in the best way possible.

Federal Information Processing Standards (FIPS) – The United States' Federal Information Processing Standards (FIPS) are publicly announced standards developed by the National Institute of Standards and Technology for use in computer systems by non-military American government agencies and government contractors.

Federal Information Security Management Act (FISMA) – Title III of the E-Government Act requiring each federal agency to develop, document, and implement an agency-wide program to provide information security for the information and information systems that support the operations and

assets of the agency, including those provided or managed by another agency, contractor, or other source.

Impact – The magnitude of harm that can be expected to result from the consequences of unauthorized disclosure of information, unauthorized modification of information, unauthorized destruction of information, or loss of information or information system availability.

Incident – An occurrence that (1) actually or imminently jeopardizes, without lawful authority, the integrity, confidentiality, or availability of information or an information system; or (2) constitutes a violation or imminent threat of violation of law, security policies, security procedures, or acceptable use policies.

Incident Management – The process responsible for tracking and recording incidents throughout the incident's lifecycle.

Influencer – One who exerts influence, a person who inspires or guides the actions of others.

Information System – A discrete set of information resources organized for the collection, processing, maintenance, use, sharing, dissemination or disposition of information.

Information Technology (IT) – Any equipment or interconnected system or subsystem of equipment that is used in the automatic acquisition, storage, manipulation, management, movement, control, display, switching, interchange, transmission, or reception of data or information by an executive agency. For purposes of the preceding definition, equipment is used by an executive agency if the equipment is used by the executive agency directly or is used by a contractor under a contract with the executive agency which:

1. requires the use of such equipment or;
2. requires the use, to a significant extent, of such equipment in the performance of a service or the furnishing of a product. The term information technology includes computers, ancillary equipment, software, firmware and similar procedures, services (including support services) and related resources.

Integrity – Guarding against improper information modification or destruction and includes ensuring information non-repudiation and authenticity. The property whereby an entity has not been modified in an unauthorized manner.

Key Performance Indicators (KPIs) – a quantifiable measure of performance over time for a specific objective. KPIs provide targets for teams to shoot for, milestones to gauge progress, and insights that help people across the organization make better decisions.

Large Scale Scrum (LeSS) – Agile Framework for scaling scrum to multiple teams who work together on a single product.

Lead Time (LT) – Lead time represents the total time it takes a person or team to complete a task—it is the combination of value added and non-value added.

Maturity Assessment – A Maturity Assessment evaluates the attributes of an organization’s processes and methods to determine the ability to consistently and continuously contribute to achieving

organizational objectives. Organizations with a high ability to contribute to these objectives, are considered mature.

Mean Time to Recover (MTTR) – The measure of the mean duration for restoration/fix of a build failure.

Mission Critical – Any telecommunications or information system that is defined as a national security system (Federal Information Security Management Act of 2002 - FISMA) or processes any information the loss, misuse, disclosure, or unauthorized access to or modification of, would have a debilitating impact on the mission of the agency.

National Institute of Standards and Technology (NIST) – A physical sciences laboratory and non-regulatory agency of the United States Department of Commerce. Its mission is to promote American innovation and industrial competitiveness.

Operational Controls – The security controls (e.g., safeguards or countermeasures) for an information system that primarily are implemented and executed by people (as opposed to systems).

Performance (Compliance) – Reviews, monitors, and assesses the activities and outcomes generated, against the expectations set by the board. Also, sustainability with a focus on strategy and creating a business model that can survive in both predictable and turbulent times.

Performance Progress Report (PPR) – A standard, government-wide performance progress reporting format used by Federal agencies to collect.

Plan of Action and Milestones (POA&M) – A document that identifies tasks needing to be accomplished. It details resources required to accomplish the elements of the plan, any milestones in meeting the tasks, and scheduled completion dates for the milestones.

Program – A program is the process of translating broadly stated mission needs into a set of operational requirements from which specific performance specifications are derived. A program consists of a functional area that supports a mission and has associated information systems and budgetary resources. A program is an organized set of activities directed towards a common purpose, objective, goal, or understanding proposed to carry out responsibilities assigned to the organization.

Program Management Controls – Complement the security controls of an information system by focusing on the organization-wide information security requirements that are independent of any particular information system and are essential for managing information security programs. Organizations are required to implement security program management controls to provide a foundation for the organization's information security program. May also be deemed as common controls by the organization since the controls are employed at the organization level and typically serve multiple systems.

Progress – The movement towards a refined, improved, or otherwise desired state, with given metrics, milestones, and deliverables for the capability of the process.

Project – Set of activities or tasks which must be completed in order to arrive at a particular goal or outcome, which typically has a defined start and end date.

Recovery Time Objective (RTO) – The overall length of time an information system's components can be in the recovery phase before negatively impacting the organization's mission or mission/business functions.

Remediation – The act of correcting a vulnerability or eliminating a threat. Three possible types of remediation are installing a patch, adjusting configuration settings, or uninstalling a software application.

Risk – A measure of the extent to which an entity is threatened by a potential circumstance or event, and typically a function of: (1) the adverse impacts that would arise if the circumstance or event occurs, and (2) the likelihood of occurrence.

Note: Information system-related security risks are those risks that arise from the loss of confidentiality, integrity, or availability of information or information systems and reflect the potential adverse impacts to organizational operations (including mission, functions, image, or reputation), organizational assets, individuals, other organizations, and the Nation.

Risk Assessment – The process of identifying, prioritizing, and estimating risks. This includes determining the extent to which adverse circumstances or events could impact an enterprise. Uses the results of threat and vulnerability assessments to identify risk to organizational operations and evaluates those risks in terms of likelihood of occurrence and impacts if they occur. The product of risk assessment is a list of estimated potential impacts and unmitigated vulnerabilities. Risk assessment is part of risk management and is conducted throughout the Risk Management Framework (RMF).

Risk Management Framework (RMF) – A structured approach used to oversee and manage risk for an enterprise.

Safeguards – Protective measures prescribed to meet the security requirements (i.e., confidentiality, integrity, and availability) specified for an information system. Safeguards may include security features, management constraints, personnel security, and security of physical structures, areas, and devices. Synonymous with security controls and countermeasures.

Scaled Agile Framework® (SAFe®) – A set of organizational and workflow patterns for implementing agile practices at an enterprise scale. The framework is a body of knowledge that includes structured guidance on roles and responsibilities, how to plan and manage the work, and values to uphold.

Security Category – The characterization of information or an information system based on an assessment of the potential impact that a loss of confidentiality, integrity, or availability of such information or information system would have on organizational operations, organizational assets, individuals, other organizations, and the Nation.

Senior Agency Information Security Officer (SAISO) – Official responsible for carrying out the Chief Information Officer responsibilities under the Federal Information Security Management Act (FISMA) and serving as the Chief Information Officer's primary liaison to the agency's authorizing officials, information system owners, and information system security officers.

Software Development Life Cycle (SDLC) – A process used by the software industry to design, develop and test high quality software. It is a framework that defines the steps involved in the development of software at each phase.

Software Lifecycle Management (SLM) – Helps organizations maximize the value of their software and cloud portfolios.

Value Stream Management – A lean business practice that helps determine the value of software development and delivery efforts and resources. It also helps to improve the flow of value to the organization, while managing and monitoring the software delivery life cycle from end-to-end.

Value Stream Mapping (VSM) – A visual tool that displays all critical steps in a specific process and easily quantifies the time, value, and volume taken at each stage.

Workflow Process – A series of sequential tasks that are carried out based on user-defined rules or conditions, to execute a business process. It is a collection of data, rules, and tasks that need to be completed to achieve a certain business outcome.

Appendix 1 – Sample Critical Success Factors and Key Performance Indicators (KPIs)

Based on this example, here are samples of Critical Success Factors and Key Performance Indicators (KPIs) that are referenced within the value requirement to deploy value stream:

Critical Success Factors	Key Performance Indicators (KPIs)
Improve Quality	Number of escaped defects % of actual *versus* planned executed tests % of critical defects found early in unit testing *versus* UAT
Improve Project and Feature Execution	% of projects (project tasks, stories, other demand requests) on time % of healthy projects (projects without unresolved urgent issues) Deviation of planned to actual work hours Number of identified issues Number of opened risks Amount of backlog/work-in-process Arrival and departure rate for work
Improve Stewardship of IT Investment	% of actual *versus* planned project cost % of change in project cost % of budget at risk
Increase Automation Adoption	% of automated tests
Achieve Development Process Excellence	% of requirements tested, authorized, completed % of requirements traced to tests % of reviewed requirements % of successful builds % of changes resulting in Incidents Ratio of detected to closed defects at release

Critical Success Factors	Key Performance Indicators (KPIs)
Improve Early Life Success of Releases	% of Incidents during warranty period % of successful/unsuccessful deployments for the project % of emergency changes Pass rates on UAT/validated requirements
Operations and Development Collaboration	Trend on early life support/UAT success metrics % rework
Improve Financial Visibility	Planned cost *versus* actual cost
Maintain a Linkage between Business Services and IT Initiatives	Aggregate (roll up) service development costs by business service
High Quality Service Design Specifications at the Outset	% reduction in the rework required for new or changed service solutions in subsequent lifecycle stages
Integration Test Success	Trend on the number of installation errors in all the packages in the integration environment Number of applications or services that require exceptions outside of the existing infrastructure portfolio
Design-Review to Ensure Application Design Complies with all Policies, including Security	Number of application designs that pass a security policy review
Early Testing of Applications for Security Vulnerabilities	% of severity 1 security defects fixed before application is released