



VA DevOps Maturity Model (DOMM)

DOMM Implementation Plan
and Industry Best Practices

Final Product

3/23/20

Agenda

- Executive Summary
- Themes and Epics Driving the DOMM
- Implementing the DOMM
- Summary

Back-Up

- *Industry Best Practices*
- *Review of Current VA DevOps Maturity Model*
- *Expanded Team Transformation Pipeline with Custom DOMMs*

Executive Summary

VA requires **an accurate measurement tool to equip Lean, Agile, DevSecOps coaches and teams.** Armed with this customizable tool, coaches can lead inexperienced cross-functional teams through a Team Transformation Pipeline.

A DevOps Maturity Model (DOMM), when designed properly, can be the needed measurement tool. **The DOMM can be used to measure the progress of coaches and teams as they pass through the Transformation Pipeline.**

A separate cross-functional DOMM implementation team continues to refine the tool, and the **DOMM becomes increasingly accurate in its ability to score value delivery capability and correlate the score with real Customer satisfaction.**

The DOMM is a powerful tool when leaders can validate that changes in Team Performance/Maturity Levels correlate **with Increased Customer Satisfaction**

Customer Satisfaction

5

4

3

2

1

Transforming DevSecOps Teams

Team Transformation Pipeline

DOMM Level

5

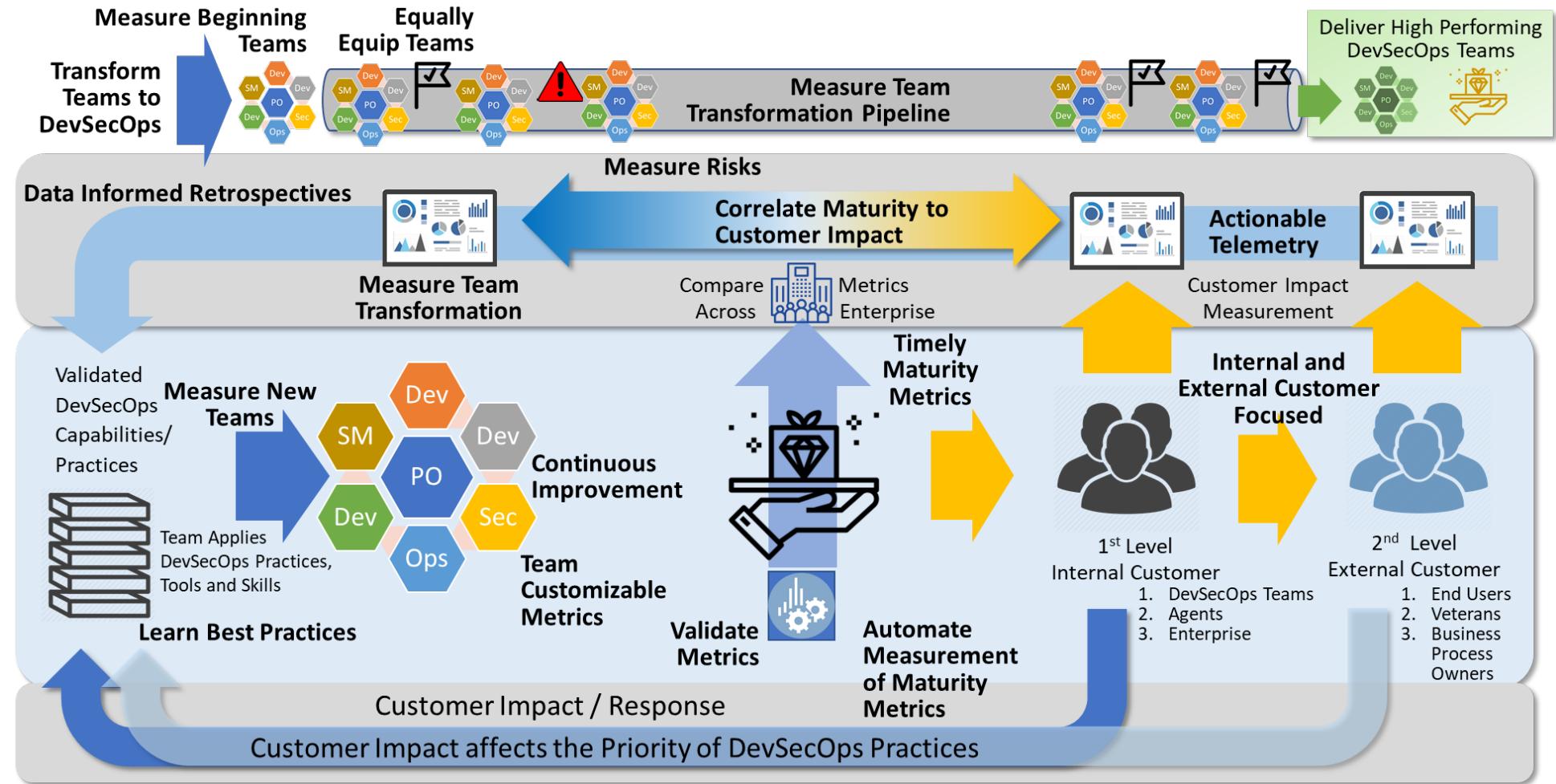
4

3

2

1





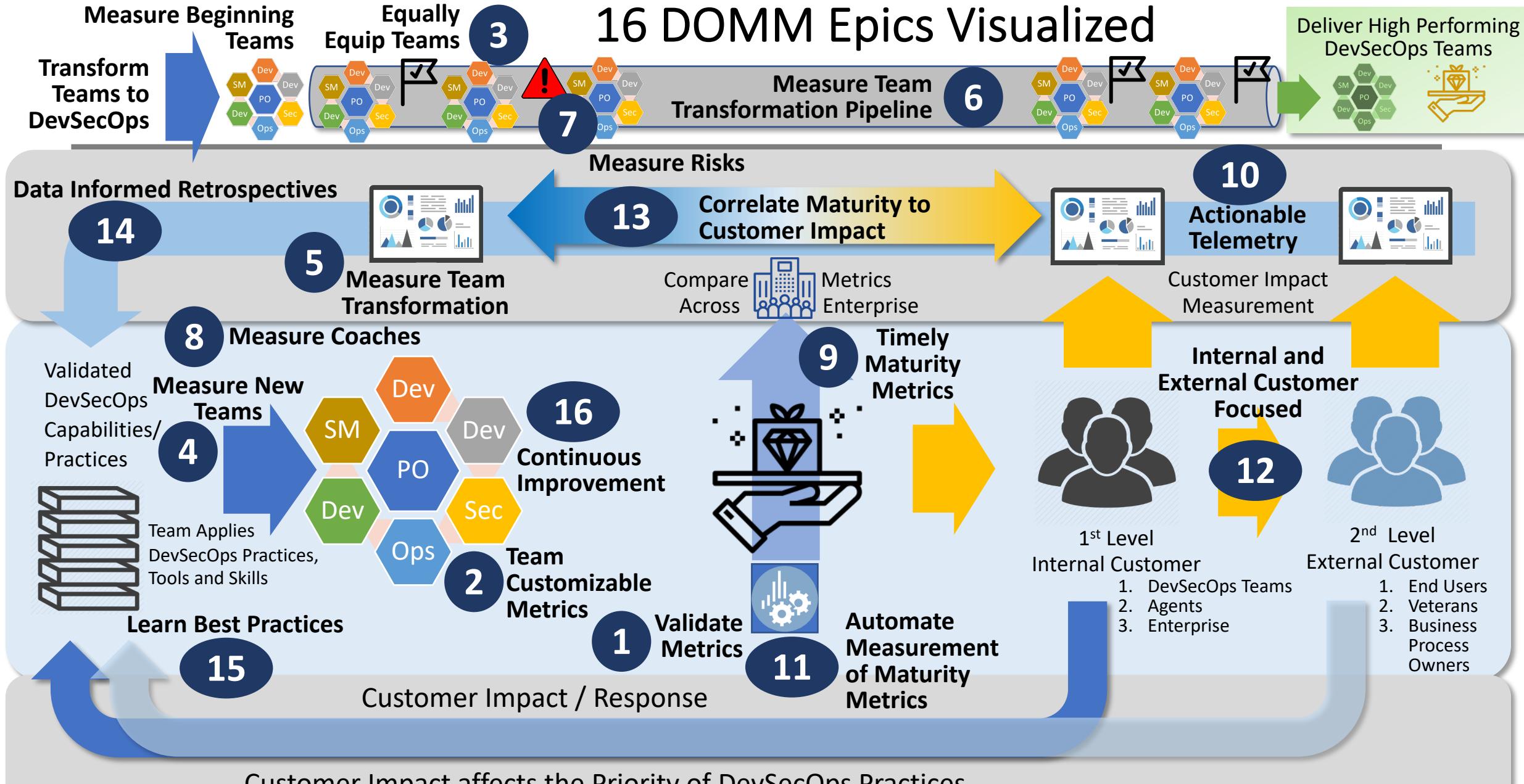
Themes and Epics Driving DOMM

Accurate Measurement, Flow, Feedback, and Continual Improvement

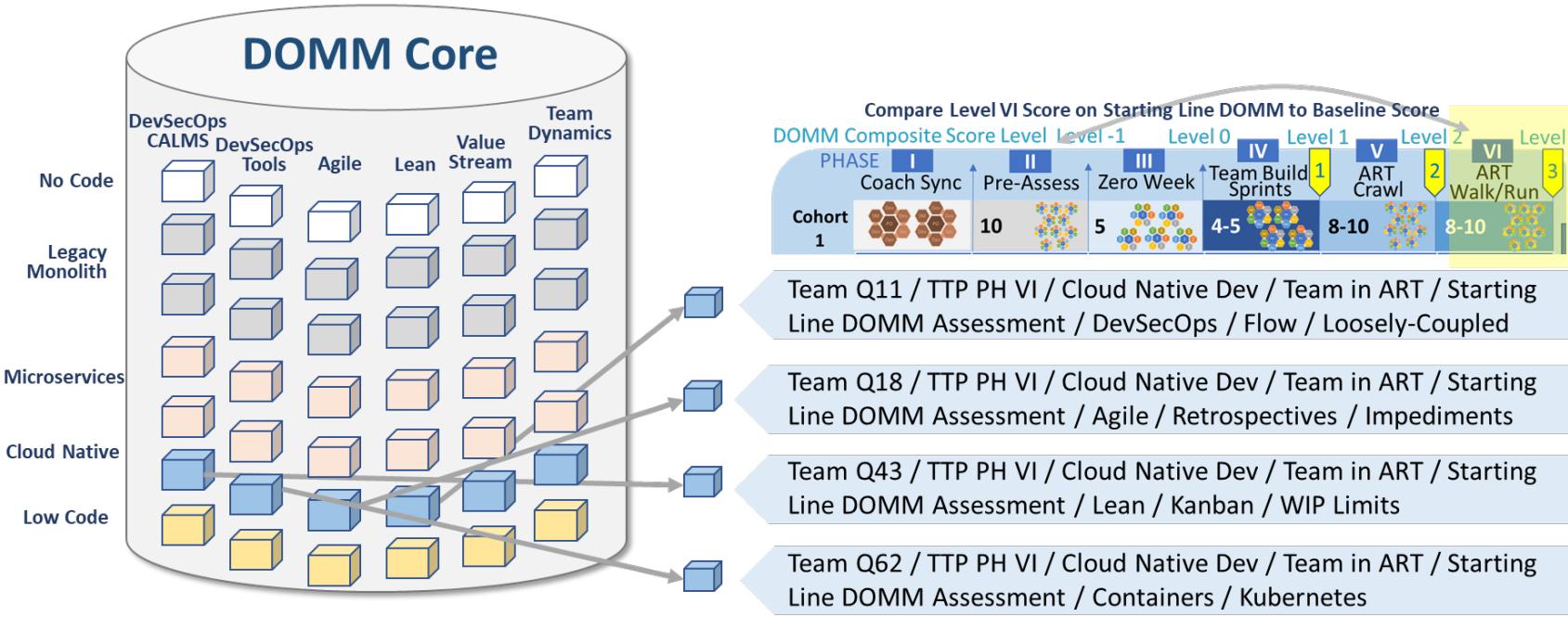
Stakeholder Feedback – What We Need From a DOMM Tool

Themes	Accurate Measurement	Flow	Feedback	Continual Improvement
Epics	1 Validate Measurements 2 Customize Assessment by Team Types 3 Provide Similar Resources to Each Team 4 Measure Beginning Teams	5 Measure Team Transformation 6 Measure the Team Transformation Pipeline 7 Measure Risks to Adoption 8 Measure Coaches	9 Timely Maturity Measurements 10 Composite Scores/ Actionable Telemetry 11 Automated Assessments 12 Internal and External Customer Focused	13 Correlate Maturity to Customer Impact 14 Conduct Data-Informed Retrospectives 15 Validate/Learn Best Practices 16 Encourage Continuous Improvement

16 DOMM Epics Visualized



Customer Impact affects the Priority of DevSecOps Practices



DOMM Core and Custom DOMMs

A Customizable and Growing Core of Criteria, Questions, and Data Points

DOMM Theme I – Accurate Measurement

Industry Best Practice

The Six Sigma process improvement framework highlights the importance of measurement and made measurement system analysis the first step after defining the problem.

The purpose of Measurement System Analysis is to qualify a measurement system for use by quantifying its accuracy, precision, and stability.

By ensuring good measurement, improvements can be trusted.

Accurate Measurement

Validate measurements

Customize Assessment by Team Types

Provide Similar Resources to Each Team

Measure Beginning Teams



Implications for DOMM Practices

A neutral party validates measurements through sampling until the validation can be automated

Using the same DOMM assessment for teams w/different missions - will result in distorted scoring for some

Measuring the performance of teams with unequal tools/resources will result in distorted scoring for some

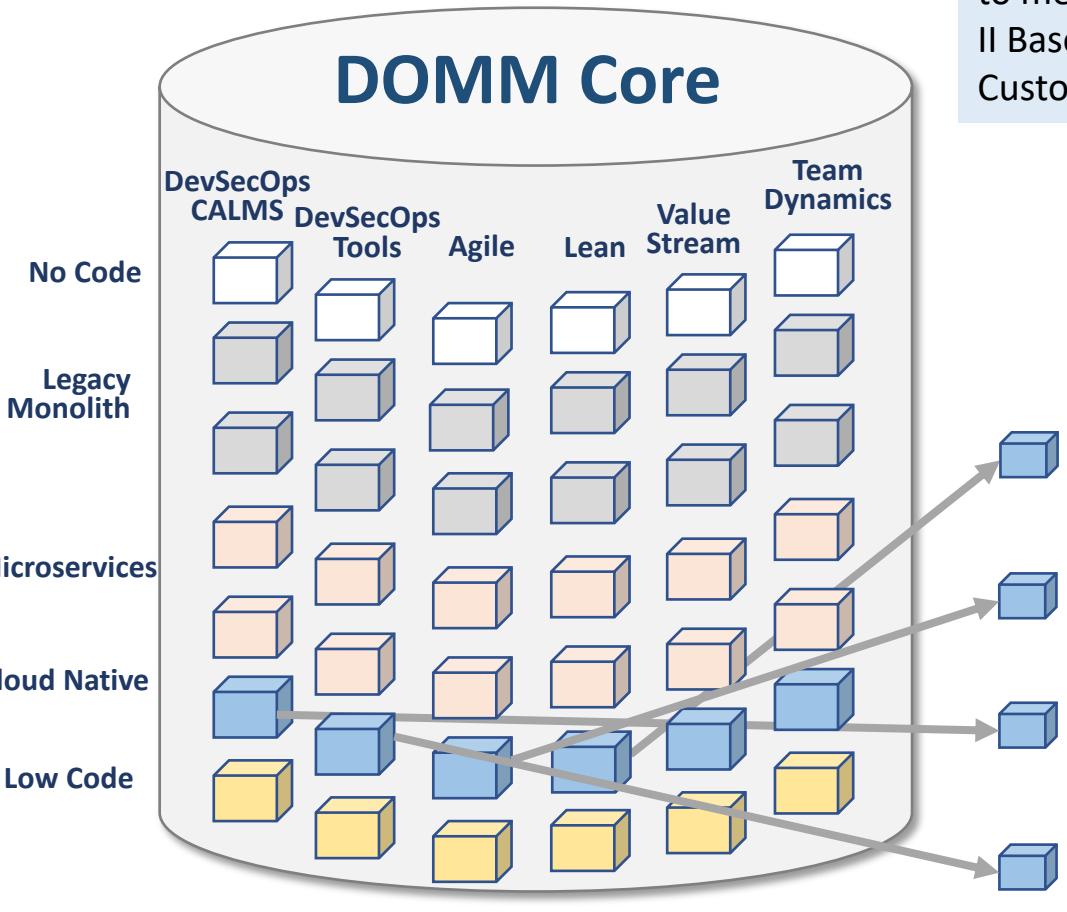
If beginning teams are not measured on a negative scale, leaders will not know how close they are to level 0

Business Impact Example

A team's automated testing has false negatives (no defect identified) and undetected code errors are resulting in avg. 15 defects per deploy. But the automated capture of the # of defects is omitting a certain type of defect resulting in a lower defect per deploy rate.

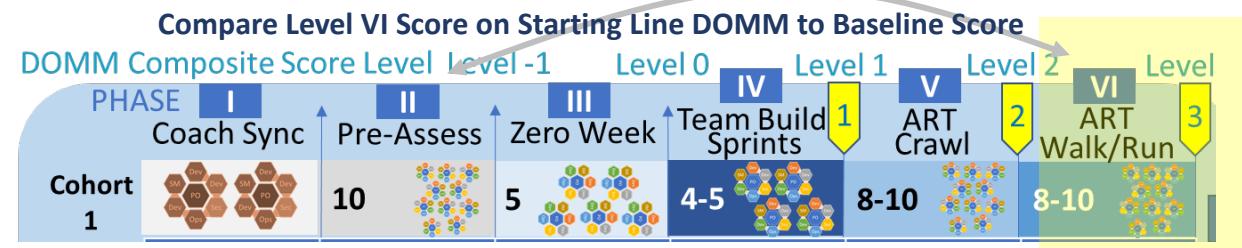
The team falsely reaches DOMM Team Level 1 and joins an ART. The team continues poor quality dev, but now the MTTR* is 3x because of the more complex setting.

Core DOMM and Custom DOMM



Custom DOMM: *Cloud Native “Starting Line” DOMM*

This Example – a Custom DOMM is derived from the DOMM Core Questions by filtering on PH VI (ART Walk/Run) for Cloud Native Teams already part of an ART. Coaches want to measure team improvement on the Starting Line DOMM Assessment from their PH II Baseline to see if Refresher Learning is Needed. Score change is correlated with Customer Impact. Scores are compared across ART to focus Coaching Resources



Team Q11 / TTP PH VI / Cloud Native Dev / Team in ART / Starting Line DOMM Assessment / DevSecOps / Flow / Loosely-Coupled

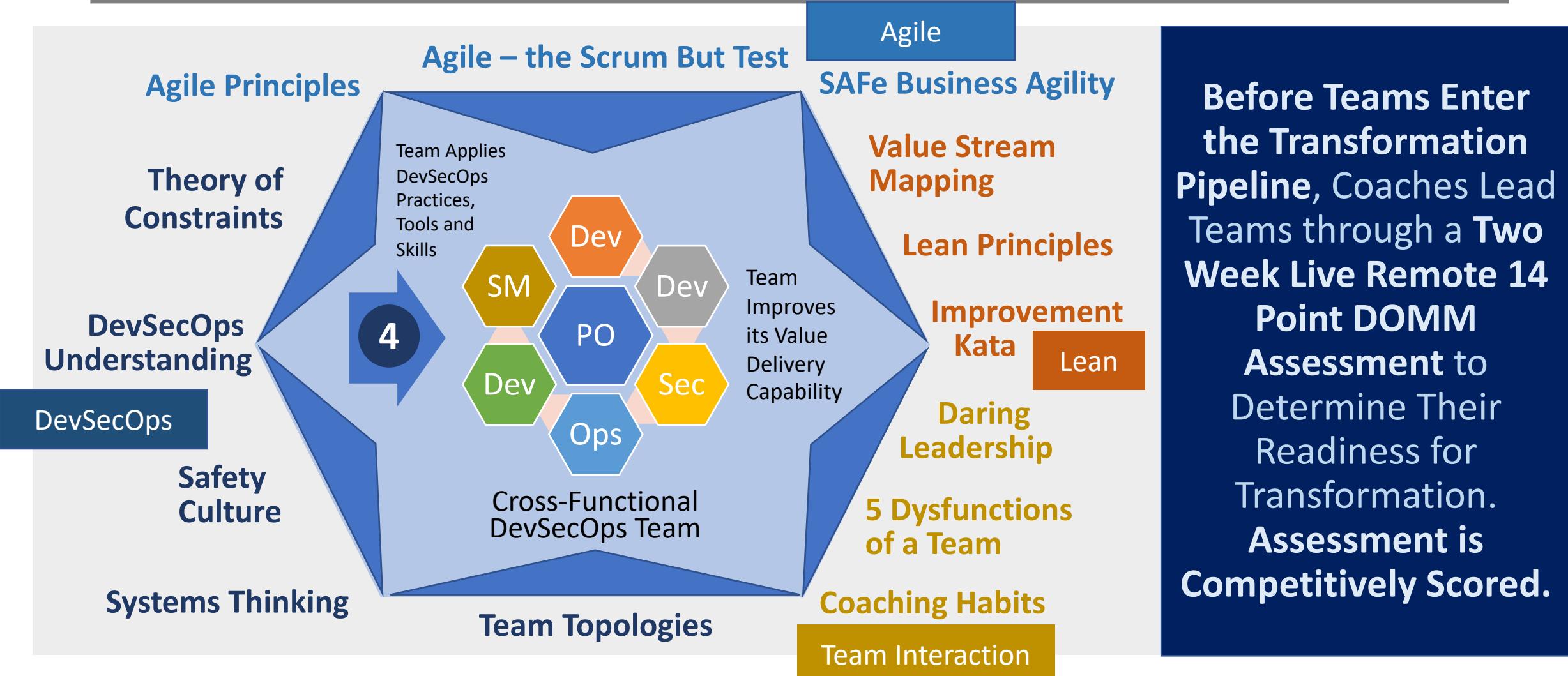
Team Q18 / TTP PH VI / Cloud Native Dev / Team in ART / Starting Line DOMM Assessment / Agile / Retrospectives / Impediments

Team Q43 / TTP PH VI / Cloud Native Dev / Team in ART / Starting Line DOMM Assessment / Lean / Kanban / WIP Limits

Team Q62 / TTP PH VI / Cloud Native Dev / Team in ART / Starting Line DOMM Assessment / Containers / Kubernetes

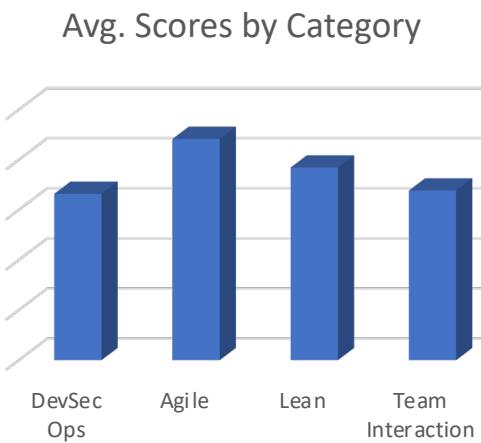
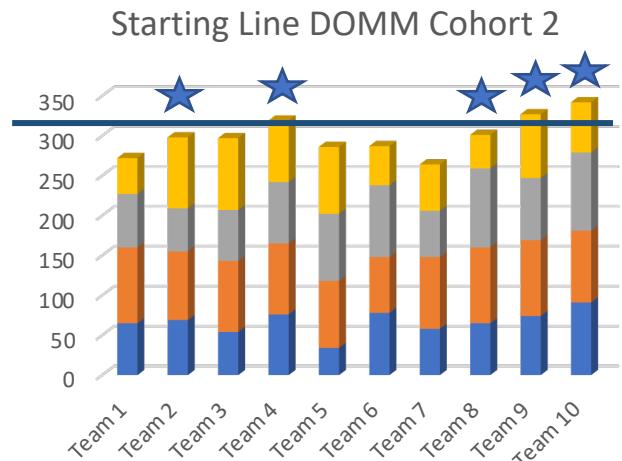
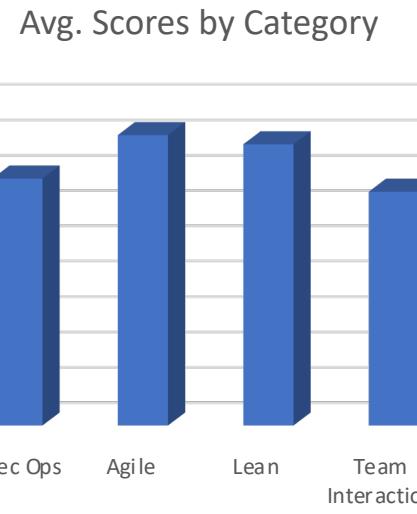
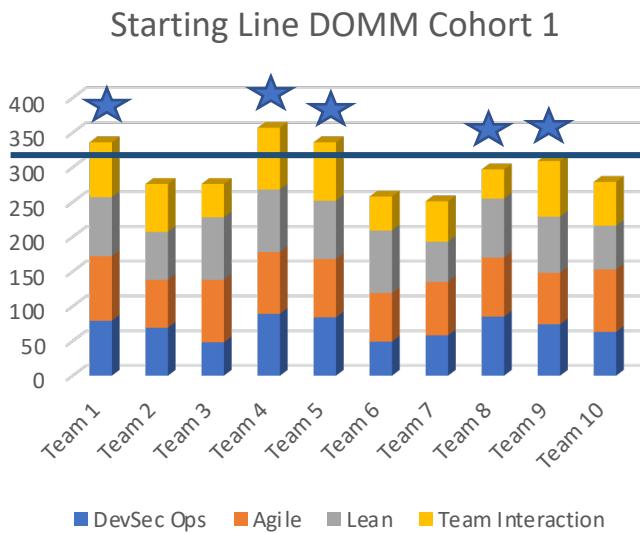
Subset of Actual DOMM Questions Shown Due To Space

The “Starting Line” DOMM Assessment



Example: Starting Line DOMM – 10 Teams Selected for TTP

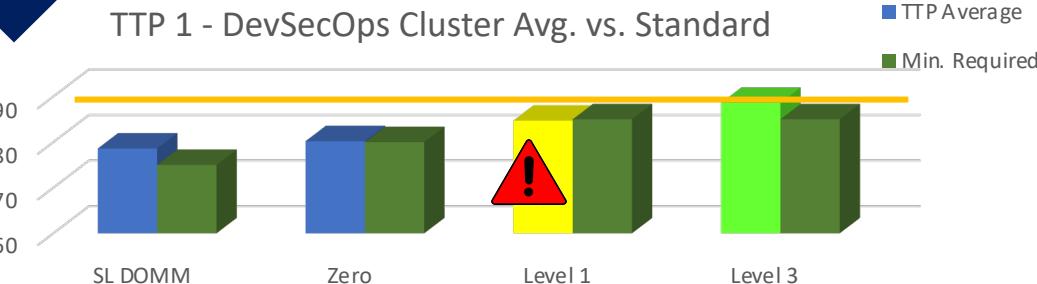
Cohort 1	DevSec Ops	Agile	Lean	Team Interaction	Composite Score	
Team 1	79	93	85	79	84	Selected
Team 2	69	69	69	69	69	Wait List
Team 3	48	90	90	48	69	Wait List
Team 4	89	89	90	89	89.25	Selected
Team 5	84	84	84	84	84	Selected
Team 6	49	70	90	49	64.5	Wait List
Team 7	58	77	58	58	62.75	Wait List
Team 8	85	85	85	42	74.25	Waiver
Team 9	74	74	81	80	77.25	Selected
Team 10	63	90	63	63	69.75	Wait List
	69.8	82.1	79.5	66.1	74.375	74.375
Min. Required	70	75	75	75	75	
Cohort 2	DevSec Ops	Agile	Lean	Team Interaction	Composite Score	
Team 1	65	95	67	45	68	Wait List
Team 2	69	86	54	89	74.5	Waiver
Team 3	54	89	64	90	74.25	Wait List
Team 4	76	89	77	77	79.75	Selected
Team 5	34	84	84	84	71.5	Wait List
Team 6	78	70	90	49	71.75	Wait List
Team 7	58	90	58	58	66	Wait List
Team 8	65	95	99	42	75.25	Waiver
Team 9	74	95	78	80	81.75	Selected
Team 10	91	90	98	63	85.5	Selected
	66.4	88.3	76.9	67.7	74.825	74.825
Min. Required	70	75	75	75	75	



Example: DevSecOps Cluster by Phase w/Telemetry

	DevSec Ops	DevSec Ops	DevSec Ops	DevSec Ops	Average Score	
Teams Entering TTP	SL DOMM	Zero	Level 1	Level 3		
Cohort 1 Team 1	79	82	85	88	84	Selected
Cohort 1 Team 2	69	61	75	85	73	Waiver
Cohort 1 Team 3	76	78	83	86	81	Selected
Cohort 1 Team 4	89	89	90	93	90	Selected
Cohort 1 Team 5	84	82	84	90	85	Selected
Cohort 2 Team 6	65	75	90	91	80	Waiver
Cohort 2 Team 7	74	77	86	85	81	Selected
Cohort 2 Team 8	85	90	85	92	88	Waiver
Cohort 2 Team 9	74	78	77	85	79	Selected
Cohort 2 Team 10	91	90	92	91	91	Selected
	78.6	80.2	84.7	88.6		
Min. Required	75	80	85	85		
Variation - Sdev	8.11	8.32	5.18	3.01		

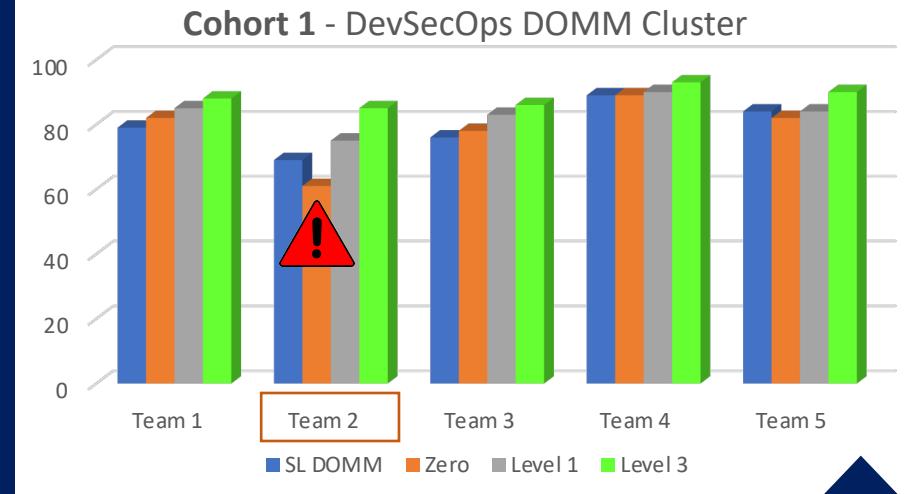
As the DOMM and associated telemetry mature, automated alerts can be sent to teams, coaches, PLM, etc. when teams are at risk in the TTP



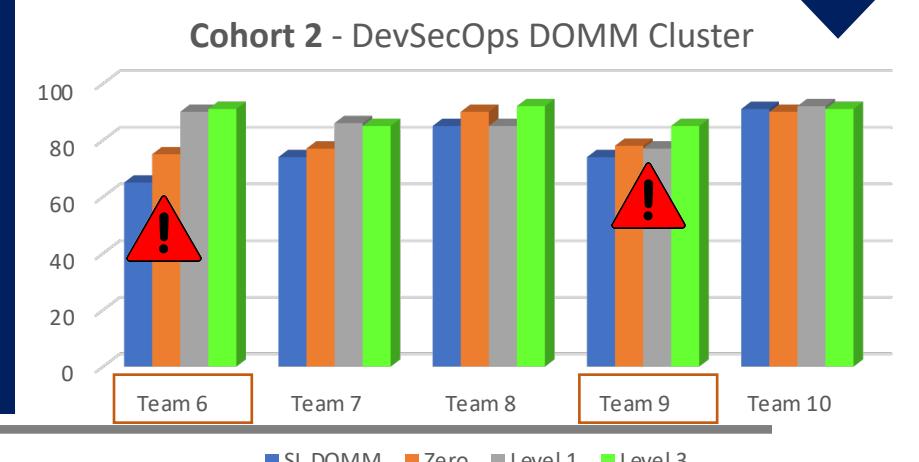
3/23/20

LTS Proprietary and Business Sensitive

After the 10 teams are selected to enter the TTP, the DevSecOps SMEs on the Cohort 1 and 2 Coaching Teams review Team scores on the DevSecOps Cluster DOMM Metrics at the end of each phase and focus efforts on lower performers to achieve cohort/TTP goals (85) and Reduce Variation (from 8.1 to 3.0 StDev.)



Dashboard Tiles Provide Insight into Team Progress for all Stakeholders



12

DOMM Theme II - Flow

Industry Best Practice

In the Toyota Production System (TPS), continuous-flow processing means the plant/process is designed so that work proceeds directly from one process to the next (without any detours into storage) - from raw materials plants through machining, to assembly, to distributors, dealers, and customers.

This is different than batch processing where large batches of work are completed and stored in queues, before being passed to the next process.

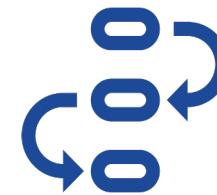
Flow

Measure Team Transformation

Measure the Team Transformation Pipeline

Measure Risks to Adoption

Measure Coaches



Implications for DOMM Practices

We want to measure how teams are progressing to a single-piece flow practice of daily code integration

We want to create a single piece flow process for teams to transform to LAD and finally to ARTs

We identify bottlenecks for teams in their delivery pipelines and in the TTP to maintain transformation velocity

Coaches have their own DOMM focused on their ability help teams achieve flow and keep TTP flowing

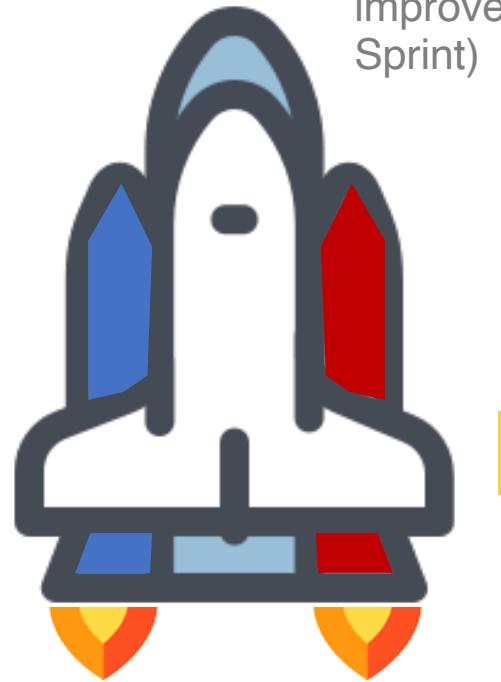
Business Impact Example

Coaches observe a team DOMM dashboard which shows significant wait time (bottleneck) in their CI/CD pipeline for change mgmt. They complain DevSecOps doesn't work – the same bottlenecks still slow deployment speed.

Coaches ask the team questions about the change management process, support their root cause analysis and help escalate the risk to the RTE* who negotiates a more automated Change Mgmt. process and expanded standard changes.

The DOMM has to Measure Coaching Teams To Help Them Improve So Transformation Teams Can Improve

LAD Teams need intensive support from cross-functional coaching teams to **achieve sufficient thrust to break old ways of doing things and develop game-changing velocity.**



*Agile and DevOps-focused
Coaching Teams Fuel
Successful Lift-Off*

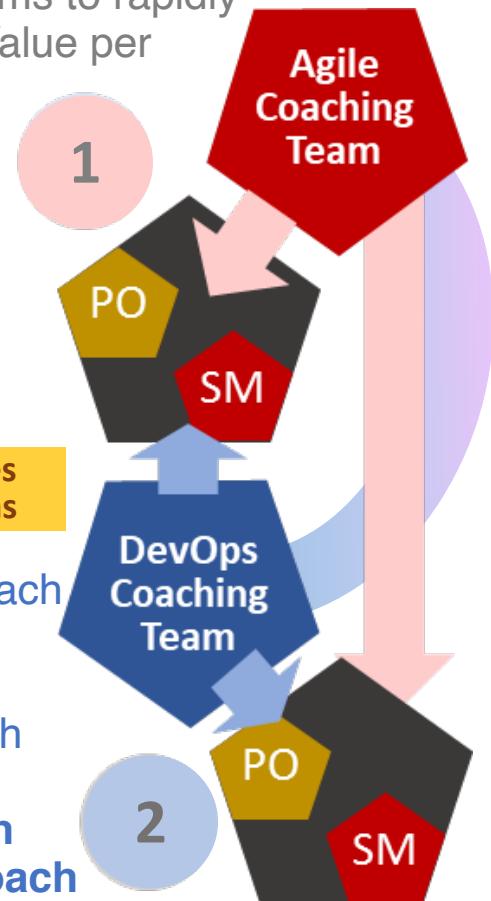
The Agile Coaching Team works with teams to rapidly improve velocity (measured by Business Value per Sprint)

1. Scrum Coach
2. Product Owner
3. UX Coach
4. Bus. Process/LSSBB
5. Development Quality
6. **VA Scrum Coach**
7. **VA Product Owner**

Coaching Teams also mentor VA Coaches who are Integrated with Coaching Teams

1. TOC**/Value Stream Coach
2. Test Automation Coach
3. Continuous Del. Coach
4. Telemetry/Visibility Coach
5. Data Analyst
6. **VA Value Stream Coach**
7. **VA Test Automation Coach**

The DevOps Coaching Team works with both teams to ensure they have the tools needed for continuous delivery



DOMM Theme III - Feedback

Industry Best Practice

Kim, Humble, and Farley highlight the power of feedback in DevOps. Feedback loops are sets of relationships between entities whereas a change in one entity causes a change in another entity and that change eventually leads to a change in the first entity.

Continuous delivery makes it economic to work in small batches. This means we can get feedback from users throughout the delivery lifecycle based on working software.

Feedback

Timely Maturity Measurements

Composite Scores/
Actionable Telemetry

Automated Assessments

Internal and External Customer Focused



Implications for DOMM Practices

Teams will be changing fast. DOMM Data on a team's progress must be recent to be relevant and accurate

Composite DOMM scores synthesize multiple team attributes to allow leaders to quickly see teams who need help

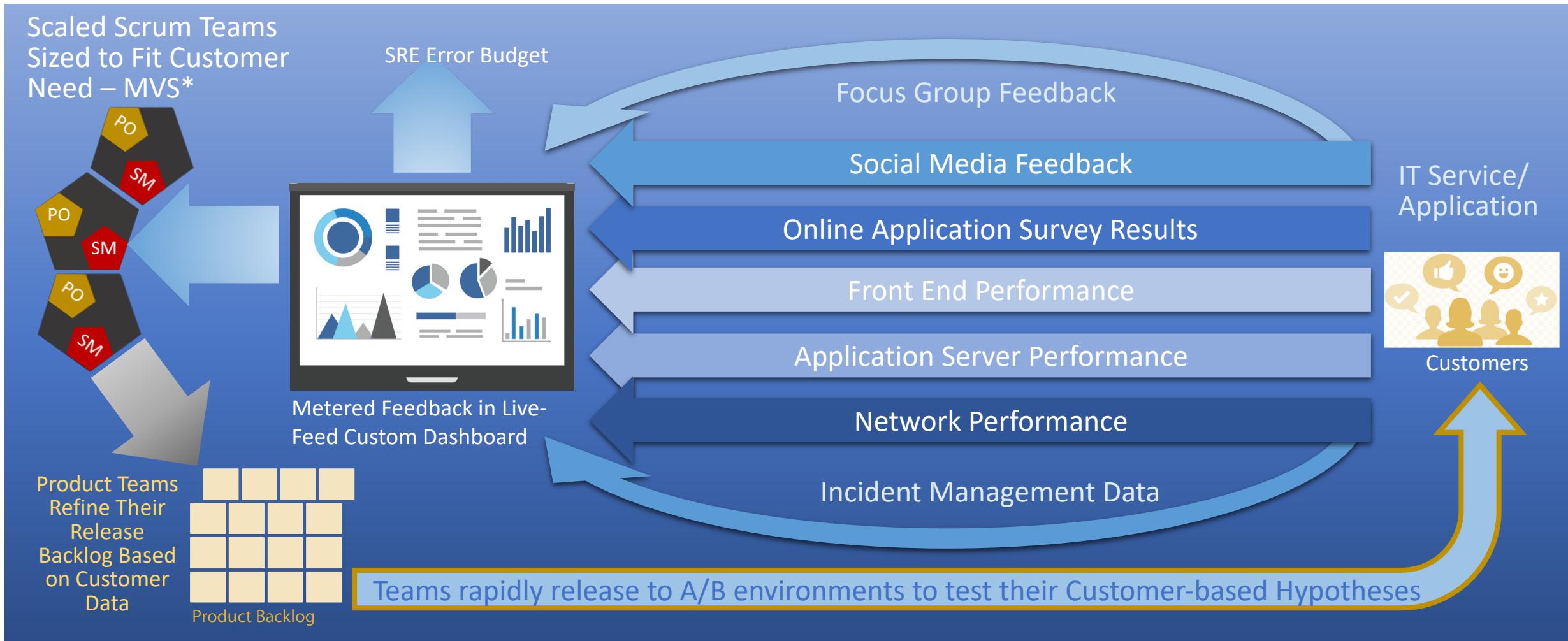
Automating as many of the DOMM assessments as possible ensures rapid feedback to teams, coaches, and leaders

Feedback must be linked to Customer Satisfaction and the Customer can be in IT, an IT Product user, or an end user

Business Impact Example

A team in the TTP logs its happiness scores daily in a simple app. The TTP Telemetry has a tile that shows trends in the happiness score. A Scrum Master Coach notices a severe dip in happiness scores for one of the teams over the last two days. She discovers a team conflict has arisen; they did not have the skills to work it out – other change metrics were starting to nose-dive. She was able to quickly bring in a team coach to help them resolve the issue and get back to happy.

Building Game-Changing Customer Feedback Channels

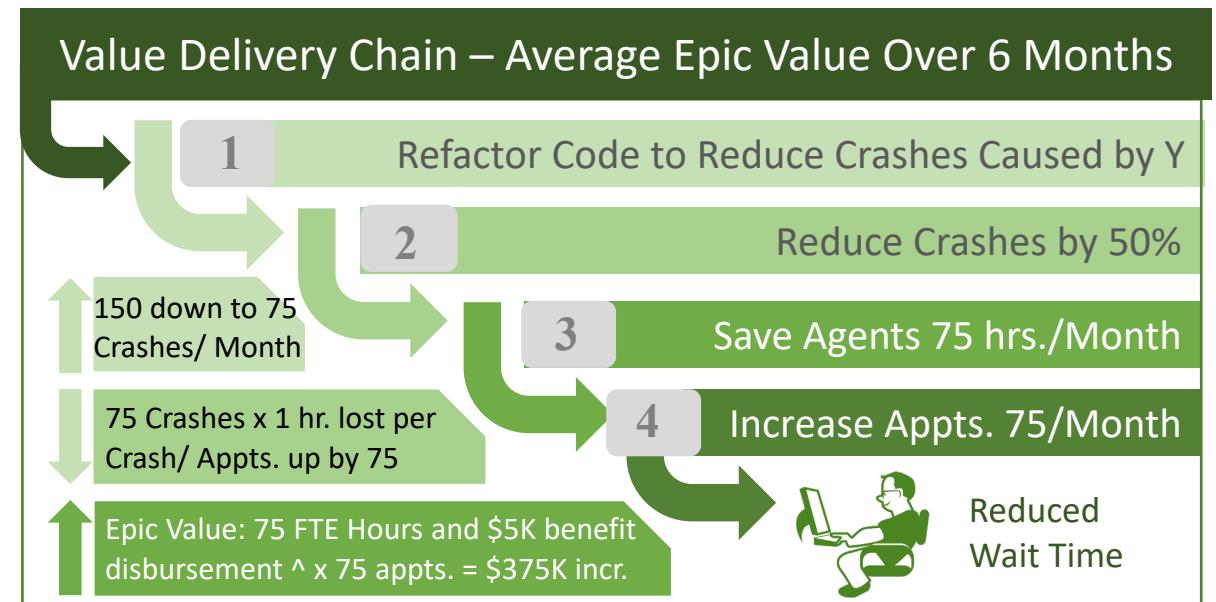
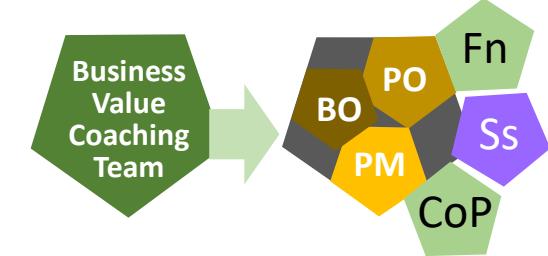


*MVS – Minimum Viable Scale

Transformation Teams Must Determine their Baseline Value Delivery in Order to Correlate Later DOMM and Value Gains

- During the Zero Week, Coaching Teams lead Transformation Teams through UX Journey Mapping and Value Stream Mapping
- A Shared Business Value Coaching Team works with Coaching Teams, Transformation Teams, and Business Owners (BOs) to:
 - Identify the **chain of value** leading to the customer (internal, external or both)
 - **Assign value** to the end customer experience and all links in the value chain – neutral party validates.
 - **Assign business value to Epics** related to the links and distribute to User Stories
 - **Develop Key Results related to org. Objectives** which act become acceptance criteria for Epics

1. Product Manager Coach
2. Org. Finance & Acct. SME
3. Bus. Modeling/Analytics
4. Data Analyst/Scientist
5. UX SME – User Feedback



DOMM Theme IV – Continuous Improvement

Industry Best Practice

W. Edwards Deming's 14 points: #5 Improve constantly and forever every process for planning, production and service.

By applying the Plan, Do, Check, Act (PDCA) or Deming cycle to the DOMM, we continuously upgrade the DOMM base on the feedback received from systems, customers, and experiments. In Japanese, this continuous improvement process is called "Kaizen" and implies incremental changes over time.

Continuous Improvement

Correlate Maturity to Customer Impact

Conduct Data-Informed Retrospectives

Validate/Learn Best Practices

Encourage Continuous Improvement



Implications for DOMM Practices

The DOMM is a measurement tool. It is not completely calibrated. As Customer Impact is correlated with DOMM we adjust the DOMM to better align it.

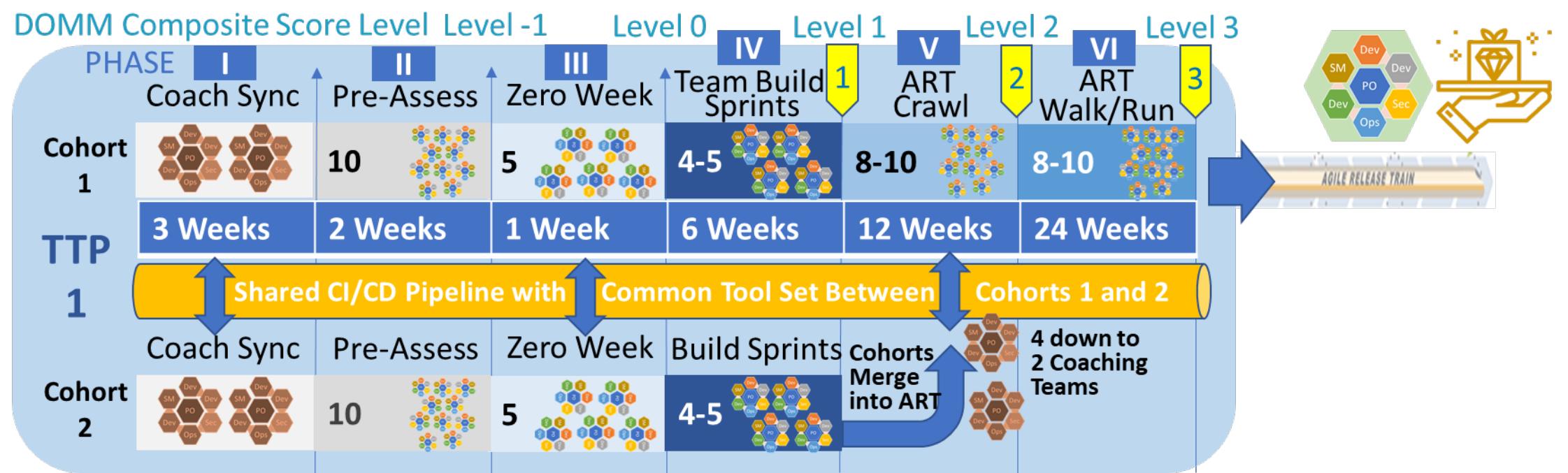
The DOMM is formatted for easy use by teams during retrospectives to identify areas of concern for improvement.

As teams learn and leaders experiment with the TTP, DOMM scores help to validate the improved practice/skill.

The DOMM levels are created so that normal teams cannot be satisfied – the highest scoring levels take years to achieve.

Business Impact Example

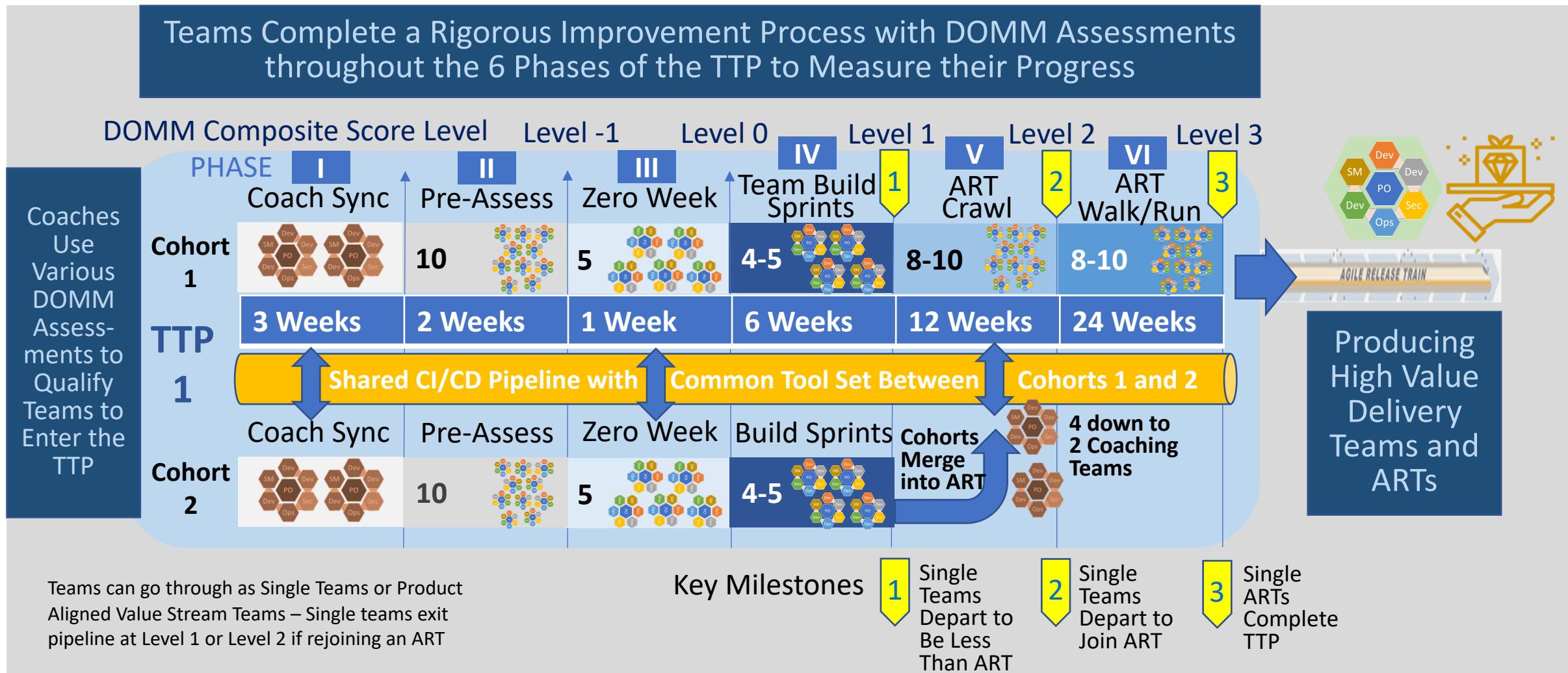
Teams and ARTs struggle to identify real areas for improvement during their Retrospectives and Inspect and Adapts. The DOMM team develops multiple short DOMM assessments teams and ARTs can access on line to walk through different questions. The questions are linked to DOMM measures they are tracking. So teams can compare the goal levels with their current levels, prioritize for improvement and create a User Story for the next sprint.



Implementing the DOMM

In the Context of the Team Transformation Pipeline

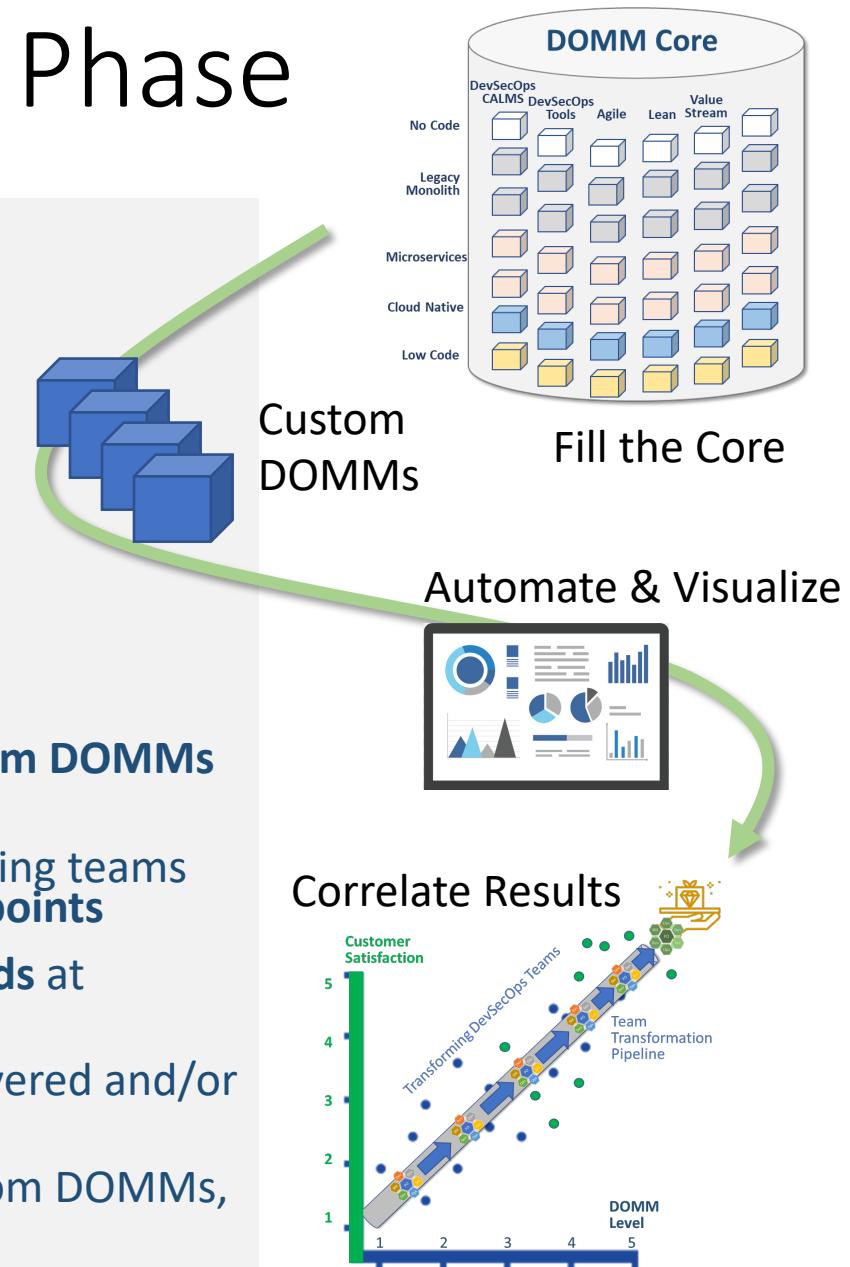
DOMM is Implemented in Parallel with TTP Implementation



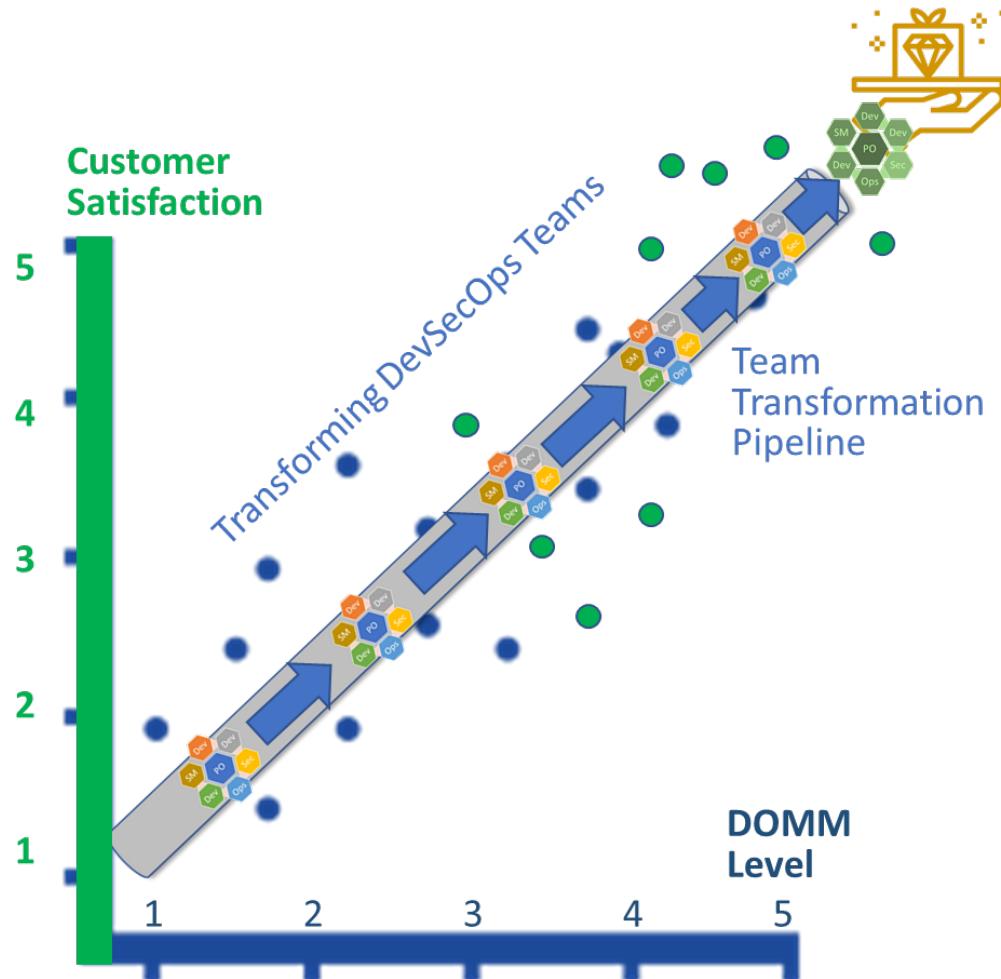
Not Shown – 2 Control Group Teams: 1 without TTP and 1 without TTP and DOMM

DOMM Implementation by TTP Phase

1. Form a **cross-functional DOMM Implementation Team**
2. Fill the **DOMM Core**
3. Categorize the Core criteria/questions across key types
4. Filter the DOMM Core to produce **Custom DOMMs**
5. Test the Custom DOMMs
6. Review the **Test results** with the Coaching Teams / Other SMEs
7. Refine the Custom DOMMs / draft Custom DOMMs for later phases
8. Coaching Teams use the Custom DOMMs to **establish baseline metrics**
9. The DOMM Implementation Team continues to **build and refine the Custom DOMMs** and maintain the **response Database**
10. Starting in Phase III – Zero Week – The DOMM team works with the Coaching teams and Transformation teams to start **automating critical performance data points**
11. The DOMM team integrates and **display DOMM data in various dashboards** at different org. levels
12. **Compare** the Transformation Team's improvements to business value delivered and/or Customer Satisfaction metrics to their increases in DOMM Level
13. The DOMM team **continues to apply LAD principles** to DOMM Core, Custom DOMMs, and DOMM delivery and data analysis.



Correlating DOMM Improvements with Customer Impact



The DOMM is a powerful tool when leaders can validate that changes in Team Performance/Maturity Levels correlate well with Increased Customer Satisfaction

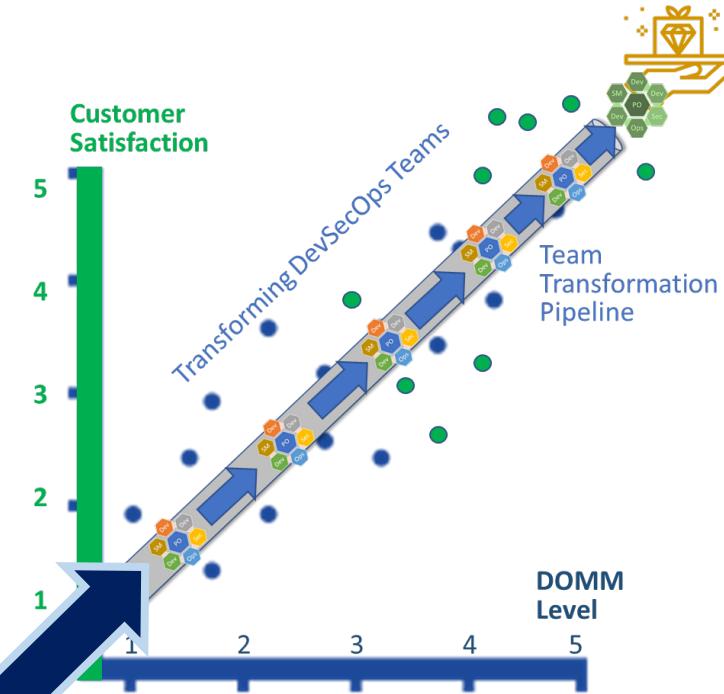
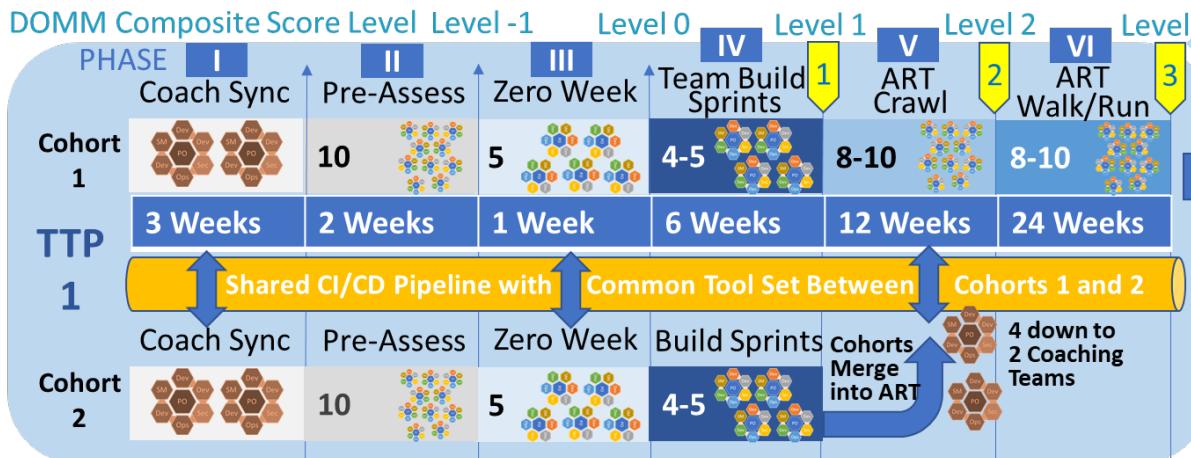
Team Baselines on Custom DOMMs and Individual Questions/Criteria allow VA to consider correlations between improvements in DOMM levels and Customer value delivery (as measured by satisfaction or NPS or similar).

The DOMM helps determine if the investment in LAD transformation paid off for Customers.

Example: Team A's Value Stream-evaluated Avg. Business Value Delivery increased from 10 hrs. / week to 100 hrs. a month as their DOMM increased from level -1 to level 3.

Summary

Measuring DevSecOps Team performance with a DevOps Maturity Model and relating performance levels to value delivery and Customer Satisfaction are critical to the success of VAs current transformation to a Lean, Agile, DevSecOps IT framework.



Using VA inspired Epics, LTS has developed a simple, intuitive, and feasible plan to implement a DevOps Maturity Model capable of enabling rapid transformation while maintaining high quality, high value-delivery teams.

Back-Up Slides

Industry Best Practices

Major DevOps Maturity Models from Industry

Key Concepts from Surveyed Maturity Models

Speed and Stability

Google/Accelerate 2019 State of DevOps

4 Levels

Low, Medium, High, Elite Performers

1. Deploy Rate
2. Deploy Cycle Time
3. Change Failure Rate
4. Time to Restore

highest performers are twice as likely to meet or exceed their organizational performance goals.

ELITE PERFORMERS

Comparing the elite group against the low performers, we find that elite performers have...



208
TIMES MORE
frequent code deployments



106
TIMES FASTER
lead time from commit to deploy



2,604
TIMES FASTER
time to recover from incidents



7
TIMES LOWER
change failure rate
(changes are 1/7 as likely to fail)

Security

Puppet State of DevOps 2019

6 Levels of Maturity

1. Foundation
2. Normalization
3. Standardization
4. Expansion
5. Automation
6. Self-Service



5 Levels of Security Integration

1. No Integration
2. Minimal Integration
3. Selective Integration
4. Significant Integration
5. Full Integration

Security becomes a shared responsibility across delivery teams that are empowered to make security improvements

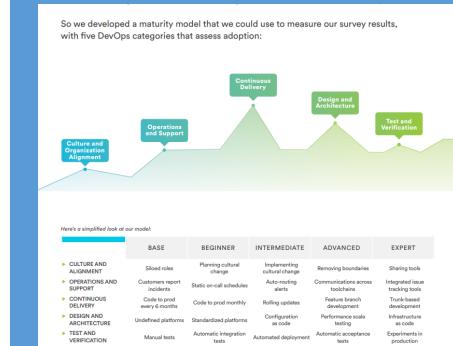
Cultural Change

xMatters/Atlassian Maturity Model

5 Levels of Maturity along 5 Major Criteria

1. Culture and Alignment
2. Operations and Support
3. Test and Verification
4. Continuous Delivery
5. Design and Architecture

Measure Growth in Business Value (Every 3 -9 months vs. Continuous Delivery)



Continuous Improvement

Plutora Maturity Model

4 Levels of Maturity

1. Testing the Water
2. Holding Your Breath
3. Diving in Headfirst
4. Improving Lap Times

*DevOps is about adopting a **mindset of continuous improvement**.*



Enterprise Integration

QA Symphony / Cloud Bees Maturity Model

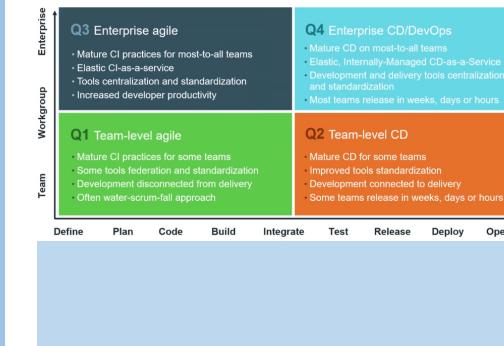
4 Quadrant Model:

Upstream Capability

1. Team Agility
2. Enterprise Agility

Downstream Capability

1. Team DevOps
2. Enterprise CI/CD



Google Accelerate DevOps Maturity – 4 Major Metrics

Key Criteria – from lowest to high		Maturity
Deployment Frequency		
On demand – multiple deploys per day	Elite	
Between once a day and once a week	High	
Between once a week and once a month	Medium	
Between once a month and once every six months	Low	
Lead Time for Changes – code commit to code running in production		
Less than a day	Elite	
Between one day and one week	High	
Between one week and one month	Medium	
Between one month and six months	Low	
Change Failure Rate – changes released to production that result in degraded service		
0-15%	Elite	
0-15%	High	
0-15%	Medium	
46-60%	Low	
Time to Restore – Unplanned outage or service disruption		
Less than an hour	Elite	
Less than a day	High	
Less than a day	Medium	
Between one week and one month	Low	

Core Maturity Model Philosophy: Our research continues to show that the industry-standard Four Key Metrics of software development and delivery drive organizational performance in technology transformations. This year's report revalidates previous findings that it is possible to optimize for stability without sacrificing speed.

Levels are: Elite, High, Medium, and Low Performers

Take Away Important Idea: “For organizations seeking guidance on how to improve, we point to the only real path forward: Start with foundations, and then adopt a continuous improvement mindset by identifying your unique constraint (or set of constraints). Once those constraints no longer hold you back, repeat the process.”

Assessment / Scoring Method – Elite showed BOTH fastest delivery and safest releases. Clustered data and looked - found 2 main clusters of correlated results – Software Delivery & Ops and Productivity. Looked at clusters of other characteristics of Elite performers.

<https://cloud.google.com/devops/state-of-devops/>

Puppet State of DevOps Maturity Model – Security

Key Criteria – from lowest to high	Stage
Monitoring and alerting are configurable by the team operating the service	Stage 0
Reuse deployment patterns for building applications or services	Foundation
Reuse testing patterns for building applications or services	
Teams contribute improvements to tooling provided by other teams	
Configurations are managed by a configuration management tool	
Application development teams use version control	Stage 1
Teams deploy on a standard set of operating systems	Normalization
Teams deploy on a single standard operating system	Stage 2
Build on a standard set of technology	Standardization
Individuals can do work without manual approval outside team	Stage 3
Deployment patterns for building apps/services are reused	Expansion
Infrastructure changes are tested before deploying to production*	
System configurations are automated	Stage 4
Provisioning is automated	Automated
System configs are in version control*	Infrastructure
Infrastructure teams use version control*	Delivery
Application configs are in version control*	
Security policy configs are automated	
Incident responses are automated	Stage 5
Resources are available via self-service	Self-Service
Applications are rearchitected based on business needs*	
Security teams are involved in technology design and development*	Business Sensitive

Core Maturity Model Philosophy: We found the organizations that had evolved the furthest on their DevOps journeys were successfully applying automation to security considerations. At Stage 4 of DevOps evolution, for example, teams were automating security policy configurations. This helped teams progress to Stage 5, where we found the key practice of automated incident response. We also learned that organizations at Stage 5 involved their security teams in technology design and deployment.

Levels are: Stages 1-5 with the highest level being “Self-Service”. Firms that have achieved higher levels of security integration are much more likely to be at a high stage of DevOps evolution.

Take Away Important Idea: Security becomes a shared responsibility across delivery teams that are empowered to make security improvements. Security teams are able to act in an advisory role, leading to time-saving and security-enhancing capabilities such as automated incident response. Further, these teams were able to implement transparent security policies as code.

Assessment / Scoring Method – Teams or leaders compare with characteristics of various stages.

https://media.webteam.puppet.com/uploads/2019/11/2019-state-of-devops-report-puppet-circleci-splunk_sml-1-1.pdf?_ga=2.99531606.993443541.1581997312-1930196998.1581997312

xMatters/Atlassian DevOps Maturity Model – Culture Change

Key Criteria	Major Criteria Categories	Core Maturity Model Philosophy: DevOps is a culture and environment where building, testing, and releasing software can happen rapidly, frequently, and more reliably. Really just building on State of DevOps Report. 63% of Atlassian companies have adopted DevOps practices.
Dev and Ops Collaboration?	Culture and Alignment	
What can your company monitor?	Operations and Support	
Common Set of Tools – Dev and Ops?	Culture and Alignment	
Monitoring Tools Predict Incidents?	Operations and Support	
How is Tool generated data shared between Ops and Dev?	Culture and Alignment	
Deploy Frequency	Continuous Delivery	
How do you learn about incidents?	Operations and Support	
What types of testing are used?	Test and Verification	
Typical Release Process?	Test and Verification	
Primary Structure for Dev. Software?	Design and Architecture	
How do you deploy infrastructure?	Design and Architecture	
How are major incidents handled?	Operations and Support	
How are incidents routed?	Operations and Support	
How do applications perform at release	Test and Verification	
Duplicate tickets created after Maj. Incident started?	Operations and Support	
Manual or automated process to keep stakeholders updated?	Operations and Support	

Plutora DevOps Maturity Model – Continuous Improvement

Key Criteria	Description of Highest Level
Collaborations between teams	Teams collaborate to understand metrics
Automated configuration management	Automated test suite allows deployment soon after code is finished. Deploy 1x a day or more.
Release Management	Know how many defects and impact on system. Ops staff can use hard #s to describe risk of adding new features.
Continuous Integration	Code compliance and quality are determined soon after writing.
Product mindset	Change data is tied to Customer Satisfaction. Backlog prioritization based on hard data.
Compliance difficulty	Compliance org. has input into technical team decisions. Sign off on code shortly after written.
Continuous Improvement Mindset	Continuous Improvement is in the orgs. DNA. Teams can provide data on how much they have improved and over what time window.

Assessment / Scoring Method –

Teams read the description of each level and compare what they are doing to the description. The level description which best fits them is their maturity level. Each level describes how all 7 criteria look for that maturity ranking.

Core Maturity Model Philosophy: Instead of approaching DevOps from a yes/no perspective, it's far better to treat it like a living organism. DevOps is about adopting a mindset of continuous improvement. The maturity of a DevOps organization is another place where that mindset must take hold.

There are a number of facets common to every mature DevOps culture. By naming and understanding them, it's possible to identify areas where a business's culture is strong and areas where that same business is weak.

Levels are: Testing the Water, Holding Your Breath, Diving in Head First, Improving Lap Times

Take Away Important Idea: That's the key trait of DevOps maturity: the ability to recognize places where a team is falling short and identify small changes to make over time in order to fix those things. Everything else flows from that.

<https://www.plutora.com/devops-at-scale/maturity-level>

QA Symphony/Cloud Bees DevOps Maturity Model – 4Qs

Key Criteria	Description of Highest Level
Team Agile Upstream	Practice Agile at Team Level through integration
Team Agile Downstream	Practice Agile at Team Level from Test to Operate
Enterprise Agile Upstream	Practice Agile at Enterprise Level through integration
Enterprise CD/DevOps	Practice Agile at Enterprise Level from Test to Operate
Continuous Integration	Mature for some teams to all teams
Tool Standardization	From limited to centralized and standardized
Dev / Ops Collaboration	Disconnected Dev and Ops to CD as a Service
Deploy Frequency	Some teams to most teams release in weeks/days/hours

Assessment / Scoring Method –

Teams or leaders compare with Quadrants.

Not really a scoring mechanism.

1. Determine current quadrant
2. Establish objectives
3. Map the journey through quadrants
4. Cross chasms, connect quadrants

Core Maturity Model Philosophy: Flexible Approach that points out that big gains in productivity will not occur until you have Enterprise Agile Upstream and Enterprise CD/DevOps. As each maturity quadrant is achieved, the organization grows in productivity.

Drive Change through the DevOps Trinity:

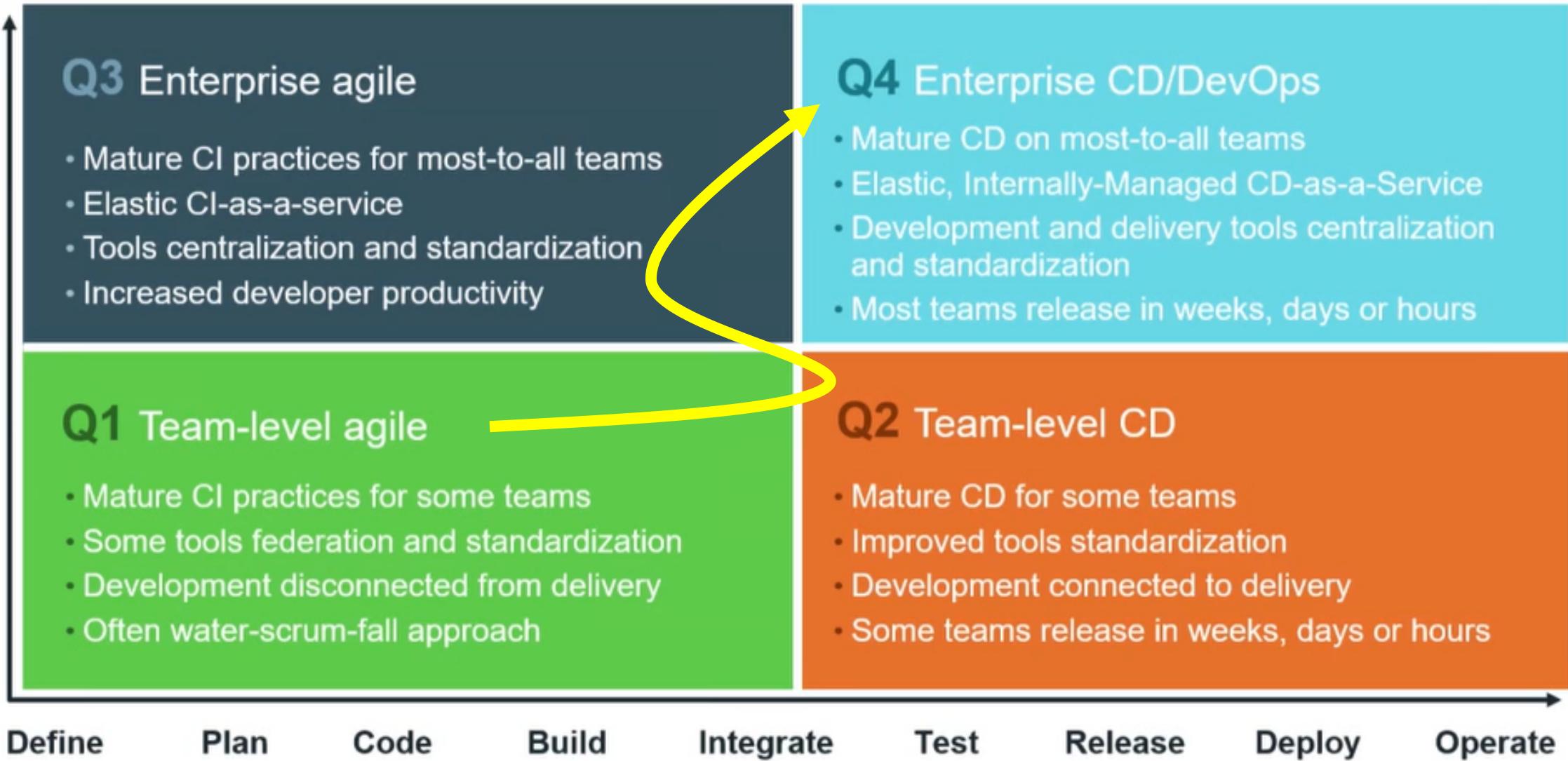
1. People and Culture
2. Process and Practices
3. Tools and Technology

Levels are: Quadrants 1-4 – Team through Enterprise upstream and/or downstream

Take Away Important Idea: Full DevOps maturity means the whole enterprise has adopted agile and continuous delivery / DevOps practices (upstream and downstream). Discusses enterprise journey to maturity.

<https://www.qasymphony.com/landing-pages/thank-you-understanding-devops-journey-brian-dawson/>

Enterprise
Workgroup
Team



Source: QA Symphony / Cloud Bees

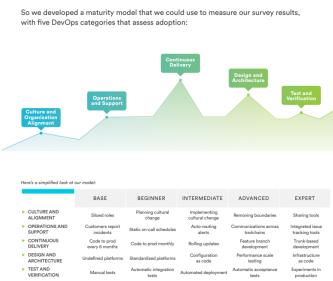
Using Industry Best Practice to Meet Stakeholder DOMM Needs

1

Ensure Team Value Delivery Practices Are Measurable



Takeaway
Use xMatters/Atlassian Focus on Cultural Change to Help Predict Team Success



2

Provide for Team and ART Level Customizable Metrics



Use Accelerate's Distinction Between SDO and Productivity Goals to Customize and Plutora's Continuous Improvement Focus



3

Provide Enterprise-Wide Understanding of DevSecOps Value Delivery Capabilities



Use QA Symphony 4Qs concept to measure growth from team level practices to enterprise



4

Focus on the Customer Impact of the Team's Value Delivery



Use Accelerate's focus on impacting org. performance and Atlassian Regular Check on Business Value



Back-Up Slides

Review of Current VA DevOps Maturity Model

Current DOMM

Criteria Category / Sub Categories	Low Criteria	High Criteria	Relation to DevSecOps Concept
Culture	Level 1 – Initial	Level 5 – Optimized, Standard, Automat.	Flow/Feedback/Cont. Learning; CALMS
Strategy/Requirements	Lacking Clear Req.	Clear & Communicated	Lean- Make only what Customer requires
Communication	Restricted	Effective/ Feedback/Share	S in CALMS for Sharing
Lessons Learned / Countermeasures	Reactive	Learn from Success & Failure – Bottleneck	Continuous Learning, Theory of Constraints
Collaboration / Ownership	Rare Innovation	Innovation Culture/empowered individ.	Automation, Security Part of Team, Early Involve
Automation of Security	No security plan	Automated Security Processes	Automation for speed AND quality
Process			
Automated Processes	Few	Many	Automation
Minimization of Risk Through Auto.	Manual	Automated	Automation
Optimized Productivity / Tracking	Lack of Processes	Defined, Refined Processes	Measure/ Metrics / Flow / Automation
Integrated Processes	Minimal Integration	Integrated Processes	DevOps focuses more on integrated modularity
Proactive Security	Incomplete Sec.	Complete Maturity	Security Part of Team, Early Involve
Process Metrics Established	Lack of Standards	Defined Standards	Measure / Metrics / Telemetry
Technology			
Self-Service Platforms	Lack Self-Service	Many Self-Service Capabilities	Automation, Lean
Self Remediation / Learning	Minimal	Improvement Culture – Teams and up	Continuous Learning
IT Resources Provisioned Through IaaC/IaaS	Few	Many	Automation, Lean
Integrated Analytics	Few	Many	Measure / Metrics / Telemetry
Integrated Platforms	Few	Many	DevOps focuses more on integrated modularity
Security	Minimal	Enhanced	Security Part of Team, Early Involve

Comments on Current DOMM

- The current DOMM is endeavoring to evaluate Maturity, in general, around the core DevSecOps framework/objectives: *Process Flow, Feedback, Continuous Learning, and CALMS (Culture, Automation, Lean, Measure, Sharing)*
- If the current DOMM only consists of this single sheet, it *lacks specificity in how each criteria will be measured and what is an acceptable measurement for each maturity level e.g. what percentage of a team's security processes need to be automated to get a "5"*
- The DOMM has some criteria re: team maturity, but *seems to be focused largely at the enterprise/large org. level* – “...well articulated business strategy with clear business requirements.” But these criteria can be adapted for teams
- *The DOMM does not specifically measure at the next level of detail* – e.g. “Integrated Platforms” – How is the integration measured – is the organization more mature if it has more integrated platforms or if a few platforms are more integrated? How do we measure broad collaboration versus “Dependency on ‘experts’”?
- *It is not clear how the criteria are connected to Customer Impact* – e.g. Level 4 Culture – “Security Incidents are monitored and evaluated” – but thousands of Customers could still be impacted by the incident
- *The DOMM values standardization (All of Level 3 focus, but used throughout) which is not necessarily a DevSecOps principle* – does a semi-autonomous team, working in a largely de-coupled environment on a modular product have to be in lock-step with standardized processes? Do standards, rigidly enforced, work against innovation and learning?
- The DOMM itself can be very manual to administer – 3 core criteria categories: Culture/Process/Technology with ~6 sub-criteria per category – the evaluation could easily take 3 people a complete day to evaluate a single team on all 18 criteria – Assuming a team of 7 people, *the process consumes 10 total people x 8 hours = 80 hours and yields only a few static metrics (e.g. Culture 3, Processes 2, Technology 2) for one team at one timestamp. Multiplied by 100 teams and the investment is significant: 8000 hours for 3 metrics at one point in time per team.*
- Does the value of this measurement outweigh the value of what 200 team members could produce (e.g. automated tests, tech debt reduction, etc.) if left alone for one week (i.e. $200 \times 40 = 8000$ hours)?

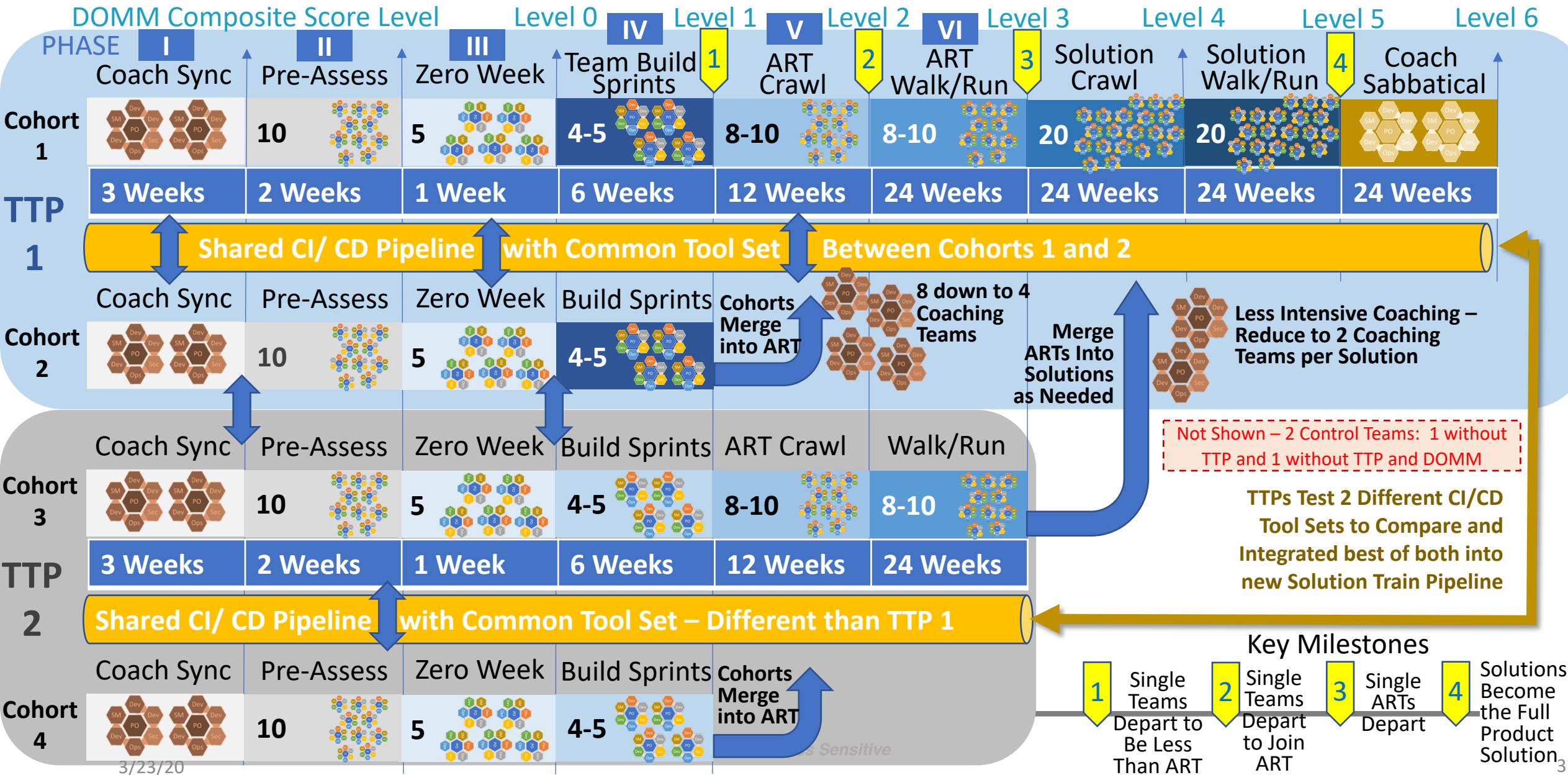
Back Up Slides

Expanded TTP and Phased Implementation of DOMM

Overview of the Team Transformation Pipeline (TTP) Value Streams – “A/B

Testing

Teams can go through as Single Teams or Product Organized Value Stream Teams – Single teams exit pipeline at Level 1 or Level 2 if rejoining an ART



DOMM Type by Phase	DOMM Score		Level -1	Level 0	Level 1	Level 2	Level 3	Level 4	Level 5	Level X
	I 3 weeks	II 2 weeks	III 1 week	IV 6 weeks	V 12 weeks	VI 24 weeks	VII 24 weeks	VIII 12 weeks	VII 12 weeks	IX 24 weeks
Team Transformation Pipeline		Rehearsal of Concept (ROC) for TTP and Refine KPIs/Measures	Test TTP DOMM	Level 0 TTP DOMM	Level 1 TTP DOMM	Level 2 TTP DOMM	Level 3 TTP DOMM	Level 4 TTP DOMM	Level 5 TTP DOMM	Select elite performing team(s) to do coaching sabbatical
Team Transformation		Teams Forming – No Team DOMMs	Starting Line DOMM	Zero Week DOMM	Level 1 DOMM. (add DevSecOps Tool skills)	Level 2 DOMM with ART Criteria Included	Level 3 DOMM with Value Delivery Correlation	Level 4 DOMM with added Solution and Security criteria emphasis	Level 5 DOMM With final Solution Train DOMM	Level X DOMM
Coaching DOMM		-Take Team Coaching DOMM -Take and Help Refine “Starting Line” -Coaching Delivery Baseline	Facilitate Virtual Team Starting Line DOMM and take Team Coaching Focus Area DOMM	VSM Coaching DOMM	Team Coaching DOMM	ART Merge and Crawl Coaching DOMM	ART Walk and Run Coaching DOMM	Solution Merge and Crawl Coaching DOMM	Solution Walk and Run Coaching DOMM	Support Transition of Elite Performing Teams to become Coaching Teams
DOMM Refinement		Test the Starting Line DOMM, begin DOMM data automation and telemetry, Automation DOMM	Evaluate Starting Line DOMM, facilitation, and Accuracy, Validity, Reliability (AVR) DOMM	Business Value Audit DOMM	User Experience DOMM along with initial value delivery correlation	Automation DOMM to see how far progressed in automating DOMM data flows from tools, etc.	Multi-DOMM data review and analysis – State of DevOps report Lead Value Deliver Correlation	Build/Demo Executive Dashboard – Single Pane of Glass	Multiple Categories for DOMM/ Customer Satisfaction Correlation	Train team members from each ART team to Customize DOMM

A DOMM team works with Coaches and uses the first 12 weeks to iteratively test and refine one level at a time staying ahead of the team transformation

Various Custom Composite DOMM Criteria Scores are Developed

