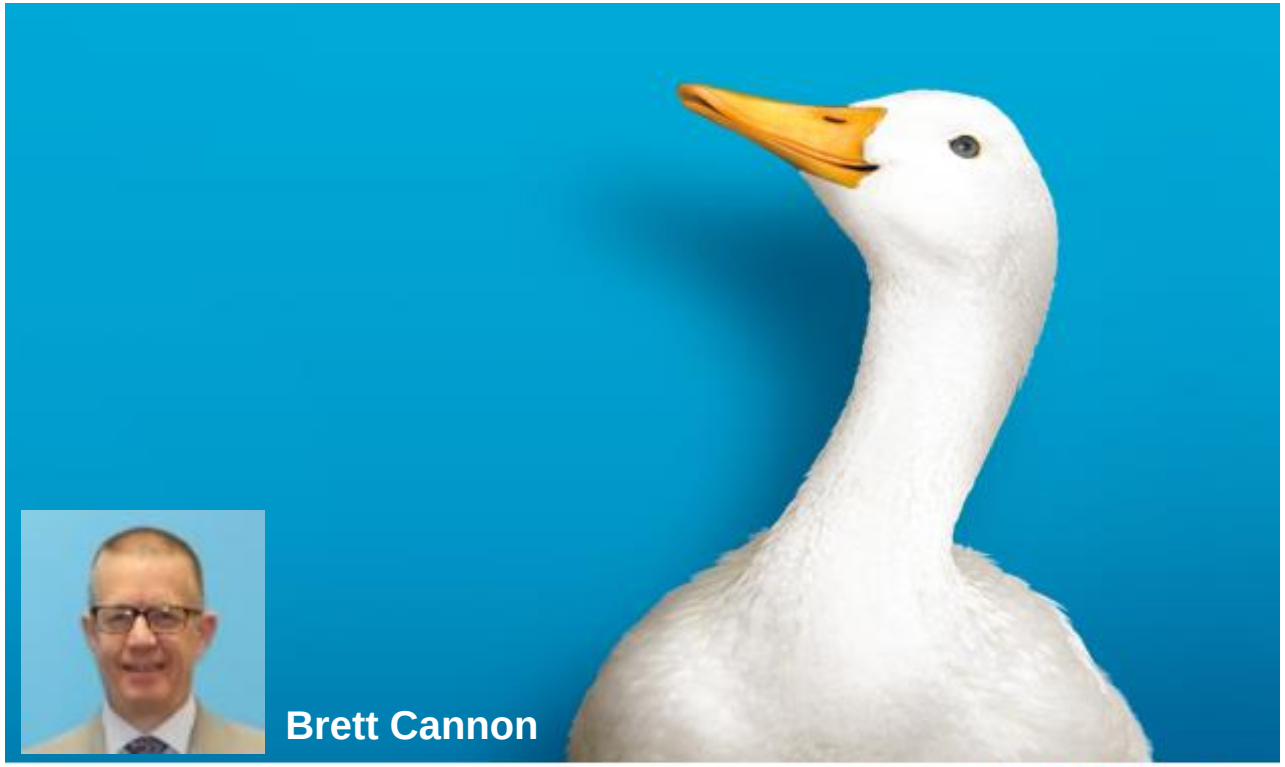
DevOps Journey of Journeys at Aflac

Achieving Enterprise-scale DevOps without the one-size-fits-all mandate antipattern



Agenda

About us, our organizations, and our roles
The challenge
Key enablers
Framework for DevOps/Agile Enablement
Progress and Outcomes (so far)
Continuing Challenges

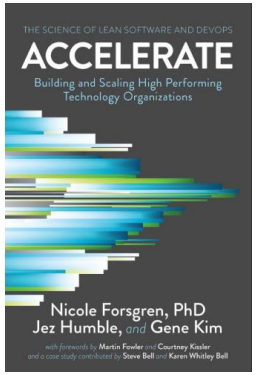
The Challenges

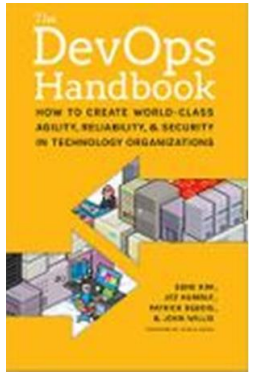
We have had great DevOps progress and results for several teams, however …

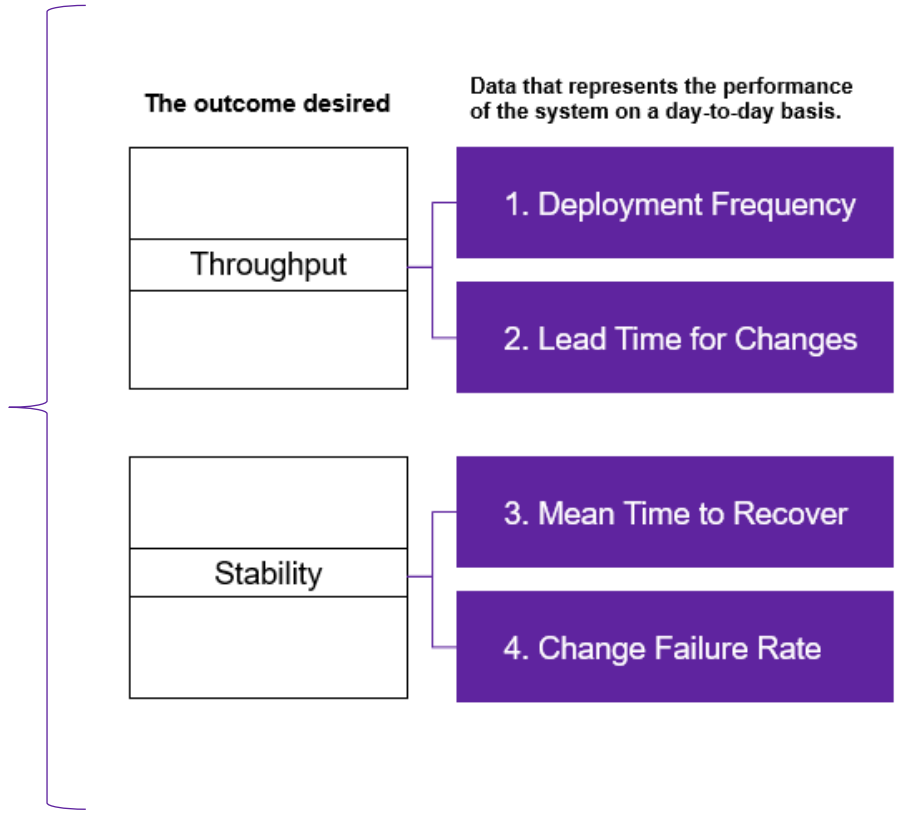
- There were inconsistent results across the enterprise
  (best where there is strong leadership engagement)
- It was difficult for teams to make time for improvements
- There was more work than people available
- Teams struggled with organization-wide impediments
- Many teams did not know how to ‘get started’

We know that …

Highly evolved, high-performance teams focus on outcomes (business value) and continuously improve these four metrics for software delivery

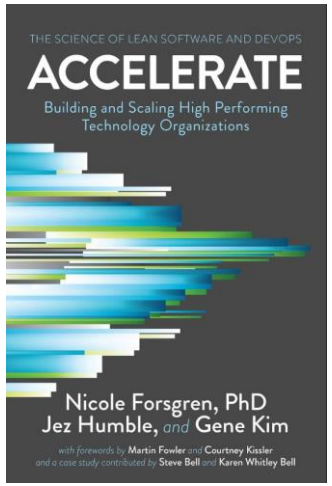






And we know that …

There is a predictive relationship of improving the four “DORA” metrics to higher organization performance



2x

more likely to exceed profitability, market share & productivity goals

2x

more likely to achieve organizational and mission goals, customer satisfaction, quantity & quality goals

2.2x

higher employee Net Promoter Score

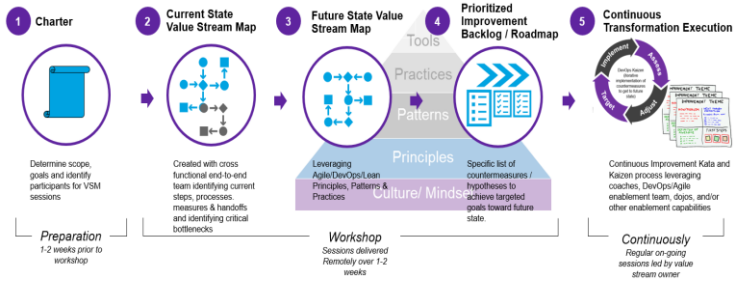
50%

higher market capitalization growth over 3 years*

These high-performance organizations continuously increase the autonomy and fast flow of cross-functional teams.

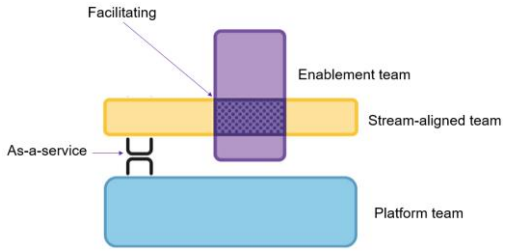
This requires …

1. Culture and discipline of top-down and bottom-up continuous improvement
2. Continually address top impediments to fast flow of value
3. Autonomous teams to make changes with minimal handoffs, approvals, blocking dependencies, and cognitive load
4. Loosely-coupled teams AND services independently developing and releasing



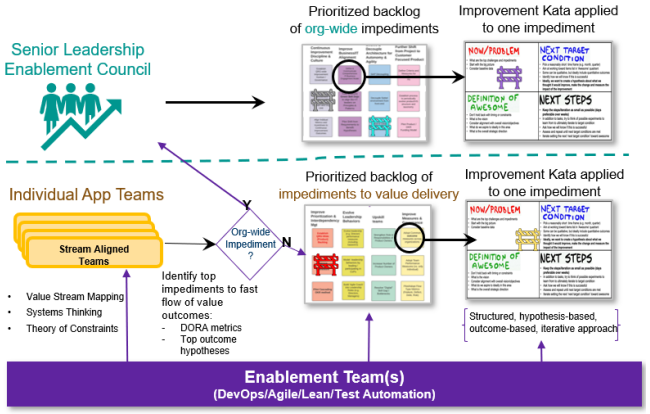
Value Stream Mapping

End-to-end current state and future state mapping for each team to align and prioritize top impediments to fast flow



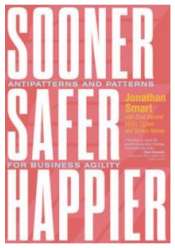
Team Topologies

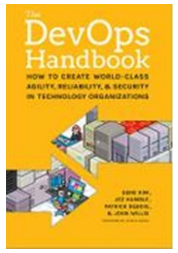
Team Topology constructs, including the formation of distinct platform and enablement teams are a key ingredient to the secret sauce of enabling these three unlocks

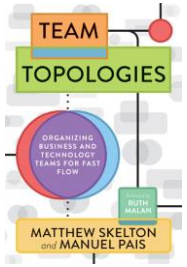


Continuous Improvement Framework

A two-tier framework (top leadership AND team level) to institutionalize continuous improvement discipline and culture for measurable outcomes







DXC TECHNOLOGY Aflac

Continuous Improvement Discipline and Culture with focus on outcomes is the “Secret Sauce”

“Improving daily work is more important than doing daily work”

- Dr. Steven Sears

"The important thing is not your process. The important thing is your process for improving your process."

- Henrik Kniberg

“The critical success factor of digital transformation is the ability to continuously manage and improve the continuous improvement process”

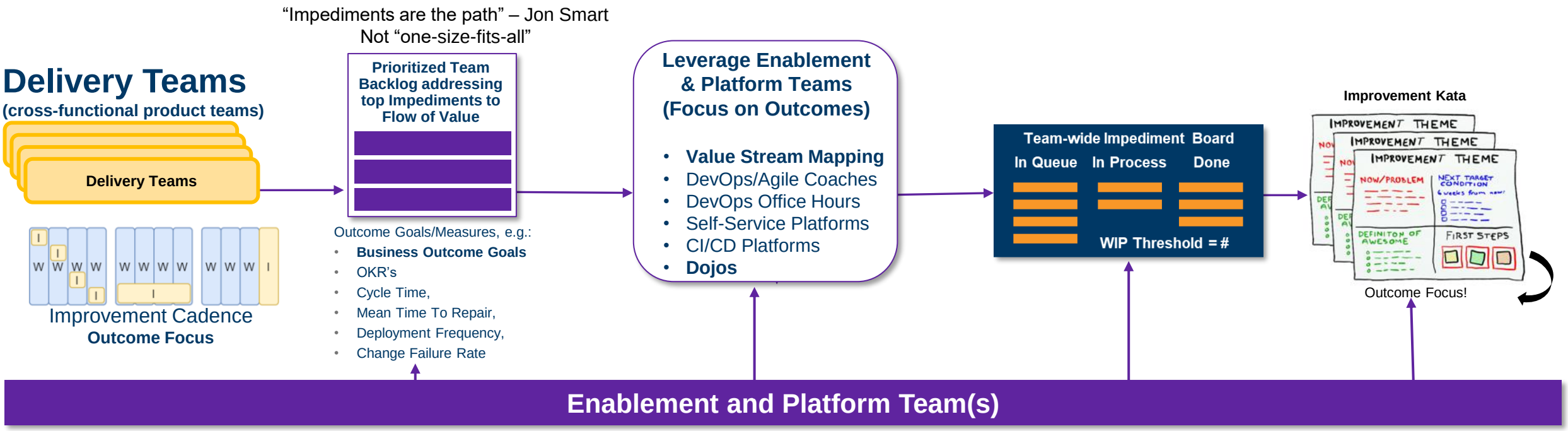
– Micro Hering

“Impediments are not IN the path Impediments ARE the path”

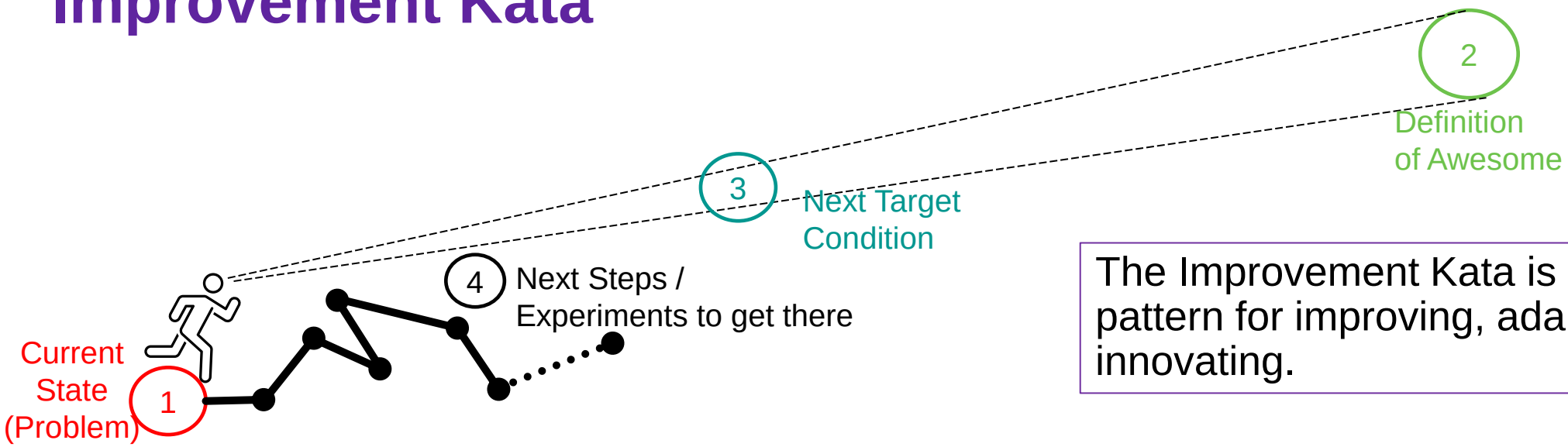
- Jonathan Smart

Journey of Journeys

Rather than a one-size-fits-all, prescriptive, mandated single journey, we **enabled** each team to own and drive results based on their goals and impediments



Improvement Kata



The Improvement Kata is an effective pattern for improving, adapting and innovating.

1 Now / Problem

Identify the high-level challenge and describe **the current state**.
Can include the top priority problems, and baseline data.
This gets updated each time we achieve the condition set in #3 – or if we run out of time.

3 Next Target Condition

Set our expectation for where we want to be in designated timeframe. "Within Q3 we want to be able to XYZ." Ideally, the **hypotheses** is measureable so that we know we achieved target condition. Once there, we evaluate where we are and, from there, set the next Short-term Target.

2 Definition of Awesome

The ideal future state. What awesome would look like.
The vision which determines the direction in which we want to travel.
This rarely changes.

4 Next Steps

For continuous improvement, we identify the **experimental steps**, techniques and countermeasures we can take immediately to move us toward the short-term target. After each experiment, we **evaluate and adjust**.

Dojo summary

Epic	We are moving away from	To go to	To improve Cycle Time and Quality
CI and CD pipeline	• Time consuming/Minimal review process • Silo'd teams • Delay approval / Manual process/ Long process of deployment	• Work as a single team. • Systematic peer review • Working in Isolation – limit the risk / increase the efficiency. • Fast feedback from pipeline in context • Fast flow/delivery	Cycle time: As we continue enhancing our pipeline, we can shift to production **elevation In Minutes**
Test Automation	• Manual testing • Testing as a phase • Lengthy feedback loops	• Automated and continuous testing • Short and quick feedback loop for developers	Cycle time: cut testing phase **from 2 weeks to few hours** with continuous testing
0 down-time deployment	• Push Deployment • Maintaining Multiple VMs • Rebuilding the entire VM for a single application	• Pull Deployment • Single VM for multiple versions of an application • Updating a single application only	Cycle time: Reduce Delivery times **from 40 hours to 30 seconds**

Results of Initial Lighthouse DevOps Efforts

1 Hypothesize

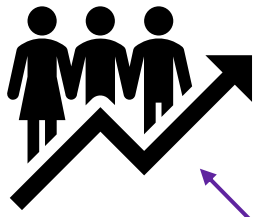
2 Experiment

3 Learn

- **We hypothesized that:**
  - We can pinpoint the biggest impediments to flow with Value Stream Mapping for lighthouse value streams and Change Management
  - Make progress to improve **cycle time** and **quality** with DevOps Dojos and DevOps Kaizen
- **We verified that:**
  - We reduced cycle time by 50% and MTTR by 60% for a claims system
  - We reduced deployment time by 75% for an agent system, 55% for a group insurance management system – 85% reduction in manual tasks
  - We significantly increased quality thru automated pipeline testing in all three value streams
  - We have implemented light-weight change management in lighthouse value streams, showing the way for others
- **We learned that:**
  - There are multiple journey categories, each with both common and unique patterns – Distributed, Cloud, Mainframe, SAAS, etc
  - Need platforms built and treated as products
  - Efforts need to be sustained and led from the top
  - Need stronger business engagement

Continuous Improvement Framework

Enablement Council



Senior Leaders

1. Prioritize org-wide impediments
2. Coach, support, drive the continuous improvement discipline & culture
3. Actively live the process we want all teams to do

Initial Org-wide impediment / countermeasure backlog

Example Org-wide Impediments

- Inflexible annual budgeting process
- Agile teams impeded by release freezes
- Contractor hiring takes too long
- Waterfall-like project, prioritization, approval & portfolio processes
- Business/IT alignment (IT as order takers)
- More doing agile than being agile
- Lack of psychological safety
- Increasing tech debt

Organizes\ Trains

Manages Backlog

Core Team

Manages Kanban

Monthly Leadership Enablement Council Sessions

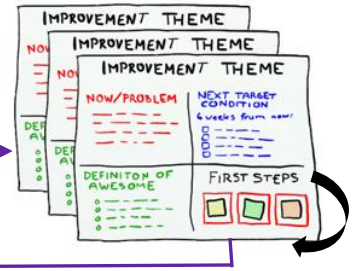
- Council launches improvement kata work to address next top impediment(s)
- Focus is next target conditions, outcomes, learnings
- Kata learnings from teams and from leaders are shared

Org-wide Impediment Board

In Queue | In Process | Done

WIP Threshold = #

Improvement Kata



*"Impediments are the path" – Jon Smart
Not "one-size-fits-all"

Delivery Teams

(cross-functional product teams)

Delivery Teams

Improvement Cadence

Outcome Focus

Prioritized Team Backlog of Top Impediments to Flow of Value

Outcome Goals/Measures, e.g.:

- Business Outcome Goals
- OKR's
- Cycle Time,
- Mean Time To Repair,
- Deployment Frequency,
- Change Failure Rate

Leverage Enablement & Platform Teams (Focus on Outcomes)

- Value Stream Mapping
- DevOps/Aglie Coaches
- Self-Service Platforms
- CI/CD Platforms
- Dojos

Org-wide Impediment ?

Y

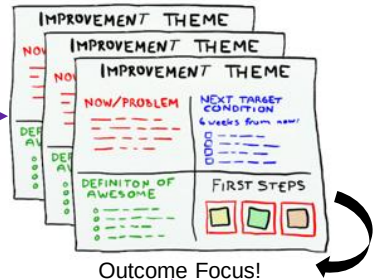
N

Team-wide Impediment Board

In Queue | In Process | Done

WIP Threshold = #

Improvement Kata



Outcome Focus!

Enablement and Platform Team(s)

Leadership Learning Sessions

Leadership Role in Continuous Improvement
Improvement Kata
Team Topologies – Organizational architecture
Psychological Safety
Optimizing and Architecting for Fast Flow of Value
Feedback Loops
Maximizing Outcome Focus
Work in Progress and Visibility of Work
Developing a Learning Organization

Progress and Outcomes (so far)

- Leaders engaged and driving
  - Executive leaders are actively being trained
  - Workshop needs and priorities are becoming leader directed
- Improvement Kata is being embraced and embedded in culture – leaders and delivery teams
- Many delivery teams are hitting their kata “next target conditions”
- Top org-wide impediments are starting to be addressed

Next steps and challenges

- Address next top org-wide impediments (continuously)
- Formally account for continuous improvement within planning process
- Train the rest of the leadership
- Solidify plan for scale DevOps practices and principles across the organization
- Continue to build out platform and enablement teams