HOW TO BE A GREAT DEVOPS LEADER

Mustafa Kapadia
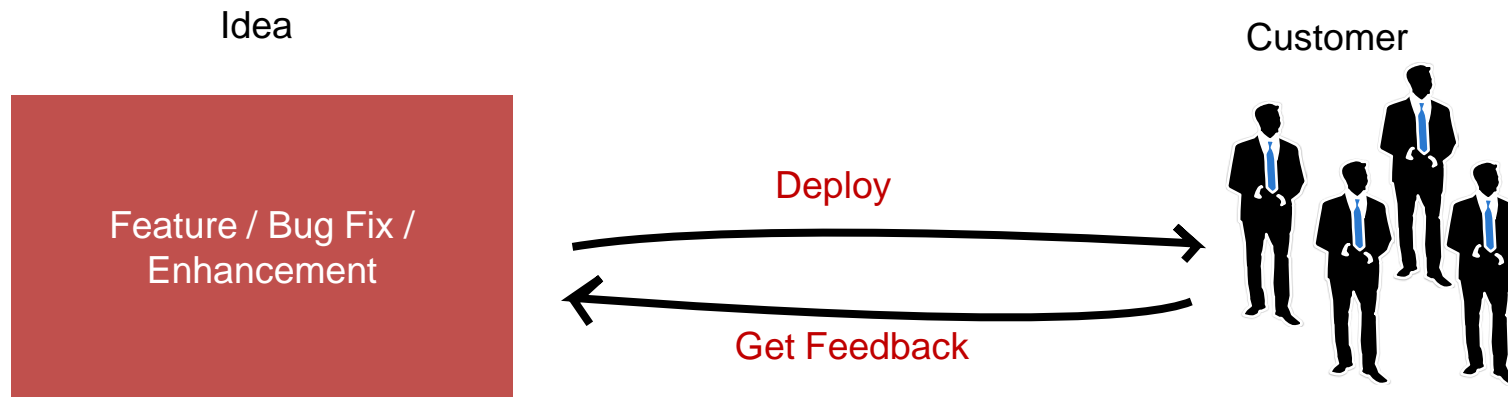IBM, Global Business Consulting

Agenda

- What is DevOps?
- Leadership Skills
- Building A Vision
- Typical Transformation Journey
- Rules Of The Road

What Is DevOps

Time To Market

Idea

Feature / Bug Fix / Enhancement

Customer

Deploy

Get Feedback

Optimize the flow of work across your SDLC so that you can

1. Bring ideas to market faster and

2. Be more responsive to your customers

Key difference between Traditional & DevOps software delivery models

Key Characteristics		Traditional IT	DevOps
	Key Driver	Cost	Flow (Time)
	Organization	Skill Centric Silos	Autonomous Teams
	Batch Size	Large	Micro (MVP)
	Scheduling	Complex, Time Consuming	Decentralized
	Release	High Risk Event	Non Event
	Culture	*"I did my job"*	*"Its ready to deploy"*
Impact	Timeliness	40% Large Projects Are Late	Deliver At Business Cadence
	Features	Upto 40% Never Used*	Deliver Only What Is Needed & Iterate
	Success	Cost	Business KPIs

*Standish Group, Feature Usability Study

Optimizing the flow of work requires changes to people, process, and technology

Key Building Blocks

People & Process (New Ways Of Working)

	Speed As A New Metric	*Velocity impacts both cost & quality*
	Small Batches	*Create MVP, continuously test, and constantly iterate*
	Autonomous Teams	*Cross functional, co-located, dedicated, product focused teams + support*

Technology

	Automate	*Push button code deploy across delivery pipeline – build, test, deploy, rollback, monitoring etc.*
	Flexible Infrastructure	*Provision a full stack environment on demand that looks just like production*

DEVOPS LEADER

Who is a DevOps Leader?

A DevOps Leader

- Is a tactical or strategic individual who helps design, influence, implement or motivate the cultural transformation proven to be a critical success factor in DevOps adoption

- Understands the human dynamics of cultural change and is equipped with practices, methods and tools to engage people across the DevOps spectrum

- Embraces systems thinking, agility, lean principles and the flow of the value stream from idea to realization

Everyone is a leader!

DevOps Leader Skill Set

Technical Aptitude

- Environment Management & Configuration
- Release & Deploy
- Testing
- API
- MicroServices
- Agile

Emotional Intelligence

- Builds a vision
- Challenges Teams Status Quo
- Inquisitive
- Inspires pride in being part of the team
- Cares about individuals
- Acknowledges improvements
- Celebrates success

Transformational vs. Servant Leaders

DevOps Leaders Are Transformational

Servant leaders and transformational leaders differ in their focus

- Servant leaders focus on their followers' development and performance
- Transformational leaders focus on getting followers to identify with the organization and engage in support of organizational objectives

A transformational leader's influence is seen in their support of their teams' work, both in technical practices and in the teams' product management capabilities.

The positive (or negative) influence of leadership flows all the way through to IT performance and organizational performance.

Source: 2017 State of DevOps Report

https://puppet.com/resources/whitepaper/state-of-devops-report

10

Dimensions of transformational leadership

Vision
- Understands organizational direction.
- Understands team direction.
- Understands 5-year horizon for team.

Intellectual stimulation
- Challenges team status quo.
- Challenges team to constantly ask new questions.
- Challenges team on basic assumptions about the work.

Personal recognition
- Commends team for better-than-average work.
- Acknowledges improvement in quality of work.
- Personally compliments individuals' outstanding work.

Inspirational communication
- Inspires pride in being part of the team.
- Says positive things about the team.
- Inspires passion and motivation; encourages people to see that change brings opportunities.

Supportive leadership
- Considers others' personal feelings before acting.
- Is thoughtful of others' personal needs.
- Cares about individuals' interests.


Source: 2017 State of DevOps Report (Puppet Labs/DORA

BUILDING A VISION

Every successful DevOps transformation starts with a Value Stream
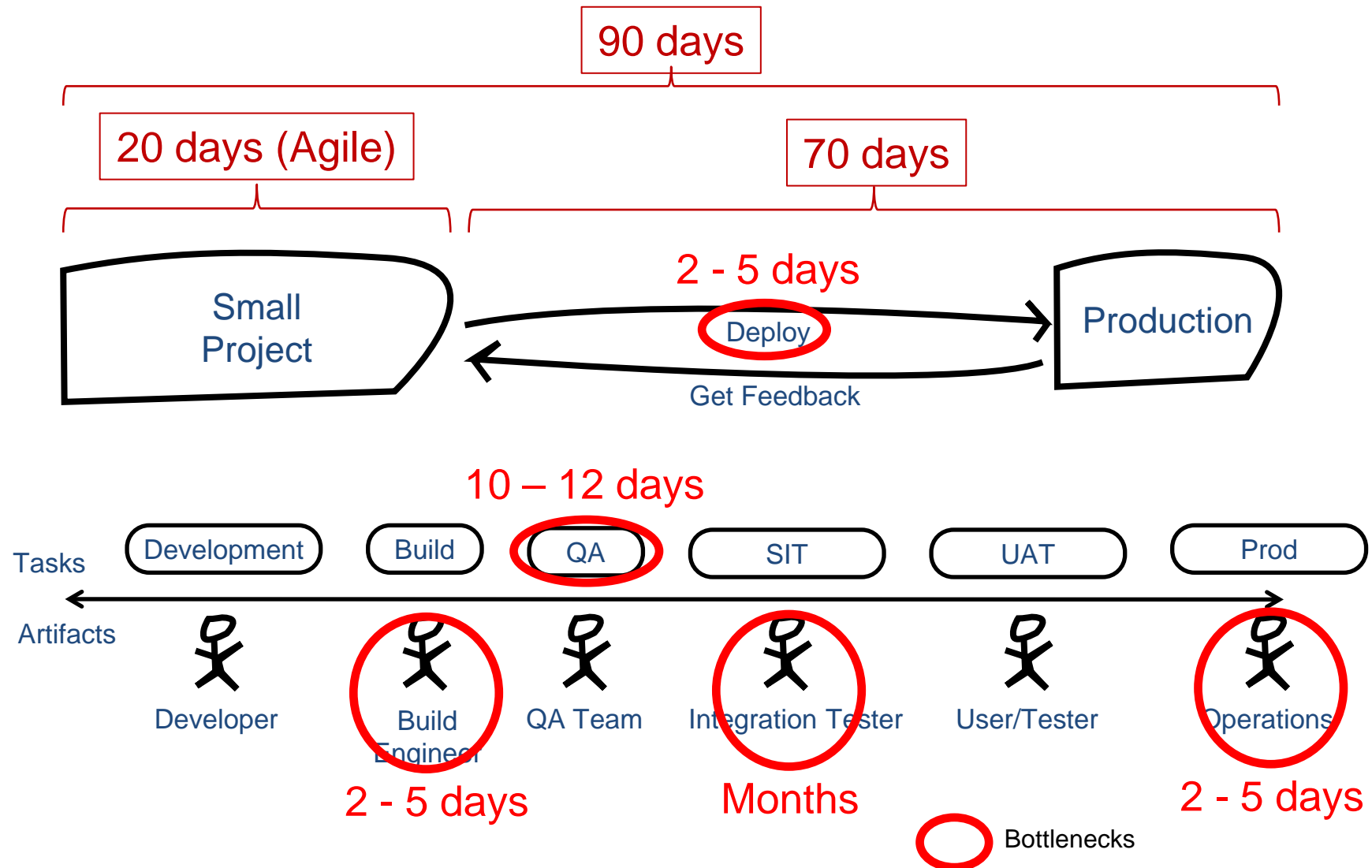
Value Stream Mapping (VSM)

- It is a time diagnostic of your software delivery lifecycle
- A flow based representation of how end to end work is currently done
- VSM is a tool that is used to evaluate processes to identify bottlenecks, waste, and ultimately improvement opportunities

# Why Use a Value Stream Map To Create A Vision
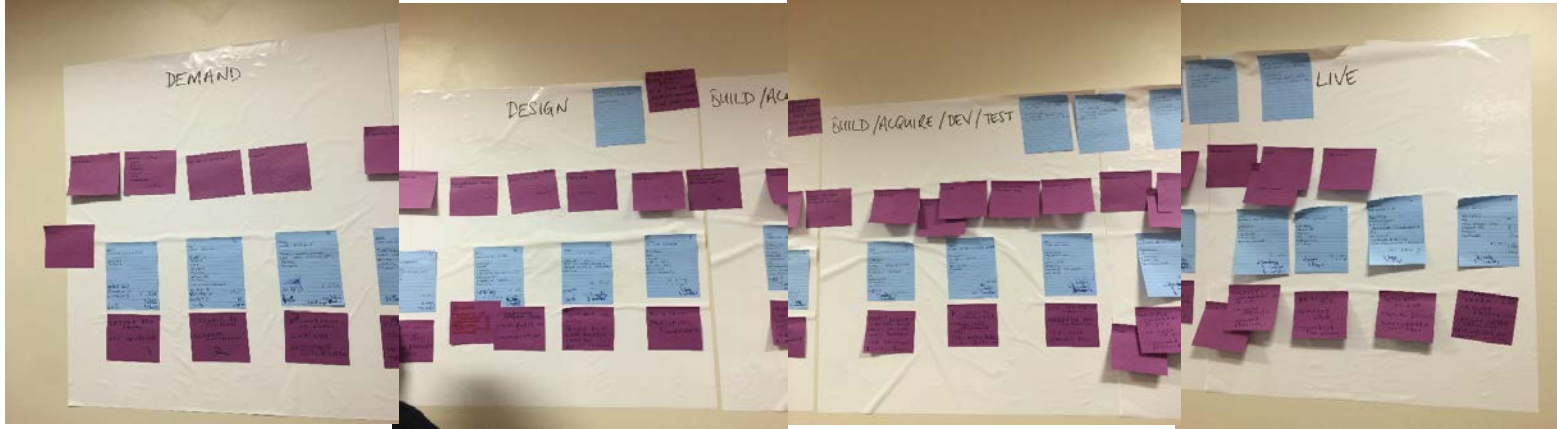
- End to end map of the current "flow of value" is usually not available within the enterprise

- Current metrics track cost and capacity, but not impact of time

- Most effective Lean method to identify waste

- Creates a common language when completed

- Provides a framework for future state design

A Simple Value Stream Mapping Exercise

90 days

20 days (Agile)

70 days

2 - 5 days

Small
Project

Deploy

Production

Get Feedback

10 – 12 days

Tasks

Development | Build | QA | SIT | UAT | Prod

Artifacts

Developer | Build Engineer | QA Team | Integration Tester | User/Tester | Operations
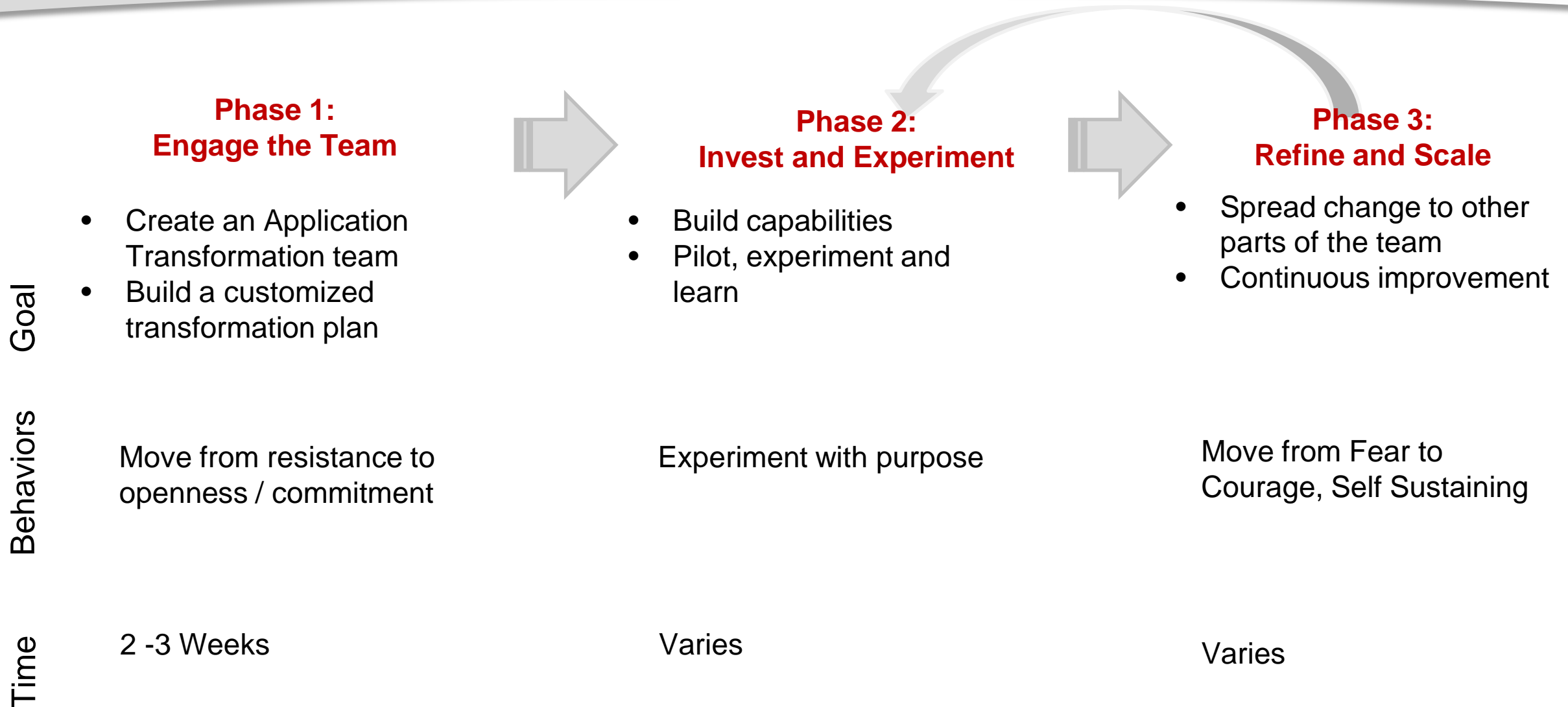
2 - 5 days

Months

2 - 5 days

Bottlenecks

Value Stream Mapping is an Ongoing Exercise

You never get to a stable future state – you have to keep reiterating (continuous improvement)

Next Future State

Future State

Current State

Original State
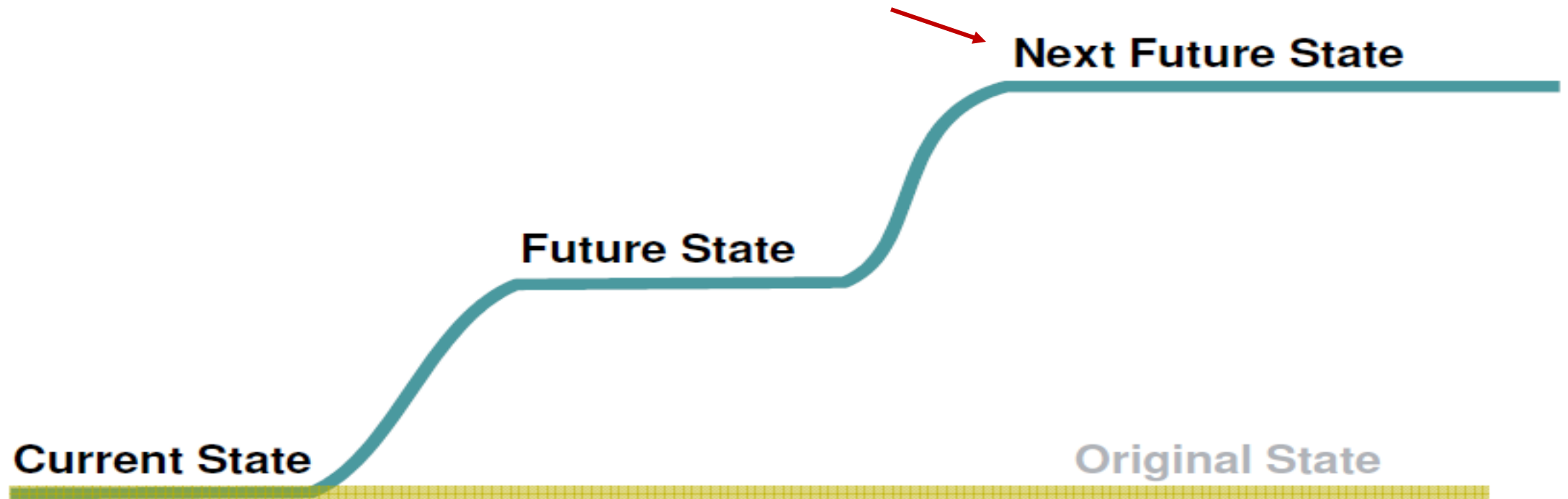
TYPICAL TRANSFORMATION JOURNEY

Application teams go through the following three phases during the transformation journey

Phase 1:
Engage the Team

Phase 2:
Invest and Experiment

Phase 3:
Refine and Scale

Goal

- Create an Application Transformation team
- Build a customized transformation plan

- Build capabilities
- Pilot, experiment and learn

- Spread change to other parts of the team
- Continuous improvement

Behaviors

Move from resistance to openness / commitment

Experiment with purpose

Move from Fear to Courage, Self Sustaining

Time

2 -3 Weeks

Varies

Varies

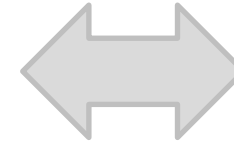# Key activities during the transformation journey

Phase 1:
Engage the Team

- Create a transformation Team
- Develop initial Value Stream Map
- Define what best looks like
- Determine Transformation Plan
- Agree on Metrics
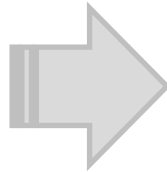
Phase 2:
Invest and Experiment

- Mobilize Technology Capability Teams
- Train team
- Pilot new operating model
- Prepare for Failure
- Track Metrics

Phase 3:
Refine and Scale

- Codify new ways of working-
- Update Value Stream Map
- Identify new areas for improvement
- Repeat Phase 2
- Share / Experiences, Celebrate Success
- Track Metrics

In order to support these Application Teams, the DevOps Leader must invest in both technical and coaching capabilities

	New Technology Capability Teams	Transformation Adoption Team
Goal	Build new technology capabilities	Assist applications teams adopt new capabilities
Key Responsibilities	• Design and Build new technology capabilities • Determine the right tooling • Assists Application teams during transformation • Domain experts that SMEs can call on • Transition new capabilities to a support organization / support it themselves	• Guide the team through the 3 Phases (coaching) • Assist Adoption on the new capabilities • Keeping the change on track • Train Application Teams • Conduct Value Stream Mapping workshops
Technical Domains	Environments Configuration (Cloud) Build / Deploy / Release Automation Test Automation API / MicroServices Add (as needed)	

RULES OF THE ROAD

Rules Of The Road

- Pick the right applications to transform

- Perform a value stream to understand where waste is

- Prioritize bottlenecks, build a roadmap

- Build a business case

- Attack one bottleneck at a time

- Restructure the work for better flow

Rules Of The Road

- Organize teams around value

- Create squads

- Make sure information is created / consumed in real time

- Create a change management plan

- Use time as a key performance metric

- And don't forget about the Tools

Question and Answers

Thank You

Mustafa Kapadia

Devops Consulting Service Line Leader

Mobile: (623) 202 8767

Email: m.a.kapadia@us.ibm.com

Sign Up For My Monthly Newsletter – Digital Top 5