

PeopleCert DevOps Fundamentals

Training material



Copyright Details

The contents of this workshop are protected by copyright and can be reproduced under the Terms of Use agreed between PeopleCert and the ATO using this material only.

Material in this presentation has been sourced from the bibliography listed in the certifications Syllabus.

No part of this document may be reproduced in any form without the written permission of PeopleCert International Ltd. Permission can be requested at www.peoplecert.org.

e-mail: info@peoplecert.org, www.peoplecert.org

Copyright © 2017-2019 PeopleCert International Ltd.

All rights reserved. No part of this publication may be reproduced or transmitted in any form and by any means (electronic, photocopying, recording or otherwise) except as permitted in writing by PeopleCert International Ltd. Enquiries for permission to reproduce, transmit or use for any purpose this material should be directed to the publisher.

DISCLAIMER

This publication is designed to provide helpful information to the reader. Although every care has been taken by PeopleCert International Ltd in the preparation of this publication, no representation or warranty (express or implied) is given by PeopleCert International Ltd. as publisher with respect as to the completeness, accuracy, reliability, suitability or availability of the information contained within it and neither shall PeopleCert International Ltd be responsible or liable for any loss or damage whatsoever (indicatively but not limited to, special, indirect, consequential) arising or resulting of virtue of information, instructions or advice contained within this publication.)

10

- 100

100

Introduce yourself

What is your name?
What organization do you work for?
What is your role?



Start and
end times



Participation
is essential!



Lunch and
breaks



Exam



Feedback

Course Contents



Introduction:

What is DevOps?



Module 1:

The Urgency For DevOps



Module 2:

Key DevOps Principles & Concepts



Module 3:

The Full Stack –
People & Culture



Module 4:

The Full Stack –
Processes & Practices



Module 5:

The Full Stack –
Technology & Automation



Wrap Up & Review

DevOps Maturity
Assessment Revisited

Agenda

Day 1

- ⑩ Introduction: What is DevOps?
- ⑩ Module 1: The Urgency For DevOps
- ⑩ Module 2: Key DevOps Principles & Concepts
- ⑩ Begin Module 3: The Full Stack – People & Culture Layer

Day 2

- ⑩ Continue Module 3
- ⑩ Module 4: The Full Stack – Processes & Practices Layer
- ⑩ Begin Module 5: The Full Stack – Technology & Automation Layer

Day 3

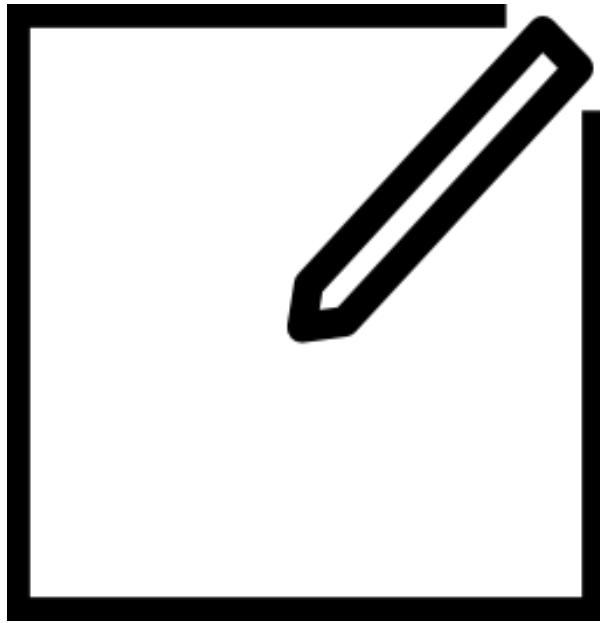
- ⑩ Continue Module 5
- ⑩ Wrap Up & Review



PeopleCert DevOps Fundamentals

INTRODUCTION

What Is DevOps?



A Quick Exercise

Before we define it, what does DevOps mean to you right now?



Defining DevOps

DevOps is a coined noun used to describe an evolution of **existing IT best practices** from ITIL, Lean and Agile into a **development and operations approach that supports automation and continuous delivery**, and **encourages a culture of collaboration and learning** to help IT deliver business value better, faster and cheaper than ever before.



What DevOps Is



A Movement,
Philosophy or
Ideology



Development &
Operations



Culture



Communication &
Collaboration



Automation



Speed & Change



Reliability &
Stability



What DevOps Is Not



A specific job title or team



Only for startups or web applications



A way to eliminate Operations



Only a collection of tools for automation



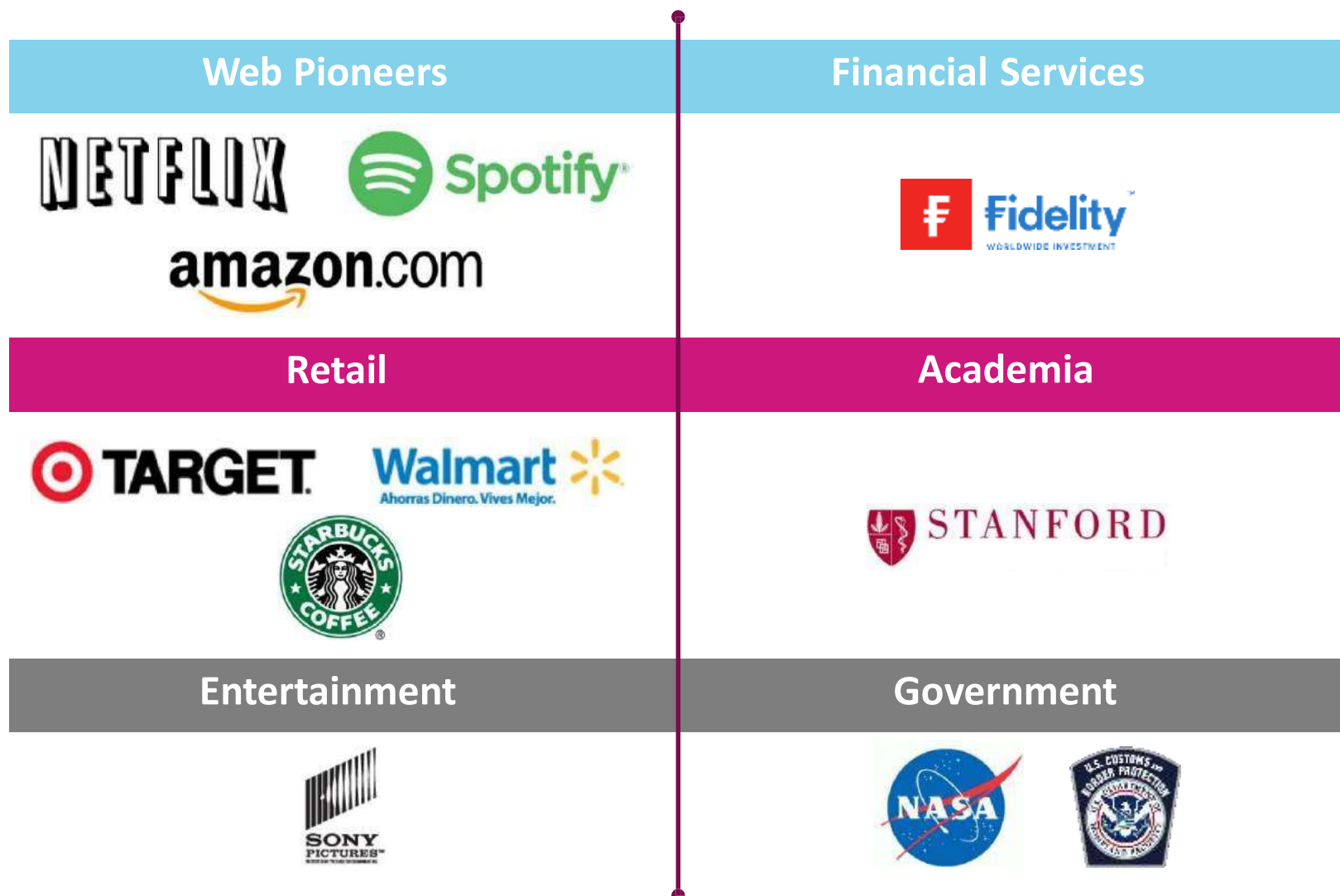
Unworkable in a regular environment

A woman with long blonde hair, wearing a blue shirt and a watch, is sitting at a wooden desk. She is holding a black pen and writing in a spiral-bound notebook. On the desk, there is also a white mug on a saucer, another notebook, and a laptop. The background is slightly blurred, showing a window and some office equipment. A red text box is overlaid on the right side of the image.

**DevOps is built on
collective experience.**



DevOps Across Industries





The DevOps Full Stack



Culture



The DevOps Full Stack



Practices



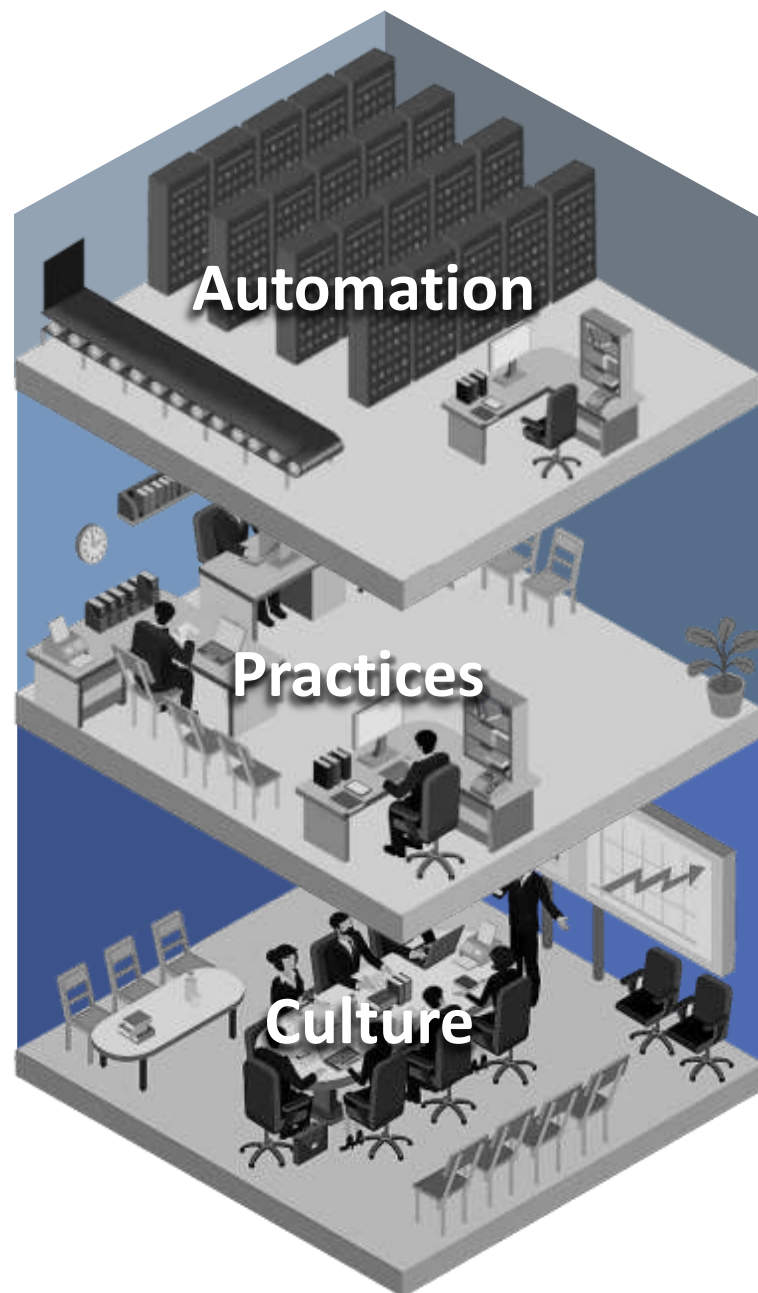
The DevOps Full Stack



Automation



The DevOps Full Stack





PeopleCert DevOps Fundamentals

MODULE 1:

The Urgency For DevOps

- **The Business Value Delivery Problem**
- **The IT Value Delivery Problem**



"I have better and more effective technology at home than I do at work!"

An illustration from a top-down perspective shows four people sitting around a circular teal table. The person at the top left is a woman with a red beanie and orange shirt, looking at a tablet. The person at the top right is a man in a dark suit, holding a coffee cup. The person at the bottom right is a man in a dark suit, looking at a laptop. The person at the bottom left is a man in a white shirt and glasses, looking at a laptop. The table is cluttered with various items: a smartphone, a coffee cup, a tablet, a laptop, a folder, a pen, and some papers. Four speech bubbles point to each person, containing text about common IT complaints.

"IT has a solution looking for a problem."

"Internal IT costs too much and delivers too little!"

"IT does not understand business priorities!"

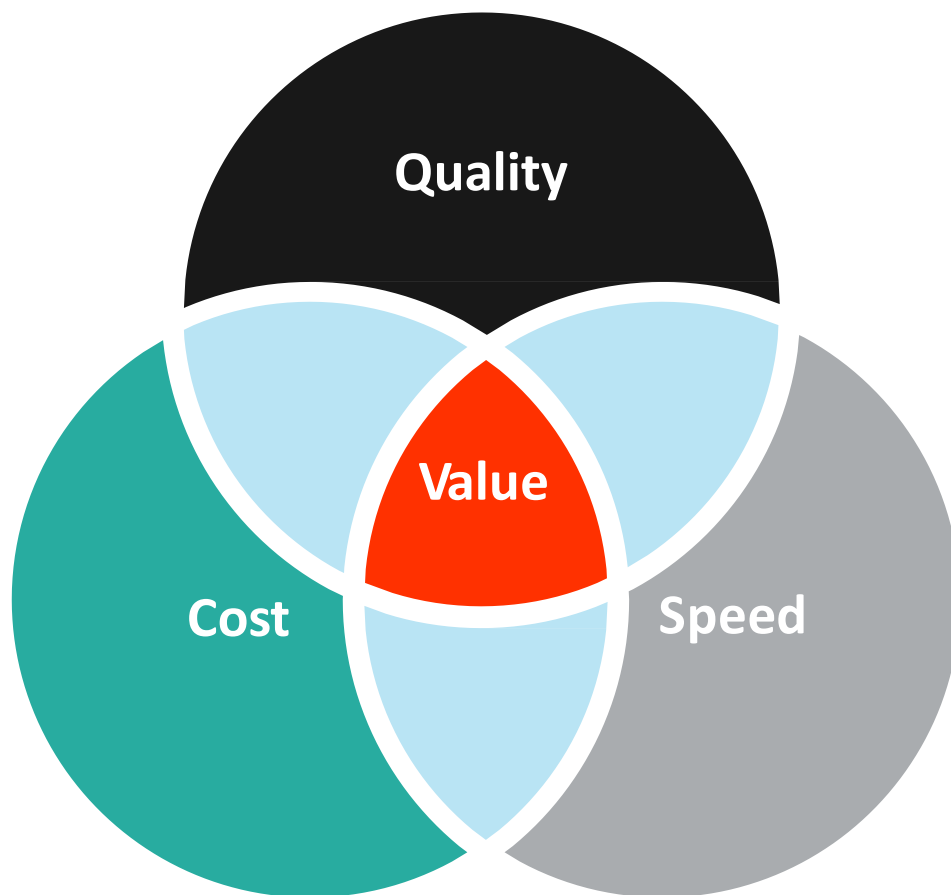


The Business Value Delivery Problem

Business Value is the level to which a service meets a customer's expectations or exceeds them.

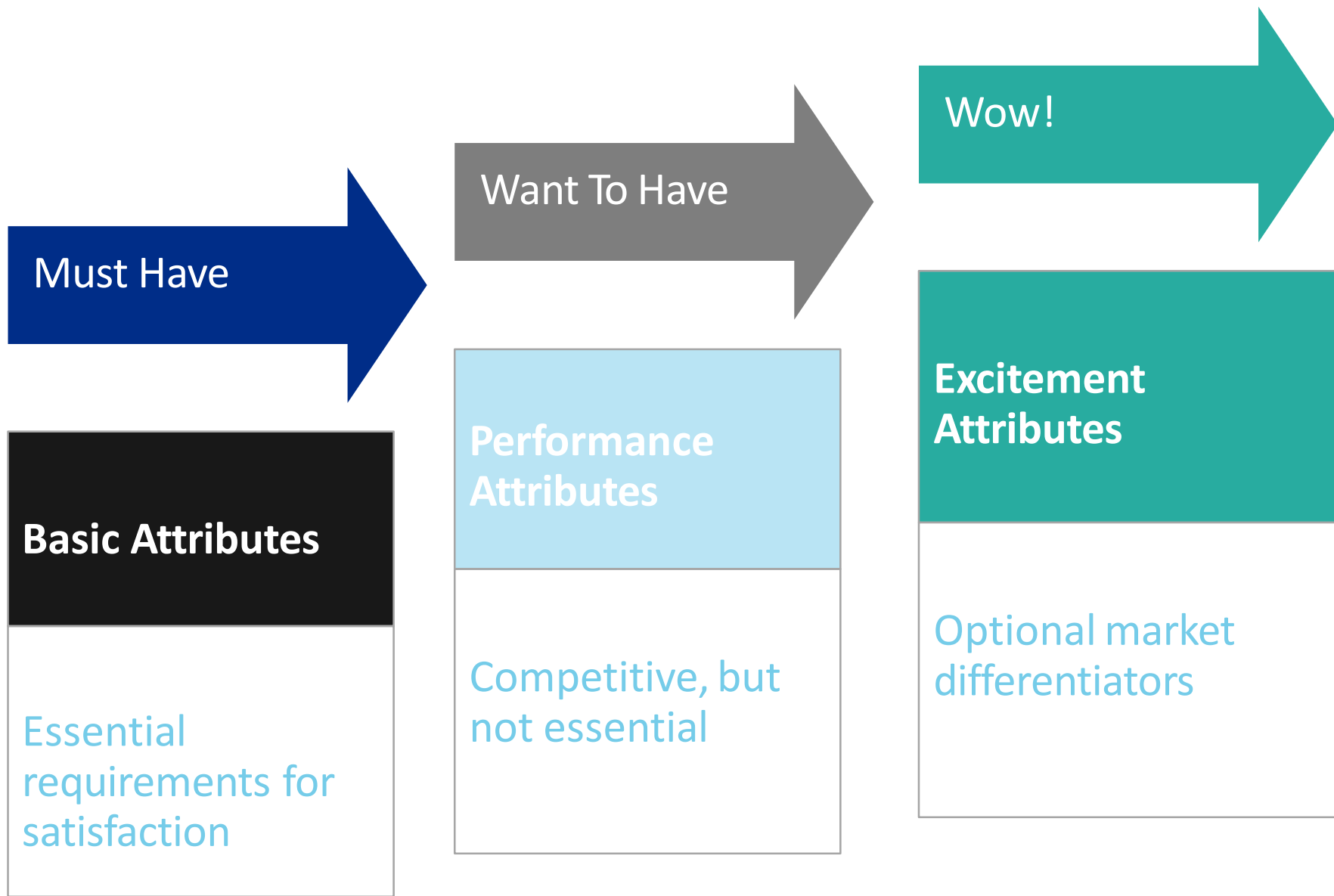


Dimensions Of Business Value





The Customer Value Hierarchy





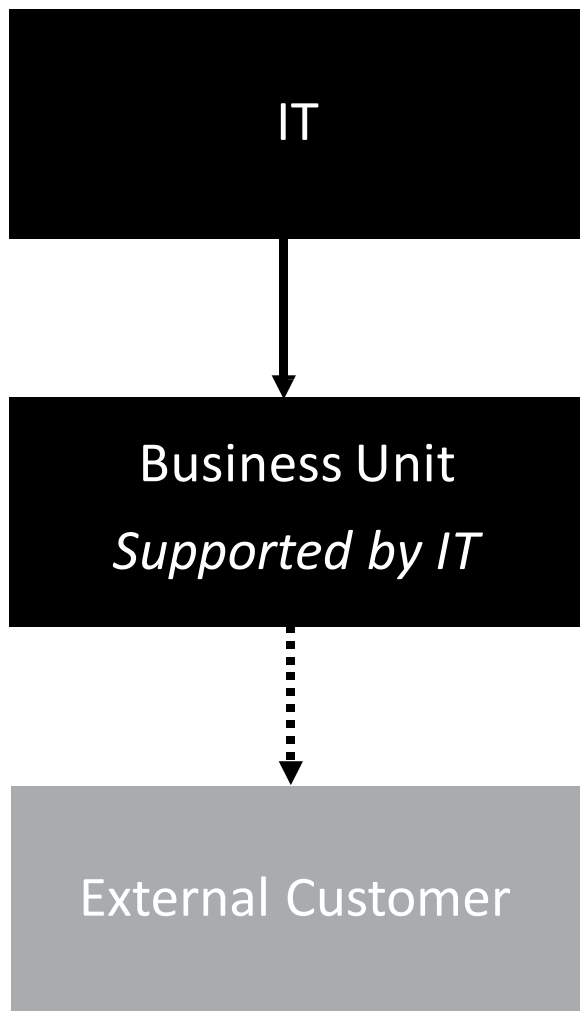
Kano Model Example

	Basic Attribute “Must Have”	Performance Attribute “Want To Have”	Excitement Attribute “Wow!”
70s-80s	▪ Checking accounts	▪ Credit Cards	▪ ATMs ▪ Debit cards ▪ Telephone banking
90s-00s	▪ Checking accounts ▪ Credit Cards ▪ ATMs	▪ Debit cards ▪ Telephone banking	▪ Online banking ▪ Online investing
Today	▪ Credit cards ▪ ATMs ▪ Online banking ▪ Debit cards ▪ Telephone banking	▪ Chip & PIN cards ▪ Online investing	▪ Digital deposits ▪ Email money transfers ▪ Online customer service

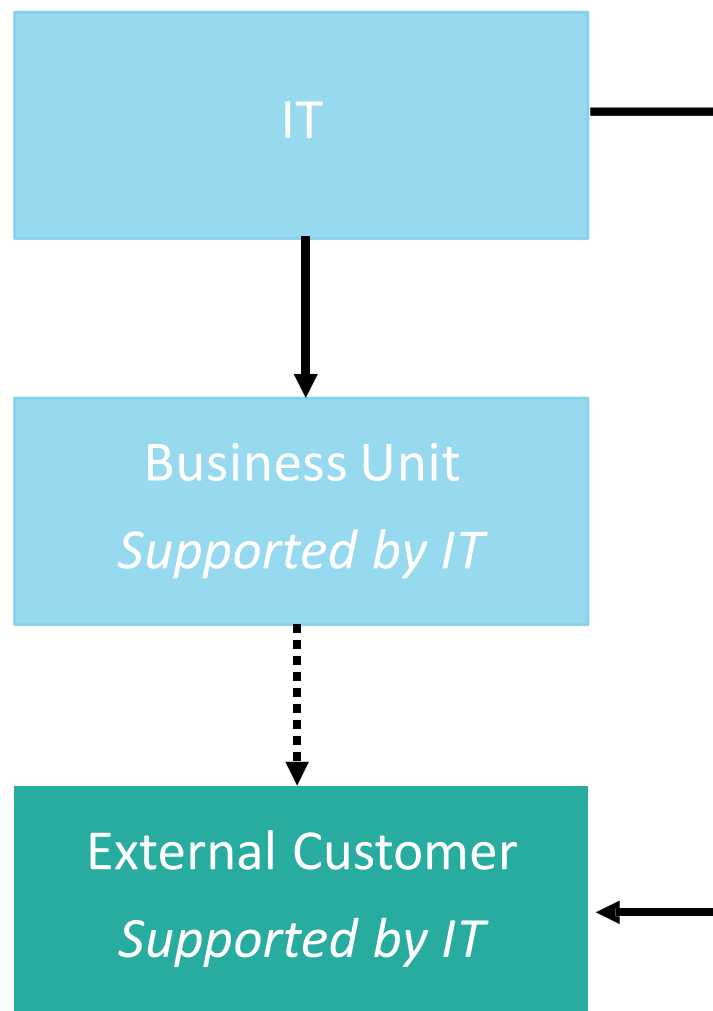


IT Impacts Business Value

THEN

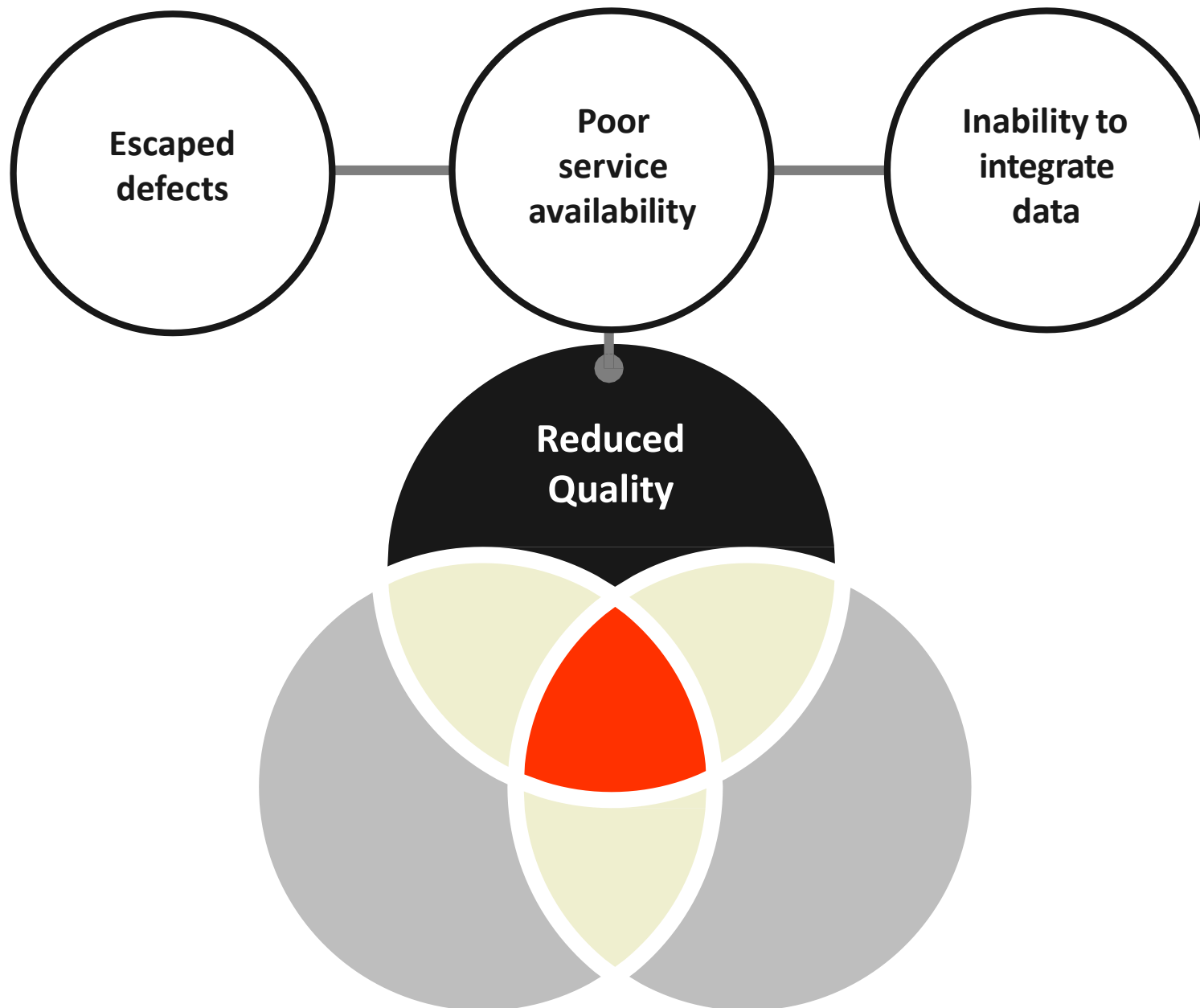


NOW



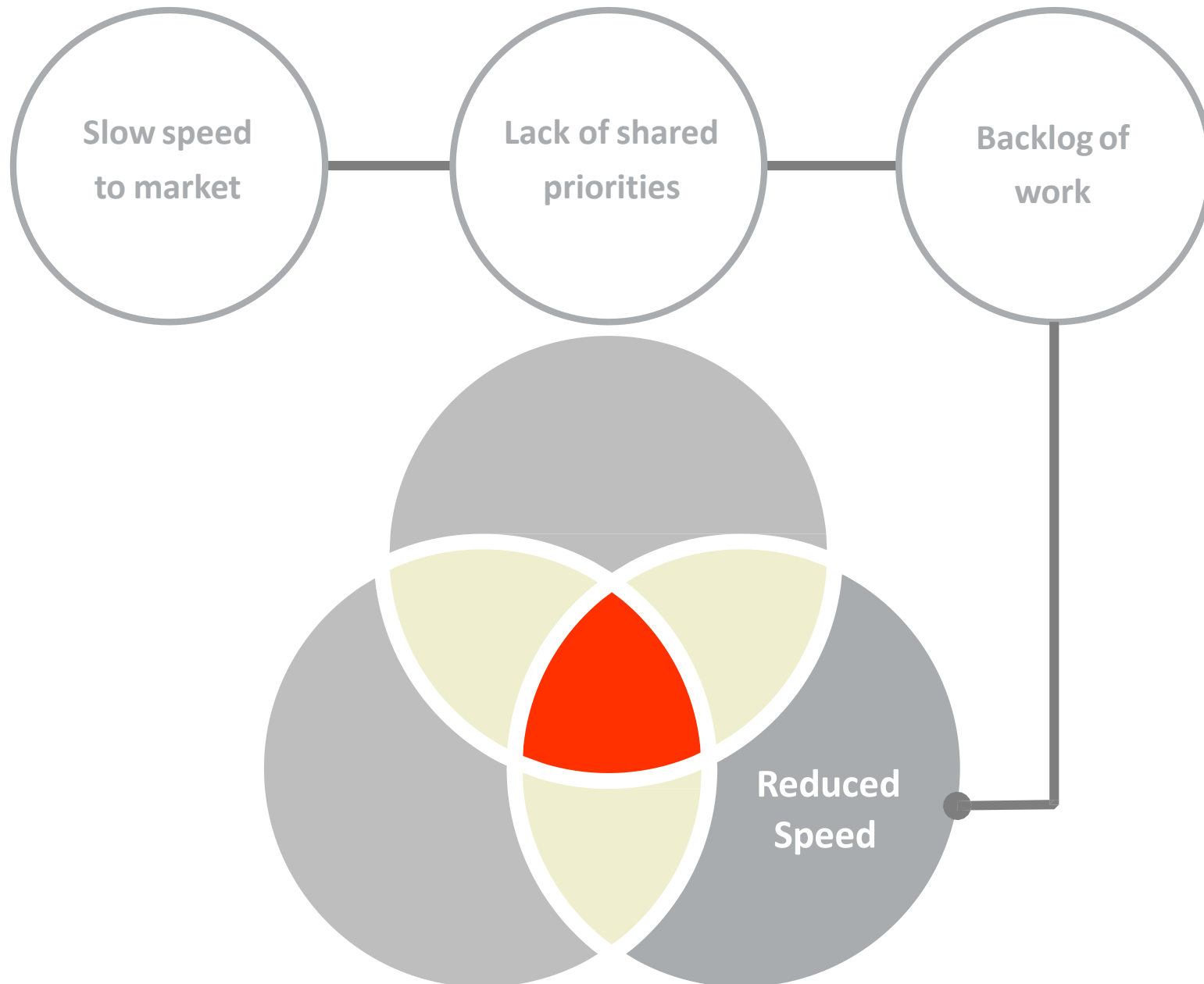


IT Impacts Business Value



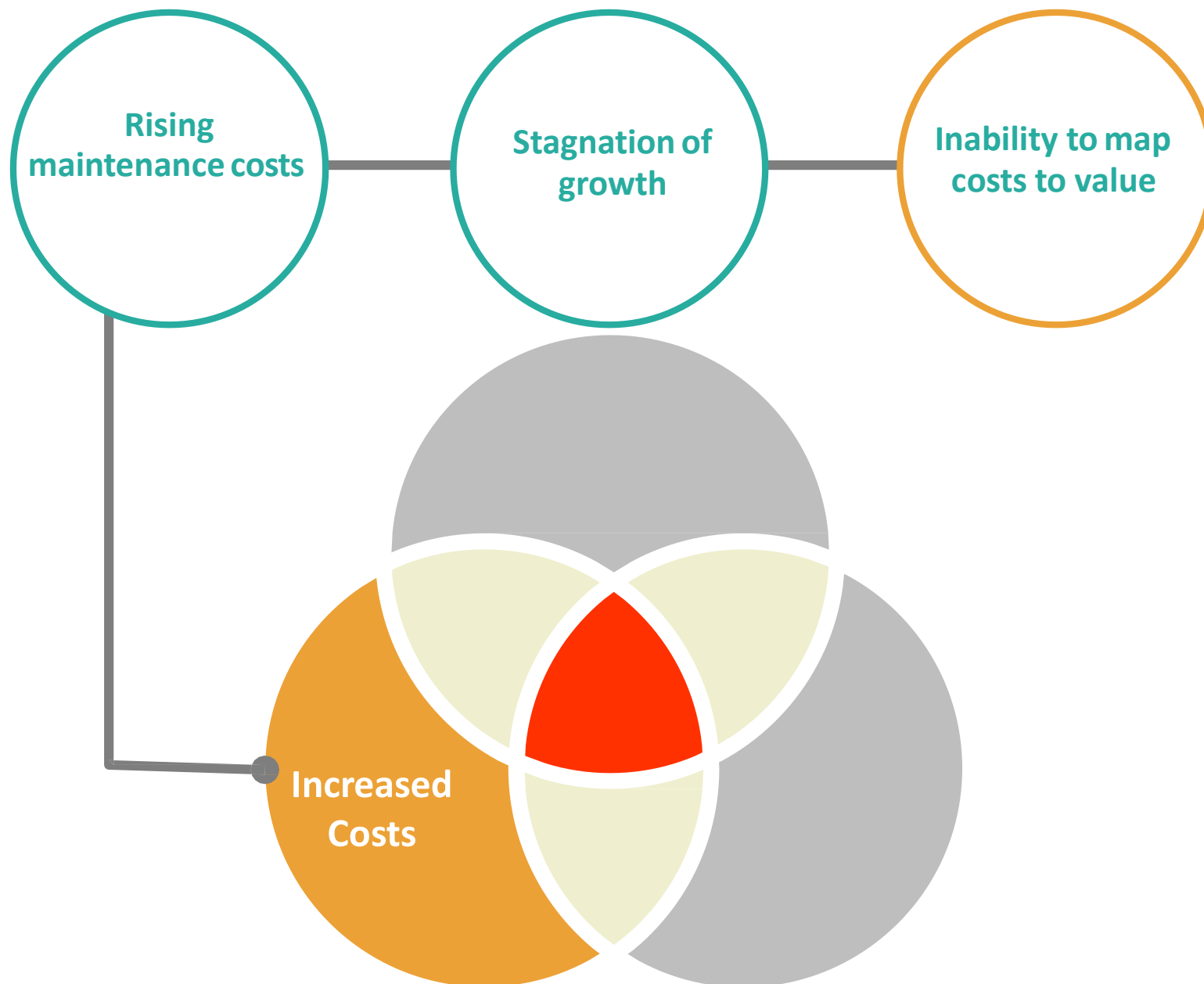


IT Impacts Business Value





IT Impacts Business Value





“Companies live and die on their ability to discover new businesses and create ongoing value for customers. This has always been true, but never more so than in the past few years. Competitive pressure is increasing, fueled by rapid changes in technology and society.”

Jez Humble, Joanne Molesky & Barry O'Reilly
Lean Enterprise



External Drivers Of Change

The Emergence Of
Cloud Technology

Global Supply

Workforce Automation

Digitally Disruptive Competition



The Emergence Of Cloud Technology

“

Cloud computing is about much more than technological capabilities. Technology is the mechanism, but, as in any shift in business, the driver is economics.

Nicholas Carr

The Big Switch: Rewiring The World From Edison To Google

”



“

More people in more places can now compete, connect and collaborate with equal power and equal tools than ever before.

Thomas Friedman
The World Is Flat

”



Workforce Automation

“

AI and robotics increasingly take on the routine aspects of both blue and white collar jobs, leaving the non-routine to humans. That challenges the very essence of what organizations retain as human work. The reconfiguration of these non-routine activities will yield new and different types of jobs.

Ravin Jesuthasan & John Boudreau

*Thinking Through How Automation Will Affect
Your Workforce*

”



Digitally Disruptive Competition

“

When the technology that has potential for revolutionizing an industry emerges, established companies typically see it as unattractive, they ignore it. But then another company steps in to bring the innovation to a new market. There disruptive technology becomes established, small scale innovation rapidly raises technology performance expectations.

Joseph Bower
Disruptive Change: When Trying Harder Is Part Of The Problem

”



Disruptive Competition In Action

Macy's is founded in
New York City



1858

1925



Sears is founded as a mail
order catalog company

The post-war boom in suburban
development encourages the
spread of shopping malls



1950s

1962



The first Walmart opens,
ushering in the era of the big-
box discount department store

Amazon.com is founded
by Jeff Bezos



1994



The Pace Of Digital Disruption Today

Sears, Amazon and rivals

■ Average ■ SHLD ■ AMZN



Source: Investopedia



The Organizational Goals Of DevOps

Faster time to market

Better integration between IT and internal partners and stakeholders

Improved responsiveness

Improved code and deployment quality

Increased productivity

Better visibility and transparency

Increased flexibility and agility

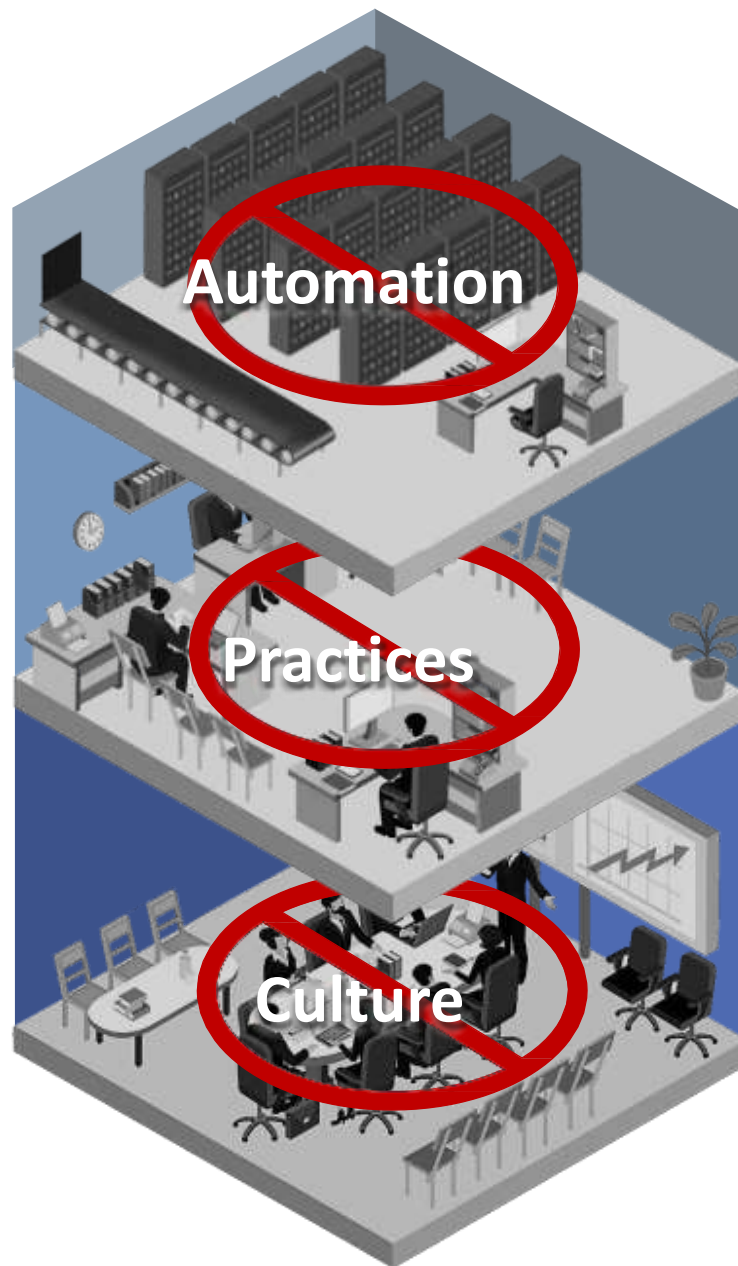
**DevOps seeks to facilitate a
necessary digital
transformation.**



The IT Value Delivery Problem



Why Is IT Unable To Deliver Value?





How Silos Became The Norm

Adam Smith advocates for task specialization in *The Wealth Of Nations*



1776

Frederick W. Taylor introduces the idea of “scientific management”



1911

1908



Henry Ford's assembly line-produced Model T revolutionizes American industry

Present Day



A focus on task specialization, control and measurement are all still present



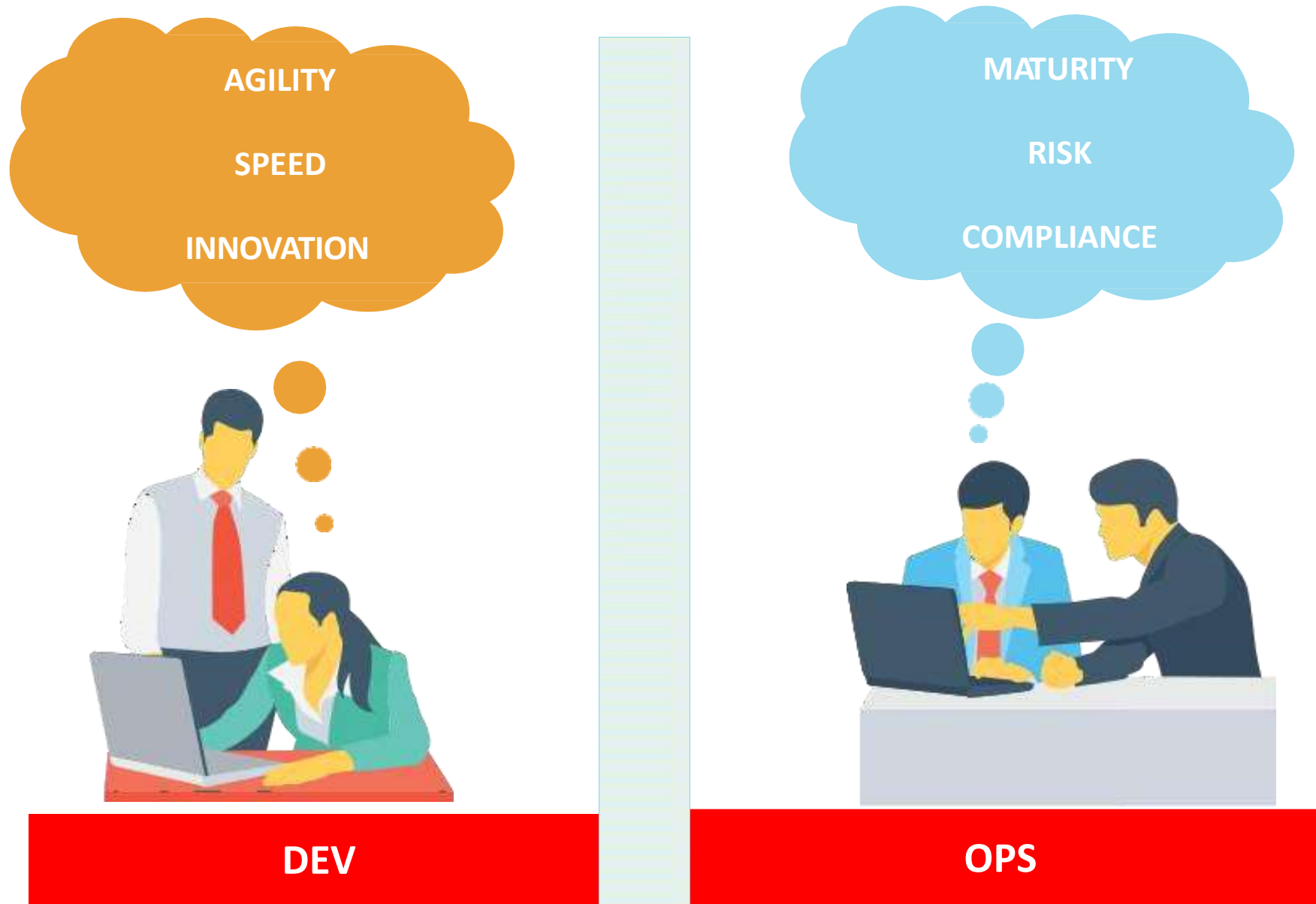
**Current organizational structures
encourage a silo mentality and
discourage collaboration.
Today challenge is to switch
from internal competition to
collaboration**

Silos increase awareness of where you are, but they prevent you from seeing the larger system – where you are on the map – to ultimately get where you need to be.





Silos Create A Wall Of Confusion





**Local optimization creates
process challenges.**



The Negative Influence Of Bureaucracy

Strengths Of Bureaucracy	Weaknesses Of Bureaucracy
■ A division of labor into spheres of influence, specialization ■ A definite hierarchy & authority ■ Clear norms and rules ■ Job assignment based on skills & qualification ■ Promotion by seniority ■ Oversight & governance over the incumbent of each office ■ Better than feudal/traditional forms where people got appointed by favoritism or bribes	■ Becomes an Iron Cage of Control ■ Red Tape grows over time as a reaction to process failure ■ Hard to change or reverse ■ Divisions of labor promote silo behaviors & culture ■ Hierarchy can mean long approval times (e.g. must go up and down chains of command to get things done) ■ Decision bias towards silo vs. the overall organizational goals



Complex Processes Result In Waste

Incidents that occur hundreds of times without anyone looking at the root of the problem

Supplier contracts that expire without knowledge until an incident occurs

IT assets that are poorly tracked, controlled or inventoried

Incident tickets that disappear into an IT back office black hole until someone shouts loudly enough

Multiple Service Desks all with their own tools and separate processes

Massive amounts of wasted server capacity due to a lack of capacity and demand management

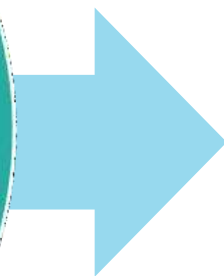
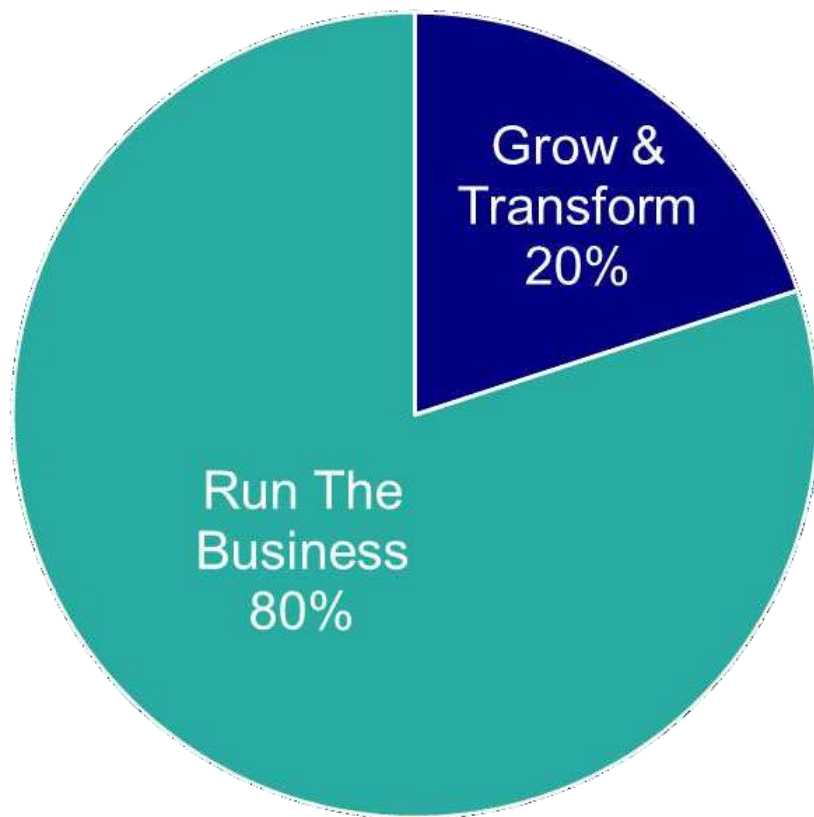
Redundant or duplicate tools purchased by various IT departments in the same organization

Multiple Change Control processes due to political boundaries

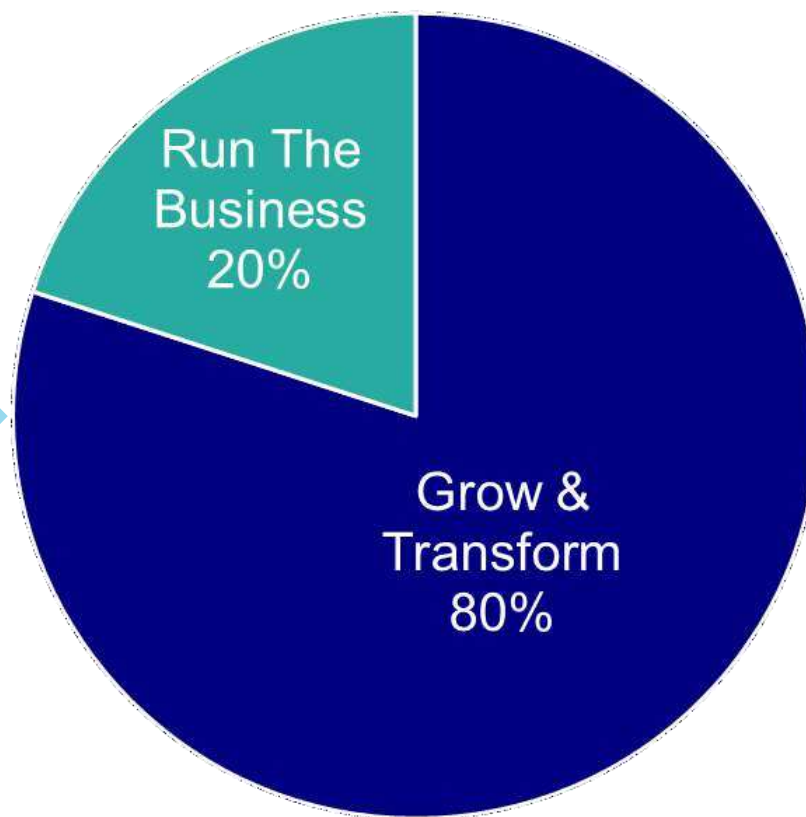


A Big Shift Is Required

Current Allocation Of IT Budget



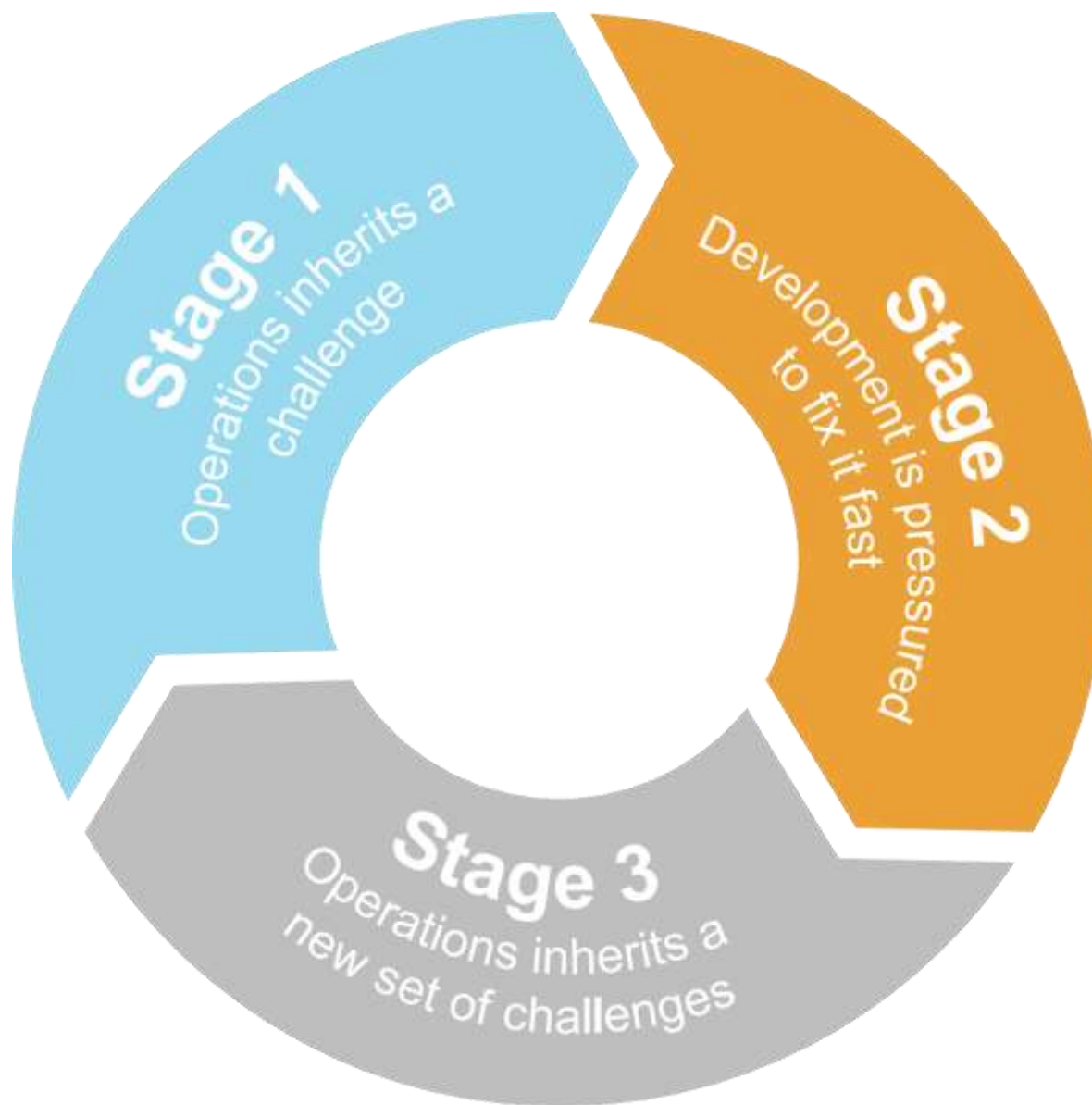
Ideal Allocation Of IT Budget



**The lack of capacity to
innovate, grow or transform
chains IT to legacy systems
and tools.**

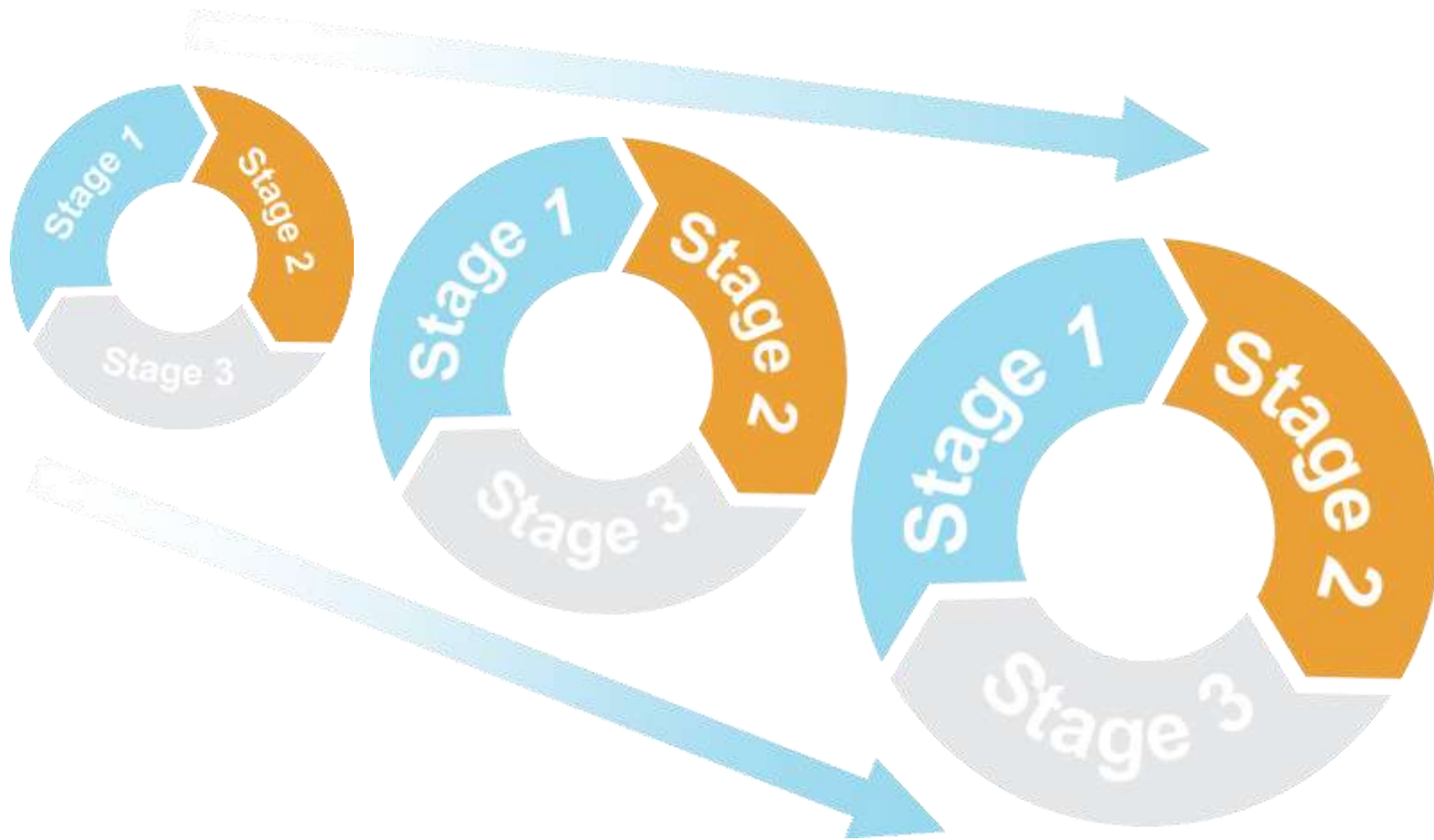


IT Downward Spiral



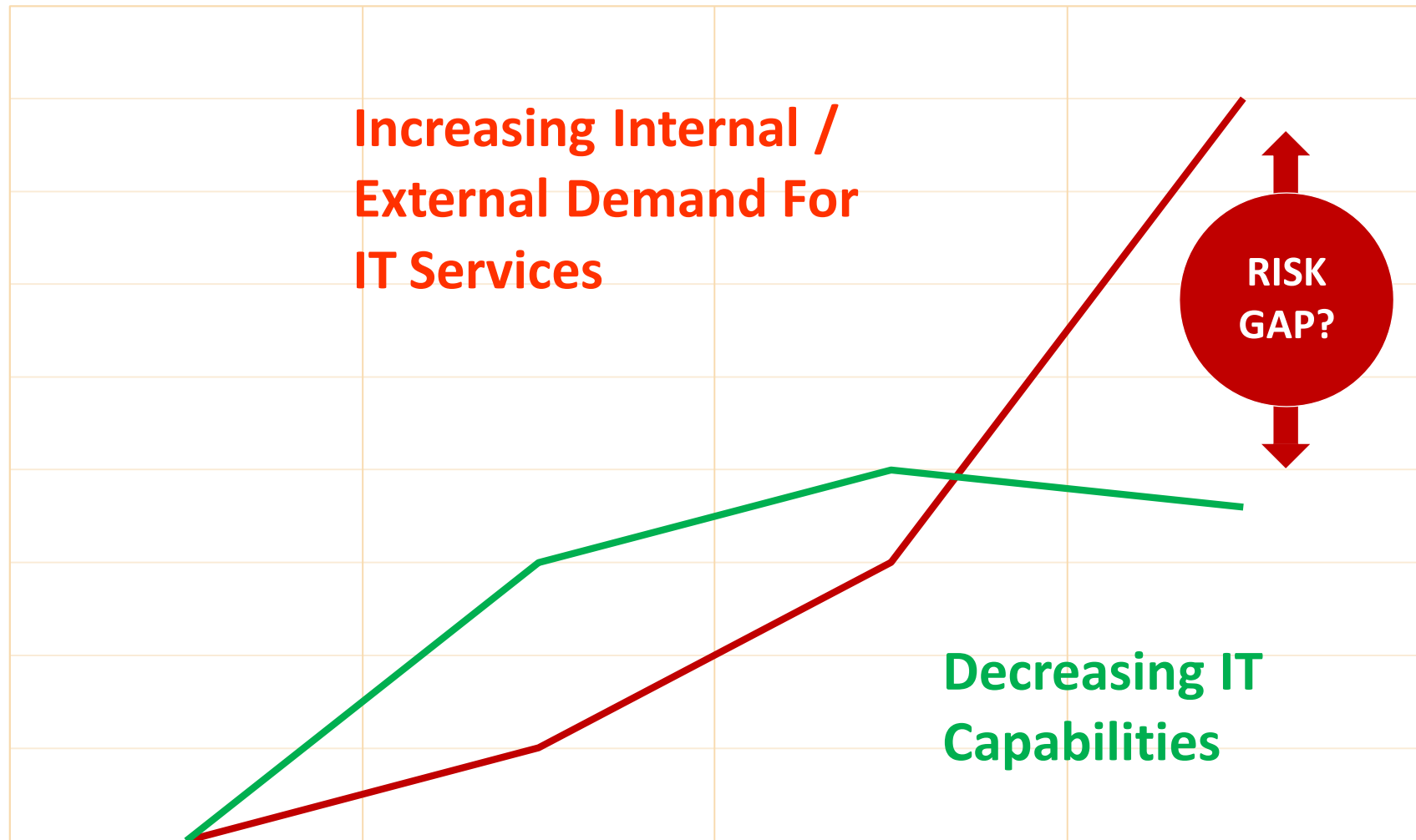


The Accumulation Of Technical Debt





What Happens If IT Can't Deliver Value?





IT Goals Of DevOps

Faster delivery of
features

Increased
collaboration and
communication

Faster resolution of
problems

Continuous
software delivery

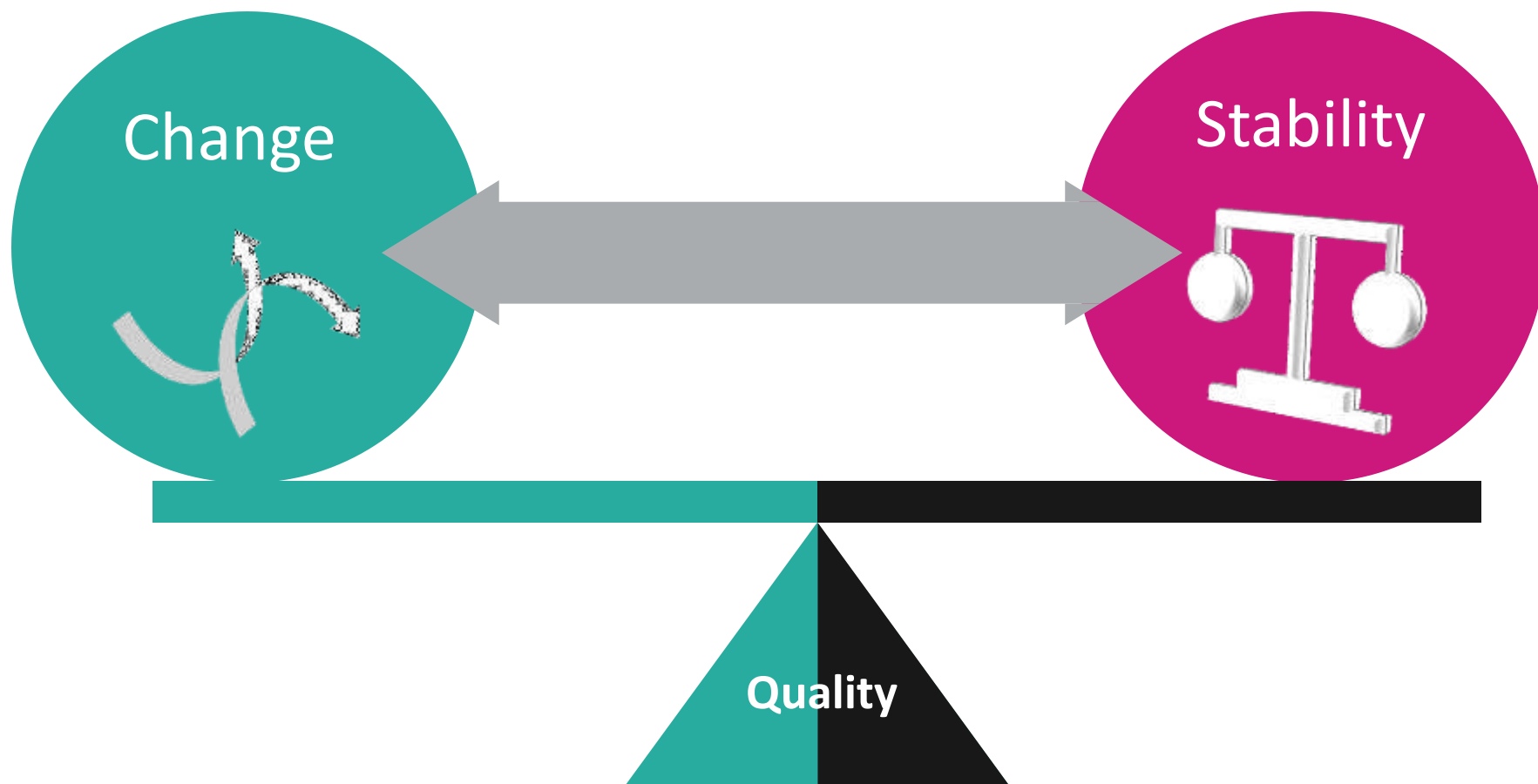
More productive
teams

Reduced
complexity to
manage

More time to
innovate



DevOps Helps IT Find The Balance





PeopleCert DevOps Fundamentals

MODULE 2:

Key DevOps Principles & Concepts

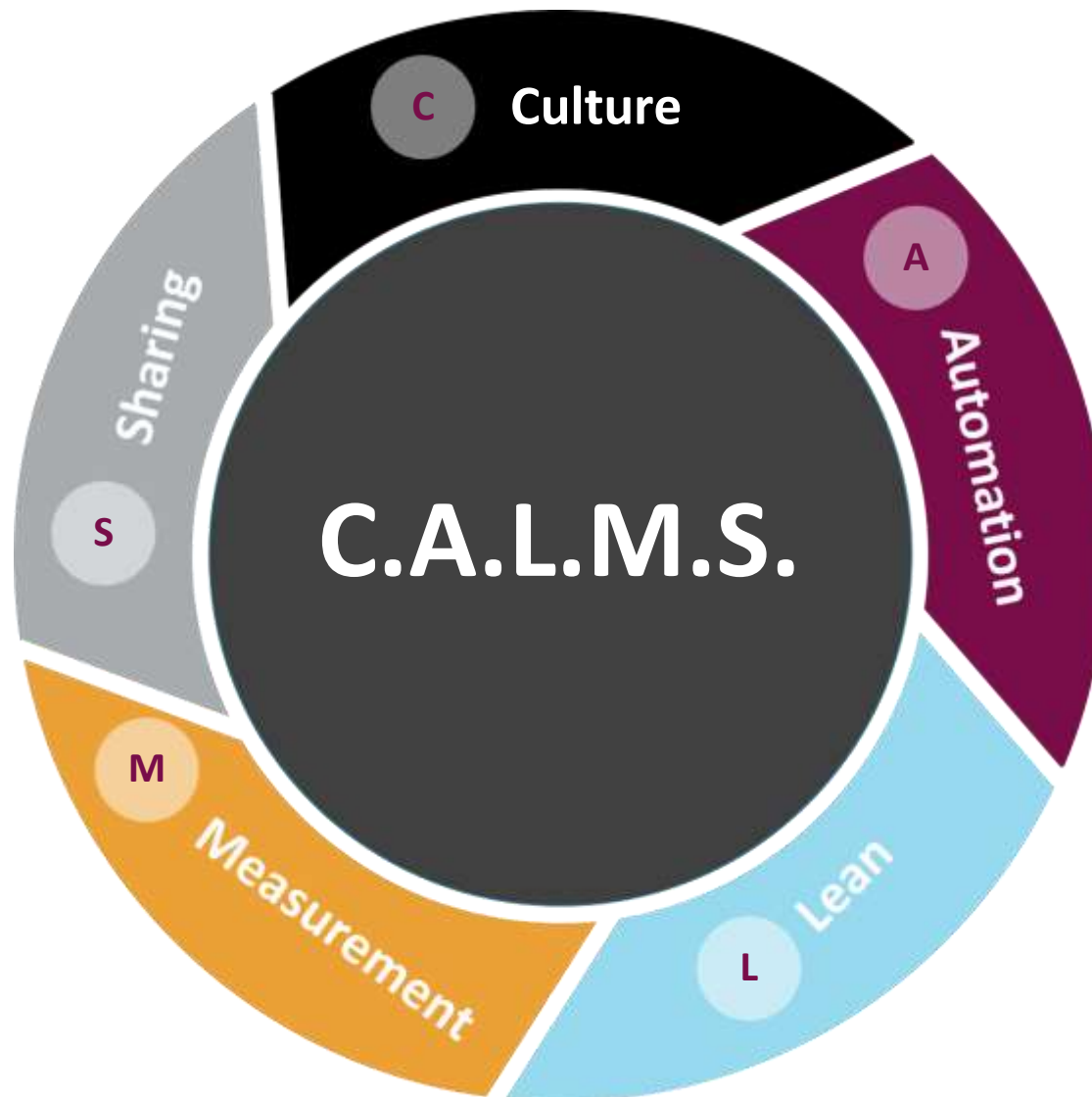
- **C.A.L.M.S. & The Three Ways**
- **Continuous Delivery Across The Deployment Pipeline**
- **The Scope Of The Full Stack**



C.A.L.M.S. & The Three Ways

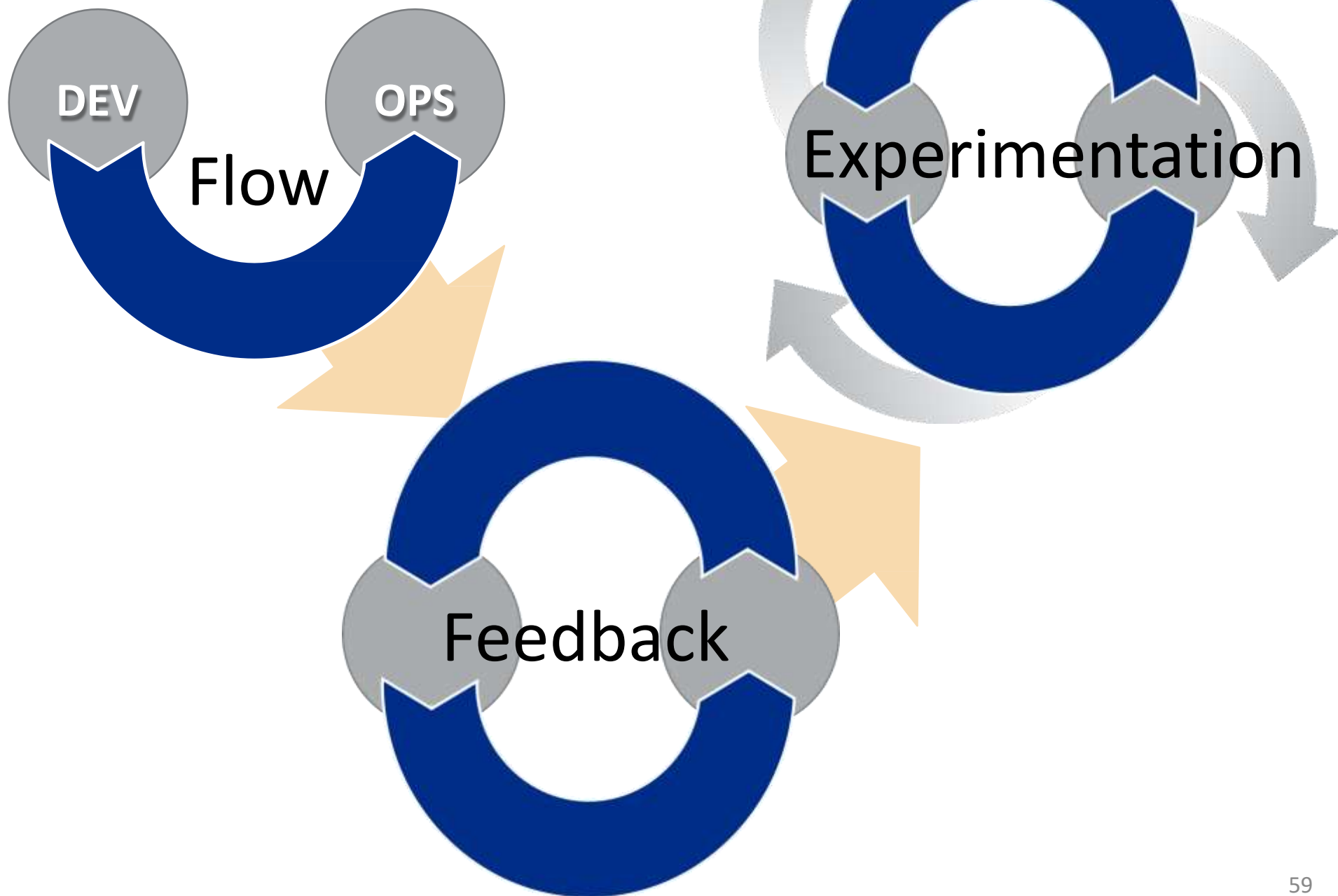


C.A.L.M.S.



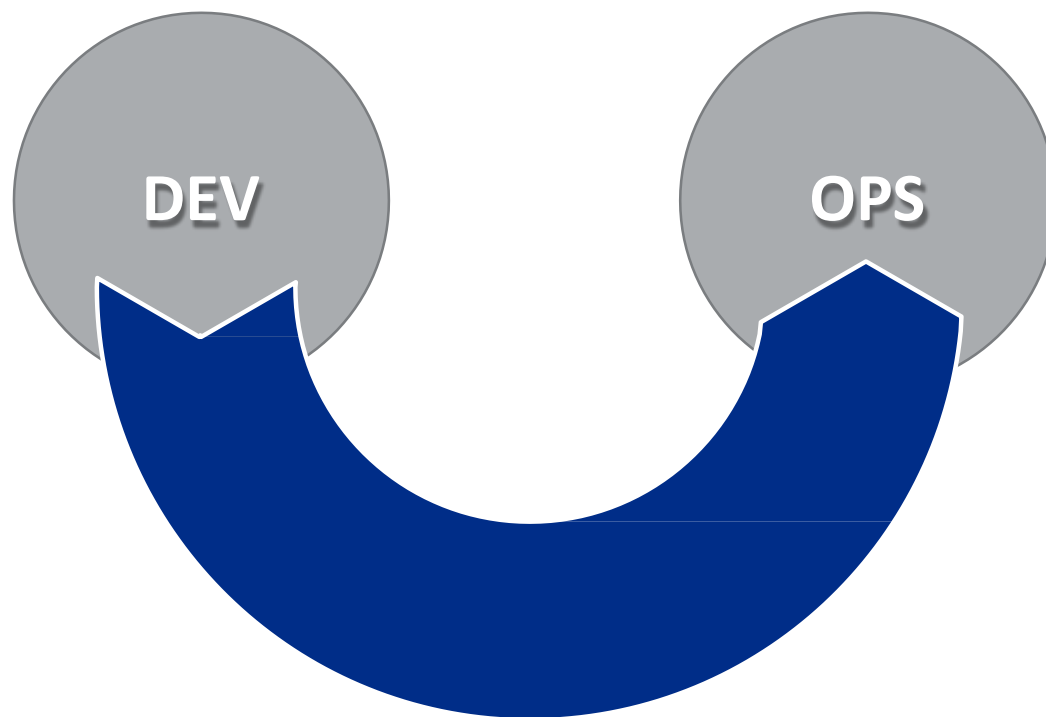


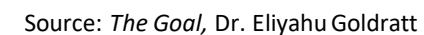
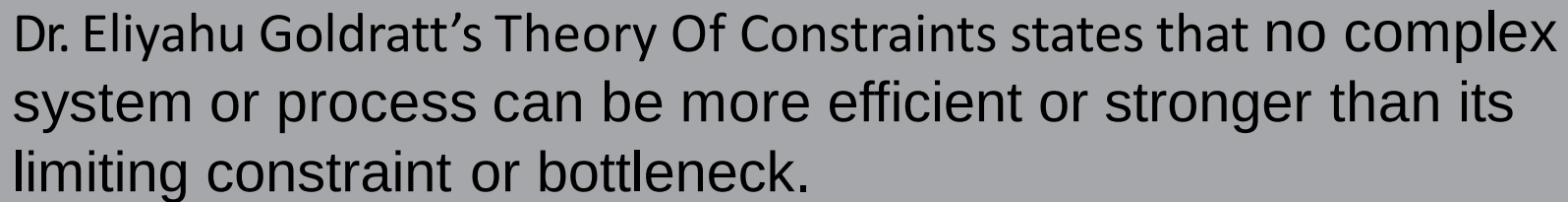
The Three Ways





The First Way: Flow







How Do You Increase Flow?

**Increase
transparency
and visibility
of work**

**Place limits on
Work In
Progress (WIP)**

**Work in
smaller
batches**

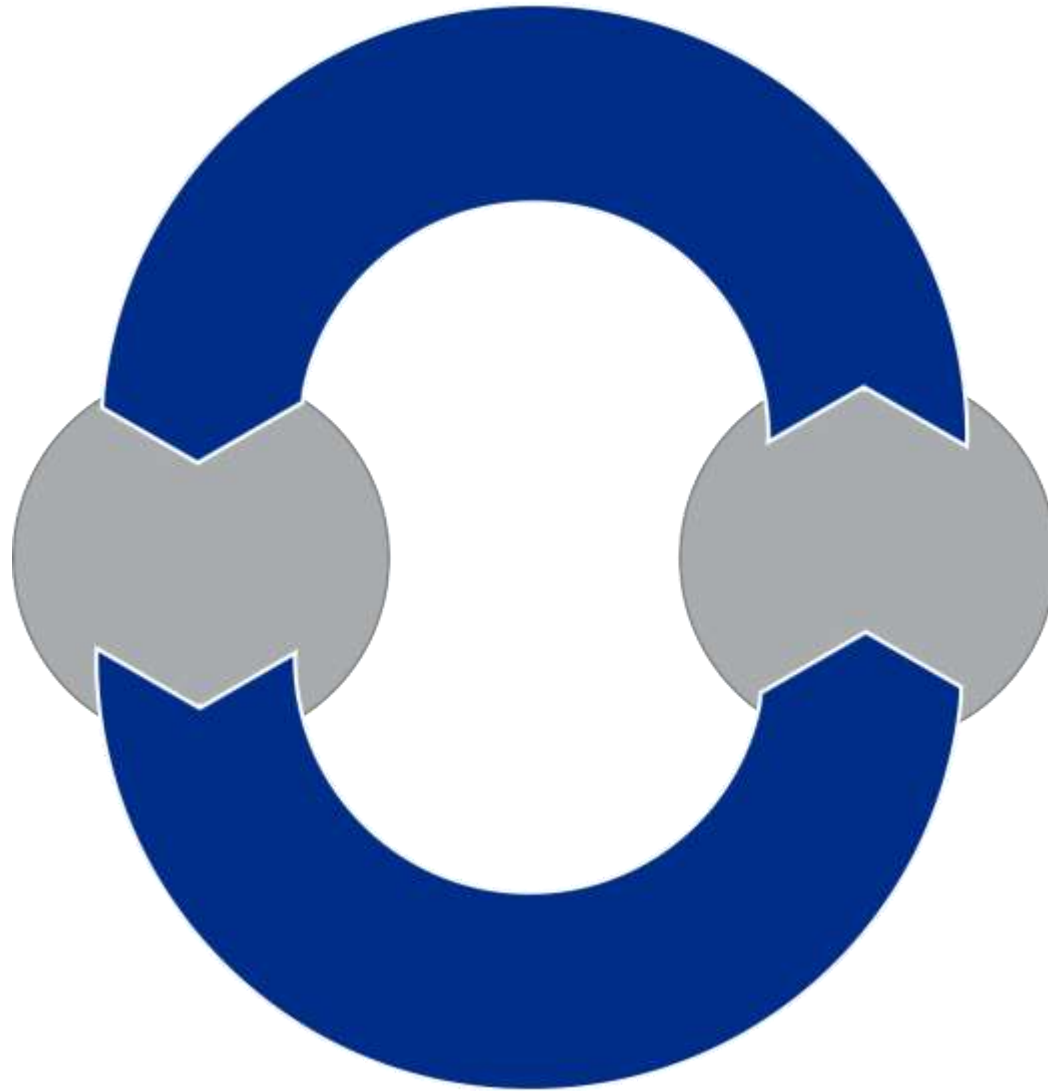
**Reduce and
limit the
number of
handoffs**

**Identify and
prioritize
constraints or
bottlenecks**

**Continually
improve to
eliminate
waste
wherever
possible**



The Second Way: Feedback





How Do You Increase Quality & Resiliency?

**Detect
problems
earlier, as
they occur**

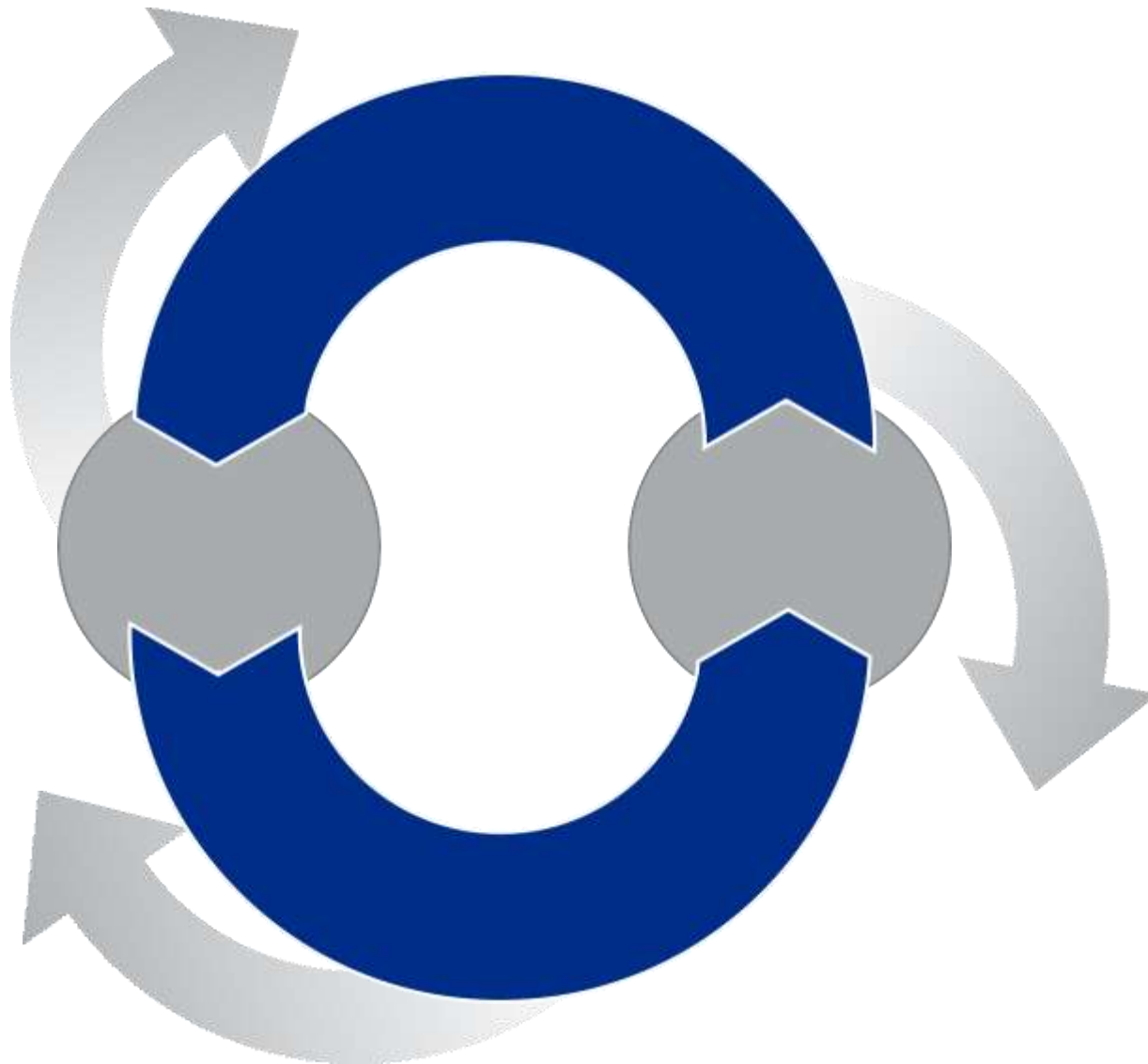
**Swarm to
solve
problems
and gain
new
knowledge**

**Push quality
closer to the
source**

**Optimize
globally vs.
locally**



The Third Way: Continual Experimentation





How Do You Encourage Experimentation?

Enable a safe
culture for
learning

Institutionaliz
e continual
daily
improvement

Turn local
knowledge
into global
improvements

Implement
resilience
patterns in
daily work

Transform
leadership to
encourage
learning

Approach failure
as a learning
opportunity and
plan on failing
faster



Continuous Delivery Across The Deployment Pipeline



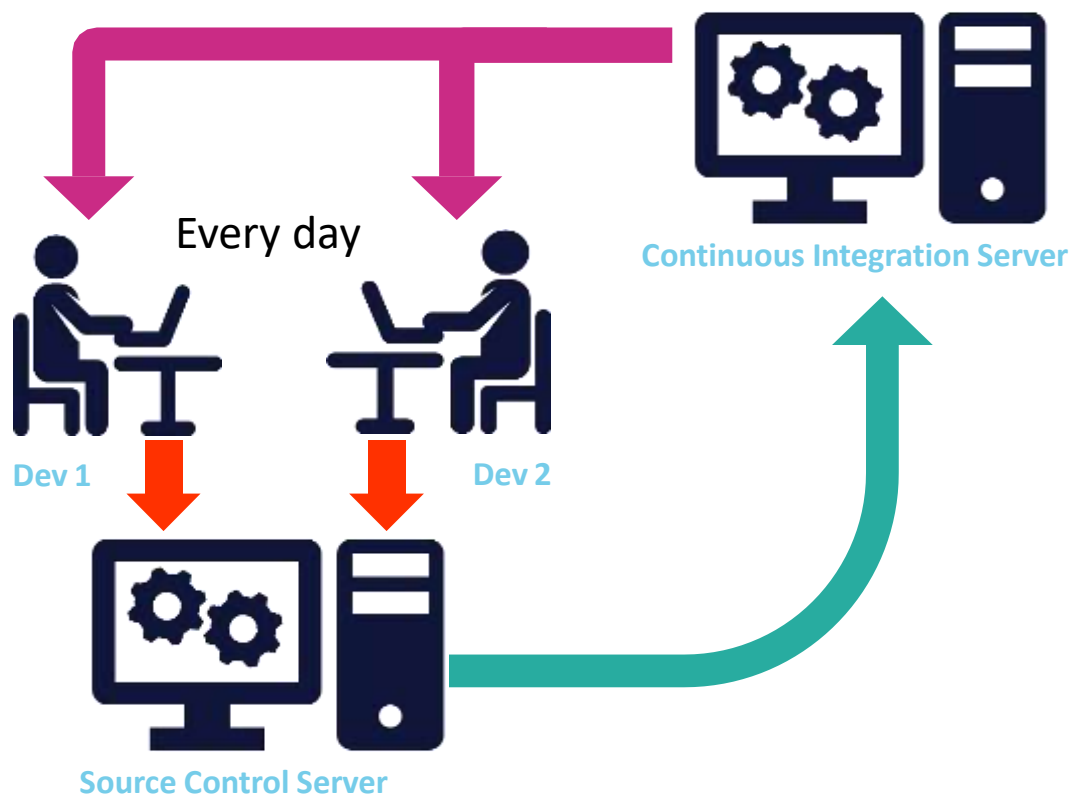
The Deployment Pipeline



The Deployment Pipeline is a **process model** that gives you a starting point from which to provide transparency and controls as work moves through the various sets of tests and deployments towards release.

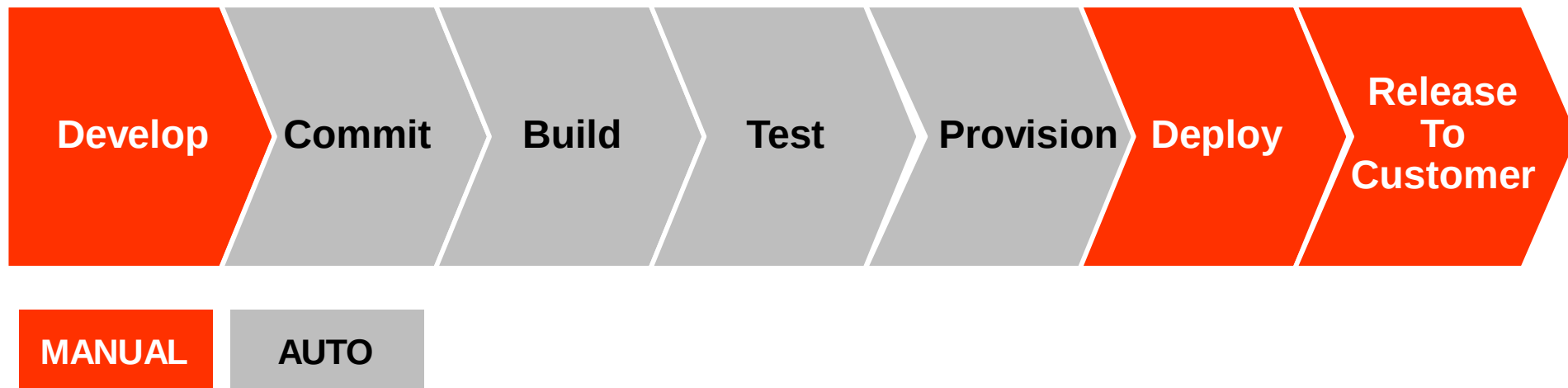


Continuous Integration





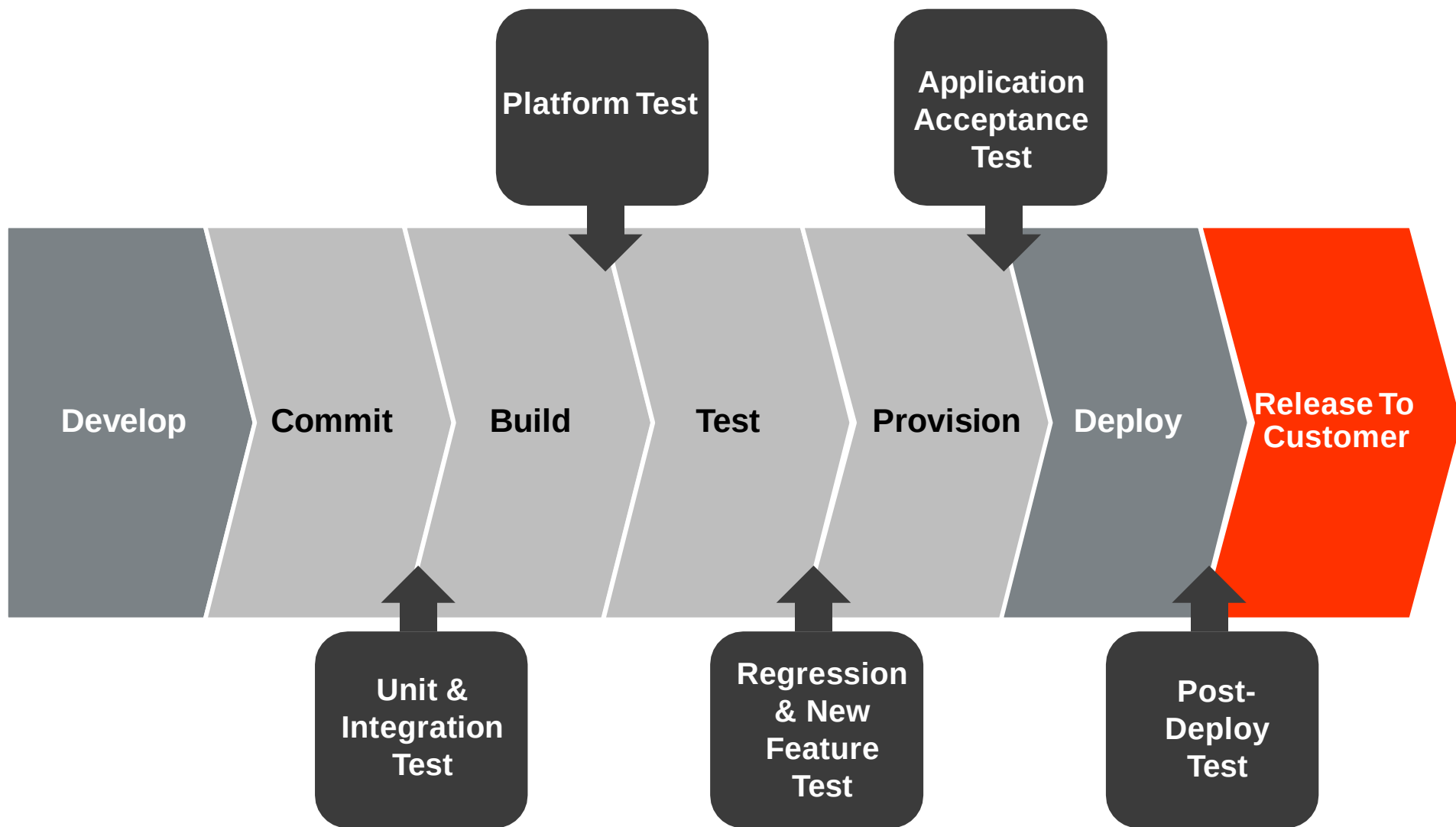
Continuous Delivery



Continuous Delivery is a set of practices designed to ensure that code is always able to be deployed rapidly and safely throughout its lifecycle to production, achieved by pushing the executables into a production-like environment and conducting automated testing to detect problems.



Continuous Testing



MANUAL

AUTO



Functional & Non-Functional Testing

Functional Testing



Unit Tests



API



Integration



**System
Testing**

Non-Functional Testing



Performance



Security



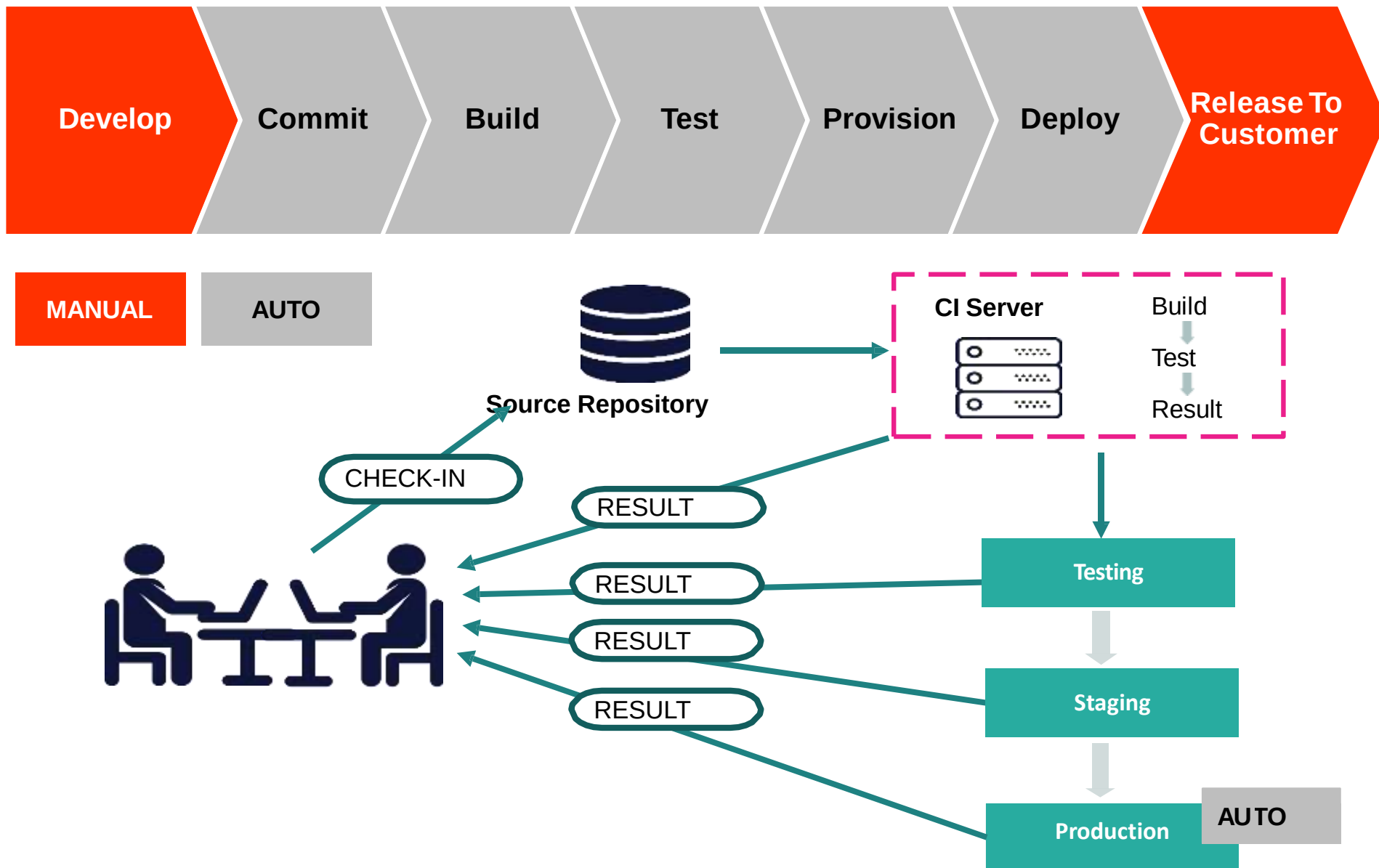
Compliance



Capacity



Continuous Deployment





The Benefits Of Continuous Delivery

Continuous Delivery across the deployment pipeline means...

Effective
test data
management

Comprehensive,
fast and reliable
test and
deployment
automation

Trunk based
development
and continuous
integration

Application
code, app and
system
configuration all
in version
control

Incorporating
security (and
security teams)
into the delivery
process

Which leads to...

Less rework

Lower levels of
deployment
pain

Higher level of
IT performance
(higher
throughput and
stability)

Lower change
failure rates

Identifying
strongly with
the organization
you work for

Higher levels of organizational performance.



The Scope Of DevOps



DevOps Changes Everything





DevOps Transforms People & Culture





DevOps Transforms Processes & Practices

#1

**Voice Of The
Customer**

#2

**Relationship
Management**

#3

**Lean Process
Optimization**

#4

**Value Stream
Mapping**

#5

**Knowledge
Management**

#6

**Visual
Management**

#7

Agile Scrum

#8

**Shift Left
Testing**

#9

**Change
Control**

#10

**Service Service
Configuration
Management**

#11

**Release &
Deployment
Management**

#12

**Incident
Management**

#13

**Problem
Management
& Kaizen**

#14

**Continual
Improvement**

#15

Antifragility

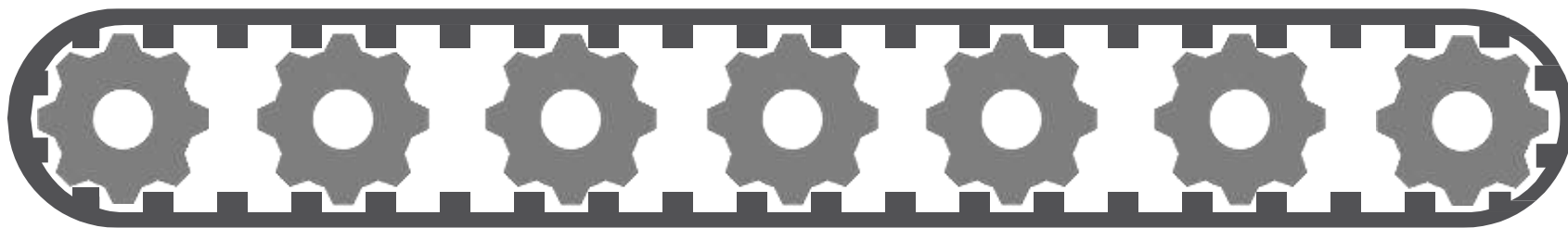


DevOps Transforms Automation

Cloud Technology &
Virtualization

Toolchains For
Automation Of The
Deployment Pipeline

New Architectures For
Continuous Delivery &
Deployment





PeopleCert DevOps Fundamentals

MODULE 3:

The Full Stack: People & Culture

- What Is A DevOps Culture?
- Transformational Leadership
- What Is A DevOps Team?

“

Culture is a pattern of shared tacit assumptions that was learned by a group as it solved its problems of external adaptation and internal integration, that has worked well enough to be considered valid and, therefore to be taught to members as the correct way to perceive, think, and feel in relation to those problems.

”

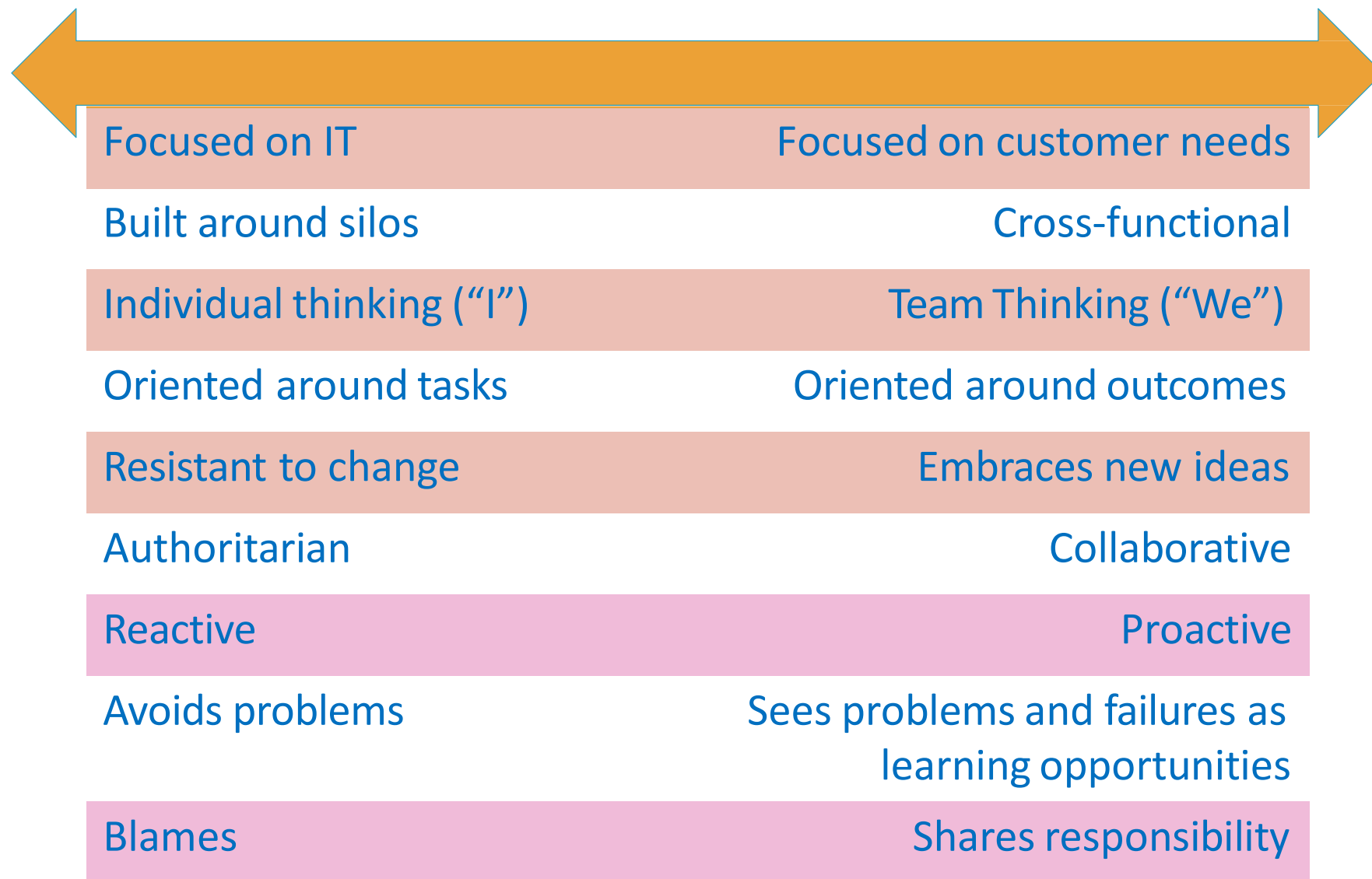
Edgar Schein

A close-up photograph of a person's hands and arms working on a server rack. The person is wearing a white t-shirt. The server rack is filled with various network equipment, including a blue switch or router with many white and blue cables plugged into its ports. The cables are bundled and organized. The background is slightly blurred, showing more of the server environment.

DevOps is a transformation of people and culture.



What Is Your Organizational Culture?





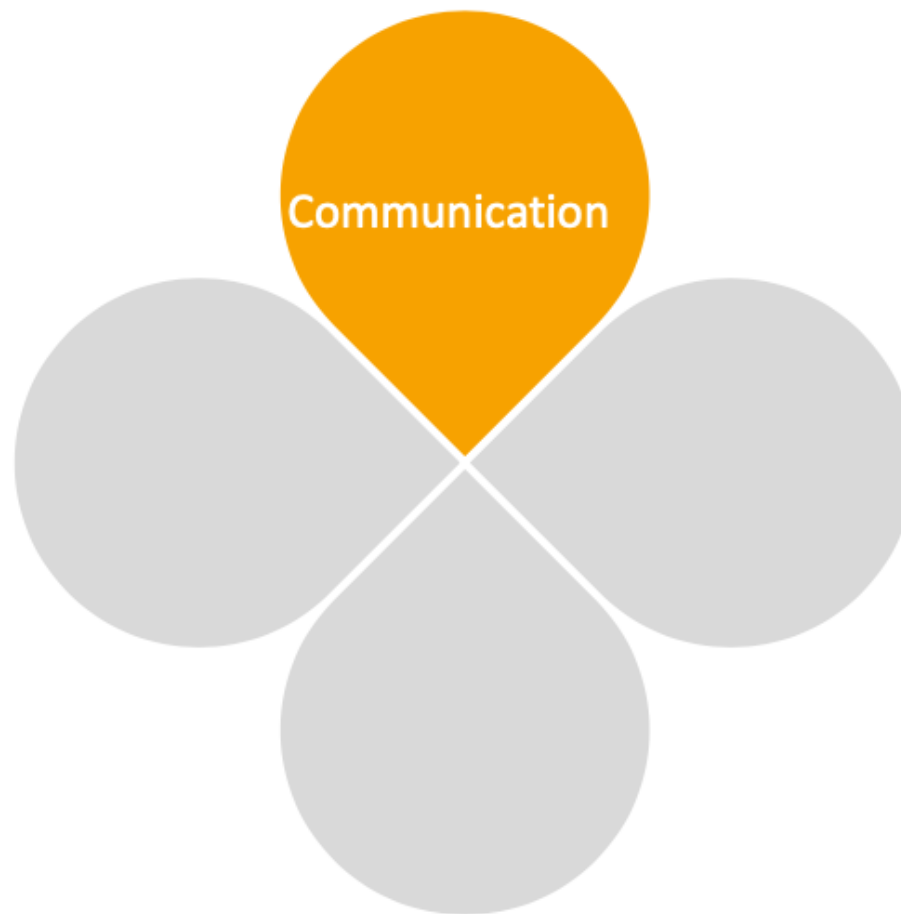
What Is A DevOps Culture?

Breaking down the
“wall of confusion” between
Dev and Ops means
collaboration.



Barriers To Communication

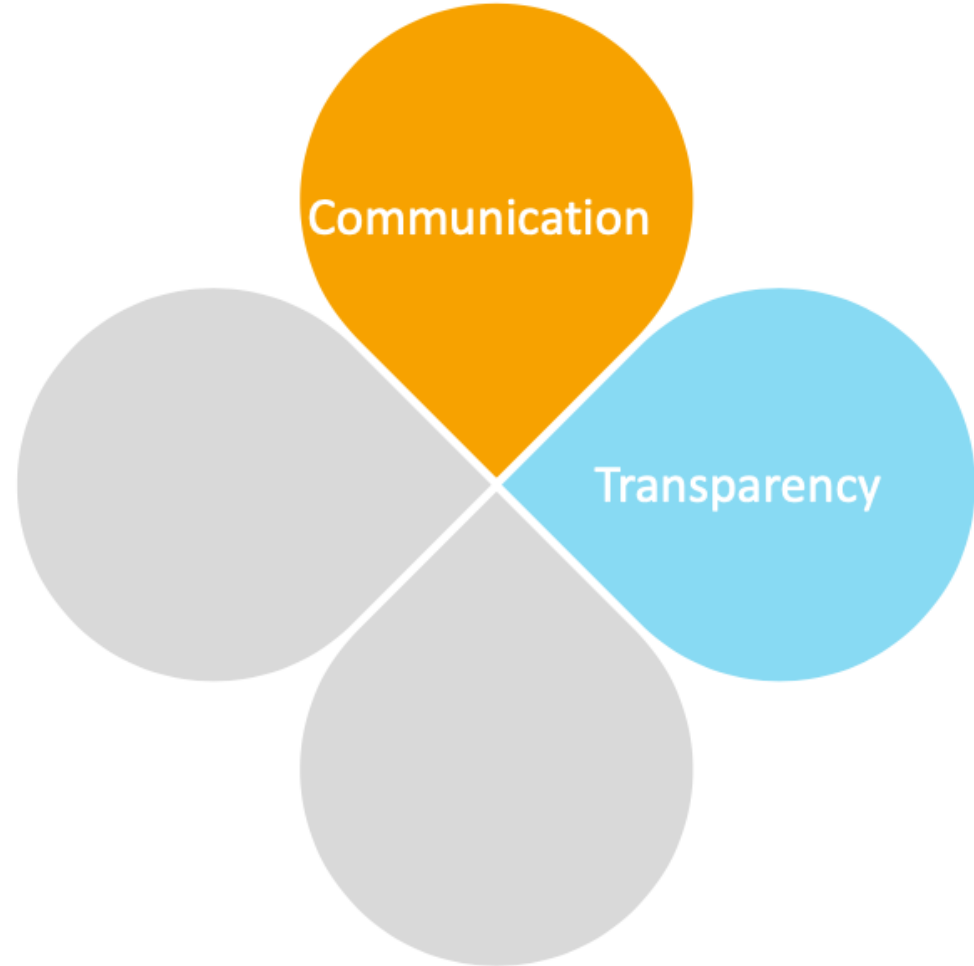
- Long wait time for responses to questions or problems
- Inability to request help or ask questions and receive a prompt response
- Insufficient tools





Barriers To Transparency

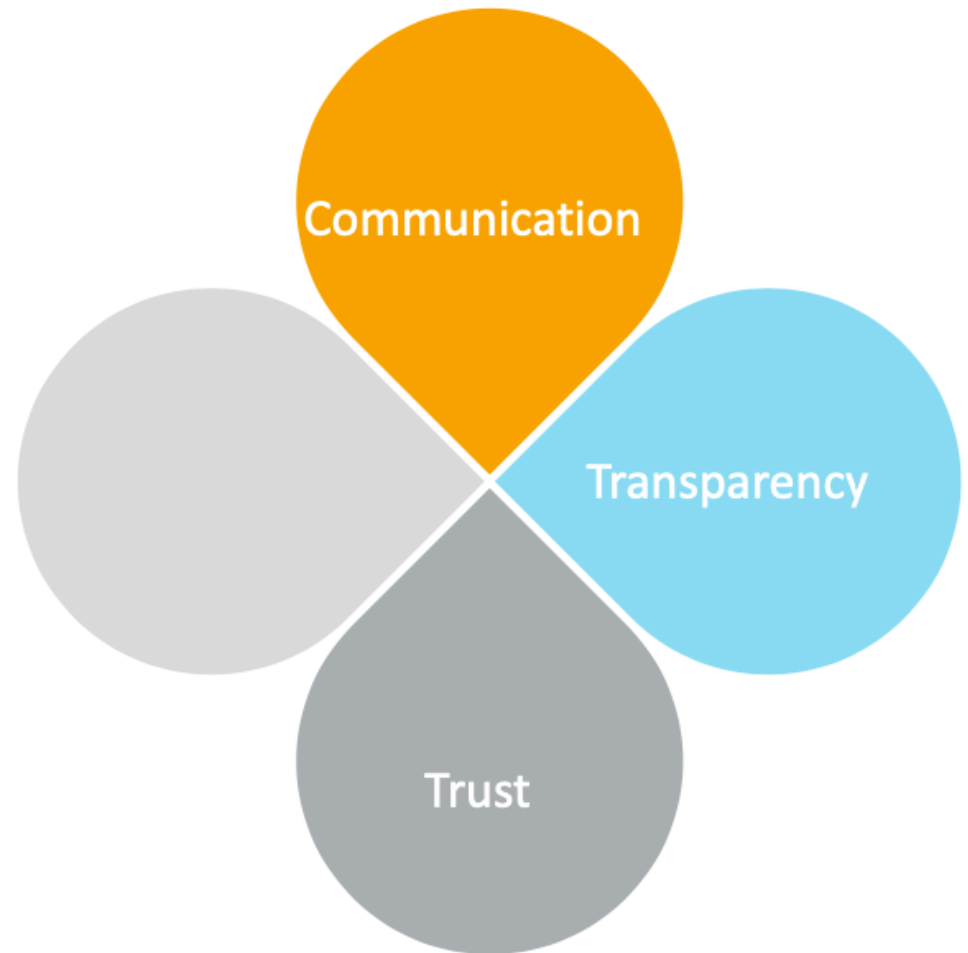
- Lack of understanding about why each side does things the way it does
- Invisibility of work that means you cannot see what others are working on or have done
- Complex processes that lack documentation or clarity
- Lack of shared knowledge





Barriers To Trust

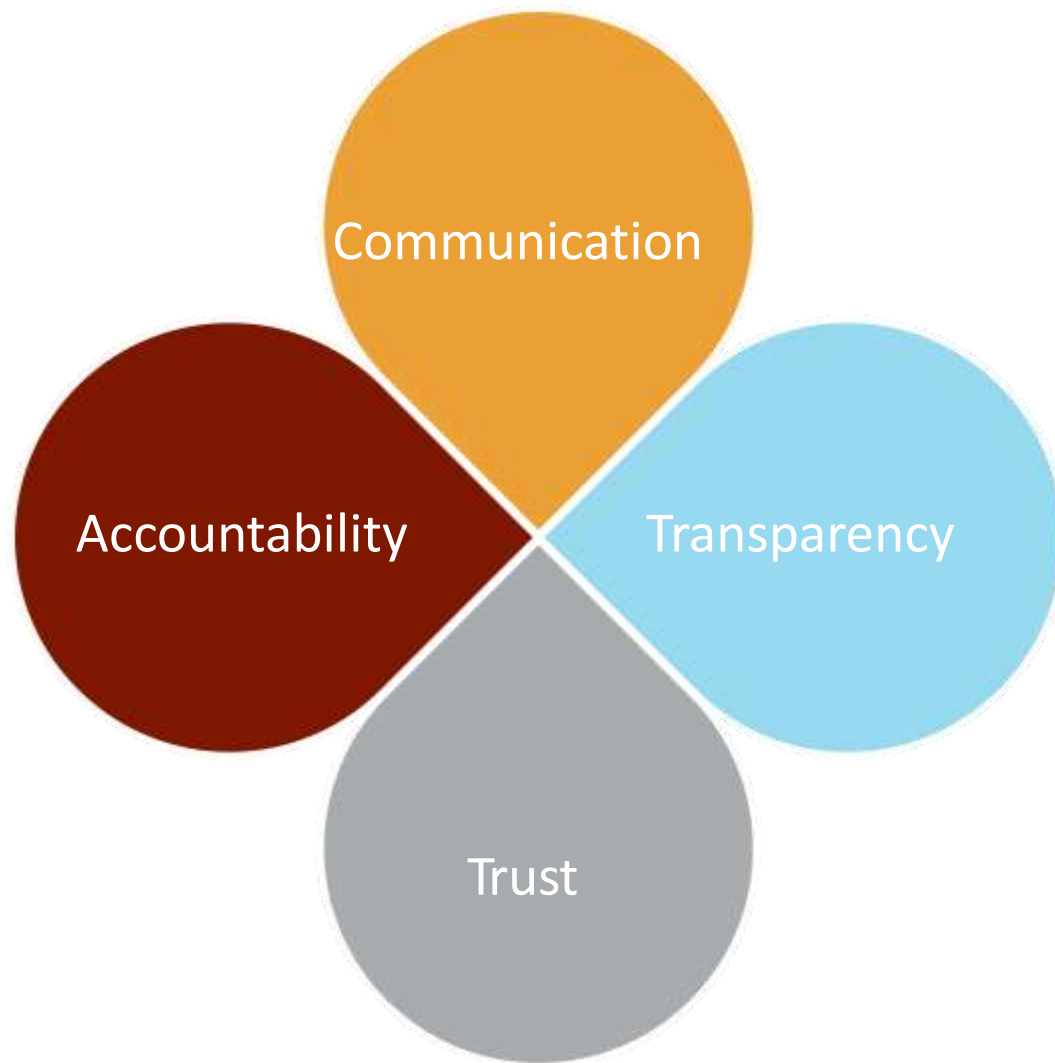
- Lack of ability to prioritize each other's needs or understand each other's challenges
- Belief that others do not share priorities or goals
- Lack of appreciation or recognition of skills or abilities
- Rejection of new ideas or innovations





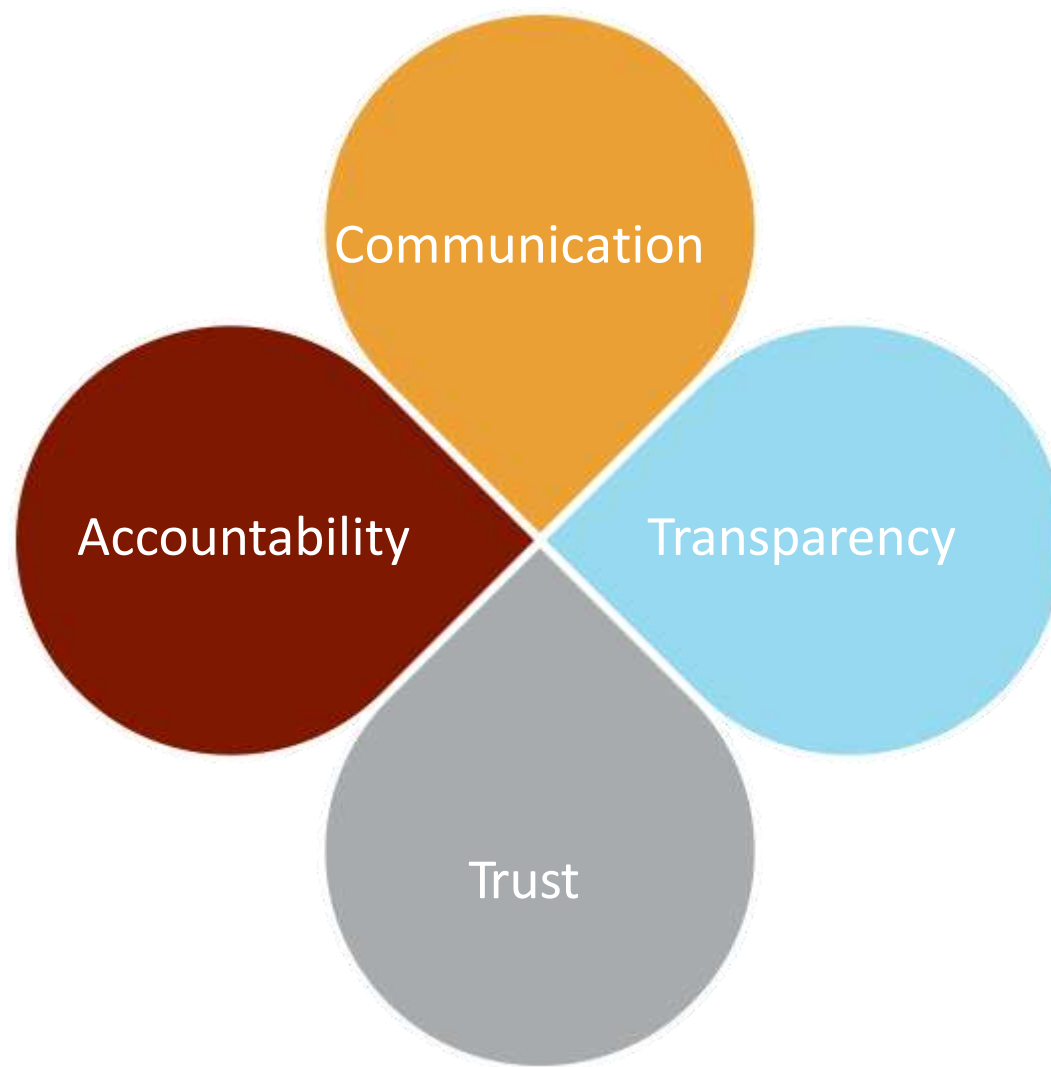
Barriers To Accountability

- Overlap of roles and responsibilities that causes conflict
- Blaming or scapegoating when failure occurs
- Lack of feedback from Ops to Dev
- Hierarchical organizational structure where leaders hoard information



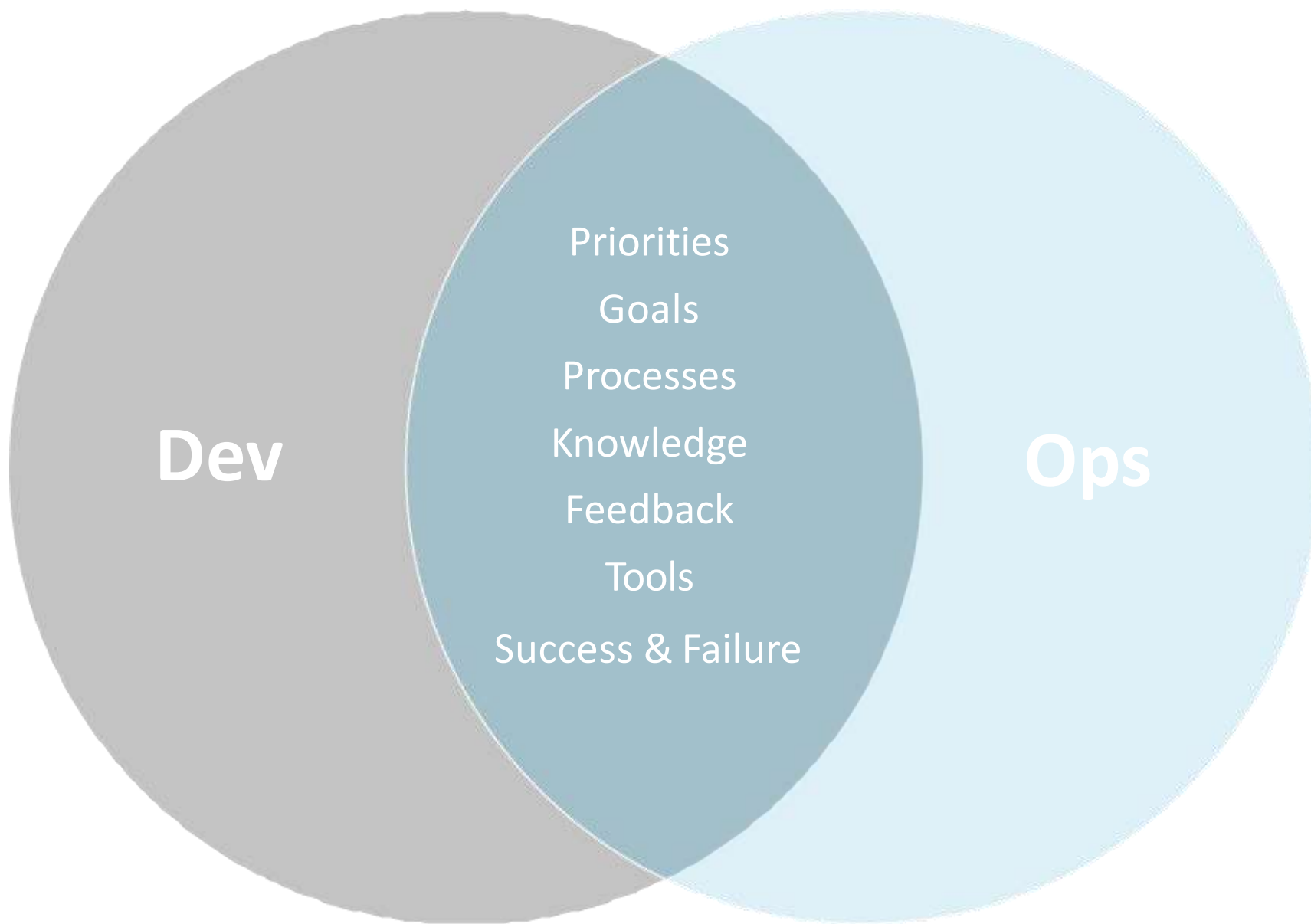


Key 4 Characteristics Of Collaboration





DevOps Is About Sharing

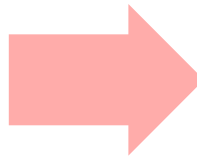




DevOps Culture Is Generative (Westrum)

Pathological

- Power oriented
- Low level of cooperation
- Messengers are “shot”
- Responsibilities are shirked
- Bridging is discouraged
- Failure leads to scapegoating
- Novelty is crushed



Generative

- Information is actively sought
- Messengers are trained
- Responsibilities are shared
- Bridging is rewarded
- Failure causes enquiry
- Novelty is implemented

Source: A Typology Of Organisational Cultures, R. Westrum



DevOps Encourages Experimentation

DevOps encourages failure, but responsible failure – calculated risks, failing fast in small ways, and then learning and implementing solutions that actually work in the aftermath.

“

The biggest job we have is to teach a newly hired employee how to fail intelligently. We have to train him to experiment over and over and to keep on trying and failing until he learns what will work.

Charles F. Kettering

”



Transformational Leadership



What Is Transformational Leadership?

Establishing and supporting generative
and high-trust cultural norms

Implementing technologies and processes that enable
developer productivity, reducing code deployment lead times
and supporting more reliable infrastructures

Supporting team experimentation and innovation,
to create and implement better products faster

Working across organizational silos to achieve strategic alignment



Transformational Leadership is a model in which leaders inspire and motivate followers to achieve higher performance by appealing to their values and sense of purpose, facilitating wide-scale organizational change.

2017 State Of DevOps Report





Transformational Leader Characteristics

Vision

Inspirational
Communication

Intellectual
Stimulation

Supportive
Leadership

Personal
Recognition



Source: 2017 State Of DevOps Report, Puppet Labs



Transformation Requires Vision



An aspirational expression of what the organization would like to achieve that serves as a clear guide for future action.

“

‘Starting with the End in Mind’ means to start with a clear understanding of your destination. If you know where you are going you will better understand where you are now so that the steps you take are always in the right directions. By keeping the end in mind you make sure the actions you take each day contribute to your overall goals.

”

Stephen Covey



Contrasting Leadership Styles

Traditional Leadership

- Works to a financial plan
- Demands quick results
- Leads with answers
- Climbs the ladder rapidly or finds a new ladder
- Sees teams as a means to an end
- Proud of having an open door policy

Transformational Leadership

- Works towards a vision
- Demonstrates patience
- Leads with questions
- Gradually earns way up the ladder
- Develops individuals and enable effective teamwork
- Learns and leads by being present and involved

“

The culture of Do and Tell does not teach us how to change pace, decelerate, take stock of what we are doing, observe ourselves and others, try new behaviors, build new relationships.

Edgar Schein

Humble Inquiry: The Gentle Art Of Asking Instead Of Telling

”



Authority Distribution In Teams

Setting Overall Direction	Management Responsibilities			
Designing The Team & Its Context				
Monitoring & Managing Work Processes			Team Responsibilities	
Executing The Task				
	Manager-Led Team	Self-Managing (Self-Organizing) Team	Self-Directed Team	Self-Governed Team

Source: *The Psychology Of Self-Management In Organizations* by Richard Hackman



Traditional vs. Self-Organized Teams

Traditional Teams	Self-Organized Teams
Individual focuses on their responsibilities	Individual is asked to focus on end-to-end process
Individual is accountable only for goal achievement	Individual is asked to be transparent, adaptable
Manager controls daily tasks	Team controls daily activities as a group
Management style is direct and authoritative	Manager acts as a coach/facilitator
Leadership is top down	Leadership is shared by the team
Many narrow jobs performed by individuals	Process is subdivided into tasks, team completes the whole task



DevOps Structures & Teaming

“

Organizations which design systems are constrained to produce designs which are copies of the communication structures of these organizations.

- **Conway's Law**

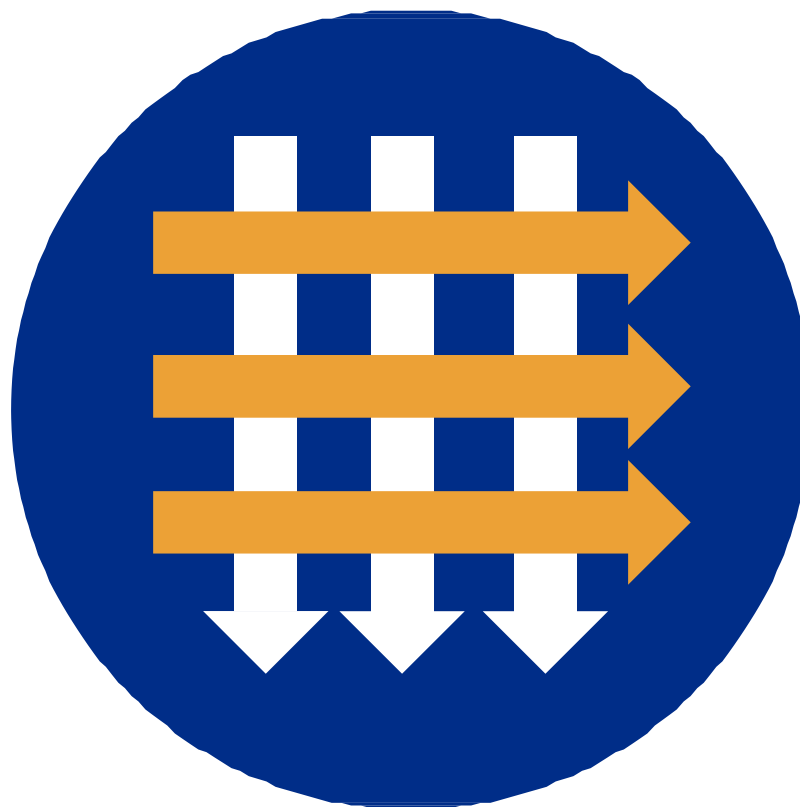
”



Traditional Organizational Models



Vertical



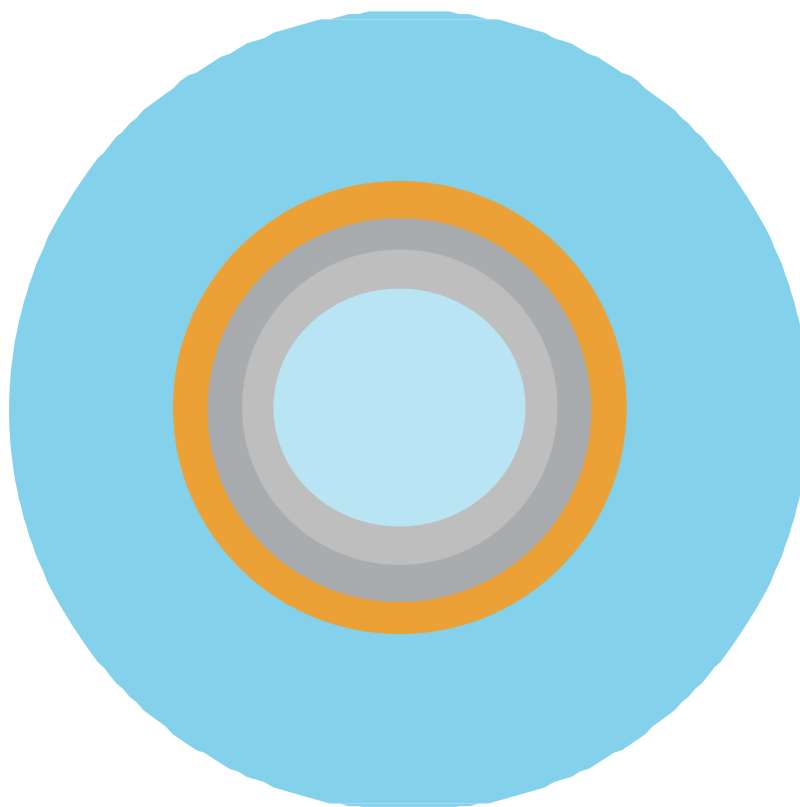
Matrix



Organizational Models For DevOps



Product



Adaptive



The ideal organizational structure for implementing **DevOps** depends upon many variables.

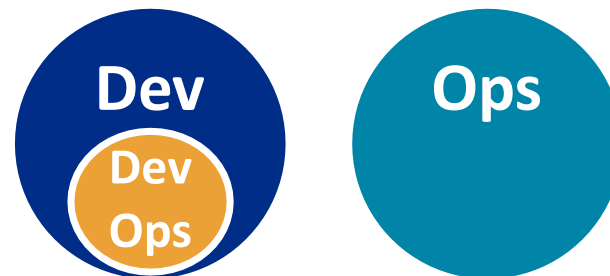


DevOps Structures That Don't Work

DevOps Team Silo:



Dev Doesn't Need Ops:



DevOps As a Tools Team:



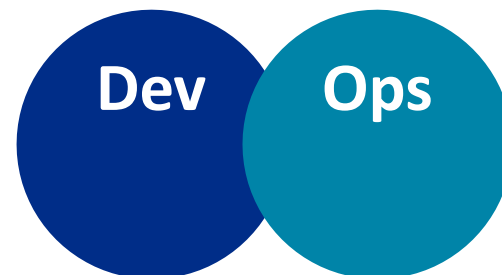
DevOps = DevOps Engineer:



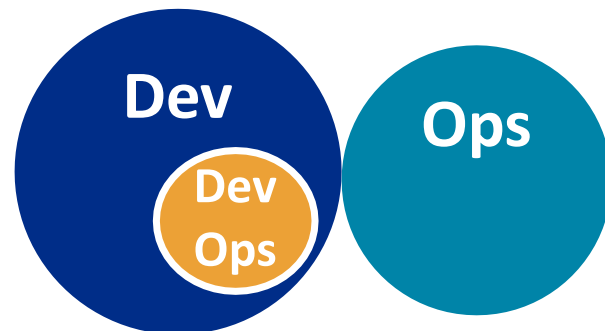


Ideal Structures For DevOps

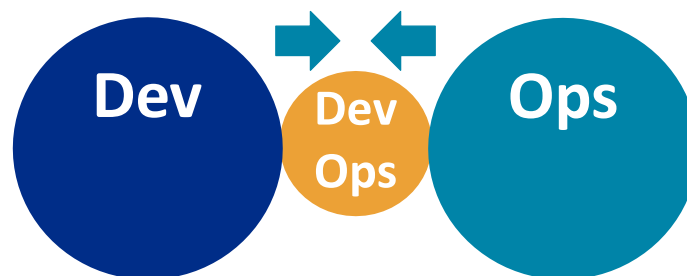
Dev & Ops Collaboration:



Ops As Infrastructure
As A Service:

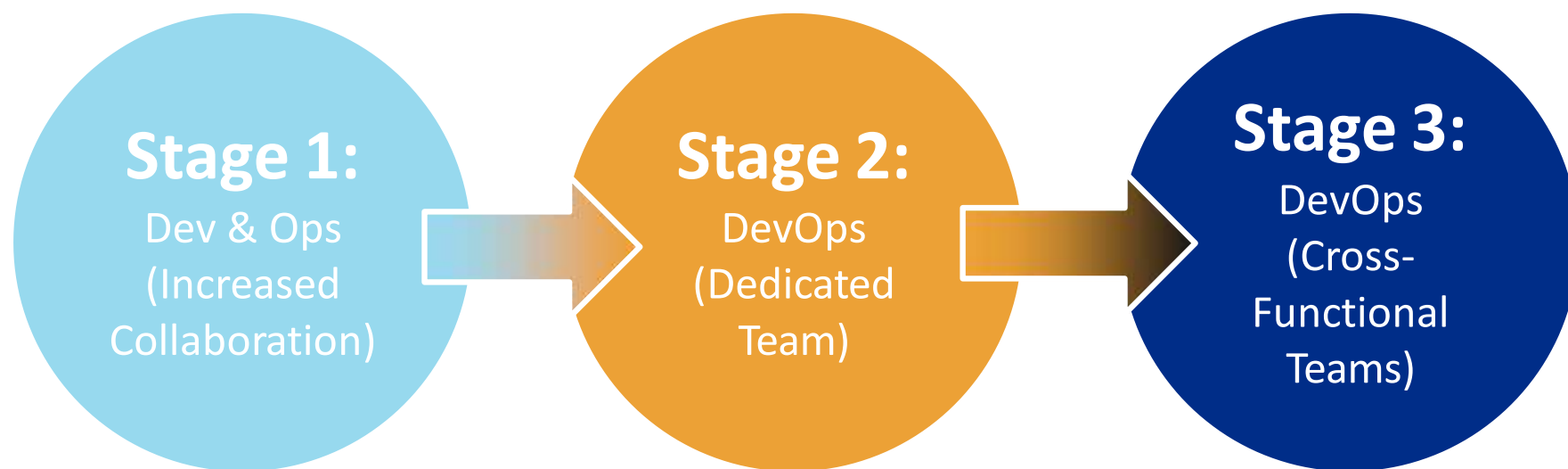


DevOps Team with Expiry:





The Evolution Of DevOps Teams





Getting Away From Task Orientation

Discourages creativity and encourages focus on task execution only

Jobs become tedious and boring

Individuals become entrenched in one specialization

Leaders are encouraged to micromanage and hoard information

Micromanagement leads to lack of focus on relationships and innovation

Promotes an attitude of completing your task and not caring about anything else

Promotes a culture of blaming



Building A Cross-Functional Team



**Include all
needed
disciplines**

**Balance
technical skill
levels**

**Balance
domain
knowledge**

Seek diversity

**Consider
persistence**

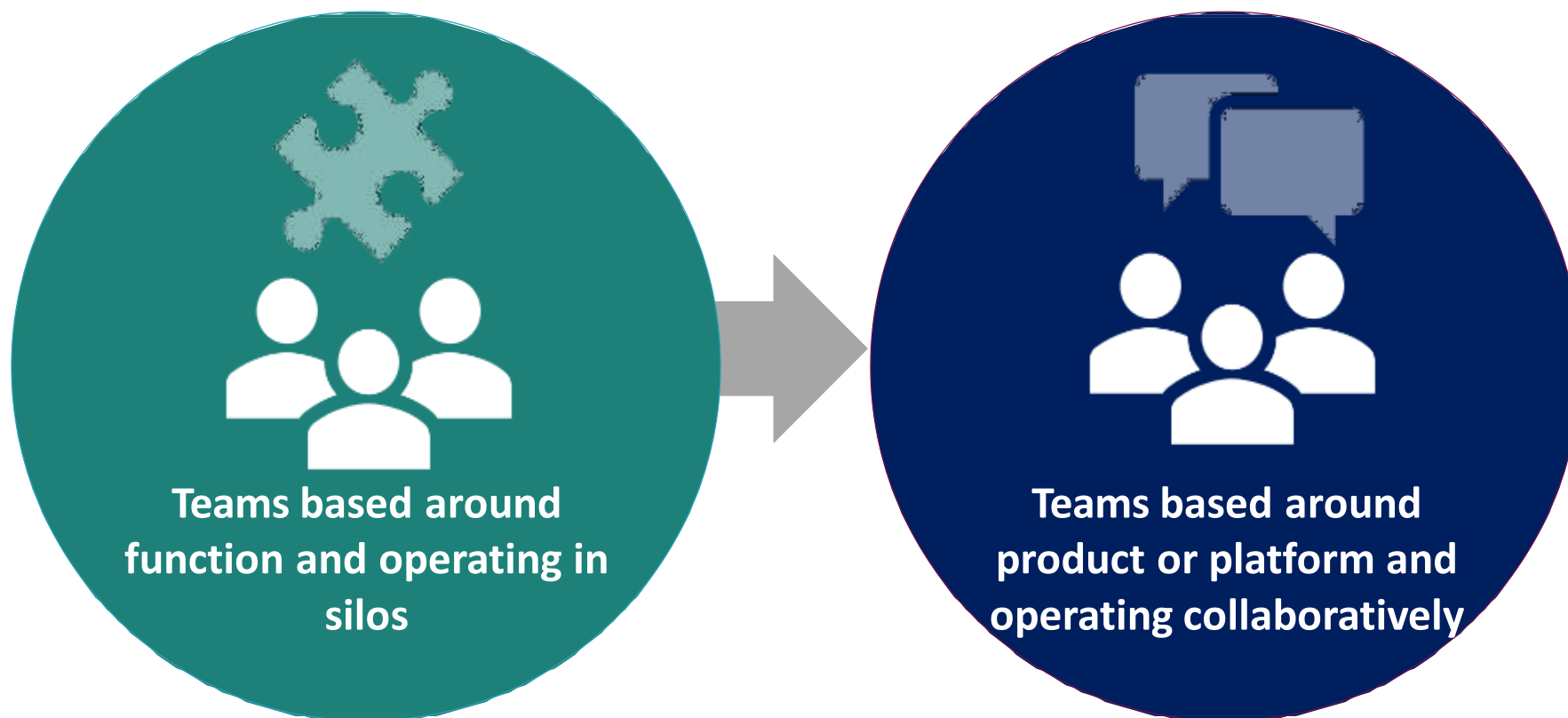


DevOps Requires Practical Specialists



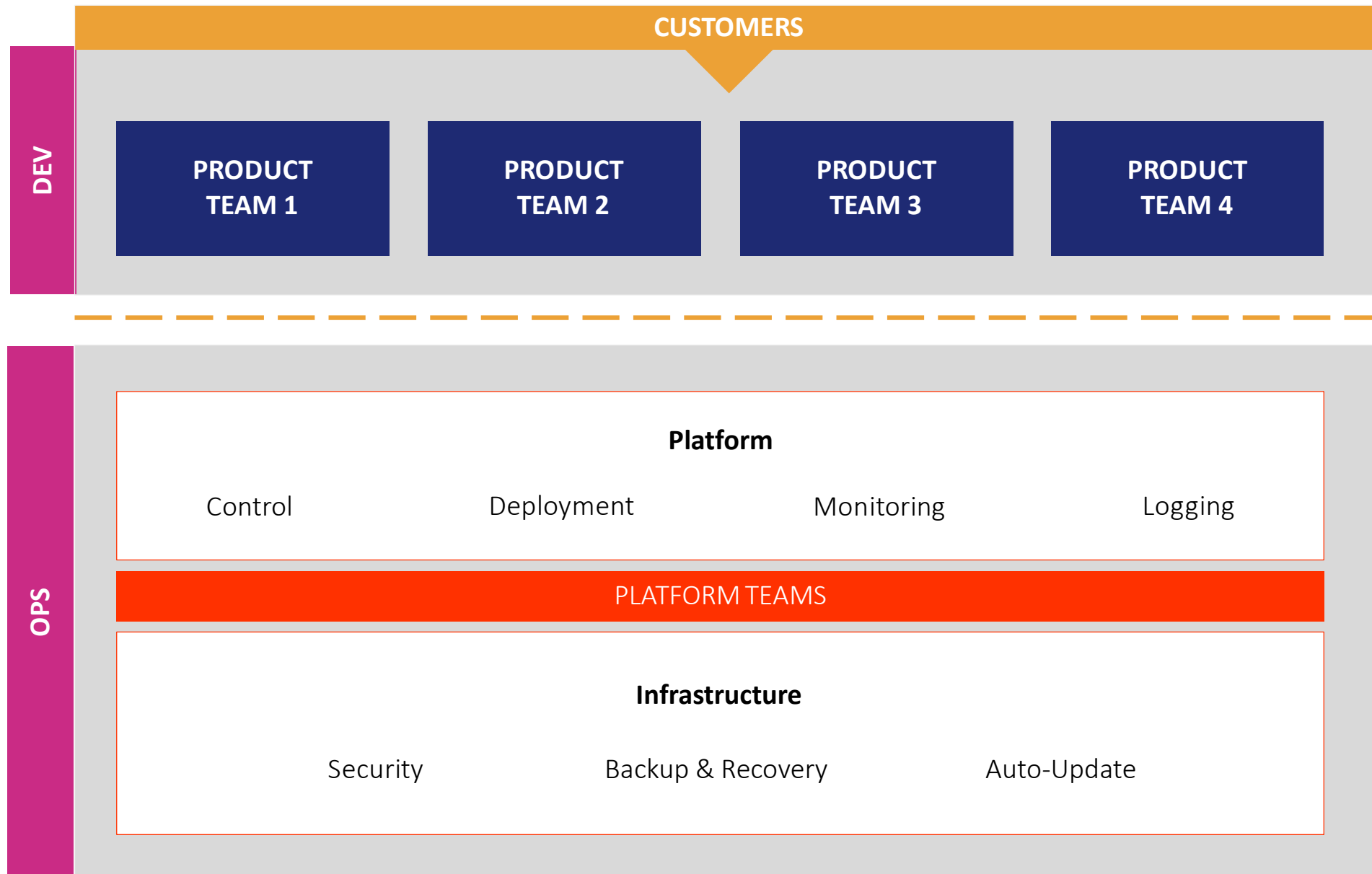


A Shift To Product & Platform Teams





Product & Platform Teams





PeopleCert DevOps Fundamentals

MODULE 4:

The Full Stack: Processes & Practices

- **The Evolution Of DevOps Practices**
- **The 15 Essential Practices Of DevOps**



The Evolution Of DevOps Practices



DevOps and its resulting technical, architectural and cultural practices represent a convergence of many philosophical and management movements.

Gene Kim, Jez Humble, Patrick Debois, John Willis, & John Allspaw
The DevOps Handbook



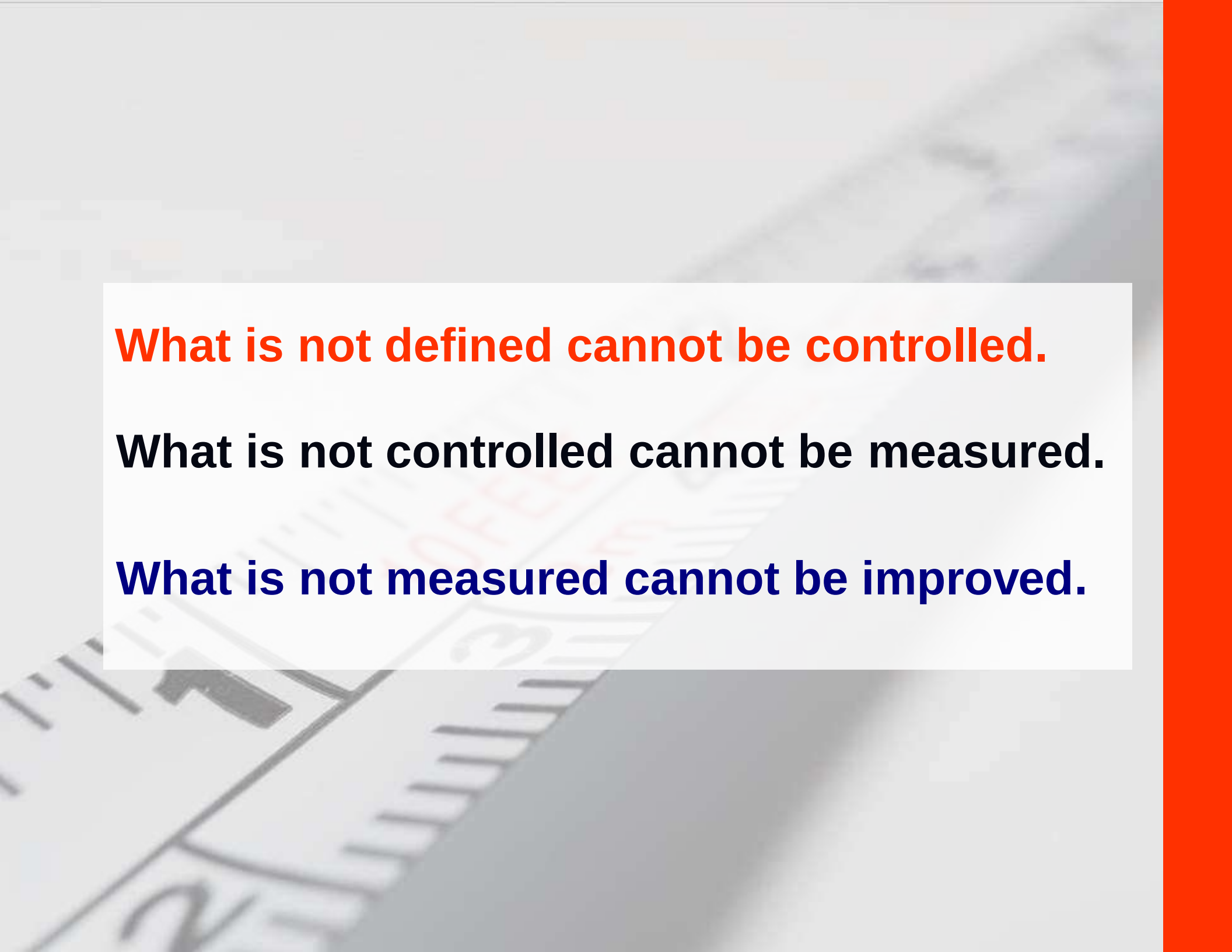
Service Management is a set of specialized **organizational capabilities for enabling **value** for customers in the form of **services**.**

“

It is my firm belief that ITSM and the DevOps Movement are not at odds. Quite to the contrary, they're a perfect cultural match.

- Gene Kim

”

The background of the slide features a close-up, slightly blurred image of a ruler and a pencil. The ruler is positioned diagonally, with markings in millimeters and centimeters visible. A pencil lies parallel to the ruler. The overall color palette is muted, with greys, whites, and a hint of the pencil's wood and lead.

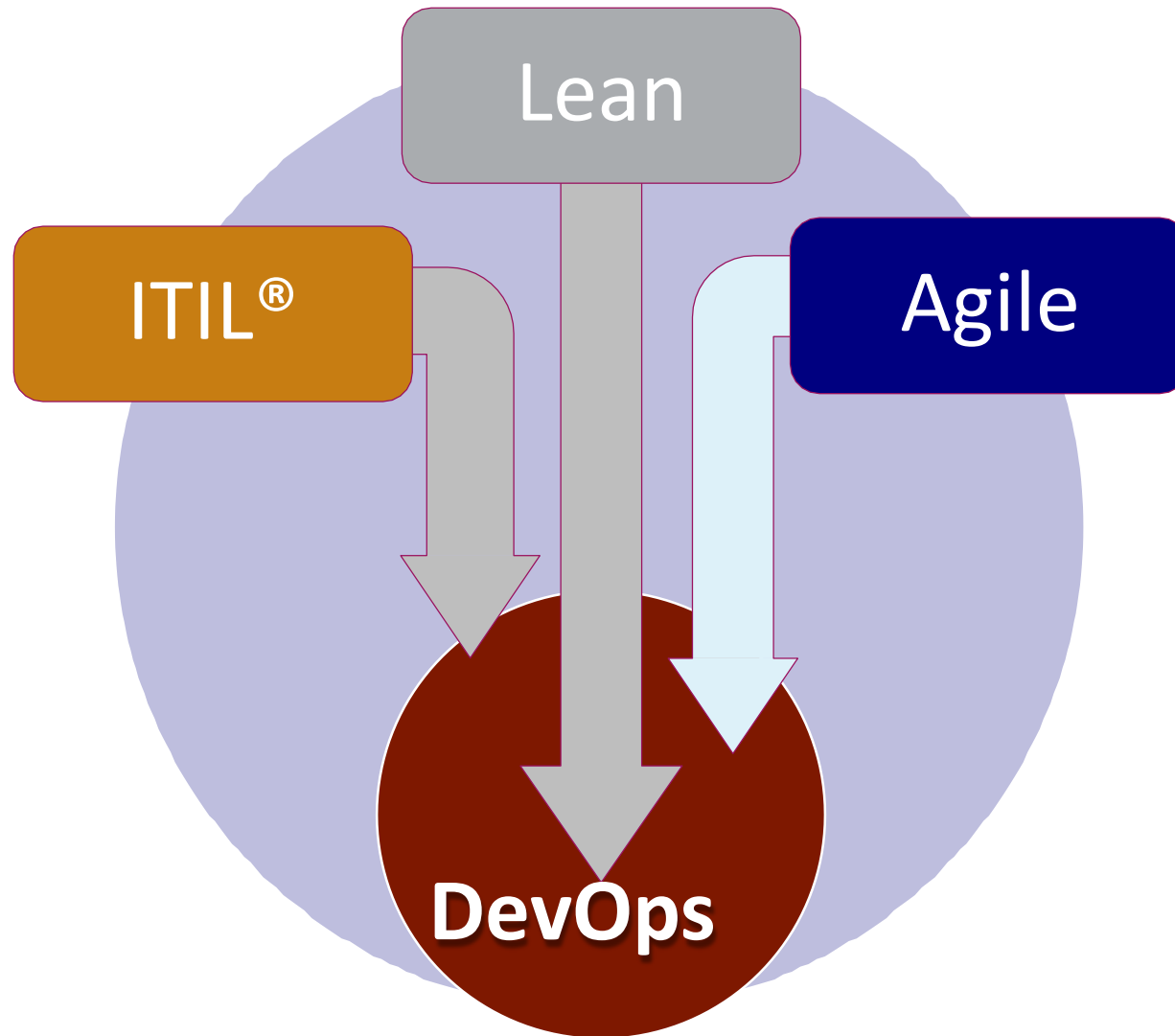
What is not defined cannot be controlled.

What is not controlled cannot be measured.

What is not measured cannot be improved.

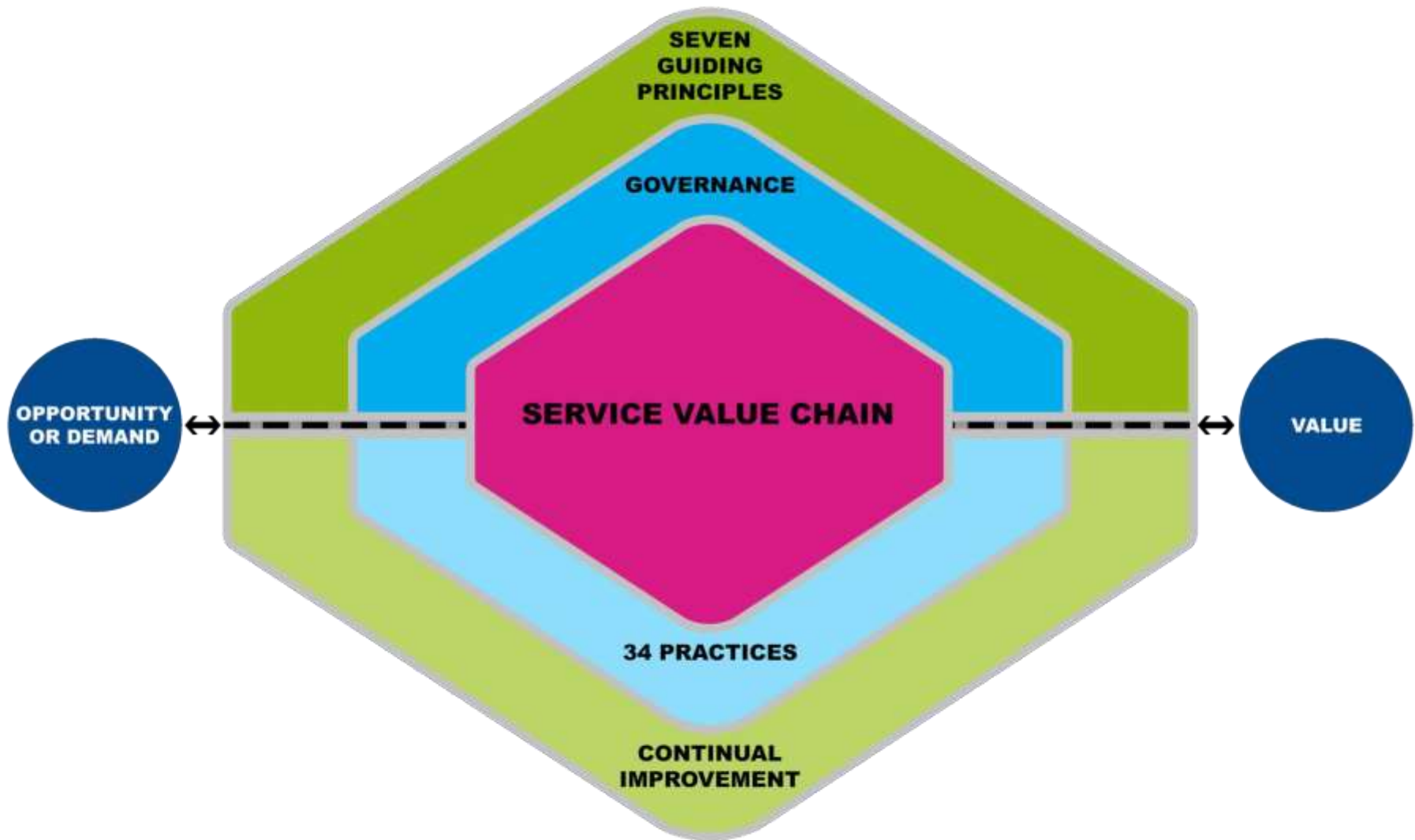


DevOps Leverages Best Practices



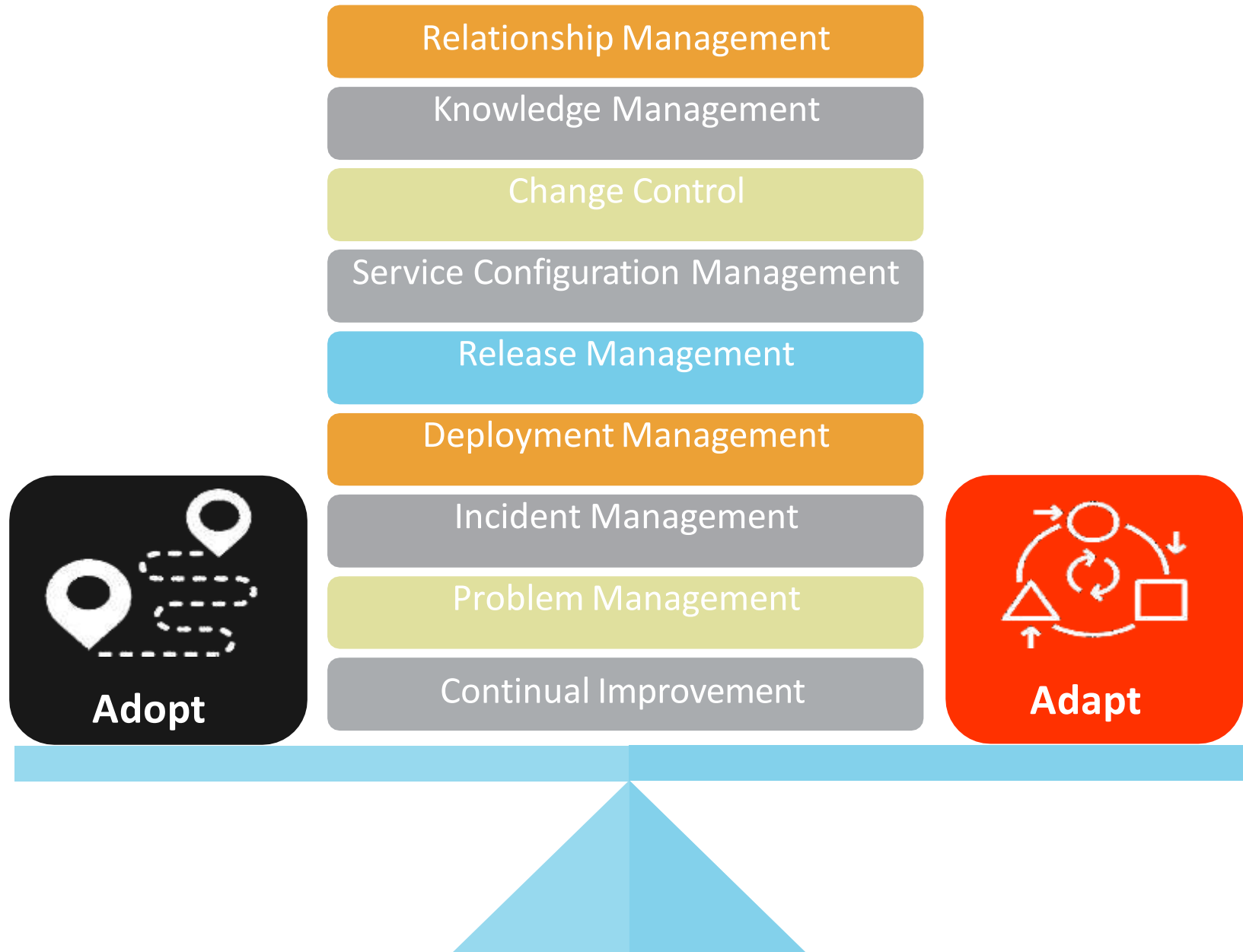


The ITIL Service Value System





Adopting & Adapting ITIL Practices for DevOps





Principles Of Lean Thinking



1. Define Value



2. Map The
Value Stream



3. Create Flow



4. Initiate Pull



5. Pursue
Perfection

Lean is not about doing more *with* less. It is about doing more *by doing* less. Those are different things.



The Agile Manifesto



Individuals &
Interactions



Working
Software



Customer
Collaboration



Responding To
Change

VS.



Processes &
Tools



Comprehensive
Documentation



Contract
Negotiation



Following A Plan



Agile Principles

Satisfy customers
through early and
continuous
software delivery

Accommodate
changing
requirements
throughout the
development
process

Deliver working
software
frequently

Enable
collaboration
between the
business
stakeholders and
developers
throughout the
project

Support, trust and
motivate the
people involved

Enable face-to-
face interactions

Measure progress
as working
software

Support a
sustainable
development pace
with Agile
processes

Ensure that
attention to
technical detail
and design
enhances agility

Focus on
simplicity

Encourage
architectures,
requirements and
designs with self-
managing teams

Reflect regularly
on how to become
more effective



DevOps Speeds Things Up



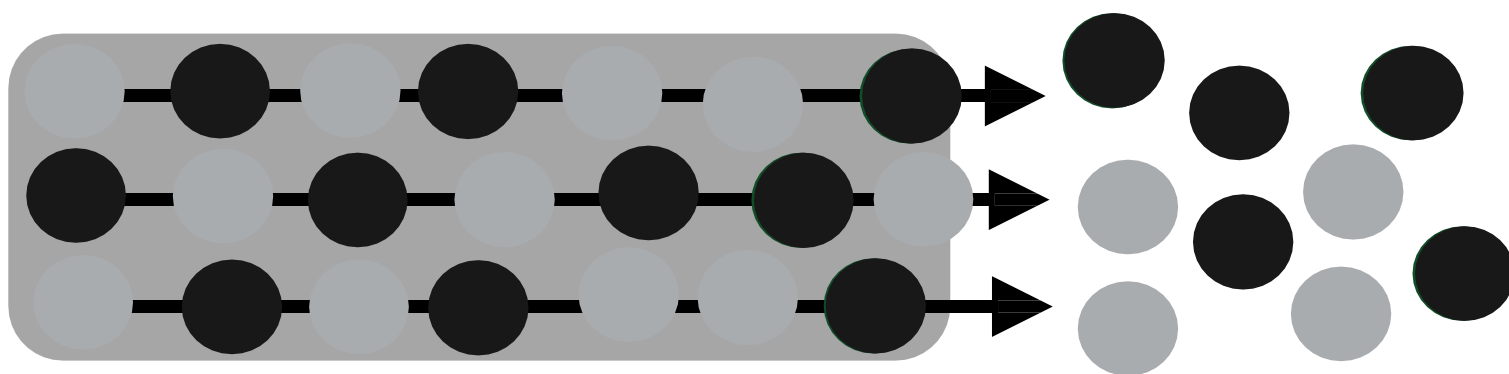
Internal Stakeholders



Dev



Ops





The 15 Essential Practices Of DevOps



Some, eager to start on the DevOps path, begin by researching which tools they should buy. More important than specific tools, though, are the technical practices that enable you to achieve the very things that most people turn to DevOps for.

Puppet Labs

2017 State of DevOps Report





The 15 Essential Practices Of DevOps

#1

**Voice Of The
Customer**

#2

**Relationship
Management**

#3

**Lean Process
Optimization**

#4

**Value Stream
Mapping**

#5

**Knowledge
Management**

#6

**Visual
Management**

#7

Agile Scrum

#8

**Shift Left
Testing**

#9

**Change
Control**

#10

**Service
Configuration
Management**

#11

**Release &
Deployment
Management**

#12

**Incident
Management**

#13

**Problem
Management
& Kaizen**

#14

**Continual
Improvement**

#15

Antifragility

The 15 Essential Practices
are critical to the **Three
Ways** and CALMS, and to
enabling **Continuous
Delivery and Deployment.**



**Align yourself around business value by defining
Voice Of The Customer**



$$\text{Utility} + \text{Warranty} = \text{Value}$$



Utility

- **Functional** requirements
- Fit for purpose
- What the service does

VALUE

VALUE

Warranty

- **Non-functional** requirements
- Fit for use
- How the service is delivered





Lean Voice Of The Customer (VOC)



**Who is the
customer?**



**What does the
customer want?**



**What adds value to
the customer?**



**What is the customer
willing to pay for?**



Understanding Critical To Quality

**“MUST HAVES” =
CRITICAL TO QUALITY**

**What does the customer
need? What is value?**

VS.

“NICE TO HAVES”

**What does the customer
want? What adds value?**

**Defining VOC should be at the
core of how you understand
Business Value.**



**Increase collaboration and break down silos with
Relationship Management**



Relationship Management

Relationship Management seeks to establish and maintain the relationship between IT and internal business stakeholders, identify business value and ensure that IT understands and is able to deliver it.



**Business
Relationship
Manager
(BRM)**

The person who really makes the Relationship Management practice effective by acting as a liaison between IT and internal business partners and stakeholders

A photograph of a man and a woman shaking hands over a table. The man is on the left, seen in profile, with a beard and wearing a dark suit. The woman is on the right, smiling, wearing a grey blazer over a white top and a gold necklace. They are shaking hands over a table that has a brown paper coffee cup, a teal cup on a saucer, and a small glass vase with dried flowers. The background is a bright window with a view of a city skyline.

The BRM enables collaboration by acting as a facilitator, making introductions and ensuring that everyone understands each other.



**Eliminate waste wherever possible and reduce handoffs
with Lean Process Optimization**

“

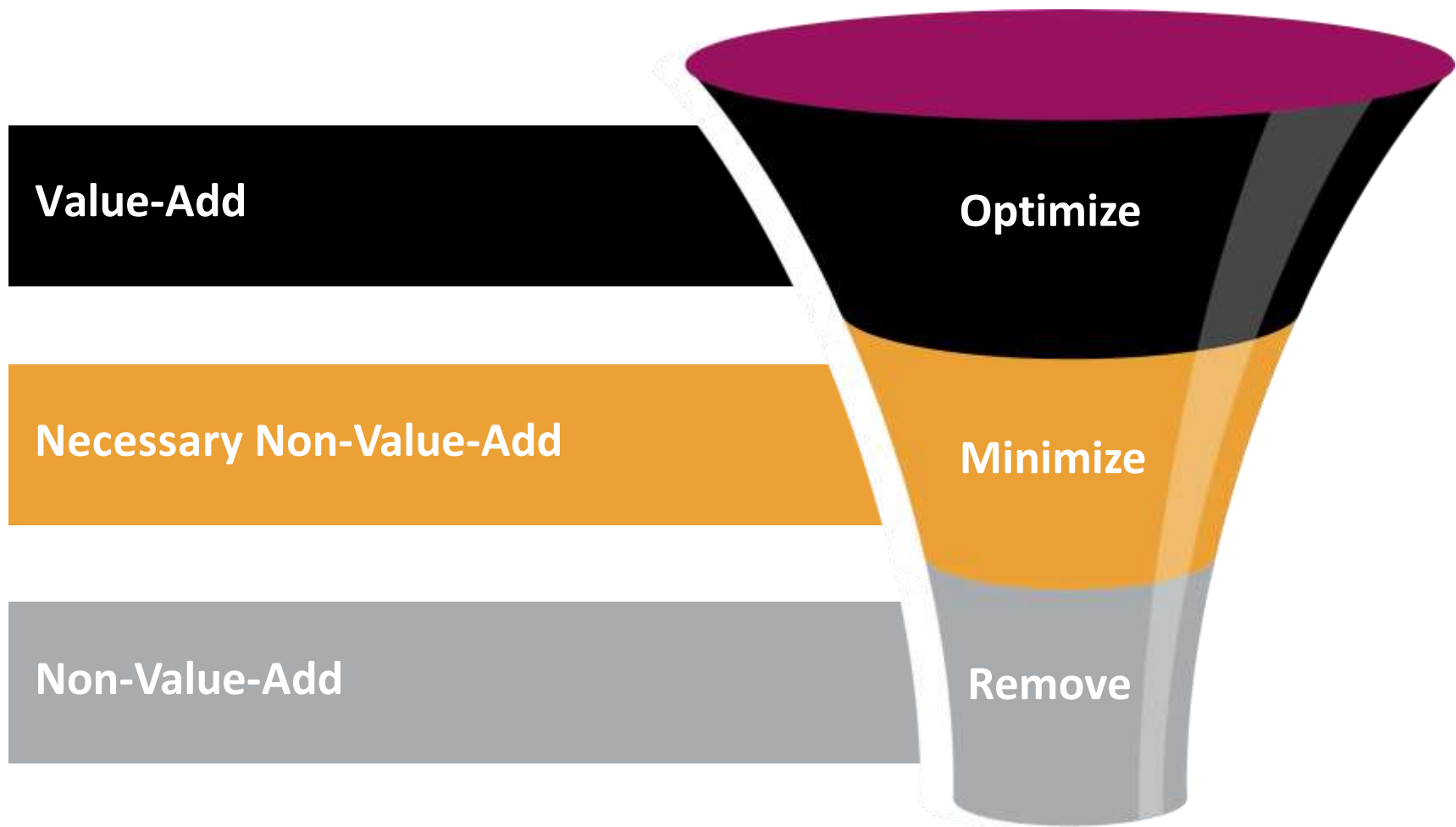
*The most dangerous kind of waste
is the waste we do not recognize.*

- Shigeo Shingo

”

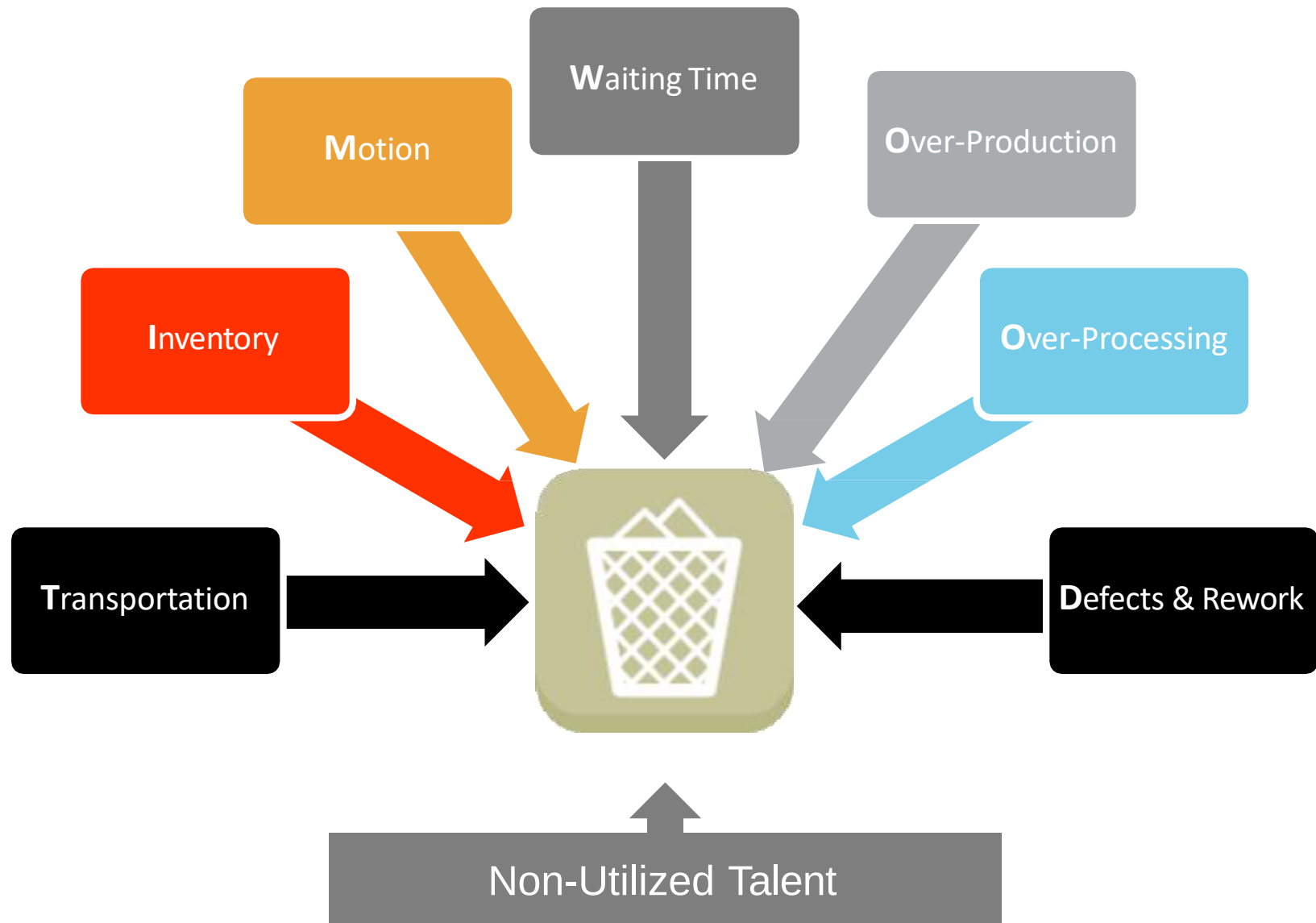


Lean Process Optimization





Lean Waste: “Muda”





Other Forms Of Lean Value Loss

Mura

Muri

Variances &
Variation

Overburden &
Inflexibility



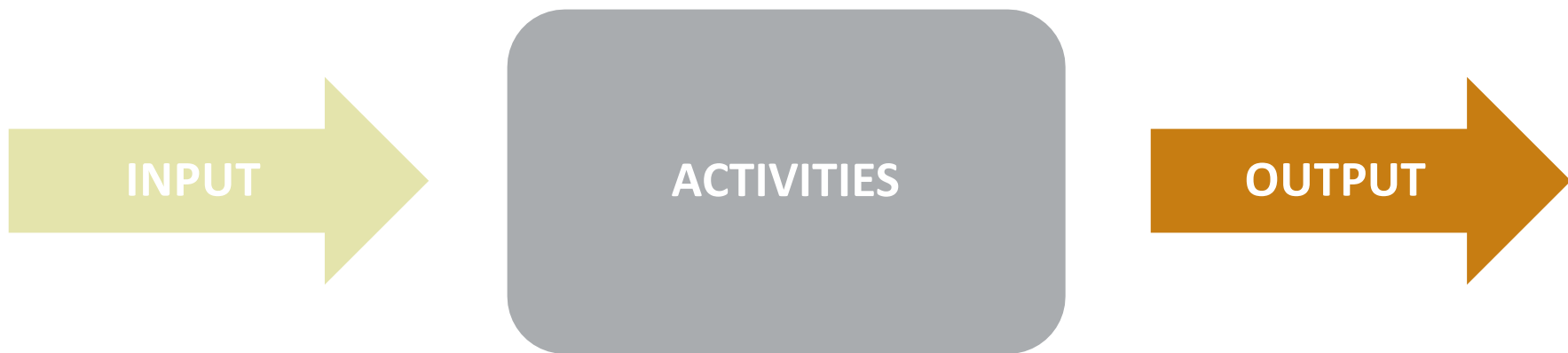


Identify constraints and bottlenecks with Value Stream Mapping



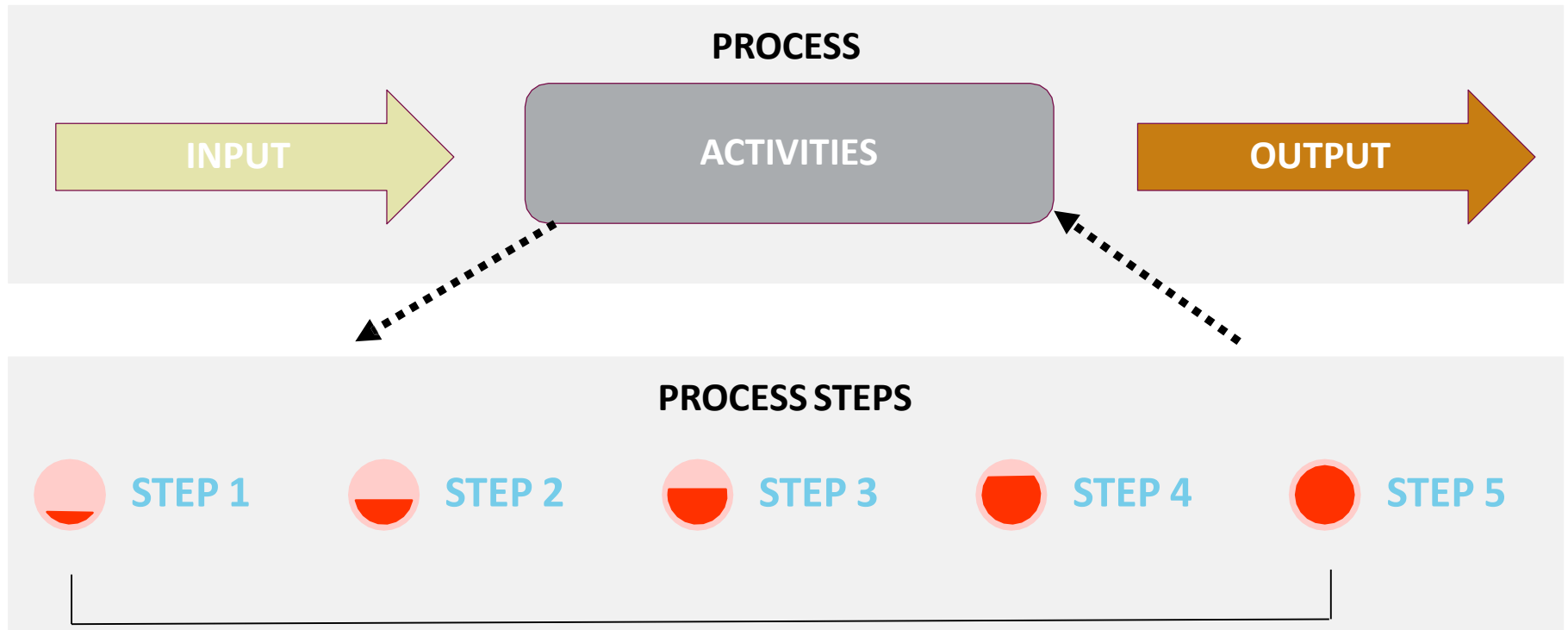
What Is A Process?

A **Process** is a structured set of activities designed to accomplish a specific objective. They take one or more defined inputs and turn them into defined outputs.





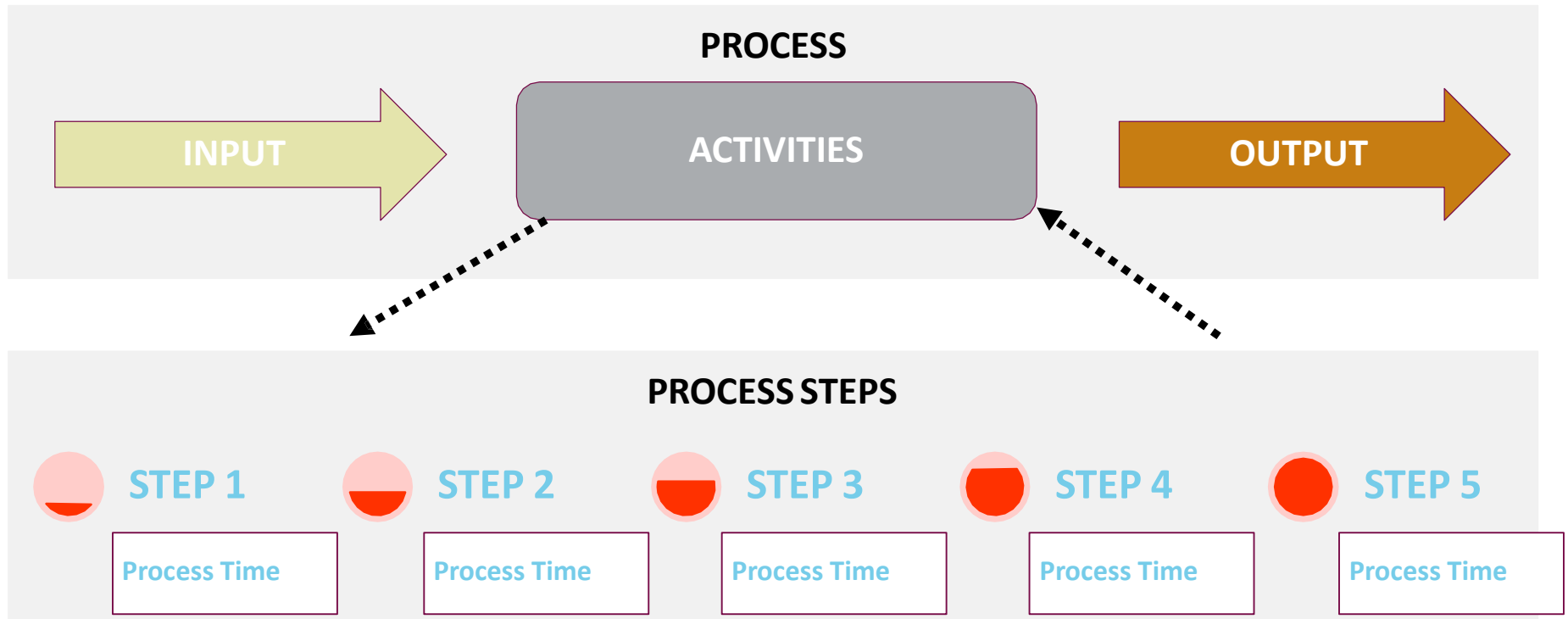
Value Stream Mapping



Total Cycle Time is the total time from the beginning to the end of your process



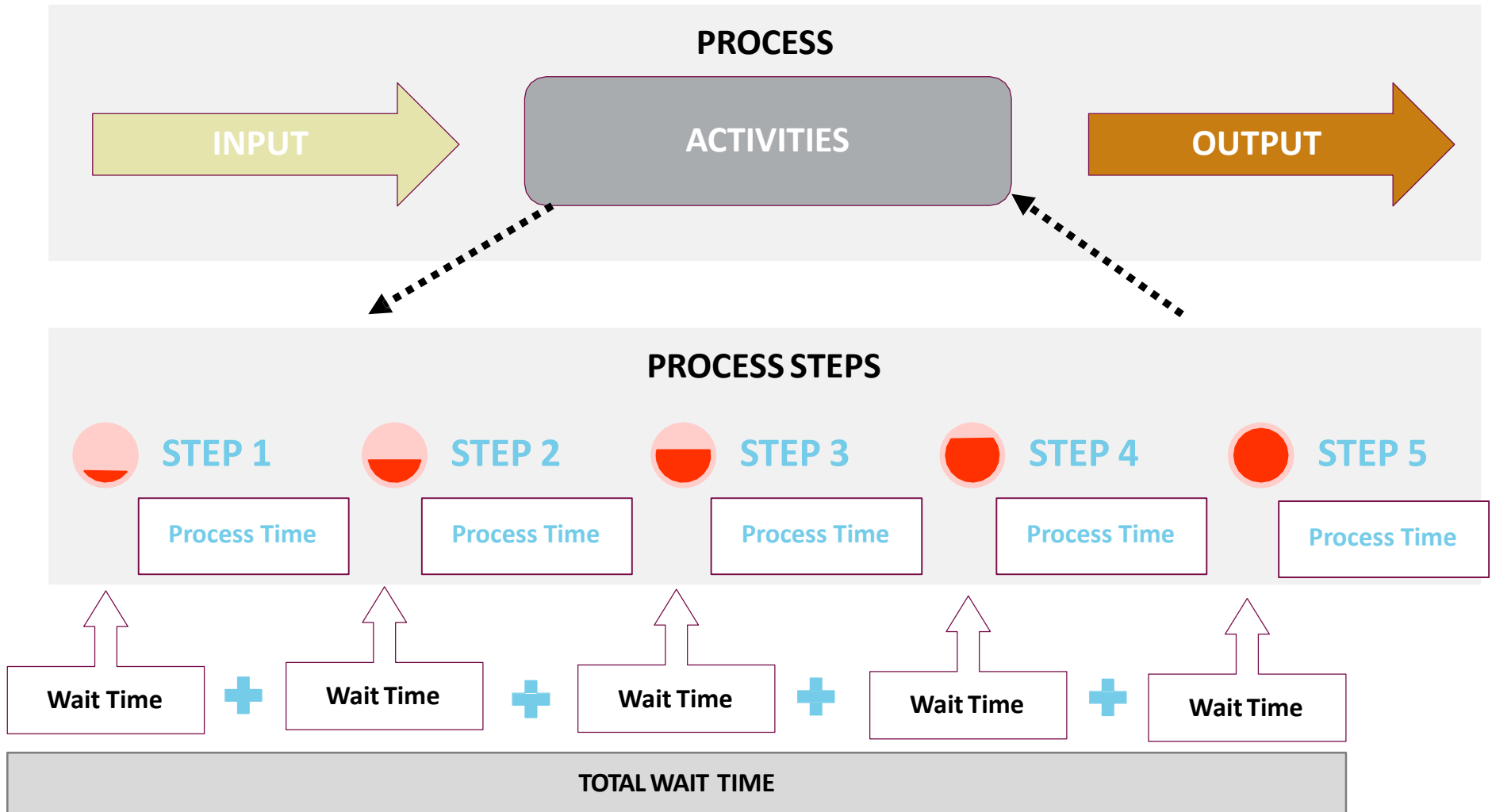
Value Stream Mapping



Process Time is the total time spent actually creating products or services.



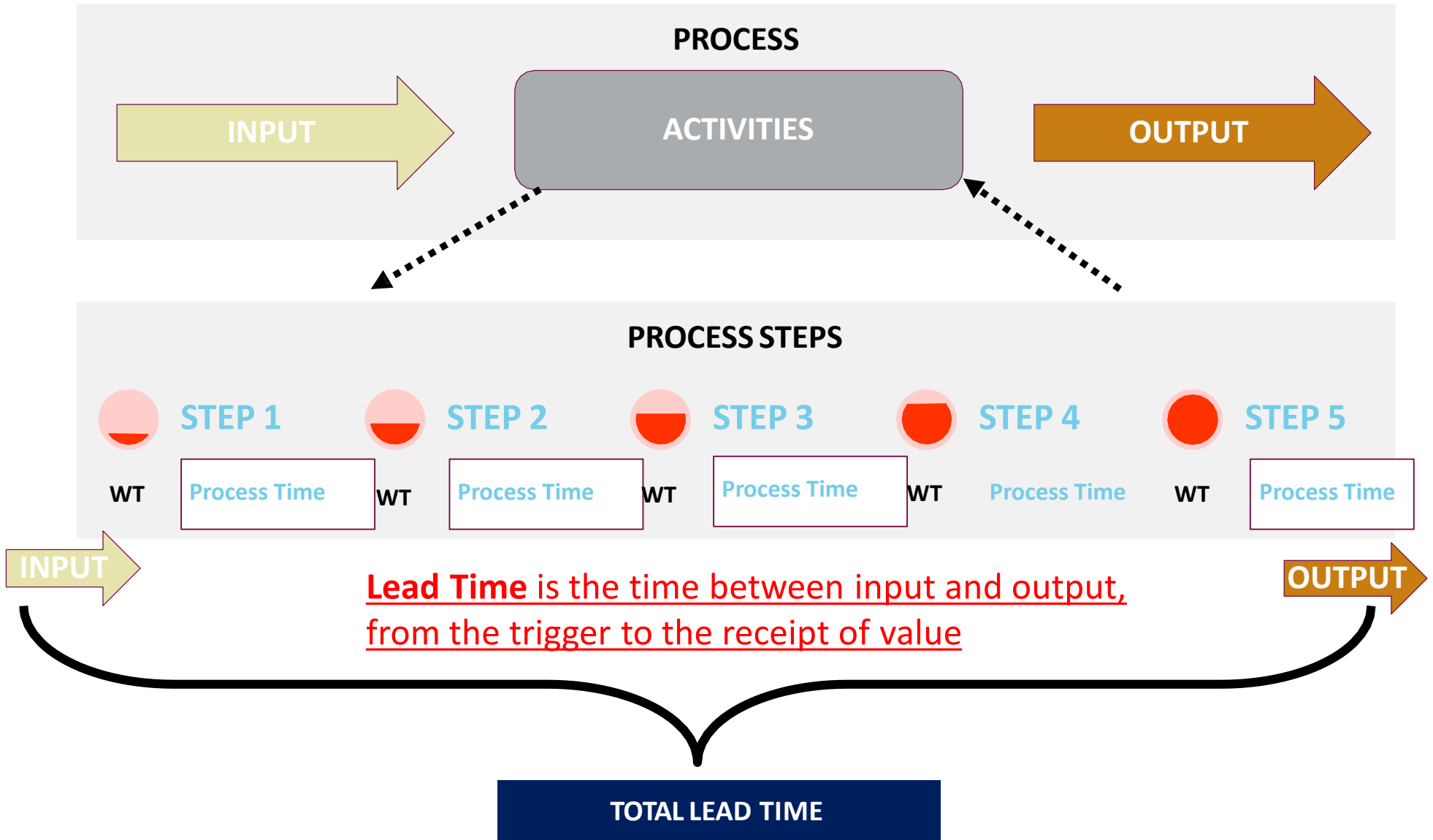
Value Stream Mapping



Wait Time is the time spent waiting for next steps to begin.



Value Stream Mapping



**Value Stream Mapping
provides valuable data based
on clear measurements that
you can use to optimize work.**

A large orange circle containing the white text "#5".

#5

Enable knowledge sharing and create a culture of learning with Knowledge Management



Knowledge Management

Knowledge Management is the practice of **embedding and making accessible the essential knowledge that is required** to support a DevOps environment and ensure stable, effective services.

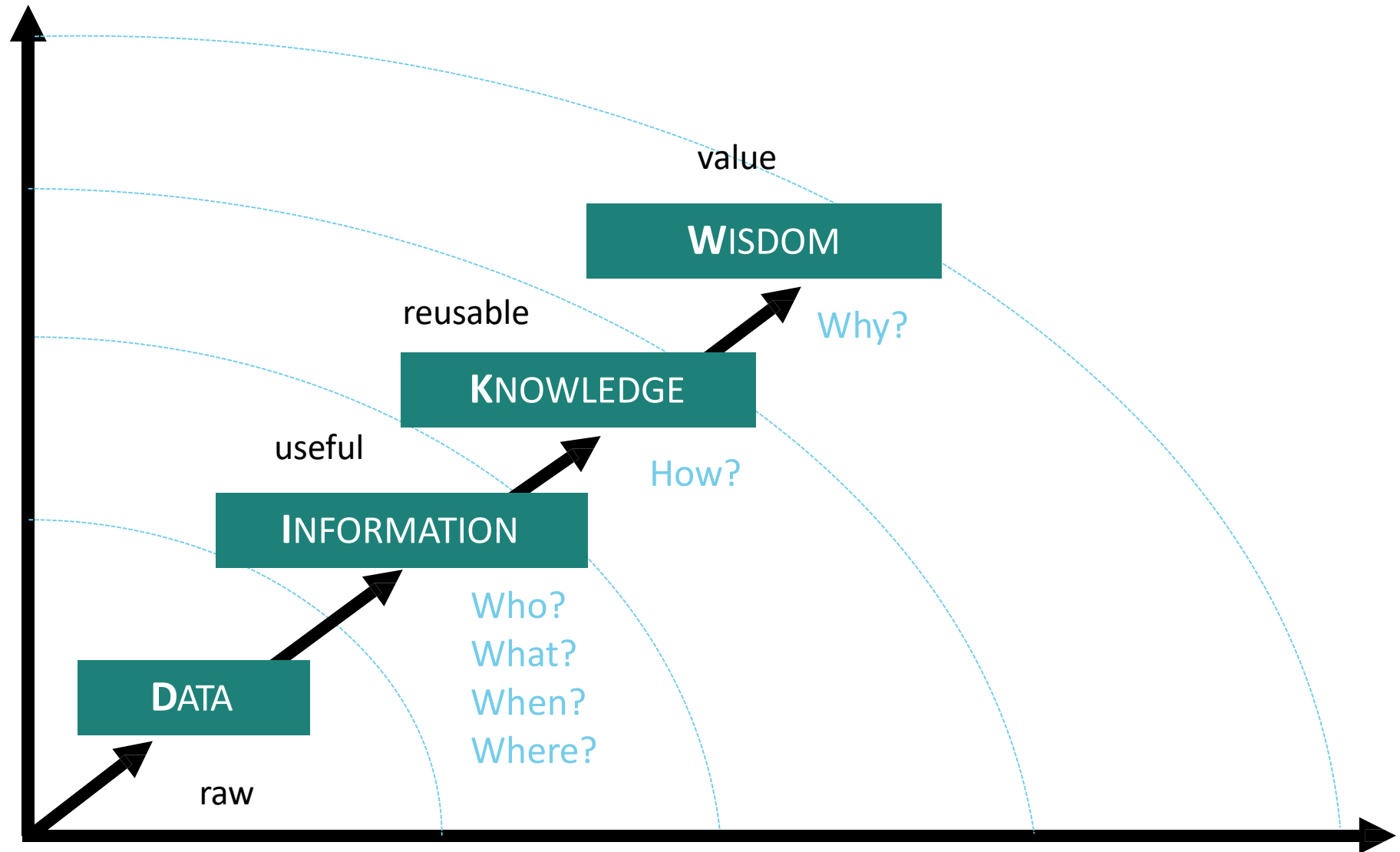
Share perspectives, ideas, experience and information

Ensure these are available in the right place, at the right time, to enable informed decisions

Improve efficiency by reducing the need to rediscover knowledge



The DIKW Model





Knowledge Management Systems

Collaborative software technologies, such as Slack or an internal wiki

Knowledge databases, such as a document managementsystem

Mentoring and job shadowing

Embedding diverse skillsets on cross-functional teams

Informal gatherings such as knowledge fairs to brainstorm ideas or teach others

Visual Management tools such as Kanban



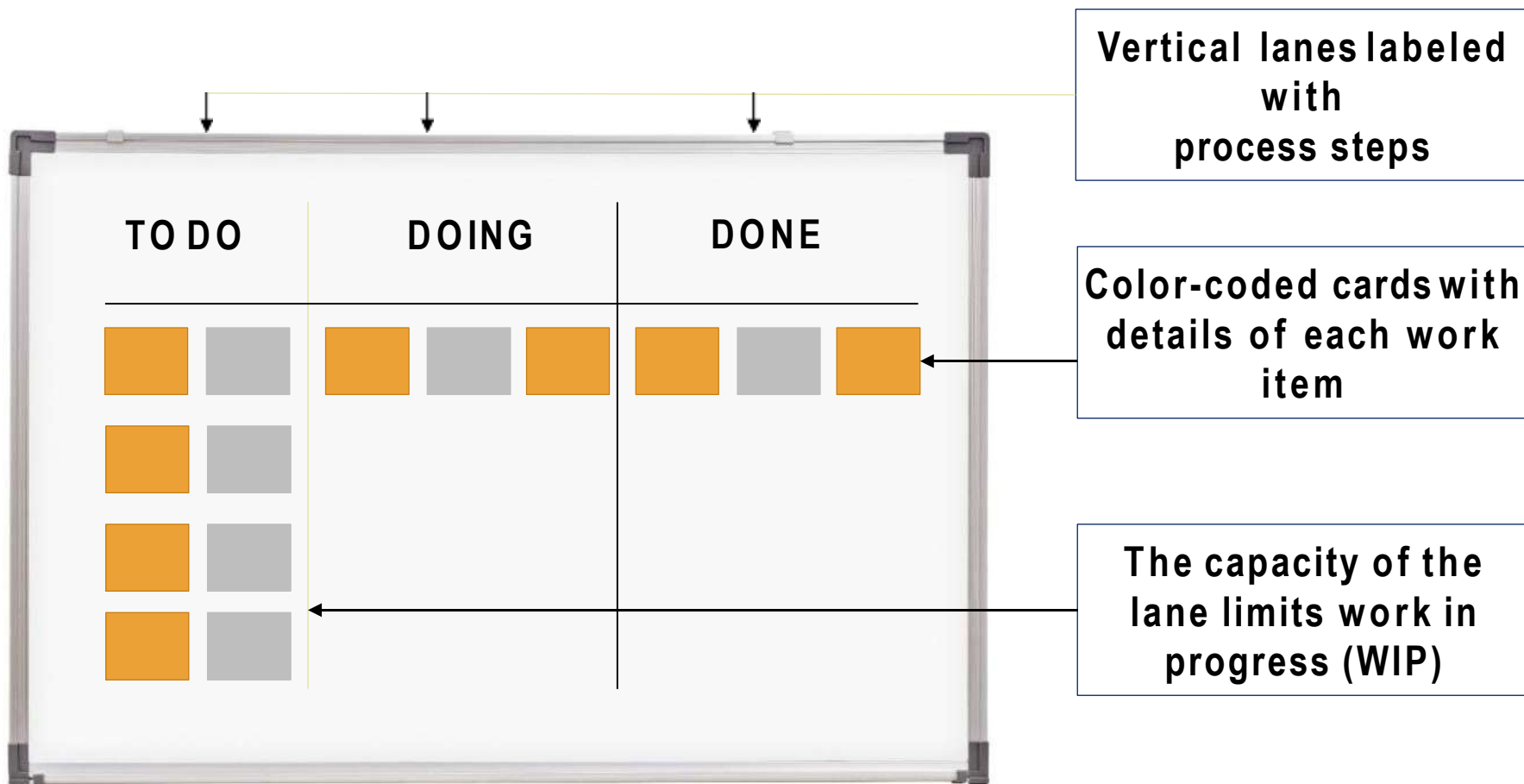
Create transparency in the value stream and make work visible with Visual Management



Visual Management involves using visual controls in plain sight rather than text-based ones, where it is easier to interpret, remember and keep at the forefront of one's mind.

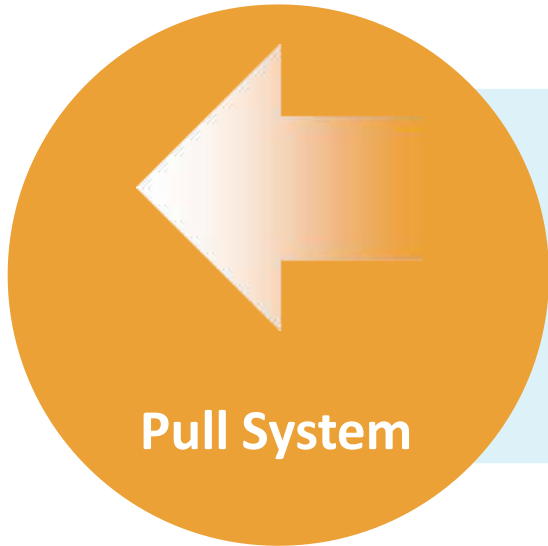


Utilizing Kanban

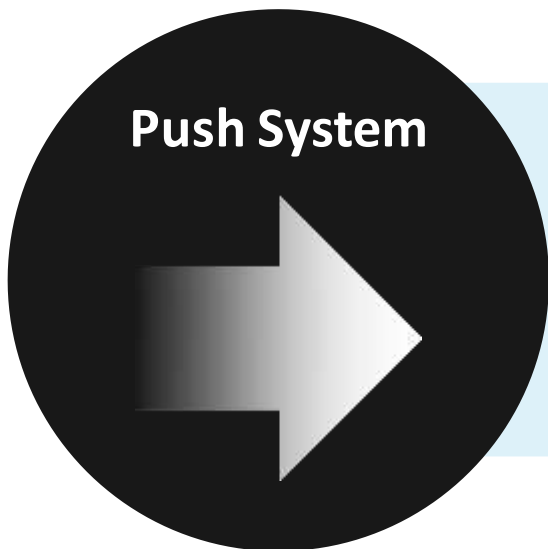




Push vs. Pull Systems



Products or services are pulled through the process by customer demand



Products or services are pushed through the process based on forecasted demand

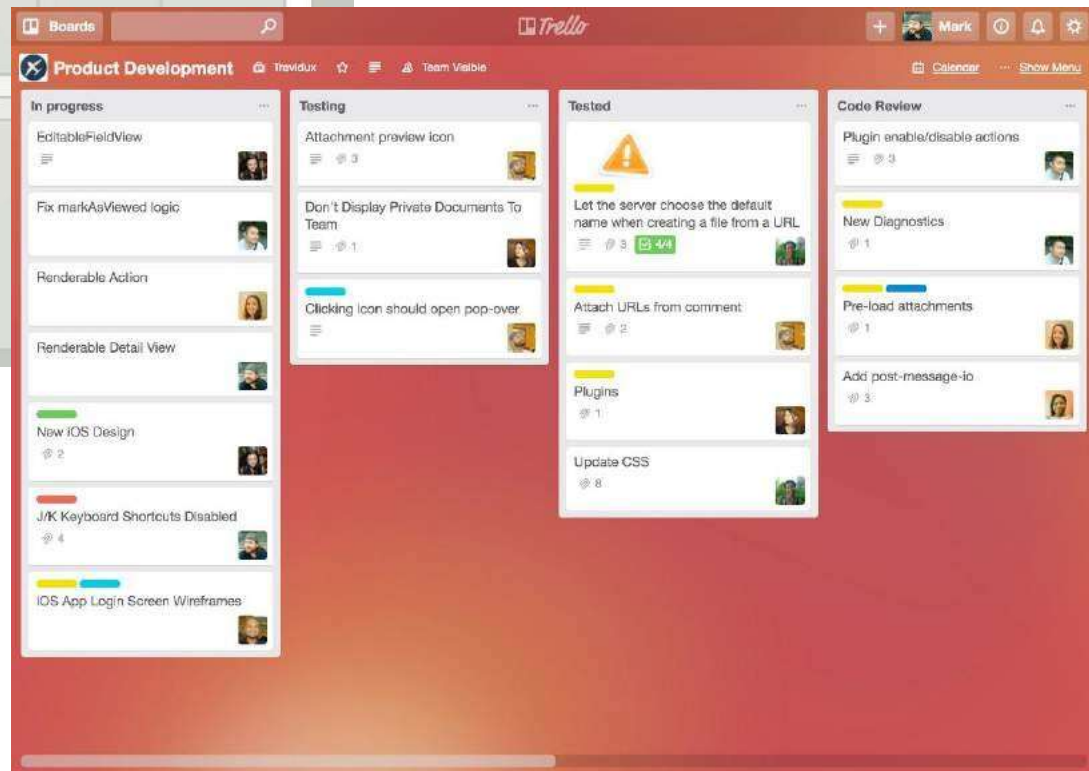
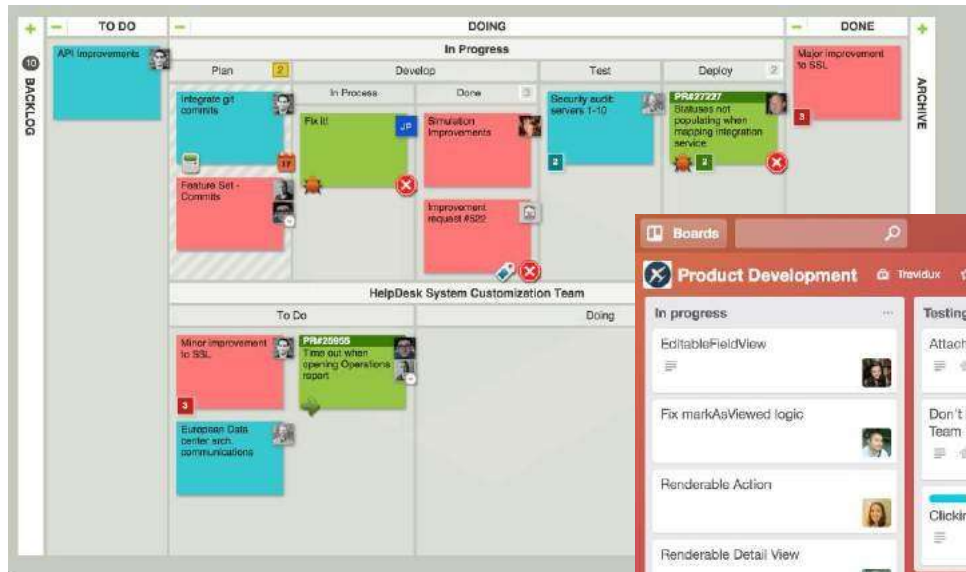


WIP Limits

Backlog	Selected (3)	Review/Fix (3)		Validate (4)		Merge (3)		Update Docs (3)		Done
		In Progress	Done	In Progress	Done	In Progress	Done	In Progress	Done	
    	  	 		  		  				   



Distributed Kanban Examples



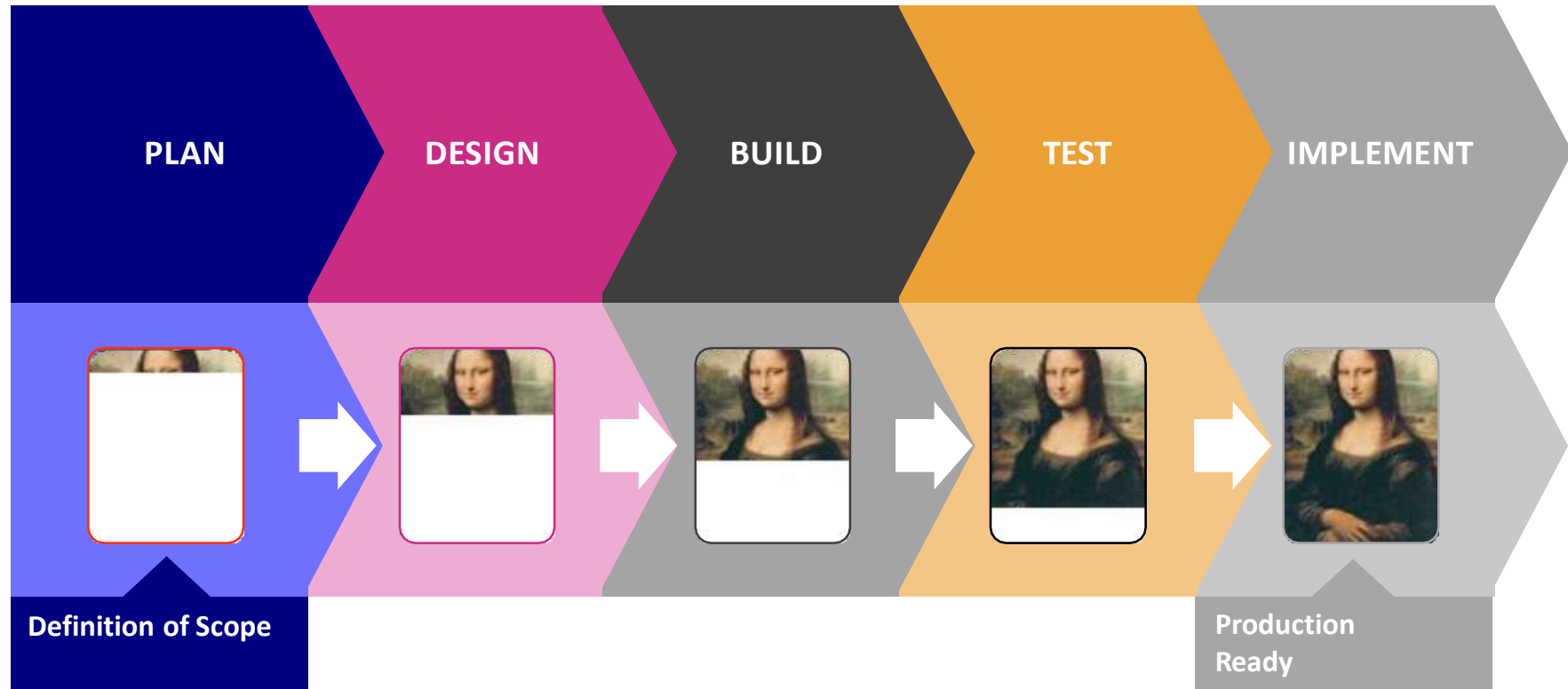
Sources: Leankit.com (left) and Trello.com (right)



**Work in smaller batches and create feedback loops with
Agile Scrum**

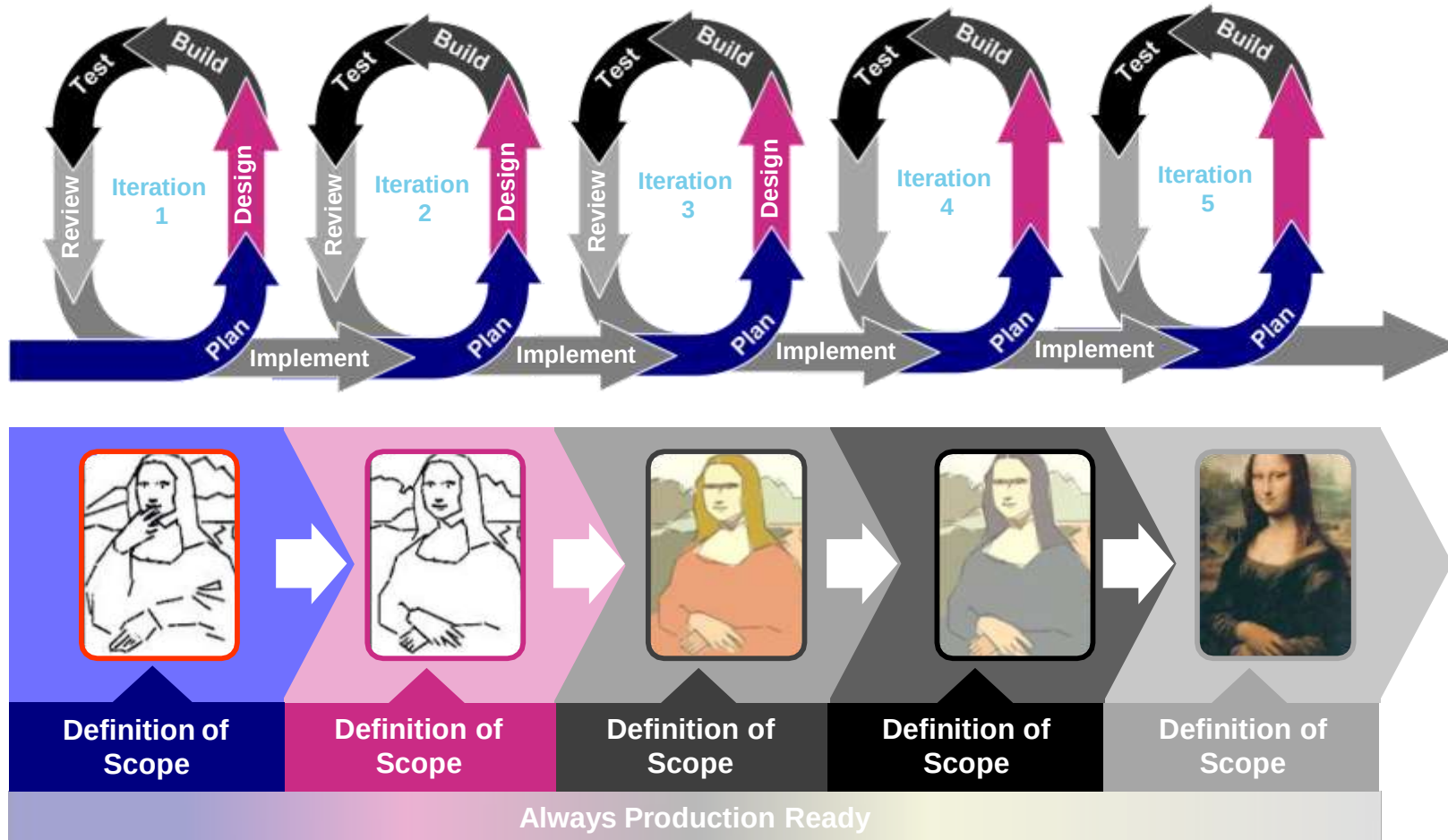


Traditional Waterfall Project Management



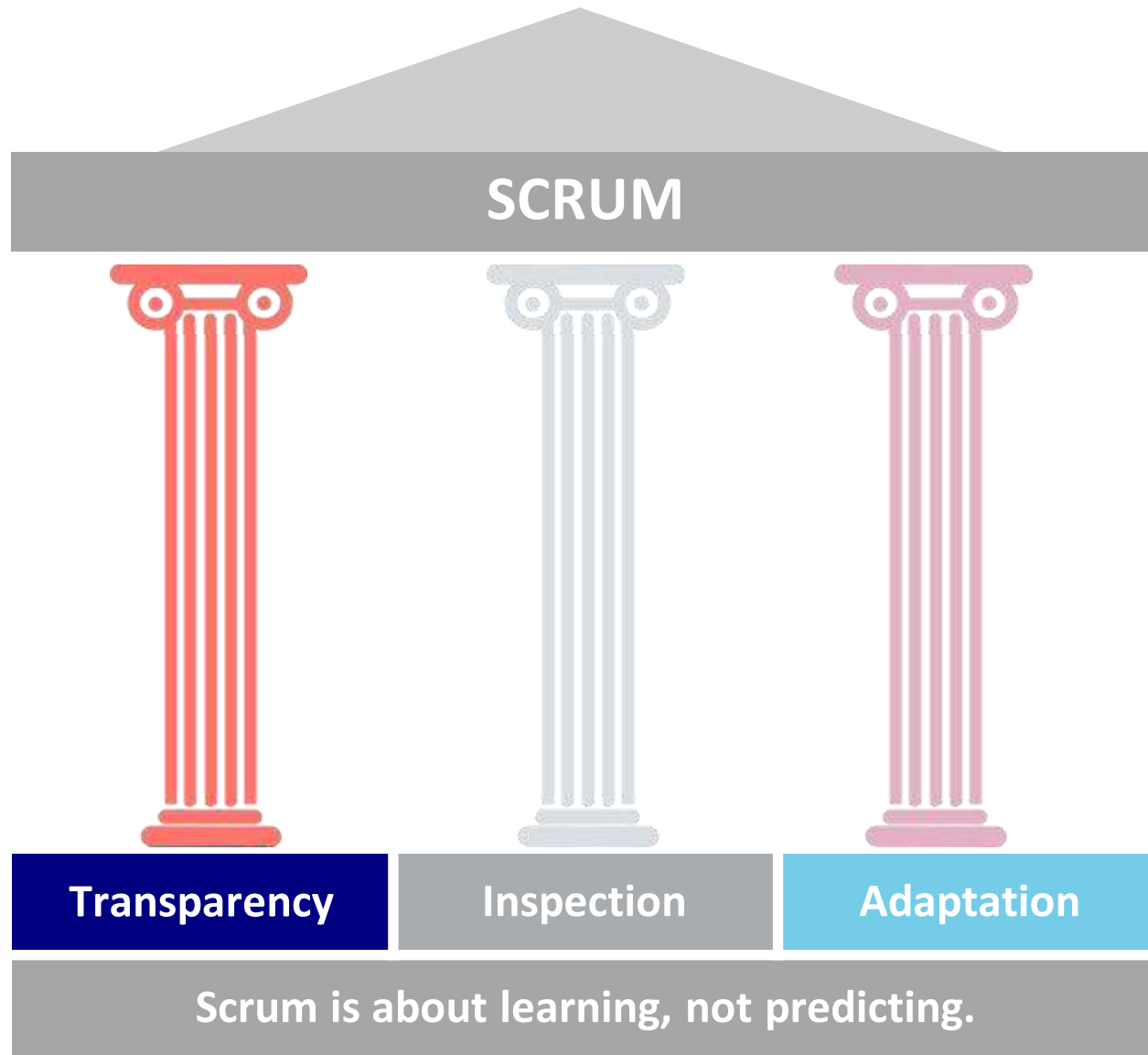


Agile Project Management



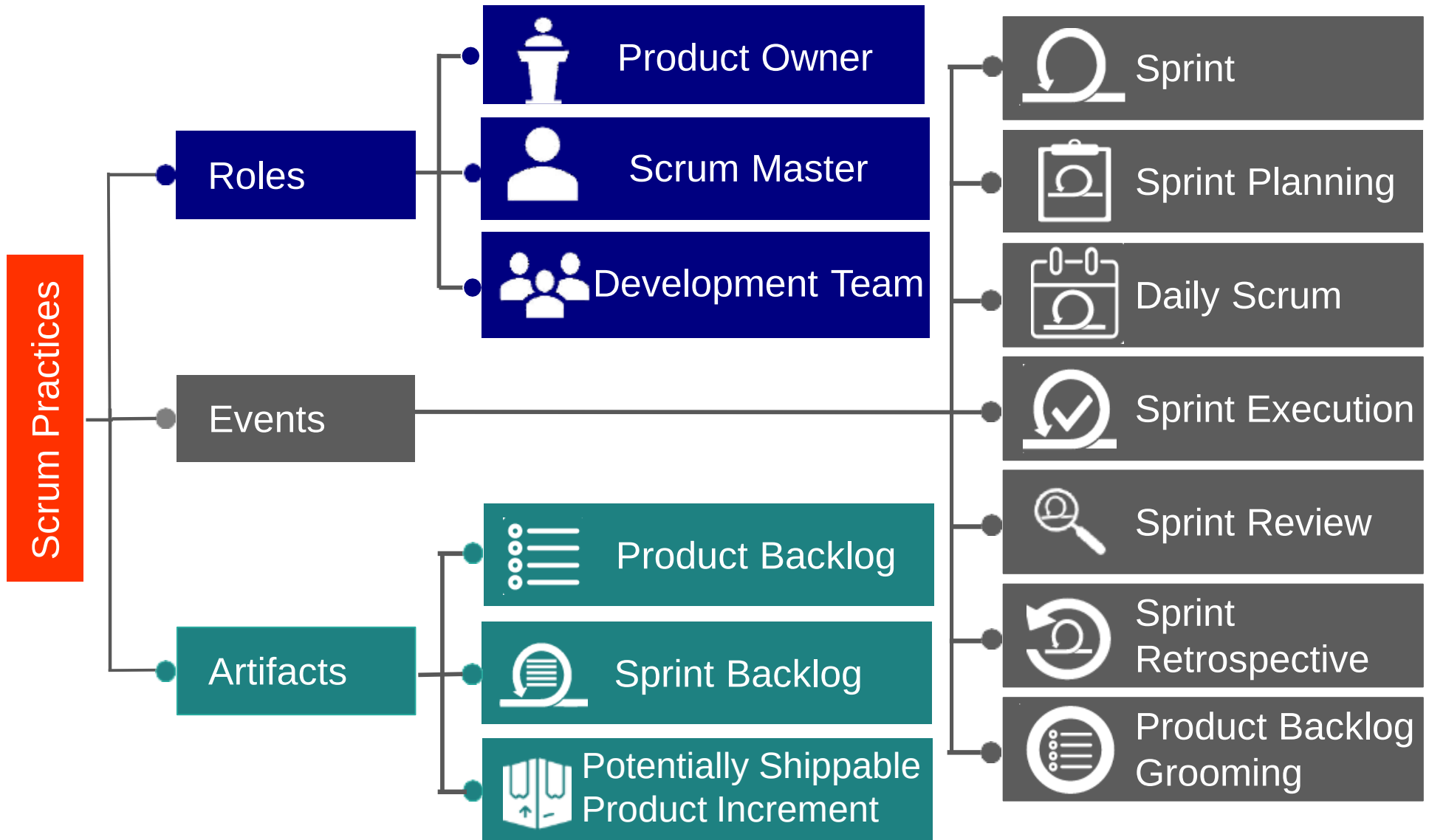


Understanding Scrum





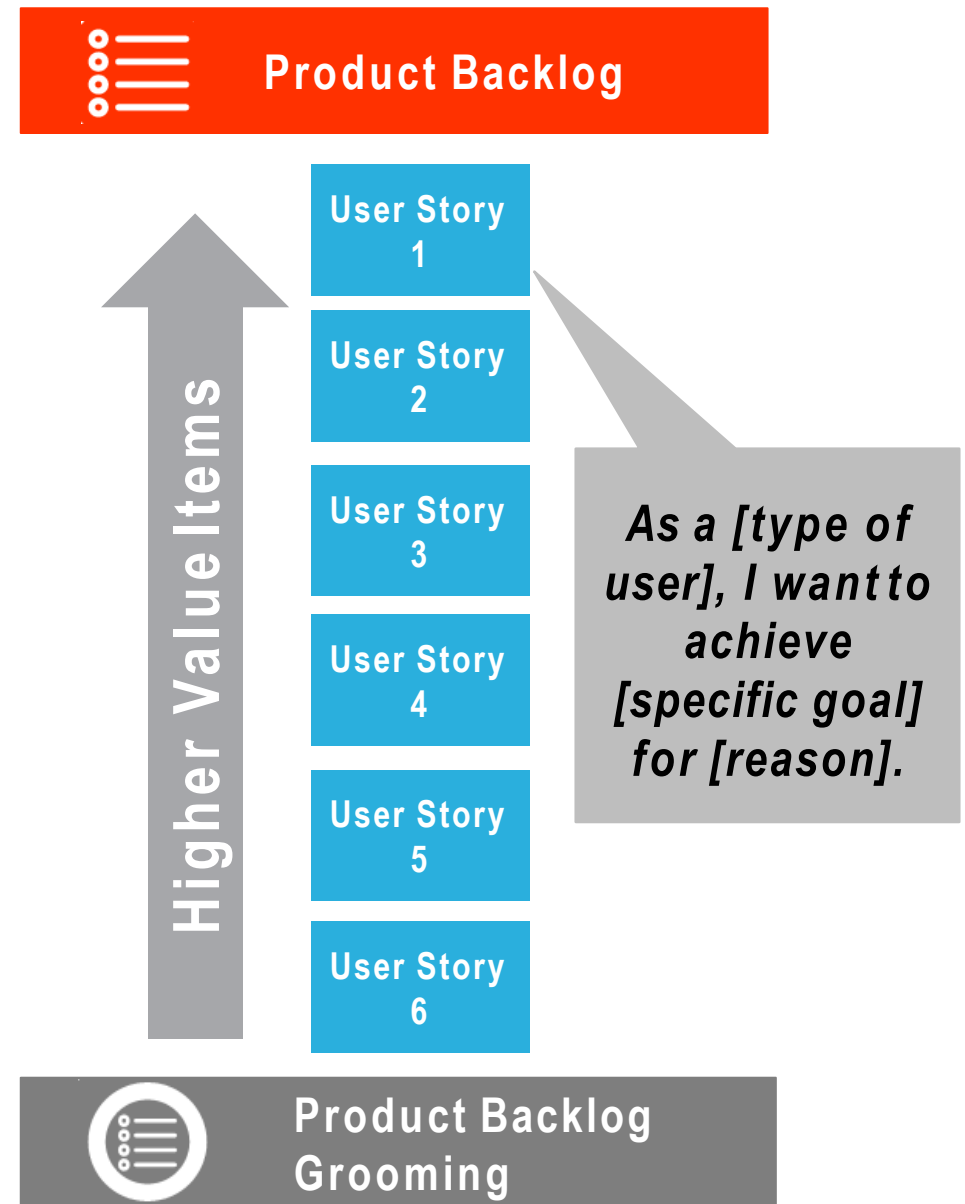
Scrum At-A-Glance





The Product Backlog

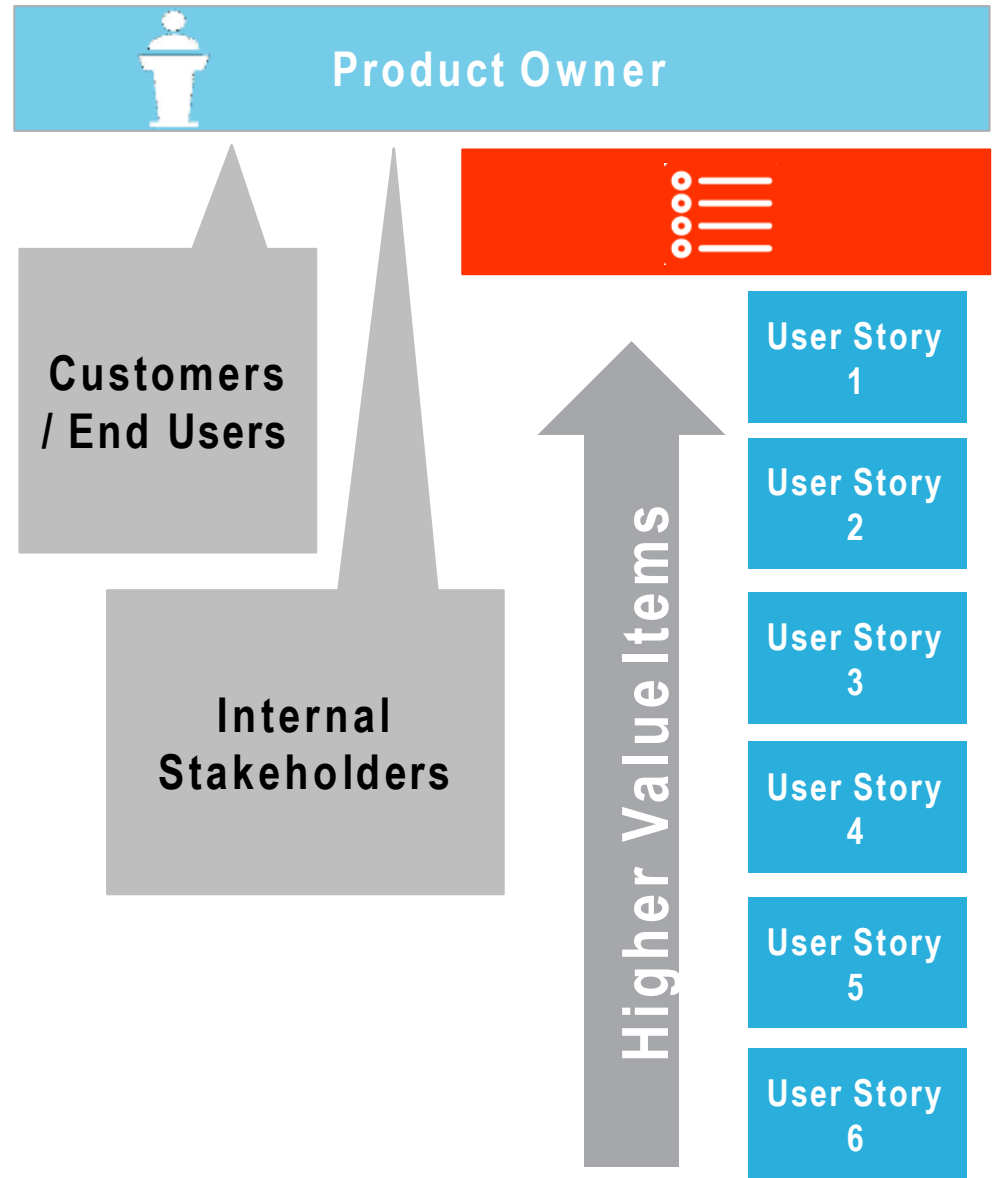
In a Scrum, product **features** are looked at from the perspective of the end user and called **User Stories**. These User Stories are collected and prioritized according to their business value in the **Product Backlog**.





The Product Owner

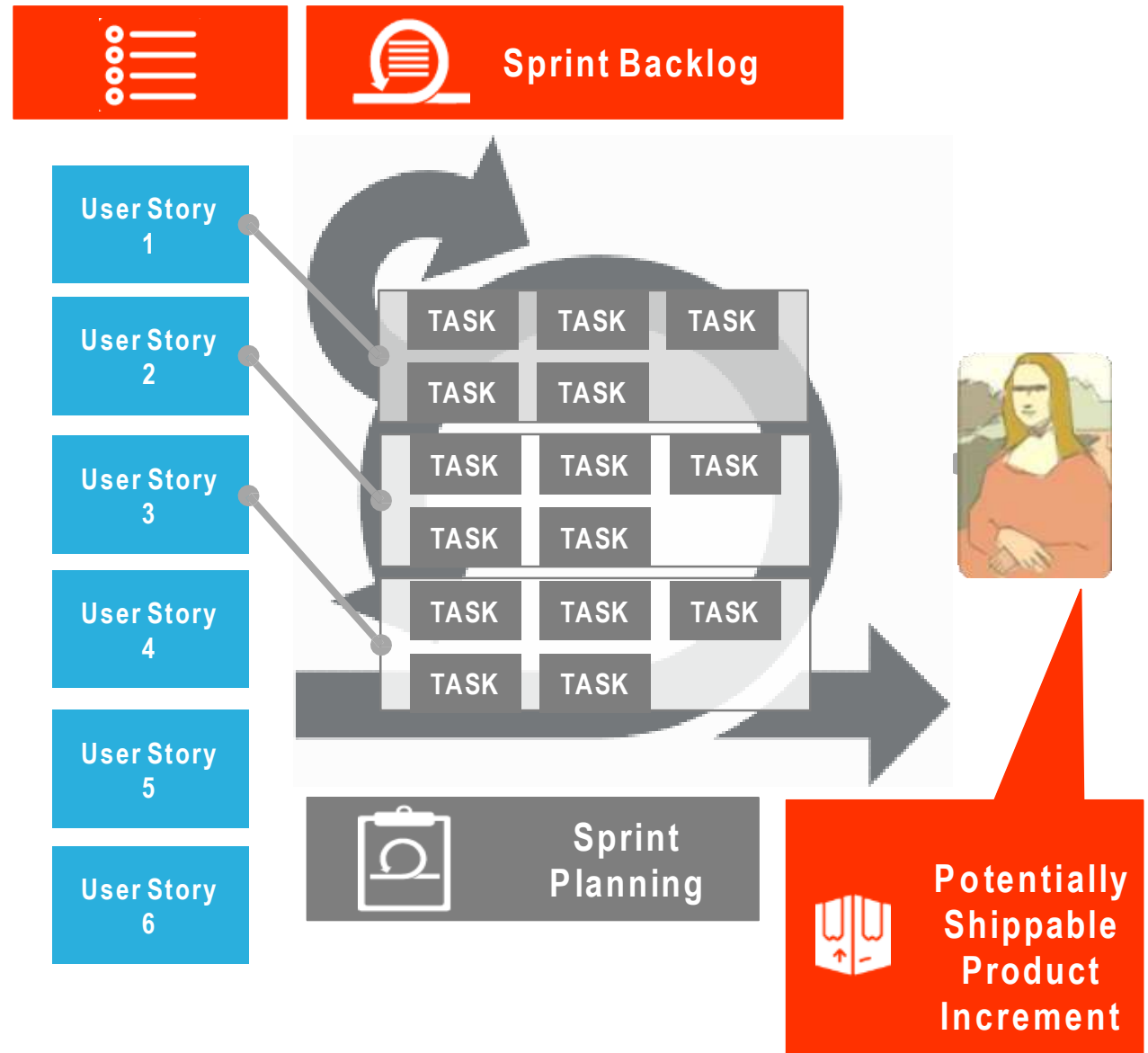
The **Product Owner** is responsible for creating and maintaining the Product Backlog. They are in constant communication with the customer and collaborate with the team.





The Sprint Backlog

The **Sprint Backlog** is the result of the Product Backlog being refined and prioritized, a focused list of items to be worked on during a Sprint.

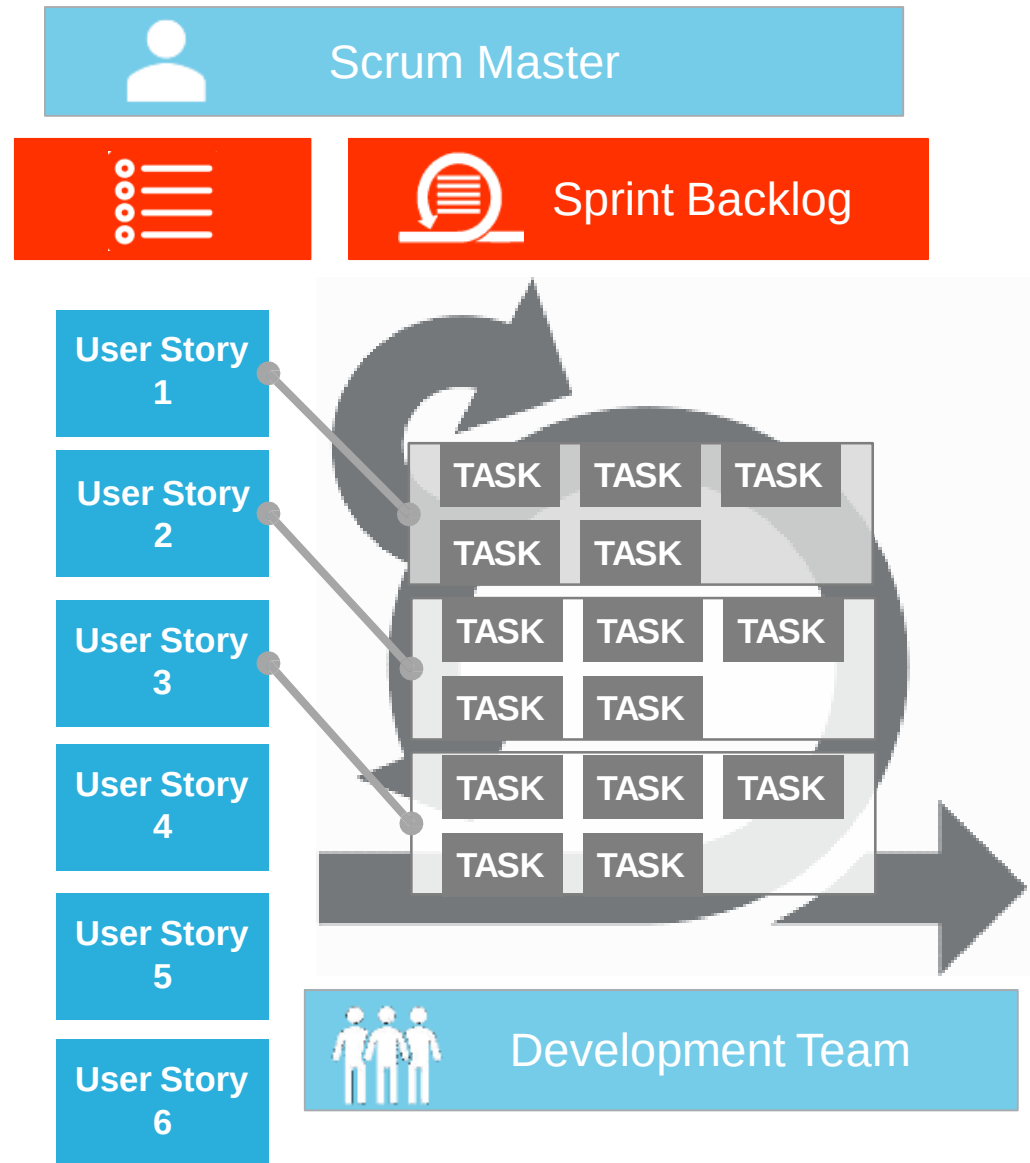




The Scrum Master & Development Team

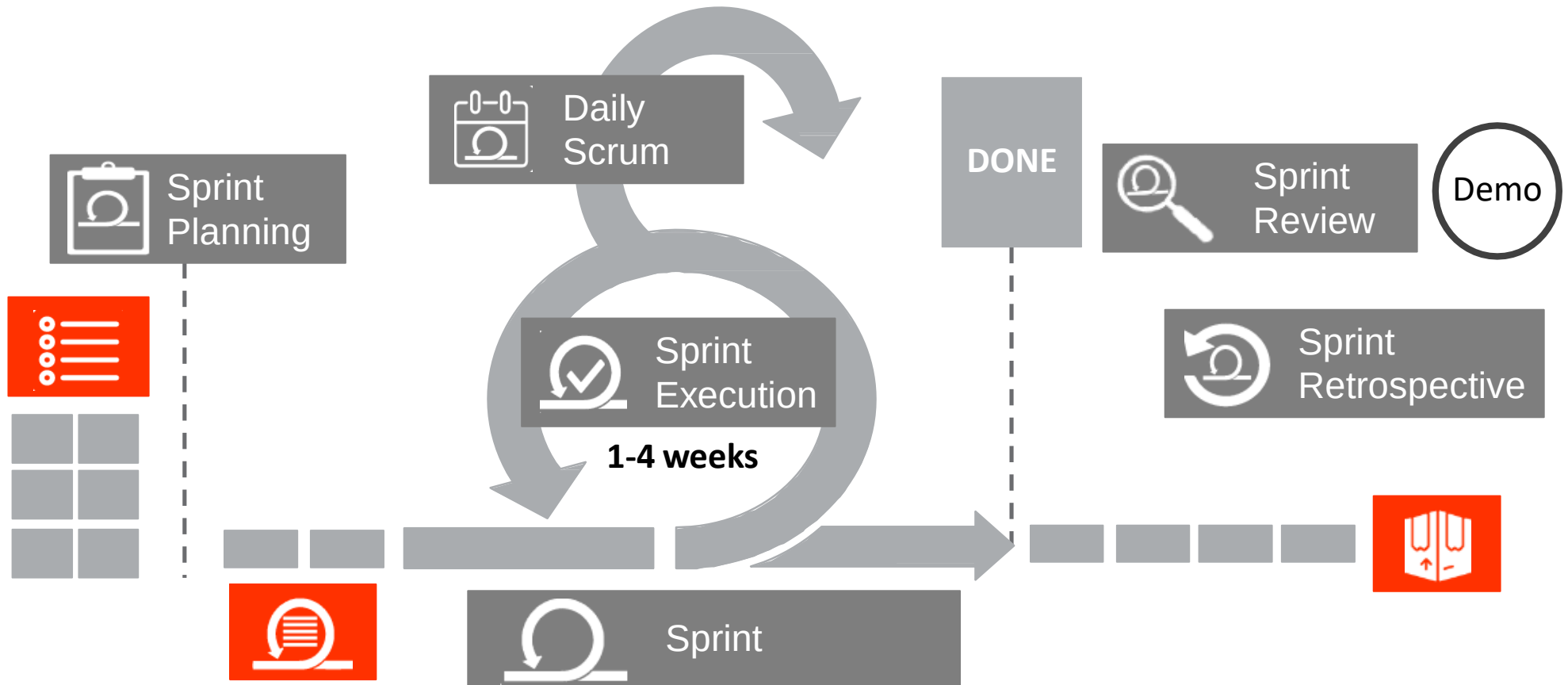
The **Scrum Master** ensures that the Scrum Framework is followed entirely and correctly and can fill in for the Product Owner.

The **Development Team** own and manage the Sprint Backlog, and are accountable to build, test and implement the Product Backlog items brought over.



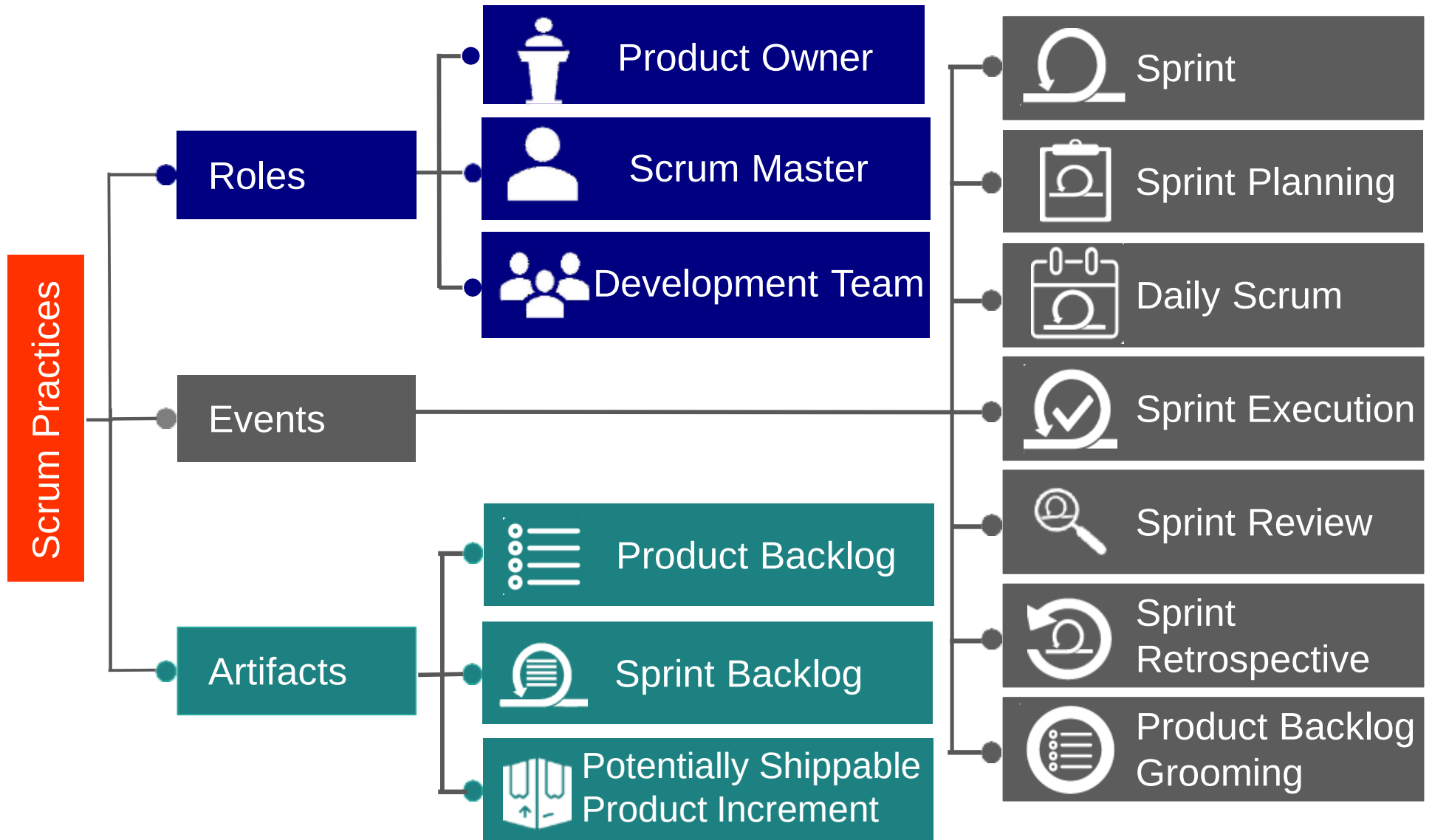


Scrum Events & Timeboxing



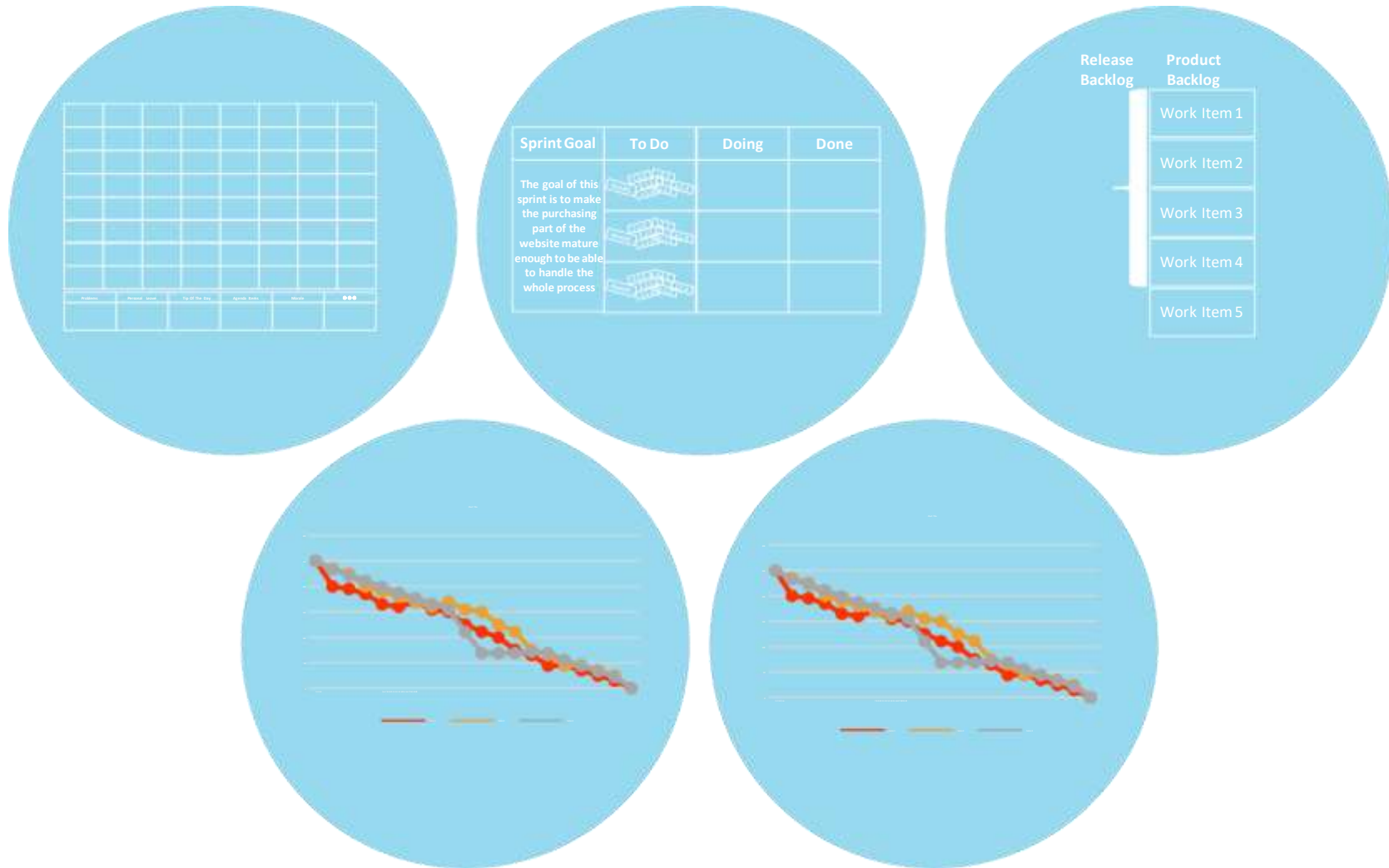


Scrum At-A-Glance





Visual Management For Scrum



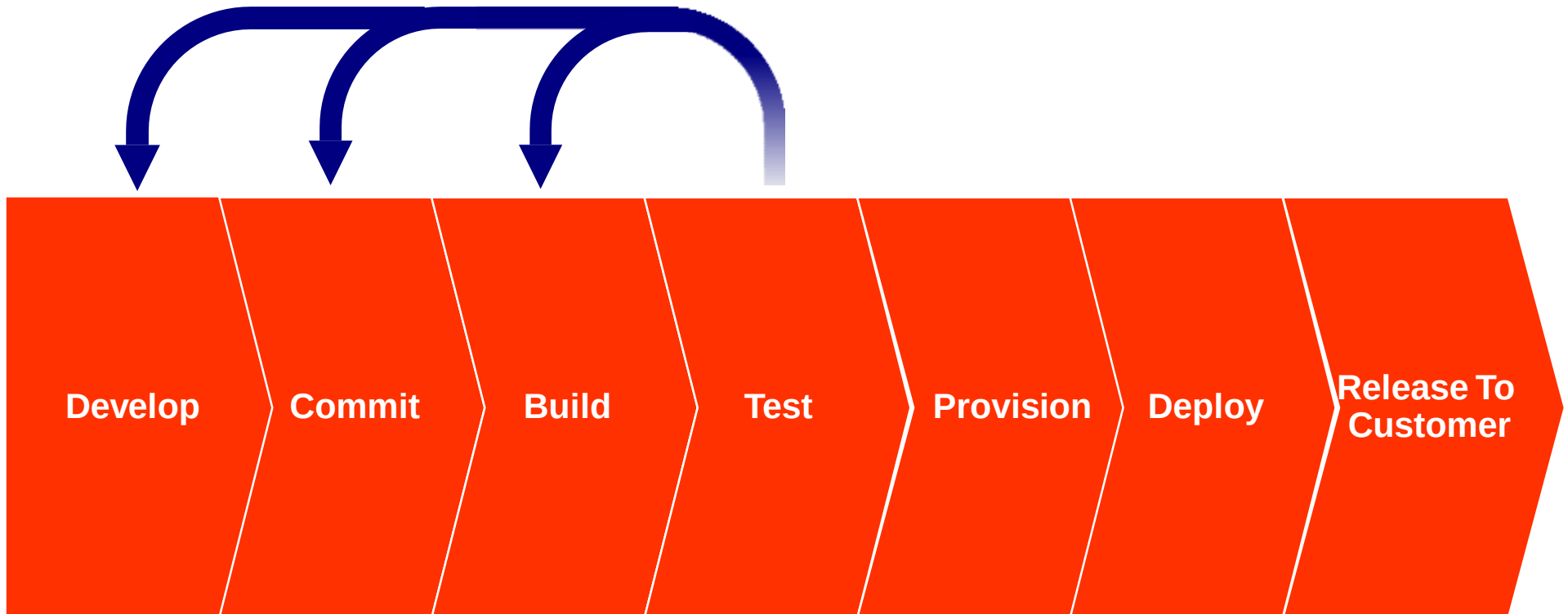


Detect problems as they occur and fail faster with Shift Left Testing.



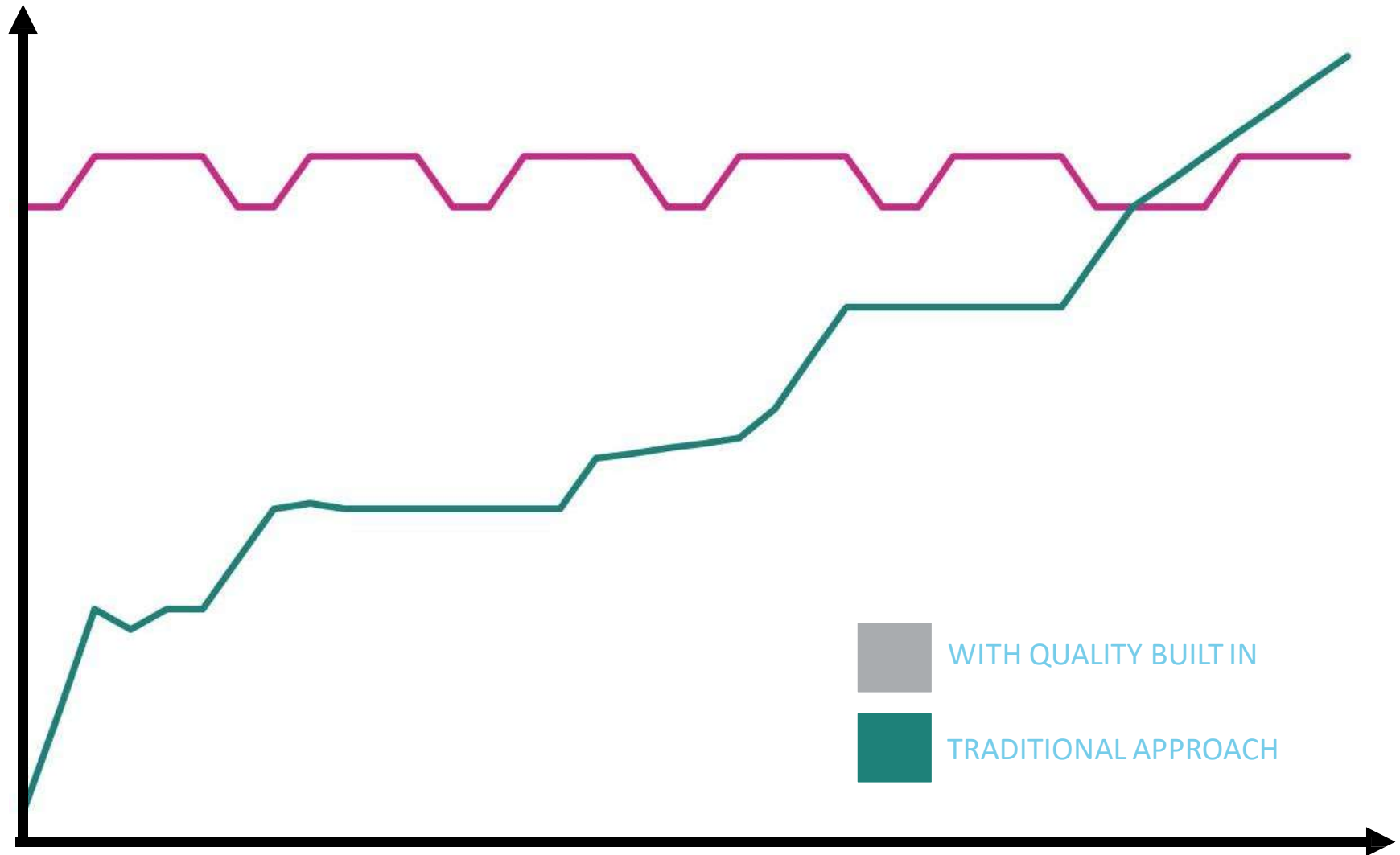
The Shift Left

DevOps advocates the power of **feedback** and **Shift Left thinking** encourages the shifting of testing and feedback earlier in development. This allows us to be more adaptive and responsive to customer requests and change, and for early detection and correction of errors.





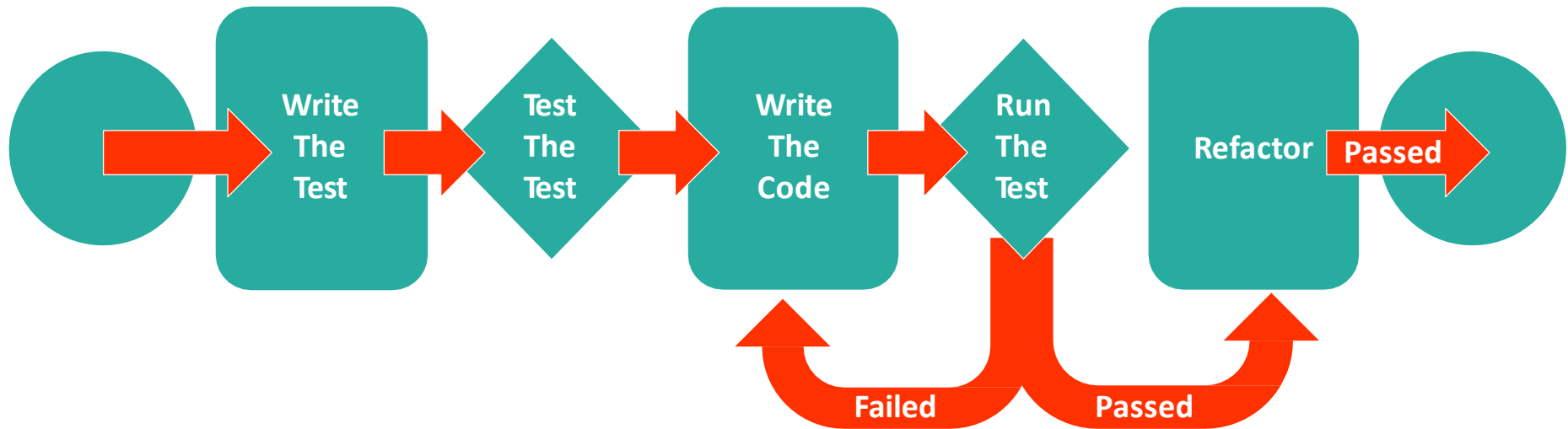
Putting Quality First



Source: Practice To Perfect: The Quality First Model



Test Driven Development





**Achieve the right balance of flexibility and stability with
Change Control**



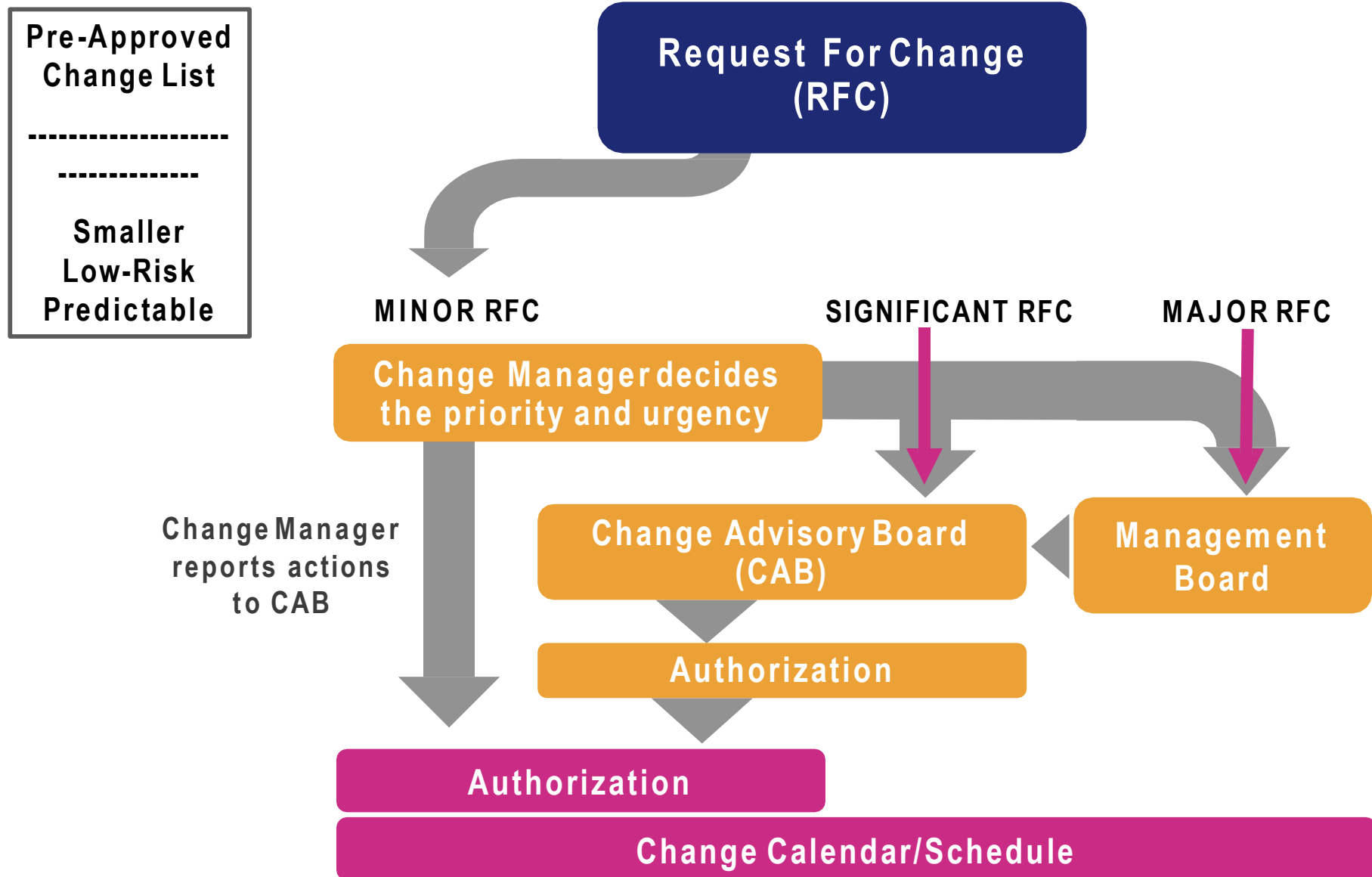
Change Types

A change is any **addition, modification, or removal** of something that could potentially have an effect on an IT service.





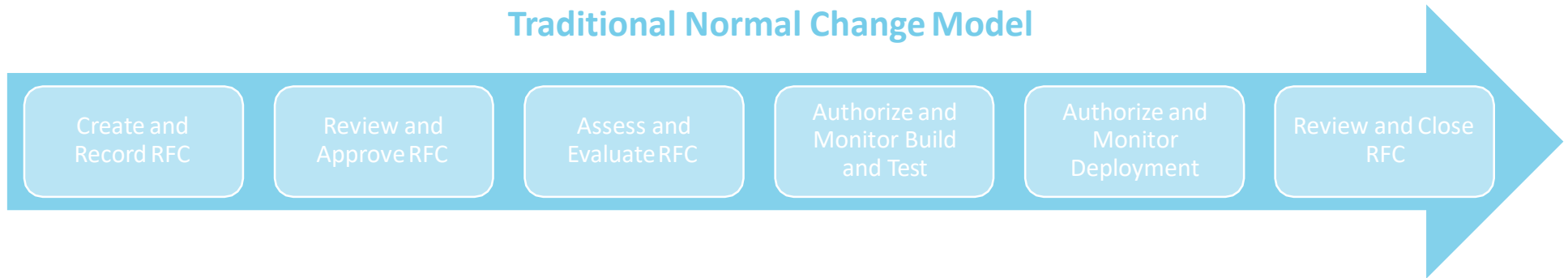
The Traditional Change Approval Process



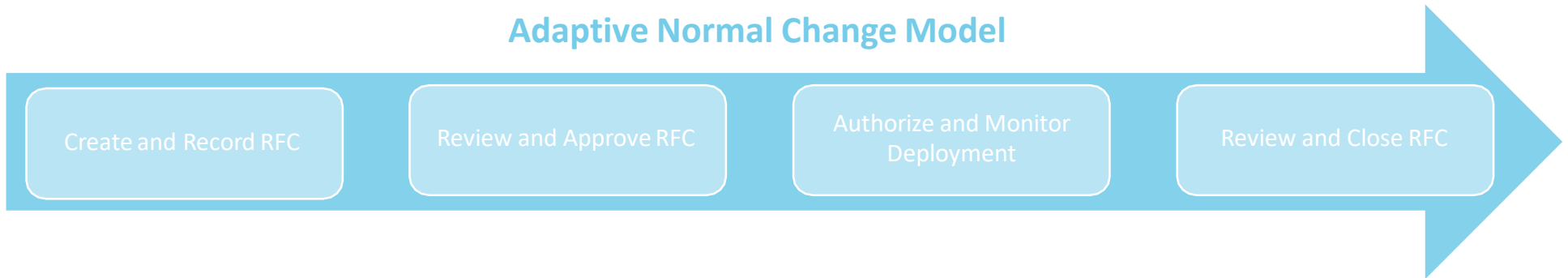


Normal Change Approvals For DevOps

Traditional Normal Change Model



Adaptive Normal Change Model



Plan

Perform

Inspect & Adapt



Product Backlog
Grooming



Sprint
Planning



Sprint

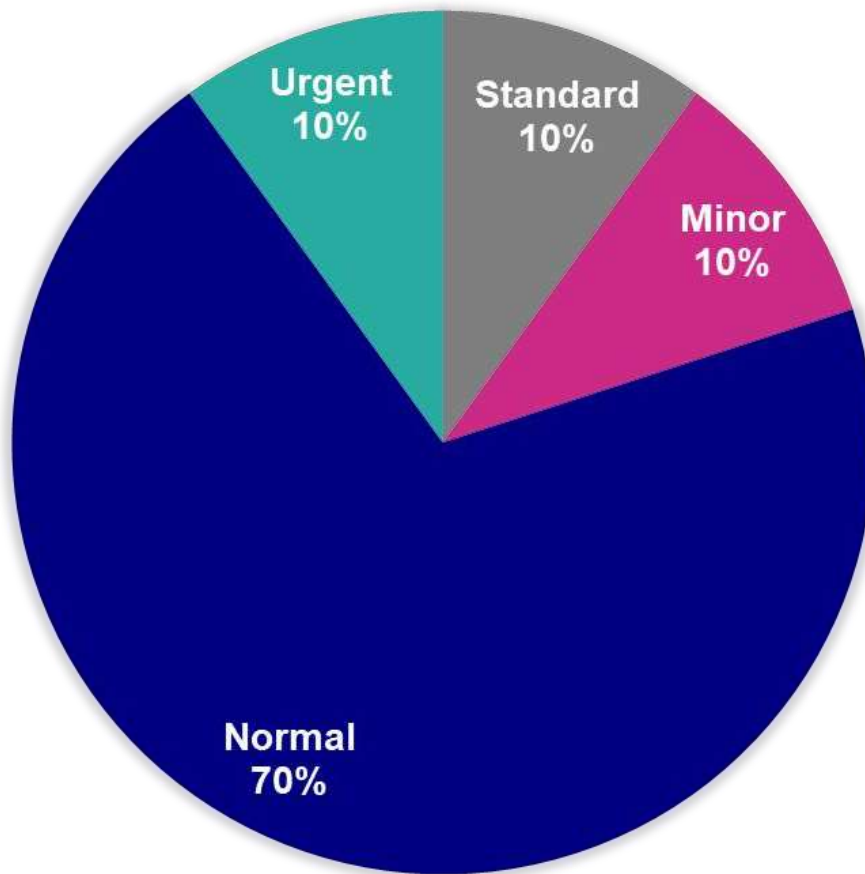


Sprint
Review

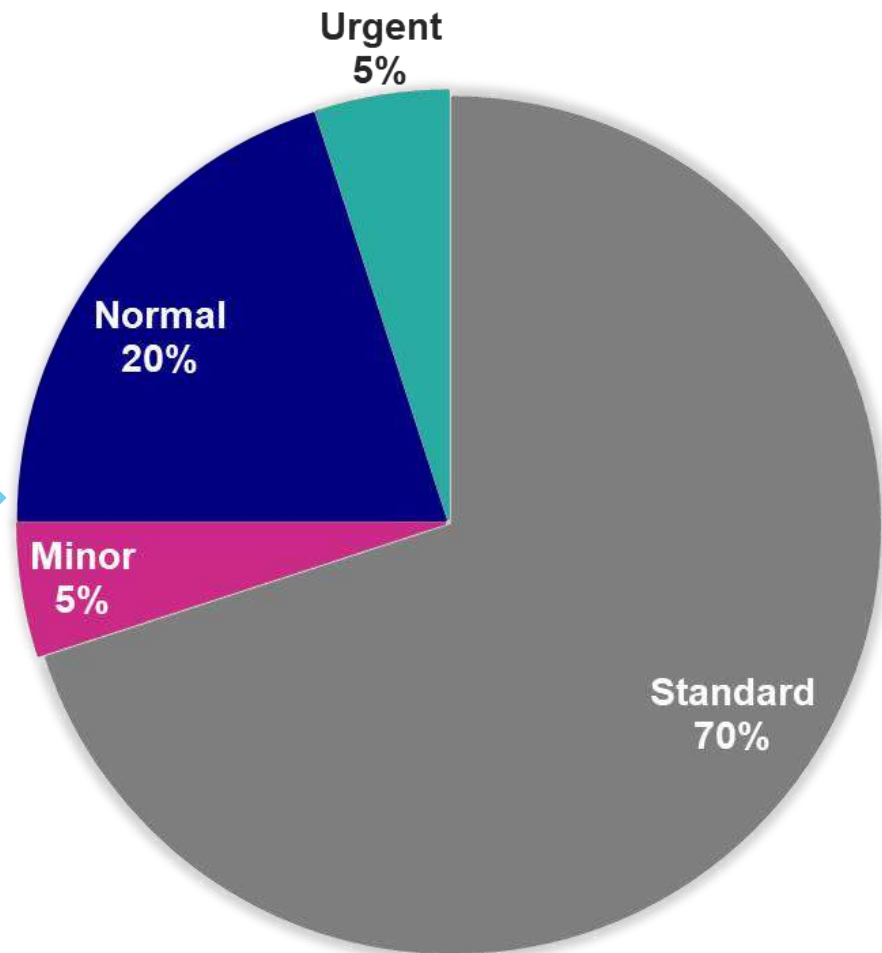


DevOps & Change Control

Change Control
BEFORE DEVOPS



Change Control
AFTER DEVOPS



A large orange circle containing the white text "#10".

#10

**Embed information and resources where you need them,
when you need them with Service Configuration
Management**



Service Configuration Management For DevOps

In a DevOps environment, Service Configuration Management encompasses **the practices and tooling that automate delivery and operation of infrastructure**. Tools are used to:

Model infrastructure

**Continually enforce
desired configurations**

**Remediate
unexpected changes
or “configuration
drift”**



puppet





Infrastructure As Code (IAC)

Automated Service Configuration Management is achieved in DevOps by treating **infrastructure as code** that can be managed with the same tools and processes that developers use, such as version control, code review, automated testing and small deployments.

Using definition files

Implementing self-documenting systems and processes

Versioning everything

Continuously testing systems and processes

Emphasizing small changes rather than big batches

Keeping services available continuously

A large orange circle containing the white text "#11".

#11

**Support automation and create flow with Release &
Deployment Management**



- ✓ **Automated Service Configuration Management and Infrastructure As Code support **automated Release Management** and enable more frequent deployment**
- ✓ **Deploy new Build and Release on **Virtual Machines** avoid stress, overtime and service outage**



Release & Deployment Management

Together, Release and Deployment Management provide **guidance related to planning, scheduling and controlling releases** as they transition from the development environment into production.

Deployment refers to the moving of new or changed service components to live environments

Release refers to the actual making of new and changed services and features available.



How does this work in a culture in which teams have more control over production deployments?



Release & Deployment Management Techniques

Automate Service Configuration Management with IAC

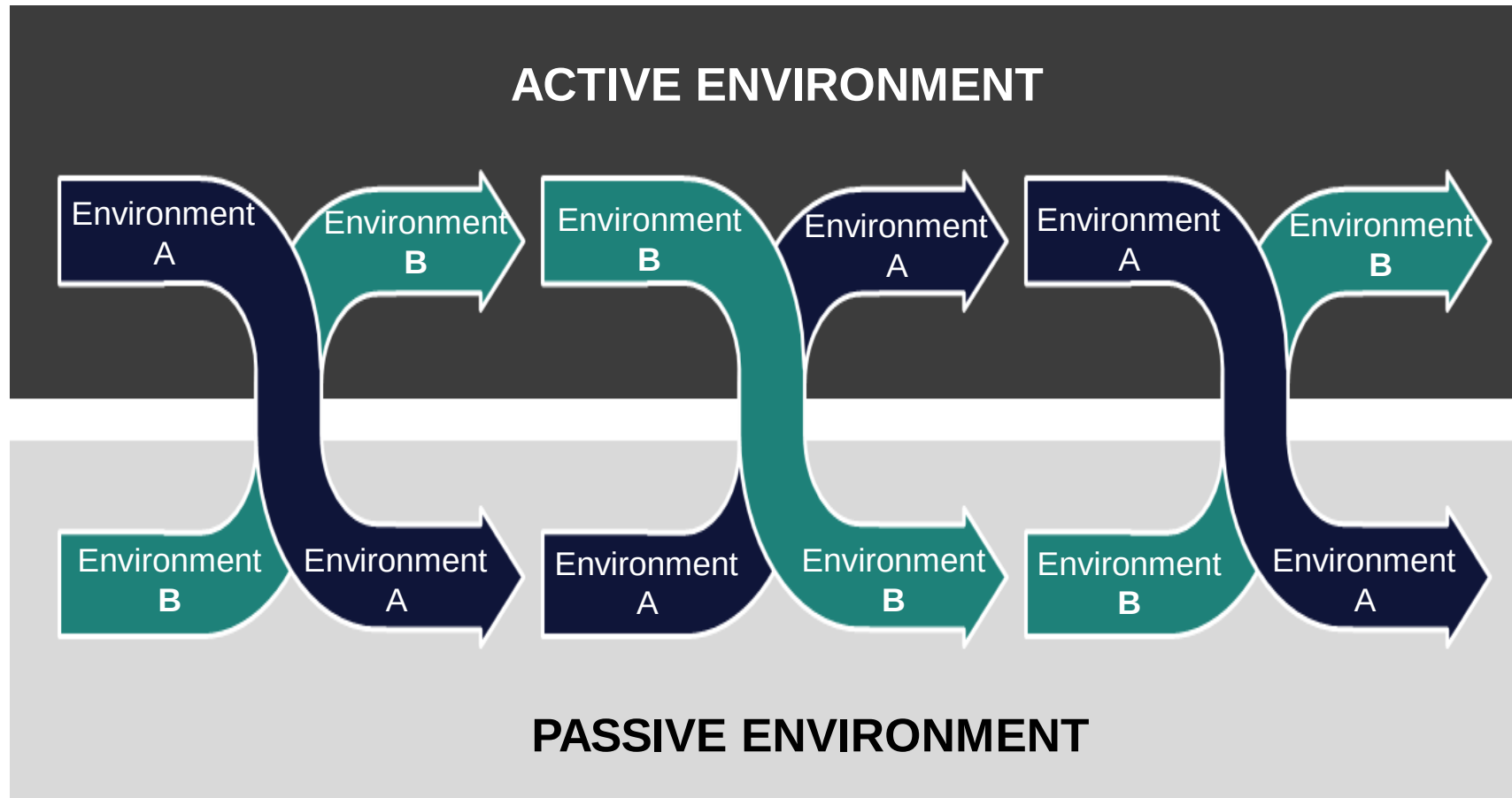
- Increase reliability
- Make the release process repeatable
- Improve audits with living documentation

Rebuild servers regularly

Implement zero-downtime deployments



Blue-Green Deployment

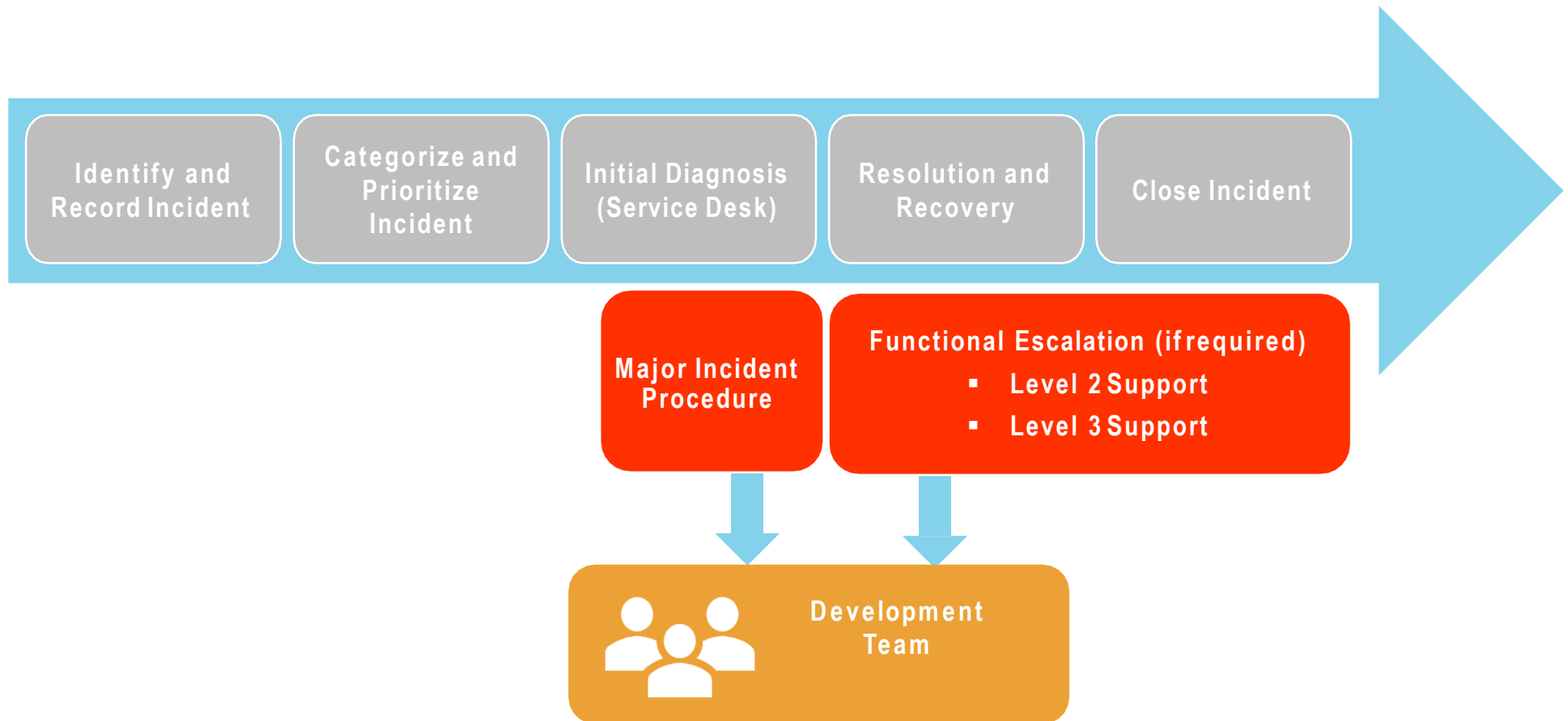




Swarm to solve incidents with Incident Management



Adapting Incident Management for DevOps





Incidents vs. Problems



Incident

- An unplanned interruption to an IT service or reduction in the quality of an IT service
- Failure of a Configuration Item that has not yet impacted service

Problem

- The underlying cause of one or more Incidents
- The cause is not usually known at the time a problem record is created, and the Problem Management process is responsible for further investigation

A large orange circle containing the white text "#13".

#13

Turn knowledge into improvements that push quality closer to the source with Problem Management & Kaizen



Adapting Problem Management for DevOps





Problem Solving With Lean Kaizen

KAIZEN

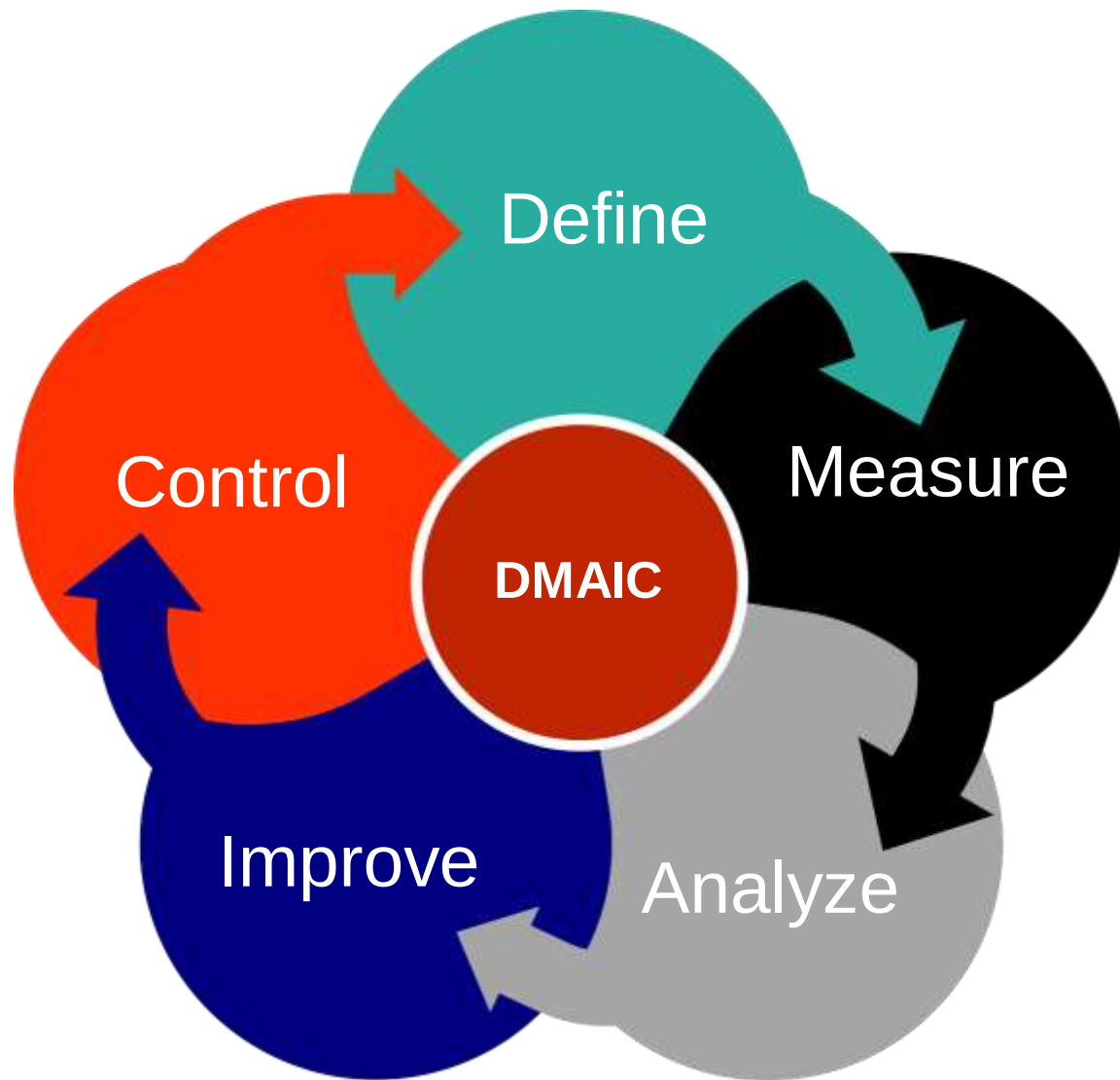
改善

CHANGE

FOR THE BETTER



The DMAIC Cycle For Improvement



A large orange circle containing the white text "#14".

#14

**Institutionalize daily improvement with
Continual Improvement models**

“

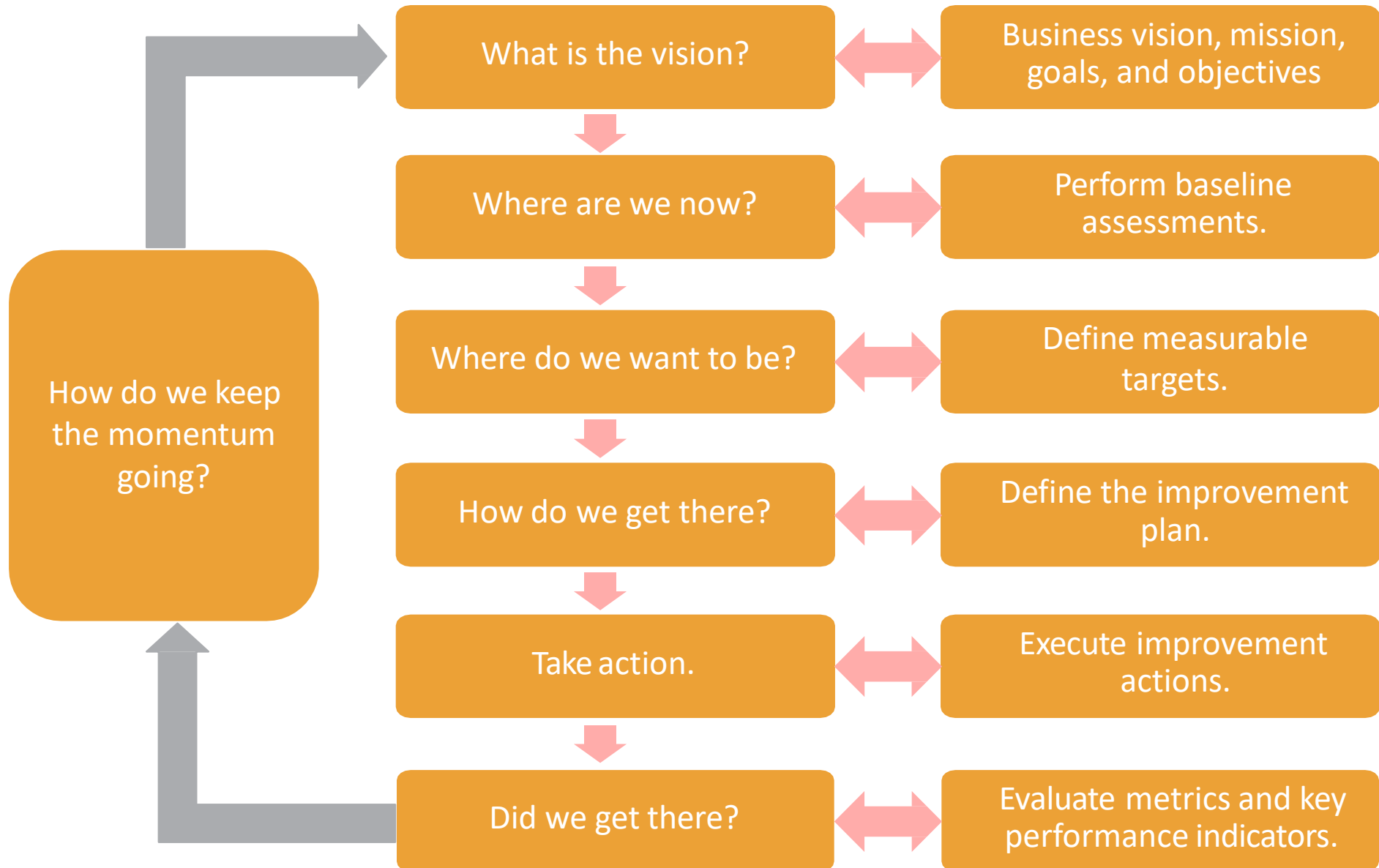
Today's standardization...is the necessary foundation on which tomorrow's improvements will be based. If you think "standardization" as the best you know today, but which is to be improved tomorrow – you get somewhere. But if you think of standards as confining, then progress stops.

• Henry Ford

”



The Continual Improvement Model





The Continual Improvement Register

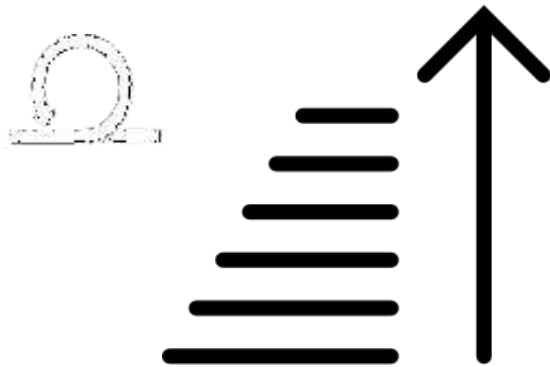
The Continual Improvement Register is a document which **provides structure and visibility to continual improvement efforts**, and which records and categorizes each opportunity for improvement by timeframe, effort level, urgency, impact and priority.

Reference ID	Date Logged	Logged By	Effort Level	Timeframe	Description	Reason	Urgency	Impact	Priority

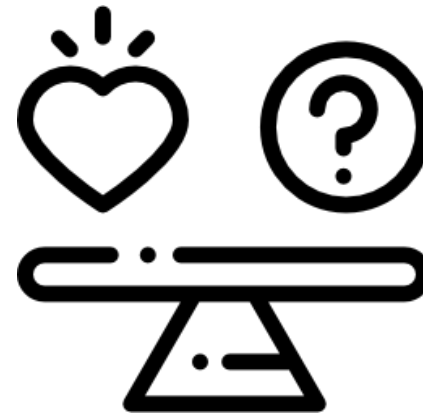


Continual Improvement Supports Everything In DevOps

Continual Improvement should be implemented with the goal of improvement that supports the Full Stack and the Three Ways, with objectives that:



**Align goals and priorities
around delivering
business value**



**Balance improvements
to time and cost
efficiencies with
improvements to quality
and effectiveness**





Embrace failure as a learning opportunity with a focus on Antifragility.



Antifragility Goes Beyond Resilience

Disaster Recovery

The means to respond to worst-case scenarios and protect critical systems from incidents and disruption



Resilience

The means to respond to, but also resist, incidents and disruptions of all kinds



The means to not only respond to and resist incidents and disruptions of all kinds, but to use them as an opportunity for learning and adaptation



Embracing Failure As A Learning Tool

NETFLIX



In 2011,
Netflix sets
loose Chaos
Monkey.

**THE
SIMIAN
ARMY**

Soon, they
have a
collection of
cloud testing
tools.



Source: "The Netflix Simian Army", The Netflix Tech Blog

“

Chaos Engineering is the discipline of experimenting on a distributed system in order to build confidence in the system's capability to withstand turbulent conditions in production.

The Principles Of Chaos Engineering

”



The 15 Essential Practices Of DevOps

#1

**Voice Of The
Customer**

#2

**Relationship
Management**

#3

**Lean Process
Optimization**

#4

**Value Stream
Mapping**

#5

**Knowledge
Management**

#6

**Visual
Management**

#7

Agile Scrum

#8

**Shift Left
Testing**

#9

**Change
Control**

#10

**Service
Configuration
Management**

#11

**Release &
Deployment
Management**

#12

**Incident
Management**

#13

**Problem
Management
& Kaizen**

#14

**Continual
Improvement**

#15

Antifragility



PeopleCert DevOps Fundamentals

MODULE 5:

The Full Stack:

Technology & Automation

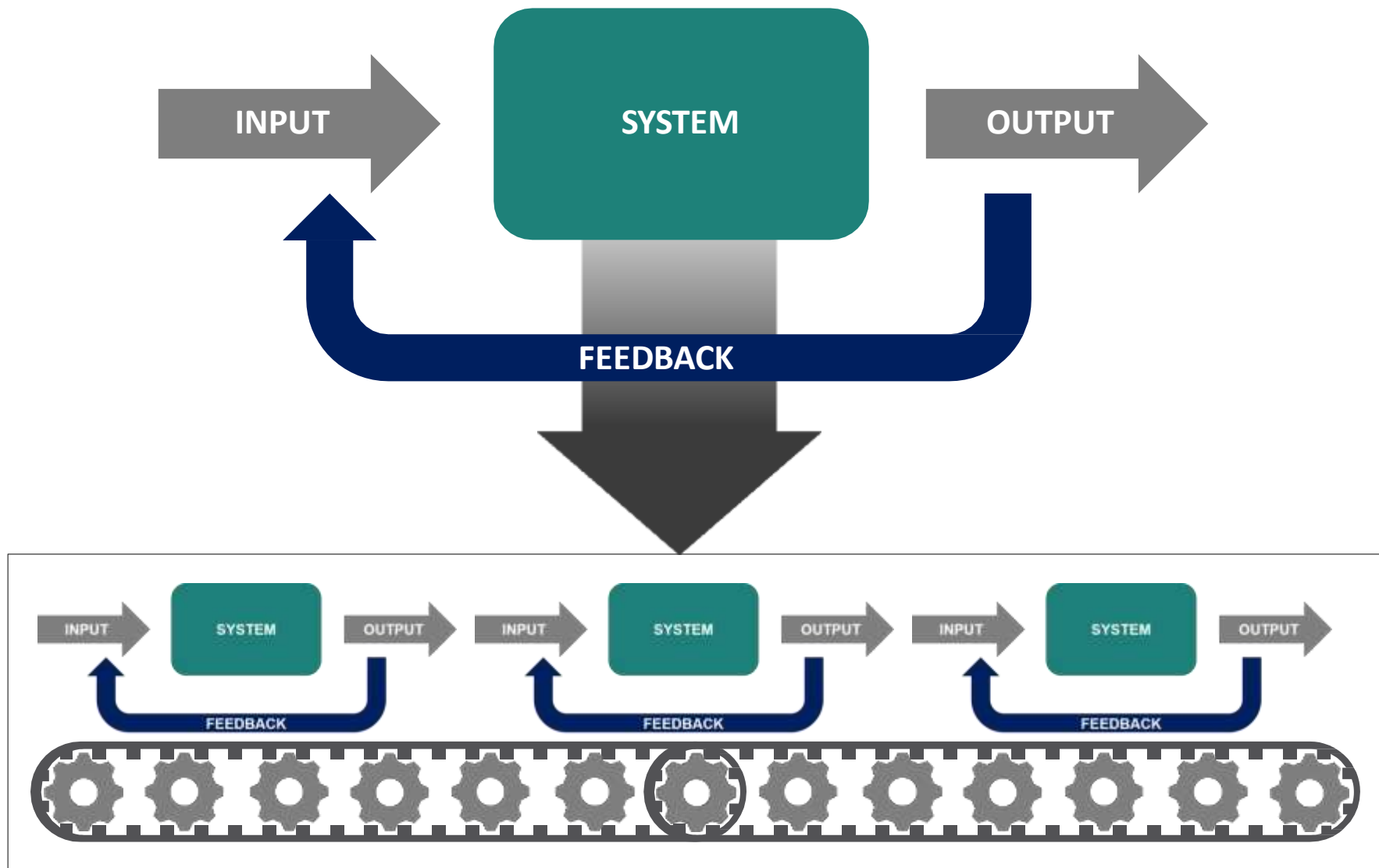
- Automation For The Deployment Pipeline
- Cloud Technology & Virtualization
- Architecting For Continuous Delivery



Automation For The Deployment Pipeline



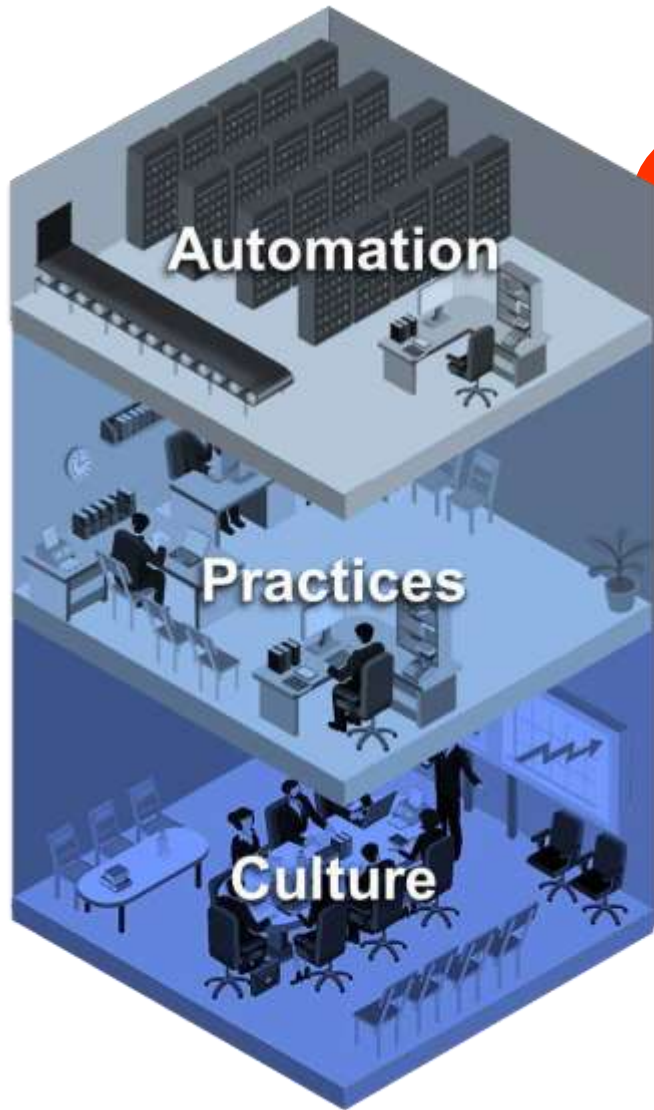
Understanding Automation



Source: *Service Automation Framework*, Jan-Willem Middelburg



Automation & The Full Stack

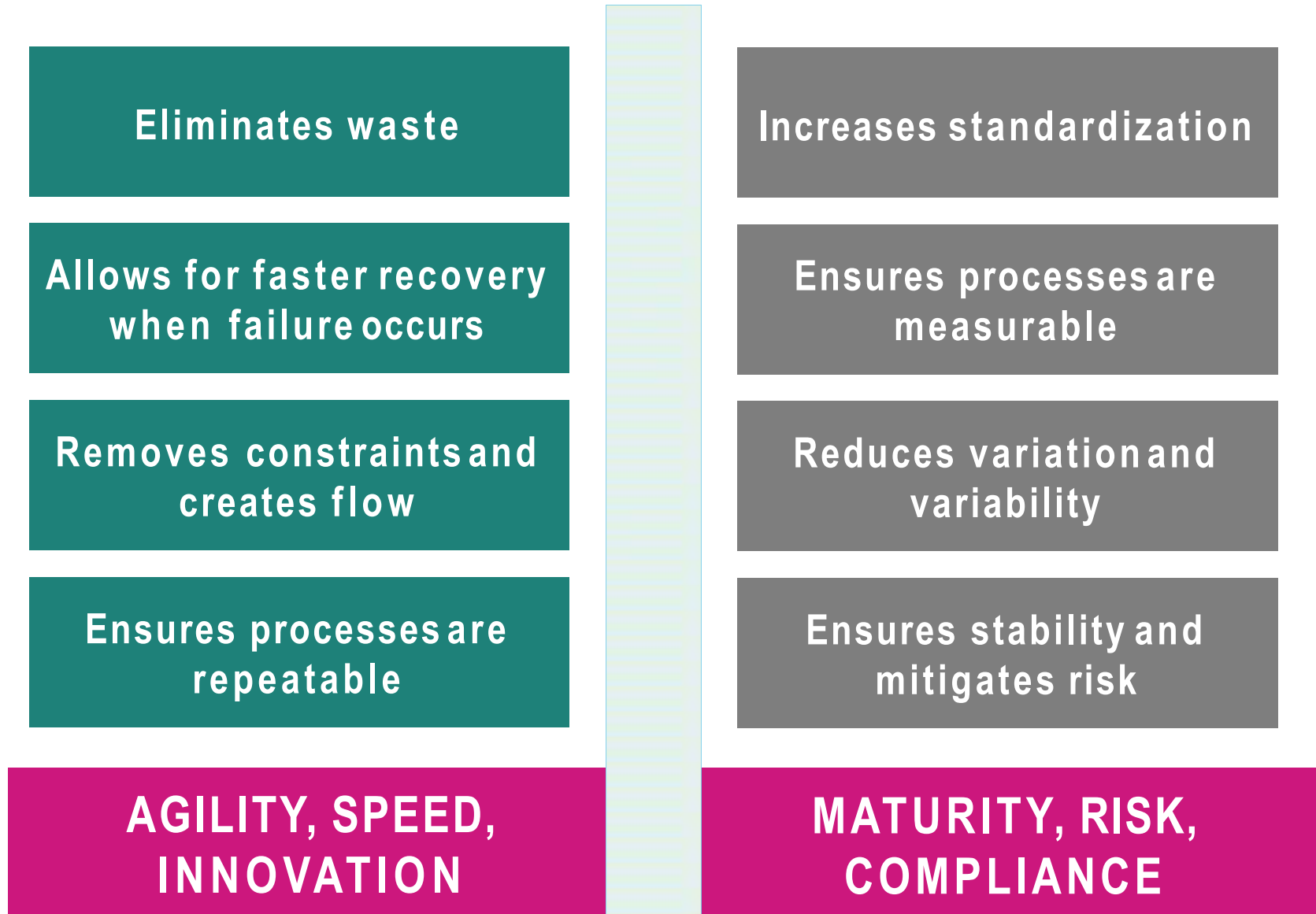


*The first rule of any technology used in a business is that automation applied to an efficient **operation** will magnify the efficiency. The second is that automation applied to an inefficient operation will magnify the inefficiency.*

Bill Gates

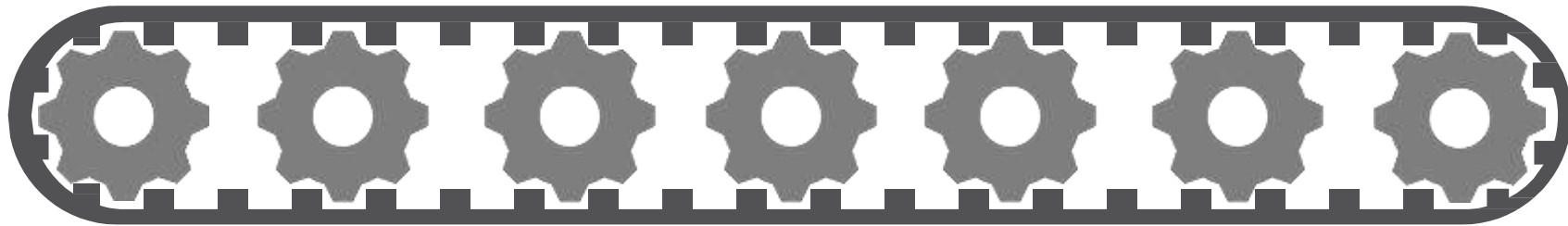


Automation Means Flexibility & Stability





Deployment Pipeline Automation



Ensure the quality
of code

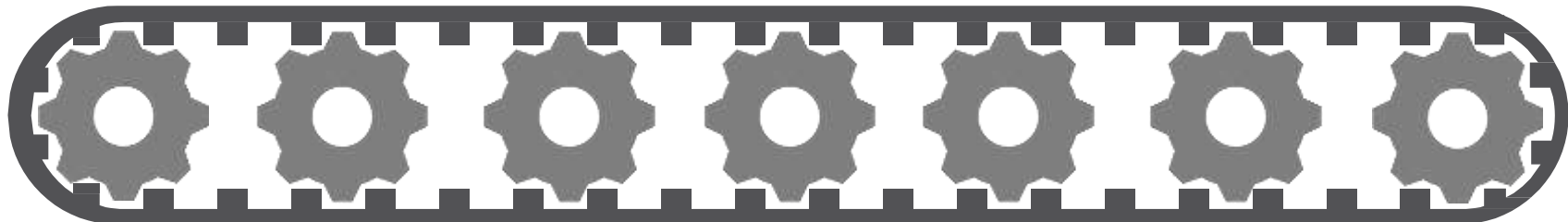
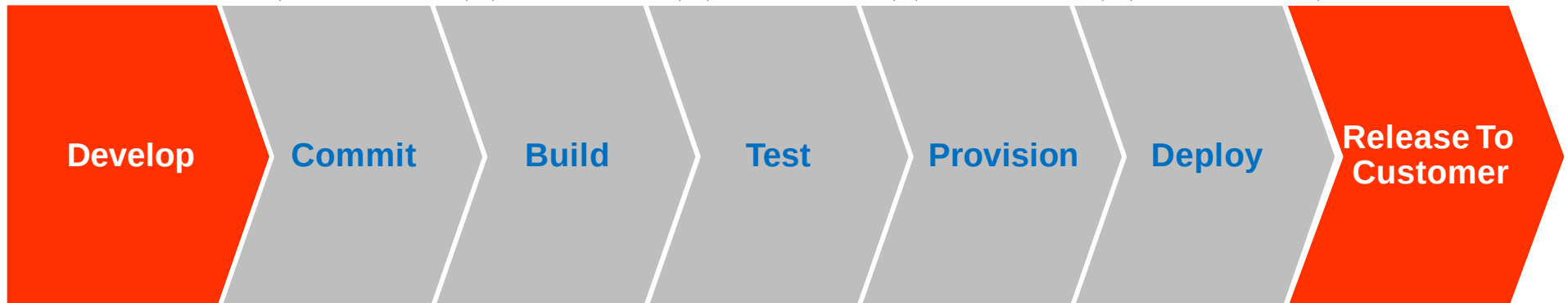
Ensure the state of
environments

Ensure that
software is
deployed correctly

Ensure stability
and performance



Tools For A DevOps Toolchain





Other Tools That Support DevOps

**Database
Management**

**Monitoring &
Dashboards**

**Release
Management**

**Workflow &
Ticketing**

**Service
Configuration**

Collaboration

Security

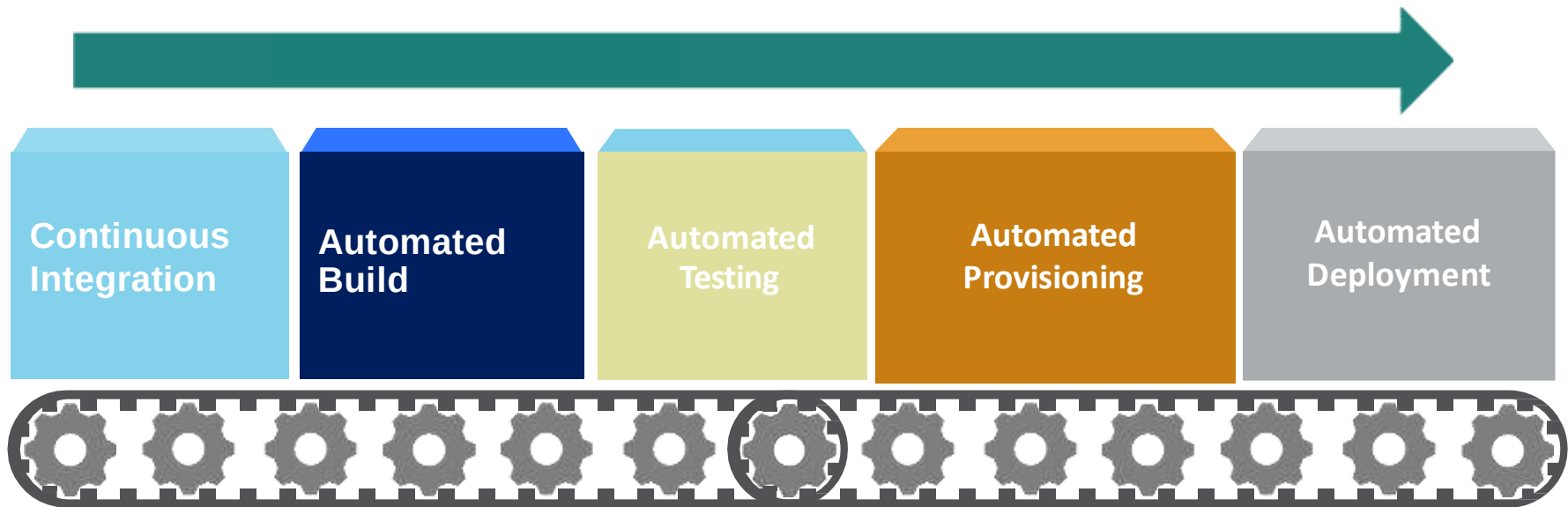
Networking

**Change
Control**

Orchestration



Stages Of DevOps Automation



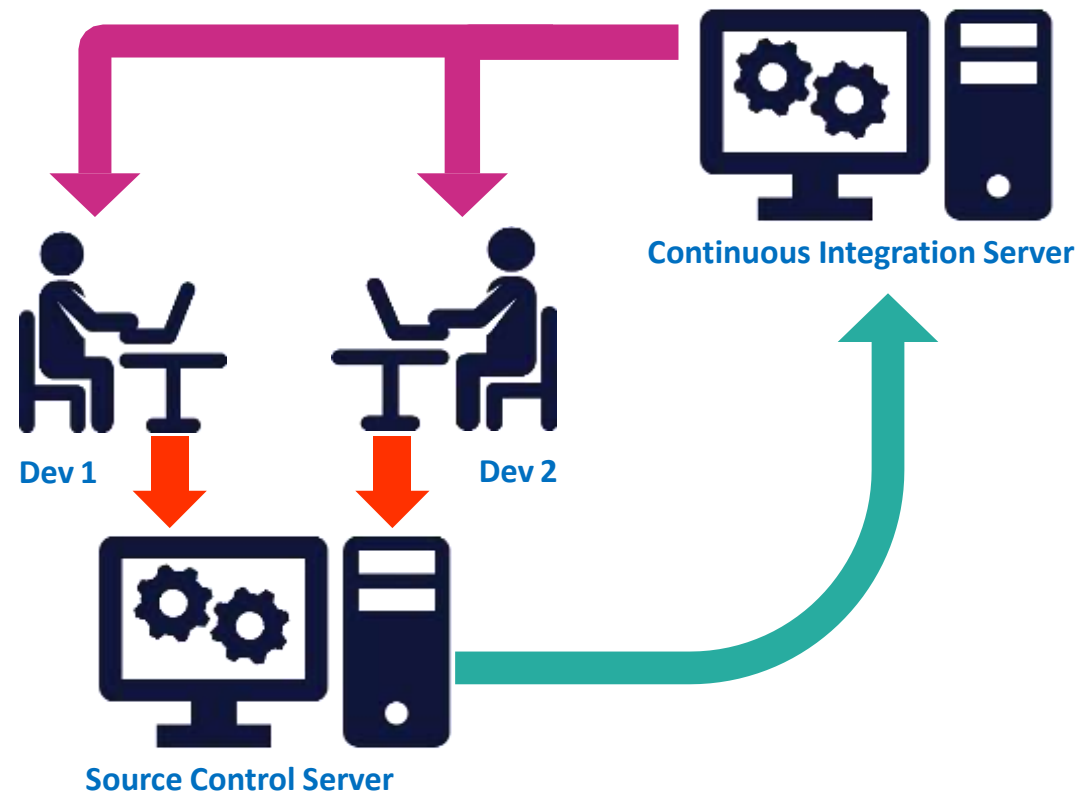


Continuous Integration



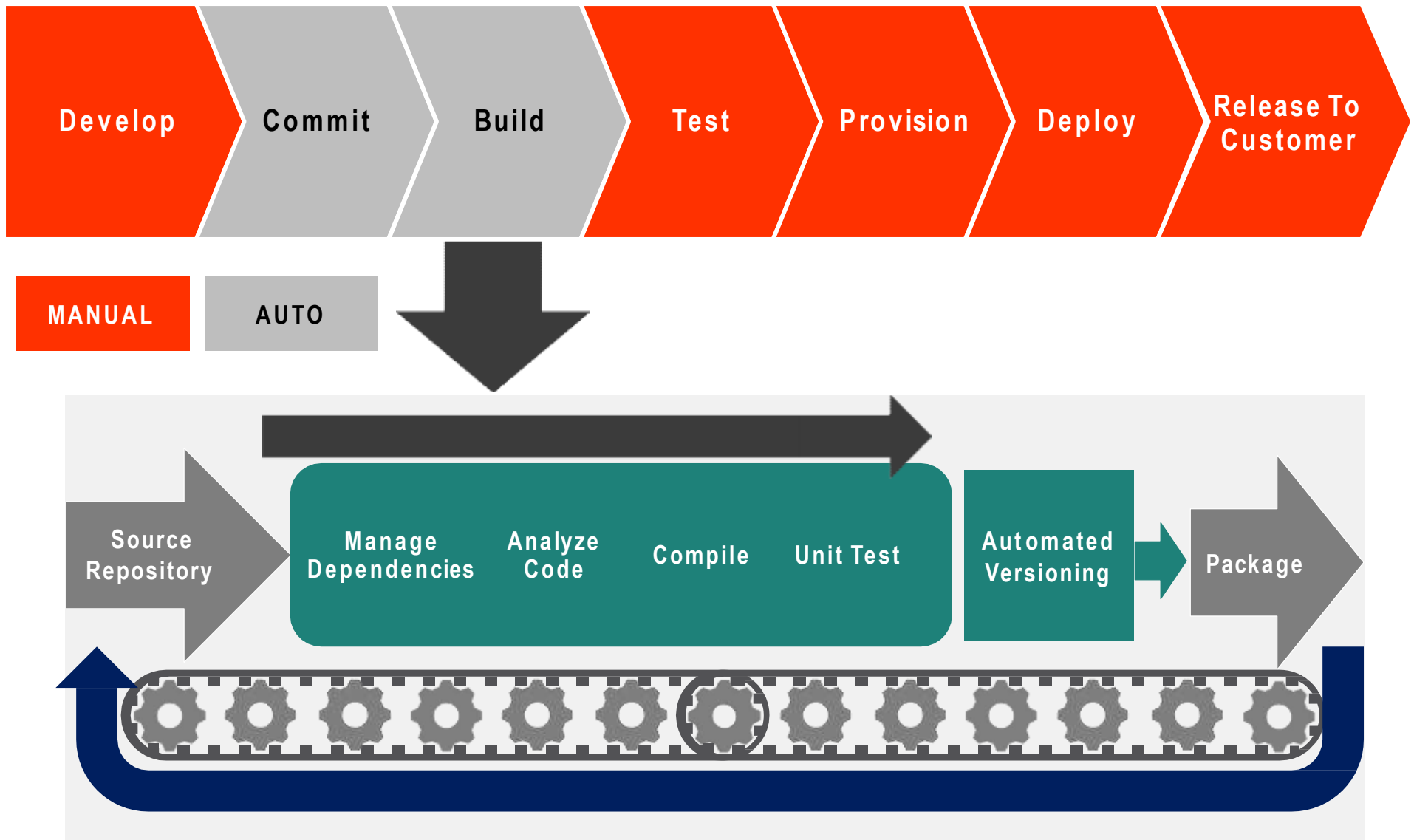
MANUAL

AUTO



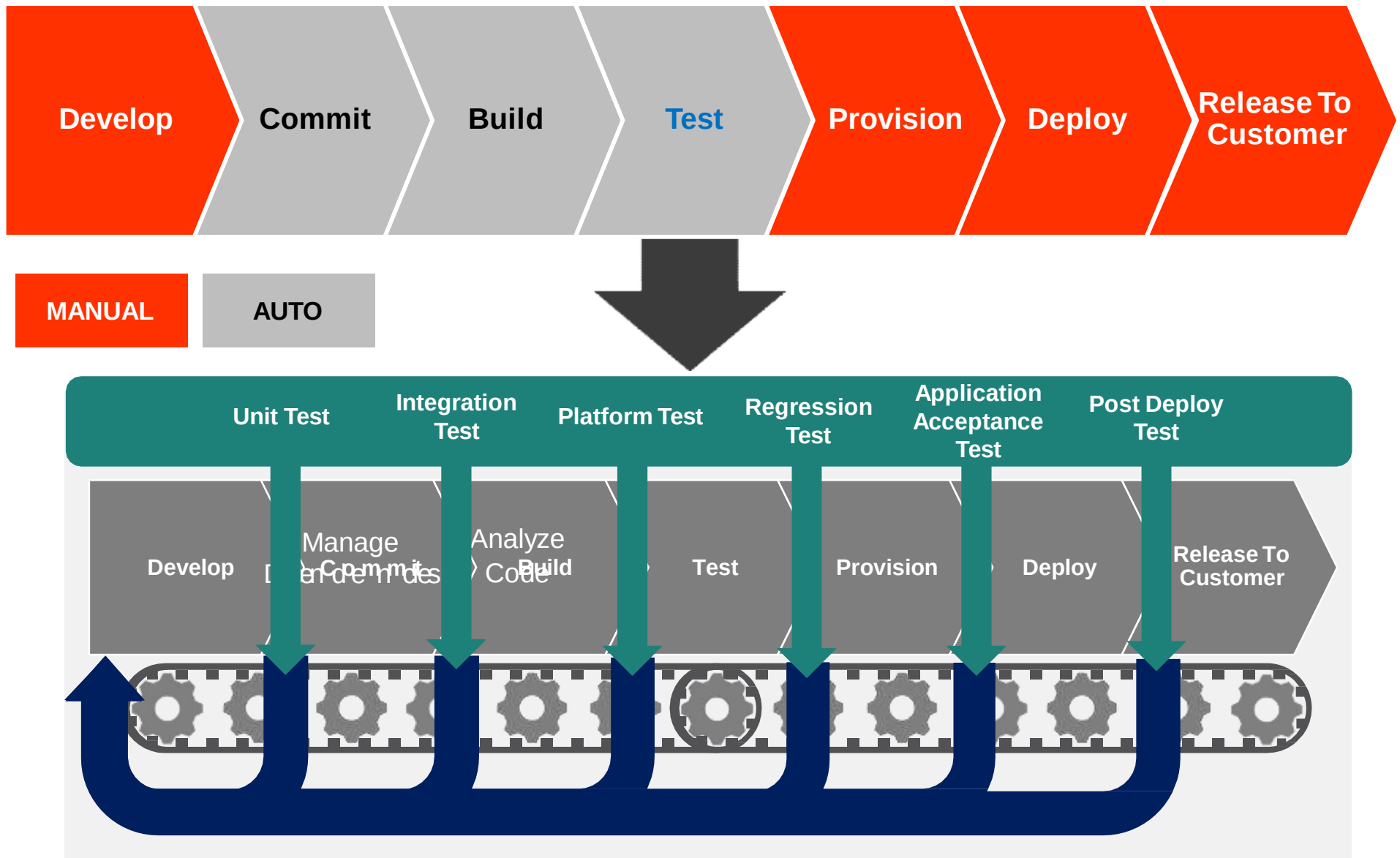


Automated Build



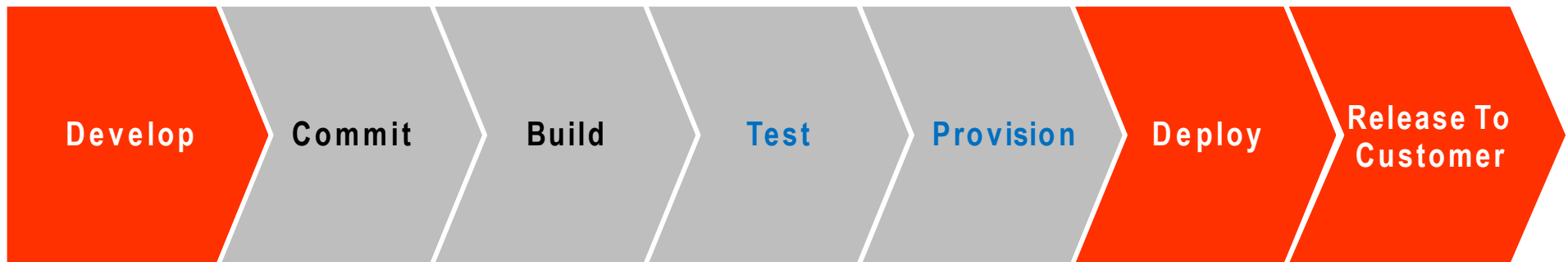


Automated End To End Testing



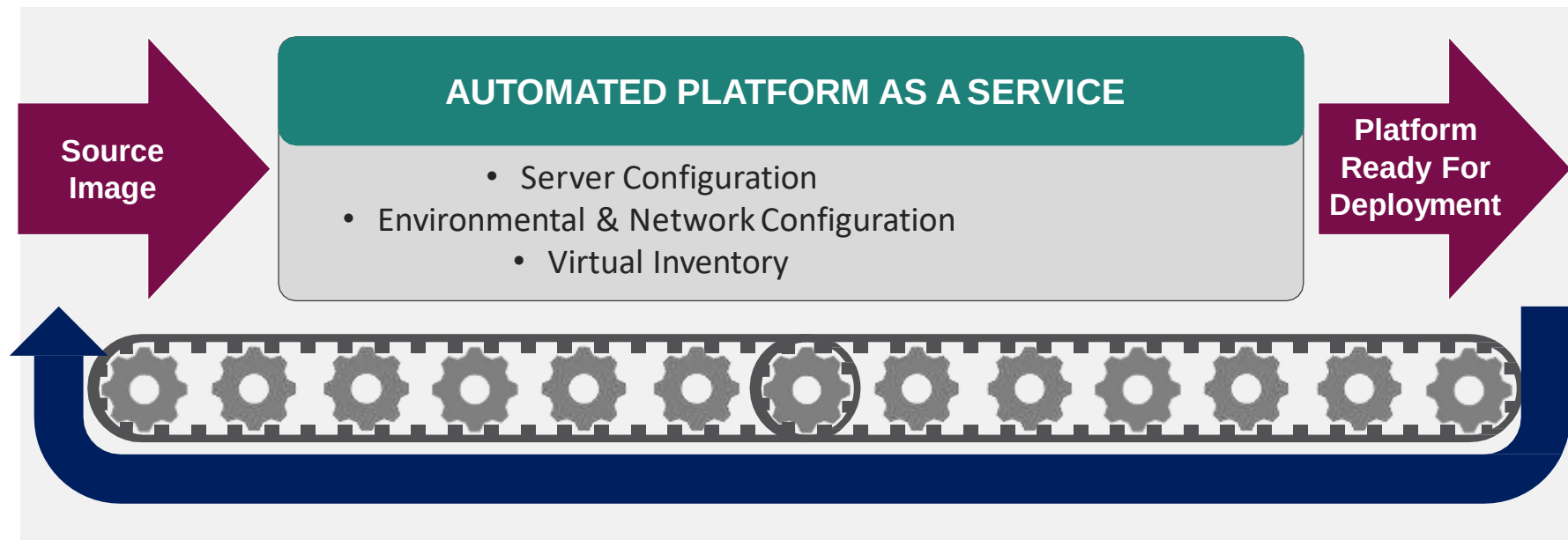


Automated Provisioning



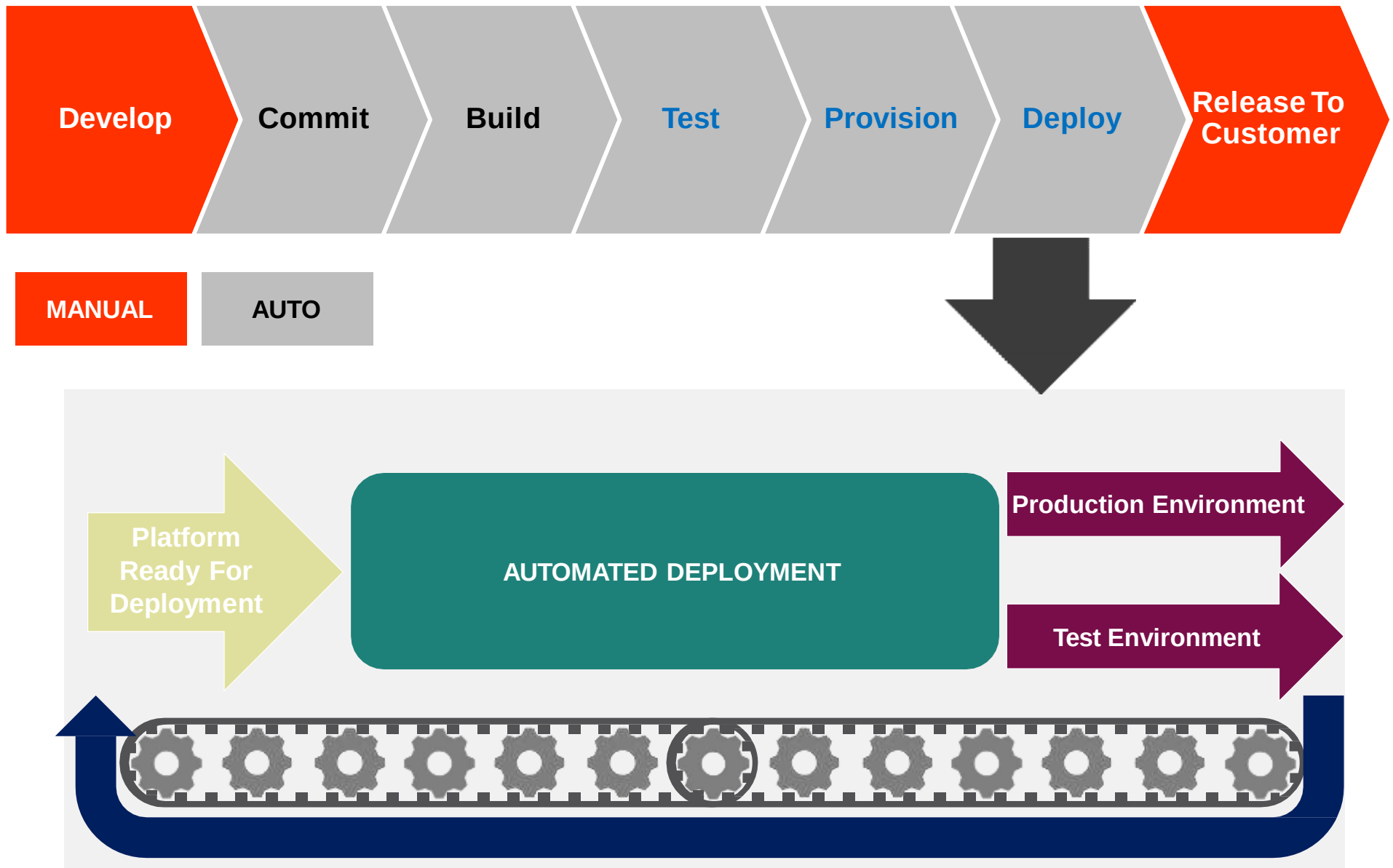
MANUAL

AUTO



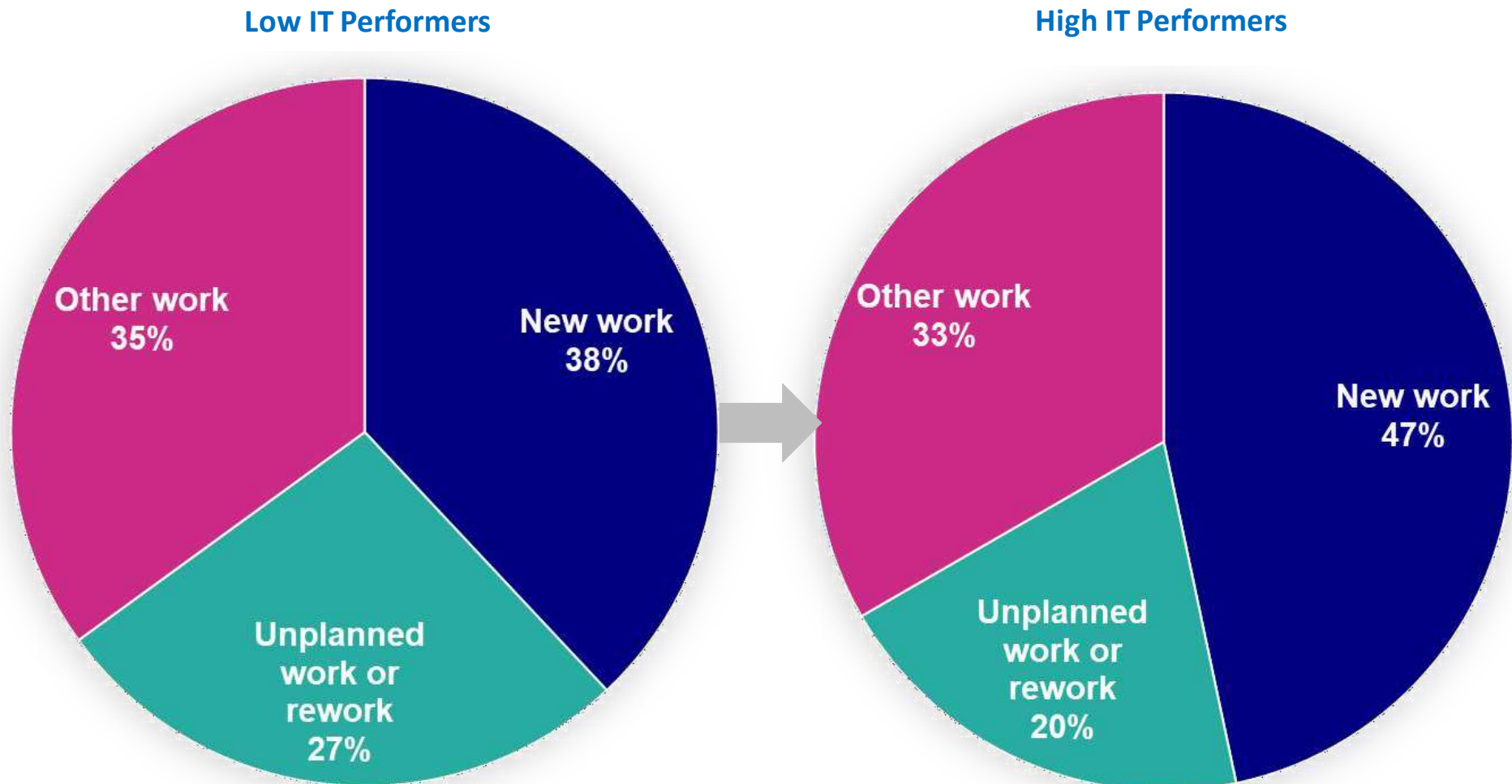


Automated Deployment





Automation Shifts Focus



Source: 2016 State of DevOps Report



Cloud Technology & Virtualization

“

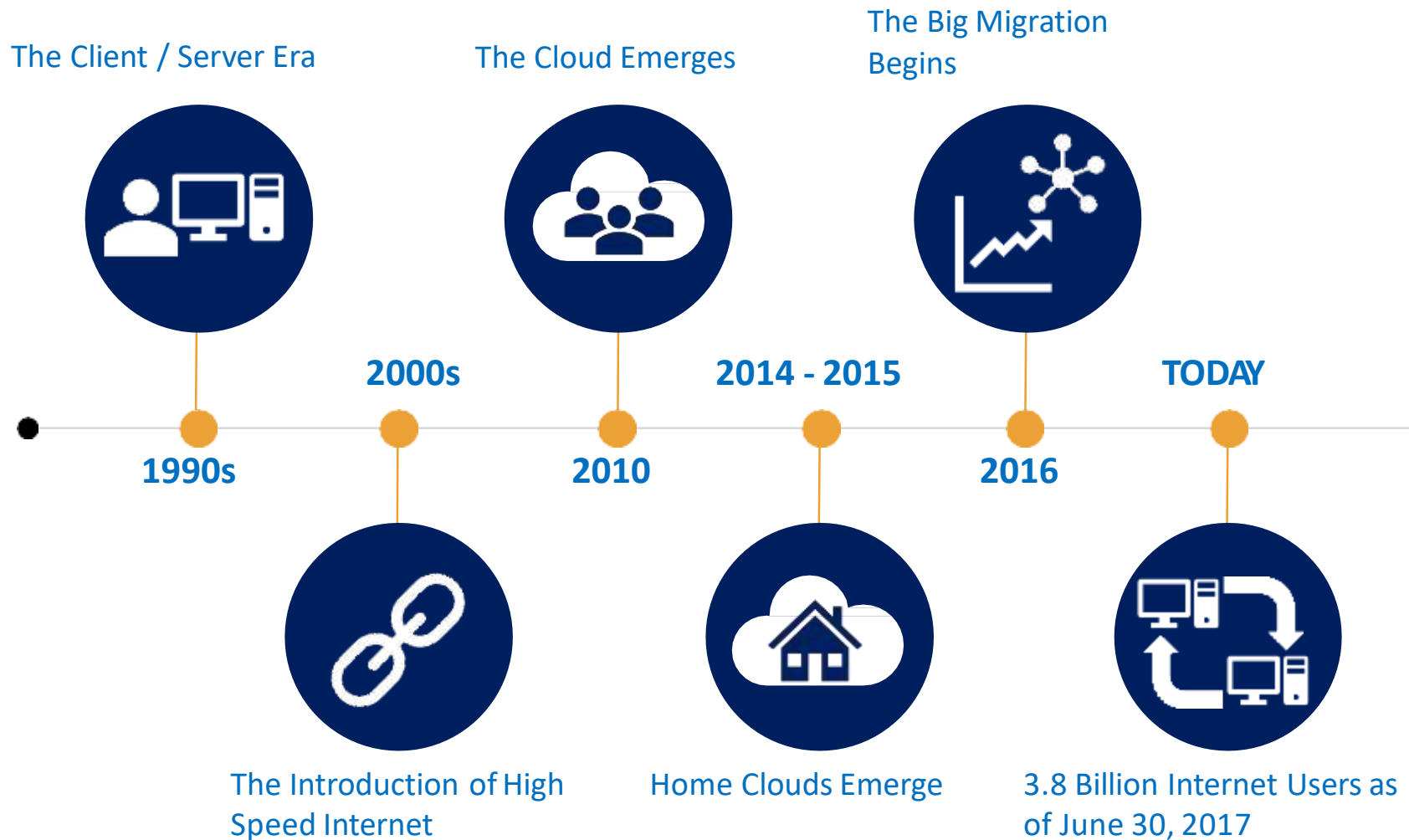
Cloud is about how you do computing, not where you do computing.

- Paul Maritz, CEO of VMware

”

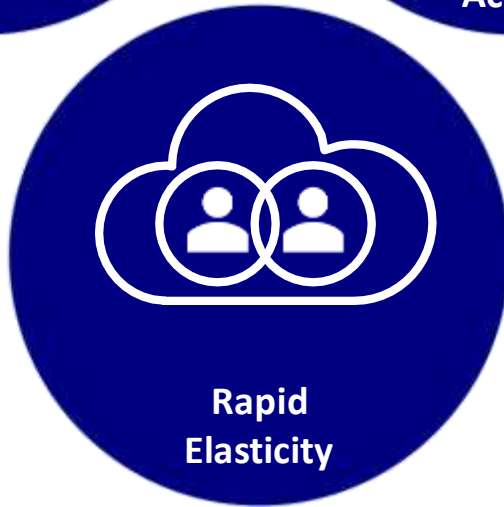
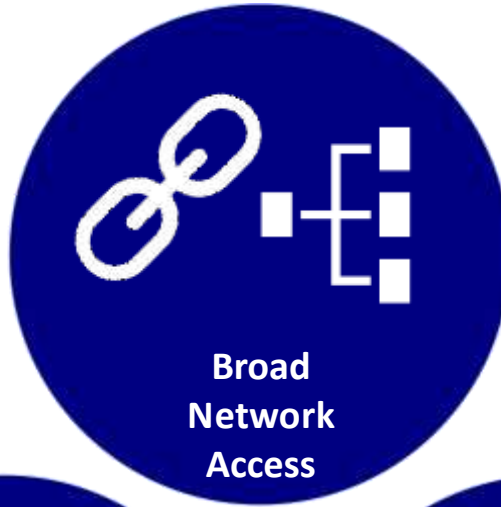


The Emergence Of Cloud Technology





Characteristics Of Cloud Computing

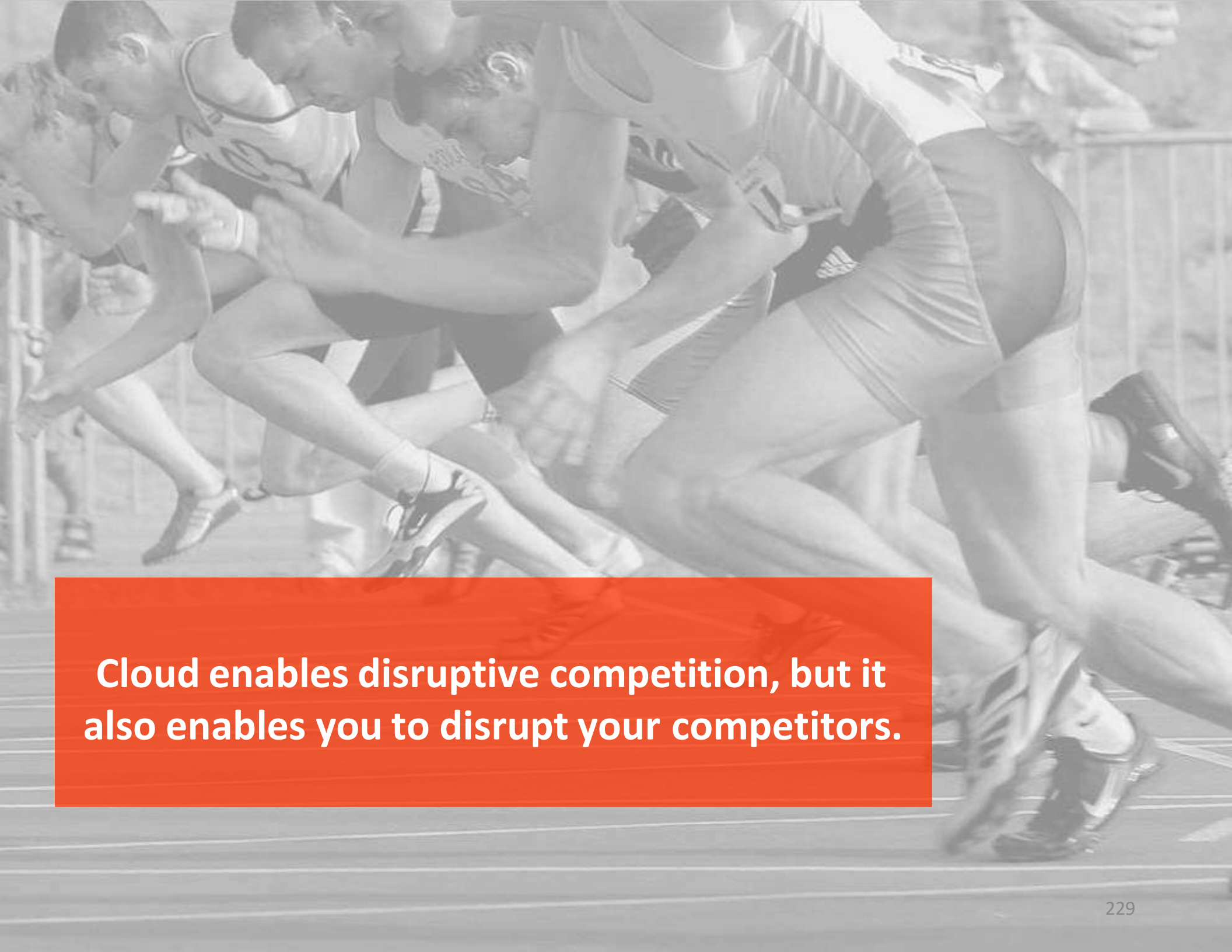


**Broad
Network
Access**

**Resource
Pooling**

**Rapid
Elasticity**

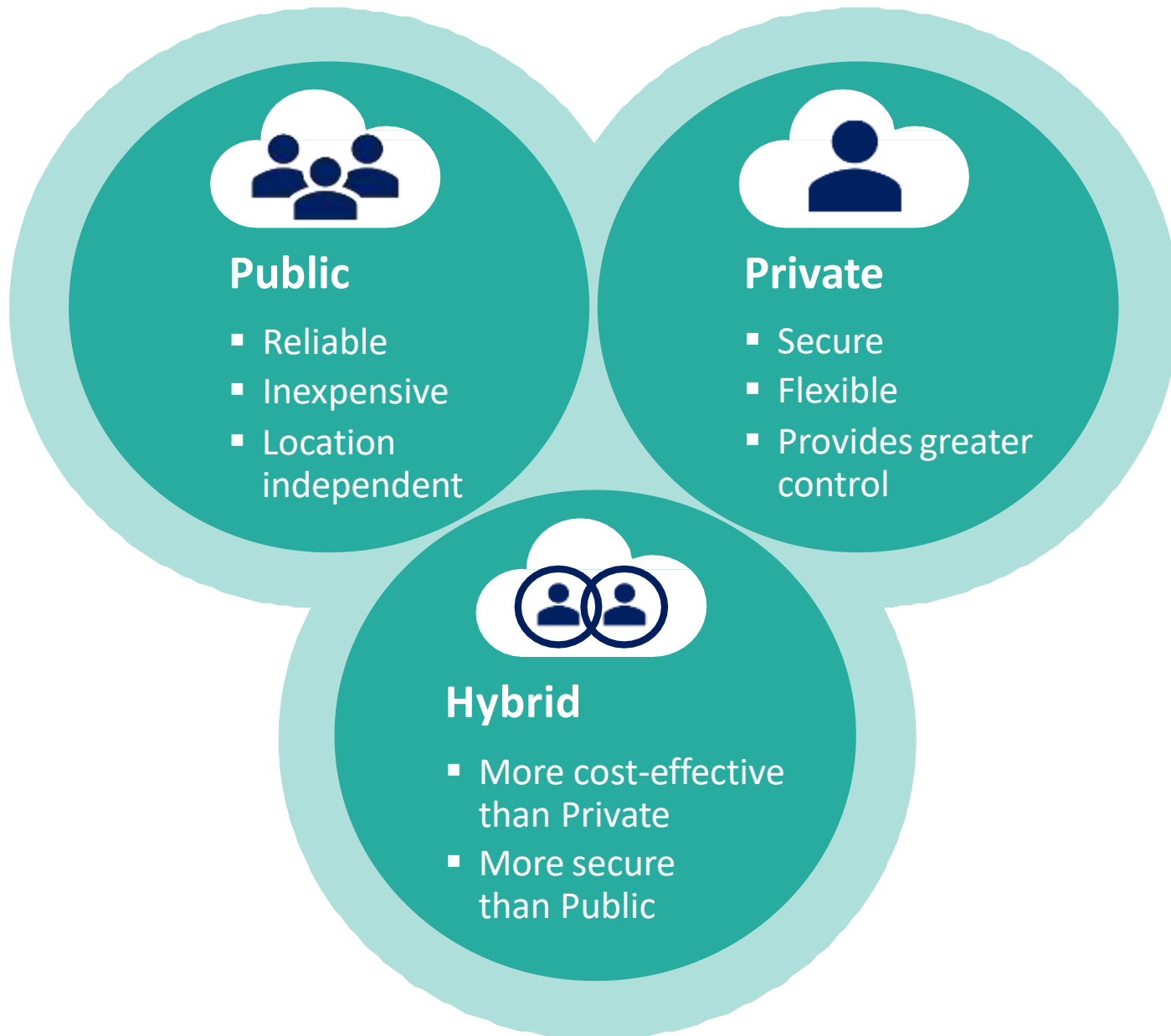
**Measured
Service**



Cloud enables disruptive competition, but it also enables you to disrupt your competitors.



Cloud Computing Deployment Models

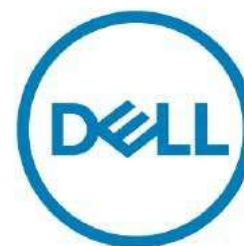




Public Cloud Computing Providers

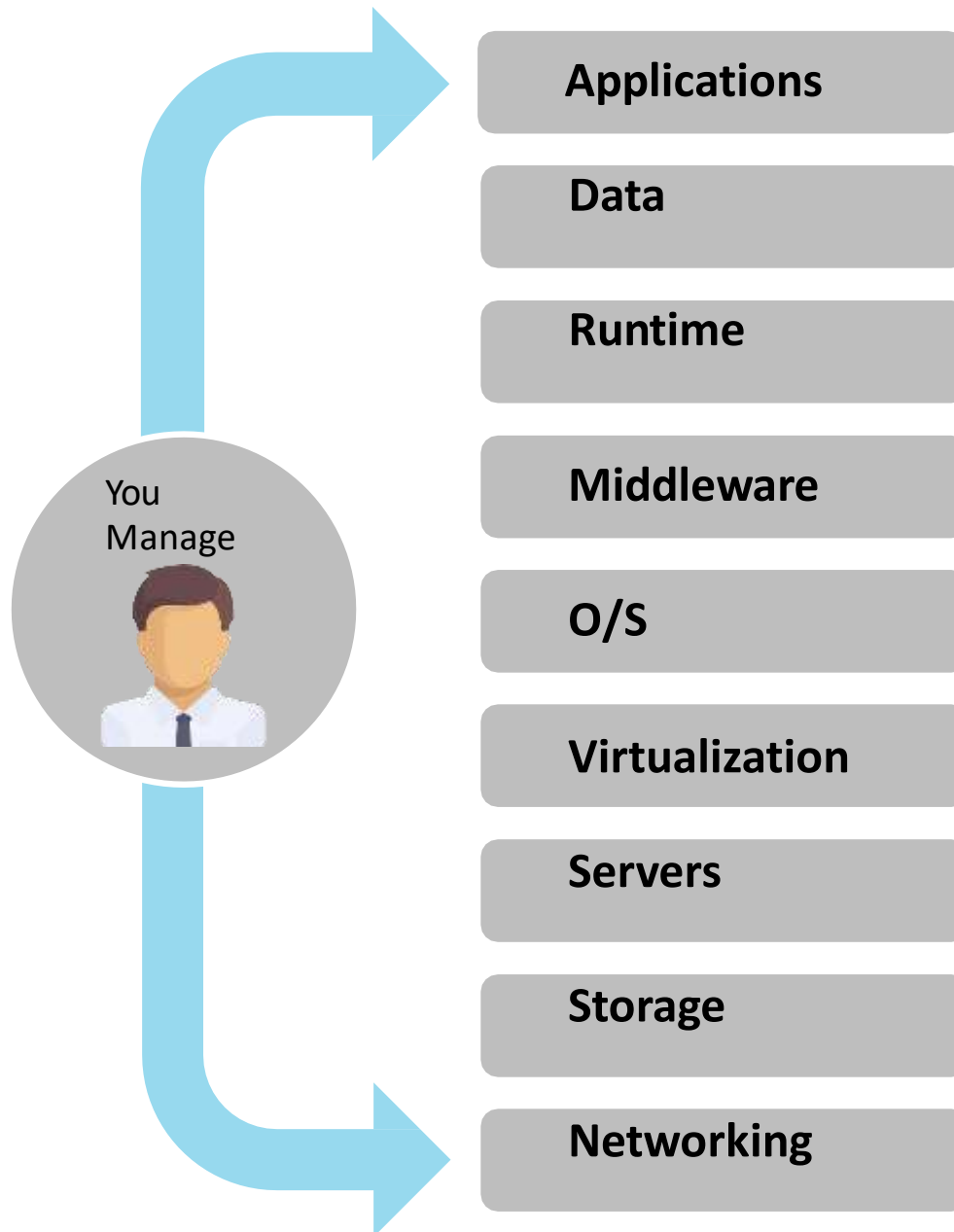


Microsoft Azure





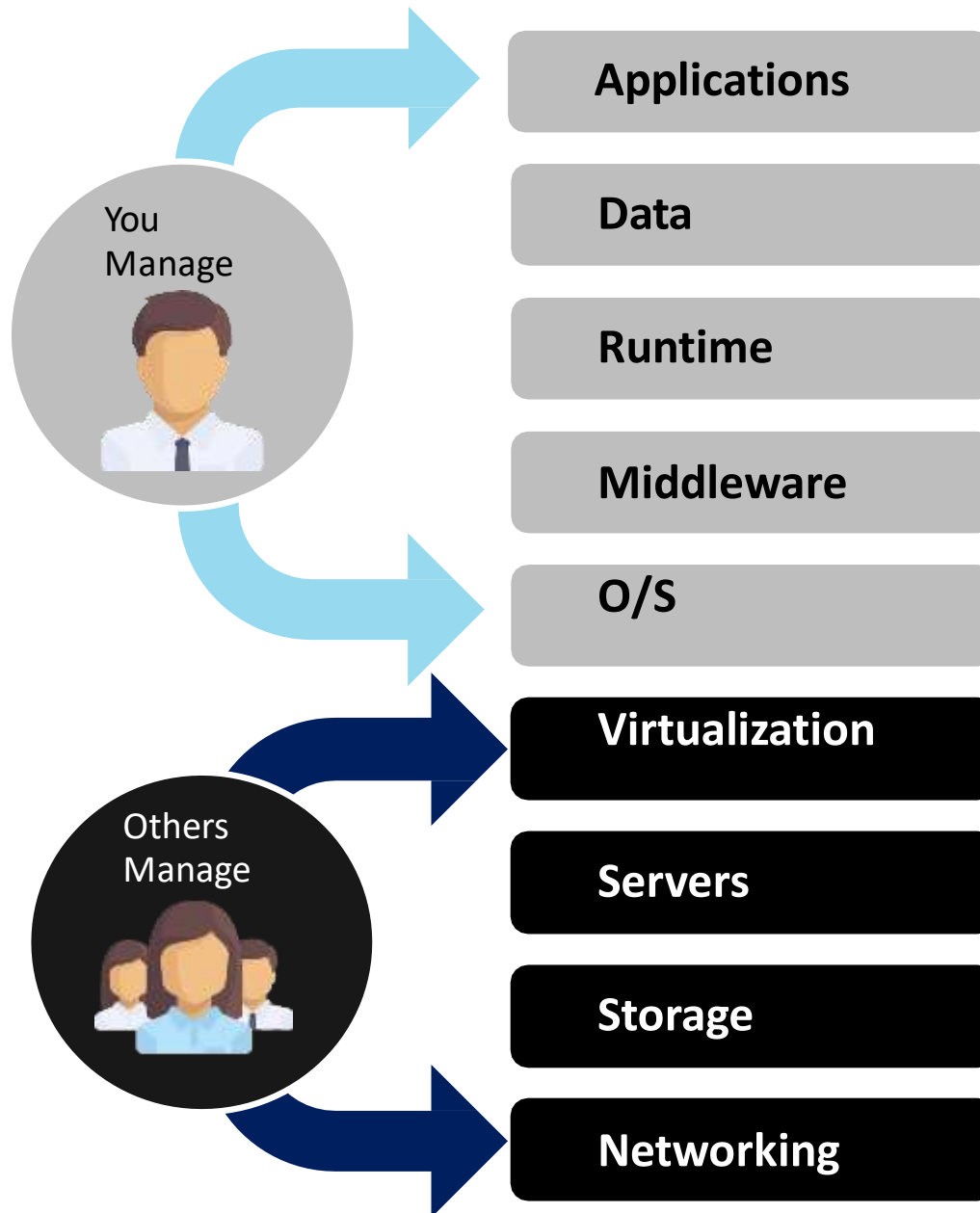
Traditional On-Premise Cloud Services



Source: "Cloud Computing Defined: Characteristics & Service Levels", IBM.com



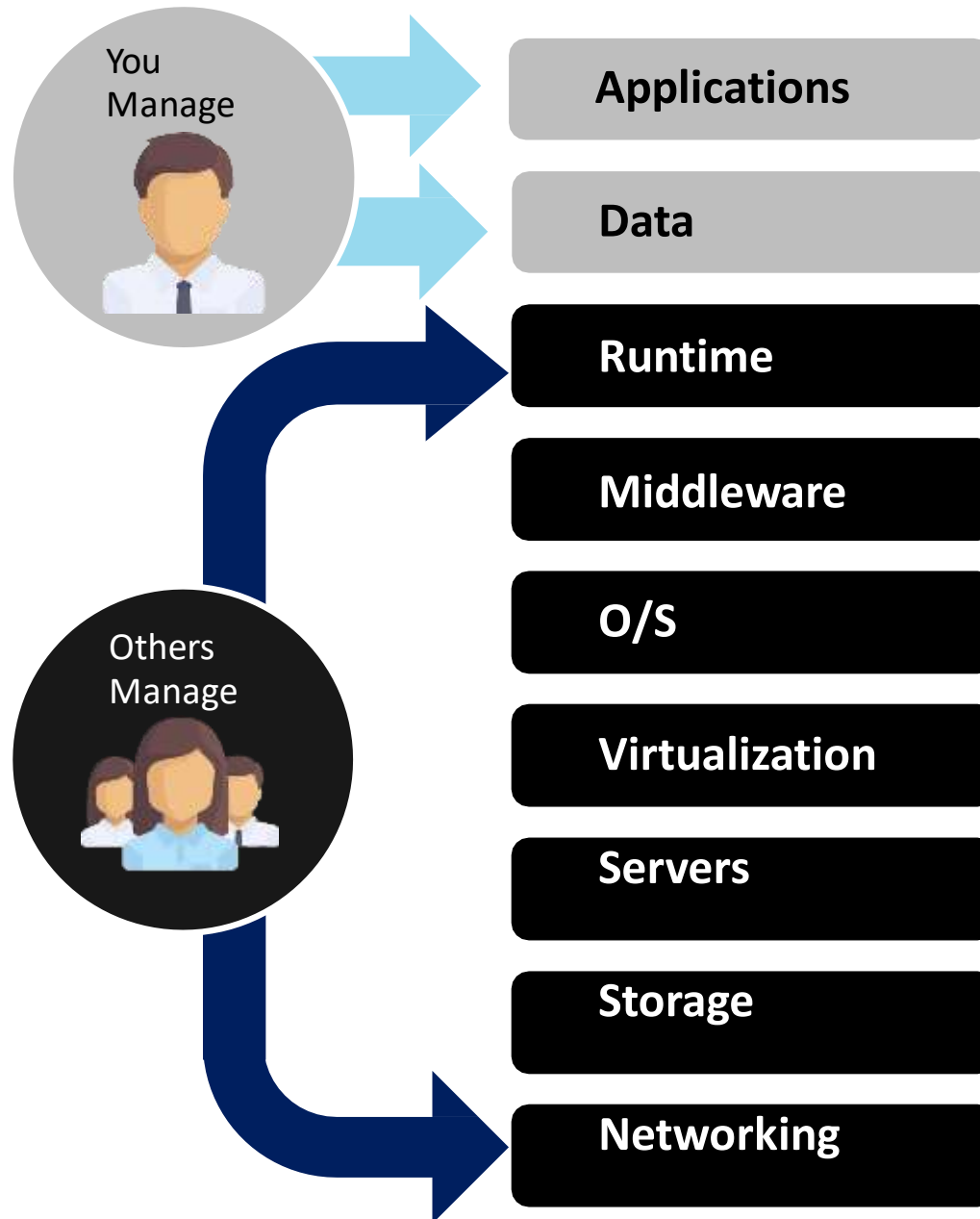
Infrastructure As A Service (IaaS)



Source: "Cloud Computing Defined: Characteristics & Service Levels", IBM.com



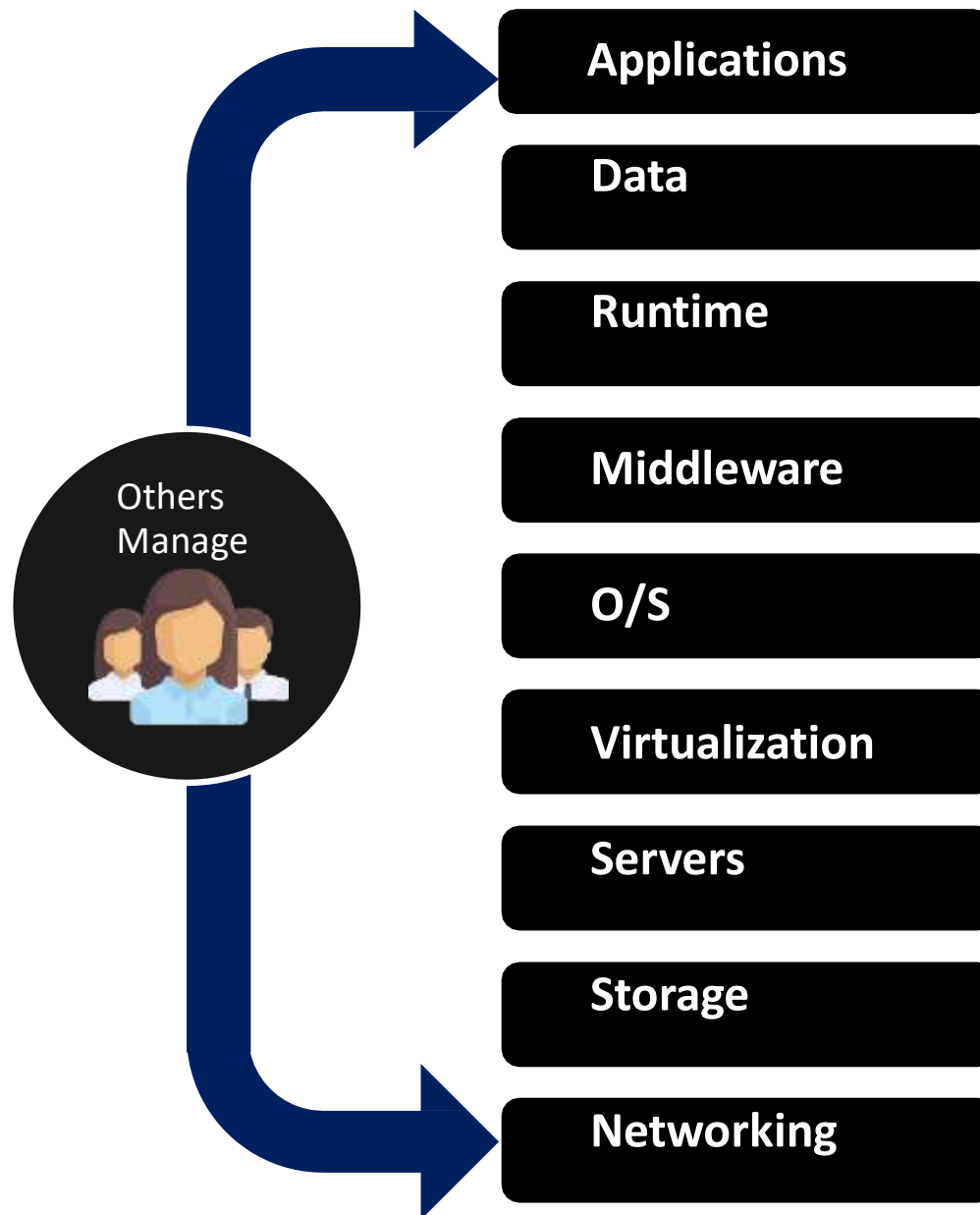
Platform As A Service (PaaS)



Source: "Cloud Computing Defined: Characteristics & Service Levels", IBM.com



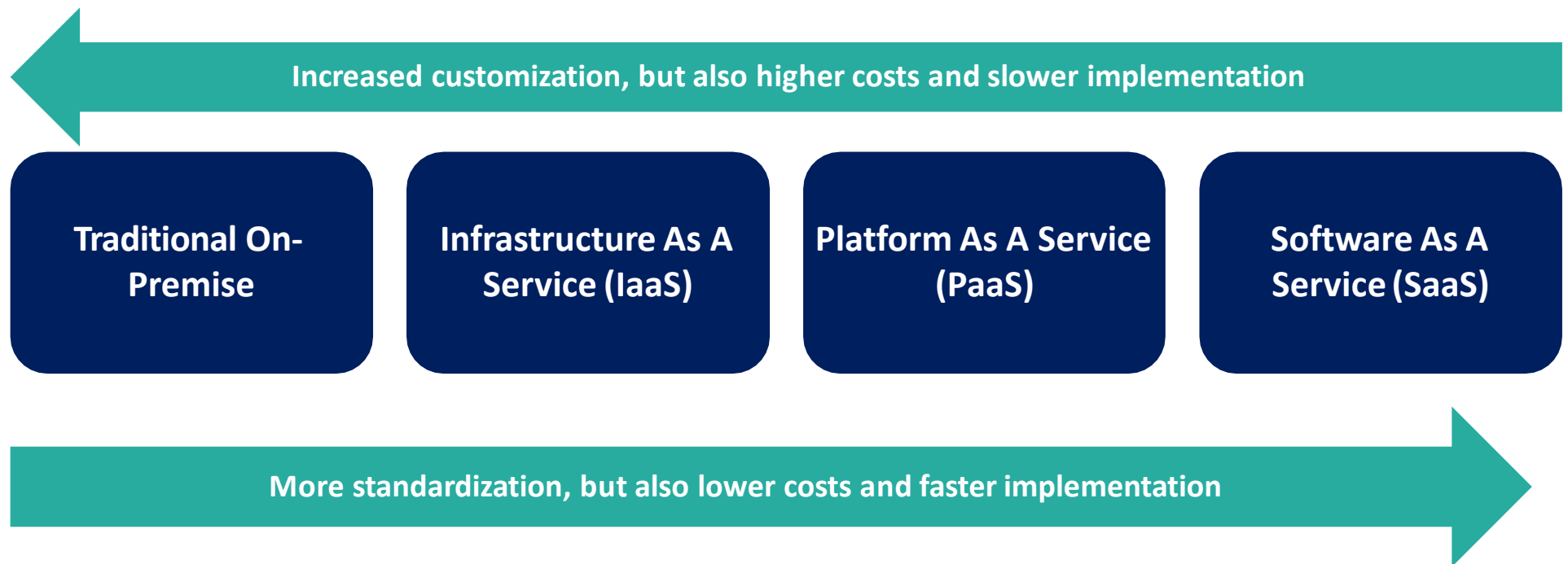
Software As A Service (SaaS)



Source: "Cloud Computing Defined: Characteristics & Service Levels", IBM.com

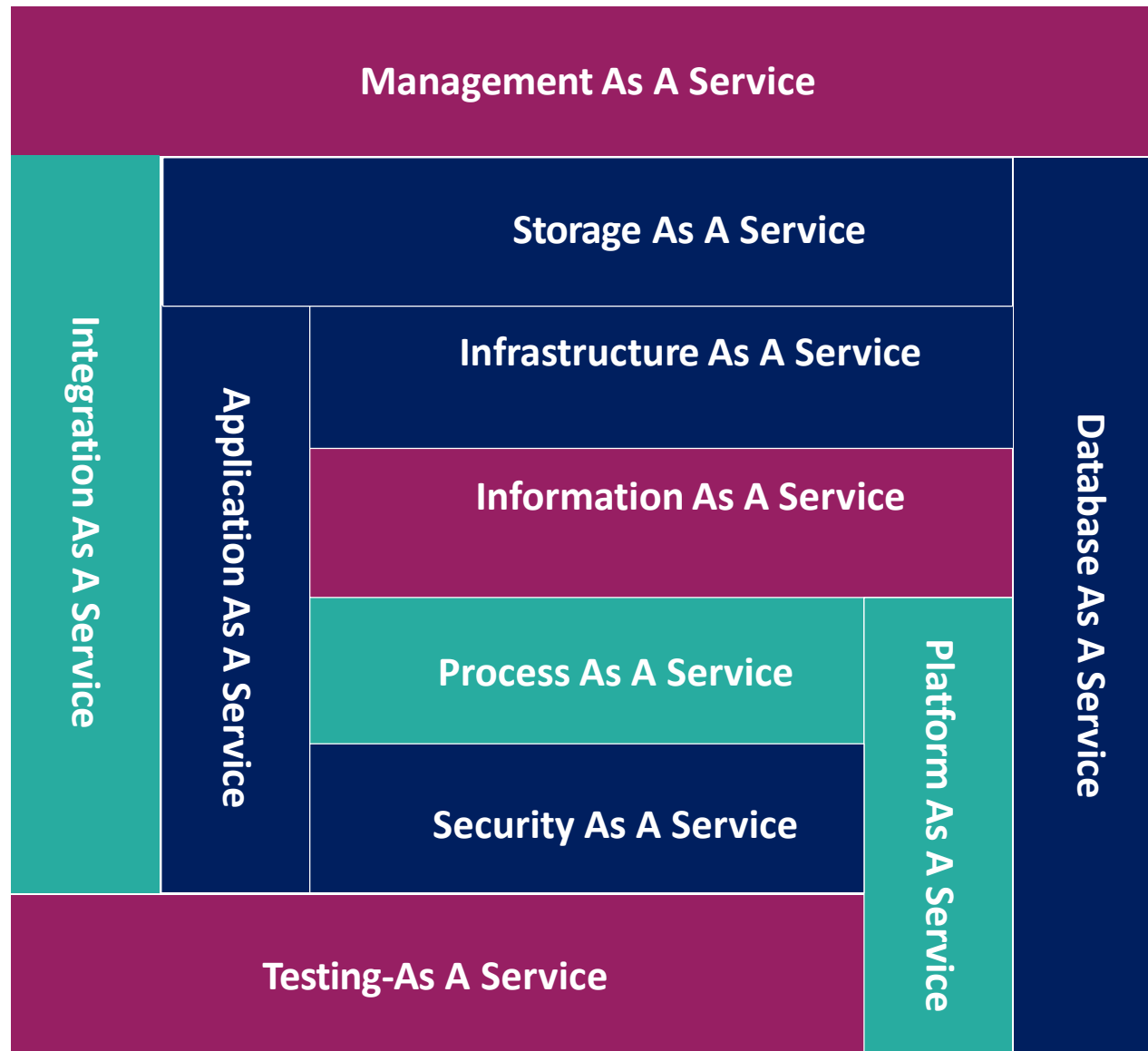


Cloud Computing Service Models



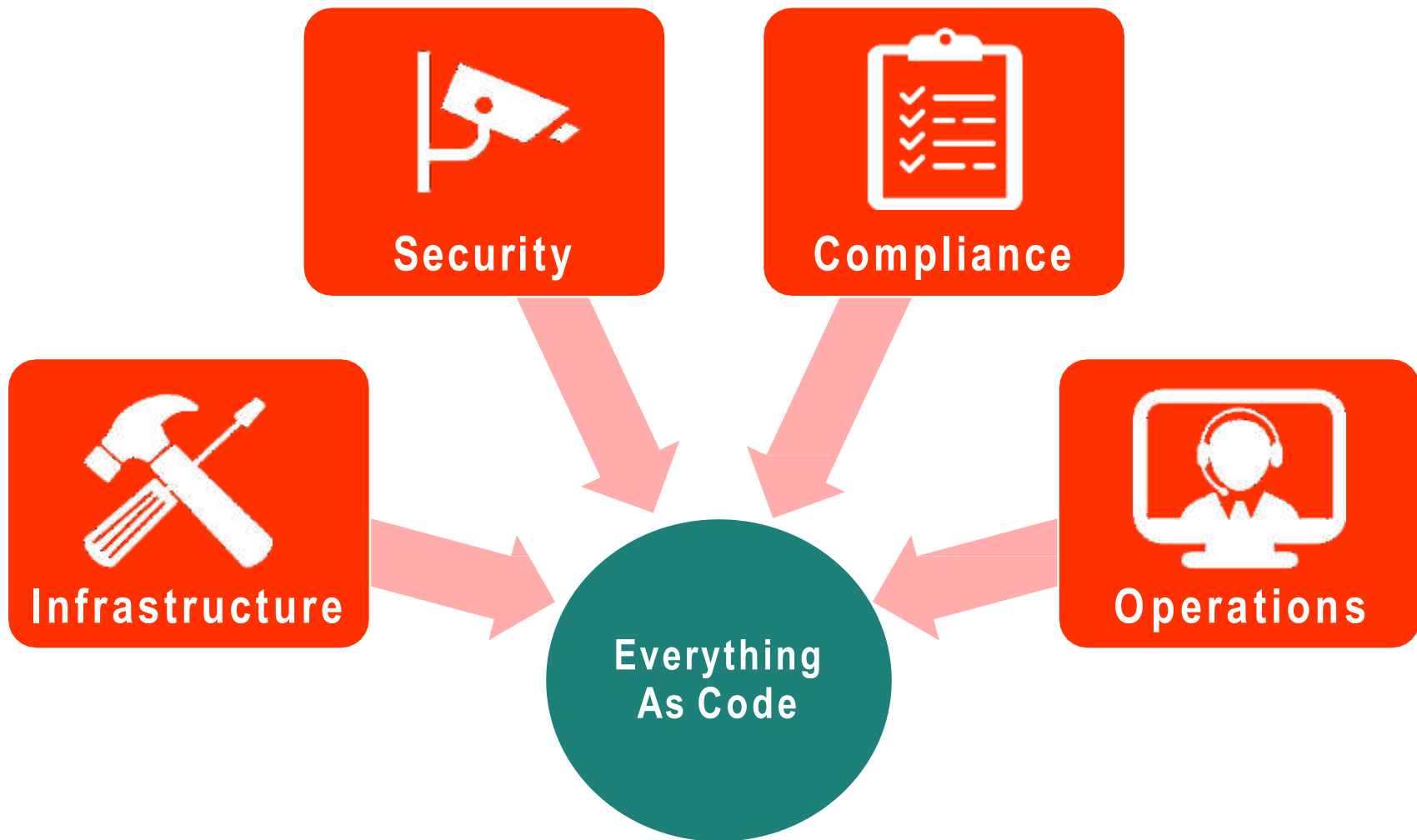


Everything As A Service (XaaS)





Everything As Code

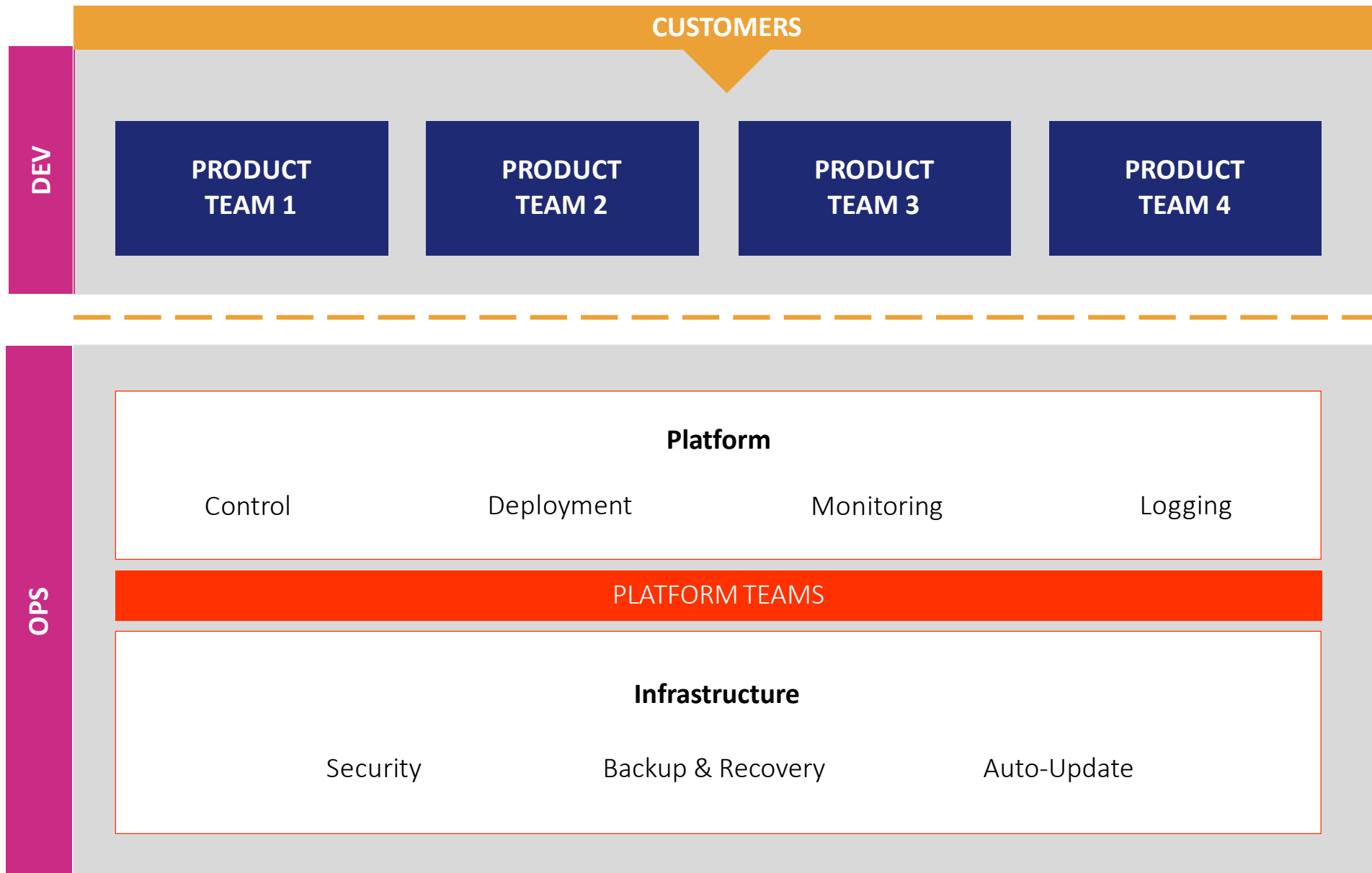




Architecting For Continuous Delivery

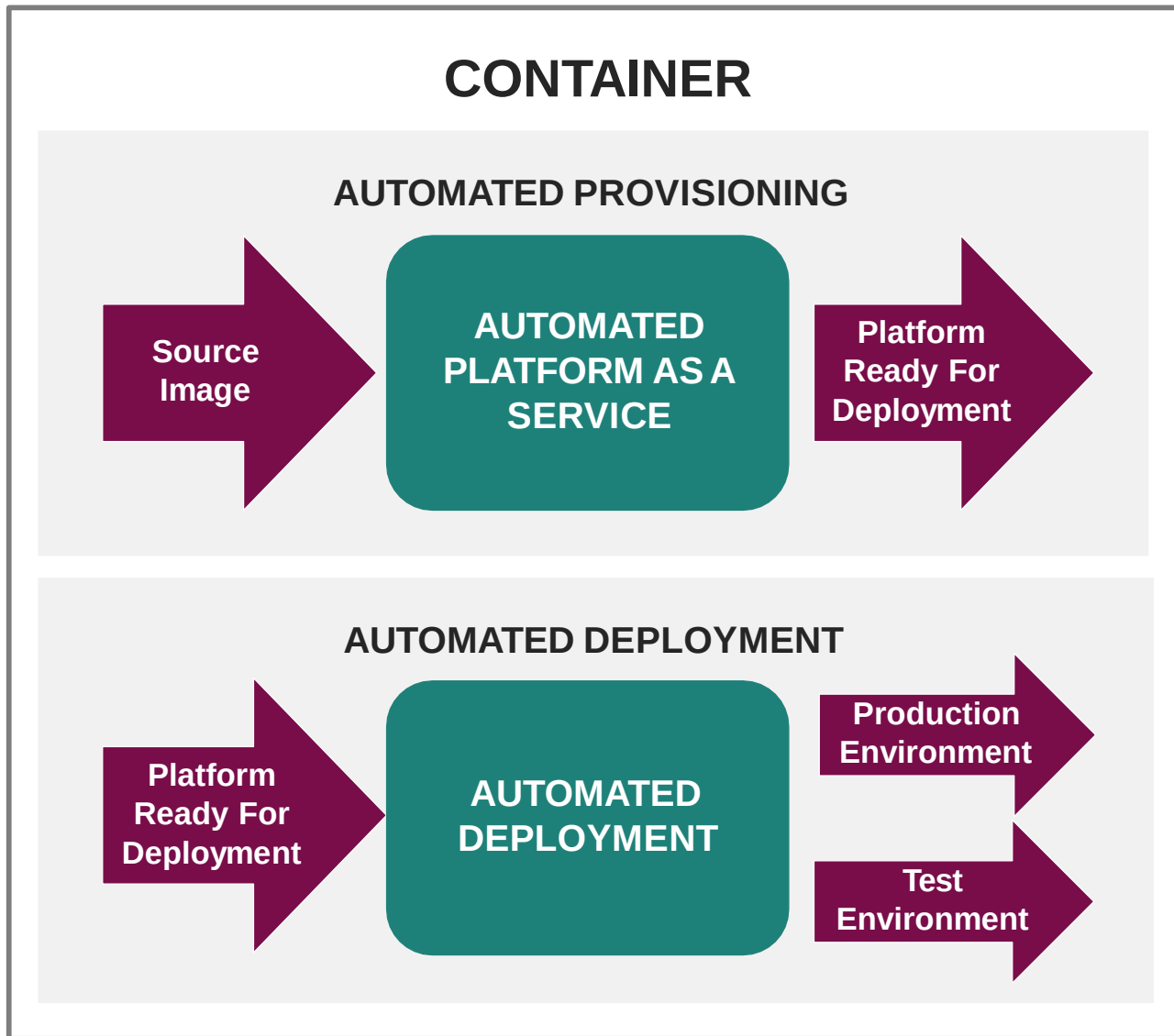


Product & Platform Teams





Containerization





Microservice Architectures

Service-Oriented Architecture (SOA)

All functions in a single monolithic process.

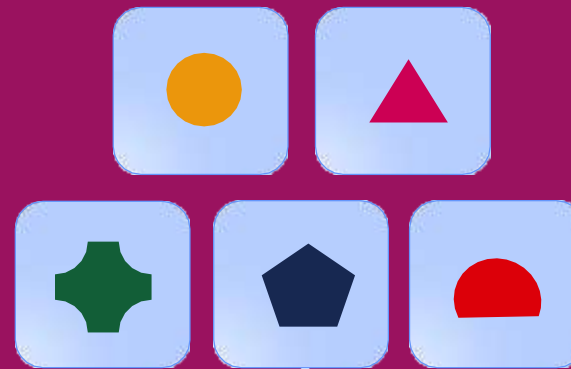


Scaled by replicating entire monolith across nodes.



Microservices Architecture (MSA)

One function, One microservice.

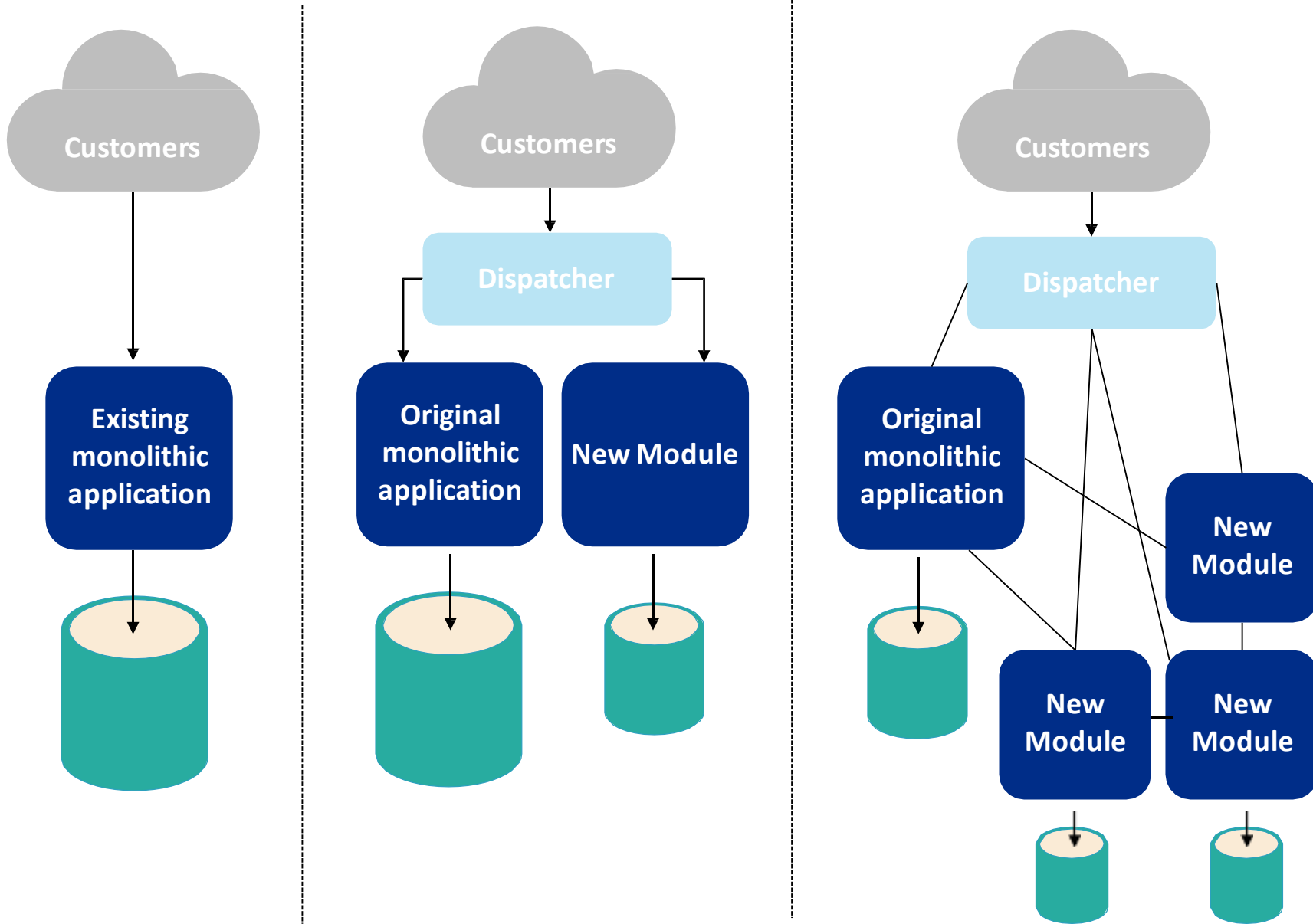


Scaled by distributing services across nodes.





Strangler Application Pattern

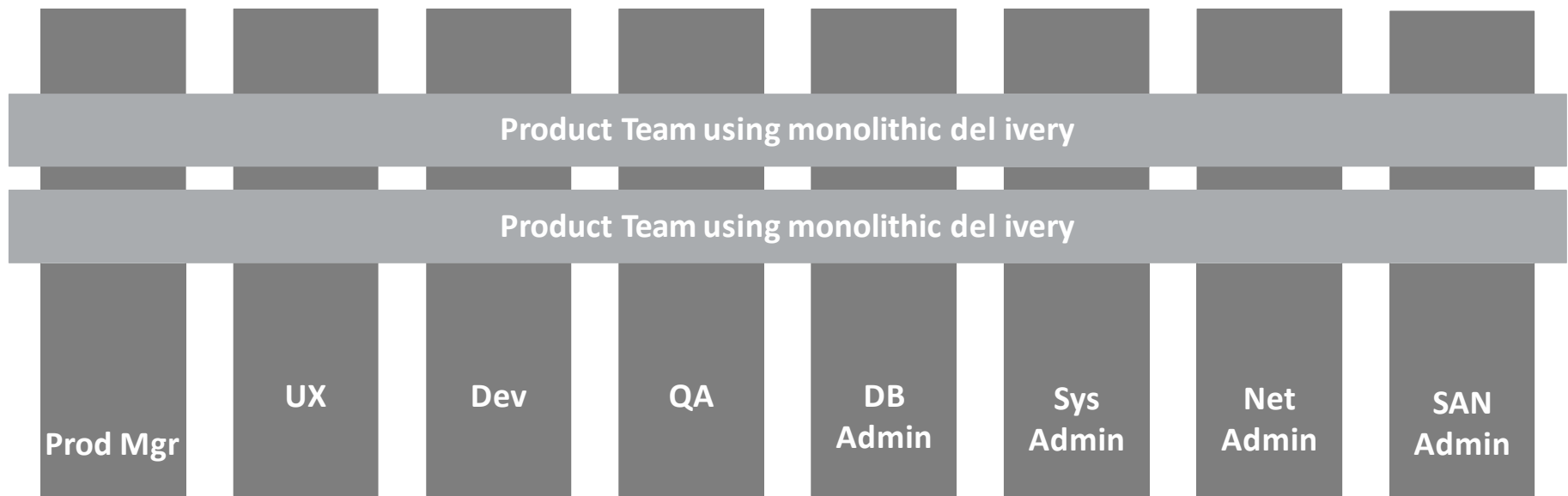


<https://youtu.be/EVvnBgSX04c>

Source: ContinuousDelivery.com

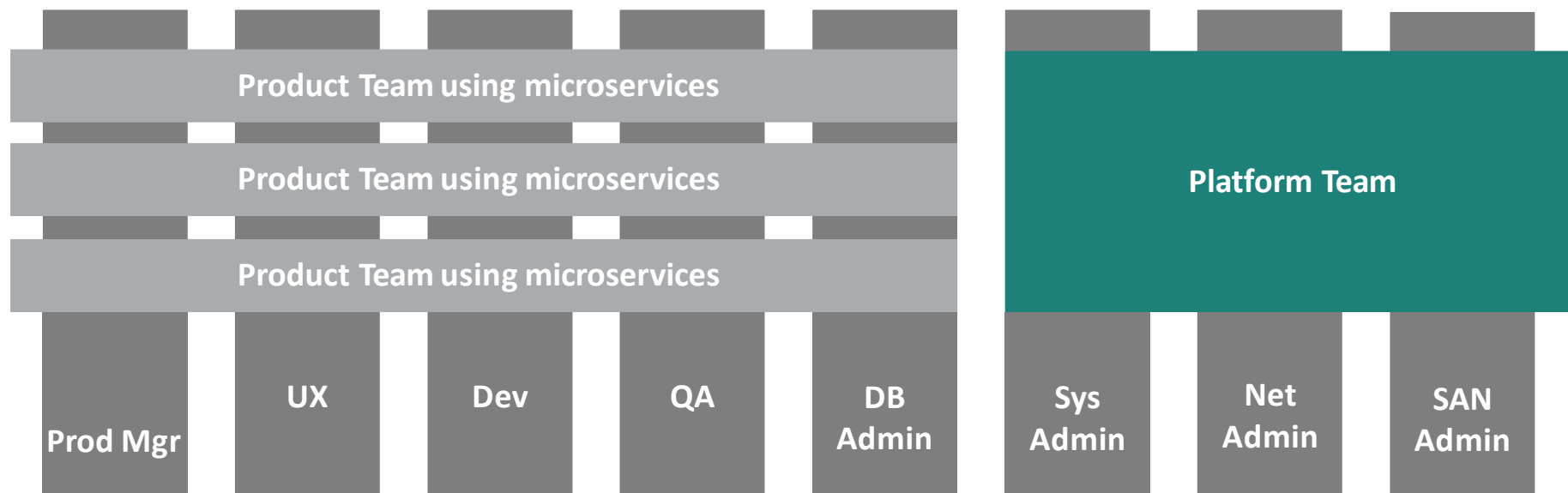


Microservices & Teaming





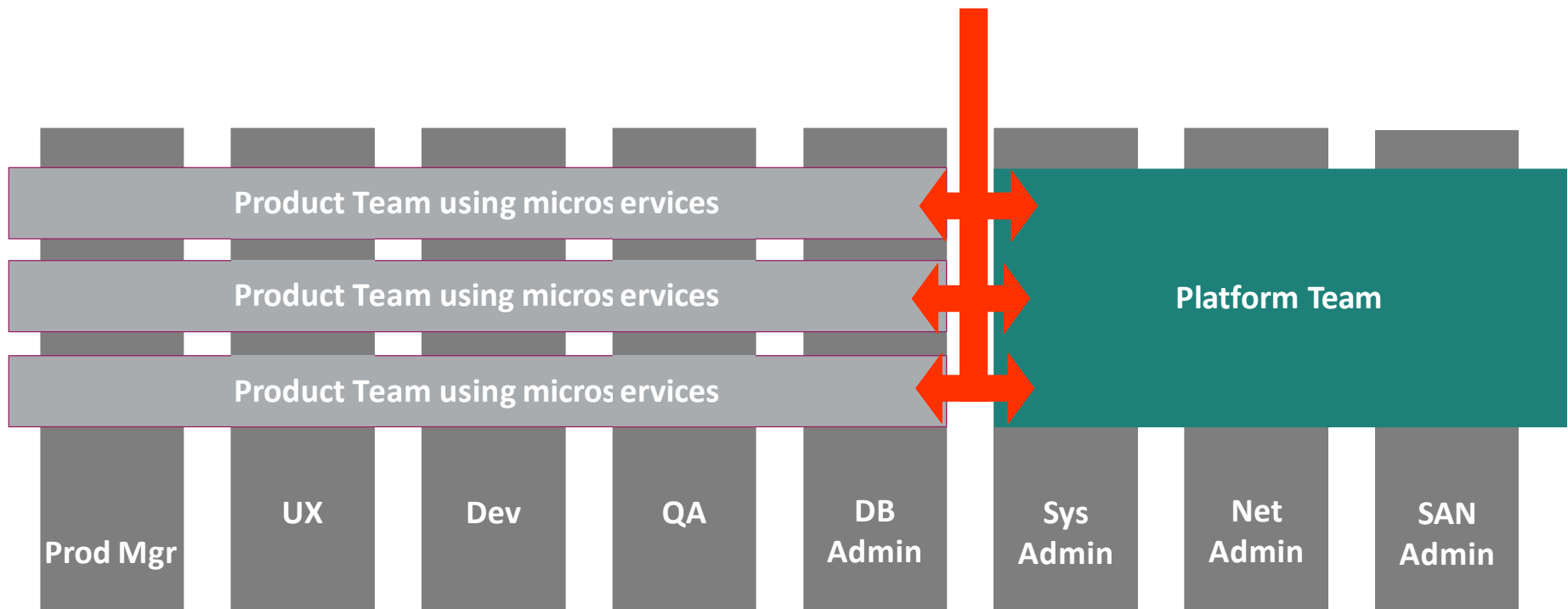
Microservices & Teaming





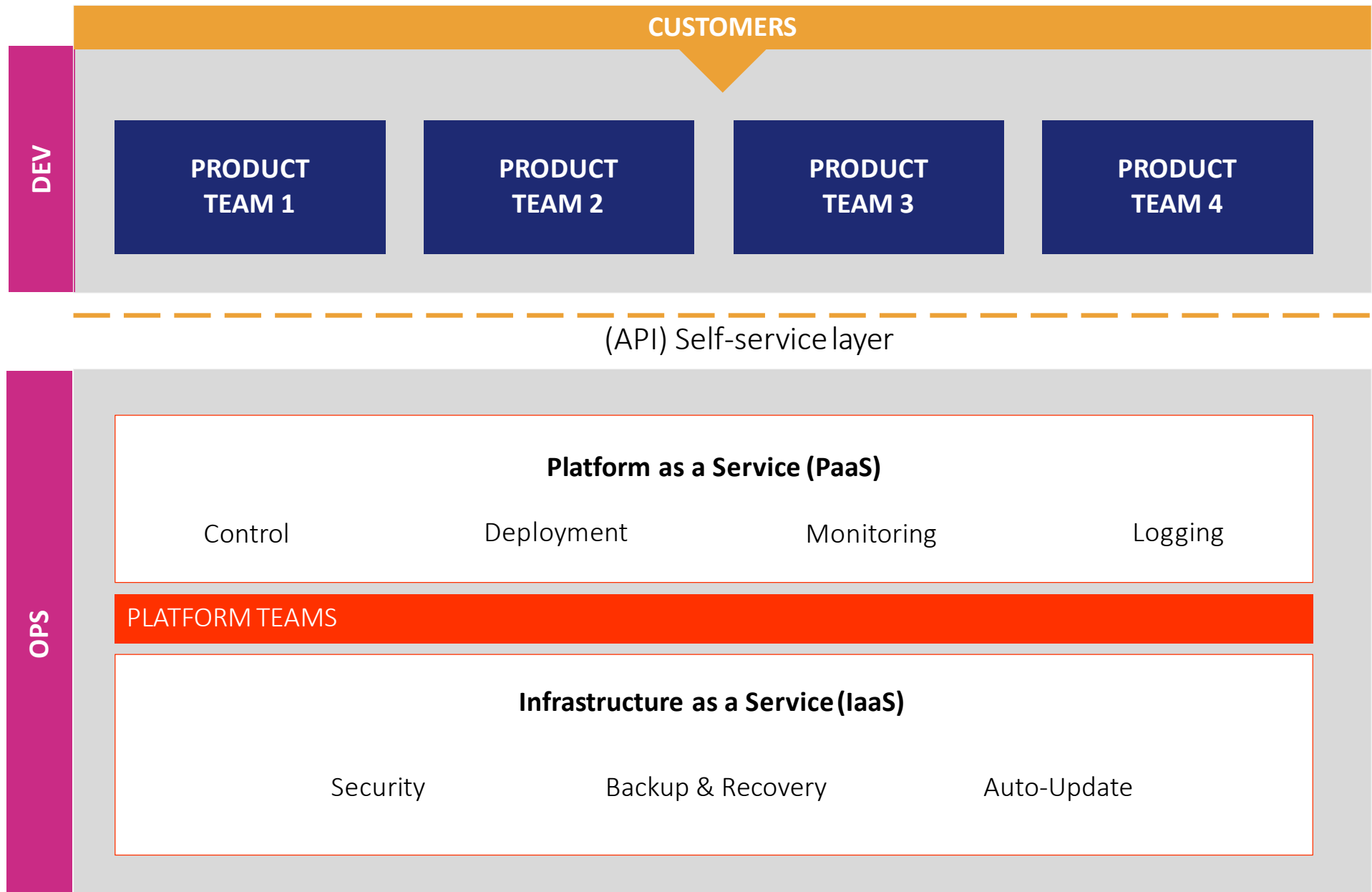
Application Programming Interfaces (APIs)

Application Programming Interfaces (APIs) provide a set of clearly defined methods of communication between the Product and Platform teams. They simplify the relationship between the teams by standardizing and automating many of the routine functions that the teams interact to complete.





Product & Platform Teams





PeopleCert DevOps Fundamentals

Wrap Up & Review



Goals Of DevOps

Organizational Goals

Faster time to market

Better integration between IT
and the business

Improved responsiveness

Improved code and deployment quality

Increased productivity

Better visibility and transparency

Increased flexibility and agility

IT Goals

Faster delivery of features

Increased collaboration and communication

Faster resolution of problems

Continuous software delivery

More productive teams

Reduced complexity to manage

More time to innovate

Ultimately, the goal of IT is to provide the value that the business requires.

DevOps addresses the business problem because it addresses the IT value delivery problem.

It breaks down the “wall of confusion” between Dev and Ops.



DevOps Transforms Processes & Practices

#1

**Voice Of The
Customer**

#2

**Relationship
Management**

#3

**Lean Process
Optimization**

#4

**Value Stream
Mapping**

#5

**Knowledge
Management**

#6

**Visual
Management**

#7

Agile Scrum

#8

**Shift Left
Testing**

#9

**Change
Control**

#10

**Service
Configuration
Management**

#11

**Release &
Deployment
Management**

#12

**Incident
Management**

#13

**Problem
Management
& Kaizen**

#14

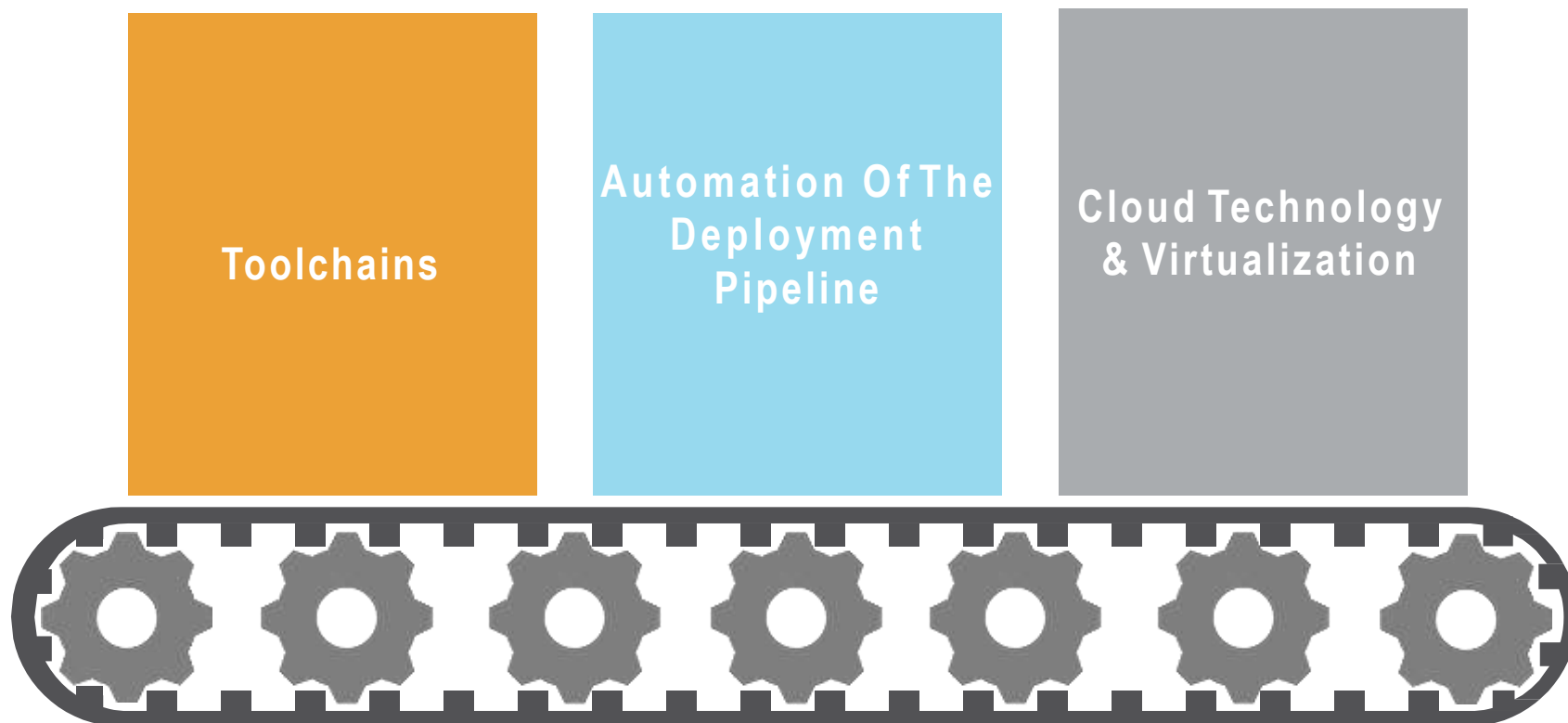
**Continual
Improvement**

#15

Antifragility

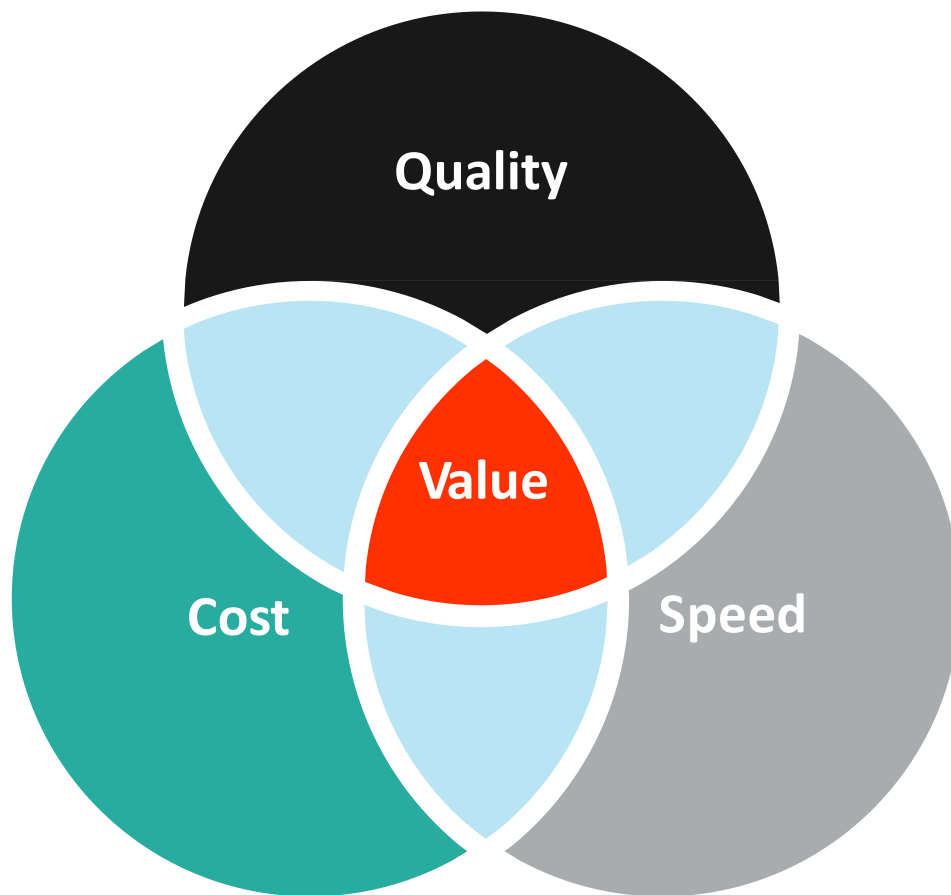


DevOps Transforms Technology & Automation



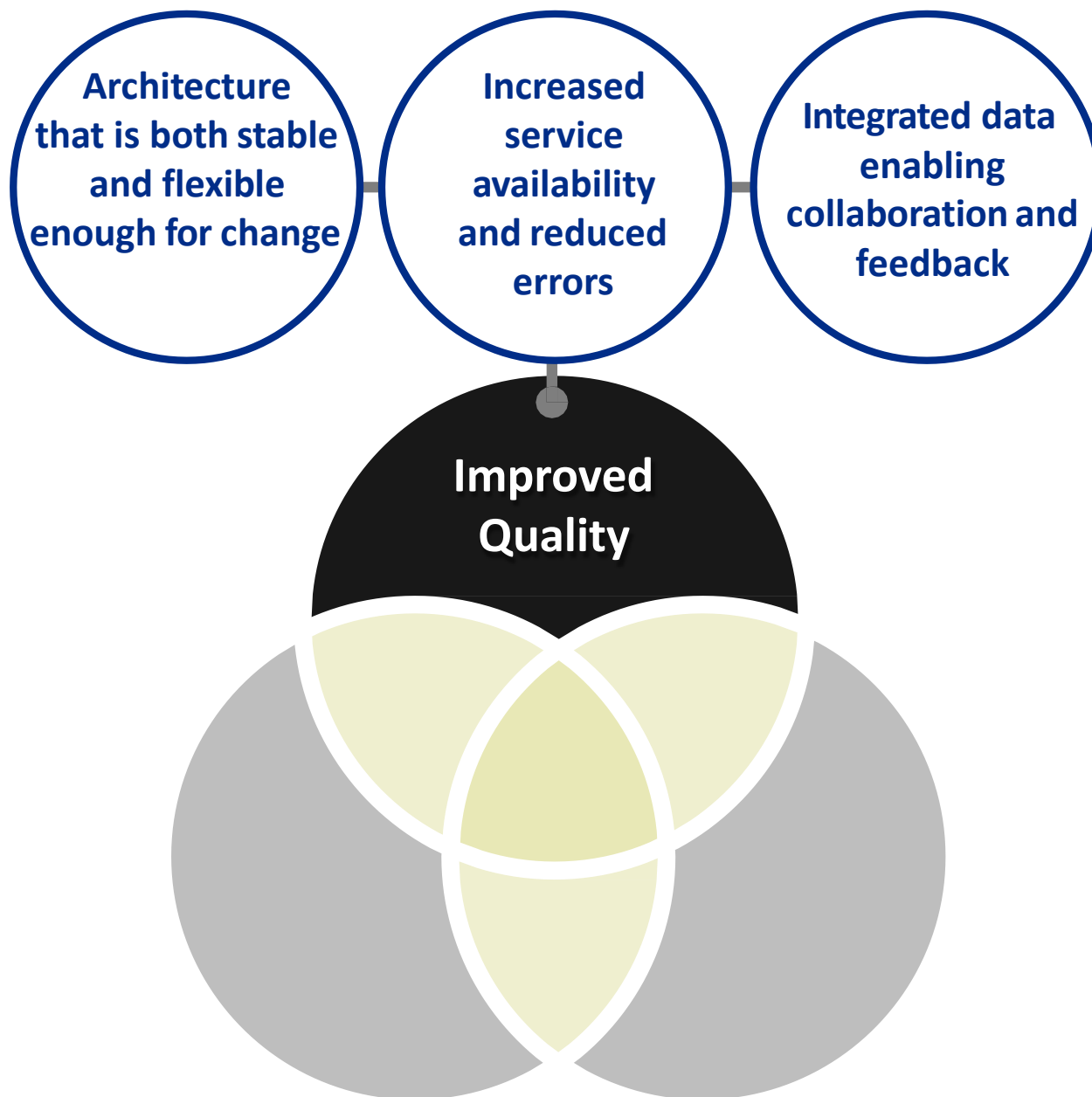


DevOps Delivers Value



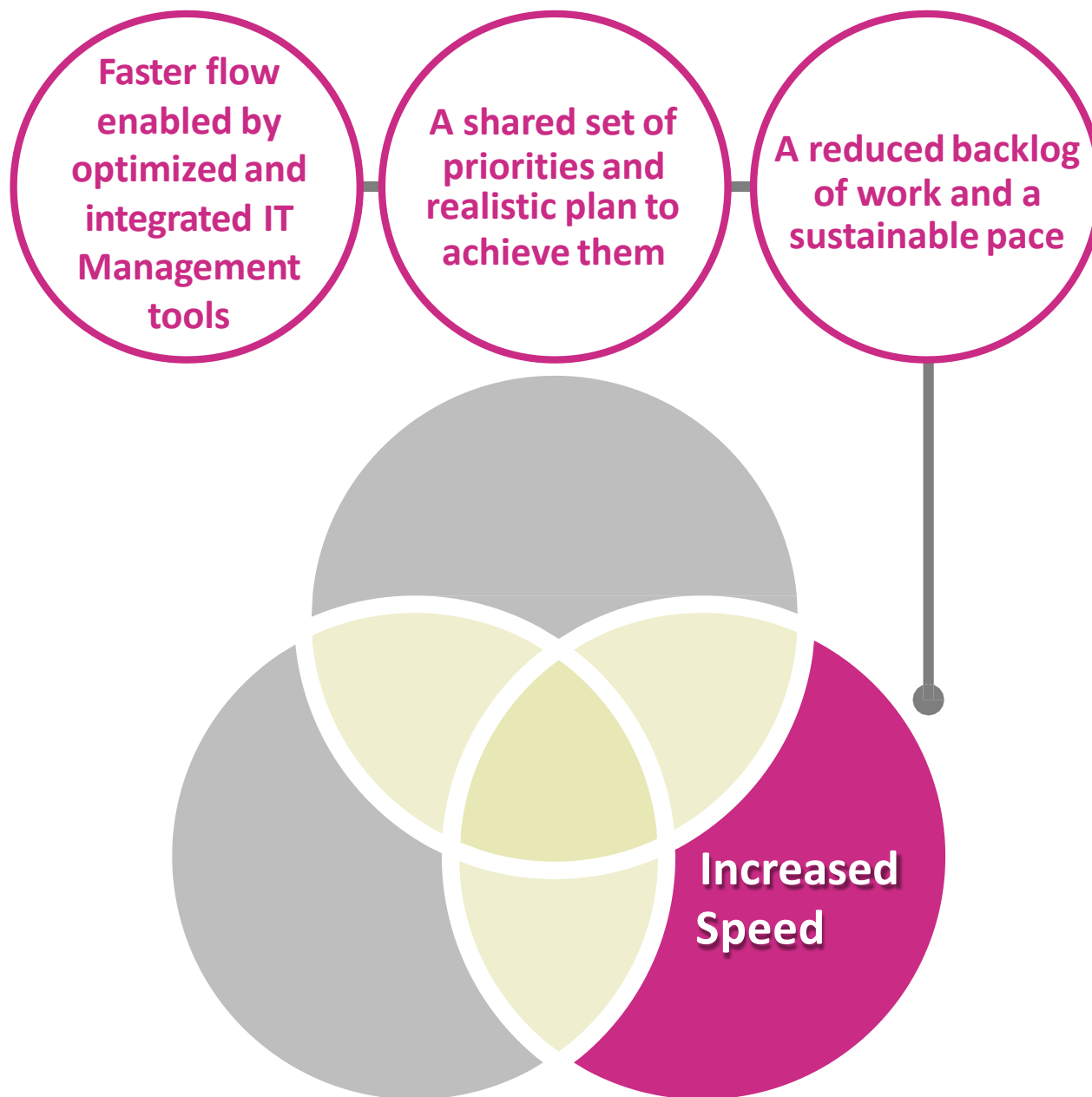


DevOps Improves Quality



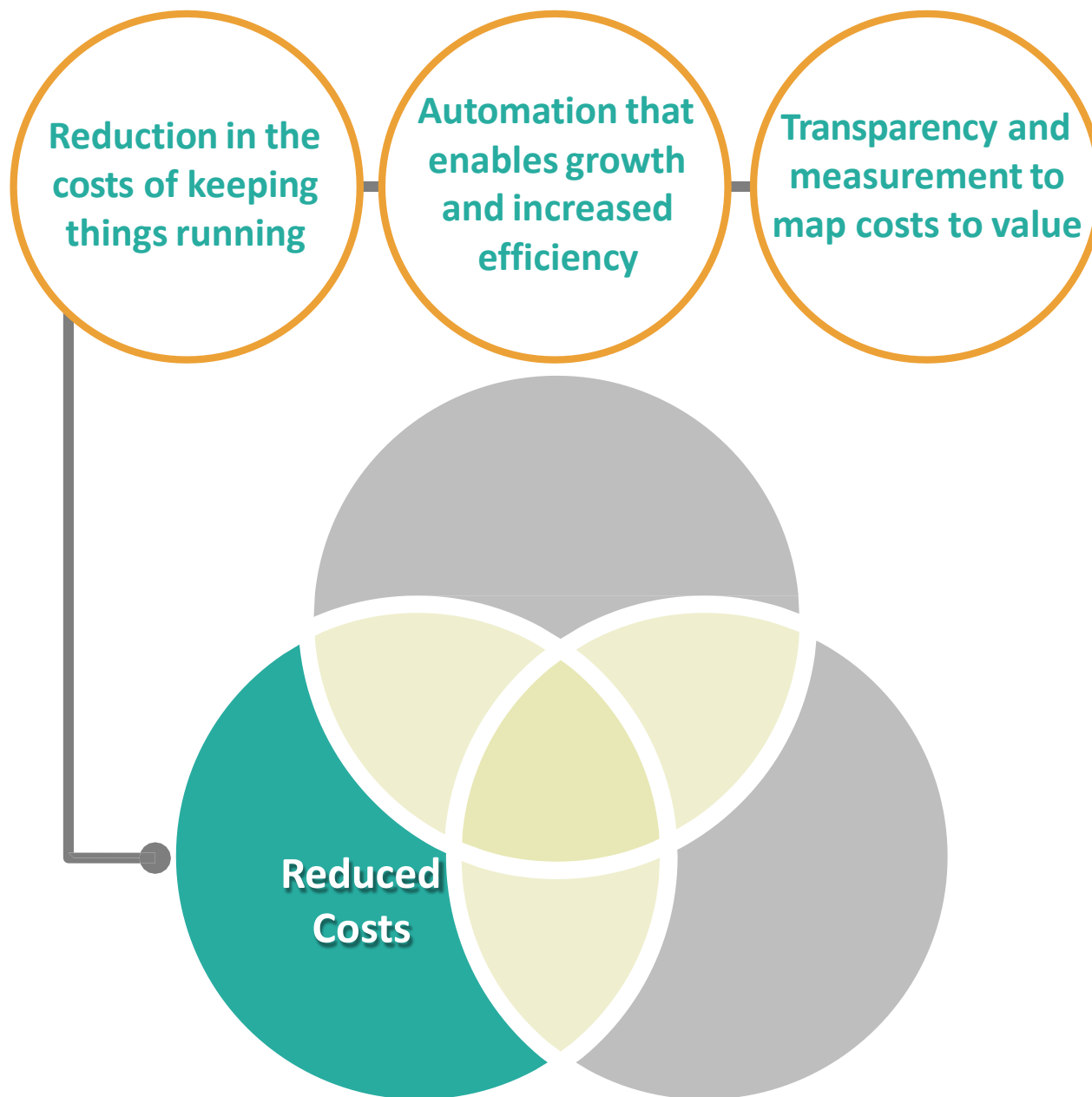


DevOps Increases Speed



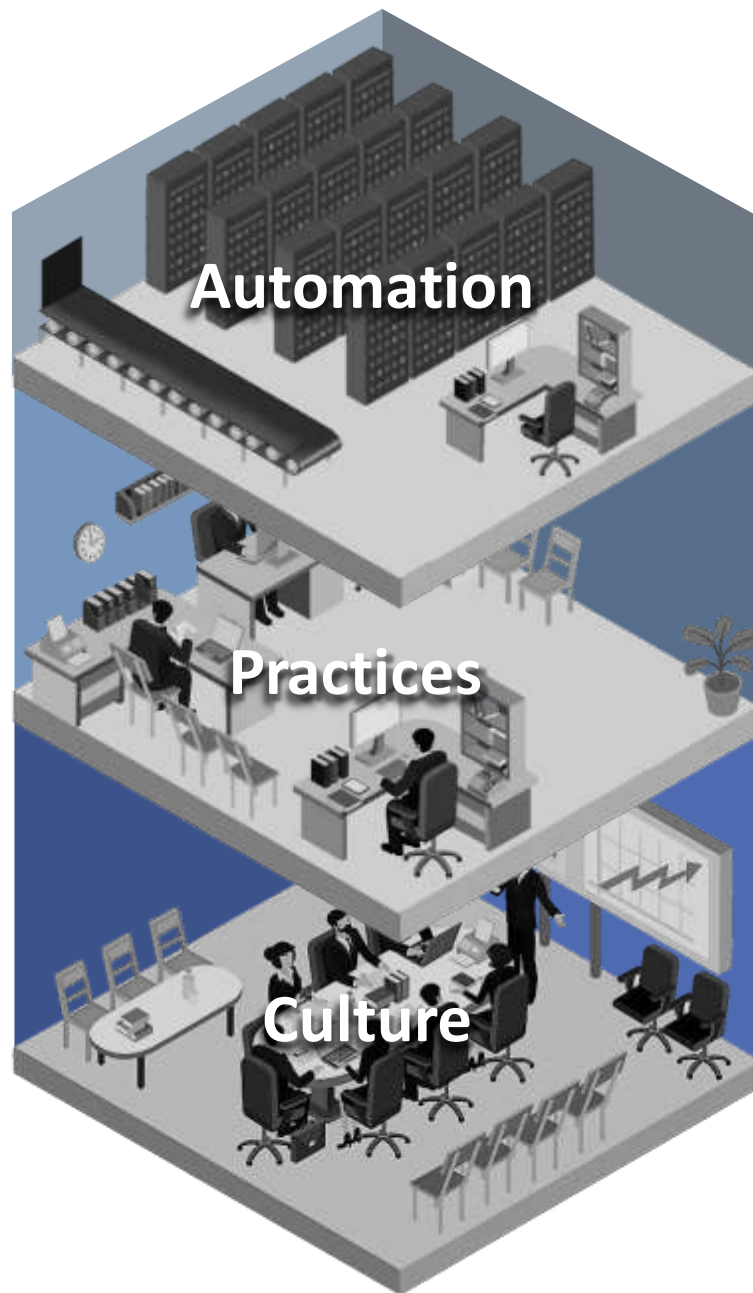


DevOps Reduces Cost





The Impact Of The Full Stack



Thank You!

For latest news and updates follow us.



[linkedin.com/company/peoplecert-group/](https://www.linkedin.com/company/peoplecert-group/)

twitter.com/peoplecert

[youtube.com/channel/UCLBidKZS9Xk08f_PM5Edtfq](https://www.youtube.com/channel/UCLBidKZS9Xk08f_PM5Edtfq)

facebook.com/peoplecert.org

