

Leading SAFe®

Thriving in the Digital Age with Business Agility

SAFe® Course - Attending this course gives students access to the SAFe® Agilist exam and related preparation materials.

5.0.1



SCALED AGILE®

© Scaled Agile, Inc.

Logistics

- ▶ Course meeting times
- ▶ Breaks
- ▶ Facilities
- ▶ Technology requirements
- ▶ Working agreements



Discussion: Introductions

Duration
5 min

- ▶ **Step 1:** Introduce yourself to someone you don't know
- ▶ **Step 2:** Share something you know about SAFe and the role of the Lean-Agile Leader



Course goals

At the end of this course you should be able to:

- ▶ Lead the transformation to Business Agility with SAFe
- ▶ Become a Lean-Agile leader
- ▶ Understand Customer needs with Design Thinking
- ▶ Enable Agile Product Delivery
- ▶ Initiate Lean Portfolio Management



Activity: Course goals overview

Duration
5 min

- ▶ **Step 1:** Introduce yourself to someone new and share three things you would like to take away from the course.
- ▶ **Step 2:** Review the outlined course goals posted on flip chart sheets around the room.
- ▶ **Step 3:** You have three votes total. Place a dot on the goals that are most relevant to you.

Course map

- ▶ Lesson 1: Thriving in the Digital Age with Business Agility
- ▶ Lesson 2: Becoming a Lean-Agile leader
- ▶ Lesson 3: Establishing Team and Technical Agility
- ▶ Lesson 4: Building Solutions with Agile Product Delivery
- ▶ Lesson 5: Exploring Lean Portfolio Management
- ▶ Lesson 6: Leading the Change
- ▶ Lesson 7: Becoming a Certified SAFe Agilist

Leading SAFe®

Lesson 1: Thriving in the Digital Age with Business Agility



SAFe® Course - Attending this course gives students access to the SAFe® Agilist exam and related preparation materials.

SCALED AGILE®

© Scaled Agile, Inc.

Learning objectives

At the end of this lesson you should be able to:

- ▶ 1.1 Thrive in the digital age
- ▶ 1.2 Describe SAFe as an operating system for Business Agility
- ▶ 1.3 Explain the Seven Core Competencies of Business Agility

1.1 Thrive in the digital age

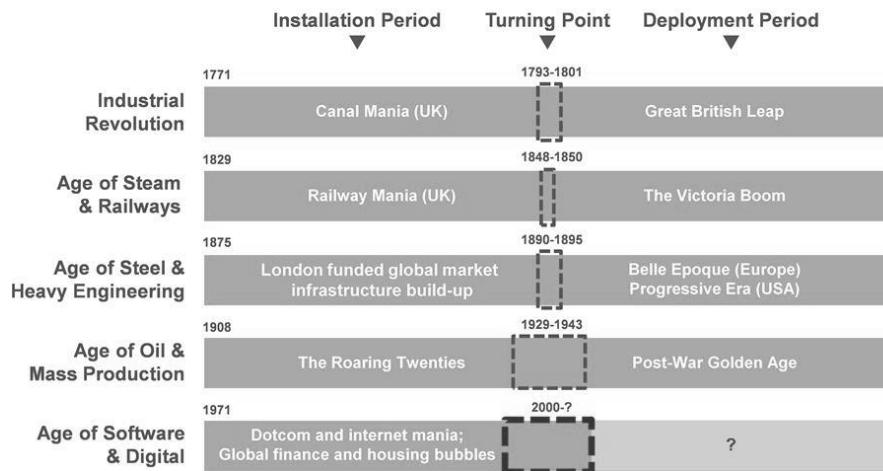
“

Those who master large-scale software delivery will define the economic landscape of the 21st century.

—Mik Kersten



Five technological revolutions



Adapted from Technological Revolutions and the Age of Software, Carlota Perez

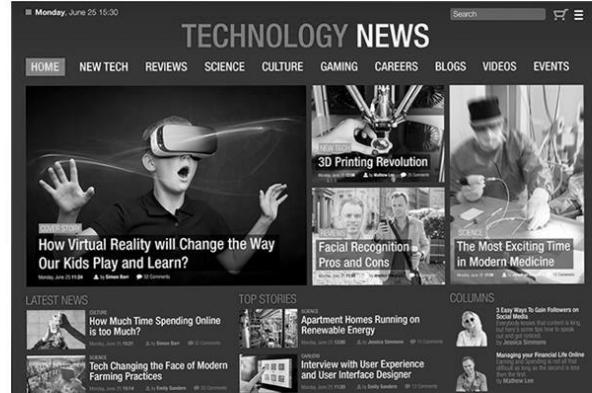
Production capital follows financial capital

- ▶ **Installation Period** – New technology and financial capital combine to create a “Cambrian explosion” of new entrants, disrupting the entire industries from the previous age
- ▶ **Turning Point** – Existing business either master the new technology or decline and become relics of the last age
- ▶ **Deployment Period** – Production capital of the new technological giants starts to take over



What stage are we in?

- ▶ "BMW Group's CEO expects that in their future more than half of its R&D staff will be software developers." (Mik Kersten, Project to Product)
- ▶ "Amazon and Whole Foods Merger to Introduce Cross-Platform Selling and Lower Prices" (Forbes, August 2017)
- ▶ The market cap of Tesla (\$43B market cap, \$21B revenue) now exceeds the market cap of Ford (\$36.2B market cap, \$160B revenue) 8:1 value ratio (September 2019)
- ▶ Apple is now the biggest watchmaker in the world (Investopedia 2019)



Competing in the Age of Software

“

The problem is not with our organizations realizing that they need to transform; the problem is that organizations are using managerial frameworks and infrastructure models from past revolutions to manage their businesses in this one.

—Mik Kersten

Rethinking the organization

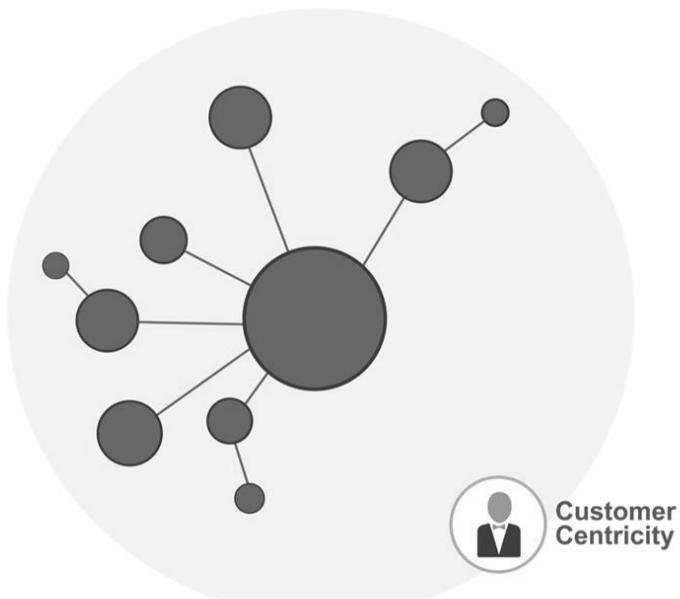
“

The world is now changing at a rate at which the basic systems, structures, and cultures built over the past century cannot keep up with the demands being placed on them.

—John P. Kotter

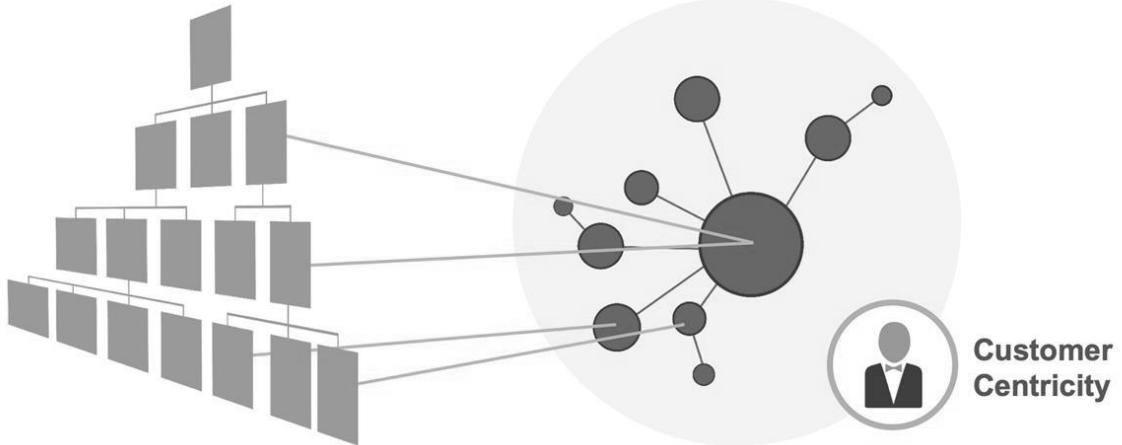


We started with a network



We add hierarchy for stability and execution

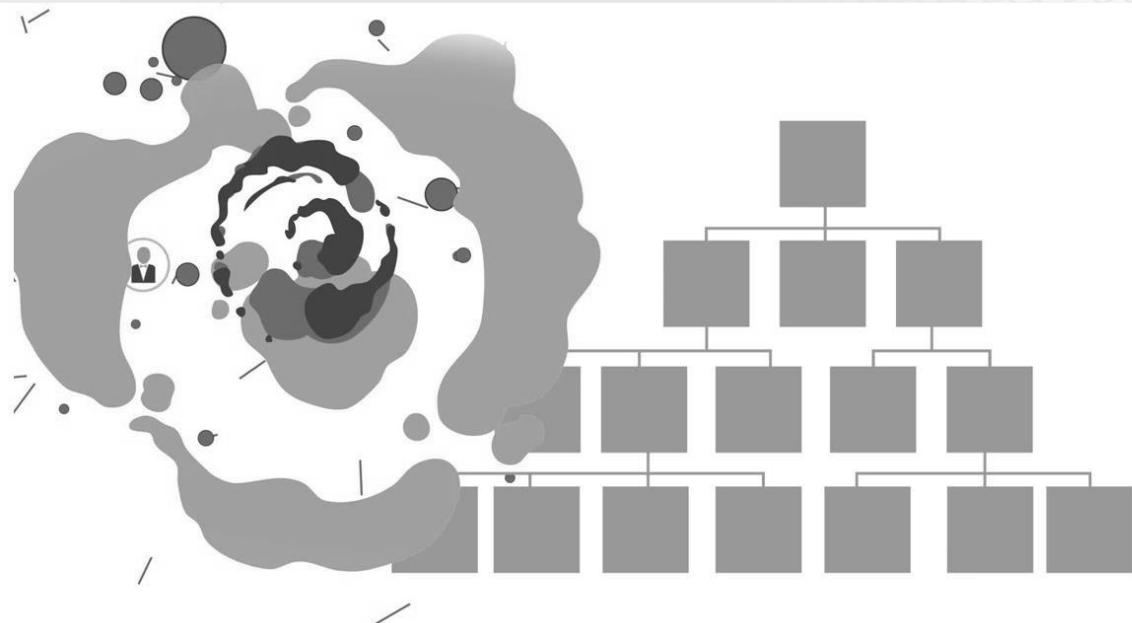
Speed of Innovation



SCALED AGILE® © Scaled Agile, Inc.

17

Guess what happens?



18

“

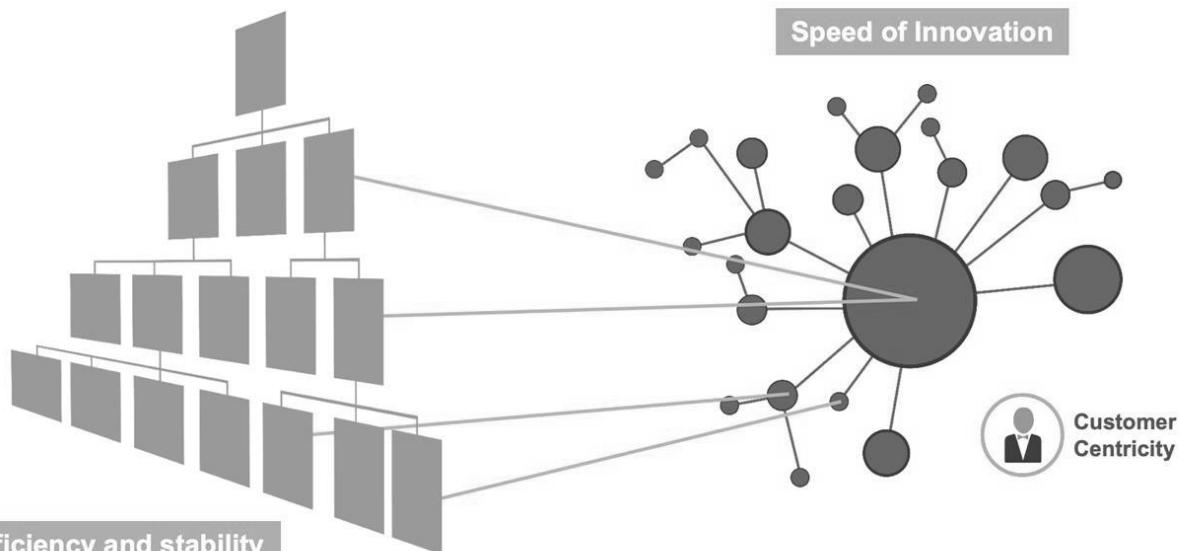
The solution is not to trash what we know and start over but instead to reintroduce a second system—one which would be familiar to most successful entrepreneurs.

You need a dual operating system.

—John P. Kotter

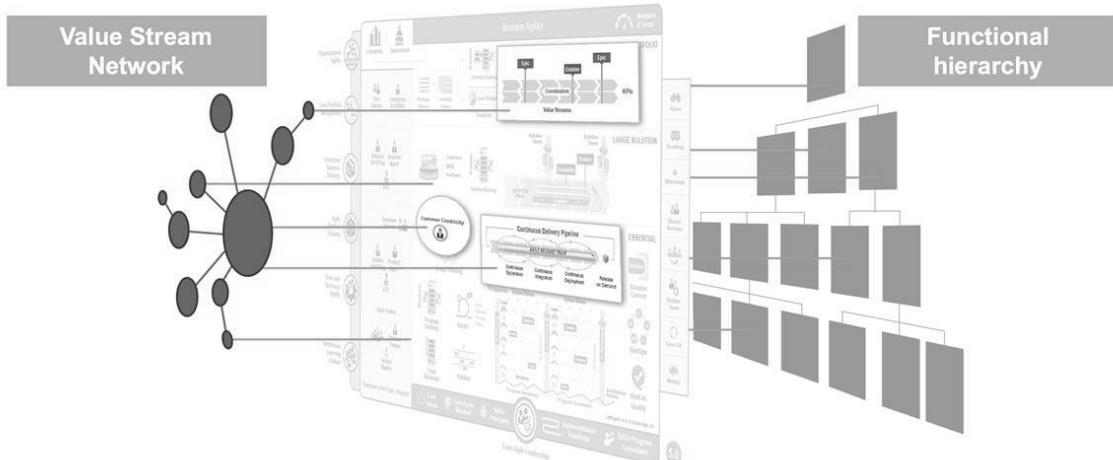
19

We need a dual operating system for Business Agility



20

And we have just such an operating system at our fingertips



1.2 Describe SAFe as an operating system for Business Agility



Every business is a software business now.

Achieving a state of **business agility** means that the entire organization—*not just development*—is engaged in continually and proactively delivering innovative business solutions faster than the competition.

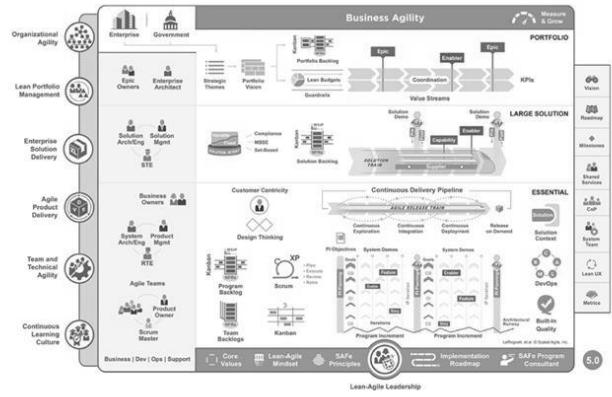
— *Dean Leffingwell*
Creator of SAFe 23

Business Agility requires technical agility and a business-level commitment to product and Value Stream thinking.

And it requires that **everyone involved in delivering business Solutions** use Lean and Agile practices.



SAFe® for Lean Enterprises is a knowledge base of proven, integrated principles, practices, and competencies for achieving Business Agility by implementing Lean, Agile, and DevOps at scale.

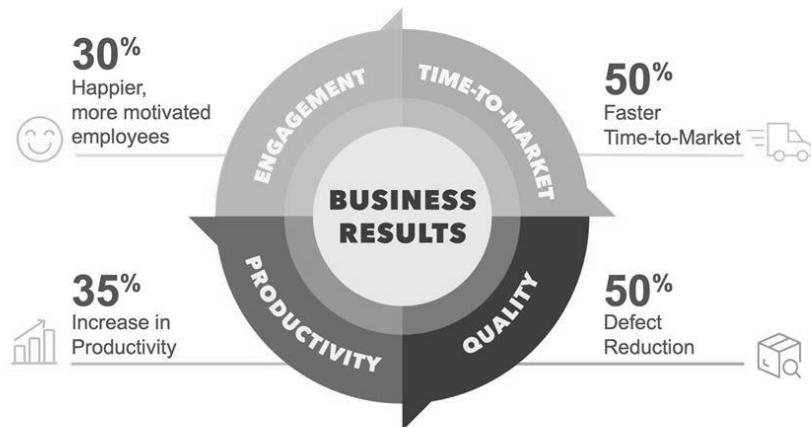


SCALED AGILE® © Scaled Agile, Inc.

25

Why SAFe?

SAFe business benefits are derived directly from case studies written by SAFe customers



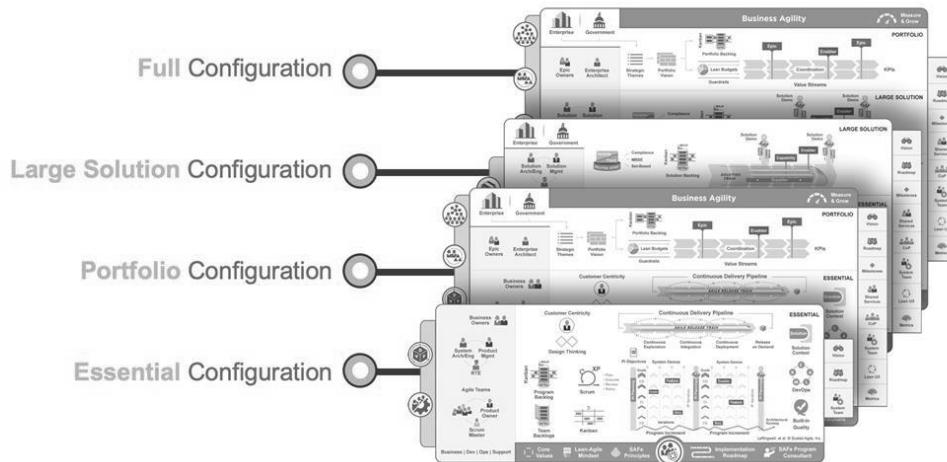
Source: Typical results from scaledagile.com/case-studies

SCALED AGILE® © Scaled Agile, Inc.

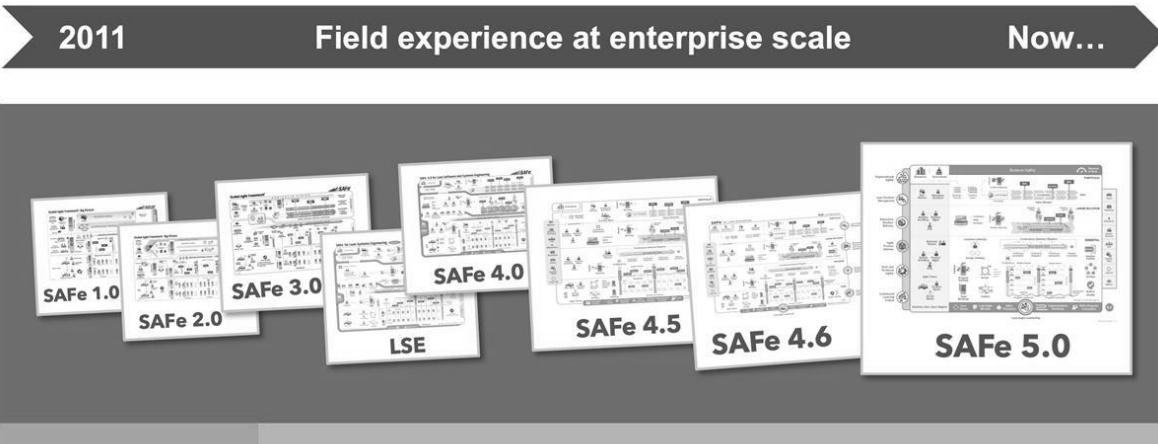
26

SAFe configurations

Four configurations provide the right solution for each Enterprise.

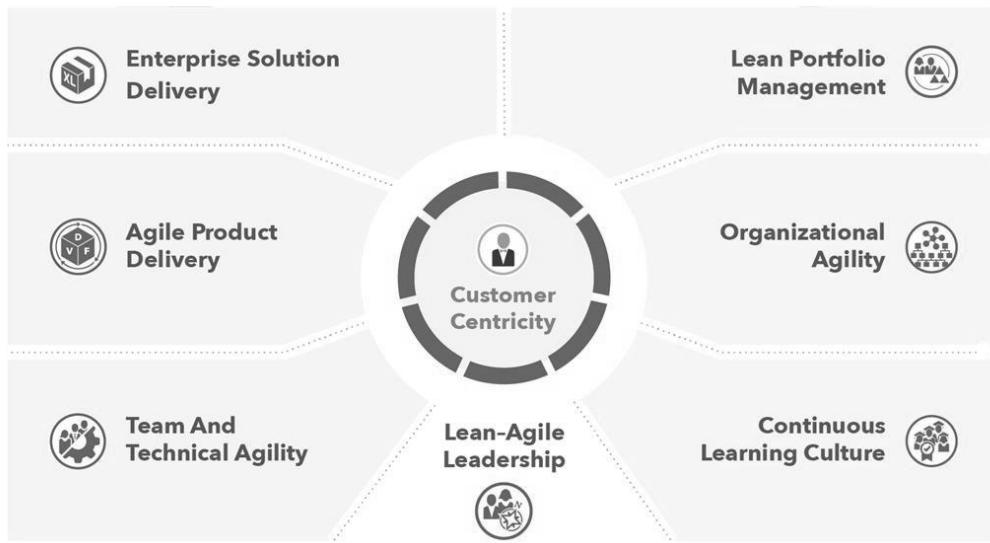


SAFe: Roots, past, present and future



Lean product development | Agile development | DevOps | Systems thinking

The Seven Core Competencies of Business Agility



1.3 Explain the Seven Core Competencies of Business Agility

Why Team and Technical Agility?

Agile Teams and teams of Agile Teams create and support the business Solutions that deliver value to the Enterprise's customers. Consequently, an organization's ability to thrive in the digital age is entirely dependent on the ability of its teams to deliver Solutions that reliably meet a customer's needs.

31

Team and Technical Agility

The Team and Technical Agility competency describes the critical skills and Lean-Agile principles and practices that high-performing Agile Teams and teams of Agile Teams use to create high-quality solutions for their customers.

Agile Teams

Scrum Master, Product Owner

Built-In Quality

- Establish Flow
- Pairing and Peer Review
- Collective Ownership and Standards
- Automation
- Definition of Done

Teams of Agile Teams

- Business
- Product Mgmt.
- Hardware
- Software
- Quality
- Testing
- Compliance
- Operations
- Security

AGILE RELEASE TRAIN

32

Why Agile Product Delivery?

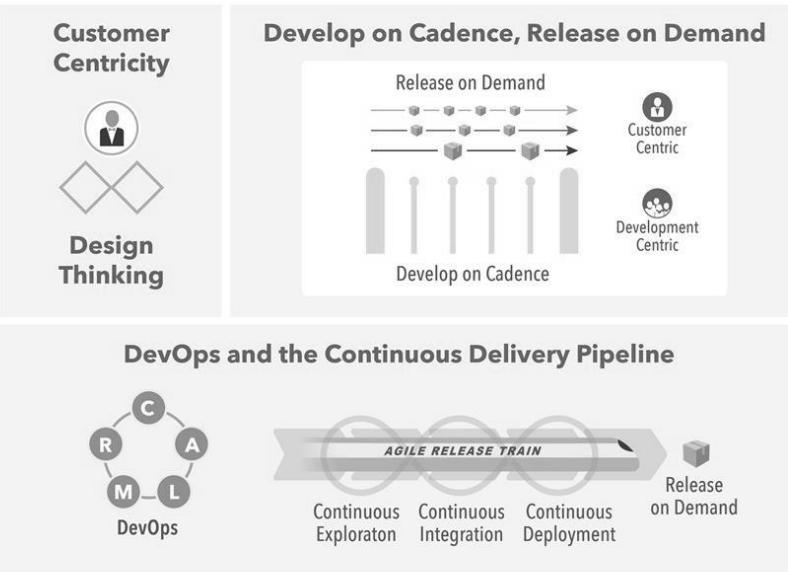
In order to achieve Business Agility, Enterprises must rapidly increase their ability to deliver innovative products and services. To be sure that the Enterprise is creating the right Solutions for the right customers at the right time, they must balance their execution focus with a customer focus.



33

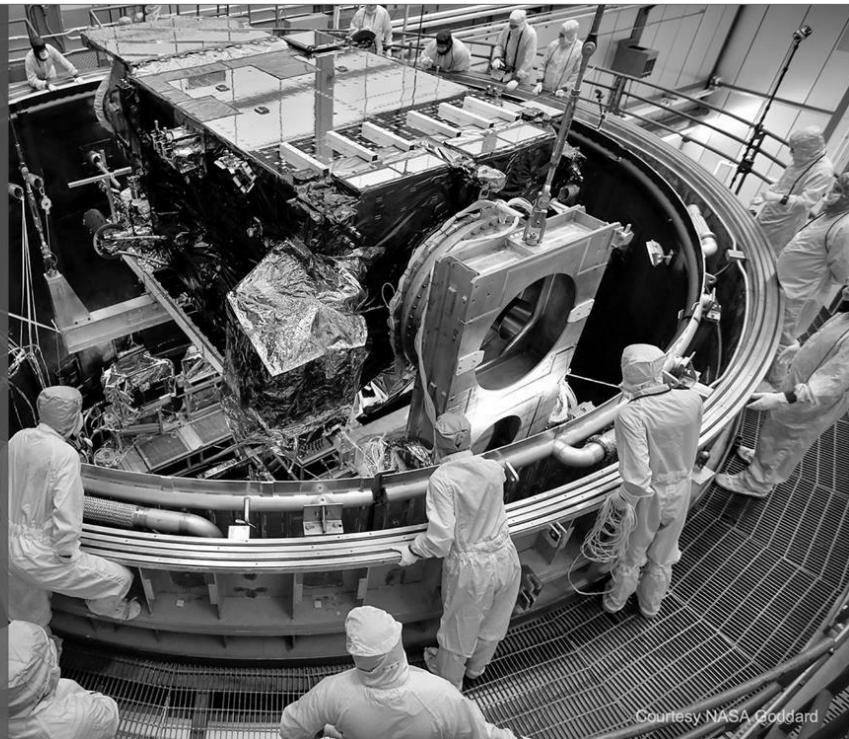
Agile Product Delivery

Agile Product Delivery is a customer-centric approach to defining, building, and releasing a continuous flow of valuable products and services to customers and users.



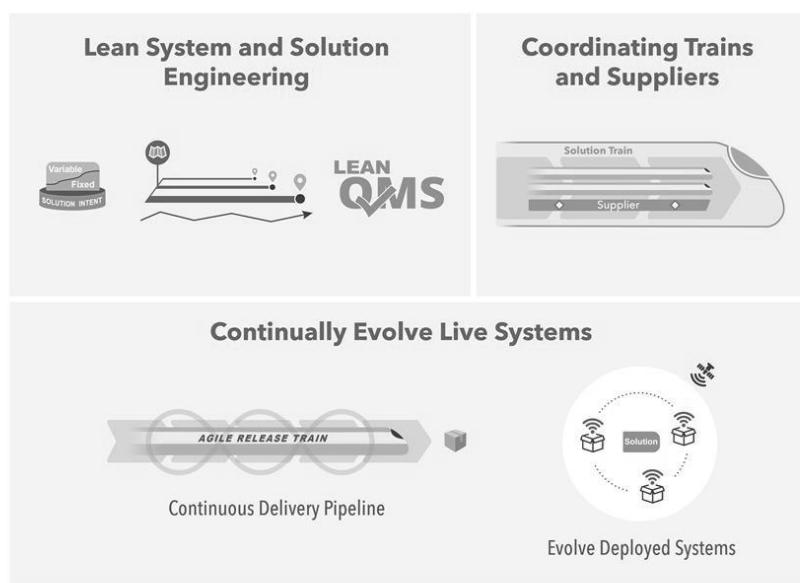
Why Enterprise Solution Delivery?

Humanity has always dreamed big; and scientists, engineers, and software developers then turn those big dreams into reality. That requires innovation, experimentation, and knowledge from diverse disciplines. Engineers and developers bring these innovations to life by defining and coordinating all the activities to successfully specify, design, test, deploy, operate, evolve, and decommission large, complex solutions.



Enterprise Solution Delivery

The Enterprise Solution Delivery competency describes how to apply Lean-Agile principles and practices to specification, development, deployment, operation, and evolution of the world's largest and most sophisticated software applications, networks, and cyber-physical systems.



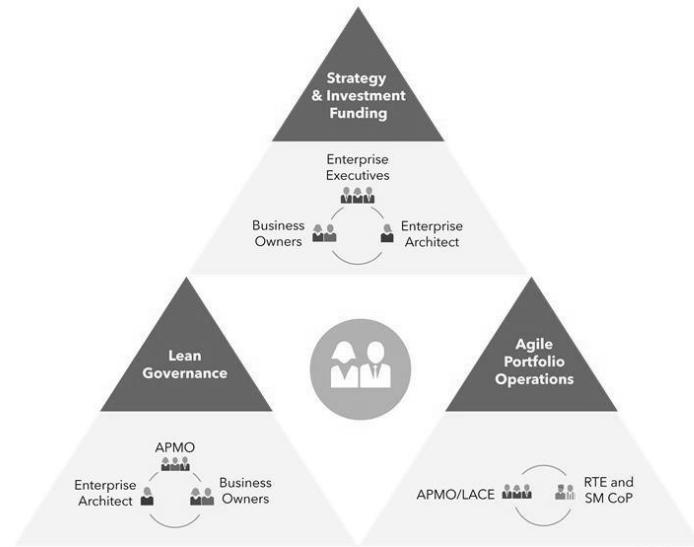
Why Lean Portfolio Management?

Traditional approaches to portfolio management were not designed for a global economy or the impact of digital disruption. These factors put pressure on enterprises to work under a higher degree of uncertainty, and yet deliver innovative solutions much faster.



Lean Portfolio Management

The Lean Portfolio Management competency aligns strategy and execution by applying Lean- and systems-thinking approaches to strategy and investment funding, Agile portfolio operations, and governance.



Why Organizational Agility?

Without Organizational Agility, Enterprises simply cannot respond sufficiently to the challenges and opportunities that today's rapidly changing markets present. Without it, employees and the Enterprises associate an individual's value with their functional skills, rather than business outcomes.



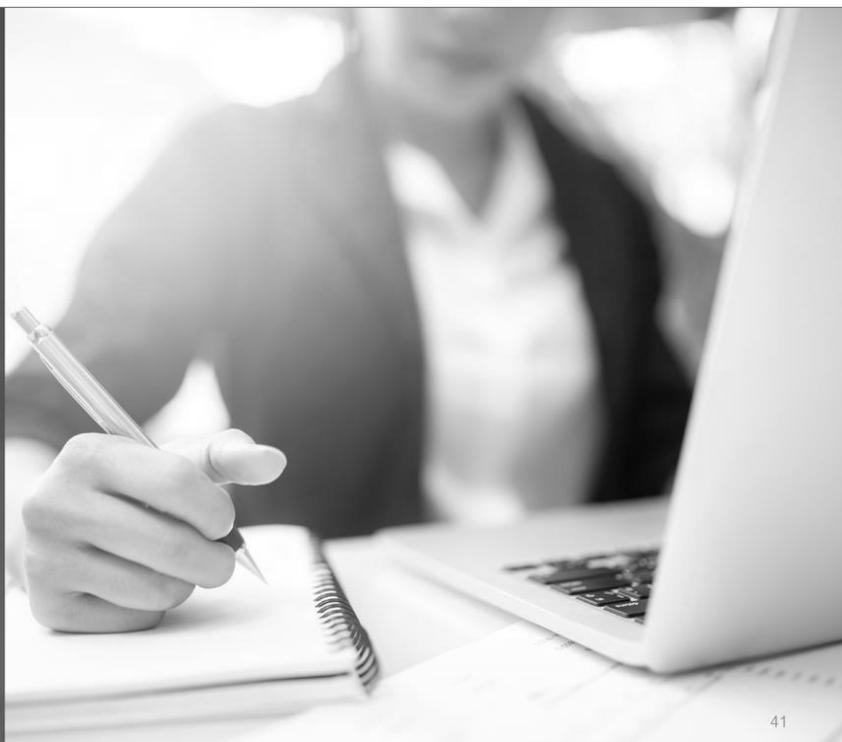
Organizational Agility

The Organizational Agility competency describes how Lean-thinking people and Agile Teams optimize their business process, evolve strategy with clear and decisive new commitments, and quickly adapt the organization as needed to capitalize on new opportunities.



Why Continuous Learning Culture?

In order to thrive in the current climate, organizations must evolve into adaptive engines of change, powered by a culture of fast and effective learning at all levels. Learning organizations leverage the collective knowledge, experience, and creativity of their workforce, customers, supply chain, and the broader ecosystem.



41

Continuous Learning Culture

The Continuous Learning Culture competency describes a set of values and practices that encourage individuals, and the Enterprise as a whole, to continually increase knowledge, competence, performance, and innovation



Why Lean-Agile Leadership?

An organization's managers, executives, and other leaders are responsible for the adoption, success, and ongoing improvement of Lean-Agile development and the competencies that lead to Business Agility. Only they have the authority to change and continuously improve the systems that govern how work is performed.



Lean-Agile Leadership

The Lean-Agile Leadership competency describes how Lean-Agile Leaders drive and sustain organizational change and operational excellence by empowering individuals and teams to reach their highest potential.



The management challenge



It is not enough that management commit themselves to quality and productivity, they must know what it is they must do.

Such a responsibility cannot be delegated.

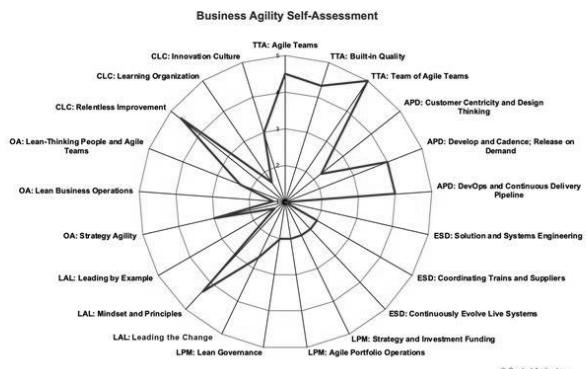
—W. Edwards Deming

...and if you can't come, send no one”

—Vignette from “Out of the Crisis,” W. Edwards Deming

Measure and Grow

Measure and Grow is the way portfolios evaluate their progress in Business Agility and determine their next improvement steps.



Lesson review

In this lesson you:

- ▶ Discussed what is necessary to be able to thrive in the digital age
- ▶ Described SAFe as your operating system for Business Agility
- ▶ Explained the Seven Core Competencies of Business Agility

Leading SAFe®

Lesson 2: Becoming a Lean-Agile Leader



SAFe® Course - Attending this course gives students access to the SAFe® Agilist exam and related preparation materials.

SCALED AGILE®

© Scaled Agile, Inc.

Learning objectives

At the end of this lesson you should be able to:

- ▶ 2.1 Embrace the Lean Agile Mindset
- ▶ 2.2 Apply Lean and Agile at scale with the SAFe Principles

2.1 Embrace the Lean-Agile Mindset

Exemplifying SAFe core values

Alignment

- ▶ Provide the relevant briefings and participate in Program Increment (PI) Planning
- ▶ Help with backlog visibility, review, and preparation
- ▶ Help with Value Stream organization and coordination
- ▶ Constantly check for understanding
- ▶ Communicate the mission, visions and strategy at every opportunity

Built-in Quality

- ▶ Demonstrate quality by refusing to accept or ship low-quality work
- ▶ Support investments in capacity planning for maintenance and reduction of technical debt
- ▶ Ensure UX, architecture, operations, security, compliance, and others, are part of the flow of work

Transparency

- ▶ Visualize all relevant work
- ▶ Take ownership and responsibility for errors and mistakes
- ▶ Admit your own mistakes
- ▶ Support others who acknowledge and learn from their mistakes—never punish the messenger

Program Execution

- ▶ Participate as an active business owner in PI execution
- ▶ Celebrate high quality and predictably delivered Program Increments
- ▶ Aggressively remove impediments and demotivators



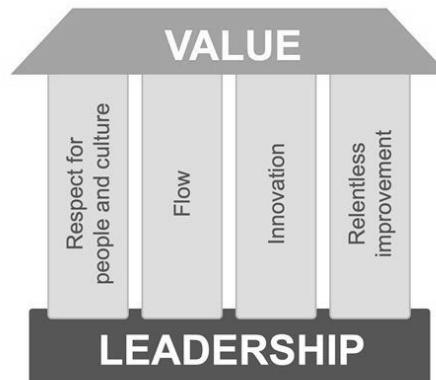
Taking Action: Exemplifying SAFe's core values

Duration
5 min

- ▶ **Step 1:** Individually choose one of SAFe's four core values: Alignment, Transparency, Built-In Quality, and Program Execution
- ▶ **Step 2:** In your group, discuss how can you exemplify that core value in your organization
- ▶ **Step 3:** Write down one example in the Action Plan in your workbook



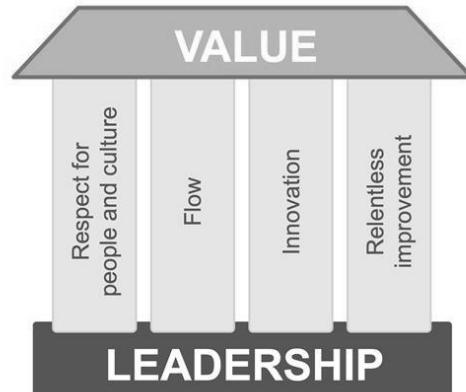
SAFe House of Lean



Value

Achieve the shortest sustainable lead time with:

- ▶ The best quality and value to people and society
- ▶ High morale, safety, and Customer delight

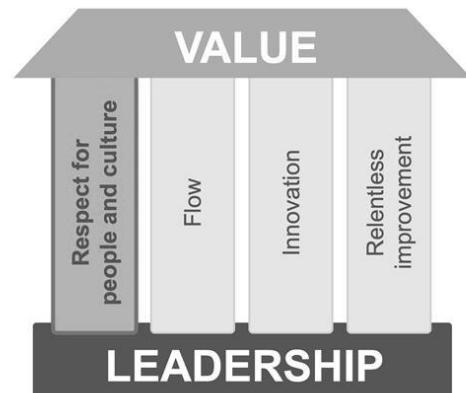


*There is only one boss. The customer.
And he can fire everybody in the company*

—Sam Walton

Respect for people and culture

- ▶ Generative culture
- ▶ People do all the work
- ▶ Your Customer is whoever consumes your work
- ▶ Build long-term partnerships based on trust
- ▶ To change the culture, you have to change the organization

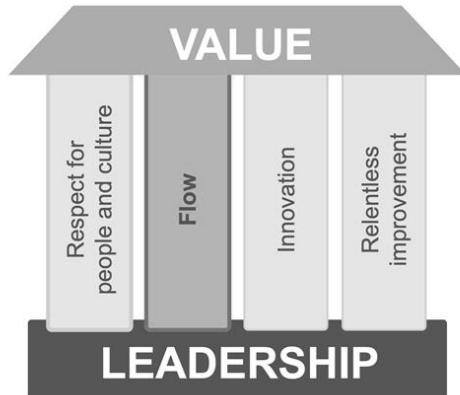


Culture eats strategy for breakfast.

—Peter Drucker

Flow

- ▶ Optimize sustainable value delivery
- ▶ Build-in quality
- ▶ Understand, exploit, and manage variability
- ▶ Move from projects to products

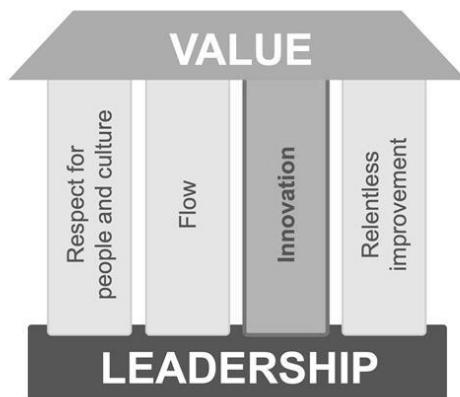


Operating a product development process near full utilization is an economic disaster.

—Don Reinertsen

Innovation

- ▶ Innovative people
- ▶ Provide time and space for innovation
- ▶ Go see
- ▶ Experimentation and feedback
- ▶ Innovation riptides
- ▶ Pivot without mercy or guilt

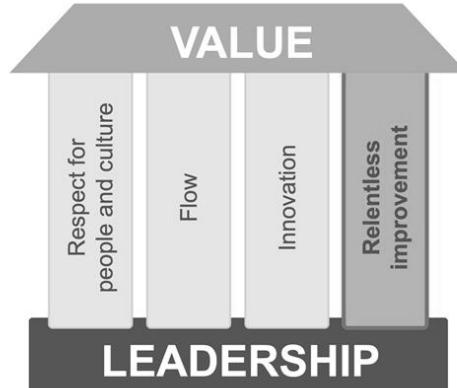


Innovation comes from the producer.

—W. Edwards Deming

Relentless Improvement

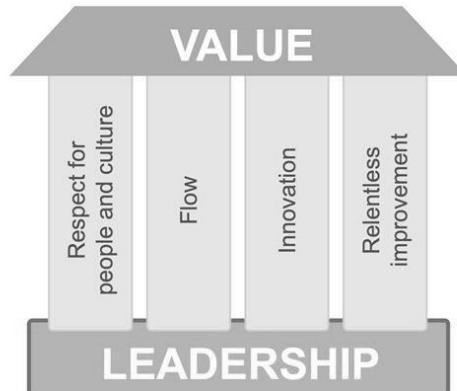
- ▶ A constant sense of danger
- ▶ Optimize the whole
- ▶ Problem-solving culture
- ▶ Base improvements on facts
- ▶ Reflect at key Milestones



Those who adapt the fastest win.

Leadership

- ▶ Lead by example
- ▶ Adopt a growth mindset
- ▶ Exemplify the values and principles of Lean-Agile and SAFe
- ▶ Develop people
- ▶ Lead the change
- ▶ Foster psychological safety



*People are already doing their best;
the problems are with the system. Only
management can change the system*

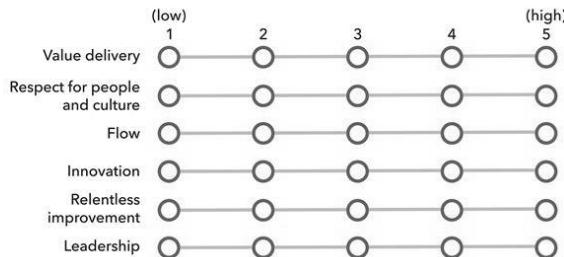
—W. Edwards Deming



Activity: Assessing a Lean mindset

Duration
5 min

- ▶ **Step 1:** Assess where your team stands in embracing a Lean mindset.
- ▶ **Step 2:** Discuss the results of the self-assessment. Do you have similar low or high scores?



Taking Action: Improving the Lean-Agile mindset

Prepare 3 min Share 2 min

- ▶ **Step 1:** Select one of the lowest scores in the assessment.
- ▶ **Step 2:** Brainstorm one to three actions you could take to improve this area.
- ▶ **Step 3:** Share your ideas with your group. Give and receive constructive suggestions on how the ideas offered can improve the mindset scores.
- ▶ **Step 4:** Write down one idea in your Action Plan and be prepared to share.



The Agile Manifesto

We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan

That is, while there is value in the items on the right, we value the items on the left more.

 agilemanifesto.org

The Agile Manifesto Principles

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

The Agile Manifesto Principles

7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity—the art of maximizing the amount of work not done—is *essential*.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.



Activity: Agile principles at scale



- ▶ **Step 1:** Review the principles behind the Agile Manifesto
- ▶ **Step 2:** Select one principle per group
- ▶ **Step 3:** Categorize as:
 - Works as is
 - Not applicable
 - Requires rethinking for scale
- ▶ **Step 4:** Share your findings with the class

2.2 Apply Lean and Agile at scale with the SAFe Principles

SAFe Lean-Agile Principles

#1 Take an economic view

#2 Apply systems thinking

#3 Assume variability; preserve options

#4 Build incrementally with fast, integrated learning cycles

#5 Base milestones on objective evaluation of working systems

#6 Visualize and limit WIP, reduce batch sizes, and manage queue lengths

#7 Apply cadence, synchronize with cross-domain planning

#8 Unlock the intrinsic motivation of knowledge workers

#9 Decentralize decision-making

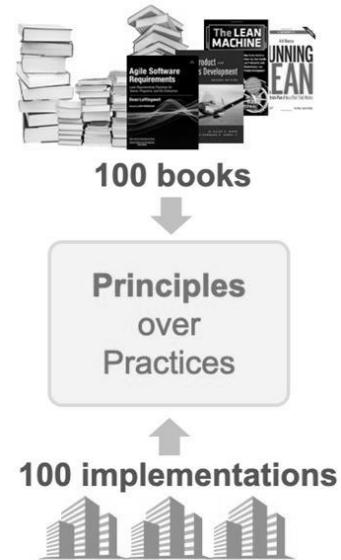
#10 Organize around value

Why focus on the principles?

A common disease that afflicts management the world over is the impression that "Our problems are different." They are different to be sure, but the principles that will help to improve the quality of products and services are universal in nature.

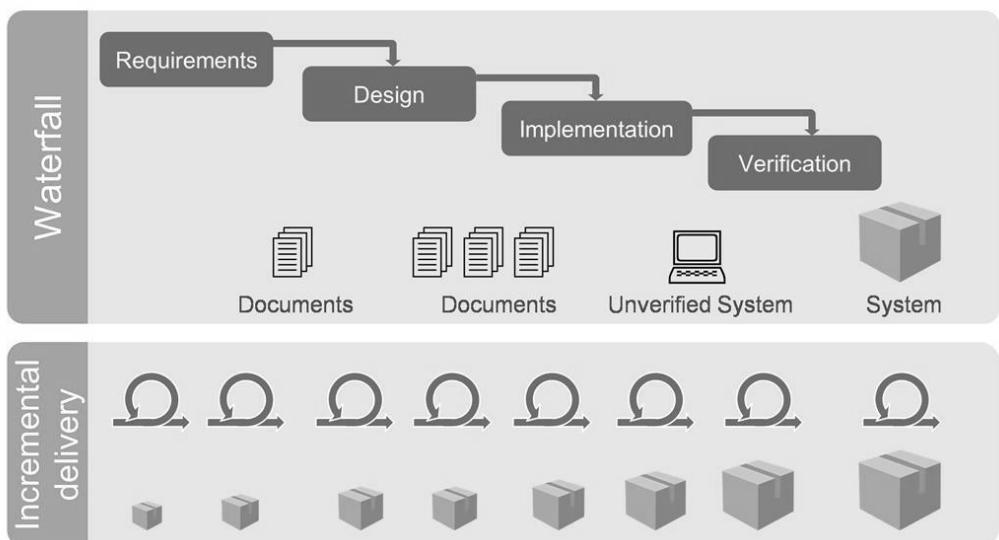
—W. Edwards Deming

- ▶ A Lean-Agile transformation will deliver substantial benefits
- ▶ However, it is a significant change, and every implementation is different
- ▶ Leaders should understand why the practices work; it's part of 'knowing what it is they must do'
- ▶ If a practice needs to change, understanding the principles will assure the change moves the Enterprise in the right direction



#1 Take an economic view

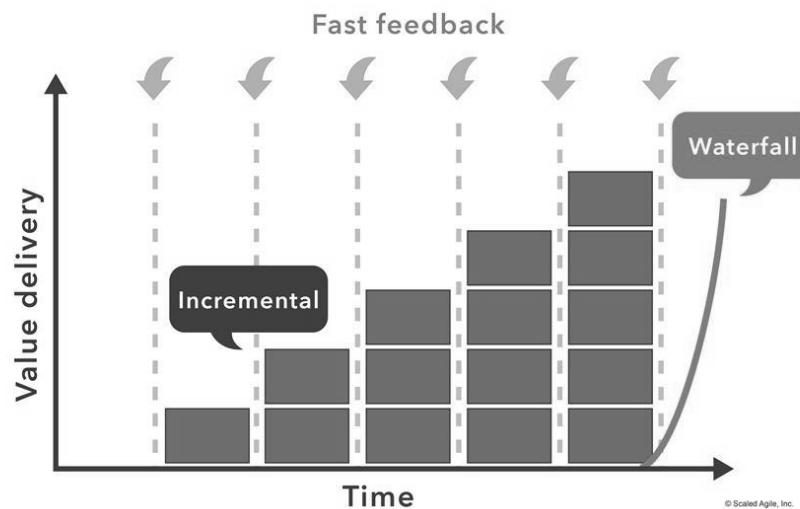
Agile economics: Deliver early and often



SCALED AGILE® © Scaled Agile, Inc.

70

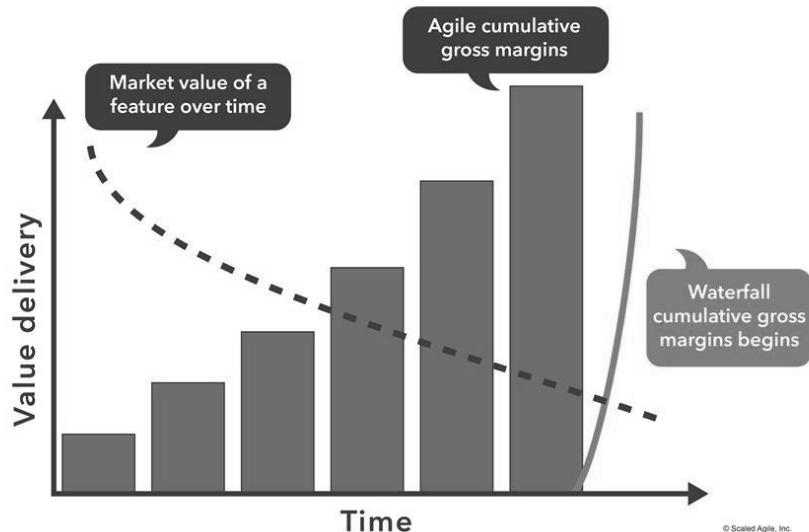
Deliver value incrementally



SCALED AGILE® © Scaled Agile, Inc.

71

Early delivery has higher value

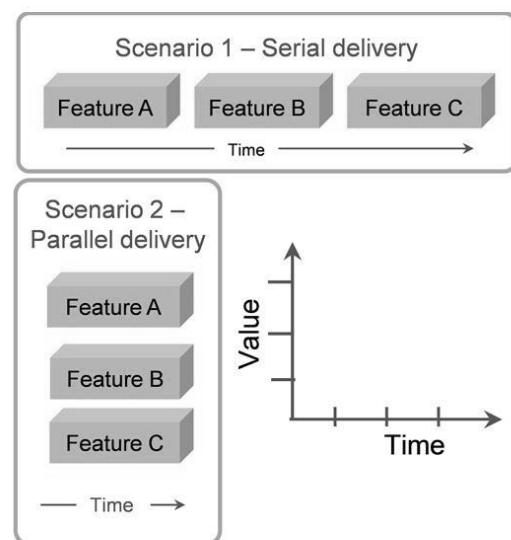


Activity: Accelerating value delivery

Prepare
5 min

Share
2 min

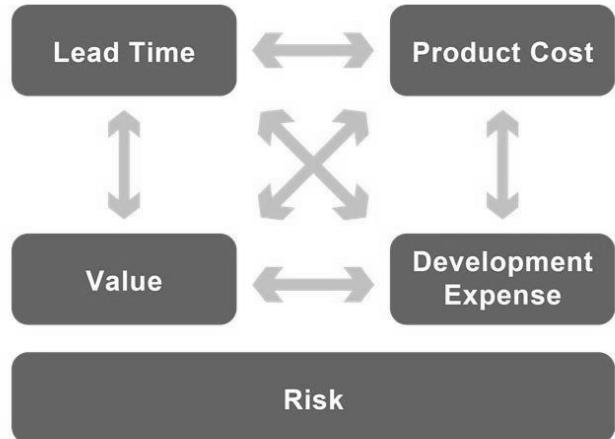
- ▶ **Step 1:** Consider that your backlog has three Features. Each will take the entire team one month and delivers one unit of value.
- ▶ **Step 2:** Plot the value delivery of serial and simultaneous/parallel implementation scenarios for delivering the Features.
 - **NOTE:** Assume 20% task switching overhead for each team member in Scenario 2
 - **HINT:** Plot the serial case first



Solution economic trade-offs

Understanding tradeoff parameters:

- ▶ Sequence jobs for maximum benefit
- ▶ Do not consider money already spent
- ▶ Make economic choices continuously
- ▶ Empower local decision making
- ▶ If you only quantify one thing, quantify the cost of delay



© Scaled Agile, Inc.

#2 Apply systems thinking

Systems thinking



A system must be managed. It will not manage itself.

Left to themselves, components become selfish, independent profit centers and thus destroy the system...

The secret is cooperation between components toward the aim of the organization.

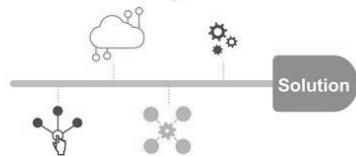
—W. Edwards Deming

Attributes of systems thinking

The Solution and the Enterprise are both affected by the following:

- ▶ Optimizing a component does not optimize the system
- ▶ For the system to behave well as a system, a higher-level understanding of behavior and architecture is required
- ▶ The value of a system passes through its interconnections
- ▶ A system can evolve no faster than its slowest integration point

The Solution itself is a system



The Enterprise building the system is a system too

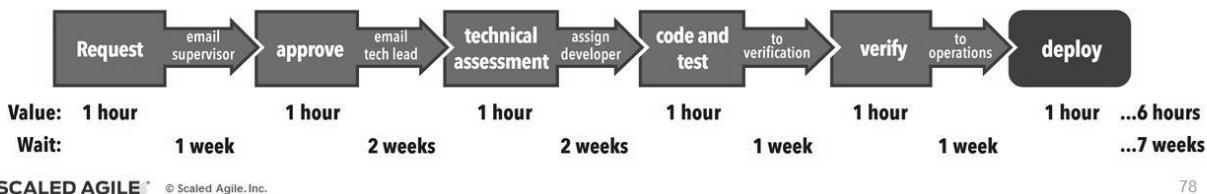


Optimize the full Value Stream

All we are doing is looking at the timeline, from when the customer gives us an order to when we collect the cash. And we are reducing the timeline by reducing the non-value added wastes. —Taiichi Ohno

- ▶ Most problems with your process will surface as *delays*
- ▶ Most of the time spent getting to market is a result of these delays
- ▶ Reducing delays is the fastest way to reduce time-to-market

Focus on the delays!



SCALED AGILE® © Scaled Agile, Inc.

78



Discussion: Identifying delays



- ▶ **Step 1:** Identify three delays from your context and write them down.
- ▶ **Step 2:** Write down what you think might be some potential causes for the delays.
- ▶ **Step 3:** Consider how systems thinking relates to finding possible solutions for the delays. Who is ultimately responsible for the optimization of the full Value Stream?
- ▶ **Step 4:** Share your insights with the class.

SCALED AGILE® © Scaled Agile, Inc.

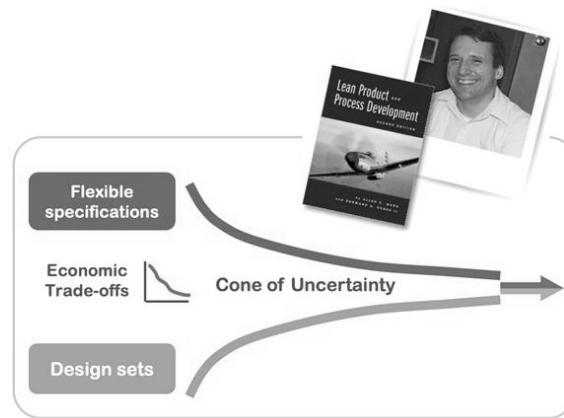
79

#3 Assume variability; preserve options

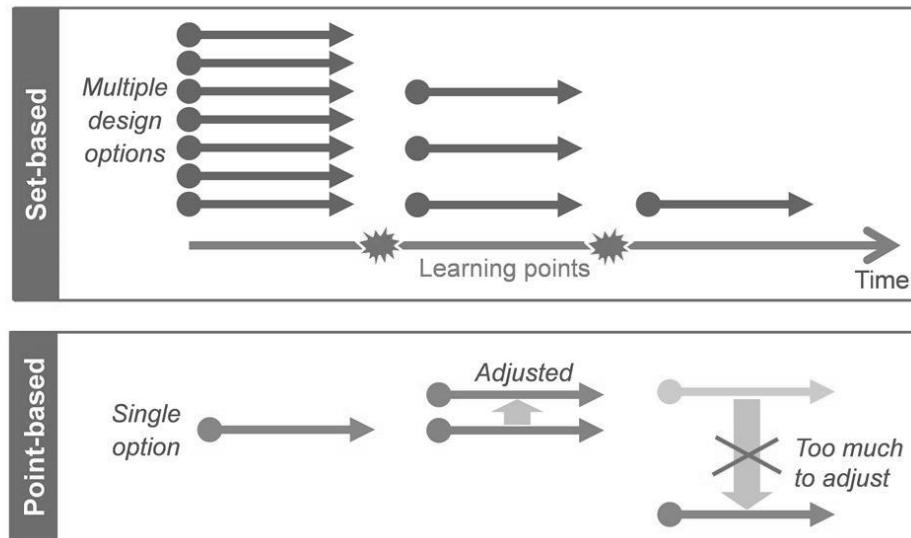
Development occurs in an uncertain world

Aggressively evaluate alternatives. Converge specifications and solution set. —Allen Ward

- ▶ You cannot possibly know everything at the start
- ▶ Requirements must be flexible to make economic design choices
- ▶ Designs must be flexible to support changing requirements
- ▶ Preservation of options improves economic results



Apply a set-based approach



#4 Build incrementally with fast, integrated learning cycles

Apply fast learning cycles

- ▶ Improves learning efficiency by decreasing the time between action and effect
- ▶ Reduces the cost of risk-taking by truncating unsuccessful paths quickly
- ▶ Is facilitated by small batch sizes
- ▶ Requires increased investment in development environment

The shorter the cycles, the faster the learning

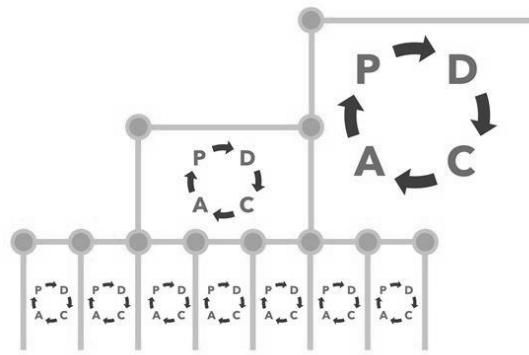
The iterative learning cycle



©Scaled Agile, Inc.

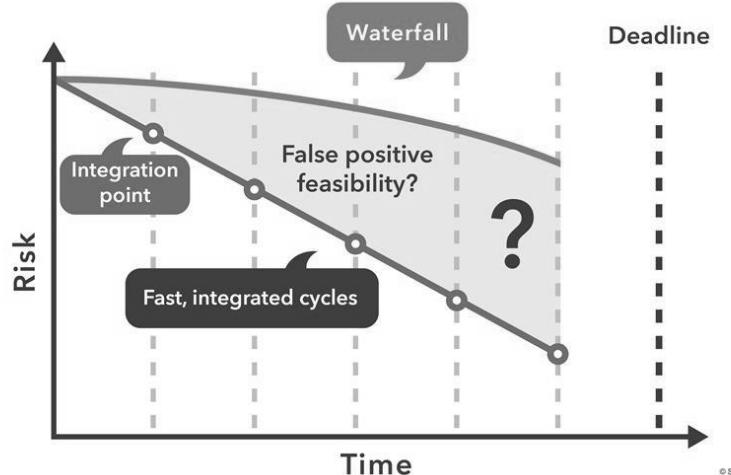
Integration points control product development

- ▶ Integration points accelerate learning
- ▶ Development can proceed no faster than the slowest learning loop
- ▶ Improvement comes through synchronization of design loops and faster learning cycles



©Scaled Agile, Inc.

Integration points reduce risk



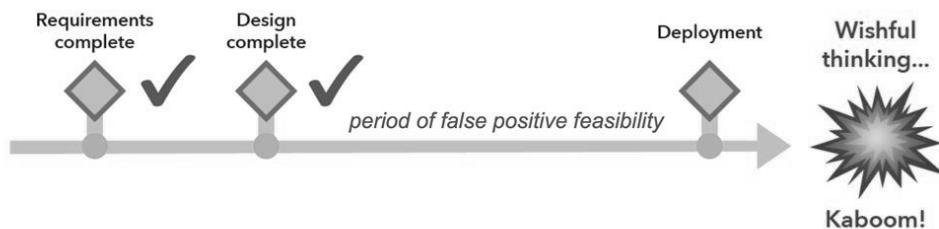
© Scaled Agile, Inc.

#5 Base milestones on objective evaluation of working systems

The problem of phase-gate milestones

There was in fact no correlation between exiting phase gates on time and project success... the data suggested the inverse might be true. —Lean Machine

- ▶ They force design decisions too early; this encourages false-positive feasibility.
- ▶ They assume a ‘point’ Solution exists and can be built correctly the first time.
- ▶ They create huge batches and long queues, and they centralize requirements and design in program management.

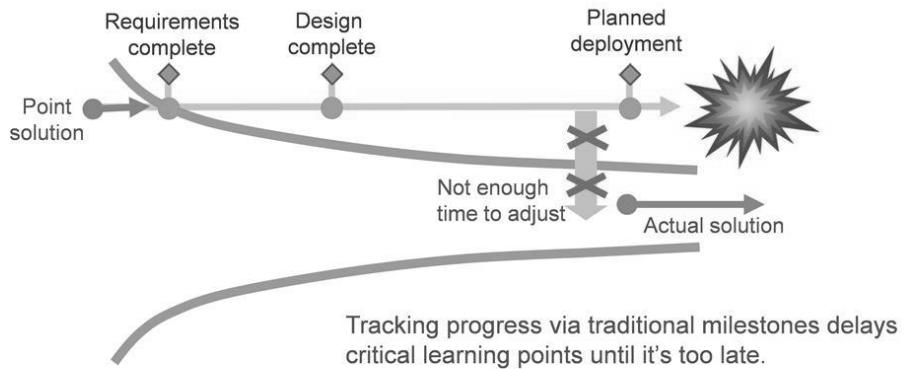


SCALED AGILE® © Scaled Agile, Inc.

88

The problem of phase-gate milestones

Phase gates fix requirements and designs too early, making adjustments too late and costly as new facts emerge.

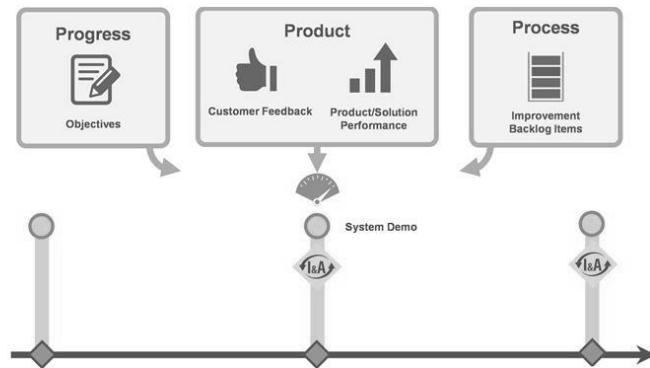


SCALED AGILE® © Scaled Agile, Inc.

89

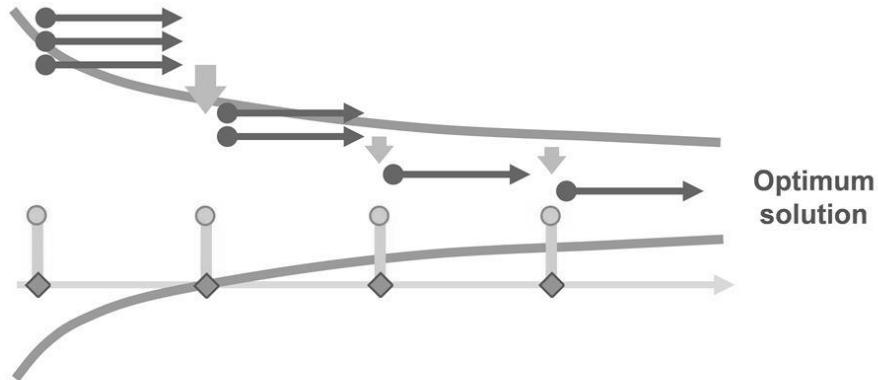
Apply objective Milestones

Program Increment (PI) System Demos are orchestrated to deliver objective progress, product, and process Metrics.



Iterate to the optimum solution

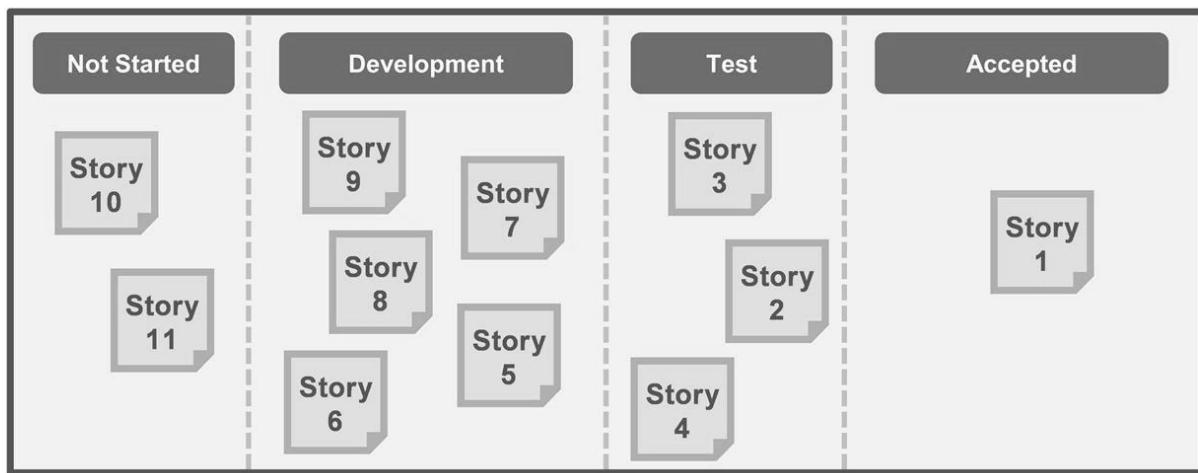
Objective Milestones facilitate learning and allow for continuous, cost-effective adjustments towards an optimum Solution.



#6 Visualize and limit WIP, reduce batch sizes, and manage queue lengths

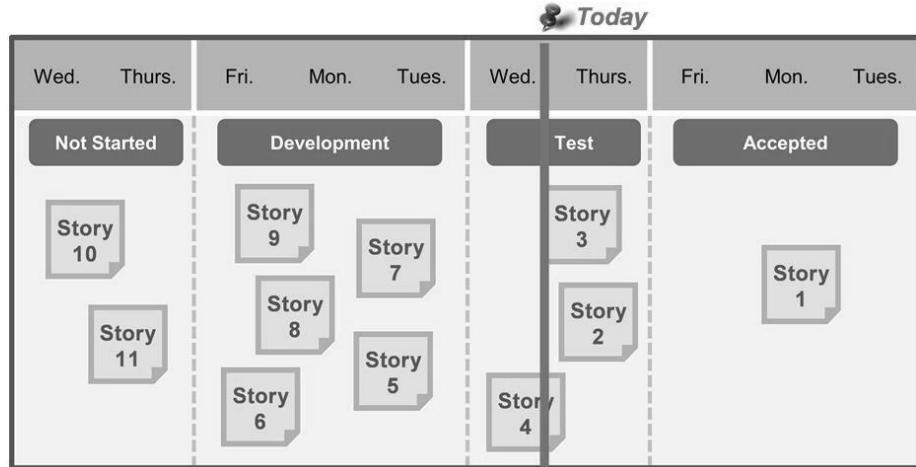
An example from the field

How is this team doing? How do you know that?



Visualize to increase understanding

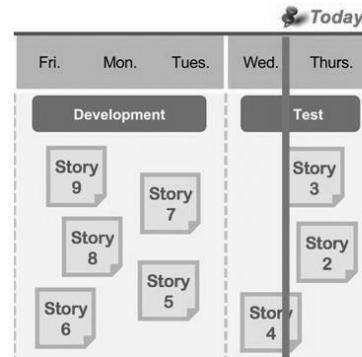
Now how do you think they are doing?



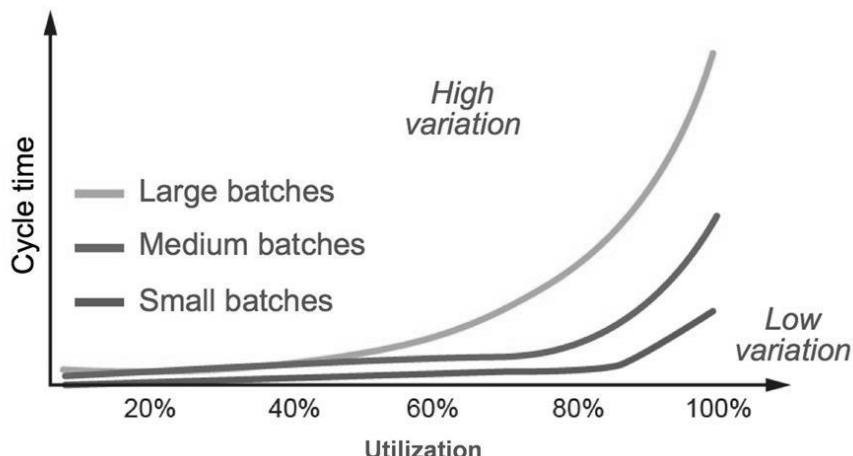
Activity: WIP improvement opportunities



- ▶ **Step 1:** Look at the BVIR graphic together on the following page in your workbook.
- ▶ **Step 2:** As a class, discuss what the effect would be of a three-story WIP constraint on Development and Test.
- ▶ **Step 3:** Consider this scenario: You're a developer. You just finished story 6. What would you do if:
 - There is no WIP constraint
 - The three-story WIP constraint is in place
- ▶ **Step 4:** Which scenario has the highest throughput?



Reduce batch size for higher predictability



Source: *Implementing Lean Software Development*, Poppendieck, Mary.



Activity: Experience a large batch size



- ▶ **Step 1:** Create groups of five people with 10 coins per group. Designate one person as the timekeeper. The remaining four people will be processing the coins.
- ▶ **Step 2:** Person by person, flip each coin one at a time, recording your own results (heads or tails).
- ▶ **Step 3:** Pass all coins at the same time to the next person, who repeats step two, until all four people are done
- ▶ **Step 4:** The timekeeper stops the timer and records the total time





Activity: Experience a small batch size

Duration
5 min

- ▶ **Step 1:** Ensure that the timekeeper is ready to start the timer
- ▶ **Step 2:** This time, each person flips one coin at a time, records the result (heads or tails), and immediately passes each coin to the next person
- ▶ **Step 3:** The timekeeper will stop the timer when the last person flips the last coin and records the result



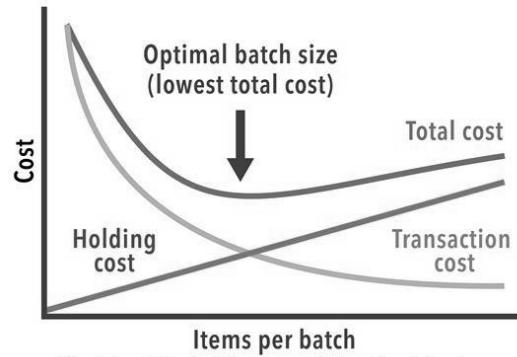
The importance of small batches

- ▶ Large batch sizes increase variability
- ▶ High utilization increases variability
- ▶ Severe project slippage is the most likely result
- ▶ Small batches go through the system faster with lower variability
- ▶ The most important batch is the handoff batch

Finding optimal batch size

Optimal batch size is an example of a U-curve optimization.

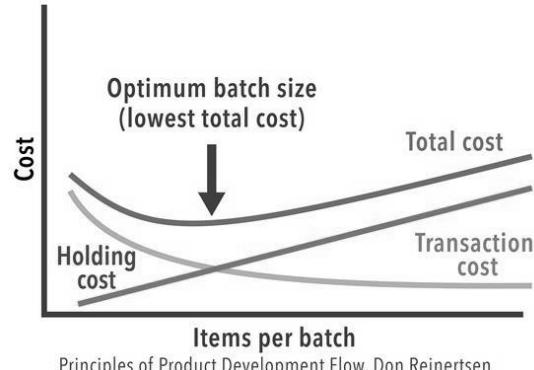
- ▶ Total costs are the sum of holding costs and transaction costs
- ▶ Higher transaction costs make optimal batch size bigger
- ▶ Higher holding costs make batch size smaller



Reducing optimal batch size

Reducing transaction costs reduces total costs and lowers optimum batch size.

- ▶ Reducing batch size:
 - Increases predictability
 - Accelerates feedback
 - Reduces rework
 - Lowers cost
- ▶ Batch size reduction probably saves **twice** what you think





Video: Formula 1 Pit Stops: 1950 and Today

Duration
2 min



Manage queue lengths

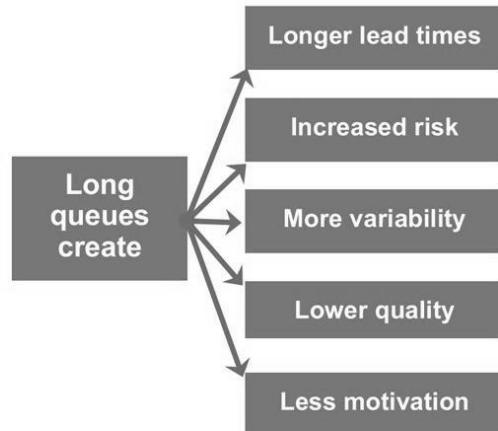
Email from a client service organization:

Thank you for contacting us. 

We are experiencing increased volumes and apologize in advance for the delay.

Our goal is to contact you within . . .

Long queues: All bad



Principles of Product Development Flow, Don Reinertsen

Reduce queue lengths

► Understand Little's Law

- Faster processing time decreases wait
- Shorter queue lengths decrease wait

► Control wait times by controlling queue lengths:

- WIP limits, small batches, defer commitments

$$W_q = \frac{L_q}{\lambda}$$

Average wait time = average queue length
divided by average processing rate

Example - Given average processing speed of 10 Features per quarter and a committed set of 30 Features, a new Feature will experience approximate wait time of:

$$\frac{30 \text{ items}}{10 \text{ items/Q}} = 3Q$$

#7 Apply cadence, synchronize with cross-domain planning

Cadence and synchronization

Cadence

- ▶ Converts unpredictable events into predictable occurrences and lowers cost
- ▶ Makes waiting times for new work predictable
- ▶ Supports regular planning and cross-functional coordination
- ▶ Limits batch sizes to a single interval
- ▶ Controls injection of new work
- ▶ Provides scheduled integration points

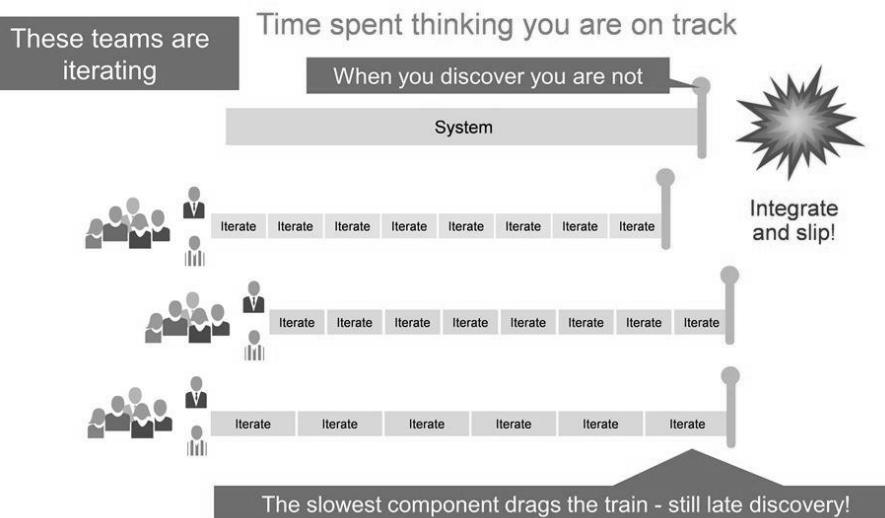
Note: Delivering on cadence requires scope or capacity margin

Synchronization

- ▶ Causes multiple events to happen simultaneously
- ▶ Facilitates cross-functional tradeoffs
- ▶ Provides routine dependency management
- ▶ Supports full system and integration and assessment
- ▶ Provides multiple feedback perspectives

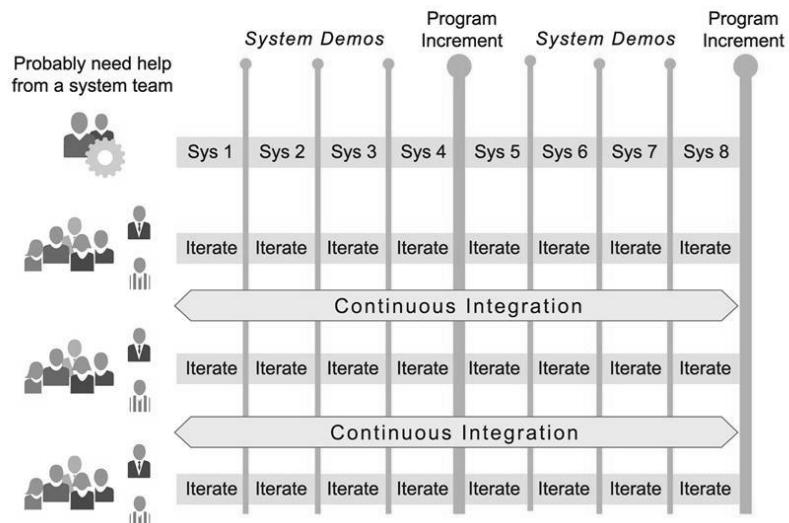
Note: To work effectively, design cycles must be synchronized

Cadence without synchronization is not enough



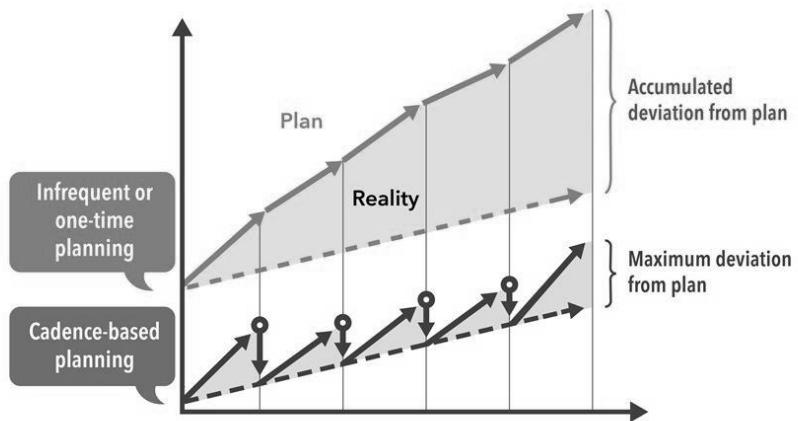
Synchronize to assure delivery

This system is iterating



Control variability with planning cadence

Cadence-based planning limits variability to a single interval.



Synchronize with cross-domain planning

Future product development tasks can't be pre-determined. Distribute planning and control to those who can understand and react to the end results.

—Michael Kennedy, *Product Development for the Lean Enterprise*

- ▶ All stakeholders meet face-to-face (but typically in multiple locations)
- ▶ Management sets the mission with minimum possible constraints
- ▶ Requirements and design happen
- ▶ Important stakeholder decisions are accelerated
- ▶ Teams create and take responsibility for plans



#8 Unlock the intrinsic motivation of knowledge workers

On managing knowledge workers

- ▶ Workers themselves are most qualified to make decisions about how to perform their work
- ▶ Workers must be heard and respected for management to lead effectively
- ▶ Knowledge workers have to manage themselves: they need autonomy
- ▶ Continuing innovation has to be part of the work and the responsibility of knowledge workers

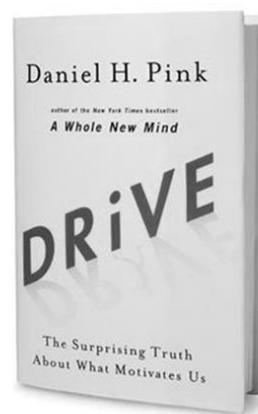


Workers are knowledge workers if they know more about the work they perform than their bosses.

— Peter Drucker

Unlocking intrinsic motivation with autonomy, mastery, and purpose

- ▶ *Autonomy* is the desire to be self-directing and have control over what we work on, how we do our work, and who we work with
- ▶ *Mastery* is the urge to get better at what we do and improve our personal and team skills
- ▶ *Purpose* is the desire to do something that matters and has meaning



#9 Decentralize decision-making

Video: Greatness by David Marquet

Duration
10 min



Decentralize decision-making

Define the economic logic behind a decision; empower others to make the changes.

| Centralize | De-centralize everything else |
|---|--|
| <ul style="list-style-type: none">▶ Infrequent - Not made very often and usually not urgent (example: <i>internationalization strategy</i>)▶ Long-lasting - Once made, highly unlikely to change (example: <i>common technology platform</i>)▶ Significant economies of scale - Provide large and broad economic benefit (example: <i>compensation strategy</i>) | <ul style="list-style-type: none">▶ Frequent - Routine, everyday decisions (example: <i>Team and Program Backlog</i>)▶ Time critical - High cost of delay (example: <i>point release to customer</i>)▶ Require local information - Specific and local technology or customer context is required (example: <i>Feature criteria</i>) |



Activity: Decentralize decision-making



- ▶ **Step 1:** Consider three significant decisions you are currently facing. Write them in the table provided in your workbook.
- ▶ **Step 2:** Rate each decision based on the frequency, time criticality, and economies of scale, assigning the value of 2 or 0.
- ▶ **Step 3:** Add the total values: 0 – 3 centralize and 4 – 6 decentralize

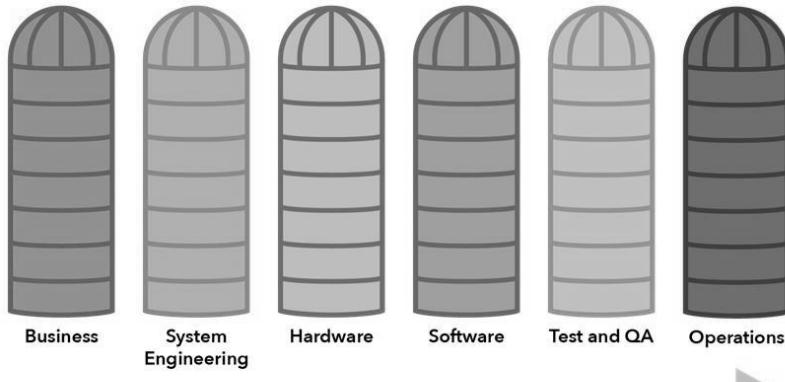
| Decision | Frequent? Y=2 N=0 | Time-critical? Y=2 N=0 | Economies of scale? Y=0 N=2 | Total |
|----------|----------------------|---------------------------|--------------------------------|-------|
| | | | | |
| | | | | |
| | | | | |

Keys to practicing decentralized decision making

- ▶ Openly discuss how decisions are made and explore opportunities to move authority for those decisions closer to where the work is performed.
- ▶ Establish a decision-making framework that equips knowledge workers with the information to make good decisions.
- ▶ Provide clarity on organizational objectives, coach effective problem-solving, and provide opportunities to exercise and cultivate decision-making abilities.
- ▶ Take responsibility for making and communicating strategic decisions—those that are infrequent, long lasting, and have significant economies of scale. Decentralize all others.

#10 Organize around value

Value doesn't follow silos



- Value delivery is inhibited by hand-offs and delays
- Political boundaries can prevent cooperation
- Silos encourage geographic distribution of functions
- Communication across silos is difficult

© Scaled Agile, Inc.

SCALED AGILE® © Scaled Agile, Inc.

120

Instead, organize around the flow of value

- ▶ A Value Stream is the sequence of steps used to deliver value to the Customer
- ▶ It includes the whole sequence—concept or customer order—to delivery of value and/or receipt of cash
- ▶ It contains the people who do the work, the systems, and the flow of information and materials

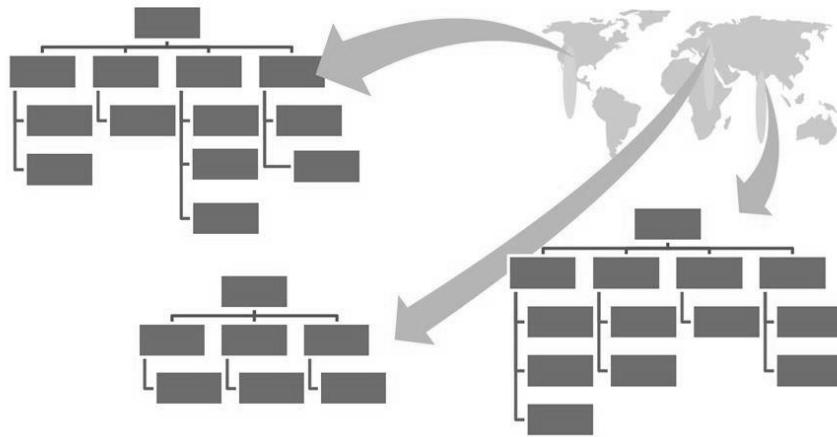


SCALED AGILE® © Scaled Agile, Inc.

121

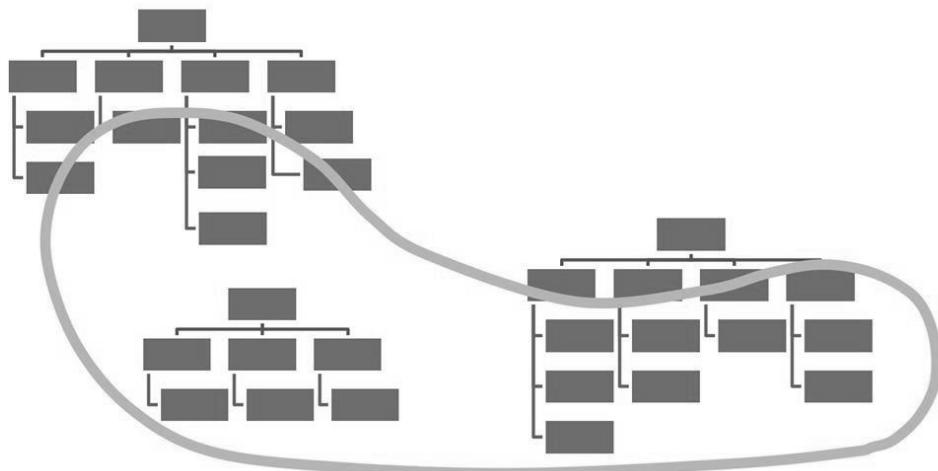
Value at scale is distributed

Value often flows across organizational boundaries.



Find the ‘kidney’

Use this thinking tool to identify the Value Stream within which to build one or more Agile Release Trains



Principles are great, but ...

*Clarity on how to **think**, without clarity on
how to **act**, leaves people unmoved.*

—Daniel Pink

... it's time to put this thinking to work.
Let's start doing.

124



Taking Action: Advocating SAFe Principles



- ▶ **Step 1:** Individually identify three actions you can take to model and advocate SAFe Principles in your Enterprise.
- ▶ **Step 2:** Write them down in your Action Plan.
- ▶ **Step 3: In your group,** share some of the insights you gained from SAFe Principles.



Lesson review

In this lesson you:

- ▶ 2.1 Embraced the Lean-Agile Mindset
- ▶ 2.2 Applied Lean and Agile at scale with the SAFe Principles

Leading SAFe®

Lesson 3: Establishing Team and Technical Agility



SAFe® Course - Attending this course gives students access to the SAFe® Agilist exam and related preparation materials.

SCALED AGILE®

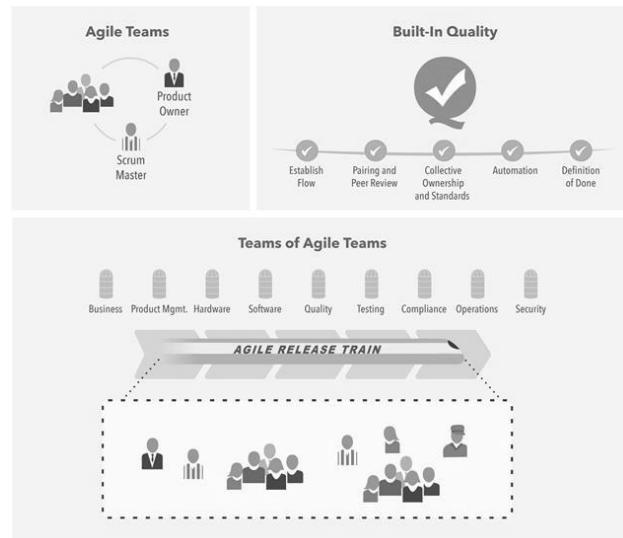
© Scaled Agile, Inc.

Learning objectives

At the end of this lesson you should be able to:

- ▶ 3.1 Form cross-functional Agile Teams
- ▶ 3.2 Build quality in
- ▶ 3.3 Organize Agile Release Trains (ARTs) around the flow of value

Team and Technical Agility

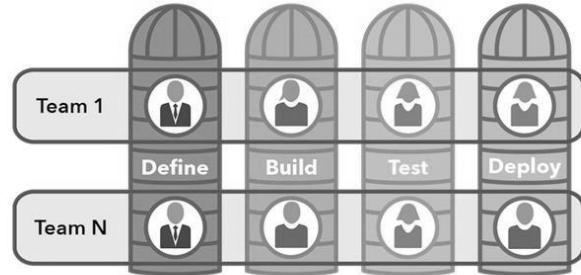


3.1 Form cross-functional Agile Teams

Build cross-functional Agile Teams

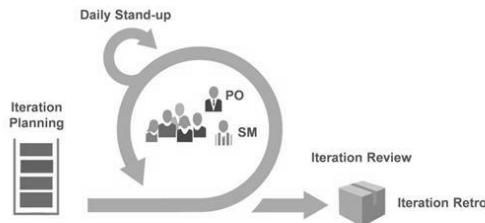
Agile Teams are cross-functional, self-organizing entities that can define, build and test, and where applicable, deploy increments of value

- ▶ Optimized for communication and delivery of value
- ▶ Deliver value every two weeks
- ▶ Two specialty roles:
 - Scrum Master
 - Product Owner

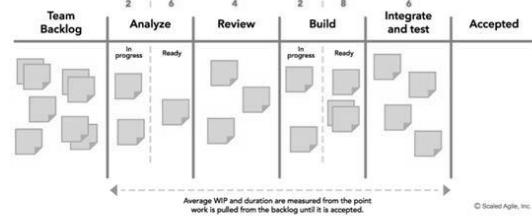


Teams execute Iterations with Scrum and Kanban

Scrum is built on transparency, inspection, adaptation, and short learning cycles



Kanban visualizes and optimizes the flow of work through the system



Agile business teams foster true Business Agility



Roles and responsibilities on the Agile Team



Agile Team

- Create and refine User Stories and acceptance criteria
- Define, build, test, and deliver Stories
- Develop and commit to team PI Objectives and Iteration plans
- Five to eleven members



Scrum Master

- Coaches the Agile Team and facilitates team meetings
- Removes impediments and protects the team from outside influence
- Attends scrum of scrum meetings



Product Owner

- Defines and accepts Stories
- Acts as the Customer for developer questions
- Works with Product Management to plan Program Increments (PI)



Activity: Identify team names and roles

Duration
2 min

- ▶ **Step 1:** Your team is your group. Create a team name.
- ▶ **Step 2:** Select a Scrum Master for your team.
- ▶ **Step 3:** Select a Product Owner for your team.
- ▶ **Step 4:** Make sure the team name and the names of the people selected are visible to all other teams
 - **Note:** In the next lesson, your team will experience PI Planning



3.2 Build quality in

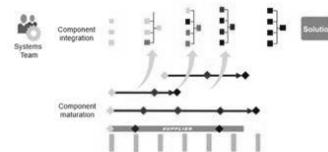
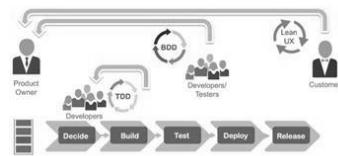
Build quality in

You can't scale crappy code (or hardware, or anything else).

- ▶ Ensures that every increment of the Solution reflects quality standards
- ▶ Is required for high, sustainable development velocity
- ▶ Many practices apply to every team, whether business or technology:
 - Establish flow
 - Peer review and pairing
 - Collective ownership and standards
 - Automation
 - Definition of done

Built-in Quality practices for technology-focused teams

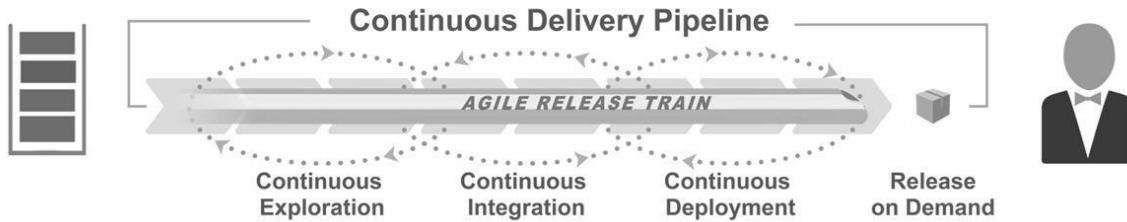
- ▶ Include software quality practices (most inspired by XP) like, Agile testing, behavior-driven development, test-driven development, refactoring, and code quality, Agile architecture
- ▶ Support hardware quality with exploratory, early iterations, frequent system-level integration, design verification, Model-Based Systems Engineering (MBSE), and Set-Based Design



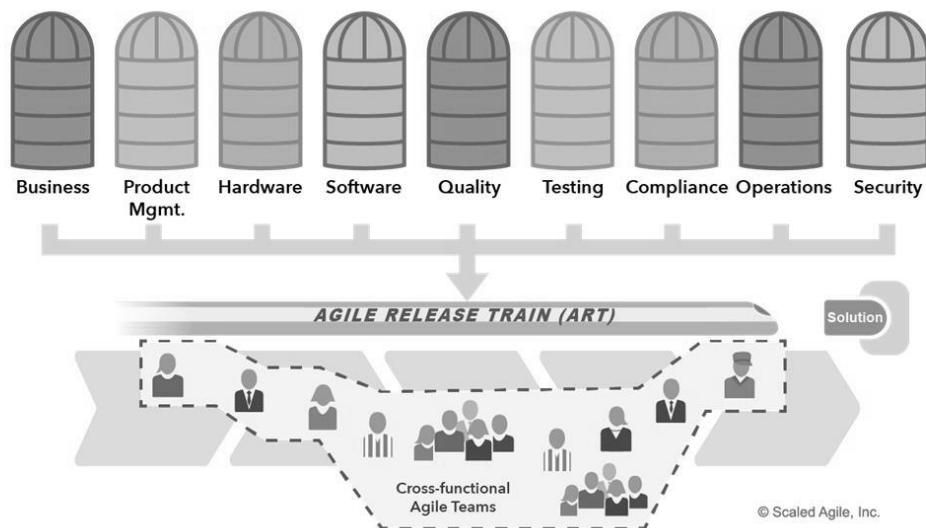
3.3 Organize Agile Release Trains (ARTs) around the flow of value

Agile Release Trains (ARTs) continuously deliver value

- ▶ A virtual organization of 5 – 12 teams (50 – 125+ individuals)
- ▶ Synchronized on a common cadence, a Program Increment (PI)
- ▶ Aligned to a common mission via a single Program Backlog



Create cross-functional Agile Release Trains



Roles on the Agile Release Train



Release Train Engineer acts as the chief Scrum Master for the train.



Product Management owns, defines, and prioritizes the Program Backlog.



System Architect/Engineering provides architectural guidance and technical enablement to the teams on the train.



System Team provides processes and tools to integrate and evaluate assets early and often.



Business Owners are key stakeholders on the Agile Release Train.

AGILE RELEASE TRAIN

Lesson review

In this lesson, you:

- ▶ Discussed how to form cross-functional Agile Teams
- ▶ Explored how to build quality in
- ▶ Discussed how to organize Agile Release Trains (ARTs) around the flow of value

Leading SAFe®

Lesson 4: Building Solutions with Agile Product Delivery



SAFe® Course - Attending this course gives students access to the SAFe® Agilist exam and related preparation materials.

SCALED AGILE®

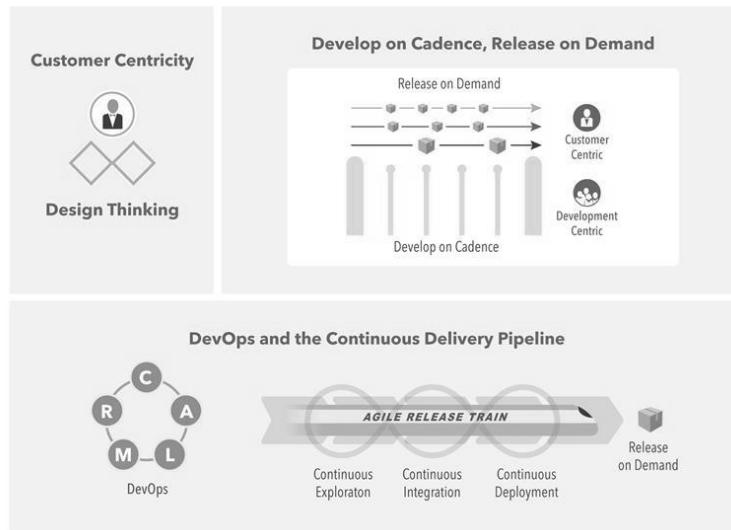
© Scaled Agile, Inc.

Learning objectives

At the end of this lesson you should be able to:

- ▶ 4.1 Apply Customer Centricity with Design Thinking
- ▶ 4.2 Prioritize the Program Backlog
- ▶ 4.3 Participate in PI Planning
- ▶ 4.4 Develop on Cadence; Release on Demand
- ▶ 4.5 Build a Continuous Delivery Pipeline with DevOps

Agile Product Delivery



4.1 Apply Customer Centricity with Design Thinking



Discussion: Customer Centricity

Prepare
3 min

Share
2 min

► **Step 1:** Discuss as a group:

- Why is it important to maintain focus on the Customer?
- What are some of the characteristics of a customer-centric Enterprise?

► **Step 2:** Be prepared to share with the class.



Why Customer Centricity?

Customer-centric Enterprises deliver whole-product Solutions that are designed with a deep understanding of Customer needs.

Customer-centric businesses generate:

- greater profits
- increased employee engagement
- more satisfied customers.



Customer-centric governments and nonprofits create:

- the resiliency, sustainability, and alignment needed to fulfill their mission.

Customer Centricity is a mindset

Whenever a customer-centric Enterprise makes a decision, it fully considers the effect it will have on its end users.

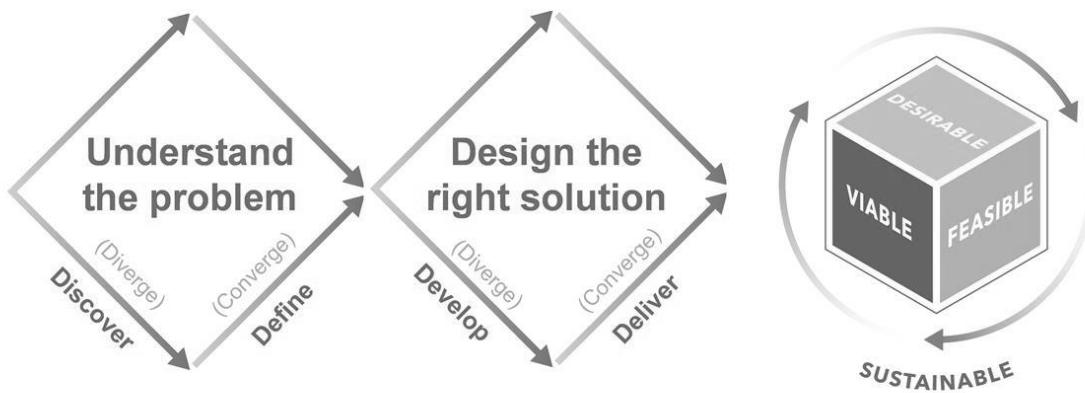


SCALED AGILE® © Scaled Agile, Inc.

150

What is Design Thinking?

Design Thinking is an iterative Solution development process that promotes a holistic approach to delighting stakeholders.



SCALED AGILE® © Scaled Agile, Inc.

151

Use personas to understand Customers

Personas are fictional characters based upon your research. They represent the different people who might use your product or Solution in a similar way.

- ▶ Convey the problems they're facing in context (i.e., their work environment) and key triggers for using the product
- ▶ Capture rich, concise information (photographs, family stories, jobs, etc.) that inspire great products without unnecessary details



Cary the Consumer

Age: 36
Location: Reno, Nevada, USA
Time in App: 10 minutes

| | | |
|--|--|--|
| <i>"I'm a working dad with three children ages 3, 6, and 10. I'm also in a band, which means I want to spend as much time as possible with my kids and my band. I need my package delivered on time so that I can maximize time with my family."</i> | | |
| I like technology! I have an iPhone, iPad, and nice home Wi-Fi setup | I'm not home on some weekends | I'd rather order online than dial the phone and talk to somebody |
| My wife also works during the week, so she doesn't have much spare time to help | Text is my favorite form of communication with suppliers | I don't own a computer, only tablets and phones |

Use empathy maps to identify with Customers

- ▶ The empathy map is a tool that helps teams develop deep, shared understanding and empathy for the Customer
- ▶ Use it to design better user experiences and Value Streams

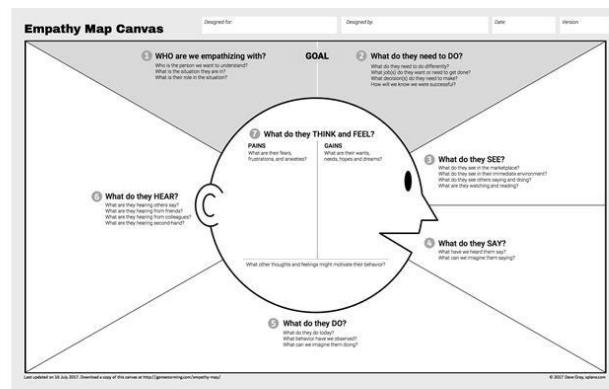




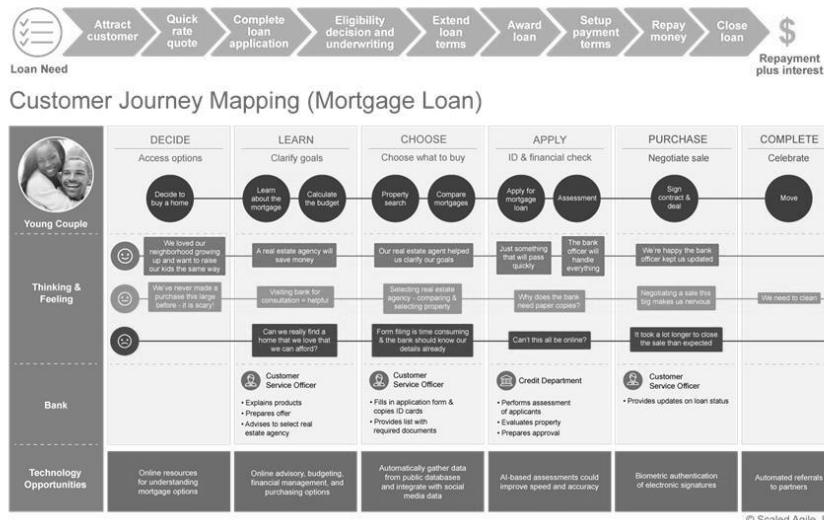
Activity: Empathy mapping

Prepare 7 min Share 3 min

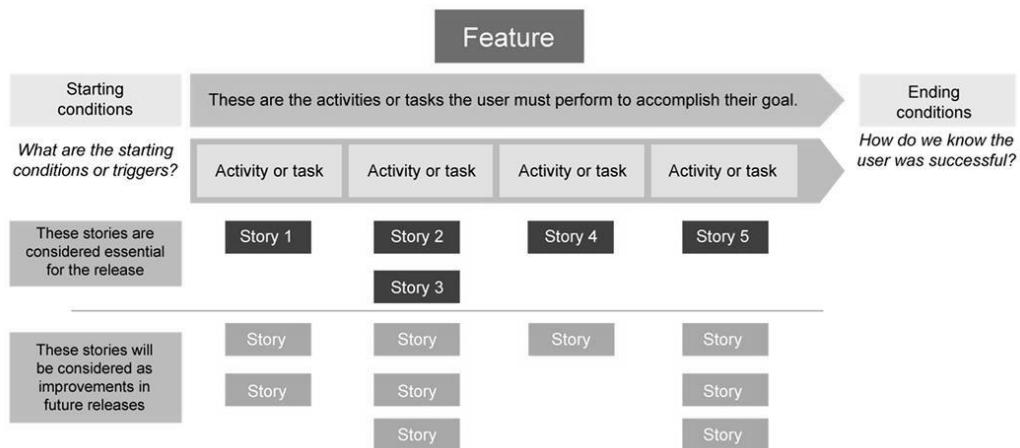
- Step 1:** In your group, create an empathy map using the example in your workbook on a flip chart or use the template provided.
- Step 2:** Select a user or customer of a product or service from one of the companies at your table.
- Step 3:** Following the sequence of numbers, fill in each section of the empathy map.
- Step 4:** Discuss with your group how the empathy map can inform Solution development. Be prepared to share your insights with the class.



Use journey maps to design the end-to-end Customer experience



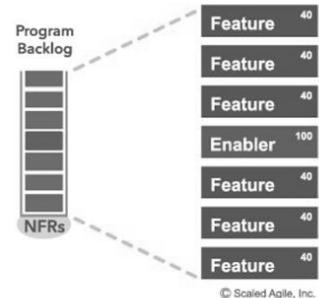
Use story maps to capture workflows



4.2 Prioritize the Program Backlog

Features are managed through the Program Backlog

The Program Backlog is the holding area for upcoming Features, that will address user needs and deliver business benefits for a single Agile Release Train (ART).



Vision aligns everyone on the product's direction

The Vision is a description of the future state of the product

- ▶ How will our product solve our customer's problems?
- ▶ What Features does it have?
- ▶ How will it differentiate us?
- ▶ What Nonfunctional Requirements does it deliver?



Features represent the work for the Agile Release Train

- ▶ Feature is an industry-standard term familiar to marketing and Product Management
- ▶ A benefit hypothesis justifies Feature implementation cost and provides business perspective when making scope decisions
- ▶ Acceptance criteria are typically defined during Program Backlog refinement
- ▶ Reflect functional and nonfunctional requirements
- ▶ Fits in one PI

Example:

Multi-factor authentication

Benefit hypothesis

Enhance user security via both password and a device.

Acceptance criteria

1. USB tokens as a first layer
2. Password authentication second layer
3. Multiple tokens on a single device
4. User activity log reflecting both authentication factors



Activity: Describe three Features



- ▶ **Step 1:** Individually identify three Features from your context
- ▶ **Step 2:** In your workbook, write down the Features and the benefit hypothesis for these Features
- ▶ **Step 3:** Choose one of the Features and write down some acceptance criteria for it

Example:

Feature:
Multi-factor authentication

Benefit Hypothesis:
Enhance user security via both password and a device.

Features are implemented by Stories

- ▶ Stories are small increments of value that can be developed in days and are relatively easy to estimate
- ▶ Story user-voice form captures role, activity, and goal
- ▶ Features fit in one PI for one ART; Stories fit in one Iteration for one Team

Enabler Story

Relocate mount for obstacle sensor to the top bracket so that it has a full 360° around the vehicle.

Enabler Stories represent different types of work, such as: *Exploration, Architecture, Infrastructure, Compliance*

Business Feature

Feature: Avoid obstacles unique to government installations
Benefit hypothesis: Characterize sensor's ability to detect and process obstacles unique to government installations

User Story

As an obstacle sensor I can track a single obstacle that continually changes speed and directions - like carts, pedestrians, forklifts, etc. So that vehicle control can respond to the obstacle's dynamic behavior

Estimate Stories with relative Story points

- ▶ A Story point is a singular number that represents:
 - Volume: How much is there?
 - Complexity: How hard is it?
 - Knowledge: What do we know?
 - Uncertainty: What's not known?
- ▶ Story points are relative. They are not connected to any specific unit of measure.

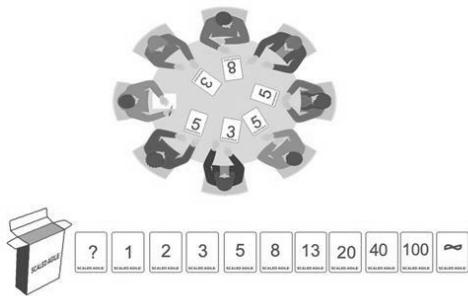
How big is it?



Guidance: Compared with other Stories, an 8-point Story should take relatively four times longer than a 2-point Story.

Apply estimating poker for fast, relative estimating

- ▶ Estimating poker combines expert opinion, analogy, and disaggregation for quick but reliable estimates
- ▶ All team members participate



Steps

- 1 Each estimator gets a deck of cards
- 2 Reads a job
- 3 Estimators privately select cards
- 4 Cards are turned over
- 5 Discuss differences
- 6 Re-estimate

Source: Mike Cohn, *Agile Estimating and Planning*

Estimation is a whole-team exercise

- ▶ Increases accuracy by including *all* perspectives
- ▶ Builds understanding
- ▶ Creates shared commitment



The whole team estimates Stories

Warning: Estimation performed by a Manager, Architect, or select group negates these benefits.

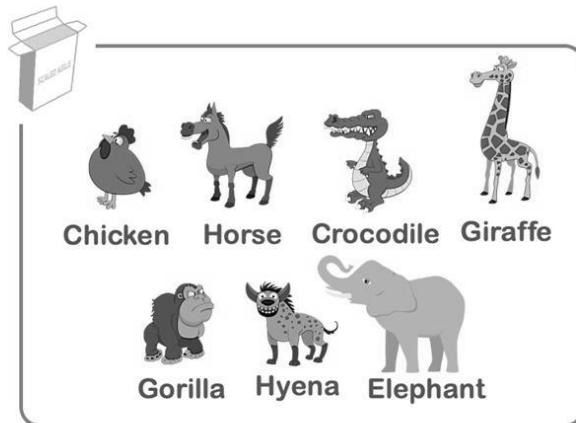


Activity: Relative size estimating



Use estimating poker to relatively estimate the mass of a set of animals.

- ▶ **Step 1:** In your groups, identify the smallest animal and mark it as 1
- ▶ **Step 2:** Estimate the remaining animals using values 1, 2, 3, 5, 8, 13, 20, 40, 100



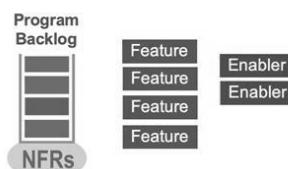
Prioritize Features for optimal ROI

In a flow system, *job sequencing* is the key to improving economic outcomes.

To prioritize based on Lean economics, we need to know two things:

- ▶ The cost of delay (CoD) in delivering value
- ▶ What is the cost to implement the valuable thing?

If you only quantify one thing, quantify the Cost of Delay. - Donald G. Reinertsen

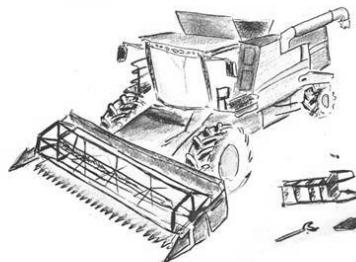


WSJF Done Right with Dean Leffingwell

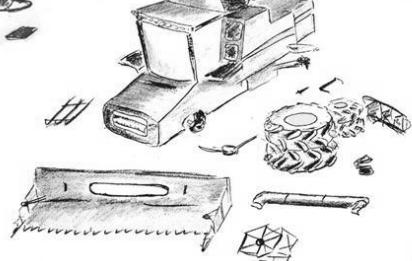
<https://share.vidyard.com/watch/an1uBZCb5UofhY9WTmH2Xx?>

Example with equal CoD: Which job first?

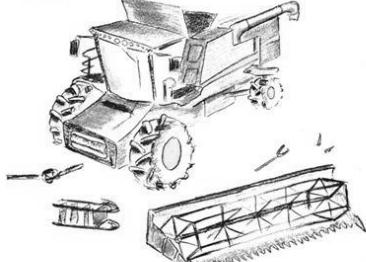
A \$\$, 1 day



C \$\$, 10 days

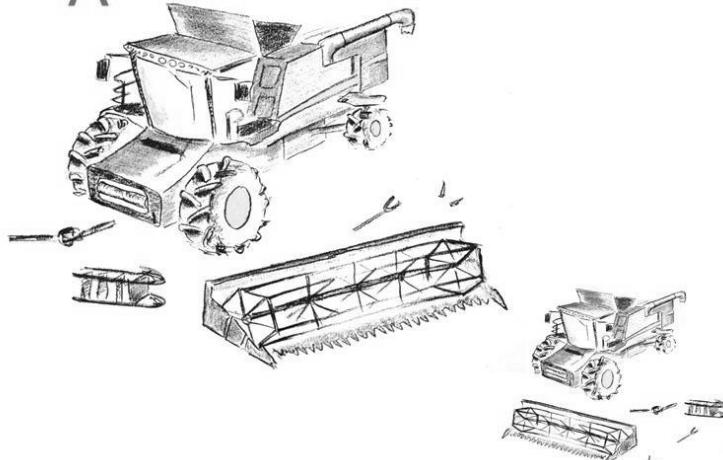


B \$\$, 3 days

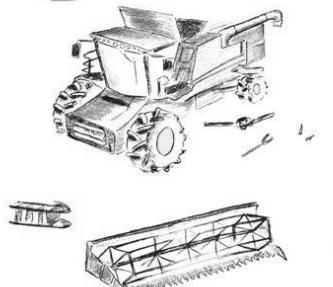


Example with equal duration: Which job first?

A \$\$\$, 3 days



B \$\$, 3 days

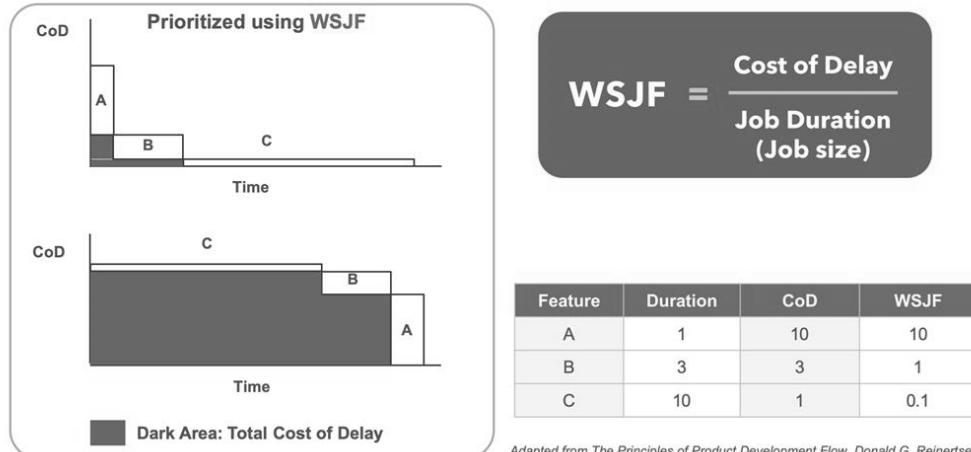


C \$, 3 days

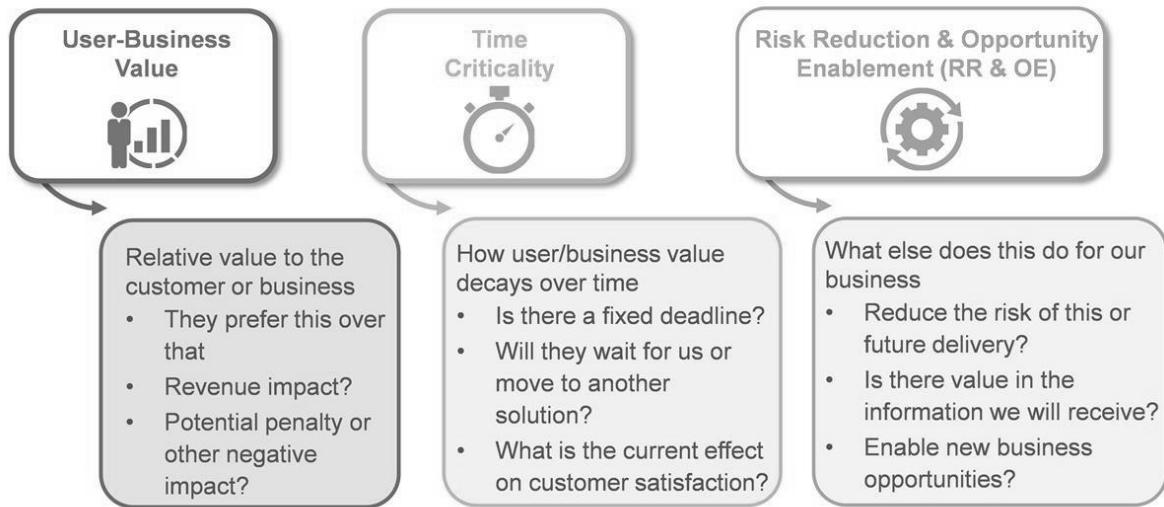


General case: Any Cost of delay (CoD) and duration

In the general case, give preference to jobs with shorter duration and higher CoD, using Weighted Shortest Job First (WSJF):



Components of cost of delay



Calculate WSJF with relative estimating

In order to calculate WSJF, teams need to estimate cost of delay and duration

- ▶ For duration, use job size as a quick proxy for duration
- ▶ Relative estimating is a quick technique to estimate job size and relative value
- ▶ WSJF stakeholders: Business Owners, Product Managers, Product Owners, and System Architects

$$\text{WSJF} = \frac{\text{User - Business Value} + \text{Time Criticality} + \text{Risk Reduction and/or Opportunity Enablement}}{\text{Job Size}}$$

© Scaled Agile, Inc.



Activity: Weighted Shortest Job First (WSJF) prioritization



- ▶ **Step 1:** Prioritize three of the Features you identified earlier using WSJF
- ▶ **Step 2:** Share some insights from this activity with the class

| Feature | User- business value | Time criticality | RR OE value | CoD | Job size | WSJF |
|---------|----------------------|------------------|---------------|-----|----------|------|
| | + | + | = | ÷ | = | |
| | + | + | = | ÷ | = | |
| | + | + | = | ÷ | = | |

Scale for each parameter: 1, 2, 3, 5, 8, 13, 20

Note: Do one column at a time, start by picking the smallest item and giving it a "1"

There must be at least one "1" in each column!

4.3 Participate in PI Planning

Program Increment Planning

Program Increment (PI) Planning is a cadence-based, face-to-face event that serves as the heartbeat of the Agile Release Train (ART), aligning all the teams on the ART to a shared mission and Vision.

- ▶ Two days every 8 – 12 weeks (10 weeks is typical)
- ▶ Everyone attends, in person if at all possible
- ▶ Product Management owns Feature priorities
- ▶ Agile Teams own Story planning and high-level estimates
- ▶ Architect/Engineering and UX work as intermediaries for governance, interfaces, and dependencies





Video: The Power of PI Planning

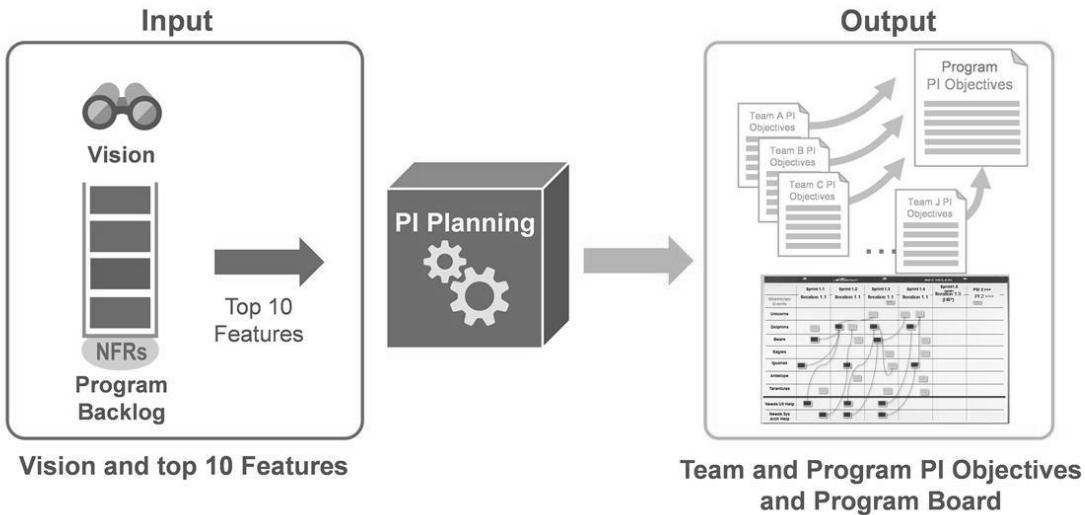
Duration
2 min



The benefits of PI Planning

- ▶ Establishing face-to-face communication across all team members and stakeholders
- ▶ Aligning development to business goals with the business context, Vision, and Team/Program PI Objectives
- ▶ Identifying dependencies and fostering cross-team and cross-ART collaboration
- ▶ Providing the opportunity for ‘just the right amount’ of architecture and Lean User Experience (UX) guidance
- ▶ Matching demand to capacity, eliminating excess work in process (WIP)
- ▶ Fast decision making

The PI Planning process



Align to a mission with PI Objectives

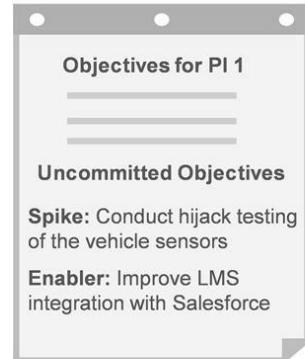
- ▶ Objectives are business summaries of what each team intends to deliver in the upcoming PI.
- ▶ They often map directly to the Features in the backlog.
- ▶ Other examples:
 - Aggregation of a set of Features
 - A Milestone like a trade show
 - An Enabler Feature supporting the implementation
 - A major refactoring

| Objectives for PI 1 | | Business Value | Actual Value |
|---|--|----------------|--------------|
| Structured location and validation of locations | | | |
| • Navigate autonomously from distribution center to top 5 most frequent destinations | | | |
| • Park at 1 building that requires parallel parking | | | |
| • Reduce GPS signal loss by 25% | | | |
| • Build and demonstrate proof of concept for next generation vehicle navigation systems | | | |
| Uncommitted Objectives | | | |
| • Spike: conduct hijack testing of the vehicle sensors | | | |

Maintain predictability with uncommitted objectives

Uncommitted objectives help improve the predictability of delivering business value.

- ▶ They are planned and aren't extra things teams do 'just in case you have time'
- ▶ They are not included in the commitment, thereby making the commitment more reliable
- ▶ If a team has low confidence in meeting a PI Objective, encourage them to move it to uncommitted
- ▶ If an item has many unknowns, consider moving it to uncommitted and put in early spikes
- ▶ Uncommitted objectives do count in velocity/capacity.



Prepare to experience a simulated PI Planning event

The flow of the simulation



You will be presented with the program Vision

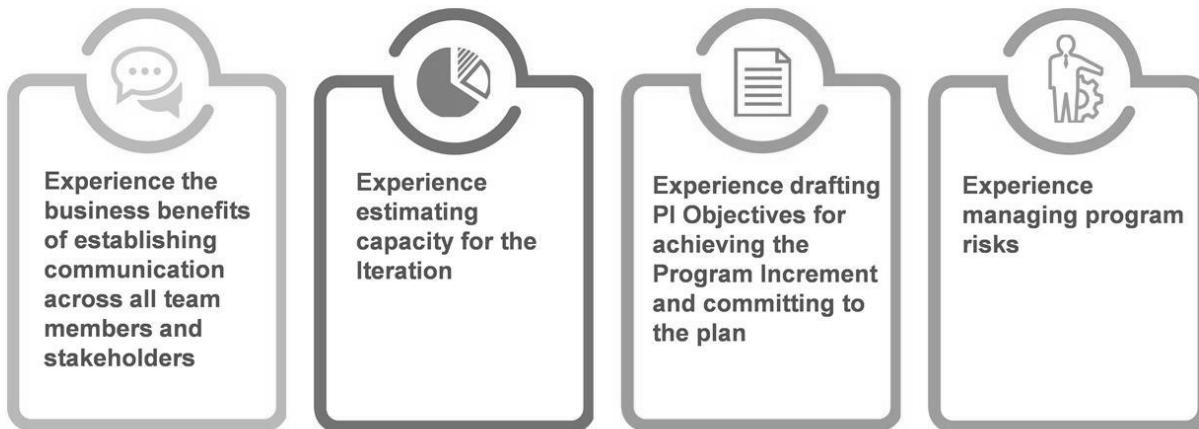
You will be involved in planning two Iterations considering Stories and Features

You will be drafting PI Objectives based on the program Vision and Features

You will be collaborating with the Business Owners to assign business value to the PI Objectives

Outcomes of the PI Planning simulation

Actively participating in a simulated PI Planning event will enable you to:



Activity: Identify program roles

Duration
3 min

- ▶ **Step 1:** Identify program roles for the simulation
- ▶ **Step 2:** Ensure that you have all key roles required for the PI Planning simulation

| Simulation role | Assigned to |
|--|-------------|
| Executive | Volunteer |
| Product Manager | Volunteer |
| System Architect, UX and Development Manager | Volunteer |

Example: Your Instructor will be the RTE. A volunteer will be the Product Manager, etc.



Simulation: Why are we here?



Alignment to a common mission

We are here to gain alignment and commitment around a clear set of prioritized objectives. I will now review the agenda for the next two days of the PI Planning Event.

PI Planning Agenda



Simulation: Day 1 agenda



Presented by RTE

| | | | |
|---------------|---|--|---|
| 8:00 ▶ 9:00 | Business Context | | ▶ State of the business and upcoming objectives |
| 9:00 ▶ 10:30 | Product/Solution Vision | | ▶ Vision and prioritized Features |
| 10:30 ▶ 11:30 | Architecture Vision and development practices | | ▶ Architecture, common frameworks, etc. ▶ Agile tooling, engineering practices, etc. |
| 11:30 ▶ 1:00 | Planning context and lunch | | ▶ Facilitator explains planning process |
| 1:00 ▶ 4:00 | Team breakouts | | ▶ Teams develop draft plans and identify risks and impediments ▶ Architects and Product Managers circulate |
| 4:00 ▶ 5:00 | Draft plan review | | ▶ Teams present draft plans, risks, and impediments |
| 5:00 ▶ 6:00 | Management review and problem solving | | ▶ Adjustments made based on challenges, risks, and impediments |



Simulation: Day 2 agenda



PI Planning Agenda
DAY 2



Presented by RTE

| | | | |
|------------------|--|--|--|
| 8:00 - 9:00 | Planning adjustments | | ▶ Planning adjustments made based on previous day's management meeting |
| 9:00 - 11:00 | Team breakouts | | ▶ Teams develop final plans and refine risks and impediments ▶ Business Owners circulate and assign business value to team objectives |
| 11:00 - 1:00 | Final plan review and lunch | | ▶ Teams present final plans, risks, and impediments |
| 1:00 - 2:00 | Program risks | | ▶ Remaining program-level risks are discussed and ROAMed |
| 2:00 - 2:15 | PI confidence vote | | ▶ Team and program confidence vote |
| 2:15 - ??? | Plan rework if necessary | | ▶ If necessary, planning continues until commitment is achieved |
| After commitment | Planning retrospective and moving forward | | ▶ Retrospective ▶ Moving forward ▶ Final instructions |

SCALED AGILE® © Scaled Agile, Inc.

186



Simulation: Briefings

Executive



Product Manager



System Architect



SCALED AGILE® © Scaled Agile, Inc.

187



Simulation: Planning guidance



Expect this first PI Planning to feel a bit chaotic. Future PI Planning meetings will become more routine.



Product Owners: You have the content authority to make decisions at the user Story level



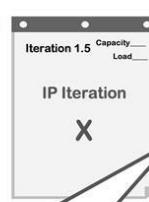
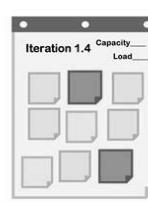
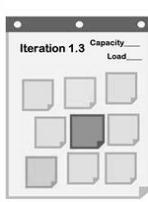
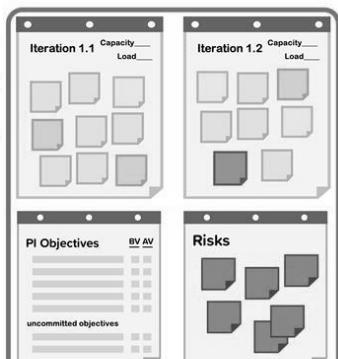
Scrum Masters: Your responsibility is to manage the timebox, the dependencies, and the ambiguities



Agile Team: Your responsibility is to define users Stories, plan them into the Iteration, and work out interdependencies with other teams



Simulation: Planning requirements

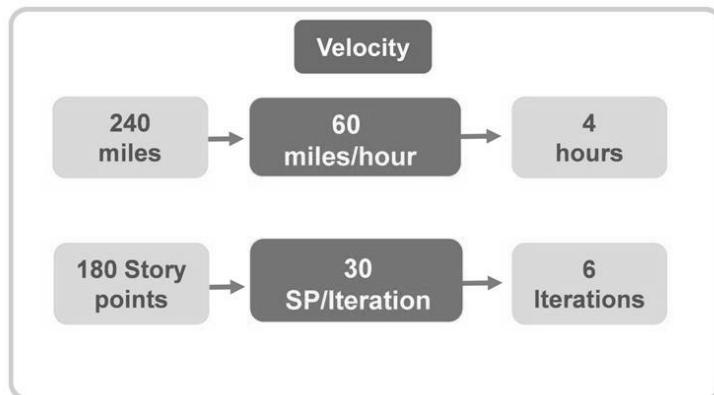


Focus on the highlighted area for this simulation.





Simulation: Using historical data to calculate velocity



Establish velocity by looking at the average output of the last Iterations.



Simulation: Calculate your capacity

Calculating Iteration capacity

- ▶ For every full-time Agile Team member contributing to Solution development, give the team 8 points (adjust for part-timers).
- ▶ Subtract 1 point for every team member vacation day and holiday.
- ▶ Find a small Story that would take about a half day to develop and a half day to test and validate. Call it a 1.
- ▶ Estimate every other Story relative to that one.

Example:

A 7-person team composed of 3 developers, 2 testers, 1 Product Owner, and 1 Scrum Master

Exclude The Scrum Master, Product Owner, and vacation time from the calculation

Calculated capacity:
 $5 \times 8 \text{ points} = 40 \text{ points per Iteration}$



Activity: Calculate your capacity

Duration
5 min

- ▶ **Step 1:** Review the example on the previous slide
- ▶ **Step 2:** Calculate your own capacity for the next two, 2-week Iterations
 - The first Iteration starts Monday
 - Use your real availability
- ▶ **Step 3:** Make sure you have your team's capacity calculated



Activity: Team breakout #1

Duration
45 min

You will be planning a short Program Increment with two Iterations.

- ▶ **Step 1:** Setup the team area. Enter the capacity for each Iteration.
- ▶ **Step 2:** Pick up a Feature from the Product Manager.
- ▶ **Step 3:** Estimate the Stories using Story Points.
- ▶ **Step 4:** Load the Stories into the Iterations.
- ▶ **Step 5:** Write the PI Objectives using clear statements.
- ▶ **Step 6:** Identify the uncommitted objectives.
- ▶ **Step 7:** Identify any program risks and dependencies.





Activity: Scrum of scrums (SoS) sync

Duration
5 min

- ▶ **Step 1:** Observe the SoS sync, conducted by the RTE
- ▶ **Step 2:** Each team's Scrum Master provides the team's current status and addresses the questions from the RTE
- ▶ **Step 3:** The RTE holds a meet-after after the sync (limited to 1 – 2 topics for the simulation)

| SoS Sync Questions | Team 1 | Team 2 |
|--|--------|--------|
| Have you identified the capacity for each Iteration in the PI? | | |
| Have you identified most of the Stories for the first two Iterations and begun estimating? | | |
| Have you begun resolving dependencies with other teams? | | |
| Are you discussing trade-offs and conflicting priorities with your Business Owners? | | |
| Have you identified any program risks? | | |
| Will you be ready to start writing PI Objectives in the next 15 minutes? | | |
| Is there anything you need to discuss with other Scrum Masters? If so, stay for the meet-after | | |



Activity: Draft plan review

Prepare
10 min

- ▶ **Step 1:** Present the summary of your team's first two Iterations and one or more draft PI Objectives
- ▶ **Step 2:** Make sure that you have included the following:
 - Capacity and load for each Iteration
 - Draft PI Objectives
 - Program risks and impediments

Management review and problem-solving

At the end of day 1, management meets to make adjustments to scope and objectives based on the day's planning.

Common questions during the managers' review:

- ▶ What did we just learn?
- ▶ Where do we need to adjust? Vision? Scope? Team assignments?
- ▶ Where are the bottlenecks?
- ▶ What features must be de-scoped?
- ▶ What decisions must we make between now and tomorrow to address these issues?



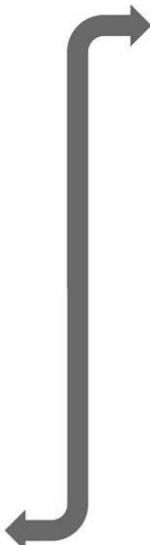
Activities during day 2

Day 1

| | | |
|---------------|---|--|
| 8:00 ▶ 9:00 | Business Context | |
| 9:00 ▶ 10:30 | Product/Solution Vision | |
| 10:30 ▶ 11:30 | Architecture Vision and development practices | |
| 11:30 ▶ 1:00 | Planning context and lunch | |
| 1:00 ▶ 4:00 | Team breakouts | |
| 4:00 ▶ 5:00 | Draft plan review | |
| 5:00 ▶ 6:00 | Management review and problem solving | |

Day 2

| | | |
|------------------|---|--|
| 8:00 ▶ 9:00 | Planning adjustments | |
| 9:00 ▶ 11:00 | Team breakouts | |
| 11:00 ▶ 1:00 | Final plan review and lunch | |
| 1:00 ▶ 2:00 | Program risks | |
| 2:00 ▶ 2:15 | PI confidence vote | |
| 2:15 ▶ ??? | Plan rework if necessary | |
| After commitment | Planning retrospective and moving forward | |



Make planning adjustments

- ▶ Based on the previous day's management review and problem-solving meeting, adjustments are discussed.
- ▶ Possible changes:
 - Business priorities
 - Adjustment to Vision
 - Changes to scope
 - Realignment of work and teams



Team breakout #2

Based on new knowledge and a good night's sleep, teams work to create their final plans.

- ▶ In the second team breakout, Business Owners circulate and assign business value to PI Objectives from low (1) to high (10)
- ▶ Teams finalize the Program Increment plan
- ▶ Teams also consolidate program risks, impediments, and dependencies
- ▶ Uncommitted objectives provide the capacity and guard band needed to increase the reliability of cadence-based delivery

| Team A | PI Objectives | BV |
|---|---------------|----|
| ■ Proof of concept with mock sounds | 10 | |
| ■ Help with radar POC | 4 | |
| ■ Decide to create or buy engine noises | 3 | |
| <hr/> Uncommitted <hr/> | | |
| ■ Proof of concept with real sounds | 7 | |



Activity: Setting business value

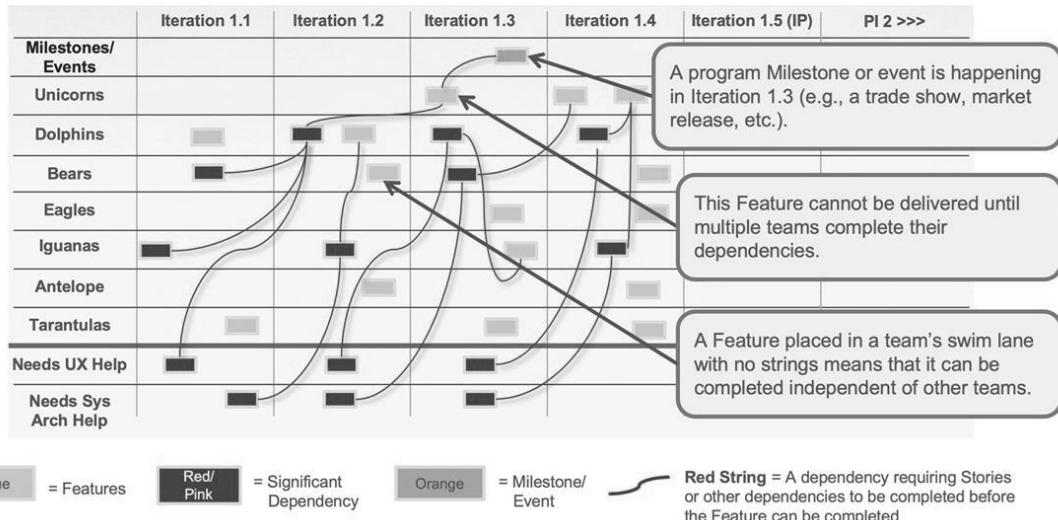
Duration
10 min

The instructor will demonstrate assigning business value for one team's objectives.

- ▶ **Step 1:** Bring the Business Owners to one team's draft plans
- ▶ **Step 2:** The Business Owners will set value on a scale of 1 – 10 for each identified objective
- ▶ **Step 3:** Observe the discussion that would take place, illustrating the larger purposes and thought processes around assigning business value

| Team A | PI Objectives | BV |
|---|---------------|----|
| ■ Proof of concept with mock sounds | 10 | |
| ■ Help with radar POC | 4 | |
| ■ Decide to create or buy engine noises | 3 | |
| Uncommitted | | |
| ■ Proof of concept with real sounds | 7 | |

Program Board - Feature delivery, dependencies, and Milestones



Final plan review

Teams and Business Owners peer-review all final plans.



Final plan review

Used with permission of SEI Global Wealth Services

Building the final plan

- ▶ Final plans are collected at the front of the room
- ▶ Final plans are reviewed by all teams
- ▶ Business Owners are asked whether they accept the plan
- ▶ If so, the team's plan and program risk sheet are brought to the front of the room
- ▶ If not, the plans stay in place, and the team continues planning after the review



A team's final plan

Used with permission of Discount Tire Corporation

Addressing program risks

After all plans have been presented, remaining program risks and impediments are discussed and categorized.

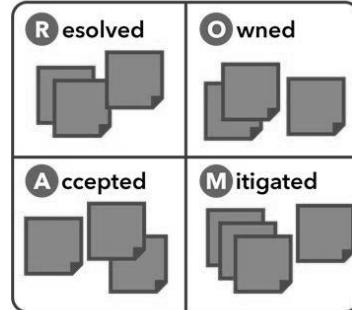
ROAMing risks:

Resolved - Has been addressed. No longer a concern.

Owned - Someone has taken responsibility.

Accepted - Nothing more can be done. If risk occurs, release may be compromised.

Mitigated - Team has plan to adjust as necessary.

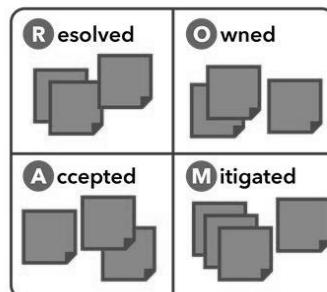


Activity: Manage program risks



The instructor will demonstrate ROAMing one to two risks for one team.

- ▶ **Step 1:** Pick one to two risk examples.
- ▶ **Step 2:** Read them in front of all teams and stakeholders.
- ▶ **Step 3:** Ask if anyone can own, help mitigate, or resolve the risks. Otherwise, accept as is.
- ▶ **Step 4:** Put each risk into a corresponding quadrant of the ROAM sheet for the program.



Confidence vote: Team and program

After dependencies are resolved and risks are addressed, a confidence vote is taken by the team and program.

A commitment with two parts:

1. Teams agree to do everything in their power to meet the agreed-to objectives
2. In the event that fact patterns dictate that it is simply not achievable, teams agree to escalate immediately so that corrective action can be taken



No confidence Little confidence Good confidence High confidence Very high confidence



Run a planning meeting retrospective

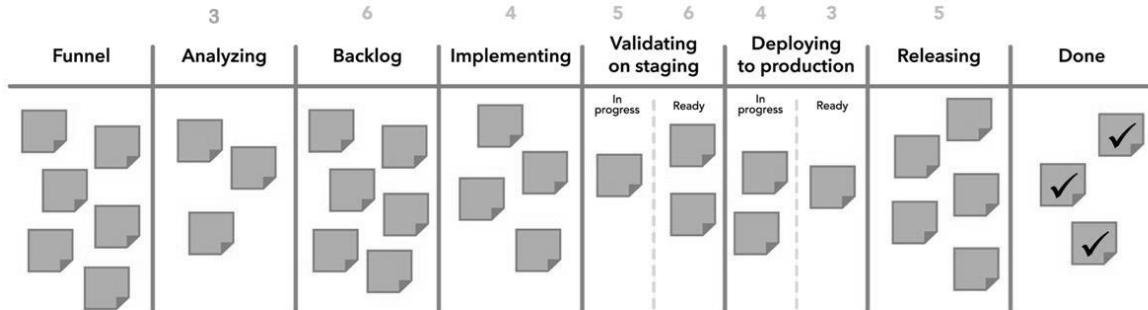
The PI planning event will evolve over time. Ending with a retrospective will help continuously improve it.



A Team's Retrospective

4.4 Develop on Cadence; Release on Demand

Manage the flow of work with the Program Kanban

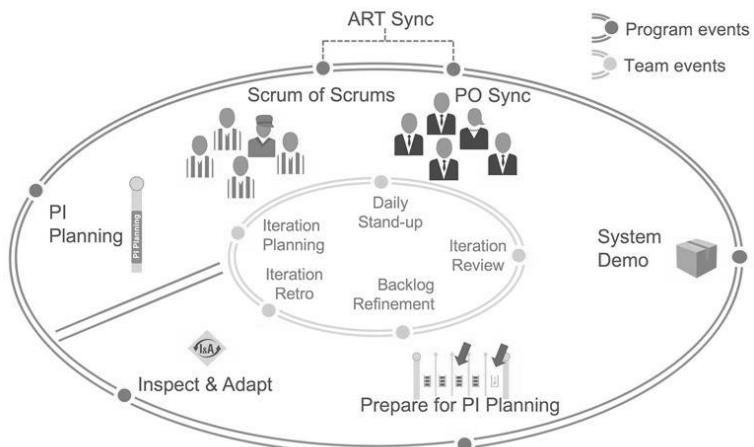


An example of a Program Kanban



Program events drive the train

Program events create a closed-loop system to keep the train on the tracks.



ART sync is used to coordinate progress

Programs coordinate dependencies through sync meetings.



Scrum of scrums

- ▶ Visibility into progress and impediments
- ▶ Facilitated by RTE
- ▶ Participants: Scrum Masters, other select team members, SMEs if necessary
- ▶ Weekly or more frequently, 30–60 minutes
- ▶ Timeboxed and followed by a 'Meet After'

ART Sync

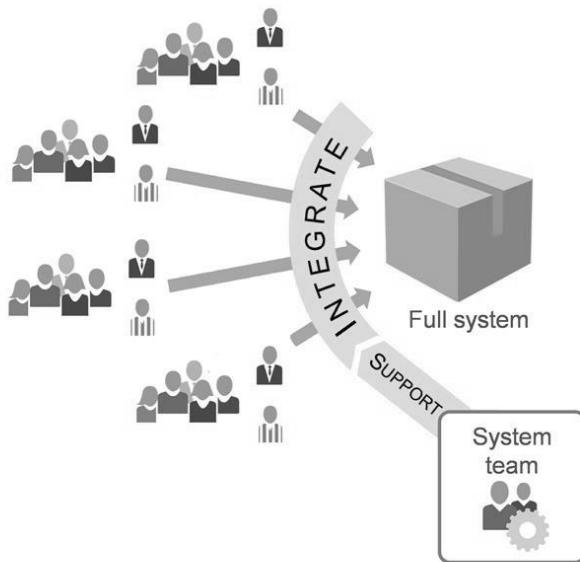


PO Sync

- ▶ Visibility into progress, scope, and priority adjustments
- ▶ Facilitated by RTE or PM
- ▶ Participants: PMs, POs, other stakeholders, and SMEs as necessary
- ▶ Weekly or more frequently, 30–60 minutes
- ▶ Timeboxed and followed by a 'Meet After'

Demo the full system increment every two weeks

- ▶ Features are functionally complete or 'toggled' so as not to disrupt demonstrable functionality
- ▶ New Features work together and with existing functionality
- ▶ Happens after the Iteration review (may lag by as much as one Iteration, maximum)
- ▶ Demo from a staging environment which resembles production as much as possible



Innovation and Planning (IP) Iteration

Provide sufficient capacity margin to enable cadence. —Donald G. Reinertsen

Facilitate reliability, Program Increment readiness, planning, and innovation

- ▶ **Innovation:** Opportunity for innovation, hackathons, and infrastructure improvements
- ▶ **Planning:** Provides for cadence-based planning
- ▶ Estimating **guard band** for cadence-based delivery

Example IP Iteration calendar

| Monday | Tuesday | Wednesday | Thursday | Friday |
|--------|-----------------------|---|---|---|
| 1 | 2 | 3 | 4 | 5 |
| | | Buffer for leftover work | | |
| | | Final verification and validation, and documentation (if releasing) | | |
| | | Innovation | | |
| | | PI planning readiness | | |
| 8 | 9 | 10 | 11 | 12 |
| | Continuing education | | PI planning | Optional time for distributed planning |
| | Innovation continues | Inspect & Adapt Event | Business context Product / solution vision Architecture vision and development practices Planning requirements and lunch Team breakouts Draft plan review Management review and problem-solving | Planning adjustments Team breakouts Final plan review and lunch Program risks PI confidence vote Plan rework if necessary Planning retrospective and moving forward |
| | PI planning readiness | | | |

Without the IP Iteration...

- ▶ Lack of delivery capacity buffer impacts predictability
- ▶ Little innovation; tyranny of the urgent
- ▶ Technical debt grows uncontrollably
- ▶ People burn out
- ▶ No time for teams to plan, demo, or improve together



Improving results with the Inspect and Adapt event

Three parts of Inspect and Adapt:

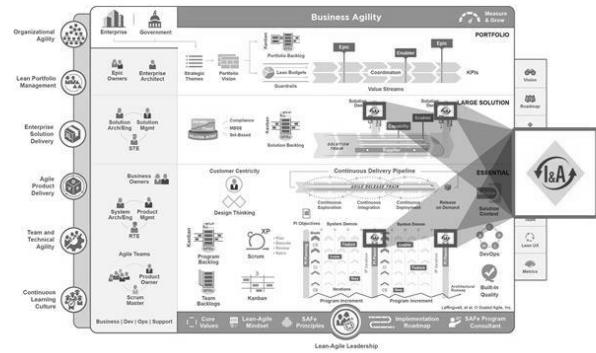
1. The PI System Demo

2. Quantitative and Qualitative Measurement

3. Problem-Solving Workshop

Timebox: 3 – 4 hours per PI

Attendees: Teams and stakeholders



PI System Demo

At the end of the PI, teams demonstrate the current state of the Solution to the appropriate stakeholders.

- ▶ Often led by Product Management, POs, and the System Team
- ▶ Attended by Business Owners, program stakeholders, Product Management, RTE, Scrum Masters, and teams



Program performance reporting

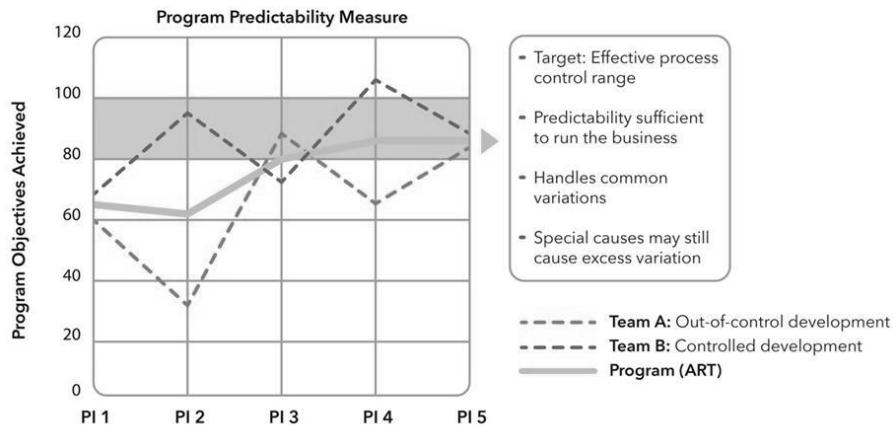
As part of the PI System Demo, teams compare planned vs actual PI Objectives.

- ▶ Teams meet with their Business Owners to self-assess the business value they achieved for each objective
- ▶ Each team's planned vs actual business value is then rolled up to the program predictability measure.

| Objectives for PI 3 | Business Value | |
|---|----------------|-----------|
| | Plan | Actual |
| - Structured locations and validation of locations | 7 | 7 |
| - Build and demonstrate a proof of concept for context images | 8 | 8 |
| - Implement negative triangulation by: tags, companies and people | 8 | 6 |
| - Speed up indexing by 50% | 10 | 5 |
| - Index 1.2 billion more web pages | 10 | 8 |
| - Extract and build URL abstracts | 7 | 7 |
| Uncommitted Objectives | | |
| - Fuzzy search by full name | 7 | 0 |
| - Improve tag quality to 80% relevance | 4 | 4 |
| Totals | 50 | 45 |
| % Achievement: | 90% | |

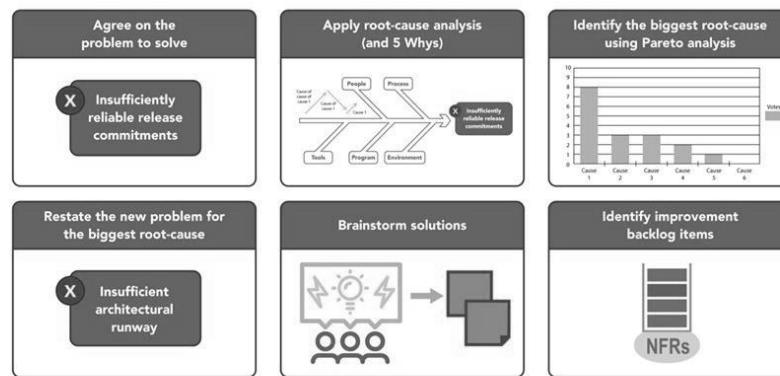
Quantitative and qualitative measurement

The report compares actual business value achieved to planned business value.



The problem-solving workshop

Teams conduct a short retrospective to systematically address the larger impediments that are limiting velocity.



4.5 Build a Continuous Delivery Pipeline with DevOps



Activity: DevOps myth or fact



- ▶ **Step 1:** Take the myth or fact quiz in your workbook
- ▶ **Step 2:** Check your results with the answer key at the bottom of page that follows the quiz





Video: What is DevOps?

Duration
2 min



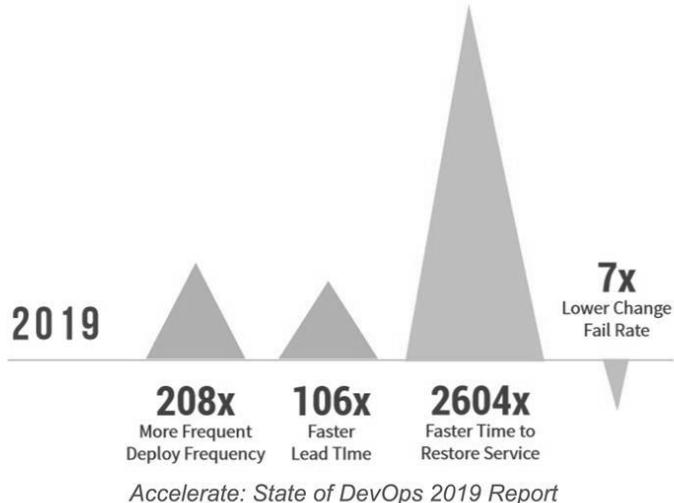
Who is DevOps?



SCALED AGILE® © Scaled Agile, Inc.

225

Achieve higher performance with DevOps

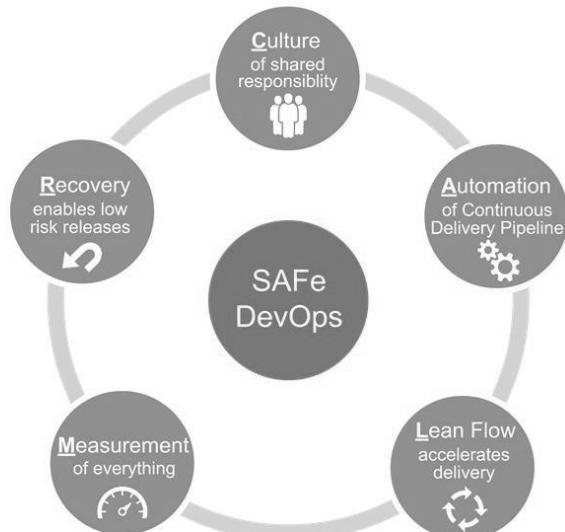


SCALED AGILE® © Scaled Agile, Inc.

226

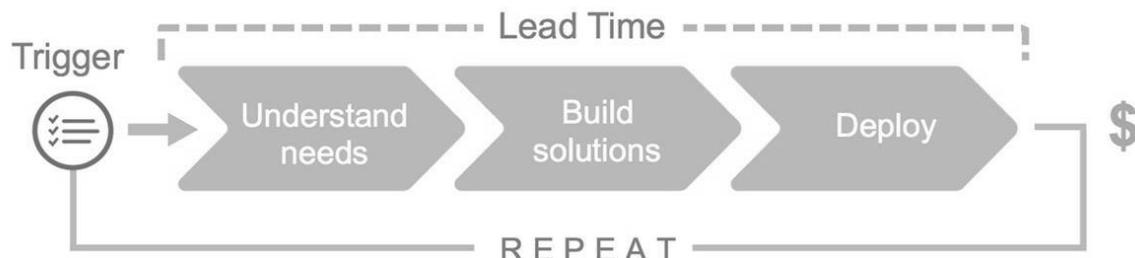
A CALMR approach to DevOps

- ▶ **Culture** - Establish a culture of shared responsibility for development, deployment, and operations.
- ▶ **Automation** - Automate the Continuous Delivery Pipeline.
- ▶ **Lean flow** - Keep batch sizes small, limit WIP, and provide extreme visibility.
- ▶ **Measurement** - Measure the flow through the pipeline. Implement full-stack telemetry.
- ▶ **Recovery** - Architect and enable low-risk releases. Establish fast recovery, fast reversion, and fast fix-forward.



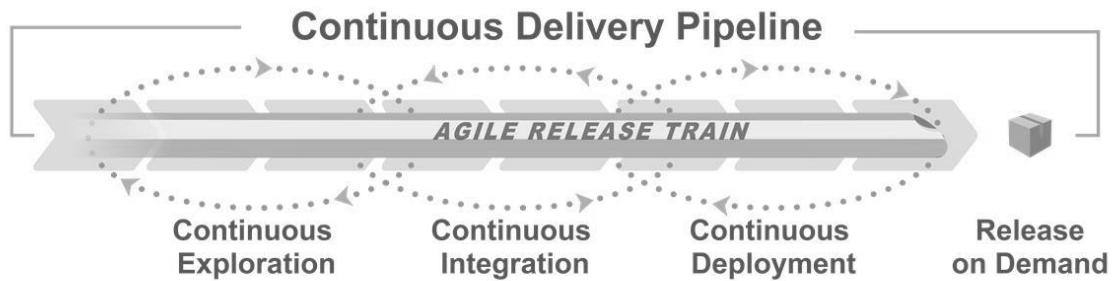
DevOps is in the Value Stream

Value occurs only when the end users are operating the Solution.

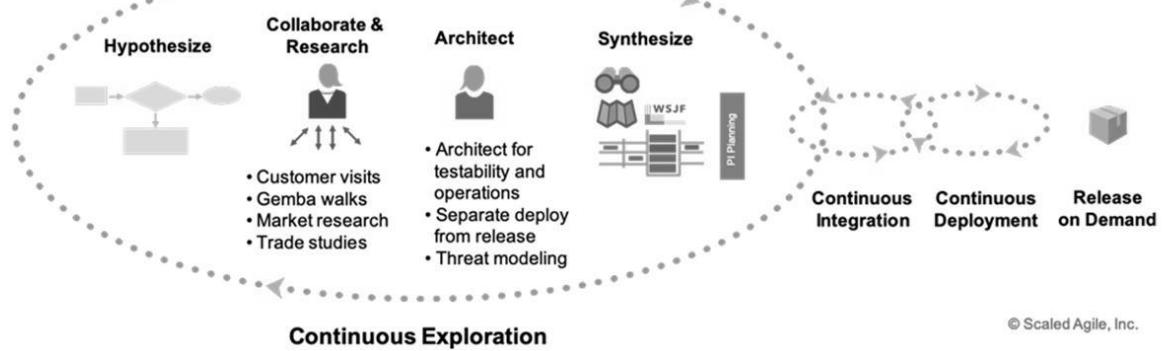


DevOps isn't optional. The only question is how efficient it is.

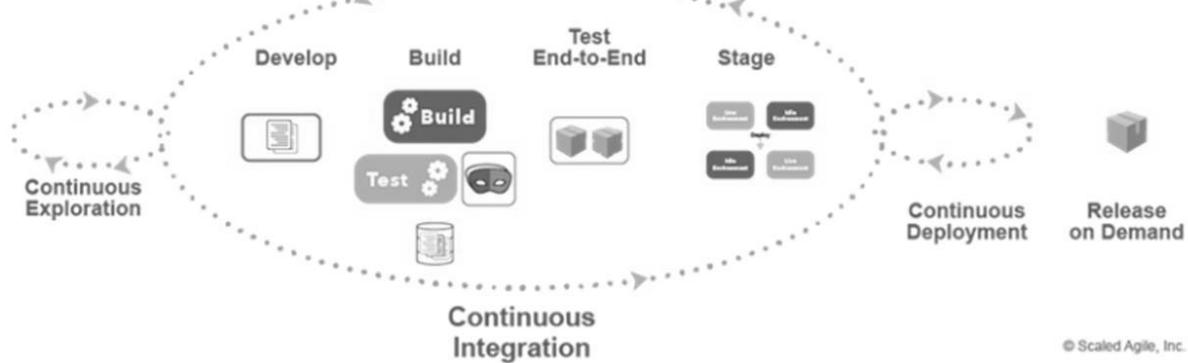
The Continuous Delivery Pipeline enables the flow of value



Continuous Exploration – Understand Customer needs

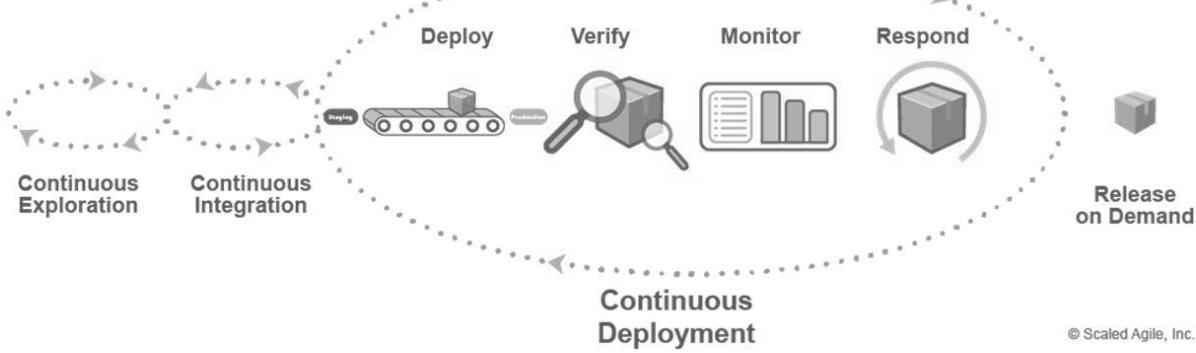


Continuous Integration – A critical technical practice of the ART



© Scaled Agile, Inc.

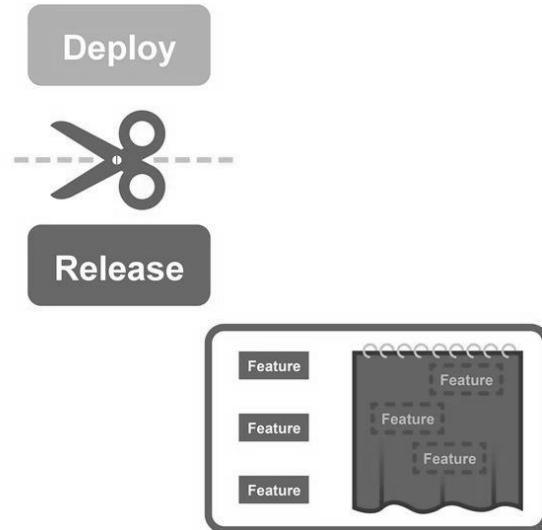
Continuous Deployment – Getting to production early



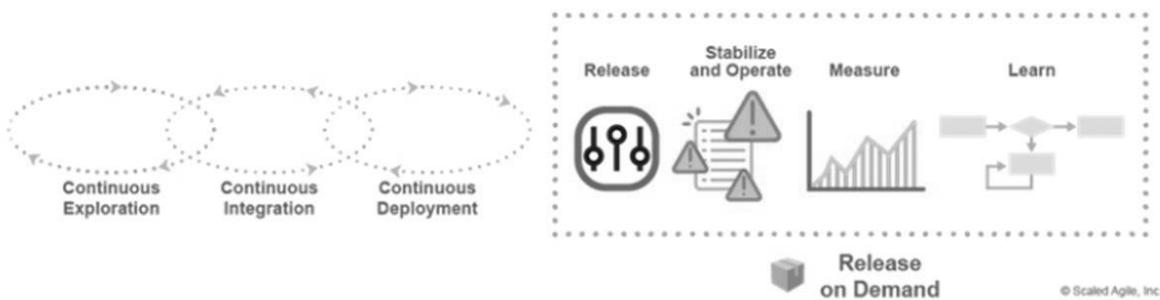
© Scaled Agile, Inc.

Separate deploy from release

- ▶ Separate deploy to production from release
- ▶ Hide all new functionality under feature toggles
- ▶ Enables testing background and foreground processes in the actual production environment before exposing new functionality to users



Release on Demand – Making value available when it's needed

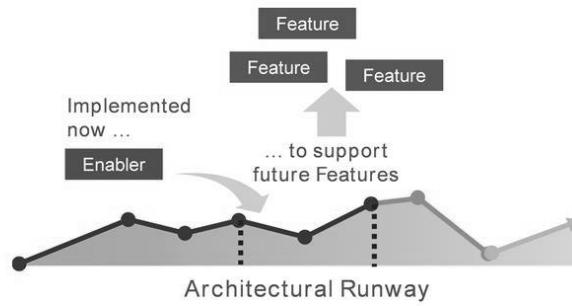


Architect for releasability

Architectural Runway is existing code, hardware components, marketing branding guidelines, etc., that enable near-term business Features.

- ▶ Enablers build up the runway
- ▶ Features consume it
- ▶ Architectural Runway must be continuously maintained
- ▶ Use capacity allocation (a percentage of train's overall capacity in a PI) for Enablers that extend the runway

Example:
A single sign-on mechanism will enable sign-on in multiple applications.



Taking Action: Improving Agile Product Delivery



- ▶ **Step 1:** Consider the practices and the events that support Agile Product Delivery, as discussed earlier
- ▶ **Step 2:** Identify three minimum viable improvements you could execute to improve Agile Product Delivery. Write them down in your Action Plan
- ▶ **Step 3:** Share your insights with the class



Lesson review

In this lesson you:

- ▶ Applied Customer Centricity with Design Thinking
- ▶ Prioritized the Program Backlog
- ▶ Participated in PI Planning
- ▶ Explored how to develop on cadence and release on demand
- ▶ Discussed how to build a Continuous Delivery Pipeline with DevOps

Leading SAFe®

Lesson 5: Exploring Lean Portfolio Management



SAFe® Course - Attending this course gives students access to the SAFe® Agilist exam and related preparation materials.

SCALED AGILE®

© Scaled Agile, Inc.

Learning objectives

At the end of this lesson, you should be able to:

- ▶ 5.1 Define the SAFe portfolio
- ▶ 5.2 Connect the portfolio to the Enterprise strategy
- ▶ 5.3 Maintain the Portfolio Vision
- ▶ 5.4 Establish portfolio flow
- ▶ 5.5 Fund Value Streams

5.1 Define a SAFe portfolio

The role of Lean Portfolio Management (LPM)

*Most strategy dialogues end up with executives talking at cross-purposes because ... nobody knows exactly what is meant by **vision** and **strategy**, and no two people ever quite agree on which topics belong where.*

That is why, when you ask members of an executive team to describe and explain the corporate strategy, you frequently get wildly different answers. We just don't have a good business discipline for converging on issues this abstract.

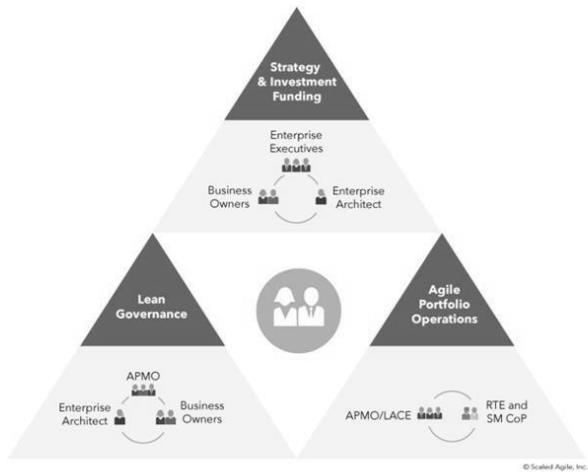
—Geoffrey Moore, *Escape Velocity*



Lean Portfolio Management empowers the portfolio

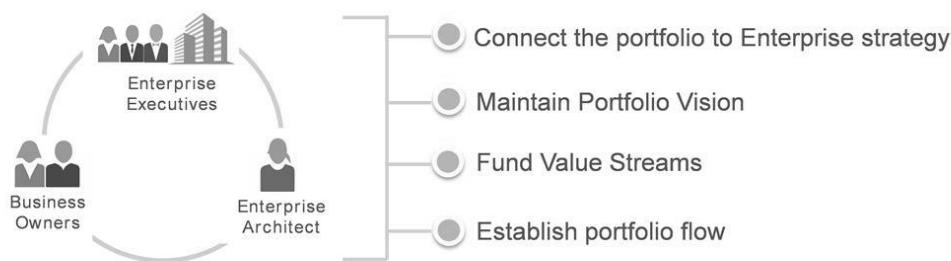
The LPM function governs a SAFe portfolio, providing three essential collaborations to realize its responsibilities:

- ▶ *Strategy and investment funding*
- ▶ Agile portfolio operations
- ▶ Lean governance



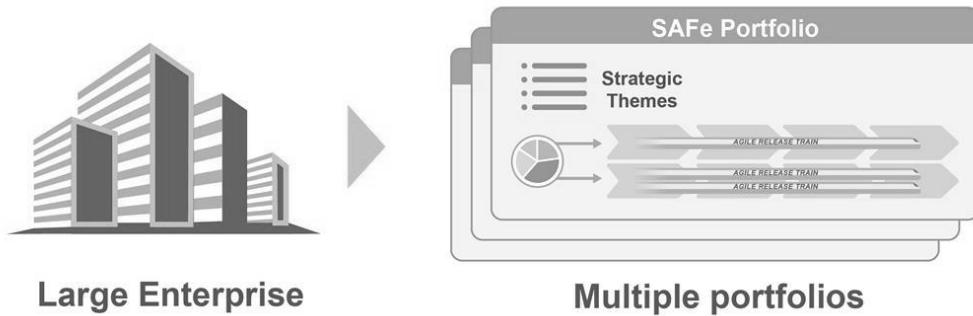
Strategy and investment funding is a collaboration

- ▶ Key stakeholders collaborate to develop and communicate the portfolio strategy
- ▶ They provide Lean Budgeting and funding to the Value Streams that develop and maintain the portfolio products and services
- ▶ They build a Portfolio Kanban system to establish flow



Large Enterprises will have multiple portfolios

In larger Enterprises, there can be multiple SAFe portfolios, typically one for each line of business, business unit, or division.



Define the portfolio with the Portfolio Canvas

- ▶ The Portfolio Canvas is a template for identifying a specific SAFe portfolio
- ▶ It defines the domain of the portfolio and other key elements

SCALED AGILE

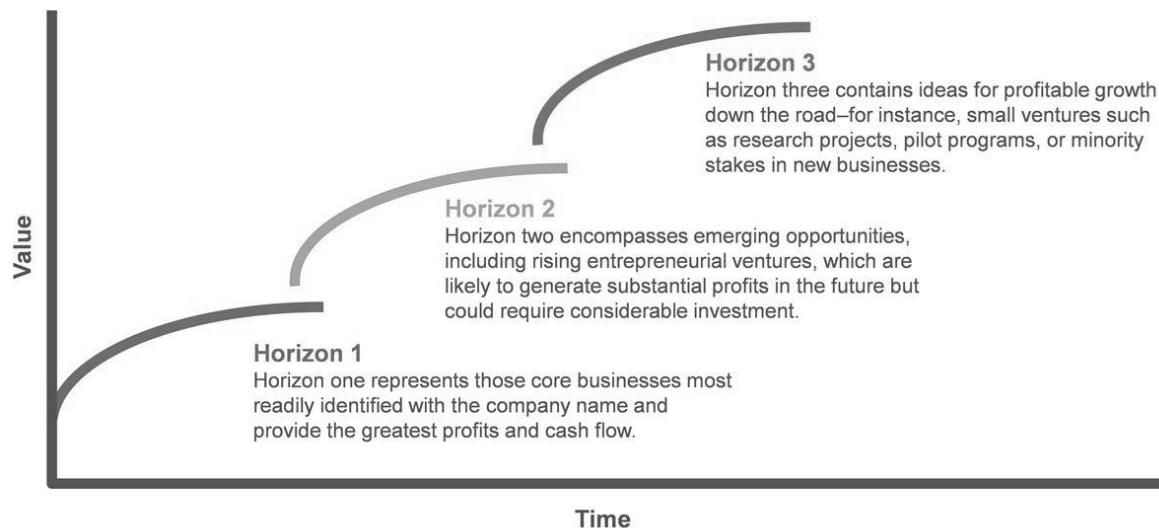
Portfolio Canvas

Portfolio Name: _____ Date: _____

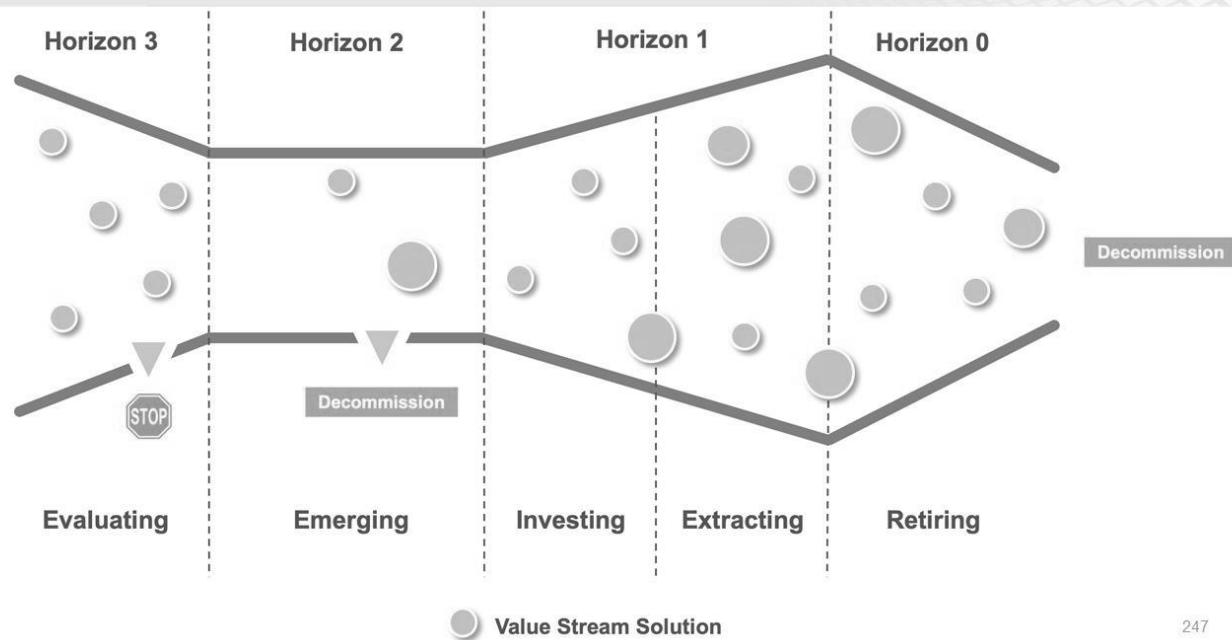
| Value Propositions | | | | | | |
|--------------------|-----------|-------------------|----------------|------------------------|-----------------|----------------|
| Value Streams | Solutions | Customer Segments | Channels | Customer Relationships | Budget | KPIs / Revenue |
| | | | | | | |
| Key Partners | | ∅ | Key Activities | | ☒ | Key Resources |
| | | | | | | |
| Cost Structure | | | | ∅ | Revenue Streams | |
| | | | | | | |

The Portfolio Canvas is adapted from The Business Model Canvas (<http://www.businessmodelgeneration.com>). This work is licensed under the Creative Commons Attribution-Share Alike 3.0 Unported License. To view a copy of this license visit: <http://creativecommons.org/licenses/by-sa/3.0/>.

Categorize investments by horizon (maturity stage)

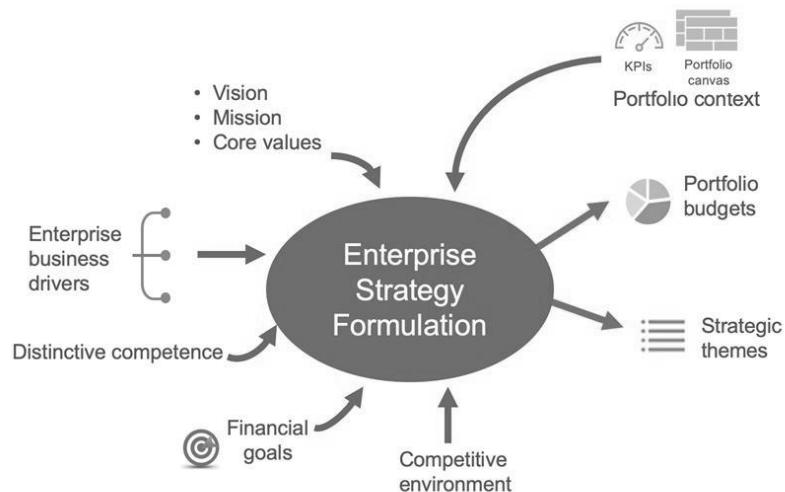


Map Solutions by horizon



5.2 Connect the portfolio to Enterprise strategy

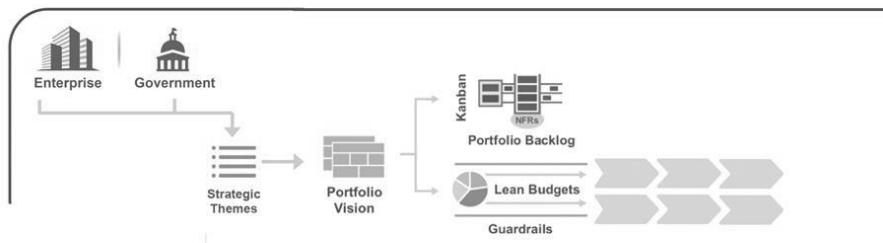
A model for Enterprise strategy formulation



Adapted from Jim Collins, *Beyond Entrepreneurship*

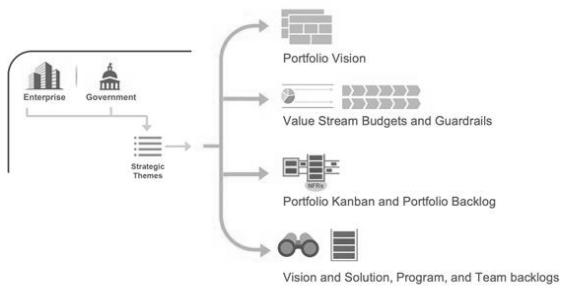
Establish Strategic Themes

- ▶ Differentiation from the current state to the desired future state
- ▶ A collaboration between LPM and the larger Enterprise
- ▶ Enterprise business drivers drive Strategic Themes
- ▶ Portfolio context influences Strategic Themes



Strategic Themes influence what gets built

- ▶ Strategic Themes are differentiating, specific, and itemized business objectives that connect a portfolio to the strategy of the Enterprise.
 - Provide context for decision-making, inputs to the Vision, budget, and backlog
 - Adjust ART and Value Stream funding to track changing strategic priorities
 - Assist with Epic evaluation and decision-making
 - Influence each Program Vision and Roadmap





Activity: Identify Strategic Themes

Prepare
5 min

Share
2 min

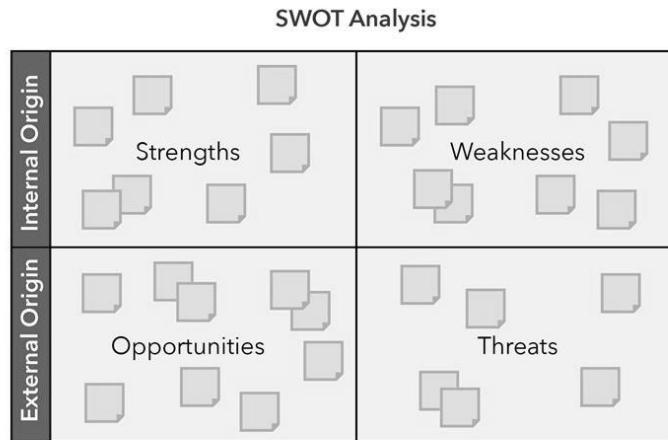
- ▶ **Step 1:** Identify three Strategic Themes that help define the strategy of your portfolio in the upcoming year
- ▶ **Step 2:** Discuss:
 - Are these *differentiators* for your business, as opposed to ‘business-as-usual’ items?
- ▶ **Step 3:** Be prepared to share with the class



5.3 Maintain the Portfolio Vision

Identify opportunities for the portfolio's future state with SWOT

- ▶ Establishes an understanding of your organization's strengths and weaknesses
- ▶ Identifies the most significant opportunities and potential threats



Identify options with 'TOWS'

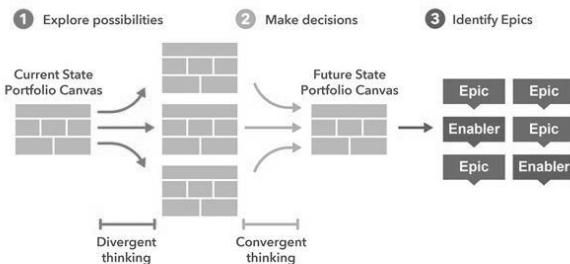
- ▶ SWOT and TOWS are both concerned with strengths, weaknesses, opportunities, and threats—the key difference is that TOWS focuses on action
- ▶ TOWS is used to identifying strategic options, Epics and other to create a better future state

TOWS Strategic Options Matrix

| External Opportunities (O) | | External Threats (T) | |
|----------------------------|--|---|---|
| Internal Strength (S) | | SO | ST |
| 1. | | How can your strengths be used to exploit and maximize opportunities? | How can you apply your strengths to overcome present and potential threats? |
| 2. | | | |
| 3. | | | |
| 4. | | | |
| Internal Weaknesses (W) | | WO | WT |
| 1. | | How can your opportunities be leveraged to overcome weaknesses? | How can you minimize weaknesses and avoid threats? |
| 2. | | | |
| 3. | | | |
| 4. | | | |

Envision the future state

- ▶ The portfolio canvas captured current state
- ▶ Use SWOT and TOWS to brainstorm potential future states
- ▶ Evaluate your options, and pick a future state
- ▶ Identify the Epics that will get you there



Express the future state as a Vision

A long view:

- ▶ How will our portfolio of future solutions solve the larger customer problems?
- ▶ How will these solutions differentiate us?
- ▶ What is the future context within which our solutions will operate?
- ▶ What is our current business context, and how must we evolve to meet this future state?



Vision: A postcard from the future



- ▶ Aspirational, yet realistic and achievable
- ▶ Motivational enough to engage others on the journey

Result: Everyone starts thinking about how to apply their strengths in order to get there.

Switch: *How to Change Things When Change is Hard*, Heath and Heath, Broadway Books, 2010

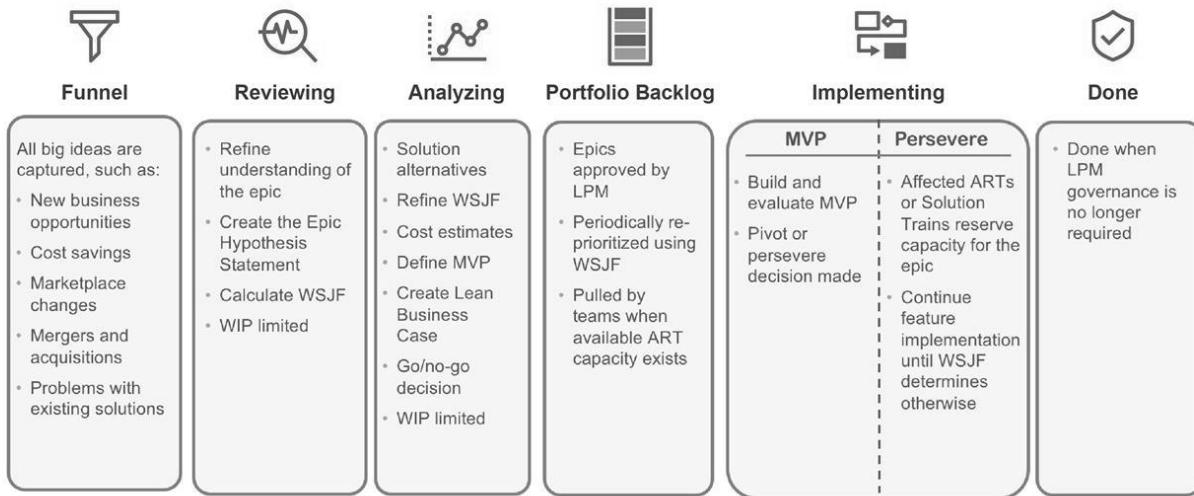
5.4 Establish portfolio flow

Govern Epic flow with the Portfolio Kanban

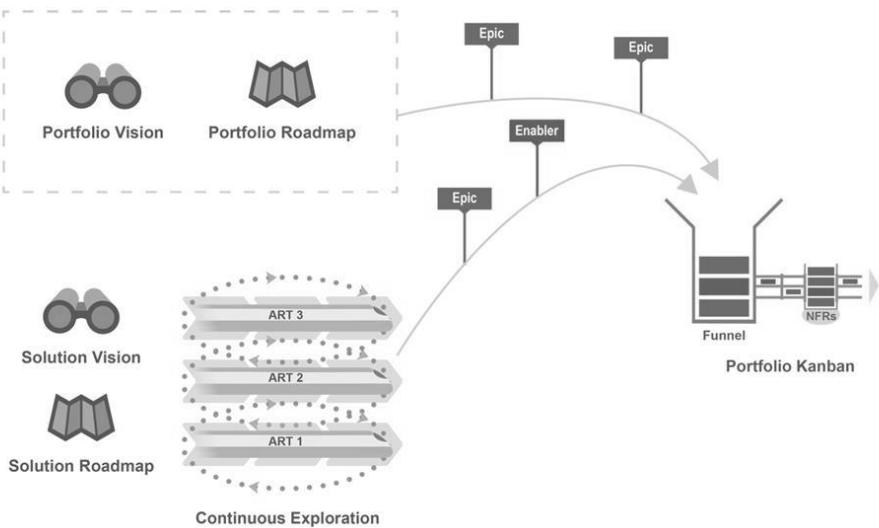
- ▶ Makes largest business initiatives visible
- ▶ Brings structure to analysis and decision-making
- ▶ Provides WIP limits to ensure the teams analyze responsibly
- ▶ Helps prevent unrealistic expectations
- ▶ Helps drive collaboration amongst the key stakeholders
- ▶ Provides a transparent and quantitative basis for economic decision-making



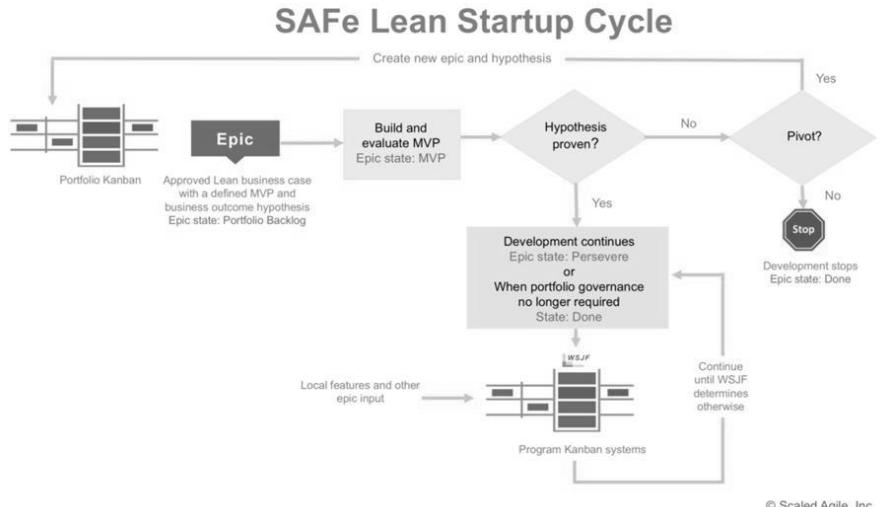
Portfolio Kanban



Feeding the portfolio funnel



MVPs foster innovation and control scope



© Scaled Agile, Inc.

Epic hypothesis statement template

| Epic Hypothesis Statement | |
|---|---|
| Funnel Entry Date: | <The date that the epic entered the funnel.> |
| Epic Name: | <A short name for the epic.> |
| Epic Owner: | <The name of the epic owner.> |
| Epic Description: | <An elevator pitch (value statement) that describes the epic in a clear and concise way.> For <customers> who <do something> the <solution> is a <something - the 'how'> that <provides this value> unlike <competitor, current solution or non-existing solution> our solution <does something better - the 'why'> |
| Business Outcomes: | <The measurable benefits that the business can anticipate if the epic hypothesis is proven to be correct.> |
| Leading Indicators: | <The early measures that will help predict the business outcome hypothesis. For more on this topic, see the Innovation Accounting advanced topic article.> |
| Nonfunctional Requirements (NFRs): | <Nonfunctional requirements (NFRs) associated with the epic.> |

© Scaled Agile, Inc.



Activity: Epic writing

Prepare
10 min

Share
3 min

- ▶ **Step 1:** In your group, identify an Epic from one of your contexts
- ▶ **Step 2:** Write the Epic hypothesis statement
- ▶ **Step 3:** Discuss:
 - What could be an MVP to validate this Epic?

| Epic Hypothesis Statement | |
|---|---|
| Funnel Entry Date: | <The date that the epic entered the funnel.> |
| Epic Name: | <A short name for the epic.> |
| Epic Owner: | <The name of the epic owner.> |
| Epic Description: | <An elevator pitch (value statement) that describes the epic in a clear and concise way.> For <customers> who <do something> the <solution> is a <something - the 'how'> that <provides this value> unlike <competitor, current solution or non-existing solution> our solution <does something better - the 'why'> |
| Business Outcomes: | |
| Leading Indicators: | <The measurable benefits that the business can anticipate if the epic hypothesis is proven to be correct.> |
| Nonfunctional Requirements (NFRs): | <Nonfunctional requirements (NFRs) associated with the epic.> |

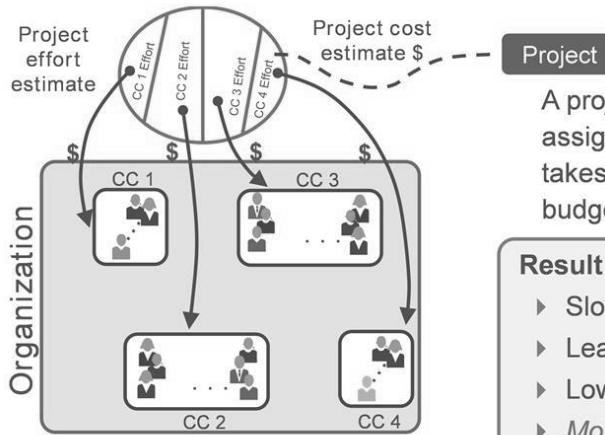
© Scaled Agile, Inc.

264

5.5 Fund Value Streams

Problem: Cost-center budgeting

Traditional project-based, cost-center budgeting creates overhead and friction, lowers velocity.



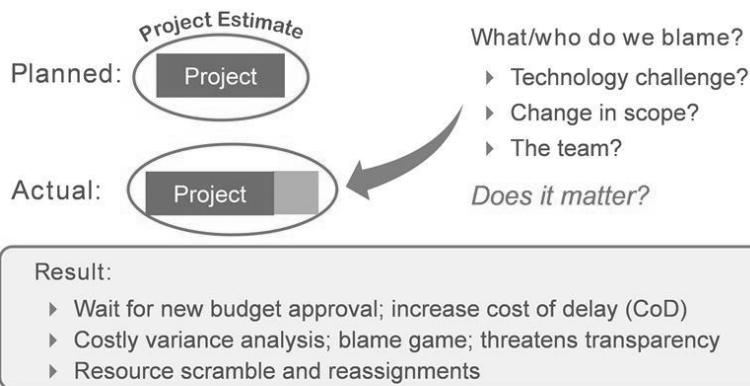
A project requires collaboration of cost centers, assignment of people, budget, and schedule. It takes multiple budgets to build a single project budget.

Result:

- ▶ Slow, complex budgeting process
- ▶ Leads to utilization-based planning and execution
- ▶ Low program throughput
- ▶ *Moves the people to the work*

Problem: Projects increase cost of delay

When overruns happen, project accounting and re-budgeting increases cost of delay and impacts culture.



Solution: Fund Value Streams not projects

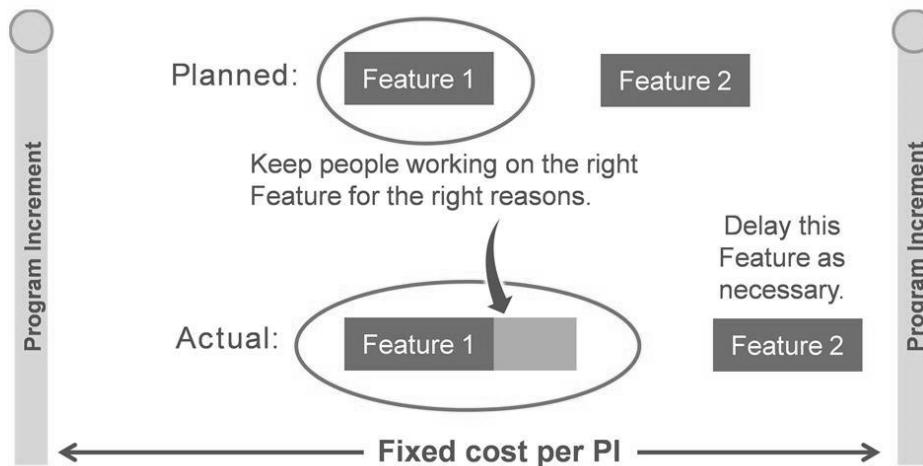
Funding Value Streams provides for full control of spend, with:

- ▶ No costly and delay-inducing project cost variance analyses
- ▶ No resource reassessments
- ▶ No blame game for project overruns



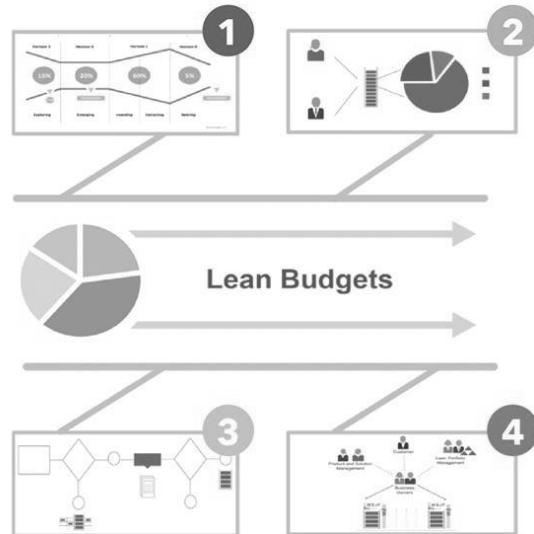
Control costs with increased flexibility

ART budgets and resources are unaffected by Feature cost overruns or changing priorities.



Maintain the Guardrails

- 1) Apply investment horizons
- 2) Utilize capacity allocation
- 3) Approve Epic initiatives
- 4) Continuous Business Owner engagement



Lesson review

In this lesson you:

- ▶ Defined a SAFe portfolio
- ▶ Connected the portfolio to Enterprise strategy
- ▶ Explored tools for maintaining the Portfolio Vision
- ▶ Discussed how to establish portfolio flow
- ▶ Discussed how to fund Value Streams

Leading SAFe®

Lesson 6: Leading the Change



SAFe® Course - Attending this course gives students access to the SAFe® Agilist exam and related preparation materials.

SCALED AGILE®

© Scaled Agile, Inc.

Learning objectives

At the end of this lesson you should be able to:

- ▶ 6.1 Lead by example
- ▶ 6.2 Lead the change

6.1 Lead by example

Leading by example

Setting an example is not the main means of influencing others, it is the only means. — Albert Einstein

- ▶ **Authenticity** requires leaders to model desired professional and ethical behaviors.
- ▶ **Emotional intelligence** describes how leaders identify and manage their emotions and those of others through self-awareness, self-regulation, motivation, empathy, and social skills
- ▶ **Lifelong learning** depicts how leaders engage in ongoing, voluntary, and self-motivated pursuit of knowledge and growth, and they encourage and support the same in others
- ▶ **Growing others** encourages leaders to provide the personal, professional, and technical guidance and resources each employee needs to assume increasing levels of responsibility
- ▶ **Decentralized decision-making** moves the authority for decisions to where the information is

Leading by Example



- ⌚ Authenticity
- ⌚ Emotional Intelligence
- ⌚ Lifelong Learning
- ⌚ Growing Others
- ⌚ Decentralized Decision-Making

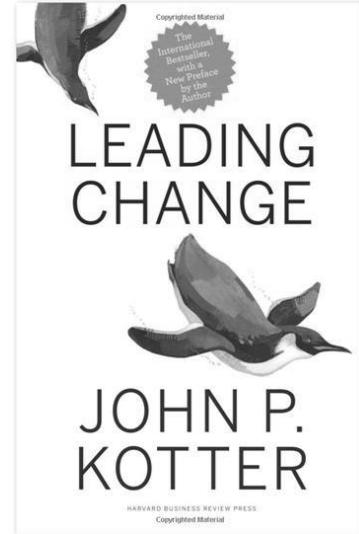
Leaders provide the organization with patterns of expected behaviors

| Pathological Culture <i>Power-oriented</i> | Bureaucratic Culture <i>Rule-oriented</i> | Generative Culture <i>Performance-oriented</i> |
|--|---|--|
| Low cooperation | Modest cooperation | High cooperation |
| Messengers blamed | Messengers neglected | Messengers trained |
| Responsibilities shirked | Narrow responsibilities | Responsibilities shared |
| Collaboration discouraged | Collaboration tolerated | Collaboration encouraged |
| Failure leads to scapegoating | Failure leads to justice | Failure leads to improvement |
| Innovation crushed | Innovation leads to problems | Innovation implemented |

6.2 Lead the change

Keys to leading successful change

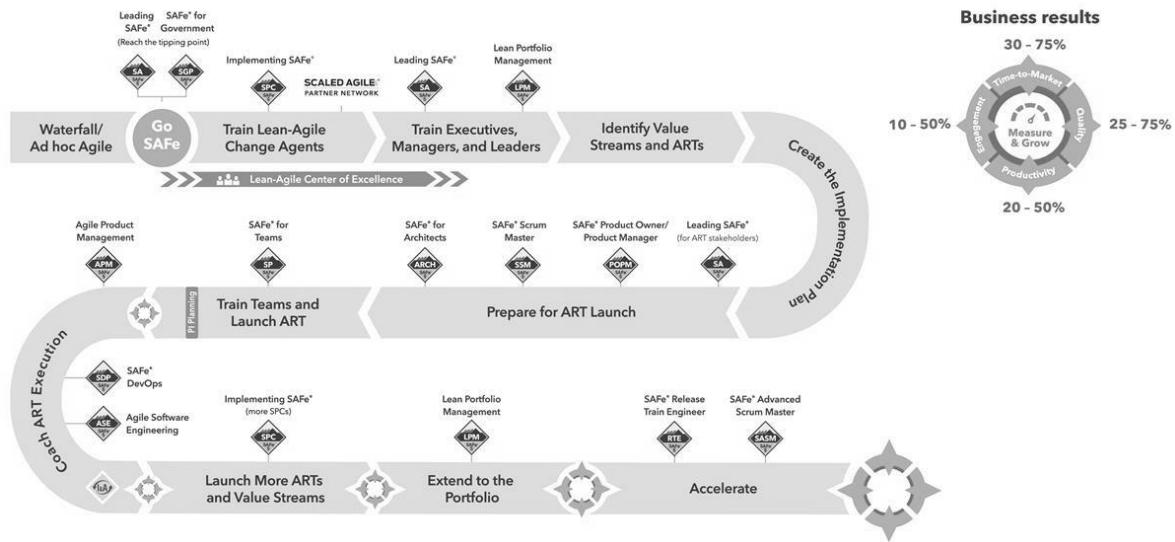
- ▶ Establish a sense of urgency
- ▶ Create a powerful guiding coalition
- ▶ Develop the vision and strategy
- ▶ Communicate the vision
- ▶ Empower employees for broad-based action
- ▶ Generate short-term wins
- ▶ Consolidate gains and produce more wins
- ▶ Anchor new approaches in the culture



SCALED AGILE® © Scaled Agile, Inc.

278

SAFe® Implementation Roadmap



SCALED AGILE® © Scaled Agile, Inc.

279



Action Plan: Leading the change

Prepare
5 min

Share
2 min

- ▶ **Step 1:** Identify three action items you can do in the next month to start leading the SAFe transformation.
- ▶ **Step 2:** Find a partner and share your ideas.
- ▶ **Step 3:** Discuss:
 - What outcomes do you hope to achieve with your Action Plan?



Lesson review

In this lesson you:

- ▶ Explored how to lead by example
- ▶ Identified actions to take for leading the change

Leading SAFe®

Lesson 7: Becoming a Certified SAFe Agilist



SAFe® Course - Attending this course gives students access to the SAFe® Agilist exam and related preparation materials.

SCALED AGILE®

© Scaled Agile, Inc.

Make the most of your learning



Access the SAFe Community Platform

Manage your member profile, access videos and training resources, join Communities of Practice, and more.



Prepare Yourself

Access your learning plan featuring your digital workbook, study materials, and certification practice test



Become a Certified SAFe Professional

Get certified to validate your knowledge, expand your professional capabilities, and open the door to new career opportunities.



Access SAFe Content and Tools

Access professional development resources and your trainer enablement plan to teach SAFe instructor-led courses.



Collaborate in real time with your team and others

Choose from ready-made templates to easily set up events like PI Planning and retrospectives—all with SAFe Collaborate.



Showcase SAFe Credentials

Display your digital badge to promote your SAFe capabilities and proficiencies throughout your career.



Video: Become a Certified SAFe Professional

Duration
3 min

Continue to build on the foundation of SAFe learning you began in class by studying and taking the certification exam.

Earning this certification demonstrates and establishes your new knowledge.

Certification details at:

<https://bit.ly/2zCu2pa>



SALE AGILE® © Scaled Agile, Inc.

284



Video: Welcome to the SAFe Community Platform

Duration
5 min

Want to learn more about the next steps on your SAFe Journey?

Access the SAFe Community Platform and discover all the SAFe resources available for your use!



SALE AGILE® © Scaled Agile, Inc.

285

