SAFe® for Teams

Establishing Team Agility for Agile Release Trains

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

5.1.1.1

SP SAFe® 5

SAFe® | PROVIDED BY SCALED AGILE®

Welcome to the course!

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 toolkits.

Collaborate with Your Team

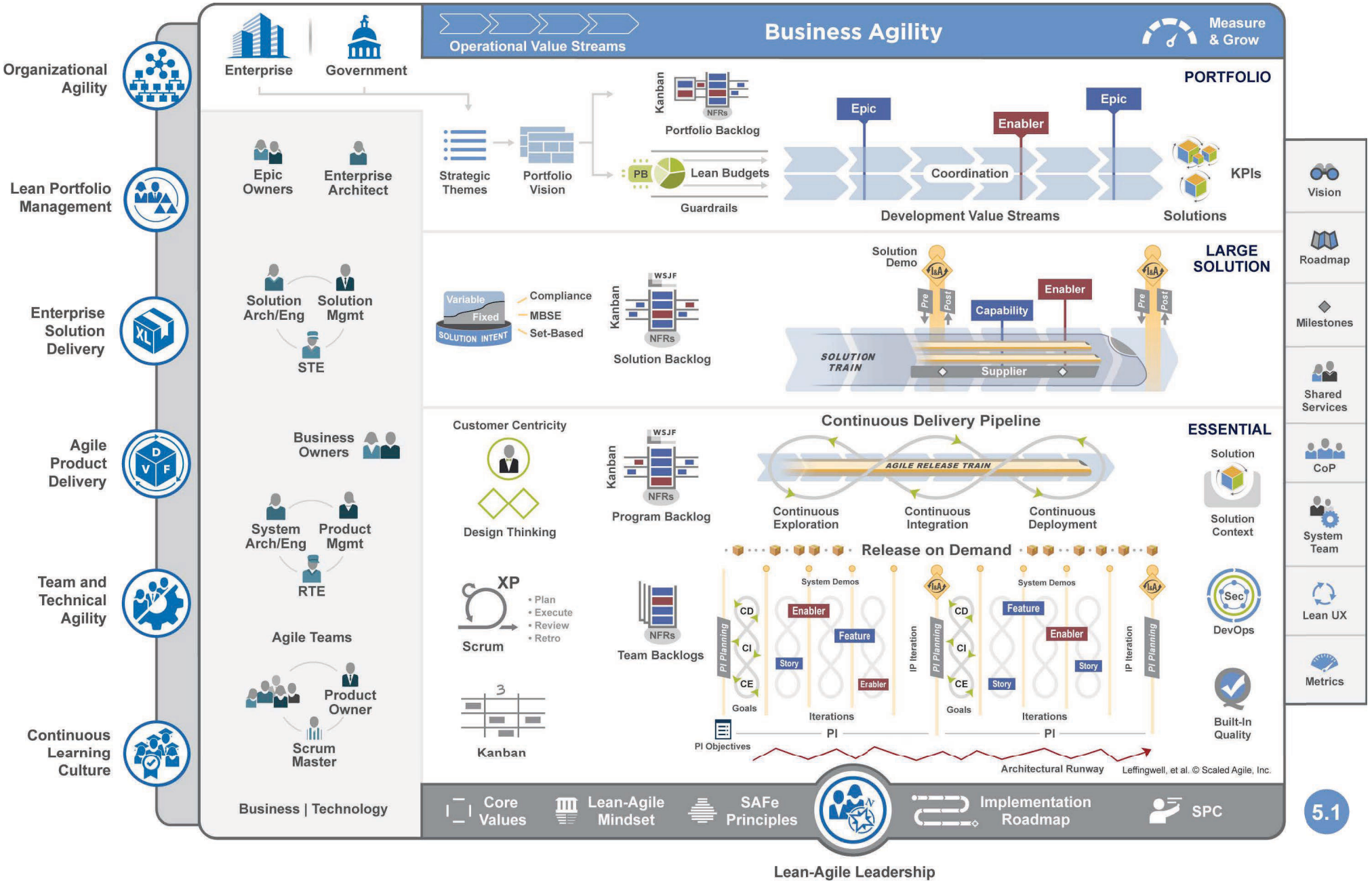
Choose from hundreds of collaboration templates to easily set up events like PI Planning and work in real time with your team and others–all with SAFe Collaborate.

Showcase SAFe Credentials

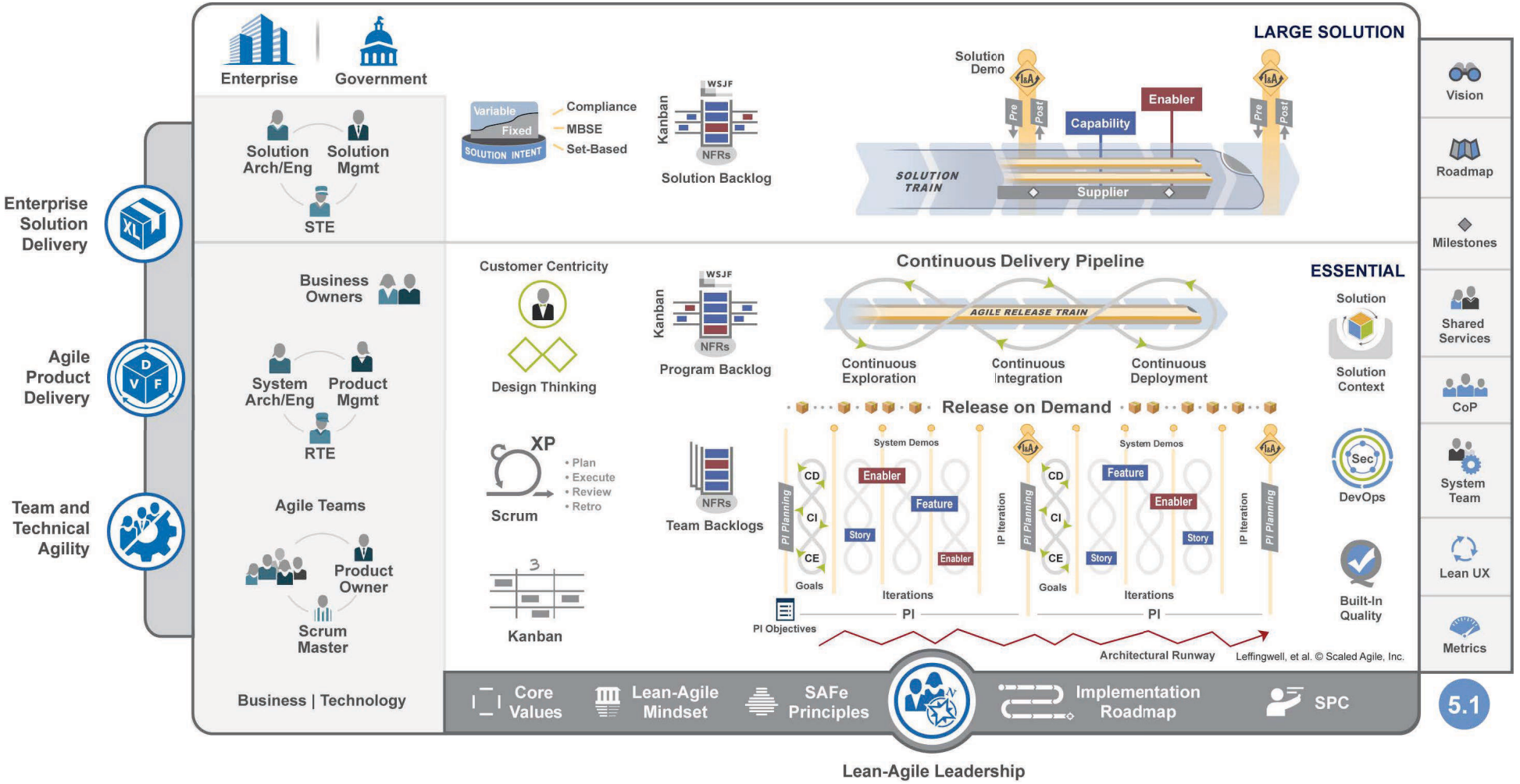
Display your digital badge to promote your SAFe capabilities and proficiencies throughout your career.

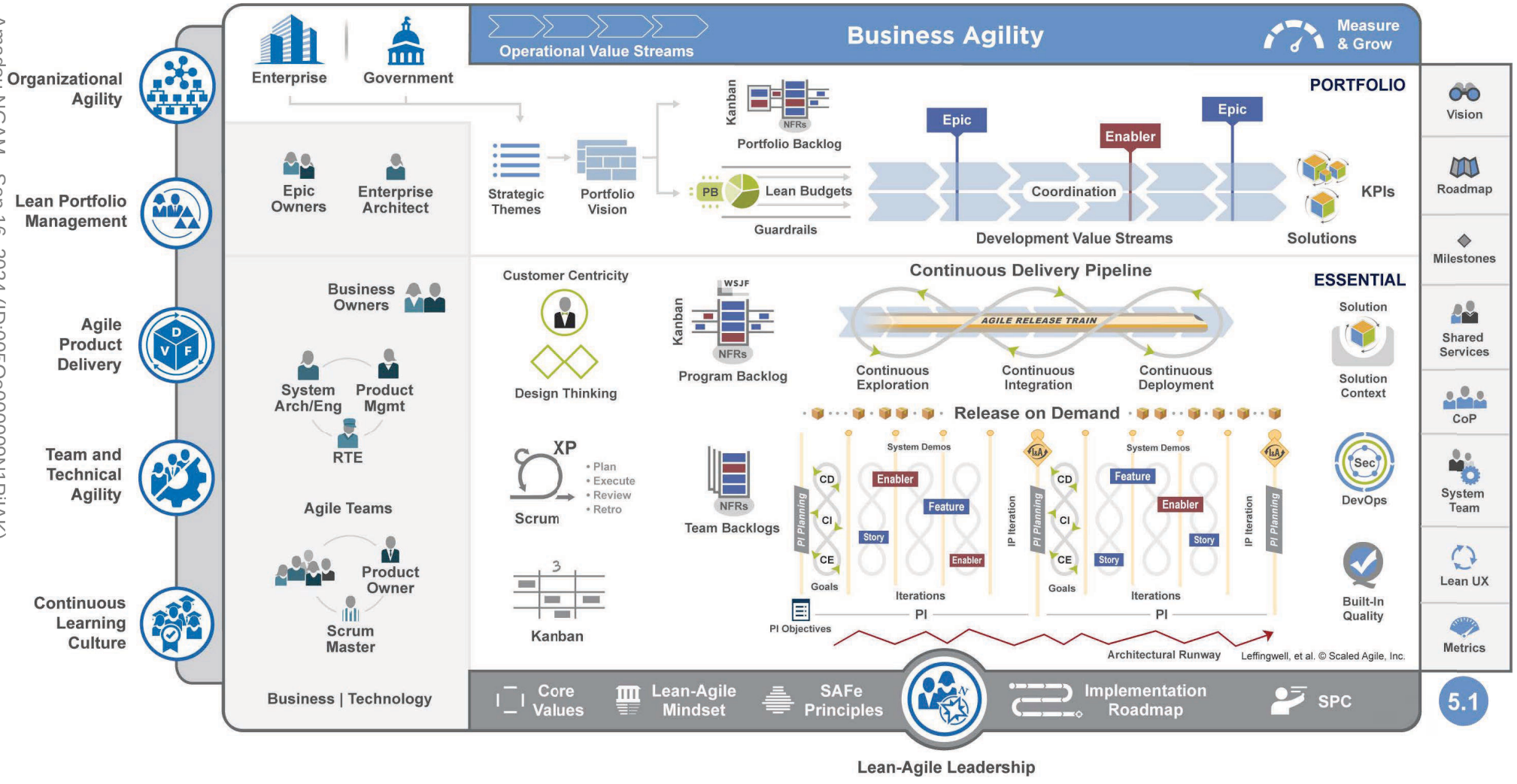
community.scaledagile.com

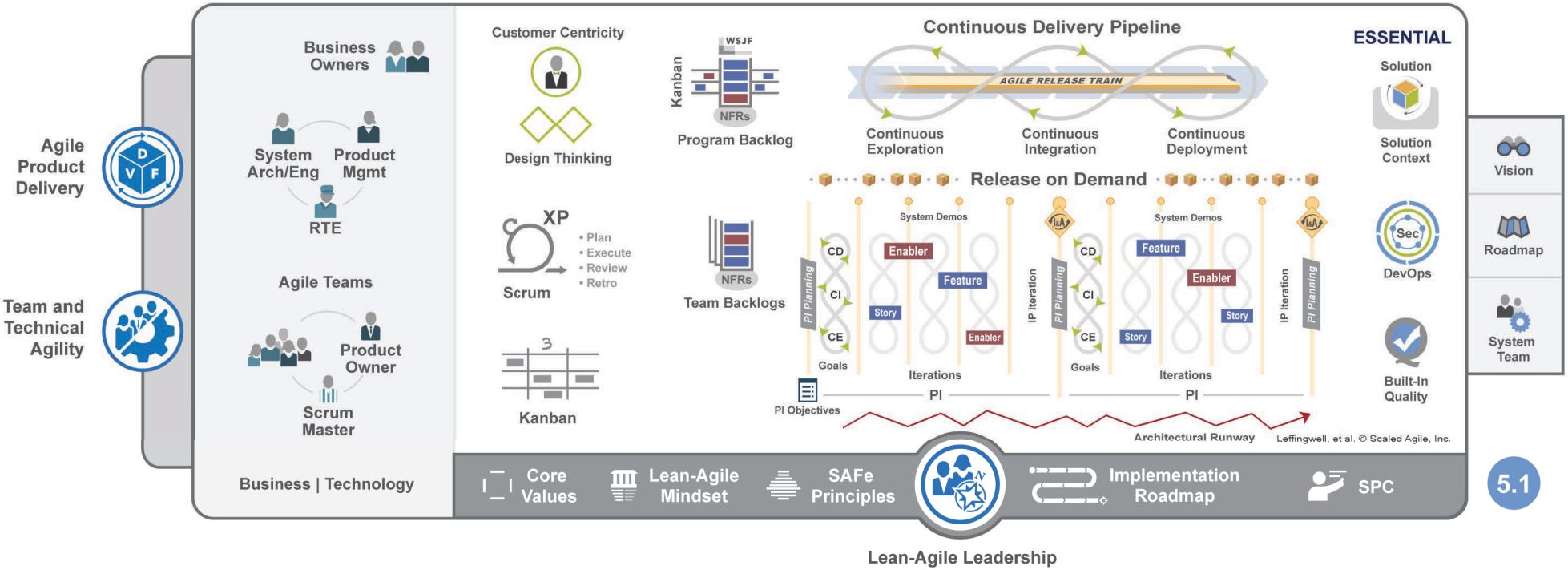
SAFe® PROVIDED BY SCALED AGILE®



SAFe® | PROVIDED BY SCALED AGILE®







BUSINESS AGILITY

MEASURE & GROW

Enterprise Solution Delivery

- Apply Lean system engineering to build really big systems
- Coordinate and align the full supply chain
- Continually evolve live systems

Lean System and Solution Engineering

Coordinating Trains and Suppliers

Continually Evolve Live Systems

Lean Portfolio Management

- Align strategy, funding, and execution
- Optimize operations across the portfolio
- Lightweight governance empowers decentralized decision-making

Agile Product Delivery

- The customer is the center of your product strategy
- Develop on cadence and release on demand
- Continuously explore, integrate, deploy, and innovate

Customer Centricity & Design Thinking

Develop on Cadence Release on Demand

DevOps and the Continuous Delivery Pipeline

Organizational Agility

- Create an enterprise-wide, Lean-Agile mindset
- Lean out business operations
- Respond quickly to opportunities and threats

Lean-thinking People and Agile Teams

Lean Business Operations

Strategy Agility

Team And Technical Agility

- High-performing, cross-functional, Agile teams
- Business and technical teams build business solutions
- Quality business solutions delight customers

Agile Teams

Teams of Agile Teams

Built-in Quality

Lean-Agile Leadership

- Inspire others by modeling desired behaviors
- Align mindset, words, and actions to Lean-Agile values and principles
- Actively lead the change and guide others to the new way of working

Leading by Example

Mindset & Principles

Leading Change

Continuous Learning Culture

- Everyone in the organization learns and grows together
- Exploration and creativity are part of the organization's DNA
- Continuously improving solutions, services, and processes is everyone's responsibility

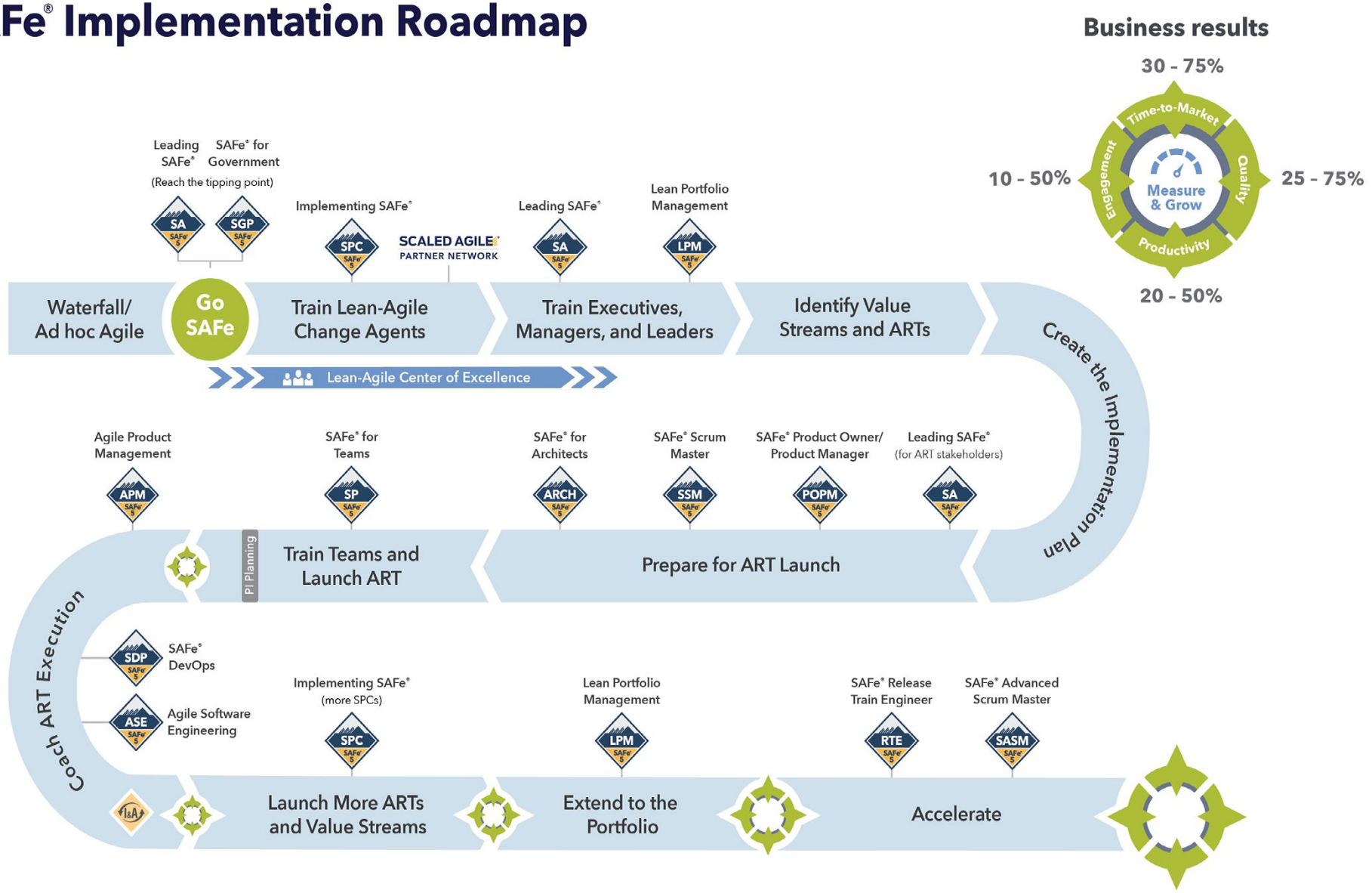
Learning Organization

Innovation Culture

Relentless Improvement

SAFe PROVIDED BY SCALED AGILE

SAFe® Implementation Roadmap



SCALED AGILE®

1/23/20 scaledagileframework.com/implementation-roadmap

SAFe® Courses and Certifications

Course	Description	Certification
Leading SAFe®	Thriving in the Digital Age with Business Agility	*with SAFe® 5 Agilist Certification*
Implementing SAFe®	Achieving Business Agility with the Scaled Agile Framework	*with SAFe® 5 Program Consultant Certification*
SAFe® for Government	Applying Lean-Agile Practices in the Public Sector with SAFe®	*with SAFe® 5 Government Practitioner Certification*
Lean Portfolio Management	Aligning Strategy with Execution	*with SAFe® 5 Lean Portfolio Manager Certification*
SAFe® Product Owner/Product Manager	Delivering Value through Effective Program Increment Execution	*with SAFe® 5 Product Owner/Product Manager Certification*
Agile Product Management	Using Design Thinking to Create Valuable Products in the Lean Enterprise	*with SAFe® 5 Agile Product Manager Certification*
SAFe® Scrum Master	Applying the Scrum Master Role within a SAFe® Enterprise	*with SAFe® 5 Scrum Master Certification*
SAFe® Advanced Scrum Master	Advancing Scrum Master Servant Leadership with SAFe®	*with SAFe® 5 Advanced Scrum Master Certification*
SAFe® Release Train Engineer	Facilitating Lean-Agile Program Execution	*with SAFe® 5 Release Train Engineer Certification*
SAFe® for Architects	Architecting for Continuous Value Flow with SAFe®	*with SAFe® 5 Architect Certification*
SAFe® DevOps	Optimizing Your Value Stream	*with SAFe® 5 DevOps Practitioner Certification*
SAFe® for Teams	Establishing Team Agility for Agile Release Trains	*with SAFe® 5 Practitioner Certification*
Agile Software Engineering	Enabling Technical Agility for the Lean Enterprise	*with SAFe® 5 Agile Software Engineer Certification*

Table of Contents

Privacy Notice

Your name, company, and email address will be shared with Scaled Agile, Inc. for course fulfillment, including testing and certification. Your information will be used in accordance with the Scaled Agile privacy policy available at *https://www.scaledagile.com/privacy-policy/*.

SAFe® for Teams

Establishing Team Agility for Agile Release Trains

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

5.1.1.1

SAFe® | PROVIDED BY SCALED AGILE®

Logistics

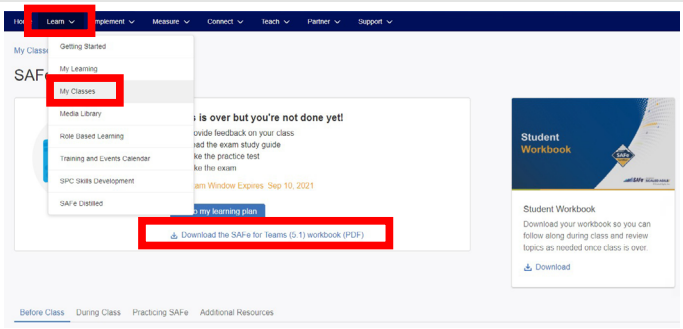
- Course meeting times
- Breaks
- Facilities
- Technology requirements
- Working agreements

1-2

Activity: Access the Class Page

Duration: 5 min

- **Step 1**: Navigate to the Class Page on the SAFe Community Platform
- **Step 2**: Select Learn, then My Classes, then SAFe for Teams (5.1)
- **Step 3**: Click on the link to Download the SAFe for Teams (5.1) workbook (PDF)



SAFe® | COMMUNITY

Visit the SAFe for Teams Class Page to download the workbook

https://bit.ly/CP-SAFeforTeams

1-3

Course outline

- Lesson 1: Introducing SAFe
- Lesson 2: Building an Agile Team
- Lesson 3: Planning the Iteration
- Lesson 4: Executing the Iteration
- Lesson 5: Executing the Program Increment (PI)
- Lesson 6: Practicing SAFe

1-4

Introducing the SAFe Practitioner Action Plan

In your workbook you will find the *Action Plan*

At the end of each lesson, you will have an opportunity to add ideas, insights, and improvement items as a takeaway from each of the lessons

1-5

Lesson 1

Introducing SAFe®

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

SP SAFe 5

SCALED AGILE®

Lesson Topics

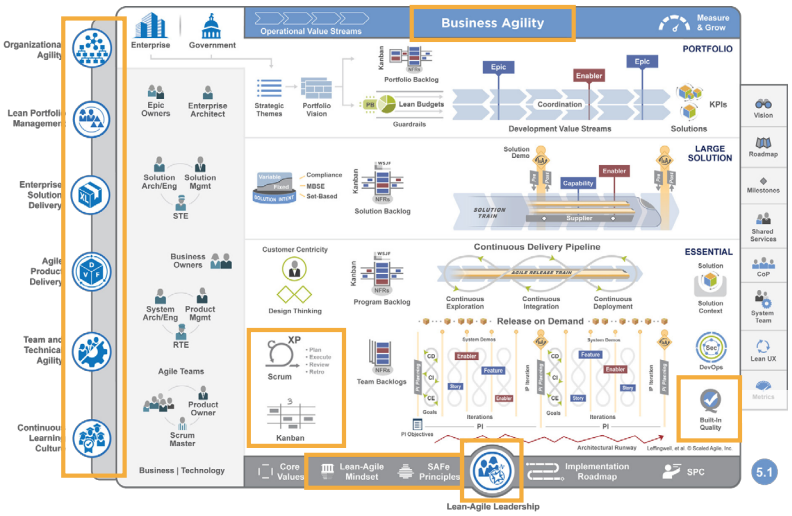
1.1 The Scaled Agile Framework

1.2 The Seven Core Competencies of Business Agility

1.3 The Lean-Agile Mindset

1.4 Lean and Agile at scale with the SAFe principles

1.5 Scrum, Kanban and Quality practices



1-7

Learning objectives

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

- Describe what is necessary to thrive in the digital age
- Recognize SAFe as an operating system for Business Agility
- Summarize the seven core competencies of Business Agility
- Embrace the Lean-Agile Mindset
- Identify the SAFe Lean-Agile Principles
- Describe Scrum, Kanban, and XP practices

SCALED AGILE © Scaled Agile, Inc.

1-8

1.1 The Scaled Agile Framework

1-9

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

1-10

A new economic landscape

- "BMW Group's CEO expects that in their future more than half of its research and development staff will be software developers." (Mik Kersten, *Project to Product*)
- The market cap of Tesla ($464B market cap, $24B revenue) now exceeds the market cap of Ford ($33B market cap, $156B revenue) at a 14:1 value ratio (November 2020)
- Apple is now the biggest watchmaker in the world (Investopedia 2019)

SCALED AGILE

Rethinking the organization

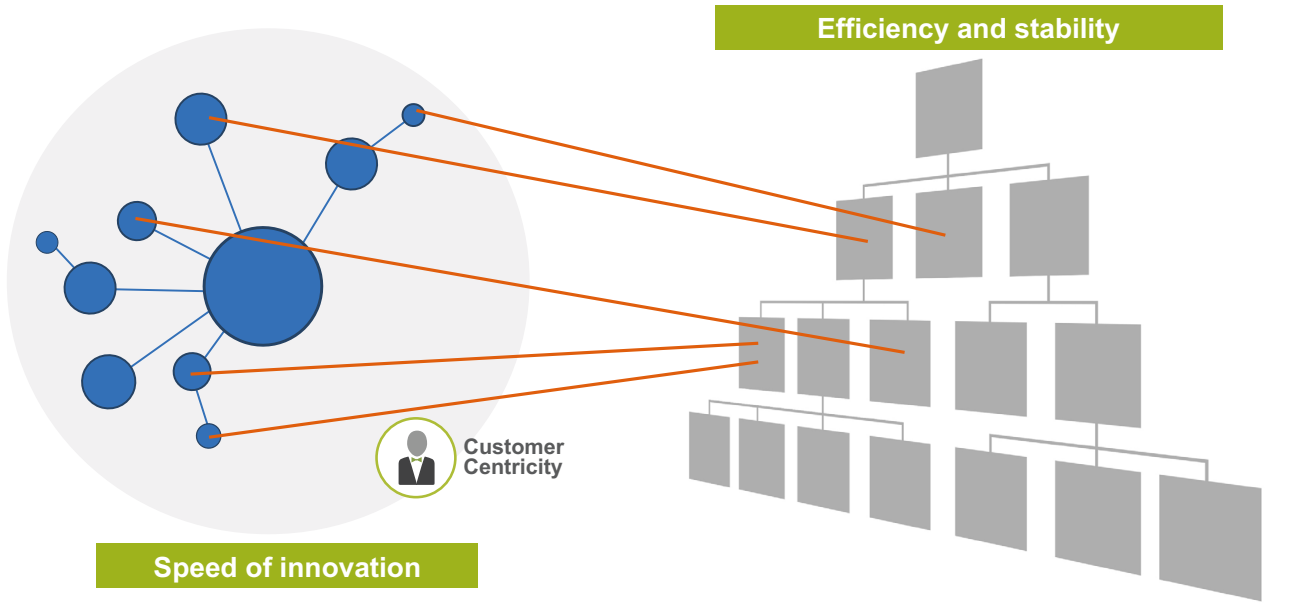
"The world is now changing at a rate at which the basic systems, structures, and cultures built over the past century cannot keep up with the demands being placed on them.

—John P. Kotter

> The solution is not to trash what we know and start over but instead to reintroduce, in an organic way, a second system—one which would be familiar to most successful entrepreneurs.
>
> —John P. Kotter

1-13

We need a dual operating system for Business Agility

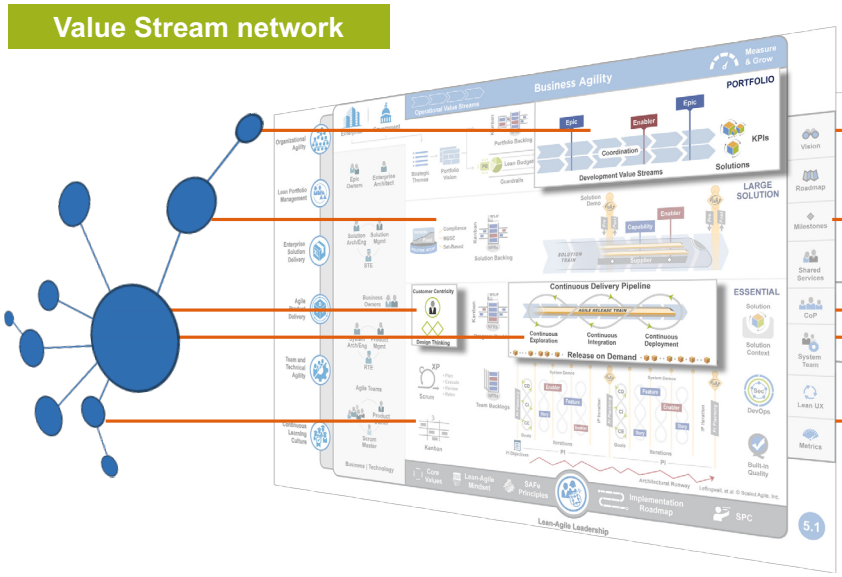


SCALED AGILE © Scaled Agile. Inc.

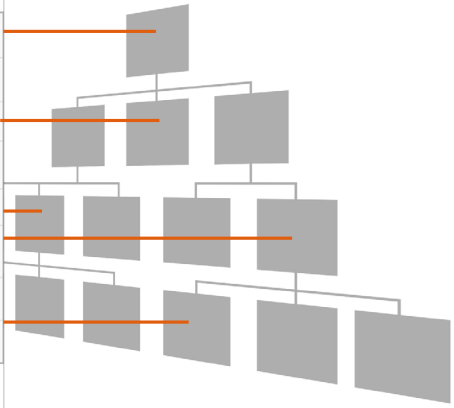
1-14

And we have just such an operating system at our fingertips

Value Stream network



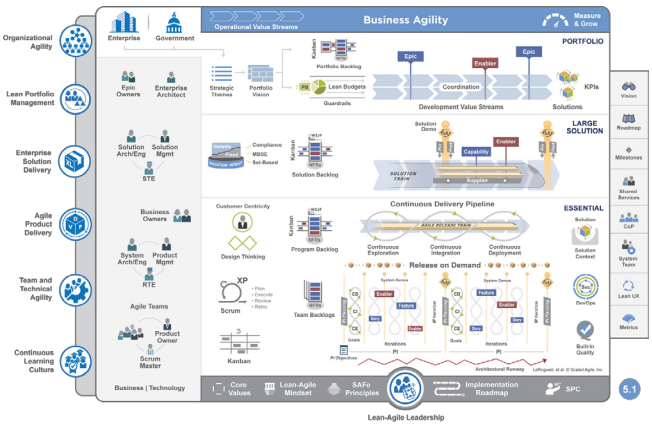
Functional hierarchy



SCALED AGILE © Scaled Agile, Inc.

1-15

SAFe® 5 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.



1-16

Why SAFe?

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

Typical results from https://bit.ly/SAI-CustomerStories

1-17

SAFe: Roots, past, present, and future

2011 — Field experience at Enterprise scale — Now…

Lean product development | Agile development | DevOps | Systems thinking

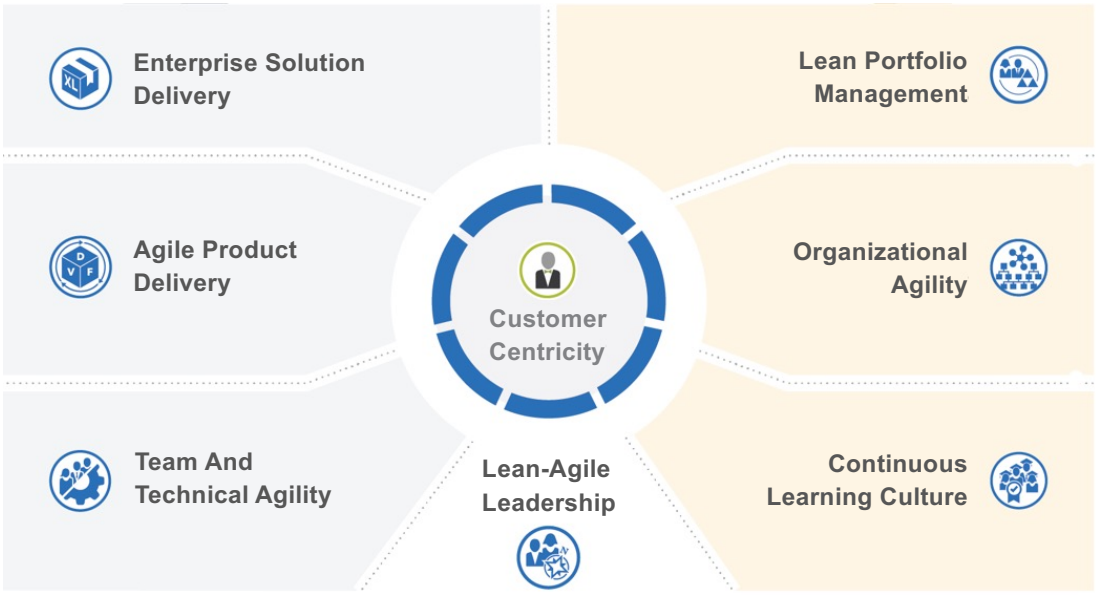
SCALED AGILE © Scaled Agile, Inc.

1-18

1.2 The Seven Core Competencies of Business Agility

1-19

The Seven Core Competencies of Business Agility



1-20

Team and Technical Agility

- High-performing, cross-functional Agile teams
- Teams of business and technical teams build Solutions
- Quality business Solutions delight Customers

Agile Teams

Teams of Agile Teams

Built-In Quality

Agile Product Delivery

- The Customer is the center of your product strategy
- Decouple the release of value from the development cadence
- Continuously explore, integrate, deploy, and release

Customer Centricity and Design Thinking

Develop on cadence and release on demand

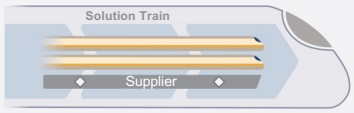
DevOps and the Continuous Delivery Pipeline

Enterprise Solution Delivery

- Apply Lean system engineering practices to build really big systems
- Coordinate and align the full supply chain
- Continue to enhance value after release

Lean System and Solution Engineering

Coordinate Trains and Suppliers



Continuously Evolve Live Systems

Lean Portfolio Management

- Align strategy, funding, and execution
- Optimize operations across the portfolio
- Lightweight governance empowers decentralized decision-making

Organizational Agility

- Create an enterprise-wide, Lean-Agile mindset
- Map and continuously improve business processes
- Respond quickly to opportunities and threats

Lean-thinking People and Agile Teams

Lean Business Operations

Strategy Agility

Continuous Learning Culture

- Everyone in the organization learns and grows together
- Exploration and creativity are part of the organization's DNA
- Continuously improving solutions, services, and processes is everyone's responsibility

Learning Organization

Innovation Culture

Relentless Improvement

Lean-Agile Leadership

- Inspire others by modeling desired behaviors
- Align mindset, words, and actions to Lean-Agile values and principles
- Actively lead the change and guide others to the new way of working

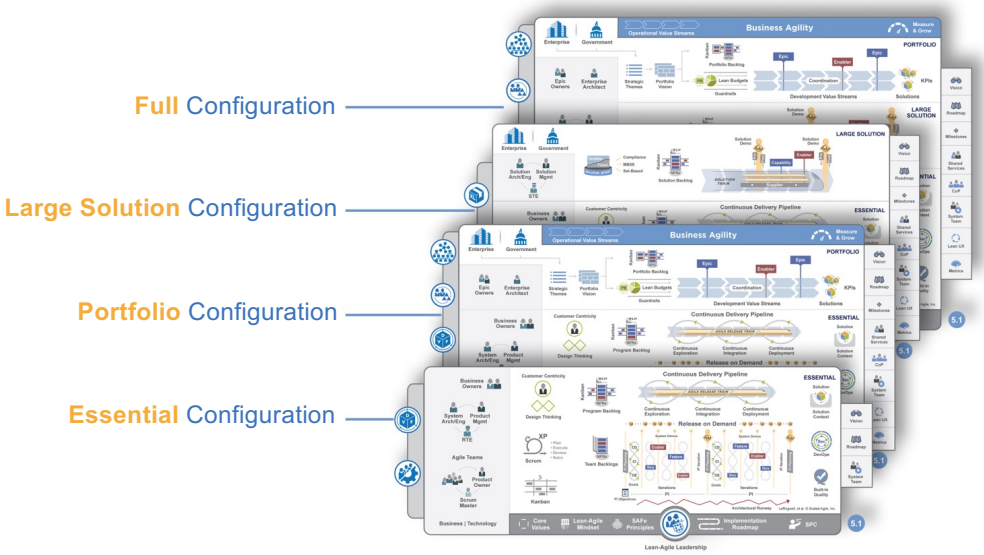
Leading by Example

Mindset & Principles

Leading Change

SAFe configurations

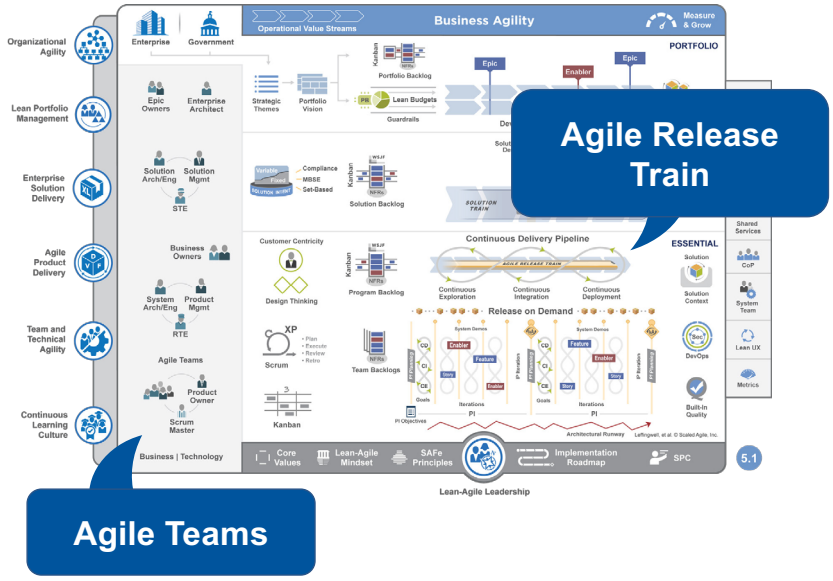
Four configurations provide the right solution for each Enterprise.



Positioning an Agile Team in a SAFe Enterprise

Essential SAFe contains:

Roles and activities, and events and processes which Agile Teams use to build and deliver value in the context of the ART

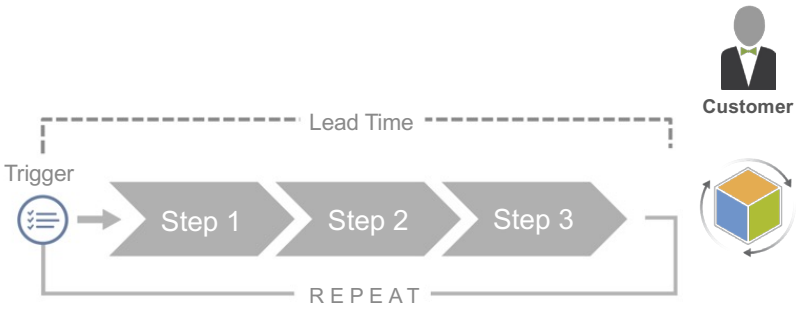


1.3 The Lean-Agile Mindset

Lean thinking

- ✓ Precisely specify value by product
- ✓ Identify the Value Stream for each product
- ✓ Make value flow without interruptions
- ✓ Let the Customer pull value from the producer
- ✓ Pursue perfection

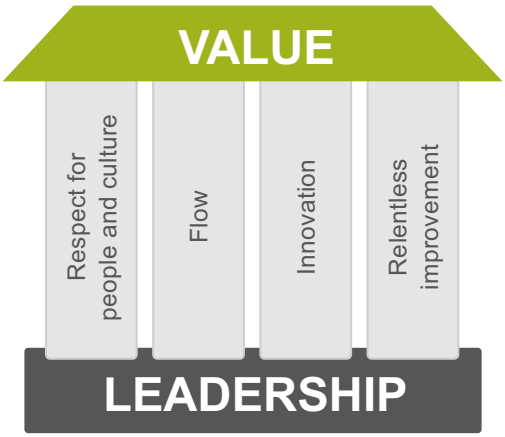
Lean Thinking: Banish Waste and Create Wealth in your Corporation, James Womack and Daniel Jones



1-31

SAFe House of Lean

The Lean thinking mindset is embodied in the SAFe House of Lean



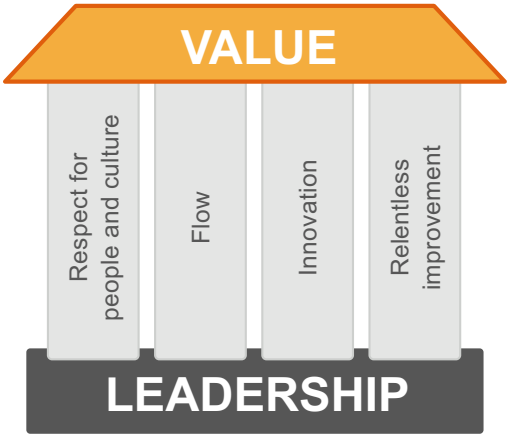
SCALED AGILE © Scaled Agile, Inc.

1-32

Value

Achieve the shortest sustainable lead time with:

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

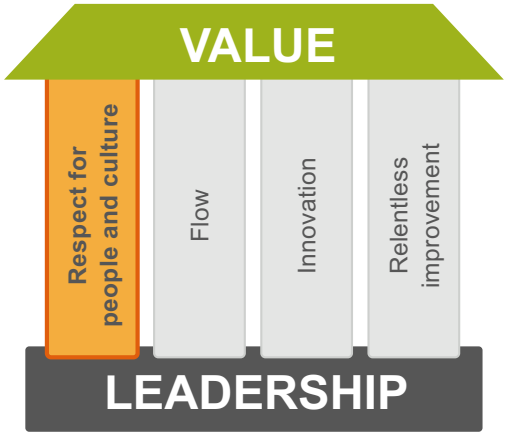


There is only one boss. The customer. And he can fire everybody in the company.

—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

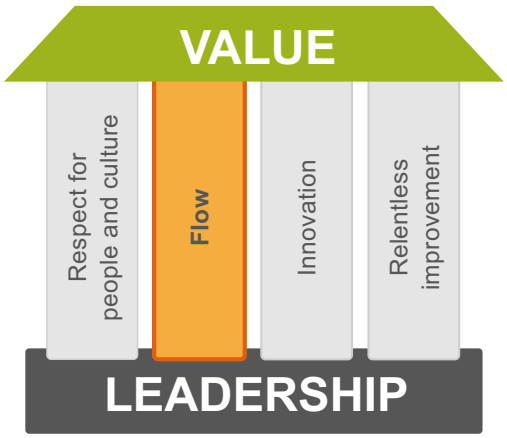


Culture eats strategy for breakfast.

—Peter Drucker

Flow

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

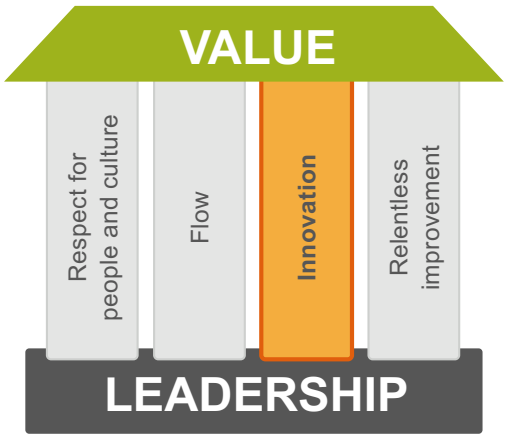


Operating a product development process near full utilization is an economic disaster.

—Don Reinertsen

Innovation

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

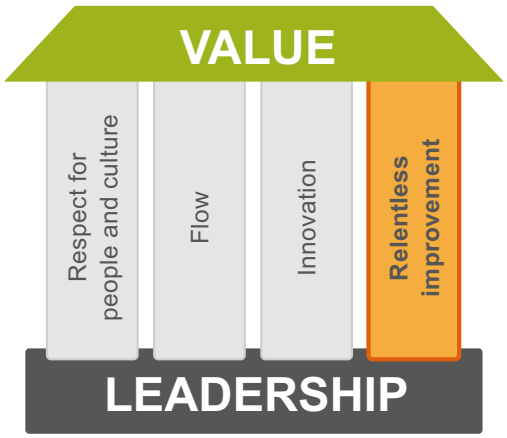


Innovation comes from the producer.

—W. Edwards Deming

Relentless improvement

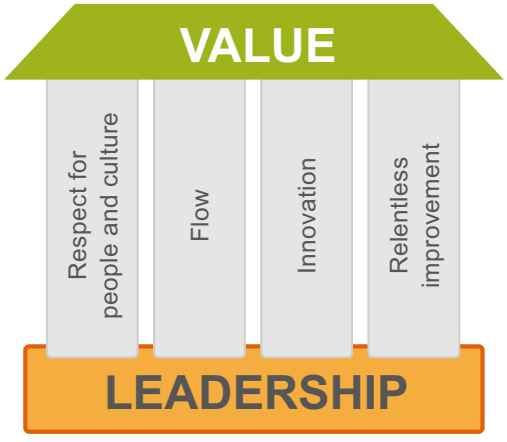
- A constant sense of danger
- Optimize the whole
- Problem-solving culture
- Base improvements on facts
- Reflect at key Milestones



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



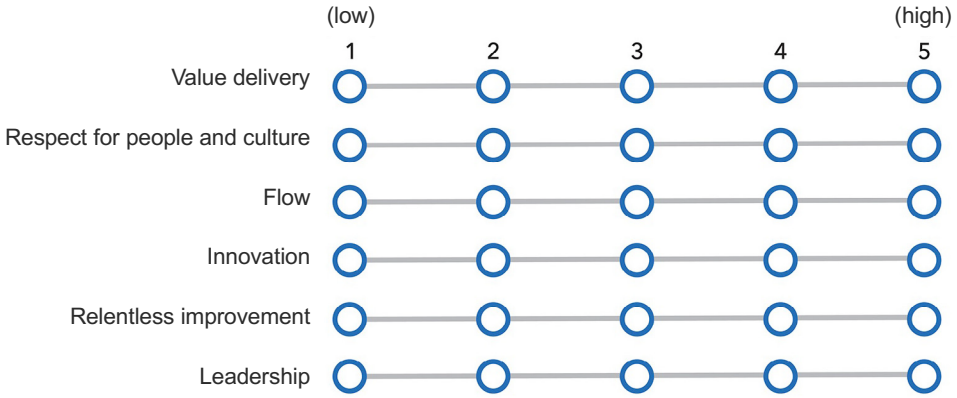
People are already doing their best; the problems are with the system. Only management can change the system.

—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?

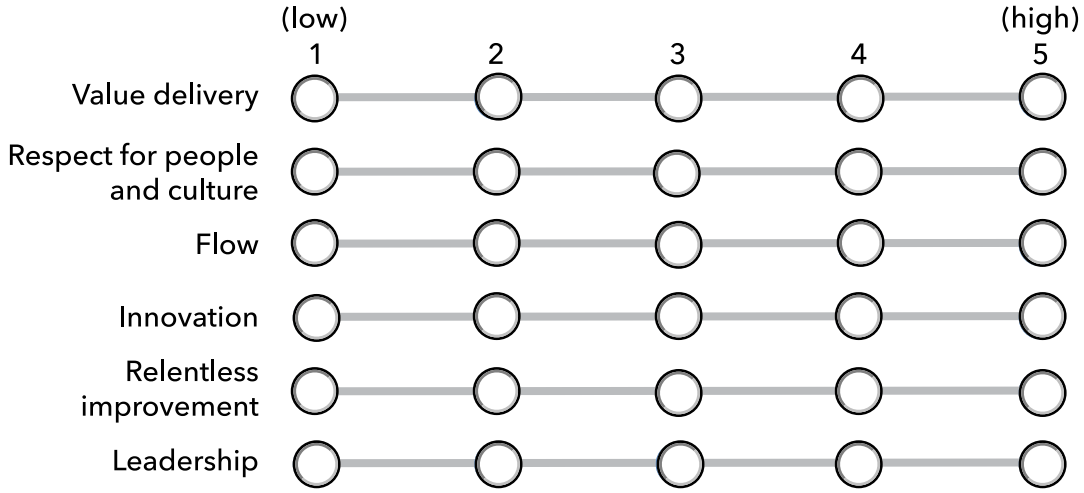


1-39

Lean Mindset Self-Assessment

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?



Notes

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.

https://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.

The Agile Manifesto Principles

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.
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.

The Agile Manifesto Principles

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 Manifesto principles

Prepare 5 min

Share 3 min

- **Step 1:** Review the principles behind the Agile Manifesto.
- **Step 2:** As a group, select one or more principles to discuss.
- **Step 3:** In your group, discuss how these principles apply in your context.
- **Step 4:** Provide an example of how you would apply one of the principles to your context and share with class.

1.4 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

#7Apply cadence, synchronize with cross-domain planning

#8 Unlock the intrinsic motivation of knowledge workers

#9 Decentralize decision-making

#10 Organize around value

1-46

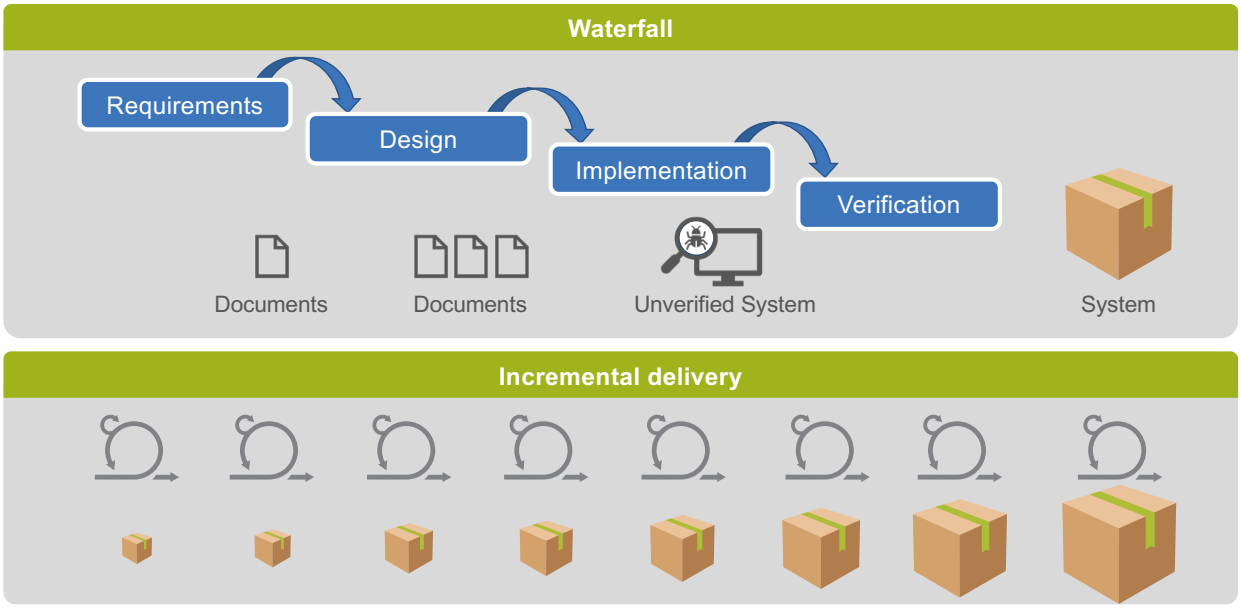
Video: Building Incrementally: Economic Advantage

Duration 3 min

https://bit.ly/Video-BuildingIncrementally

1-47

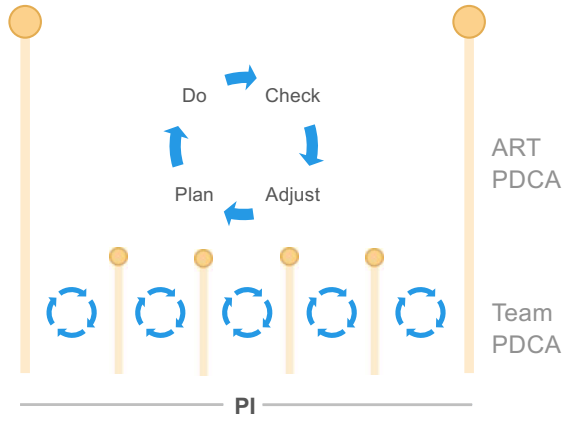
Agile economics: Deliver early and often

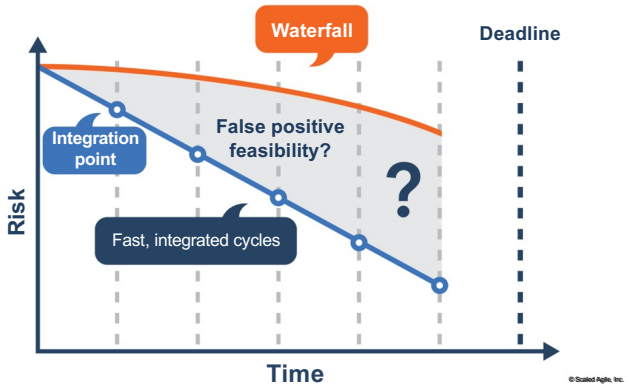


SCALED AGILE © Scaled Agile, Inc.

1-48

Use Iterations and Program Increments to learn quickly



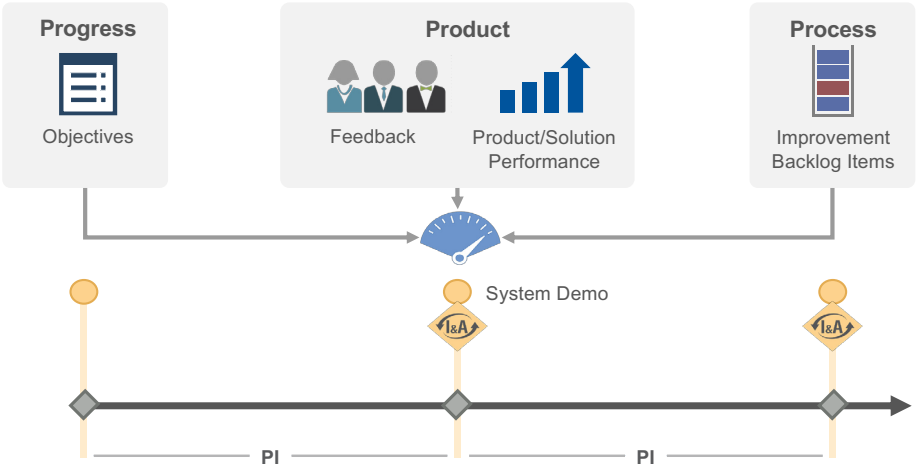


SCALED AGILE © Scaled Agile, Inc.

1-49

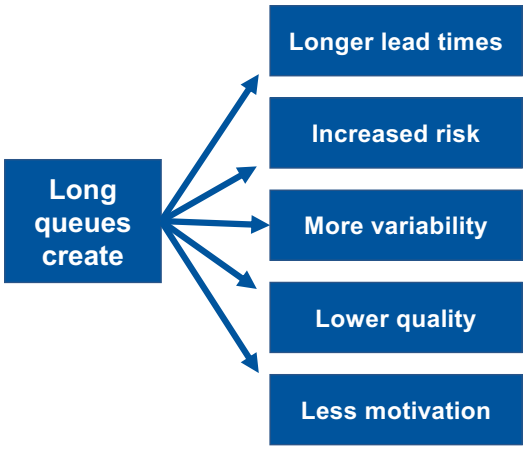
Base milestones on objective evaluation of working systems

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



Visualize and limit WIP, reduce batch size, and manage queue lengths

Long queues: All bad



Principles of Product Development Flow, Don Reinertsen

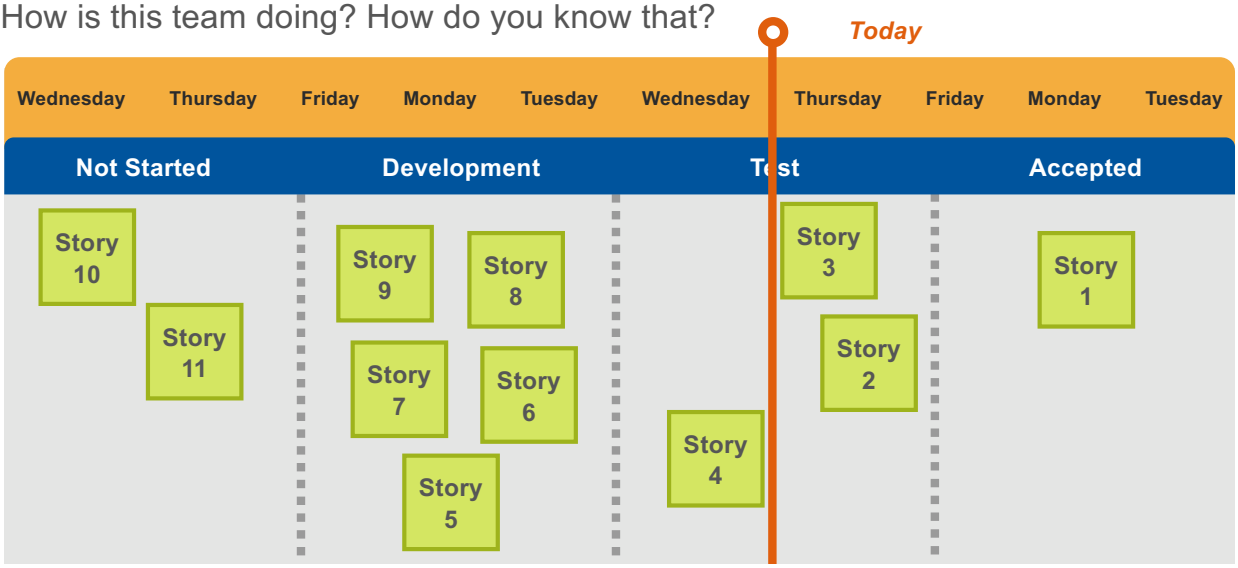
Little's Law

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

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

Visualize and limit work in progress

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



1-52

Activity: Experience a large batch size

Duration 5 min

- **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 process each coin.
- **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

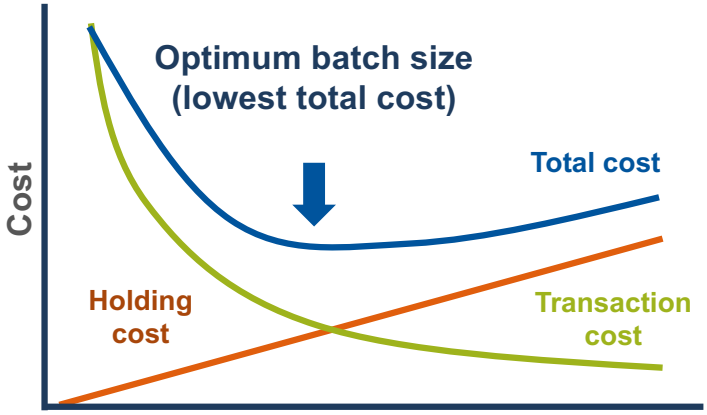
1-53

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 processes one coin at a time 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

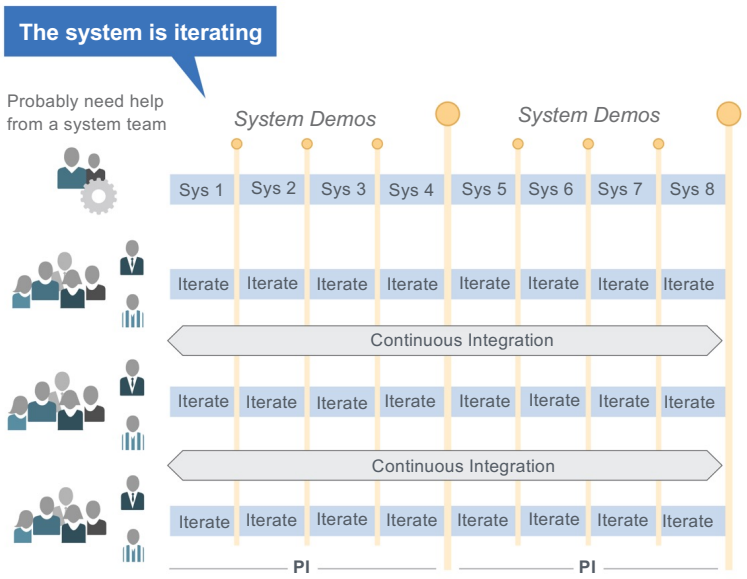
Reduce batch size for higher predictability



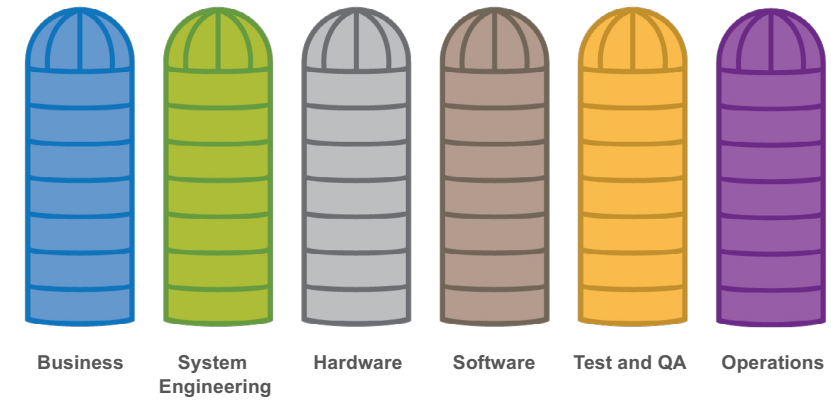
Principles of Product Development Flow, Don Reinertsen

Apply cadence, synchronize with cross-domain planning

Cadence-based planning limits variability.



Value doesn't follow silos

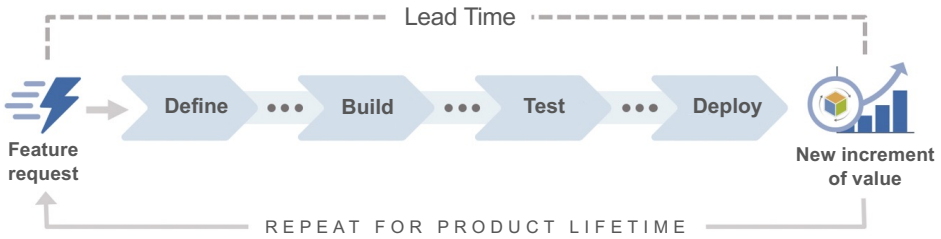


- Value delivery is inhibited by handoffs and delays
- Political boundaries can prevent cooperation
- Silos encourage geographic distribution of functions
- Communication across silos is difficult

Instead, organize around Development Value Streams

The aim of development is in fact the creation of profitable operational value streams.
—Allen C. Ward

- Includes activities from recognizing an opportunity through release and validation
- Contains the steps, the flow of information and material, and the people who develop the Solutions used by the Operational Value Streams



1.5 Scrum, Kanban, and Quality Practices

From traditional development to Agile

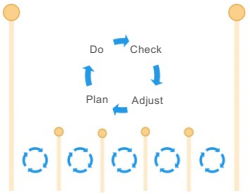
Instead of a large group…

…working on all of the requirements…

…and integrating and delivering value toward the end of development,

Have small teams working together as a program…

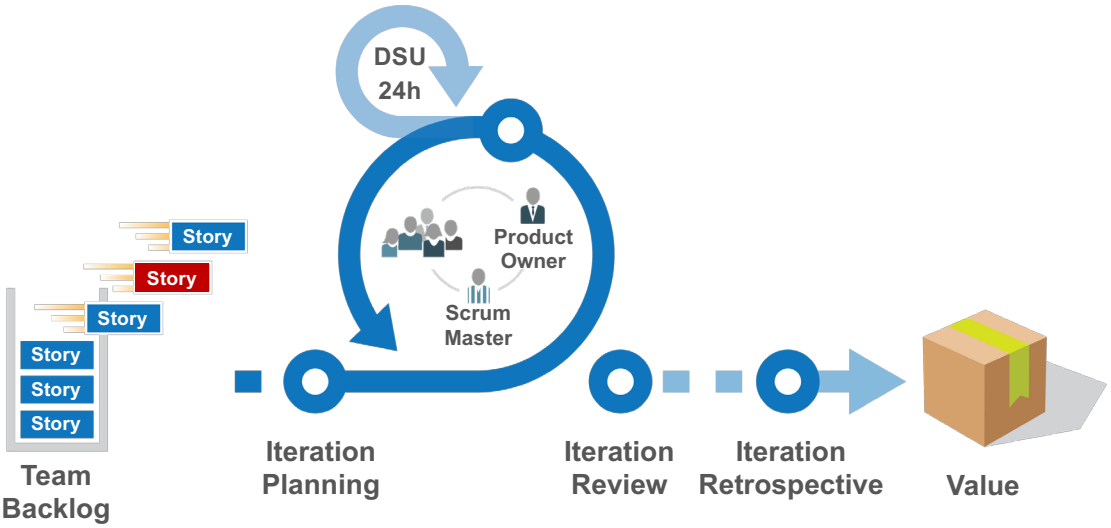
…working on small batches of requirements…



…and delivering value in short timeboxes with frequent integration and improvement cycles.

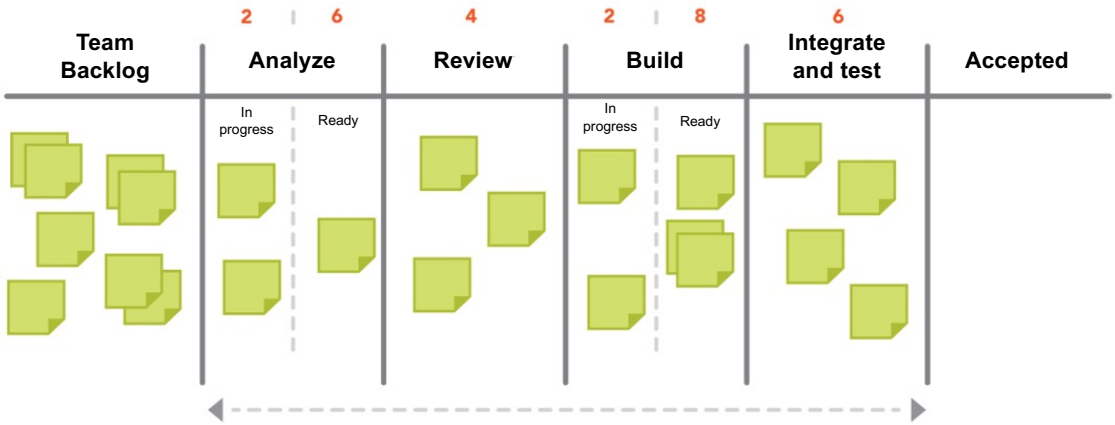
Agile for teams: Scrum

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



Agile for teams: Kanban

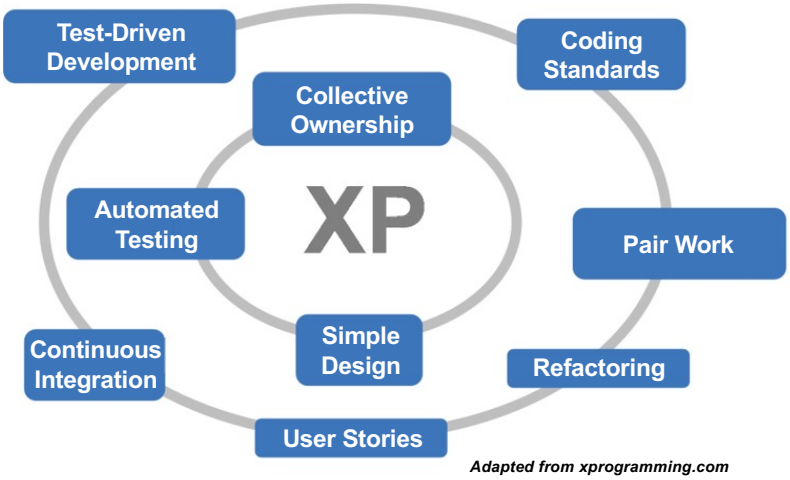
Visualize work flow. Limit work in process. Improve flow.



1-62

Quality practices provides the basis for Technical Agility

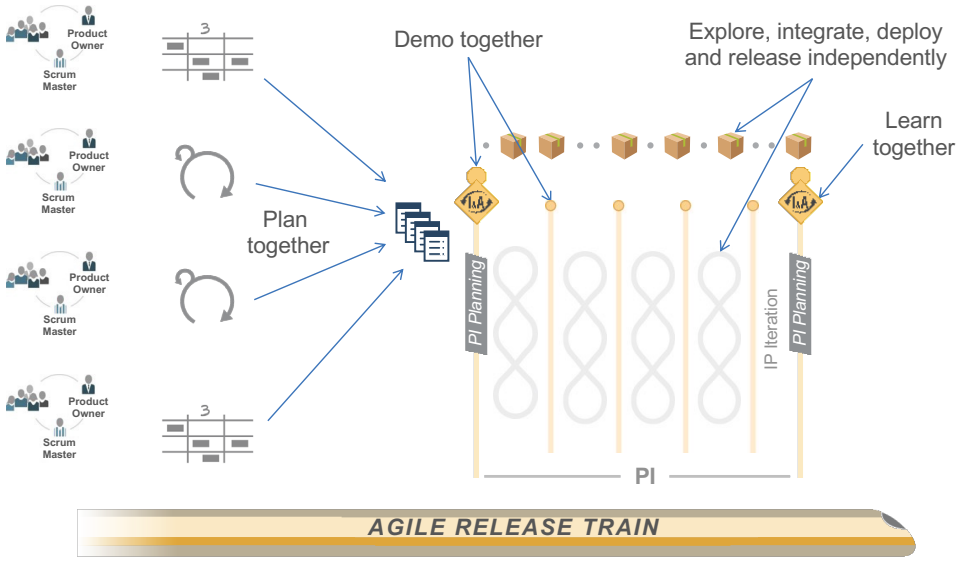
Examples of quality practices inspired by eXtreme Programming (XP)



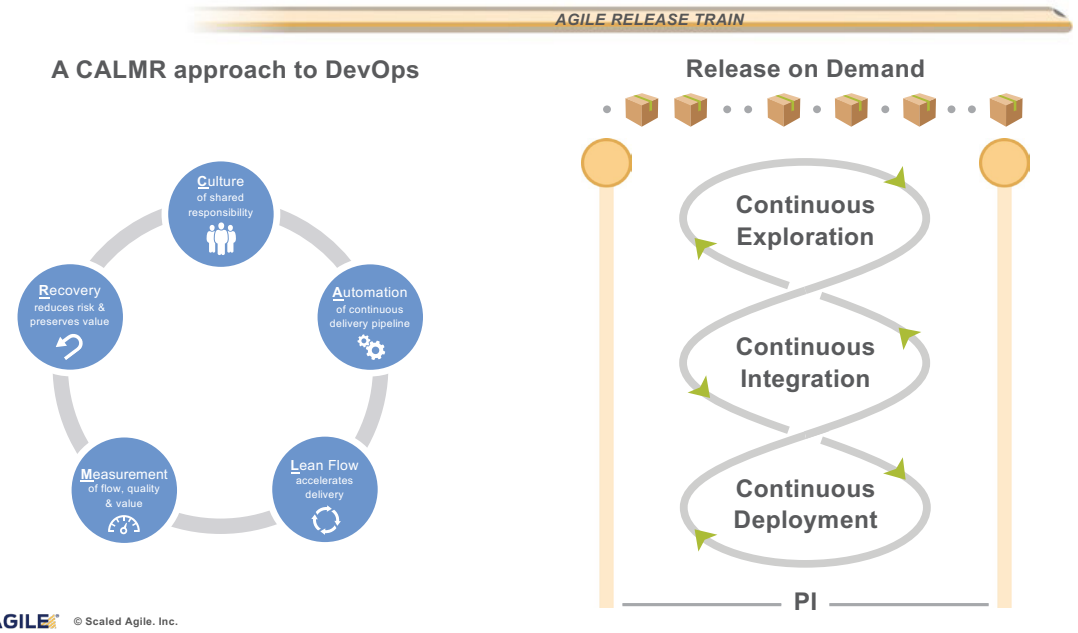
Adapted from xprogramming.com

1-63

Teams in SAFe are part of an Agile Release Train



The ART and teams continuously deliver value



Action Plan: Introducing SAFe

Prepare 5 min

Share 3 min

- **Step 1:** As a team, brainstorm one to three actions you could take to improve in any areas related to this lesson
- **Step 2:** Individually write down at least one improvement item
- **Step 3:** Share one item you discussed as a team and one item you individually wrote in your Action Plan

1-66

Introducing SAFe

Lesson review

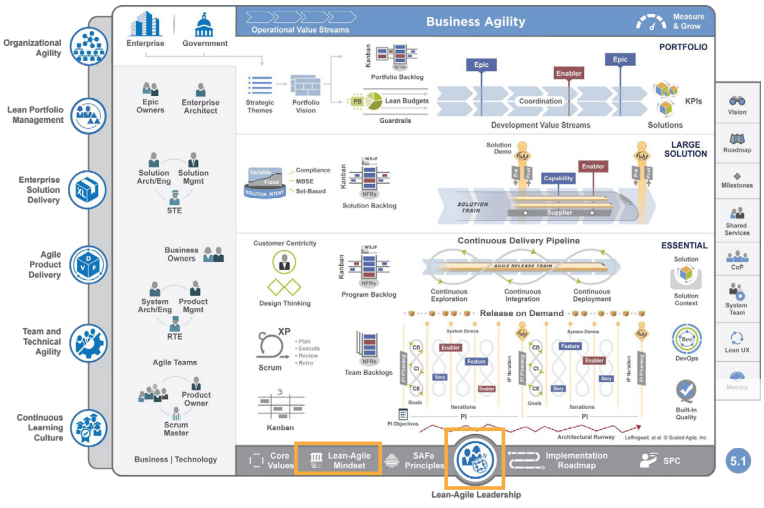
In this lesson you:

- Described what is necessary to thrive in the digital age
- Recognized SAFe as an operating system for Business Agility
- Explored the seven core competencies of Business Agility
- Explored the Lean-Agile Mindset
- Reviewed the SAFe Lean-Agile Principles
- Described Scrum, Kanban, and XP practices

Articles used in this lesson

Read these Framework articles to learn more about topics covered in this lesson

- "Lean-Agile Mindset" https://v5.scaledagileframework.com/lean-agile-mindset/
- "Lean-Agile Leadership" https://v5.scaledagileframework.com/lean-agile-leadership/
- "SAFe Lean-Agile Principles" https://v5.scaledagileframework.com/safe-lean-agile-principles/



Continue your SAFe journey with the following resources

Watch this three-minute video, *Navigating the Big Picture*, to understand how to use the SAFe Big Picture. **https://bit.ly/Video-NavigatingtheBigPicture**	Build your knowledge of the goals and methods of SAFe to achieve Business Agility with the "What is SAFe for Lean Enterprise" e-learning. **https://bit.ly/Community-GettingStarted**
Watch this four-minute video, *Lean-Agile Mindset,* to learn why a Lean-Agile mindset is an important Enabler for Business Agility. **https://bit.ly/Video-LeanAgileMindset**	Access the SAFe Collaborate template, "Experience Batch Sizes," to practice putting large and small batches through a system. **https://bit.ly/Template-BatchSize**
Download and share the "Introducing SAFe" toolkit to familiarize people in your organization with SAFe. **https://bit.ly/Community-ToolkitsandTemplates**	Complete the e-learning, "Agile Basics," to learn more about what Agile is and how it supports value delivery. **https://bit.ly/Community-GettingStarted**

1-69

Lesson notes

Enter your notes below. If using a digital workbook, save your PDF often so you don't lose any of your notes.

Lesson 2

Building an Agile Team

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

SP SAFe® 5

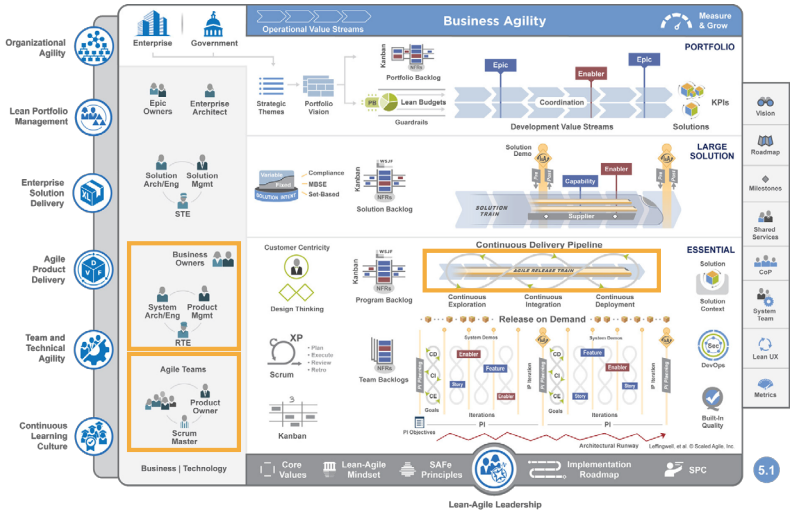
SCALED AGILE®

Lesson Topics

2.1 Forming cross-functional Agile Teams

2.2 The Scrum Master and Product Owner roles

2.3 Organizing Agile Release Trains



2-2

Learning objectives

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

- Build your Agile Team
- Explain the Scrum Master and Product Owner roles
- Identify the people and teams on the train, and learn about their roles

2.1 Forming cross-functional Agile teams

The power of a high performing team

We, the work, and the knowledge are all one.

- A self-organizing team dynamically interacts with itself and the organization
- Team members create new points of view and resolve contradictions through dialogue
- The team is energized with intentions, vision, interest, and mission
- Leaders provide autonomy, variety, trust, and commitment

2-5

Teams create and challenge norms

- There is creative chaos via demanding performance goals
- The team is challenged to question every norm of development
- Equal access to information at all levels is critical

2-6

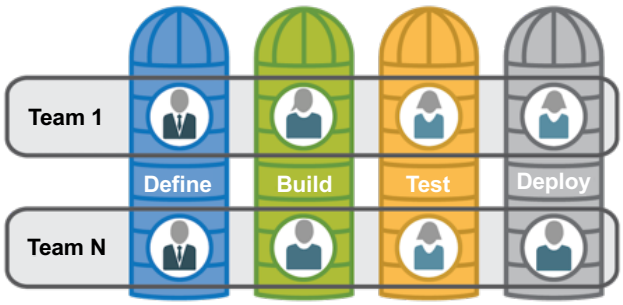
“It is amazing what you can accomplish if you do not care who gets the credit.”
—Harry S. Truman

2-7

Build cross-functional Agile Teams

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

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

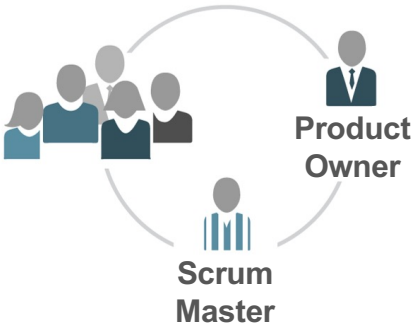


SCALED AGILE © Scaled Agile, Inc.

2-8

Responsibilities of the Agile Team

- 5 to 11 team members
- Create and refine Stories and acceptance criteria
- Define, build, test, and deploy Stories
- Build quality in to each increment of the solution
- Develop and commit to team PI Objectives and Iteration goals



Discussion: Agile Teams in your workplace

Prepare 5 min | Share 3 min

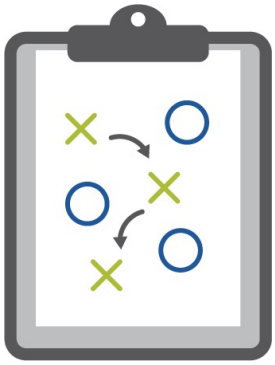
- **Step 1:** Discuss these questions
  - What would a cross-functional team look like at your workplace?
  - How would this change the results your team delivers?
- **Step 2:** Be prepared to share with the class

Activity: Experience teams – Purpose

Duration 5 min

Purpose

- Experience how an Agile Team functions
- Get as many balls through the group as possible within two minutes
- After two minutes, the group is allowed an additional minute (one-minute Retrospective) to discuss the process and how it could be improved. The game is played a total of three times. (three two-minute Iterations).

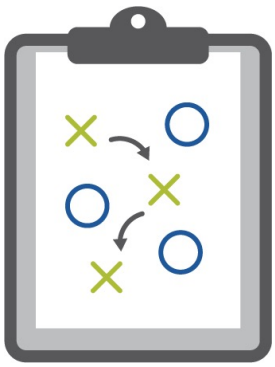


Activity: Experience teams – Rules

Duration 5 min

Rules

- Everyone is part of one big group
- Each ball must have air-time
- Each ball must be touched at least once by everyone
- Balls cannot be passed to your direct neighbor (to your immediate left or right)
- Each ball must return to the same person who introduced it into the system
- There are a total of three Iterations



Activity: Experience teams

Prepare 15 min | Share 5 min

- **Step 1:** Organize into small groups.
- **Step 2:** As a group, estimate how many balls you think you can process in two minutes.
- **Step 3:** Run the two-minute Iteration following the rules (get as many balls through the group as possible)
- **Step 4:** Take one minute (run the one-minute Retrospective) to discuss how you can improve the process

Note: To get credit, you must provide an estimate for the number of balls you think you can process before each Iteration.

- **Step 5:** Summarize your experience as a group:
  - What were some of the challenges?
  - How were you able to improve the process with each Iteration?

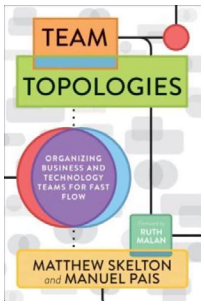
Teams on the ART are organized for flow

Stream-aligned team – organized around the flow of work and has the ability to deliver value directly to the Customer or end user.

Complicated subsystem team – organized around specific subsystems that require deep specialty skills and expertise.

Platform team – organized around the development and support of platforms that provide services to other teams.

Enabling team – organized to assist other teams with specialized capabilities and help them become proficient in new technologies.



More information in the Advanced Topic Article:
https://v5.scaledagileframework.com/organizing-agile-teams-and-arts-team-topologies-at-scale/

Applying the four topologies

- **Stream-aligned Teams (best)**
  - By product, Solution, or service
  - By Customer or market segment
  - By Solution feature areas
  - By steps in the Customer journey
  - By value streamlets
  - New product innovation
- **Complicated Subsystem Teams**
  - Highly specialised system components
  - Safety critical systems elements
  - Specialty algorithm or business rules
  - Part of a cyber-physical system

Applying the four topologies - continued

- **Platform Teams**
  - Sets of services consumed by other Teams
- **Enabling Teams**
  - DevOps implementation
  - Automated testing
  - Continuous integration and build tooling
  - Engineering quality practices
  - Security Environments and configuration

Activity: Building your team

Prepare

10 min

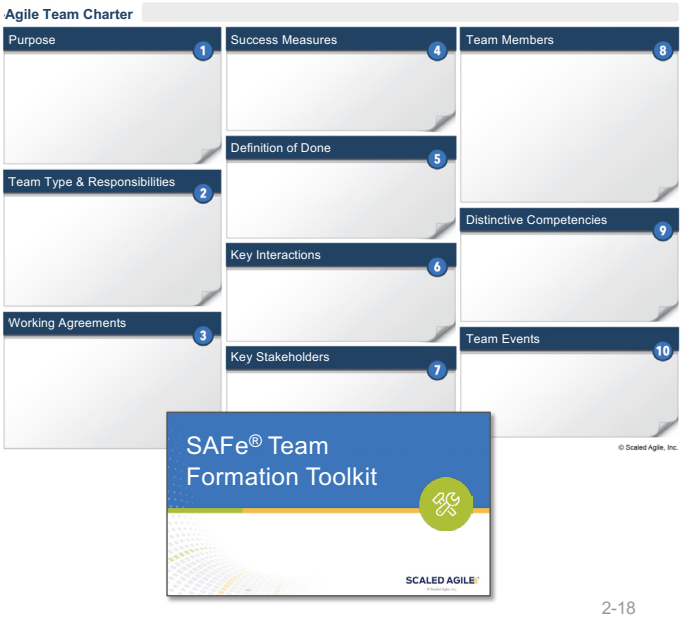
- **Step 1:** As a group, discuss each person's responsibilities and skill sets.
- **Step 2:** Create a group name. Note: Names should not be the names of components, subsystems, or Feature areas. Instead, create a fun name, a mascot, and a cheer.
- **Step 3:** Discuss your role as one of the four team topologies.
- **Step 4:** Discuss what your group is responsible for and what other things you can do.
- **Step 5:** Prepare a short presentation about your group (name, role on the train, and special skills that other groups should know about).

2-17

Facilitate an Agile Team Charter workshop

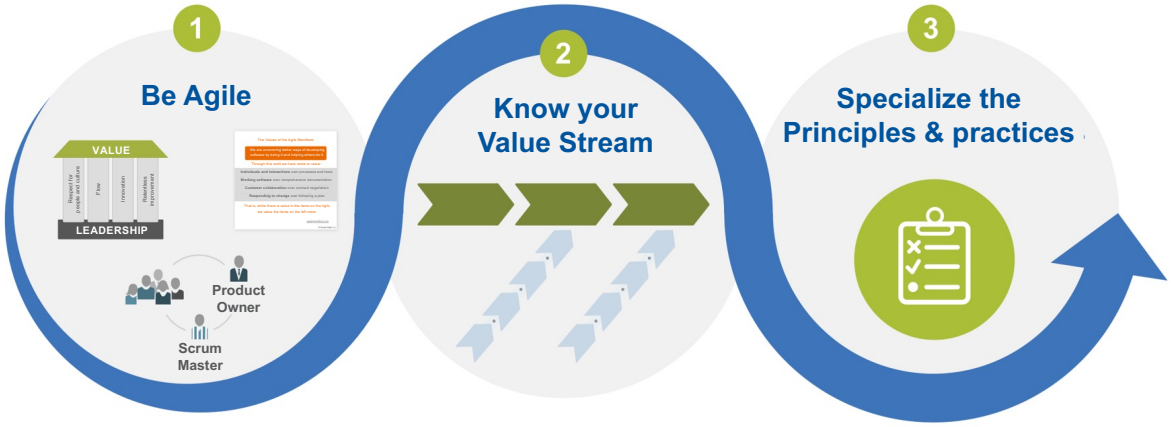
Part of the SAFe Team Formation Toolkit.

- The Agile Team Charter helps teams to clearly define their purpose, responsibilities and success criteria amongst other critical elements.
- The process of completing the Agile Team Charter provides the opportunity for teams to discuss and reflect both on how they want to work together and with other teams on the ART.
- For each of the 10 boxes on the charter an interactive exercise is included to generate discussion and the required output.



2-18

Extend into the business with Agile business teams



Agile Team maturity cycle

2.2 The Scrum Master and Product Owner roles

Agile Teams have two speciality roles

Scrum Master

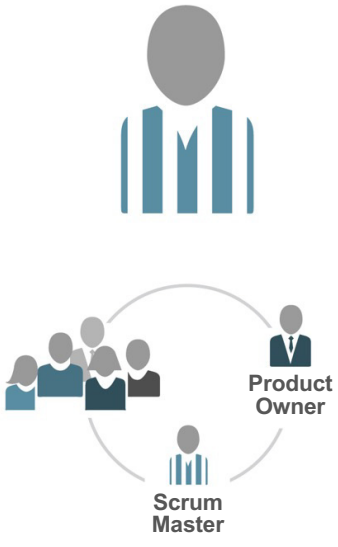
- Coaches the Agile Team in self-management
- Helps the team focus on creating increments of value each iteration
- Facilitates the removal of impediments to the team's progress
- Ensures that all team events take place, are productive and kept within the timebox

Product Owner

- Contributes to the Vision and Roadmap
- Acts as the Customer for team questions
- Creates, clearly communicates and accepts Stories
- Prioritizes the Team Backlog

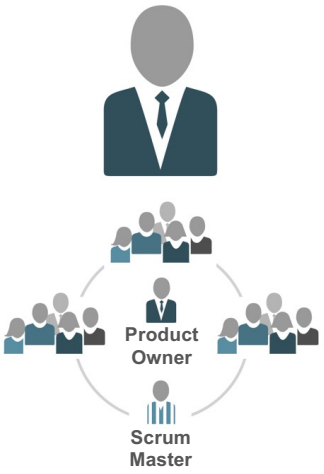
The Scrum Master in the Enterprise

- Coordinates with other Product Owners, the System Team, and Shared Services in the PI Planning meetings
- Works with the above teams throughout each Iteration and PI
- Coordinates with other Scrum Masters and the Release Train Engineer in the Scrum of Scrums
- Helps team understand and operate within its capacity
- Helps team operate under architectural and portfolio governance, system-level integration, and System Demos
- Fosters team adoption of Agile technical practices



The Product Owner in the Enterprise

- Establishes the sequence of backlog items based on program priorities, events, and dependencies with other teams
- Operates as part of an extended Product Management Team
- Understands how the Enterprise backlog structure operates with Epics, Capabilities, Features, and Stories
- Work with Product Management to plan Program Increments (PI)
- Uses PI Objectives and Iteration Goals to communicate with management
- Coordinates with other Product Owners, the System Team, and Shared Services in the PI Planning meetings
- Works with other Product Owners and the Product Management team throughout each Iteration and PI

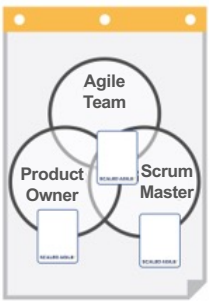


Activity: Scrum roles and responsibilities

Prepare 8 min

Share 2 min

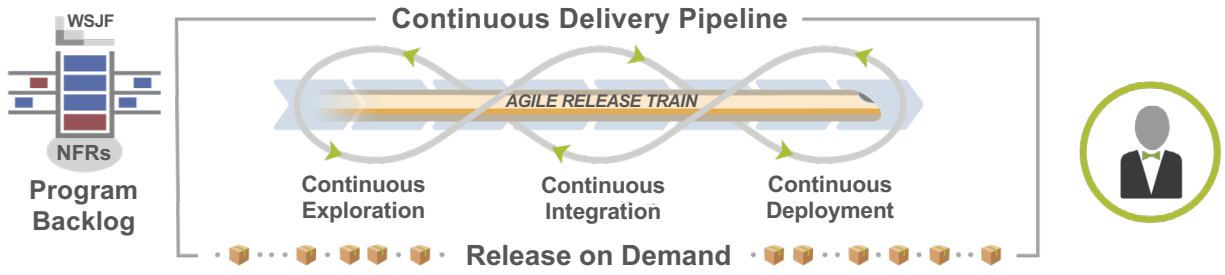
- **Step 1:** With your group, draw the following Venn diagram on a flipchart sheet
- **Step 2:** Review the responsibility cards
- **Step 3:** Place them either in the role or at an intersection of the Venn diagram
- **Step 4:** Present your Venn diagram to the class



2.3 Organizing Agile Release Trains

Agile Release Trains (ARTs)

- 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



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.

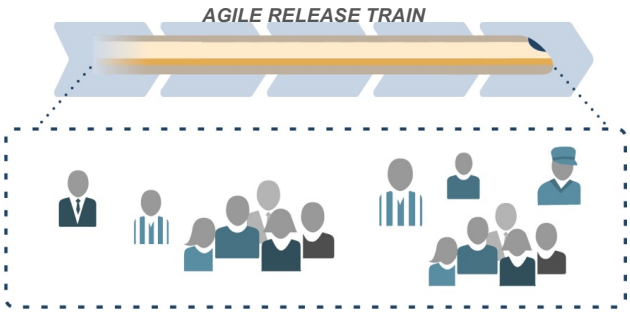
Business Owners are key stakeholders on the Agile Release Train.

2-27

Activity: Know the people on the train

Share 20 min

- **Step 1:** The RTE introduces themself
- **Step 2:** The RTE presents the main players on the train:
  - Product Management
  - System Architect/Engineering
  - Lean UX
  - Shared Services
- **Step 3:** Each group presents itself (name, area of responsibility, special skills)



2-28

Action Plan: Building an Agile Team

Prepare 5 min

Share 3 min

- **Step 1:** As a group, brainstorm one to three actions you could take to improve in any areas related to this lesson
- **Step 2:** Individually write down at least one improvement item
- **Step 3**: Share one item you discussed as a group and one item you individually wrote in your Action Plan

Building an Agile Team

Lesson review

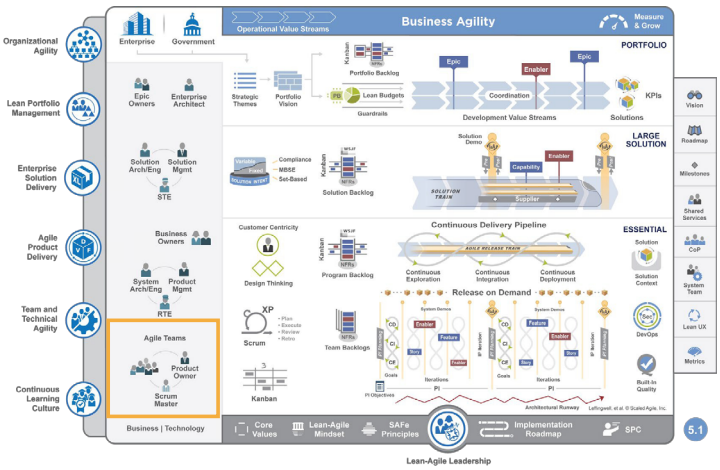
In this lesson you:

- Built your team and learned about their roles
- Explored the roles of the Scrum Master and the Product Owner
- Identified the people and teams on the train, and learned about their roles

Articles used in this lesson

Read these Framework articles to learn more about topics covered in this lesson

- "Agile Teams" https://v5.scaledagileframework.com/agile-teams/
- "Product Owner" https://v5.scaledagileframework.com/product-owner/
- "Scrum Master" https://v5.scaledagileframework.com/scrum-master/
- "Organizing Agile Teams and ARTs: Team Topologies at Scale" https://v5.scaledagileframework.com/organizing-agile-teams-and-arts-team-topologies-at-scale/



Continue your SAFe journey with the following resources:

Watch this three-minute video, *Welcome to Your Scrum Team,* for an overview of what to expect in your first days on a Scrum team, and the role each team member plays in executing as a team. **https://bit.ly/Video-WelcometoYourScrumTeam**	Download and use the "Team Formation Toolkit (5.1)" to build your Agile Team charter and clearly define your purpose, responsibilities and success criteria, and other critical elements necessary for your new team to flourish. **https://bit.ly/Community-ToolkitsandTemplates**
Use the "Measure and Grow Workshop Toolkit (5.1)" to run a workshop to identify growth opportunities for your team as you work toward mastering SAFe and Business Agility. **https://bit.ly/Community-ToolkitsandTemplates**	Share the "Essential SAFe Toolkit (5.1)" with your team to help ensure a shared understanding of the basic building blocks needed for a successful SAFe implementation. **https://bit.ly/Community-ToolkitsandTemplates**
Facilitate an exercise using the "Experience Teams!" SAFe Collaborate template to practice working as a team and using retrospectives to improve team performance. **https://bit.ly/Template-ExperienceTeams**	Use the "Agile Team Charter Virtual Workshop" in SAFe Collaborate to help a remote or distributed team capture the ideas and decisions from the "Team Formation Toolkit (5.1)." **https://bit.ly/Template-SAFeAgileTeamCharter**

2-32

Lesson notes

Enter your notes below. If using a digital workbook, save your PDF often so you don't lose any of your notes.

Lesson 3

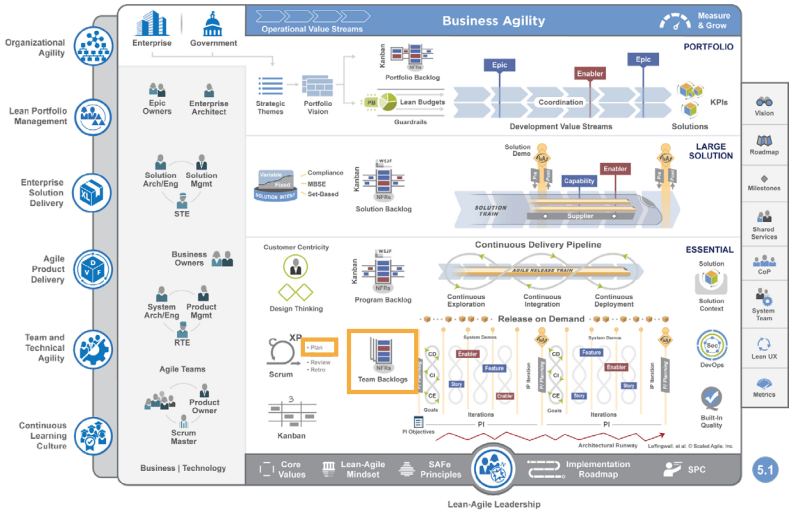
Planning the Iteration

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

SP
SAFe® 5

SCALED AGILE®

Lesson Topics



3-2

Learning objectives

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

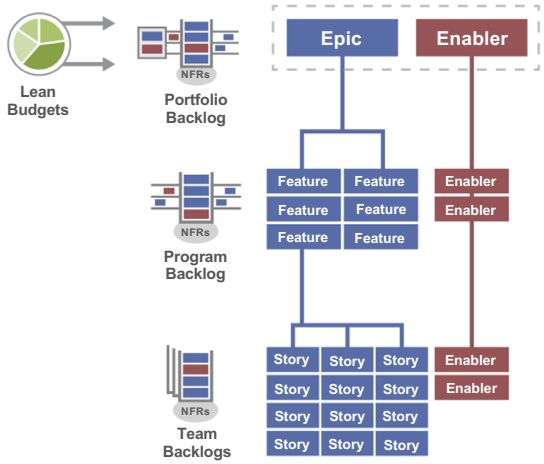
- Identify the qualities of well-written Features and Stories
- Prepare your backlog by breaking down Features into Stories
- Discuss the key considerations for sequencing work
- Apply estimation poker for fast, relative estimation
- Explain how to conduct an Iteration Planning event
- Explore Kanban as a team method for visualizing workflow

3.1 Preparing the Team Backlog

Define Features for the Program Backlog

Features are services that fulfill stakeholder needs.

- Feature is an industry-standard term familiar to marketing and Product Management
- Describe Features with a short phrase and a benefit hypothesis that clearly expresses their value
- Identify, prioritize, estimate, and maintain Features in the Program Backlog



Features represent the work for the Agile Release Train

- The Feature benefit hypothesis justifies development cost and provides business perspective for decision-making
- Features contain acceptance criteria typically defined during Program Backlog refinement
- They reflect functional and non-functional requirements
- Features fit in one PI

In-service software update

Benefit hypothesis
Significantly reduced planned downtime

Acceptance criteria
1. Nonstop routing availability
2. Automatic and manual update support
3. Rollback capability
4. Support through existing admin tools
5. All enabled services are running after the update

Example Feature

Example Features

Software Example

Multi-factor authentication

Benefit hypothesis

Enhanced user security will reduce risk of a system data breach

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
5. Data breach tests pass

Business Example

Create GDPR Incident Response Plan

Benefit hypothesis

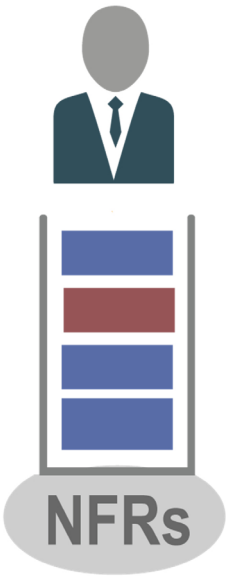
Organizational readiness to quickly respond to incidents

Acceptance criteria

1. Incident response plan is fully documented
2. Incident response plan is reviewed and approved by PO
3. Incident response is compliant with legal requirements

The Team Backlog

- Contains all the work for the team
- Created by the Product Owner and the team
- Prioritized by the Product Owner
- Contains User and Enabler Stories
  - User Stories provide Customers with value
  - Enabler Stories build the infrastructure and architecture that makes User Stories possible
- Stories for the next Iteration are more detailed than Stories for later Iterations
- Nonfunctional requirements (NFRs) are a constraint on the backlog



User Stories

- User Stories are short descriptions of a small piece of desired functionality, written in the user's language
- The recommended form of expression is the user-voice form, as follows:
  As a (user role), I want to (activity), so that (business value)

As a driver, I want to limit the amount of money before I fuel so that I can control my expenditure

As a driver, I want to get a receipt after fueling so that I can expense the purchase.

As the Finance Department, we want to print receipts only for drivers who request them so that we save on paper

SCALED AGILE © Scaled Agile. Inc.

3-9

Using personas to better understand users

Personas are detailed fictional characters acting as a representative user.

Jane - Mileage sensitive
- Law-abiding driver
- Obeys all traffic signs
- Wants to save on gas

Bob - Time sensitive
- Impatient driver
- Ignores traffic signs if they slow him down

As Jane, I want to travel at the legal limit and operate in an energy saving manner so that I do not get a ticket and I save money

As Bob, I want to travel at the maximum speed the roadway and my vehicle safely allows so that I arrive quickly

SCALED AGILE © Scaled Agile. Inc.

3-10

INVEST in a good Story

I	**Independent**	Write Stories that can be developed separately
N	**Negotiable**	Write Stories in which scope can be negotiated
V	**Valuable**	Write Stories that are valuable to the Customer
E	**Estimable**	Write Stories that can be estimated
S	**Small**	Write Stories that can fit in an Iteration
T	**Testable**	Write Stories that are testable

Writing good Stories: The 3Cs

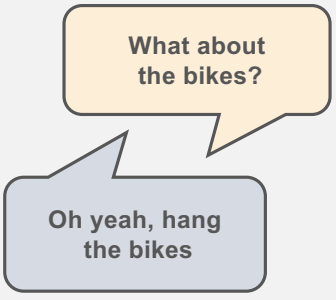
Card

Written on a card or in the tool and may annotate with notes

As a spouse, I want a clean garage so that I can park my car and not trip on my way to the door.

Conversation

The details are in a conversation with the Product Owner



Confirmation

Acceptance criteria confirm the Story correctness

- **Tools have been put away**
- **Items on the floor have been returned to the proper shelf**
- **Bikes have been hung**

Source: 3Cs coined by Ron Jeffries

Enabler Stories

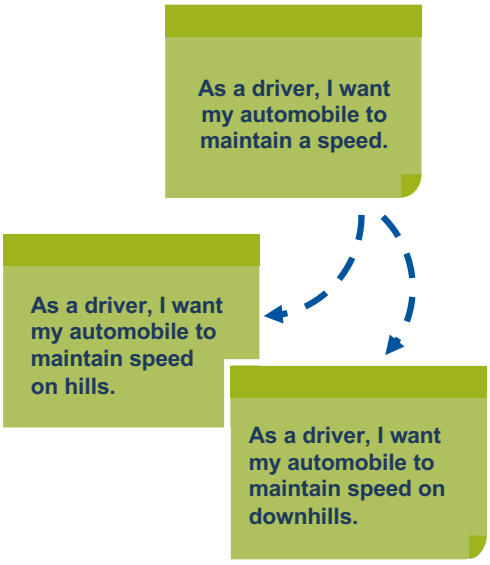
Enabler Stories build the groundwork for future User Stories. There are four types of Enabler Stories:

- **Infrastructure:** Build development and testing frameworks that enable a faster and more efficient development process
- **Architecture:** Build the Architectural Runway, which enables smoother and faster development
- **Exploration:** Build understanding of what is needed by the Customer to understand prospective Solutions and evaluate alternatives
- **Compliance:** Facilitate specific activities such as verification and validation, documentation, signoffs, regulatory submissions, and approvals

Splitting Stories

- **In support of small batches for flow, decrease size to minimum:**
  - Split Stories into essential and non-essential parts and eliminate the non-essential
  - Ensure you have something releasable
- **In support of feedback:**
  - Deploy small Stories to get technical/user feedback quickly (maximize feedback)
- **In support of Iteration Planning:**

  Split Stories so they fit into an Iteration



Apply some common splitting techniques

Splitting techniques:

- Business rule variations (e.g. single variation, then remainder)
- Workflow steps (for multi-step stories)
- Simple/complex (e.g. search for single word, then for phrases)
- Scenarios (e.g. use case exceptions)

Activity: Break Features into Stories

Duration 15 min

Work with your group to break Features from the Program Backlog into Stories.

- **Step 1:** Select one Feature from the Program Backlog (your own or use the example provided in your workbook)
- **Step 2:** As a group, break the Feature into Stories in a way that they still retain business value
- **Step 3:** Write these User Stories in the User Story format:
  - **As a (user role), I want (activity) so that (business value).**

Program Backlog Example Features

Feature: **Flexible search**

Benefit: Users will have a flexible, easy-to-use search capability to locate books.

Description: Search by author, title, or genre from a single search field. Mispelling subsitutions (i.e., "Did you mean ...?"). Present results as per-match algorithm.

Feature: **Shopping Cart**

Benefit: Users can manage items in a shopping cart for immediate or future purchase.

Description: Users can easily access their cart from any page, view the same infromation displayed in the book list, change the quantity, remove it from their cart, or save it for later. A subtotal for all items in their shopping cart should be displayed at the bottom. Items saved for later should appear below that.

Feature: **Purchase by credit card**

Benefit: Users can purchase products from us (as soon as implemented–only beta up until then)

Description: Users can select from their preferred credit card and shipping address as defined in their profile or add new ones. Visa, mastercard, Discover, and Diners Club ar required. American Express is optional. Must be PCI compliant.

Feature: **Shipping method selection**

Benefit: Users can select a shipping method based on cost, delivery speed, and carrier.

Description: Users can select a shipping method based on the price, delivery speed, and estimated delivery date for all major carriers (USPS, UPS, and FedEx).

Feature: **Profile management**

Benefit: Users can create and maintain their profiles rather than enter in their information each time they order.

Description: Users can manage their login credentials (ID, password), personal information (name, email address, home address), nickname for book rating and commenting, credit card information (multiple), and shipping address (multiple). Physical addresses, email addresses, and credit card info should be verified as valid. Passwords must meet current security standards.

Feature: **Book detail**

Benefit: Users can see informative and enticing details about a book.

Description: Display book name, book cover (which can be enlarged when clicked), author and bio, book description, genre, publishing info (publisher, release date, etc.), bok rating, and comments. Hyperlink author's name to a list of other books by the same author.

Feature: **Book list sorting**

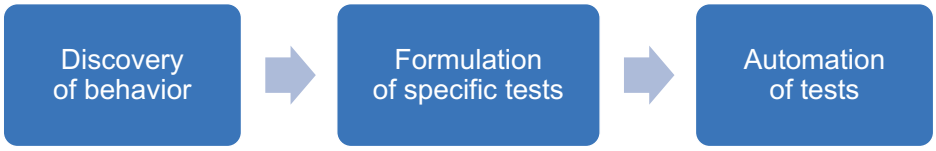
Benefit: Users can sort a list of books in a number of ways to more easily find what they are looking for.

Description: Sort by book title, author, price, book rating, and release date. Allow for users to select the number of search results to appear on each page.

Behavior-driven development: From ambiguity to precision

- Behavior is often first described in general terms, which can be ambiguous
- Specific examples of behavior provide better understanding
- The examples can directly become tests, or they can lead to specific behaviors which then are transformed into tests

Example



Acceptance criteria

- Acceptance criteria provide the details of the Story from a testing point of view
- They are created by the Agile Team
- Can be written in the 'Given-When-Then' format

As a driver, **I want** to limit the amount of money before I fuel **so that** I can control my expenditure.

Acceptance Criteria:

1. **Given** that the driver indicated a maximum amount of money
   When the fuel cost reaches the amount
   Then the fueling process stops automatically
2. …

As a driver, **I want** to get a receipt after fueling so that I can expense the purchase.

Acceptance Criteria:

1. **Given** that the fueling is over
   When the driver asked for the receipt
   Then it is printed and includes:
   amount fueled, amount paid, tax, vehicle number, date, time

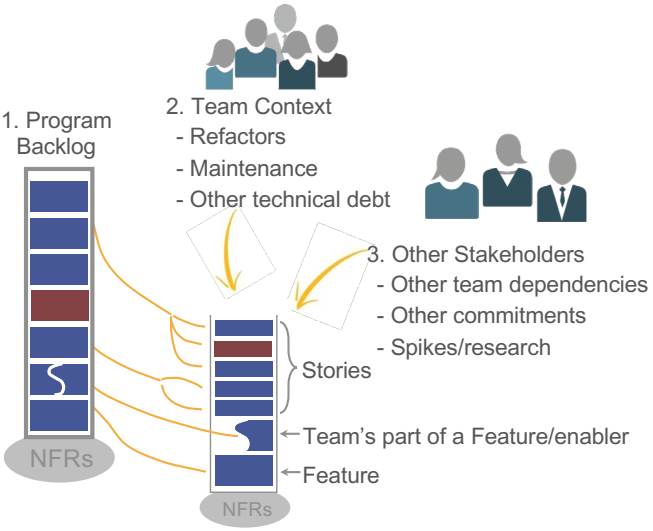
Activity: Write acceptance criteria

Prepare 10 min | Share 10 min

- **Step 1:** With your group, write acceptance criteria in the Given-When-Then format for three User Stories you have created.
- **Step 2:** Make sure the acceptance criteria are testable.
- **Step 3:** Discuss with your group:
  - Did writing acceptance criteria in the Given-When-Then format identify the need for any additional details?
- **Step 4:** Be prepared to share with the class.

Sequencing Stories

- The Product Owner and the team sequence stories based on:
  - Priorities inherited from the Program Backlog
  - Capacity allocations for defects, maintenance, and refactors
  - Dependencies with other stories, teams, events, Milestones and releases.
- Initial sequencing happens during PI Planning
- Adjustments happen at Iteration boundaries

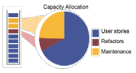


3.2 Planning the Iteration

Plan and commit with Iteration Planning

Purpose	Define and commit to what will be build in the Iteration
Process	• The Product Owner defines what • The team defines how and how much • Four hours max
Result	Iteration Goals and Backlog of the team's commitment
Reciprocal commitment	• Team commits to delivering specific value • Business stakeholder commit to leaving priorities unchanged during the Iteration

Iteration Planning flow

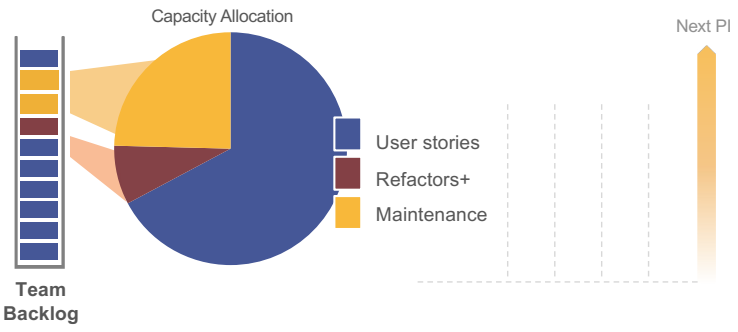
1. Establishing capacity 
2. Story analysis and estimating 
3. Detailing Stories 
4. Developing Iteration goals 
5. Committing to Iteration goals 

Iteration Planning

- **Timebox:** Four hours or less
- This event is **by** and **for** the team
- SMEs may attend as required

Capacity allocation for a healthy balance

- By having capacity allocation defined, the Product Owner doesn't need to prioritize unlike things against each other
- Once the capacity allocation is set, the PO and team can prioritize like things against each other



Capacity allocation

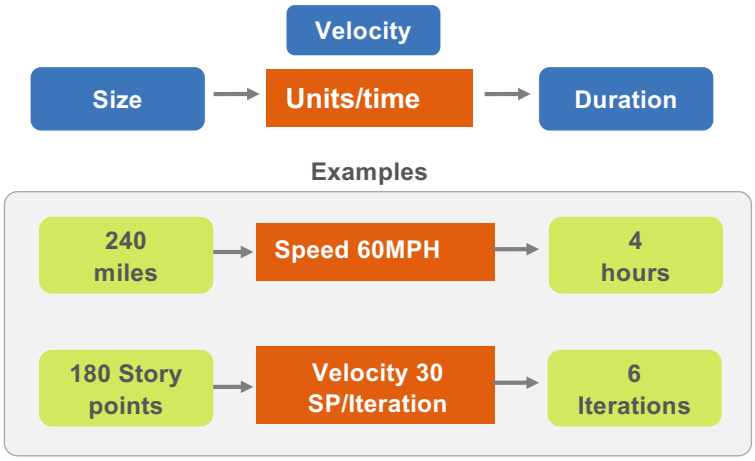
- Helps alleviate velocity degradation due to technical debt
- Keeps existing customers happy with bug fixes and enhancements
- Can change at Iteration or PI boundaries

Using size to estimate duration

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

Definition of Velocity

Velocity is the number of points of stories accepted in the Iteration. Make sure to always use the average velocity for the most recent Iterations.



3-25

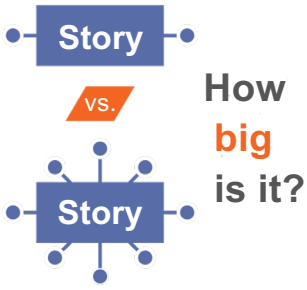
Story analysis and estimation

- The Product Owner presents Stories in order of priority
- Each Story
  - Is discussed and analyzed by the team
  - Has its acceptance criteria refined
  - Is estimated
- The process continues until the total Story points of the estimated Stories has reached the capacity of the team

3-26

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.
  - An 8-point Story should take four times longer than a 2-point story to complete
  - Typically, a 1-point story would take one day to develop and test



3-27

Apply estimating poker for fast, relative estimating

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

?	1	2	3	5	8	13	20	40	100	∞

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

Source: Mike Cohn, Agile Estimating and Planning

Note: estimating poker is a form of the **Wideband Delphi** estimation method, a consensus-based technique for estimating effort.

3-28

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*

How much time to spend estimating

A little effort helps a lot. A lot of effort only helps a little.

Activity: Estimate Stories

Prepare 10 min

Share 3 min

- **Step 1:** As a group, use estimating poker to estimate three Stories created in the previous activity.
- **Step 2:** Share with the class:
  - Where do you find challenges when engaged in Story estimation?
  - Are you as a team aligned around the combination of qualities that represent a Story point (volume, complexity, knowledge, uncertainty)?

Tasking Stories

Some teams break their Stories into tasks and forecast them in hours to better understand their capacity and capabilities.

Team members discuss:

- Who would be the best person to accomplish it?
- Approximately how long would it take?
- What dependencies might it have to other Stories?

Tasking stories is an optional practice in SAFe

Iteration Goals

Iteration goals provide clarity, commitment, and management information. They serve three purposes.

Align team members to a common purpose

Align teams to common Program Increment (PI) Objectives and manage dependencies

Provide transparency and management information

Iteration goals: Examples

Software Example

Iteration Goals

1. Finalize and push last-name search and first-name morphology
2. Index 80% of remaining data
3. Other Stories:
   - Establish search replication validation protocol
   - Refactor artifact dictionary schema

Business Example

Iteration Goals

1. Roll out the GDPR incident report procedures
2. Prepare for external audit
3. Obtain approvals for financial report

Commit to the Iteration goals

Team commitments are not just to the work. They are committed to other teams, the program, and the stakeholders.

A team meets its commitment:

By doing everything they said they would do,
- or -
in the event that it is not feasible, they must immediately raise the concern.

Commitment

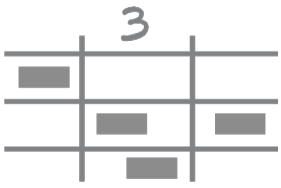
Too much holding to a commitment can lead to burnout, inflexibility, and quality problems.

Adaptability

Too little commitment can lead to unpredictability and lack of focus on results.

Iteration planning for Kanban teams

- Some teams have a more responsive nature to their work, such as maintenance teams and System Teams
- These teams may find less value in trying to plan the Iteration in detail, and will tend towards Kanban
- Kanban teams still publish Iteration goals, and integrate with other teams continuously or on cadence
- They commit to the goals as well as service level agreements (response time) for incoming work based on their known historical lead time
- They participate in PI Planning, System Demos and Inspect and Adapt like all other teams



Kanban

Video: Implementing Kanban

Duration 3 min

https://bit.ly/Video-ImplementingKanban

Action Plan: Planning the Iteration

Prepare 5 min

Share 3 min

- **Step 1:** As a group, brainstorm one to three actions you could take to improve in any areas related to this lesson
- **Step 2:** Individually write down at least one improvement item
- **Step 3**: Share one item you discussed as a team and one item you individually wrote in your Action Plan

Planning the Iteration

Lesson review

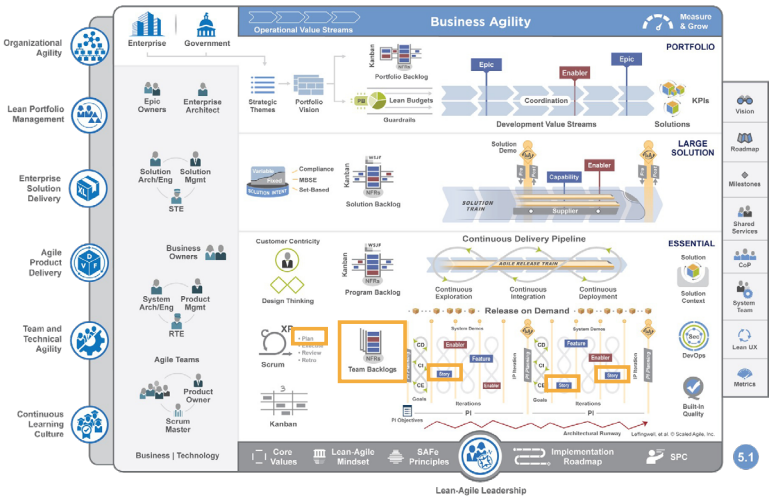
In this lesson you:

- Identified the qualities of well-written Features and Stories
- Prepared your backlog by breaking down Features into Stories
- Discussed the key considerations for sequencing work
- Applied estimation poker for fast, relative estimation
- Reviewed the process for planning the Iteration
- Explored Kanban as a team method for visualizing workflow

Articles used in this lesson

Read these Framework articles to learn more about topics covered in this lesson

- "Story" https://v5.scaledagileframework.com/story/
- "Team Backlogs" https://v5.scaledagileframework.com/team-backlog/
- "Iteration Planning" https://v5.scaledagileframework.com/iteration-planning/



Continue your SAFe journey with the following resources:

Watch the two videos in this playlist to learn more about the two different types of stories. https://bit.ly/Video-StoriesPlaylist	Access the “SAFe ART and Team Events” on the SAFe Community Platform for additional tools and guidance for preparing to facilitate SAFe events. https://bit.ly/Community-SAFeARTandTeamEvents
Download and read the “Story Writing and Splitting Guide” to learn what makes a good story and how to split stories. https://bit.ly/Community-StoryWritingGuide	Practice story splitting with your team using this SAFe Collaborate template, “Story Splitting on an Agile Team” https://bit.ly/Template-StorySplitting
Download the “SAFe Iteration Execution Toolkit (5.1)” for a set of tools and guides for facilitating significant Iteration events. https://bit.ly/Community-ToolkitsandTemplates	Download the “SAFe Remote ART Toolkit 5.1” for additional tools and guides for releasing value with distribute or remote teams. https://bit.ly/Community-ToolkitsandTemplates

3-41

Lesson notes

Enter your notes below. If using a digital workbook, save your PDF often so you don't lose any of your notes.

Lesson 4

Executing the Iteration

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

SP
SAFe® 5

SCALED AGILE®

Lesson Topics

4.1 Visualizing the flow of work

4.2 Measuring the flow of work

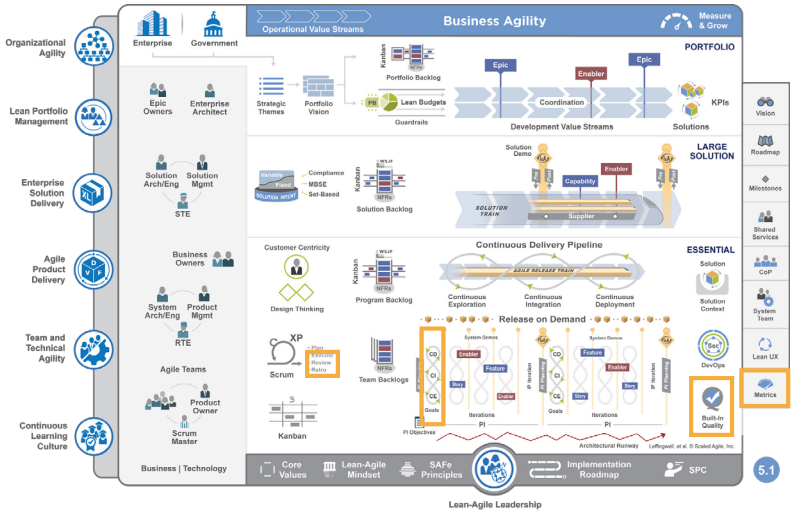
4.3 Building quality in

4.4 Continuously integrating, deploying, and releasing

4.5 Improving flow with communication and synchronization

4.6 Demonstrating value with the Iteration Review

4.7 Improving with the Iteration Retrospective



4-2

Learning objectives

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

- Visualize your current flow of work
- Apply cumulative flow diagrams for measuring flow
- Identify the different aspects of Built-in Quality
- Recognize the importance of a Continuous Delivery Pipeline
- Describe the communication and synchronization benefits of the daily stand-up
- Explain how to run a successful backlog refinement session
- Describe the purpose of the Iteration Review and create a definition of done
- Participate in a retrospective and identify improvements

4.1 Visualizing the flow of work

Video: Designing your team's Kanban system

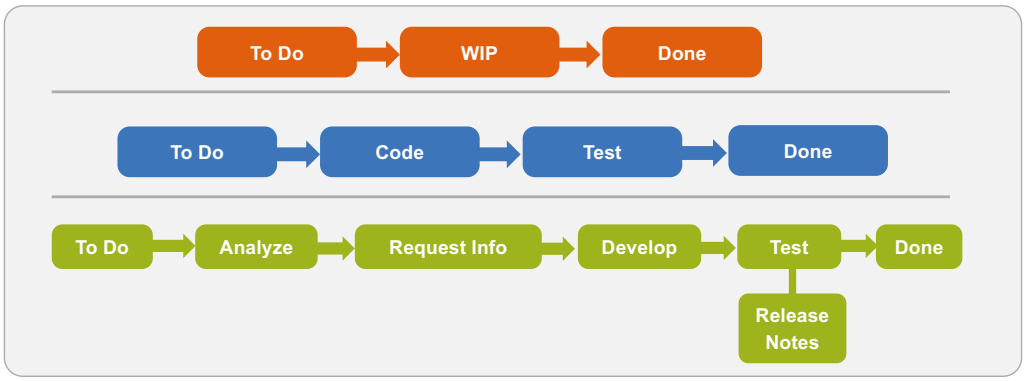
Duration 4 min

https://bit.ly/Video-DesignKanban

Visualize the flow of work

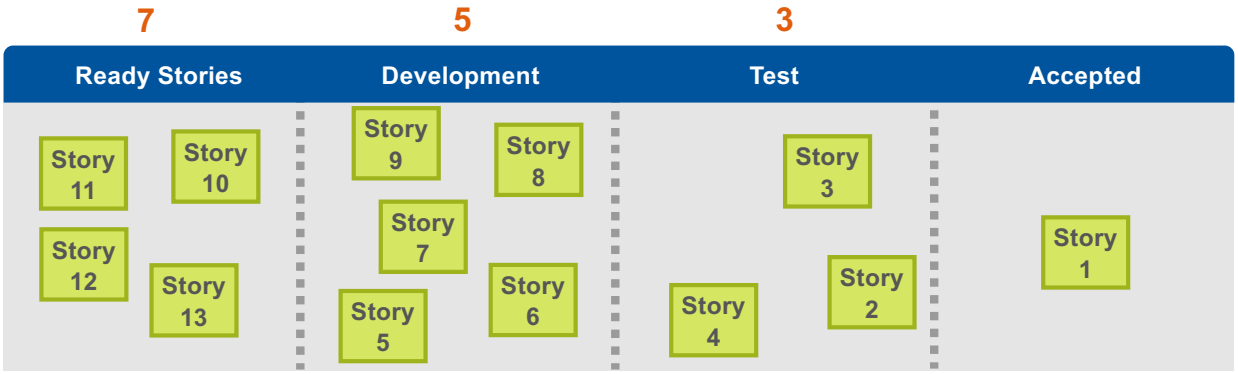
- What is the flow of work for your team?
- What are the steps it takes to get a Story to done?

Example: Flow of work



Setting WIP limits

- WIP limits improve the flow of work
- Some steps have no WIP limits, while others serve as buffers and have minimum as well as maximum WIP



Activity: Visualize the flow of work

Prepare 8 min

Share 2 min

- **Step 1:** Consider the Stories you estimated in the previous lesson
- **Step 2:** Define the steps you need to turn the Stories into value
- **Step 3:** As a group, build your current flow of work and assign WIP limits
- **Step 4:** Be prepared to share your flow of work with the class

4.2 Measuring the flow of work

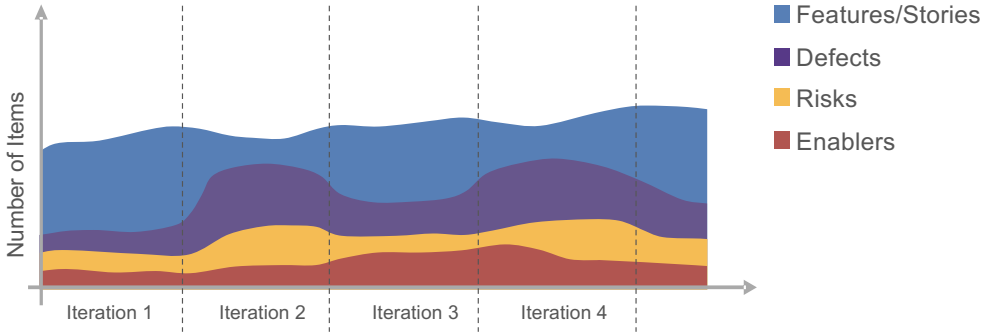
SAFe defines six flow metrics

Metric	Description
Flow Distribution	The proportion of each backlog item type in the flow
Flow Velocity	Number of items completed in a given time
Flow Time	Time elapsed from when an item enters the workflow to when it is released to the customer
Flow Load	Total work-in-progress (across all steps of the flow)
Flow Efficiency	The portion of time backlog items are actively worked on to the total time elapsed
Flow Predictability	Overall planned vs. actual business value

The first five metrics are from Mik Kersten's Flow Framework described in his book Project to Product

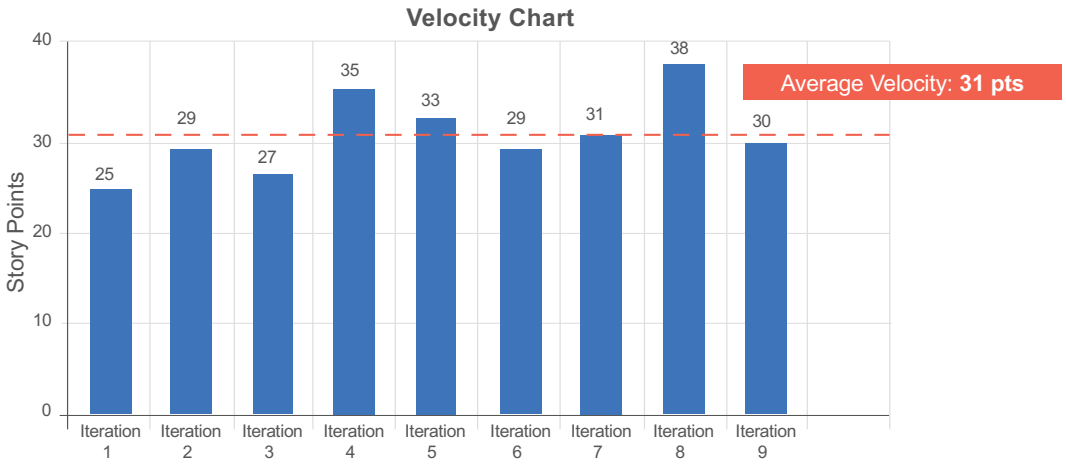
Flow distribution ensures a healthy balance of work item types

What does it measure? Flow distribution measures the amount of each type of work in the system over time.



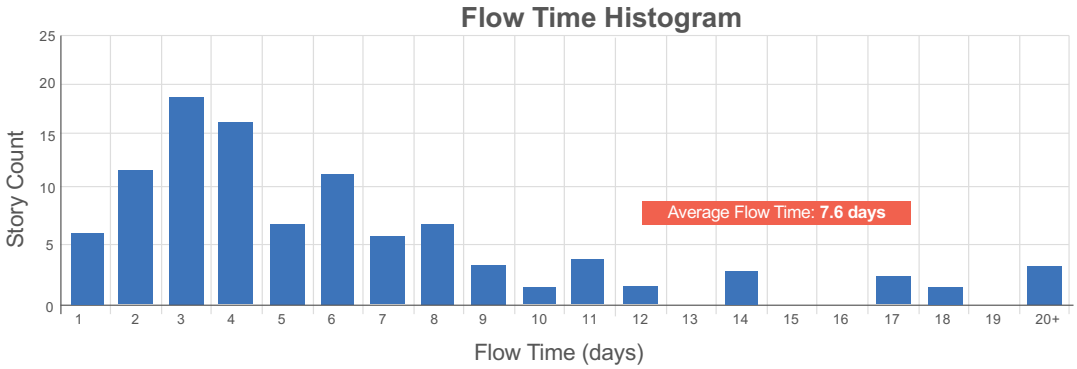
Flow velocity tracks team performance each Iteration

What does it measure? Flow velocity measures the number of backlog items completed in a given timeframe



Flow time ensures teams deliver value in the shortest possible time

What does it measure? Flow time measures the elapsed time from when an item enters the system (developer starts working) to the moment it is delivered to the customer.

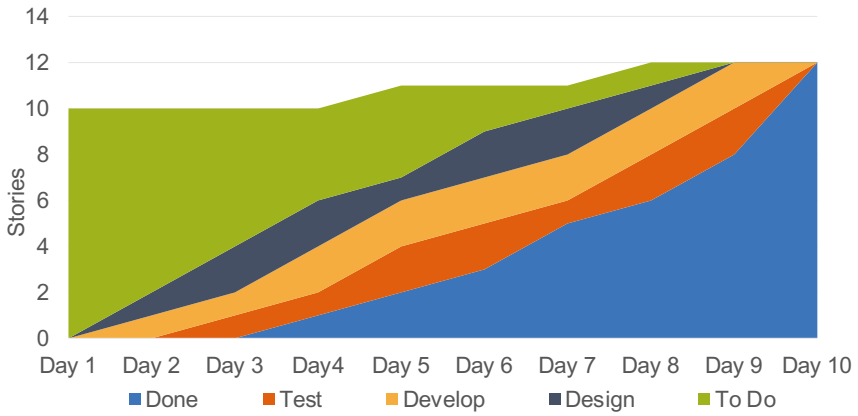


SCALED AGILE © Scaled Agile. Inc.

4-13

Flow load is a leading indicator of excess work in process (WIP)

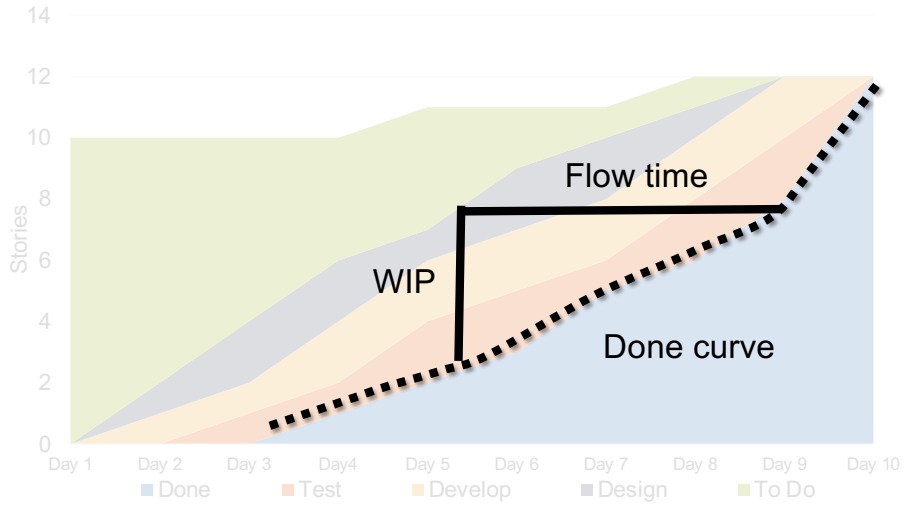
What does it measure? Flow load indicates how many items are currently in the system.



SCALED AGILE © Scaled Agile. Inc.

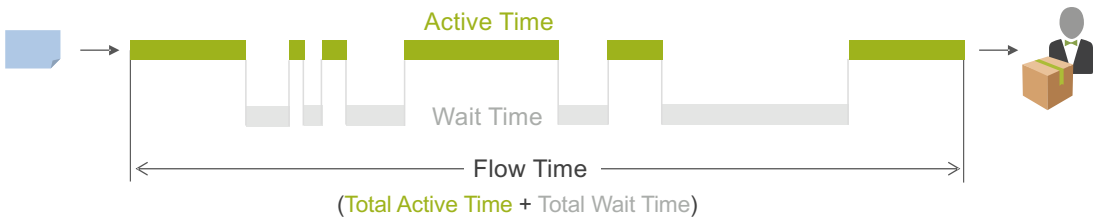
4-14

What can you learn from a cumulative flow diagram (CFD)?



Flow efficiency highlights waste, bottlenecks, and delays in the system

What does it measure? Flow efficiency measures how much of the overall flow time is spent in value-added work activities vs. waiting between steps.



$$\text{Flow Efficiency} = \frac{\text{Total Active Time}}{\text{Flow Time}}$$

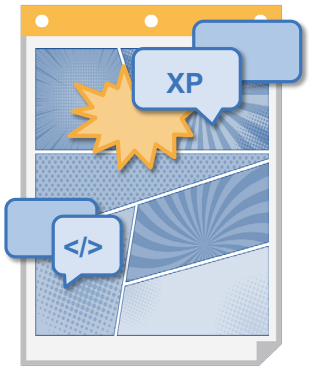
4.3 Building quality in

Activity: Expressing “Build quality in”

Prepare 10 min

Present 5 min

- **Step 1:** In your group, discuss what building quality in means to you and your organization
- **Step 2:** Create a poster to reflect the different aspects of building quality in
- **Step 3:** Present your poster to the class



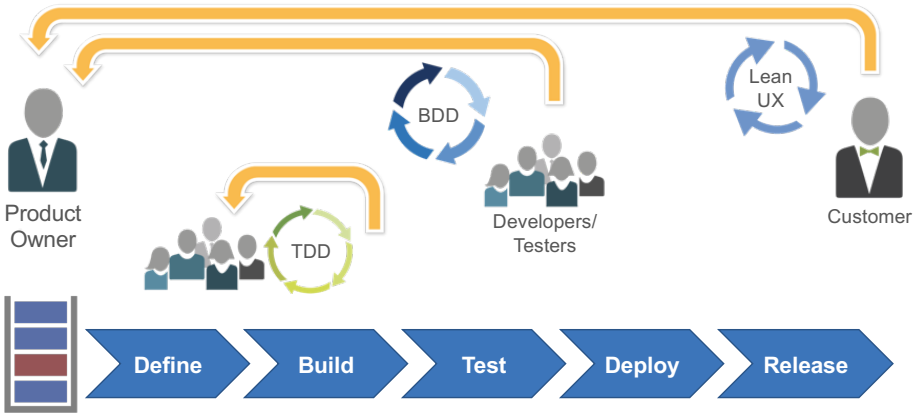
Building quality in

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

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

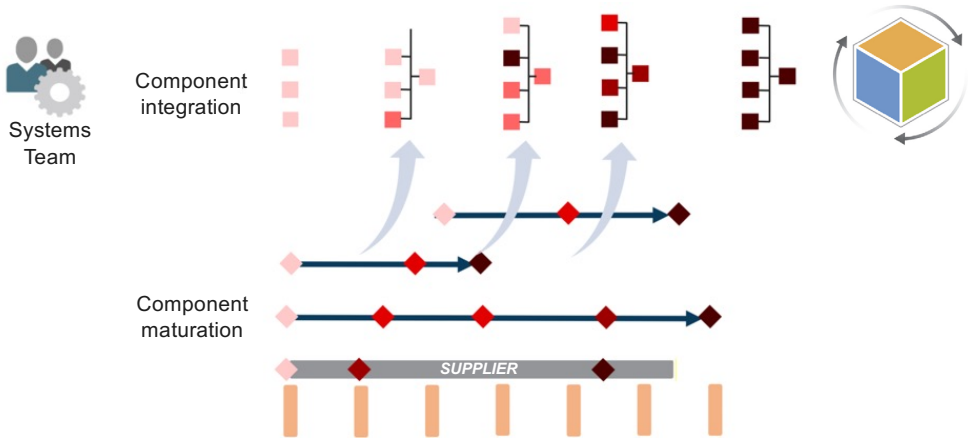
Built-in Quality practices for software teams

They include software quality practices (most inspired by XP) like Agile testing, behavior-driven development, test-driven development, refactoring, code quality, and Agile architecture.



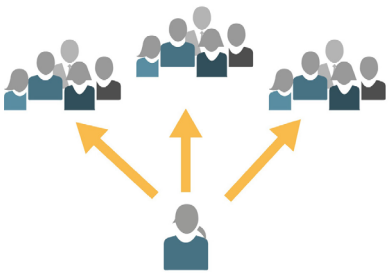
Built-in Quality practices for hardware teams

Support hardware quality with exploratory, early iterations, frequent system-level integration, design verification, Model-Based Systems Engineering (MBSE), and set-based design.



Emergent design and intentional architecture

- Every team deserves to see the bigger picture
- Every team is empowered to design their part
- Emergent design - Teams grow the system design as user stories require
- Intentional architecture - Fosters team alignment and defines the Architectural Runway
- A balance between emergent design and intentional architecture is required for speed of development and maintainability



Architectural Runway

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

Architectural Runway: Examples

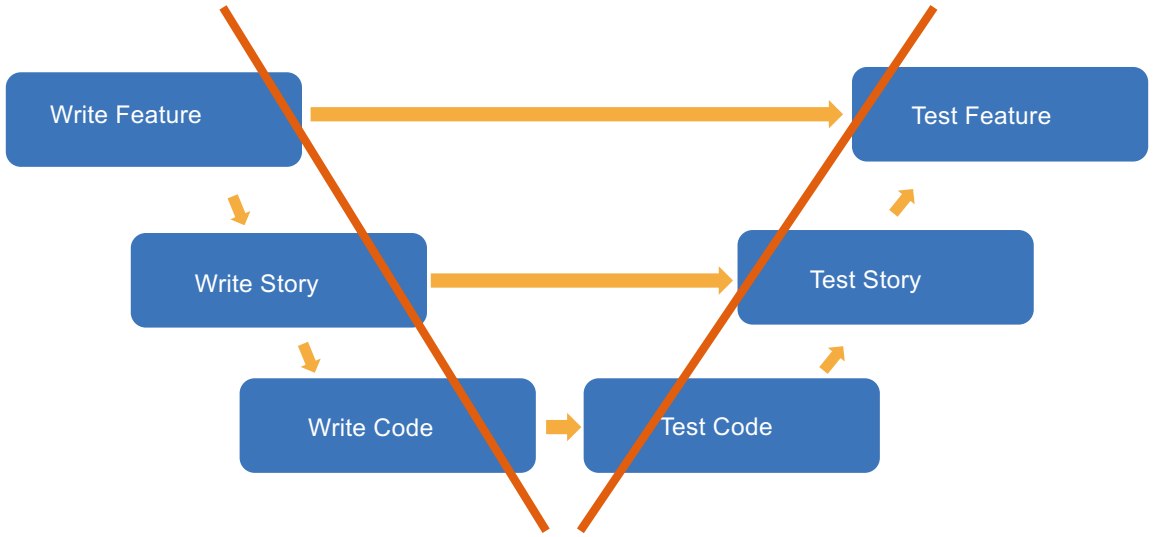
Software Example

A new, fuzzy search algorithm will enable a variety of future Features that can accept potentially erroneous user input

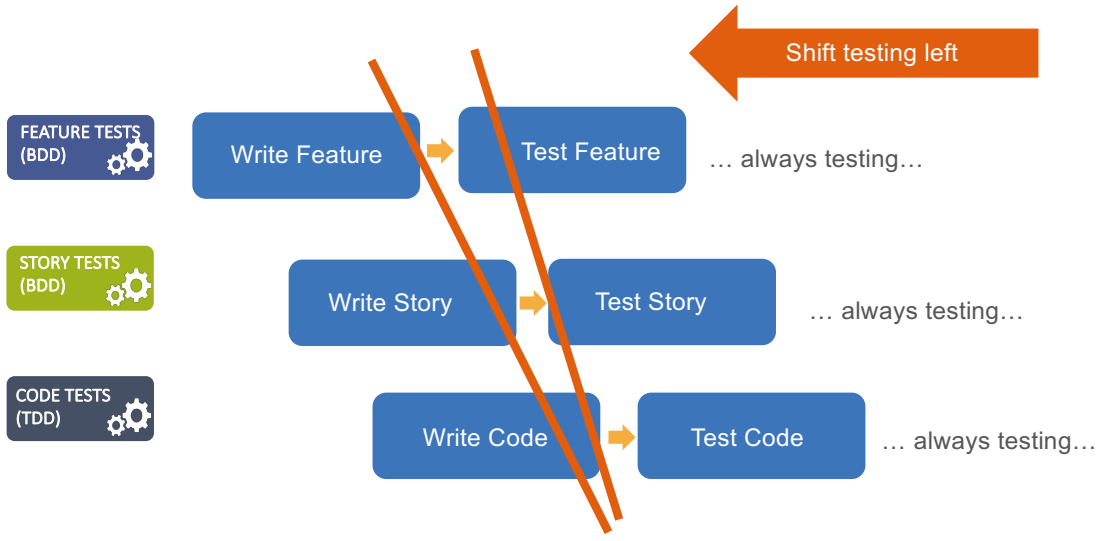
Business Example

HR Example: A job architecture and hiring strategy for Agile talent to enable the company's growth

Traditional testing (V-Model) delays feedback

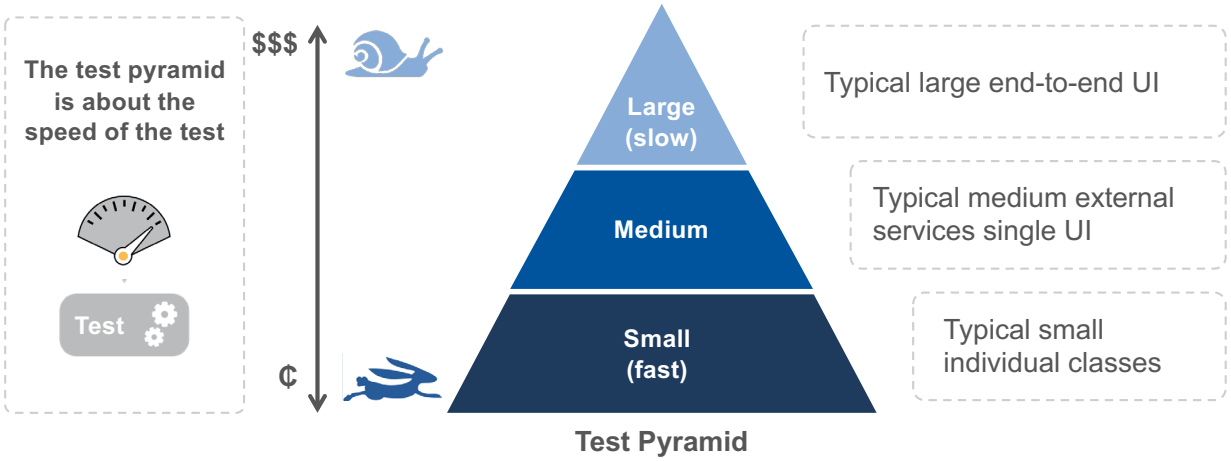


Shift testing left for fast and continuous feedback



Test first naturally creates a pyramid of tests

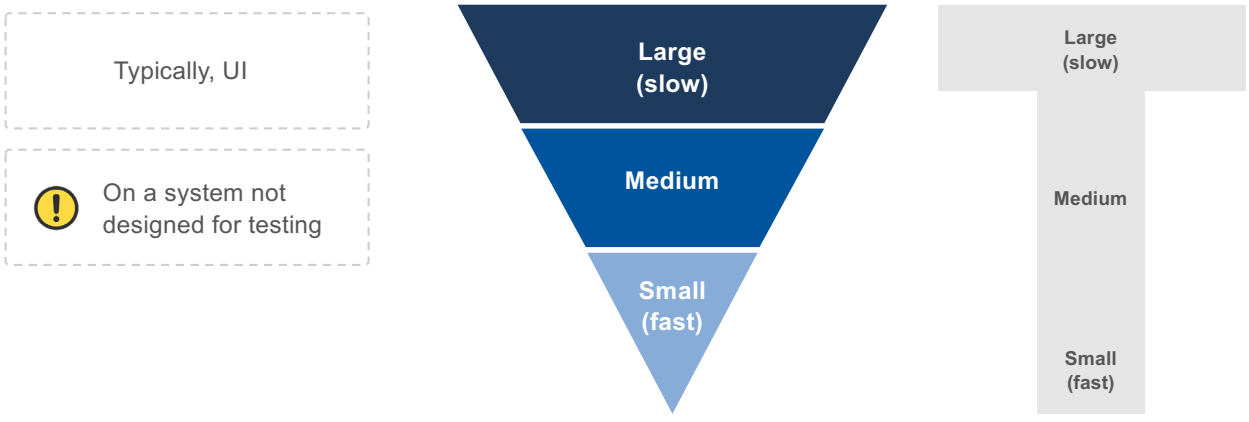
The test pyramid advocates a balanced portfolio of tests with many small, low-level, automated tests and fewer large, manual tests.



Test Pyramid

An inverted Test Pyramid is a test strategy anti-pattern

Slows development, delays feedback, encourages larger batches



4.4 Continuously integrating, deploying, and releasing

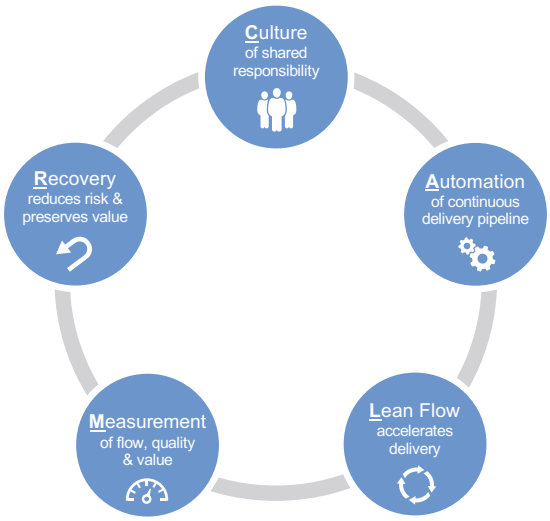
Video: What is DevOps?

Duration 5 min

https://bit.ly/Video-WhatisDevOps

A CALMR approach to DevOps

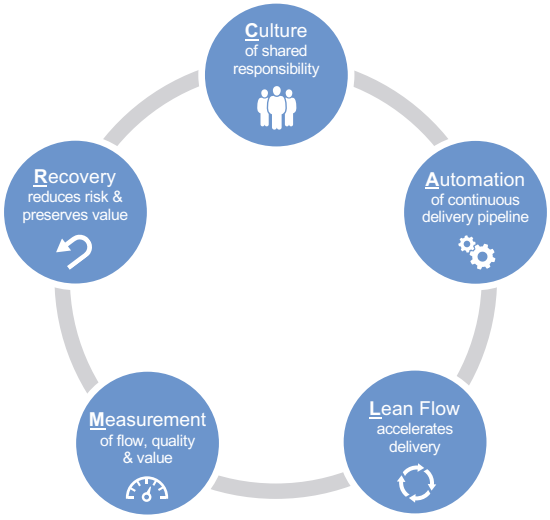
- **C**ulture - Establish a culture of shared responsibility for development, deployment, and operations.
- **A**utomation - Automate the Continuous Delivery Pipeline.
- **L**ean flow - Keep batch sizes small, limit WIP, and provide extreme visibility.
- **M**easurement - Measure the flow through the pipeline. Implement full-stack telemetry.
- **R**ecovery - Architect and enable low-risk releases. Establish fast recovery, fast reversion, and fast fix-forward.



Discussion: CALMR Approach

Duration 3 min

- **Step 1:** Discuss as a team:
  - What opportunities do you have in your current context to apply the concepts in the CALMR approach?
- **Step 2:** Be prepared to share with the class



Video: Continuous Delivery Pipeline

Duration: 5 min

https://bit.ly/Video-CDP

4-33

Discussion: Continuous delivery culture

Prepare: 5 min

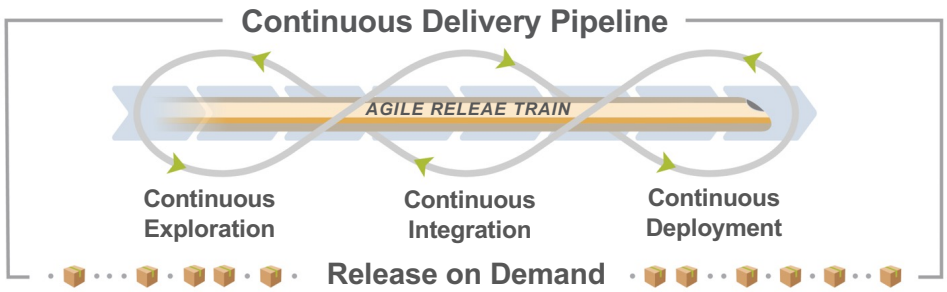
Share: 3 min

- **Step 1:** As a group, discuss the following:
  - How is your culture or environment ready for continuous delivery?
  - What does “continuous” mean to you and your group?
- **Step 2:** Be prepared to share some insights with the class

4-34

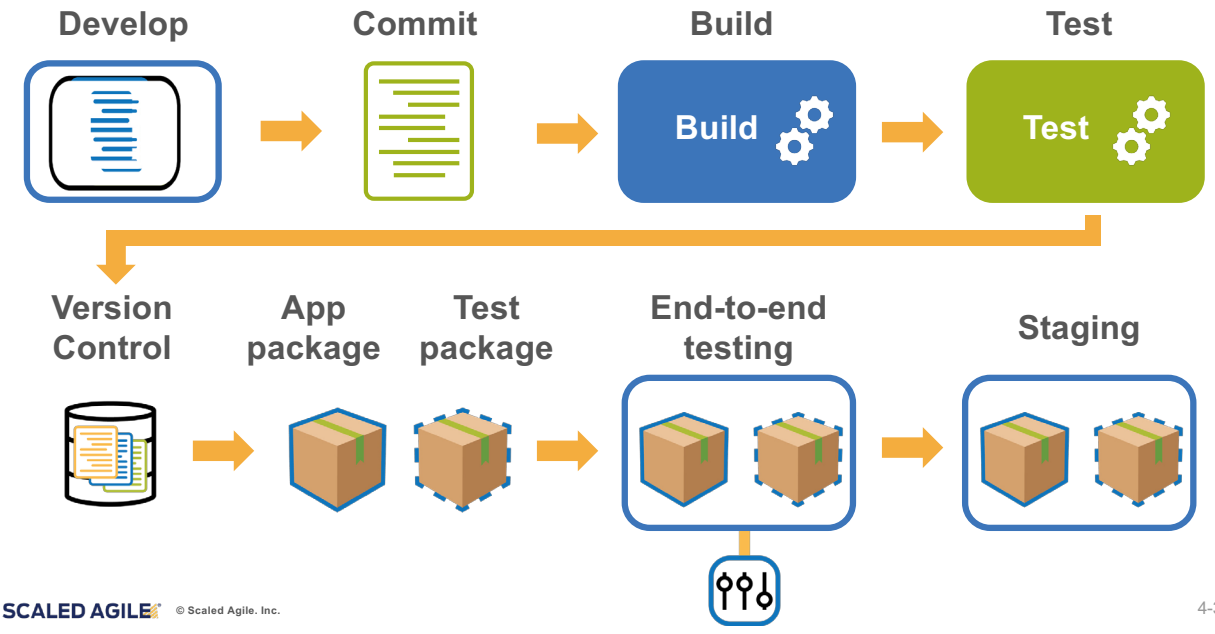
Building the Continuous Delivery Pipeline with DevOps

- The Continuous Delivery Pipeline (CDP) represents the workflows, activities, and automation needed to deliver new functionality more frequently.
- Each Agile Release Train builds and maintains, or shares, a pipeline.
- Organizations map their current pipeline into this new structure and remove delays and improve the efficiency of each step.



SCALED AGILE © Scaled Agile, Inc. 4-35

Continuous code integration

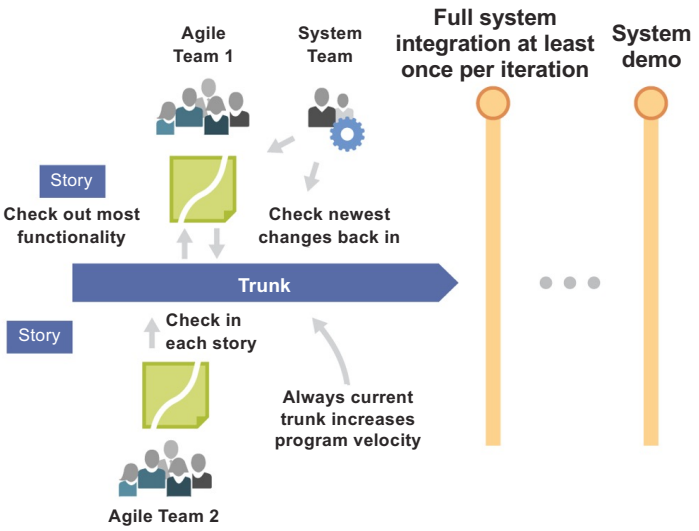


SCALED AGILE © Scaled Agile, Inc. 4-36

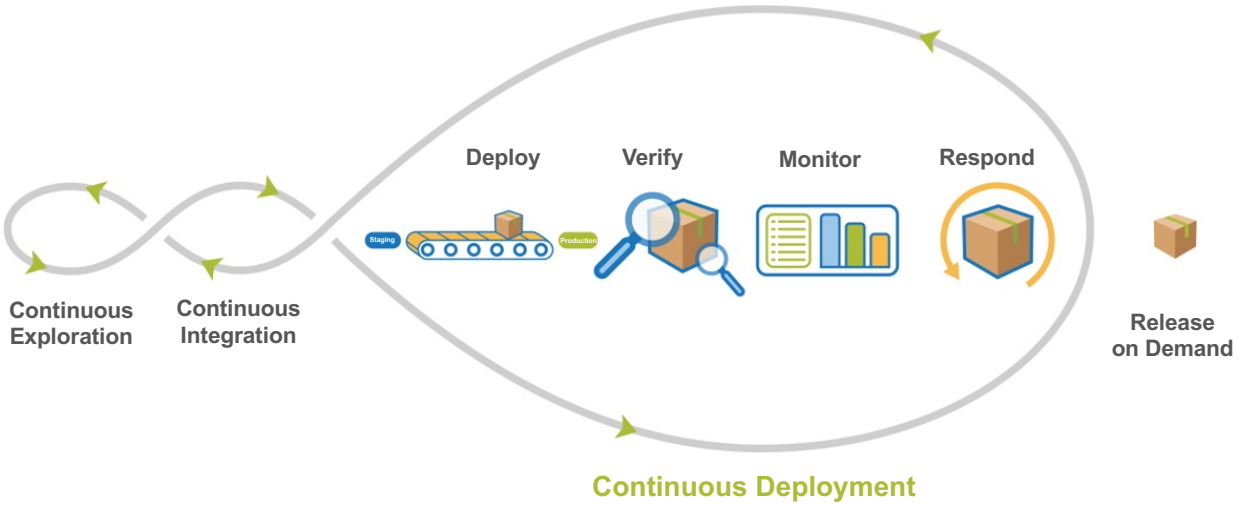
Trunk-based development

Teams continuously integrate assets (leaving as little as possible to the System Team).

- Avoid physical branching for software
- Frequently integrate hardware branches

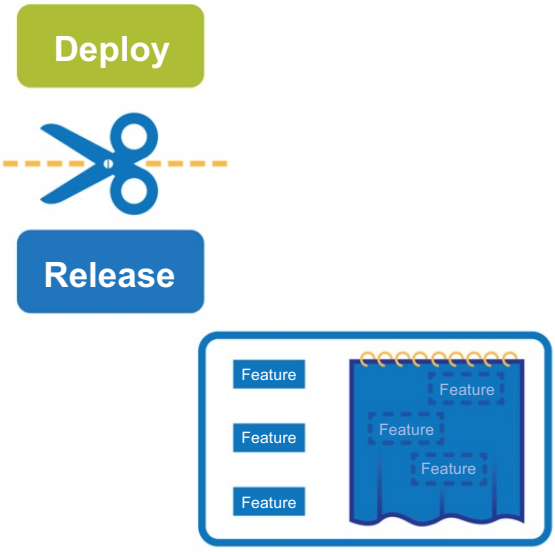


Continuous Deployment – Getting to production early

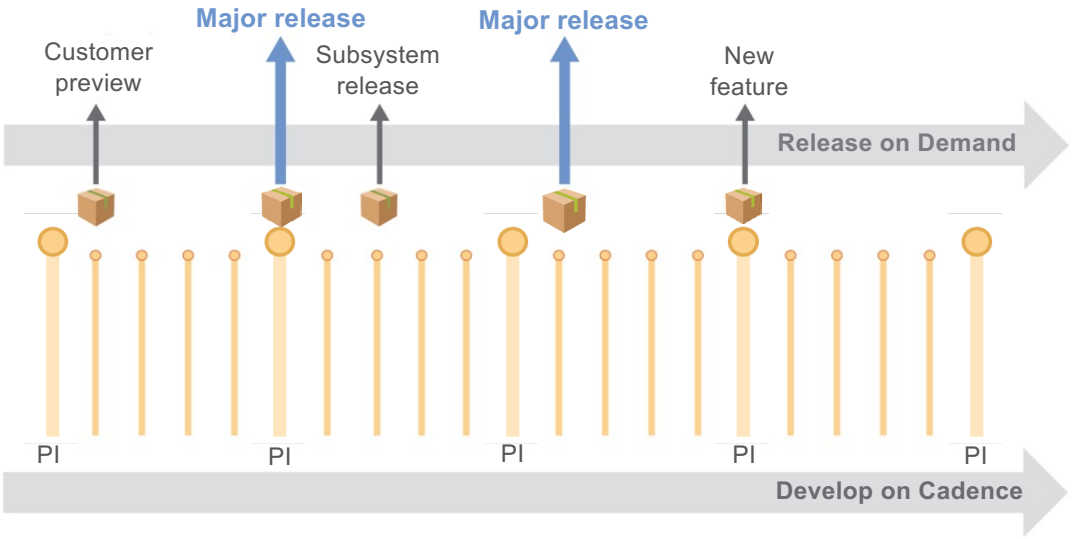


Separate deploy from release

- Separate deploy to production from release
- Hide new functionality under feature toggles
- Enables testing background and foreground processes in the actual production environment before exposing new functionality to users
- Timing of the release becomes a business decision



Develop on cadence. Release on Demand.



Discussion: Continuous integration and deployment challenges

Prepare 15 min | Share 5 min

- **Step 1:** Working in your groups, think about the various aspects of environment, culture, tools, and people and discuss the following:
  - What are the challenges to continuously integrating?
  - What are the challenges to continuously deploying?
- **Step 2:** As a group, prepare a list of three to five items that make it hard to continuously integrate and deploy. What may be some ways to solve them?
- **Step 3:** Be prepared to share with the class.

4.5 Improving flow with communication and synchronization

Communication and synchronization with daily stand-ups

Basic daily stand-Up (DSU) agenda

Each person answers:

1. What did I do yesterday to advance the Iteration Goals?
2. What will I do today to advance the Iteration Goals?
3. Are there any impediments that will prevent the team from meeting the Iteration Goals?

The meet-after agenda

1. Review topics the Scrum Master wrote on the meet-after board
2. Involved parties discuss, uninvolved people leave

4-43

Activity: Reenact the daily stand-up (DSU)

You will participate in and observe a reenactment of the daily stand-up meeting. Four to five volunteers will play the role of *team member*. Your instructor will play the role of the *Scrum Master*.

- **Step 1**: As an observer, take notes and reflect on the following:
  - How long do you think the meeting should be?
  - Where should it take place?
  - What is the main purpose of the daily stand-up?
- **Step 2:** Share some of your insights as an observer

SCALED AGILE © Scaled Agile. Inc.

4-44

Reenact the Daily Stand-Up (DSU)

How long do you think the meeting should be?

Where should it take place?

What is the main purpose of Daily Stand-Up?

Video: Backlog Refinement Workshop

Duration 5 min

https://bit.ly/Video-RunningBacklogRefinement

The backlog refinement workshop

The backlog refinement workshop is a preview and elaboration of upcoming Stories.

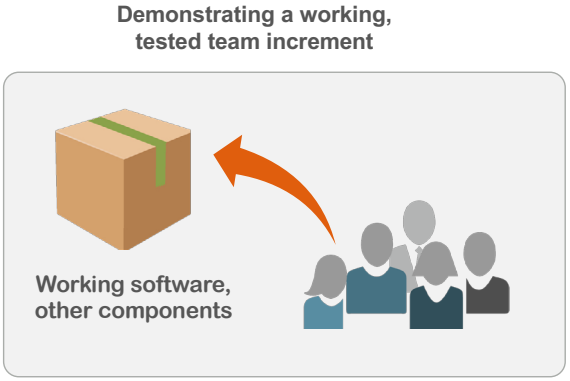
- Helps the team think about new Stories prior to Iteration Planning
- Provides enough time to identify and resolve dependencies and issues that could impact the next Iteration
- The team can improve Stories, add acceptance criteria, and identify missing information
- Most of the focus is on the next Iteration, but it allows time to discuss future Iterations and even Features for the next PI

4.6 Demonstrating value with the Iteration Review

The Iteration Review

- The Iteration Review provides the true measure of progress by showing working software functionality, hardware components, etc.
- Preparation starts with planning
- Teams demonstrate every Story, spike*, refactor, and non-functional requirement (NFR)
- Attendees are the team and its stakeholders

*A spike is a research Story, considered an exploration style Enabler



Iteration Review guidelines

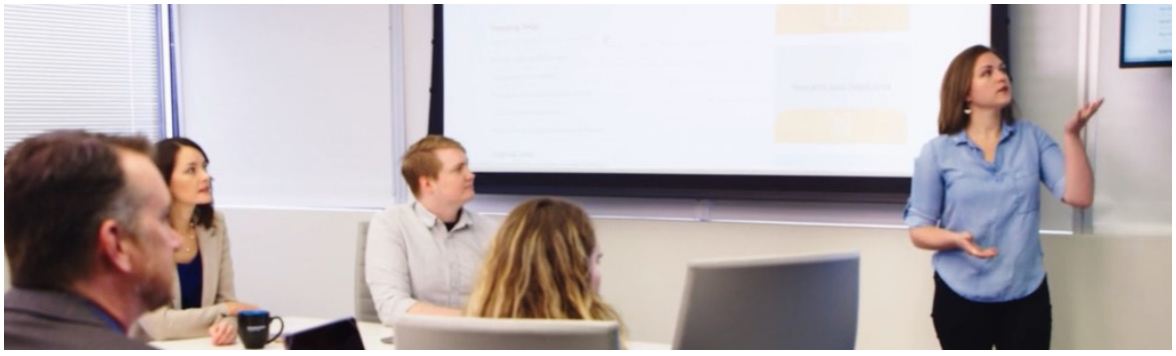
- **Timebox:** 1 to 2 hours.
- **Preparation:** Review preparation should be limited to 1 to 2 hours. Minimize presentation. Work from the repository of Stories.
- **Attendees:** If a major stakeholder cannot attend, the Product Owner should follow up individually.

Sample Iteration Review Agenda

1. Review business context and Iteration Goals
2. Demo and solicit feedback of each Story, spike, refactor, and NFR
3. Discuss Stories not completed and why
4. Identify risks, impediments
5. Revise team backlog and Team PI Objectives as needed

Two views from the Iteration Review based on a working system

- **How we did in the Iteration**
  - Did we meet the goal?
  - Story-by-Story review
- **How we are doing in the PI**
  - Review of PI Objectives
  - Review of remaining PI scope and reprioritizing if necessary



Scalable definition of done

Team Increment	System Increment	Solution Increment	Release
• Stories satisfy acceptance criteria • Acceptance tests passed (automated where practical) • Unit and component tests coded, passed, and included in the BVT • Cumulative unit tests passed • Assets are under version control • Engineering standards followed • NFRs met • No must-fix defects • Stories accepted by Product Owner	• Stories completed by all teams in the ART and integrated • Completed features meet acceptance criteria • NFRs met • No must-fix defects • Verification and validation of key scenarios • Included in build definition and deployment process • Increment demonstrated, feedback achieved • Accepted by Product Management	• Capabilities completed by all trains and meet acceptance criteria • Deployed/installed in the staging environment • NFRs met • System end-to-end integration verification, and validation done • No must-fix defects • Included in build definition and deployment/transition process • Documentation updated • Solution demonstrated, feedback achieved • Accepted by Solution Management	• All capabilities done and meet acceptance criteria • End-to-end integration and solutions V&V done • Regression testing done • NFRs met • No must-fix defects • Release documentation complete • All standards met • Approved by Solution and Release Management

4-51

Discussion: What is your definition of done?

Prepare 5 min

Share 3 min

- **Step 1:** As a group, craft a definition of what it means to you to finish a Story
- **Step 2:** Considering the criteria in the Team Increment described in the previous slide, discuss some criteria that comprise your definition of done
- **Step 3:** Be prepared to share with the class

SCALED AGILE © Scaled Agile. Inc.

4-52

4.7 Improving with the Iteration Retrospective

Iteration Retrospective

- **Timebox:** an hour or less
- **Purpose:** Pick one or two items that can be improved upon for the next Iteration
- **Outcome:** Enter the improvement items in the Team Backlog

Sample Agenda

Part 1: Quantitative

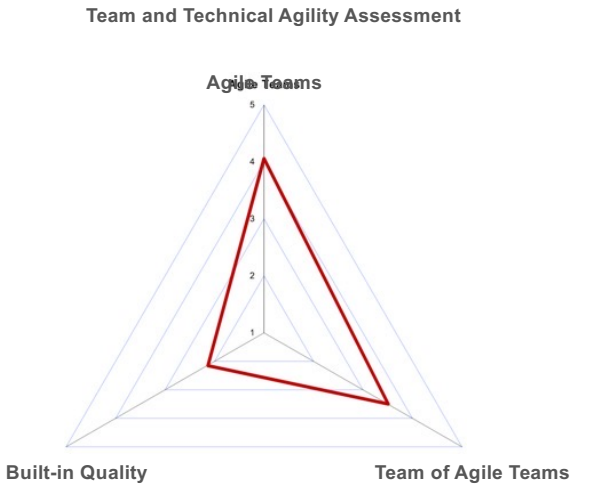
1. Review the improvement backlog items targeted for this Iteration. Were they all accomplished?
2. Did the team meet the goals (yes/no)?
3. Collect and review the agreed-to Iteration print metrics.

Part 2: Qualitative

1. What went well?
2. What didn't?
3. What can we do better next time?

Team and Technical Agility (TTA) competency assessment

- Periodically measure the progress being made toward the three TTA dimensions
- Identify specific practices for potential improvement
- Reassess periodically to observe trends



Assessments are available on the SAFE Community Platform:
https://bit.ly/Community-MeasureAndGrow

Sample Iteration Metrics

Functionality	Iteration 1	Iteration 2	Iteration 3
Velocity planned			
Velocity actual			
# Stories planned			
# Stories accepted			
% Stories accepted			
Quality			
Unit test coverage %			
# Defects			
# New test cases			
# New test cases automated			
Total tests			
Total % tests automated			
# Refactors			

Activity: Simulate an Iteration Retrospective

Prepare 8 min | Share 2 min

In your group, facilitate an Iteration retrospective of this course so far.

- **Step 1**: Pick someone in your group to play the role of the Scrum Master to facilitate the Iteration retrospective event
- **Step 2:** As a group, participate in the retrospective by discussing the following:
  - What went well?
  - What didn't go so well?
  - What can be done better?
- **Step 3:** Share some of your group's insights with the class

Action Plan: Executing the Iteration

Prepare 5 min | Share 3 min

- **Step 1:** As a team, brainstorm one to three actions you could take to improve in any areas related to this lesson
- **Step 2:** Individually write down at least one improvement item
- **Step 3:** Share one item you discussed as a team and one item you individually wrote in your Action Plan

Executing the Iteration

Lesson review

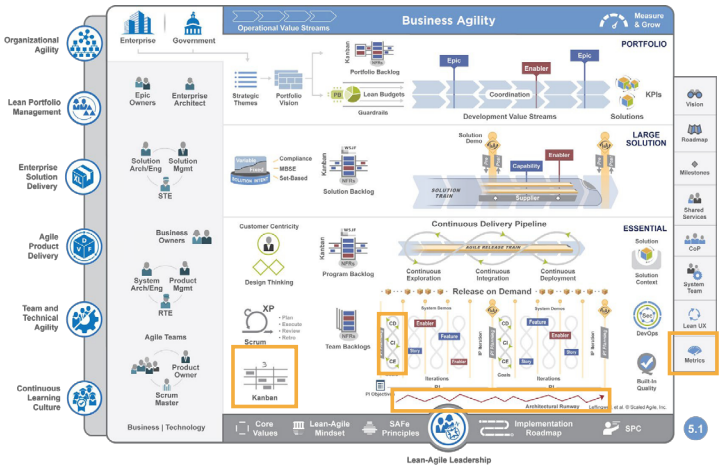
In this lesson you:

- Visualized your current flow of work
- Learned how to apply burn-up and cumulative flow diagrams for measuring flow
- Reflected on the different aspects of Built-in Quality
- Recognized the importance of a continuous delivery pipeline
- Experienced the communication and synchronization benefits of the daily stand-up
- Explained how to run a successful backlog refinement session
- Discussed the purpose of the Iteration Review and created a definition of done
- Participated in a retrospective and identified improvements

Articles used in this lesson

Read these Framework articles to learn more about topics covered in this lesson

- "Kanban" https://v5.scaledagileframework.com/team-kanban/
- "Metrics" https://v5.scaledagileframework.com/metrics/
- "Architectural Runway" https://v5.scaledagileframework.com/architectural-runway/
- "Continuous Integration" https://v5.scaledagileframework.com/continuous-integration/



Continue your SAFe journey with the following resources:

Watch this three-part video series, *Kanban for Teams*, to walk through the processes of setting up, implementing, and incrementally improving a team Kanban. https://bit.ly/Playlist-KanbanTeams	Use the SAFe Collaborate template, "Determine the Team's Definition of Done," with your team to agree upon guardrails, rules, and processes to ensure built-in quality for your solutions. https://bit.ly/Template-DetermineDoD
Use the SAFe Collaborate template "Experience Batch Sizes" with your team to explore the impact of reducing batch sizes on overall throughput. https://bit.ly/Template-BatchSize	Watch this sixty-minute video, *Community Webinar: DevSecOps in Real Life,* to delve into the most important practices that fuel the Continuous Delivery Pipeline and how they are implemented in real life. https://bit.ly/Community-DevSecOpsWebinar
Watch this five-part video series, *Running Effective Iteration Events,* to explore each of the key team-level events in SAFe. https://bit.ly/Playlist-EffectiveIterationEvents	Use the SAFe Collaborate template, "Retrospective Plus, Minus, Delta," to run an effective retrospective with your team. https://bit.ly/Template-PlusMinusDelta

4-61

Lesson notes

Enter your notes below. If using a digital workbook, save your PDF often so you don't lose any of your notes.

Lesson 5

Executing the Program Increment (PI)

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

SP SAFe® 5

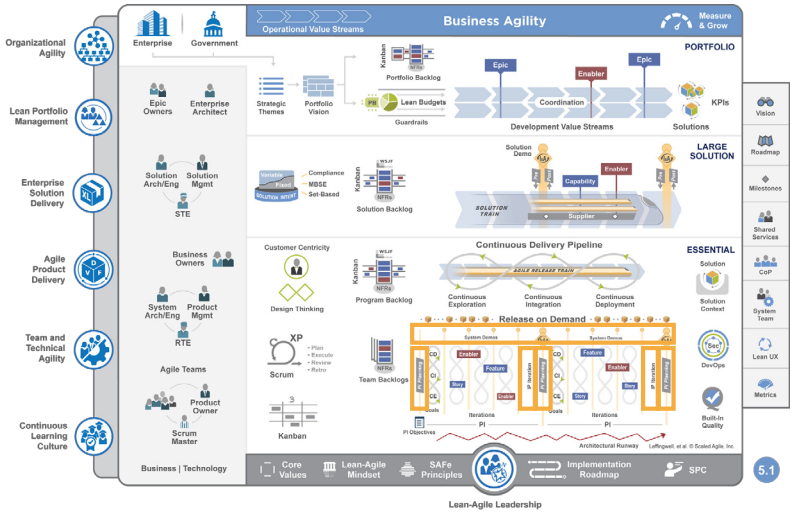
SCALED AGILE®

Lesson Topics

5.1 PI Planning

5.2 ART Sync and System Demo

5.3 Innovation and Planning (IP) Iteration and the Inspect & Adapt event



5-2

Learning objectives

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

- Explain the benefits of planning together as a whole ART
- Participate in a PI Planning event
- Recognize the importance of integrating and demonstrating together with the System Demo
- Recognize the need for the IP Iteration to provide time and space for the ART to learn together
- Explain how to improve as an ART through the Inspect and Adapt event

5.1 PI Planning

Video: Introduction to PI Planning

Duration 3 min

https://bit.ly/Video-PIPlanningOverview

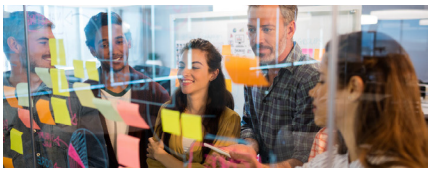
5-5

What is PI Planning?

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

- Two days every 8 – 12 weeks (10 weeks is typical)
- Everyone plans together
- Product Management owns Feature priorities
- Development teams own Story planning and high-level estimates
- Architect/Engineering and UX work as intermediaries for governance, interfaces, and dependencies

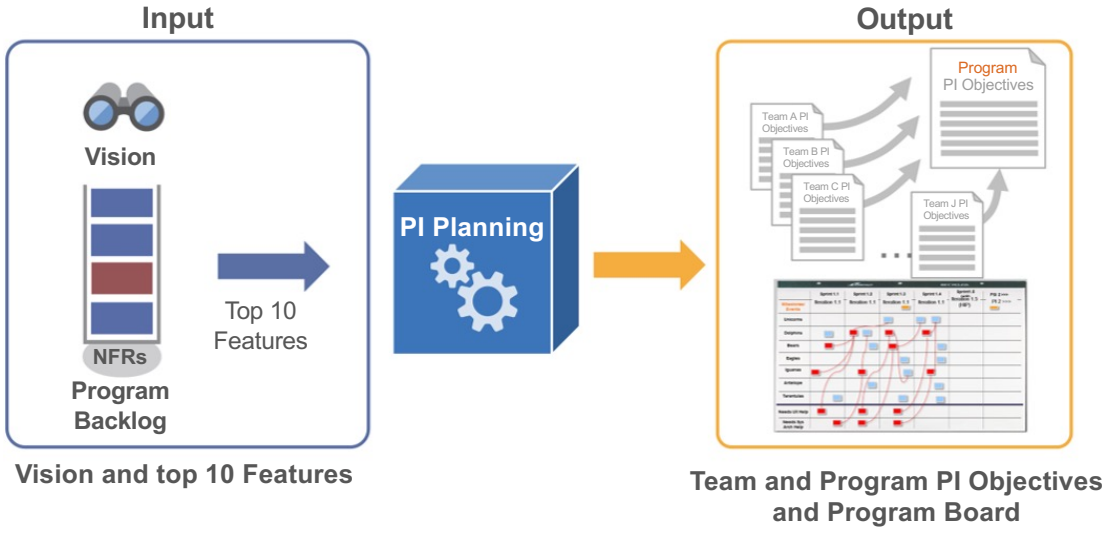
PI Planning



Agile Team

5-6

The PI Planning process



Create Alignment with PI Objectives

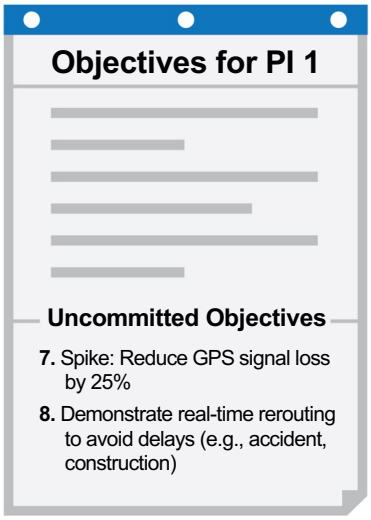
- Objectives are business summaries of what each team intends to deliver in the upcoming PI.
- They often directly relate to intended Features in the backlog.
- Other examples:
  - Aggregation of a set of Features
  - A Milestone like a trade show
  - An Enabler Feature supporting the implementation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

Objectives for PI 1	BV	AV
1. Show routing calculations between the 5 most frequent destinations		
2. Navigate autonomously from distribution center to the most frequent destination		
3. Parallel park for a delivery		
4. Return to the distribution center after delivery		
5. Include traffic data in route planning		
6. Recall a delivery that is already in progress		
Uncommitted Objectives		
7. Spike: Reduce GPS signal loss by 25%		
8. Demonstrate real-time rerouting to avoid delays (e.g., accident, construction)		

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, it should be moved to uncommitted
- If an objective has many unknowns, consider moving it to uncommitted and put in early spikes
- Uncommitted objectives count when calculating load



SMART team PI Objectives

- **S**pecific - States the intended outcome as simply, concisely, and explicitly as possible (Hint: Try starting with an action verb).
- **M**easurable - It should be clear what a team needs to do to achieve the objective. The measures may be descriptive, yes/no, quantitative, or provide a range.
- **A**chievable - Achieving the objective should be within the team's control and influence
- **R**ealistic - Recognize factors that cannot be controlled.(Hint: Avoid making overly optimistic assumptions)
- **T**ime-bound - The time period for achievement must be within the PI, and, therefore, all objectives must be scoped appropriately.

Activity: Identify ART roles

Duration 3 min

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

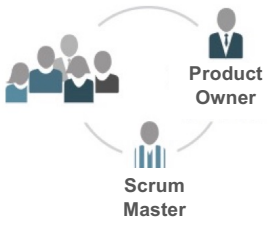
Simulation role	Assigned to
Executive	Volunteer
Product Manager	Volunteer
System Architect, UX, Development Manager	Volunteer

Note: Your Instructor will play the role of the RTE.

Activity: Identify team names and roles

Duration 3 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's name and the names of the people selected are visible to all other teams.



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.

Simulation: Day 1 agenda

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

Simulation: Day 2 agenda

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

Simulation: Briefings

Executive

Product Manager

System Architect

Simulation: Planning guidance

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

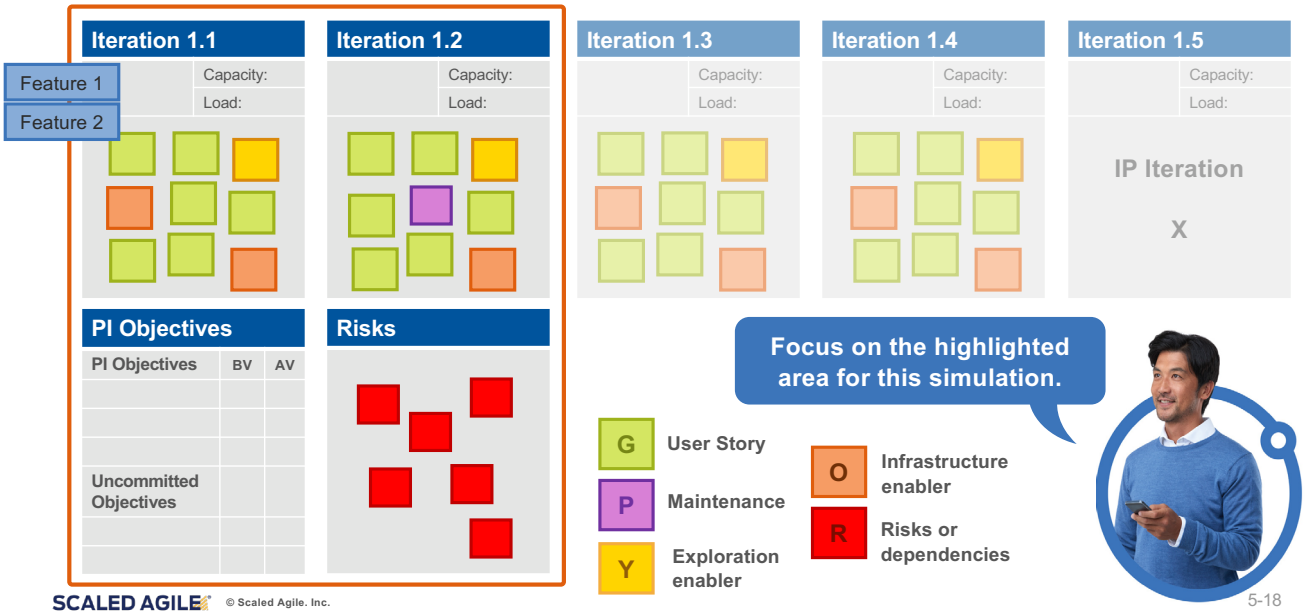
Product Owners: You have the content authority to make decisions at the user Story level

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

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

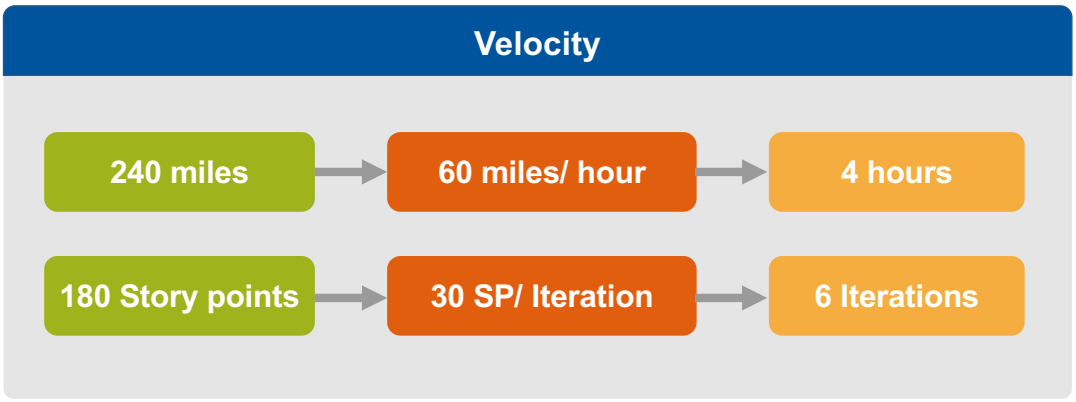
5-17

Simulation: Planning requirements



5-18

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 x 8 points = 40 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 group's capacity calculated

Activity: Team breakout #1

Duration: 50 min

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

- **Step 1:** Setup the workspace. 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)

Duration: 5 min

- **Step 1:** Observe the SoS, conducted by the RTE
- **Step 2:** Each Scrum Master provides the 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)

Scrum of Scrums questions are on the following slide.

5-23

Activity: Scrum of Scrums (SoS)

Duration: 5 min

SoS Sync Questions	Team 1	Team 2	Team 3	Team 4	Team 5
Have you identified the capacity for each Iteration of 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 tradeoffs 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.					

5-24

Activity: Draft plan review

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

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

5-27

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

SCALED AGILE © Scaled Agile. Inc.

5-28

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

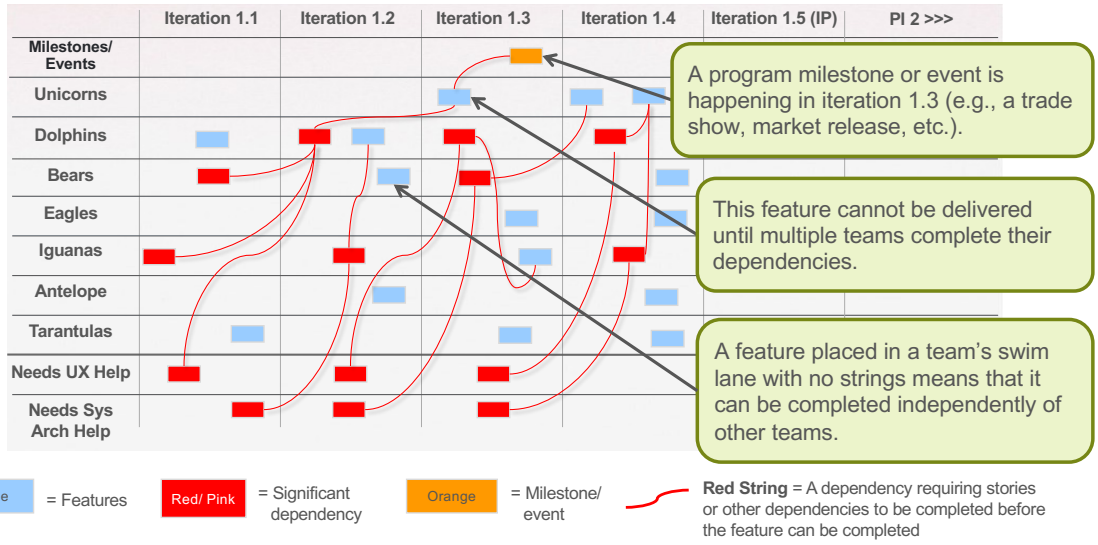
Objectives for PI 1	BV	AV
1. Show routing calculations between the 5 most frequent destinations	10	
2. Navigate autonomously from distribution center to the most frequent destination	8	
3. Parallel park for a delivery	7	
4. Return to the distribution center after delivery	10	
5. Include traffic data in route planning	7	
6. Recall a delivery that is already in progress	7	
Uncommitted Objectives		
7. Spike: Reduce GPS signal loss by 25%	2	
8. Demonstrate real-time rerouting to avoid delays (e.g., accident, construction)	5	

Video: Program Board

Duration: 7 min

https://bit.ly/Video-ProgramBoard

Program board: Feature delivery, dependencies, and Milestones



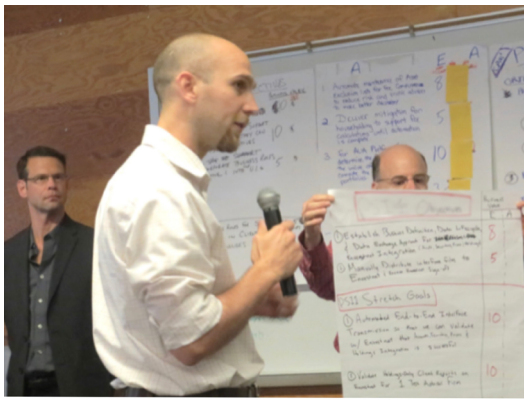
5-31

Final plan review

Teams and Business Owners peer-review all final plans.

Final plan review agenda

1. Changes to capacity and load
2. Final PI Objectives with business value
3. Program risks and impediments
4. Q&A session



Final plan review

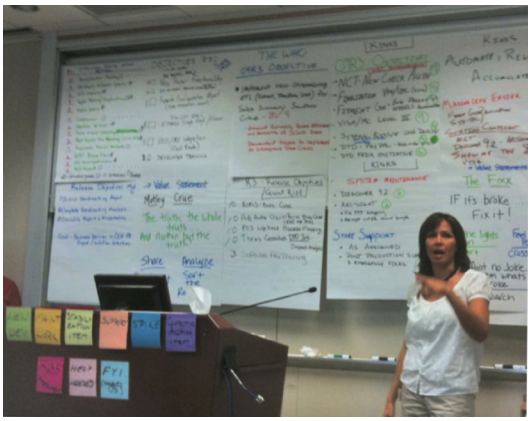
Used with permission of SEI Global Wealth Services

SCALED AGILE © Scaled Agile, Inc.

5-32

Building the final plan

- Final plans are reviewed by all teams
- Business Owners are asked whether they accept the plan
- If so, the plan is accepted
- If not, the plans stay in place, and the team continues planning after the review



A team presenting their final plan

Used with permission of Discount Tire Corporation

5-33

Addressing program risks

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

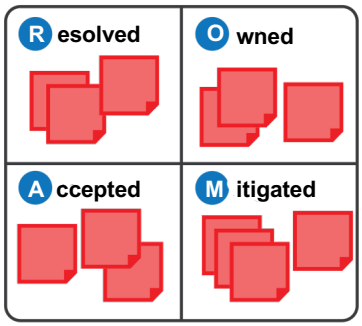
ROAMing risks:

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.



5-34

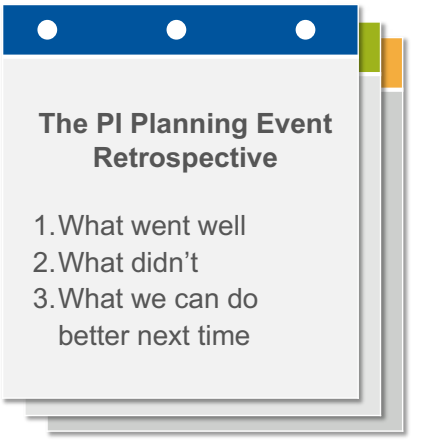
Confidence vote: Team and program

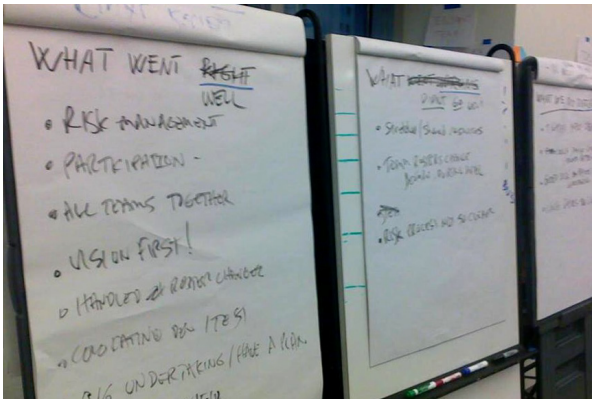
After dependencies are resolved and risks are addressed, a confidence vote is taken by the team and program.

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

Run a planning meeting retrospective

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

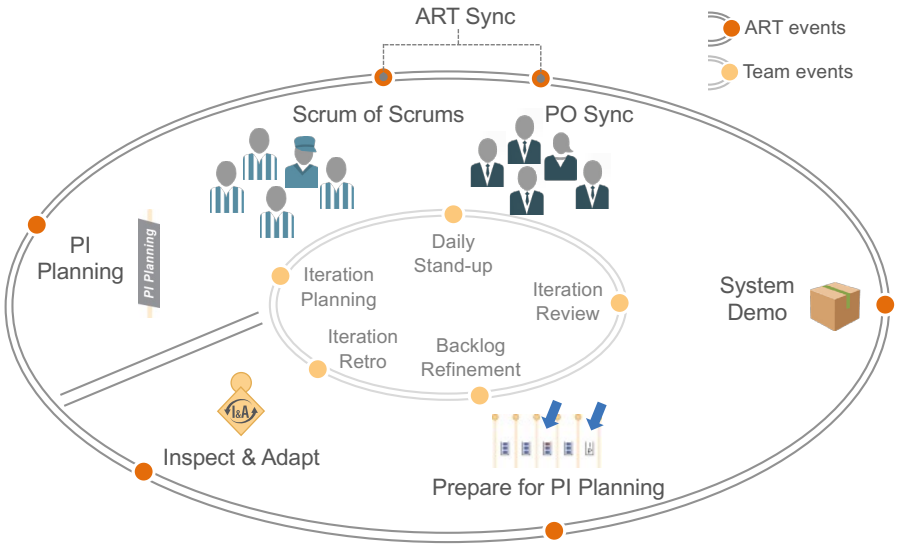




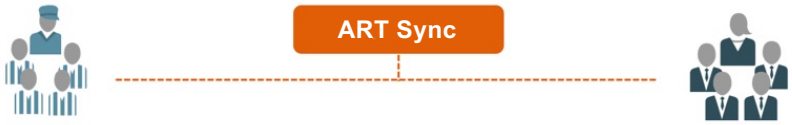
5.2 ART Sync and System Demo

ART events drive the train

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



ART Sync is used to coordinate progress



Scrum of Scrums

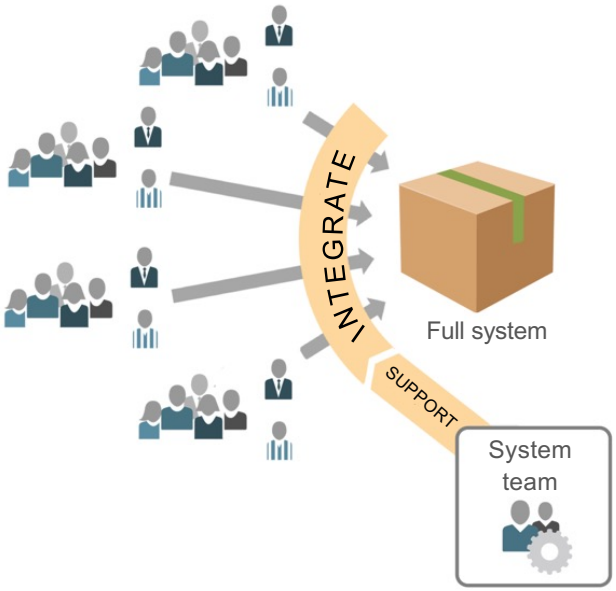
- Visibility into progress and impediments
- Facilitated by RTE
- Participants: Scrum Masters, other select team members, SMEs if necessary
- Weekly or more frequently, 30-60 minutes
- Timeboxed and followed by a 'Meet After'

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

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



Discussion: System Demo challenges

Prepare 3 min

Share 2 min

- **Step 1:** As a team, discuss the following:
  - What are challenges to having a new system increment every two weeks?
- **Step 2:** List three to five challenges and some ways to solve them
- **Step 3:** Be prepared to share with the class

5.3 Innovation and Planning (IP) Iteration and the Inspect & Adapt event

Innovation and Planning (IP) Iteration

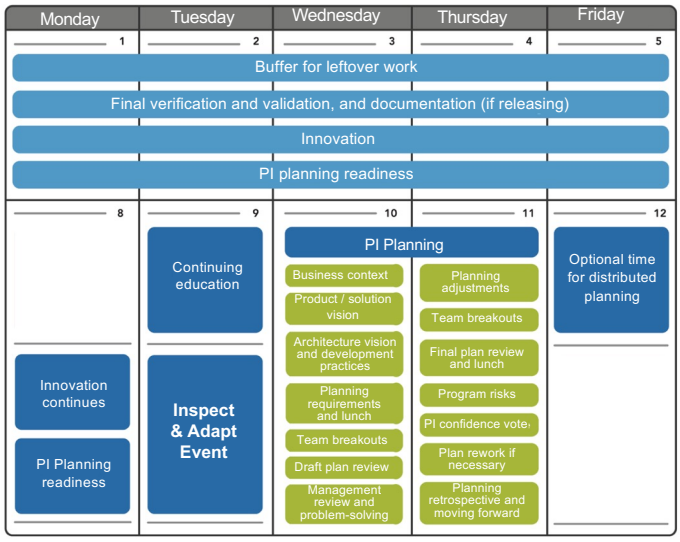
Provide sufficient capacity margin to enable cadence. —Donald G. Reinertsen

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

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

Example IP Iteration calendar

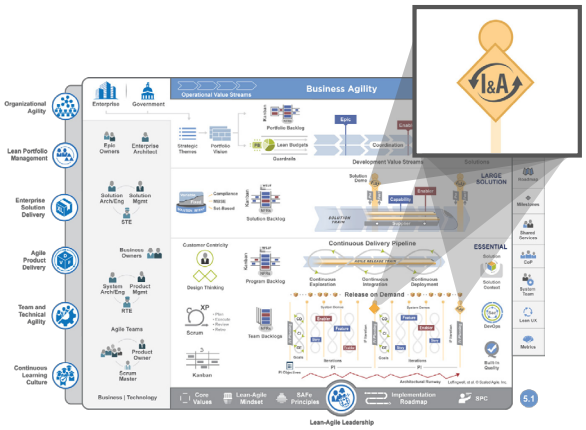


Improving results with the Inspect and Adapt event

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



1. 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, ART stakeholders, Product Management, RTE, Scrum Masters, and teams

Program performance reporting

Prior to or as part of the PI System Demo, teams review the business value achieved for each of their PI Objectives.

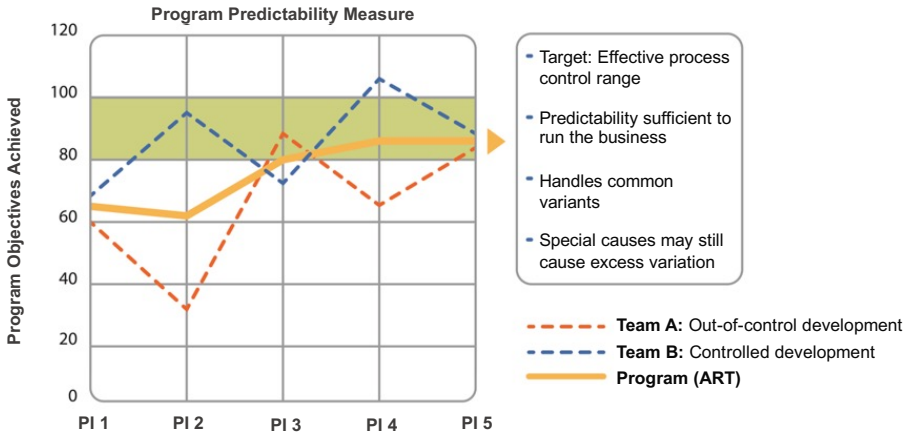
- Teams meet with their Business Owners to self-assess the business value they achieved for each objective
- Each team's planned vs actual business value is then rolled up to the program predictability measure.

Objectives for PI 3	Business Value: Plan	Business Value: 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%		

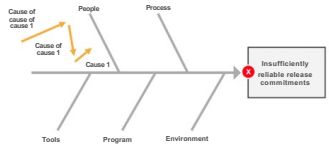
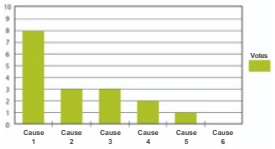
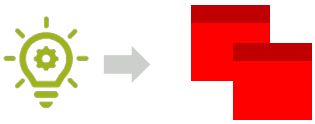
Measure ART Predictability

The Program Predictability measure compares actual business value achieved to planned business value.

Flow



3. The Problem-Solving Workshop

Agree on the problem to solve	Apply root-cause analysis and 5 whys	Identify the biggest root-cause using Pareto analysis
X Insufficiently reliable release commitments		
Restate the new problem for the biggest root-cause	**Brainstorm solutions**	**Identify improvement backlog items**
X Insufficient architectural runway		

5-49

Action Plan: Executing the PI

Prepare 5 min

Share 3 min

- **Step 1:** As a team, brainstorm one to three actions you could take to improve in any areas related to this lesson
- **Step 2:** Individually write down at least one improvement item
- **Step 3:** Share one item you discussed as a team and one item you individually wrote in your Action Plan

5-50

Executing the PI

Lesson review

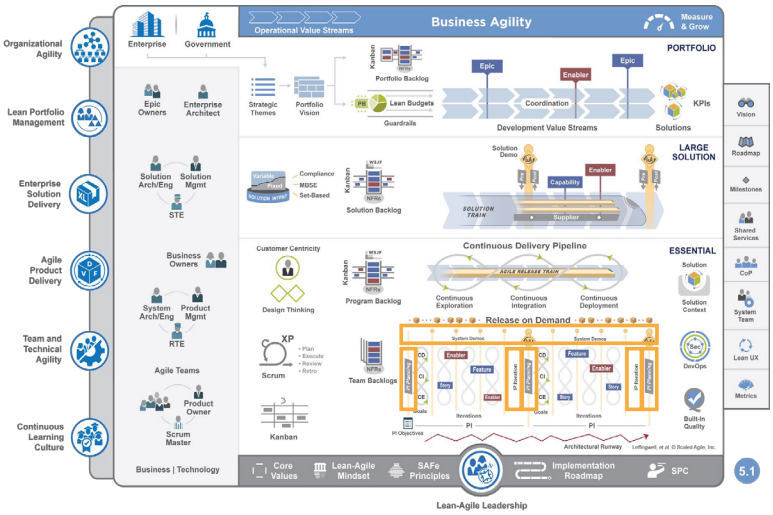
In this lesson you:

- Explored the benefits of planning together as a whole ART
- Participated in a PI Planning event
- Recognized the importance of integrating and demonstrating together with the System Demo
- Recognized the need for the IP Iteration to provide time and space for the ART to learn together
- Explored how to improve as an ART through the Inspect and Adapt event

Articles used in this lesson

Read these Framework articles to learn more about topics covered in this lesson

- "PI Objectives" https://v5.scaledagileframework.com/pi-objectives/
- "Innovation and Planning Iteration" https://v5.scaledagileframework.com/innovation-and-planning-iteration/
- "Inspect and Adapt" https://v5.scaledagileframework.com/inspect-and-adapt/
- "System Demo" https://v5.scaledagileframework.com/system-demo/



Continue your SAFe journey with the following resources:

Watch this quick one-minute video, *SAFe Developer Stories: My First PI Planning*, to hear about Elizabeth Flournoy's first PI Planning experience. https://bit.ly/Video-FirstPIPlanning	Download the "SAFe PI Planning Toolkit 5.1" for resources to support preparation, coordination, and communication to guide an ART through its PI Planning event. https://bit.ly/Community-ToolkitsandTemplates
Watch this six-minute video, *Preparation for PI Planning*, to learn about three areas an ART should focus on when preparing for PI Planning. https://bit.ly/Video-PIPlanningPrep	Download and use the resources found in the "SAFe PI Execution Toolkit" to support an ART in successful PI Execution. https://bit.ly/Community-ToolkitsandTemplates
Watch and share the videos in this playlist, "Agile Software Engineering Vlog" to learn how to move from software engineering to Agile software engineering. https://bit.ly/Playlist-AgileSoftwareEngineering	

5-53

Lesson notes

Enter your notes below. If using a digital workbook, save your PDF often so you don't lose any of your notes.

Lesson 6

Practicing SAFe

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

SCALED AGILE®

Video: SAFe Certification Benefits

Duration 3 min

https://bit.ly/Video-SAFeCertificationBenefits

SCALED AGILE®

6-2

A Path Towards Certification

Access exam study guides and practice tests

Download your certificate of course completion

Take the **Certification Exam**

Showcase your **Digital Badge** and get recognized as Certified SAFe Professional

Becoming a Certified SAFe Professional

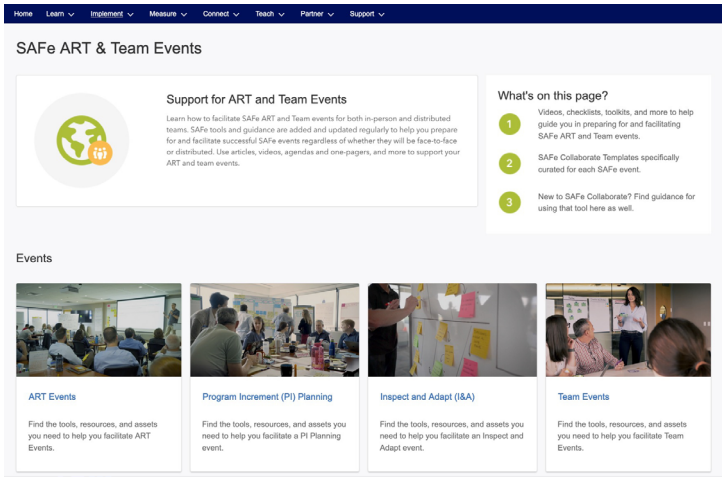
Video: Welcome to the SAFe Community Platform

Duration 3 min

https://bit.ly/Video-WelcomeSAFeCommunityPlatform

SAFe ART and Team Events

SAFe ART and Team Events: Use checklists, guides, templates, videos, and more to support your ART and Team Events



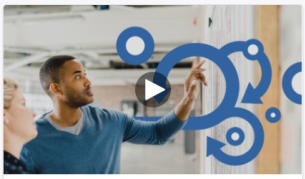
SAFe Toolkits

Explore ready-to-use templates and job resources to help execute SAFe events and workshops more effectively

- SAFe® Remote ARTs Toolkit
- SAFe® Measure and Grow Workshop Toolkit
- SAFe® Iteration Execution Toolkit
- SAFe® PI Planning Toolkit
- SAFe® PI Execution Toolkit
- Team Formation Toolkit

E-learning Resources

Discover and develop skills through self-paced, interactive e-learning modules to achieve your personal and professional goals

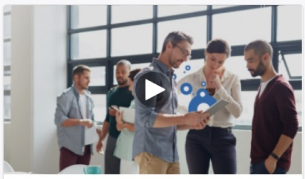


Agile Basics

E-learning

Learn what Agile is, where it comes from, why it continues to be used and needed, and how it supports teams and organizations to do what they do better.

30 - 45 Minutes

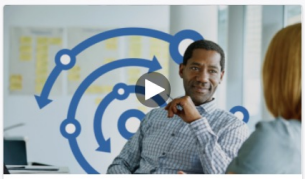


What is SAFe for Lean Enterprises

E-learning

Become more familiar with the goals and methods of SAFe to achieve Business Agility.

15 - 30 Minutes



SAFe Foundations: Core Values

E-learning

Build your understanding of the core values of SAFe and how they are applied in practice.

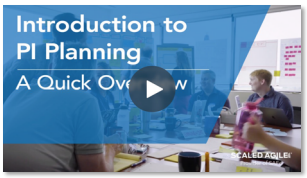
15 - 30 Minutes

SCALED AGILE © Scaled Agile, Inc.

6-7

Community Video Hub

Access videos to support your learning and grow your skills



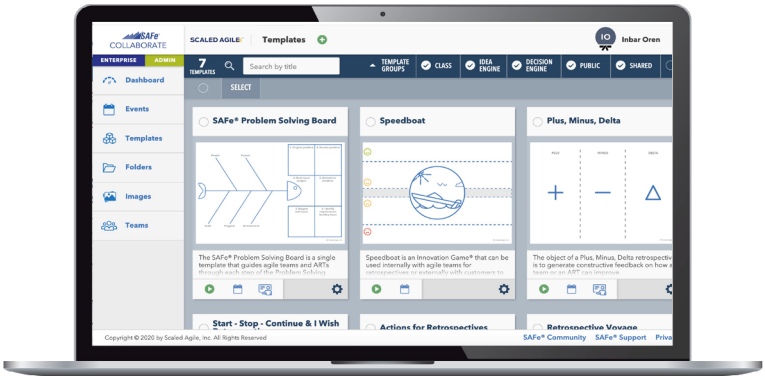
SCALED AGILE © Scaled Agile, Inc.

6-8

SAFe Collaborate

Organize and run virtual SAFe events in real-time

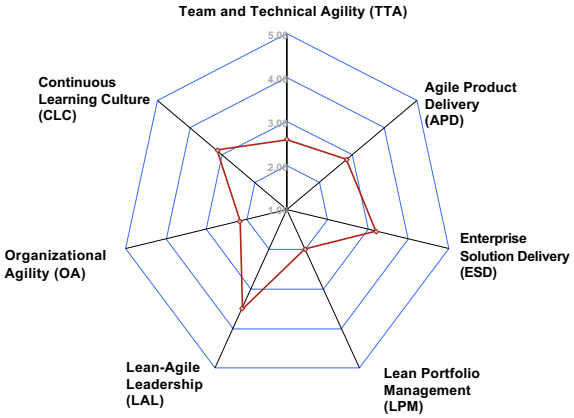
SAFe Collaborate is a visual, cloud-based workspace where organizations can orchestrate virtual SAFe events activities easily and effectively with predesigned and customizable templates.



Measure and Grow

Evaluate progress towards business agility with the SAFe assessments, Measure and Grow workshop and our assessment partners

Business Agility Assessment



agilityhealth® enabling business agility

comparative agility

Team and Technical Agility Assessment

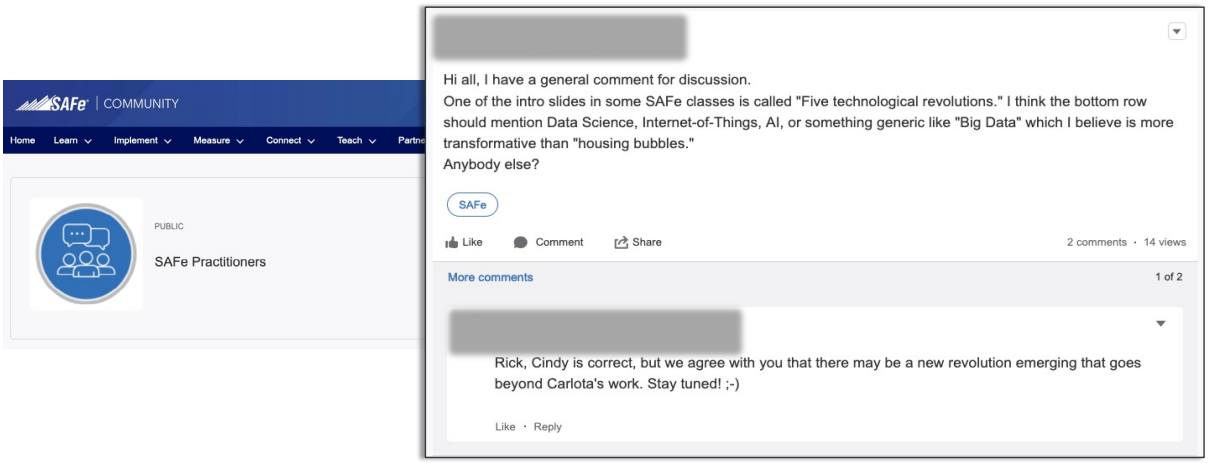
Competency-based self-assessments

Each core competency has a downloadable assessment, along with three Growth recommendations, available in SAFe Collaborate. To access these assets directly, visit the SAFe Community Platform Measure and Grow page: https://bit.ly/Community-MeasureAndGrow

- Organizational Agility
- Lean Portfolio Management
- Enterprise Solution Delivery
- Agile Product Delivery
- **Team and Technical Agility**
- Continuous Learning Culture
- Lean-Agile Leadership

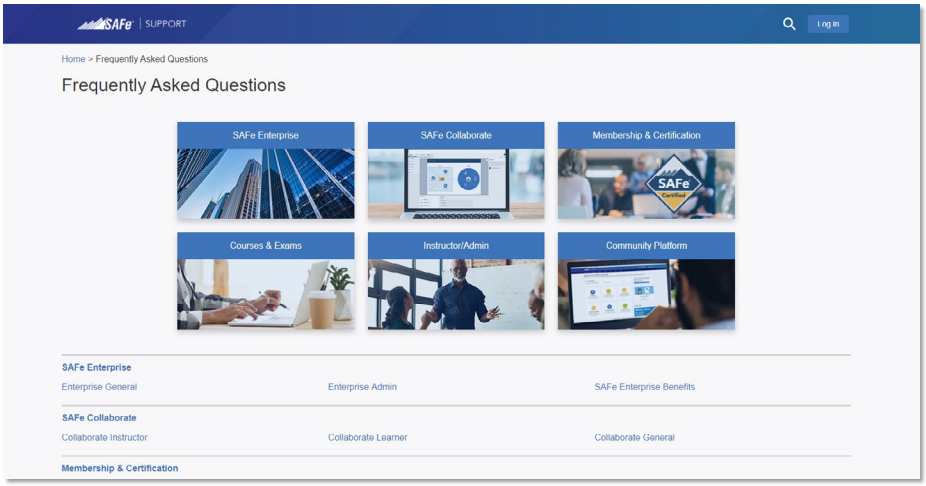
SAFe Forums

Join the SAFe Practitioners Community Forum to connect with a community of SAFe Practitioners



SAFe FAQs

When you need support, check the FAQ page for your question or contact SAI support directly.



6-13

Good luck on your SAFe Practice with the

SAFe Community Platform!

https://community.scaledagile.com/

SAFe® COMMUNITY

6-14

SAFe Glossary

SAFe Glossary: Visit the Scaled Agile Framework site (*v5.scaledagileframework.com/glossary/*) to download glossaries translated into other languages.