

Lesson 5

Executing the Program Increment

1. Applying SAFe in the Lean Enterprise
2. Relating a Lean-Agile Mindset to the PO/PM Roles
3. Collaborating with Lean Portfolio Management
4. Continuously Explore Customer Needs
5. Executing the Program Increment
6. Defining the PO/PM Roles and Responsibilities
7. Creating your PO/PM Action Plan

SAFe® Course: Attending this course gives learners access to the SAFe Product Owner/Product Manager exam and related preparation materials.

Learning objectives

- 5.1 Create alignment with PI Planning
- 5.2 Decompose Features into Stories
- 5.3 Plan the Iteration
- 5.4 Execute the PI
- 5.5 Release on Demand

5.1 Create alignment with PI Planning

Participate in PI planning

The Product Manager and Product Owners play key roles before, during, and after the PI Planning event.



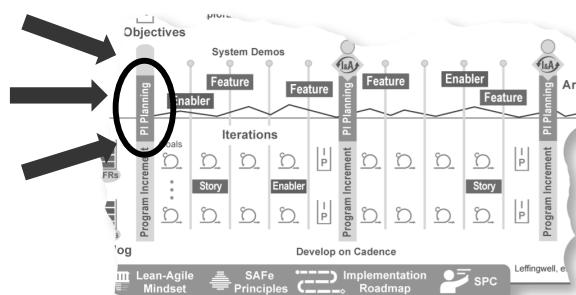


Schrodors PI Planning Event Kickoff

<https://vimeo.com/169066536>

Inputs of PI Planning

- Inputs of PI Planning include:
- ▶ Business context
 - ▶ Roadmap and Vision
 - ▶ Top 10 Features from the single ART Program Backlog

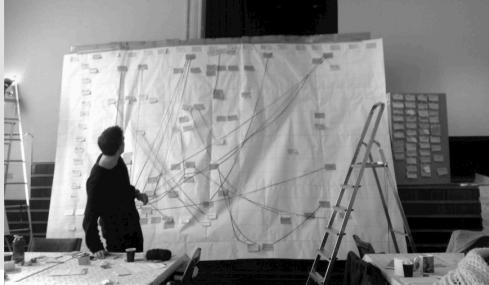


Successful PI Planning Outputs

A successful PI planning event delivers two primary outputs:

1. Committed PI Objectives (SMART)
2. Program Board

<u>Objectives for PI 1</u>	<u>Business Value</u>
<ul style="list-style-type: none">› Structured location and validation of locations› Build and demonstrate a proof of concept for context images› Implement negative triangulation by: tags, companies, and people› Speed up indexing by 50%› Index 1.2 B more web pages› Extract and build URL abstracts	
<u>Stretch Objectives for PI 1</u>	
<ul style="list-style-type: none">› Fuzzy search by full name› Improve tag quality to 80% relevance	



5.7

PI Planning preparation

Prior to the PI Planning event, the PM should collaborate with POs to:



Create/update the Vision.



Prepare and estimate the top 10 Features list from the Program Backlog.



Socialize the above artifacts with business, stakeholders, and architects to validate the Feature list and set expectations for the PI Planning meeting.

What POs and PMs do during PI Planning – Day 1

- ▶ Communicate:
 - Program Vision
 - Present the top 10 Features
- ▶ Collaborate to decompose Features into Stories
- ▶ Negotiate scope
- ▶ Review draft PI plans and provide feedback
- ▶ Participate in management review of draft PI plans



You are here! Day 1 agenda

8:00-9:00	Business context		State of the business and upcoming objectives
9:00-10:30	Product/Solution Vision		Vision and prioritized Features
10:30-11:30	Architecture Vision and development practices		▶ Architecture, common frameworks, etc. ▶ Agile tooling, engineering practices, etc.
11:30-1:00	Planning context and lunch		Facilitator explains planning process
1:00-4:00	Team breakouts SoS	1 2 3 4	▶ Teams develop draft plans and identify risks and impediments ▶ Break Features into Stories ▶ Architects and Product Managers circulate
4:00-5:00	Draft plan review		Teams present draft plans, risks, and impediments
5:00-6:00	Management review and problem solving		Adjustments made based on challenges, risks, and impediments

Management review and problem-solving

At Day 1 end, 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? Resources?
- ▶ Where are the bottlenecks?
- ▶ What Features must be de-scoped?
- ▶ What decisions must we make between now and tomorrow to address these issues?



Used with permission of Hybris Software

Exercise: Problem Solving Workshop - PO and PM roles



Think about the Management Review and Problem-Solving Workshop.

- ▶ What activities are required for the PO?
- ▶ What activities are required for the PM?
- ▶ Capture ideas and share with the class



What POs and PMs do during PI Planning – Day 2

- ▶ Participate in final PI plan review
- ▶ Establish business value with Business Owners
- ▶ Accept Team Objectives
- ▶ Provide feedback on program risks
- ▶ Participate in confidence vote, rework (if applicable), and planning retrospective



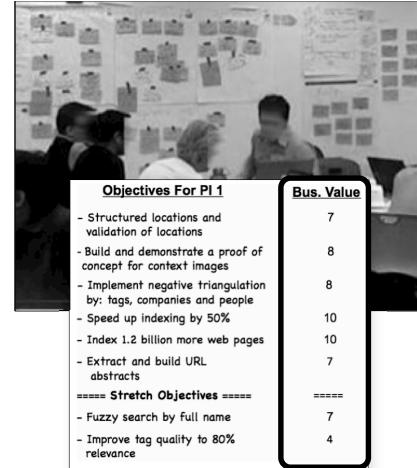
Day 2 agenda

8:00-9:00	Planning adjustments		Planning adjustments made based on previous day's management meeting
9:00-11:00	Team breakouts	1 2 3 4	<ul style="list-style-type: none">▶ Teams develop final plans and refine risks and impediments▶ Business Owners circulate and assign business value to team objectives
11:00-1:00	Final plan review and lunch		Teams present final plans, risks, and impediments
1:00-2:00	Program risks		Remaining program-level risks are discussed and ROAMED
2:00-2:15	PI confidence vote		Team and program confidence vote
2:15-???	Plan rework if necessary	1 2 3 4	If necessary, planning continues until commitment is achieved
After commitment	Planning retrospective and moving forward		<ul style="list-style-type: none">▶ Retrospective▶ Moving Forward▶ Final Instructions

Supporting the Team Objectives

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
- ▶ Stretch objectives provide the capacity and guard band needed to increase cadence-based delivery reliability



5.15

Exercise: Supporting Team Objectives with Business Value



- ▶ Read the Guidance Article on "The Role of PI Objectives" (page 116 in your workbook)
- ▶ In your workbook, reason about how the POs and PMs will support the assignment of Business Value
- ▶ Be prepared to share your results

Objectives For PI 1		Bus. Value
- Structured locations and validation of locations		7
- Build and demonstrate a proof of concept for context images		8
- Implement negative triangulation by: tags, companies and people		8
- Speed up indexing by 50%		10
- Index 1.2 billion more web pages		10
- Extract and build URL abstracts		7
===== Stretch Objectives =====		=====
- Fuzzy search by full name		7
- Improve tag quality to 80% relevance		4



5.16

Confidence vote: Team and Program Levels

After dependencies are resolved and risks are addressed, a confidence vote is taken at the Team and Program Levels.

'Fist of five' confidence vote

- ▶ Range of 1-5
- ▶ 1 = No confidence
- ▶ 5 = Very high confidence

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



SCALED AGILE® © Scaled Agile, Inc.

Used with permission of Standard Bank

5.17

Exercise: PO/PM day 1 and 2 PI Planning activities



Refer to your workbook as needed and summarize the things you need to be doing as a PO/PM:

- ▶ Before PI Planning in preparation
- ▶ During PI Planning on day 1
- ▶ During PI Planning on day 2



SCALED AGILE® © Scaled Agile, Inc.

5.18

What POs and PMs do after PI Planning



Update the Roadmap



Continue to collaborate with each other



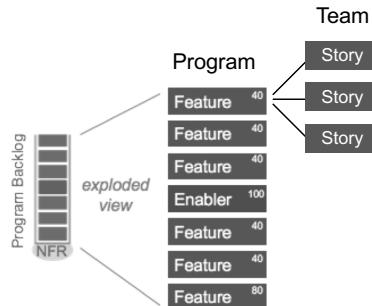
Prepare for the next PI!

5.2 Decompose Features into Stories

Feature decomposition

Feature breakdown into User Stories happens prior to, during, and after the PI Planning event.

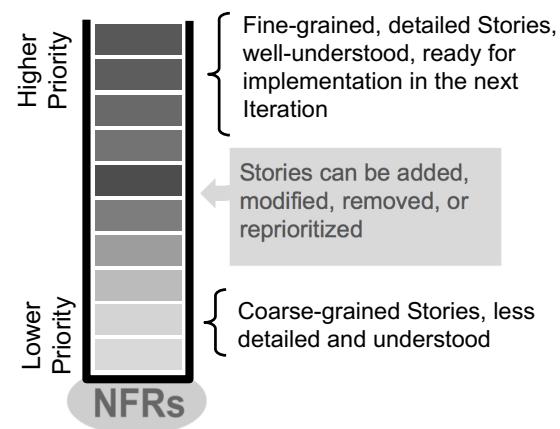
Before, During, and After



Features drive User Stories

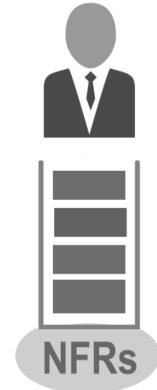
User Stories (from XP) replace traditional requirements expressions.

- ▶ The Team Backlog consists of backlog items, many of which are elaborated as User Stories that express the needs of the stakeholders
- ▶ User Stories are not requirements. They are negotiable expressions of intent.

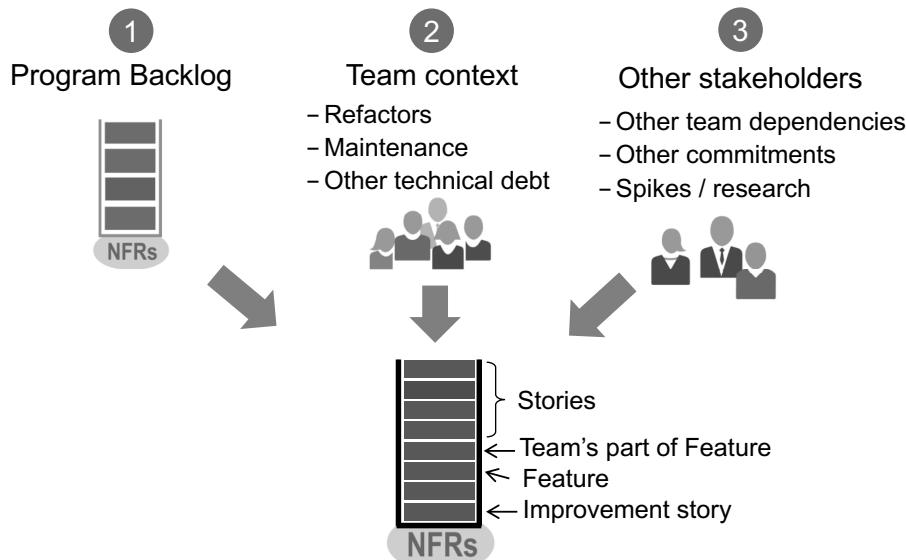


The Team Backlog

- ▶ Contains all the work the team needs to work on
- ▶ Contains User and Enabler Stories
 - User Stories provide customers with value
 - Enabler Stories build the infrastructure and architectures that makes User Stories possible
- ▶ Stories in the backlog are prioritized
- ▶ Stories for the next Iteration are more detailed than Stories for later Iterations
- ▶ Non-Functional Requirements (NFRs) are a constraint on the Backlog



Three primary sources of Team Backlog items



Backlog Item	Size
1. Login portal	5
2. Display minutes	3
3. Support ticket call	8
4. Remote login help	13
5. Update profile	2
6. Buy more time	8
7. Single sign-on	20
8. Log rotation	5
9. Timer display	3
10. Automatic logout	1
11. Usage warning	2
12. 300 logins/min	13
13. Location tracking	5
17. Intrusion detection	13
18. Update MySQL DB	20
19. Update Web Stack	13
20. Update Linux Kernel	8
21. Support Novel Auth	13
22. Support RADIUS Auth	8
23. Scan & Block Interface	40
24. AP Manager Interface	20

A Team Backlog

Backlogs bring stakeholder ideas together

Where Good Ideas Come From



<https://youtu.be/Mb0ssmoXG1I>

User Stories

- ▶ Containers for user or customer value
- ▶ Written using the following template:

As a <user role>, I want <activity> so that <business value>

- **User role** is the description of the person doing the action
- **Activity** is what they can do with the system
- **Business value** is why they want to do the activity

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 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 can save on paper.

User Story guidelines — The 3 Cs

Card	Conversation	Confirmation
<p>Written on a card or in the tool and may annotate with notes</p> <p>As a spouse, I want a clean garage so that I can park my car and not trip on my way to the door</p>	<p>The details are in a conversation with the Product Owner</p> <p>What about the bikes?</p> <p>Oh yeah, hang the bikes</p>	<p>Acceptance criteria confirm the Story correctness</p> <ul style="list-style-type: none">▶ Tools have been put away▶ Items on the floor have been returned to the proper shelf▶ Bikes have been hung

Source: 3 Cs coined by Ron Jeffries

5.27

INVEST in a good Story

I	Independent
N	Negotiable
V	Valuable
E	Estimable
S	Small
T	Testable

Enabler Stories

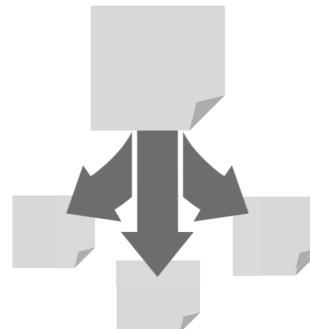
Enabler Stories build the groundwork for future User Stories.

There are generally four types of Enabler Stories:

1. Infrastructure – Build development and testing frameworks that enable a faster and more efficient development process
2. Architecture – Build the Architectural Runway, which enables smoother and faster development
3. Exploration – Build understanding of what the customer needs to understand prospective Solutions and evaluate alternatives
4. Compliance - Compliance enablers are used to schedule and manage specific compliance activities, including Verification and Validation (V&V), documentation and signoffs, and regulatory submissions and approvals.

10 patterns for breaking Features into Stories

- | | | | |
|---|--------------------------|----|------------------------|
| 1 | Work flow steps | 6 | Data methods |
| 2 | Business rule variations | 7 | Defer system qualities |
| 3 | Major effort | 8 | Operations |
| 4 | Simple/complex | 9 | Use case scenarios |
| 5 | Variations in data | 10 | Break out a spike |



Exercise: Break Features into Stories



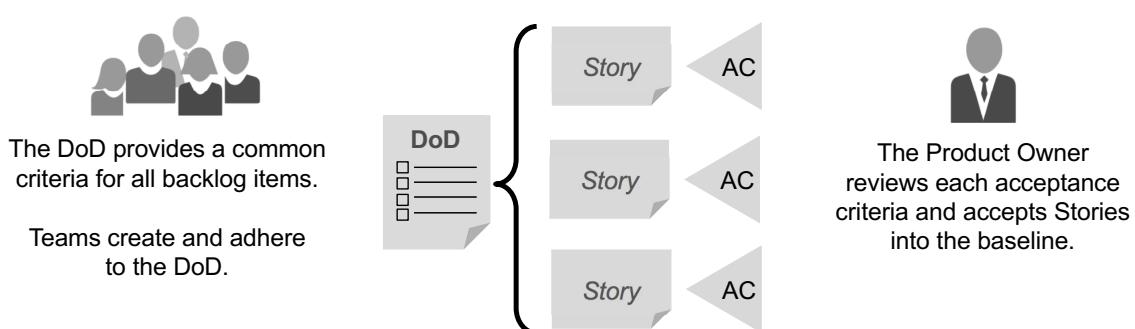
Using the information in your workbook (pages 127-131), break your Features into Stories that are small enough to fit into an Iteration.

- ▶ Break a Feature into at least five Stories
- ▶ Try to create Stories that are less than a week in size
- ▶ Identify spikes as needed
- ▶ You should have at least five Stories from your Feature
- ▶ See your workbook for examples



Accepting User Stories

A Story is accepted when it satisfies the Definition of Done (DoD) and is accepted by the Product Owner.



Acceptance criteria

- ▶ Acceptance criteria provide the details of the Story from a testing point of view
- ▶ Acceptance criteria are created by the team and the PO

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

Acceptance criteria:

1. The fueling process stops automatically on the exact value
2. I can stop fueling before the limit has been reached and will only be charged for the amount fueled

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

Acceptance criteria :

1. Receipt includes: Amount fueled, Amount paid, Tax, Vehicle number, Date, Time

Exercise: Write acceptance criteria



- ▶ Write acceptance criteria for 2 to 3 Stories you have identified
- ▶ Make sure the acceptance criteria is testable
- ▶ Ensure that the Story meets the INVEST criteria to prepare for estimation



Estimation is a whole-team exercise

Agile Teams estimate Stories; POs provide clarification, but do not estimate the work

- ▶ Usually occurs during the backlog refinement event
- ▶ Increases accuracy by including all perspectives
- ▶ Builds understanding
- ▶ Creates shared commitment



Estimation performed by a manager, architect, or select group negates these benefits.

Estimate Stories with relative Story points

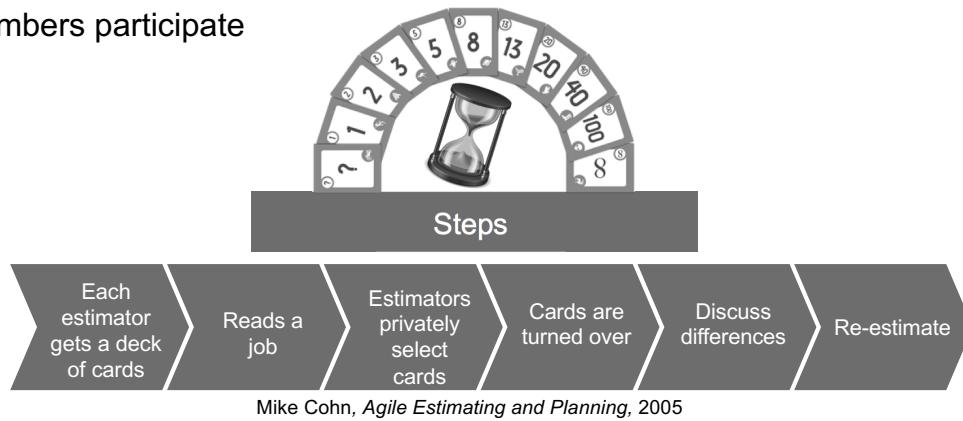
- ▶ A Story point is a singular number that represents:
 - Volume: how much is there?
 - Complexity: how difficult is it?
 - Knowledge: what do we know?
 - Uncertainty: what is not known?
- ▶ Story points are relative. They are not connected to any specific unit of measure.
- ▶ Compare with other stories (an 8-point story should take 4X longer than a 2-point story)

How big is it?



Apply Estimating Poker for fast, relative estimating

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



Exercise: Relative size estimating



Use Estimating Poker to relatively estimate a group of Stories.

- ▶ Use one set of Stories from your team
- ▶ Identify the smallest Story that takes about a day to develop and mark it as 1
- ▶ Estimate the remaining Story using the values 1, 2, 3, 5, 8, 13, 20, 40, 100



5.3 Plan the Iteration

Plan and commit

Purpose

Define and commit to what will be built 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 commits to leaving priorities unchanged during the Iteration

Iteration Goals

Iteration Goals provide clarity, commitment, and management information.

They serve three purposes:

1. Align team members to a common purpose
2. Align Program Teams to common PI Objectives and manage dependencies
3. Provide continuous management information

Iteration Goals example

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

Commit to the Iteration Goals

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 a red flag.

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.

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

Exercise: Iteration Planning - PO and PM roles



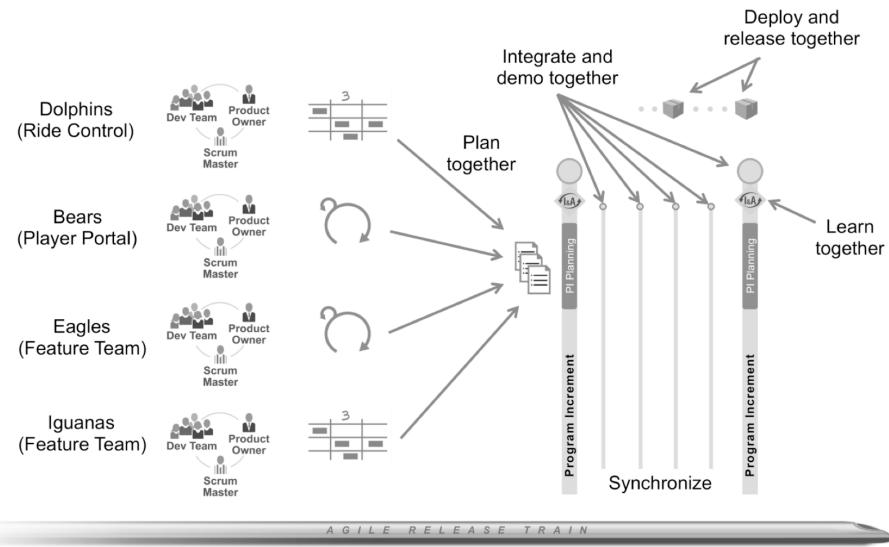
Iteration Planning

- ▶ What activities are required for the PO?
- ▶ What activities are required for the PM?
- ▶ Capture ideas and share with the class



5.4 Execute the PI

Agile Teams are on the Train

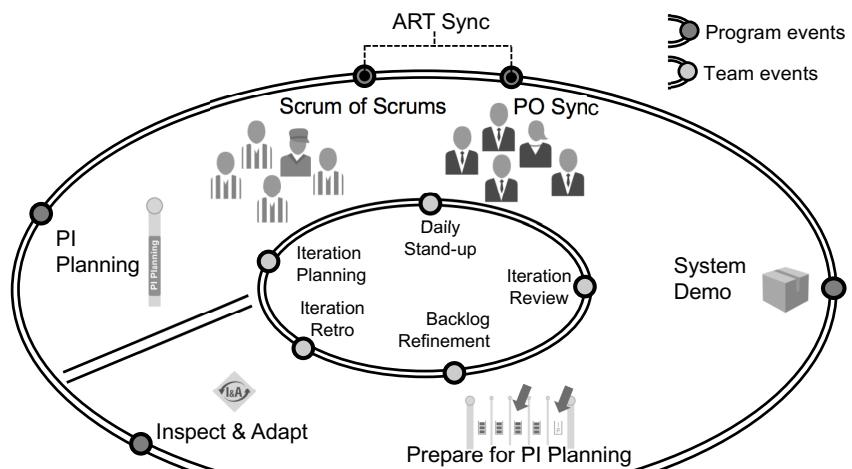


SCALED AGILE® © Scaled Agile, Inc.

5.45

Program Execution

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



SCALED AGILE® © Scaled Agile, Inc.

5.46

The Daily Stand-up (DSU)

The DSU is the key to team synchronization and self-organization.

The DSU (or Daily Scrum) is **not** a daily status meeting for management. It is used to:

- ▶ Share information about progress
- ▶ Coordinate activities
- ▶ Raise blocking issues
- ▶ Most importantly, ask, “*Will we still meet our Iteration Goals and our commitment?*”



- Every day at the same time in front of the team board
- Timebox of 15 minutes
- Not a problem-solving session
- Update the board

ART Sync

Programs coordinate dependencies through sync meetings.



Scrum of Scrums

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

Exercise: Roleplay the events



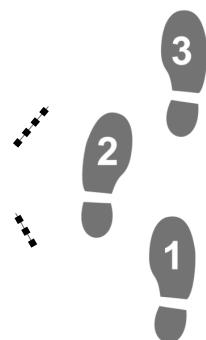
- ▶ Daily Stand Up and ART Sync
 - One half roleplays a typical Daily Stand-up
 - Other half roleplays a typical ART Sync



Sequencing Stories and Team Backlog Prioritization

Primary economic prioritization happens at the Program Backlog. Agile Teams sequence work for efficient execution of business priorities.

- ▶ The Product Owner and the team sequence work based on:
 - Story priorities inherited from Program Backlog priorities
 - Events, Milestones, Releases, and other commitments made during PI Planning
 - Dependencies with other teams
 - Local priorities
 - Capacity allocations for defects, maintenance, and refactors
- ▶ Initial sequencing happens during PI Planning
- ▶ Adjustments happen at Iteration boundaries



The Backlog Refinement Session

What it is:	How it's done:
<p>The backlog refinement session is a preview and elaboration of upcoming Stories.</p> <ul style="list-style-type: none">▶ Helps the team 'sleep' on new Stories prior to Iteration Planning▶ Provides enough time to identify and resolve dependencies and issues that could impact the next Iteration▶ Creates strong process foundation for test automation and Acceptance Test-Driven Development (ATDD)	<p>It is not a single event but a continuous effort.</p> <ul style="list-style-type: none">▶ The PO synthesizes input from Mgmt, subject matter experts, Customers, and other stakeholders as needed▶ The PO always has:<ul style="list-style-type: none">– One to two Iterations of Stories– High-level acceptance criteria– A sense for capacity allocation and relative priorities within types

Exercise: Backlog Refinement



- ▶ At your table, discuss the backlog refinement session, preparation, and guidelines for the Team Backlog
- ▶ Consider the following questions
 - What does the PO/PM need to do to prepare?
 - What is the PO/PM's involvement in these activities?

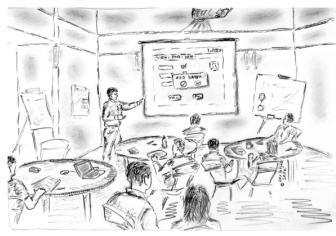


The Iteration Review: Two views

The Iteration Review provides measures of progress into the program:

1. How we did on the Iteration

- ▶ Did we meet the goals?
- ▶ Story by Story review



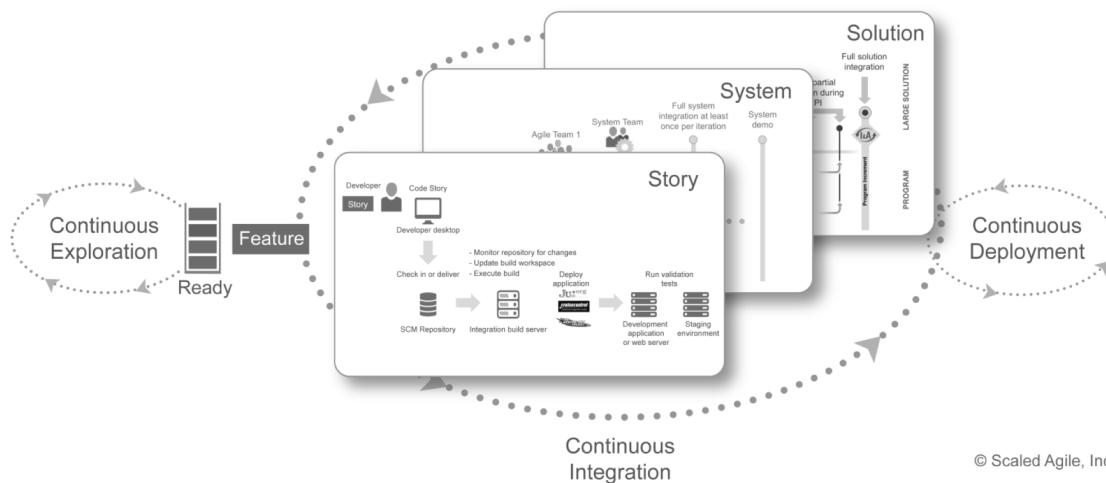
2. How we're doing on the PI

- ▶ Review of PI objectives
- ▶ Review remaining PI scope and reprioritize if necessary

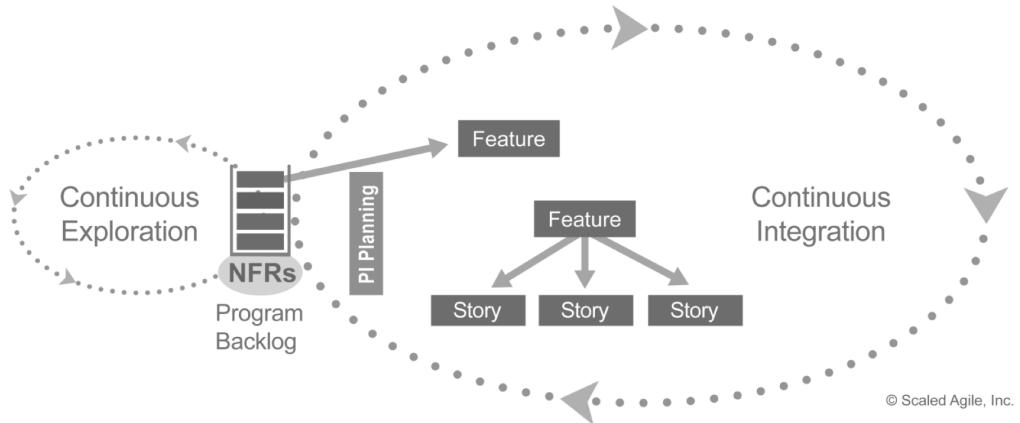
Attendees are the team and its stakeholders

Ad hoc demos – Don't wait for the Iteration Review.
Demo Stories when they are done.

Continuous Integration



Continuously Integrate and Test Stories

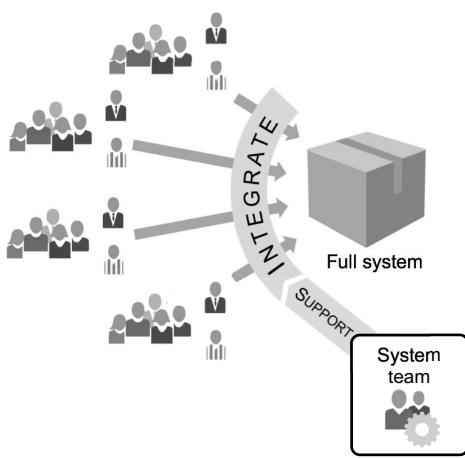


SCALED AGILE® © Scaled Agile, Inc.

5.55

System Demo every two weeks

Demonstrate the full Solution increment to stakeholders every Iteration.



- ▶ An integrated Solution demo
- ▶ Happens after the teams' demo (may lag by as much as one Iteration, maximum)
- ▶ Demo from the staging environment, or the nearest proxy



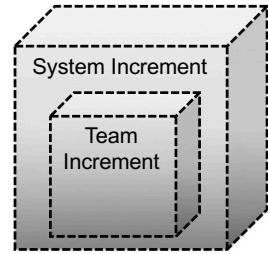
SCALED AGILE® © Scaled Agile, Inc.

5.56

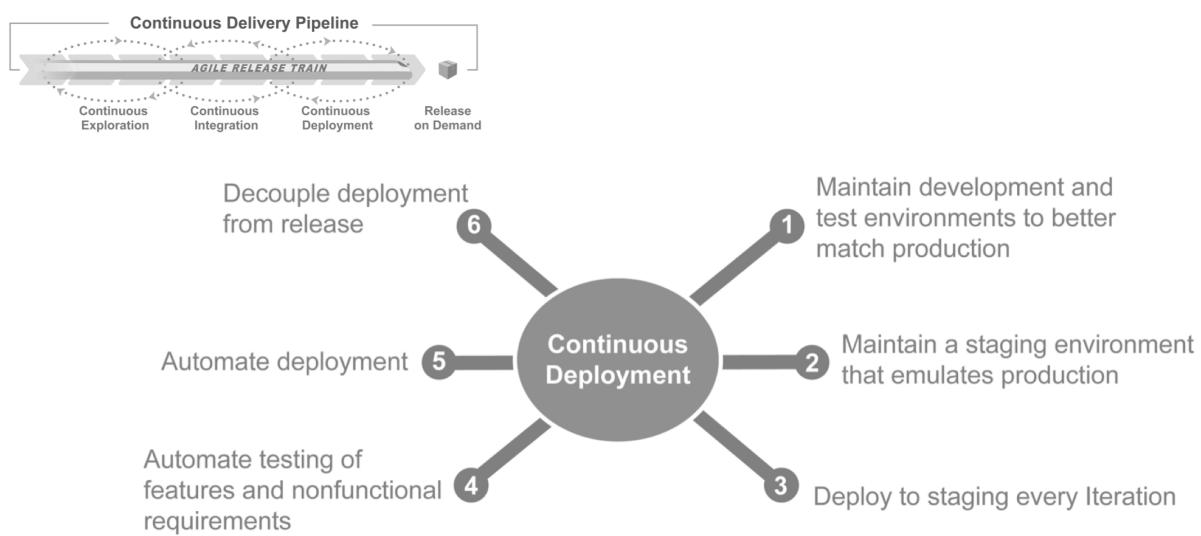
New system increment every two weeks

Every two weeks, teams evaluate the status of the new, integrated system increment.

- ▶ Features are functionally complete or ‘toggled’ so as not to disrupt demonstrable functionality
- ▶ New Features work together, and with existing functionality
- ▶ Architectural Runway work in process is scaffolded and toggled
- ▶ System is continually verified via Story and Feature acceptance tests
- ▶ All practical NFR testing is done continuously



Continuous Deployment



Create your DevOps culture with a CALMR approach



SCALED AGILE® © Scaled Agile, Inc.

5.59

Exercise: Create an action poster!



- ▶ Connect actions to Continuous Delivery Pipeline
 - One group selects Continuous Integration (pages 151-157)
 - One group selects Continuous Deployment (pages 158-162)
 - One group selects DevOps (pages 163-168)
- ▶ Using the articles in the workbook (starting on page 151), create a poster of actions you can take as a PO/PM to increase these capabilities within your enterprise



SCALED AGILE® © Scaled Agile, Inc.

5.60

Iteration Retrospective

At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

—Agile Manifesto

- ▶ 30 – 60 minutes
 - ▶ Pick 1 – 2 things that can be done better, target for next Iteration
 - ▶ Enter improvement items into the Team Backlog

The Innovation and Planning (IP) Iteration

Sunday	Monday	Tuesday	Wednesday	Thursday	Friday	Saturday																													
31	1	2	3	4	5	6																													
		Validation (if shipping)																																	
		Innovation / hackathon / spikes for next PI																																	
		PI Planning readiness																																	
7	8	9	<div style="background-color: #333; color: white; padding: 10px; text-align: center;"> Continuing education Inspect and Adapt workshop </div> <div style="background-color: #333; color: white; padding: 10px; text-align: center;"> PI Planning <table border="1" style="width: 100%; border-collapse: collapse;"> <tr><td>8:00-9:00</td><td>Business Context</td><td>8:00-9:00</td><td>Planning Adjustments</td></tr> <tr><td>9:00-10:30</td><td>Product/Solution Vision</td><td>9:00-11:00</td><td>Team Breakouts</td></tr> <tr><td>10:30-11:30</td><td>Architecture Vision & Development Practices</td><td>11:00-1:00</td><td>Final Plan Review & Lunch</td></tr> <tr><td>11:30-1:00</td><td>Planning Requirements & Lunch</td><td>1:00-2:00</td><td>Program Risks</td></tr> <tr><td>1:00-4:00</td><td>Team Breakouts</td><td>2:00-2:15</td><td>PI Confidence Vote</td></tr> <tr><td>4:00-5:00</td><td>Draft Plan Review</td><td>2:15-3:00</td><td>Plan Rework if Necessary</td></tr> <tr><td>5:00-6:00</td><td>Management Review & Problem Solving</td><td>3:00-After Commitment</td><td>Planning Retrospective & Moving Forward</td></tr> </table> </div>					8:00-9:00	Business Context	8:00-9:00	Planning Adjustments	9:00-10:30	Product/Solution Vision	9:00-11:00	Team Breakouts	10:30-11:30	Architecture Vision & Development Practices	11:00-1:00	Final Plan Review & Lunch	11:30-1:00	Planning Requirements & Lunch	1:00-2:00	Program Risks	1:00-4:00	Team Breakouts	2:00-2:15	PI Confidence Vote	4:00-5:00	Draft Plan Review	2:15-3:00	Plan Rework if Necessary	5:00-6:00	Management Review & Problem Solving	3:00-After Commitment	Planning Retrospective & Moving Forward
8:00-9:00	Business Context	8:00-9:00	Planning Adjustments																																
9:00-10:30	Product/Solution Vision	9:00-11:00	Team Breakouts																																
10:30-11:30	Architecture Vision & Development Practices	11:00-1:00	Final Plan Review & Lunch																																
11:30-1:00	Planning Requirements & Lunch	1:00-2:00	Program Risks																																
1:00-4:00	Team Breakouts	2:00-2:15	PI Confidence Vote																																
4:00-5:00	Draft Plan Review	2:15-3:00	Plan Rework if Necessary																																
5:00-6:00	Management Review & Problem Solving	3:00-After Commitment	Planning Retrospective & Moving Forward																																
						13																													

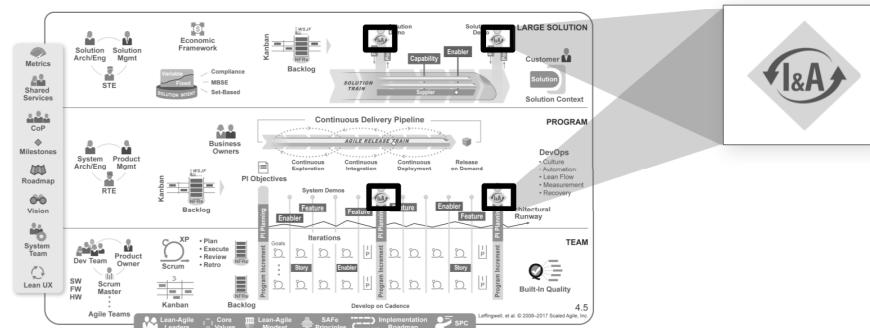
Inspect and Adapt

Three parts:

1. The PI System Demo
2. Quantitative measurement
3. The problem-solving workshop

► Attendees: Teams and stakeholders

► Timebox: 3 – 4 hours per PI



SCALED AGILE® © Scaled Agile, Inc.

5.63

Exercise: Reflecting on Program Increment events



- Map out your program events on a calendar, and socialize with another team
- Think about the PO and PM roles, and place the program events in the column in which that role would add the most value for them to attend/participate.
- Prepare to share with the class



SCALED AGILE® © Scaled Agile, Inc.

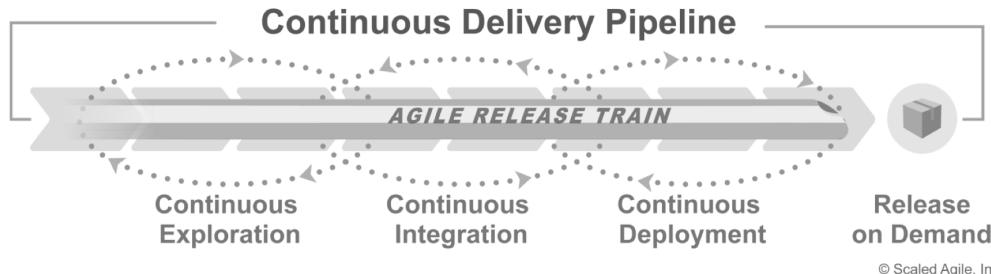
5.64

5.5 Release on Demand

SAFe Definition of Done

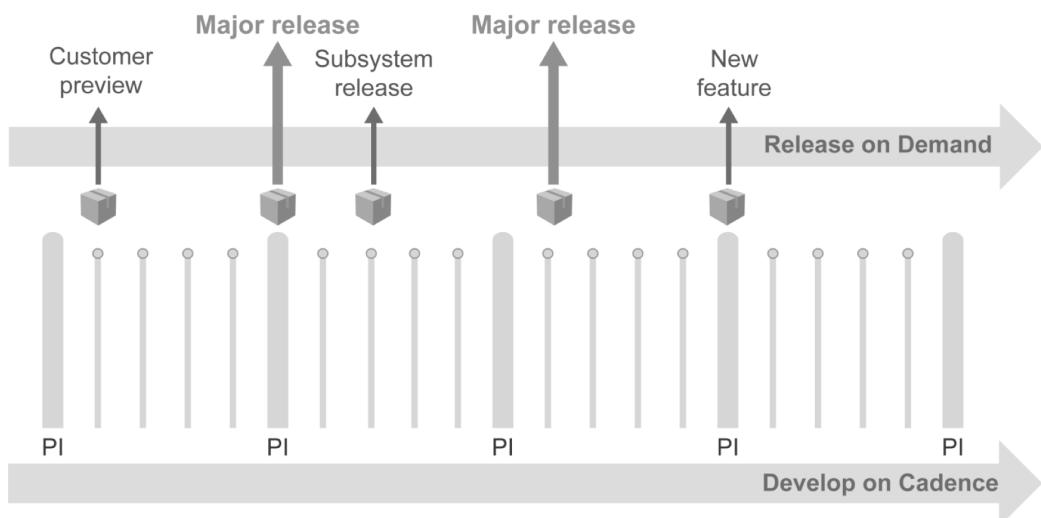
Team Increment	System Increment	Solution Increment	Release
<ul style="list-style-type: none">Stories satisfy acceptance criteriaAcceptance tests passed (automated where practical)Unit and component tests coded, passed, and included in the BVTCumulative unit tests passedAssets are under version controlEngineering standards followedNFRs metNo must-fix defectsStories accepted by Product Owner	<ul style="list-style-type: none">Stories completed by all teams in the ART and integratedCompleted features meet acceptance criteriaNFRs metNo must-fix defectsVerification and validation of key scenariosIncluded in build definition and deployment processIncrement demonstrated, feedback achievedAccepted by Product Management	<ul style="list-style-type: none">Capabilities completed by all trains and meet acceptance criteriaDeployed/installed in the staging environmentNFRs metSystem end-to-end integration, verification, and validation doneNo must-fix defectsIncluded in build definition and deployment/transition processDocumentation updatedSolution demonstrated, feedback achievedAccepted by Solution Management	<ul style="list-style-type: none">All capabilities done and meet acceptance criteriaEnd-to-end integration and solution V&V doneRegression testing doneNFRs metNo must-fix defectsRelease documentation completeAll standards metApproved by Solution and Release Management

Release on Demand

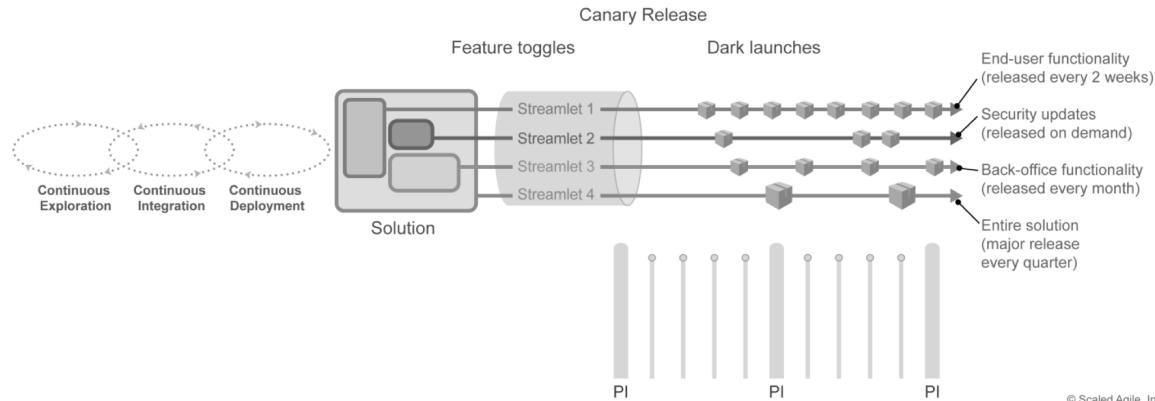


For more on the Continuous Delivery Pipeline, watch the “Introduction to Continuous Delivery Pipeline” video on the SAFe Community Platform

Decouple Cadence. Release on Demand



Architect the Solution for Incremental Release



Exercise: Release on Demand strategies



► Your Release on Demand strategies:

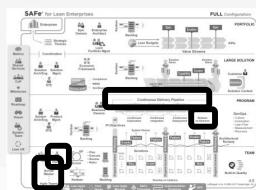
- What can you do to evolve your Definition of Done?
- What can you do to better foster architecting the solution for incremental release?
- What might you change in the future?



Lesson summary

In this lesson, you:

- ▶ Created alignment with PI Planning
- ▶ Decomposed Features into stories
- ▶ Explored planning the Iteration
- ▶ Discussed events to execute the PI
- ▶ Integrated Release on Demand strategies



Suggested Scaled Agile Framework reading:

"Scrum Master", "Agile Teams", "Product Owner", "Continuous Delivery Pipeline", and "Release on Demand" articles

Exercise: This lesson's key learnings



Summarize key learnings and insights from this lesson in your workbook.

