

Lesson 3

Exploring Agile and Scrum Anti-Patterns

1. Exploring the Scrum Master Role in the SAFe Enterprise
2. Applying SAFe Principles: A Scrum Master's Perspective
3. Exploring Agile and Scrum Anti-Patterns
4. Facilitating Program Execution
5. Improving Flow with Kanban and XP
6. Building High-Performing Teams
7. Improving Program Performance with Inspect and Adapt

SAFe® Course Attending this course gives students access to the SAFe® Advanced Scrum Master exam and related preparation materials.

Learning objectives

- 3.1 Explore anti-patterns associated with the Product Owner role
- 3.2 Learn how Stories and tasks may lead to anti-patterns
- 3.3 Identify context-specific anti-patterns in your environment

3.1 Explore anti-patterns associated with the Product Owner role

Recognizing anti-patterns

As an Agile coach, the Scrum Master must learn to recognize anti-patterns in the process.

- ▶ Anti-patterns can be structural or behavioral
 - Structural example: Team has more than one Product Owner
 - Behavioral example: Partially completed Stories are being carried over from Iteration to Iteration
- ▶ Anti-patterns can be internal or external
 - Internal example: Developers don't work collaboratively on Stories
 - External example: Lack of coordination with other teams leads to excessive WIP



Many anti-patterns can be traced to the PO role

Underperforming in the Product Owner role can lead to dysfunction on the team.



Key responsibilities of the Product Owner:

- ▶ Facilitate Team Backlog refinement
- ▶ Prepare for and participate in Iteration Planning
- ▶ Elaborate Stories and Enablers 'just in time'
- ▶ Address team questions; be the 'voice of the customer'
- ▶ Accept Stories
- ▶ Participate in the Iteration Review and retrospective
- ▶ Coordinate with other Product Owners to manage dependencies

Exercise: Anti-patterns that involve the Product Owner

- ▶ Brainstorm dysfunctions that may arise from the interaction between the Product Owner and the rest of the team or from the role itself
- ▶ Discuss how you might address such dysfunction
- ▶ Write down two or three anti-patterns identified with:
 - Name
 - Brief description
 - Impact
 - Solution(s)

PREPARE



SHARE



Exercise: Scrum Master anti-patterns

- ▶ Watch the following video and try to identify anti-patterns
- ▶ Discuss in your group



<https://youtu.be/GGbsgs611MM>

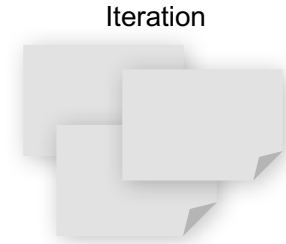


3.2 Learn how Stories and tasks may lead to anti-patterns

Big Stories are a frequent source of anti-patterns

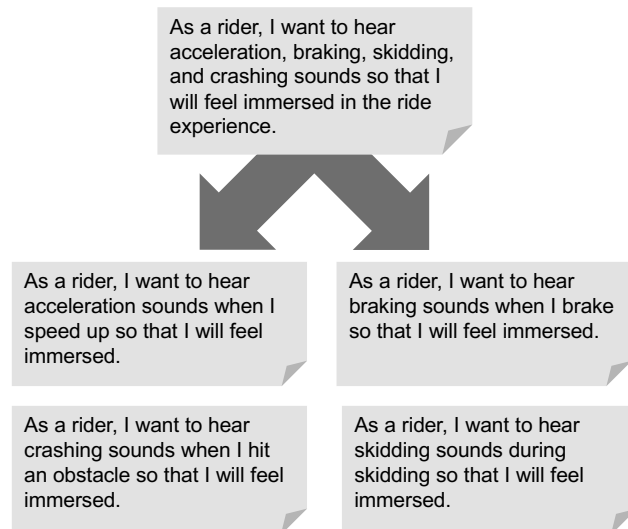
A team that can't iterate isn't able to inspect and adapt.


- ▶ Big Stories do not support team iteration
- ▶ Smaller Stories allow for faster, more reliable implementation
- ▶ Splitting bigger Stories into smaller ones is an essential skill



Ways to split a Story

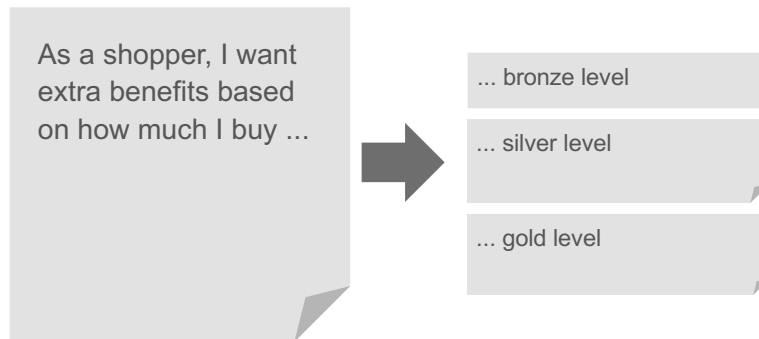
- ▶ By business rule variations
- ▶ By use case scenario
- ▶ Simple/complex



More on Story splitting at:
 www.scaledagileframework.com/story

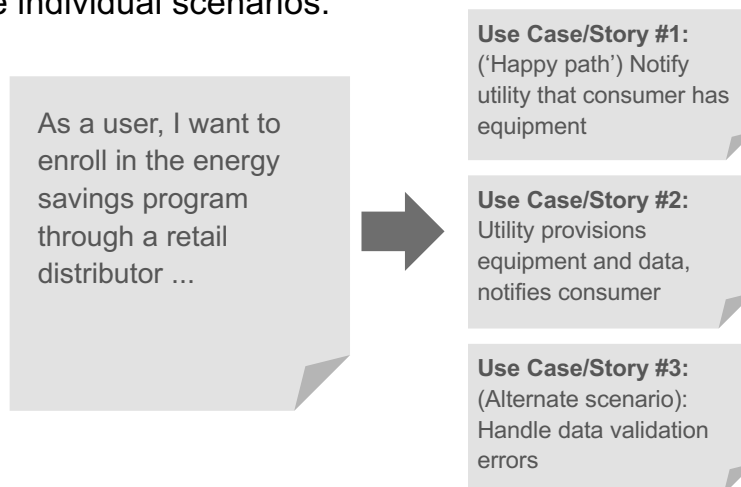
Split by business rule variations

Business rule variations often provide a straightforward splitting scheme.



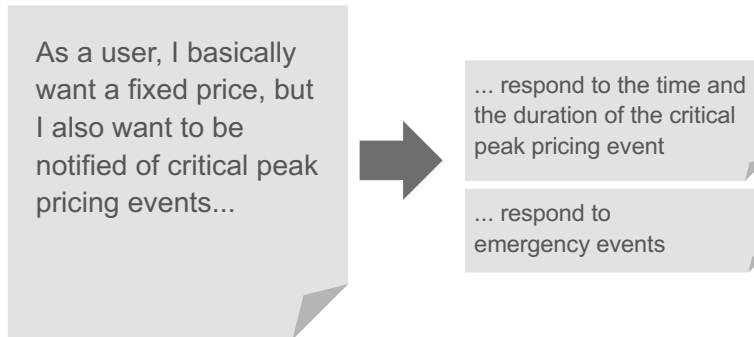
Split by use case scenarios

If use cases are used to represent complex interaction, the Story can be split via the individual scenarios.



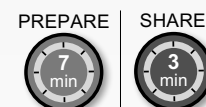
Split by simple/complex

Simplify! What's the simplest version that can possibly work?



Exercise: Splitting Stories

- ▶ Select a big Story from your actual backlog that is too big to complete in an Iteration
- ▶ Using Post-its and the patterns in the previous slides, split the Story into two to three smaller Stories that could be completed in an Iteration
- ▶ Ideally two to three days of work
- ▶ Be ready to share your work



Exercise: A Story that can't be split?

Your team is insisting that big Stories in the backlog just cannot be split into smaller ones:

- ▶ How might you coach the team to reconsider?
- ▶ What specific steps would you take to shift their mindset?
- ▶ Be ready to discuss

PREPARE



SHARE



Exercise: Tasking

- ▶ Is tasking an anti-pattern?
- ▶ Discuss at your table and be ready to present your conclusions

User Story

Task 1

Task 2

Task 3

PREPARE



SHARE



3.3 Identify context-specific anti-patterns in your environment

PO, backlog, planning, and commitment anti-patterns

PO and backlog

- ▶ Product Owner and team reach Iteration Planning without preparation
- ▶ There is more than one PO per team
- ▶ PO is not sufficiently involved during Iteration execution

Planning

- ▶ Planning is based on tasks, not on User Stories and acceptance criteria



Commitment

- ▶ Team does not commit to clear Iteration goals

Execution, demos, and retrospectives anti-patterns

Execution

- ▶ Developers don't work collaboratively on User Stories
- ▶ Waterfalling Iterations - Team integrates and tests Stories only at Iteration end
- ▶ 'Done isn't done'; debt is carried forward Iteration to Iteration

Demos

- ▶ Team delays demos or extends Iteration
- ▶ Story reported but not demonstrated (non-UI Stories, spikes, refactors, etc.)



Retrospectives

- ▶ 'Idea fest' instead of focus on near-term, incremental improvements

Exercise: Anti-patterns

- ▶ Which of the anti-patterns resonate most strongly?
- ▶ What other anti-patterns are you aware of?

PO, backlog, planning, and commitment anti-patterns

PO and backlog <ul style="list-style-type: none">▶ Product Owner and team reach Iteration Planning without preparation▶ There is more than one PO per team▶ PO is not sufficiently involved during Iteration execution	Planning <ul style="list-style-type: none">▶ Planning is based on tasks, not on User stories and acceptance criteria
Commitment <ul style="list-style-type: none">▶ Team does not commit to clear Iteration goals	

SCALED AGILE © Scaled Agile, Inc.

Execution, demos, and retrospectives anti-patterns

Execution <ul style="list-style-type: none">▶ Developers don't work collaboratively on user stories▶ Waterfalling Iterations - Team integrates and tests Stories only at Iteration end▶ 'Done isn't done'; debt is carried forward Iteration to Iteration	Demos <ul style="list-style-type: none">▶ Team delays demos or extends Iteration▶ Story reported but not demonstrated (non-UI stories, spikes, refactors, etc.)
Retrospectives <ul style="list-style-type: none">▶ 'Idea fest' instead of focus on near-term, incremental improvements	

SCALED AGILE © Scaled Agile, Inc.

PREPARE



SHARE



Exercise: Let's solve some of these...

- ▶ Pair with somebody, and discuss whether there are anti-patterns that could be fully resolved by the Scrum Master
- ▶ Pick two to three such anti-patterns, and for each one build a list of action items that would allow the Scrum Master to solve it
- ▶ Be ready to present

PREPARE



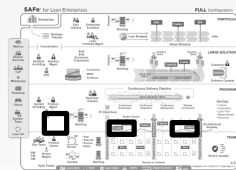
SHARE



Lesson summary

In this lesson, you:

- ▶ Considered some typical examples of Agile and Scrum anti-patterns
- ▶ Explored anti-patterns in your context



Suggested Scaled Agile Framework reading:

- "Product Owner" article
- "Stories" article

Other suggested reading:

- "Seven Sins of Scrum and other Agile Anti-patterns"
www.infoq.com/news/2016/03/agileindia-7sins-scrum