

Lesson 2

Building an Agile Team

- 1. Introducing the Scaled Agile Framework
- 2. Building an Agile Team
- 3. Planning the Iteration
- 4. Executing the Iteration
- 5. Executing the PI

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

Learning objectives

- 2.1 Build your team
- 2.2 Explore the Scrum Master and Product Owner roles
- 2.3 Meet the teams and people on the train

2.1 Build your team

The Agile Team

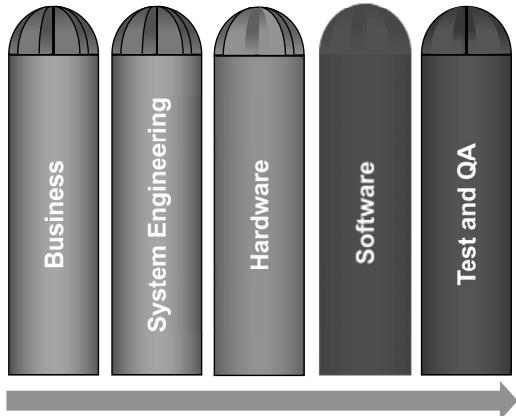
The ‘relay race’ approach to product development ... may conflict with the goals of maximum speed and flexibility. Instead, a holistic or ‘rugby’ approach—where a team tries to go the distance as a unit, passing the ball back and forth—may better serve today’s competitive requirements.



[Youtu.be/KWnwI0-aeq0](https://youtu.be/KWnwI0-aeq0)

—Hirotaka Takeuchi and Ikujiro Nonaka,
“The New New Product Development Game,”
Harvard Business Review, January 1986

Value doesn't follow silos

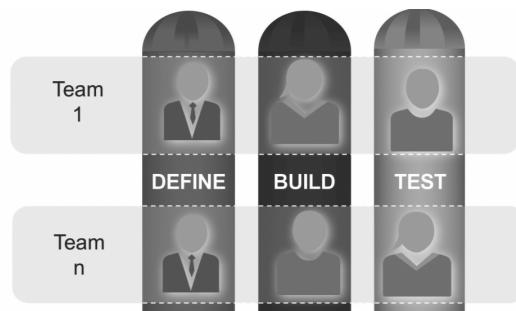


- Optimized for vertical communication
- Friction across the silos
- Location via function
- Political boundaries between functions

Management challenge:
Connect the silos

Build cross-functional Agile Teams

- ▶ Cross-functional, self-organizing entities that can define, build, and test a feature or component
- ▶ Optimized for communication and delivery of value
- ▶ Deliver value every two weeks



The power of “ba”

We, the work, and the knowledge are all one

- ▶ Dynamic interaction of individuals and the organization in the form of a self-organizing team; the fuel of “ba” is its self-organizing nature
- ▶ Team members create new points of view and resolve contradictions through dialogue
- ▶ “Ba” is energized with intentions, vision, interest, and mission
- ▶ Leaders provide autonomy, variety, trust, and commitment
- ▶ 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



Nishida, 1921, 1970; *Hitotsubashi on Knowledge Management*, Takeuchi and Nonaka, 2004



youtu.be/yiKFYTFJ_kw

2.7

SCALED AGILE® © Scaled Agile, Inc.

Teamwork

It is amazing what you can accomplish if you do not care who gets the credit.

—Harry Truman

SCALED AGILE® © Scaled Agile, Inc.

2.8

Exercise: Ball game

- ▶ Game objective is to process as many balls as possible as one team
- ▶ Balls must be passed with air time between any two people. They must be touched by each person. A ball must return to start point before it is counted complete.
- ▶ You will have 2 minutes to organize and begin your first Iteration
- ▶ You will do 5 iterations. Each iteration is 2 minutes, followed by a 1 minute retrospective.
- ▶ To get credit, you must provide an estimate for the number of balls you think you can process before each iteration



Collocated teams enhance productivity

Collocation:

- ▶ Critical for the Agile Team to be effective
- ▶ Recommended for programs to have efficient product development flow
- ▶ Distributed development must be compensated with efficient remote interaction (video-conferencing, sharing and collaboration tools, Agile project management tools, etc.)



Team working area

Shared team area



SCALED AGILE® © Scaled Agile, Inc.

2.11

Agile Teams power the train



Development Team

- ▶ Creates and refines user stories and acceptance criteria
- ▶ Defines/builds/tests/delivers Stories
- ▶ Develops and commits to Team PI Objectives and Iteration plans



Product Owner

- ▶ Defines and accepts Stories
- ▶ Acts as the Customer for developer questions
- ▶ Works with Product Management to plan Releases
- ▶ A team has only one Product Owner, who may be dedicated to one or two teams



Scrum Master

- ▶ Runs team meetings, coaches Agile mindset and practices
- ▶ Removes impediments, protects the team from outside influence
- ▶ Attends Scrum of Scrums meetings
- ▶ May be a part-time role for a team member (25 – 50%), or a single Scrum Master may be shared across 2 – 3 teams

SCALED AGILE® © Scaled Agile, Inc.

2.12

System Team

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



- ▶ Builds the development infrastructure and manages environments
- ▶ Assists with test automation strategies and adoption
- ▶ Provides/supports full system integration
- ▶ Performs end-to-end system and performance testing
- ▶ Stages and supports the System Demos

Organizing teams around value

Organize for the larger purpose

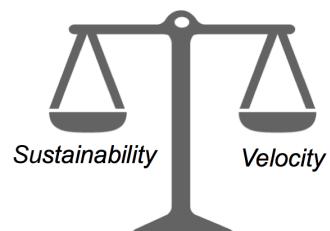
- ▶ Maximize velocity by minimizing dependencies and handoffs, while sustaining architectural robustness and system qualities

A team can be organized around

- ▶ Features
- ▶ Components

Far less desirable

- ▶ Architectural layer
 - Platform, middleware, UI, DB, business logic, etc.
- ▶ Other
 - Programming language, spoken language, technology, location



Finding the right trade-off

Most large programs have a mix.

Lean toward Feature Teams:

- ▶ Fastest velocity
- ▶ Minimize dependencies
- ▶ Develop T-shaped skills

Use Component Teams when:

- ▶ High reuse, high technical specialization, critical NFRs
- ▶ Creating each component as a “potentially replaceable part of the system, with well-defined interfaces”

Generally avoid organizing around architectural layers, as they create team coupling and don't provide a technical separation of concerns.



Exercise: Build your team

Full ART Only

As a **team**, you want to understand:

- ▶ What are your responsibilities and skill sets?
- ▶ Define our team name. Team names should not be the names of components, subsystems, or feature areas.
- ▶ Discuss your role as a feature or component team
- ▶ Discuss what your team is responsible for, and what other things you can do
- ▶ Choose a name for your team
- ▶ Prepare a short presentation about your team (name, role on the train, and special skills on the team that other teams should know about)



Exercise: Build your team

- ▶ After class go back to your team and discuss:
 - What is our team name and why was it chosen?
 - Are we a feature or a component team?



2.2 Explore the Scrum Master and Product Owner roles

Roles: Scrum Master

- ▶ Coaches the team
- ▶ Ensures that the team follows Agile principles and practices
- ▶ Facilitates processes and meetings
- ▶ Removes impediments and barriers
- ▶ Protects the team from external forces



The Scrum Master in the Enterprise

- ▶ Coordinates with other Scrum Masters, the System Team, and shared resources in the ART PI Planning meetings
- ▶ Works with the above teams throughout each Iteration and PI
- ▶ Coordinates with other Scrum Masters and the Release Train Engineer in Scrum of Scrums
- ▶ Fosters normalized estimating within the team
- ▶ Helps teams operate under architectural and portfolio governance, System Level integration, and System Demos
- ▶ Fosters adoption of Agile technical practices



Roles: Product Owner

- ▶ Member of the Agile Team
- ▶ A single voice for the Customer and stakeholders
- ▶ Owns and manages the Team Backlog
- ▶ Defines and accepts requirements
- ▶ Makes the hard calls on scope and content



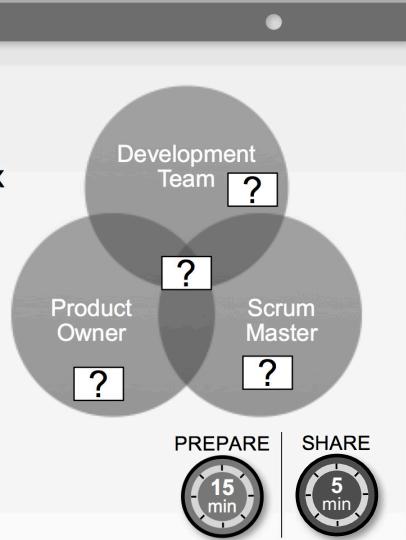
The Product Owner in the Enterprise

- ▶ 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, usually reporting via a “fat dotted line” to Product Management
- ▶ Understands how the Enterprise Backlog Model operates with Epics, Capabilities, Features, and Stories
- ▶ 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



Scrum roles and responsibilities card game

- ▶ In your group, draw the following Venn diagram
- ▶ Review the responsibilities sheet in the Appendix and create sticky notes for each responsibility, placing them either in a role or at an intersection
- ▶ Prepare to discuss your decisions



2.3 Meet the teams and people on the train

The Agile Release Train

- ▶ A virtual organization of 5 – 12 teams (50 – 125+ individuals) that plans, commits, and executes together
- ▶ Program Increment (PI) is a fixed time box; default is 10 weeks
- ▶ Synchronized Iterations and PIs
- ▶ Aligned to a common mission via a single Program Backlog
- ▶ Operates under architectural and UX guidance
- ▶ Frequently produces valuable and evaluable System Level Solutions



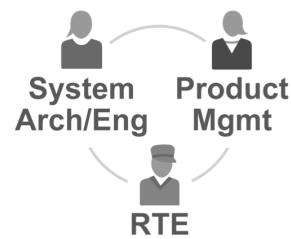
SCALED AGILE® © Scaled Agile, Inc.

2.25

The ART roles

Three primary roles help ensure successful execution of the Vision and Roadmap initiatives at the Program Level:

1. The Release Train Engineer is a servant leader who facilitates and guides the work of the ART. He acts like a chief Scrum Master.
2. Product Management is the main content authority guiding the train. They own and prioritize the Program Backlog.
3. The System Architect/Engineer has the technical responsibility for the overall architectural and engineering design of the system. She provides architectural and technical guidance to the teams on the train.



SCALED AGILE® © Scaled Agile, Inc.

2.26

Exercise: Know the people on the train

The RTE presents himself and the main players on the train:

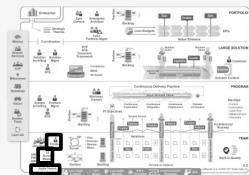
- ▶ Product Management
- ▶ System Architect/Engineering
- ▶ Lean UX
- ▶ DevOps
- ▶ Shared Services
- ▶ Each team presents itself (name, area of responsibility, special skills)



Lesson summary

In this lesson, you:

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



Suggested Scaled Agile Framework reading:

- “Agile Teams” article
- “Scrum Master” article
- “Product Owner” article