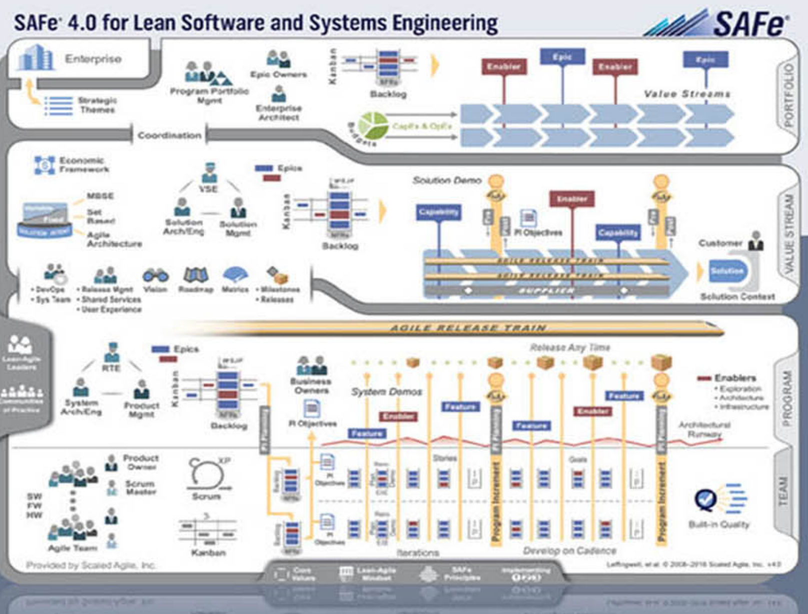
SAFe® 4.0

SAFe® REFERENCE GUIDE

SCALED AGILE FRAMEWORK® FOR LEAN SOFTWARE AND SYSTEMS ENGINEERING



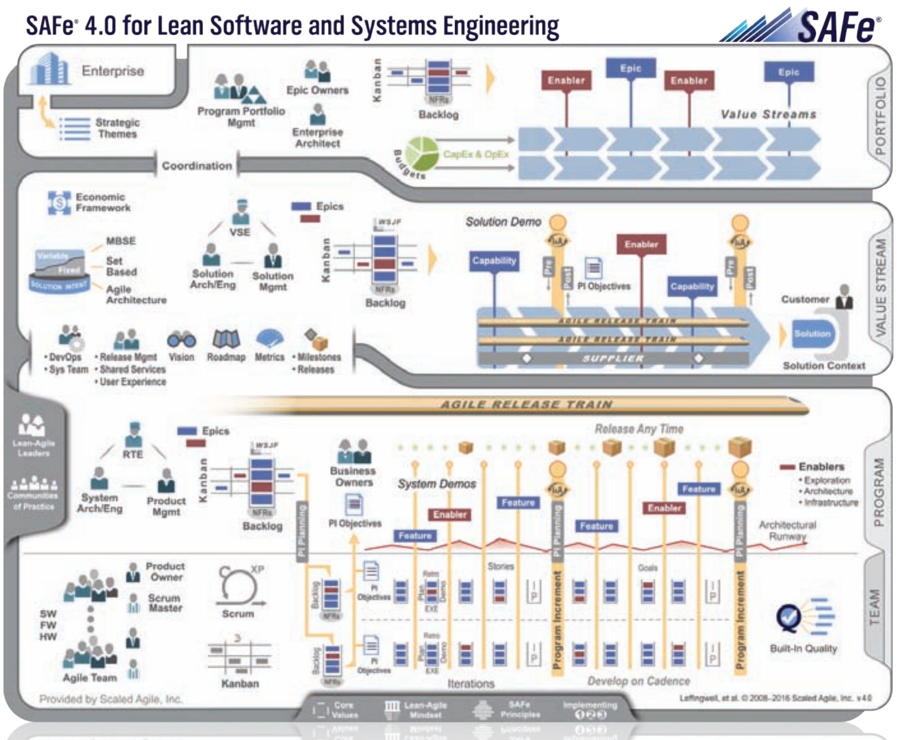
Dean Leffingwell

with Alex Yakyma, Richard Knaster, Drew Jemilo, and Inbar Oren

SAFe® 4.0

SAFe® REFERENCE GUIDE

SCALED AGILE FRAMEWORK® FOR LEAN SOFTWARE AND SYSTEMS ENGINEERING



Dean Leffingwell

with Alex Yakyma, Richard Knaster, Drew Jemilo, and Inbar Oren

Library of Congress Control Number: 2016943742

ISBN-13: 978-0-13-451054-5

ISBN-10: 0-13-451054-2

Text printed in the United States on recycled paper at RR Donnelley in Crawfordsville, Indiana.

First printing, August 2016

Contents

Preface

On behalf of the entire Scaled Agile, Inc., team and the SAFe contributors, it is my personal pleasure to introduce the *SAFe 4.0 Reference Guide*.

SAFe is an online, freely revealed knowledge base of proven success patterns for implementing Lean-Agile software and systems development at enterprise scale. It provides comprehensive guidance for work at the enterprise Portfolio, Value Stream, Program, and Team levels.

Why SAFe?

The world's economy, and the health and welfare of society as a whole, is increasingly dependent on software and systems. In support of this need, systems builders are creating increasingly complex software and cyber-physical systems of unprecedented scope and complexity with requirements for utility and robustness exceeding those that have come before them.

The methods that systems builders use to create these systems must keep pace with this larger mandate. However, the assumptive, one-pass, stage-gated, waterfall methods of the past are not scaling to the new challenge. New development methods are needed. Agile shows the greatest promise, but it was developed for small teams and, by itself, does not scale to the needs of larger enterprises and the systems they create. What's needed is a new way of working, one that applies the power of Agile but leverages the more extensive knowledge pools of systems thinking and Lean product development. The Scaled Agile Framework (SAFe) is one such approach.

SAFe is provided by Scaled Agile, Inc., where our core belief is simple: Better systems and software make the world a better place. Our mission is to assist those who build these systems through development and publication of the SAFe framework, as well as accompanying certification, training, and courseware. As case studies on the Scaled Agile Framework website (www.scaledagileframework.com) show, many enterprises—large and small—are getting outstanding business benefits from applying SAFe.

These typically include:

- 20 – 50% increase in productivity
- 50%+ increases in quality
- 30 – 75% faster time to market
- Measurable increases in employee engagement and job satisfaction

As you can imagine, with results like those, SAFe is spreading rapidly around the world. The majority of Fortune 100 U.S. companies have certified SAFe practitioners and consultants already on site, as do an increasing percentage of the Global 1000 enterprises. SAI also has an extensive network of more than 80 global partners providing consulting and implementation services in almost every region of the world. And while every business situation is unique, we have found that the straightforward Implementation 1-2-3 strategy always delivers results.

Our commitment is to continuously evolve SAFe to provide value to the industry—better systems, better business outcomes, better daily lives for the people who build the world's most important new systems—but only you, the adopters and practitioners, can tell us whether or not we have accomplished that. As we are fond of saying, "without you, SAFe is just a website."

Why a *Reference* Guide?

The SAFe website is quite comprehensive, containing hundreds of pages of guidance for the various roles, responsibilities, activities, and artifacts that constitute SAFe, along with the foundational elements of values, Lean-Agile mindset, principles, and practices. This Reference Guide is intended to provide a hard-copy or e-copy companion to help you understand and apply SAFe, with the goal of helping you accomplish your mission of building better systems.

More on SAFe

For more on SAFe, please browse the site, read the blog, watch the "updates" field, and follow us on Twitter (@ScaledAgile), where we will notify you of new developments. Also, click on the site's "Presentations & Downloads" tab to find free posters for the Big Picture, the House of Lean, and SAFe Lean-Agile Principles. You'll also find SAFe videos and recorded webinars, free presentations, and more. Finally, be sure to check out our corporate site www.ScaledAgile.com. Even better, attend a Training and Certification course (www.scaledagile.com/which-course); perhaps I will see you there. Stay SAFe!

—Dean Leffingwell and the Scaled Agile Team

Acknowledgments

Scaled Agile Framework Contributors

Alex Yakyma, SAFe Fellow and Principal Consultant
Alex is a SAFe methodologist, trainer, and principal consultant who has been involved with the development and field implementations of the Scaled Agile Framework since its inception. Alex's broad prior experience as an engineer, development manager, and program manager in highly distributed multicultural environments provides the experience needed to assist enterprises with improving their system development capabilities at the Program, multi-program, and Portfolio levels. Alex has published a number of articles and white papers on Agile and Lean and is the author of *Pacific Express*, a novella about launching an Agile Release Train.

Drew Jemilo, SAFe Fellow and Principal Consultant
Drew is a principal contributor to the Scaled Agile Framework, a consultant, and an instructor. Drew met Dean Leffingwell in early 2009 when he was developing a scaled Agile methodology for a management consulting company to bridge their strategic business framework with Agile. Since then, they have worked together with global clients to synchronize distributed teams using Agile Release Trains in the United States, Europe, and India.

Richard Knaster, SAFe Fellow and Principal Consultant
Richard has more than 25 years' experience in software development in roles ranging from developer to executive and has been involved in Agile for more than a decade. Prior to joining Scaled Agile, Inc., Richard worked at IBM, where his career spanned from product line management (PPM domain) and professional services to chief methodologist, Agile and Lean. Richard is a certified IBM Thought Leader and an Open Group Distinguished IT Specialist. He is also a certified SPC, PSM, Agile Certified Practitioner, PMP, and a contributor to the Disciplined Agile Delivery framework and PMI Portfolio/Program Management standards.

Inbar Oren, SAFe Fellow and Principal Consultant
Inbar has more than 20 years' experience in the high-tech market. For more than a decade, he has been helping development organizations—in both software and integrated systems—improve results by adopting Lean-Agile best practices. Previous clients include Cisco, Woolworth, Amdocs, Intel, and NCR. Inbar's current focus is on working with leaders at the Program, Value Stream, and Portfolio levels to help them bring the most out of their organizations and build new processes and culture.

SAFe Community Contributors

We are also indebted to those SAFe Program Consultant Trainers (SPCTs) and SAFe Program Consultants (SPCs) who are doing the hard work of applying the framework in various enterprises every day. Many have contributed indirectly in discussions, certification workshops, LinkedIn forums, and more. More specifically, the following individuals have directly provided content that is included either here or in Guidance articles on the Scaled Agile Framework website (www.scaledagileframework.com).

- **Harry Koehnemann, SPCT** – Special contributor to SAFe for Lean Systems Engineering and 4.0 systems engineering content
- **Ken France, SPCT** – Guidance article: "Mixing Agile and Waterfall Development in the Scaled Agile Framework"
- **Scott Prugh, SPC** – Guidance article: "Continuous Delivery"
- **Eric Willeke, SPCT** – Guidance articles: "Role of PI Objectives," "A Lean Perspective on SAFe Portfolio WIP Limit"
- **Jennifer Fawcett, SAFe Fellow and Principal Consultant** – Product Manager and Product Owner contribution and focus
- **Colin O'Neill, SPCT** – SAFe 1.0 – 2.5 contributor
- **Gareth Evans, SPCT** – Guidance article: "Lean Software Development in SAFe"
- **Gillian Clark, SPCT** – Guidance article: "Lean Software Development in SAFe"
- **Maarit Laanti, SPC** – "Lean-Agile Budgeting" guidance and white paper
- **Steven Mather, SPC** – SAFe 2.0 glossary draft
- **Al Shalloway, SPCT** – Concept development and community support

Additional Acknowledgments

The Contributors to Agile Software Requirements

The initial concepts behind the framework were first documented in the 2007 text *Scaling Software Agility: Best Practices for Large Enterprises*, by Dean Leffingwell. But the framework itself was first documented in Dean's 2011 book *Agile Software Requirements: Lean Requirements for Teams, Programs, and the Enterprise* (ASR), so it's appropriate to repeat and update the book acknowledgments here.

Thanks to the ASR reviewers, Gabor Gunyho, Robert Bogetti, Sarah Edrie, and Brad Jackson. Don Reinertsen provided permission to use elements of his book, *The Principles of Product Development Flow*. Thanks to my Finnish collaborators: Juha-Markus Aalto, Maarit Laanti, Santeri Kangas, Gabor Gunyho, and Kuan Eeik Tan. Alistair Cockburn, Don Widrig, Mauricio Zamora, Pete Behrens, Jennifer Fawcett, and Alexander Yakyma contributed directly to book content. Even that list is not exhaustive; many others—Mike Cottmeyer, Ryan Shriver, Drew Jemilo, Chad Holdorf, Keith Black, John Bartholomew, Chris Chapman, Mike Cohn, Ryan Martens, Matthew Balchin, and Richard Lawrence—contributed words, thoughts, or encouragement.

A Special Acknowledgment to the Agile Thought Leaders
Of course, SAFe stands on the shoulders of many who came before us, particularly the Agile thought leaders who created the industry movement. It starts with the signers of the Agile Manifesto and continues with those outspoken thought leaders who have helped move the industry toward the new paradigm. The following have contributed most directly to our understanding of Agile development: Kent Beck, Alistair Cockburn, Ron Jeffries, Mike Cohn, David Anderson, Jeff Sutherland, Martin Fowler, Craig Larman, Ken Schwaber, Scott Ambler, and Mary and Tom Poppendieck. Still others are acknowledged in the Bibliography.

A Special Acknowledgment to the Lean Leaders
In extending Agile to the enterprise and developing the broader Lean-Agile paradigm, we are fortunate to stand on the shoulders of Lean thought leaders as well, including Don Reinertsen, Jeffrey Liker, Taichi Ohno, Eli Goldratt, Dr. Alan Ward, Jim Sutton, Michael Kennedy, Dantar Oosterwal, Steve Womack, and Daniel Jones. Still others are acknowledged in the Bibliography.

And to W. Edwards Deming
Finally, where would we be without the seminal works of W. Edwards Deming, to whom we perhaps owe the deepest gratitude of all? He was a visionary and systems thinker, whose tireless quest for the underlying truths and unwavering belief in people and continuous improvement led to a set of transformational theories and teachings that changed the way we think about quality, management, and leadership.

This page intentionally left blank

Introduction to the Scaled Agile Framework (SAFe)

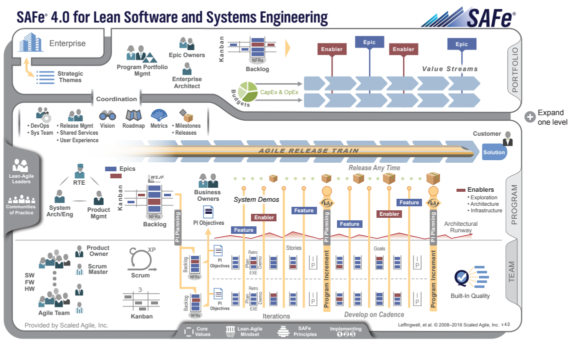
The Scaled Agile Framework (SAFe) is a freely revealed knowledge base of proven, integrated patterns for enterprise-scale Lean-Agile development. It is scalable and modular, allowing each organization to apply it in a way that provides better business outcomes and happier, more engaged employees.

SAFe synchronizes alignment, collaboration, and delivery for large numbers of Agile Teams. It supports both software solutions and complex cyber-physical systems that require thousands of people to create and maintain. SAFe was developed in the field, based on helping Customers solve their most challenging scaling problems. SAFe leverages three primary bodies of knowledge: Agile development, Lean product development and flow, and systems thinking.

Overview

The SAFe website (www.scaledagileframework.com) provides comprehensive guidance for scaling development work across all levels of an enterprise. SAFe's interactive "Big Picture" (Figure 1) provides a visual overview of the framework. Each icon on the website is selectable, navigating the user to an article that provides extensive guidance on the topic area, along with links to related articles and further information.

The Big Picture has two views. The default "3-level view" (below left) is well suited for solutions that require a modest number of Agile Teams. The "4-level view" (below right) supports those building large solutions that typically require hundreds or more practitioners to construct and maintain.



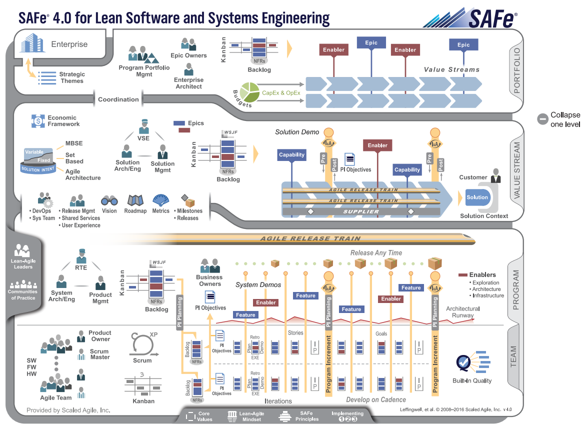

SAFe® 4.0 for Lean Software and Systems Engineering
Enterprise
Solution Demo
SUPPLIER
VALUE STREAM
PROGRAM
TEAM
Collapse one level
Provided by Scaled Agile, Inc.

Figure 1. Big Picture: 3-level and 4-level SAFe

SAFe provides three, and optionally four, organization levels, as well as a foundation, as follows:

- **Team Level** – SAFe is based on Agile Teams, each of which is responsible for defining, building, and testing stories from their backlog. Teams employ Scrum or Kanban methods, augmented by quality practices, to deliver value in a series of synchronized, fixed-length iterations.
- **Program Level** – SAFe teams are organized into a virtual program structure called the "Agile Release Train" (ART). Each ART is a long-lived, self-organizing team of 5 to 12 Agile Teams—along with other stakeholders—that plan, commit, execute, inspect and adapt, and deliver solutions together.
- **Value Stream Level** – The optional value stream level supports the development of large and complex solutions. These solutions require multiple, synchronized ARTs, as well as stronger focus on solution intent and solution context. Suppliers and additional stakeholders contribute as well.
- **Portfolio Level** – The portfolio level organizes and funds a set of value streams. The portfolio provides solution development funding via Lean-Agile budgeting and provides necessary governance and value stream coordination.
- **Foundation Layer** – The foundation layer holds various additional elements that support development. Elements include guidance for Lean-Agile Leaders, communities of practice, core values, the Lean-Agile mindset, the nine Lean-Agile principles that guide SAFe, and an overview of implementation strategy.

Foundation Layer

The foundation layer of SAFe (the shadow backdrop on the big picture) contains the aspects of SAFe that are critical, necessary, and supportive of value delivery, but are not specific practices. This layer contains the following:

- **Lean Agile Leaders** – The ultimate responsibility for the success of the enterprise, and thereby any significant change to the way of working, lies with management. To this end, SAFe describes a new style of leadership, one that is exhibited by SAFe's *Lean-Agile Leaders*.
- **Communities of Practice** – The Lean approach to aligning around Value Streams typically causes the Lean enterprise to pivot from a functional organization to a more flexible and adaptive line-of-business approach. In response, SAFe also supports communities of practice, informal groups of team members and other experts who share practical, functional knowledge in one or more relevant domains.
- **Core Values** – There are four primary *core values* that help make SAFe effective: Alignment, Built-in Quality, Transparency, and Program Execution.

- **Lean-Agile Mindset** – SAFe Lean-Agile Leaders are lifelong learners and teachers who understand and embrace Lean and Agile principles and practices, and teach them to others. To achieve that effectively, leaders must first be trained in, and then become trainers of, these leaner ways of thinking and operating. This mindset is exhibited in part by the House of Lean and the Agile Manifesto.
- **Lean-Agile Principles** – SAFe's practices are grounded on nine fundamental principles that have evolved from Agile principles and methods, Lean product development, systems thinking, and observation of successful enterprises. These are:

  #1 – Take an economic view

  #2 – Apply systems thinking

  #3 – Assume variability; preserve options

  #4 – Build incrementally with fast, integrated learning cycles

  #5 – Base milestones on objective evaluation of working systems

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

  #7 – Apply cadence, synchronize with cross-domain planning

  #8 – Unlock the intrinsic motivation of knowledge workers

  #9 – Decentralize decision-making
- **Implementing 1-2-3** – Based on the learnings from hundreds of SAFe implementations, a basic "Implementing SAFe 1-2-3" pattern for successfully adopting SAFe has emerged. The pattern is, first, *train implementers and Lean-Agile change agents* (SPCs). In turn, these external or internal consultants can then *train all executives, managers, and leaders*. After this, SPCs can *train teams and launch Agile Release Trains*.

Team Level

The team level provides an organization, artifact, role, and process model for the activities of Agile Teams, as illustrated in Figure 2.

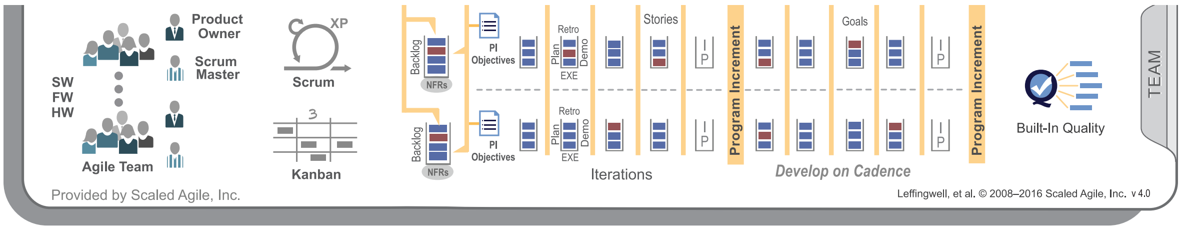


Figure 2. SAFe team level

All SAFe teams are part of one Agile Release Train (ART)—the central construct of the program level. Each Agile Team is responsible for defining, building, and testing stories from their team backlog in a series of fixed-length iterations, using common iteration cadence and synchronization to align with other teams, so that the entire system is iterating.

Teams use Scrum or Team Kanban, along with the Built-in Quality practices, to deliver working software every two weeks. The system demo creates a routine "pull event," which pulls the effort of the different teams together, bringing forward the hard work of integration and testing that phase-gated models often leave until too late in the life cycle.

Each team has five to nine members and includes all the roles necessary to build a quality increment of value in each iteration. Roles include the Scrum Master, Product Owner, dedicated individual contributors, and any specialty resources the team needs to deliver value.

A summary of this level is provided in the "Introduction to the Team Level" overview article.

Program Level

The heart of SAFe is the program level, illustrated in Figure 3, which revolves around the organization called the "Agile Release Train," and which incorporates the team level by reference.

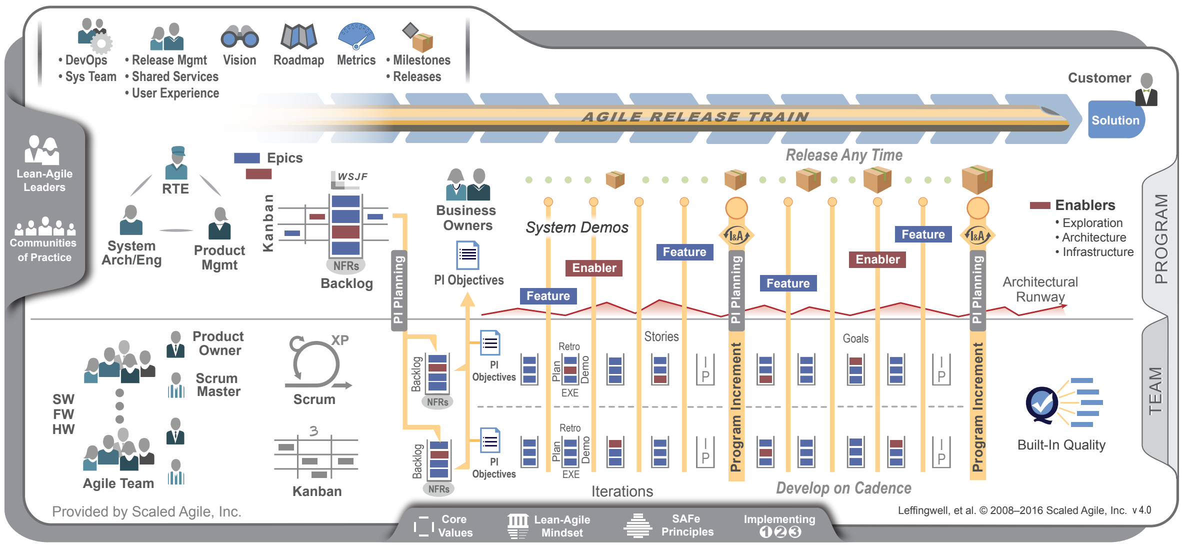


Figure 3. Program level

SAFe program level teams, roles, and activities are organized around the ART metaphor, a team of Agile Teams that delivers a continuous flow of incremental releases of value.

ARTs are virtual organizations formed to span functional boundaries, eliminate unnecessary handoffs and steps, and accelerate the delivery of value via implementation of SAFe Lean-Agile principles and practices.

While it is called the “program level,” ARTs are generally very long-lived and therefore have a more persistent self-organization, structure, and mission than a traditional “program,” which more classically has a start and an end date, as well as temporarily assigned resources. It is the long-lived, knowledge acquiring, flow-based, and self-organizing nature of the ART that powers the SAFe portfolio.

Value in SAFe is delivered by Agile Release Trains, each of which realizes a portion of a value stream (or, in some cases, the entire value stream). They deliver value incrementally in program increments (PIs) of 8 to 12 weeks in duration; each PI is a multiple-iteration timebox during which a significant, valuable increment of the system is developed. Each ART is composed of 5 to 12 Agile Teams (50 – 125+ people) and includes the roles and infrastructure necessary to deliver fully tested, working, system-level solutions. Many release trains are virtual, spanning organizational and geographic boundaries; others follow a line of business or product line management reporting structure.

A summary of this level is provided in the “Introduction to the Program Level” overview article.

The Spanning Palette

There are a number of additional icons indicated on this level; they are located at the conjunction of the value stream and program level on the Big Picture. This is called the “spanning palette” and is illustrated in Figure 4.

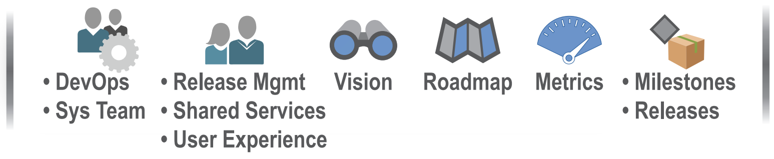


Figure 4. The spanning palette

Each of these artifacts and roles contributes to the ART and program level, as described in the “Vision,” “Roadmap,” “Metrics,” “Milestones,” “Releases,” “DevOps,” “System Team,” “Release Management,” “Shared Services,” and “User Experience” articles. However, these elements also “span” the levels because many of them are also useful at the other levels.

Value Stream Level

The value stream level is optional in SAFe. Enterprises that build systems that are largely independent, or that can be built with a few hundred practitioners, may not need these constructs, and in that case the portfolio can operate with the 3-level view. Even then, however, those are far from trivial systems, and the constructs at the value stream level can be used in 3-level SAFe as needed.

The value stream level helps enterprises that face the largest systems challenges: those building large-scale, multidisciplinary software and cyber-physical systems. Building such solutions in a Lean-Agile manner requires additional constructs, artifacts, and coordination. The constructs of the value stream level are illustrated in Figure 5.

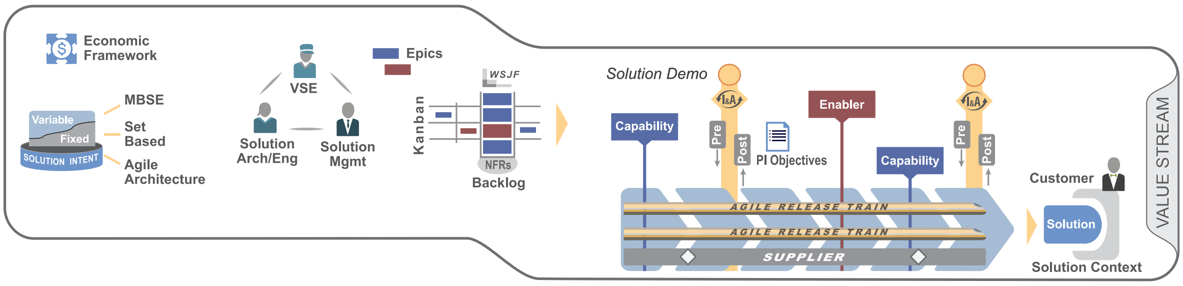


Figure 5. Value stream level

This level contains an economic framework, intended to provide financial boundaries for value stream and ART decision-making; solution intent as a repository for intended and actual solution behavior; solution context, which describes the way the solution fits in the deployment environment; and capabilities, describing the larger behaviors of the solution.

Like the program level, the value stream level is organized around program increments, which are synchronized across all the ARTs in the value stream. It provides for cadence and synchronization of multiple ARTs and Suppliers, including pre- and post-PI planning meetings and the solution demo. It also provides additional roles, specifically Solution Management, Solution Architect/Engineering, and the Value Stream Engineer.

A summary of this level may be found in the "Introduction to the Value Stream Level" overview article.

Portfolio Level

The SAFe portfolio is the highest level of concern in SAFe. As illustrated in Figure 6, each SAFe portfolio has the value streams, people, and processes necessary to provide funding and governance for the products, services, and solutions required to fulfill the overall business strategy.

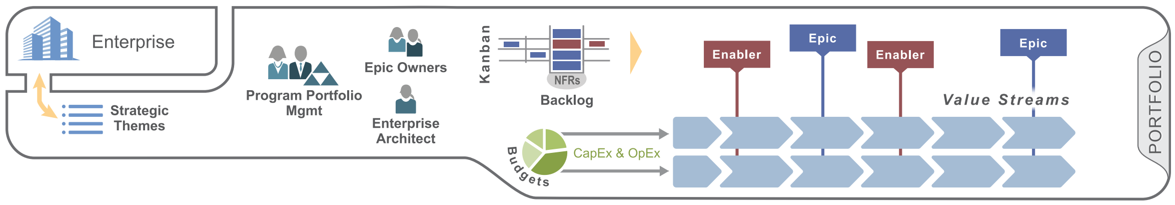


Figure 6. SAFe portfolio level

It provides the basic constructs for organizing the Lean-Agile Enterprise around the flow of value via one or more value streams, each of which develops the systems and solutions necessary to meet the strategic intent. The portfolio level encapsulates these elements and also provides the basic budgeting and other governance mechanisms that are necessary to ensure that the investment in the value streams provides the returns necessary for the enterprise to meet its strategic objectives.

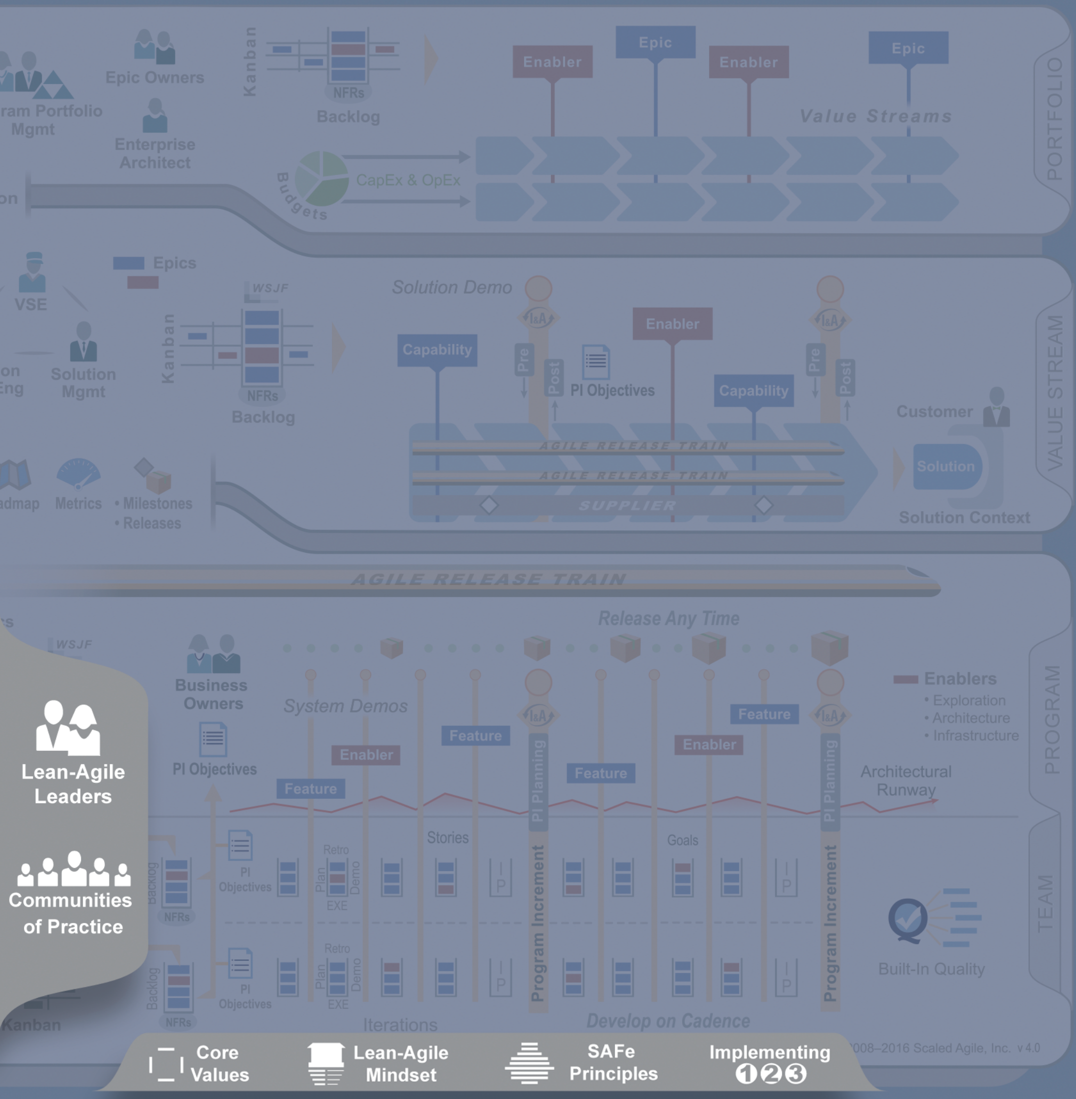
The portfolio has a bidirectional connection to the business. One direction provides the strategic themes that guide the portfolio to the larger, and changing, business objectives. The other direction indicates a constant flow of portfolio context back to the enterprise.

The primary elements of the portfolio are value streams (one or more), each of which provides funding for the people and other resources necessary to build the solutions that deliver the value. Each value stream is a long-lived series of system definition, development, and deployment steps used to build and deploy systems that provide a continuous flow of value to the business or Customer. Program Portfolio Management represents the stakeholders who are accountable to deliver the business results.

A summary of this level may be found in the “Introduction to the Portfolio Level” overview article.

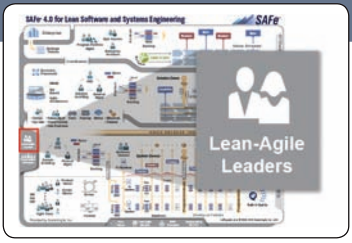
Part 1
The SAFe Foundation

[scaledagileframework.com



This page intentionally left blank

Lean-Agile Leaders



A leader is one who knows the way, goes the way, and shows the way.
—John C. Maxwell

Abstract

The philosophy of SAFe is simple: As the enabler for the teams, the ultimate responsibility for adoption, success, and ongoing improvement of Lean-Agile development lies with the Enterprise's existing managers, leaders, and executives. Only they can change and continuously improve the systems in which everyone operates. To achieve this, leaders must be trained, and become trainers, in these leaner ways of thinking and operating. Many need to offer a new style of leadership, one that truly teaches, empowers, and engages individuals and teams to reach their highest potential.

While some of these management roles and titles do not appear specifically on the Big Picture, they serve a critical function nonetheless by providing the personnel, resources, management, direction, and support necessary to help the enterprise achieve its mission. This article describes the principles of these *Lean-Agile Leaders*.

Details

SAFe *Lean-Agile Leaders* are lifelong learners and teachers who help teams build better systems through understanding and exhibiting the Lean-Agile Mindset, SAFe Principles, and systems thinking. Such leaders exhibit the behaviors below.

#1 – Lead the Change

The work of steering an organization toward Lean and Agile behaviors, habits, and results cannot be delegated. Rather, Lean-Agile Leaders exhibit urgency for change, communicate the need for the change, build a plan for successful change, understand and manage the change process, and address problems as they come up. They have knowledge of organizational change management and take a systems view with respect to implementing the transformation.

#2 – Know the Way; Emphasize Lifelong Learning

Create an environment that promotes learning. Encourage team members to build relationships with Customers and Suppliers and expose them to other world views. Strive to learn and understand new developments in Lean, Agile, and contemporary management practices. Create and foster formal and informal groups for learning and improvement. Read voraciously from the recommended reading list and on other topics. Share selected readings with others and sponsor book club events for the most relevant texts.

Allow people to solve their own problems. Help them identify a given problem, understand the root causes, and build solutions that will be embraced by the organization. Support individuals and teams when they make mistakes, otherwise learning is not possible.

#3 – Develop People

Employ a Lean leadership style, one that focuses on developing skills and career paths for team members rather than on being a technical expert or coordinator of tasks. Create a team jointly responsible for success. Learn how to solve problems together in a way that develops people's capabilities and increases their engagement and commitment. Respect people and culture.

#4 – Inspire and Align with Mission; Minimize Constraints

Provide mission and vision, with minimum specific work requirements. Eliminate demotivating policies and procedures. Build Agile Teams and trains organized around value. Understand the power of self-organizing, self-managing teams. Create a safe environment for learning, growth, and mutual influence. Build an Economic Framework for each Value Stream and teach it to everyone.

#5 – Decentralize Decision-Making

(See "SAFe Principle #9" for further discussion.)

Establish a decison-making framework. Empower others by setting the mission, developing people, and teaching them to problem-solve. Take responsibility for making and communicating strategic decisions—those that are infrequent, long lasting, and have significant economies of scale. Decentralize all other decisions.

#6 – Unlock the Intrinsic Motivation of Knowledge Workers

(See "SAFe Principle #8" for further discussion.)

Understand the role that compensation plays in motivating knowledge workers. Create an environment of mutual influence. Eliminate any and all management by objectives (MBOs) that cause internal competition. Revamp personnel evaluations to support Lean-Agile principles and values. Provide purpose and autonomy; help workers achieve mastery of new and increasing skills.

Role of the Development Manager

As an instantiation of the principles of Lean and Agile development, SAFe emphasizes the values of nearly autonomous, self-organizing, cross-functional teams and Agile Release Trains. This supports a leaner management infrastructure, with more empowered individuals and teams and faster, local decision-making. Traditional, day-to-day employee instruction and activity direction is no longer required.

However, all employees still need someone to assist them with career development; set and manage expectations and compensation; and provide the active coaching they need to advance their technical, functional, individual, and team skills and career goals. They also have a right to serve as an integral member of a high-performing team.

In addition, self-organizing ARTs do not fund themselves or define their own mission. That remains a management responsibility, as it is an element of implementation of strategy.

Much of this responsibility traditionally falls to the traditional role of the *development manager*, and the adoption of Lean-Agile development does not abrogate their responsibilities. However, in SAFe these responsibilities fall to those who can adapt, thrive, and grow in this new environment.

Responsibilities

The development manager (or engineering manager for system development) is a manager who exhibits the principles and practices of Lean-Agile leadership as described above. Further, the manager has personal responsibility for the coaching and career development of direct reports, takes responsibility for eliminating impediments, and actively evolves the systems in which all knowledge workers operate. They have final accountability for effective value delivery as well. A summary of responsibilities is highlighted below.

Personnel and Team Development

- Attract, recruit, and retain capable individuals
- Build high-performing teams; establish mission and purpose for individuals and teams
- Perform career counseling and personal development
- Listen and support teams in problem identification, root cause analysis, and decision-making
- Participate in defining and administering compensation, benefits, and promotions
- Eliminate impediments and evolve systems and practices in support of Lean-Agile development
- Take subtle control in assignment of individuals to teams; address issues that teams cannot unblock; make personnel changes where necessary

- Evaluate performance, including team input; provide input, guidance, and corrective actions
- Serve as Agile coach and advisor to Agile Teams
- Remain close enough to the team to add value and to be a competent manager; stay far enough away to let them problem-solve on their own

Program Execution

- Help in building Agile Milestones and Roadmaps, as well as the building plans that enable them
- Help develop, implement, and communicate the economic framework
- Participate in Inspect and Adapt workshops
- Protect teams from distractions and unrelated or unnecessary work
- Assist the Release Train and Value Stream Engineers with PI Planning readiness and Pre- and Post-PI Planning activities
- Participate in PI planning, System Demo, and Solution Demo
- Build partnerships with Suppliers, subcontractors, consultants, partners, and internal and external stakeholders
- Provide other resources as necessary for teams and ARTs to successfully execute their Vision and roadmap

Alignment

- Work with Release Train and Value Stream Engineers and system stakeholders to help ensure alignment and effective execution of Strategic Themes
- Work with the System Architect/Engineer, Product Managers, and Product Owners to establish clear content authority
- Continuously assist in aligning teams to the system mission and vision
- Help ensure the engagement of Business Owners, Shared Services, and other stakeholders

Transparency

- Create an environment where the *facts are always friendly*
- Provide freedom and safety so individuals and teams are free to innovate, experiment, and even fail on occasion
- Communicate openly and honestly with all stakeholders
- Keep backlogs and information radiators fully visible to all
- Value productivity, quality, transparency, and openness over internal politics

Built-in Quality

- Understand, teach, or sponsor technical skills development in support of high-quality code, components, systems, and Solutions
- Foster Communities of Practice
- Understand, support, and apply Agile Architecture

LEARN MORE

[1] Manifesto for Agile Software Development. http://agilemanifesto.org/.

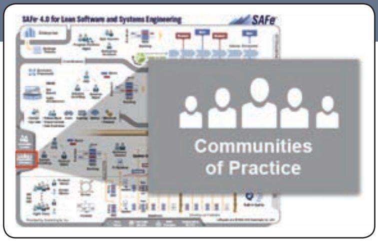
[2] Reinertsen, Donald. *The Principles of Product Development Flow: Second Generation Lean Product Development.* Celeritas Publishing, 2009.

[3] Rother, Mike. *Toyota Kata: Managing People for Improvement, Adaptiveness, and Superior Results*. McGraw-Hill, 2009.

[4] Liker, Jeffrey and Gary L. Convis. *The Toyota Way to Lean Leadership: Achieving and Sustaining Excellence Through Leadership Development*. McGraw-Hill, 2011.

This page intentionally left blank

Communities of Practice



It's said that a wise person learns from his mistakes. A wiser one learns from others' mistakes. But the wisest person of all learns from others' successes.

—Zen proverb adapted by John C. Maxwell

Abstract

A *Community of Practice* (CoP) is an informal group of team members and other experts, acting within the context of a program or enterprise, that has a mission of sharing practical knowledge in one or more relevant domains. CoPs are not new, nor are they mandated by Agile development. However, the Lean approach to aligning around Value Streams optimizes for delivery of value, which is a good thing. Over time, this typically causes the Lean Enterprise to pivot from a vertical, *functional* organization to a more flexible, *horizontal line-of-business* organization that can deliver value more rapidly. Further, within value streams, SAFe promotes long-lived Agile Release Trains (ARTs), which are built of people allocated to them for an extended period. What happens when practitioners of a discipline (whether or not their organization has become horizontal), who are from different programs but often have the same reporting structure, meet regularly, are led by managers and experts from their domain, and advance their specialist skills? Enter the SAFe CoP (*Guild*, in Spotify terminology [1]).

Details

Lean-Agile promotes cross-functional teams and programs that facilitate value delivery in the Enterprise. Similarly, Lean thinking emphasizes organizing people with different skills around a Value Stream. However, developers need to talk with other developers within or outside of the team context, testers need to talk with other testers, Product Owners need to communicate with their peers from other Agile Teams, and so on. This is critical for leveraging the multiple experiences and different types of practical knowledge available from different people at scale. That is what drives craftsmanship and persistent knowledge acquisition and facilitates the adoption of new methods and techniques.

Such inter-team communication is often supported by *Communities of Practice* (CoP)—informal working groups designed specifically for efficient knowledge-sharing and exploration across teams and groups of professionals, as shown in Figure 1.

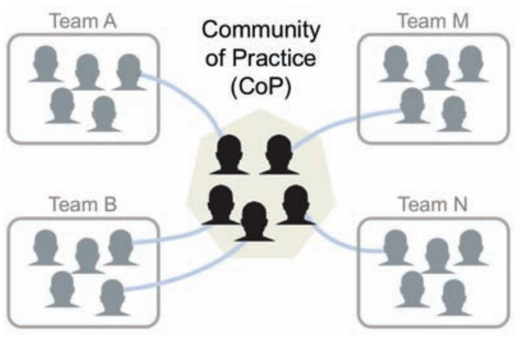


Figure 1. Community of practice: Members normally work in their Agile Teams but also regularly share best practices

Organizing a Community of Practice

A Lean enterprise has the ability to identify the relevant domains where communities of practice would be beneficial and then to foster and support such communities once they are in place.

CoPs can be ad hoc and need driven. They may or may not be permanent; they may form and disband based on current need and context. For example, an automated testing CoP could be composed of test engineers and developers who are interested in advancing these skills. An architecture and design CoP would foster the adoption of practices such as emergent design, intentional system architecture, Continuous Integration, and refactoring. It could also support the effort put into building and maintaining the Architectural Runway, foster designing for testability and deployability, deprecate old platforms, and more. Still others may be formed around Agile coaching, continuous integration, continuous delivery, coding standards, and other new practices and processes. Similarly, Scrum Masters from different Agile Teams may form a CoP to exchange facilitation best practices and experiences in building highly productive Agile Teams.

Operating a Community of Practice

A CoP is defined by the knowledge specialization of its members. Each typically has a specific learning objective, Roadmap, and backlog. Membership is fluid and changes as members take on different roles, as new needs arise, or as individual members gain the knowledge they need. CoPs may be fostered or initiated spontaneously. They are largely self-organizing, although a leader or Scrum Master equivalent (the *Guild Coordinator* in Spotify terminology) may organize the initiative and help maintain its momentum. CoPs meet regularly for knowledge-exchange sessions and maintain and evolve internal community websites and wikis to institutionalize their knowledge. The CoP exists only for so long as the members believe they have something to learn or contribute.

However, CoPs are created for the purpose of learning and exchanging experiences, not for coordinating dependencies or current tasks.

For instance, the Scrum Master CoP would foster learning new facilitation techniques, while actual coordination and dependency management for current work in process would happen among the same people during the Scrum of Scrums.

The Innovation and Planning Iteration presents a great time for CoPs to hold learning sessions, formal or informal, as well as other activities such as coding dojos, coaching clinics, and the like.

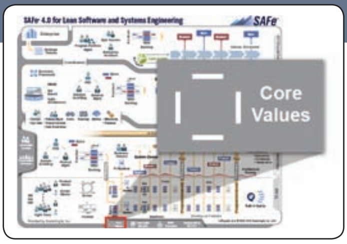
It is the role of Lean-Agile Leaders to encourage and support people's desire to improve as this both helps the enterprise and builds the intrinsic motivation of knowledge workers, as is evident in SAFe Principle #9–*Decentralize decision-making*.

LEARN MORE

[1] Scaling Agility @ Spotify with Tribes, Squads, Chapters, and Guilds. https://dl.dropboxusercontent.com/u/1018963/Articles/SpotifyScaling.pdf.

This page intentionally left blank

SAFe Core Values



Find people who share your values, and you'll conquer the world together.
—John Ratzenberger

Abstract

Core Values are the fundamental beliefs of a person or organization. The core values are the guiding principles that dictate behavior and action. Core values can help people to know what is right from wrong; where to put their focus and help companies to determine if they are on the right path and fulfilling their business goals; and they create an unwavering and unchanging guide.

A Lean-Agile Mindset, Lean-Agile Leaders, SAFe Principles, and the extensive benefits that Lean-Agile development provides all play important roles in defining what makes SAFe safe. But in synthesis, there are four *Core Values* that SAFe honors, supports, and helps deliver: *Alignment, Built-in Quality, Transparency,* and *Program Execution*. If an Enterprise does those four things well, a lot of goodness will surely follow.

Details

SAFe is broad and deep and is based on both Lean and Agile principles. That's what it's built on, but what does SAFe itself stand for? SAFe upholds four *Core Values: Alignment, Built-in Quality, Transparency,* and *Program Execution*.

These are illustrated in Figure 1, and each is discussed in the paragraphs that follow.

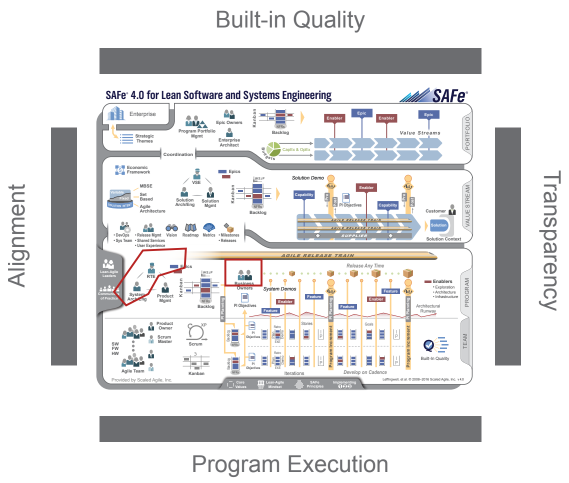


Figure 1. SAFe core values: alignment, built-in quality, transparency, program execution

Alignment

Like cars out of alignment, misaligned companies can develop serious problems. They are hard to steer and they don't respond well to changes in direction [1]. Even if it's clear where everyone thinks they're headed, the vehicle is unlikely to get them there.

Alignment scales. It is a necessary condition to be able to address the business reality of fast-paced change, turbulent competitive forces, and geographically distributed teams. While empowered Agile Teams are good (even great), the responsibility for strategy and alignment cannot rest with the accumulated opinions of the teams, no matter how good they are. Rather, alignment must be based on the Enterprise business objectives. Here are some of the ways in which SAFe supports alignment:

- It starts at the strategy level of the portfolio, is reflected in Strategic Themes and the Portfolio Backlog, and then moves down through the Vision, Roadmap, and Program Backlogs to the Team Backlogs. All is visible. All is debated. All is resolved. All is known.
- It is supported by clear lines of content authority, starting at the portfolio and then resting primarily with the Product and Solution Management roles, and extending to the Product Owner role
- PI Objectives and Iteration Goals are used to communicate expectations and commitments
- Cadence and synchronization are applied to ensure that things stay in alignment, or that they drift only within reasonable economic and time boundaries

- Program architecture, User Experience guidance, and governance help ensure that the Solution is technologically sound, robust, and scalable
- Lean prioritization keeps the stakeholders engaged in continuous, agreed-to, rolling-wave prioritization, based on the then-current context and changing fact patterns

Alignment, however, does not imply or encourage command and control. Instead, it provides a foundation for the enterprise where business objectives and outcomes are the continued focus. It also encourages decentralized technical and economic decision-making, thereby enabling those who implement value to make better local decisions.

Built-in Quality

Built-in Quality ensures that every increment of the solution reflects quality standards. Quality is not "added later." Built-in quality is a prerequisite of Lean and flow; without it, the organization will likely operate with large batches of unverified, unvalidated work. Excessive rework and slower velocities are the likely outcome. There can be no ambiguity about the importance of built-in quality in large-scale systems. It is mandatory.

Software
In complex solutions, *software* functionality often represents a fast-changing and increasingly high-investment area. In addition, given the high levels of complexity and the manual nature of much of the work, it is often the source of many solution defects. The relatively lower cost of change encourages rapid adaptation, which is good. But if attention is not paid, the software design may quickly erode, negatively affecting quality and velocity.

Put simply, *you can't scale crappy code*. The Agile Manifesto certainly focused on quality: "Continuous attention to technical excellence and good design enhances agility" [2]. To address software quality in the face of rapid change, software practitioners have developed and evolved a number of effective practices, many of which are largely inspired by eXtreme Programming. These include:

- Test-First: Test-Driven Development (TDD), Acceptance Test-Driven Development (ATDD), and Behavior-Driven Development (BDD)
- Continuous Integration
- Refactoring
- Pair work
- Collective ownership

Hardware
But coding aside, no one can scale crappy components or systems, either. Hardware elements—electronics, electrical, fluidics, optics, mechanical, packaging, thermal, and many more—are a lot less "soft." Errors here can introduce a much higher cost of change and rework. Tips to avoid this include:

- Frequent design cycles and integration [3]
- Collaborative design practices
- Model-Based Systems Engineering
- Set-Based Design
- Investment in development and test infrastructure

System Integration
Eventually, different components and subsystems—software, firmware, hardware, and everything else—must collaborate to provide effective solution-level behaviors. Practices that support solution-level quality include:

- Frequent system and solution-level integration
- Solution-level testing of functional and Nonfunctional Requirements
- System and Solution Demos

Transparency

Solution development is hard. Things go wrong or do not work out as planned. Without transparency, facts are obscure and hard to come by. This results in decisions based on speculative assumptions and lack of data. No one can fix a secret.

For that *trust* is needed, because without trust no one can build high-performing teams and programs, or build (or rebuild) the confidence needed to make and meet reasonable commitments. Trust exists when one party can confidently rely on another to act with integrity, particularly in times of difficulty. And without trust, working environments are a lot less fun and motivating.

Building trust takes time. Transparency is the enabler for trust. SAFe helps an enterprise achieve transparency:

- Executives, Portfolio Managers, and other stakeholders are able to see into the Portfolio Kanbans and program backlogs, and they have a clear understanding of the PI goals for each train
- Programs have visibility into the team's backlogs, as well other program backlogs
- Teams and programs commit to short-term, clear, and visible commitments. They routinely meet them.

- Programs Inspect and Adapt with all relevant stakeholders; lessons learned are incorporated.
- Teams and programs have visibility into business and architecture Epic Kanban systems. They can see what might be headed their way.
- Status reporting is based on objective measures of working systems
- Everyone can understand the velocity and WIP of the teams and programs; strategy and the ability to execute are aligned

Program Execution

Of course, none of the rest of SAFe matters if teams can't execute and continuously deliver value. Therefore, SAFe places an intense focus on working systems and resultant business outcomes. This isn't only for the obvious reasons. History shows us that while many enterprises start the transformation with Agile Teams, they often become frustrated as even those teams struggle to deliver larger amounts of solution value reliably and efficiently.

That is the purpose of the Agile Release Train, and that is why SAFe focuses implementation initially at the Program Level. In turn, the ability of Value Streams to deliver value depends on the ability of the ARTs.

But with *alignment*, *transparency,* and *built-in quality* on the team's side, they have a little "wind at their back." That enables a focus on *execution*. And if they struggle—and they will, because complex solution development is *hard*—they have the cornerstone of the inspect and adapt workshops. In that way, they close the loop and execute better and better during each Program Increment.

But program execution can't just be a team-based, bottom-up thing. Successful Lean-Agile execution at scale requires not just the teams but the active support of their Lean-Agile Leaders, who couple their internal leadership with an orientation toward system and Customer outcomes. That creates a persistent and meaningful context for the teams and their stakeholders.

That's the way the successful teams and programs are doing it, and that's why they are getting the many benefits—employee engagement, productivity, quality, and time to market—that Lean-Agile enterprises so enjoy.

LEARN MORE

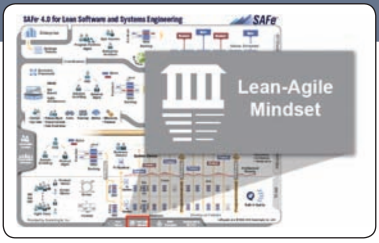
[1] Labovitz, George H. and Victor Rosansky. *The Power of Alignment: How Great Companies Stay Centered and Accomplish Extraordinary Things*. Wiley, 1997.

[2] AgileManifesto.org.

[3] Oosterwal, Dantar P. *The Lean Machine: How Harley-Davidson Drove Top-Line Growth and Profitability with Revolutionary Lean Product Development*. Amacom, 2010.

This page intentionally left blank

Lean-Agile Mindset



It all starts with a Lean-Agile Mindset.
—SAFe Authors

Abstract

SAFe is based on a number of newer paradigms in modern systems and software engineering, including Lean and systems thinking, product development flow, and Agile development. As reflected at the Team Level, Agile provides the tools needed to empower and engage teams to achieve unprecedented levels of productivity, quality, and engagement. But a broader and deeper *Lean-Agile Mindset* is needed to support Lean and Agile development at scale across the entire Enterprise.

Thinking Lean
Much of the thinking in Lean is represented in the SAFe "House of Lean" icon. It is organized around six key constructs. The "roof" represents the goal of delivering Value, the "pillars" support that goal via Respect for People and Culture, Flow, Innovation, and Relentless Improvement. Lean-Agile Leadership provides the foundation on which everything else stands.

Embracing Agility
In addition, SAFe is built entirely on the skills, aptitude, and capabilities of Agile Teams and their leaders. And while there is no one definition of what an Agile method is, the *Agile Manifesto* provides a unified value system that has helped inaugurate Agile methods into mainstream development.

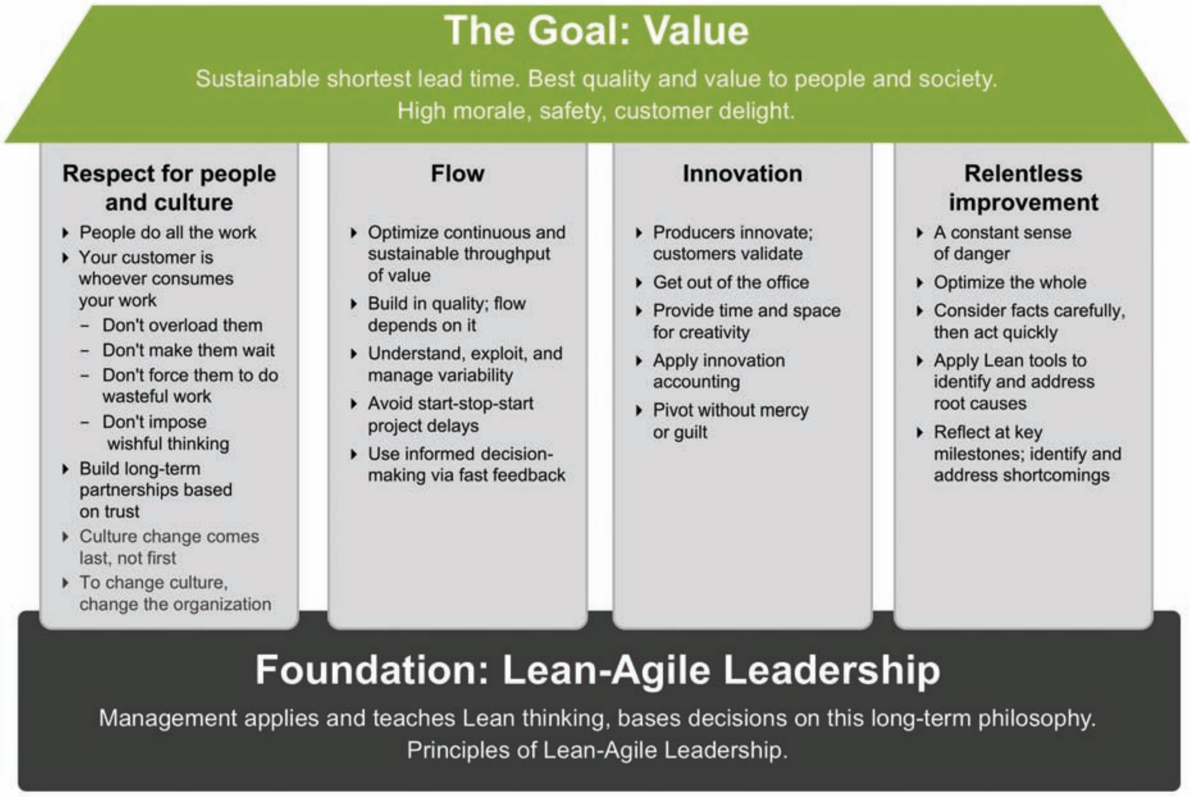
Together, these create the *Lean-Agile Mindset*, part of a new management approach and an enhanced culture, one that provides the leadership needed to drive a successful transformation, and one that helps both individuals and businesses achieve their goals.

Details

The SAFe House of Lean

While initially derived from Lean manufacturing [1], the principles and practices of Lean thinking as applied to software, product, and systems development are now deep and extensive.

For example, Ward [2], Reinertsen [3], Poppendieck [4], Leffingwell [5], and others have described aspects of Lean thinking that put many of the core principles and practices into a product development context. In combination of these factors, we present the *SAFe House of Lean*, as illustrated in Figure 1, which is inspired by "houses" of Lean from Toyota and others.



The Goal: Value
Sustainable shortest lead time. Best quality and value to people and society.
High morale, safety, customer delight.
Respect for people and culture
People do all the work
Your customer is whoever consumes your work
Don't overload them
Don't make them wait
Don't force them to do wasteful work
Don't impose wishful thinking
Build long-term partnerships based on trust
Culture change comes last, not first
To change culture, change the organization
Flow
Optimize continuous and sustainable throughput of value
Build in quality; flow depends on it
Understand, exploit, and manage variability
Avoid start-stop-start project delays
Use informed decision-making via fast feedback
Innovation
Producers innovate; customers validate
Get out of the office
Provide time and space for creativity
Apply innovation accounting
Pivot without mercy or guilt
Relentless improvement
A constant sense of danger
Optimize the whole
Consider facts carefully, then act quickly
Apply Lean tools to identify and address root causes
Reflect at key milestones; identify and address shortcomings
Foundation: Lean-Agile Leadership
Management applies and teaches Lean thinking, bases decisions on this long-term philosophy.
Principles of Lean-Agile Leadership.

Figure 1. The SAFe House of Lean

The Goal – Value

The goal of Lean is inarguable: to deliver the *maximum Customer value in the sustainably shortest lead time*, while providing the highest possible quality to Customers and society as a whole. High morale, safety, and Customer delight are further tangible targets and benefits.

Pillar 1 – Respect for People and Culture

SAFe is a systematic framework for implementing Lean-Agile development at scale, but it does not instantiate itself, nor does it perform any real work. *People do all the work*. Respect for people and culture is a fundamental value of the SAFe House of Lean. People are empowered to evolve their own practices and improvements.

Management challenges people to change and may even indicate what to improve, but the teams and individuals learn problem-solving and reflection skills, and they make the appropriate improvements.

Culture is the driving force behind this behavior. To evolve a truly Lean organization, the culture will need to change. In order for that to happen, the organization and its leaders must change first. And culture and people are not solely an internal construct. The culture of the organization extends to long-term relationships with Suppliers, partners, Customers, and the broader community that supports the Enterprise.

Where there is urgency for positive change, improvements in culture can be achieved gradually by, first, understanding SAFe values and principles; second, implementing SAFe practices; and third, delivering positive results. Changes to culture will follow naturally.

Pillar 2 – Flow

The key to successful execution in SAFe is establishing a continuous flow of work that supports incremental value delivery, based on continuous feedback and adjustment. Establishing continuous flow is critical to fast value delivery; effective quality practices; continuous improvement; and effective, evidence-based governance. The principles of flow, reflected in this pillar of the House of Lean, constitute an important subset of the SAFe Lean-Agile Principles and are instantiated in various practices throughout. These include understanding the full Value Stream, visualizing and limiting WIP, reducing batch sizes and managing queue lengths, and prioritizing work based on the cost of delay. Lean also has a primary focus on Built-in Quality, fast feedback, and the identification and constant reduction of delays and non-value-added activities.

These constructs provide a pivotal change to a better understanding of the system development process and provide new thinking, tools, and techniques that leaders and teams can use to move from phase-gated processes to more continuous value delivery.

Pillar 3 – Innovation

Flow builds a solid foundation for the delivery of value, but without innovation, both product and process will stagnate. Innovation is a critical part of the SAFe House of Lean. In support of innovation, Lean-Agile Leaders:

- "Get out of the office" and into the actual workplace where value is produced and products are created and used (*gemba*). As Taiichi Ohno put it, "No useful improvement was ever invented at a desk."
- Provide time and space for people to be creative. Time for innovation must be purposeful. Innovations can rarely occur in the presence of 100% utilization and continuous firefighting. SAFe's Innovation and Planning Iteration is one such opportunity.

- Apply innovation accounting [6]. Establish non-financial, non-vanity Metrics that provide fast feedback on the important elements of the new innovation.
- Validate the innovation with Customers, then *pivot without mercy or guilt* when the hypothesis needs to change

Pillar 4 – Relentless Improvement

The fourth pillar is relentless improvement. With this pillar, the organization is guided to become a learning organization through continuous reflection and relentless improvement. A constant sense of competitive danger drives the learning organization to aggressively pursue improvement opportunities. Leaders and teams do the following systematically:

- Optimize the whole, not the parts, of both the organization and the development process
- Consider facts carefully, then act quickly
- Apply Lean tools and techniques to determine the root cause of inefficiencies and apply effective countermeasures quickly
- Reflect at key Milestones to openly identify and address the shortcomings of the process at all levels

Foundation – Leadership

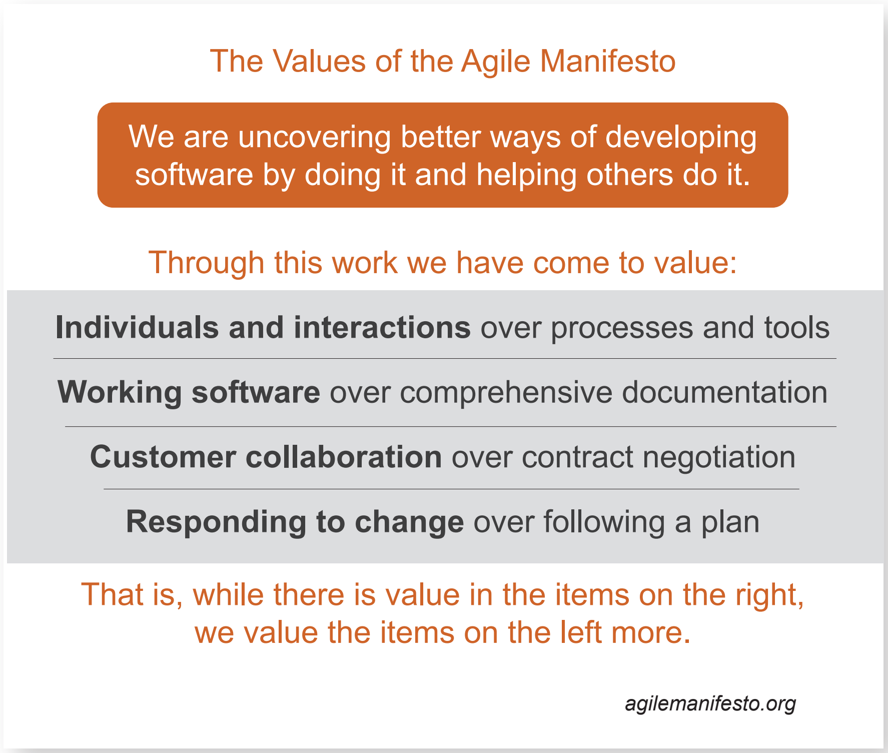
The foundation of Lean is leadership, which is the ultimate enabling force for team success. Here, SAFe's philosophy is simple: *The ultimate responsibility for adoption and success of the Lean-Agile paradigm lies with the enterprise's existing managers, leaders, and executives*. "Such a responsibility cannot be delegated" (Deming [7]) to Lean/Agile champions, Lean/Agile working groups, development teams, a PMO, process teams, outside consultants, or any other party. To achieve success, leaders must be trained in these new and innovative ways of thinking and exhibit the principles and behaviors of Lean-Agile leadership.

Lean thinking deviates from common experience with Agile, which was often introduced as a team-based process that tended to exclude management. That does not scale. Here is a key differentiator between traditional Agile and one of the key drivers for SAFe:

> In traditional Agile, the expectation has been that management simply *supports* the teams and helps eliminate impediments as they arise. In Lean-Agile development, the expectation is that management *leads* the teams, embraces the values of Lean, is competent in the basic practices, proactively eliminates impediments, and takes an active role in driving organizational change and facilitating relentless improvement.

The Agile Manifesto

In the 1990s, in response to the many challenges of waterfall development methods, a number of lighter-weight and more iterative development methods arose. In 2001, many of the leaders of these methods came together in Snowbird, Utah. While there were differences of opinion on the specific merits of one method over another, the attendees agreed that their common values and beliefs dwarfed the differences in approach. The result was a *Manifesto for Agile Software Development* [8], which was a turning point that helped unify the approach and started to bring the benefits of these innovative methods to the industry at large. The Manifesto consists of a value statement, as exhibited in Figure 2, and a set of principles, as exhibited in Figure 3.



The Values of the Agile Manifesto

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

Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

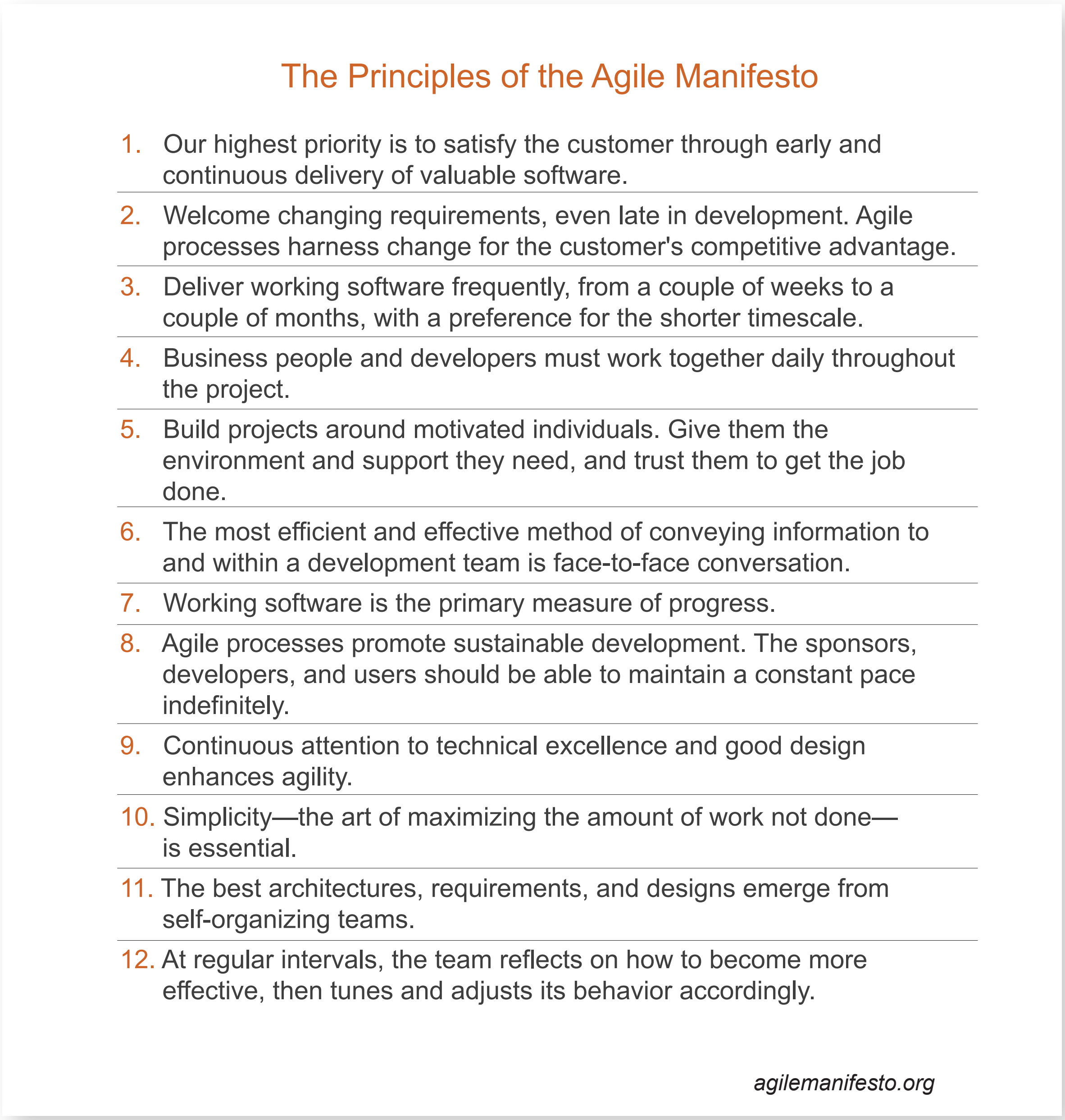
Customer collaboration over contract negotiation

Responding to change over following a plan

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

agilemanifesto.org

Figure 2. Values of the Agile Manifesto



The Principles of the Agile Manifesto

1. Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
2. Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
3. Deliver working software frequently, from a couple of weeks to a couple of months, with a preference for the shorter timescale.
4. Business people and developers must work together daily throughout the project.
5. Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
6. The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.
7. Working software is the primary measure of progress.
8. Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
9. Continuous attention to technical excellence and good design enhances agility.
10. Simplicity—the art of maximizing the amount of work not done—is essential.
11. The best architectures, requirements, and designs emerge from self-organizing teams.
12. At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

agilemanifesto.org

Figure 3. Principles of the Agile Manifesto

Along with the various Agile methods, the Manifesto provides the Agile foundation for effective, empowered, self-organizing teams. SAFe extends this foundation to the level of teams of teams and applies Lean thinking to understand and relentlessly improve the systems that support the teams in their critical work.

LEARN MORE

[1] Womack, James P., Daniel T. Jones, and Daniel Roos. *The Machine That Changed the World: The Story of Lean Production—Toyota's Secret Weapon in the Global Car Wars That Is Revolutionizing World Industry*. Free Press, 2007.

[2] Ward, Allen and Durward Sobeck. *Lean Product and Process Development*. Lean Enterprise Institute, 2014.

[3] Reinertsen, Donald G. *The Principles of Product Development Flow: Second Generation Lean Product Development*. Celeritas, 2009.

[4] Poppendieck, Mary and Tom Poppendieck. *Implementing Lean Software Development: From Concept to Cash*. Addison-Wesley, 2006.

[5] Leffingwell, Dean. *Agile Software Requirements: Lean Requirements Practices for Teams, Programs, and the Enterprise*. Addison-Wesley, 2011.

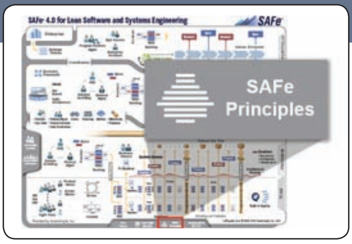
[6] Ries, Eric. *The Lean Startup: How Today's Entrepreneurs Use Continuous Innovation to Create Radically Successful Businesses*. Crown Business, 2011.

[7] Deming, W. Edwards. *Out of the Crisis.* MIT Center for Advanced Educational Services, 1982.

[8] Manifesto for Agile Software Development. http://agilemanifesto.org/.

This page intentionally left blank

SAFe Principles



The impression that "our problems are different" is a common disease that afflicts management the world over. They are different, to be sure, but the principles that will help to improve the quality of product and service are universal in nature.

—W. Edwards Deming

SAFe is based on a number of immutable, underlying Lean and Agile principles. These are the fundamental tenets, the basic truths and economic underpinnings that drive the roles and practices that make SAFe effective. The nine principles are:

#1-Take an economic view

#2-Apply systems thinking

#3-Assume variability; preserve options

#4-Build incrementally with fast, integrated learning cycles

#5-Base milestones on objective evaluation of working systems

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

#7-Apply cadence, synchronize with cross-domain planning

#8-Unlock the intrinsic motivation of knowledge workers

#9-Decentralize decision-making

Why the Focus on Principles?

Building enterprise-class software and cyber-physical systems is one of the most complex challenges the industry faces today. Millions of lines of software, complex hardware and software interactions, multiple concurrent platforms, demanding and unforgiving nonfunctional requirements—these are just a few of the challenges systems builders face.

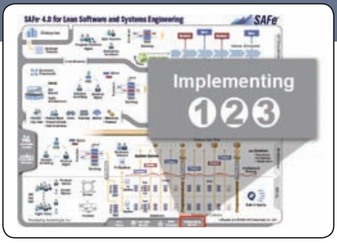
Of course, the enterprises that build these systems are increasingly complex, too. They are bigger and more distributed than ever. Mergers and acquisitions, distributed multinational (and multilingual) development, offshoring, and the rapid growth that success requires are all part of the solution—but also part of the problem as well.

Fortunately, we have an amazing and growing body of knowledge to help us address this challenge. These include Agile principles and methods, Lean and Systems thinking, product development flow, Lean process and product development, and more. Many thought leaders have gone down this path before us and left a trail to follow in the hundreds of books and references we can draw on.

SAFe's goal is to synthesize some of this body of knowledge and the lessons learned from hundreds of deployments into a single framework—a system of integrated, proven practices that has been demonstrated to bring substantial improvements in employee engagement, time to market, solution quality, and team productivity. However, given the complexity of the industry challenges already discussed, there is truly no off-the-shelf solution to the unique challenges every enterprise faces. This means that some tailoring and customization may be required, as not every SAFe-recommended practice will apply equally well in every circumstance. Therefore, we always endeavor to make certain that SAFe practices are grounded in fundamental, and reasonably immutable, principles. In that way, we can be confident that they apply well in the general case. And when and if they don't, the underlying principles can guide those doing the implementation to make sure that they are moving on a continuous path to the "shortest sustainable lead time, with best quality and value to people and society." There is value in that too.

The nine SAFe Principles are discussed in greater detail in the next chapter.

Implementing 1-2-3



It is not enough that management commit themselves to quality and productivity, they must know what it is they must do. Such a responsibility cannot be delegated.

—W. Edwards Deming

Abstract

Implementing the changes necessary to become a Lean-Agile technology enterprise is a substantial change for most organizations. Embracing a Lean-Agile Mindset, understanding and applying the Lean-Agile principles, and effectively implementing the SAFe practices all come *before* the business benefits. And, of course, the culture must evolve, too.

While SAFe is a freely revealed body of knowledge, available to all, it does not implement itself, nor does it prescribe the organizational change management process that is typically required for successful implementation. We leave that to the enterprise, because only they know their specific context, and they—typically assisted by their partners—must own the transformation.

But many enterprises have gone down this path already (see the "Case Studies" articles online at www.scaledagileframework.com), and the lessons learned are now becoming more widely accessible. Based on the learnings from hundreds of SAFe implementations, Scaled Agile, Inc., the owner of SAFe, has developed a basic *Implementing SAFe 1-2-3* pattern for successful SAFe adoption. It provides a simple roadmap that helps gets everyone aligned to a common implementation strategy.

This article describes an overview of this successful pattern for SAFe implementation, along with pointers to the growing community of service providers who are ready and willing to help your enterprise make this critical transformation.

Details

Figure 1 on the next page provides a high-level summary of the *Implementing SAFe 1-2-3* approach. Each of the numbered items in this strategy is described in the paragraphs that follow.

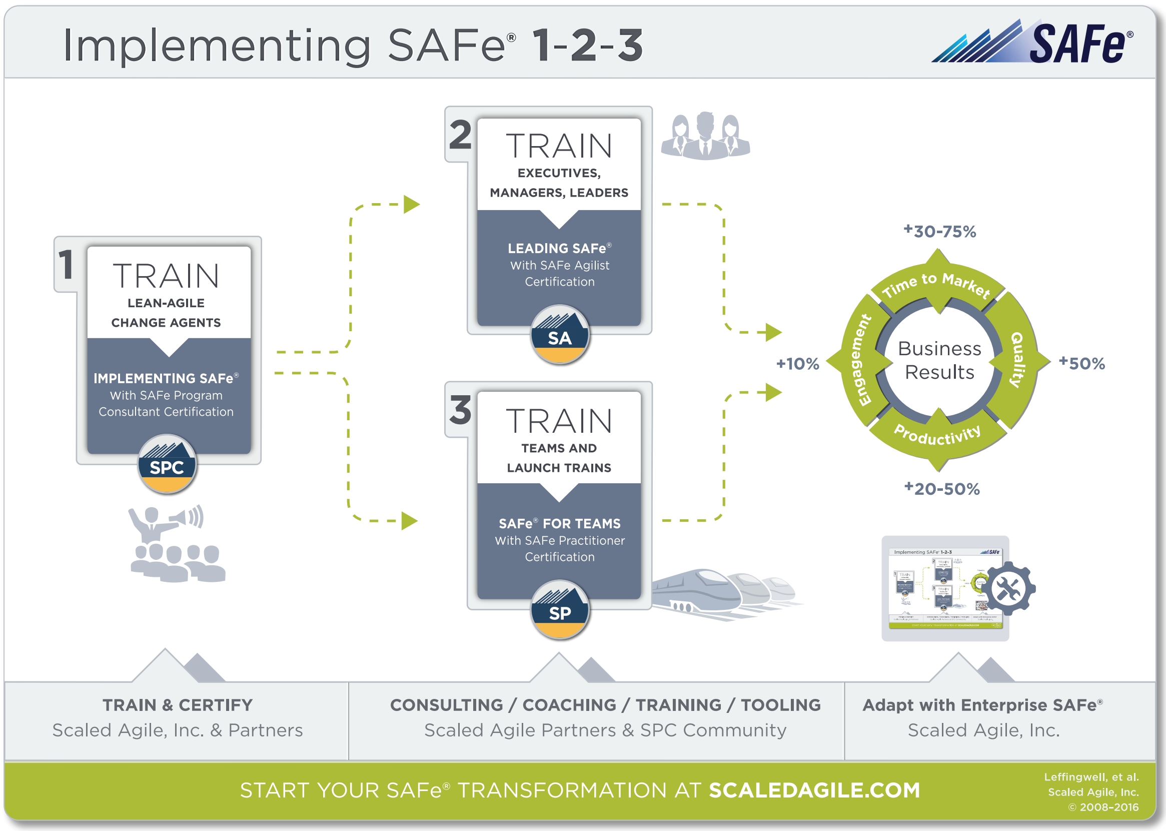

Implementing SAFe® 1-2-3
SAFe®
1 TRAIN LEAN-AGILE CHANGE AGENTS
IMPLEMENTING SAFe® With SAFe Program Consultant Certification
SPC
2 TRAIN EXECUTIVES, MANAGERS, LEADERS
LEADING SAFe® With SAFe Agilist Certification
SA
3 TRAIN TEAMS AND LAUNCH TRAINS
SAFe® FOR TEAMS With SAFe Practitioner Certification
SP
+30-75%
Time to Market
+10%
Engagement
Business Results
Quality
+50%
Productivity
+20-50%
TRAIN & CERTIFY
Scaled Agile, Inc. & Partners
CONSULTING / COACHING / TRAINING / TOOLING
Scaled Agile Partners & SPC Community
Adapt with Enterprise SAFe®
Scaled Agile, Inc.
START YOUR SAFe® TRANSFORMATION AT SCALEDAGILE.COM
Leffingwell, et al. Scaled Agile, Inc. © 2008–2016

Figure 1. Implementing SAFe 1-2-3

1. Train Implementers and Lean-Agile Change Agents

Given the scope, challenge, and impact of rollouts, successful adoption of SAFe requires most enterprises to use a combination of internal and external change agents, leaders, mentors, and coaches. These people need to be skilled in teaching and delivering SAFe. To achieve this, Scaled Agile, Inc. provides an *Implementing SAFe 4.0 with SPC Certification* program. After taking this class, attendees will be able to:

- Lead an enterprise Agile transformation with SAFe
- Implement SAFe
- Launch Agile Release Trains and proctor and continuously improve the trains via Inspect and Adapt workshops

Those who take and pass the optional SPC Certification exam (included) will be licensed to:

- Train managers and executives in Leading SAFe and act as a SAFe Agilist (SA) certifying agent
- Train practitioners in SAFe 4.0 for Teams and act as a SAFe Practitioner (SP) certifying agent

The Implementing SAFe 4.0 with SAFe Program Consultant (SPC4) Certification course is delivered by certified SPC Trainers (SPCT) in open enrollment or on-site settings worldwide. Service providers who specialize in SAFe and Scaled Agile Partners can be found online at www.scaledagile.com. There are many independent SPCs as well; these can be found at the Scaled Agile SPC membership site at www.scaledagileacademy.com.

2. Train all Executives, Managers, and Leaders

It is critical that executives, managers, and leaders understand what is required to lead a Lean-Agile transformation, including how and why SAFe works. To help achieve this, Scaled Agile, Inc., provides a two-day course, *Leading SAFe 4.0, Leading the Lean-Agile Enterprise with the Scaled Agile Framework.* After attending, participants will be able to:

- Adopt a Lean-Agile mindset
- Apply Lean and Agile principles; base daily decisions on this long-term philosophy; understand, exhibit, and teach these principles
- Understand the practices, roles, activities, and artifacts of the Scaled Agile Framework
- Unlock the intrinsic motivation of knowledge workers
- Learn the practices and tools of relentless improvement and teach employees problem-solving and corrective-action skills
- Become hands-on in the new process adoption, eliminate impediments, and facilitate organizational change management
- Take responsibility for Lean-Agile implementation success

The audience for this class is executives, managers, and change agents responsible for leading a Lean-Agile change initiative, whereby they gain the knowledge necessary to lead the SAFe adoption.

A certification exam is optional for this course. Those who pass the optional exam will be certified as SAFe Agilist (SA), and will receive one year's membership to that community and its benefits.

The *Leading SAFe 4.0* course is delivered by certified SPC consultant/trainers in open enrollment or on-site settings worldwide. Service providers include Scaled Agile, Inc.; Scaled Agile Partners; and independent SPCs.

3. Train Teams and Launch Agile Release Trains

The primary value delivery mechanism in the enterprise is the Agile Release Train, but starting these trains is not a trivial task. One proven starting mechanism is an Agile Release Train Quickstart.

Suitable after some significant up-front preparation, the QuickStart is a one-week training and immersion program that:

- Organizes 50 – 100 team members into Agile Teams, training them simultaneously in the principles of Lean, Agile, and SAFe
- Aligns the teams on the train to a common mission and spends two days in face-to-face support of planning the next Program Increment
- Introduces prospective Product Owners and Scrum Masters to the skills and activities unique to their roles in the new Agile enterprise
- Builds context and a cadence-based, rolling-wave planning and delivery model that continuously incorporates business objective setting and program commitments, effective and reliable program execution, and adaptive feedback

SPCs can provide these services and download and use the SAFe ART Launch Pack (member login required) to prepare for a successful launch. It contains the tools to prepare the organization, programs, teams, and individuals for success and continuous improvement. You may also want to consider licensing the ART Training and Launch Pack Bundle. This bundle provides both the courseware and tools needed to quickly and effectively launch Agile Release Trains.

Supporting Consulting Activities

Once the enterprise has a critical mass of in-house Lean-Agile Leaders, and a few Agile Release Trains rolling, a variety of consulting activities may be applicable and beneficial. These could include *coaching the train, training specialist roles,* and *continuous improvement*.

Coaching the Train
By sharing their knowledge and experience, coaches can help teams and individuals improve their newfound skills by:

- Providing program consulting and team coaching to build the organization's Lean-Agile capabilities
- Facilitating Agile Release Train readiness, including Program Backlog refinement and more
- Facilitating inspect and adapt workshops
- Facilitating Portfolio planning workshops
- Implementing relevant Metrics and governance
- Mentoring executives, managers, and other program stakeholders in SAFe adoption
- Shadowing and mentoring Release Train Engineers

Many of the these activities are supported by various Scaled Agile workshop kits, which are available to SPCs in good standing.

Training Specialist Roles
It is important to train specialists—including prospective Product Owners and Scrum Masters—in the principles and practices unique to their roles. Training courses for this purpose include:

- *SAFe 4.0 Scrum Master Orientation* – Half- to 1-day orientation to the role of a SAFe Scrum Master
- *SAFe 4.0 Product Manager / Product Owner with PMPO certification* – This 2-day certification course is for Product Managers, Business Owners, and Product Owners who will learn how to manage and prioritize backlogs, participate in SAFe events, define and support epics, capabilities, features, and user stories, and manage stakeholders at the various levels of the enterprise.
- *SAFe 4.0 Advanced Scrum Master with ASM Certification* – This 2-day advanced, certification course prepares current Scrum Masters for their leadership role in facilitating Agile team, program, and enterprise success. It enhances the Scrum paradigm with an introduction to scalable engineering and DevOps practices; the application of Kanban to facilitate flow; supporting interactions with architects, product management, and other critical stakeholders; and tips and techniques for building high-performing Agile teams.

Note: The courseware offerings are always advancing, so be sure and check ScaledAgile.com for the latest updates.

Continuous Improvement
Once the transformation is under way, there are a variety of opportunities for sustaining and enhancing improvements in speed and quality that can be best facilitated by the extensive community of skilled professionals. These activities can include Agile Release Train health checks; Portfolio, Value Stream, Agile Release Train, and Team agility self-assessments; and facilitated Inspect and Adapt sessions. For help, we again refer you to your in-house SPCs, Scaled Agile Partners, and other independent SPCs.

Guidance and Governance with Enterprise SAFe

For those who would benefit from being able to modify a custom version of the Scaled Agile Framework website, all SAFe content is available for enterprise licensing. Organizations that are scaling Lean-Agile best practices leverage Enterprise SAFe so they can have access to the most up-to-date content for their teams and the thought leaders at Scaled Agile, Inc.

Enterprise SAFe allows organizations to align around common process objectives while providing the ability to adapt SAFe to their unique needs and culture. Enterprise SAFe allows organizations to create a custom version of the Scaled Agile Framework website while maintaining automated updates as the methodology advances.

Fully Adaptable to Your Organization's Context
Provisioned by Scaled Agile, Inc., and built on WordPress, Enterprise SAFe supports adaptation that enables organizations to revise the graphical representation of the SAFe Big Picture as well as the entire SAFe content offering. Content is controlled locally by the enterprise via a set of tools that support accepting or rejecting framework updates from Scaled Agile, Inc., as well as adding custom content, such as articles, icons, labels, and graphics.

Enterprise SAFe features include:

- A WordPress publishing platform that allows you to start capturing the specifics of your custom SAFe implementation in a matter of minutes
- A customizable Big Picture in Adobe Illustrator that allows you to capture key modifications to the framework at the front end of the website
- A PowerPoint version of all SAFe artwork that allows you to change the graphics that are integral to the story
- Provisions to modify or extend SAFe with your custom process content
- Local control of custom content via a set of tools that allow reviewing differences between local pages and content updates, and accepting or rejecting updates
- Administration utility for easy management of large numbers of user accounts

Enterprise SAFe is provisioned by Scaled Agile, Inc., via a private and secure cloud-based website.

Index