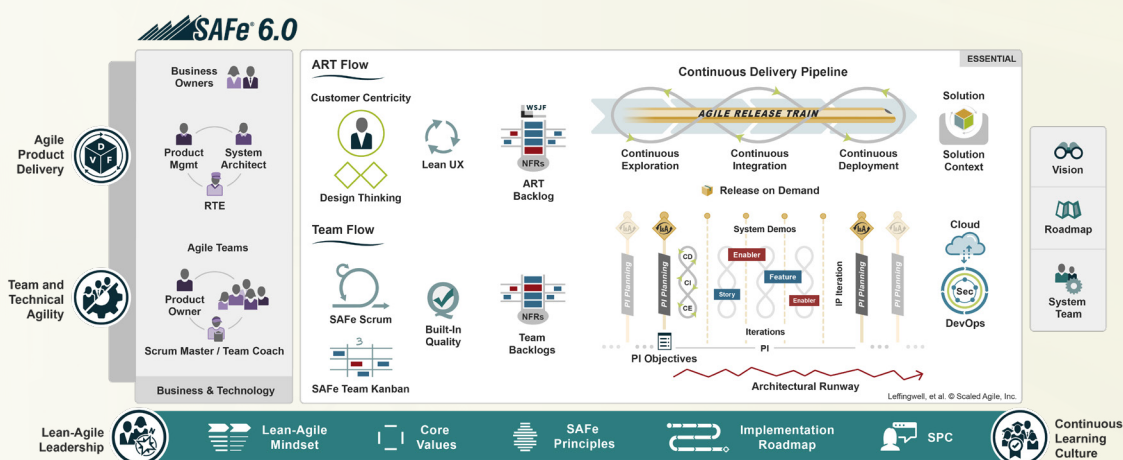# ESSENTIAL SAFe®

The Scaled Agile Framework® (SAFe®) was designed to help Enterprises deliver value continuously and more efficiently on a regular and predictable schedule, making them more Agile in the marketplace and more competitive in their industry.

Over time, the Framework has grown to accommodate the full spectrum of complexity in software and systems development. From international bond trading and medical devices to memory chips and fighter aircraft, SAFe has proven to scale to all situations. But, with such a robust framework, the question becomes: how closely does an organization need to follow various SAFe practices to get the desired result?

In addition, we've observed that not every implementation realizes the full business benefits that others achieve. When diagnosing, we've found that the less successful Enterprises have skipped some of the most essential practices. It's easy to see how that can happen. After all, it's a big framework, how would an Enterprise know what's essential?

To that end, Essential SAFe is a subset that describes the minimal elements necessary for success. If you are incorporating these 10 critical success factors for each Agile Release Train (ART) in your portfolio, you're well on your way to realizing the full benefits of SAFe.

ESSENTIAL SAFe®

Lean-Agile Principles

SAFe practices are grounded in fundamental principles. That's why you can be confident that they apply well in your case. And if the practices don't directly apply, the underlying principles can guide you 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."

Real Agile Teams and ARTs

Real Agile Teams and Trains that are fully cross-functional have everything, and everyone, necessary to produce a working, tested increment of the Solution. They are self-organizing and self-managing, which enables value to flow more quickly, with a minimum of overhead. Product Management, System Arch, and Release Train Engineer provide content and technical authority, and an effective development process. Product Owners and Scrum Masters/Team Coaches help the Dev Teams meet their objectives. The Customer is integrally engaged throughout the development process.

Cadence and Synchronization

Cadence provides a rhythmic pattern, the dependable heartbeat of the development process. It makes routine that which can be routine. Synchronization allows multiple perspectives to be understood and resolved at the same time. For example, synchronization is used to pull the disparate assets of a system together to assess Solution-level viability.

PI Planning

No event is more powerful in SAFe than Planning Interval (PI) Planning. It's the cornerstone of the PI, which provides the rhythm for the ART. When 100 or so people work together toward a common mission, vision, and purpose, it's amazing how much alignment and energy it creates. Gaining that alignment in just two days can save months of delays.

Customer Centricity, DevOps and Release on Demand

DevOps provides the Culture, Automation, Lean-flow, Measurement, and Recovery (CALMR) capabilities to enable an Enterprise to bridge the gap between development and operations. Releasability focuses on the Enterprise's ability to deliver value to its Customers more often and according to the demand of the market. Together DevOps and releasability allow an organization to achieve better economic results by having more frequent releases and faster validation of hypotheses.

Without a shared understanding of these principles:

- [ ] There is no systematic way to adapt practices to specific context
- [ ] Business outcomes do not significantly improve
- [ ] Practices and measures that were once beneficial become problematic
- [ ] Lean-Agile Mindset is unachievable
- [ ] Conflict and disagreement on processes and practices are difficult to resolve

Without real Agile Teams and Trains:

- [ ] Responsibilities are unclear, delaying decision-making
- [ ] Lack of cross-functional skills causes over-specialization and bottlenecks that inhibit flow
- [ ] Teams locally optimize and can't deliver end-to-end value
- [ ] No architectural and User Experience integrity; Solution Features and components evolve incompatibly
- [ ] Vision and requirements are not clear, and prioritization is extremely difficult

Without cadence and synchronization:

- [ ] No steady development rhythm
- [ ] Gradual decline into disorder and lack of predictability
- [ ] Hard to schedule planning, retrospectives, demos, and other key events
- [ ] Difficult to adjust to changing priorities
- [ ] Teams are constantly overloaded

Without PI Planning:

- [ ] Stakeholders, teams, and management are not aligned
- [ ] Demand doesn't match capacity, no predictability, excess Work In Process (WIP)
- [ ] Lack of trust between stakeholders and teams
- [ ] Late discovery of dependencies causes delays
- [ ] Low commitment, ownership, and employee engagement

Without Customer Centricity, DevOps and Release on Demand:

- [ ] Lower Customer satisfaction and bad Customer experiences
- [ ] Reduced deployment quality and high production defects
- [ ] Value delivery is seriously delayed; more frequent releases are not possible, increasing time-to-market
- [ ] Large batches of code are pushed to production, resulting in production errors and emergencies
- [ ] Friction between development and operations limits collaboration, learning, and cultural change

System Demo

The primary measure of the ART's progress is the objective evidence provided by a working Solution in the System Demo. Every two weeks, the full system—the integrated work of all teams on the train for that Iteration—is demoed to the train's stakeholders. Stakeholders provide the feedback the train needs to stay on course and take corrective action.

Without System Demo:

- ☐ Teams are 'sprinting,' but the system is not
- ☐ Chronic lack of trust between stakeholders and teams
- ☐ Lack of feedback to Iterate to the right solution
- ☐ False progress and poor quality
- ☐ 'Waterfalled PIs'—problems and risks are discovered too late

Inspect & Adapt

Inspect & Adapt (I&A) is a significant event held every PI. A regular time to reflect, collect data, and solve problems, the Inspect & Adapt assembles teams and stakeholders to assess the Solution and define and take action on the improvements needed to increase the velocity, quality, and reliability of the next PI.

Without Inspect & Adapt:

- ☐ No systemic improvement, problems persist
- ☐ No means to measure or establish delivery predictability
- ☐ Improvement efforts address symptoms, not root causes
- ☐ Leaders who could change the system are not engaged
- ☐ Low morale

IP Iteration

The Innovation and Planning Iteration occurs every PI and serves multiple purposes. It acts as an estimating buffer for meeting PI Objectives and provides dedicated time for innovation, continuing education, and PI Planning and Inspect & Adapt events. It is like extra oxygen in the tank: without it, the train may start straining under the 'tyranny of the urgent.'

Without the IP Iteration:

- ☐ Lack of estimating buffer and poor predictability
- ☐ 'Tyranny of the urgent' Iteration inhibits innovation
- ☐ Technical debt grows uncontrollably
- ☐ Lots of overtime, and people burn out
- ☐ No time for teams to plan together, demo together, and improve together

Architectural Runway

Architectural Runway consists of the existing code, components, and technical infrastructure necessary to support implementation of high-priority, near-term features, without excessive delay and redesign. Without enough investment in the Architectural Runway, the train will slow down to allow time to redesign for each new Feature.

Without Architectural Runway:

- ☐ Architecture progressively decays under the 'urgency of now'
- ☐ Velocity peaks for a while, then falls off
- ☐ Infrequent and irregular releases
- ☐ Solution robustness, maintainability, and quality decay
- ☐ Unsustainable development pace

Lean-Agile Leadership

For SAFe to be effective, the Enterprise's leaders and managers must take responsibility for Lean-Agile adoption and success. Executives must become leaders who are trained—and become trainers in—these leaner ways of thinking and operating. Without leadership taking responsibility for the implementation, the transformation will likely fail to achieve the full benefits.

Without Lean-Agile Leadership:

- ☐ Teams cannot learn from their leaders
- ☐ The transformation is fatally impaired
- ☐ Agile development with traditional governance results in 'Agile in name only'
- ☐ Lead time increases due to frequent escalation of decisions
- ☐ People not allowed to experiment, fail, innovate, and learn

SAFe®
PROVIDED BY SCALED AGILE

ESSENTIAL SAFe® Self-Diagnostic

1. On the previous two pages, mark the symptoms that exist in your enterprise.

2. Then shade in one box in the following table for each symptom identified.

3. The things you need to work on should be obvious. Take action!

EXAMPLE

| Lean-Agile Principles | Real Agile Teams and ARTs | Cadence & Synchronization | PI Planning | Customer Centricity, DevOps and Releasability | System Demo | Inspect & Adapt | IP Iteration | Architectural Runway | Lean-Agile Leadership |

These 10 critical success factors also help an organization achieve three Lean Enterprise competencies: Lean-Agile Leadership, Team and Technical Agility, and Agile Product Delivery. Moreover, Essential SAFe provides the foundation for achieving the Business Solutions and Lean Systems Engineering and Lean Portfolio Management competencies.

TO LEARN MORE, READ THE GUIDANCE ARTICLE AT

scaledagileframework.com/essential-safe/