SAFe® 4.0 Glossary

Scaled Agile Framework® Terms and Definitions

English

Provided by Scaled Agile, Inc.

Leffingwell, et al. © 2008–2016 Scaled Agile, Inc. v4.0
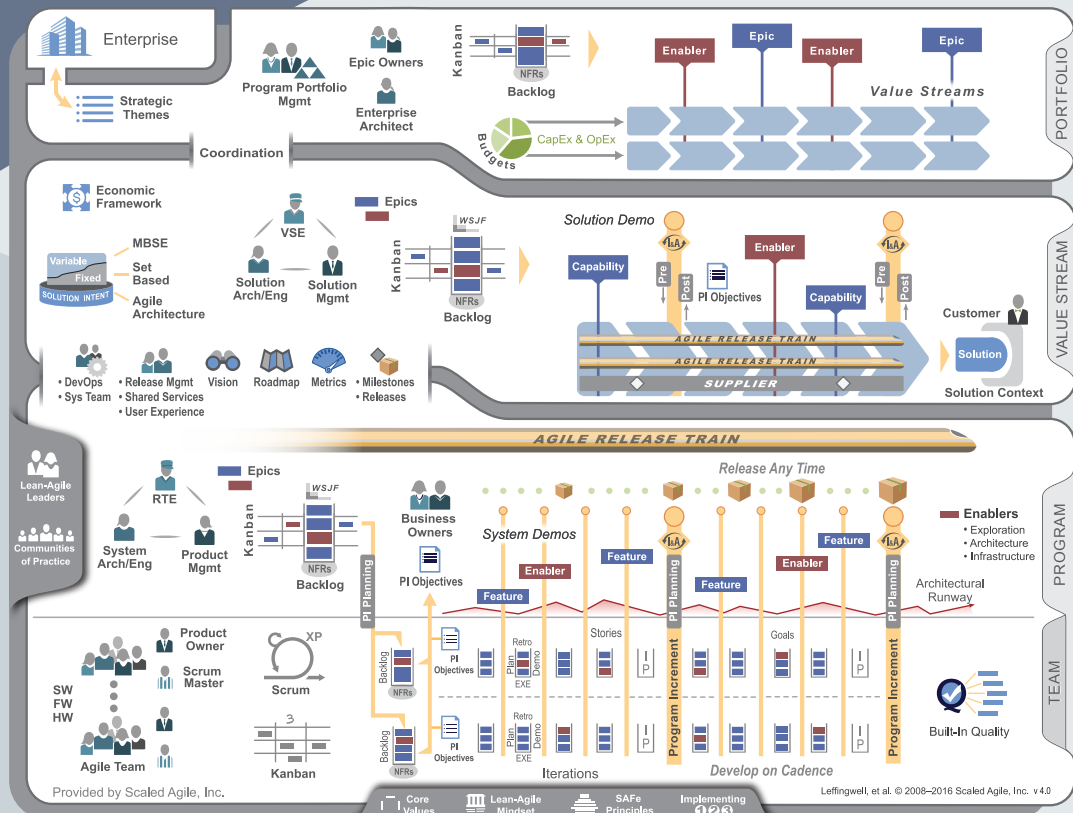
Agile Architecture

Agile architecture is a set of values and practices that advances the design and architecture of a system while implementing new business functions.

Agile Release Train

The Agile Release Train (ART) is a long-lived team-of-Agile-teams, which along with other stakeholders, develops and delivers solutions incrementally, using a series of fixed iterations within a program increment timebox. The ART aligns teams to a common business and technology mission.

Agile Teams

Agile teams are a group of three to nine dedicated individual contributors, covering all the roles necessary to define, build, and test a quality increment of value in an iteration.

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.

Budgets

See Lean-Agile budgeting.

Built-in Quality

Built-in quality practices ensure that each solution element, at every increment, meets appropriate quality standards throughout development.

Business Epic

See epic.

Business Owners

Business Owners are a small group of stakeholders who have the primary technical, fitness for use, governance and Return on Investment (ROI) responsibility for a solution developed by an Agile Release Train. They are key stakeholders on the train and actively participate in certain events.

Capability

A capability is a high-level solution behavior that typically spans multiple Agile Release Trains. They are sized and split as necessary to fit within a single program increment.

CapEx and OpEx

CapEx and OpEx describe Lean-Agile financial practices for tracking capital expenses (CapEx), and operating expenses (OpEx) in a value stream budget. In some cases, CapEx may include capitalized labor associated with development of intangible assets such as software, intellectual property and patents.

Communities of Practice

Communities of Practice (CoPs) are groups of people who have a common interest in a specific technical or business domain. They collaborate regularly to share information, improve their skills and performance and advance the general knowledge of the domain.

Continuous Integration
Continuous integration is a built-in quality practice, where team members constantly integrate and verify their work, using automated build-and-test environments that quickly identify problems and defects.

Coordination
See value stream coordination.

Core Values
Core values define the ideals and beliefs that are key to SAFe's effectiveness. They are: alignment, built-in quality, transparency, and program execution.

Customer
The customer is anyone who consumes the work of a value stream. Customers are an integral part of the agile development process and value stream.

Develop on Cadence
Develop on cadence is a strategy for managing the inherent variability in solution development, by making sure important events and activities on a regular, predictable schedule.

DevOps
DevOps is a mindset, culture, and set of technical practices that foster communication, collaboration, and close cooperation among all the professionals needed to develop, test, deploy, and maintain a solution.

Economic Framework
The economic framework is a set of decision rules aligning everyone to the financial objectives of the mission, defines economic tradeoff parameters, and assures operation within the budget provided by the program portfolio.

Enablers
Enablers further the exploration, infrastructure, and architecture development activities needed to support future business functionality. Enablers arise from the backlog and occur at all levels of the framework where they are described as enabler portfolio epics, enabler capabilities, enabler features, or enabler stories.

Enterprise
The enterprise represents the business entity that has the ultimate strategic, fiduciary, and governance authority for all the development value streams that make up a portfolio.

Enterprise Architect
The Enterprise Architect fosters adaptive design and engineering practices, and drives strategic architectural initiatives for a portfolio. Enterprise Architects facilitate the reuse of ideas, components, and proven patterns across solutions.

Epic
An epic is a significant initiative that typically affects multiple value streams and ARTs. Epics require analysis using a lightweight business case and financial approval before implementation. There are two types of epics: business epics and enabler epics, which may occur at the portfolio, value stream, and program levels.

Epic Owners

Epic Owners are responsible for coordinating portfolio epics through the portfolio kanban system. They develop the business case and, when approved, work directly with the key stakeholders on the selected trains to help realize the implementation.

Feature

A feature is a service provided by the system that fulfills stakeholder needs and can be delivered by a single ART. Each feature includes a statement of benefits and acceptance criteria, and is sized to fit within a program increment.

Implementing 1-2-3

Implementing 1-2-3 is a proven success pattern for launching SAFe. It describes the three basic steps: 1) train Lean-Agile change agents, 2) train executives, managers, and leaders, 3) train teams and launch Agile Release Trains.

Innovation and Planning Iteration

The Innovation and Planning iteration occurs every PI and serves multiple purposes. It acts as an estimating buffer for meeting PI objectives, as well as providing dedicated time for innovation, continuing education, and PI planning and Inspect and Adapt (I&A) events.

Inspect and Adapt

Inspect and Adapt (I&A) is a significant event, held at the end of each PI, where the current state of the solution is demonstrated and evaluated. Teams then reflect, and identify improvement backlog items via a structured, problem-solving workshop.

Iteration Execution

Iteration execution is how Agile teams manage their work throughout the iteration timebox, resulting in a high-quality, working, tested system increment. Each iteration follows a standard pattern: plan the iteration, commit to a goal, execute, demonstrate the work to the key stakeholders, and hold a retrospective.

Iteration Goals

Iteration goals are high-level summaries of the business and technical goals that the team and Product Owner agree to accomplish in an iteration. They serve as a communication mechanism within the team, and to the team's stakeholders, and help ensure alignment with the PI objectives.

Iteration Planning

Iteration planning is an event at which all team members determine how much of the team backlog they can commit to deliver during an upcoming iteration. The team summarizes the work as a set of committed iteration goals.

Iteration Retrospective

The iteration retrospective is a meeting where the team members discuss the results of the iteration, review their practices and identify ways to improve.

Iterations

Iterations are a standard, fixed-length timebox during which teams deliver incremental value in the form of working, tested software and systems. Iterations lengths may be chosen from one to three weeks, with two weeks being the suggested, and most common, duration.

Lean-Agile Budgeting

Lean-Agile budgeting is a set of practices that fund and empower value streams, while maintaining financial and fitness-for-use governance through objective evaluation of working systems, and dynamic budget adjustments.

Lean-Agile Leaders

Lean-Agile leaders are lifelong learners who are responsible for the successful adoption of SAFe and the results it delivers. They help teams build better systems by learning, exhibiting, teaching and coaching SAFe's Lean-Agile principles and practices.

Lean-Agile Mindset

A Lean-Agile mindset combines the concepts of the Agile Manifesto and Lean thinking, serving as the basis for SAFe principles and practices.

Metrics

Metrics are agreed-upon measures used to evaluate how well the enterprise is adopting SAFe and progressing toward the portfolio, value stream, ART and team business and technical objectives.

Milestones

Milestone are used to track progress toward a specific goal or event. There are three types of SAFe milestones: fixed-date milestones, program increment milestones and learning milestones.

Model-Based Systems Engineering

Model Based Systems Engineering (MBSE) is a methodology that focuses on the development of various related system models that are used to define and design the system. Models provide an efficient way to explore, update and communicate system aspects to stakeholders while significantly reducing or eliminating dependence on traditional documents.

Nonfunctional Requirements

Non-functional requirements (NFRs) define system attributes such as security, reliability, performance, maintainability, scalability, and usability. They serve as constraints or restrictions on the design of the system across the different backlogs.

Program Increment Planning

Program Increment (PI) planning is a cadence-based, face-to-face planning event that serves as the heartbeat of the Agile Release Train, aligning all the teams on the ART to a common goal.

Portfolio Backlog

The portfolio backlog is a prioritized list of epics that have been approved for implementation through the portfolio kanban system.

Portfolio Business Epic
Portfolio business epics are large business initiatives that often span value streams and require multiple program increments to deliver.

Portfolio Kanban
The portfolio kanban is a method used to visualize and manage the analysis, prioritization and flow of portfolio epics from ideation to implementation and completion.

Portfolio Level
The portfolio level contains the roles, artifacts and processes needed to meet the strategic intent of the portfolio. It defines the value streams and funding for the people and other resources that build the solutions.

Pre- and Post-PI Planning
Pre- and post-PI (Program Increment) planning meetings are used to prepare for, and follow-up, PI planning for multiple ARTs and suppliers in a large value stream.

Product Management
Product Management is the content authority for the program level and backlog. They are responsible for identifying customer needs, prioritizing features and developing the program vision and roadmap.

Product Owner
The Product Owner is the content authority for the team level. They are responsible for the team backlog, prioritizing and accepting stories, and representing the customer to the Agile team.

Program Backlog
The program backlog is a prioritized list of features intended to address user needs and deliver business benefits. It also includes the enabler features necessary to build the architectural runway.

Program Epics
Program epics are initiatives significant enough to require analysis using a lightweight business case and financial approval before implementation. Their scope is limited to a single Agile Release Train and may take several Program Increments to develop.

Program Increment
A Program Increment (PI) is a timebox in which Agile Release Trains deliver incremental value in the form of working, tested software and systems. PIs are typically eight to twelve weeks long, and the most common pattern for a PI is four development iterations, followed by one Innovation and Planning iteration.

Program Kanban
The program kanban is a method used to visualize and manage the analysis, prioritization and flow of program epics and features from ideation to completion for a single ART.

Program Level
The program level contains the roles and activities needed to continuously deliver solutions via an Agile Release Train.

Program PI Objectives
Program PI objectives are an integrated summary of all the teams' PI objectives for a train. They are used to communicate the plan to stakeholders and to measure accomplishments of the train for a program increment.

Program Portfolio Management
Program Portfolio Management is a function that contains the individuals who have the ultimate decision-making authority of a portfolio. It is responsible for strategy, investment funding, common program management elements, and portfolio governance.

Release
A release is the private or public delivery of valuable, working, and fully tested and validated solution increments.

Release Any Time
Release any time is a practice whereby the Agile Release Train separates the development cadence from the solution release cycle. Within appropriate technical and business governance, trains may release whatever is needed, at any time.

Release Management
Release Management is a function that communicates release status to stakeholders, validates that the solutions meets the relevant quality and governance criteria, and provides the final authorization for the release.

Release Train Engineer
The Release Train Engineer (RTE) is a servant leader and coach for the Agile Release Train, who facilitates the train's processes, events, and execution. He or she escalates impediments and helps manage risk, value delivery, and continuous improvement.

Roadmap
The roadmap is a schedule of events and milestones that communicate planned deliverables over a timeline. It includes commitments for the planned PI and offers visibility into the forecasted deliverables of the next few PIs.

SAFe Principles
SAFe principles define the nine core beliefs, fundamental truths and economic premises that drive effective roles, and practices and successful SAFe implementation.

Scrum Master
Scrum Masters are servant leaders and coaches for an Agile team. They help educate the team in Scrum, Extreme Programming (XP), Kanban and SAFe, and ensure that the process is being followed. They also help remove impediments, and foster an environment for high-performing team dynamics, continuous flow, and relentless improvement.

ScrumXP

ScrumXP is a lightweight process for cross-functional, self-organized teams to deliver value within the context of SAFe. ScrumXP combines the power of Scrum project management practices with XP-inspired technical practices.

Set-Based Design

Set-based design is a practice in which the requirements and design options are kept flexible for a longer period of time. Instead of choosing a single "point" solution upfront, set-based design is used to identify design options, and eliminate poorer choices over time, thereby enabling flexibility in the design process.

Shared Services

Shared Services represents the specialty roles, people and services necessary for the success of an Agile Release Train or value stream, but can't be dedicated to any specific train full-time (e.g. security specialists, database administrators).

Solution

A solution is a final product, service, or system delivered to the customer or, enables an operational value stream within the organization.

Solution Architect/Engineering

Solution Architect/Engineering represents the individuals and teams who have the technical responsibility for the overall architecture and engineering design of the solution. They help align the value stream and Agile Release Trains to a common technological and architectural vision.

Solution Context

Solution context identifies critical aspects of the target solution's environment. It identifies impact on the requirements, usage, installation, operation, and support of the solution itself.

Solution Demo

The solution demo is where the results of all the development efforts from multiple ARTs—along with the contributions from suppliers—are integrated, evaluated, and made visible to customers and other stakeholders.

Solution Intent

Solution intent represents the repository for storing, managing, and communicating knowledge of the current and intended solution, including traceability between the items where required.

Solution Management

Solution Management is the content authority for the value stream level. They work with customers to understand their needs, create the vision and roadmap, define requirements, and guide work through the value stream kanban.

Spanning Palette

The spanning palette contains various roles and artifacts that may be applicable at any level of the framework. It is used to apply various SAFe elements to a specific team, program, value stream, or portfolio context.

Stories
Stories are short, simple descriptions of a small piece of desired functionality, written in the user's language. Each story supports incremental development by implementing a small, vertical slice of system functionality.

Strategic Themes
Strategic themes are itemized, differentiated business objectives that connect a portfolio to the enterprise's business strategy. They provide business context for decision-making within the portfolio, and they serve as inputs to the budget, portfolio, solution, and program backlog.

Supplier
A supplier is an organization that develops and delivers components and subsystems that help value streams deliver value to their customers.

System Architect/Engineering
System Architect/Engineering represents an individual or small team that defines a common technical and architectural vision for the solution under development. They participate in defining the system and subsystems, interfaces, validate technology assumptions, and evaluate alternatives.

System Demo
The system demo occurs at the end of every iteration, and provides an integrated view of the new features, which have been delivered by all the teams in the ART for the most recent iteration. It provides the Agile Release Train with an objective measure of progress during a PI.

System Team
The System Team is a special Agile team that provides assistance in building and using the Agile development environment, including continuous integration and test automation. The System Team integrates assets from Agile teams, performs end-to-end solution testing where necessary, and assisting with deployment. They often facilitate the system demo.

Team Backlog
The team backlog contains user and enabler stories that originate from the program backlog, as well as stories that arise locally from the team's specific context. It represents all the things a team needs to do to advance their portion of the system.

Team Demo
The team demo is used to measure progress and get feedback at the end of each iteration by demonstrating every story, spike, refactor, and new non-functional requirement developed in the recent iteration.

Team Kanban
Team kanban is a method that facilitates the flow of value by visualizing Work in Process (WIP) and establishing WIP limits, measuring throughput, and continuously improving the process.

Team Level
The team level contains the roles, activities, and process model for the teams who power the Agile Release Train.

Team PI Objectives

Team PI objectives describe the business and technical goals that an Agile team intends to achieve in the upcoming PI. They summarize and validate business and technical intent, which enhances communication, alignment, and visibility.

Test-First

Test-first is a built-in quality practice to develop and test a system in small increments. Tests are developed in advance of the functionality and are used to better define the intended system behavior, before the system is implemented.

User Experience

User experience designers support a consistent user experience across the components and systems of the larger solution, while Agile teams have responsibility for implementing the solution, including the user-facing elements.

Value Stream Backlog

The value stream backlog is the repository for all upcoming capabilities and enablers, each of which can span multiple Agile Release Trains and is used to advance the solution and build the architectural runway.

Value Stream Coordination

Value stream coordination provides guidance for managing dependencies across value streams in a portfolio.

Value Stream Engineer

The Value Stream Engineer (VSE) is a servant leader and coach for a value stream. VSEs facilitate value stream processes, events and execution, as well as driving continuous improvement.

Value Stream Epics

Value stream epics are significant initiatives that require analysis, using a lightweight business case, and financial approval before implementation. Their scope is limited to a to a single value stream and may take several PIs to develop.

Value Stream Kanban

The value stream kanban is a method used to manage the analysis, prioritization and flow of value stream epics and capabilities, from ideation to completion.

Value Stream Level

The value stream level provides the roles, responsibilities and activities necessary to support those building large and complex solutions, which typically require multiple Agile Release Trains, as well as the contributions of suppliers. This optional level is used by enterprises that face the largest systems challenges, which require multidisciplinary software and system professionals.

Value Stream PI Objectives

Value stream PI objectives are the business and technical goals identified by value stream stakeholders for a PI. They communicate to stakeholders what the value stream will deliver in the upcoming PI. They are used for large value streams that have multiple ARTs or suppliers.

Value Streams

Value streams represent the series of steps that an enterprise uses to build solutions that provide a continuous flow of value to a customer. Value streams are the primary means for understanding business objectives, organizing teams and trains, and delivering end value. They are realized by Agile Release Trains.

Vision

The vision is a description, or future view of the solution to be developed, reflecting customer and stakeholders needs, as well as the proposed features and capabilities. It provides the larger, contextual overview and purpose of the solution under development.

Weighted Shortest Job First

Weighted Shortest Job First (WSJF) is a prioritization model used to sequence "jobs" (features, capabilities) so as to produce maximum economic benefit in a flow-based system. WSJF is calculated as the cost of delay divided by job duration (or job size as a proxy).

Guide to acronyms and abbreviations

ART	Agile Release Train
BO	Business Owner
BV	Business Value
BVIR	Big Visual Information Radiator
CFD	Cumulative Flow Diagram
CapEx	Capital Expenses
CI	Continuous Integration
CoD	Cost of Delay
CoP	Community of Practice
DoD	Definition of Done
DSU	Daily Stand-up
EA	Enterprise Architect
EO	Epic Owner
FW	Firmware
HW	Hardware
I&A	Inspect and Adapt
IP	Innovation and Planning (iteration)
MBSE	Model-Based Systems Engineering
NFR	Non-functional Requirements
OE	Opportunity Enablemen
OpEx	Operating Expenses
PDCA	Plan, Do, Check, Adjust
PI	Program Increment
PM	Product Manage

PM/PO	Product Manager / Product Owner
PO	Product Owner
PPM	Program Portfolio Management
ROAM	Resolved, Owned, Accepted, Mitigated
RR	Risk Reduction
RTE	Release Train Engineer
S4T	SAFe® for Teams
SAFe®	Scaled Agile Framework
SA	SAFe® Agilest
SM	Scrum Master
SMART	Specific, Measurable, Achievable, Realistic, Time-bound
SoS	Scrum of Scrums
SP	SAFe® Practitioner
SPC	SAFe® Program Consultant
SW	Software
UX	User Experience
VS	Value Stream
VSE	Value Stream Engineer
WIP	Work in Process
WSJF	Weighted Shortest Job First
XP	Extreme Programming

SAFe Certifications

There's a certification for everyone on the the SAFe Learning Path

---

SAFe 4.0 Program Consultant Certification

A SAFe Certified Program Consultant 4.0 (SPC4) is a SAFe implementation professional responsible for training leaders, change agents, consultants, and team members to drive a Lean-Agile transformation at enterprise scale. Key areas of competency include designing a SAFe implementation, launching and facilitating an Agile Release Train (ART), and extending the Lean-Agile portfolio by launching additional ARTs.

SAFe Agilist Certification

A SAFe Certified Agilist (SA) is a SAFe enterprise leadership professional who is part of a Lean-Agile transformation. Key areas of competency include the application of Lean-Agile principles, execution, and release of value via Agile Release Trains (ARTs), and building an Agile portfolio with Lean-Agile budgeting. SA4 is used to indicate SAFe 4.0 certified.

SAFe 4.0 Release Train Engineer Certification

A SAFe Certified Release Train Engineer (RTE) is a SAFe professional responsible for driving end-to-end delivery of value in a single program or value stream in a Lean-Agile enterprise. Key areas of competency include facilitating Program Increment planning, leading program execution, and coaching programs and teams in relentless improvement of value delivery

---

SAFe Product Manager/Product Owner Certification

A SAFe Certified Product Manager/Product Owner (PMPO) is a SAFe product manager or product owner professional who works with customers and development organizations to identify and write requirements. Key areas of competency include identifying customer needs, writing epics, capabilities, features and stories, and prioritizing work.

SAFe Scrum Master Certification

A SAFe Certified Scrum Master (SSM) is a SAFe Scrum Master professional responsible for integrating Scrum practices into an organization. Key areas of competency include using Scrum and Kanban to facilitate team events, supporting program execution, and coaching Agile teams.

SAFe Advanced Scrum Master Certification

A SAFe Certified Advanced Scrum Master (SASM) is a SAFe Scrum Master professional responsible for improving team results. Key areas of competency include identifying and solving team anti-patterns, improving value flow using Kanban and engineering practices, supporting program-level execution, and using problem solving techniques to drive relentless improvement.

SAFe Practitioner Certification

A SAFe Certified Practitioner (SP) is a SAFe team member professional responsible for using Scrum, Kanban, and XP in a SAFe environment. Key areas of responsibility include planning Program Increments and iterations, breaking requirements into stories, developing incrementally with built-in quality, demoing value at a team and program level, and problem solving impediments to drive relentless improvement.

Find out which SAFe training and certification path is right for you at:

www.scaledagile.com/which-course

FIND SAFe® TRAINING NEAR YOU

"In just two weeks, I feel like the 156-year historical cultural barrier we've created has been broken. Thank you for helping us."
—Senior Level Architect, Fortune Global 500 Company

Training is at the core of customer success

Our most popular courses are offered in 35 countries and 350 cities worldwide through Scaled Agile and our Global Partner Network, all focused on empowering the enterprise to realize the maximum benefit that comes from adopting SAFe.

Find a SAFe training class near you at:
www.scaledagile.com/calendar

Learn more

If you would like to learn more about SAFe, visit these websites.

SAFe Case Studies

www.scaledagileframework.com/case-studies

Scaled Agile Framework

www.scaledagileframework.com

Introduction to SAFe

www.scaledagileframework.com/introduction-to-safe

Role-based SAFe training and certification

www.scaledagile.com

SAFe presentations and videos

www.scaledagileframework.com/videos-and-presentations

Books on SAFe

www.scaledagileframework.com/books

About Scaled Agile, Inc.

Based in Boulder, Colorado, USA, Scaled Agile's mission is to help large enterprises achieve better outcomes, increase employee engagement, and improve business economics through adoption of Lean-Agile principles and practices based on the Scaled Agile Framework® (SAFe®).

Scaled Agile supports thousands of practitioners of the Framework through training, certification, consulting services, and a global partner network that reaches over 35 countries and 350 cities. As a member of the Pledge 1% corporate philanthropy movement, Scaled Agile contributes to the local community and seeks to inspire and encourage other companies to do the same.

Contact:

Scaled Agile, Inc.

5480 Valmont Rd, Suite 100

Boulder CO 80301 USA

+1.303.554.4367

support@scaledagile.com

www.scaledagile.com