Purpose of the Scrum@Scale Guide

Scrum, as originally outlined in the Scrum Guide, is a framework for developing, delivering, and sustaining complex products by a single team. Since its inception, its usage has extended to the creation of products, processes, services, and systems that require the efforts of multiple teams. Scrum@Scale was created to efficiently coordinate this new ecosystem of teams in a way that optimizes the overall strategy of the organization. It achieves this goal through setting up a "minimum viable bureaucracy" via a scale-free (below) architecture, which naturally extends the way a single Scrum Team functions across the organization.

This guide contains the definitions of the components that make up the Scrum@Scale framework, including its scaled roles, scaled events, and enterprise artifacts, as well as the rules that bind them together.

Dr. Jeff Sutherland developed Scrum@Scale based on the fundamental principles behind Scrum, Complex Adaptive Systems theory, game theory, and his work in biology. This guide was developed with the input of many experienced Scrum practitioners based on the results of their field work.

Why Scrum@Scale?

Scrum was designed for a single team to be able to work at its optimal capacity while maintaining a sustainable pace. In the field, it was found that as the number of Scrum Teams within an organization grew, the output (working product) and velocity of those teams began to fall (due to issues like cross-team dependencies and duplication of work). It became obvious that a framework for effectively coordinating those teams was needed in order to achieve linear scalability. Scrum@Scale is designed to accomplish this goal via its scale-free architecture.

Scale-free architectures are commonly found in biological systems like the human body, and in chip design that requires putting billions of transistors on a chip. The internet is designed to be scale-free by every node having the same structure as every other node. By utilizing a scale-free architecture, an organization is not constrained to grow in a particular way determined by a set of arbitrary rules; rather it can grow organically based on its unique needs and at a sustainable pace of change that can be accepted by the groups of individuals that make up the organization. The simplicity of the Scrum@Scale model is essential to a scale-free architecture and carefully avoids introducing extra complexity that will cause productivity per team to decrease as more teams are created.

Scrum@Scale is designed to scale across the organization as a whole: all departments, products and services. It can be applied across multiple domains in all types of organizations in industry, government, or academia.

Definition of Scrum@Scale

Scrum: A framework within which people can address complex adaptive problems, while productively and creatively delivering viable products of the highest possible value.

The Scrum Guide is the minimal feature set that allows inspection and adaptability via radical transparency to drive innovation, customer satisfaction, performance, and team happiness.

Scrum@Scale: A framework within which networks of Scrum Teams can address complex adaptive problems, while creatively delivering products of the highest possible value.

NOTE: These "products" may be hardware, software, complex integrated systems, processes, services, etc., depending upon the domain of the Scrum Teams.

Scrum@Scale is:

- · Lightweight - the minimum viable bureaucracy
- · Easy to understand - consists of only Scrum Teams
- · Hard to master - requires implementing a new operating model

Scrum@Scale is a framework for scaling Scrum. It radically simplifies scaling by using Scrum to scale Scrum.

Scrum@Scale consists of components that allow an organization to customize their transformational strategy and implementation. It gives them the ability to target their incrementally prioritized change efforts in the area(s) they deem most valuable or most in need of change and then progress on to others.

In Scrum, the removal of impediments is a central component to increase team speed. In Scrum@Scale impediment removal is applied at scale to increase the speed of the entire organization.

In Scrum, care is taken to separate accountability of the "what" from the "how." The same care is taken in Scrum@Scale so that jurisdiction and accountability are expressly understood in order to eliminate wasteful organizational conflict that keeps teams from achieving their optimal productivity.

In separating these two jurisdictions, Scrum@Scale contains two cycles: the Scrum Master Cycle (the "how") and the Product Owner Cycle (the "what"), each touching the other at two points. Taken together, these cycles produce a powerful framework for coordinating the efforts of multiple teams along a single path.

Values-Driven Culture

Besides separating accountability of the "what" and the "how," Scrum@Scale further aims to build healthy organizations by creating a values-driven culture in an empirical setting. The Scrum values are: Openness, Courage, Focus, Respect, and Commitment. These values drive empirical decision making, which depends on the three pillars of Transparency, Inspection, and Adaptation.

Openness supports Transparency into all of the work and processes, without which there is no ability to inspect them honestly and attempt to adapt them for the better. Courage refers to taking the bold leaps required to deliver value quicker in innovative ways. Focus and Commitment refer to the way we handle our work obligations, putting customer value delivery as the highest priority. Lastly, all of this must occur in an environment based on Respect for the individuals doing the work, without whom nothing can be created.

Scrum@Scale helps organizations thrive by supporting a transformational leadership model which fosters a positive environment for working at a sustainable pace and putting commitment to deliver customer-facing value at the forefront of our efforts.

The Components of the Scrum@Scale® Framework

© 2019 Scrum Inc.
www.scruminc.com

3
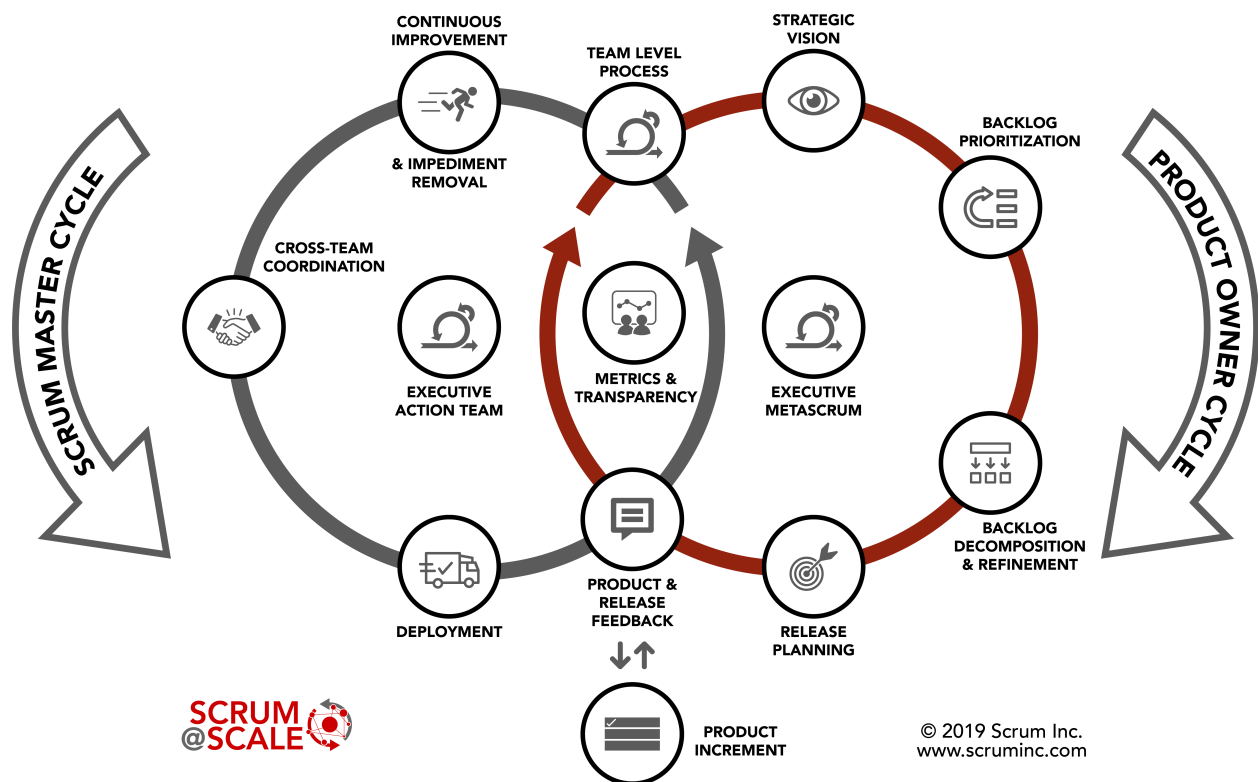
Scaled Structure: the Scrum of Scrums

A set of Scrum Teams that need to coordinate in order to deliver value to customers comprise a **Scrum of Scrums (SoS)**. This team is responsible for a fully integrated set of potentially shippable increments of product at the end of every Sprint from all participating teams. A SoS functions as a Release Team and must be able to directly deliver value to customers.

A Scrum of Scrums operates as a Scrum Team with scaled versions of the Scrum roles, events and artifacts. Given the scaled nature of the SoS there are some additional considerations.

SoS Roles

The SoS needs to have all of the skills necessary to deliver a fully integrated potentially shippable Product Increment at the end of every Sprint and to facilitate cross-team coordination where necessary. (It may need experienced architects, QA Leaders, members of the Product Ownership Team, and other operational skill sets.)

Product Owner Team

For each SoS, there is an associated group of Product Owners called a **Product Owner Team** which aligns the teams' priorities along a single enterprise backlog so that they can coordinate their Scrum Team's backlogs and build alignment with stakeholders.

A team's Product Owner is responsible for the composition and prioritization of the team backlog and may pull backlog items from the shared SoS backlog into the team backlog or generate independent backlog items at their discretion.

The main functions of the Product Owner Team are to:

- · create an overarching vision for the product and make it visible to the organization.
- · build alignment with key stakeholders to secure support for backlog implementation.
- · generate a single, prioritized backlog, ensuring that duplication of work is avoided.
- · assure impediments get resolved by properly prioritizing them in the backlog.
- · create a minimal, uniform "Definition of Done" that applies to all teams.
- · resolve dependencies raised by the teams.
- · generate a coordinated Release Plan and forecast beyond the current Release Plan (often called a roadmap).
- · decide upon and monitor metrics that give insight into the product and the market.

Product Owner Teams, just like SoS, function as Scrum Teams on their own. As such, they need to have someone acting as the Scrum Master and keeping the team on track in discussions. They also need a single person who is responsible for coordinating the generation of a single Product Backlog for all teams within the Scrum of Scrums. This person is designated as the **Chief Product Owner**.

Chief Product Owner (CPO)

The **Chief Product Owner** (CPO) coordinates priorities among Product Owners who work with individual teams. They align backlog priorities with Stakeholder and Customer needs. It may be an individual team PO who plays this role as well, or there may be a person specifically dedicated to this role. The responsibilities of the CPO are the same as that of a team PO, but at scale, the CPO is responsible for:

- · setting a strategic vision for the whole SoS.
- · creating a single, prioritized backlog of value to be delivered by all of the teams.
    - · Note: these items are typically larger Product Backlog Items that will need further refinement and decomposition by the team POs.
- · working closely with their associated Scrum of Scrums Master (below) so that the Release Plan that the Product Owner team generates can be deployed efficiently.
- · monitoring customer product feedback, as well as product feedback from the SoS, and adjusting the backlog accordingly.
- · at the executive/meta level, leading the MetaScrum event where the overall Product Backlog is presented and alignment with stakeholders is achieved.

Scrum of Scrums Master (SoSM)

The Scrum Master of the Scrum of Scrums is called the **Scrum of Scrums Master (SoSM)**. The SoSM is accountable for the release of the joint teams' effort and must:

- · make progress visible.
- · make an impediment backlog visible to the organization.
- · remove impediments that the teams cannot address themselves.
- · facilitate prioritization of impediments, with particular attention to cross-team dependencies and the distribution of backlog.
- · improve the effectiveness of the Scrum of Scrums.
- · work closely with the Product Owner Team to deploy a potentially releasable Product Increment at least every Sprint.
- · coordinate the teams' deployment with the Product Owner's Release Plans.

SoS Events

Scaled Daily Scrum (SDS)

Since the SoS needs to be responsive to impediments raised by participating teams, at least one representative (usually the team's Scrum Master or whoever best understands the teams' impediment) of each of the participating teams needs to attend a **Scaled Daily Scrum (SDS)**. The SDS event mirrors the Daily Scrum in that it optimizes the collaboration and performance of the network of teams. Any person or number of people from participating teams may attend as needed. Additionally, the SDS:

- is time-boxed to 15 minutes or less.
- must be attended by a representative of each team, including the Product Owner Team.
- is a forum where team representatives discuss what is going well, what is getting done, and how teams can work together more effectively.

Some examples of what might be discussed are:

- What impediments does my team have that will prevent them from accomplishing their Sprint Goal (or impact the upcoming release)?
- Is my team doing anything that will prevent another team from accomplishing their Sprint Goal (or impact their upcoming release)?
- Have we discovered any new dependencies between the teams or discovered a way to resolve an existing dependency?
- What improvements have we discovered that can be leveraged across teams?

Dealing with Impediments in a SoS

The SoSM should facilitate the refinement of an impediment backlog wherein impediments are identified as "ready" and prioritized to be removed. The teams then determine how best to remove them and how they will know when they are "done." In some cases, impediment resolution may require product development, in which case, involvement from the SoS CPO and Product Owner Team will be necessary.

Particular attention should be paid to the SoS Retrospective in which the teams' representatives share any learnings or process improvements that their individual teams have succeeded with, in order to share best practices across the teams within the SoS.

Reference Model

When implementing large networks of teams, it is critical to develop a scalable **Reference Model** for a small set of teams. Any deficiencies in a Scrum implementation will be magnified when multiple teams are deployed. Many of the initial scaling problems will be organizational policies and procedures or development practices that block high performance and frustrate teams.

Therefore, the first challenge is to create a small set of teams that implements Scrum well. This is best accomplished by the creation of an **Executive Action Team (EAT)** (below), which is accountable for the development and execution of the transformation strategy. The EAT must be comprised of individuals who are empowered, politically and financially, to ensure the existence of the Reference Model. This set of teams works through organizational issues that block agility and creates a Reference Model for Scrum that is known to work in the organization and can be used as a pattern for scaling Scrum across the organization.

As the Reference Model of teams accelerates, impediments and bottlenecks that delay delivery, produce waste, or impede business agility become apparent. The most effective way to eliminate these problems is to spread Scrum across the organization so that the entire value stream is optimized.

Scrum@Scale achieves linear scaling in productivity by saturating the organization with Scrum and distributing velocity and quality organically, consistent with the organization's specific strategy, products, and services.

Leadership Teams

Executive Action Team (EAT)

The **Executive Action Team (EAT)** fulfills the Scrum Master role for the entire agile organization. This leadership team creates an agile bubble in the organization where the Reference Model operates with its own guidelines and procedures that integrates effectively with any part of the organization that is not agile. It owns the agile ecosystem, implements the Scrum values, and assures that Scrum roles are created and supported.

The EAT is the final stop for impediments that cannot be removed by the SoS that feed it. Therefore, it must be comprised of individuals who are empowered, politically and financially, to remove them. The function of the EAT is to coordinate multiple SoS and to interface with any non-agile parts of the organization. As with any Scrum Team, it needs a PO, a SM and a transparent backlog.

Sample Diagram showing an EAT coordinating 5 groupings of 25 teams:
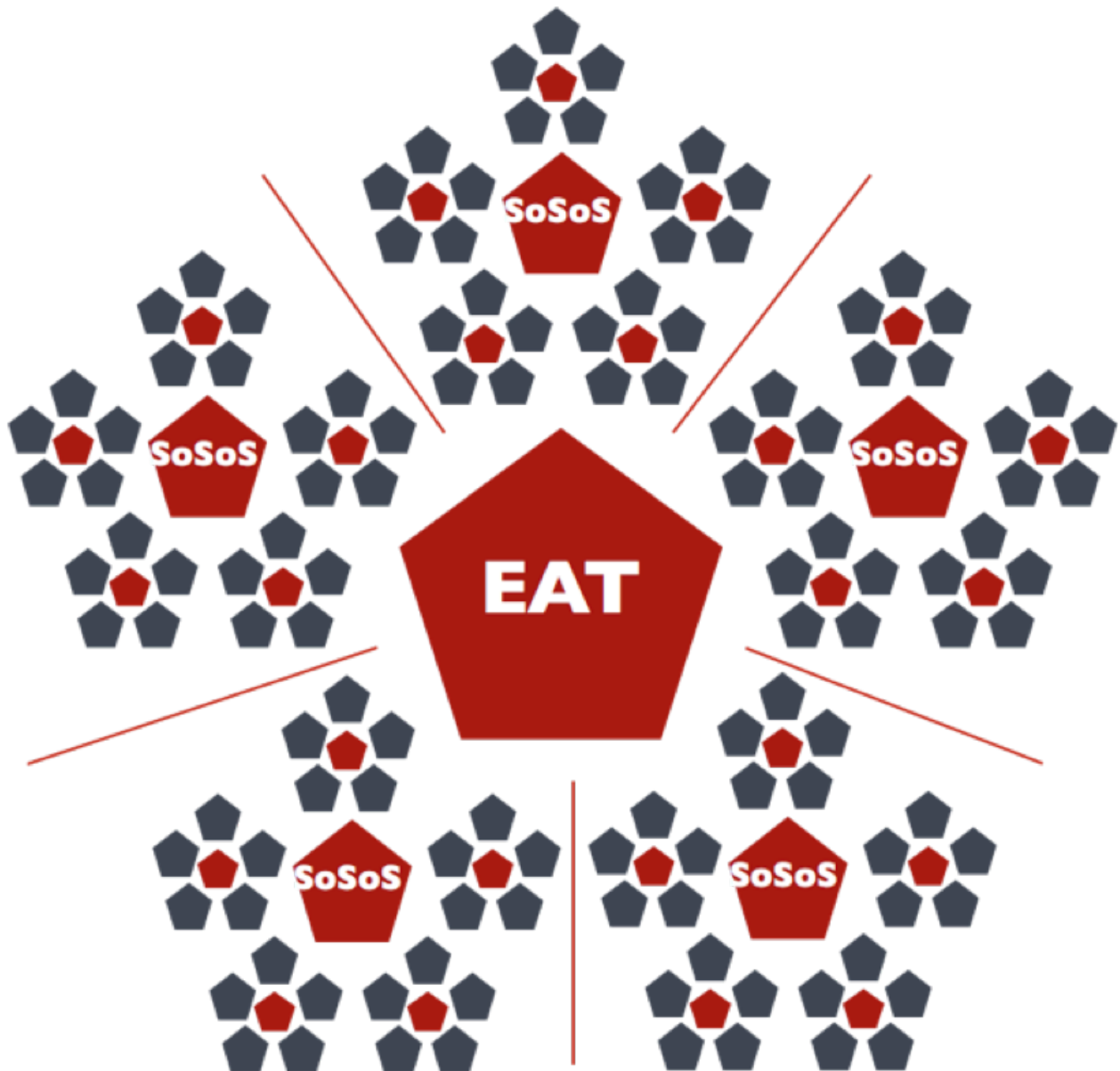
EAT Backlog and Responsibilities

The EAT is accountable for the quality of Scrum within the organization. As such, the entire SM organization reports to the EAT.

The EAT's role is to create an Organizational Transformation Backlog (a prioritized list of the agile initiatives that need to be accomplished) and see that it is carried out. The EAT ensures that a Product Owner organization is created and funded and that this organization

is represented on the EAT to support these efforts.

The EAT's responsibilities include but are not limited to:

- · creating an agile operating system for the Reference Model as it scales through the organization, including corporate operational rules, procedures, and guidelines to enable agility.
- · measuring and improving the quality of Scrum in the organization.
- · building capability within the organization for business agility.
- · creating a center for continuous learning for Scrum professionals.
- · supporting the exploration of new ways of working.

Executive MetaScrum Team (EMT)

The **Executive MetaScrum Team (EMT)** fulfills the Product Owner role for the entire agile organization. This leadership team owns the organizational vision and sets the strategic priorities, aligning all the teams around common goals. The Product Owner Teams enable a network design of Product Owners which is infinitely scalable with their associated SoS. The Product Owner Team for the entire agile organization meets with key stakeholders at the **MetaScrum Event** (below).

Sample diagram showing an EMT coordinating 5 groups of 25 teams:

At least once per Sprint, the Executive MetaScrum Team holds a stakeholder alignment meeting, the **MetaScrum Event**.

- · At the MetaScrum event, decisions affecting the entire organization get made. Changing direction of the organization, having to fire 20
- · The Chief Product Owner presents the Product Backlog to the MetaScrum Team, business owners who control funding, personnel, and customer commitments. They address any changes needed to strategy, funding, resource allocation, and deployments, and collaborate with the Chief Product Owner to build an agreement on a Product Backlog that they will support until the next MetaScrum Event.

Scaling

Scaling the SoS

Depending upon the size of the organization or implementation, more than one SoS may be needed to deliver a very complex product. In such cases, a **Scrum of Scrum of Scrums (SoSoS)** can be created out of multiple Scrums of Scrums. The SoSoS is an organic pattern of Scrum Teams which is infinitely scalable. Each SoSoS should have SoSoSMs and scaled versions of each artifact and event.

Scaling the SoS reduces the number of communication pathways within the organization so that complexity is encapsulated. The SoSoS interfaces with a SoS in the exact same manner that a SoS interfaces with a single Scrum Team, which allows for linear scalability.

Sample Diagrams:
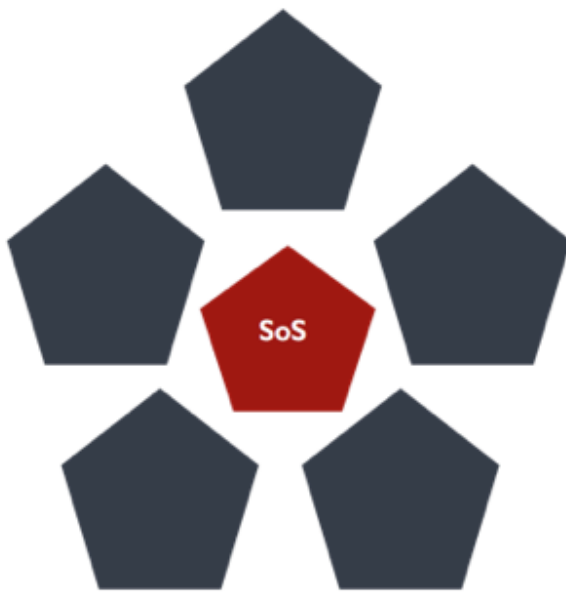
SoS of 5 Teams   SoSoS of 25 Teams

NOTE: While the Scrum Guide defines the optimal team size as being 3 to 9 people, Harvard research determined that optimal team size is 4.6 people (on average).[1] Experiments with high performing Scrum Teams have repeatedly shown that 4 or 5 people doing the work is the optimal size. It is essential to linear scalability that this pattern be the same for the number of teams in a SoS. Therefore, in the above and following diagrams, pentagons were chosen to

---

[1] Hackman, J Richard, *Leading teams: Setting the stage for great performances*, Harvard Business Press, 2002

represent a team of 5. These diagrams are meant to be examples only, your organizational diagram may differ greatly.

Scaling the Product Owner Team

Just as SoS can grow into SoSoS, Product Owner Teams also expand by the same mechanism. There is no specific term associated with these expanded units, nor do the CPO's of them have specific expanded titles. We encourage each organization to develop their own. For the following diagrams, we have chosen to add an additional "Chief" to the title of those PO's as they magnify out.

Sample diagrams:

MetaScrum of 5 Teams   MetaScrum of 25 Teams

Scrum Master Cycle - Coordinating the "How"

The SM organization (SM's, SoSM's, and EAT) work as a whole to complete the components of the Scrum Master Cycle. The components unique to the Scrum Master Cycle are:

- · **Continuous Improvement and Impediment Removal**,
- · **Cross-Team Coordination**, and
- · **Deployment**.

Continuous Improvement and Impediment Removal

In order to scale, impediments need to be removed as soon as possible. Unresolved impediments scale with the implementation and cripple productivity. We remove impediments to guarantee a scale-free architecture.

Continuous Improvement and Impediment Removal are repeatedly applied to:

- · identify impediments and reframe them as opportunities,
- · ensure visibility in the organization to effect change,
- · maintain a healthy and structured environment for prioritizing and removing impediments, and
- · verifying the resulting improvements.

Cross-Team Coordination

When multiple teams have to cooperate on the creation of a shared product, they have a strong need to coordinate.

Cross-Team Coordination is essential in Scrum@Scale to:

- · coordinate similar processes across multiple related teams,
- · mitigate cross-team dependencies to ensure they don't become impediments, and
- · maintain alignment of team norms and guidelines for consistent output.

Deployment

Since the goal of the SoS is to function as a release team, the deployment of product falls under their scope, while what is contained in any release falls under the scope of the Product Owners. Therefore, the goals of the Deployment are to:

- · deliver a consistent flow of valuable finished product to customers.
- · integrate the work of different teams into one seamless product.
- · ensure high quality of the customer experience.

Product Owner Cycle - Coordinating the "What"

The PO organization (the Product Owners, the CPO's, and the Executive MetaScrum) work as a whole to satisfy the components of the Product Owner Cycle. The components unique to the Product Owner Cycle are:

- · **Strategic Vision**,
- · **Backlog Prioritization**,
- · **Backlog Decomposition and Refinement**, and
- · **Release Planning**.

Strategic Vision

A compelling vision attracts both customers and great employees. Therefore, in the PO cycle, a Strategic Vision needs to be formulated and communicated both internally and externally.

The goals of setting a Strategic Vision are to:

- · clearly align the entire organization along a shared path forward,
- · compellingly articulate why the organization exists,
- · describe what the organization will do to leverage key assets in support of its mission, and
- · respond to rapidly changing market conditions.

Backlog Prioritization

Proper Prioritization Prevents Poor Performance. Prioritizing the backlog can reveal activities that have negative value, no value or little value. Identifying them and eliminating them provides focus and prevents waste. Therefore, in Scrum@Scale, a crucial component of the PO cycle is Backlog Prioritization.

The goals of Backlog Prioritization are to:

- · identify a clear ordering for products, features, and services to be delivered;
- · reflect value creation, risk mitigation and internal dependencies in ordering of the backlog, and
- · prioritize the high-level initiatives across the entire agile organization prior to Backlog Decomposition and Refinement.

Backlog Decomposition and Refinement

Backlog gets pulled by teams. In order for backlog from higher level teams to be pulled, it might need to be broken down and understood better. This is done in the Backlog Decomposition and Refinement component of the PO cycle.

The goals of Backlog Decomposition and Refinement are to:

- · break complex products and projects into independent functional elements that can be completed by one team in one Sprint.
- · capture and distill emerging requirements and customer feedback.
- · ensure all backlog items are truly "Ready" so that they can be pulled by the individual teams.

Release Planning

The goals of Release Planning are to:

- · forecast delivery of key features and capabilities.
- · communicate delivery expectations to stakeholders.
- · update prioritization, as needed.

Note that Release Planning may encompass one or many releases of the product to a customer. It is a higher-level planning horizon than a single Sprint, usually covering a period of 1 to 6 months.

Connecting the PO/SM Cycles

The PO and SM Cycles have two touchpoints:

- · **Team-Level Process, and**
- · **Product and Release Feedback.**

Both cycles need **Metrics and Transparency**.

Team-Level Process

The **Team-Level Process** is Scrum as prescribed by the Scrum Guide. It constitutes the first touch point between the Scrum Master and Product Owner Cycles. The goals of the Team-Level Process are to:

- · maximize the flow of completed and quality tested work.
- · increase performance of the team over time.
- · operate in a way that is sustainable and enriching for the team.
- · accelerate the customer feedback loop.

Product and Release Feedback

The **Product and Release Feedback** component is the second point where the PO and SM Cycles touch. Product feedback drives continuous improvement through adjusting the Product Backlog while Release feedback drives continuous improvement through adjusting the Deployment mechanisms. The goals of obtaining and analyzing feedback are to:

- · validate our assumptions.
- · understand how customers use and interact with the product.
- · capture ideas for new features and functionality.
- · define improvements to existing functionality.
- · update progress towards product/project completion to refine release planning and stakeholder alignment.
- · identify improvements to deployment methods and mechanisms.

Metrics and Transparency

Radical transparency is essential for Scrum to function optimally, but it is only possible in an organization that has embraced the Scrum values. It gives the organization the ability to honestly assess its progress and to inspect and adapt its products and processes. This is the foundation of the empirical nature of Scrum as laid out in the Scrum Guide.

Both the SM and PO Cycles require metrics that will be decided upon by the separate SM and PO organizations. Metrics may be unique to both specific organizations as well as to specific functions within those organizations. Scrum@Scale does not require any specific set of metrics, but it does suggest that at a bare minimum, the organization should measure:

- · Productivity - e.g. change in amount of Working Product delivered per Sprint
- · Value Delivery - e.g. business value per unit of team effort

· Quality - e.g. defect rate or service downtime
· Sustainability - e.g. team happiness

The goals of having Metrics and Transparency are to:

· provide all decision makers, including team members, with appropriate context to make good decisions.
· shorten feedback cycles as much as possible to avoid over-correction.
· require minimal additional effort by teams, stakeholders or leadership.

Some notes on Organizational Design

The scale-free nature of Scrum@Scale allows the design of the organization to be component-based, just like the framework itself. This permits for rebalancing or refactoring of teams in response to the market. As an organization grows, capturing the benefits of distributed teams may be important. Some organizations reach talent otherwise unavailable and are able to expand and contract as needed through outsourced development. Scrum@Scale enables linear scalability both in size and global distribution while avoiding long lag times, compromised communications, and inferior quality.[2]
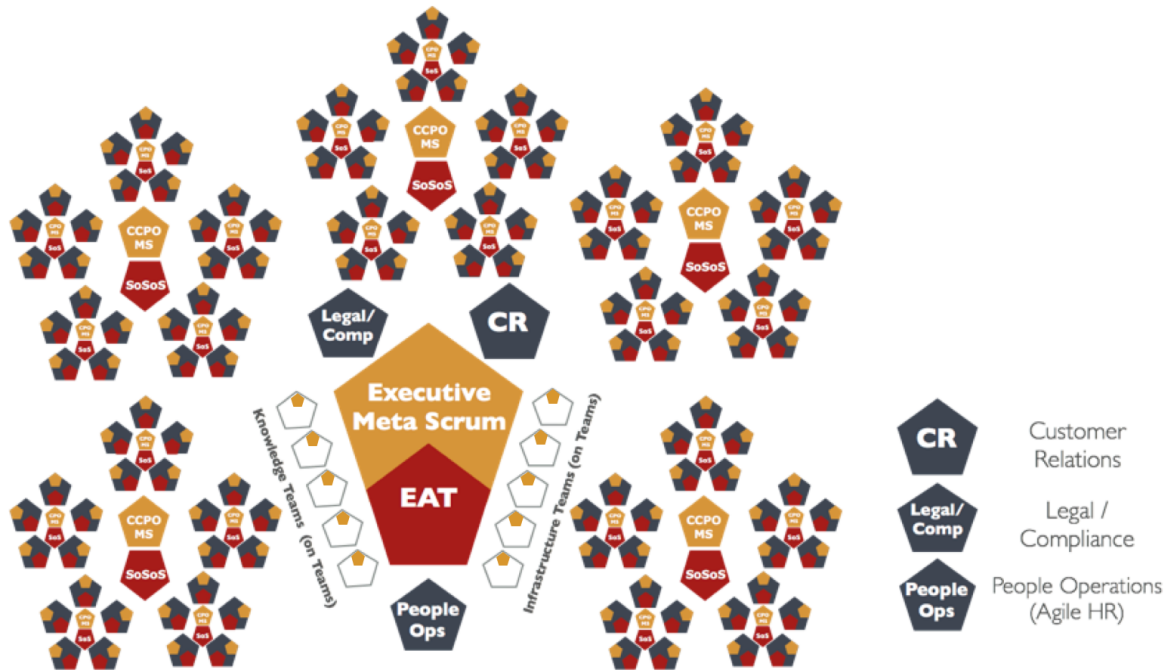

5 SoS's with 2, 3, 4, & 2x5 Teams   3 SoSoS's with 10, 13, & 15 teams

Sample diagrams:

---

[2]Sutherland, Jeff and Schoonheim, Guido and Rustenburg, Eelco and Rijk, Maurits, "Fully distributed scrum: The secret sauce for hyperproductive offshored development teams", AGILE'08. Conference, IEEE: 339-344, 2008

In this organizational diagram, the **Knowledge and Infrastructure Teams** represent virtual teams of specialists of which there are too few to staff each team. They coordinate with the Scrum Teams as a group via service-level agreements where requests flow through a PO for each specialty who converts them into a transparent ordered backlog. An important note is that these teams are NOT silos of individuals who sit together (this is why they are represented as hollow pentagons); their team members sit on the actual Scrum Teams, but they make up this virtual Scrum of their own for the purpose of backlog dissemination and process improvement.

Customer Relations, Legal / Compliance, and People Operations are included here since they are necessary parts of organizations and will exist as independent Scrum Teams on their own, which all of the others may rely upon.

A final note on the representation of the EAT and EMT: in this diagram, they are shown as overlapping since some members sit on both of the teams. In very small organizations or implementations, the EAT and EMT may consist entirely of the same team members.

End Note

Scrum@Scale is designed to scale productivity, to get the entire organization delivering twice the value at half the cost in a significantly improved work environment. Large organizations that properly implement the framework can cut the cost of their products and services while improving quality and innovation.

Scrum@Scale is designed to saturate an organization with Scrum. All teams, including Leadership, Human Resources, Legal, Consulting and Training, and product and service teams, implement the same style of Scrum while streamlining and enhancing an organization.

Well implemented Scrum can run an entire organization and Scrum@Scale becomes the operating system for the organization.

Acknowledgements

---

[3]Sutherland, Jeff, "Inventing and Reinventing SCRUM in five Companies", Sur le site officiel de l'alliance agile, 2001

[4]Sutherland, Jeff, "Future of scrum: Parallel pipelining of sprints in complex projects", Proceedings of the Agile Development Conference, IEEE Computer Society 90-102, 2005.

[5]Sutherland, Jeff and Altman, Igor, "Take no prisoners: How a venture capital group does scrum", Agile Conference, 2009. AGILE'09, IEEE 350-355. 2009