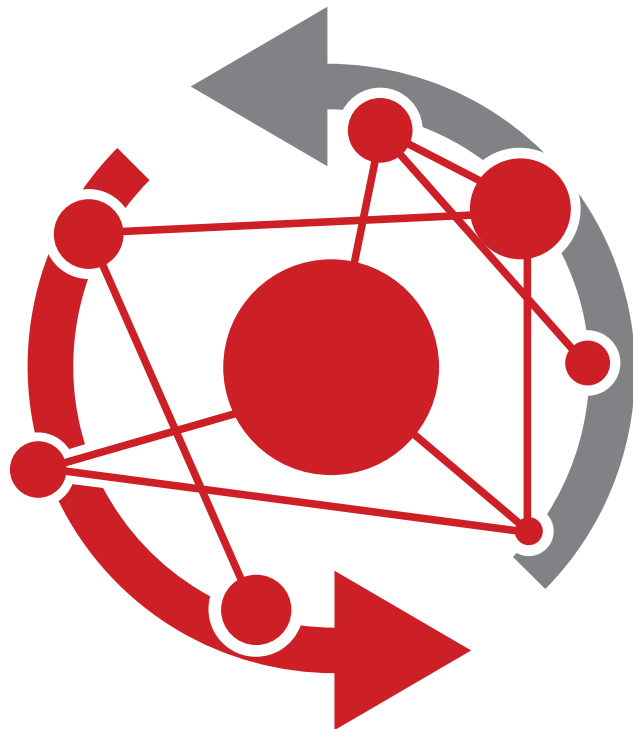# The Scrum@Scale® Guide

The Definitive Guide to Scrum@Scale:
Scaling that Works

Version 1.0 – 10 February 2018

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 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 object-oriented technology. This guide was developed with the input of many experienced Scrum practitioners based on the results of their field work. The goal of this guide is for the reader to be able to implement Scrum@Scale on their own.

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 optimal 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.

By utilizing a scale-free architecture, the 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.

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 (n): A framework within which people can address complex adaptive problems, while productively and creatively delivering 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, performance, and team happiness.

Scrum@Scale (n): A framework within which networks of Scrum teams operating consistently with the Scrum Guide 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
- · Simple to understand - consists of only Scrum teams
- · Difficult to master - requires implementing a new operating model
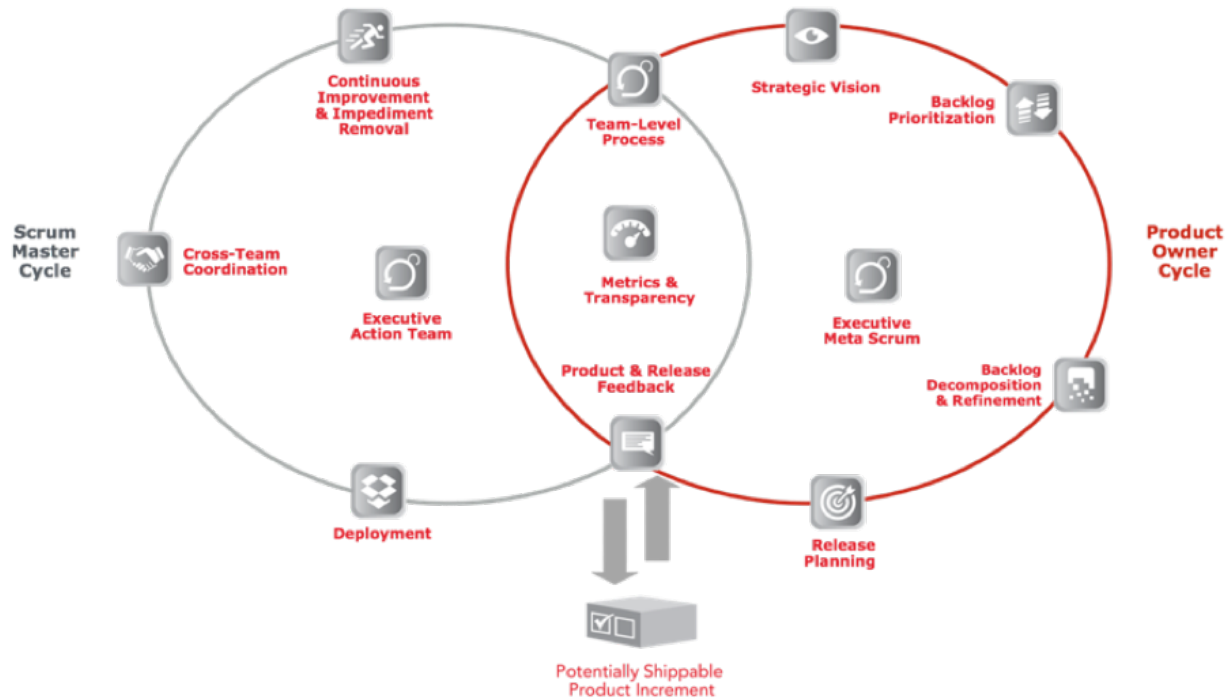
Scrum@Scale is a framework for scaling Scrum. It radically simplifies scaling by using Scrum to scale Scrum. It consists only of Scrum teams coordinated via Scrum of Scrums and MetaScrums.

The component-based nature of Scrum@Scale allows an organization to customize their transformational strategy and implementation. It gives them the ability to target their transformation efforts in the area(s) they deem most valuable or most in need of change and then progress on to others.

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 keep 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.

The Components of the Scrum@Scale® Framework

Continuous
Improvement
& Impediment
Removal

Team-Level
Process

Strategic Vision

Backlog
Prioritization

Scrum
Master
Cycle

Cross-Team
Coordination

Metrics &
Transparency

Product
Owner
Cycle

Executive
Action Team

Executive
Meta Scrum

Product & Release
Feedback

Backlog
Decomposition
& Refinement

Deployment

Release
Planning

Potentially Shippable
Product Increment

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 depend 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 both a servant-leadership and intent-based leadership model,[1] which fosters a positive environment for working at a sustainable

---

[1]Marquet, L David, Turn the Ship Around!: How to Create Leadership at Every Level, Greenleaf Book Group, 2012

pace and putting commitment to deliver customer-facing value at the forefront of our efforts.

Getting Started with Scrum@Scale

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.

Therefore, the first challenge is to create a small set of teams that implements Scrum well. 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, product, and services.

Scrum Master Cycle

Team-Level Process

The **Team-Level Process** constitutes the first touch point between the Scrum Master and Product Owner Cycles, and is laid out clearly in the Scrum Guide. It is composed of three artifacts, five events, and three roles. The goals of the team level process are to:

- · maximize the flow of completed and quality tested work.
- · increase velocity a little each Sprint.
- · operate in a way that is sustainable and enriching for the team.

Coordinating the "How" - The Scrum of Scrums

A set of the teams that have a need to coordinate comprise a **"Scrum of Scrums" (SoS)**. The SoS is a "team of teams,"[2] which hold a **Scaled Daily Scrum (SDS)** event with a representative from each team (usually the team's Scrum Master, although any person or number of people may attend). The SDS exists to coordinate teams and remove impediments to delivering value.

The SDS event mirrors the Daily Scrum in that it optimizes the collaboration and performance of the network of teams. Additionally, the SDS:

- · is time-boxed to 15 minutes or less.
- · must be attended by a representative of each team.
- · is a forum where team representatives address 3 simple questions:
  - · 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?

This team of Scrum Masters is a Scrum Team unto itself which is responsible for a fully integrated set of potentially shippable increments of product at the end of every Sprint from all participating teams. The SoS team needs to be responsive in real-time to impediments raised by participating teams.

A SoS functions as a Release Team and must be able to directly deliver value to customers. To do so effectively, it needs to be consistent with the Scrum Guide; that is, have its own roles, artifacts, and events. This includes a Backlog Refinement event wherein they decide what impediments are "ready" to be removed, how best to remove them, and how the team will know it is "done." 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 standardize those practices across the teams within the SoS.

It needs to have all of the skills necessary to deliver a fully integrated potentially shippable product at the end of every Sprint. It has Product Owner representation to resolve prioritization issues. It may need experienced architects, QA Leaders, and other operational skill sets. When starting Scrum@Scale the teams may not have an infrastructure that supports continuous deployment. This may force the SoS to set up an "integration team" or "release team" that generates the extra work required to overcome engineering deficiencies. The SoS

---

[2]McChrystal, General Stanley and Collins, Tantum and Silverman, David and Fussell, Chris, Team of teams: New rules of engagement for a complex world, Penguin, 2015

is encouraged to address impediments to integration and deployment aggressively as it creates an environment for hyper-productivity, e.g. Amazon has 3300 Scrum teams deploying on average more than once per second[3].

The Scrum of Scrums Master (SoSM)

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

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

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 those 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 SoSoSM's and scaled versions of each artifact & 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.

---

[3]Monica, R. (2018). Personal Communication: Amazon Scrum Implementation. J. Sutherland. Seattle, Amazon.

Sample Diagrams:
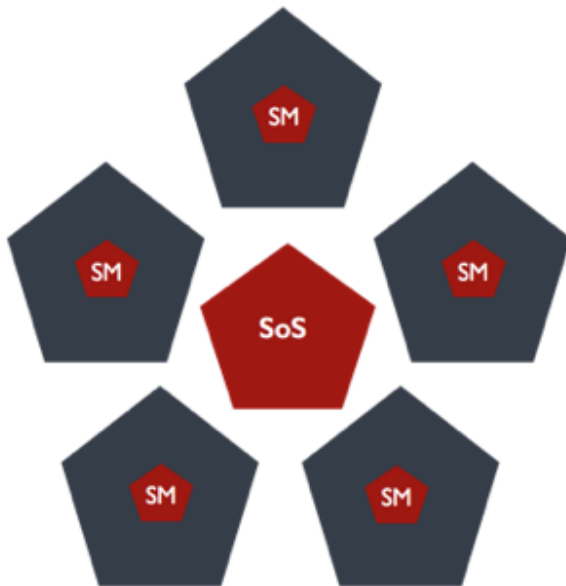
SoS of 5 Teams   SoSoS of 25 Teams

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

The Executive Action Team

The Scrum of Scrums for the entire agile organization is called the **Executive Action Team (EAT)**. The EAT is the final stop for impediments that cannot be removed by the SoS's 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's (or SoSoS's). As with any Scrum team, it needs a PO and SM. It would be best if the EAT met daily as a Scrum team. They must meet at least once per Sprint and have a transparent backlog.

---

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

Sample Diagram showing an EAT coordinating 5 groupings of 25 teams:
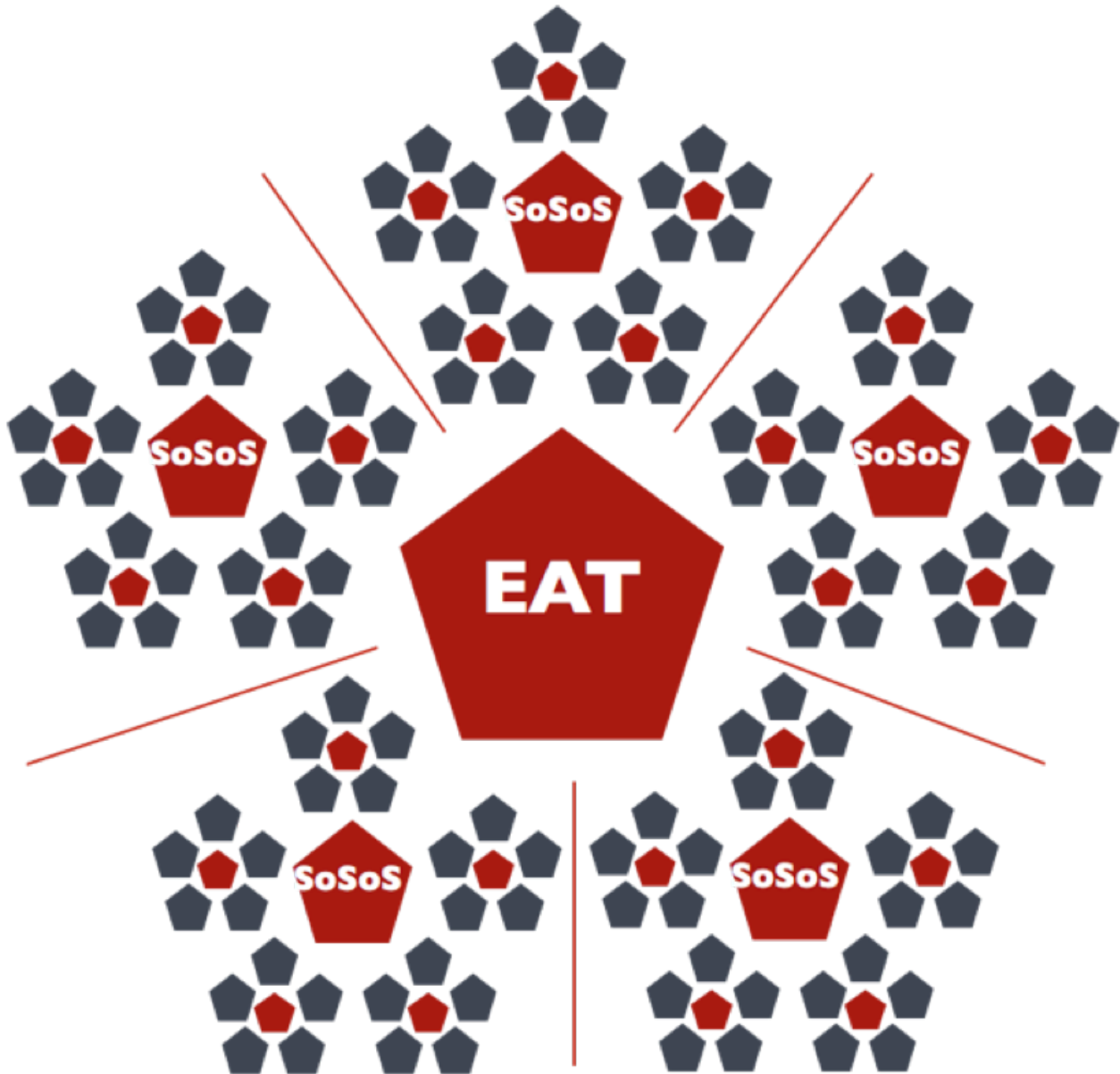
The EAT's Backlog & Responsibilities

Scrum is an agile operating system that is different from traditional project management. The entire SM organization reports into the EAT, which is responsible for implementing this agile operating system by establishing, maintaining, and enhancing the implementation in the organization. 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. For example, if there is a traditional Product Development Life Cycle in the old

organization, a new agile Product Development Life Cycle needs to be created, implemented, and supported. It will typically support quality and compliance issues better than the old method but be implemented in a different way with different rules and guidelines. There are many other aspects of organizational development and governance that may need retuning.

The EAT is accountable for the quality of Scrum within the organization. Its 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.

Finally, the EAT must set up and support a corresponding Product Owner organization through associations of PO's that mirror the SoS's and scale their PO functions. These teams of PO's and key stakeholders are known as **MetaScrums**.

Outputs/Outcomes of the Scrum Master Organization

The SM organization (SoS, SoSoS, and EAT) work as a whole to complete the other components of the Scrum Master Cycle: **Continuous Improvement and Impediment Removal, Cross-Team Coordination, and Deployment**.

The goals of Continuous Improvement and Impediment Removal are to:

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

The goals of Cross-Team Coordination are to:

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

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 MetaScrum

A group of Product Owners who need to coordinate a single backlog that feeds a Scrum of Scrums are themselves a team called a **MetaScrum**. For each SoS there is an associated MetaScrum. A MetaScrum aligns the teams' priorities along a single path so that they can coordinate their backlogs and build alignment with stakeholders to support the backlog. MetaScrums hold a scaled version of Backlog Refinement.

- Each team PO (or proxy) must attend
- This event is the forum for Leadership, Stakeholders, or other Customers to express their preferences

This event occurs as often as needed, at least once per Sprint, to ensure a Ready backlog. The main functions of the MetaScrum are to:

- create an overarching vision for the product & 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.
- create a uniform "Definition of Done" that applies to all teams in the SoS.
- eliminate dependencies raised by the SoS.
- generate a coordinated Release Plan.
- decide upon and monitor metrics that give insight into the product.

MetaScrums, just like SoS's, function as Scrum teams on their own. As such, they need to have someone who acts as a SM and keeps 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 of the teams covered by the MetaScrum. This person is designated as the **Chief Product Owner**.

The Chief Product Owner (CPO)

Through the MetaScrums, Chief Product Owners coordinate priorities among Product Owners who work with individual teams. They align backlog priorities with Stakeholder and

Customer needs. Just like a SoSM, they may be an individual team PO who chooses to play this role as well, or they may be a person specifically dedicated to this role. Their main responsibilities are the same as a regular PO's, but at scale:

- Setting a strategic vision for the whole product.
- Creating a single, prioritized backlog of value to be delivered by all of the teams.
    - these items would be larger stories than that for a team PO.
- Working closely with their associated SoSM so that the Release Plan that the MetaScrum team generates can be deployed efficiently.
- Monitoring customer product feedback and adjusting the backlog accordingly.

Scaling the MetaScrum

Just as SoS's can grow into SoSoS's, MetaScrums can 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.

Some sample diagrams:

MetaScrum of 5 Teams   MetaScrum of 25 Teams

NOTE: As mentioned above, these pentagons represent the ideal sized Scrum teams and ideal sized MetaScrums. These diagrams are meant to be examples only, your organizational diagram may differ greatly.

The Executive MetaScrum (EMS)

The MetaScrums enable a network design of Product Owners which is infinitely scalable alongside their associated SoS's. The MetaScrum for the entire agile organization is the **Executive MetaScrum**. The EMS owns the organizational vision and sets the strategic priorities for the whole company, aligning all the teams around common goals.

Sample diagram showing an EMS coordinating 5 groups of 25 teams:
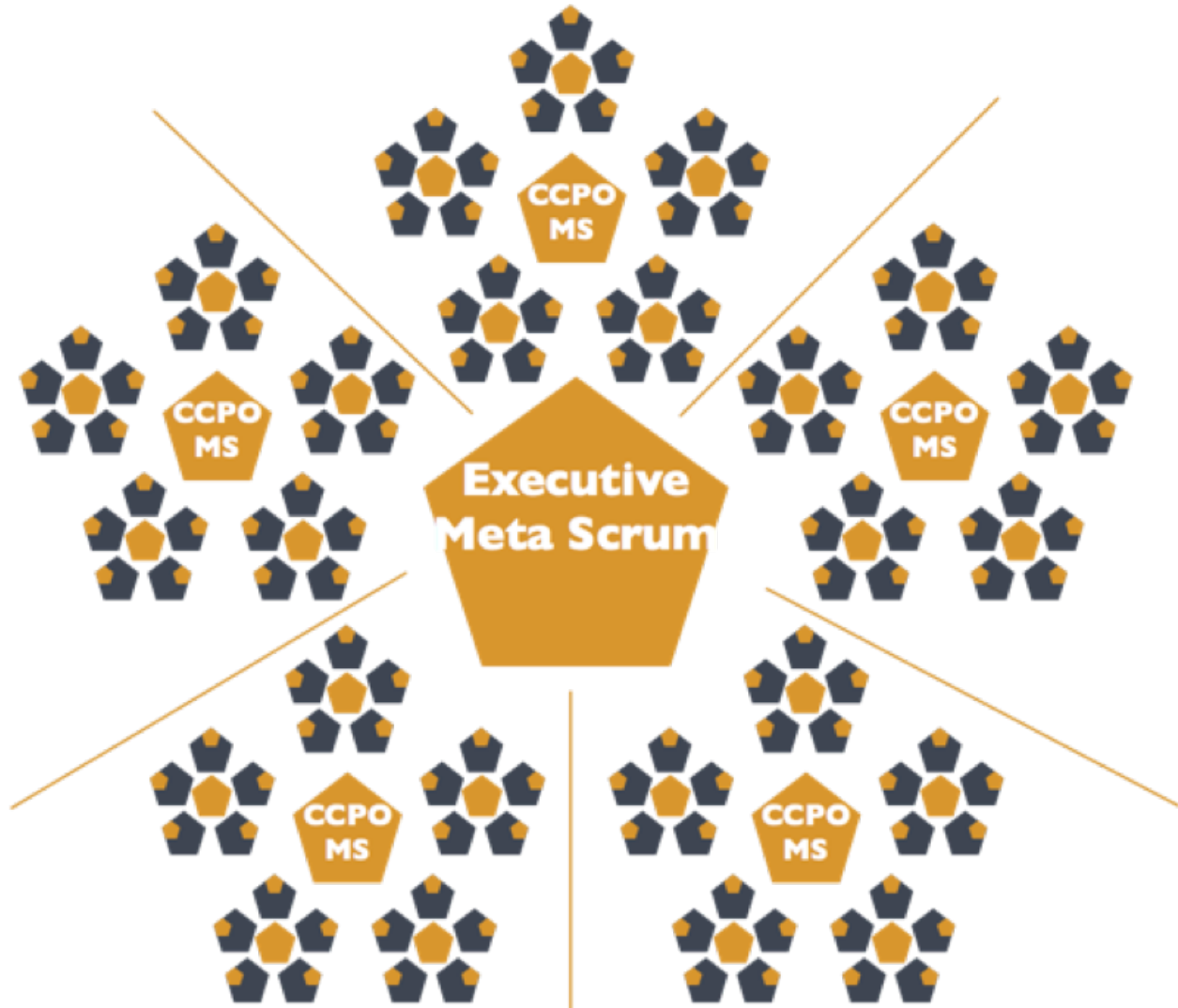
Outputs/Outcomes of the Product Owner Organization

The PO organization (various MetaScrums, the CPO's, and the Executive MetaScrum) work as a whole to satisfy the components of the Product Owner Cycle: **Strategic Vi-**

sion, Backlog Prioritization, Backlog Decomposition & Refinement, and Release Planning.

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.
- · update continuously to respond to rapidly changing market conditions.

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.
- · prioritize the high-level initiatives across the entire agile organization prior to Backlog Decomposition and Refinement.

The goals of Backlog Decomposition & Refinement are to:

- · break complex projects and products 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.

The goals of Release Planning are to:

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

Connecting the PO/SM Cycles

Understanding Feedback

The **Feedback** component is the second point where the PO & 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 & 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 & 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
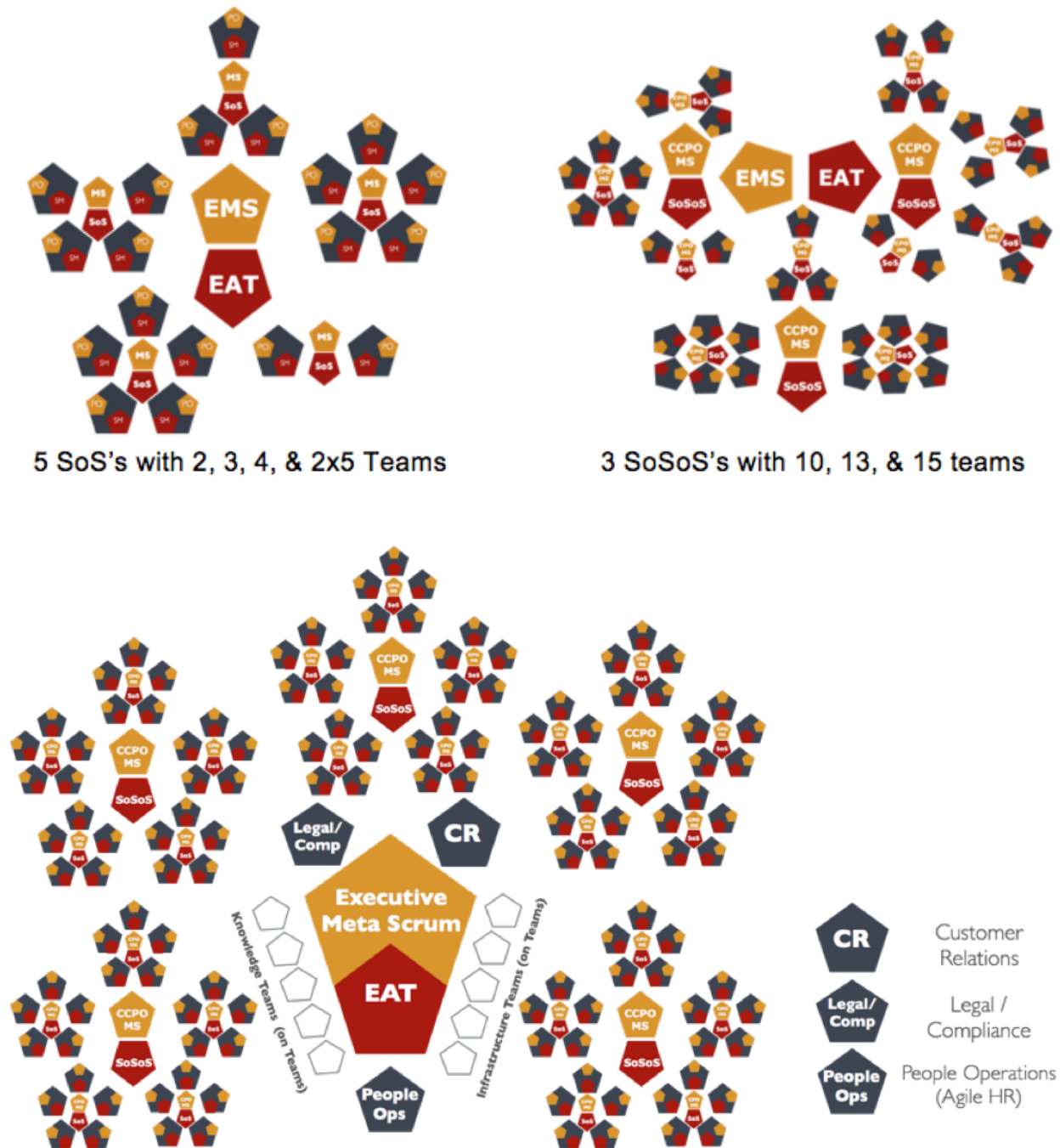- Sustainability - e.g. team happiness

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 shows how to do this while avoiding long lag times, compromised communications, and inferior quality,

enabling linear scalability both in size and global distribution.[5]

5 SoS's with 2, 3, 4, & 2x5 Teams

3 SoSoS's with 10, 13, & 15 teams

CR	Customer Relations
Legal/Comp	Legal / Compliance
People Ops	People Operations (Agile HR)

In this organizational diagram, the **Knowledge & 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

---

[5]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

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 & EMS: in this diagram, they are shown as overlapping since 2 members sit on both of the teams. In very small organizations or implementations, the EAT & EMS may consist entirely of the same team members.

End Note

Scrum@Scale is designed to scale productivity, to get the entire organization doing twice the work in half the time with higher quality and 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 & Training, and product & service teams, implement the same style of Scrum while streamlining and enhancing an organization.

Well implemented Scrum can run an entire organization.

Acknowledgements

We acknowledge IDX for the creation of the Scrum of Scrums which first allowed Scrum to scale to hundreds of teams,[6] PatientKeeper for the creation of the MetaScrum,[7] which enabled rapid deployment of innovative product, and OpenView Venture Partners for scaling Scrum to the entire organization.[8] We value input from Intel with over 25,000 people doing Scrum who taught us "nothing scales" except a scale-free architecture, and SAP with the

---

largest Scrum team product organization who taught us management involvement in the MetaScrum is essential to get 2,000 Scrum teams to work together.

The agile coaches and trainers implementing these concepts at Amazon, GE, 3M, Toyota, Spotify, and many other companies working with Jeff Sutherland have been helpful in testing these concepts across a wide range of companies in different domains.

And finally, Avi Schneier and Alex Sutherland have been invaluable in formulating and editing this document.