

Scrum@Scale[®] 指南

Scrum@Scale 权威指南：
有效规模化



版本 1.01 – 2018.06.18

版权所有©2006-2018 Jeff Sutherland 与 Scrum Inc.， Scrum@Scale 是 Scrum Inc.的注册商标。
本指南基于署名-相同方方共享许可协议 4.0 发布。

©2006-2018 Jeff Sutherland and Scrum Inc., All Rights Reserved Scrum@Scale is a registered trademark of Scrum Inc.

Released under Creative Commons 4.0 Attribution-Sharealike License

目录

SCRUM@SCALE 指南的目的	3
为什么要 Scrum@Scale?	3
Scrum@Scale 的定义	3
价值观驱动的文化	5
开始使用 Scrum@Scale	6
SCRUM MASTER 环路 SCRUM MASTER CYCLE	6
团队级过程	6
高管行动团队	9
产品所有者环路 PRODUCT OWNER CYCLE	11
协调“什么”（“What”） – MetaScrum	11
高管 MetaScrum（EMS）	13
连接 PO/SM 环路	14
理解反馈	14
度量和透明性	14
关于组织设计的一些说明	15
结束语	17
致谢	17
参考文献	18
翻译	18

Scrum@Scale 指南的目的

正如 Scrum 指南中最初概述的那样，Scrum 是一个用于单个团队开发、交付和维护复杂产品的框架。自创建以来，其使用已扩展到需要多个团队合作来创建产品、流程、服务和系统。创建 Scrum @ Scale 是为了有效协调这一新型的团队生态系统，从而优化组织的整体战略。它通过自由扩展（scale-free）架构建立“最简可行的官僚体系”来实现这一目标，自然地将单个 Scrum 团队的功用（function）扩展至整个组织中。

本指南包含组成 Scrum @ Scale 框架的组件的定义，包括其规模化角色（scaled roles）、规模化事件（scaled events）和企业级工件（enterprise artifacts），以及将它们组织在一起的规则。

基于 Scrum、复杂自适应系统理论（Complex Adaptive Systems theory）、博弈论（game theory）和面向对象技术（object-oriented）等背后的基本原理，Jeff Sutherland 博士开发了 Scrum @ Scale。本指南是在采纳许多经验丰富的 Scrum 实践者的输入，基于他们的现场工作（field work）成果基础上开发的。本指南的目标是让读者能够自行实施 Scrum @ Scale。

为什么要 Scrum@Scale?

Scrum 旨在让单个团队能够在保持可持续开发速度的同时以最佳能力工作。在该领域，人们发现随着组织中 Scrum 团队数量的增长，这些团队的产出（可工作产品）和团队速度开始下降（由于跨团队依赖性和工作重复性等问题）。很明显，为了实现线性可扩展性，需要一个有效协调这些团队的框架。Scrum @ Scale 旨在通过其自由扩展的架构（scale-free architecture）来实现这一目标。

通过使用自由扩展的架构，组织不会受到一组任意规则（a set of arbitrary rules）确定的特定方式的限制；而是可以基于其独特的需要（need）和可持续的变革速度进行有机增长，这些变化可以被组织的成员所接受。Scrum@Scale 模型的简单性对于自由扩展的架构是必不可少的，并且谨慎地避免引入额外的复杂性，额外的复杂性将导致每个团队的生产力随着多团队的创建而降低。

Scrum @ Scale 旨在扩展至整个组织范围：所有部门、产品和服务。它可以应用于不同领域，包括工业、政府或学术界中的各类组织。

Scrum@Scale 的定义

Scrum: 是一个框架，在此框架中，人们可以解决复杂适应性问题，同时高效并创造性地交付可用最高价值的产品。

Scrum 指南是最小的特性集（the minimal feature set），可通过彻底的透明性实现检视和适应性，从而驱动创新、客户满意度、绩效和团队幸福感。

Scrum @ Scale: 是一个框架，在此框架中，一致采纳 Scrum 指南进行运作的 Scrum 团队网络（networks of Scrum teams）可以解决复杂适应性问题，同时创造性地交付最高价值的产品

注: 这些“产品”可以是硬件、软件、复杂集成系统、流程和服务等，这取决于 Scrum 团队的所处领域。

Scrum @ Scale 是：

- 轻量的——最简可行的官僚体系
- 易于理解的—— 仅由 Scrum 团队组成
- 难以精通的——需要实施全新的运作模型

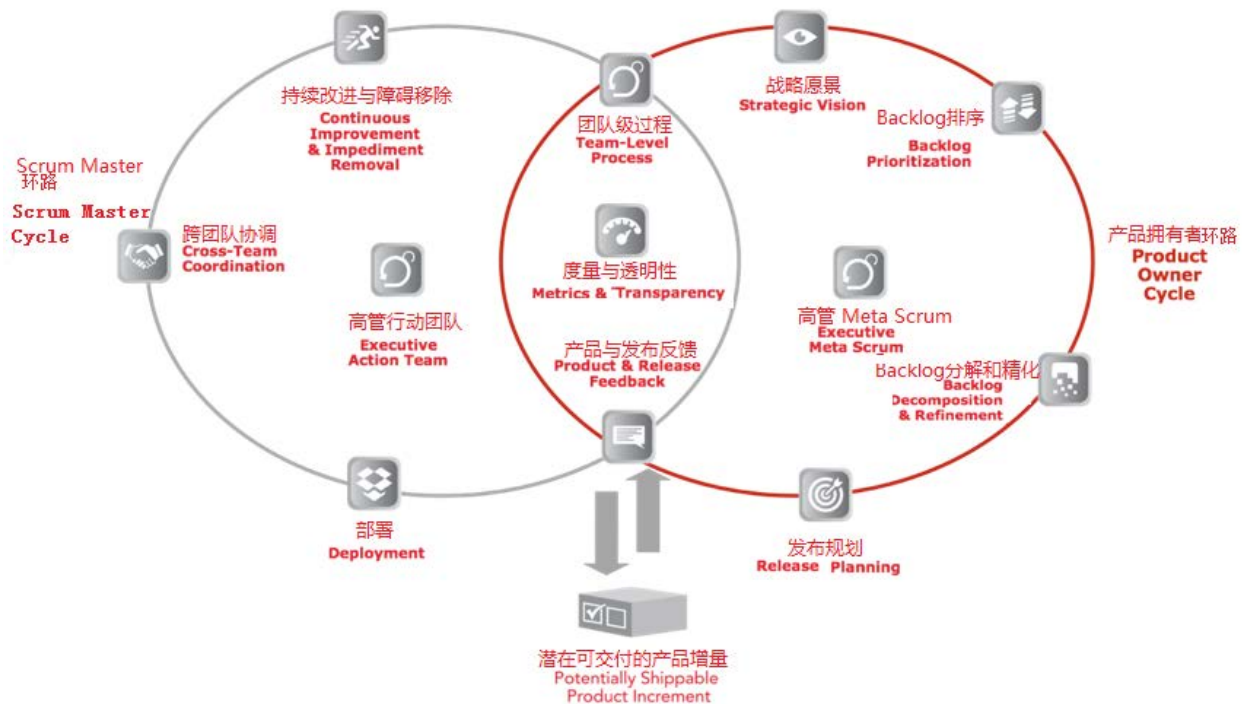
Scrum@Scale 是一个规模化 Scrum 的框架。它通过使用 Scrum 来扩展 Scrum 从根本上简化了规模化 Scrum。

在 Scrum 中，注意需将“什么（what）”与“如何（how）”的问责区分开来。Scrum @ Scale 也采取同样的措施，以便明确理解权限和职责，以消除浪费——使团队无法实现最佳生产力的组织冲突。

Scrum @ Scale 包含允许组织定制其转型策略和实现方式的组件。它使他们能够在他们认为最有价值或最需要改变的领域增量式地优先进行转型工作，然后进入其他领域。

在分离这两个权限时，Scrum @ Scale 包含两个环路（cycle）：Scrum Master Cycle（“如何”）和 Product Owner Cycle（“什么”），彼此具有两个相互触点。总而言之，这些环路造就一个强大的框架，协调多个团队向同一方向努力。

Scrum@Scale 框架的组件



价值观驱动的文化

除了区分“什么”和“如何”的问责之外，Scrum @ Scale 还旨在通过在实证背景（empirical setting）下创造价值观驱动的文化来建立健康的组织。Scrum 的价值观是：开放，勇气，专注，尊重和承诺。这些价值观推动经验决策（empirical decision），这取决于透明，检视（inspection）和适应（adaptation）三大支柱。

开放支持所有工作和过程的透明性，没有这种透明度，就没有能力诚实地检视它们并试图使它们更好地适应。勇气是指以创新的方式实现更快速地提供价值所需的大胆飞跃。

专注和承诺是指我们处理我们的工作职责的方式，将客户价值交付作为最高优先事项。最后，所有一切都必须发生在尊重个人的工作环境中，没有它们，什么都无法创造。

Scrum @ Scale 通过转型领导模型帮助组织蓬勃发展，这为可持续发展创造了积极的工作环境，将承诺交付面向客户价值放在我们努力的第一位。

开始使用 Scrum@Scale

在实施大型团队网络时，为少数团队开发可扩展的**参考模型（Reference Model）**至关重要。部署多个团队时，Scrum 实施中的任何缺陷都会被放大。许多最初的规模化问题将会是那些阻碍高性能和挫伤团队的组织政策和程序或开发实践。

因此，第一个挑战就是建立能够很好地实施 Scrum 的少量团队。这组团队通过解决阻碍敏捷性的组织问题开展工作，并为已知在组织中运行的 Scrum 创建参考模型，并可用作在整个组织范围内规模化 Scrum 的模式。

随着团队参考模型的加速（运作），延迟交付、产生浪费或妨碍业务敏捷性的障碍和瓶颈会变得明显。消除这些问题的最有效方法是将 Scrum 扩散至整个组织以便优化整个价值流。

Scrum @ Scale 通过组织浸透在 Scrum 中，并将速度和质量有机结合起来，与组织的特定战略、产品和服务相一致，从而实现了生产力的线性扩展。

Scrum Master 环路 Scrum Master Cycle

团队级过程

团队级过程构成 Scrum Master 环路和产品所有者环路之间的第一个触点，在 Scrum 指南中有明确的规定。它由 3 个工件，5 个事件和 3 个角色组成。团队级过程的目标是：

- 最大化完成和高质量测试工作的流（flow）。
- 随着时间的推移提升团队绩效。
- 以一种可持续和丰富的方式运作。

协调“如何做事”（“How”） – Scrum of Scrums

一组需要协作的团队组成一个“**Scrum of Scrums**”（SoS）。这一团队本身就是一个 Scrum 团队，负责所有参与团队在每次 Sprint 结束时对产品进行完全地集成出一个潜在可交付产品增量。SoS 充当发布团队，必须能够直接为客户交付价值。为了有效地做到这一点，它需要与 Scrum 指南保持一致；也就是说，有自己的角色、工件和事件：

角色：

它需要具备在每个 Sprint 结束时提供完全集成的潜在可交付产品所需的所有技能。（它可能需要经验丰富的架构师，QA 领导者和其他操作技能组合 skill sets）。SoS 团队需要实时响应参与团队提出的障碍，参与团队的 Scrum Master 必须在 SoS。它有产品所有者（Product Owner, PO）代表来解决优先级问题。SoS 的 Scrum Master 称之为 **Scrum of Scrums Scrum Master (SoSM)**。

事件：

这包括一个 Backlog Refinement 事件，在这个事件中，他们决定哪些障碍已经“准备好”要被移除，如何最佳地移除它们，以及团队如何知道它是“完成的”。应特别关注 SoS 回顾，其中团队代表分享他们各自团队成功的任何学习收获或过程改进，以便在 SoS 内的各个团队中实现这些实践的标准化。SoS 举行**规模化每日 Scrum (Scaled Daily Scrum, SDS)**。SDS 事件镜像了每日 Scrum，它优化了团队网络的协作和性能 (performance)。SoS 团队需要实时响应参与团队提出的障碍，参与团队的 Scrum Master 必须出席 SDS。它应该有 PO 代表来解决优先级问题。来自参与团队的任何人或数量的人按照需要都可以参加。SDS 的存在是为了协调团队并移除交付价值的障碍。

另外，SDS：

- 时间限制在 15 分钟以内。
- 每个团队必须派代表出席。
- 是一个团队代表们解决 3 个简单问题的论坛：
 - 我的团队有什么障碍阻止他们完成他们的 Sprint 目标（或影响即将发布的版本）？
 - 我的团队是否有任何工作阻碍其他团队完成 Sprint 目标（或影响其即将发布的版本）？
 - 我们是否发现了团队之间的任何新的依赖关系，或者发现了一种解决现有依赖关系的方法？
 - 我们发现哪些改进可以跨团队使用？

Scrum of Scrums Scrum Master (SoSM)

SoSM (the Scrum of Scrums Scrum Master, SoSM) 为联合团队的工作 (effort) 的发布负责, 并且他们必须做到:

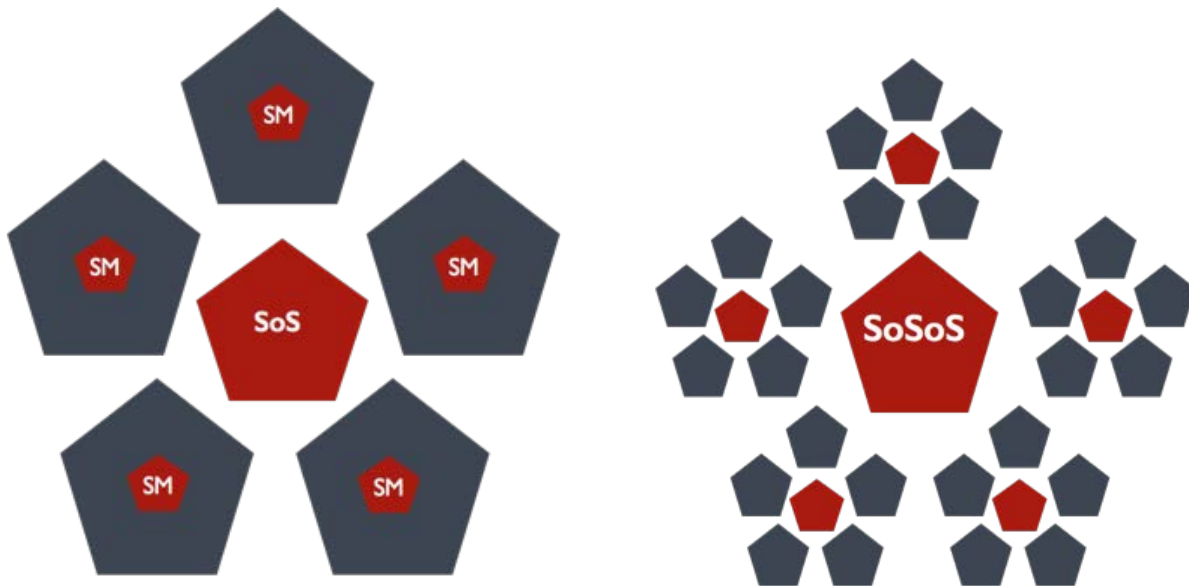
- 使过程可见。
- 使组织可以看到障碍 backlog。
- 移除团队无法解决的障碍。
- 引导 (facilitate) 障碍优先级排序, 特别关注跨团队的依赖关系和 Backlog 的分配 (distribution)。
- 提高 Scrum of Scrums 的效果。
- 与 PO 紧密合作, 每个 Sprint 部署至少一个潜在可发布的产品增量。
- 利用 PO 的发布计划协调多团队的部署工作。

规模化 SoS

根据组织或实施的规模, 可能需要多个 SoS 来交付非常复杂的产品。在这些情况下, 可以在多个 Scrum 的 Scrum 中创建一个 **Scrum of Scrum of Scrum (SoSoS)**。SoSoS 是一个有机模式, 其中 Scrum 团队可以无限扩展。每个 SoSoS 都应该有 SoSoSM 和每个工件和事件的规模化版本。

规模化 SoS 减少了组织内部的沟通路径数量, 因此复杂性被封装起来。SoSoS 与 SoS 接口的方式完全相同, 即 SoS 与单个 Scrum 团队接口, 从而实现线性可扩展性。

示例图:



5 个团队的 SoS

25 个团队的 SoSoS

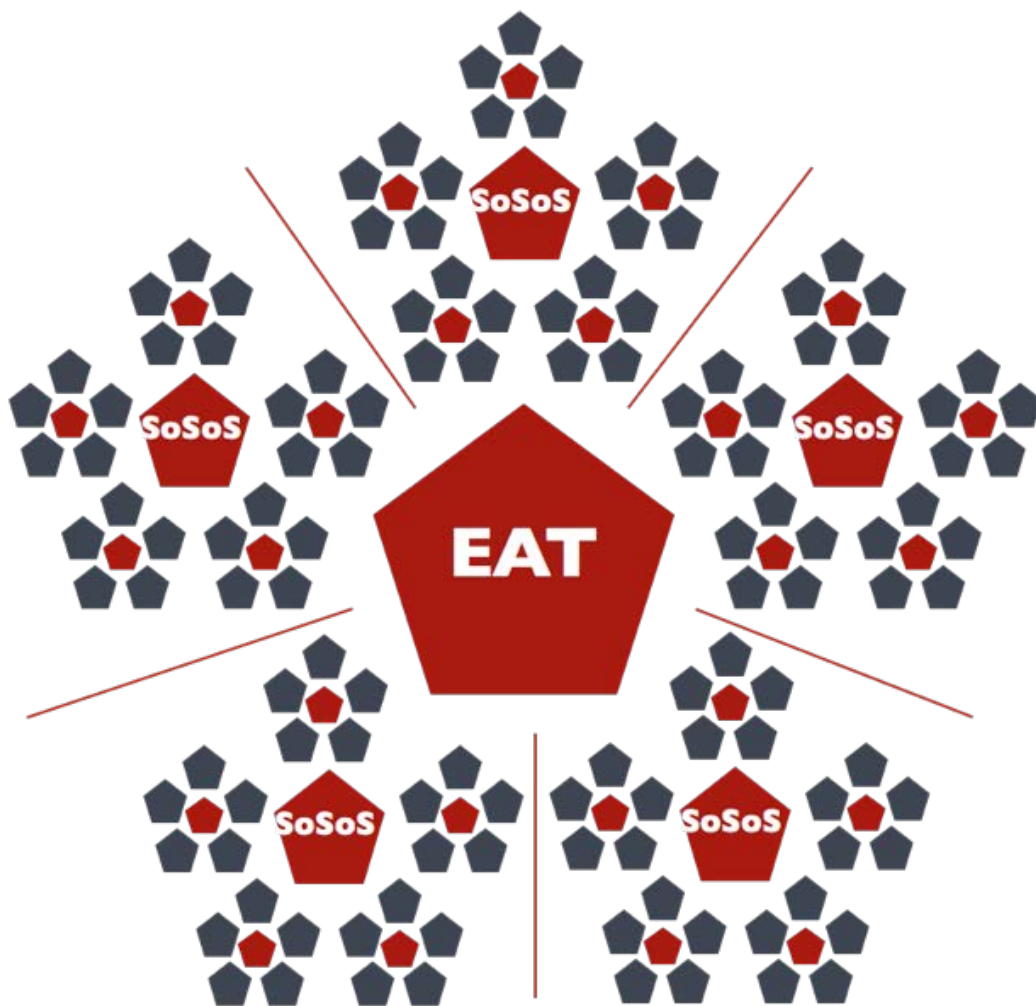
注: 虽然 Scrum 指南将最佳团队规模定义为 3 到 9 人, 但哈佛大学的研究确定最优团队规模为 4.6 人¹。高绩效 Scrum 团队的实验一再表明, 4 或 5 人做这项工作是最优规模。线性可扩展性至关重要, 这种模式对于 SoS 中的团队数量是相同的。因此, 在上图和下图, 选择五边形来表示一个 5 人团队。这些图仅仅是示例, 您的组织图可能会与此差别很大。

高管行动团队

整个敏捷组织的 Scrum of Scrums 被称为**高管行动团队（Executive Action Team, EAT）**。这一领导团队在组织中创建了一个敏捷穹顶（agile bubble），参考模型以其自己的指导方针和程序运行，这些指导方针和程序可以有效地与组织的任何非敏捷部分集成。它拥有敏捷生态系统，实现敏捷价值观，并确保创建和支持 Scrum 角色。

EAT 是所有障碍（SoS 无法解决的）的终点站，这些障碍无法由产生它的 SoS 移除。因此，它必须由在政治和财务上得到充分授权的个人组成，以便将障碍移除。EAT 的功能是协调多个 SoS（或 SoSoS）。与任何 Scrum 团队一样，它需要一个 PO 和一个 Scrum Master。如果 EAT 也像 Scrum 团队一样每天见面，那是最好。每个 Sprint 他们必须至少见面一次，并且需要有透明化的 Backlog。

示例图显示了一个协调 5 组 25 个团队的 EAT：



EAT 的 Backlog 和责任

Scrum 是一个敏捷运作系统，与传统的项目管理不同。整个 SM 组织报告给 EAT，后者负责通过在组织内建立、维护和增强其敏捷运作系统。

EAT 的角色是创建一个组织转型 Backlog（一份已排序的列表，包括需要完成的敏捷举措），并确保其得到执行。例如，如果组织原已存在传统的产品开发生命周期，则需要创建、实现和支持新的敏捷产品开发生命周期。它通常会比旧方法更好地支持质量和合规事项，但以不同的规则和指导方针以不同的方式实施。EAT 确保创建和资助 PO 组织，并且在 EAT 中有代表该组织以支持这些工作。

EAT 对组织内部的 Scrum 运行质量负责。其职责包括但不限于：

- 为参考模型创建敏捷运作系统，以便可以扩展至整个组织，包括适当的企业运营规则、程序（procedures）和指南以提升敏捷性。
- 衡量和改进组织中 Scrum 的运行质量。
- 在组织内部构建业务敏捷性的能力。
- 为 Scrum 专业人员创建一个持续学习中心。
- 支持探索新型工作方式。

最后，EAT 必须建立并支持相应的产品所有者组织，通过比照 SoS，聚焦 PO 群体建立和支持相应的 PO 协会并扩展其 PO 职能。这些 PO 团队和关键利益相关者团队被称为 **MetaScrum**。

Scrum Master 环路的输出/成果

SM 组织（SoS，SoSoS 和 EAT）作为一个整体来完成 Scrum Master 循环的其他组成部分：**持续改进和移除障碍，跨团队协调和部署。**

持续改进和移除障碍的目标是：

- 识别障碍并将它们重新框架（reframe）为机遇。
- 维护一个健壮和结构化的环境来排序和移除障碍，然后验证最终的改进。
- 确保组织中的可见性以促成变革。

跨团队协调的目标是：

- 协调跨多个相关团队的类似过程。
- 缓解跨团队依赖关系，以确保它们不会变成障碍。
- 保持团队规范和准则的一致性，以实现一致的输出。

由于 SoS 的目标是作为发布团队发挥作用，因此产品的部署属于他们的范围，而任何发布中包含的内容都属于产品所有者的范围。因此，部署的目标是：

- 为客户提供一致的有价值的产品流。
- 将不同团队的工作整合成一个无缝产品。
- 确保高质量的客户体验。

产品所有者环路 Product Owner Cycle

协调“什么”（“What”） – MetaScrum

一组产品所有者有必要协调一个唯一的 Backlog，以提供团队网络来工作，他们自己就形成一个团队称之为 **MetaScrum**。对于每个 SoS 都有一个关联的 MetaScrum。MetaScrum 将团队的优先顺序沿着同一路径进行调整，以便他们能够协调其 Backlog，并与利益相关方保持一致以支持 Backlog。

MetaScrum 举行规模化版本的 Backlog Refinement，**规模化 Backlog 精化会议**（Scaled Backlog Refinement meeting）：

- 每个团队的 PO（或代理）必须参加。
- 这一事件是领导者、利益相关者或其他客户表达他们各自选择倾向的论坛。

这一事件按需要发生，每个 Sprint 至少一次，以确保一个“准备就绪”的 Backlog。MetaScrum 的主要职责是：

- 为产品创建一个总体愿景并使其对组织可见。
- 与关键利益相关方保持一致，以确保支持 Backlog 的实现。
- 创建惟一的优先级已排序的 Backlog；确保避免重复工作。
- 创建适用于 SoS 中所有团队的统一的“完成的定义（Definition of Done）”。
- 消除 SoS 引起的依赖关系。
- 生成一份整合的发布计划。
- 识别并监控能够洞察产品的度量指标，并基于其进行决策。

类似于 SoS，多个 MetaScrum 本身也作为 Scrum 团队来运作。所以，需要有人承担 SM，并使团队在讨论中保持正常的轨道上。他们还需一名负责协调为 MetaScrum 覆盖的所有团队创建惟一的 Product Backlog 的人员。此人被指定为**首席产品所有者（Chief Product Owner, CPO）**。

首席产品所有者(CPO)

通过 MetaScrum，首席产品所有者与各个团队产品所有者协调优先级。他们将 Backlog 优先级与利益相关方和客户需求相对齐。就像 SoSM 一样，他们也可能是某一个团队的 PO 扮演这个角色，或者他们可能是由专人承担这一角色。他们的主要职责与常规 PO 相同，但在规模化时：

- 为整个产品制定战略愿景。
- 创建一个由所有团队交付价值的惟一优先级已排序的 Backlog。
 - 这些事项 (item) 相对于一个团队 PO 的故事而言可以是规模更大的故事。
- 与相应的 SoSM 密切合作，以便可以高效部署 MetaScrum 团队创建的发布计划。
- 监控客户对产品反馈并相应地调整 Backlog。

规模化 MetaScrum

就像 SoS 可以增长为 SoSoS 一样，MetaScrum 也可以通过相同的机制进行扩展。没有与这些扩展单元相对应的特定术语，它们的 CPO 也没有特定的扩展头衔。我们鼓励每个组织自行开发。下图中，我们选择在这些 PO 的标题放大时添加一个额外的“首席 Chief”来突显。

一些示例图：



5 个团队的 MetaScrum



25 个团队的 MetaScrum

注：如上所述，这些五边形代表理想规模的 Scrum 团队和理想规模的 MetaScrum。这些图只是为了举例，你的组织图可能会与此差别很大。

高管 MetaScrum (EMS)

MetaScrum 使 PO 及其与之对应的 SoS 能够以一种的网络设计进行无限地扩展。整个敏捷组织的 MetaScrum 是**高管 MetaScrum**(Executive MetaScrum)。EMS 拥有组织愿景，并为整个公司设定战略优先级，使所有团队围绕共同目标进行对齐。

示例图显示了一个 EMS 协调分为 5 组的 25 个团队：



产品所有者组织的输出/成果

PO 组织（各种 MetaScrum, CPO 和高管 MetaScrum）作为整体工作以满足产品所有者环路的组件：**战略愿景、Backlog 优先级、Backlog 分解和精化**(Backlog Decomposition & Refinement)以及发布计划。

设置战略愿景的目标是：

- 整个组织沿着一条共享路径前行清晰对齐。
- 清晰而有力地阐述组织为什么而存在。

- 描述组织会做什么从而调用其关键资产来支持其使命。
- 响应快速变化的市场状况。

优先级排序的目标是：

- 针对要交付的产品、功能和服务，识别出一个清晰明确的排序。
- Backlog 的排序反映价值创造、风险缓解和内部依赖关系。
- 在 Backlog 分解和精化之前，先在整个敏捷组织的高层举措进行排序。

Backlog 分解和精化的目标是：

- 将复杂的产品和项目分解成独立的功能元素，可由一个团队在一个 Sprint 内完成。
- 捕获和提炼涌现的需求和客户反馈。
- 确保所有 Backlog 项真正“准备就绪”，以便它们可以被各个团队拉取。

发布计划的目标是：

- 预测关键特性和能力的交付。
- 向利益相关方沟通交付预期。
- 根据需要更新优先级。

连接 PO/SM 环路

理解反馈

反馈组件是 PO 和 SM 循环交叉的第二个接触点。产品反馈通过调整 Product Backlog 来推动持续改进，同时通过调整部署机制，发布反馈推动持续改进。获取和分析反馈的目标是：

- 验证我们的假设。
- 理解客户如何使用产品并与产品互动。
- 捕获新特性和新功能的创意。
- 定义针对现有功能的改进。
- 朝着产品/项目完成的方向更新进度，以优化发布计划并与利益相关者对齐。
- 识别出部署方法和机制的改进项。

度量和透明性

彻底的透明性对于 Scrum 的最佳状态运行的本质，但它只能在拥抱 Scrum 价值观的组织中才可行。它使组织能够诚实地评估其进展，并检视和适应其产品及过程。这是 Scrum 指南中规定的 Scrum 经验主义的基石。

SM 和 PO 环路各自需要的度量分别由 SM 和 PO 组织来决定。衡量对于这两个特定组织以及这些组织内的特定功能来说可能是独一无二的。Scrum @ Scale 并不要求任何特定的度量集，但它确实推荐，至少在最低限度上，组织应该度量：

- 生产力——例如，每个 Sprint 交付的可工作产品的总数变化
- 价值交付——例如，单位团队投入能够带来的业务价值
- 质量——例如，缺陷率或服务宕机时间
- 可持续性——例如，团队幸福感

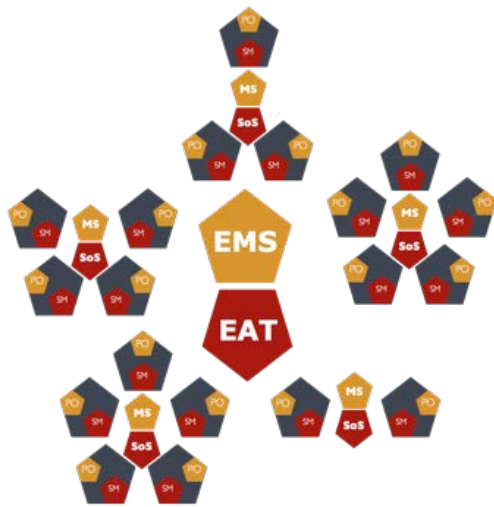
设定这些度量指标和透明性的目标是：

- 为包括团队成员在内的所有决策者提供适当的背景以便作出好的决策。
- 尽可能缩短反馈周期以避免矫枉过正。
- 最小化地要求团队、利益相关者或领导者进行额外投入。

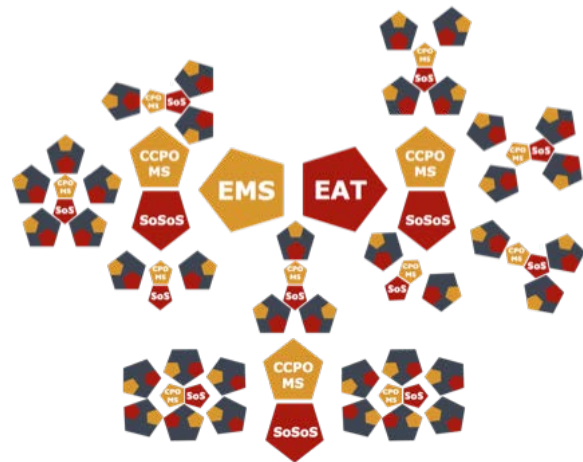
关于组织设计的一些说明

Scrum@Scale 自由扩展的本性，允许将组织设计为基于组件，就像框架本身一样。它允许重新平衡或重构团队以响应市场。随着组织的增长，分布式团队带来的益处可能也很重要。一些组织在无法获取人才的时候则通过外包开发来拓展和签约。Scrum@Scale 展示如何扩展这种情况，同时避免过长的延迟时间、妥协的沟通以及低劣的质量，使组织在规模和地理分布上兼具线性扩展性。²

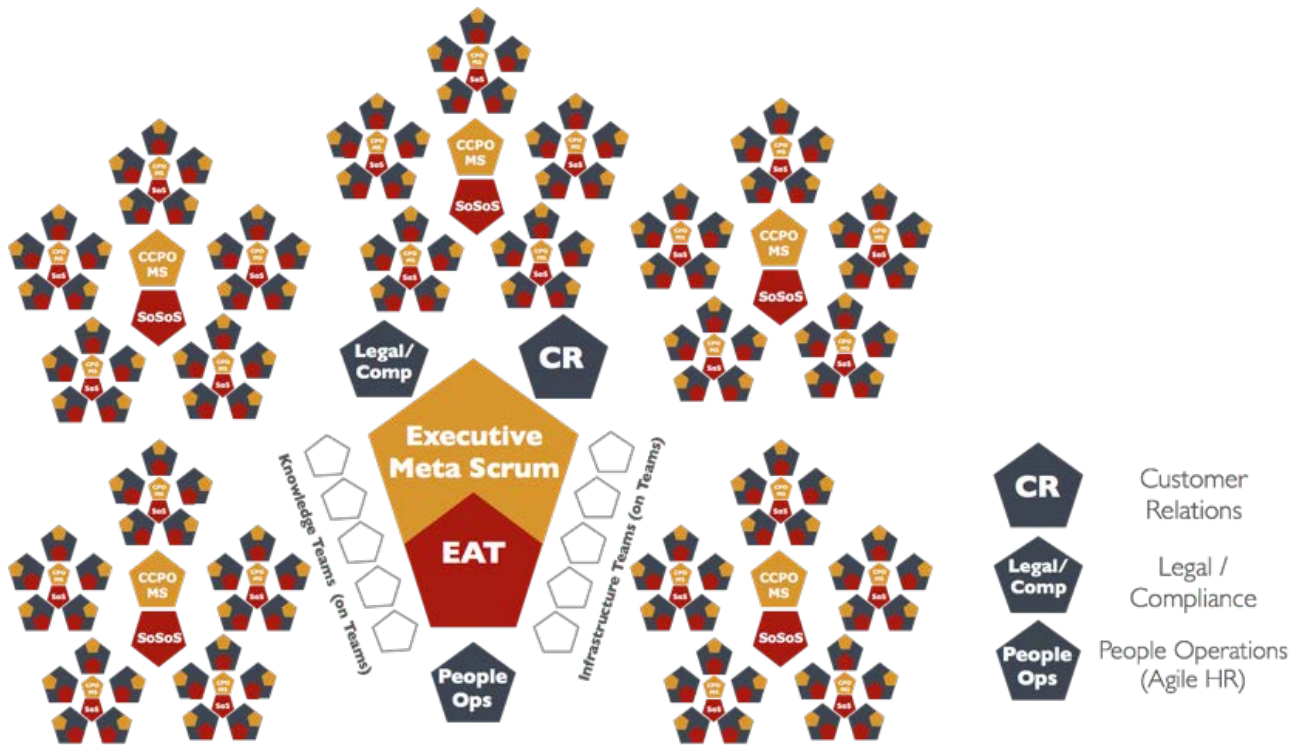
一些示例图：



5 个 SoS 分别包含 2, 3, 4 和 2 个 5 个团队



3 个 SoSoS 分别包含 10, 13 和 15 个团队



在这个组织图中，**知识与基础设施团队(knowledge & Infrastructure Teams)**表示一些虚拟团队，这些专家的数量太少，难以保证在每个团队都配备。他们作为一个组与多个 Scrum 团队进行协调，遵照服务水平协议，每个专业方面请求都流经同一 PO，他将那些请求转换为透明的已排序的 Backlog。值得注意的是，这些团队并不是坐在一起的一群各自为政的个体（这是为什么他们被标记为中空多边形）；这些团队成员都坐在实际的 Scrum 团队当中，但是他们组成这个虚拟 Scrum 是为了传播 Backlog 和过程改进。

客户关系(Customer Relations, CR)、法务/合规(Legal/Compliance, Legal/Comp)和人力运营(People Operations, Agile HR)也包含在这里，因为他们组织中必要的部分，他们将独立于 Scrum 团队而存在，其他人依赖于他们。

关于 EAT 和 EMS 的最后一点：在这个图中，由于有 2 个成员同时存在于这两个团队中，所以两者看起来是重叠了。在非常小的组织或者实施中，EAT 和 EMS 可以是同一批人组成。

结束语

Scrum@Scale 是为了扩大生产力而设计的，使得整个组织在一个显著改善的工作环境中能够高质量地做到事半功倍。在大型组织中适当地应用本框架能够削减产品和服务的成本，并且提升质量和创新能力。

Scrum@Scale 是为了让 Scrum 浸透组织而设计。所有团队，包括领导层、人力资源、法务、咨询与培训、以及产品和服务团队，他们在精简和提升组织的时候采用同一种 Scrum 风格。

良好实施的 Scrum 可以运作整个组织。

致谢

我们感谢 IDX 创建了 Scrum of Scrums，它允许 Scrum 扩展至上百个团队，³ 感谢 PatientKeeper 创建了 MetaScrum，⁴ 它使得创新产品能快速部署，感谢 OpenView Venture Partners 将 Scrum 扩展至整个组织。⁵ 我们珍视来自英特尔的 25,000 多人实施 Scrum 的输入，教会了我们——“无尺度（nothing scales）”——除了一个自由扩展的架构，还要感谢具有最大的 Scrum 团队的 SAP 产品组织，教会了我们让 2000 多个 Scrum 团队一起工作的必要因素就是让管理层参与到 MetaScrum 中。

敏捷教练和培训师们与 Jeff Sutherland 一起工作，在亚马逊、GE、3M、丰田、Spotify、Maersk、Comcast、AT&T 和许多其他公司实施了这些概念，这对于在更广范围的不同领域的公司中验证这些概念是非常有帮助的。

最后，Avi Schneier 和 Alex Sutherland 制定和编辑本文的工作是无价的。

参考文献

1. Hackman, Richard. *Leading Teams: Setting the Stage for Great Performances*. Boston: Harvard Business School Press, 2002.
2. Sutherland, Jeff and Schoonheim, Guido and Rustenburg, Eelco and Rijk, Maurits, “Fully Distributed Scrum: The Secret Sauce for Hyperproductive Offshored Development Team.” AGILE’ 08. Conference, IEEE:339-344, 2008
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

翻译

本简体中文指南是从 Jeff Sutherland 撰写《Scrum @ Scale 指南》，Scrum Inc 所提供的英文原版（2018 版）翻译而来。原文链接：<https://www.scrumatscale.com/scrum-at-scale-guide/>

译者：周建成，敏捷教练（zhoujiancheng@gmail.com , <http://www.lakethinker.com> ）。