

Scrum 指南 2020 中文版

Scrum 指南的目的

在 1990 年代初，我们开发了 Scrum。在 2010 年，我们撰写首版 Scrum 指南，以帮助全世界的人们理解 Scrum。自那时起，我们通过小的功能更新对 Scrum 指南进行了演进。我们是 Scrum 指南的共同后盾。

Scrum 指南包含了 Scrum 的定义。框架的每个元素都有其特定的目的，这对于使用 Scrum 实现全部价值和成果是至关重要的。如果更改 Scrum 的核心设计或理念、遗漏元素或不遵循 Scrum 的规则，将掩盖问题并限制 Scrum 的好处，甚至可能变得毫无用途。

我们关注到 Scrum 在日益复杂的世界中应用越来越广泛。我们很荣幸地看到 Scrum 在许多本质上具有复杂性的工作领域中被采用，超越了 Scrum 的起源领域——软件产品开发。随着 Scrum 的应用范围不断扩大，developers、研究人员、分析师、科学家和其他专家都能在工作中应用。因此，我们在 Scrum 中使用“developers”一词并不是为了排除其他使用者，而是为了简化统称。如果您从 Scrum 中获得价值，那么您可以将自己视为其中一员。

在使用 Scrum 时，可能可以找到、应用和设计适合本文中所描述的 Scrum 框架的模式、过程和见解。它们的描述超出了 Scrum 指南的目的，因为它们因 Scrum 的使用具体情境不同而不同。这些使用 Scrum 框架的特定技巧有很大的差异，因此不在本文中描述。

Scrum 的定义

Scrum 是一个轻量的框架，它通过提供针对复杂问题的自适应解决方案来帮助人们、团队和组织创造价值。

简而言之，Scrum 需要 Scrum Master 营造一个环境，从而：

1. 一名 Product Owner 将解决复杂问题所需的工作整理成一份 Product Backlog。
2. Scrum Team 在一个 Sprint 期间将选择的工作转化为价值的 Increment。
3. Scrum Team 和利益攸关者检视结果并为下一个 Sprint 进行调整。
4. 重复

Scrum 是易于理解的。原封不动地去尝试，并确定其哲学、理论和结构是否有助于实现目标和创造价值。Scrum 框架故意不完整，仅定义了实施 Scrum 理论所需的部分。Scrum 建立在其使用者的集体智慧之上。Scrum 的规则没有为人们提供详细的使用说明，而是指导他们之间的关系和互动。

在 Scrum 框架中，可以使用各种不同的过程、技术和方法。Scrum 可以将一些已有的实践包装进来，也可以甄别出非必须的实践。Scrum 可以凸显当前管理、环境和工作技术的相对成效，以便可以进行改进。

Scrum 理论

Scrum 基于经验主义和精益思维。经验主义主张知识源自实际经验以及根据当前观察到的事物作出的判断所获得。精益思维减少浪费，专注于根本。

Scrum 采纳一种迭代和增量的方法来优化对未来的预测性并控制风险。Scrum 让一群共同拥有所有技能和专长的人员参与进来完成工作，并根据需要分享或获得所需技能。

Scrum 将 4 个正式事件组合在一起以在一个容器型事件 Sprint 中进行检视和适应。这些事件之所以起作用，是因为它们实现了基于经验主义的 Scrum 的三个支柱：透明、检视和适应。

透明

涌现的过程和工作必须对执行工作的人员和接受工作的人员都是可见的。在 Scrum 中，重要的决策是基于其 3 个正式工件的感知状态。透明度较低的工件可能导致做出降低价值并增加风险的决策。

透明使检视成为可能。没有透明的检视会产生误导和浪费。

检视

Scrum 工件和实现商定目标的进展必须经常地和勤勉地检视，以便发现潜在的不良的差异或问题。为了帮助检视，Scrum 以 5 个事件的形式提供了稳定的节奏。

检视使适应成为可能。没有适应的检视是毫无意义的。Scrum 事件旨在激发改变。

适应

如果过程的任何方面超出可接受的范围或所得的产品不可接受，就必须对当下的过程或过程处理的内容加以调整。调整工作必须尽快执行以最小化进一步的偏差。

当所涉人员没有得到授权或不能自我管理 (self-managing) 时，则适应将变得更加困难。在通过检视学到任何新东西时，Scrum Team 会做出相应调整。

Scrum 价值观

Scrum 的成功应用取决于人们变得更加精通践行并内化 5 项价值观

承诺，专注，开放，尊重和勇气

Scrum Team 致力于达成其目标并且相互支持。他们的主要关注点是 Sprint 的工作，以便尽可能地向着这些目标获取最好的进展。Scrum Team 及其利益攸关者对工作和挑战持开放态度。Scrum Team 成员相互尊重，彼此是有能力和独立的人，并因此受到与他们一起工作的人的尊重。Scrum Team 成员有勇气做正确的事并处理那些棘手的问题

这些价值观为 Scrum Team 的工作、行动和行为指引方向。做出的决定、采取的步骤以及使用 Scrum 的方式应强化这些价值观，而不是削弱或破坏它们。Scrum Team 成员通过 Scrum 事件和工件来学习

和探索这些价值观。当 Scrum Team 和与他们一起工作的人们体现这些价值观时，基于经验主义的 Scrum 的三个支柱：透明、检视和适应，就成为现实，并在每个人之间构建信任。Scrum 价值观

Scrum Team

Scrum 的基本单位是小团队，称为 Scrum Team。Scrum Team 由一名 Scrum Master，一名 Product Owner 和 Developers 组成。在 Scrum Team 中，没有子团队或层次结构。Scrum Team 是具有凝聚力的专业团体，一次专注于一个目标，即 Product Goal。

Scrum Team 是跨职能的（cross-functional），这意味着团队成员具有在每个 Sprint 中创造价值而所需的全部技能。他们也是自管理的，这意味着他们在团队内部决定谁做什么、何时做以及如何做。

Scrum Team 规模足够小以保持灵活，同时足够大以便可以在一个 Sprint 中完成重要的工作，通常只有 10 人或更少。总的来说，我们发现较小的团队沟通更好，效率更高。如果 Scrum Team 变得太大，则应考虑将他们重组为多个具有凝聚力的 Scrum Team，每个团队都专注于同一产品。因此，他们应该共享相同的 Product Goal、Product Backlog 和 Product Owner。

Scrum Team 负责所有与产品相关的活动，包括与利益攸关者的协作、验证、维护、运营、实验、研究和开发，以及可能需要进行的其他任何活动。组织组建并授权 Scrum Team 自行管理他们自己的工作。以可持续的速度在 Sprint 中工作可以提高 Scrum Team 的专注度和一致性。

整个 Scrum Team 都有责任在每个 Sprint 中创建有价值的、有用的 Increment。Scrum 在 Scrum Team 中定义了三种特定的职责：Developers、Product Owner 和 Scrum Master。

Developers

Developers 是 Scrum Team 中致力于创建每个 Sprint 可用 Increment 的任何方面的人员。

Developers 所需的特定技能通常很广泛，并且会随着工作领域的不同而变化。但是，Developers 始终要负责：

- 为 Sprint 创建计划，即 Sprint Backlog；
- 通过遵循 Definition of Done 来注入质量；
- 每天根据 Sprint Goal 调整计划；
- 作为专业人士对彼此负责。

Product Owner

Product Owner 负责将 Scrum Team 的工作所产生的产品价值最大化。如何做到这一点可能在组织、Scrum Team 和个体之间存在很大差异。

Product Owner 还负责对 Product Backlog 进行有效管理，包括：

- 开发并明确地沟通 Product Goal；
- 创建并清晰地沟通 Product Backlog 条目（items）；

- 对 Product Backlog 条目进行排序；
- 确保 Product Backlog 是透明的、可见的和可理解的。

Product Owner 可以自己做上述工作，或者也可以将职责委托他人。然而无论如何，Product Owner 是负最终责任的人。

为保证 Product Owner 取得成功，整个组织必须尊重他们的决定。这些决定在 Product Backlog 的内容和顺序中可见，并在 Sprint Review 时透过可检视的 Increment 予以体现。

Product Owner 是一个人，而不是一个委员会。在 Product Backlog 中，Product Owner 可以代表许多利益攸关者的期望要求。那些想要改变 Product Backlog 的人可以尝试去说服 Product Owner 来做到这一点。

Scrum Master

Scrum Master 负责按照 Scrum 指南的游戏规则来建立 Scrum。他们通过帮助 Scrum Team 和组织内的每个人理解 Scrum 理论和实践来做到这一点。

Scrum Master 对 Scrum Team 的效能负责。他们通过让 Scrum Team 在 Scrum 框架内改进其实践来做到这一点

Scrum Masters 是真正的领导者，服务于 Scrum Team 和作为更大范围的组织。

Scrum Master 以多种方式服务于 Scrum Team ，包括：

- 作为教练在自管理和跨职能方面辅导 Scrum Team 成员；
- 帮助 Scrum Team 专注于创建符合 Definition of Done 的高价值 Increment ；
- 促使移除 Scrum Team 工作进展中的障碍；
- 确保所有 Scrum 事件都发生并且是积极的、富有成效的，并且在时间盒（timebox）内完成。

Scrum Master 以多种方式服务于 Product Owner ，包括：

- 帮助找到有效定义 Product Goal 和管理 Product Backlog 的技巧；
- 帮助 Scrum Team 理解为何需要清晰且简明的 Product Backlog 条目；
- 帮助建立针对复杂环境的基于经验主义的产品规划（empirical product planning）；
- 当需要或被要求时，引导利益攸关者协作。

Scrum Master 以多种方式服务于组织，包括：

- 带领、培训和作为教练辅导组织采纳 Scrum ；
- 在组织范围内规划并建议 Scrum 的实施；
- 帮助员工和利益攸关者理解并实施针对复杂工作的经验主义方法（empirical approach）；

- 消除利益攸关者和 Scrum Teams 之间的隔阂。

Scrum 事件

Sprint 是所有其他事件的容器。Scrum 中的每个事件都是检视和适应 Scrum 工件的正式机会。这些事件都是为实现所需的透明度而特别设计的。未能按规定运作任何事件将导致失去检视和适应的机会。

Scrum 使用事件来创造规律性，并以此最小化对 Scrum 中未定义的会议的需要。

最理想的是，所有事件都在同一时间同一地点举行，以便减少复杂性。

Sprint

Sprint 是 Scrum 的核心，在这里创意 (idea) 转化为价值。

它们是固定时长的事件，为期一个月或更短，以保持一致性。前一个 Sprint 结束后，下一个新的 Sprint 紧接着立即开始。

实现 Product Goal 所需的所有工作，包括 Sprint Planning、Daily Scrum、Sprint Review 和 Sprint Retrospective，都发生在 Sprint 内。

在 Sprint 期间：

- 不能做出危及 Sprint Goal 的改变；
- 不能降低质量；
- Product Backlog 按需进行精化 (refined)，和
- 随着学到更多，可以与 Product Owner 就范围加以澄清和重新协商。

Sprint 通过确保至少每个日历月一次对达成 Product Goal 的进展进行检视和适应，来实现可预测性。当 Sprint 的长度拉得太长时，Sprint Goal 可能会失效，复杂性可能会上升，同时风险可能会增加。可以使用较短的 Sprint 来产生更多的学习周期，并将成本与工作投入 (effort) 的风险限制在更短的一段时间内。每个 Sprint 都可以视为一个短期的项目。

存在各种各样的实践来预测进展，例如，燃尽图 (burn-downs)、燃起图 (burn-ups) 或累积流图 (cumulative flows)。尽管被证明是有用的，然而这些实践并不能用来取代经验主义的重要性。在复杂的环境中，未来将要发生什么是未知的。只有已经发生的事情才能用来做前瞻性的决策。

如果 Sprint Goal 已过时，那么就可以取消 Sprint。只有 Product Owner 有取消 Sprint 的权力。

Sprint Planning

Sprint Planning 通过安排在 Sprint 中要做的工作来启动 Sprint。最终的计划是由整个 Scrum Team 协作创建的。

Product Owner 要确保与会者准备好讨论最重要的 Product Backlog 条目，以及它们如何映射到 Product Goal。Scrum Team 还可以邀请其他人参加 Sprint Planning 以提供建议。

Sprint Planning 处理以下话题：

话题一：为什么这次 Sprint 有价值？

Product Owner 提议产品如何在当前的 Sprint 中增加其价值和效用。然后，整个 Scrum Team 将共同制定一个 Sprint Goal，用以沟通当前 Sprint 对利益攸关者有价值的原因。必须在 Sprint Planning 结束之前最终确定 Sprint Goal。

话题二：这次 Sprint 能完成 (Done) 什么？

通过与 Product Owner 讨论，Developers 从 Product Backlog 中选择一些条目，放入当前 Sprint 中。Scrum Team 可以在此过程中精化这些 Product Backlog 条目，从而增加理解和信心。

选择在 Sprint 中可以完成多少任务可能会有挑战。但是，Developers 对他们以往的表现，他们在即将到来的 Sprint 内的产能以及对他们的 Definition of Done 了解得越多，他们对 Sprint 预测就越有信心。

话题三：如何完成所选的工作？

对于每个选定的 Product Backlog 条目，Developers 都会规划必要的工作，以便创建符合 Definition of Done 的 Increment。这通常是通过将 Product Backlog 条目分解为一天或更短的较小条目来完成的。Developers 自行决定如何完成这一工作。没有人告诉他们如何将 Product Backlog 条目转化为价值的 Increment。

Sprint Goal、这次 Sprint 所选出的 Product Backlog 条目加上如何交付它们的计划称之为 Sprint Backlog。

Sprint Planning 是有时间盒限定的，以一个月的 Sprint 来说最多为 8 个小时。对于更短的 Sprint，Sprint Planning 所需时间通常会更短。

Daily Scrum

Daily Scrum 的目的是检视达成 Sprint Goal 的进展，并根据需要调整适应 Sprint Backlog，以调整即将进行的计划工作。

Daily Scrum 是一个属于 Scrum Team 的 Developers 的 15 分钟的事件。为了降低复杂性，它在 Sprint 的每个工作日都在同一时间同一地点举行。如果 Product Owner 或 Scrum Master 正在积极处理 Sprint Backlog 条目，那么他们将作为 Developers 参与其中。

Developers 可以选择他们想要的任何举行 Daily Scrum 的结构和技术，只要他们专注于实现 Sprint Goal 的进展，并为下一工作日的工作制定可行的计划即可。这可以创建专注点并改进自我管理。

Daily Scrum 改善沟通，发现障碍，促进快速决策，从而消除其他会议的需要。

Daily Scrum 并不是唯一一次允许 Developers 调整计划的时间。他们可以在一天中任何时间碰面，详细讨论如何调整适应或重新规划 Sprint 的剩余工作。

Sprint Review

Sprint Review 的目的是检视 Sprint 的成果并确定未来的适应性。Scrum Team 向关键利益攸关者展示他们的工作结果，并讨论 Product Goal 的进展情况。

在 Sprint Review 期间，Scrum Team 和利益攸关者将评审在这次 Sprint 中完成了什么，以及环境发生了什么变化。基于这些信息，与会者可以就下一步的工作进行协作。Product Backlog 也可能会进行调整以适应新的机会。Sprint Review 是一个工作会议，Scrum Team 应避免将其仅限于展示。

Sprint Review 是 Sprint 的倒数第二个事件，Sprint Review 是有时间盒限定的，以一个月的 Sprint 来说，最多为 4 个小时。对于更短的 Sprint，Sprint Review 通常所需的时间更短。

Sprint Retrospective

Sprint Retrospective 的目的是规划提高质量和效能的方法。

Scrum Team 检视最近 Sprint 中有关个体、交互、过程、工具和他们的 Definition of Done 的情况如何。被检查的元素通常随工作领域而变化。识别使他们误入歧途的假设，并探究其起源。Scrum Team 讨论在 Sprint 期间哪些进展顺利，遭遇到哪些问题以及这些问题是如何解决（或未解决）的。

Scrum Team 识别出最有用的改变以提高其效能。最有影响力的改进将尽快得到执行。甚至可以将它们添加到下一个 Sprint 的 Sprint Backlog 中。

Sprint Retrospective 结束 Sprint。它是有时间盒限定的，以一个月的 Sprint 来说，最多为 3 个小时。对于更短的 Sprint，Sprint Retrospective 通常所需的时间更短。

Scrum 工件

Scrum 的工件代表工作或价值。它们旨在最大限度地提高关键信息的透明度。因此，为适应而检视它们的每个人对工件都有相同的基础。

每个工件都包含一个承诺，以确保它提供可增强透明度并聚焦于可度量进展的信息：

- 对于 Product Backlog 而言，它是 Product Goal。
- 对于 Sprint Backlog 而言，它是 Sprint Goal。
- 对于 Increment 而言，它是 Definition of Done。

这些承诺的存在是为了强化经验主义和 Scrum Team 及其利益攸关者的 Scrum 价值观。

Product Backlog

Product Backlog 是一份涌现的和有序的清单，它列出了改进产品所需的内容。它是 Scrum Team 所承担工作的唯一来源。

能够被 Scrum Team 在一个 Sprint 中完成（Done）的 Product Backlog 条目被认为准备就绪，在 Sprint Planning 事件中可供选择。它们通常在精化活动后获得这种透明度。Product Backlog 精化是

将 Product Backlog 条目分解并进一步定义为更小更精确的行为。这是一项持续进行的活动，为 Product Backlog 条目增添细节，例如描述、优先顺序和规模。这些属性通常随工作领域而变化。

将要做这项工作的 Developers 负责使其适当的大小。Product Owner 可以通过帮助 Developers 理解和权衡取舍来影响他们。

承诺：Product Goal

Product Goal 描述了产品的未来状态，可以作为 Scrum Team 制定计划的目标。Product Goal 在 Product Backlog 中。Product Backlog 的其余部分涌现，用来定义“做什么”将实现 Product Goal。

产品是传递价值的载体。它具有明确的边界、已知的利益攸关者和定义明确的用户或客户。产品可以是一种服务、实体产品或其他更抽象的东西。

Product Goal 是 Scrum Team 的长期目标。他们必须先实现（或放弃）一个目标，然后再开始下一个目标。

Sprint Backlog

Sprint Backlog 由 Sprint Goal（为什么做）、为 Sprint 选择的 Product Backlog 条目（做什么）以及交付 Increment 的可执行计划（如何做）组成。

Sprint Backlog 是 Developers 为其制定的计划。它是 Developers 在 Sprint 期间为实现 Sprint Goal 而计划完成的工作，是一个高度可视且实时的工作画面。因此，随着学到更多，Sprint Backlog 在整个 Sprint 期间会进行更新。它应该有足够的细节，以便他们可以在 Daily Scrum 中检视其进展。

承诺：Sprint Goal

Sprint Goal 是 Sprint 的单个目标。尽管 Sprint Goal 是 Developers 的承诺，但它为实现该目标所需的确切工作方面提供了灵活性。Sprint Goal 还创造了连贯性和专注点，鼓励 Scrum Team 一起工作而不是分开独自行动。

Sprint Goal 在 Sprint Planning 事件中确定，然后添加到 Sprint Backlog 中。当 Developers 在 Sprint 期间工作时，他们将 Sprint Goal 铭记在心。如果需要做的工作与预期的不同，他们将与 Product Owner 协作，在不影响 Sprint Goal 的情况下，协商本次 Sprint Backlog 的范围。

Increment

一个 Increment 是迈向 Product Goal 的一块坚实垫脚石。每个 Increment 都是之前所有的 Increment 累加起来的，并经过彻底地验证，以确保整合在一起的所有 Increment 都能工作。为了提供价值，Increment 必须是可用的。

在一个 Sprint 中可以创建多个 Increment。Increment 的总和在 Sprint Review 中展示，从而支持经验主义。但是，Increment 可以在 Sprint 结束之前交付给利益攸关者。Sprint Review 决不应该被视为发布价值的关口

一项工作除非符合 Definition of Done ，否则不能将其视为 Increment 的一部分。

承诺：Definition of Done

Definition of Done 是当 Increment 符合产品所需的质量度量标准时对其状态的正式描述。当一个 Product Backlog 条目符合 Definition of Done 时，就会产生一个 Increment。

Definition of Done 通过使每一个人对作为 Increment 的一部分、什么样的工作算是已完成的工作有一个共同的理解来创建透明。如果一个 Product Backlog 条目不符合 Definition of Done ，那么它就不能发布，甚至不能在 Sprint Review 中展示它。相反，它返回到 Product Backlog 中以供将来考虑。

如果 Increment 的 Definition of Done 是组织标准的一部分，那么所有 Scrum Team 都必须以此为最低标准来遵守。如果它不是组织标准的一部分，那么 Scrum Team 必须制定适合于该产品的 Definition of Done 。

Developers 需要遵守 Definition of Done。如果有多个 Scrum Team 在同一产品上一起工作，那么他们必须一起制定并遵守同样的 Definition of Done 。

结束语

Scrum 是免费的，并在本指南中提供。本文概述的 Scrum 框架是不可改变的。虽然仅实施部分的 Scrum 是可能的，但其结果就不是 Scrum 了。Scrum 仅以完整形式而存在，唯其如此才能有效成为其他技术、方法和实践的容器。

2021 年 4 月 13 日