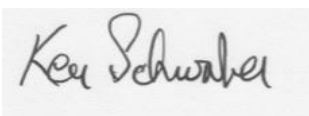
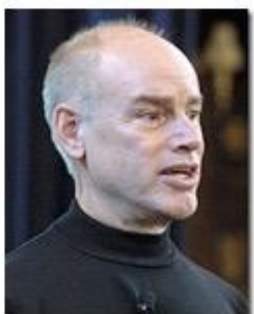


# Scrum 指南™

---

Scrum 权威指南：  
游戏规则

A handwritten signature in black ink, reading "Jeff Sutherland".A handwritten signature in black ink, reading "Ken Schwaber".

2016 年 7 月

由 Ken Schwaber 和 Jeff Sutherland 开发并维护

# 目录

---

|                   |    |
|-------------------|----|
| Scrum 指南的目的.....  | 4  |
| Scrum 的定义.....    | 4  |
| Scrum 理论.....     | 5  |
| Scrum 价值观.....    | 6  |
| Scrum 团队.....     | 7  |
| 产品负责人.....        | 7  |
| 开发团队.....         | 8  |
| Scrum Master..... | 8  |
| Scrum 事件.....     | 10 |
| Sprint.....       | 10 |
| Sprint 计划会议.....  | 11 |
| 每日 Scrum 站会.....  | 13 |
| Sprint 评审会议.....  | 13 |
| Sprint 回顾会议.....  | 14 |
| Scrum 工件.....     | 15 |
| 产品待办列表.....       | 15 |
| Sprint 待办列表.....  | 16 |
| 增量.....           | 17 |
| 工件透明.....         | 17 |
| “完成” 的定义.....     | 17 |
| 结束语.....          | 18 |

©2016 Scrum.Org and ScrumInc. Offered for license under the Attribution Share-Alike license of Creative Commons, accessible at <http://creativecommons.org/licenses/by-sa/4.0/legalcode> and also described in summary form at <http://creativecommons.org/licenses/by-sa/4.0/>. By utilizing this Scrum Guide you acknowledge and agree that you have read and agree to be bound by the terms of the Attribution Share-Alike license of Creative Commons.

|              |    |
|--------------|----|
| 致谢 .....     | 19 |
| 人们 .....     | 19 |
| 历史.....      | 19 |
| 翻译.....      | 19 |
| 版本变更记录 ..... | 19 |

## Scrum 指南的目的

Scrum 是用于开发和持续支持复杂产品的一个框架。本指南包含了 Scrum 的定义，其中包括 Scrum 的角色、事件、工件，以及把它们组织在一起的规则。Ken Schwaber 和 Jeff Sutherland 创造了 Scrum，Scrum 指南也由他们撰写与提供。总之，他们是 Scrum 指南的后盾。

## Scrum 的定义

**Scrum (名词):** Scrum 是一个框架，在此框架中人们可以解决复杂的自适应难题，同时也能高效并创造性地交付尽可能高价值的产品。

Scrum 是：

- 轻量级的
- 易于理解的
- 难以精通的

Scrum 是一个过程框架，自上世纪 90 年代初以来，它就已经被应用于管理复杂产品的开发上。Scrum 并不是构建产品的一种过程或一项技术，倒不如说，它是一个框架，在此框架中您可以使用各种不同的过程和技术。Scrum 让您的产品管理和开发实践的相对成效更加清楚地显现出来，因此您可以去改进它们。

Scrum 框架由 Scrum 团队以及与之相关的角色、事件、工件和规则组成。框架中的每个部分都有其特定的目的，其对于 Scrum 的成功与使用是至关重要的。

Scrum 的规则把事件、角色和工件组织在一起，管理它们之间的关系和交互。对于 Scrum 的规则描述将会贯穿全文。

使用 Scrum 框架的其它不同特定技巧将不在本文中描述。

## Scrum 理论

Scrum 基于经验过程控制理论，或称之为经验主义。经验主义主张知识源自实际经验以及从当前已知情况下做出决定所获得。Scrum 采用一种迭代、增量式的方法来优化对未来的预测和管理风险。

透明、检视和适应是经验过程控制的三大支柱，支撑起每一个经验过程控制的实施。

### 透明

过程中的关键环节对于那些对产出负责的人必须是显而易见的。要拥有透明，就要为这些关键环节制定统一的标准，这样所有留意这些环节的人都会对观察到的事物有统一的理解。

例如：

- 所有参与者谈及过程时必须使用统一的术语。同时，
- 负责完成工作和验收工作的人必须对“完成”的定义，有一致的理解。

### 检视

Scrum 的使用者必须经常检视 Scrum 的工件和完成 Sprint 目标的进展，以便发现不必要的差异。检视不应该过于频繁而阻碍工作本身。当检视是由技能娴熟的检视者在工作中勤勉地执行时，效果最佳。

### 适应

如果检视者发现过程中的一个或多个方面偏离于可接受范围以外，并且将会导致产品不可接受时，就必须对过程或过程化的内容加以调整。调整工作必须尽快执行如此才能最小化进一步的偏离。

Scrum 规定了 4 个正式事件，用于检视与适应上，这 4 个事件在 Scrum 事件章节中会加以描述：

- Sprint 计划会议
- 每日 Scrum 站会
- Sprint 评审会议
- Sprint 回顾会议

## Scrum 价值观

当承诺、勇气、专注、开放和敬重五大价值观为 Scrum 团队所践行与内化时，Scrum 的透明、检视和适应三大支柱成为现实，并且在每个人之间构建信任。Scrum 团队成员通过 Scrum 事件、角色和工件来学习和探索这些价值观。

Scrum 的成功应用取决于人们变得更为精通践行五项价值观。人们致力于实现 Scrum 团队的目标。Scrum 团队成员有勇气去做正确的事并处理那些棘手的问题。每个人专注于 Sprint 和 Scrum 团队目标的工作。Scrum 团队及其利益攸关者同意将所有工作和执行工作的挑战进行公开。Scrum 团队成员相互敬重，彼此成为更有能力和独立的人。

## Scrum 团队

Scrum 团队由一名产品负责人、开发团队和一名 Scrum Master 组成。Scrum 团队是跨职能的自组织团队。自组织团队自己选择如何以最好的方式来完成工作，而不是由团队之外的人来指导。跨职能团队拥有完成工作所需的全部技能，不需要依赖团队之外的人。Scrum 的团队模式乃是设计用来提供最佳的灵活性、创造力和生产力。

Scrum 团队迭代增量式地交付产品，籍此最大化获得反馈的机会。增量式交付“完成”的产品保证了一个可工作产品的潜在可用版本总是存在。

## 产品负责人

产品负责人负责最大化产品和开发团队工作的价值。如何实现这一点的方式会随着组织、Scrum 团队和单个团队成员的不同而不同。

产品负责人是负责管理产品待办列表的唯一责任人。产品待办列表的管理包括：

- 清晰地表述产品待办列表项；
- 对产品待办列表项进行排序，使其最好地实现目标和使命；
- 优化开发团队所执行工作的价值；
- 确保产品待办列表对所有人是可见、透明和清晰的，同时显示 Scrum 团队下一步要做的工作；以及，
- 确保开发团队对产品待办列表项有足够深的了解。

产品负责人可以亲自完成上述工作，或者也可以让开发团队来完成。然而无论何者，产品负责人是负最终责任的人。

产品负责是一个人，而不是一个委员会。产品负责人可能会通过产品待办列表展现一个委员会的期望要求，但是想要改变产品待办列表项的优先级必须经过产品负责人。

为保证产品负责人的工作取得成功，组织中的所有人员都必须尊重他 / 她的决定。产品负责人对产品待办列表的内容和排序的决定必须是可见的。任何人都不得要求开发团队按照另一套需求开展工作，另一方面开发团队也不允许去做任何其他他人所说的。

## 开发团队

开发团队包含了各种专业人员，负责在每个 **Sprint** 结束时交付潜在可发布并且“完成”的产品增量。只有开发团队成员才能创建增量。

开发团队由组织组建并得到授权，团队自己组织和管理他们的工作。由此产生的正面效应能最大化开发团队的整体效率和效用。

开发团队具有下列特点：

- 他们是自组织的。没有人（即使是 **Scrum Master**）有权告诉开发团队应该如何把产品待办列表变成潜在可发布的功能增量；
- 开发团队是跨职能的，团队作为一个整体，拥有创建产品增量所需的全部技能；
- **Scrum** 不认可开发团队成员的头衔，不管承担哪种工作他们都叫开发人员，即只有开发人员这一头衔。此规则无一例外；
- **Scrum** 不认可开发团队中所谓的“子团队”，无论是否有特别的专业领域，例如无论是测试还是业务分析的成员都不能划分为“子团队”。此规则无一例外；同时，
- 开发团队中的每个成员也许有特长和专注的领域，但是责任属于整个开发团队。

## 开发团队的规模

开发团队最佳规模是足够小以保持敏捷性，同时足够大可以在 **Sprint** 内完成重要的工作。少于 3 个人的开发团队，成员之间没有足够的互动，因而生产力的增长不会很大。过小的团队在 **Sprint** 中可能会遭遇到技能上的约束，进而导致开发团队无法交付潜在可发布的产品增量。超过 9 人的团队则需要过多的协调沟通工作。过大的开发团队会产生太多的复杂性，不便于经验过程管理。产品负责人和 **Scrum Master** 角色不包含在此数字中，除非他们同时也参与执行 **Sprint** 待办列表中的工作。

## Scrum Master

**Scrum Master** 负责保证所有人都能正确地理解并实施 **Scrum**。**Scrum Master** 要确保 **Scrum** 团队遵循 **Scrum** 的理论、实践和规则。

**Scrum Master** 对 **Scrum** 团队而言，他 / 她是一位服务型领导。**Scrum Master** 帮助 **Scrum** 团队之外的人了解他 / 她们如何与 **Scrum** 团队交互是有益的，通过改变他 / 她们与 **Scrum** 团队的互动方式来最大化 **Scrum** 团队所创造的价值。



## Scrum Master 服务于产品负责人

Scrum Master 以各种方式服务于产品负责人，包括：

- 找到有效管理产品待办列表的技巧；
- 帮助 Scrum 团队理解为何需要清晰且简明的产品待办列表项；
- 理解在经验主义的环境中的产品规划；
- 确保产品负责人懂得如何来安排产品待办列表使其达到最大化价值；
- 理解并实践敏捷性； 以及,
- 按要求或需要引导 Scrum 事件。

## Scrum Master 服务于开发团队

Scrum Master 以各种方式服务于开发团队，包括：

- 在自组织和跨职能方面给予开发团队指导；
- 帮助开发团队创造高价值的产品；
- 移除开发团队工作进展中的障碍；
- 按要求或需要引导 Scrum 事件； 以及,
- 在 Scrum 还未完全采纳和理解的组织环境中指导开发团队。

## Scrum Master 服务于组织

Scrum Master 以各种方式服务于组织, 包括：

- 带领并指导组织采纳 Scrum ；
- 在组织范围内规划 Scrum 的实施；
- 帮助员工和利益攸关者理解并实施 Scrum 和经验产品开发；
- 引发能够提升 Scrum 团队生产效率的改变； 以及,
- 与其他 Scrum Master 一起工作，增加组织中 Scrum 应用的有效性。

## Scrum 事件

Scrum 使用固定的事件来产生规律性，以此来减少 Scrum 之外的其它会议的必要。所有事件都是有时间盒限定的事件，也就是说每一个事件限制在最长的时间范围内。一旦 Sprint 开始，它的持续时间是固定的，不能缩短或者延长。而其他事件则可以在该事件的目标达成之后可以立即终止，如此确保时间被适当地使用而不会造成过程中的浪费。

Sprint 除了本身作为一个事件以外，它还是其他所有事件的容器，在 Scrum 中的每个事件都是用来进行检视和适应的正式机会。这些事件都是被特别设计用来确保达成透明和检视。如果 Sprint 不能成功地包含这些事件中的任何一个事件，导致透明化的降低，同时也会丧失进行检视与适应的机会。

## Sprint

Sprint 是 Scrum 的核心，其长度（持续时间）为一个月或更短时间的限时，在这段时间内构建一个“完成的”、可用的和潜在可发布的产品增量。在整个开发过程期间，Sprint 的长度通常保持一致。前一个 Sprint 结束后，新的下一个 Sprint 紧接着立即开始。

Sprint 由 Sprint 计划会议、每日 Scrum 站会、开发工作、Sprint 评审会议和 Sprint 回顾会议构成。

在 Sprint 期间：

- 不能做出有害于 Sprint 目标的改变；
- 不能降低质量的目标； 以及,
- 随着对信息掌握的增加，产品负责人与开发团队之间对范围内要做的事可能会要澄清和重新协商。

每个 Sprint 都可以被视为一个项目，为期不超过一个月。就如同项目一样，Sprint 被用于完成某些事情。每个 Sprint 都会定义要开发什么，还有一份设计过和灵活的计划用来指导如何做这些事、工作内容和最终产品。

Sprint 的长度限制在一个月內。当 Sprint 的长度太长的话，对要构建什么的定义就有可能改变，复杂性也有可能增加，同时风险也有可能增加。Sprint 通过确保至少每月一次对达成目标的进度进行检视和适应，来实现可预测性。Sprint 同时也把风险限制在一个月成本上。

## 取消 Sprint

Sprint 可以在 Sprint 时间盒结束之前取消。只有产品负责人才有取消 Sprint 的权力，虽然他或她做这样的决定也可能受到来自利益攸关者、开发团队或是 Scrum Master 的影响。

如果 Sprint 目标过时，那么 Sprint 就会被取消。比如公司的发展方向或者市场上或技术上的状况发生改变，这些变化都可能导致 Sprint 被取消。总的来说，如果某个 Sprint 对其所在环境来说失去了价值和意义，那么它就应该被取消。然而，由于 Sprint 的时间都很短，所以取消 Sprint 的意义不大。

当取消某个 Sprint 时，任何做完和“完成”的产品待办列表项都需要评审。假如成果的任何部分具有潜在可发布的话，产品负责人通常会接受这个成果。所有未完成的产品待办列表项都会被放回到产品待办列表中，并重新估算。花在它们身上的工作会很快地贬值，所以必须经常性地重估。

取消 Sprint 会消耗资源，因为每个人都必须重新集合在另一个 Sprint 计划会议来开始另一个 Sprint。取消 Sprint 通常会对 Scrum 团队造成重创，这种情况非常罕见。

## Sprint 计划会议

Sprint 中要做的工作在 Sprint 计划会议中来做计划。这份工作计划是由整个 Scrum 团队共同协作完成的。

Sprint 计划会议是限时的，以一个月的 Sprint 来说最多 8 小时为上限。对于较短的 Sprint，会议时间通常会缩短。Scrum Master 要确保会议顺利举行，并且每个参会者都理解会议的目的。Scrum Master 要教导 Scrum 团队遵守时间盒的规则。

Sprint 计划会议回答以下问题：

- 接下来的 Sprint 交付的增量中要包含什么内容？
- 要如何完成交付增量所需的工作？

### 话题一：这次 Sprint 能做什么？

开发团队预测在这次 Sprint 中要开发的功能。产品负责人讲解 Sprint 的目标以及达成该目标所需完成的产品待办列表项。整个 Scrum 团队协同工作来理解 Sprint 的工作。

Sprint 会议的输入是产品待办列表、最新的产品增量、开发团队在这个 Sprint 中能力的预测以及开发团队的以往表现。开发团队自己决定选择产品待办列表项的数量。只有开发团队可以评估接下来的 Sprint 可以完成什么工作。

在开发团队预测完这个 **Sprint** 中可交付的产品待办列表项之后，**Scrum** 团队草拟一个 **Sprint** 目标。**Sprint** 目标是在这个 **Sprint** 通过实现产品待办列表要达到的目的，同时它也为开发团队提供指引，使得开发团队明确开发增量的目的。

## 话题二：如何完成所选的工作？

在设定了 **Sprint** 目标并选出这个 **Sprint** 要完成的产品待办列表项之后，开发团队将决定如何在 **Sprint** 中把这些功能构建成“完成”的产品增量。这个 **Sprint** 中所选出的产品待办列表项加上交付它们的计划称之为 **Sprint** 待办列表。

开发团队通常从设计整个系统开始，到如何将产品待办列表转换成可工作的产品增量所需要的工作。工作有不同的大小，或者不同的预估工作量。然而，在 **Sprint** 计划会议中，开发团队已经挑选出足够量的工作，以此来预估他们在即将到来的 **Sprint** 中能够完成。在 **Sprint** 计划会议的最后，开发团队规划出在 **Sprint** 最初几天内所要做的工作，通常以一天或更少为一个单位。开发团队自组织地领取 **Sprint** 待办产品列表中的工作，领取工作在 **Sprint** 计划会议和 **Sprint** 期间按需进行。

产品负责人能够帮助解释清楚所选定的产品待办列表项，并作出权衡。如果开发团队认为工作过多或过少，他们可以与产品负责人重新协商所选的产品待办列表项。开发团队也可以邀请其他人员参加会议，以获得技术或领域知识方面的建议。

在 **Sprint** 计划会议结束时，开发团队应该能够向产品负责人和 **Scrum Master** 解释他们将如何以自组织团队的形式完成 **Sprint** 目标并开发出预期的产品增量。

## Sprint 目标

**Sprint** 目标是在当前 **Sprint** 通过实现产品待办列表要达到的目的。它为开发团队提供指引，使得团队明确为什么要构建增量。**Sprint** 目标在 **Sprint** 计划会议中确定。**Sprint** 目标为开发团队在 **Sprint** 中所实现的功能留有一定的弹性。所选定的产品待办列表项会提供一个连贯一致的功能，也即是 **Sprint** 目标。**Sprint** 目标可以是任何其他的连贯性来促使开发团队一起工作而不是分开独自做。

开发团队必须在工作中时刻谨记 **Sprint** 目标。为了达成 **Sprint** 目标，需要实现相应的功能和实施所需的技术。如果所需工作和预期的不同，开发团队需要与产品负责人沟通协商 **Sprint** 待办列表的范围。

## 每日 Scrum 站会

每日 Scrum 站会是一个以 15 分钟为限的事件，它让开发团队同步开发活动，并为接下来的 24 小时制定计划。这需要检视上次每日站会以来的工作和预测下次每日站会之前所能完成的工作。

每日 Scrum 站会在同一时间同一地点举行，以便降低复杂性。在会议上，每一个开发团队成员都需要说明：

- 昨天，我为帮助开发团队达成 Sprint 目标做了什么？
- 今天，我为帮助开发团队达成 Sprint 目标准备做什么？
- 是否有任何障碍在阻碍我或开发团队达成 Sprint 目标？

开发团队借由每日 Scrum 站会来检视完成 Sprint 目标的进度，并检视完成 Sprint 待办列表的工作进度趋势。每日 Scrum 站会优化了开发团队达成 Sprint 目标的可能性。每天，开发团队应该知道如何以自组织团队来协同工作以达成 Sprint 目标，并在 Sprint 结束时开发出预期中的增量。开发团队或者开发团队成员通常会在每日 Scrum 站会后立即聚到一起进行更详细的讨论，或者为 Sprint 中剩余的工作进行调整或重新计划。

Scrum Master 确保开发团队每日站会如期举行，但开发团队自己负责召开会议。Scrum Master 教导开发团队将每日 Scrum 会议时间控制在 15 分钟内。

Scrum Master 强制执行每日 Scrum 站会规则——只有开发团队成员才能参加。

每日 Scrum 站会增进交流沟通、减少其他会议、发现开发过程中需要移除的障碍、突显并促进快速地做决策、提高开发团队的认知程度。这是一个进行检视与适应的关键会议。

## Sprint 评审会议

Sprint 评审会议在 Sprint 快结束时举行，用以检视所交付的产品增量并按需调整产品待办列表。在 Sprint 评审会议中，Scrum 团队和利益攸关者协同讨论在这次 Sprint 中所完成的工作。根据完成情况和 Sprint 期间产品待办列表的变化，所有参会人员协同讨论接下来可能要做的事情来优化价值。这是一个非正式会议，并不是一个进度汇报会议，演示增量的目的是为了获取反馈并促进合作。

对于长度为一个月的 Sprint 来说，评审会议的限时为 4 小时。对于较短的 Sprint 来说，会议的时间会有所缩短。Scrum Master 要确保会议举行，并且每个参会者都明白会议的目的。Scrum Master 教导大家遵守时间盒的规则。

Sprint 评审会议包含以下内容：

- 产品负责人邀请 Scrum 团队和主要的利益攸关者参加会议；
- 产品负责人说明哪些产品待办列表项已经“完成”和哪些没有“完成”；
- 开发团队讨论在 Sprint 期间哪些工作做的很好，遭遇到什么问题以及问题是如何解决的；
- 开发团队演示“完成”的工作并解答关于所交付增量的问题；
- 产品负责人讨论当前的产品待办列表的情况。他 / 她根据到目前为止的进度来预测可能的完成日期（如果有需要的话）；
- 参会的所有人就下一步的工作进行探讨，这样，Sprint 评审会议就能够为接下了的 Sprint 计划会议提供有价值的输入信息；
- 评审市场或潜在的产品使用方式所带来的接下来要做的最有价值的东西的改变；同时，
- 为下个预期产品版本的发布评审时间表、预算、潜力和市场。

Sprint 评审会议的结果是一份修订后的产品待办列表，阐明很可能进入下个 Sprint 的产品待办列表项。产品待办列表也有可能为了迎接新的机会而进行全局性地调整。

## Sprint 回顾会议

Sprint 回顾会议是 Scrum 团队检视自身并创建下一个 Sprint 改进计划的机会。

Sprint 回顾会议发生在 Sprint 评审会议结束之后，下个 Sprint 计划会议之前。对于长度为一个月的 Sprint 来说，会议的限时为 3 小时。对于较短的 Sprint 来说，会议时间通常会缩短。Scrum Master 要确保会议举行，并且每个参会者都明白会议的目的。Scrum Master 教导大家遵守时间盒的规则。Scrum Master 作为 Scrum 过程的责任者，作为团队的一员参加该会议。

Sprint 回顾会议的目的在于：

- 检视前一个 Sprint 中关于人、关系、过程和工具的情况如何；
- 找出并加以排序做得好的和潜在需要改进的主要方面；同时，
- 制定改进 Scrum 团队工作方式的计划。

©2016 Scrum.Org and ScrumInc. Offered for license under the Attribution Share-Alike license of Creative Commons, accessible at <http://creativecommons.org/licenses/by-sa/4.0/legalcode> and also described in summary form at <http://creativecommons.org/licenses/by-sa/4.0/>. By utilizing this Scrum Guide you acknowledge and agree that you have read and agree to be bound by the terms of the Attribution Share-Alike license of Creative Commons.



Scrum Master 鼓励 Scrum 团队在 Scrum 的过程框架内改进开发过程和实践，使得他们能在下个 Sprint 中更高效更愉快。在每个 Sprint 回顾会议中，Scrum 团队通过适当地调整“完成”的定义的方式来计划提高产品质量。

在 Sprint 回顾会议结束时，Scrum 团队应该明确接下来的 Sprint 中需要实施的改进。在下一个 Sprint 中实施这些改进是基于 Scrum 团队对自身的检视而做出的适当调整。虽然改进可以在任何时间执行，Sprint 回顾会议提供了一个专注于检视和适应的正式机会。

## Scrum 工件

Scrum 的工件以不同的方式表现工作任务和价值，可以用来提供透明以及检视和适应的机会。Scrum 所定义的工件是特别地设计的，是为了给关键信息提供最大透明化，因此每个人对工件都需要有相同的理解。

### 产品待办列表

产品待办列表是一份有序列表，其中包含产品需要的一切可能的东西，也是产品需求变动的唯一来源。产品负责人负责管理产品待办列表的内容、可用性和排序。

产品待办列表永远是不完整的。最早开发的产品待办列表只列举最初所知的以及理解最透彻的需求。产品待办列表会随着产品及其应用环境的改变而演进。产品待办列表是动态的，需要持续更新以反映出产品需要什么来保持其适用性、竞争力和有用。只要产品存在，产品待办列表也就同样存在。

产品待办列表列出所有的特性、功能、需求、增强和修复等对未来要发布的产品进行的改变。产品待办列表项具有这些属性：描述、次序、估算和价值。

随着产品的使用、价值的获取和获得市场的反馈，产品待办列表会成长为更大和更详尽的列表。因为需求永不停止改变，所以产品待办列表就如一份活的工件。业务需求、市场形势或者技术的变化都会引起产品待办列表的改变。

多个 Scrum 团队常常会一起参与对同一产品的开发。一个产品只有一个产品待办列表用于描述下一步产品开发工作。那么这就可能需要使用能够对产品待办列表项进行分组的属性。

产品待办列表精化指的是为产品待办列表项增添细节、估算和排序的动作。这是一个持续的过程，产品负责人和开发团队协同工作在产品待办列表项的细节上。在产品待办列表精化过程中，产品待办列表项被重新评审和修改。Scrum 团队决定如何来完成精化以及何时来完成。精化的工作通常占用开发团队不超过 10% 的产能。然而，产品负责人或者其他人在产品负责人的斟酌下，产品待办列表项可以在任何时间来更新。

©2016 Scrum.Org and ScrumInc. Offered for license under the Attribution Share-Alike license of Creative Commons, accessible at <http://creativecommons.org/licenses/by-sa/4.0/legalcode> and also described in summary form at <http://creativecommons.org/licenses/by-sa/4.0/>. By utilizing this Scrum Guide you acknowledge and agree that you have read and agree to be bound by the terms of the Attribution Share-Alike license of Creative Commons.

排序越高的产品待办列表项通常比排序低的更清晰同时包含更多细节。根据更清晰的内容和更详尽的细节信息就能做出更为准确的估算；同样，排序越低，则细节信息越少。产品待办列表项中那些即将会占用开发团队下一个 Sprint 大部分时间的项会被加以精化，因此，任一产品待办列表项都能够在 Sprint 的时间盒期限内适当地“完成”。这些能够被开发团队在一个 Sprint 中“完成”的产品待办列表项称为“准备就绪”，它们将作为 Sprint 计划会议中的待选产品列表项。产品待办列表项的足够透明程度通常要经过上述的精化活动来获得。

开发团队负责所有估算工作。产品负责人可以通过帮助开发团队更好地理解需求，并根据情况权衡取舍来影响他们，但是最终估算是由开发团队决定的。

### 监控目标实现的进度

在任何时刻，达成目标的剩余工作是可以累计的。产品负责人至少在每个 Sprint 评审会议中都必须跟踪剩余工作总量。产品负责人比较这次的剩余工作量与之前 Sprint 评审会议时的剩余工作量，来评估在期望的时间点达成目标的进度。这个信息对所有的利益攸关者都是透明的。

各种不同趋势走向的实践已经被使用在预测进度方面，例如，燃尽图（burn-downs）、燃烧图（burn-ups）或者累积流图（cumulative flows）。这些工具都被证实是有用的。然而，它们并不能用来取代经验主义的重要性。在复杂的环境中，未来将要发生的事是无法预知的。只有已经发生的事才能用来做前瞻性的决策。

### Sprint 待办列表

Sprint 待办列表是一组为当前 Sprint 选出的产品待办列表项，同时加上交付产品增量和实现 Sprint 目标的计划。Sprint 待办列表是开发团队对于下一个产品增量所需的那些功能以及交付那些功能到“完成”的增量中所需工作的预测。

Sprint 产品待办列表将开发团队用来达成 Sprint 目标的所有工作变得清晰可见。

Sprint 产品待办列表是拥有足够细节的计划，任何进度的变化可以在每日 Scrum 站会中清晰地看到。开发团队在 Sprint 期间修改 Sprint 待办列表，使得 Sprint 待办列表在 Sprint 期间涌现。涌现发生在开发团队按计划开展工作并学习到更多的关于哪些工作是达成 Sprint 目标所必需的工作时。

当新工作出现时，开发团队需要将其加入到 Sprint 待办列表中去。随着工作的执行或完成，剩余的工作量被估算并更新。当计划中的某个部分失去开发意义，就可以将其移除。在 Sprint 期间，只有开发团队可以改变 Sprint 待办列表。Sprint 待办列表是高度可见的，是对开发团队计划在当前 Sprint 内工作完成情况的实时反映，该列表由开发团队全权负责。



## 监控 Sprint 进度

在 Sprint 的任何时间点都可以计算 Sprint 待办列表中所有剩余工作的总和。开发团队至少在每日 Scrum 站会时跟踪剩余的工作量，预测达成 Sprint 目标的可能性。通过在 Sprint 中不断跟踪剩余的工作量，开发团队可以管理自己的进度。

## 增量

增量是一个 Sprint 完成的所有产品待办列表项的总和，以及之前所有 Sprint 所产生的增量的价值总和。在 Sprint 的最后，新的增量必须是“完成”的，这意味着它必须可用并且达到了 Scrum 团队“完成”的定义的标准。无论产品负责人是否决定真正发布它，增量必须可用。

## 工件透明

Scrum 依赖于透明。优化价值和控制风险的决定都是基于所获知的工件状态。当工件的状态是完全透明时，这些做出的决定才有一个坚实的基础；当工件的状态是不完全透明时，这些做出的决定就会有瑕疵，而价值也可能因此遭受损失，同时风险也可能会因此而增加。

Scrum Master 必须和产品负责人、开发团队和其他相关人员一起合作，以确保所有工件都是完全透明的。有些实践就是为应对不完全透明的状态而生的，Scrum Master 必须帮助每个人，让他们能够在遇到不透明的情况下采取最合适的实践。Scrum Master 能够通过检视工件、嗅探模式、倾听周围的声音以及观察预期和实际结果的差异来发现不完全透明。

Scrum Master 的职责就是和 Scrum 团队以及组织一起合作增加工件的透明化。这一工作通常包括学习、说服和改变。透明化不会在一夜之间发生，但是这是一条必经之路。

## “完成”的定义

当产品待办列表项或增量被描述为“完成”时，每个人都必须理解“完成”意味着什么。虽然在不同 Scrum 团队之间会存在巨大的差别，但是每个团队成员必须对完成工作意味着什么有相同的理解以确保透明化。这就是 Scrum 团队的“完成”定义，用来评估产品增量是否完成。

这一定义也同时被用来指导开发团队了解在 Sprint 计划会议时能够选择多少产品待办列表项。每个 Sprint 的目标在于交付符合 Scrum 团队当前“完成”的定义的潜在可交付功能增量。

开发团队在每个 **Sprint** 都交付产品功能增量。这一增量是可用的，所以产品负责人可以选择立即发布它。如果“完成”的定义对增量来说是开发组织的惯例、标准或指南，那么所有 **Scrum** 团队都必须遵守它，以此为最低标准。如果增量“完成”的定义不是开发组织的惯例，那么 **Scrum** 团队中的开发团队就必须制定适合于产品的“完成”的定义。如果系统或产品发布由多个 **Scrum** 团队一起开发，那么所有 **Scrum** 团队中的开发团队必须一起参与制定“完成”的定义。

每个增量都添加至之前的所有增量上，并且经过彻底地测试，以此确保整合在一起的所有增量都能工作。

随着团队的成熟，“完成”的定义会扩大，包含更为严格的标准来保证更高的质量。任何产品或系统都应该对其上面开发的工作有“完成”的定义。

## 结束语

**Scrum** 是免费的，在本指南中提供。**Scrum** 的角色、工件、事件和规则是不可改变的。虽然只实施部分的 **Scrum** 是可能的，但这样就不是 **Scrum** 了。**Scrum** 只以整体的形式而存在，唯其如此才能作为其他技术、方法和实践的容器运作良好。

## 致谢

### 人们

在千千万万对 Scrum 作出贡献的人中，我们要特别感谢那些在其最初十年提供帮助的人们。首先，要提到 Jeff Sutherland 以及与他一道工作的 Jeff McKenna，还有 Ken Schwaber 及与其一道工作的 Mike Smith 与 Chris Martin。在随后的几年中，许多人都作出了贡献，没有他们的帮助，Scrum 不会被提炼至如今这般。

### 历史

Ken Schwaber 和 Jeff Sutherland 在 1995 年举行的 OOPSLA 大会上首次联合公开演讲 Scrum。那次演讲本质上记录了 ken 和 Jeff 在之前几年间使用 Scrum 时所学到的经验。

在软件开发的世界中，Scrum 的历史已经算是很长了。我们对首批尝试和提炼 Scrum 的公司：Individual, Inc.、Fidelity Investments 和 IDX（现在的 GE 医疗）表示致敬。

Scrum 指南记录 Jeff Sutherland 和 Ken Schwaber 开发和培育 Scrum 已经超过 20 多年了。其他的一些资源从模式、流程和见解方面为 Scrum 框架提供了补充，从而优化生产效率、价值、创造力及提升自豪感。

### 翻译

本中文指南（2016 版）是从 Jeff Sutherland 和 Ken Schwaber 联合撰写的《Scrum 指南》（2016 版）的英文原版翻译而来。

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

## 版本变更记录

2016 版与 2013 版之间的区别是增加了“Scrum 价值观”一节。其内容如下：

当承诺、勇气、专注、开放和敬重五大价值观为 Scrum 团队所践行与内化时，Scrum 的透明、检视和适应三大支柱成为现实，并且在每个人之间构建信任。Scrum 团队成员通过 Scrum 事件、角色和工件来学习和探索这些价值观。

Scrum 的成功应用取决于人们变得更为精通践行五项价值观。人们致力于实现 Scrum 团队的目标。Scrum 团队成员有勇气去做正确的事并处理那些棘手的问题。每个人专注于 Sprint 和 Scrum 团队目标的工作。Scrum 团队及其利益攸关者同意将所有工作和执行工作的挑战进行公开。Scrum 团队成员相互敬重，彼此成为更有能力和独立的人。