

# Scrum 指南

---

Scrum 的权威指南：  
游戏规则



*Jeff Sutherland*



*Ken Schwaber*

2013 年 7 月

*由 Ken Schwaber 和 Jeff Sutherland 开发并维护*

---

# 目录

---

Scrum 指南的目的.....	3
Scrum 的定义.....	3
Scrum 理论.....	3
透明性.....	3
检视.....	4
调整.....	4
Scrum 团队.....	4
产品负责人.....	4
开发团队.....	5
Scrum Master.....	5
Scrum 事件.....	6
Sprint.....	7
Sprint 计划会议.....	8
每日 Scrum 站会.....	9
Sprint 评审会议.....	9
Sprint 回顾会议.....	10
Scrum 工件.....	11
产品待办列表.....	11
Sprint 待办列表.....	12
增量.....	12
工件的透明性.....	12
“完成”的定义.....	13
结束语.....	13
致谢.....	13
人们.....	14
历史.....	14
翻译.....	14

## Scrum 指南的目的

Scrum 是用于开发和支持复杂产品的框架。这份指南包含了 Scrum 的定义，其中包括 Scrum 的角色、事件、工件，以及把它们组织到一起的规则。Ken Schwaber 和 Jeff Sutherland 创造了 Scrum，Scrum 指南也由他们撰写提供。他们是 Scrum 指南的后盾。

## Scrum 的定义

Scrum: Scrum 是一个框架，在这个框架中人们可以解决复杂的自适应问题，同时也能高效并有创造性地交付尽可能高价值的产品。

Scrum 是：

- 轻量级的
- 容易理解的
- 难以精通的

自上世纪 90 年代初期以来，Scrum 就已经应用于管理复杂产品的开发。Scrum 不是开发产品的一种流程或一项技术，而是一个框架，在这个框架里可以应用各种流程和技术。Scrum 能使产品管理和开发实践的相对功效（relative efficacy）显现出来，以便进行改进。

Scrum 框架由 Scrum 团队及其相关的角色、事件、工件和规则组成。框架中的每个模块都有其特定的目的，对 Scrum 的成功实施和运用都至关重要。

Scrum 的规则把事件、角色和工件组织在一起，管理着它们之间的关系和交互。Scrum 的规则会贯穿这份文档。

实施 Scrum 的方案根据情况不同而不同，在这里不作介绍。

## Scrum 理论

Scrum 基于经验型流程控制理论，或者称为经验主义。经验主义主张知识源于经验，而决策基于已知的事物。Scrum 采用迭代增量式的方法来优化可预测性和管理风险。

透明性、检视、调整是经验型流程的三大支柱，支撑起每个经验型控制流程的实施。

## 透明性

流程中的关键环节必须为那些对产出负责的人可见。要拥有透明性，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情有统一的理解。

例如：

*©2014 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.*

- 所有参与者谈及流程的时候都必须使用统一的术语
- 负责完成工作和验收工作的人必须对“完成”有一致的定义。

## 检视

Scrum 的使用者必须经常检视 Scrum 的工件和完成 Sprint 目标的进度，以发现不必要的偏差。检视不应该过于频繁而阻碍了工作本身。当熟练的检视者认真履行检视工作时，效果最佳。

## 调整

如果检视者发现流程中的一个或多个方面背离了可接受的标准，并且将会导致产品不合格时，就必须对流程本身或者流程化的内容进行调整。调整工作必须尽快实施以最小化进一步的偏差。

Scrum 指定了进行检视和调整的 4 个正式事件，将在“Scrum 事件”一节中详细描述：

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

## Scrum 团队

Scrum 团队由产品负责人、开发团队和 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最好地完成工作，而不是由团队外的人指导。跨职能团队拥有完成工作所需要的全部技能，不需要依赖团队以外的人。这种团队模式的目的是最大限度地优化灵活性、创造力和生产效率。

Scrum 团队迭代增量式地交付产品，最大化获得反馈的机会。增量式地交付“完成”的产品保证了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以及开发团队工作的价值。实现这一点的方式会随着组织、Scrum 团队以及单个团队成员的不同而不同。

产品负责人是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：

- 清晰地表达产品待办列表项
- 对产品待办列表项进行排序，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可见、透明、清晰，并且显示 Scrum 团队的下一步工作
- 确保开发团队对产品待办列表项有足够的理解

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

产品负责人是一个人，而不是一个委员会。产品负责人可能会通过产品待办事项列表展现一个委员会的需求，但要想改变某项的优先级必须先经过产品负责人。

为保证产品负责人的工作顺利进行，组织中的所有人员都必须尊重他的决定。产品负责人所作的决定通过产品待办列表的内容和排序来表达。任何人都不得要求开发团队按照另一套需求开展工作，开发团队也不允许听从任何其他人的指令。

## 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。只有开发团队的成员才能开发增量。

开发团队由组织组建并授权，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有效性。

开发团队有以下几个特点：

- 他们是自组织的，没有人（即使是 Scrum Master 都不可以）告诉开发团队如何把产品待办事项列表变成潜在可发布的功能。
- 开发团队是跨职能的，团队作为一个整体，拥有开发产品增量所需要的全部技能。
- Scrum 不认可开发团队成员的头衔，无论承担哪种工作他们都叫做开发人员。此规则无一例外。
- Scrum 不认可开发团队中的所谓“子团队”，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。
- 开发团队中的每个成员可以有特长和专注领域，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，足够大以完成重要的工作。少于 3 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 中可能会受到技能的约束，无法交付可发布的产品增量。大于 9 人的团队需要过多的沟通协调工作。过大的团队会产生太多复杂性，不便于经验过程管理。产品负责人和 Scrum Master 的角色不包含在此数字中，除非他们也参与执行 Sprint 代表事项列表中的工作。

## Scrum Master

Scrum Master 负责确保所有人都能正确地理解并实施 Scrum。因此，Scrum Master 要确保 Scrum 团队遵循 Scrum 的理论、实践和规则。

Scrum Master 是 Scrum 团队中的服务型领导。Scrum Master 帮助 Scrum 团队外的人员了解他们如何与 Scrum 团队交互是有益的，通过改变他们与 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“清晰准确的产品待办列表项”的重要性
- 在经验主义的环境中理解长期的产品规划
- 确保产品负责人懂得如何安排产品待办列表项来最大化价值
- 理解并实践敏捷
- 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团队指导
- 协助开发团队开发高价值的产品
- 移除开发团队工作中的障碍
- 按要求或需要引导 Scrum 事件
- 在 Scrum 还未被完全采纳和理解的组织环境下指导开发团队

### Scrum Master 服务于组织

Scrum Master 以各种方式服务于组织，包括：

- 带领并指导组织采用 Scrum
- 在组织范围内计划 Scrum 的实施
- 帮助员工及相关干系人理解并实施 Scrum 和经验型产品开发
- 发起能够提升 Scrum 团队生产效率的改变
- 与其他 Scrum Master 一起工作，增加组织中 Scrum 实施的有效性

## Scrum 事件

Scrum 中指定了一些常规性事件，以减少 Scrum 之外的会议。Scrum 中的事件是有时间盒限定的，也就是说每个事件都有时间限制的。一旦 Sprint 开始，它的周期也就固定下来了，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以后立即终止，这样就确保了在这些事件上花费的时间不会影响项目的进度。

Sprint 除了本身作为一个事件以外，还是其他所有事件的容器。Scrum 中的每个事件都是进行检视和调整的机会。这些事件被特别用来确保至关重要的透明性和检视。如果 Sprint 不能成功地包含这些事件中的任何一个，透明性就会降低，同时也丧失了进行检视和调整的机会。

## Sprint

Sprint 是 Scrum 的核心，其周期为小于或者等于一个月，其产出是“完成的”、可用的、潜在可发布的产品增量。Sprint 的长度在整个开发过程中保持一致。新的 Sprint 在上一个 Sprint 完成之后立即开始。

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 中：

- 不能做出有害于 Sprint 目标的改变
- 不能降低产品质量
- 随着对信息掌握的增加，产品负责人和开发团队可以澄清或者重新商讨开发范围

每个 Sprint 都可以被视为一个项目，为期不超过一个月。和普通项目一样，Sprint 的目标也是完成一些事情。每个 Sprint 都会定义要开发什么东西，还有一份设计和灵活的计划能够指导开发过程、工作内容和最终结果。

Sprint 的周期被限制在一个月內。如果 Sprint 周期过长，对“要构建什么东西”的定义就有可能改变，复杂度和风险也有可能增加。Sprint 通过确保至少每月一次对达成目标的进度进行检视和调整，来实现可预见性。Sprint 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，但他做这样的决定也可能是受到相关干系人、团队或是 Scrum Master 的影响。

如果某个 Sprint 的目标过时了，那么也许就需要取消该 Sprint。比如公司的发展方向，或是市场、技术条件等发生了改变，这些都可能导致 Sprint 被取消。总的来说，如果某个 Sprint 对于其所处环境来说失去了价值和意义，那么它就应该被取消。然而，因为 Sprint 周期都较短，所以一般来说取消 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。假如有些已经潜在可交付，那产品负责人就会采纳它。所有未完成的就要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迅速贬值，所以需要频繁地重估。

取消 Sprint 会消耗资源，因为每个人都需要参加额外的 Sprint 计划会议来启动新的 Sprint。取消 Sprint 通常会对团队造成重创，这种情况非常罕见。

## Sprint 计划会议

Sprint 计划会议的目的就是要为这个 Sprint 的工作做计划。这份计划是由整个 Scrum 团队共同协作完成的。

对于周期为一个月的 Sprint，计划会议的时限为 8 小时。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，并且每个参与者都明白会议的目的，同时还要教导大家遵守时间盒的规则。

Sprint 计划会议要解决以下两个问题：

- 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：接下来的 Sprint 交付的增量中要包含什么内容？

开发团队预计这个 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要完成的产品待办列表项。整个 Scrum 团队为了更好地了解 Sprint 的工作进行讨论。

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 中预计可接受的工作量和以往的表现。开发团队自己决定选择待办列表项的数量。只有开发团队本身可以评估接下来的 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后，Scrum 团队会制定一个 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，它也为开发团队提供指引，使团队明确开发增量的目的。

### 话题二：要如何完成交付增量所需的工作？

设定了 Sprint 目标并挑选出这个 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以及交付它们的计划统称为 Sprint 待办列表。

开发团队通常先由系统设计开始，设计把产品待办列表转换成可工作的产品增量所需要的工作。工作的大小或预估的工作量可能会不同。然而，在 Sprint 计划会议中，开发团队已经挑选出足够的工作量，并且预计他们在即将到来的 Sprint 中能够完成。开发团队所计划的 Sprint 最初几天的工作在会议结束前分解为工作量等于或少于一天的任务。开发团队自组织地领取 Sprint 待办列表中的工作，领取工作在 Sprint 计划会议和 Sprint 期间按实际情况进行。

产品负责人对选定的产品待办列表项作出澄清，并协助团队权衡取舍。如果团队认为工作量过大或太小，就可以和产品负责人重新协商选中的产品待办列表项。开发团队也可以邀请其他人员参加会议，以获得技术或者领域知识方面的建议。



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 目标并开发期望的产品增量。

## 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，使团队明确构建增量的目的，其在 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。Sprint 目标可以是为唯一功能服务的产品待办列表项的集合，也可以是能够促使开发团队向着同一目标前进的其他工作，而不应该是孤立的工作。

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

## 每日 Scrum 站会

每日 Scrum 站会是以 15 分钟为限的事件，开发团队成员在这里分享各自的工作情况，并为接下来的 24 小时制定计划。这需要检视上个每日站会以来的工作和预测下个每日站会之前所能完成的工作。每日站会在同一时间同一地点进行来降低复杂度。会议上，每个开发团队成员都需要说明：

- 昨天我为开发团队达成 Sprint 目标做了什么
- 今天我准备如何帮助团队达成 Sprint 目标
- 有什么事情阻碍了我帮助团队达成 Sprint 目标

开发团队用每日站会来评估完成 Sprint 目标的进度，并评估完成 Sprint 待办列表的进度趋势。每日站会优化开发团队达成 Sprint 目标的可能性。每天，开发团队应该知道如何以自组织的形式协同工作以达成 Sprint 的目标，并在 Sprint 结束时开发出预期中的增量。整个开发团队或者部分团队成员经常在每日站会后聚集到一起进行更详细的讨论，或者为 Sprint 中剩余的工作做调整或重新计划。

Scrum Master 确保开发团队每日站会如期举行，开发团队自己则负责召开会议。Scrum Master 指导团队把会议控制在 15 分钟内。

Scrum Master 强制执行每日站会的规则——只有开发团队的成员才能参加。

每日站会可以增强交流沟通、省略其他会议、确定开发过程中需要移除的障碍、强调和提倡快速决策、提高每个成员对项目的认知程度。这是进行检视和调整的关键会议。

## Sprint 评审会议

Sprint 评审会议在 Sprint 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事项列表。在 Sprint 评审会议中，Scrum 团队和相关干系人讨论 Sprint 中完成的工作。然后，根据完成情况和 Sprint 期间产品待办列表的变化，与参会人员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，而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。

对于周期为一个月的 Sprint，评审会议的限时为 4 小时。对于时间少于一个月的 Sprint 来说，会议的长度会有所缩短。Scrum Master 要确保会议顺利举行，并且每个参与者都明白会议的目的，同时还要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 团队以及相关干系人参加会议
- 产品负责人说明哪些工作“完成”了，哪些工作没有“完成”
- 开发团队讨论在 Sprint 中哪些工作进展顺利、遇到了什么问题、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
- 产品负责人描述当前产品待办列表的完成情况，并根据进度推测可能的完成日期（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 评审会议就能为接下来的 Sprint 计划会议提供有价值的信息。
- 评审市场或者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
- 为下个产品版本的发布评审时间表、预算、潜在功能和市场

Sprint 评审会议的结果是一份修订的产品待办列表，确定很可能进入下个 Sprint 的产品待办列表项。也有可能为了迎接新机遇而全局调整产品待办列表。

## Sprint 回顾会议

Sprint 回顾会议是 Scrum 团队检视自身并创建下个 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 的计划会议之前。对于长度为一个月的 Sprint，会议限时为 3 小时。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，并且每个参与者都明白会议的目的，同时还要教导大家遵守时间盒的规则。Scrum Master 作为 Scrum 流程的监督者，也需要作为团队的一员参加该会议。

Sprint 回顾会议的目的是：

- 对前一个 Sprint 周期中的人、关系、过程和工具进行检视
- 找出做得好的和潜在需要改进的主要方面，并进行排序
- 制定改进 Scrum 团队工作方式的计划

Scrum Master 鼓励团队在 Scrum 的流程框架内改进开发流程和实践，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 团队对自己的检视而做出的调整。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调整的正式机会。

## Scrum 工件

Scrum 的工件以不同的方式表现工作任务和价值，可以用来提供透明性以及检视和调整的机会。Scrum 中的工件就是为了最大化关键信息的透明性，因此每个人都需要有相同的理解。

### 产品待办列表

产品待办列表是一个有序的列表，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，最早的列表只列举出最初所知的以及理解最透彻的需求。产品待办列表根据产品及其应用环境的改变而演进。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合理、有竞争力以及有用。只要产品存在，产品待办列表就存在。

产品待办列表列出了所有的特性、功能、需求、改进和修复等对未来要发布的产品进行的改变。产品待办列表项包含描述、次序、估算和价值。

随着产品的使用、价值的获取以及获得市场的反馈，产品待办列表变成了更大、更详尽的列表。因为需求永远不会停止改变，所以产品待办列表是个不断更新的工件。业务需求、市场形势和技术的变化都会引起产品待办列表的改变。

多个 Scrum 团队常常会一起开发某个产品。但描述下一步产品开发工作的产品待办列表只能有一个。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。

“产品待办列表细化”指的是为列表项补充细节、估算和排序。这是一个持续不断的过程，产品负责人和开发团队协作讨论产品待办列表项的细节，并对列表项进行评审和修改。何时如何进行细化的工作是 Scrum 团队的决定。细化的工作通常占用开发团队不超过 10% 的时间。然而，产品负责人可以根据自己的判断随时更新产品待办列表。

排序越高的产品待办列表项通常比排序低的更清晰、更具体。根据更清晰的内容和更详尽的信息就能做出更准确的估算；排序越低，细节信息越少。开发团队在接下来的 Sprint 中将要进行开发的产品待办列表项是细化过的，因此，任一列表项都能够在 Sprint 的时限内“完成”。我们把这些能够在 Sprint 中“完成”的列表项称为“准备就绪的”（Ready），它们将作为 Sprint 计划会议中的待选列表项。这种细化列表项的工作能为产品待办列表项提供足够的透明性。

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

## 监控实现目标的进度

在任何时间，达成目标的剩余工作量是可以累计的。产品负责人至少要在每个 Sprint 评审会议的时候追踪剩余工作总量。产品负责人比较这个数量与之前 Sprint 评审时的剩余工作量，来评估在希望的时间点达成目标的进度。这个信息对所有的相关干系人都透明。

各种趋势燃尽图（burn-downs）、燃烧图（burn-ups）和累积型的工作流（cumulative flows）都能用来预测进度。实践证明这些工具都是有用的。然而，这并不能代替经验主义的重要性。在复杂的环境下，将要发生的事情是未知的，只有已经发生的事情才能用来做前瞻式的决策。

## 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，外加交付产品增量和实现 Sprint 目标的计划。Sprint 待办事项列表是开发团队对于哪些功能要包含在下一个增量中，以及交付那些功能到“完成”的增量中所需工作的预测。

开发团队选中用于达成 Sprint 目标的工作正因为 Sprint 待办列表变得清晰可见。

Sprint 待办列表拥有足够的细节，因此能够在每日站会中对进度的变化有清楚的认识。开发团队会在 Sprint 期间修改 Sprint 待办列表，Sprint 待办列表也会随着团队工作的开展以及对工作需求了解的深入涌现出来。

当新工作出现时，开发团队需要将其追加到 Sprint 待办列表中去。随着任务的进行或者完成，团队需要估算并更新剩余的工作量。如果计划中某个部分失去开发的意义，就可以将其删除。在 Sprint 期间只有开发团队可以修改 Sprint 待办列表。Sprint 待办列表是高度可见的，是对团队计划在当前 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 中的任意时间点都可以计算 Sprint 待办列表中所有剩余工作的总和。开发团队至少在每日站会时跟踪剩余的工作量，预测达成 Sprint 目标的可能性。团队通过在 Sprint 中不断跟踪剩余的工作量来管理自己的进度。

## 增量

增量是一个 Sprint 完成的所有产品待办列表项，以及之前所有 Sprint 所产生的增量价值的总和。在 Sprint 的结尾，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。

## 工件的透明性

*©2014 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 依赖于透明性。我们作出的优化价值和控制风险的决策都是基于所获知的工件状态。如果工件的状态是完全透明的，那么作出的决策就等于有了一个坚实的基础；否则，作出的决策就是有缺陷的，而价值也有可能因此遭受损失，风险也可能会因此而增加。

Scrum Master 必须和产品负责人、开发团队以及其他相关干系人一起合作，以确保所有工件都是完全透明的。有些实践就是为了应对不完全透明的状态而生的，Scrum Master 必须帮助每个人，让他们都能够在遇到不透明的情况下采取最合适的措施。Scrum Master 可以通过检视工件、嗅探（可能造成不透明的）模式、留心倾听以及观察预期和实际结果的差别来检查是否有不透明的表象。

Scrum Master 的职责就是要和 Scrum 团队以及企业一起增加工件的透明性。这项工作通常需要一个学习、说服和改变的过程。获得透明性无法一步登天，但这是一条必经之路。

## “完成”的定义

当产品待办列表项或者增量被描述为“完成”的时候，每个人都必须理解“完成”意味着什么。虽然这在不同的 Scrum 团队之间会有巨大的差别，但是团队成员必须对完成工作意味着什么有相同的理解，这样才能保证透明性。这就是 Scrum 团队的“完成”定义，用来评估产品增量是否完成。

这个定义也同时被用来指导开发团队了解在 Sprint 计划会议时能选择多少产品待办列表项。每个 Sprint 的目标都是交付符合 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。这个增量是可用的，所以产品负责人可以选择立即发布它。如果开发部门制定了“完成”的定义作为规范、标准或者指引，那么所有 Scrum 团队都必须遵守；如果“完成”的定义还没制定，那么 Scrum 团队中的开发团队就必须制定符合产品的“完成”的定义。如果系统或者产品由多个团队开发，那么所有 Scrum 团队中的开发团队必须一起参与制定。

每个增量都添加到之前的所有增量上，并经过充分测试，以此保证所有的增量都能工作。

随着 Scrum 团队的成熟，“完成”的定义会扩大，包含更严格的标准来保证更高的质量。任何产品和系统都应该对在其上面开发的工作有“完成”的定义。

## 结束语

Scrum 是免费的，在这份指南中提供。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 只以整体的形式存在，才能作为其他技术、方法论和实践的容器运作良好。

## 致谢

*©2014 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.*

## 人们

在千千万万 Scrum 的贡献者中，我们要特别感谢那些在其最初 10 年提供帮助的人们。首先，要提到 Jeff Sutherland 及与他工作的 Jeff McKenna 和 Ken Schwaber 及与他工作的 Mike Smith 和 Chris Martin。许多人在随后的几年中也都作了贡献，没有他们的帮助，Scrum 不会像今天这样精炼。

## 历史

Ken Schwaber 和 Jeff Sutherland 在 1995 年的 OOPSLA 大会中首次共同演说 Scrum。那次演讲本质上记录了 Ken 和 Jeff 在之前几年使用 Scrum 时所学到的经验。

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 Ken Schwaber 和 Jeff Sutherland 开发和培育 Scrum 已经超过 20 年了。其他的一些资源从模式、流程和见解方面为 Scrum 提供了补充，从而提高生产效率、价值、创造力以及提升自豪感。

## 翻译

这份中文版指南是由李麟德 (Derek Li) 和王军 (Jim Wang)，基于鲍央舟和孙媛翻译的 2011 版的 Scrum Guide，根据 2013 版的更新进行补充和修改。原版的 Scrum Guide 由 Ken Schwaber 和 Jeff Sutherland 撰写。