

# 敏捷读书之重回根本：《Scrum指南》100问

本文是对Scrum指南的解析，以展示这个简单而有力的软件开发管理方法。

本文推荐的读者有三大群体：

1. Scrum Master。Scrum Master是Scrum的首要用户，他/她需要掌握和运用Scrum，并教练PO和Team运用Scrum，以及影响与团队工作相关的人员支持Scrum的运用。Scrum Master的工作通常不被正确理解。本文对此正本清源。
2. 企业高管。企业高管需要理解Scrum的理念和章法，才能决定和支持在企业中运用Scrum。
3. 所有其他对Scrum感兴趣的人员，或潜在直接或非直接用户。

本文分七个部分：

第一部分：Scrum的概述，作者和历史

第二部分：Scrum综述

第三部分：Scrum角色

第四部分：Scrum仪式（也叫会议或事件）

第五部分：Scrum物件（也叫工件）

第六部分：Scrum的规则和可执行性

第七部分：对Scrum指南可能的歧义问题

## 第一部分：Scrum的概述，作者和历史

### 什么是Scrum

Scrum是开发和维护复杂产品的框架方法。Scrum的定义包含其角色，事件，物件，和规则四种要素。Scrum的功效通过这四种要素的正确运用获得。Scrum是一个框架，在其中可以容纳与之兼容的其他方法，比如XP。Scrum的用户可以在Scrum框架之内优化Scrum。Jeff Sutherland和Ken Schwaber是Scrum的发明者，Scrum指南的撰写者和持续支持的后盾。

## 什么是Scrum指南

Scrum指南是Scrum的官方知识体系SBOK (Scrum Body of Knowledge)。Scrum指南的官方网站为：<http://scrumguides.org/>，也可通过Scrum联盟访问。

Scrum创始人Jeff Sutherland简介：



Jeff做过11家不同软件公司的CTO/CEO，对业务流程和打造高绩效团队有深刻的理解。Jeff 1993年发明Scrum方法，1995年与Ken一道把Scrum正式化。Scrum现在被全球大量软件开发公司使用。Jeff已将Scrum扩展至财务，保健，高等教育和通信等行业。

Scrum创始人Ken Schwaber简介：



Ken在1990年代早期与Jeff一道发明Scrum。Ken是2001年敏捷宣言起草人之一。Ken创建了敏捷联盟和Scrum联盟。Ken是30年软件开发老兵，写过三本关于Scrum的书：《Scrum敏捷软件开发》，《Scrum敏捷项目管理》，和《企业与Scrum》。

## Scrum的历史

1990年代早期Jeff和Ken发明了Scrum。1995年Jeff和Ken在美国德州奥斯汀的OOPSLA大会演说Scrum并发表了名为《Scrum软件开发流程》的论文。

Scrum的名字来自1986年的开创性论文《新的新产品开发游戏》，其作者是知名管理思想家野中郁次郎和竹内宏高。他们使用Scrum这个橄榄球术语来强调团队的重要性，并用橄榄球这种团队运动的特征来比拟新产品开发游戏的成功状况。论文中的研究表明，

被赋予目标而不是任务的小的自组织团队在开发复杂的新产品时能取得突出的绩效。最优秀的团队呈现为，被给予方向，和被给予空间制定战术，以完成共同目标。团队自治，以达卓越。

软件开发领域的Scrum框架，实现了论文中描述的原理，以用于开发与维护复杂软件产品。

对于软件开发而言，经验主义的过程优于预定义的过程，这一点也得到过程控制研究专业教授的认可。

经验主义用于复杂的工作，在其中未知的状况大于已知。在高度可变和不确定性之下，预定义过程的价值不大。

最早试用和完善Scrum的公司，略。

2001年2月，Jeff和Ken等17人起草了敏捷软件开发宣言。之后，Ken创建了敏捷联盟，并成为首任主席。

2001年，在Kent Beck鼓励下，Ken与Mike Beedle合写了第一本Scrum书：《Scrum敏捷软件开发》。

2002年，ken与Mike Cohn, Esther Derby成立了Scrum联盟，开启了Scrum认证项目。

2006年，Jeff成立了Scrum.inc, 并继续教授认证Scrum课程。

2009年，Ken离开Scrum联盟，成立了Scrum.org, 并开启了另一Scrum认证系列。

2010年Jeff和Ken首次发布Scrum指南，2011，2013，和2016年都有版本更新。Jeff和Ken以此建立了全球认可的Scrum知识体系SBOK即《Scrum指南》。

Scrum自1995年公布以来，被世界上大量软件开发公司使用，是敏捷方法论大家庭里最广泛运用的方法。目前，关于Scrum的书已超过1000本。

Scrum已被成功的运用到其他领域：制造，市场，运营，和教育等。

## 第二部分：Scrum综述

Scrum的定义是什么？

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以简驭繁，通过一套简单的规则，以目标为导向，彰显人的价值，达到上面的效果。运用Scrum的成功与对Scrum的采纳深度成正比。而对Scrum的采纳深度依赖于CEO的认知和组织文化，以信任，责任，教练和赋能取代命令和控制。

## Scrum 的三个特点

- 轻量级的：从流程的角度来看，Scrum是轻流程，不需要仅为证明遵从流程而专门做出的物件。流程的存在是为了方便沟通，达成目标，而不是推卸责任。
- 易于理解的：Scrum中只包含屈指可数的几个角色，事件，物件和把它们贯穿起来的规则。Scrum指南是一份只有17页的免费的PDF文件。Scrum指南中的每一句话都很容易看明白。
- 难以精通的：传统角色和Scrum角色之间只隔着一层窗户纸，这个窗户纸就是敏捷宣言的四个价值观。有的人一下子就能捅破，有的人怎么都不能捅破。

敏捷宣言的四个价值观：

敏捷软件开发宣言 <http://agilemanifesto.org/>

我们一直在实践中探寻更好的软件开发方法，身体力行的同时也帮助他人。由此我们建立了如下价值观：

**个体和互动** 高于 流程和工具

**工作的软件** 高于 详尽的文档

**客户合作** 高于 合同谈判

**响应变化** 高于 遵循计划

也就是说，尽管右项有其价值，我们更重视左项的价值。

与Scrum匹配的文化，是开放，信任，授权，赋能。把人当作有理想，会思考，能行动的成年人。

在一个团队当中，Scrum Master要首先精通Scrum，然后教导PO和Team精通Scrum。

## Scrum是一个怎样的框架

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。

正确运用Scrum，可以把问题清楚的显现出来，团队可以据此解决问题，持续改善。

在这其中存在着双重PDCA或双环学习。分别是关于产品和工作方法的设想-执行-验证-调整。Scrum Master要使这个双环学习发生，而这个过程的要点是深刻理解Scrum的所有要素，并使之在实际中发生。

## Scrum的要素有哪些

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。

正确使用Scrum的要点是深刻了解每一要素的目的。

## Scrum 的理论

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。

人是经验主义的载体。人的学习能力是经验主义传播的载体。Scrum的理论是经验主义，而Scrum的要素也有利于经验主义发挥价值。

经验主义作用的发挥，仰赖其三大支柱。

## Scrum的三大支柱

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。

这三个支柱应成为所有人员的基础性共识。是Scrum的基础性规则。

## 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都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。

Scrum Master要与团队一起识别哪些物件需要透明，以及以何种方式透明。

## 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。

Scrum Master可与团队一道定制需要检视的内容和检视的方法，并使之可执行。

## 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

适应通常与检视配对使用，Scrum Master要与团队一起制定适应的策略。

## 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。

团队可制定属于自己的价值观，可以包含但不限定于Scrum的五个价值观。

Scrum Master要和团队一道，把价值观，支柱，事件，角色和工件分解细化成可执行的东西，这是Scrum成功的保证，也是Scrum Master自我发展的方向 and 价值的体现。

## 第三部分：Scrum角色

### Scrum 团队的组成

Scrum 团队由一名产品负责人、开发团队和一名 Scrum Master 组成。

Scrum团队周边也可能有一些其他角色，他们是Scrum的配角，应配合Scrum团队的工作模式。

实行Scrum的组织，应最小化不在Scrum团队中的人员和最大化Scrum团队的能力。

## 什么是自组织和跨职能

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 Master可在Scrum框架之内采用任何最佳实践打造自组织团队，并在自组织和跨职能方面获得组织支持。

## Scrum中的迭代与增量

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。

迭代是把大项目变成小项目，增量是确保每迭代都有用户可用的输出。

## 产品负责人的核心职责

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。

一方面是确保所有工作项都是有价值的，另一方便确保团队先做价值最大的工作。高质量的产品待办事项列表，附以合适的沟通，是最大化产品和团队工作价值的核心载体。

## 产品负责人管理产品待办列表的职责有哪些

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。

- 确保开发团队对产品待办列表项有足够深的了解。

Scrum Master需辅助产品负责人和团队让上述事项发生。

## 产品负责人的责任和权威保证

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。

## 开发团队的组成，核心责任，成立与基础工作方式

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。

## 开发团队的特点

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。

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。

Scrum Master可创建舞台，教练团队成员Scrum的工作方式，但不可指挥和监控团队。Scrum Master可引导团队发展成跨职能团队，并取得组织的支持。

理想团队呈现的状态：目标共享，协作互助，个人在团队中得到提升。



## 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的核心职责

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 认可信任与教练优于命令和控制，才能做好服务型领导。

Scrum Master 可通过培训，引导和教练让团队理解和运用 Scrum。

Scrum Master 是通过影响力来工作。但一个组织的 Scrum 实施也不排除权力因素。

## 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。
  - 按要求或需要引导 Scrum 事件。

一方面产品待办列表要有价值和清楚明白，另一方面要在迭代和互动的环境中做规划。

## 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。
- 在 Scrum 还未完全采纳和理解的组织环境中指导开发团队。

有三个要点：一是以目标和高价值的产品为中心，二是自组织和跨职能的工作，三是移除障碍。

## 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。
- 与其他 Scrum Master 一起工作，增加组织中 Scrum 应用的有效性。

这一部分通常由敏捷教练担当。而一个成熟的Scrum Master就已经是敏捷教练了。

## 第四部分：Scrum仪式（也叫会议或事件）

### 什么是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。

实践中Scrum会议的无效性通常是不能正确和系统化的理解Scrum每一事件的目的。

## Sprint本身是一个事件

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的构成

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。

## Sprint中的变与不变

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。

## Sprint是一个利于检视和适应，降低复杂度和风险的项目

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 计划会议的目的，时长和基本方式

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？

## Sprint 计划会议话题一：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 建议的 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 计划会议的输入

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。

## 团队在计划会上制作 Sprint 目标

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 团队一起制作一个 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。

## Sprint计划会话题二：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。

## 产品负责人在计划会上的互动

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。

## 计划会结束，做出承诺

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。

## 什么是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。

## Sprint目标的运用

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站会的时间地点及核心内容

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，并为接下里的 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。

## 每日站会上的三个问题

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？

## 每日站会与迭代目标

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。

## 每日站会的参与者

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。

## 每日站会的好处

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 评审会议的目的与时间

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 评审会议的内容

- 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。
- 为下个预期产品版本的发布评审时间表、预算、潜力和市场。

## 评审会议的输出

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 回顾会议的核心目的和时间

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

- 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。
- 制定改进 Scrum 团队工作方式的计划。

## 回顾会议的方式

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。

## 回顾会议的输出

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。

## 第五部分：Scrum 物件或工件

### Scrum 工件的目的

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。

### 什么是产品待办列表

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。

### 产品待办列表的属性

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。

### 产品待办列表的演变

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。

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。

## 产品待办列表的分组

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。

## 产品待办列表的精化

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。

## 产品待办列表的“准备就绪”

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。

## 谁负责估算

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 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。

## 迭代待办列表的所有者及使用

Sprint 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。

## 第六部分：Scrum的规则和可执行性

Scrum的规则散落在Scrum指南的各处。本节是对Scrum规则的不完全提炼。这里所说的规则，也并非狭义的规则，更多是指做事方式。一方面是让各方认识到Scrum Master有大量工作要做，另一方面让Scrum Master有着力点。

1. 目标。目标有不同层面，包括产品愿景，产品路线图，迭代目标，甚至迭代内的阶段性目标。要在不同时机让团队明了不同层面的目标，视完成团队目标为主要驱动

力。

2. 迭代目标。在迭代计划会上由团队一起制定目标。在每日站会上检视目标的达成进度。
3. 团队。在不同活动中以不同方式引导团队共同参与。例如在估算时使用估算纸牌。在每日站会结束时，随机一人总结迭代进展情况。
4. 问题解决和持续改善。探索问题解决和持续改善的管理方法，辅助和引导团队识别问题，解决问题，持续改善。
5. 沟通。打磨出适合的沟通渠道和沟通模式。比如说团队每天的时间可分出欢迎打扰时间和免打扰时间。
6. 应变。在Scrum框架内的应变。准备和选取迭代待办事项需尽量慎重。迭代中内容的增删以不违背迭代目标为前提。对可见的干扰预留缓冲。尽量采用短的迭代。
7. 透明。识别出透明的内容和透明的方式。
8. 检视。识别出检视的内容和检视的方式，做成在不同事件中的检查列表。
9. 适应。识别出适应的内容和适应的方式。
10. 五价值观。跟行为映射起来，作为检视的内容。比如说不迟到，对应尊重等价值观。
11. 自组织。培养团队集体做决定的方式等。
12. 跨职能。计划交叉学习和技能备份，鼓励合作。
13. 迭代与增量。检视每迭代是否有可工作的软件。
14. 产品待办列表。遵守清晰，有估算，有优先级等规则。
15. 承诺。练习在迭代计划会上做承诺的习惯。
16. 准备好的定义。与团队一起制定准备好的定义。
17. 完成的定义。与团队一起制定完成的定义。
18. 监控什么及如何监控。识别并检视适应。
19. 工作协定。按需建立工作协定并可视化。

## 第七部分：对Scrum指南可能的歧义问题

Scrum Master与Scrum的关系

Scrum Master由组织授权。授权者应理解Scrum, 对Scrum Master对PO和团队的教练, 及影响周边人员支持Scrum的工作方式等提供支持。

Scrum Master需精通Scrum, 确保团队正确应用Scrum, 并与管理层一道排除Scrum实施的障碍。

## 关于会议长度

Scrum指南中所建议的会议长度是一个时间盒, 代表最大长度。而如果实际中会议时间太短, 也可能是没有正确理解每个会议的目的。

## 关于每日站会中是否可以有团队之外的成员

不反对参观者, 但要事先让团队了解参观的目的。参观者不可以发言, 不可以破坏团队的安全感。

## 关于此规则无一例外

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。

特别强调这一点, 一是鼓励团队共享目标和责任, 二是鼓励沟通交流合作, 三是鼓励学习不同技能, 四是这样要求的目的, 是使工作安排上不会出现瓶颈。这一切都可以有计划的进行。

## 关于开发团队

有人提议将开发团队更名为交付团队。相对于交付团队, 有一个产品探索团队。

Scrum指南对这两者并不做清晰的界定。可以有一个虚拟的产品探索团队, 以产品负责人为中心, 调度相关人员包括开发团队。所以保留开发团队的名字可能更好。虚拟概念上可以有产品探索团队和交付团队。

## 关于是否把产品待办列表精化作为一个事件或会议

产品待办列表精化可以是一个会议, 也可以不以会议的形式进行。实践当中多是两者结合。这或者是Scrum指南没把它定为一个事件的原因。

## 关于每日站会三个问题是否必须回答

不一定要从形式上回答, 但要涵盖到三个问题的内涵。一是关于进度, 二是关于下一步计划, 三是关于障碍。并且要从团队的角度看。还有一个隐含的问题是对团队目标完成

的预测性。Scrum Master可与团队一道定制属于自己的每日站会检查列表。前提是以Scrum指南为指导。

关于没有人（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

长期目标是团队成为自组织和跨职能的团队，在此之前，团队依然可以求助。

关于在计划会上制定迭代目标的时机

产品负责人可以在会议开始时提议一个迭代目标，在会议中间，如果确定了待办事项，团队可以基于产品负责人提议的目标一起制定一个迭代目标，如果无法在中间确定要做的待办事项，可以在会议的最后再制定迭代目标。

一个迭代是否可以发布多次

可以。迭代计划可据此相应调整。Scrum指南中对此并无明确说明，或许是因为这种情况还没有成为常态。

关于准备好的定义

Scrum指南中有提到准备好的定义，但没有提高到完成的定义的高度。关于这一点，可能的争议是，准备好的定义有可能会变成一个流程和工具。或许等有更多的实践支持，Scrum指南才会把准备好的定义正式化。

关于不要把产品负责人变成一个代理

一个组织需要正确理解Scrum并确定正确的产品负责人角色，并尊重其权威。

关于估算

估算的作用，对于产品层面是预测大概什么时候做完，对于迭代层面是帮助作出承诺。在保证这两个目的的前提下，估算可以灵活的方式进行，比如数故事个数。

关于产品负责人如何最大化团队工作的价值

要有清晰的可被团队理解的产品愿景，产品路线图，故事地图，史诗，故事包括验收标准。价值最大的工作先做。

关于工作协定

Scrum指南中没有明确说明。完成的定义等也是一种工作协定。还可以有其他工作协定。

## 关于预测与承诺

预测是基于估算的判断，承诺是尽最大努力而为的行为。在一个透明和不指责的环境中，这两点的区别和联系会比较容易的呈现。

## 关于一对一的回顾会议

可以有。可以作为Scrum Master对团队成员教练和合作改善的一部分。

## 关于产品愿景

Scrum指南中并没有包含。从实践当中来说，产品愿景会起到很好的激励作用。产品负责人需要制定一个产品愿景。

## 关于迭代目标是否算Scrum物件

目前迭代目标并没有放在Scrum指南的物件章节下。Scrum的标准物件只有三个：产品待办列表，迭代待办列表，产品增量。其他如迭代目标也可视为物件。

## 产品待办列表对所有人透明的所有人是谁

团队及有兴趣的人。有兴趣的人的首选是通过产品待办列表的状态了解产品团队的进展，所有人都盯着同一个方向，而不是人盯人。

## 关于业务分析师在Scrum中的角色

被产品负责人和团队封装吸纳。

## 关于Scrum Master移除障碍

有些障碍具体是由团队参与移除，有些是Scrum Master为主协调，有些需要管理层支持。

## 关于每日站会是否增加一问：我们是否能完成迭代目标

可以增加。可以每天随机一人总结回答这个问题。

关于是否需要障碍列表

团队以适合自己的方式管理。

关于是否需要改善列表

团队以适合自己的方式管理。

关于回顾会是否可以谈个人

尽量避免，不绝对，取决于团队的心理接受程度。

关于每日站会是否需要一个可视化的定时器

有条件的话可以有。

关于是否需要限制正在进行的任务数量

Scrum Master可观察并引导团队讨论。

关于Scrum Master在回顾会议中的角色

Scrum指南中只讲了回顾会议的目的，时间和要在Scrum框架内改善，但没讲回顾会议进行的方式。Scrum Master可探索任何能达到会议目的及适合团队的方式。

关于Scrum规则

在Scrum指南中，Scrum规则散落在各处。本文的Scrum规则一节尝试有一些提炼。

关于质量

可通过迭代目标和完成的定义涵盖。

关于可持续开发的步伐

在敏捷宣言的原则中有。在Scrum指南中没有明确说明。敏捷宣言可视为Scrum的一部分或前提。在Scrum中需强调可持续的开发步伐，而不要被冲刺的字面含义误导。

Scrum Master是否可以作为开发人员



可以。

产品负责人是否可以作为开发人员

可以。不过产品负责人的工作通常比较饱满。

产品负责人和Scrum Master 可以是同一个人吗

不可以。两者之间存在着有益的良性冲突，一个提升责任，一个提升信任。

## 结语

Scrum总结起来是关于目的，人和流程。

关于目的：把产品的大概念按颗粒度分解为史诗和故事，在时间轴上分解为产品路线图，发布，迭代和每日工作，每个工作项在不同的时机按不同的颗粒度有清晰的表示，把目的灌输在其中，辅以适当的沟通，起到目的驱动的效果。

关于人：以信任和责任为基础原则。责任体现在目的的颗粒化，清晰化，迭代化。信任包括管理层对团队的信任，授权团队自组织的工作。还包括团队内部的互相信任，这样才能共享目标，协作互助，共同进步。相信团队有目标，会思考，善行动。焦点在团队，而不在个人。

关于流程或规则：流程和规则是为了辅助人，方便人，解放人。是为了方便目的的表达，方便人的合作。是为了帮助人达到高绩效，和帮助产品成功。这是流程和规则的出发点。

真正的卓越，根植于组织内成员的快乐之中。员工们的快乐带来的绩效优化，是Scrum模式产生威力的关键因素。