



# SCRUM

2010 年 2 月

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

## 致谢

### 概要

Scrum 基于业界认可的最佳实践，这些实践已在过去的几十年被使用并证实有效。之后，Scrum 被置于基于经验过程的理论中。正如 Jim Coplien 一次对 Jeff 所说：“每个人都会喜欢 Scrum；因为这是当我们被逼到墙角时的自然反应。”

### 人

在千千万万对 Scrum 做出贡献的人中，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 及与之工作的 Jeff McKenna, Ken Schwaber 和 Mike Smith 还有 Chris Martin。Scrum 在 1995 年的 OOPSLA 上首次被正式介绍和发布。在之后 5 年中，Mike Beedle 和 Martine Devos 做出了重大贡献。还有所有其他人，没有你们的帮助，Scrum 不会被提炼至今天的高度。

### 历史

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。

### 翻译

这份指南是从 Ken Schwaber 和 Jeff Sutherland 的英文原版翻译而来。感谢孙媛和鲍央舟为翻译做出贡献。

## 目标

自从上世纪90年代初期，Scrum方法就已经应用于开发复杂的产品。本篇文章介绍了如何应用Scrum构建产品。Scrum不是一种过程，也不是一项构建产品的技术，而是一个框架，在这个框架里可以应用各种过程和技术。Scrum的作用就是让开发实践方法的相对功效显现出来以便随时改进，**同时也为开发复杂产品提供了框架。**

## Scrum 理论

Scrum是以经验过程控制理论为依据，采用迭代、增量的方法来提高产品开发的可预见性并控制风险。Scrum的三大支柱支撑起每个经验过程控制的实现。

### 第一大支柱是高透明度

高透明度确保管理结果的人看得到那些影响结果的过程方面。这些过程方面不仅要透明，而且那些被观察到的方面也必须被充分了解。这就是说，当某人检验某个过程并认为完成了某些任务时，这个完成必须等同于他们的完成定义。

### 第二大支柱是检验

开发过程中的各方面必须做到经常性的检验，以确保及时发现过程中的重大偏差。在确定检验频率时，需要考虑到检验会引起所有过程发生变化。当规定的检验频率超出了过程检验所能允许的程度，那么就会出现問題。幸运的是，软件开发并不会出现这种情况。另一个因素就是检验工作成果人员的技能水平和勤勉程度。

### 第三大支柱是适应

如果检验员经检验发现过程中的一个或多个方面不满足可接受标准，并且最终产品是不合格的，那么检验员就必须对过程或是材料进行调整。调整工作必须尽快实施以减少进一步的偏差。

Scrum中有三个进行检验和适应的时刻：每日例会是用来检验朝向Sprint目标的工作进程，调整以优化次日的工作价值。另外，Sprint评审和计划会议是用来检验朝向发布目标的工作进程，调整以优化下一个Sprint的价值。最后，Sprint回顾会议是用来评审完成的Sprint，并确定什么样的调整可以使下一Sprint的效率更高、

结果更令人满意和更易于工作。

## Scrum 内容

Scrum框架包括一组**Scrum团队**和与其相关的事物：**时间盒**、**工件**和**规则**。

Scrum团队的目标是提高灵活性和生产能力。为此，他们自组织、跨职能，并且以迭代方式工作。每个Scrum团队都有三个角色：1) **ScrumMaster**，负责确保成员都能理解并遵循过程；2) **产品负责人**，负责最大化Scrum团队的工作价值；3) **团队**，负责具体工作。团队包括的开发人员具备开发所需的各种技能，负责在每个Sprint结束之前将产品负责人的需求转化成为潜在可发布的产品模块。

Scrum利用时间盒实现规律性。被时间盒限定的Scrum要素有：**发布计划会议**、**Sprint计划会议**、**Sprint**、**每日例会**、**Sprint评审会议**和**Sprint回顾会议**。Scrum的核心是**Sprint**，即贯穿于开发工作中保持不变的一个月（或更短时间）迭代。所有的Sprint都采用相同的Scrum框架，并且都交付潜在可发布的最终产品增量。Sprint的交替没有间隔期。

Scrum 采用四个主要的工件：**产品待办事项列表**是囊括了开发产品可能需要的所有事项的优先排列表。**Sprint 待办事项列表**包含了在一个Sprint内将产品待办事项列表转化成最终可交付产品增量的所有任务。燃尽图是用来衡量剩余的待办事项列表。**发布燃尽图**衡量在一个发布计划的时间段内剩余的产品待办事项列表。

**Sprint 燃尽图**衡量在一个Sprint时间段内剩余的**Sprint 待办事项列表**条目。

**规则**将Scrum的时间盒、角色和工件联系起来。对于Scrum规则的描述将贯穿整个文章。例如，Scrum规则规定：只有团队成员，即承诺将产品待办事项列表转换成产品增量的人员，才可以在每日例会上发言。在“提示”框中描述的实现方法并不是规则而是建议。

### 提示

当规则未明确时，Scrum的使用者们要自己想出如何去做。不要试图去寻找完美的解决方法，因为问题通常变化的很快。相反的，尝试一些做法并观察效果如何。Scrum经验本性中的检验和适应的特性会指导你。

## Scrum 角色

Scrum团队包括ScrumMaster、产品负责人和团队。Scrum团队成员被称为“猪”。产品负责人是产品待办事项列表的“猪”。团队是Sprint任务的“猪”。ScrumMaster是Scrum过程的“猪”。其他人被称为“鸡”。“鸡”没有权利要求“猪”如何去开展工作。“鸡”和“猪”的比喻来自于以下的故事：

*一只鸡对一头猪说：“我们合伙开家饭店吧！”*

*猪想了想，说：“那我们给这个饭店起什么名字呢？”*

*鸡说：“鸡蛋和火腿！”*

*猪回答到：“那还是算了吧，你要做的只是下几只鸡蛋，我把命都搭上了！”*

## ScrumMaster

ScrumMaster负责确保Scrum团队遵守Scrum价值、实践和规则；帮助Scrum团队和整个组织采用Scrum；通过指导和引导，教授Scrum团队更高效工作、生产出高质量的产品；帮助Scrum团队理解并采用自组织和跨职能。ScrumMaster也帮助Scrum团队在现有的组织环境中做到最好，即使环境并没有优化到适应复杂产品开发的程度。当ScrumMaster帮助进行改变时，这些活动被称为“去除障碍”。ScrumMaster的角色是Scrum团队的服务型领导者。

### 提示

*ScrumMaster 与客户和管理层共同确定和具体化产品负责人人选。ScrumMaster 负责教授产品负责人如何进行工作。产品负责人应当了解如何使用 Scrum 优化产品开发带来的价值。如果他们不能做到这点，那么我们认为 ScrumMaster 是负有责任的。*

### 提示

*ScrumMaster 可以是团队的成员，例如，他可以是承担 Sprint 任务的开发人员。但是，当他需要在排除障碍和执行任务之间选择权衡的时候，这种身兼二职的情况常常会对做决定带来矛盾。ScrumMaster 绝对不能是产品负责人。*

## 产品负责人

产品负责人是管理产品待办事项列表、确保团队工作价值的唯一责任人。他负责维护产品待办事项列表，确保每个成员明晰列表内容、明确哪些条目具有最高优先级，从而了解下个需要开发的条目。

产品负责人是一个人，而不是一个委员会。可能会有一些委员会向产品负责人提出建议或影响他的决策，但要想改变某条目的优先级必须先说服产品负责人。采用Scrum的企业可能发现这样会对他们制定优先级和需求的方法产生影响。

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的决定。任何人都不得要求团队按照另一套优先级开展工作，团队也不允许听从任何人带有威胁恐吓性的指令。产品负责人所作的决定需要明确的包括在产品待办事项列表的目录和优先级中。这就要求产品负责人竭尽全力，同时也使其成为一个费心费力但又值得去做的角色。

### 提示

*对于商业开发，产品负责人可能就是产品经理。对内部开发而言，产品负责人可能来自其业务会被该产品自动化的职能部门经理。*

### 提示

*产品负责人可以是团队成员，承担开发任务。但是这样可能会牵掣其精力，影响产品负责人和利益相关人协调工作。但是，产品负责人绝对不能是ScrumMaster。*

## 团队

开发人员组成的团队负责在每个Sprint将产品待办事项列表转化成为潜在可交付的功能增量。团队同时是跨职能的；团队成员必须具备创造产品增量所需要的技能。团队成员常常具备如编程、质量控制、业务分析、架构、用户界面设计或数据库设计等的专业技能。然而，团队成员所共同分享的技能即访问需求并将需求转化成可用产品的能力，往往比那些专业技能更重要。因为是架构师或设计人员而不愿意编码的人不适合成为团队成员。每个人都尽心尽力参与，即使需要学习新技能或巩固现有技能。团队中没有头衔的概念，这一规则无例外。团队也不允许包括测试或业务分析等在特定领域工作的子团队。

团队同时是自组织的。任何人，包括 ScrumMaster 都没有权利规定团队如何将产品待办事项列表转化成可交付的功能增量，而是由团队自己确定。每个团队成员利用自己的专业技能，解决遇到的问题。这种协同配合提高了团队整体的效率和效力。

团队理想规模是 7 个（加或减 2 个）人。如果成员少于 5 人，那么相互交流就减少了，团队的生产力也会下降。更重要的是，团队在 Sprint 中可能会受到技能限制，从而导致无法交付可发布的产品模块。如果成员多于 9 人，那么成员之间就需要太多的协调沟通工作。大型团队会产生太多复杂性，不便于经验过程控制。但是，我们也遇见过一些超过规模上或下限人数的团队取得成功。产品负责人和 ScrumMaster 这两个角色不计入成员人数，除非他们也是“猪”。

团队的构成在Sprint结束时可能会发生变化，每次团队成员的变化，都会降低通过自组织而获取的生产力。因此，改变团队构成时务必要谨慎。

## 时间盒

Scrum中的时间盒包括：**发布计划会议、Sprint、Sprint计划会议、Sprint评审会议、Sprint回顾会议和每日例会。**

## 发布计划会议

发布计划会议的目的是建立Scrum团队与组织内的其他部门能够理解和沟通的计划和目标。发布计划会议需要回答以下两个问题：“我们如何以最佳方式将愿景转化为成功的产品？我们怎样才能达到或甚至超越客户的满意度和投资回报？”发布计划确定发布目标、具有最高优先级的产品待办事项列表、重大风险和发布所包含的全部特性和功能。另外，发布计划会议确立大致交付日期和费用，如果没有任何变化就应当保持该日期和费用。因此，组织就可以检验开发进程，以每个Sprint为基础调整发布计划。

发布计划会议是非强制性的。如果Scrum团队没有召开该会议就开始工作，那么缺乏的工件就会变得非常明显并成为需要解决的障碍。解决障碍的工作将成为产品待办事项列表中的条目。

产品应用Scrum以迭代的方式构建，每个Sprint中都会创造出产品增量，通常先开发价值最高和风险最大的产品。产品增量随着Sprint交替而完成，每个增量都是整个产品的可交付部分。当创造出足够多有价值、对投资人有益的增量时，产品就可以发布了。

大多数组织已经有自己的发布计划过程，并且在大多数过程中的大部分计划已经在发布之初完成，而且也不会更改。在Scrum发布计划会议上确立一个整体目标和预期结果。发布计划会议通常不超过组织构建传统发布计划的15-20%时间。然而，Scrum发布在每个Sprint评审和计划会议上都执行及时计划，同时，在每日例会上执行每日及时计划。总的看来，Scrum发布可能比传统的发布计划耗费略多的工作。

发布计划会议需要为发布估算并优先排列产品待办事项列表。完成这项工作的技巧很多，但都不属于Scrum的范围，虽说如此，这些技术仍有其使用价值。



## Sprint

一个Sprint就是一个迭代，Sprint是以时间盒限定的。在Sprint中，ScrumMaster负责确保不出现任何影响Sprint目标的变化。团队构成和质量目标在Sprint中均保持不变。Sprint包括，或者说由Sprint计划会议、开发工作、Sprint评审会议和Sprint回顾会议组成。Sprint一个紧跟一个进行，之间没有任何时间间隔。

### 提示

*如果团队发现承诺的任务过多，可以和产品负责人沟通，以减小Sprint中选定的产品待办事项列表范围。如果团队发现有富余的时间，可以和产品负责人商议追加额外的产品待办事项列表。*

项目旨在完成某个目标，对于软件开发而言，其目的在于构建一个产品或系统。每个项目都包括构建目标的定义、构建计划、工作完成进度和最终产品。每个项目都应该制定范围，即计划精准的时间范围。如果时间过长，某些定义可能会发生变化，另外也会牵扯进很多变量，因而引发巨大的风险等等。Scrum框架适合开发周期不超过一个月的项目，项目过于复杂势必会使开发周期延长，从而增加风险。至少每个月应该针对项目的可预见性进行控制，这样每个月都可以遏制项目变得不可控制或不可预测而产生的风险。

### 提示

*当团队刚刚开始使用Scrum，为期两周的Sprint可以让团队在学习进步，避免在不确定中挣扎。两周时间的Sprint可以通过将两个增量添加在一起，从而和其他团队同步。*

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利益相关人、团队或是ScrumMaster的影响。那么，在什么情况下必须取消Sprint呢？如果某个Sprint的目标过时了，那么管理层就需要取消该Sprint。比如公司的发展方向，或是市场、技术等情况发生变化，这些变化都可能导致取消Sprint。总的来说，如果某个Sprint对于其所在环境来说失去价值和意义，那么它就应该被取消。然而，因为Sprint周期都较短，所以很少发生取消Sprint的情况。

当某个 Sprint 被取消时，任何做完和“完成”的产品待办事项列表条目都需要评审。假如有些条目已经相当于潜在可交付的产品增量，那么它们是可以被采纳的。其他的条目就都要复原到最初的估算，放回到产品待办事项列表中去，任何相关工作就都被看作是徒劳的了。Sprint 终止会消耗资源，因为每个人都需要在新召开的 Sprint 计划会议中重新组合，开始另一个 Sprint。Sprint 终止对团队的不利影响非常大，被终止的情况也非常的少见。

## Sprint 计划会议

Sprint 计划会议制定迭代计划。对于一个月为周期的 Sprint，计划会议的时间盒限定为 8 小时。对于较短的 Sprint，根据 Sprint 的长度，按比例缩小会议的时间。（比如，两周的 Sprint 对应 4 小时的 Sprint 计划会议。）Sprint 计划会议的内容包括以下两个部分：在第一部分中决定 Sprint 需要做什么。第二部分（4 小时的时间盒对应一个月的 Sprint），团队研究在 Sprint 内如何将功能构建成产品增量。

Sprint 计划会议包含两部分内容：“做什么”和“怎么做”。一些 Scrum 团队将这两部分结合起来。第一部分，Scrum 团队处理“做什么”的问题。产品负责人把具有最高优先级的产品待办事项列表呈现给团队，并一起决定接下来的 Sprint 中开发什么功能。Sprint 会议需要投入的要素包括产品待办事项列表、最新的产品增量、团队的能力和以往的表现。团队自己决定选择待办事项列表的数量，因为只有团队可以评估在接下来的 Sprint 内可以完成什么工作。

选定产品待办事项列表后，Sprint 目标也就明确了。Sprint 目标通过实现产品待办事项列表来达到。Sprint 目标为团队提供指导，使团队明确构建增量的目的。Sprint 目标是发布目标的子集。

确定 Sprint 目标的原因是在功能方面为团队留出一定的回旋余地。例如，上一个 Sprint 的目标可能是：“通过一种安全、可重获的交易中间件实现客户账户修改功能的自动化”。所以当团队工作时，就会紧记这个目标。为了达到这个目标，团队就需要实现该功能和技术。如果团队感觉实现过程比预期的难度大，那么就可以和产品负责人协商，只实现部分功能。

在 Sprint 计划会议的第二部分，团队需要处理“怎么做”的问题。在 Sprint 计

划会议的第二部分（4小时时间盒对应一个月的Sprint）中，团队需要弄清楚如何将“做什么”会议阶段选定的产品待办事项列表转化成完成的增量。通常团队会先以设计展开工作，设计过程中，团队确定任务，这些任务就是将产品待办事项列表转化成可用软件的具体工作。任务需要被分解，以便在一天之内完成。这个任务列表就是Sprint待办事项列表。团队自组织承担该列表中的工作，也可以在Sprint计划会议或在Sprint中及时确定。

产品负责人会参加Sprint计划会议的第二部分，对产品待办事项列表做一说明，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、确定产品待办事项列表。团队也可以邀请其他人员参加会议，以寻求技术和领域建议。新组建的团队常常在这次会议中第一次意识到：整个团队要么一起成功，要么一起灭亡，没有个体这个概念。团队同时认识到必须依靠自己。正因为意识到这一点，整个团队才逐渐自组织并形成自己的特色，真正成为一个团队。

### 提示

*通常，Sprint 计划会议上只设计出 Sprint 待办事项列表的 60-70%。剩余部分后续完成，或者给出大体的估算并在 Sprint 中再分解。*

## Sprint 评审会议

Sprint末尾时要举行Sprint评审会议，一个月的Sprint通常对应4小时时间盒的评审会议。对于时间少于一个月的Sprint来说，根据Sprint的长度，按比例缩小会议的时间。（比如，两周的Sprint对应2小时的Sprint评审会议。）在Sprint评审会议中，Scrum团队和利益相关人沟通Sprint中完成了哪些工作。然后，根据完成情况和Sprint期间产品待办事项列表的变化，他们确定接下来的工作。这是一个非正式会议，会议中进行功能演示，以促进下一步工作的互助与合作。

评审会议至少要包含以下因素：产品负责人确定完成了哪些工作和剩余哪些工作。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。然后，团队演示完成的工作并答疑。产品负责人和与会人员讨论产品待办事项列表，

并根据不同的速率计划出可能的完成日期。接着，整个团体就哪些工作已经完成，同时这对下一步工作有何意义进行探讨。Sprint评审会议为接下来的Sprint计划会议提供了宝贵的参考信息。

## Sprint 回顾会议

在Sprint评审会议结束之后和下个Sprint计划会议之前，Scrum团队需要举行Sprint回顾会议。对于长度为一个月的Sprint，这是一个3小时时间盒的会议（根据Sprint的长度，按比例缩小会议的时间）。在这个会议上，ScrumMaster鼓励团队在Scrum过程框架和时间的范围内，对自己的开发过程进行改进，使下一个Sprint的效率更高、更易工作。许多书籍都介绍了召开回顾会议的技巧。

回顾会议旨在对前一个Sprint周期中的人、关系、过程和工具进行检验。检验应当确定并重点发展那些进展顺利的，和那些如果采用不同方法可以取得更好效果的条目。这些包括：Scrum团队构成、会议安排、工具、“完成”定义、沟通方法和将产品待办事项列表条目转化成“完成”工作的过程。在Sprint回顾会议的末尾，Scrum团队应该确定将要在下个Sprint中实现的有效改进方法。这些变化更适应于经验检验。

## 每日例会

团队每天进行15分钟的检验和适应的会议就称为Scrum每日例会。每日例会在各个Sprint都是在同一时间，同一地点进行。会议上，每个团队成员需要汇报以下三个问题：

1. 从上次会议到现在都完成了哪些工作；
2. 下次每日例会之前准备完成什么；
3. 工作中遇到了哪些障碍。

每日例会可以增强交流沟通、省略其他会议、确定并排除开发遇到的障碍、强调和提倡快速决策、提高每个成员对项目的认知程度。

ScrumMaster要确保团队举行每日例会，团队则负责召开会议。ScrumMaster指导团队执行规则来控制会议时间，并保证团队成员进行简短有效的汇报。

ScrumMaster也执行“鸡”不允许在会议上发言或以各种方式干扰会议的规则。

每日例会不是进度汇报会议。这个会议是为将产品待办事项列表条目转化成为增量的人（团队）召开的。团队承诺实现Sprint目标和完成产品待办事项列表条目。每日例会是检验朝向Sprint目标的进程（三个问题）。跟踪会议通常是对Sprint中的下一步工作进行调整，目的在于增加团队实现目标的可能性。这是Scrum经验过程中的重要检验和适应的会议。

## Scrum 工件

Scrum的工件包括产品待办事项列表、发布燃尽图、Sprint待办事项列表和Sprint燃尽图。

## 产品待办事项列表和发布燃尽图

产品待办事项列表列出团队正在开发的产品需求。产品负责人负责产品待办事项列表的内容、可用性和优先级。产品待办事项列表永远不会是全面的，最初的版本只列出最基本的和众所周知的需求。产品待办事项列表根据产品和开发环境的变化而演进。待办事项列表是动态的，它经常发生变化以确保产品更合理、更具竞争力和更有用。只要产品存在，产品待办事项列表就存在。

产品待办事项列表中包含产品开发和交付成功产品需要的所有条件和因素。表中列出了所有的特性、功能、技术、改进方法和故障修复等对未来发布产品进行的改变。产品待办事项列表条目包含描述、优先级和估算的特征。优先级是以风险、价值和必要性驱动的。评估这些特征的技巧有很多。

### 提示

产品待办事项列表条目通常是以用户故事的形式展现。也有可能是用户案例，但是用户案例更适用于开发生命关键型或任务关键型软件。

产品待办事项列表按照优先级排序。优先级高的产品待办事项列表需要立即进行开发。优先级越高，产品待办事项列表越紧急，就越仔细斟酌，而且对其价值的意见越一致。优先级高的产品待办事项列表更清晰直观，细节信息比低优先级待办事项列表的多，根据清晰的内容和详尽的信息做出的估算就更具价值。优先级越低，细节信息越少，少到只能勉强辨认出该条目。

随着产品的应用、价值的增加、市场的反馈，产品待办事项列表变成了庞大、更详尽的列表。因为需求不断发生变化，所以产品待办事项列表是个不断更新的文档。业务需求、市场形势、技术和员工的变化都会引起产品待办事项列表的变化。为了减少返工，只需要细化最高优先级的条目。在接下来的若干Sprint内将要进行开发的产品待办事项列表条目是细粒度的，已经被分解过，因此，任何一个条目在Sprint内都可以被完成。

若干个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事项列表只能有一个。那么这就需要使用产品待办事项列表的特征对条目进行分组。分组可以根据特性集、技术或架构进行。这也是Scrum团队经常采用的一种组织

### 提示

*Scrum 团队通常利用每个 Sprint 的 10%时间，提炼产品待办事项列表以达到上面的要求。当达到粒度要求时，产品待办事项列表顶端的（最高优先级、最高价值）条目会被分解，分解后的条目适合在单个 Sprint 内完成。在提炼阶段就已经对这些条目进行分析和考虑。当进行 Sprint 计划会议时，这些高优先级的条目就能充分地被理解，也可以很容易挑选出来。*

### 提示

*验收测试常常作为产品待办事项列表条目的属性。验收测试经常以条目开发完成后应实现功能的可测试描述取代细节文字描述。*

### 提示

*在某些组织中，向待办事项列表中追加的工作任务比完成的还要多。这就可能造成绘制的趋势图曲线是水平甚至是上升的。为了应对这种情况并保证高透明度，在追加或减少工作量时，标记新的底线。只有工作量发生重大变化时才有可能需要增加或除去底线，而且也应该详细记入文档。*



工作的方法。

发布燃尽图记录了在一段时间内产品待办事项列表的总剩余估算工作量。估算工作量以Scrum团队和组织决定的单位为标准，时间是以Sprint为单位。

在发布计划会议中对产品待办事项列表条目进行原始估算，之后创建条目。在产品待办事项列表提炼阶段，这些条目会被评审和修改。然而，对产品待办事项列表的估算可以随时更新，团队负责所有的估算工作。产品负责人在协助和指导团队权衡取舍时可能对团队所做决定带来一定影响。但是，最后的估算是由团队进行。产品负责人总是留有一份最新的产品待办事项列表/发布待办事项列表燃尽图。可以根据剩余工作量的变化绘制一张趋势图。

### 提示

*趋势图曲线对于发布的前两三个Sprint的可参考性也许不大，除非这些团队曾经在一起工作过，对产品认识深刻，并且理解基本技术。*

## Sprint 待办事项列表和 Sprint 燃尽图

Sprint待办事项列表包含团队需要执行的任务，从而将产品待办事项列表条目转化成“完成”的增量。许多条目在Sprint计划会议中已经定义。这些就是团队确认的完成Sprint目标所必须做的工作。Sprint待办事项列表条目必须被分解。进行充分的分解，那么过程中发生的变化就可以在每日例会上得到理解。一天或一天之内的工作量对Sprint待办事项列表条目来说是比较适合的。

团队在整个Sprint内都会修改Sprint待办事项列表，同时在Sprint内合并Sprint待办事项列表。因为它是针对每个成员的任务，所以可能有时任务会偏多或偏少，亦或某项任务耗费的时间超出或提前于预期。当出现新工作时，团队需要将其追加到Sprint待办事项列表中去。随着任务进行或者被完成，需要更新每项任务的估算剩余工作量。如果某项任务失去开发的意义，就可以将其除去。在Sprint内只有团队可以对Sprint待办事项列表进行修改，也只有团队可以对列表的内容或估算进行修改。Sprint待办事项列表是高可见度的，是对团队计划在当前Sprint内完成工作的实时反映，并且，该列表只属于团队。

Sprint待办事项列表燃尽图展现的是当前Sprint内剩余的Sprint待办事项列表工作数量。创建该图需要通过累计Sprint中每日待办事项列表估算来确定剩余工作量。Sprint内的剩余工作量就是所有Sprint待办事项列表的剩余工作总量。每天对这些数据进行跟踪，并绘制燃尽图，时刻显示剩余工作量。用线将图上的点依次连接起来，团队就可以控制完成Sprint工作的进程。所耗时长并不属于Scrum的关注范围，剩余工作量和完成日期才是利益相关因素。

### 提示

尽可能在一大张纸上手绘燃尽图，并将其悬挂在团队工作区域内。因为团队更倾向于去看一张大幅、清晰可见的图，而不是去打开Excel表格或其他文档工具。

Scrum的一个规则是关于每个Sprint目标的，即严格按照“完成”的定义开发潜在可交付的产品增量。

## 完成

Scrum要求团队在每个Sprint内构建产品功能的增量。这个增量必须是潜在可交付的，因为产品负责人可能会要求立即实现该功能。为了做到这一点，增量必须是最终产品完整的一部分，必须是“完成”的。每个增量都必须可以和前面所有增量累加起来，并且进行过彻底的测试，以保证所有的增量能兼容工作。

在产品开发中，宣称功能已经完成会让有些人认为这个功能至少拥有整洁的代码、经过重构、进行了单元测试、通过构建、完成了验收测试。另外一些人也许只认为该功能只是构建了代码。如果每个人对“完成”定义的理解不同，经验过程的另外两个支柱也就无法发挥作用。当有人说某个工作“完成”了，每个人都需要明白“完成”的含义。

“完成”解释了当团队在Sprint中承诺“进行”某个产品待办事项列表条目工作的意义。有一些产品不包

### 提示

“未完成”的工作常常会累积到产品待办事项列表中的条目，称为“未完成工作”或“实现工作”。随着这些工作的积累，产品待办事项列表燃尽图会比未进行累积时更精确。



含文档，所以“完成”的定义不包含对文档的要求。真正“完成”的增量包含了增量和其中所有产品待办事项列表条目必须的分析、设计、重构、编码、文档和测试工作。测试包括单元测试、系统测试、用户测试和回归测试，另外还有非功能测试如性能测试、稳定性测试、安全性测试和集成测试。“完成”也包含所有的国际化工作。有些团队尚没有能力将实现他们“完成”定义的所有要求都包含在内。这一点必须告知产品负责人。产品在实现、投入使用之前要把所有的剩余工作都完成。

## 最后总结

一些组织没有能力在一个 Sprint 内构建完整的增量。这可能是因为他们还没有自动化测试基础结构来完成所有的测试。这种情况下，需要为每个增量创建两种归类：“完成”的工作和“未完成”的工作。“未完成”的工作是每个增量中需要稍后完成的部分。因为增量符合“完成”的定义，而且产品负责人同样清楚这个定义，所以他就会知道在 Sprint 结束时应该检验什么。“未完成”的工作会追加到产品待办事项列表的“未完成工作”条目中，因此随着条目的累积，就可以准确地反映到发布燃尽图中。这个技巧为发布的进程创造了高透明度。Sprint 评审中的检验和适应也符合这个透明的标准。

例如，如果某个团队没有能力对每个产品待办事项列表条目进行性能、回归、稳定性、安全性和集成测试，那么这部分的工作相对于可以完成的工作（分析、设计、重构、编码、文档、单元测试和用户测试）来说就被累积下来。比如说，这部分有六项“完成”的和四项“未完成”的工作，即团队完成了一个产品待办事项列表条目的六个单位工作（团队根据其对如何去“完成”的理解基础上估算），那么当工作完成后，就在“未完成工作”产品待办事项列表条目中追加四个单位的工作。

Sprint 一个接一个进行，每个增量的“未完成”工作也就累积起来，这些累积的工作必须在产品发布之前得到解决。未完成工作以线性累积，尽管事实上这些工作根据不同组织特性会呈指数形式累积。发布 Sprint 可以追加到任何发布末尾，以解决“未完成”的工作。因为“未完成”的工作不是以线性形式增加的，因此追加的 Sprint 数目是不可预知的。