

在 Scrum 中实施敏捷建模

张逸

www.agiledon.com

摘要：Scrum 是一种敏捷过程，它使用迭代和增量方式管理和控制复杂的软件与产品开发。由于没有提供核心的价值观与指导原则，同时也缺乏具体的实践方法，Scrum 在实际开发过程中存在明显的不足。作为一种基于实践的建模方法，敏捷建模恰好可以弥补 Scrum 的不足。本文从敏捷建模提出的价值观、核心原则和核心实践等三个方面，阐明了敏捷建模与 Scrum 之间的契合关系，并给出了在 Scrum 敏捷过程中实施敏捷建模的实践方法。

关键词 Scrum; 敏捷建模; 建模活动; 模型

1. Scrum 敏捷框架

1.1 Scrum 概述

Scrum 是一种敏捷过程，它使用迭代和增量方式管理和控制复杂的软件与产品开发。Scrum 的开发流程非常简单。首先，Product Owner 根据客户的需求编写 Product Backlog，然后召开计划会议，评估各项功能的相对工作量，并确定 Sprint 的愿景和目标，生成 Sprint Backlog。然后，在 Sprint（即迭代）的开发过程中，召开每日会议，Scrum Master 通过它了解开发的进展以及出现的问题，检查团队成员的工作与进度。迭代结束后，团队会召开评审会议，向项目关系人展示可运行的增量版本，并检查是否达到了 Sprint 的目标。评审会议之后的回顾会议则会总结以往的实践经验，以提高团队生产力。

Scrum 的核心在于迭代。团队首先浏览开发需求，考虑可用技术，并对自身技术及能力做出评估。然后共同确定构建功能的方案，并每日调整方法，以应对新的复杂问题、困难和出乎意料的情况。团队找出并选择最佳方案去完成任务。此创造性过程便是 Scrum 生产力的核心^[1]。Scrum 的所有实践就是围绕着一个迭代和增量的过程开展的。

1.2 Scrum 的不足

与 XP (eXtreme Programming, 极限编程) 不同，Scrum 并没有提供核心的价值观与指导原则，也缺乏具体的实践方法，例如结对编程、测试驱动开发。Scrum 仅仅规定了实施的基本流程与检查表，它是一个开放的管理框架，重心在于项目管理，而不是指导团队成员如何进行开发。这既是 Scrum 的优点，因为它很灵活，能够适应大多数场景，也可以兼容并包地引入其他方法学所提倡的实践；同时也是 Scrum 存在的固有缺陷，使得它难以被

实践。如果没有一位优秀的 Scrum Master，而团队成员又缺乏自我组织和管理的的能力，就会让开发过程变得一团糟，团队成员将会无所适从。

在开发实践方面，Scrum 可以借鉴 XP 提倡的结对编程以及测试驱动开发实现编码，通过重构对编码进行调整以适应需求的变化。但是，Scrum 在建模方面却是一片空白。例如，Scrum 对于如何创建 Product Backlog，如何建立架构模型，以及如何在编码之前进行必要的模型设计，都没有给出具体的解决方案。缺乏正确的建模活动，就可能会对 Scrum 开发过程造成阻碍，影响团队达成 Sprint 的目标。

2. 敏捷建模与 Scrum 的契合

敏捷建模 (Agile Modeling) 是一个基于实践的建模方法，包括一系列以特定原则和价值为导向的，可被软件专业人员应用到日常工作实际做法^[2]。敏捷建模有效地将建模敏捷化，利用敏捷方法的思想对传统的建模理念进行了重新梳理，使其更加适用于敏捷开发。敏捷建模描述了一种建模的风格。当它应用于敏捷的环境中时，能够提高开发的质量和速度，同时避免过度简化和不切实际的期望。

敏捷建模可以弥补 Scrum 在建模方面的不足。如果说 Scrum 是一个对开发过程的所有活动进行了规定的基本框架，则敏捷建模由于其对建模活动的核心关注，极大地丰富和增强了 Scrum 的软件过程。建模在所有的软件开发中都是不可缺少的一个重要环节。传统的建模活动，常常会重视对文档与工具的使用，要求创建的模型涵盖软件开发过程的方方面面。这种重量级的建模活动与敏捷开发方法在核心思想上是相悖的。敏捷方法需要敏捷的建模，Scrum 自然也不例外。

敏捷建模定义了一组与轻量级的建模有关的价值观、原则和实践，并说明如何把它们付诸实施。本文将从敏捷建模的价值观、核心原则和核心实践三个方面讨论敏捷建模与 Scrum 的契合。

2.1 敏捷建模的价值观与 Scrum 的契合

敏捷建模的价值观包括交流、简单、反馈、勇气和谦虚。前面四条来自于 XP 的价值观，但完全可以说是敏捷开发的价值观。敏捷软件开发宣言强调与客户交流和团队合作。宣言对可工作软件的重视甚于详尽的文档，凸现了简单的价值观。宣言对变更的重视体现了反馈的重要性，以及拥抱变化的勇气。Scrum 同样体现了敏捷建模的第五条原则——谦虚。Scrum 将整个团队定义为一种角色，作为一个整体负责将 Sprint Backlog 转化为可运行的产品。

在开发过程中，团队成员需要管理自身的工作，同时对每次迭代和整个项目共同负责。如果没有谦虚的精神，Scrum 的团队是无法运作的。

2.2 敏捷建模的核心原则与 Scrum 的契合

敏捷建模提出了十一条核心原则。Ambler 认为，只有完全采纳这些原则，才能真正地宣称自己在进行敏捷建模。Scrum 虽然没有提出具体的指导原则，但在 Scrum 框架和实施流程中，仍然体现了部分敏捷建模的核心原则。表 1 展现了在 Scrum 项目中敏捷建模核心原则的适用性。

表 1 在 Scrum 项目中敏捷建模核心原则的适用性

敏捷建模的核心原则	与 Scrum 的契合
软件是你的首要目标	Scrum 坚持所有的 Sprint 都结束于演示，其目的就是要交付可工作的软件。
支持后续工作是你的第二目标	Scrum 认为，需求列表是推动迭代的主要力量，只要项目有资金，迭代就不会停止。项目的后续工作属于需求列表的内容。
轻装前进	Scrum 的最终产出物除了可工作的软件外，只包括 Product Backlog 和 Sprint Backlog。
主张简单	Scrum 主张在一开始就要保持设计尽可能简单。
包容变化	Scrum 要求 Product Owner 根据不断变化的商业环境对产品作出调整。
递增的变化	Scrum 属于增量式开发，要求团队在每个 Sprint 周期内完成一部分产品功能增量。
有目的地建模	与建模相关的原则，Scrum 并未要求
多种模型	与建模相关的原则，Scrum 并未要求
你需要一个技术知识工具箱	团队的基本要求。
高质量的工作	Scrum 要求开发过程具有可视性，提倡对最后结果会产生影响的各个方面必须是清晰可见的，同时要求频繁的检查，以及对不合格的内容进行调整。
快速反馈	Scrum 每日会议、评审会议与回顾会议反映了这一原则。

2.3 敏捷建模的核心实践与 Scrum 的契合

敏捷建模的精华在于它的实践，但敏捷建模的实践是在价值观和原则指引下体现的。它的核心实践分为四类，即迭代和增量建模、团队协作、简单性和验证。实际上，敏捷建模的实践并没有超出敏捷开发的范围之外，只不过它的关注对象被界定为建模活动而已。因此，敏捷建模的实践完全可以应用在 Scrum 的开发过程中。

迭代和增量建模实践与 Scrum 完全吻合，因为 Scrum 本身就是一种迭代和增量开发。既然建模活动贯穿整个项目开发周期，因而建模采用迭代与增量的方式自然顺理成章。Scrum 定义了团队角色，从而突出了团队成员的协作，成员作为一个整体参与到软件开发过程中。在 Scrum 中，每个成员都可能是建模人员，例如 Product Owner 对需求进行建模，对用户界面进行建模，团队成员对设计进行建模。简单性实践要求建模人员使用最简单的工具，创建简单的内容，简单地描述模型。归根结底，模型只需要传达它应该展现的内容，不管是需求分析，还是架构设计，都应该尽可能地保持简单，既不需要考虑格式，也不需要考虑完整，甚至可以丢弃那些已经实现了的模型。Scrum 大量使用了白板、索引卡、即时贴等简单工具，创建的模型非常简单，甚至是临时的。Scrum 同样重视对产品的验证，避免出现错误或与需求产生偏差。

3. 贯穿 Scrum 敏捷过程的敏捷建模

3.1 Scrum 软件生命周期

Scrum 并没有明确划分项目开发过程的阶段，而是将几种会议（计划会议、评审会议和回顾会议）定义为软件开发的里程碑。如果借用软件生命周期的概念，我们可以将 Scrum 划分为初始阶段、计划阶段、冲刺阶段与发布阶段。初始阶段的活动主要包括组建团队、准备资源和编写 Product Backlog。计划阶段包括了 Sprint 的两次计划会议。冲刺阶段即一个完整的 Sprint 迭代，周期通常不超过六个星期。发布阶段包括评审会议与回顾会议。此阶段结束后，将发布一个达到 Sprint 目标的增量版本。至于产品的维护，则属于 Product Backlog 的一部分，列入每次迭代的范围。

3.2 初始阶段的敏捷建模

Product Backlog 的编写与建模活动息息相关。Product Backlog 是 Scrum 的核心，也是一切的起源。从根本上说，它就是一个需求、或故事、或特性等组成的列表，按照重要性的级别进行了排序。它里面包含的是客户想要的东西，并用客户的术语加以描述^[3]。编写 Product Backlog 就是对需求进行建模。根据敏捷建模“主张简单”的原则，我们在描述 Backlog 的条目时，通常借鉴 XP 对用户故事的描述方式，而不是采用用例驱动的模式。可以使用 Excel 工具来创建一个 Backlog 表。敏捷建模认为“内容比形式更重要”，在表示 Backlog 时，我们甚至可以使用即时贴，将其展示在白板上，使每个人都能直接看到需求模型。

编写 Product Backlog 时，项目的利益相关人必须积极参与，和 Product Owner 一起确定 Backlog 的条目以及优先级。Product Backlog 应该能够“包容变化”，Product

Owner 通过与项目关系人的讨论，可以增加新的功能，或者根据新的需求变化对其进行修改。根据敏捷建模的“多种模型”核心原则，我们也可以在 Product Backlog 基础上，使用用例模型或用户界面模型，帮助说明 Backlog 的业务流程，促进开发人员对需求的理解。

3.3 计划阶段的敏捷建模

计划阶段仅仅包括两次计划会议，每次计划会议大约持续四个小时。第一个计划会议主要确定 Sprint 的目标以及 Sprint Backlog。第二个计划会议则需要确定 Sprint Backlog 中每个任务的承担人，并根据实际情况裁减 Sprint Backlog，生成最终的 Sprint Backlog。

敏捷建模认为，项目关系人应积极参与到需求建模活动中。Scrum 负责需求建模的主要是 Product Owner 和客户。在计划会议中，Product Owner 必须出席会议，以便对 Backlog 的需求条目进行解释，帮助团队理解需求。

敏捷建模的核心实践要求“与他人一起建模”。在计划会议中，团队常常会对功能需求进行拆分，其目的主要是为了更容易对工作量进行估算，但另一方面也是对需求的一种细化。一种最佳实践是将需求条目细分为任务。任务与需求条目的区别在于，需求条目属于可交付的内容，是 Product Owner 以及他所代表的利益相关人所关心的。而任务则不可交付，它常常代表了分析、建模、编码、测试等实现需求条目的各个环节。在拆分任务期间，并不会真正开展建模活动，但团队成员在了解到需求的具体细节时，实际上已经开始考虑需求的实现。

计划会议会对 Sprint Backlog 进行工作量估算。Scrum 建议发挥集体的智慧。方法是利用计划纸牌。团队中的每个人都可以在深思熟虑之后，出示自己手里的纸牌，根据出示纸牌的数值取平均值，就可以大致获得该需求条目的工作量。这种方式符合敏捷建模“简单”的价值观。在讨论 Sprint Backlog 的需求细节时，则可以使用索引卡。根据每个需求的重要程度与优先级依次将索引卡张贴在墙上。索引卡属于敏捷建模的临时模型，在失去价值之后可以考虑丢弃，而只保留更新后的 Sprint Backlog。

3.4 冲刺阶段的敏捷建模

整个冲刺阶段就是一个迭代周期，即一次 Sprint。在 Sprint 的开始阶段，我们可以根据 Sprint Backlog 建立一个任务列表模型，以及一个能够直观反映开发进度和开发效率

的 Burndown (燃尽) 模型, 并形成任务板。任务板要尽量简单, 只需要保留必要的列。Scrum 要求召开站立的每日会议, 通常就在任务板前完成。团队成员一边描述昨天已经做的和今天要做的事情, 一边移动任务板上对应的即时贴。每日会议结束, 则任务模型也随之更新, 最后由 Scrum Master 负责更新 Sprint Backlog 和 Burndown 模型。

在冲刺阶段引入敏捷建模非常必要, 它有助于解决团队在开发过程中遇到的需求、设计以及开发方面的问题。一个方法是召集相关人员举行简短的设计会议, 这在敏捷建模中称为专门或即兴建模会议。通常的流程是: 首先与项目关系人探讨相关的需求, 可能需要创建基本用户界面模型或者讨论业务规则的逻辑; 随后继续前进讨论这些需求潜在的解决方案, 这时常常会画一张白板草图来帮助讨论; 再往后就是继续前进编码并测试这个解决方案^[4]。

Scrum 团队没有设计人员、建模人员和编码人员之间的区别, 它是自组织、自管理的团队。团队的每一个成员都具有项目中所有方面的参与权力, 不存在单一的团队成员对系统架构、需求或者测试负责的情况^[5]。这正是敏捷建模“有效团队协作的实践”的运用。

在冲刺阶段, 通过引入敏捷建模, 我们可以在开发过程中创建架构模型、类结构模型和测试用例模型等内容。根据项目的实际情况, 我们可以选择使用 UML 建模工具, 或直接利用简单的白板工具创建架构模型, 利用 CRC 卡展现类的结构模型。我们还可以借助一些需求模型以及用户界面模型, 深入对需求的理解。

3.5 发布阶段的敏捷建模

Scrum 评审会议实际上就是一次增量产品的演示和发布。在进行 Sprint 演示时, 需要确保清晰阐述了 Sprint 目标, 并让演示关注于业务层次, 而不要考虑技术细节。如果我们在冲刺阶段严格地遵循了持续集成原则, 就可以在每次 Sprint 之后发布一个增量版本, 供用户使用。这实际上是“快速反馈”和“包容变化”核心原则的体现。通过每次迭代发布的版本, 可以及时获得客户的反馈, 验证实现是否与需求相符。如果出现偏差, 或者客户提出新的建议和变化, 就可以将其列入到下次 Sprint 的范围和目标中。

回顾会议在 Scrum 中是一项特殊的活动。审视和适应的能力是 Scrum 的基础, 这也是开展回顾会议的目的。在回顾会议期间, 项目团队会分析 Sprint 中的成功经验和遇到的障碍。Scrum 回顾会议不会涉及建模活动, 但它对于敏捷建模而言却具有促进作用, 因为我们可以从回顾会议中总结敏捷建模应用的得与失。例如我们可以讨论建模活动是否过于复杂, 是否需要引入其它的建模工具, 哪些模型属于临时模型或契约模型。简而言之, 在回顾会议中, 我们可以检查团队的建模活动是否背离了敏捷建模的价值观、原则和实践。

4. 结束语

在 Scrum 项目中, 建模活动仍然属于必不可少的一个环节, 但却是很多 Scrum 实践容易忽视或轻视的一部分。Scrum 敏捷框架的不足主要体现于此。如果将敏捷建模的价值观、原则与实践应用到 Scrum 的整个开发过程中, 将有利于规范 Scrum 的建模活动。二者的关系是框架与细节的有机契合。Scrum 提供了一个基础的框架, 对敏捷开发过程中的所有活动进行了规定, 而敏捷建模的重点则是全部软件过程的一部分, 因而需要与另一个完整的过程结合, 以增强这些过程。敏捷建模是 Scrum 的有效补充, 在 Scrum 中实施敏捷建模, 能够提高 Scrum 的可操作性, 并在建模活动方面给与指导与规范。敏捷建模帮助 Scrum 团队找到建模的最佳点, 保证我们既进行了足够的建模, 以保证有效地研究和记录系统, 但又没有过多地建模以致变成减慢项目进展的负担。

参考文献

- [1] Ken Schwaber. Agile Project Management with Scrum[M]. 上海: 世界图书出版公司, 2007: 6.
- [2] 王毅嘉, 张为群. 一种基于 UML 和敏捷建模的 JEMM 方法研究[J]. 西南师范大学学报(自然科学版), 2005, 30(3): 426.
- [3] Henrik Kniberg. Scrum and XP from the Trenches [M/OL]. C4Media, 2008, <http://infoq.com/minibooks/scrum-xp-from-the-trenches>
- [4] Scott W Ambler. 敏捷建模——极限编程和统一过程的有效实践[M]. 张嘉路, 朱鹏, 程宾, 译. 北京: 机械工业出版社, 2003: 120.
- [5] Robert C Martin. 敏捷软件开发——原则、模式与实践[M]. 邓辉, 译. 北京: 清华大学出版社, 2003: 7.