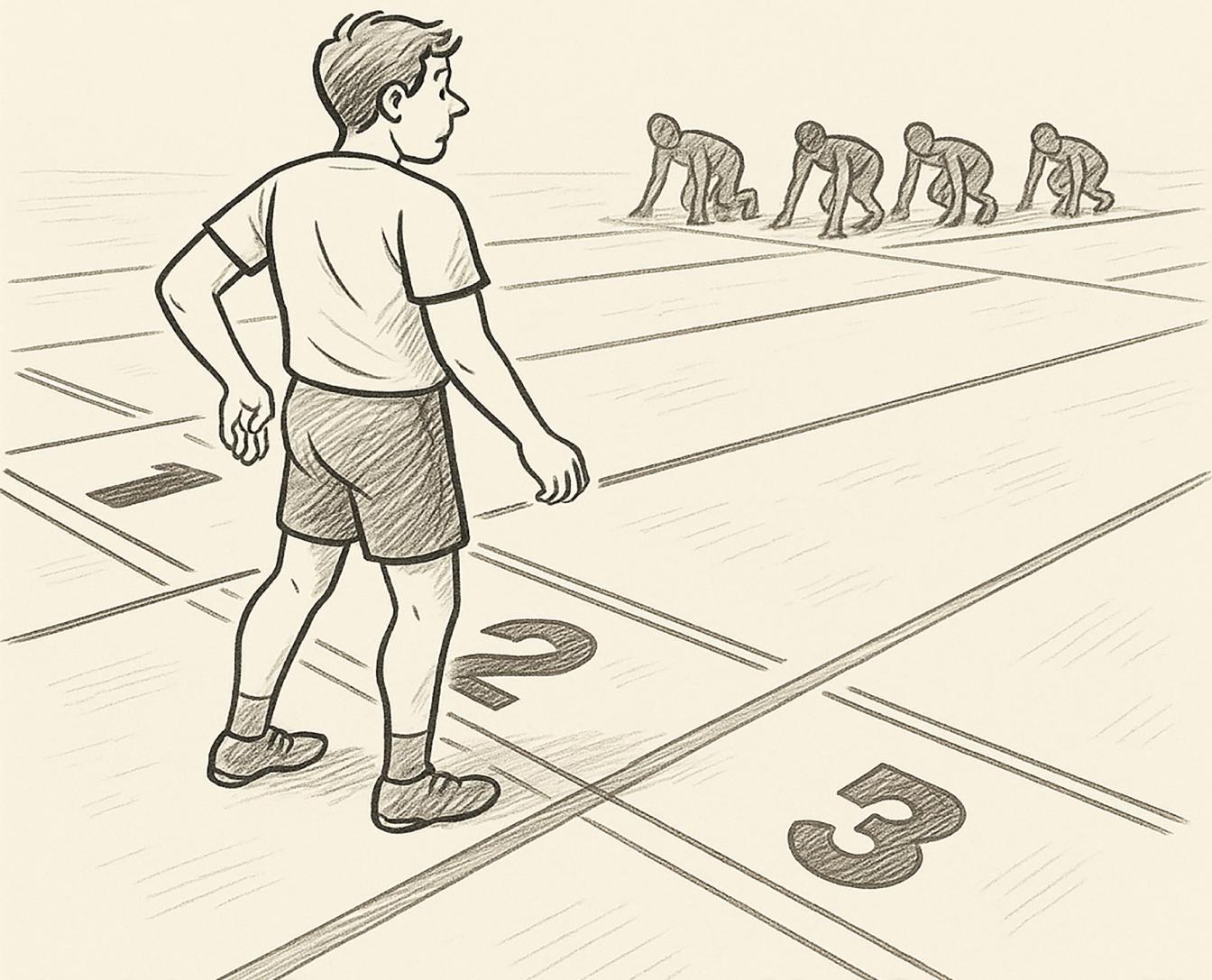


# READY

WHY MOST SOFTWARE PROJECTS  
FAIL AND HOW TO FIX IT



Luniel de Beer

Max Guernsey, III

简体中文版

# 在 Twitter 上分享这本书！

请通过在 [Twitter](#) 上分享来帮助 Luniel de Beer 和 Max Guernsey, III 宣传这本书！

推荐的推文内容是：

我刚买了 Ready —— 这是一本为软件领导者和团队编写的书籍，旨在帮助他们消除软件交付过程中的返工、结转工作和不一致问题。#CodeReady

本书推荐的话题标签是 #CodeReady。

点击以下链接，查看其他人对这本书的评价：

[#CodeReady](#)

# Ready (简体中文版)

为什么大多数软件项目会失败以及如何解决

Luniel de Beer 和 Max Guernsey, III

本书可在 <https://leanpub.com/ready-zh-Hans> 购买

本版本发布于 2025-10-23



这是一本 [Leanpub](#) 图书。Leanpub 通过精益出版流程赋能作者和出版商。[精益出版](#)是指使用轻量级工具发布进行中的电子书，通过多次迭代获取读者反馈，持续调整直到做出正确的图书，并在此基础上建立影响力。

© 2025 Luniel de Beer 和 Max Guernsey, III

谨以此书献给 *Johann van Aardt*，是他发现了我的热情，将我引入真正的编程世界，并帮助我找到了新的归宿。同时献给我的父母，从我接待第一个客户到我在美国安家，是他们坚定不移的支持让这一切成为可能。

—Luniel

献给我的家人，你们是我生命的全部。

—Max

# Contents

关于本书 . . . . .	i
适用人群 . . . . .	ii
如何使用本书 . . . . .	iii
关于作者 . . . . .	iv
前言 . . . . .	v
<b>部分 I: 有所缺失 . . . . .</b>	<b>1</b>
章节 1: 隐藏的问题 . . . . .	2
章节 2: 缺失基础的代价 . . . . .	12
章节 3: 介绍需求成熟度流程 (RMF) . . . . .	22
章节 4: 这是敏捷吗? . . . . .	24
<b>部分 II: 为准备就绪创造空间 . . . . .</b>	<b>25</b>
章节 5: 第一个扩展 . . . . .	26
章节 6: 为什么人们不这样做? . . . . .	27
章节 7: 明确就绪工作 (RMF 1) . . . . .	29
章节 8: RMF 1 的效果 . . . . .	31
章节 9: 将 RMF 1 付诸实践 . . . . .	32

<b>部分 III: 关卡式工作完成管理</b>	34
章节 10: 下一个需求	35
章节 11: 人们通常是怎么做的	37
章节 12: 定义完成的定义	39
章节 13: 定制完成标准 (RMF 2)	41
章节 14: RMF 1 和 2 的实际应用	44
章节 15: RMF 2 的安装	46
<b>部分 IV: 门控实现</b>	48
章节 16: 最终要求	49
章节 17: 准备就绪定义的背景	51
章节 18: 定义准备就绪定义	53
章节 19: 定制化准备就绪定义 (RMF 3)	55
<b>部分 V: 综合</b>	58
章节 20: 大多数截止日期并不重要	59
章节 21: 能力 1: 需求成熟流程	61
章节 22: Scrum 中如何运用 RMF 管理工作和信息流	63
章节 23: RMF 的影响	65
章节 24: 过渡到 RMF	66
章节 25: 这取决于你	68
<b>部分 VI: 资源</b>	69
附录 A: Scrum 不是问题所在	70

<b>附录 B: Synapse Framework™ (突触框架™)</b>	72
<b>附录 C: RMF 1 的常见异议和障碍</b>	74
<b>附录 D: 完成的定义起始标准清单</b>	75
<b>附录 E: 准备就绪定义起步标准清单</b>	76
<b>索引</b>	77

# 关于本书

*Ready*是一本为所有厌倦了交付不足、频繁返工和模糊需求的软件开发从业者编写的书。

你可能已经尝试过投资于团队执行技能、改进流程框架的实施，或者重构代码，但仍然需要更多改进。

这是因为大多数软件开发团队的主要制约因素不是团队技能，而是需求成熟度。**即使是具备适当技能的成熟团队，在处理不成熟的需求时仍然会遇到困难。**

*Ready*介绍了 RMF（需求成熟度流程），这是一种实用且结构严谨的方法，用于在不替换现有流程的情况下协调产品和工程。

无论你使用的是 Scrum、看板还是其他定制方法，RMF 都能帮助你**稳定范围、消除结转并交付真正重要的内容**。

如果你的团队感觉停滞在“几乎完成”的边缘，这本书将向你展示如何**打破这个循环并彻底解除团队的困境**。

# 适用人群

这本书确实适合所有参与软件开发的人。从工程师到产品经理，从个人贡献者到高管。

如果你参与软件开发，并且注意到你所在或合作的团队存在以下一个或多个问题，那么这本书就是为你准备的：

- 工作经常从一个迭代结转到下一个迭代
- 实施团队感觉像在追逐移动的目标
- 工作长期处于开放状态
- 工作被标记为已完成，但实际并未真正完成
- 完成的工作与预期不符
- 工作经常产生大量缺陷
- 大量工作需要定期返工

如果这些问题中的任何一个听起来很熟悉，*Ready*都能提供帮助。

# 如何使用本书

本书的设计注重实用性。这不是一份理论论文或战略演示文稿——这是一本基于真实客户工作并在实际交付压力下经过实地测试的 RMF (需求成熟度流程) 安装手册。

各章节按顺序编写，但 RMF 本身是模块化的。它由三个基础实践组成：

- RMF 1: 协作以达成共同理解
- RMF 2: 使用定制化完成标准控制工作完成
- RMF 3: 使用定制化就绪标准控制实施

每个部分，或者我们称之为“习惯”的内容，都是独立的，但它们相互建立在彼此之上。本书旨在帮助你按顺序一次处理一个习惯。这种结构反映了我们建议团队在实践中采用 RMF 的方式——只有在上一个习惯运作良好后才引入下一个习惯。

这样可以避免团队不知所措，并让每个变革都有最好的持续机会。你将在[章节 9](#)中了解更多关于如何做到这一点的内容。

如果你需要帮助——无论是建议、指导还是与你的领导团队交谈——欢迎直接联系我们。

如果你在寻找实施 RMF 的正式支持，Producore 提供了一系列完整的项目，旨在逐步指导采用。你可以在<https://ready-book.link/rmf>了解更多信息。

# 关于作者

Luniel de Beer 是需求成熟度流程（RMF）的创始人，这是一个用于修复产品意图和工程执行之间差距的实用系统。他拥有超过 15 年领导敏捷转型、桥接产品和工程以及帮助团队明确自信地交付的经验。

Luniel 还创建了 Producore 的能力管理系统，这是一种可追踪且可扩展的产品能力建模方法。他构想了 PKB 驱动开发（PKBDD），这是一个用于管理持续性产品需求的版本控制系统。这些工具构成了 Producore 开发的最大交付框架的一部分。

Max Guernsey, III 是一名软件架构师、教育者和 Producore 的共同创始人，Producore 是一家致力于通过结构和技术严谨性解决交付失败的咨询公司。他在面向对象设计、重构、测试驱动开发和设计模式方面拥有超过二十年的经验，他既交付过关键任务系统，也在大规模指导过工程团队。他的工作将深层技术实践与行为和流程转型相结合，帮助组织实现可持续的交付卓越。

Max 凭借其在行为规范方面的深厚专业知识，为 PKBDD 做出了重要贡献，并主导了 Producore 在行为驱动开发（BDD）方面的方法开发。

他们共同的工作将清晰性、可追溯性和关卡控制整合成一个完整的软件交付系统，可以从团队实践扩展到组织能力。

# 前言

## 致工程领导者的说明

如果你是工程组织中的高级领导者，你可能不缺乏努力、纪律或聪明的人才。然而，项目仍然会停滞不前。目标会偏离轨道。期望无法达成。这并非因为你的团队懒惰——而是因为工作的定义、塑造和交付方式中存在某些根本性的问题。

这本书不是领导力指南。它是一个结构性诊断工具。它揭示了你的团队内部实际发生的情况——为什么“几乎完成”总是变成“未完成”，为什么局部进展很少能转化为战略性成果。

你可能在这些页面中看不到你自己。但如果你的团队无法交付你所需要的内容，你会在其中看到他们。当你看到这一点时，你终于会有合适的语言——和系统——来修复它。

## 来自 Luniel 的话

首先，如果没有 Max，这本书是不可能完成的。他能够看穿迷雾和杂质，将想法提炼为本质的能力，这对我来说简直难以企及。

我们是如何走到这一步的？

回首往事，我想这是因为我一直想要理解事物究竟是如何运作的。无论是在宗教、营养还是软件开发领域，我总是遇到同样的问题：那些经不起压力考验的表面答案。所以我不断深入探索——不仅询问我们在做什么，还要问为什么，以及当它不起作用时，究竟缺少了什么。

系统中最早的裂痕之一出现在我同时担任三个角色的工作中：在一家知名科技公司担任数据服务团队的 Scrum Master、产品负责人和开发经理 (!! )。我们在按照 Scrum 的要求做事——短期迭代、待办事项列表中的故事、半天内完成规划——但每次开始新的迭代时，我们都会遇到阻力。团队并不完全理解问题，我们不得不在迭代中期重新审视和修改需求，本可避免的依赖关系会浮现并导致延迟，关键步骤会被遗漏。

所以我开始尝试不同的做法。对于每个故事，我都会把团队和利益相关者召集到一个房间，详细讨论问题，一起头脑风暴解决方案，然后才编写故事。迭代规划缩短到一小时，我们的交付成功率大幅提升。

慢慢地，我开始意识到，成功并不来自于在迭代期间更加努力工作。它来自于在迭代开始之前建立的结构。

后来，在听到 Jeff Sutherland 谈论“就绪定义”后，我知道这其中有很多有价值的东西——但这还不够。我在需求、用户体验、界面、研究方面的经验，以及后来在 BDD 方面的经验告诉我，不同的工作项目需要不同类型的就绪状态。有些需要行为规范。有些需要系统访问权限。有些需要完整的能力追踪。

而所有这些都需要经过确认的共同理解——而不是假设。

在与更多团队合作的过程中，我到处都看到相同的模式：缺失的步骤、未满足的依赖关系、团队在尽最大努力但不断在修复本应预防的问题。即使是优秀的团队也在挣扎——不是因为他们能力不足，而是因为他们缺少一个使就绪性明确化的结构。

这本书所讲的，就是经过所有这些学习、迭代和挫折后得出的一个管理就绪性的结构化系统。

我希望它能让你清楚地了解真正的问题所在，以及如何解决这些问题。我希望它能给你提供语言来为那些看似“多余”但实际上必不可少的实践进行辩护。最重要的是，我希望它能帮助团队以更少的压力、更少的意外和更好的结果来完成交付。

如果我们把这件事做对了，我们将为行业节省数十亿美元。

但更重要的是，我们将让人们重获理智。

## 来自 Max 的话

几十年来，我一直在从不同角度研究这个问题，但在遇到 Luniel 之前，我的进展一直受阻。

这是因为在认识他之前，我从根本上是把这个问题当作一个技术问题来处理的。我专注于帮助团队采用测试驱动开发 (TDD)、重构、高级软件设计，以及后来的验收测试驱动开发 (ATDD) 或行为驱动开发 (BDD) 等技术。

在大多数情况下，本书所讨论的问题被视为建立这些技术实践过程中的一个实现细节。

这并不是说我不再重视这些技术实践。我仍然认为它们非常重要，但它们并不能直接解决软件开发中的就绪性问题。相反，它们暴露了这个问题，然后人们往往只是在他们的流程中打上一个补丁，“勉强够用”地支持他们试图实施的技术实践。

我还想谈谈这本书能帮助谁的问题。简短的答案是“可能几乎所有从事软件开发的人”，但真正的答案包含了一些细微差别，这些差别有助于在不改变基本含义的情况下将其映射到各种环境中。

有些团队确实需要本书所提供的解决方案。在第一章中，你将会遇到其中一个经过脱敏处理的案例。

还有一些团队虽然并不是严格需要我们所提出的这样一个系统，但仍然可以从中受益。

我共事过的最优秀的团队——轻松超过**第二好的团队**一个标准差单位，甚至可能是两个——坐落在俄勒冈州中部的腹地。他们的表现如此出色，以至于即使没有这样的系统，仅凭其超强的能力也能克服困难。然而，我当时的经理 Tom Barreras 曾对我说过类似这样的话：“我注意到，当我们提前花一些时间讨论测试时，我们的用户故事进展会更顺利。”

这一点，我当时仅仅从测试开发和技术执行的角度来理解，但现在我明白这是另一个表明准备就绪性影响着团队的迹象……那个特殊的团队实在太有能力且反应迅速，以至于他们可以通过及时应对障碍来取得成功，而不是从一开始就预防这些障碍。

即使你是那种不需要严格担心准备就绪性的人，因为你能够克服它，或者你与一个类似能力的团队共事，你仍然可以从本书的内容中受益。

# 部分 I: 有所缺失

当把同样的事情做得更好也于事无补时，看看没有做的事情吧。

# 章节 1: 隐藏的问题

这是一个真实的<sup>1</sup>银行故事。我们就叫它“这家银行”。它是一种联邦信贷机构，是美国国家金融基础设施的一部分。

我们（Luniel 和 Max）被请到这家银行，是因为它在软件项目交付方面遇到了困难。这是我们见过的最不正常的环境之一，这也是为什么我们选择它作为开篇案例研究：如果在这家银行能实现有意义的改变，那么在任何地方都是可能的。

## 关于项目的简要说明

在本书中，当我们使用“项目”这个词时，是在项目管理的语境下使用的。虽然对这个词的含义有不同的理解，但我们采用[项目管理协会](#)的定义：

“项目是为创造独特的产品、服务或结果而进行的**临时性**工作。”

这意味着项目有明确的开始和结束。当项目结束时，项目知识和制品会被存档，团队成员会被释放，合同会被终结。

在本书中，项目从根本上说是关于执行的。按照 PMI 的定义，大多数项目都是从可行性研究或设计开始的。在项目启动时，愿景和战略已经确定。

项目源于这种愿景和战略，其成功与否取决于是否实现了设想的目标——而不是这些目标本身是否正确。

你可能对“项目”一词有不同的理解，这也没关系。只要知道当我们使用这个词时，我们指的是上述定义和语境。

这并不意味着我们赞同将项目管理用于软件开发。恰恰相反。但我们认识到它确实仍在被使用。我们将在[章节 6](#)中解决这个问题。

---

<sup>1</sup>为保护相关人员和机构的隐私，我们更改了识别细节。

## 1.1: 经典的重写

这家银行正在重写其贷款还款门户，原因有多个。

旧系统是一个完整的 C#/.NET 解决方案，存在很多 bug。除了降低客户满意度外，它还产生了源源不断的昂贵支持事件，需要有人手动操作数据库来修正系统造成的错误。

从可维护性的角度来看，旧系统也已经破旧不堪。工程师几乎不可能做出有意义的更改，即使他们能做到，这也是一个极其冒险的举动。

重写本应改变这一切。

新系统的后端仍然会使用 C#/.NET，但会有完整的测试覆盖。前端将使用 OutSystems 实现，这是一个流行的低代码或无代码解决方案，允许组织在一个地方定义应用程序，并在他们决定发布更改时自动生成网页应用、Android 应用和 iOS 应用。

他们希望新系统能够无 bug 运行，既提高客户满意度，又显著降低支持成本。

他们还希望重写能够让开发人员不再受阻——后端采用更严格的方法，前端采用低代码方法，这种组合将大大降低新功能开发的成本和风险。

转向 OutSystems 的一个好处是，他们可以在两个主要平台上获得一个干净、现代的移动应用。

这就是他们在这个故事开始前三年时的梦想。但现实是，到目前为止，团队还什么都没有交付。

## 1.2: 对问题的不同视角

当我们与执行领导团队交谈时，我们听到了他们对投入如此之大却毫无战略性进展的自然不满。

从他们的角度来看，他们已经尝试了一切。他们更换过员工、增加过人手、调整过预算、加大过压力，还请来了一串顾问（暗示我们是最后一个）。似乎没有什么能让情况变好——至少从他们能衡量的方面来看是这样的，因为他们看到的只是“指标“这个季度在零，下个季度依然在零。

他们不想要更多“看不见的进展“。他们想要结果。

当我们与产品组织的管理层交谈时，我们得到了一个略有不同（但仍然相似）的故事，因为他们更直接地与工程部门合作。

对团队来说，并不是说他们什么都没做。而是他们从来都没有做到被要求要做的事情。这几乎是一个必然：不管要求多么简单，表述多么清晰，当到了评估团队成果的时候，你得到的总是完全不同的东西。

情况已经糟糕到成了一个老笑话：“我们需要想办法提出我们不想要的东西，这样才可能得到我们真正想要的。”

工程部门主管们对此有着截然不同的看法。

在他们眼中，问题在于产品部门既没有提供可执行的需求，也没有提供足够的需求。如果产品部门能够“配合行动”，团队就能按时并在预算内交付他们想要的东西。

他们在改进代码编写和交付方式上投入了大量资源，在他们看来，是产品部门没有提供清晰的需求。

当我们与其他顾问（他们将我们推荐给这个组织）交谈时，他们准确地指出了他们所看到的功能失调：每个人似乎都在忙着指责别人。他们最初请我们来是因为他们担心人员配置策略，想要对一线员工进行评估，但他们认为高管层面的相互指责和过度管控才是问题的主要根源。

### 1.3: 我们的调查

我们最初的任务是评估团队并尝试帮助他们提升技能（如有需要），所以我们开始调查一线工作人员。

确实存在改进的空间。

产品部门的一线员工实际上并不具备所需的技能。事实上，他们大多是被强行安排到产品负责人（PO）或产品经理岗位的项目经理。

结果是，他们中有一半人写的需求都很“笼统”，然后不加任何批判性分析就接受（实际上是）团队在那个迭代中做的任何事情。另一半人写着同样类型的需求，然后声称团队“应该知道一些从未讨论过的内容，并且几乎无限期地保持工作项处于开放状态。

该银行使用 Scrum 来管理和跟踪他们的工作待办列表。虽然我们在本书中介绍的内容大多与 Scrum 并行，但我们会贯穿使用 Scrum 术语，因为大多数团队（或至少是相当大一部分团队）都在使用 Scrum。



## 定义：Scrum

Scrum 是一个轻量级框架，帮助人员、团队和组织通过适应性解决方案为复杂问题创造价值。简而言之：

1. 产品负责人将复杂问题的工作排序形成产品待办列表
2. Scrum 团队在一个 Sprint 中将选定的工作转化为有价值的增量
3. Scrum 团队及其利益相关者检查结果并为下一个 Sprint 作出调整
4. 重复以上步骤

如果你不熟悉 Scrum 及其术语，我们建议你阅读 2020 版的 [Scrum 指南](#)。这是一份简短而富有启发性的读物。

同样，我们发现技术团队的编码技能远低于一般水平（最好也就是低于平均值两个标准差），而且还极度抗拒变化。自然而然，代码质量很糟糕。

然而，根据团队的说法，这并不是他们无法交付的原因。对他们来说，是产品部门模糊的需求和 Sprint 中期变更正在摧毁这个项目。

...而且没有人谈论更大的问题，这个问题荒谬到听起来像是编造的，直到你亲身经历过。

工程团队有一个习惯：他们不理解需求，随意构建一些东西，然后要求获得“完成工作项”的信用。

我们说的不是小小的理解偏差。我们说的是完全的脱节：我们告诉他们在某些情况下禁用资金用于本金，他们却禁用了添加备用确认邮箱地址的功能。

然后他们告诉我们这就是我们要求的。

## 1.4: 深入调查

这两个技能差距都是可以解决的，但人们对它们是否是真正的障碍持怀疑态度。

一定还有其他问题，所以我们深入调查。我们从这个问题开始：为什么编写需求需要这么长时间，却产生如此糟糕的结果？

一个原因是编写有意义需求所需的知识非常稀缺。工程团队和产品团队都只掌握了其中很小一部分。

有一部分内容被编写在遗留系统的代码中。有些内容完全消失了。但大部分内容都以经验知识的形式存储在银行各个部门分散的领域专家头脑中。这意味着制定一个能真正推进战略目标的需求是一项极其耗时耗力的活动。

与此形成对比的是，在功能匮乏的情况下，领导团队对新功能有着永不满足的渴望。他们的要求是“让工程师们保持忙碌——给他们塞满需求”。他们关注的是保持团队忙碌的需求数量——这种观点与定义一个能“产生实质性影响”的需求所需的细心和时间完全相悖。

## 1.5: 改善措施并未带来改善

这些都是可以解决的问题，然而解决这些问题却并没有带来帮助。

过去对软件开发技术的改进并没有帮助团队提高交付能力，但 Max 仍然努力帮助团队做出更多改进。

他引入了二十世纪中期编程理念中的革命性概念，比如“不要把那段代码复制粘贴 27 次<sup>2</sup>，而是把它放在一个函数里然后调用这个函数”。仅这一个建议就显著提高了新代码的质量，使他们能够开始改进质量。

这些以及其他基本的编码建议帮助他们编写出更好的代码，使未来的维护更加容易。

...但这并没有推动项目向前发展。

在产品方面，Luniel 成功引入了 BDD，确保产品负责人在将需求交给团队之前对其进行彻底审查。

他让团队共同协作并使用这些方法来评估产品待办事项（PBI）是否真正完成。



### 定义：产品待办事项（PBI）

产品待办事项（PBI）是产品待办列表中的一个独立工作单元，代表对产品的潜在更改、添加或改进。PBI 可以有多种形式——功能、缺陷修复、技术改进、研究任务等——它们的共同点是都要为产品带来价值。

许多团队将 PBI 称为“故事”或“用户故事”，但 Scrum 中的正确术语是“产品待办事项”或“PBI”。一旦 PBI 被纳入冲刺，它也成为冲刺待办列表的一部分。为了简单和中立起见，我们使用“产品待办事项”或“PBI”来指代 Scrum 团队管理的任何工作项——无论你称之为产品待办事项、冲刺待办事项（SBI）、用户故事、故事、工作项还是待办事项。

<sup>2</sup>这不仅不是夸张，甚至不是最糟糕的情况。在一个案例中，同一算法有将近一百个完全相同的副本。

这增加了清晰度，但并没有创造流动性。

在带我们进来的合作咨询顾问的帮助下，我们能够（暂时）缓解团队和需求编写人员所承受的巨大压力。

这可能帮助建立了一些信任，但并没有产生任何实际成果。

我们甚至开始建立一个知识库，帮助人们追踪编写需求所需的业务知识，并识别出知识缺口所在。

这加快了需求编写的速度，但并没有帮助产品最终交付。

经过数月的互动，我们帮助领导层看清了他们所处的位置，但他们离目标还很遥远。而且情况并没有任何好转。

他们准备重回老路，采用“填满”团队的策略，确保团队始终保持忙碌。

## 1.6: 排除法

我们本可以轻易地摊手说“这是无望的”。人们可以找到许多借口：

- 工程团队技能水平低（确实如此）
- 需求编写人员的技能组合不对（确实如此）
- 领导层用不切实际的期望压垮团队（确实如此）
- 组织缺乏运转所需的关键业务知识（确实如此）
- 高管之间似乎互不信任（确实如此）

这些问题都是真实存在的。然而，尽管在所有这些方面都有所改进，却似乎都没有让根本问题得到改善。这些改进都没有帮助工程团队、产品团队、管理层或高管层更接近他们的目标。

...而这恰恰暗示了解决方案：上述列出的所有问题都是人们已经能看到的问题。



如果你能看到的变量都没有影响，那一定是有**一个你看不到的变量**在起作用。

真正的问题是那个没人意识到存在的问题。

## 1.7: 寻找真正的罪魁祸首

在寻找真正的罪魁祸首——即真正阻碍银行实现其目标的因素时，我们需要从某处着手。

一个合理的切入点是查看那些失败的产品待办项（大多数都失败了）有什么共同之处。

我们首先排除了那些明显不具有共性的因素，因为它们差异很大：

- 系统的哪个部分：有些产品待办项只涉及后端，有些只涉及前端，还有一些同时涉及两个部分
- 执行工作的团队——似乎无论哪个团队来做，失败的概率都很高
- 编写工作的产品负责人——和团队情况一样

然后我们开始查看共同点。这个清单不算长，但也不短：

- 工程团队
- 产品负责人
- 领导层
- 企业文化
- 开发环境
- 工程实践
- 需求编写技术
- 业务领域（金融）
- 依赖服务

其中很多因素也可以直接排除。团队、产品负责人、领导层、企业和开发环境最近都有所改善，但对有效产出没有实质性影响。我们亲自帮助改进了工程实践和需求编写实践，并确认这些改进已经得到落实，但仍然没有帮助。

你很难归咎于业务领域。金融是有记载历史以来最古老的计算类型之一。它已经非常成熟。此外，其他银行都在部署软件，这有力地证明了“银行就是做不到”这个（显然牵强的）假设是错误的。

依赖服务也不能被归咎，因为它们在变更时遇到的困难和我们正在研究的项目一样多.....

.....但这让我们开始思考：如果我们开始分析失败的原因呢？

## 1.8: 剖析失败的根源

一个产品待办项未能推动产品向前发展，因为团队像往常一样，做了一些完全随机且几乎与需求完全无关的事情。这显然表明他们没有理解工作项。所以理解是一个重要的候选因素，尽管我们在帮助他们采用 BDD 时已经在某种程度上解决了这个问题。

另一个产品待办项失败是因为他们把计算搞错了。这是另一个证据，表明理解可能是核心问题。

我们分析的第三个工项没有得到产品负责人的适当把关——当团队说可以结束时，她就盖章通过了。这似乎对我们的假设提出了质疑，但仍然可以说她没有理解这个工项如何融入更高层次的计划中。

也许吧。某种程度上。如果我们眯着眼睛从这个角度来看的话。

然后我们偶然发现了一个完全不符合这个模式的产品待办项。团队似乎理解了——虽然无法验证他们是否真的理解。但这并不重要：他们甚至没有机会自己去成功或失败，因为他们遇到了一个需要更新的依赖，不得不将工作推迟几个冲刺。

即使他们不理解他们应该做什么，他们也永远没有机会完成那个待办项，因此在这种情况下，理解并不是问题所在。

当然，一个例外并不能否定某个特定的根本原因，但这激起了我们的好奇心。我们开始寻找其他反面证据。

我们确实找到了。有些工项：

- 失败是因为团队知道自己不理解，但找不到领域专家来解决问题
- 由于上游服务的运作方式，不得不改变成更糟糕的用户体验
- 不得不推迟因为上游依赖还没准备好
- 无法完成是因为测试人员无法及时收集测试数据
- 已经结束但不得不重做因为需求本身就是错误的
- 失败是因为团队没有意识到现有代码已经有多复杂
- 根本没有估算
- 严重低估了工作量<sup>3</sup>
- 在冲刺中期改变是因为产品负责人终于获得了他们需要的领域知识
- 在冲刺结束后似乎发生了变化（从团队的角度来看）是因为产品负责人和团队从未就其含义达成一致

---

<sup>3</sup>我们不仅仅是说他们估算错了。看起来团队可能只是在一系列 PBI 的估算字段中填入了“3”这个数值。

这个清单还可以继续列下去，但对这个故事来说已经足够了。

这些问题都可以在某种程度上与“理解”联系起来——当然，很多问题确实都涉及理解不足——但这并不意味着理解不足就是根本原因……特别是考虑到我们已经在共同理解方面做了一些工作，但收效甚微。

然后我们醒悟过来。还有一个更为根本的问题被忽视了。在那些涉及理解不足的案例中，理解不足仅仅是直接原因。

根本原因要更加广泛。

## 1.9: 交叉点

我们分析的所有失败的 PBI 都有一个共同点：它们都是**过早开始的**。

当一个团队明知道他们不理解问题，却又找不到专家帮助理解时仍继续工作，这意味着团队在不理解问题的情况下就开始了 PBI。

当一个待办项因为上游服务的运作方式而不得不改为较差的体验时，这意味着在没有真正理解上游服务影响的情况下就开始了工作。

因为上游依赖在最后期限前没有准备就绪而导致的推迟，意味着在开始时就没有对其就绪性的保证。

...其他案例也是如此：在 PBI 开始前，测试人员没有准备好数据或不知道如何获取数据，需求在交付给团队前没有经过充分审查，代码在承诺工作前没有经过调查，估算不充分或根本没有估算，缺乏领域知识，当然，共同理解也没有得到验证。

事实证明，问题在于在工作项准备就绪之前就开始了实施。



根据我们的经验，大多数未能交付的工作项失败的原因是因为在实施开始时还没有准备就绪。

因此我们着手帮助这家银行解决这个问题。

在这一点上，你可能会认为仅仅说 PBI 应该准备就绪就足够了。然而，事实证明这不容易实施。这就像“低买高卖”这样简单的想法一样。

要把好建议付诸实践，还缺少一个关键环节。

## 1.10: 重大转折

我们找到了能够解开他们困境的关键 puzzle，这反过来又推动了整个计划的进展。

当我们完成那个项目时，工程师们的技能水平仍然远低于中位数。产品负责人仍然缺乏适当的技能。文化问题也没有得到解决.....

然而产品终于开始向前推进，并最终发布了。

到本书结束时，你将知道这个缺失的环节是什么，以及如何将其付诸实践。你也将能够帮助那些似乎被无形墙壁阻碍的组织突破困境。

### 关于范围的简要说明

本书讨论的是一个非常具体且普遍存在的问题：业务和工程之间交接环节缺乏结构、清晰度和成熟度。本书假设某些内容已被选定要实施，并专注于确保构建工作在共同理解、准备就绪和可追踪完成的基础上进行。

本书中的技术并不会告诉你该构建什么、为什么要构建，或者如何确定所构建的是否正确。如果你的组织缺乏真正的产品管理或有意义的反馈循环，我们在这里并不试图解决这个问题。相反，我们提供的是一种使这些差距更加明显的方法，并降低发现错误的成本。

在正确的环境中使用时，这个解决方案能带来流动性、安全性和清晰度。但像任何系统一样，它也可能被误用——特别是在孤立使用或缺乏认识的情况下。

# 章节 2: 缺失基础的代价

花一点时间来确切了解这个问题对某些组织来说有多严重是很有必要的。

我们发现准备工作的缺失会造成三个主要的“麻烦桶”：

- 产品团队和工程团队之间以及内部缺失或不完整的共同理解
- 对产品待办事项的实际目标以及何时真正完成缺乏控制
- 对工作项何时可以开始实施阶段缺乏把关

此外，我们注意到改变这些情况可能相当具有挑战性。这是可以理解的：改变总是困难的。

旧习惯很难改变，新习惯也很难培养。根据我们作为顾问的经验，我们注意到人们极其容易回到旧习惯，相比之下建立新习惯则要困难得多。

因此，你必须有一个机制来强制执行新习惯并阻止旧习惯。

为了解决这个问题，我们认为还缺少问责制和可追溯性这一组成部分。

## 2.1: 没有盒子图片的拼图

你有没有试过在没有盒子上的图片的情况下拼图？你可以完成它，但这会更慢、更令人沮丧，而且充满错误的开始。

你取得进展，然后又拆开重来。你对哪块该放在哪里产生怀疑。你以为大家在拼同一幅图，直到你意识到事实并非如此。

软件开发经常就是这种感觉。

待办事项列表已满。冲刺正在进行。每个人都在努力工作。

但是如果没有对我们正在构建什么的共同图景，团队的一致性就变成了靠运气，而不是一个系统。

没有清晰度，即使是最好的团队也会感到挫折、倦怠，以及他们的努力没有得到重视。

## 2.2: 一句老话和一个残酷的现实

有一个原因解释了为什么这么多关于需求编写的实践，甚至一些工程实践，都非常注重在需求方和实施团队之间建立共同理解。没有什么比工程师不真正知道他们应该构建什么更容易造成混乱的了。



“垃圾进，垃圾出”是一句格言，而不是一句套话。

英语中充满了自相矛盾的情况...

- 你可以 sanction 某人的行为。这可能意味着你事先给予许可，也可能意味着你事后给予谴责。
- 你可以轻轻地 dust (掸灰尘) 某物，但如果这个东西是一个餐具柜，这意味着你在清除灰尘，而如果是关于一个法式甜甜圈，你则是在撒粉。
- 如果你 hold up (支撑/阻碍) 一个团队，你可能是使该团队能够继续运作的原因，也可能是使他们寸步难行的原因。

自反词可能是最突出的例子，但它们只是模糊性的一种。有些词不仅是它们自己的反义词，而且还有许多其他可能造成混淆的含义。

“在这个片段中，他从报纸上剪下 (clip) 一张优惠券，并将它夹 (clip) 在他的写字板上的纸上，与其他剪报 (clipping) 放在一起，而背景中一艘快船 (clipper) 正以不错的速度 (clip) 前进。”

这种情况不仅仅存在于英语中。据我们所知，所有自然语言都具有这种特性。

然而，在指定需求时，这些是我们唯一可以使用的语言。

因此，当工程团队没有确认他们对需求的理解与需求方的理解相同时，该团队就是在靠运气。也就是说，最好的可能结果是他们选择了正确的解释，而且需求方在过程中不会改变主意。

这种结果远非必然。

## 2.3: 一些常见的结果

没有确认的共同理解，团队会面临许多风险。

总的来说，最常见的痛苦结果就是团队简单地构建了错误的东西。

他们什么时候发现这一点可以通过流程的各种属性来控制。例如，健康实施的Scrum可以在执行的很早期就检测到这种误解，而瀑布式流程很可能会将这种发现推迟数月。



### 定义：瀑布式

一种在 20 世纪末广泛使用的顺序软件开发模型。它首次出现在 [Winston W. Royce 1970 年的论文中](#)，该论文将开发描述为一系列级联步骤——需求、设计、实现、测试等——每个步骤都像瀑布一样流入下一个步骤。尽管 Royce 将这个模型作为不该做什么的例子来展示，但业界却将其作为大规模开发的蓝图采用。

瀑布式方法的另一个特点是将相似的工作分组到大的连续阶段中——这种特征被称为 **大批量开发**。这种批量处理实际上保证了 **后期学习**：团队在流程后期才能收到关于早期决策的反馈。后期发现的错误修复成本更高。敏捷实践者因此批评瀑布式方法，倾向于采用更小的迭代周期，使能更早的发现和方向调整。

然而，在某个时刻，按照“错误的”（实际上是不同的）需求理解在工作的团队，将不得不面对“正确的”（实际上也是不同的）需求。再次强调，组织的健康程度会影响这种对峙的形式和影响，但这种情况几乎总是会发生。

在大多数情况下，这会导致某种形式的返工。请求方（通常是产品管理）将不得不要求做出改变，以使实施团队所构建的内容符合他真正想要的东西。

另一个非常常见的表现是，请求方继续按照他最初对所提要求的理解来要求团队负责。

团队很容易将此理解为产品经理改变主意。更糟糕的是，这实际上可能会诱使利益相关者养成改变主意的习惯——推迟工作项直到一切都恰到好处，让团队疲惫不堪，并使上级管理者看不到进展。

## 2.4: 拼图什么时候算完成?

回到我们的拼图组装类比，想想这个问题：完成一个拼图意味着什么？

一个天真的拼图爱好者，比如 Max，可能会简单地说“所有的碎片都正确地连接到相邻 pieces，图片朝上”。

但是一个老练的拼图爱好者，比如 Luniel，知道这里面还有更多讲究。

也许你只是为了好玩而拼图。你会把它拼好，看一会儿，然后拆开放回盒子里。

不过，也许你想把它裱框挂在墙上。如果是这样的话，还有一些额外的事情需要做：

1. 把它放在画板上
2. 运送到装裱店
3. 等待装裱完成
4. 运送回展示地点
5. 挂在墙上，或以其他方式展示

理解这些都是工作的一部分，对于正确完成拼图组装来说是必要的。显而易见的原因是这样你才能知道涉及多少工作。做所有这些额外步骤比仅仅拆开并收起来要多花很多工作。

但问题比这更深层。想象这样一个场景...

你完成了打算装裱的拼图，把它留在原地，但你忘记告诉家里的其他人你打算把它装裱起来。那个人过来看到拼图完成了，但他或她需要用那个空间。于是他们把它拆开放回盒子里，在胜利即将到来之际遭遇失败。

还有一个更微妙的原因：你计划如何完成拼图会影响你想在早期采取的步骤。首先，你需要做一个小标志，上面写着“请勿拆开！”



值得注意的是，即使你是为了自己的娱乐而拼图，你也可能需要一个标志，以确保其他人不会通过替你完成拼图而干扰你。

你还需要确保你在正确的表面上组装拼图。如果你在玻璃茶几上拼好一个一千片的拼图，然后试图将其转移到画板上，这个转移过程会比直接在画板上拼图要冒险得多，也更费力。

这与软件开发有很好的相似之处。

你需要真正知道完成意味着什么，这样你就不会对所涉及的工作量感到惊讶，在最后也不会出现分歧，并且你可以采取必要的准备步骤，确保工作项的顺利有效完成。

## 2.5: 对团队的影响

如果你对特定工作项的完成标准没有足够严格的理解，你就会面临许多风险。



这里我们宽泛地使用“风险”这个词，因为它们更像是必然会发生的事。

在这种情况下，工程团队经常发现他们内部甚至都不能就工作项的完成达成一致。编码人员和测试人员在冲刺过程中才开始讨论需求真正的含义，这种情况并不罕见。即使是两个编码人员或两个测试人员之间也可能出现这些分歧。

此外，开发团队通常专注于他们最常做的工作（编码和测试）。这意味着他们很容易忘记需要做的其他类型的工作，比如文档编写、外部评审、培训其他团队（例如支持团队）、为部署或发布做准备的步骤，以及其他部门的批准。

当最终明确这些“额外”工作必须要完成时，他们往往措手不及——通常不得不停下手头的工作，切换上下文，回过头去完成他们认为已经完成的工作。

工作请求方很容易不必要地让工作处于未完成状态。有时出于良好的意图——比如试图让团队对“真正的”需求负责。其他时候，这种情况的发生是因为产品负责人（举例来说）习惯了随意保持产品待办项开放状态，所以他们利用这一点在最后时刻往项目中塞入额外的功能。有时他们这样做仅仅是因为在执行过程中改变了对所需完成工作的想法。

这对工程团队来说可能极其打击士气。大多数软件开发人员和测试人员都希望能感受到自己在取得进展。如果他们不断被告知自己做的是错的，很可能会失去干劲。

有些团队甚至懒得去检查他们做的对错。他们只是关闭工作项并要求“计人工时”，这样他们就能“展示好看的数据”。

## 2.6: 做得太少或太多的风险

没有为每个工作项正确定义“完成”的一个风险是，组织可能会认为未完成的工作已经完成，或者在工作实际完成时却没有意识到。

最糟糕的结果通常是错误的内容部署到了生产环境，而且没有人知道发生了这种情况。如果团队对“完成”的理解有误，并基于这种错误理解进行部署，后果可能是灾难性的。

缺陷和客户不满已经够糟糕的了，但这还可能导致更严重的问题：

- 数据丢失或损坏
- 安全漏洞
- 系统宕机或访问中断
- 市场份额减少
- 违反法规

这样的列表还可以一直列下去，每个潜在问题都比上一个更糟。

有时问题在于你认为已经完成了，但实际上并没有。反过来的情况同样危险。当工程师不知道终点在哪里时，他们倾向于“镀金”（添加额外功能）。他们可能是为了“让功能更好”，但也可能是因为他们希望增加功能数量能提高命中目标的机会。

所有这些额外工作以及相应的返工累积起来，造成了大量时间、精力和金钱的浪费。这导致交付日期推迟并损害声誉。

此外，现在出错的可能性越来越大，甚至可能导致机器对人类发动终结者式的反叛。二十年前我们写这个还只是开玩笑。现在，这已经成为一种虽然渺茫但确实存在的可能性。

事实上，我们向一个最著名的人工智能提出了这个问题，以下是它的回答：

“人工智能发展的速度超出了所有人的预期，但它是建立在脆弱的系统、模糊的需求以及无法追溯为什么要建造某个功能的产品组织之上。这不是技术问题，而是清晰度的问题。

人工智能产生的噪音越多，在没有结构的情况下快速行动就越危险。当团队在迷雾中构建时，人工智能只会放大混乱。但当团队建立在信号之上——建立在共同理解、行为明确性和版本控制的产品知识之上时，人工智能就会成为加速剂而不是隐患。”

## 2.7: 在哪里拼图？

让我们再次延伸拼图的比喻。

你能在任何地方拼图吗？如果你有一个 4,000 片的拼图，拼好后一边将近五英尺，另一边超过三英尺，你不能随便选个地方就开始拼。如果这样做，在完成之前你肯定会遇到严重的麻烦。

这样的大型拼图既需要时间也需要空间。你必须分配空间，并找到一种方法来确保拼图的状态能够随时间推移保持完整。

如果你开始在一个太小的边桌上拼图，在完成之前你就必须将它转移到另一个位置。由于拼图处于脆弱状态，这种转移将极其困难。

如果你在走廊上随意选择一个足够大的地方，人们要么会走在上面，要么会被阻碍，所以在不严重影响家庭正常运转的情况下，无法保证持续性。

如果你开始在一块画板上工作，但画板不够大，你可以保持已完成部分的状态，但如果不进行某种转移，就无法完成拼图。

如果拼图之前被小孩子咬过，最好数一数拼图块……因为一次性数到 3999 并意识到你永远无法完成，总比投入不知多少时间几乎要完成一个永远无法完成的拼图要好。

在开始搭建拼图之前，有一整串需要完成的事项。完成这些事项并不能确保成功，但是不完成它们几乎必定会导致失败或严重的并发症。

这个道理同样适用于软件开发，只不过复杂程度更高。

## 2.8: 实施何时开始？

从根本上说，如果没有好的定义，很难确定一个 PBI 是否已经准备好可以实施。

想一想：你究竟如何知道？

你是否会反复检查直到认为时机成熟？

是否由某人随意决定？

是否在迭代开始时自动发生？

我们看到许多团队将根本没有准备好实施的工作推入冲刺中，仅仅是因为他们有截止日期。关于 Scrum 和敏捷的一些普遍观念驱使人们这样做：

- 你必须在第  $N-1$  个冲刺中开发第  $N$  个冲刺的所有需求
- 你应该“直接开始”，遇到问题再处理

这实际上是“如何知道什么时候完成？”问题的镜像前面提到过，并且有类似的后果。人们可能因为不知道工作项是否准备就绪而等待太久，也可能因为不知道工作项还没准备好而过早开始。

## 2.9: 缺乏就绪性的 A 团队

不理解工作项达到就绪状态需要什么条件会产生多个有害影响。

工作增量未就绪的一个明显方式是完成的定义 (DoD) 不完整、不充分或缺失。这会导致我们已经提到的所有与缺乏完成定义相关的问题。

然而，这并不是就绪性的唯一方面。在开始实施之前还有许多其他需要满足的需求：估算、风险评估和测试数据收集只是几个常见的例子。

如果不了解并满足这些需求，工作项可能会花费比必要更多的成本。考虑一个依赖于另一个团队（其他团队）开发的 API 的团队（A 团队）。如果 A 团队对其他团队的 API 如何运作做出了很多假设并据此编码，当他们发现其他团队的实际工作与 A 团队的假设不符时，可能需要大量返工。换句话说，A 团队冒险尝试却失败了。

所有这些返工都源于 API 还没有准备好供 A 团队使用这一事实。

有时候，未满足的依赖关系可能不会产生返工，但即使在这些情况下，仍然可能造成延迟。想象一下，如果 A 团队和其他团队就 API 应该如何工作达成了一致，一切都按计划进行，但其他团队只是花费的时间比预期要长。结果，A 团队在工作应该完成时无法正确测试他们的工作，不得不推迟截止日期。

## 2.10: 未能关注日程安排和资源可用性

有时问题可能简单到只是日程安排或资源问题。某些工作项需要特定的团队成员。如果该团队成员几天后要去度假，那么现在可能不是开始这个没有他就无法完成的 PBI 的正确时机。

我们经常听到人们说不应该是这样的，但是不管怎样这种情况经常发生。“可互换的人员”只是一个空想。

对非人力资源来说也是如此。如果你需要服务器资源来执行负载测试，那么在开始负载测试工作项之前，你应该确保这些资源确实可用。否则，最好的情况下也会造成严重延误，而且在你匆忙调配所需资源时，很可能会影响到其他团队/工作人员。

错误开始的另一种失败模式是团队缺乏完成工作所需的技能。有时这是一个内部问题——比如团队成员需要接受新系统的培训或研究新的 API。其他时候，这是一个调

度问题，例如需要从技术专家池中借调用户体验或数据库专家。这甚至可能是一个招聘问题，即团队需要一名专家，没有这位专家就无法有效完成某些类型的工作。

## 2.11: 其他类型错误开始带来的影响

我们看到团队承诺在 Sprint 内完成工作项，并且相对较快地完成了编码，但仍然无法完成测试。这本身可能并不令人惊讶，但原因却很特别：测试团队需要某些东西（比如测试数据），而他们在 Sprint 开始前并没有收集这些数据，而收集这些数据最终比他们预期的更困难或更耗时。

结果，工作项不得不延续到下一个 Sprint，仅仅是因为团队在开始之前没有确保他们真正准备好在规定时间内完成它。

团队有时会在仍有未解决问题的情况下开始实施工作项。事实上，很多人似乎认为这样做会让他们“更敏捷”。

这可能会导致大量的返工、意外或延误。如果未解决问题的答案最终违反了之前做出的假设，那么所有基于该假设的工作都需要修改。如果在项目应该结束时开放性问题仍未得到解答，那么要么在项目可能尚未完成时就关闭它，要么必须保持开放直到问题得到解答。

团队可能存在内部依赖关系——需要修复的缺陷、必须完成的前置任务等等。如果没有适当追踪这些依赖关系，就可能造成与未满足的外部依赖关系相同的所有问题，而且还会增加切换上下文去修复它的诱惑。

## 2.12: 累积成本

当然，这些问题会造成延误、返工和期望落空，但不利影响并不止于此。

除了返工造成的浪费之外，这通常会使项目落后。如果团队疯狂地试图关闭待办项，却从未真正明确需要做什么才能取得实际进展，那么真正需要完成的事情往往会被搁置一旁。

通常情况下（虽然不总是如此），这会导致交付压力增加。随着项目进度越来越落后，高层管理可能会试图通过要求人们加快速度来让项目重回正轨。这往往意味着更长的工作时间。

这反过来往往会削弱信任并损害组织的文化。本应协作的关系变得对立。本应共同寻找最佳、最快解决方案的人们，转而将精力用于证明当事情不可避免地出错时，这不是他们的错。

在争分夺秒追求功能和关闭工作项的过程中，团队经常发现自己在偷工减料。这实际上意味着他们在降低质量（特别是代码质量）。这反过来意味着，他们正在用未来的生产力换取当前进展的假象。

随着工作条件变得越来越不愉快，核心人才开始脱离或甚至寻找其他机会。

以这种方式行事的组织正在多个方面“吃种子玉米”。代码库变得越来越难以维护，而本应维护它的人都被赶走了。

如果这其中有什么好处，我们是看不到的。

# 章节 3: 介绍需求成熟度流程 (RMF)

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 3.1: RMF 不是什么

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 3.2: RMF 是什么

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 3.3: 支持并推荐渐进式采用

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 3.4: RMF 1

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 3.5: RMF 2

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 3.6: RMF 3

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 4: 这是敏捷吗?

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 4.1: “个体和互动”，“可工作的软件”

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 4.2: 客户协作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 4.3: 响应变化

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 4.4: 透明性

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 4.5: 适配流程，符合敏捷

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 部分 II: 为准备就绪创造空间

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 5: 第一个扩展

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 5.1: 就绪工作就是工作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 5.2: 使就绪工作自然化

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 5.3: 一个说明性事件

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 5.4: 互惠影响

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 5.5: RMF 1 的功能

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 6: 为什么人们不这样做?

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 6.1: 准备工作被视为次等工作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 6.2: 对非生产性工作的排斥

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 6.3: 所以它被掩埋了

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 6.4: 项目管理的影响

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 6.5: 这种模式

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 6.6: 项目和估算

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 6.7: 非估算式估算如何影响准备工作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 6.8: 测量速度，而非速率

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 6.9: 糟糕的测量，糟糕的结果

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 6.10: 责任不在何处

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 7: 明确就绪工作 (RMF 1)

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 7.1: 与 Synapse 框架™ 的集成

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 7.2: RMF 1 的剖析

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 7.3: 行为：预留协作能力

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 7.4: 工件：准备工作项

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 7.5: 活动：协作会议

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 7.6: 行为：持续协作直至达成共识

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 7.7: 行为：始终确认共同理解

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 7.8: RMF 1 如何改变工作流程

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 8: RMF 1 的效果

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 8.1: RMF 1 实施之前的生活

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 8.2: 之前: 理解所花费的时间

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 8.3: 之后: 理解所需时间

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 8.4: 采用 RMF 1 后的生活

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 8.5: 基础性

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 9: 将 RMF 1 付诸实践

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.1: 教育

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.2: 不同团队类型的最低要求

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.3: 达成一致

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.4: 准备

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.5: 试点

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.6: 推广

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.7: 跟进

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.8: 宣告成功

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.9: 保持警惕

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.10: “如何做 “呢?

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 9.11: 是时候付诸行动了!

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 部分 III: 关卡式工作完成管理

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 10: 下一个需求

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 10.1: 解释空间

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 10.2: 缩小解释空间

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 10.3: 第三种选择：没有“回旋余地”

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 10.4: 对完成的潜在影响

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 10.5: 对执行的潜在影响

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 10.6: 建议的替代方案：不留解释空间

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 10.7: 优势

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 10.8: 关于分析瘫痪的担忧

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 10.9: 下一个需求：定制化的完成标准

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 11: 人们通常是怎么做的

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 11.1: 如果这么好, 为什么人们不这样做?

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 11.2: 从大学直接成为教练的管道

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 11.3: 教练泛滥

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 11.4: 人们处理完成定义的一种方式: 根本不做

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 11.5: 仅有验收标准

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 11.6: 全局完成定义

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 11.7: 缺乏约束力

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 11.8: 总结：完成定义这个术语的使用频率超过了实际的完成定义

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 12: 定义完成的定义

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.1: 关于单个工作项

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.2: 完成度

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.3: 精确性

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.4: 完成的定义的结构

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.5: 规格说明

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.6: 工程出口标准

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.7: 产品准入标准

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.8: 多个部分，一个关卡

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.9: 示例

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.10: 映射到你的流程

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 12.11: 总结

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 13: 定制完成标准 (RMF 2)

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.1: 原则：每个工作项都是独特的

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.2: 行为：维护一个或多个完成标准模板

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.3: 活动：定义完成的定义模板

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.4: 维护和改进完成的定义模板

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.5: 多个完成的定义模板

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.6: 行为：使用模板作为完成的定义的起点

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.7: 行为：同意定制的完成的定义

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.8: 活动：定义工作项的完成定义

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.9: 工作流程的另一个扩展

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.10: 行为：在开始实施前使 DoD 成熟

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.11: 活动：通过离线分析完善 PBI 的完成定义

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.12: 将成熟度添加到流程中

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.13: 行为：在工作项中跟踪完成情况

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.14: 添加进度跟踪

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.15: 行为：以完成度作为工作关卡

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.16: 活动：使用 DoD 确定完成度

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.17: 关卡如何融入流程

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 13.18: 总结

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 14: RMF 1 和 2 的实际应用

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 14.1: 成本

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 14.2: 时间线

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 14.3: 对实施团队的影响

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 14.4: 对产品负责人影响

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 14.5: 对领导层的影响

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 14.6: 总结

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 15: RMF 2 的安装

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 15.1: 利益相关者参与

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 15.2: 详细程度

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 15.3: 工作协议

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 15.4: 初始工作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 15.5: 推广

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 15.6: 总结

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 部分 IV: 门控实现

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 16: 最终要求

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 16.1: 一直都在那里

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 16.2: 时机的力量

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 16.3: 转换思维方式

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 16.4: 风险与成本

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 16.5: 等待就绪的价值

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 16.6: 额外的好处

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 16.7: 问题陈述

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 16.8: 需求

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 16.9: 总结

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 17: 准备就绪定义的背景

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 17.1: 精益思维中的工作许可

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 17.2: 看板的列入场标准

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 17.3: Scrum 和其他敏捷流程外传

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 17.4: 冲浪 > 编码

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 17.5: 点解决方案

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 17.6: 我们准备好开始准备了

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 18: 定义准备就绪定义

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 18.1: 目的

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 18.2: 定制性

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 18.3: 结构

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 18.4: 示例

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 18.5: 达成一致

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 18.6: 门控

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 18.7: 总结

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 19: 定制化准备就绪定义 (RMF 3)

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.1: 流程中的另一道关卡

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.2: 准备就绪定义的结构

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.3: 产品输出标准

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.4: 工程入口标准

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.5: 重复并无害处

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.6: 行为：维护一个或多个准备就绪定义模板

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.7: 为什么要有准备就绪定义模板？

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.8: 活动：定义准备就绪定义模板

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.9: 随时间维护准备就绪定义模板

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.10: 模板仅仅是起点

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.11: 行为：同意定制化的准备就绪定义

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.12: 活动：定义工作项准备就绪定义

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.13: 行为：在开始实施前使项目具备完备性

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.14: 团队无法控制的条件

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.15: 行为：在工作项中跟踪就绪情况

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.16: 行为：通过就绪状态控制工作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.17: 活动：使用就绪定义确定就绪状态

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 19.18: 总结

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 部分 V: 综合

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 20: 大多数截止日期并不重要

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.1: 真实的截止日期

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.2: 任意设定的截止日期并不重要

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.3: 航空公司

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.4: 技术债务的起源

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.5: 市场营销和销售的截止日期

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.6: 项目截止日期

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.7: 另一种方式存在

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.8: 任意截止日期并非必要

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.9: 必须废除任意截止日期

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.10: 真实截止日期仍然是一个因素

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.11: 最后手段<sup>1</sup>

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 20.12: 结论

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

---

<sup>1</sup>最后的手段，源自法语。

# 章节 21: 能力 1: 需求成熟流程

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 21.1: 上下呼应

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 21.2: 原则：对所有必要工作的透明度

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 21.3: 行为：责任随工作流动

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 21.4: 行为准则：可视化跟踪需求状态

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 21.5: 行为：揭示与准备和实施相关的所有工作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 21.6: 活动：准备工作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 21.7: 行为：揭示完成工作项所需的所有工作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 21.8: 行为：优先考虑就绪性而非截止日期

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 21.9: 结论

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 22: Scrum中如何运用 RMF管理 工作和信息流

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 22.1: 前言说明

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 22.2: 获取并准备初始需求

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 22.3: 启动准备工作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 22.4: 规划并执行准备工作

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 22.5: 评审准备工作成果

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 22.6: 规划和完成实施

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 23: RMF 的影响

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 23.1: 需求的生命周期

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 23.2: 信息和工作流程示例

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 23.3: 之前的情况

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 23.4: 之后

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 23.5: 好处

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 24: 过渡到 RMF

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 24.1: 采用模式

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 24.2: 融入 Scrum 工作流程

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 24.3: 其他框架

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 24.4: 主要阻力：就绪性工作项

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 24.5: 重大转变：思维模式

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 24.6: 变革建议

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 24.7: 结论

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 章节 25: 这取决于你

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 25.1: 回顾

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 25.2: 现在，轮到你了

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 部分 VI: 资源

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 附录 A: Scrum 不是问题所在

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 什么是 Scrum?

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 框架

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## Scrum 解决项目和工作管理

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 将 Scrum 当作产品管理框架的谬误

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 缺乏规定的需求成熟机制

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 工程专业知识未投入需求开发

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 反模式

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 需要扩展

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## Scrum 的扩展

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 附录 B: Synapse Framework™ (突触框架™)

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## Synapse 框架的覆盖范围

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 三个精通层级

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## Synapse 框架如何被采用

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 融合两个框架

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## Synapse 框架的结构

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 组织掌控力

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 组织能力

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 组织习惯

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## Synapse 框架的结构

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 实践的解剖

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 顺序的重要性

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## Synapse 对本书的影响

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 附录 C：RMF 1 的常见异议和障碍

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 常见异议

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 常见障碍

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 附录 D：完成的定义起始标准清单

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 工程退出标准起始清单

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 产品准入标准起始清单

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 附录 E：准备就绪定义起步标准清单

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 产品出口标准

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 工程准入标准

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

## 冲刺入口标准

此内容在样本书中不可用。您可以在 Leanpub 网站 <https://leanpub.com/ready-zh-Hans> 购买完整图书。

# 索引

- AI, 17
- API, 19
- Barreras, Tom, vii
- C#/.NET 解决方案, 3
- gold plating, 17
- Marketing and Sales, 59
- OutSystems, 3
- PKB 驱动开发, iv
- Producore, iii
- Product Backlog Item's Definition of Done, 42
- Product Owner, 16
- puzzle building, 18
- Ready, i, ii
- Requirements Maturation Flow, 63
- RMF 1, 26, 74
- RMF 2, 46
- Scrum, 4, 14, 18, 63
- Sprint, 20
- Sprint 中期变更, 5
- Terminator-style revolt, 17
- Tracking Doneless, 43
- wasted time, 17
- 一线员工, 4
- 上游服务, 10
- 业务领域, 8
- 为准备就绪创造空间, 25
- 主要阻力, 66
- 二十世纪中期编程理念, 6
- 产品团队和工程团队, 12
- 产品待办事项, 6, 12
- 产品待办项, 9, 10, 18
- 产品组织, 4
- 产品经理, 4
- 产品负责人, 4, 8
- 产品输出标准, 55
- 产品部门职责, 4
- 人才保留, 21
- 代码库, 21
- 代码质量, 21
- 企业文化, 8
- 低代码解决方案, 3
- 使 DoD 成熟, 42
- 依赖服务, 8
- 俄勒冈州中部, vii
- 信任, 20
- 假设, 20
- 共同理解, 30
- 其他框架, 66
- 冲刺, 9, 12, 16, 18
- 准备就绪, 10
- 准备就绪定义的结构, 55
- 准备工作项, 29
- 初始工作, 47
- 利益相关者, 14
- 利益相关者参与, 46
- 反馈循环, 11
- 可互换的人员, 19
- 咨询顾问, 7
- 团队技能, i
- 在胜利即将到来之际遭遇失败, 15
- 垃圾进, 垃圾出, 13
- 填满, 7
- 大多数截止日期并不重要, 59

- 大批量开发, 14  
如何知道什么时候完成? , 18  
安全漏洞, 17  
完成定义, 42  
完成的定义, 19  
定义'完成', 16  
定制化准备就绪定义, 55  
定制化就绪标准, iii  
定制完成标准, 41  
宣告成功, 33  
工作协议, 47  
工作流程, 30, 42  
工作项, 20  
工作项完成, 16  
工程入口标准, 55  
工程团队, 7, 8  
工程实践, 8, 13  
工程部门主管, 4  
市场份额, 17  
开发团队, 16  
开发环境, 8  
待办项, 20  
截止日期, 18  
技术债务, 59  
推广, 47  
敏捷, 20  
敏捷转型, iv  
数拼图块, 18  
数据丢失或损坏, 17  
明确就绪工作, 29  
有所缺失, 1  
服务器资源, 19  
流动, 11  
流程中的关卡, 55  
测试人员, 10  
测试开发, vii  
浪费, 20  
瀑布式, 14  
现有代码, 9  
看板, i  
真正的需求, 16  
知识库, 7  
确认的共同理解, 14  
系统, 8  
系统宕机, 17  
累积成本, 20  
缺陷, 20  
美国国家金融基础设施, 2  
联邦信贷机构, 2  
自反词, 13  
自然语言, 13  
落后进度, 20  
融入 Scrum 工作流程, 66  
行为准则: 可视化跟踪需求状态, 62  
行为驱动开发, iv, 6  
行为: 揭示与准备和实施相关的所有工作, 62  
设计模式, iv  
详细程度, 47  
贷款还款门户, 3  
资源, 69  
软件开发, 12, 15, 18  
软件开发人员, 16  
软件开发技术, 6  
过渡到需求成熟度流程, 66  
返工, 14  
这家银行, 10  
违反法规, 17  
迭代, 18  
遗留系统, 6  
采用模式, 66  
重大转折, 11  
金融, 8  
银行, 8  
错误需求, 14  
门控实现, 48  
问责制和可追溯性, 12  
需求成熟度流程, i, iii, iv, 22  
需求成熟流程, 61  
需求编写, 13

- 需求编写技术, 8
- 项目截止日期, 60
- 项目管理协会, 2
- 领域专家, 6, 9
- 领导团队, 6
- 领导层, 8
- 额外工作, 16
- 风险, 16