# 第二部分

# 基础：怀着终点开始

# 第2章 矩阵和混乱：从这里开始

## 本章内容

* Scrumban的根基和系统思考的重要性
* 对齐的相关性
* Scrumban如何与Scrum相辅相成的高层概览

学习层级：

本章的基础概念适用于所有学习者。

相信你阅读本书是基于某个原因,也许你想要评估Scrumban能否作为当前问题的潜在解决方案。或者也许你对这个框架好奇。无论你有什么动机,背后可能有一个“烂摊子”,以及导致烂摊子的先入为主的某些想法。

作为一个管理框架,Scrumban强调连同既有的工作方式来分层分析新的原则和实践进而发现知识。在你开始这旅程时,我要挑战你如下的问题：你对现实既有的认知是否会妨碍你对于新方法的发现和理解。

在开始这段旅程时,澄清终点也是重要的。定义一个清晰的终点能够帮助你避免无目的漫游。接下来的章节将聚焦在概念上,这些概念会帮助你识别并且理解你预期的目标。

在这个旅程的第一步,我要向你展示Scrumban建立的基础以及这些基础如何解决一般的挑战。有些也许看起来与你特定的需要是不明晰的,或者无关的,我鼓励你深入进去,将会体验到它们与你最终的成功是紧密相关的。

一路平安！

Scrumban故事

通过这本书,我从多样的案例中来分析归纳经验,这些案例会将反映各种不同类型的团队和业务是如何使用Scrumban来开展他们的事务,建立更好的工作方式,为他们公司得到更好的结果。有些组织得到了明确的识别,有些组织被归类到一个虚构的组织,我把它称为曼莫斯银行,但是数据和经验都是真实的,并整体呈现在附录里。

## 2.1 第一部分 矩阵——Scrumban 的起源

不幸的,互联网上重复的博客文章和其它资源不准确地定义了Scrumban。这个框架随着时间在演进,大量不准确的描述随之而来。正因为这个原因,我相信最好理解Scrumban的方式是投入些时间来了解它的起源。

### 2.1.1 Scrumban是什么？以及Scrumban不是什么？

理解Scrumban首先应当从理解它的根源开始-Scrum和看板。以我的经验看,两者都被极大的误解了。

Jeff Sutherland 和Ken Schwaber 说：“Scrum是一个开发和维护复杂产品的框架。”[[[1]](#footnote-1)] 在这个脉络下,Scrum既是一个项目管理方法,也是一个变化和涌现行为促进框架。

作为一个项目管理框架,Scrum说明了正式的过程(比如,冲刺会议,迭代,回顾)和角色(产品主管,Scrum主管,团队成员)。作为一个变化促进者,它在历史上依赖于承诺的概念,这概念是作为带来期望改变的倾向控制机制。虽然一些认证的Scrum培训师(Certified Scrum Trainers ,缩写CST)仍然继续强调承诺,但是这一因素在近几年中被淡化了。当前最新的观点是更加看重检视和调整的机制。不幸的是,承诺和检视/调整的概念都有实用主义的局限性。

这方面一个普通的例子是在我最近花多数时间服务的大型金融机构里,尽管多个顾问进行了广泛的教练工作,这个机构的多数经理和高级经理仍然不能简单地调整他们的观念来避免把分配在一个冲刺中的工作看作是除了承诺以外的其它事情。

Scrum框架允许当团队在当冲刺待办列表中发现更多关于工作的信息时可以对承诺再进行商量。尽管如此,Scrum框架提供了很少的指导和机制来帮助组织克服现实问题,比如这个例子中管理层展现出来的顽固抗拒。

作为一个过程框架,Scrum在团队层面看起来是非常有效的[[[2]](#footnote-2)]。然而,它并不是自然的能适合于整个系统管理[[[3]](#footnote-3)]。Scrum规定了4个正式检视调整的事件：冲刺计划、每日站立会议、冲刺评审,冲刺回顾。这些都聚焦优化工作,这些工作是由7-12人的小团队来担当。

Scrum将团队的工作隔离在时间限制的开发批量(冲刺),它鼓励各种技术来尽量使这批量工作可预测(来帮助团队满足其冲刺承诺),Scrum寻求尽量隔离外部对团队的影响。在Scrum,暴露过程不足和组织功能失效的首要机制是以团队为中心。

相似的,在这些问题和功能失效实际浮现出来之前,Scrum没有给出度量或者解决它们的具体方法。在那些情形下,它鼓励组织使用其它最适合处理那些问题的工具。在我的估计中,Scrumban就是那样的一个工具。

从一个实施的观点看,对比企业原来所使用的那些方式,Scrum经常代表了一个差异较大的软件开发方式,在那些环境中引入Scrum,常常引起剧烈的和颠覆性的变化。作为一个后果,Scrum被不公平的赋予了搞颠覆的特点。事实上,并非所有激进的改变都是破坏性的。相似的,在一个环境里颠覆性的变化在另外一个环境不一定也是颠覆性的。何时,如何,何地和何人,这些Scrum实施因子在决定颠覆程度方面比Scrum框架本身经常扮演更重要的角色。

现在让我们考察看板方法,它首先是一个促进演进变化和赋能持续改善的框架。这些目标中的部分是通过使用看板系统来达成,看板系统通过可视化来强调当前情况的更深理解。看板方法可以帮助通知我们哪些变化对一个组织会是颠覆性的,哪些不是。

看板方法的目标是结构化的促使演进变化。这样的演进变化从开始就显现为过程优化。随着组织能力成熟,看板方法能支持可持续的、更大的、更激动人心的、并且可管理的变化 。认识到看板方法既不是项目管理框架,也不是软件开发框架,这个认识是极为关键的[[[4]](#footnote-4)]。事实上,它并没有告诉我们该怎么做。更重要的是,它必须与现有的过程融合。

教练提示

关键概念先行

最后,看板方法也是一个使个人、团队和组织更好地理解他们是如何工作的框架。它使用了为数并不多的机制来简化整个系统的协调,并在跨多个子系统中最大化资源灵活性。模棱两可(或者说灵活多适应性)是故意构建的,因此框架可以成功地应用于任何上下文——Scrum或其他情况。

带走关键点：  
Scrumban不仅仅是关于利用Scrum和看板的一些元素来创建一个软件开发过程。更多的,它强调在Scrum上下文中应用看板系统,并将看板方法与Scrum一起作为一种进化的工具。最终,它将帮助和增强Scrum中固有的功能,并提供新的视角和能力。

### 2.1.2 管理知识工作

在20世纪中期彼得德鲁克提出的知识工作概念是非常重要的,它是一种在现代商业领域中占主导位置的工作。建筑、工程、法律、医学和许多金融都是知识工作的例子。软件开发和大多数IT功能也属于知识工作。

知识工作与大多数其他类型的工作是不一样的,,其主要的差异性特点是“需要综合收敛、发散和创造性思维来解决非常规问题”[[[5]](#footnote-5)] (一顶帽子给 reinhardt, 施密特等)。虽然生产完整的知识工作过程与生产实体货物的过程在许多方面是相似的,但是它在几个方面是有本质上区别。

25多年前,高德拉特(Eliyahu Goldratt)书写了商业小时《目标》,被多数出版商拒绝,挨了许多批评(文学和商业两方面),但后来高德拉特的书销售了数百万本。为什么？因为人们发现他在他的故事中揭示的原则和理论给制造过程带来了新观角,这导致了实质新的好结果。他这本书的形态同样也提供了一个很好的途径,在实施变化的组织里传播所需要的变化。

高德拉特书中揭示的原则在制造过程中运作良好,而当其试图适用到知识工作时,事实证明有更多挑战。 看板方法诞生于知识工作的独特特征。 我们来看一些例子。

知识工作是一个充满活力的过程,将“生产”中的学习和创造力融合在一起。相比之下,制造与设计产品和生产过程所需的脑力大相分离。尽管制造过程可能比较复杂,大多数制造过程“简单地”重复一系列步骤,以始终如一地生产具有已知价值的产品。 要考虑到的是,在制造过程中优化知识工作方面没有受到任何影响,反之亦然。

例如,Goldratt的约束理论(TOC)中可以用于改进制造过程的一种方法是识别系统瓶颈(工作流程中容量最有限的位置)。 最大限度地提高瓶颈的生产率可以提高整体系统吞吐量,因为输出受其最低生产部件的限制。 不幸的是,在知识工作环境中并不总是实用有效的。

也许“瓶颈”在于一个初级开发人员,他没有获得足够技能来与团队中高级成员达到一样的高产能。尽管可以通过把初级开发人员换成高效资深人员来提高吞吐量,但这样做实际上可能会增加总体业务风险。那就是开发初级团队成员的失败可能会使生产过程在其他方面变得脆弱。当您讨论将一个生产机器换到另一个生产机器时,这种变化的风险是不存在的。或者瓶颈在于一位高级团队成员,因为他或她的25%时间用于帮助其他团队成员,他或她的全部生产力被限制了。对这样的高级人员保持松弛可能更重要,因为如果不这样做实际上会降低整个团队的生产力。这种动态关系在制造环境下是不存在的[[[6]](#footnote-6)]。

此外,在工作流程中,关键信息到达的方式也有明显的不同。在生产过程中,需求通常是在工作开始前已经设置,设计是在生产开始前已经确定。相反的,知识工作,设计和增强升级都是生产过程的一部分。

最后,制造和知识工作之间最根本的区别之一就是可变性。在知识工作的各个方面都有非常高的可变性——从工作的性质到完成它所需的时间和资源的数量。为了处理这可变性需要引入了不少复杂的理论和实践,而在其他情况下是不必考虑的。

Scrumban 是我所选择的框架, 用于武装个人、团队和组织,以最大化提升他们有效管理知识工作的能力。

### 2.1.3 以系统思考来开始旅程

分解思维在工作和教育系统中根深蒂固,以至于很多人实际上习惯于分解信息而不是综合信息。这种做法对改善工作习惯有很大的阻碍,因为分解性思维将把系统看作是独立事物的集合,导致人们忽略影响每个人行为结果的交互和依赖性。俗话说,"没有人是孤岛"；需要许多事物的集体协调互动来维持商品和服务的生产。即使是孤独的别动队员和专家也依靠其他人和系统来开展业务。

教练提示!

重要概念先行

如果我们试图改善我们的工作条件并最大限度地发挥我们行动的价值, 那么我们必须认识到实现这一目标的最佳方法是避免把注意力孤立地集中在系统的任何特定组件。这样做需要大量并且刻意的努力, 它解释了我们为什么需要强调带着系统思维来开始我们的旅程。

不幸的是,我们所交互的系统是伪装的。

尽管像 Scrum 这样的框架是建立在系统思想之上的, 但它们并不扩展到帮助发现和管理系统全局影响范围。Scrumban有能力使此扩展更容易。

需带走的关键  
威廉·爱德华·戴明博士(Dr. W. Edwards Deming)是系统思考的先锋,他制作了演示, 创建了统计模型, 并策划了现实世界的商业研究, 显示了以系统角度如何思考在解决各种各样问题上是至关重要的。戴明估计, 94% 的业务问题源自我们所建立的过程, 而不是在这些过程中工作的人。改进系统通常会消除大多数问题。

那么, 什么是系统, 什么使它们与众不同？以下是一些基本特征：

* 系统是一组具有统一目的的组件
* 每个组件对整体都有影响
* 每个组件至少依赖一个其他组件
* 经常有延迟的反应, 使得因果关系难以察觉
* 系统输入、过程和输出有自然的差异变化
* 组件输出的总和不等于系统的输出。
* 系统有趋向于无序化 (熵) 的趋势。

汽车是系统的一个很好例子。它由许多组件组成, 它们一起将驾驶员和乘客从一个地方运送到另一处。这些部件包括发动机、变速箱、乘客车厢、后备箱等等。当我们把车拆开时, 它就失去了所有的本质特征；它不再是一辆汽车,仅仅成为一堆零件。正如本例所示, 系统从不是其零件的总和——它是这些零件相互作用的产物。

现在假设你想提升你的汽车,希望更有趣地驾驶它,并且更经济。如果你把所有的注意力都放在发动机上,你可能不会达到你的目标。这是因为你想要实现的系统输出依赖于许多组件的相互作用,而不仅仅是发动机的性能。如果你了解这个系统视角如何适用于你的车,那么你就可以将这种思维方式扩展到你的工作上。

操作提示

你如何辨别那些表面上旨在提高升“汽车” ( 系统性改善) 的努力,在实际上只是调整“发动机”(本地优化)？通过运用本书中概述的原则和实践！这里介绍的具体例子侧重于评价度量,这些度量直接衡量您所期望的系统产出,运用严谨的方法来解决问题, 等等。一旦付诸实践, 这些工具通过侦测系统瓶颈或约束,使我们能检测出那些干扰信号只不过是“搅动问题转圈”,而不是真正改变系统的能力。

### 2.1.4 Scrumban 可以包含其他模型和框架

下一级学习：

演进到更先进的实践

Scrum 和看板方法都是可以与其他过程和技术相结合使用的框架。看板能够在其框架的可视化和度量方面无缝地将各种各样的模型和方法集成融合,这使得它成为成功的强大工具。这些功能发挥了特别重要的作用, 因为组织将 Scrumban 的能力扩展到整个企业。刚开始采纳 Scrumban 原则的团队和组织最好是一次寻求只掌握一件事。在“守”学习阶段,引入多个框架或模型并不利于获得最大成效,或者获得最大加速。

在本书的稍后部分, 我们将考虑在 Scrumban 中集成多个模型和框架的影响, 比如：A3思维、Cynefin 框架 和高级统计建模。

Later in this book, we’ll consider the impact of integrating multiple models and

frameworks within Scrumban, such as A3 Thinking, the Cynefin framework, and

advanced statistical modeling.

Why Systems Thinking Is Relevant

### 2.1.5 为什么系统思维是相关的

学习层级：

本章的基础概念适用于所有学习者。

IT组织是系统,其所属的组织也是系统。它们由不同的部分(个人、其他团队、甚至其他系统) 组成,这些组成部分作为一个功能单元来生成软件。它们有将输入转化为输出的过程 (信息在其中流入和流出)。

Scrumban故事：曼莫斯银行

在2013年,一家大型银行的IT组织已经历经了八年的 "敏捷转型"。一家外部公司带领了最初的转型,另外一家公司在几年前被雇佣来协助进行中的工作。现在, 第三家公司刚刚开始参与进来,以帮助该机构最终实现其原本这重大事业的全部利益。IT组织因为交付缓慢而且不可靠的工作正在被架在火上烤。由于技术应用的滞后能力,银行正在失去客户和商机。高级执行官们从转变中看到的好结果是零散的,相对于时间、金钱和精力的巨大投资不成比例,因此感到沮丧。

银行发现自己为什么陷入当前这种状态的原因是该组织专注于改进单个组件,而不是整个系统。这个系统抑制了他们的努力取得更好效果。如果他们及早认识到这情况,他们本来可以避免大量的痛苦和牺牲——更不用说加速他们寻求结果的过程。

*可在附录中阅读猛犸银行的完整故事。*

戴明的研究表明,当所有的联系协作和相互作用都朝着一个共同的目标共同努力时, 一个组织可以取得伟大的成果。

如果你只是提升了一个团队, 甚至一组团队,你将不会使你的开发人员感到开发软件的过程更加愉悦 (或为客户提供更好的结果)。因此,我们致力于强调如何在组织的所有层中利用 Scrumban, 在4大关键领域对齐理解和努力：

* 领导力之理解市场的需求并阐明一个旨在满足这些需求的愿景；
* 领导力之紧密联系领导愿景和员工共同的理解；
* 员工执行力之将领导愿景转换到行动
* 员工工作效力之保持行动结果与市场需要的一致性

以下分小节展开说明

#### 领导力之理解市场需求并阐明愿景

领导力之理解市场的需求并阐明一个旨在满足这些需求的愿景(图 2.1)。一个公司的领导力如果能更好地理解市场的需求, 那么它就可以更好地定义产品和服务以满足这些需求。

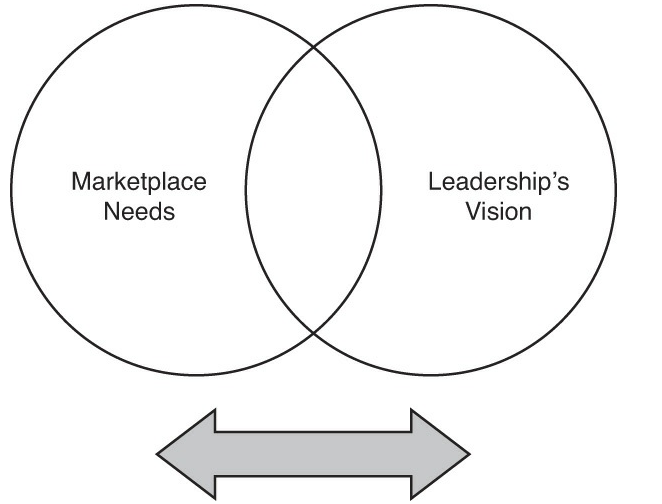


图2.1 使组织愿景与市场的实际需求保持一致, 是迈向持续成功的第一步。

Scrumban 如何提升 scrum 的能力

作为一个敏捷的框架, Scrum 通过采用正式机制(比如冲刺评审和发布计划事件)来创建短反馈循环,帮助产品负责人和IT组织更好地了解市场。这些机制旨在促进在IT应用的最终用户和开发者之间的更好合作。

但是在规定这些基本的角色和过程之外,对于组织如何实现其最重要的目标之一——提高对市场的理解,Scrum是沉默的。

然而,利用Scrumban,组织领导者可以利用框架的面向服务和可持续性的能力来提升他们对市场的理解。例如,企业领导者可以学习应用“发现看板”的原则和实践,以帮助持续地把选项和风险感知到并可视化出来。在同时执行和优化其行之有效的商业模式时存在巨大挑战,发现看板帮助企业识别并解决这些挑战,同时也为新商机来探索更多选项[[[7]](#footnote-7)]。

#### 领导力之紧密联系领导愿景和员工共同的理解

领导力之把领导愿景和员工共同的理解有效地紧密联系在一起,这共同的理解是关于员工的工作如何与实现愿景相关联(见图2.2)。组织的产品和服务最终由系统的系统产生。如果其员工不理解也不欣赏他们的工作与实现公司的市场愿景紧密有关,那么无论领导阶层如何能理解市场的需求,组织都将蒙受损失。

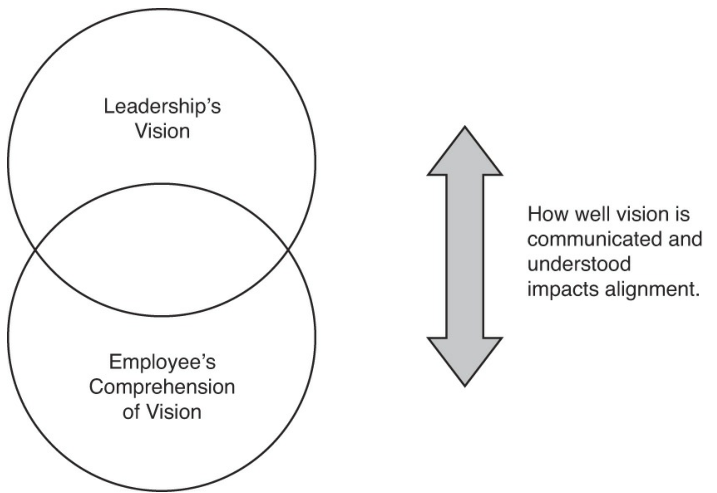


图2-2 确保员工对组织愿景的理解与预期传达的是一致的,这是一个关键对齐领域

Scrumban 如何提升 Scrum 的能力

Scrum 依赖于定义的角色和协作过程来帮助实现这种对齐。由于其规定机制是以团队为中心的,基本框架没有解决如何在整个系统中跨多个团队实现战略协调。

在 Scrum 的防御中,Scrumban 并没有制定具体的策略来实现。但是,它提供了一个集成以下工具和技术的框架,它促进了更广泛的对齐：

* 拓宽深化共享理解的可视化技术
* 强调利用经验数据和科学方法做出更好的决策
* 减少或消除自然、人类心理反应所造成障碍的技术

更有意义的是,这些和其他额外功能可以扩展到软件开发过程之外。对于提高整体的一致性来说,拥有这样一个框架是至关重要的：这个框架是可以用来改进共同的理解,并且改进跨多个业务功能的协作。

#### 员工执行力之将领导愿景转换到行动

员工执行力之能够有效地将对于领导愿景的理解转换到行动 (比如产品、服务),如图2.3 所示。正如领导必须能够有效地在整个系统中传达其愿景,因此员工必须能够将他们对这个愿景和市场的理解转化为行动,使系统能够生产出与他们理解相一致的产品和服务。

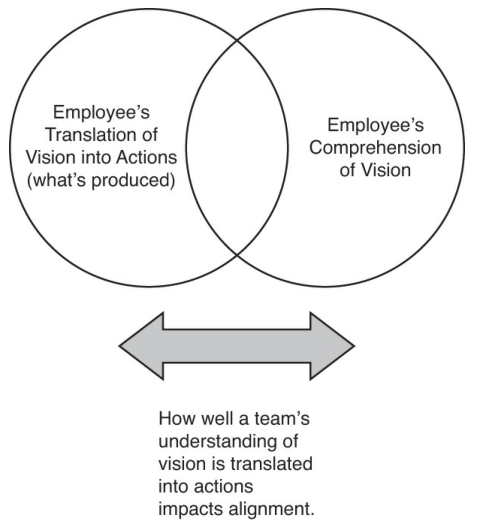


图2-3使员工能够以有效的方式工作, 这个方式使他们能够采取与他们的理解一致的行动,这对成功实现组织目标至关重要

Scrumban 如何提升 Scrum 的能力

在这一领域,Scrum 和 Scrumban 都力求通过提高质量和吞吐量来最大限度地实现的一致性。Scrumban 通过提供产品/服务交付的替代视图来增强 Scrum 的固有能力(例如, 通过使团队更容易地分析像前置时间那样的量化指标, 而不是依赖于像速率一样的主观度量)。该框架还展现了其他方法,以最大限度地提高投资回报率 (ROI),例如将高级优先级排程和自适应风险管理能力集成到日常工作流决策中。

#### 员工工作效力之保持行动结果与市场需要的一致性

见图2-4。本组织的产品和服务必须满足市场的需求。在这些领域上, 总的对齐一致不仅仅是为了取得成功,而是更好地对齐一致意味着更大的能力来保持运营和加油持续增长。

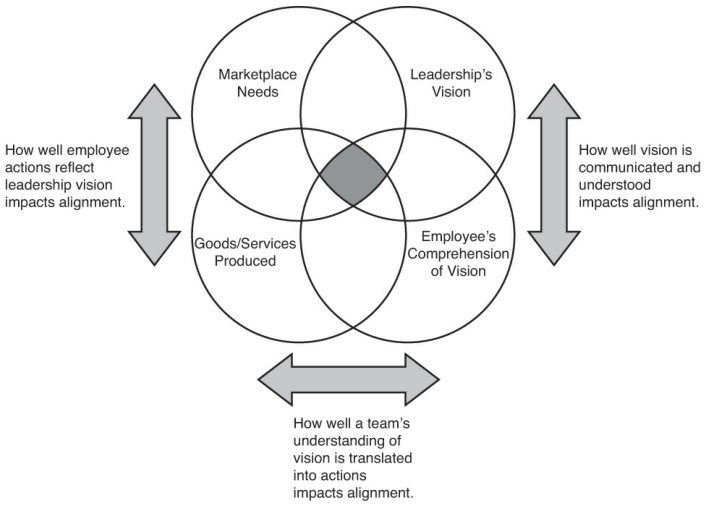


图2-4我们越是成功地在这些领域对齐一致, 我们就越能实现共同的成功。

FIGURE 2.4 The more we are successful in aligning these arenas, the more we

enable ourselves to achieve mutual success.

组织内的每一个人和团队都有助于最大限度地实现组织的一致性。这就是为什么系统思维是相关的, 为什么它在发现过程的起点线上就能找到的。Scrumban 帮助我们发现正确的问题, 分析正确的信息, 以及正确的解决方案。

### 2.1.6 在我们工作环境中的系统

概念化系统是一回事；识别它们是另一回事。特别是当我们身处其中时,识别关键系统是特别有挑战性的。第一个障碍来自于倾向于聚焦在由可见的角色和功能所定义的系统。然而,这些系统只是冰山的一角。有许多不同种类的系统要考虑。

操作提示

识别并可视化核心系统是有效使用 Scrumban 的第一步和最重要的步骤之一。尽管 Scrum 的基础也部分地依赖系统思考原则,但这个框架并不侧重于帮助我们理解它们的组件的系统性相互作用—这是区别于纯粹局部输出的系统性提升的必要步骤。

**正式系统**

* 通常是最明显的系统。例子有流程的文档、组织结构图 (通常是分层的),以及供应商和客户 (通常被视为终端节点)。
* 正规系统往往反映管理结构,而不是工作结构。

**非正式系统**

* 例子有执行实际工作的 "影子" 组织、跨组织界限的网络、在正式定义角色之外的协同工作以及 跨职能/自我导向团队。
* 这些系统通常没有正式记载, 但通常在工作场所内被认可。

**无形系统**

* 例子包括一个组织的知识和学问,它设置了限制,并反映了人们真正的立场。
* 这些系统通常被传统的管理框架所忽略,但他们向员工保证：员工的努力受到重视并与本组织的最终成功挂钩。
* 也许更好或也许更糟,管理层对这类系统的影响是最直接的。

#### **Scrumban故事：猛犸银行和西门子健康**

像任何大型公司一样, 猛犸银行充斥着有形的系统,当然它也有隐藏系统。例如,敏捷教练向来自第三方供应商的一个无形管理层汇报。你认为这个事情如何影响结局？

相似地,就算原有文化系统的外在表现被强行改变了,原有文化系统的影响也不会消失。例如, 在安静工作区中工作多年的高绩效员工 (在某些情况下, 数十年) 在被推进共享工作区时不会自动成为有价值的合作者。

最终抑制超大敏捷转型过程的投资成果的是这家银行在识别和管理所有这些系统性影响方面的失败。而我能与他们的团队取得重大收获的一个关键因素是投入时间来发现和映射这些影响。这方法可以与西门子医疗服务领导层所采取的方法进行对比。在 2005年,这个组织选择采纳敏捷实践——具体来说,Scrum/XP。到了2011年, 他们的团队已经达到了高水平的成熟度,但他们在项目群和投资组合层面上继续艰苦地应对诸多挑战。

领导层决定将看板方法引入到这个环境中。他们没有在组织的现有系统上试点,而是选择了“理想化设计”方法 (本质上假定组织完全被毁, 并在其理想的基础上实施新的设计)。当我们深入研究系统思考时,我将更详细地探讨这个策略。

在附录中可以读到猛犸银行和西门子医疗服务的完整故事。

### 2.1.7 Scrumban 理解系统的方法

从戴明的工作中我们知道,不把组织看作系统的人往往会带来复杂性和问题。另一方面,我们也知道,减少复杂性最终会增加事半功倍的杠杆,产生更多的突破,并导致更容易解决问题。Scrumban 结合了与戴明的深厚知识体系相关的一些核心原则,支持我们真正理解我们工作的系统：

* **每个系统都有变化。**需要创建最优系统的所有信息都是未知的和不可知的。在不同情况下,期望系统都有完全相同功能是愚蠢的。
* **赞成科学方法。**我们可以发现许多关于系统的有用的东西,这些是不被立刻知道的。使用科学的方法使我们能够更快地发现这些东西。
* **只有稳定的系统才能得到改进。**在工作中观察系统使我们能够检测并隔离其内置的缺陷。我们必须在改进之前稳定系统,但是我们必须依靠动态的相互作用而不是静止的观测。我们可以使用 Scrumban 的做法,首先稳定我们的系统, 然后改进它。
* **干预是适得其反的。**复杂性和熵 (倾向进入趋向混乱) 可以通过去除固有瑕疵减少。管理可能有这样一些误导：绕开缺失的资源、过多调节、强加非线性工作流等,因此管理往往成为了增加复杂性的主要根源。

实践提示

尽管 Scrumban 的核心原则和实践可以帮助我们发现系统及其属性,但这些功能并不能免除实践者的责任, 经过刻意的努力才能做到系统思考!

## 2.1 第二部分：混乱(需要Scrumban的理由)

虽然驱使团队和组织改变的力量通常是独一无二的, 但是他们在寻找新的工作方式方面有着共同的动机。你对这些动机有熟悉多少个？

Although the forces driving them are typically unique, teams and organizations share

common motivations in seeking out new ways of working. How many of these are

familiar to you?

### 2.1.1 为什么我们要 "敏捷"

学习水平:

本章所涵盖的基本概念与所有学习者相关。

让我们面对这个问题--没有人仅仅为了敏捷而追求敏捷。相反, 成为敏捷的动力是为了解决一个根本的需求,或者一个已经引起了某些人关注的问题。

在我所曾经工作过的公司中有一个相同的顾虑,就是在员工之间对于他们的组织为什么要追求“敏捷”缺乏一个共同的理解。这理解缺口解释了为什么如下的审查是有用的：对于公司和IT组织为什么投入力量来采用像Scrum那样的敏捷框架,审查其更本质的、出于业务的理由。这些理由在以下各节中加以探讨。

#### 更快的上市时间

房产的商业方面讨厌长长的交货周期,这拖延价值的实现,并可能导致生产不完整充足的产品。敏捷实践强调的,短交付周期和周期性发布的承诺非常诱人。

Scrumban如何提升Scrum的能力？

作为一种敏捷软件开发方法,Scrum 擅长于帮助团队快速地将应用程序推向市场。在这本书中,我将重点介绍 Scrumban 的精益和系统思考基础是如何添加新的能力来加快速度,这速度体现在我们为我们的客户提供正确的产品和服务。一个简单的例子就是这框架如何帮助识别和度量浪费,例如工作流中的过多等待(不工作的)。

#### 更早的投资回报

可工作软件尽早并且增量的交付能给业务带来巨大的区别。最明显的,能够向客户和最终用户显示应用程序的工作版本有利于推进销售。

Scrumban如何提升Scrum的能力？

Scrum 擅长于帮助组织在开发过程早期就取得积极的投资回报。Scrumban 有能力紧密整合各种实践来帮助组织识别并追求正确的投资回报；Scrumban还有能力更好地管理与工作相关的风险, 这些能力将使得已经巨大的收益更加巨大。一个具体的例子是 Scrumban 用于根据已定义的风险刻画来识别和度量工作的机制。

#### 构建正确的产品

即使你正在构建客户所说需要的应用程序,引入敏捷实践也可以帮助确保你的产品以客户实际使用的方式来进行构建。短交付周期让客户体验一个发展中的产品,让客户对它响应反馈,在整个开发过程中可以沟通进行必要的调整。

——————--

Even if you’re building applications that your customers say they want, Agile practices

help ensure you build products in such a way that your customers will actually use them.

Short delivery cycles let customers experience an emerging product, respond to it, and

communicate needed adjustments throughout the development process.

Scrumban如何提升Scrum的能力？

这是另外一个Scrum闪耀的舞台,但是成果回报通常依赖于实践的成熟程度——这是许多组织陷于挣扎努力的地方。使用Scrumban的组织会发现在整个产品和服务生命周期里面集成客户反馈更加容易。例如,Scrumban框架能够在开发开始之前帮助组织实质性地影响客户的期望,而不是在产品使用后根据反馈而放弃它们。我最近在ScrumDo深入利用了这个能力。

像大多数软件即服务提供者一样,ScrumDo持续利用客户反馈来沟通未来产品开发决策并识别优先级。与其它许多公司不同,它积极地通过双向沟通来塑造客户需求。根据我的经验,产品开发团队可以快速地根据开发团队交付能力来判断产品增强要求的紧急程度。这触发了与客户的直接沟通,这反过来又让客户提升了他们对自己实际需要的理解。因此,ScrumDo可以及早地修改增强需求,从而在实际客户需求和ScrumDo响应这些需求的能力之间取得更好的平衡——为所有参与方得到一个更好的成果。

#### 更好的质量

在软件开发项目中有4个主要维度：时间,成本,范围和质量。传统上,企业在时间、成本和范围上设置了严格的限制。因此, 除非企业愿意妥协其既定的限制,否则你可以调整的唯一维度就是质量。

Scrumban 如何提升 Scrum 的能力

如Scrum那样,敏捷方法对时间、成本和质量设定了限制,而将范围作为可变维度。这是有效的,因为帕累托原则告诉我们80% 的产品价值来自与其20%的功能,一旦实现了关键的20%,其它粉刷渲染性的范围几乎无关紧要。

Scrum 为管理这一目标提供了坚实的框架,但其总体效果往往依赖于组织内限定数量人员的个人能力。此外, 该框架要求在每个冲刺 (sprint) 开始时对实现某些功能作出明确承诺。在高度动态的市场中,承诺在交付点两周前就完成工作, 也可能为时过早。Scrumban 强调使用科学方法在价值流的每个决策点上不断地排定工作优先级, 使组织在可视化管理框架内整合信息和度量,而这些信息和度量有助于得出正确的决策。

#### 更好的可预见性

不幸的是, 许多 IT组织都面临着艰难的挑战,给业务部门解答如下问题：“钱花出去能够得到什么回报？”,“什么时候会看到它的好处？”。这导致了业务部门客户更加关心团队交付工作的预测性如何,而不是他们如何快速工作。可是从重要方面而言,可预见性不应该是头等大事。

Scrumban如何提升Scrum的能力？

敏捷框架旨在帮助团队自信的给出承诺并且稳妥的达成承诺。尽管如Scrum这样的方法论使团队能够更好地预测自己的工作,但是一个团队的产出在很少情况下由单独团队完成。此外,像团队估算之类的传统 Scrum 技术在组织中没法有效地规模化扩展。

Scrumban增加了能力,这些能力不仅可以帮助团队提高他们的估算精确度,而且使组织更容易地在多个团队和项目群中进行可靠的预测和协调。例如, 在Scrum框架内的多层虚拟看板系统可以实现诸如提前时间度量之类的客观绩效指标。这些指标与故事点不同,故事点在技术上不是Scrum框架的一部分,在实践中得到了广泛的使用,不同点在于这些数字的含义在不同团队并不变化。基本的排队论系统数学方法提供了概率预测法,这是通常比纯粹的基于估算的预测更可靠。

### 2.1.2 为什么有些敏捷导入停滞或者失败

尤其是 经理和高管

不管组织寻求导入敏捷实践的具体原因,为什么有些成功地成为精益、高绩效组织,而另外一些组织却得到了好坏混杂的结果？

Scrum 和看板框架都呈现了走向敏捷的路径。两者都容易得不到有力的落实执行。特别是,两个框架都容易受到一些实践者的批评,这些实践者过于专注于效率提升,而忽视了诸如系统思维和组织文化之类的关键组成部分[[[8]](#footnote-8)]。这扼杀创造力、创新和可持续性——而所有这些都是本组织的重要成果。

Scrum 实际上很擅长揭露系统性问题和功能障碍。然而, 它不是设计来帮助个人和组织理解并且管理潜伏在他们背后的系统性互动。这不是一个缺点,一个螺丝刀不能把两块板钉在一起,并不是意味着螺丝刀是一个无效的工具。

Scrumban 只是敏捷工具箱中的另一个工具。然而,它可能更贴切地形容为一把瑞士军刀。它可以让你用来加速Scrum 实践的成熟。你还可以使用它作为“拐杖”,以满足一个基础的Scrum 目的；否则,会因组织、团队或个人的不足而无法得到满足。它甚至可以影响完全在组织的 IT/软件开发领域之外的工作流程。让我们考虑其中的一些内容。

#### 将敏捷看作一个过程, 而不仅仅是作为一种心态

向敏捷方法论转型不是一个线性过程。角色不同, 价值观不同, 成功的转型需要在文化和心态上进行变化,这与在过程上的变化一样多。

达成敏捷性没有单一的配方,但许多组织都试图如Scrum原样那样去实施Scrum。戴明和其他跟随他的系统思想家早就理解一个基本真理：不同的系统引入同样的过程会产生不同的结果,同样系统在不同的时间点应用同样的过程也产生不同的结果。发生这种不同是因为没有两个系统是相同的,而单个系统总是随着时间的推移而变化。

虽然有些益处可以从遵循 "配方" 的方向得来,但这些益处的程度是由当地因素所决定的。Scrum 是灵活的,但它并没有解释如何在个人和商业文化没有准备好接受的情况下来实施变革。这就是 Scrumban 可以发挥重要作用的地方。

如果基本意图是提高组织响应市场需求的能力,那么我们需要使该组织的员工有能力了解他们工作的系统, 并使他们能够发现和决策在给定上下文环境中的最佳方法。考虑当前管理员工试图达到这个目标的方法频谱。

这方法频谱中的一个极端情况也许是最普通的方法-自上而下推进。当一个组织接受了 自上而下推进管理的方法时,它的领导者决定他们的公司需要“变得敏捷”,他们规定了实现它的方法 (例如,采用 scrum), 然后他们期望每个人都在这初始动议背后排队。

这方法频谱中另外一个极端是像Dan Mezick提出的“开放式敏捷导入”之类的方法。这个框架包含了如下一些概念：邀请 (参与是可选的) ,实验,游戏力学,通行仪式和其他。许多这些概念都让人联想到丰田的质量圈。彼得.德鲁克可能会把这种方法看作是一种鼓励 “人员活力”的方法,因为它提供了一种方法来管理个人不再保持转型之前的方法和状态,但没有完全转换到他们的新工作方式的时间段。

Mezick 的方法包含了许多与全员共治(holacracy) 相关的概念,全员共治(holacracy)是一个基于公司需要完成的任务而不是标准汇报结构的管理结构。在这个组织治理的社会系统中,权威和决策过程是分布在各个自组织团队,而不是归属于层级结构的顶端。开放式敏捷导入的目的是提供一个这样定义的过程—在其中任何团队成员都可以采取主动,并公开表达他们的关注或想法[[[9]](#footnote-9)]。

Scrumban 如何提升 Scrum 的能力

scrum 无助于我们决定哪一种方法。事实上, "开放式敏捷导入" (或 holacratic) 方法的关键方面在一些组织中可能会导致产生文化冲击,这样的文化冲击同样出现在从上向下推进的其它组织里。

Scrumban 允许我们发现给定上下文环境下的最佳方法。执行其核心做法和原则并不需要重大的变革, 而这框架作为一种手段, 通过这种手段可以按舒适和可持续的步调进行变革。

#### Scrumban故事：猛犸银行

猛犸银行一直与一家大型商业咨询公司保持合作。"古鲁" 与高级管理人员合作,为敏捷实践定义一个定制化框架。他们本质上规定了 “类 scrum” 实施框架,尽管在团队层面上缺乏许多基本特征。

当教练们开始面对实施的实际情况时,清楚地看到转型到这个 “定制化框架”将遭遇许多问题。然而,公司和其收费昂贵的顾问们都不愿意采纳额外的框架(比如看板)来帮助他们的实施。

我开始使用Scrumban作为指导新实践者在Scrum实践上的工具, 虽然我没有给它一个标签。就其他人而言,他们在做 Scrum。使用 Scrumban 的团队的表现大大超过了所有其他团队的表现。

即使像 "标签" 这样简单的东西既可以增强能力,也可以带来限制。

可在附录中阅读猛犸银行的完整故事。

#### 缺少承诺或者缺少努力

敏捷经常是被推销为解决许多问题的灵丹妙药。敏捷过程可以改善许多事情,但它们不是魔法,也不能消除组织机能障碍。

归根结底,工作是人的追求,改善任何人的追求所需要的不仅仅是期望。也就是说, 它需要刻意的承诺,使期望成为现实。它要求经历关于打破旧习惯并且建立全新习惯的艰难抗争。它还要求愿意寻求和接受外界的观点作为指南。

丰田在20世纪中期处于破产的边缘；而到本世纪末,它是世界上最成功的汽车制造商。这不可思议的转折不是简单地过程改进的结果；相反, 它是组织中每一层思维和行为变化的直接结果。丰田拥抱了新的价值观——包括对于持续改进的严肃承诺。它采取刻意的步骤来把这些价值观培育到每个员工,并特意努力更好地对齐公司和员工的目标。它强调长期的关系、双向沟通、仆人式领导和问题解决的规范方法。这些因素是其扭转背后的根本力量,它们是在组织每一个层面上作出高度认真的承诺和努力的直接结果。如果你的目标是在你的组织中创造可持续的高绩效,那么就可以从丰田走过的路中吸取经验教训。

Scrumban如何提升Scrum的能力

Scrum和Scrumban都有助于建立承诺和投入努力。二者之间的主要区别在于 Scrumban 更加看重渐进的、进化的变化,并保持与使之发生所必需的人的因素相协调。组织可以使用这种方法成为更好的 Scrum 实践者,或者发现并实施自己独特的实践做法。而另外一方面,Scrum 为组织工作过程的许多方面规定了最终的形式。Scrumban 侧重于这些形式的根本目的,并帮助组织在上下文环境需要时发现它们如何以其它方式来实现满足。

#### 屈服于幻灭

在敏捷过程中投入时间、金钱和精力之后,当公司看到他们的生产率下降时,或者当他们没有得到按预期的改进时,才会使人开始感到幻灭。虽然许多人拥有惊人的韧性(这个因素是通配符),但是随着连续令人不满的经历,它会像水库放水一样的下降。

Scrumban 如何提升 scrum 的能力

scrum 是一个强有力的框架,但是当它的一些仪式没有得到恰当的采用时,可能会导致幻灭。例如,项目经理担当 Scrum Master的角色并不少见,但继续以原来的方式运作。当发生这种情况时, Scrum 不起作用。相似的,时间箱的冲刺(Sprint)对于许多环境而言可能过于僵化,可能会造成过多的开销。冲刺 (Sprint)计划活动通常会占用整个工程团队半天或更多时间。如果确保跨多个功能的参与和协调是一个问题, 那么 Scrum的那些仪式就一定能改善问题。但如果他们不是？

If ensuring involvement and coordination across multiple functions is an issue, then Scrum’s formalities can definitely improve matters. But what if they’re not?？？？

团队估计是大多数Scrum环境中的一个核心组件,它也会导致问题。花费在试图评估和辩论某一功能或者故事大小的时间可能会适得其反,特别是如果你已经把条目分解成最小的可行颗粒度。糟糕的估算——一个非常常见的问题——实际上会导致问题。当实际工作大于预期时, 冲刺(sprint)就会超载。响应这种意外扩展对于时间和精力两者都需要。

更重要的是, 错误的估计导致不切实际的期望。在这种情况下,回顾通常会变成对估算技术、优先级变化和其他实践要素的批评——令人泄气的结果对信任建设造成负面作用。

生产率也可以在时间箱冲刺 (sprint) 范式中变得 "尖锐",处理和解决这种不一致性的内置工具很少。在冲刺 (sprint) 快要结束时的疯狂时间内, 团队成员为了达成承诺要把所有都完成而陷入苦战, 往往跟随着的是因为团队成员恢复原样而带来的崩溃。

这些例子说明了为什么幻灭呈现了一个失败转型的重要原因。从产生的心理倦怠中恢复是几乎不可能的——尤其是在资源已经非常有限的创业环境中。Scrumban整合了有助于最大限度地减少这些经历的机制。

#### 未能顾及人的因素

Failing to Account for Human Factors

改变对大多数人来说是威胁；破坏性变化尤其如此。对人们施加迅速和/或剧烈的变化是充满危险的,就算仅仅因为它所创造的心理障碍。

也就是说, 并不是所有的破坏性变化都是坏事。在某些情况下, 实际上是需要破坏性变化的。杰克. 韦尔奇在讨论他关于四种类型管理者的哲学时提到了这个概念。

韦尔奇说, 组织面临的挑战是发展和维持一个文化, 既是以价值观为中心, 又由绩效驱动。当领导者们容忍那些虽然交付、但行为表现与公司价值观不符的员工时, 他们最终将破坏自己的议程。

例如, 敏捷的一个核心价值是持续改进。如果你的组织中的一位高绩效的员工不遵循这一承诺, 他或她将最终破坏你为灌输这种价值而做出的任何努力。

Scrumban 如何提升 Scrum 的能力

对于如何最好地解决这些类型的心理障碍,Scrum是沉默的。更重要的是, 它趋向于实施的方式很少说明引入这种变化的含义。相比之下, 这些人的因素是在 Scrumban 的框架内直接管理的。

例如, 在引入Scrumban 时, 工作方式顶多只有少量立即的改变。相反, 框架强调采用诸如可视化的简单机制, 使团队能够更深入地了解其系统, 提升他们共同的理解, 并在发现改进的方法时设置变化的范围和速度。

#### 由外部顾问代理

对于使一个组织更加精干和敏捷的各种各样努力, 其长期成功取决于领导层持续的主人翁式努力, 以及持续重视通过新的过程来推动期望的业务结果。组织领导层往往会无意中把这一责任下放给外人。

我参与了一些公司的敏捷导入, 外部公司被雇佣来帮助他们执行成功采用敏捷方法所必需的战术步骤。通常, 管理层放弃了对这些公司特定领域的"主人翁"式努力—而它应该是保留在管理层控制范围内。具体的例子包括确保员工理解他们的工作如何与预期的业务结果挂钩, 并确保组织最终实现这些结果。任何外部公司都不能指导这类事情。

虽然与这个概念没有直接关系, 但值得注意的是, 过程改进咨询如何成为西方世界的一个下流生意。高级管理者已经开始认为外部顾问比他们自己的领导层更有能力可持续地改进其组织。这很少是真的--特别是在大型组织中。尽管如此, 这种心态也解释了为什么许多顾问在敏捷教练费用上获得了2000美元/天的上涨(2014年)。专家是重要的工具, 组织却需要确保他们没有被要求履行应当在组织内部的职能。

Scrumban故事：西门子健康服务

当西门子试图将 Scrumban 引入其组织时, 它采取了一个完美的方法来聘请外部顾问。它从以往的经验中知道, 要使新的进程可持续, 必须发展内部专长和能力。然而, 它也承认, 如果没有外界的帮助, 它是不可能成功的。西门子公司雇佣了一家公司, 通过起步的培训并结合持续的专家洞察力来帮助他们建立适当的基础。这些专家指导内部教练, 加深他们的理解, 使组织了解像利特尔法则和扩展度量那样的准则。西门子高管欣然承认, 如果没有这种外部专家的帮助, 他们将不会离成功那么接近。

在附录中可以阅读西门子健康医疗的完整故事。

#### 在改进之前没有稳定下来

戴明论证了为什么在试图改进现有系统之前必须稳定它们。Scrum 是一个很好的框架, 但它不是为了促使系统稳定而设计的。它依赖于其规定的团队结构来创建稳定 (例如, 它假定打破现有稳定的团队来构建7到12位成员的Scrum 团队, 然后将最终稳定下来)。此外, 框架并没有真正纳入任何机制来揭示可能何时稳定性会缺乏。因此, 许多Scrum 项目团队认为, 当人员进出团队的时候, 稳定性不会受到影响。而这导致了绩效问题, 框架可能因此受到不公平的指责。

诸如"敏捷健康评估", "敏捷空间" 等类似的服务有许多咨询公司提供, 也有许多公司购买。这类做法通常在没有获得对于要开展引入的系统的任何上下文理解下, 为组织仅仅提供一份行动检查表。使用无感知力的检查表来改进是适得其反的。

Scrumban 自身的能力, 加上那些可以 "插入" 到 Scrumban 系统的其他模型和技术, 帮助识别系统状态和在其中开展管理的最佳方法。我使用的一些外部模型包括: A3思维方式、问题解决规范方法和 Cynefin 框架, 这呈现了复杂性理论的务实应用。

#### 无法识别真正的问题, 反而干扰正常事物

从戴明的工作中得出的另一项原则是, 无法识别真正的问题或者干扰已经工作正常的事物的重要性。管理层的一个常见失误是处理错误的问题, 或者不必要地研究那些根本不是问题的表面感知问题, 这两者都可能导致管理者迫使事情变成预先确定的模子。

Scrumban 如何提升 Scrum 的能力

Scrum 暴露了问题和功能障碍, 但它不是设计来管理纠正它们的过程。相比之下, Scrumban进化变化框架直接为这种纠正过程提供了方法。

例如, 我曾经与一个Scrum 团队合作过, 它没有能力稳妥地完成一个冲刺 (sprint) 内接受的所有工作。前一位教练选择在改进工作分解技术方面上训练团队成员, 以此提高他们满足冲刺 (sprint) 承诺的能力。尽管他的意图令人钦佩, 但他却没有真正了解到底是什么推动了结果。Scrumban的系统理解和工作流可视化的引入帮助团队和他们的新教练发现他们冲刺承诺问题的根源原因。

相似的, 我还与一家大型金融机构合作, 这机构有意构建新的敏捷工作区以支持其新流程。然而, 在进行这一重大物理空间的变化时, 管理层对几十年来工作在物理立方隔间里的人员的心理影响考虑不周全, 对在另外环境中适应并采用这种新方法的障碍也考虑不足。当Scrumban 框架在已经受影响的环境中实施后, 团队就有能力克服了许多障碍。

#### 回顾时的不健康反思

我们之前提到了在解决幻灭问题时回顾聚焦偏差的问题。回顾通常会被不适当的转移关注到个人，而不是系统上。它们还可以导致分开处理改进与系统本身(例如, 改善板的使用，团队担当的改进工作与“普通工作”分开可视化)。Scrumban 通过不断强调整体系统观点和鼓励在统一框架内对所有工作进行可视化和管理，帮助抵消这些倾向。

#### 超过质量的病态偏重速度和结果，

由于过于强调速度而向市场推出劣质产品对许多组织产生了长期负面影响。正确的结果包含适当的质量和时间考虑。Scrumban 启用相应的实践和技术来帮助组织识别和管理适当的平衡。

#### 持续改进成了少数人的责任，或者只依赖少数人

彼得. 德鲁克说, 对于任何领导人最严重的指控是一旦领导人离开或死亡，他或她的组织就会垮台。对于将核心敏捷价值观的所有权委派给少数人的组织来说, 具有等效的净效果。

#### 角色: 好、坏、丑

角色携带很多行李。我们认同我们的角色。我们所做的工作体现在我们的角色中。当新的过程定义不同的角色，或为现有角色定义不同的职能时，我们需要认识到在这个过程中必须克服心理因素。

Scrumban 如何增强 scrum 的能力

因为 scrum 在没有对这个心理变化过程进行说明的情况下规定了角色，那些实施它的人往往忽视角色转变的影响。

人为地坚持现有的角色或新定义的角色通常会妨碍组织有效地演进的能力。Scrumban 原生结构就是为了应对这些挑战。其核心原则之一是尊重现有的角色, 但着眼于鼓励团队和个人以不具威胁性的方式发现自己所需的任何变化。

#### 提前设计式架构与涌现式架构

最后, 有必要介绍另一个 Scrumban 有助于阐明的尖锐领域——我们对软件开发采用了提前设计式与涌现式方法的程度。传统的方法，比如瀑布，在纲领性的计划和设计上面提前决定了多项内容。呈现出来的是一个非常慎重和故意的过程。

对比于大量的计划和文档来，敏捷实践更加倾向于创建可运行的软件，因此大多数实践者认为应当允许架构从完成以客户为中心的小需求的工作过程中涌现出来，相信这样会使得组织更加敏捷。虽然当对系统的理解更加深入时，返工仍然是需要的，但是这个方法有利于规避沉重过度架构设计的倾向(这在以前被认为是有用的)。Scrum 偏向采取这种理念。出现一个笨拙的技术债务症状通常意味着存在一个纲领性要点没有充分聚焦。

但考虑架构的目的：它是用来塑造我们的解决方案，解决可预见的技术风险，并尽量减少变化的影响。解决更大的问题和创建常用的方法 (轻量级架构机制) 是很好的架构目标， 但只有在解决方案确定的情况下才能实现。

由大型团队、重用需求、多重集成和对齐要求以及外部治理要求 (包括其他) 组成的环境可能更适合于一种提前设计式架构方法。对比环境和市场, 如初创公司和小型组织，通常更适合于一种涌现式架构方法。

Scrumban 如何增强 scrum 的能力

在这一天结束时，除非一个组织有一个能够经受住增量开发的框架, 否则它每天都会累积技术债务，这债务会减慢软件交付管道，因而开发人员最终会进行维护而不是新的开发。在我们的案例研究中, 这正是猛犸银行开始发生的事情。

对于如何最好地处理这个问题，Scrum本身保持沉默。Scrumban 帮助组织主动地处理这些开发的方方面面——首先是使工作及其结果更加明显，然后通过使团队和组织能够集成各种机制来主动地把它们作为正在进行中工作的一部分来处理。

## 2.3 整合在一起试试

到这里，我们发现和理解的第一步就结束了。你从本章中吸取的单个最重要的学习内容就是系统思维。尽管系统思维最初在70年前就已经明确表述出来，但对于这个商业世界上大多数人员来说，它仍然是一种新颖的理念。你越早适应这种新的思维方式， 你就能越早地利用本书中详细介绍的原则和实践。

系统思考能够改善理解。更好的理解，反过来，使我们对于寻找什么以及寻找什么方法去提升都有更清晰的认识。 对于寻找什么有更好的认识帮助我们去识别有意义的模式，这原本可能是发现不了的。当我们继续我们的旅程的时候，利用这些知识，成为你的优势。

1. Jeff Sutherland and Ken Schwaber. (July 2013). The Scrum Guide: The Definitive Guide to Scrum, p. 3 [↑](#footnote-ref-1)
2. 来自于两个Scrum管理平台的使用数据支持以上发现(ScrumDo.com and Rally Software) [↑](#footnote-ref-2)
3. 在企业内扩大Scrum的例子参见<http://tiny.cc/JeffSutherland> [↑](#footnote-ref-3)
4. 这种对看板方法的定义已经被它的先驱David.J.Anderson严格地应用了很多年。在2014年秋天，Anderson和其他领导者在看板社区创建了一个“带有看板的项目管理”课程，目的是展示如何使用该框架来服务基本的项目管理功能。 [↑](#footnote-ref-4)
5. Reinhardt, W.; Schmidt, B.; Sloep, P.; Drachsler, H. (2011). “Knowledge Worker Roles and Actions: Results of Two Empirical Studies.” Knowledge and Process Management. [↑](#footnote-ref-5)
6. Dimitar Bakardzhiev has suggested a seemingly valid application of TOC within a similar context. Though the scenario we describe reflects different conditions, his article shows how TOC can be successfully applied to

   many knowledge work problems. http://tiny.cc/DimiterBakLiquidity (June 2013). [↑](#footnote-ref-6)
7. <http://tiny.cc/DeliveryDiscovery> [↑](#footnote-ref-7)
8. 参见示例，http://tiny.cc/ScrumCriticism (July 2014) [↑](#footnote-ref-8)
9. 在书写本书时，最引人注目的采用全员共治结构的组织是Zappos.com(在2013年12月宣布)。 [↑](#footnote-ref-9)