

软件开发过程改进

软件开发过程改进

– from individual to team to organization

Contents

前言 Prologue

We are uncovering better ways of developing software by doing it
and helping others do it.

—Agile Software Development Manifesto

当今 IT 时代，软件开发越来越重要。但如何让软件开发也可以像其他工程领域，可以利用数字化来不断提升呢？

不少开发团队已经开始利用敏捷开发，两到四周迭代去对应客户需求不断变化。团队基于每迭代回顾、复盘其实可以利用数据，帮助进一步提升团队的质量与生产率。

本书主要是针对这点，借用一些实际的方法，场景，案例，让理解如何通过演化的过程，使个人到团队不断改善，包括：

- # 制定方式/步骤/模型
- # 针对问题的根因做试验
- # 评判下一轮迭代的效果
- # 利用数据，团队得到反馈，持续完善过程

这持续完善的过程，以前在工业生产年代已经被证明有效，戴明先生 (Deming)，与 Juran, Shewart 等质量大师已经验证了这种改进方式，配合数据统计分析，可以提升生产线的质量与生产率。

如何利用这方式把作坊式的软件开发逐步变成有序的软件工程？

与工业生产不同，软件开发很多时候都无法像工厂收集到各种数据，但软件工程师仍然可以利用数据反馈做改进。很多软件工程师虽然开发能力很强，但未养成个人收集数据的能力和习惯，也不清楚如何用数据，持续改善。本书针对这些需求，以如何改善软件质量为目标，建议一些具体的做法、工具，帮助个人和团队可以立马尝试走出量化敏捷开发的第一步。

本书主要内容包括：

- 从个人如何提升开始
- 如何作基于个人，演化成团队的提升
- 探索背后的一些管理思路，包括：
 - 德鲁克
 - 敏捷开发与 X-Y 理论

- 针对敏捷开发，如何利用度量与分析，不断提升质量

1.0	2.0	3.0	4.0	5.0
6.0	敏捷开发颜黑			7.0
8.0	9.0	10.0	11.0	12.0

T ₁	颜黑	
	BLACK	BROWN
EYE 颜黑	68	119X 黑

1.0	2.0	3.0
6.0	敏捷颜黑	7.0
8.0		9.0
10.0		11.0
12.0	13.0	14.0

1.0	2.0	3.0	4.0
6.0	敏捷颜黑		7.0
8.0			9.0
10.0			11.0
12.0	13.0	14.0	15.0

Part I

个人经验分享

我深信要提升必须源自个人，如果团队成员本身没有动力，团队无法提升。我虽然差不多天天都与软件开发团队打交道，但自己编码的时间很少，下面是相关的经验分享。

Part II

再读德鲁克与 **X-Y** 理论

探索敏捷开发背后的管理思路。

Part III

量化敏捷开发案例

最后部分的重点是量化敏捷开发，很多管理者都知道量化很重要，但往往因未把握度量与分析的思路，效果适得其反。这部分会先分享量化管理的常见问题，然后介绍一些案例，最后以形式帮助有兴趣的团队可以尽快在冲刺回顾复盘尝试。