

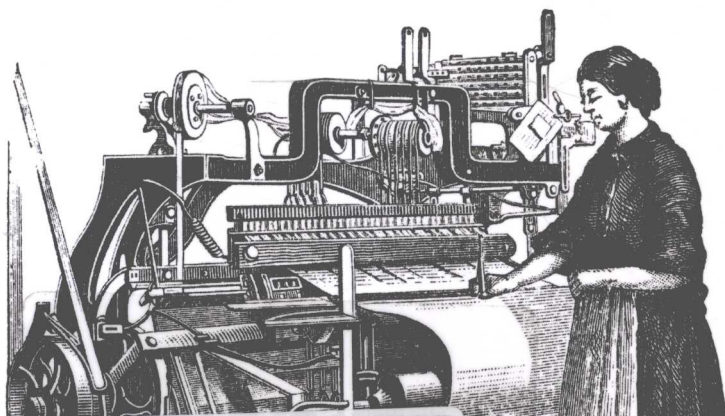
The
Pragmatic
Programmers

Driving Technical Change

Why People on Your Team Don't Act on Good Ideas,
and How to Convince Them They Should

布道之道

引领团队拥抱技术创新



[美] Terrence Ryan 著
李缨 李松峰 译

 人民邮电出版社
POSTS & TELECOM PRESS

TURING

Driving Technical Change

Why People on Your Team Don't Act on Good Ideas,
and How to Convince Them They Should

布道之道

引领团队拥抱技术创新



【美】Terrence Ryan 著
李缨 李松峰 译

人民邮电出版社
北京

图书在版编目 (C I P) 数据

布道之道 : 引领团队拥抱技术创新 / (美) 瑞恩 (Ryan, T.) 著 ; 李纓, 李松峰译. — 北京 : 人民邮电出版社, 2012. 1

书名原文: Driving Technical Change: Why People on Your Team Don't Act on Good Ideas, and How to Convince Them They Should

ISBN 978-7-115-26727-6

I. ①布… II. ①瑞… ②李… ③李… III. ①领导学—通俗读物 IV. ①C933-49

中国版本图书馆CIP数据核字(2011)第226206号

内 容 提 要

本书旨在告诉读者如何说服自己的同事采用新的工具和技术。作者总结了7种怀疑论者模式: 孤陋寡闻型、随波逐流型、百般挑剔型、激情燃尽型、时间紧迫型、发号施令型和不可理喻型。围绕说服这些怀疑论者, 书中介绍了9个实用技巧和4个策略。理解并熟练运用这些技巧和策略, 能够让你的技术布道生涯收获累累硕果。

本书适合IT行业的技术布道师、推广专家、产品经理、需求调研及实施人员阅读, 同样也适合对前沿技术时刻保持浓厚兴趣的设计和开发人员参考。

布道之道: 引领团队拥抱技术创新

-
- ◆ 著 [美] Terrence Ryan
译 李 纓 李松峰
责任编辑 朱 巍
 - ◆ 人民邮电出版社出版发行 北京市崇文区夕照寺街14号
邮编 100061 电子邮件 315@ptpress.com.cn
网址 <http://www.ptpress.com.cn>
北京隆昌伟业印刷有限公司印刷
 - ◆ 开本: 700×1000 1/16
印张: 9.25
字数: 133千字 2012年1月第1版
印数: 1—3 500册 2012年1月北京第1次印刷
著作权合同登记号 图字: 01-2011-6292号
ISBN 978-7-115-26727-6
-

定价: 29.00元

读者服务热线: (010)51095186转604 印装质量热线: (010)67129223

反盗版热线: (010)67171154

版 权 声 明

Copyright © 2010 Pragmatic Programmers, LLC. Original English language edition, entitled *Driving Technical Change: Why People on Your Team Don't Act on Good Ideas, and How to Convince Them They Should*.

Simplified Chinese-language edition copyright © 2012 by Posts & Telecom Press. All rights reserved.

本书中文简体字版由The Pragmatic Programmers, LLC. 授权人民邮电出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

致 谢

没有Dave Thomas和Andy Hunt，就没有这本书。如果不是读过《程序员修炼之道》，我不可能知道自己迫切需求的技术和工具。从看他们写的书到为他们的出版公司写书，我完成了一个自己曾经认为不可能完成的任务。

Jackie Carter是这本书的编辑。她对我的情况可以说是了如指掌，每当我写不下去的时候，她总会及时地出现。她还给我出谋划策，确保了这本书的可读性和容易理解。还要感谢本书的技术审校，感谢他们对本书的润色和修饰：Rachel Davies、Ben Nadel、Karl W. Pfalzer、Craig Riecke、Johanna Rothman和Brian Rinaldi。

我必须感谢沃顿商学院的所有同事。无论是质疑过我的人，还是被我说服的人，他们教会了我很多，特别是在做技术布道的时候，该做什么，不该做什么。其中，特别要感谢Dave Siedell、Bob Zarazowski、Gerry McCartney和Deirdre Woods，感谢他们哪怕是心里并不服气，但仍然宽宏大度，一贯支持。

我还要感谢目前Adobe公司的同事。我们这个组都是做技术推广工作的。我们的第一次内部会议考验的是每个人的音量。我特别想感谢我的顶头上司Kevin Hoyt，感谢他提出的中肯建议。Ryan Stewart则是灵感、斗志和鼓舞的源泉。Adam Lehman对我帮助很大，也是和我斗嘴的好搭档。Ben Forta是我最初的榜样，他让我知道自己必须站出来说服大家去争取更大更好的成绩。

Avish Parashar是我身边的老师，他告诉我怎么讲话、站立以及倾听。他教会了我先去做了再说。这本书就是这种思维方式的产物，如果没有他，就不会有这本书。

老爸、老妈，还有Casey，谢谢你们撑起了这个家，在这个家里，通过学习提升自己是一种高尚的行为。Jack还有Ellie，给你们换尿布、喂奶，哄你们笑，让我觉得这辈子都值得了。

最后，我要把最重量级的感谢送给我的贤妻Janice。你从来都不怀疑我能做任何事。你对我的这种信心感染了我。谢谢你相信我可以做到。如果不是为了你，我还真做不到。

目 录

第一部分 导 言

第 1 章 为什么写这本书	3
1.1 本书组织方式	4
1.2 为什么要看这本书	4
1.3 本书目标读者	5
第 2 章 开宗明义	7
2.1 什么是业内发展成果	7
2.2 怀疑者指的是谁	8
2.3 为什么需要推销	8
第 3 章 解决正确的问题	11
3.1 为什么要布道	13
3.2 考虑方案	13
3.2.1 研究问题	14
3.2.2 摸底调查	14
3.2.3 开列清单	14
3.3 面临的挑战	15
3.4 几点建议	15

第二部分 怀疑者模式

第 4 章 了解身边的人	19
第 5 章 孤陋寡闻型	21
5.1 他们为什么不用这种技术	21
5.2 深层次原因	22
5.3 有效的应对策略	22
5.4 几句忠告	22
第 6 章 随波逐流型	25
6.1 深层次原因	25

6.2	有效的应对策略	26
6.3	几句忠告	27
第 7 章	百般挑剔型	29
7.1	深层次原因	30
7.2	有效的应对策略	32
7.3	几句忠告	33
第 8 章	激情燃尽型	35
8.1	深层次原因	36
8.2	有效的应对策略	36
8.3	几句忠告	37
第 9 章	时间紧迫型	39
9.1	深层次原因	39
9.2	有效的应对策略	40
9.3	几句忠告	41
第 10 章	发号施令型	43
10.1	深层次原因	43
10.2	有效的应对策略	44
10.3	几句忠告	45
第 11 章	不可理喻型	47
11.1	深层次原因	48
11.2	有效的应对策略	48
11.3	几句忠告	49

第三部分 技 巧

第 12 章	装满工具箱	53
第 13 章	取得经验	55
13.1	技巧分析	57
13.2	怎样成为专家	57
13.2.1	研究技术和工具	57
13.2.2	实际使用	58
13.2.3	向现有专家求助	59
13.2.4	教人使用	59
13.3	适用对象	60
13.3.1	孤陋寡闻型	60
13.3.2	随波逐流型	60

13.3.3	百般挑剔型	60
13.3.4	激情燃尽型	61
13.4	陷阱	61
13.4.1	强迫别人	61
13.4.2	傲慢或霸道	62
13.5	小结	63
第 14 章	传达理念	65
14.1	技巧分析	66
14.2	掌握表达的艺术	67
14.2.1	要做人, 别做计算机	67
14.2.2	要有激情, 但不能激进	67
14.2.3	要提建议, 而不是申饬	68
14.2.4	要多听, 而不是多说	68
14.2.5	保持积极的心态	69
14.3	适用对象	69
14.3.1	孤陋寡闻型	69
14.3.2	百般挑剔型	69
14.3.3	不可理喻型	70
14.4	陷阱	70
14.5	小结	70
第 15 章	展示技术	71
15.1	技巧分析	72
15.2	把握展示时机	72
15.2.1	等待机会	73
15.2.2	创造机会	73
15.2.3	培养机会	74
15.2.4	代码评审	74
15.3	适用对象	74
15.3.1	孤陋寡闻型	74
15.3.2	百般挑剔型	75
15.3.3	时间紧迫型	75
15.3.4	不可理喻型	75
15.4	陷阱	75
15.5	小结	76
第 16 章	适当妥协	77
16.1	技巧分析	78
16.2	找到折中方案	78
16.2.1	找到条件成熟的规定	79

16.2.2	找到与规定匹配的技术	79
16.3	适用对象	80
16.3.1	时间紧迫型	80
16.3.2	发号施令型	80
16.4	陷阱	81
16.5	小结	81
第 17 章	建立信任	83
17.1	技巧分析	84
17.2	如何建立信任	85
17.2.1	不要故意撒谎	85
17.2.2	不要回避事实	85
17.2.3	永远不要制造 FUD	86
17.2.4	承认错误	86
17.3	适用对象	87
17.3.1	激情燃尽型	87
17.3.2	百般挑剔型	87
17.3.3	不可理喻型	87
17.4	陷阱	88
17.5	小结	88
第 18 章	公之于众	89
18.1	技巧分析	90
18.2	让自己成为焦点	91
18.2.1	开源你的工作	91
18.2.2	参加竞赛	92
18.2.3	为得奖而设计	92
18.2.4	让人评审你的项目	92
18.3	适用对象	93
18.3.1	孤陋寡闻型	93
18.3.2	百般挑剔型	93
18.3.3	激情燃尽型	93
18.3.4	发号施令型	94
18.4	陷阱	94
18.5	小结	94
第 19 章	注重合力	95
19.1	技巧分析	96
19.2	构造合力	96
19.3	适用对象	96
19.3.1	时间紧迫型	96

19.3.2	发号施令型	97
19.4	陷阱	97
19.5	小结	97
第 20 章	搭一座桥	99
20.1	技巧分析	101
20.2	搭一座桥	101
20.2.1	情况调查	101
20.2.2	找一座桥	102
20.2.3	搭一座桥	102
20.3	适用对象	102
20.3.1	随波逐流型	102
20.3.2	百般挑剔型	102
20.3.3	激情燃尽型	103
20.3.4	时间紧迫型	103
20.4	陷阱	103
20.5	小结	104
第 21 章	来点刺激	105
21.1	技巧分析	106
21.2	给大家提一提神吧	106
21.3	适用对象	107
21.3.1	孤陋寡闻型	107
21.3.2	随波逐流型	107
21.3.3	时间紧迫型	108
21.3.4	百般挑剔型	108
21.3.5	激情燃尽型	108
21.4	陷阱	108
21.5	小结	109

第四部分 策 略

第 22 章	简单，但不容易	113
第 23 章	无视敌人	115
23.1	怎么无视敌人	115
23.2	为什么不好处理	116
第 24 章	先易后难	117
24.1	难度分组	117
24.2	容易	118

24.2.1	孤陋寡闻型	118
24.2.2	随波逐流型	118
24.3	难	119
24.3.1	激情燃尽型	119
24.3.2	时间紧迫型	119
24.3.3	百般挑剔型	120
24.4	最难	120
第 25 章	借力支持者	123
25.1	请求帮助	123
25.2	创造布道者	124
25.3	交叉推广	125
25.4	消耗关注	126
第 26 章	说服管理层	127
26.1	希望管理层做什么	127
26.2	怎么做到	127
26.2.1	解决管理问题	128
26.2.2	用数字说话	128
26.2.3	解释为什么需要强制执行	128
26.3	接下来怎么办	129
第 27 章	最后的话	131
27.1	经验教训	131
27.1.1	成功过度	131
27.1.2	莫求回报	132
27.1.3	你可能错了	133
27.2	成功之道	134
27.3	问题会扩散	134
27.4	只有过程，没有终点	135
参考文献		136

Part 1

第一部分

导 言

本 部 分 内 容

- 第 1 章 为什么写这本书
- 第 2 章 开宗明义
- 第 3 章 解决正确的问题

第 1 章

为什么写这本书

我曾经在一家非常有代表性的公司上班，时不时和一群非常有代表性的人在一起，开那么一次非常有代表性的会议。我掌管着公司的Web应用服务器。工作职责包括维护软硬件、推行最佳实践，说服人们不断做技术升级。我一直都忙着想办法让人们升级。

事实上，会议的主题就是升级。

“我想让你们给我一个从ColdFusion 6迁移到ColdFusion 8的时间表。”我说。这已经是在类似会议上第五次给他们下最后通牒了。

不出所料，有人叹了口气说：“我们不能迁移到新服务器。每次把应用迁移到新服务器都会遇到麻烦和不兼容的问题。我们就是不想再给用户添麻烦了。”

我还击道：“这个问题很正常。你们想过在切换版本的过程中，使用单元测试来验证新版本的优势吗？并不是只有应用服务器存在这个问题，Web服务器、数据库服务器，还有你们的基础代码都存在这个问题。版本升级是谁也逃避不了的。你们可以使用单元测试嘛。”

然后，各种借口都涌过来了，“升级会占用太多时间。”“为什么这件事非做不可呢？”“我们不知道怎么做单元测试。”

我可以告诉你，我是在跟他们辩论。我还可以把辩论的情形描述得更详细一些。但那又有什么用呢？结果是明摆着的，我没有把他们争取过来，我始终无法说服他们。

在从事那项工作的时间里，我得想方设法地说服人们在某方面加以改进

或采取某种新技术。刚才描述的争论场面以及类似的情景，我经历了很多很多次了。我输掉了大部分争论，也赢了一些，有些争论到最后终于两败俱伤。不过，我倒是总结出了一些模式：

- 同一类人所持观点相同；
- 有些人始终都乐于接受新事物；
- 另一些人则在别人转变之后才肯改进；
- 有些人你永远也说服不了；
- 某些观点对有些人有用对另一些人没用；
- 有时候让管理层插手是让人们服从的唯一方法。

总结出以上模式之后，我又据以做了很多笔记，明白了某些策略只对部分人有用，永远不能一概而论。然后，我基于不同问题针对同一批怀疑者采取了相同的战术。成功率节节攀升，说服人们有所改进变得简单了。

本书的主要内容就是这些改进、模式，以及争论。但愿我的经验能成为读者的前车之鉴，让大家能够避免漫无目的的争吵，减少挫折感，真正推动组织的技术进步。

1.1 本书组织方式

这是一本讲模式的书。换句话说，本书的主题将围绕一系列重复的形式，也就是模式展开。本书两个主要部分各自包含一组模式，其中一部分模式适用于怀疑者，着重分析某些人抵制的原因。另一部分主要谈方法论，包含一套可以用来反击怀疑者的方法和技巧。如此一来，本书这两部分中各章的结构自然也就极为明晰了。

除了讨论你的同事以及如何对付他们的有关模式的这两部分之外，本书最后一部分将把焦点从模式转移到策略。策略能排序出要先说服谁，不要招惹谁，以及如何真正推动技术进步。

1.2 为什么要看这本书

本书的目标是使你能够说服自己的同事采用新工具和新技术。为了达到

这个目的，你不必像政客一样在办公室里玩阴谋诡计。当然，我可不是说不需要讲策略，只不过用不着那么邪恶罢了。

这本书要牵扯到一大批人物，其中一些人会让你觉得似曾相识——哎，这不就是那谁吗！好，只要你能让现实当中的某个人与书中的人物对号入座，就可以在本书中找到对付他们的方法。对那些怀疑者使用这些方法应该讲究策略，这些本书都会向你一一道来。然后，变化就会魔法般地发生了。

好吧，哪能靠什么魔法啊！你还得靠自己的工作和努力。只不过现在你的目标明确了。看完这本书，相信你一定会有相见恨晚的感觉。而只要你依照本书的策略行事，随着经验的增长，更多的好处还在后面等着你呢。

1.3 本书目标读者

本书的目标读者是搞技术的人，比如开发人员或者程序员。具体来说，可以是服务器管理员、网络工程师或硬件工程师，抑或数据库管理员，甚至是与技术人员协同工作的设计师。

至于你到底搞什么技术，其实并不重要。只要你掌握某种技术，需要与别人沟通，这本书都适合你看。

本书要讲的故事和描述的情景主要涉及开发人员。非常抱歉，我本人就是一名开发人员。但实际上，无论你使用什么工具、什么语言，都没有问题。说到底，不管你是.NET程序员、搞Java开发的，还是开源的拥趸，或者对某家公司的技术情有独钟，都不影响你看这本书。这本书要讲的就是怎么让你的同事改变他们的工作方式，至于改变成哪种方式，那是你自己的事。

