

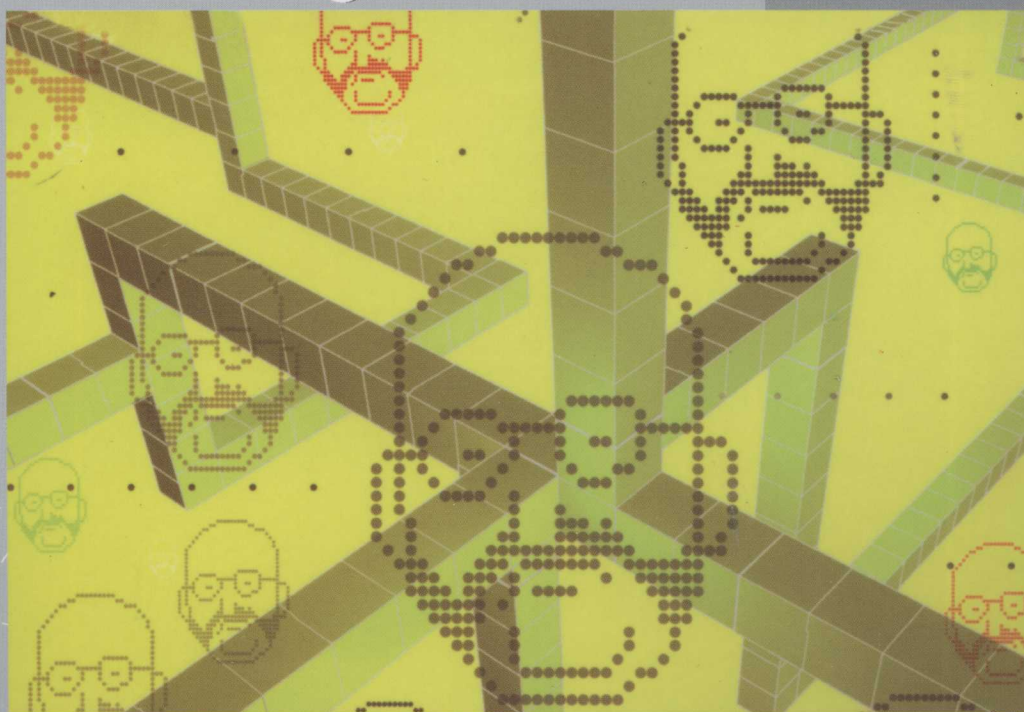


Microsoft®
Press

Microsoft
核心技术丛书

代码之道

I. M. Wright's "Hard Code"



(美) Eric Brechner 著
陆其明 译

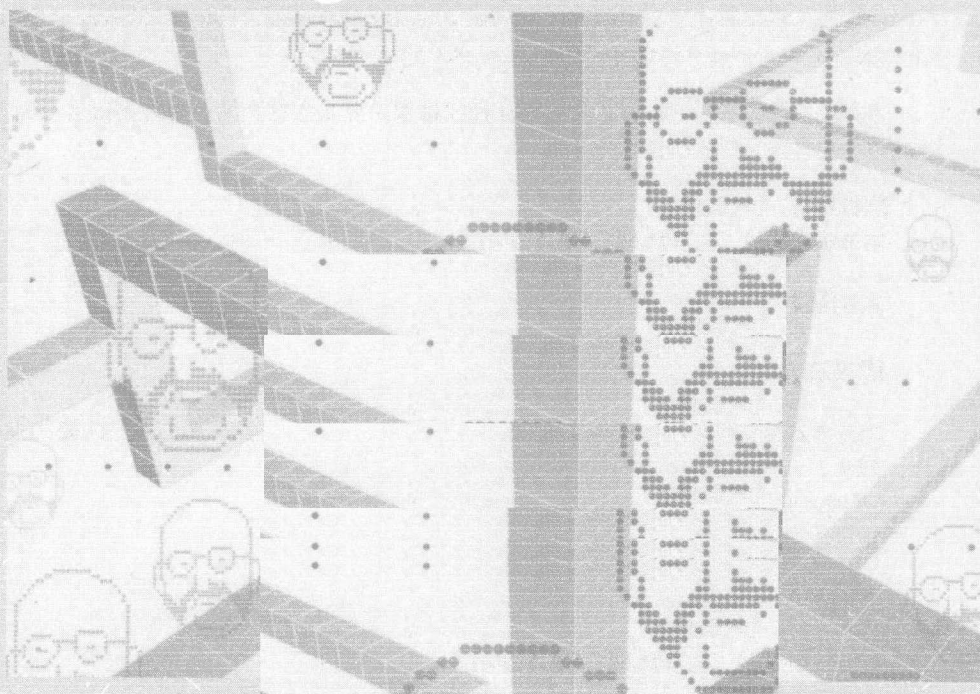


机械工业出版社
China Machine Press

《代码大全》姊妹篇

代码之道

I. M. Wright's "Hard Code"



(美) Eric Brechner 著
陆其明 译



机械工业出版社
China Machine Press

本书以一位微软内部人士的视角，揭示了关于软件编码、软件测试和项目管理中的方方面面问题。作者文笔犀利，见解独到，对软件行业内的很多常见问题提出了解决方案，并提供了最佳实践。本书详细介绍怎样提高软件的质量和值；切合实际地管理项目的时间表、风险和规范书；为常见的低效率开发瘦身；应用过程改进方法，避免固执盲从；规划一个成功的、令人满意的职业生涯；发展并管理一个欣欣向荣的团队等。

本书是作者对过去在软件行业 6 个不同的公司、28 年的工作经验的一次总结。

本书不仅是微软内部员工的必读之书，也同样适合于软件行业内其他所有工程师和管理者阅读。

Eric Brechner: I. M. Wright's "Hard Code" (ISBN: 978-0-7356-2435-1).

Copyright 2008 by Microsoft Corporation.

Original English language edition copyright © 2008 by Eric Brechner.

Published by arrangement with the original publisher, Microsoft Press, a division of Microsoft Corporation, Redmond, Washington, U. S. A. All rights reserved.

本书中文简体字版由美国微软出版社授权机械工业出版社出版。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

本书法律顾问 北京市展达律师事务所

本书版权登记号：图字：01-2008-2926

图书在版编目 (CIP) 数据

代码之道 / (美) 布莱什纳 (Brechner, E.) 著；陆其明译. —北京：机械工业出版社，2009.1

(Microsoft 核心技术丛书)

书名原文：I. M. Wright's "Hard Code"

ISBN 978-7-111-25167-5

I. 代… II. ①布… ②陆… III. 编码程序 IV. TP31

中国版本图书馆 CIP 数据核字 (2008) 第 146276 号

机械工业出版社 (北京市西城区百万庄大街 22 号 邮政编码 100037)

责任编辑：李东震

北京牛山世兴印刷厂印刷 · 新华书店北京发行所发行

2009 年 1 月第 1 版第 1 次印刷

186mm × 240mm · 13.25 印张

标准书号：ISBN 978-7-111-25167-5

定价：36.00 元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换
本社购书热线：(010) 68326294

读者对 I. M. Wright 的 “Hard Code” 栏目的喝彩

任何大型组织都会面临成为自身文化牺牲品的危险。事物的本源以及行事之道，玄之又玄的背后终是一个个自圆其说的预言。这确是任何组织发展趋势的终结所在，这对需要不断创新才能繁荣的技术公司来说，是个致命杀手。Eric Brechner 做了件难以置信的事——他亮出了手术刀，切入了这些组织结构中的软肋。他果敢地打出了重拳——偶尔也会招致恶名。他的一些行话和案例总能或多或少地为微软的员工带来启发，他的知识和才干是整个软件行业的巨大的财富。

——Clemens Szyperski，首席架构师

“I. M. Wright” 写的关于开发时间表的文章真是太棒了！它在我们部门参与的基础设施项目上同样适用。

——Ian Puttergill，部门经理

你没有受到任何死亡威胁，是吗？

——Tracey Meltzer，高级测试主管

这一定是个笑话——很坦率地说，这类纯粹的谬论是危险的。

——Chad Dellinger，企业架构师

Eric 是我本人崇拜的英雄——很大程度上是因为他长期以来一直代表着开发社区的一种声音。

——Chad Dellinger，企业架构师

软件工程师很容易就会迷失在他们的代码中，更糟糕的是，他们迷失在过程中。这时，迫切需要 Eric 在 “Hard Code” 中提出的实用建议。

——David Greenspoon，总经理

我刚刚读完这个月的栏目……我不得不指出，这是我第一次认为你的主张对公司来说完全是错误的并且会带来灾难。

——David Greenspoon，总经理

Eric 你真了不起：几个月之前，我跟我的产品单元经理和一些开发主管恰恰进行过这样的一次对话。

——Scott Cottrille，首席开发经理

我真的很喜欢这些栏目。它们是如此实用，而且令人茅塞顿开！我喜欢它们的另外一个

原因是，当我在指导初级开发人员的时候，我可以把这些栏目推荐给他们；他们也会记住这些栏目，因为它们都是那么的有趣。

——Malia Ansberry, 高级软件工程师

Eric, 干得好! 我觉得你在这个栏目中说得非常中肯。我想, 该给管理者传递这样的信息: “不要害怕试验。” 事情的真实情况跟理想化的理论之间差别是非常大的。

——Bob Fries, 合作伙伴开发经理

我只是想让你知道我有多喜欢你写的文章——它们充满智慧, 见解深刻, 你还神奇地把本来很严肃的问题变得如此有趣 (采用的方法很不错)。

——Niels Hilmar Madsen, 开发推广专员

你那篇关于死亡之旅的栏目来得正是时候。我们正打算在未来几周内开会讨论功能削减的事情呢! 那些我们以前付出很大代价才学到的教训, 不知怎么回事, 总是轻易就被忘记了; 你的栏目对大家起到了很好的提醒作用。

——Bruce Morgan, 首席开发经理

我想让你知道的是, 我真的很喜欢并感谢你在 EE 站点上发表的所有文章。不过, 直到今天, 当我读了 “Stop Writting Specs (停止写规范书)” 这个栏目之后, 我不得不说, 我强烈不同意你的观点。

——Cheng Wei, 项目经理

你到底是谁? 你跟 Eric Brechner 都做了些什么?

——Olof Hellman, 软件工程师

Eric, 我刚刚读完你写的那篇文章: “不恰当的比较” (Beyond Comparison)。你不知道我有多感激你! 你把这个观点真正地传达给了公司里面成千上万的人……你致力于正确领导和管理团队, 并把其中的奥秘跟大家分享, 对于你的这种热情我真的非常欣赏!

——Teresa Horgan, 业务规划经理

序

长期以来，我一直在阅读 Eric Brechner 以 I. M. Wright 为笔名撰写的栏目。当我第一次见到作者本人的时候，我费了好大的劲才让自己相信，我是在跟同一个人说话。栏目中的 Wright 先生非常地自信、张扬，然而站在我面前说话的人却那么谦虚、彬彬有礼、非常友好，他看起来更像 Clark Kent。[◎]

我很关注微软团队在软件开发的过程中是如何处理技术与人际交流之间的关系的；这类栏目总是我的最爱。看到大量的公司内幕被写了出来，我常常会感到吃惊——我不知道还有多少不为人知的故事没有说出来。

大型项目中的软件工程管理者面临着 3 个基本问题。第一个是，程序代码太容易被改变了。跟机械或土木工程不一样，它们在现有系统上做一次改变总是要付出实实在在地拆毁某些东西的代价，而软件程序的改变只需要敲敲键盘就行了。如果对一座桥的桥墩或一架飞机的引擎做一个错误的结构性更改，由此产生的后果，即使不是专家也很容易就能看出来。然而，如果在一个现有程序上做修改，对于其风险，即使经验丰富的软件开发者进行了充分的讨论，其结果常常还是错的。

建筑领域中的情况实际上可以很好地用来比喻软件开发。基于程序代码在系统中所处的层次，可以把其比作为“基础、框架和装饰”代码。“基础”代码具有高度的冒牌效应，它们的改动常常会引发严重的连锁反应。“装饰”代码比较容易改动，而且也需要经常改动。问题是，累积了几年的改变之后，复杂的程序就跟历经过几次装修的房子差不多了——电源插座躲到了橱柜的后面，浴室风扇的出风口通向了厨房。再做任何改变的话，其副作用或最终的结果都是很难预知的。

第二个基本问题是，软件行业还太年轻，关于可复用组件的正确标准实际上还没有被发现或建立起来。螺栓是否应该放在离开 16 英寸的地方，以同时适应水平或垂直的 4 英尺 × 8 英尺的干垒墙或夹板？我们不仅在这类问题上还没有取得一致意见，甚至我们还没有决定，是否像螺栓、干垒墙和夹板这样的组合更可取，还是我们要去发明像泥浆、稻草、石头、钢铁和碳纤维这样的组合。

最后一个问题实际上是第二个问题的另一种表现形式。每个项目中重复发明的软件组件也被重复命名了。软件行业里对现有的概念发明新的名字是很常见的，即使用的名字相同，

◎ Clark Kent 是“超人”的名字，他具有超强的本领，是一个虚构的超级英雄，美国漫画中的经典人物。——译者注

这些名字也以新的方式被重用。行业里有一个心照不宣的秘密：软件开发最佳方法的大量内容，实际上是由不同的人群参与讨论而且使用了不同的名字，他们对彼此正在说的东西都没有一个哪怕是很朦胧的想法。

表面上看来，这些都是很简单的问题。建立一些标准，然后强制实行它们。在快速进步的大容量、高价值、低成本的软件世界里，这可是一个让你的业务落败的捷径。实际情况是，软件最大的工程障碍，同时也是它最大的优势。无处不在的软件（运行在低成本的个人电脑和互联网上），已经使得以惊人的步伐去创新成为可能。

随着微软的成长，公司已经不能在最佳工程实践的研究方面大量地投入，然后经过深思熟虑，挑选出其中具有最好质量的方法。个人电脑和 Windows 的成功，已经把公司从按传统方式做些小项目的形态转变出来，转而去谱写开发有史以来最庞大、最复杂软件的新篇章。

为了能够创建出平衡风险与效率、创新的最佳系统，微软面临着持续不断的挣扎。考虑到我们的一些项目极其复杂，这些努力甚至可以称得上“英勇无畏”。在过去的一段时间以来，我们已经设置了专员，建立了专门的组织，他们都一心一意、致力于这个行业里最困难的事情——“软件发布”。我们已经学会了很多的民间传说、风俗、文化、工具、过程和大拇指规则^①，这些都有助于我们建造和发布这个世界上最复杂的软件。但与此同时，每天都处理这些问题难免也让人心惊胆战、士气受挫。Eric 的栏目正是大家跟我们一起分享和学习的极好方式。

Mike Zintel，微软公司 Windows Live 内核开发部门总监

2007 年 8 月

① Rules of Thumb，是指没有经过科学实验、直接从实践中总结出来的方法和规则；它们在很多情况下都有用，但并不是放之四海皆准。——译者注

前言

献给当初对我说“你为什么不写？”的人：Bill Bowlus

这是一本关于极其实操性的书。它会比较乏味。但也许会比较有意义，你能从中得到知识，读后甚至对你产生些许影响，但读起来肯定还是干巴巴而无趣的。为什么这么说呢？

实操性强的书是乏味的，因为它是跟具体的项目、具体的人、具体的目标以及偏好紧密相关的。一次实践能不能产生“最佳”效果，大家可能看法不一。作者必须把实践列举出来让读者自己来选，并分析在什么时候、因为什么原因作出最佳选择。虽然这种做法是现实的、负责任的，但也是令人厌烦的，最终无法取悦读者。为释疑而设计的案例研究会使文字有趣味一些，但作者仍必须把选择的机会留给读者，否则作者就会显得傲慢、教条并且死板。

然而人们喜欢看到傲慢、教条、死板的学者之间的针锋相对。大家喜欢引用学者们的只言片语，与朋友和同事一起讨论。为什么不把这些最佳实践当作观点栏目来争论呢？唯一的条件，就是要有人愿意将自己扮演成一个思想保守的傻瓜。

本书的由来

2001年4月，在我历经了Bank Leumi、Jet Propulsion Laboratory、GRAFTEK、Silicon Graphics、Boeing等公司总共16年的职业程序员生涯，以及在微软做了6年的程序员和经理之后，我转入了微软内部的一个以在公司范围内传播最佳实践为职责的团队。当时这个组正在运作发行名为《Interface》月刊网络杂志的一个项目。它很有意义，且富有知识性，但同样也是枯燥无趣的。我那时建议增加一个观点栏目。

我的上司Bill Bowlus建议由我来写。我拒绝了。作为一个半大孩子，我努力成为一个协调员，撮合多方产生成果。成为一个爱唠叨的实践学者会毁掉我的名誉和效力。因此，我当时的想法是说服一个大家公认的小心眼的工程师来写，他可能是我在微软前6年工作经历中接触过的一位最固执的开发经理之一。

但Bill指出，我有22年的开发经验，4年的开发管理经验，写作技巧也还行，而且有足够认真的态度来做这件事。我只需要放下自身的心理包袱。另外，其他开发经理都忙于日常的工作，不可能每个月来为我们写观点。最后Bill和我想出了一个用假名撰稿的点子，于是I. M. Wright的“Hard Code”栏目诞生了。

从2001年6月开始，我使用“I. M. Wright，微软逍遥的开发经理”这个笔名为微软的开发者和经理写了49个“Hard Code”观点栏目。这些栏目的标签行都打上了“绝对诚实，

重拳出击”的标语。每个月，成千上万的微软工程师和经理都在读这些栏目。

前 16 个栏目在《Interface》内部网络杂志上发表了。随后同事们（Mark Ashley 和 Liza White）给我分配了更多的主题。我和《Interface》的美工 Todd Timmcke 还一起炮制了作者的很多搞怪照片。当网络杂志停刊的时候，我才得以喘息一会儿，但也停止了写作。

14 个月之后，在我们组的同事 Amy Hamilton（Blair）、Dia Reeves、Linda Caputo、Shannon Evans 和 Marc Wilson 的帮助下，我又开始在内部站点上发表我的栏目。去年 11 月份，我将所有栏目转移到了一个内部的 SharePoint 博客上。

2007 年春天，正当我打算休掉几年前奖励给我的假期的时候，我现在的经理 Cedric Coco 给了我在休假期间将“Hard Code”栏目内容整理出版成书的授权。而微软出版社的 Ben Ryan 也同意了。

除了我已经提及的人，我还想感谢《Interface》的其他成员（Susan Fario、Bruce Fenske、Ann Hoegemeier、John Spilker 和 John Swenson），其他帮助过本书出版的人（Suzanne Sowinska、Alex Blanton、Scott Berkun、Devon Musgrave 和 Valerie Wolley），支持我的管理层（Cedric Coco、Scott Charney 和 John Devaan），我现在的和以前的、复审过我的栏目并提出过很多主题的团队（William Adams、Alan Auerbach、Adam Barr、Eric Bush、Scott Cheney、Jennifer Hamilton、Corey Ladas、David Norris、Bernie Thompson、James Waletzky、Don Willits 和 Mitch Wyle），我才华出众的中学英语老师（Alan Shapiro），以及那些慷慨给予我反馈的读者们。特别地，我还要感谢我的妻子 Karen 和我的儿子 Alex 和 Peter，他们让我做任何事情都充满信心。

本书的读者

组成本书的 49 个观点栏目最初是写给微软的开发人员和他们的经理的，这些是从我过去在软件行业 6 个不同的公司、28 年的工作经验中提炼出来的。编辑和我一起修正了文字，注解了那些微软内部的特殊用语，使得本书适合于所有软件工程师和工程经理阅读。

我在这些栏目中表达的观点是我个人的，不代表我现在和以前任职过的任何一家公司，包括微软。我在栏目中的注解以及简介中的言论同样都是我个人的，与公司无关。

本书的组织方式

根据主题的不同，我把所有栏目分成了 10 章。前 6 章剖析了软件开发流程，接下来 3 章重点讨论人的问题，最后 1 章批判软件业务运转方式。用于解决这些问题的工具、技巧和建议遍布全书。本书的最后还附有术语表方便大家参考。

每一章的各个栏目均按照当初在微软内部发表的时间顺序排列。每章开头我都给出了一个简短的介绍，随后就是当初我以“I. M. Wright”名义发表的栏目内容。编辑成书的时候，我还适时在栏目中加上了“作者注”，以解释微软的术语，提供更新内容或者额外的背景知识。

编辑和我尽力保持了原有栏目的完整性。我们做的，仅仅是纠正语法和内部引用。称得上改动的其实只有一处：就是将原来一个叫“你被解雇了”的栏目标题改成了“最艰难的工

作”，因为以前那个标题太容易让人误解了。

每个栏目都以一段激昂的演说开场，然后就是对问题根源的分析，最后以我对这个问题如何改善的建议结束。我酷爱文字游戏、头韵和通俗文化，因此栏目中充斥着对这些东西的引用。特别是大部分栏目的标题和副标题都直接取材于歌词、电影对白和有名的谚语。是的，我自娱自乐，但撰写这些栏目确实给我带来了些许乐趣以及痛快的宣泄。希望你也会喜欢！

系统要求

本书提供的工具都是微软的 Excel 2003 和 Word 2003 格式的。只要你的电脑上安装有 Word 和 Excel，你就能使用这些文件。

微软的组织结构

因为这些栏目最初是写给微软的内部员工看的，因此简要了解一下微软以及我在工作中扮演的角色会有助于更好地理解这些文字。

目前，微软的产品开发分成三大业务部门，总共有大概 25 条产品线，超过 450 个产品单元和众多的功能团队。这些部门是平台产品与服务部门、微软商业部门、娱乐与设备部门。部门内的产品线是由相关的产品套件整合在一起形成的，比如 Office System 和 Visual Studio。

每条产品线包含了大约 20 个独立的产品单元。通常情况下，这些产品单元共享源代码控制、建造、安装、工作条款跟踪和项目协调，包括价值主张、里程碑安排、发布管理和工程支持。除了这些协调服务之外，产品单元还有高度的自主权，可以对产品、流程和人员作出自己的安排。

一个典型的产品单元通常有一个产品单元经理（Product Unit Manager, PUM）和三个工程工种经理：部门项目经理（Group Program Manager, GPM）、开发经理（Development Manager）和测试经理（Test Manager）。其他工程工种，比如用户体验、内容发布（比如在线帮助）、实施，可能单独对某个产品单元负责，也可能在产品线或者整个部门中共享。

每个工种都要抽出一个或多个代表，以组成一个叫功能团队的虚拟组织，来开发具体的产品功能。这些人的工作仍然汇报给各自的工种经理。有些功能团队选择敏捷方法，有些喜欢精益模型，有些采用传统的软件工程模型，有些则根据实际情况混合了上述多种方法。

微软怎么能包容所有这些多样性和产品的区域自治、还能朝着一个共同的目标有效地工作呢？这就要靠产品线公共的项目协调了。举例来说，产品线的价值主张为所有产品单元和他们的功能团队设置并统一关键应用范例、质量尺度和框架体系。

为了促进跨部门和产品线的协调和共享，特别是为提高质量和效率，微软还设立了一个组织叫 Microsoft Trustworthy Computing and Excellence。它是我的上级组织。具体来说，我的工作职责是，要让微软全球超过 1 万名的开发者在构建令人兴奋的、高质量的用户体验时，能够做得更加开心、更富有效力。不用说，这是一项长期的工作。

我的团队每月将开发部门的领导层召集在一起，讨论问题并指导我们的工作。我们研究公司范围内乃至整个行业内的工程方法，以期找到新的机会和领域去提高。我们通过网络共享工具、最佳实务，以及实施职业生涯指导。我们举办各种活动和奖项评比，与各种团队商

讨，为开发中的各个级别、各种角色提供技术培训。非常棒的工作！另外，我每月还在写着观点栏目。

示范工具和文档

本书提到的示范工具和文档，作为本书的附属内容，都可以通过如下网址下载到：

<http://www.microsoft.com/mspress/companion/9780735624351>

在线资料列表

工 具	栏 目	章节
Sprint backlog; SprintBacklogExample.xls; SprintBacklogTemplate.xlt	The agile bullet (敏捷子弹)	2
Product backlog; ProductBacklogExample.xls; ProductBacklogTemplate.xlt	The agile bullet (敏捷子弹)	2
Spec template; Spec template.doc	Bad specs; Who is to blame (糟糕的规范书：该指责谁?)	3
Spec checklist; Spec checklist.doc	Bad specs; Who is to blame (糟糕的规范书：该指责谁?)	3
Pugh Concept Selection; PughConceptSelectionExample.xls; PughConceptSelectionTemplate.xlt	Review this (复审一下这个)	5
Inspection Worksheet; InspectionWorksheetExample.xls; Inspection WorksheetTemplate.xlt	Review this (复审一下这个)	5
Interview Role Playing, a how-to guide; InterviewRolePlaying.doc	Out of the interview loop (面试流程之外)	9

为本书提供的支持

为了保证本书及其附属内容的准确无误，我们尽了最大的努力。本书所有修正和改动都会搜集起来并放到微软的知识库中去。微软出版社通过如下站点对本书及其附属内容提供支持：<http://www.microsoft.com/learning/support/books/>。

问题和意见

如果你对本书及其附属内容有任何意见、建议或问题，并且通过访问上述站点仍然无法解决，请给微软出版社发送电子邮件：mspinput@microsoft.com，或者按照如下方式邮寄：

Microsoft Press

Attn: I. M. Wright's "Hard Code" Editor

One Microsoft Way

Redmond, WA 98052-6399

请注意，上面的地址对微软的软件产品不提供支持。

目 录

读者对 I. M. Wright 的“Hard Code”

栏目的喝彩

序

前言

第 1 章 项目的不当管理 1

2001 年 6 月 1 日：“开发时间表、飞猪和
其他幻想” 2

里氏震级估计 2

风险管理 3

客户赢了 4

2001 年 10 月 1 日：“竭尽所能：再论
开发时间表” 4

软件工程绝对是含糊的 4

相信一半你看到的，别信你听到的 5

激励：不能光靠比萨和啤酒 6

在日期上沉沦 7

2002 年 5 月 1 日：“我们还开心吗？
分诊的乐趣” 8

战争是地狱 8

这不是个人的事情 8

分诊的 5 条黄金法则 9

魔鬼藏在细节里面 10

很难进行下去，不是吗 10

谨小慎微 11

2004 年 12 月 1 日：“向死亡进军” 11

暗箭伤人 12

对失败的连祷 13

转折点 14

很少有人走过的路 14

2005 年 10 月 1 日：“揭露真相” 15

遭受错觉之苦 15

拿把叉子扎进我的身体 16

给我个坦率的回答 16

给猪抹口红 17

看看所有这些传言 18

我想知道真相 19

第 2 章 过程改进，没有魔法 20

2002 年 9 月 2 日：“六西格玛？
饶了我吧！” 21

啊！这是什么巫术 21

召集骑兵 22

在混沌之外建立秩序 22

2004 年 10 月 1 日：“精益：比帕斯雀
牛肉还好” 23

任何事情都要适中 23

俭则不匮 23

过量生产 24

走向深处 25

运输 25

多余动作 26

等待 27

过程不当 27

库存 28

缺陷 28

组合应用 28

2005 年 4 月 1 日：“客户不满” 29

但愿不知道 29

太过分，太迟了 30

敏捷错觉 31

后退你的步伐	31	坚持做一件事情	53
更多用武之地	31	你准备好了吗	53
使用正确的工具	32	2007年2月1日：“糟糕的规范书： 该指责谁？”	54
布基胶带和打包钢丝	33	树立靶子	54
客户满意	33	沟通分解	54
2006年3月1日：“敏捷子弹”	34	保持简单容易	55
真理的敌人	34	变得稳健	55
拨乱反正	35	获取反馈	56
准备改变了吗	36	集成质量检查	56
让他说话	37	差别在哪	57
你完善我	37	第4章 跨越工种	58
有点极端	38	2002年4月1日：“现代临时夫妻？ 开发与测试”	59
准备玩橄榄球	38	我怎么爱你？看看我的招术	59
最后你要知道的	39	必要的邪恶或珍贵的伙伴	60
第3章 根除低下的效率	41	每个人都要知道自己的弱点	61
2001年7月1日：“迟到的规范书： 生活现实或先天不足”	42	你完善我	61
对于每次变更，搅动，搅动，搅动	42	2004年7月1日：“感觉性急——测试者 的角色”	62
走廊会议	43	高级保护	62
委员会会议	43	改变一下对你有好处	63
规范书变更请求	43	黎明时分	63
预防是最好的治疗	44	充分利用数据	64
2002年6月1日：“闲置人手”	44	非常酷——我保证你	64
宝宝做了件极坏的事情	45	2005年5月1日：“模糊逻辑—— 君子之道”	65
告诉我该做什么	47	包罗万象	65
俭则不匮	48	他们跟我们不一样	66
2004年6月1日：“我们开会的时候”	48	通过安检	68
为什么我们会在这里	49	着手去改变	68
我们正在试图做什么	49	更好地在一起	68
为什么他们会在这里	50	2005年11月1日：“废除工种——有什么 理由搞专业化？”	69
为什么我现在才听到这个	50	历经未来的日子	69
接下去要做什么	51	考察它的极限	70
2006年7月1日：“停止写规范书， 跟功能小组呆在一起”	51	足球是门科学	70
你失去理智了吗	51		
在那里进退两难	52		
特殊要求	52		
我不记得了	53		

两者之间的距离	71	神奇的汇总会议	88
你深陷其中	71	审查的诀窍	89
第 5 章 软件质量不是梦	72	走上正道	90
2002 年 3 月 1 日:“你对你的安全		2006 年 10 月 1 日:“对质量的大胆预测”	90
放心吗?”	73	谜?我不这么认为	90
小心晃动的钟摆	73	邪恶双煞	91
做正确的事	74	嫌疑惯犯	91
安全受制于最薄弱环节	74	你会喜欢它的	91
领导、跟随或者离开	75	停止卖弄愚蠢	92
2002 年 11 月 1 日:“牛肉在哪里?		质量就是没有意外	92
为什么我们要质量”	75	第 6 章 有时间就做软件设计	94
情况变了	76	2001 年 9 月 1 日:“错误处理的灾难”	95
足够好还不行	77	恐怖,恐怖	95
艰难的选择	77	使用异常	96
终于有足够的时间了	78	别丢弃,用上它	96
再检查一遍	79	2002 年 2 月 1 日:“厨师太多烧不好菜——	
医生,治好你自己的病	79	唯一权威”	97
步步为营	80	一幅图片抵得上一千个字	97
太多疑问	80	有人确切知道现在几点了吗	97
2004 年 4 月 1 日:“软件发展之路——		只能有一个	98
从手工艺到工程”	81	万物皆有联系	98
工艺制桌子,工程造汽车	81	2004 年 5 月 1 日:“通过设计解决”	99
其实你知道	82	如何才算足够好	99
真实面对自己	82	设计完成	99
数字的含义	83	细节,细节	101
各人有各人的习性	83	让我看看你是由什么组成的	101
大处着想,小处着手	84	当心缺口	102
从优秀到卓越	84	成功处方	103
2005 年 7 月 1 日:“复审一下这个——		2006 年 2 月 1 日:“质量的另一面——	
审查”	85	设计师和架构师”	103
糟糕的混合	85	你必须比那做得更好	103
完美风暴	85	改变一下对你有好处	104
谁来负责	86	他这么做不对	104
你有什么想法	86	正确的做法	104
正是这个形式	87	下一次,试试雕塑	105
孩子,准备好了吗	87	关键要有正确的工具	105
再检查一遍	88	打破这些壁垒	106
		2006 年 8 月 1 日:“美妙隔离——	
		更好的设计”	106
		分解难做	106

正确的做法	107	逐渐长大	125
团队不需要“我”	107	我脑子里闪过的阴影和凶兆	126
循序渐进	108	不要伤害 Messenger	126
猫狗不分家	109	皆大欢喜	127
第7章 职业生涯历险	110	2005年2月1日:“最好学会平衡生活”	128
2001年12月1日:“当熟练就是目标”	111	平衡是关键	128
每个人都要知道自己的弱点	111	光说不练	128
享其成但不坐等	112	我甚至不能平衡我的支票簿	129
我希望他们尊重我	112	平衡好,一切都好	130
我们都牵连其中	112	2005年6月1日:“时间够用了”	131
2002年10月1日:“生活是不公平的—— 考核曲线”	113	直接告诉我	131
我不想再逆来顺受了	114	免受打扰之苦	131
知识就是力量	114	找到你的乐园	132
关注业务	114	我们谁也不笨	132
前进,让我快乐	115	我们必须共同承担	133
伸出手去接触某人	115	告诉我必须做什么	133
有了柠檬?制作柠檬水	116	他还是个孩子	134
改变你的主意	116	你应该休息一下	134
方向盘后面的人	116	这里秩序井然	134
2006年11月1日:“职业阶段中的角色”	117	坦诚相待	135
一个人同时扮演很多角色	117	大有可为	136
搞清楚职业阶段	118	2005年8月1日:“有理有节地控制 你的上司”	136
我是有抱负的	118	我没辙了	136
资历过高	118	知彼知己	137
我是特殊的	119	他们能自我适应	137
只能选一个	119	把水卖给鱼	138
你想成为什么	119	势利的眼睛	139
2007年5月1日:“让你自己与 世界相连”	120	付诸行动	139
你认识的人	120	敢于做梦	140
我利用习惯	121	2006年4月1日:“你在跟我说话吗? 基本沟通”	140
难道你不好奇	121	为我着想一下	141
你得到了我们的感谢	122	告诉我你想要什么	141
我回头再找你	122	你什么时候想要	142
欢迎来到这个世界	123	缩小注意力跨度	142
第8章 自我完善	124	就这样完了	143
2002年12月1日:“要么听我的, 要么走人——协商”	125	2007年3月1日:“不只是开放和诚实”	144
一个你无法拒绝的方式	125	那不是理由	144
		我会对你诚实	144

那不容易	145	放任自流	163
他们似乎有个开放政策	145	2005 年 12 月 1 日：“我能够管理”	163
无处隐藏	146	持续送出的赠品	164
跟我想的不一样	146	优秀就够了	164
走上正道	147	草率行事	164
第 9 章 成为管理者，而不是邪恶		我想要工作	165
的化身	148	我不是物件	165
2003 年 2 月 1 日：“不只是数字——		从优秀到卓越	166
生产力”	149	我服务于人	167
小心你希望得到的东西	149	2006 年 5 月 1 日：“不恰当的比较——	
扮演一个角色	150	病态团队”	167
卓越开发者的素质	150	想要挑起战争	167
你要做法官	151	这不是竞争	168
2004 年 9 月 1 日：“面试流程之外”	151	我会给你些提示	169
抱怨得不到帮助	151	团结在一起	169
90% 是准备	152	第 10 章 微软，你会喜欢它的	171
那就是问题	152	2001 年 11 月 1 日：“我是怎么学会停止	
白板编译器	153	焦虑并爱上重组的”	172
帮招聘专员准备	154	沿着巴别塔下来	172
再次帮面试官准备	154	地狱里的生活	173
友情提醒	155	很少有人走过的路	173
最后的难题	155	容忍问题还是主动去解决	174
2004 年 11 月 1 日：“最难做的工作——		2005 年 3 月 1 日：“你的产品单元经理	
绩效不佳者”	156	是个游民吗？”	175
你期望什么	156	有计划的人	175
知难而进	157	我等不及要去实施了	176
寻求专业援助	157	魔鬼藏在细节里面	176
没人想失败	158	道路规则	176
目标是成功	158	回到正确的跑道上	177
无所求，则无所获	158	2006 年 9 月 1 日：“有幸成为 Windows	
你不会总能如愿	159	的主宰者”	177
2005 年 9 月 1 日：“随波逐流——		你还有别的要求吗	178
人才的保持和流动”	159	准备轮船	178
我只是想环球旅行	160	设置路线	179
不错的水坝	160	启航	179
像河水一样流动	161	导航	180
新鲜血液	161	责任	181
分享就是关爱	162	下一代 Windows	181
成长空间	162	2006 年 12 月 1 日：“A 公司：严重的威胁	
我必须要旅行	163	还是糟糕的拼写？”	182

他们步伐踉跄，我们手舞足蹈	182	日子照过，只不过要掌握一点窍门	186
注定要失败	183	不轻易冒险	186
聪明人需要智能客户端	183	我认为他们还不能胜任	187
保持警惕	184	不再年轻了	187
一马当先	184	不要惊慌失措	187
2007年4月1日：“中年危机”	185	没有人是完美的	188
你已经变了	185	术语表	190