

The
Pragmatic
Programmers

TURING

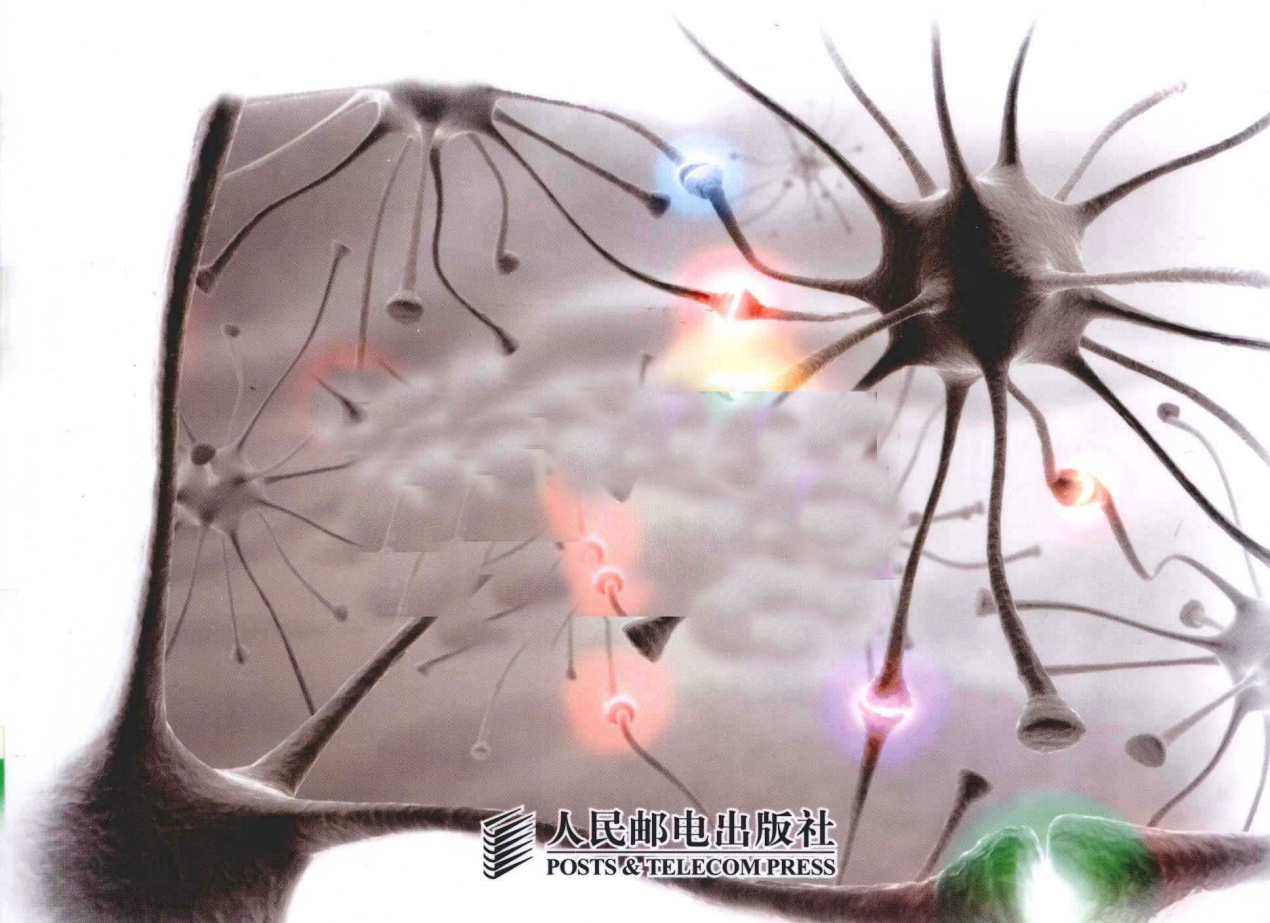
图灵程序设计丛书

Pragmatic Thinking and Learning
Refactor Your Wetware

程序员的思维修炼

开发认知潜能的九堂课

[美] Andy Hunt 著
崔康 译



人民邮电出版社
POSTS & TELECOM PRESS

TURING 图灵程序设计丛书

Pragmatic Thinking and Learning
Refactor Your Wetware

程序员的思维修炼

开发认知潜能的九堂课

[美] Andy Hunt 著
崔康 译

人民邮电出版社
北京

图书在版编目 (C I P) 数据

程序员的思维修炼：开发认知潜能的九堂课 / (美) 亨特 (Hunt, A.) 著；崔康译. — 北京：人民邮电出版社，2011.1

(图灵程序设计丛书)

ISBN 978-7-115-24233-4

I. ①程… II. ①亨… ②崔… III. ①程序设计
IV. ①TP311.1

中国版本图书馆CIP数据核字(2010)第215268号

内 容 提 要

本书解释了为什么软件开发是一种精神活动，思考如何解决问题，并就开发人员如何能更好地开发软件进行了评论。书中不仅给出了一些理论上的答案，同时提供了大量实践技术和窍门。

本书供各层次软件开发人员阅读。

图灵程序设计丛书

程序员的思维修炼：开发认知潜能的九堂课

-
- ◆ 著 [美] Andy Hunt
 - 译 崔 康
 - 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 电子函件 315@ptpress.com.cn
 - 网址 <http://www.ptpress.com.cn>
 - 北京铭成印刷有限公司印刷
 - ◆ 开本：700×1000 1/16
 - 印张：14 彩插：2
 - 字数：233千字 2011年1月第1版
 - 印数：1-3 000册 2011年1月北京第1次印刷
 - 著作权合同登记号 图字：01-2009-2909号

ISBN 978-7-115-24233-4

定价：39.00元

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

反盗版热线：(010)67171154

版 权 声 明

Copyright © 2008 Andy Hunt. Original English language edition, entitled *Pragmatic Thinking and Learning: Refactor Your “Wetware”*.

Simplified Chinese-language edition copyright © 2011 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。

译者序

这是一本教你如何对大脑“编程”的书！

运用一门程序设计语言编程对大多数普通程序员来说是“小菜一碟”，那么如何更上层楼成为一名专家级的软件开发者呢？本书给出了答案——优秀的学习能力和思考能力。作者从软件开发领域的角度，阐述了每一名程序员提升“内力”所需要的各种软性知识：从新手到专家的5个层次、人类大脑的运行机制、直觉和理性的利与弊、学习方法和实践经验的重要性、控制注意力的技巧，等等，可谓是一本程序员“素质教育”的微型百科全书。我非常支持一个白话版的“素质”定义：除了书本知识、硬性记忆以外的东西，扪心自问，包括我自己在内的大多数程序员除了固化的编程知识以外，又有多少“素质”拿得出手呢？IT领域知识更新换代之快需要我们不停地往前奔跑，当我们痛苦地追逐时尚的新鲜玩意时，更需放慢脚步，冷静地修炼自己的“内功”，以不变应万变，才能立于不败之地。如果你想改变现状，本书可以作为一个良好的起点。作者对各种软性技能都做了深入研究，并结合自己的经验总结成你可以借鉴的知识点，让你无需阅读各个领域（认知科学、神经学、行为理论）的专著，就能够汲取适合自己的精华。

在翻译本书时，我切实地感受到，虽然它文字不多、篇幅不大，但却内容丰富、引经据典，可见作者知识的渊博和写作的认真。我建议读者在阅读本书时，不要急于求成，要仔细地阅读各个章节，结合自己的日常经验体会文字背后的含义。对每一节中的“实践单元”，要立刻应用到日常工作中，观察和比较实践的前后效果，找出适合自己的行动指南！

千里之行始于足下。请翻开本书的下一页，或许可以改变你的一生。

对本书的赞誉

本书将推动你拥抱美好未来。

——Patrick Elder, 敏捷开发人员

遵循Andy推荐的具体步骤,你能够使自己最宝贵的财富(大脑)更具效率,更有创造力。请阅读本书,并按照Andy说的去做。你的思维会比以前更敏捷,工作会更出色,同时也会比以前学到更多。

——Bert Bates, *Head First* 系列书的合著者

我一直在寻觅能够帮助我提高学习能力的东西,但我还没找到可以与本书媲美的。本书提供了最好的方法,帮助你成为专家级学习者,提高你的技能,指导你如何通过快速易行的学习提高工作效率。

——Oscar Del Ben, 软件开发人员

我喜欢谈论与情境相关的书籍。本书就是这样做的,而且帮助你理解为什么情境如此重要。从德雷福斯(Dreyfus)模型(让我顿悟很多事情)到解释为何体验性培训效果显著(书中的攀岩例子),Andy以其幽默和老练的写作风格,让读者在阅读中有所收获,并提高思考和学习的能力。

——Johanna Rothman, 顾问、作家和演说家

我非常喜欢Andy的著作,其内容严谨且实用。

——Patricia Benner 博士, 旧金山加利福尼亚大学社会
与行为科学系教授、系主任

2 ► 对本书的赞誉

昨晚读完了本书的 beta 版。我非常喜欢 NFJS 研讨会上的有关讨论（赛马一节），非常希望它能成书，真的太棒了。本书的所有内容实际上改变了我的人生！

——Matt McKnight，软件开发人员

这本书太有趣了，我受益匪浅。这对我来说足够了。

——Linda Rising，国际演说家、顾问和面向对象领域专家

献给我的太太和孩子。

献给那些梦想成真的人。

目 录

第 1 章 绪论	1
1.1 再提“实用”	3
1.2 关注情境	4
1.3 所有人都关注这些技能	5
1.4 本书结构	6
1.5 致谢	9
第 2 章 从新手到专家的历程	11
2.1 新手与专家	12
2.2 德雷福斯模型的 5 个阶段	15
2.3 现实中的德雷福斯模型：赛马和赛羊	21
2.4 有效地使用德雷福斯模型	26
2.5 警惕工具陷阱	32
2.6 再一次考虑情境	34
2.7 日常的德雷福斯模型	35
第 3 章 认识你的大脑	37
3.1 双 CPU 模式	38
3.2 随时 (24×7) 记录想法	42
3.3 L 型和 R 型的特征	45
3.4 R 型的崛起	51
3.5 R 型看森林，L 型看树木	56
3.6 DIY 脑部手术和神经可塑性	57
3.7 如何更上一层楼	58
第 4 章 利用右脑	60
4.1 启动感观输入	60
4.2 用右脑画画	62
4.3 促成 R 型到 L 型的转换	66
4.4 收获 R 型线索	76
4.5 收获模式	85
4.6 正确理解	91
第 5 章 调试你的大脑	93
5.1 了解认知偏见	94

5.2 认清时代影响	102
5.3 了解个性倾向	109
5.4 找出硬件问题	112
5.5 现在我不知道该思考什么	116
第 6 章 主动学习	119
6.1 学习是什么……不是什么	119
6.2 瞄准 SMART 目标	122
6.3 建立一个务实的投资计划	126
6.4 使用你的原生学习模式	130
6.5 一起工作，一起学习	134
6.6 使用增强的学习法	136
6.7 使用 SQ3R 法主动阅读	137
6.8 使用思维导图	140
6.9 利用文档的真正力量	146
6.10 以教代学	148
6.11 付诸实践	149
第 7 章 积累经验	150
7.1 为了学习而玩耍	150
7.2 利用现有知识	154
7.3 正确对待实践中的失败	155
7.4 了解内在诀窍	158
7.5 压力扼杀认知	162
7.6 想象超越感观	165
7.7 像专家一样学习	169
第 8 章 控制注意力	171
8.1 提高注意力	172
8.2 通过分散注意力来集中注意力	177
8.3 管理知识	179
8.4 优化当前情境	185
8.5 积极地管理干扰	190
8.6 保持足够大的情境	195
8.7 如何保持注意力	199
第 9 章 超越专家	200
9.1 有效的改变	200
9.2 明天上午做什么	202
9.3 超越专家	203
附录 A 图片授权	206
附录 B 参考文献	207

第 1 章

绪 论

欢迎大家！

感谢选择这本书。我们将共同经历一次有关认知科学、神经学、学习和行为理论的旅程。你将看到人类大脑令人惊奇的工作机制，并研究如何克服这一系统的局限来提高自己的学习和思考技能。

我们将开始重构你的“湿件”——对大脑进行“重新设计”和“重新连线”，使你更高效地工作。不论你是程序员、管理者、“知识工人”、技术奇人还是思想家，或者你只是想让你的大脑聪明一点，这本书对你都会有所帮助。

我是一名程序员，所以我的例子和言论都离不开软件开发领域。但是，如果你不是开发人员，也不必担心，实际上程序设计与使用神秘、深奥的编程语言编写软件没有多大关系（尽管我们总是习惯性地认为它们很相关）。

程序设计其实就是解决问题，它需要发明、创造和灵感。不论你从事什么职业，可能都需要创造性地去解决问题。然而，对于程序员来说，既要受到数字计算机系统的严格约束，又要展开丰富而灵活的人类思考，这就会展示二者的强大力量，又会深深地暴露二者的缺陷。

无论你是一名程序员，还是一位心灰意冷的用户，可能都曾认为软件开发是人类可以想象和遇到的最艰难的工作。它的复杂性耗尽了我们的全部智慧，而一旦失败则后果是可怕的，且往往极具新闻价值。我们曾经让宇宙飞船偏离了轨道，撞向遥远的星球；让昂贵的火箭爆炸，蒙受无法弥补的实验损失；给消费者寄去索要零美元的催款信，搞得人家莫名其妙；时不时还让航空旅客滞留在机场。

刚才我们发现：这完全是我们自己的错误造成的。我们自身往往增加了程序设计的难度。随着软件行业的不断发展，我们似乎失去了作为一名软件开发

人员所必需的最基础、最重要的技能。

不过好在，我们此时此地就能改正这个错误。本书将告诉你如何去做。

过去40年中，程序员引入到程序中的缺陷的数量已经基本保持不变。尽管程序设计语言、技术、项目方法论等都在不断改进，但缺陷发生的频率仍然保持在同一水平，没能得到改善^①。

也许这是因为我们一直关注着错误的事情。尽管技术上有了很多显著的改变，但有一样东西却始终没变：我们自己——作为开发人员的人。

软件并不是在集成开发环境（IDE）或其他工具上设计出来的，它是在我们的大脑中想象和创造出来的。

软件是在头脑中创建的。
Software is created in your head.

思想和概念是需要团队（也包括付钱让我们开发软件的人）中分享和交流的。我们已经在改进基础技术——程序设计语言、工具、方法上花费了很多时间，当然这也是十分必要的，但现在是我们更进一步的时候了。

现在我们需要研究的真正难题是团队内部和团队间的交流，甚至更困难的问题是完全陈旧的思想。没有任何项目是孤岛，软件不可能孤立地创建或者运行。

Frederick Brooks 在他的里程碑式的文章《没有银弹》[Bro86]中提出：“软件产品处于应用、用户、规则和硬件^②的合力之下。这些因素总是在不断变化，迫使软件产品也随之改变。”

Brooks 的言论把我们推向了社会漩涡的中心。考虑到社会中各个相关团体的复杂交互影响和社会的持续变化，在我看来当前最重要的两项技能就是：

- 沟通能力；
- 学习和思考能力。

软件行业正在逐步提高沟通能力。特别是敏捷方法（见注解栏），强调了团队成员之间、最终客户和开发团队之间的沟通交流。类似《演说之禅：职场必知的幻灯片秘技》[Rey08]这样的大众图书突然热卖，表明越来越多的人意识到简单、有效的沟通非常重要。这是个好的开始。

① 引自 Bob Binder，基于 Capers Jones 的研究成果。

② 也就是平台。

不过，提高学习和思考能力要更难一些。

程序员需要不断地学习——不仅仅是学习新技术，还包括应用的问题域、用户社区的奇思妙想、同事的古怪习惯、行业的八卦新闻和项目演进的重要特征，我们必须学习学习再学习，持续不断地学习，然后把学习成果应用到解决日常遇到的一切新旧问题上。

也许，这些听起来都相当容易，但学习能力、批判性的思考能力和创造力——所有这些扩展思维的能力，都取决于你自己。没有人教你这些东西，你必须自己学习。我们往往错误地看待老师和学生的关系：不光是老师在教，学生也要学。学习完全取决于你自己。

我希望本书能够帮你获得又快又强的学习能力和更实用的思考能力。

什么是敏捷方法

“敏捷方法”这个词最早出现在2001年2月的一次峰会上，与会的17个人都是软件开发行业领军人物，他们创建了各种开发方法，如极限编程、Scrum、Crystal，当然也包括我们的注重实效的编程。

敏捷方法在很多重要的方面都与传统的基于计划的方法不同，最显著的就是摆脱了死板的规则，丢弃了陈旧的日程表，注重实时的反馈。

我在本书中会经常提到敏捷方法，因为很多敏捷思想和实践都是与良好的认知习惯相融合的。

1.1 再提“实用”

从最初的《程序员修炼之道：从小工到专家》[HT00]到现在的 *Pragmatic* 系列图书，你会注意到我们一直在冠以实用（pragmatic）这个词。实用主义（pragmatism）的本质就是做对你有用的事情。

在开始讨论之前，请牢记：每个人都是不同的。虽然我引用的很多研究成果都已经被大部分人所沿用，但也有一些还未广为使用。我会运用大量不同的材料，既有通过对人脑的功能性核磁共振扫描证实的科学事实，又有一般概念性理论，既有荒诞故事，也有“嘿，Fred 做了尝试，这对他管用”的日常生活例子。

在很多情况下，尤其是在讨论大脑时，根本的科学依据是未知的或不可知的。但你不必为此担心：如果某种方法是有效的，那么它就是实用的，我就会写入书中，供你思考。我希望这些方法中的大多数对你都有用。

切忌随波逐流。

Only dead fish go with the flow.

不过，总有些人与众不同，你可能就是其中一员。这也没关系，不要盲目地听从任何建议，包括我的建议。你可以用开放的思维来阅读本书，尝试执行一些建议，再判断哪些对你有用。

什么是湿件

wet•ware | wet, we(e)r | 词源：wet+software

名词，谐语。指人脑细胞或思维过程，与计算机系统相对应。

也就是说，利用计算机模型类比人类的思维过程。

随着不断成长和适应，人们需要改变自己的习惯和方法。生命中没有什么恒久不变的，只有死鱼才随波逐流，尝试改变自己。请把本书当作改变的开始。

我将会分享在我的经历中发现的实用思想和方法，剩下的就看你的了！

1.2 关注情境

万事万物都是相互联系的：自然界、社会系统、你内心的想法、计算机的逻辑——所有事物构成了一个庞大的相互联系的现实世界。没有什么事物是孤立存在的，一切都是系统和更大的情境的一部分。

由于现实世界的相关性，小的事物可能会有意想不到的巨大影响。这种不成比例的影响作用是非线性系统的标志，也许你并未注意到，现实世界毫无疑问是非线性的。

当我们试图将某个事物单独挑出来时，我们发现它与宇宙中的其他事物是息息相关的。

——约翰·缪尔（美国作家），1911年，《山间夏日》

在本书中，你会发现一些活动只有不起眼的差异，看不出它们之间能有什么区别。比如，冥想与大声说出你的想法，或者在纸上写一句话与在计算机编

辑器中输入这句话。抽象地讲，这些事情应当是完全等价的。

然而，事实并非如此。

这些活动使用了人脑中完全不同的思维路径，这些思维路径深受人类思想和思考方式的影响。思想并没有与大脑或身体的其他部分切断，它们相互间是密切相连的。这只是一个例子（在本书的后面章节将会讨论更多关于人脑的话题），但它有助于说明理解这些相互作用的系统的重要性。

在《第五项修炼》[Sen90]一书中，Peter Senge 推广了系统思维（systems thinking）这个词，描述了另外一种观察世界的方法。在系统思维中，人们试图将一个事物看作几个系统的连接点，而不是一个独立的个体。

一切都是互相关联的。
Everything is interconnected.

例如，你可能把一棵树看作一个单独、离散的对象，立在地面。但事实上，一棵树至少是两个主要系统的连接点：树叶和空气的处理循环与根和泥土的处理循环。树不是静止的，也不是孤立的。更有趣的是，几乎没有人只是系统的一个观察者，不论你是否意识到，很可能你就是这个系统的一部分^①。

诀窍 1

始终关注情境。

将这句话写下来贴在你的墙上、书桌上、会议室里、白板上，甚至任何你独立思考或与他人共同思考的地方。我们将会在本书后面讨论这个主题。

1.3 所有人都关注这些技能

在我构思如何写这本书的时候，我发现很多不同领域的人都在谈论这些我感兴趣的话题。这些领域包括：

- MBA 和高级管理人员的培训；
- 认知科学研究；
- 学习理论；
- 护理、卫生保健、航空以及其他行业；

^① 来源于 Heisenberg 的测不准原理，更一般性的观察者效应（observer effect）认为人必须通过改变系统来认识它。

- 瑜伽和冥想；
- 编程、抽象和问题解决；
- 人工智能研究。

有些东西是基础的、各领域相通的。

There's something fundamental here.

当你从以上各个领域发现事物虽有不同的表象却有着相通之处，这其实是一个信号。在如此众多的不同情境下却拥有类似的思想，那么必然存在某些根本和重要的东西。

瑜伽和冥想训练近来相当流行，却似乎想不出有什么明确的原因。大约 2005 年 10 月前后，我曾经在飞机上看到杂志上的标题醒目地写着“公司提供瑜伽和冥想训练以减少不断增加的保健成本”。

大公司以前从没有提供过类似活动，但是医疗费用的急速增长迫使它们去寻找一切解决办法。很显然，它们相信瑜伽和冥想的练习者会比普通人拥有更强健的体魄。在本书中，我们更关心这些方法与认知相关的地方，当然如果能获得全面健康那也是很不错的收获。

我也注意到 MBA 和高级管理人员的很多教程都在提倡各种思考性的、创造性的、直觉性的技能，这些东西都是当前已有的研究成果，不过还没有普及到奋战在第一线的员工，包括我们这些知识型工作者。

但是不要担心，我会在本书中讨论这些话题，非 MBA 也能享受这些成果。

1.4 本书结构

每一次美好的旅程都由一幅地图开始，我们的地图就在本节。尽管本书的章节是按顺序依次排列的，但这些章节却是相互关联的。

毕竟一切事物都是相互联系的，但是当你面对一本顺序写成的书时，却不容易领悟到这个观点。你也许无法通过书中各章节中无数个“又见”的提示，去体会到这种关联性。那么通过本节内容，我希望你能稍微明白一些各主题之间的联系。

请记住，以下的主题就是本书的方向，虽然讨论中我们还会说些别的话题。

1.4.1 从新手到专家的历程

在本书的第一部分，我们将研究一下大脑为何如此运转，一开始就引入一

个流行的专业模型。

技能获取领域的德雷福斯模型（Dreyfus model）是研究如何超越新手层次、如何不断精通技术的有效方法。我们将会探讨德雷福斯模型，并特别关注成为一名专家的关键要素：应用你自己的实践经验、理解情境和利用直觉。

1.4.2 认识你的大脑

当然，在软件开发中最重要的工具就是你自己的大脑。我们将会讨论一些认知科学和神经系统科学的基本知识，因为它们与软件开发人员的兴趣密切相关，比如把人脑模型可以类比为双 CPU、共享总线设计，以及如何对你自己的大脑作“手术”。

1.4.3 正确使用大脑

一旦对大脑有了更深的认识，我们将想方设法提高创造力和问题解决能力，以及更有效地获取经验。

我们还将探讨直觉从何而来。直觉是专家的基本特征，事实上难以驾驭。你需要它，依赖它，但或许也莫名其妙地反对经常使用它。你可能总会怀疑自己或其他人的直觉，错误地认为它是“非科学的”。

我们将探讨如何转变这种思想，给直觉更大的活动空间。

1.4.4 调试你的大脑

直觉是非常奇妙的技能，当然直觉出错时除外。在人类思考中存在着许多“已知的缺陷”：个人认知偏见、时代及同代人的影响、固有的个性，甚至是大脑底层的生物性缺陷。

这些缺陷经常会误导人们做出错误的判断，甚至走向灾难性的深渊。

了解这些常见缺陷是消除其影响的第一步。

1.4.5 积极学习

既然我们对大脑的工作方式有了深入了解，那么接着我们开始研究如何利用这个系统，引入有关学习的话题。

请注意，这里我所说的学习具有广泛的含义，不仅指对新的技术、程序设