

/THEORY/IN/PRACTICE

# 卓有成效的程序员

## The Productive Programmer

O'REILLY®

机械工业出版社  
China Machine Press



HZ BOOKS

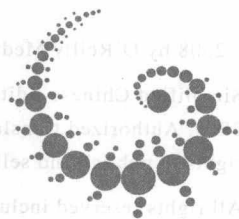
Neal Ford 著

ThoughtWorks中国公司 译

图书在版编目(CIP)数据  
卓有成效的程序员 / (美) 弗德(Ford, N.) 著; 机械工业出版社中国公司译.  
北京: 机械工业出版社, 2002.  
(O'Reilly精品图书系列)  
书名原文: The Productive Programmer  
ISBN 7-111-26495-4

# 卓有成效的程序员

## The Productive Programmer



Copyright © 2002 by Neal Ford, published by O'Reilly Media, Inc. and China Machine Press.  
All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior written permission of O'Reilly Media, Inc. or China Machine Press.

Neal Ford 著

ThoughtWorks 中国公司 译

in part in any form  
O'Reilly Media, Inc. 078-7-111-26495-4  
The Productive Programmer

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Sebastopol • Taipei • Tokyo

O'Reilly Media, Inc. 授权机械工业出版社出版

机械工业出版社

## 图书在版编目 (CIP) 数据

卓有成效的程序员 / (美) 弗德 (Ford, N.) 著; ThoughtWorks 中国公司译.  
—北京: 机械工业出版社, 2009.3

(O'Reilly 精品图书系列)

书名原文: The Productive Programmer

ISBN 978-7-111-26406-4

I. 卓… II. ①弗… ②T… III. 程序设计 IV. TP311.1

中国版本图书馆 CIP 数据核字 (2009) 第 023187 号

北京市版权局著作权合同登记

图字: 01-2009-1162 号

©2008 by O'Reilly Media, Inc.

Simplified Chinese Edition, jointly published by O'Reilly Media, Inc. and China Machine Press, 2009. Authorized translation of the English edition, 2008 O'Reilly Media, Inc., the owner of all rights to publish and sell the same.

All rights reserved including the rights of reproduction in whole or in part in any form.

英文原版由 O'Reilly Media, Inc. 出版 2008。

简体中文版由机械工业出版社出版 2009。英文原版的翻译得到 O'Reilly Media, Inc. 的授权。此简体中文版的出版和销售得到出版者和销售者的所有者——O'Reilly Media, Inc. 的许可。

版权所有, 未得书面许可, 本书的任何部分和全部不得以任何形式重制。

本书法律顾问: 北京市展达律师事务所

书 名 / 卓有成效的程序员

书 号 / ISBN 978-7-111-26406-4

责任编辑 / 周茂辉

封面设计 / Mark Paglietti, 张健

出版发行 / 机械工业出版社

地 址 / 北京市西城区百万庄大街 22 号 (邮编 100037)

印 刷 / 北京京师印务有限公司印刷

开 本 / 178 毫米 × 233 毫米 16 开本 14.5 印张

版 次 / 2009 年 3 月第 1 版 2009 年 3 月第 1 次印刷

定 价 / 45.00 元 (册)

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

## O'Reilly Media, Inc. 介绍

为了满足读者对网络和软件技术知识的迫切需求，世界著名计算机图书出版机构 O'Reilly Media, Inc. 授权机械工业出版社，翻译出版一批该公司久负盛名的英文经典技术专著。

O'Reilly Media, Inc. 是世界上在 Unix、X、Internet 和其他开放系统图书领域具有领导地位的出版公司，同时是联机出版的先锋。

从最畅销的《The Whole Internet User's Guide & Catalog》(被纽约公共图书馆评为 20 世纪最重要的 50 本书之一)到 GNN(最早的 Internet 门户和商业网站)，再到 WebSite(第一个桌面 PC 的 Web 服务器软件)，O'Reilly Media, Inc. 一直处于 Internet 发展的最前沿。

许多书店的反馈表明，O'Reilly Media, Inc. 是最稳定的计算机图书出版商——每一本书都一版再版。与大多数计算机图书出版商相比，O'Reilly Media, Inc. 具有深厚的计算机专业背景，这使得 O'Reilly Media, Inc. 形成了一个非常不同于其他出版商的出版方针。O'Reilly Media, Inc. 所有的编辑人员以前都是程序员，或者是顶尖级的技术专家。O'Reilly Media, Inc. 还有许多固定的作者群体——他们本身是相关领域的技术专家、咨询专家，而现在编写著作，O'Reilly Media, Inc. 依靠他们及时地推出图书。因为 O'Reilly Media, Inc. 紧密地与计算机业界联系着，所以 O'Reilly Media, Inc. 知道市场上真正需要什么图书。

## 业内好评

“对于程序员，过去我们一直习惯于用单纯的技术水平（也就是实现程序功能的能力）来衡量。然而这个时代其实已经过去了。”

虽然技术仍然很重要，但企业越来越多地认识到，对于程序员更全面的衡量标准，应当是生产率。只有能够以较高的效率完成对项目、对企业有价值的工作，才是团队和组织所真正需要的人才。反之，只有技术好，但不能真正促进整体价值，甚至其反作用，这样的‘技术牛人’已经没有生存空间了。”

——孟岩

《程序员》杂志总编

“卓有成效是可以学习的’，让编程工作变得卓有成效也是可以学习的，方法就在这里……我会要求我们团队的所有程序员认真研读这本书。”

——黄晶

校内网高级技术总监

“如果你想做一个真正的懒人，就请读完这本书，因为这本书是天堂；如果你不想做一个真正的懒人，那也可以读完这本书，因为它至少可以教会你掌握一些耍酷的小窍门，而且要比从前那种一遍遍敲ls或者dir滚屏更加专业。”

——李剑

InfoQ 中文站敏捷社区首席编辑

“通俗不掩睿智，腕从何妨心悟——用这句话可概括我喜欢本书的原因。”

——温昱

资深咨询顾问、《软件架构设计》作者

“如果你不能持续提高自己的价值，恐怕迟早会沦为公司的鸡肋。赢得竞争，就是让自己比别人更有效率！好习惯决定高效率，同时也希望本书能够让你的代码走正确的路……Neal Ford的这本书中有这样的一句话：‘我的使命，是让作为个体的程序员通过掌握恰当的工具和思想变得更加高效。’这本书倾注了ThoughtWorks公司精英们的心血，他们把自己的宝贵经验都分享了出来，那些工具和方法，定会让你受益匪浅。熊节，这位优秀的咨询师，他犀利的文字也为这本书凭添了几分独特，让人爱不释手。”

——胡铭娅

IT168 技术频道资深编辑

“有些知识，你可能需要工作和学习体验很久才能掌握，而本书作者 Neal Ford 已经直接把这些经验方法告诉你了……在熊节的翻译下，Neal Ford 的书读起来也比较对味，很是流畅。”

——华亮

CSDN 博客专家、Tencent 工程师

“一直以来，我都未曾在书店找到过一本关于代码和工程以外的、实战类的通用技巧性的书籍……如果你想在效率的竞赛中上百尺竿头更进一步，那这本书就是你需要的。”

——乔坦

.NET 程序员

“程序员总有学不完的东西，许多看过我写的‘程序员的十层楼’的人觉得自己仍然是‘菜鸟’。同样，当我看到 Neal 的这本书时，发现自己十几年的程序员生涯仍然是一个低效的程序员，书中介绍的许多提高效率的工具和方法以前没有用过或没有用好。”

要是在‘菜鸟’或‘大虾’阶段就能看到这样一本好书多好啊！不仅能及时掌握各种提高效率的工具和方法，更重要的是变成‘牛人’或‘大牛’后，它可以为设计高效的软件提供非常好的借鉴。”

——周伟明

多核编程专家

# 译者序

## 消除浪费，始于细节

在一次关于敏捷的讨论中，我说了一句令很多人不解的话：我不要敏捷。和很多话一样，断章取义地理解很容易造成误会。我当时说的整句话是：我不要敏捷，我要致力于消除软件开发中的一切浪费。当“敏捷”渐渐变成一个人见人爱的“大词”，越来越多的人开始发现，其实自己要的不是“be agile”（变得敏捷），而是切实地消除浪费、提高效率。

所以，作为 ThoughtWorks 员工的 Neal Ford 在他的这本书里闭口不谈“敏捷”。他只是实实在在地告诉你，作为一个程序员，你每天都在什么地方浪费着自己的生产率，以及如何去有效地消除这些浪费。

也许你甚至意识不到这些细小环节上浪费的存在。随便举个例子吧，在你一天的工作中，你有多少次从资源管理器里导航到源代码文件夹查看代码，然后又导航到另一个文件夹寻找文档，然后打开命令行窗口并进入项目目录，以及在密密麻麻的任务栏里找到正确的浏览器窗口？Neal Ford 说，这些都是浪费：做这些与核心任务（即软件开发）无关的事情是在浪费生产率。有兴趣知道这些自己每天做无数次的事还能如何改进吗？即便不是专业程序员，本书的第 2 章也将对你不无裨益。

从某种意义上来说，Neal Ford 在本书里做的事，正是现代科学管理理论的鼻祖弗雷德里克·泰勒在伯利恒钢铁厂做过的“泰勒实验”：剖析每个个体日常工作中的每个细节，对细节进行持续优化，通过对细节的改进提升生产率。在钢铁厂，泰勒的科学管理方法让一个搬运铁块的工人每天的工作效率提高了 3 倍；而在软件开发中对细节的重视甚至能让程序员的效率提升更多，因为人的体力终究有限，而脑力的开发程度还远未达到极限。

这并非痴人说梦，因为 ThoughtWorks 就是这样的例证。ThoughtWorks 有一群天才的程序员，只有近距离接触才会发现，这些人之所以能做到如此高效，很大程度上是因为他们有一些根深蒂固的好习惯，而且不断在细节上精益求精。

ThoughtWorks 中国公司的几位同事一起参与翻译本书，这也正是为了把我们的经验分享给更多人。

从注意每天的细节开始，让自己成为一个高效的程序员，其实每个人都能做到。

熊节

ThoughtWorks 咨询师

2008 年 11 月 17 日

## 作者简介

---

作为一名软件架构师与意见领袖，**Neal Ford** 供职于 ThoughtWorks（一家专注于端到端软件开发与交付的跨国 IT 咨询公司）。在加入 ThoughtWorks 之前，Neal 是 The DSW Group, Ltd. 的技术总监——这是一家在美国还算有名的培训与软件开发公司。Neal 毕业于乔治亚州立大学，他拥有计算机科学的学位，专攻语言与编译器；同时他还辅修数学，专攻统计分析。现在他是一名软件设计师和开发者，此外也编撰培训材料、杂志文章和视频演讲，他还是几本图书的作者，包括《Developing with Delphi: Object-Oriented Techniques》（由 Prentice-Hall 出版）、《Jbuilder 3 Unleashed》（由 Sams 出版）和《Art of Java Web Development》（由 Manning 出版）等。他曾担任 2006 和 2007 版《No Fluff, Just Stuff 文选》（Pragmatic Bookshelf）的编辑和作者。他擅长的编程语言包括 Java、C#/ .NET、Ruby、Groovy、函数式语言、Scheme、Object Pascal、C++ 和 C 等。他的咨询工作主要针对大规模企业应用的设计和开发。Neal 曾在美国和其他国家进行现场授课，客户包括军方和很多世界 500 强的企业。作为演讲者，他同样在全球享有盛名，曾在世界各地举办的各种大型开发者会议上发表超过 600 场演讲。如果有兴趣了解更多关于 Neal 的信息，请访问他的网站：<http://www.nealford.com>。他也希望得到读者的反馈，他的邮件地址是 [nford@thoughtworks.com](mailto:nford@thoughtworks.com)。

## 封面介绍

---

封面图像是 Corbis 的素材照片。



# 目录

序言	1
前言	3
<b>第1章 概述</b>	<b>9</b>
为什么要写一本关于程序员生产率的书	10
本书涵盖的内容	12
如何读本书	14
<b>第一部分 机制</b>	
<hr/>	
<b>第2章 加速法则</b>	<b>17</b>
启动面板	18
加速器	27
宏	42
小结	44
<b>第3章 专注法则</b>	<b>45</b>
排除干扰	46
搜索优于导航	48
找出难找的目标	50
使用有根视图	52
设好“粘性属性”	54
使用基于项目的快捷方式	55
使用多显示器	56
用虚拟桌面拆分工作空间	56
小结	58
<b>第4章 自动化法则</b>	<b>59</b>
不要重新发明轮子	61
建立本地缓存	61

自动访问网站	62
与 RSS 源交互	63
在构建之外使用 Ant	64
用 Rake 执行常见任务	65
用 Selenium 浏览网页	67
用 bash 统计异常数	67
用 Windows Power Shell 替代批处理文件	69
用 Mac OS X 的 Automator 来删除过时的下载文件	70
驯服 Subversion 命令行	72
用 Ruby 编写 SQL 拆分工具	73
我应该把它自动化吗	74
别给牦牛剪毛	76
小结	76
<b>第 5 章 规范性法则</b>	79
DRY 版本控制	80
使用标准的构建服务器	82
间接机制	83
利用虚拟平台	90
DRY 阻抗失配	91
DRY 文档	99
小结	105
<b>第二部分 实践</b>	
<b>第 6 章 测试驱动设计</b>	109
不断演化的测试	111
代码覆盖率	118
<b>第 7 章 静态分析</b>	121
字节码分析	122
源代码分析	124
用 Panopticode 生成统计数据	125
动态语言分析	128

<b>第 8 章 当个好公民</b>	131
破坏封装	132
构造函数	133
静态方法	134
犯罪行为	138
<b>第 9 章 YAGNI</b>	141
<b>第 10 章 古代哲人</b>	147
亚里斯多德的“事物的本质性质和附属性质”理论	148
奥卡姆剃刀原理	149
笛米特法则	153
“古老的”软件学说	154
<b>第 11 章 质疑权威</b>	157
愤怒的猴子	158
连贯接口	159
反目标	162
<b>第 12 章 元编程</b>	163
Java 和反射	164
用 Groovy 测试 Java	166
编写连贯接口	167
元编程的归处	168
<b>第 13 章 组合方法和 SLAP</b>	171
组合方法实践	172
SLAP	176
<b>第 14 章 多语言编程</b>	181
历史与现状	182
路在何方	185
Ola 的金字塔	190
<b>第 15 章 寻找完美工具</b>	193
寻找完美编辑器	194
编辑器参考列表	197

181	选择正确的工具	198
182	丢弃错误的工具	204
<b>第 16 章 结束语：继续对话</b>		
<b>附录 构建块</b>		
187		
191		
194		
198		
201		
204		
207		
209		
211		
214		
218		
248		
288		
294		
297		
298		
299		
301		
308		
311		
318		
321		
328		
331		
338		
341		
348		
351		
358		
361		
368		
371		
378		
381		
388		
391		
398		
401		
408		
411		
418		
421		
428		
431		
438		
441		
448		
451		
458		
461		
468		
471		
478		
481		
488		
491		
498		
501		

# 序言

在我们这个行业里，不同程序员的个人生产率可谓判若云泥——大多数人也许要花一周时间才能干完的活，有些人一天之内就搞定了。这是为什么？简单来说，这些程序员比大多数同行掌握了更多趁手的工具。说得更明白一点，他们真正了解各种工具的功用，并且掌握了使用这些工具所需的思维方式。这些“卓有成效的程序员”的秘密是采用了正确的方法论与原理，而 Neal 在本书中准确地捕捉到了这种神秘的东西。

2005 年的某日，在去机场的车上，Neal 问我：“你认为这个世界需要再多一本关于正则表达式的书吗？”然后话题就变成了“我们希望有什么样的书”，并从此播下了你手上这本书的种子。回望自己的职业生涯中从“好程序员”跃升为“卓有成效的程序员”的那个阶段，思索当时的情景和前因后果，我这样说道：“书名我还没想好，不过副标题应该叫‘用命令行作为集成开发环境’。”那时我把自己的生产率提升归功于使用 bash shell 带来的加速，但这并不是全部，更重要的是我对这些工具更加熟悉，我无须思索怎么完成一些日常工作，而是自然而然地就把它们做好。我们还花了一些时间讨论过度生产（译注）以及控制这种情况的办法。几年以后，在经过无数次的私下讨论，以及围绕这个主题做了一系列演讲之后，Neal 的大作终于得以付梓了。

在《Programming Perl》(O'Reilly 出版)一书中，Larry Wall 说到“懒惰、傲慢和缺乏耐性”是程序员的三大美德。懒惰，因为你一直致力于减少需要完成的工作总量；缺乏耐性，因为一旦让你浪费时间去做本该计算机做的事，你就会怒不可遏；还有傲慢，因为被荣誉感冲昏头的你会把程序写得让谁都挑不出毛病来。本书不会使用这几个字眼（我已经用 grep 检查过了），但你会发现同样的理念在本书的内容中得到了继承和发扬。

曾经有那么几本书，它们影响了我的职业生涯，甚至改变了我看待这个世界的方式。说实话，我真地希望早 10 年看到这本书，因为我确信它会给读者带来极其深远的影响。

David Bock  
首席咨询师  
CodeSherpas

---

译注：过度生产(hyperproductivity)是指在高效的工具和工作流程之下工作的工人得不到休息而过度疲劳、压力过大的情况。





很多年前，我曾经给一些有经验的软件开发人员上课，教他们学习新的技术（例如Java等）。这些学生之间生产率的差异一直让我感到惊讶：有些人的效率能比另一些人高出几个数量级——而且这还不是指他们使用工具的过程，而是他们与计算机之间的一般交互。我曾经跟同事开玩笑说，这个班上有些人的电脑压根不是在跑（running），简直就是在散步（walking）。这自然让我开始反思自己的生产率：我有没有让手边的（正在跑或在走的）这台电脑物尽其用？

那以后又过了几年，David Bock和我谈论起这件事。很多比较年轻的同事从来就没有认真用过命令行工具，自然也就无法理解为何这些工具能比时下那些漂亮的IDE还要高效。正如David在序言里说的，我们讨论这个问题并决定要写一本关于“高效使用命令行”的书。我们联系了出版商，然后开始从朋友、同事那里搜集各种各样的命令行“巫术”。

随后又发生了几件事：David创办了他自己的咨询公司，他的孩子也呱呱坠地——三胞胎！所以，显然已经有足够多的事情让David焦头烂额了。与此同时，我也明白了一件事：一本单纯讲述命令行技巧的书很可能会成为有史以来最乏味的书。差不多就在那个时候，我在班加罗尔的一个项目里工作，和我结对编程的搭档Mujir和我聊起代码中的模式以及如何识别这些模式。如同醍醐灌顶一般，我突然意识到在自己搜集的所有技巧中都可以看到模式的踪影。我真正想要介绍的不是一堆命令行技巧，而是那些使得软件开发者们卓有成效的模式。于是，就有了你手中的这本书。

## 本书的目标读者

这不是一本帮助最终用户更有效使用计算机的书。这是一本写给程序员的、关于如何提高生产率的书，这意味着我可以对读者作很多假设，很多基本概念也不需要浪费很多时间去解释，因为软件开发者是极其强大的计算机用户。当然，没有技术背景的用户也应该能够从本书（尤其是它的第一部分）中学到一些东西，但我的目标读者是软件开发者。

本书没有明确指定阅读顺序，所以尽情地随性翻阅吧，当然如果你喜欢从头读到尾也没有问题。书中的各个主题之间只有少许有意的关联，所以尽管从头读到尾的方式会略有优势，但还不足以成为阅读本书的不二法门。

## 体例

本书使用下列排版样式：

*斜体 (Italic)*

用于新名词、URL、邮件地址、文件名和文件扩展名。



等宽字体 (`Constant width`)

用于程序代码，同时也包括在段落正文中引用的程序元素，例如变量名、函数名、数据库、数据类型、环境变量、语句、关键字等。

加粗的等宽字体 (`Constant width bold`)

用于命令或其他需要由用户来输入的文字。

加斜体的等宽字体 (`Constant width italic`)

表示这里的文本应该替换为用户提供的值或是根据上下文决定的值。

## 使用代码示例

本书是为了帮助你完成工作而写的。一般而言，你可以在自己的程序或者文档中使用本书中的代码而无须征求我们的许可，除非你打算重新发布其中很大部分的代码。例如，在编程时用到书中的几段代码不需要征求许可，然而把O'Reilly书中的代码示例做成光盘销售或者发行就需要征求许可；引用本书中的代码示例来解答别人的问题不需要征求许可，然而把大量出自本书的示例代码放进你自己的产品文档就需要征求许可。

如果你在引用本书内容时注明它的版权，我们会非常感激，当然这不是必须的要求。一本书的版权信息通常包括书名、作者、出版社和ISBN编号，例如“《The Productive Programmer》 by Neal Ford. Copyright 2008 Neal Ford, 978-0-596-51978-0”。

如果你以别的方式需要使用本书中的代码示例，请通过 [permissions@oreilly.com](mailto:permissions@oreilly.com) 与我们联系。

## 如何联系我们

关于本书的评论和疑问都可以发送给出版社：

美国：

O'Reilly Media, Inc.  
1005 Gravenstein Highway North  
Sebastopol, CA 95472

中国：

北京市西城区西直门南大街2号成铭大厦C座807室 (100035)  
奥莱利技术咨询(北京)有限公司