

PERL HACKS™

100个业界最尖端的技巧和工具



O'REILLY®

 HACKS
SERIES™

chromatic 著
刘颖 译



清华大学出版社

PERL HACKS™

—— 100 个业界最尖端的技巧和工具



chromatic 著

刘颖 译

O'REILLY®

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

O'Reilly Media, Inc. 授权清华大学出版社出版

清华大学出版社

Copyright ©2006 by O'Reilly Media, Inc.

Authorized Simplified Chinese translation edition, by O'Reilly Media, Inc., is published by Tsinghua University Press, 2006. Authorized translation of the original English edition, 2006 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.

本书之英文原版由 O'Reilly Media, Inc. 于 2006 年出版。

本中文简体翻译版由 O'Reilly Media, Inc. 授权清华大学出版社于 2006 年出版。此翻译版的出版和销售得到出版权和销售权的所有者——O'Reilly Media, Inc. 的许可。

版权所有, 未经书面许可, 本书的任何部分和全部不得以任何形式复制。

北京市版权局著作权合同登记

图字: 01-2006-7217 号

本书封面贴有清华大学出版社防伪标签, 无标签者不得销售。

版权所有, 侵权必究。侵权举报电话: 010-62782989 13501256678 13801310933

图书在版编目 (CIP) 数据

PERL HACKS™: 100 个业界最尖端的技巧和工具 / (美) 克罗曼蒂克 (chromatic) 著; 刘颖译. —北京: 清华大学出版社, 2007.1

书名原文: Perl Hacks™

ISBN 978-7-302-14218-8

I. P… II. ①克… ②刘… III. PERL 语言—程序设计 IV. TP312

中国版本图书馆 CIP 数据核字 (2006) 第 145031 号

责任编辑: 常晓波

封面设计: Linda Palo. 张健

责任校对: 张 剑

责任印制: 王秀菊

出版发行: 清华大学出版社 地 址: 北京清华大学学研大厦 A 座

<http://www.tup.com.cn> 邮 编: 100084

c-service@tup.tsinghua.edu.cn

社 总 机: 010-62770175 邮购热线: 010-62786544

投稿咨询: 010-62772015 客户服务: 010-62776969

印 刷 者: 北京市清华园胶印厂

装 订 者: 三河市春园印刷有限公司

经 销: 全国新华书店

开 本: 152 毫米 × 227 毫米 19 印张 356 千字

版 次: 2007 年 1 月第 1 版 印 次: 2007 年 1 月第 1 次印刷

印 数: 1 ~ 3000 册

定 价: 38.00 元 (册)

本书如存在文字不清、漏印、缺页、倒页、脱页等印装质量问题, 请与清华大学出版社出版部联系调换。联系电话: (010)62770177 转 3103 产品编号: 023982-01

O'Reilly Media, Inc. 介绍

为了满足读者对网络 and 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. 授权清华大学出版社, 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, 同时也是联机出版的先锋。

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, 而现在则编写著作, 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。

目录

致谢	1
前言	7
第 1 章 生产率 Hack.....	13
1 为 Firefox 增加 CPAN 快捷方式	13
2 工作中使用 Perldoc	16
3 在线浏览 Perl 文档	18
4 使用 shell 别名	20
5 在 Vim 里自动完成 Perl 标识符	23
6 使用 Perl 最佳的 Emacs 模式	26
7 强化本地风格	28
8 不要保存不合格的 Perl	31
9 自动提交代码复验	35
10 在 Vim 里运行测试	37
11 从 Emacs 运行 Perl.....	39
第 2 章 用户交互	41
12 使用 \$EDITOR 作为用户界面	41
13 在命令行中正确交互	43

14	简化终端交互	45
15	使 Mac 发出警告信息	50
16	与图形应用程序交互	53
17	收集配置信息	59
18	重写 Web	62
 第 3 章 数据解析		66
19	把文件看成数组	66
20	反向阅读文件	68
21	用任意电子表格作为数据源	69
22	抽取数据库代码特性	74
23	构建 SQL 库	77
24	不用 SQL 动态查询数据库	78
25	绑定数据库列	80
26	迭代并产生重要数据	81
27	从迭代器里抽取多个值	83
 第 4 章 使用模块		87
28	缩短过长的类名	87
29	处理模块路径	88
30	重新加载修改过的模块	90
31	创建个人模块包	92
32	管理模块安装	94
33	预解决模块路径	97
34	创建标准模块工具箱	99
35	依据指南编写样本程序	102
36	从外部修改出错代码	104
37	与 CPAN 干杯	106
38	改进异常条件	107
39	本地搜索 CPAN 模块	110
40	打包单一模式的 Perl 应用程序	114

41 创建属于自己的语法警告	117
42 找到并报告模块 bug	119
第 5 章 对象 Hack	124
43 彻底转变对象	124
44 自由串行化（绝大多数）对象	127
45 添加特性信息	129
46 使得方法真正私有化	131
47 自动删除方法参数	135
48 控制远程对象的访问	138
49 使得对象真正多态	141
50 自动生成访问器	144
第 6 章 调试	148
51 快速定位编译错误	148
52 使得不可见的符号变得可见	150
53 用测试用例实现调试	152
54 利用注释调试	154
55 在错误中显示源码	158
56 不解析匿名函数	160
57 命名匿名子例程	163
58 找到子例程的源	165
59 定制调试器	166
第 7 章 开发技巧	170
60 重建发布	170
61 规范化测试	172
62 隔离开发者和用户测试	175
63 自动运行测试	178
64 查看测试错误诊断——用颜色！	180
65 测试动态代码	182

66	窃取基准	185
67	构建属于自己的 Perl	187
68	持久稳固地运行测试套装	189
69	在测试里仿真不利环境	194
 第 8 章 了解自己的代码		199
70	理解何时发生了什么	199
71	检查数据结构	203
72	安全找到函数	205
73	知道什么是核心并且发生在何时	207
74	跟踪所有用过的模块	209
75	找到包内的所有符号	213
76	偷窥内部闭包	215
77	找到所有全局变量	218
78	深入子例程	220
79	找到已导入的函数	224
80	概述程序大小	226
81	重用 Perl 进程	229
82	跟踪自己的 Op	230
83	编写属于自己的警告	233
 第 9 章 扩展 Perl Foo		237
84	用 dualvar 加倍数据	237
85	用真实的引用代替软引用	239
86	优化去除烦人的代码	241
87	锁定散列	243
88	清理干净作用域边缘	244
89	以奇异的方式调用函数	246
90	glob 序列	253
91	编写尽量少的错误检查代码	255

92	返回更精确的值	257
93	返回活动的值	261
94	添加属于自己的 Perl 语法	265
95	用源滤波器修改语义	268
96	使用没有 XS 的共享库	271
97	在单一 TCP 端口运行两种服务	273
98	改进调度表	277
99	跟踪自己的近似值	281
100	重载运算符	284
101	向困惑学习	289

致谢

关于作者

chromatic 为 O'Reilly Media 工作，他负责编辑 O'Reilly 网络、自由开放源码 Web 站点。在他的空闲时间，负责编写书籍，比如本书。而余下来的时间则奉献给了 CPAN、Perl 5、Perl 6、Parrot 甚至 Pugs。他住在 Oregon 的 Portland 西部，附近有一个公园和一条小溪，他希望某天能完成更多的项目，包括写一部小说、一本连环漫画册、一部电视剧本，然后好好地休息。

Dr.Damian Conway 是一位专业的 Perl 技术专家。作为许多广为流传的 CPAN 模块 (<http://search.cpan.org/~dconway>) 以及两本获得高度赞誉的 Perl 方面书籍的作者，他还是知识广博、很受欢迎的会议演说家，并且经营着一个国际 IT 培训公司——Thoughtstream (<http://damian.conway.org>)，这里向欧洲、北美以及澳大利亚提供面向初学者到大师级人物的 Perl 培训。他花费大量的时间与 Larry Wall 一起设计 Perl 6。他编写了 [Hack #5]、[Hack #13]、[Hack #14]、[Hack #34]、[Hack #54]、[Hack #86]、[Hack #90]、[Hack #92]、[Hack #93]、[Hack #94]、[Hack #95]、[Hack #99]、[Hack #100]。

本世纪初，Curtis “Ovid” Poe 显示了其不同寻常的时代感，从主机转为用 Perl 进行 Web 编程，并且敏锐地预测到网络工业的突飞猛进。不谈他次要的挫折以及为几个现在已经不存在的公司的工作经历，Ovid 坚持使用 Perl 并真正爱上了 Perl。作为用户团体、CPAN 模块作者以及受欢迎的 Perl CGI 课程中知名的演说家，Ovid 还是 Perl 基金指导委员会成员并且也是 TPF 委员会的领袖。他编写了 [Hack #10]、[Hack #31]、[Hack #50]、[Hack #56]、[Hack #57]、[Hack #58]、[Hack #60]。

其他贡献者

Perl 也是一门有丰富多彩的社区的语言, 该社区里有专家、向导、高手、东郭先生以及什么也不做的人。这样形形色色的人物中也有一些对本书的编写做出了贡献。

Adrian Howard 仍然希望 Lisp 和 Smalltalk 能够接管整个世界, 同时也能在其他事情上从使用 Perl 和 Ruby 中受益。灵敏性的追随者。测试的推崇者。实用性的狂热者。近期很是内疚成为 Ovid 的 Perl 管理人员。正在为能够买到和爱人 Kathryn 以及一小队忠实的约克郡犬共享生活的小型城堡而努力储蓄。他编写了 [Hack #15]、[Hack #63]。

Chris Dolan 是生活在 Wisconsin 州 Madison 的软件开发人员。曾获得天文学的博士学位, 他拥有极其扎实的数学和理科背景。早在 20 世纪 80 年代末还十几岁时他就开始了专业的编程生涯。在空闲时间里, 他是多个在线软件开发社团的积极参与者, 并且是一个活跃的自行车手。他编写了 [Hack #40]。

David Landgren 在 awk 无法满足处理事情的需求时开始使用 Perl 4.019。在 1999 年, 他成立了 Paris Perl Mongers 并且参与组织 YAPC::Europe 2003 会议。目前在一个大型法国人力资源公司工作, 他喜欢被称做 IT 业务经理, 在哪儿都离不开 Perl。他用其少得可怜的空闲时间编写 Perl 5 的概述; porter 的邮件发送清单。可以通过 david@landgren.net 联系他。他编写了 [Hack #98]。

David Wheeler, 一位资深 Perl hacker, 是 Bricolage 目录管理系统的首席开发人员, 并且是软件公司 Kineticcode 的主席和创始人 (<http://www.kineticcode.com>)。他在 Bricolage 的成就使得他花费整个去年开发 Kinetic 平台, 这是一个下一代企业 Web 服务应用程序平台。他住在 Oregon 州 Portland。他编写了 [Hack #7]、[Hack #28]。

Guillaume Cottenceau 是一位软件开发人员。他对计算机开始产生兴趣的年代已经极其久远以致无法说出, 喜欢用不同工具不同语言来编程, 并且很幸运地成为 Perl 编写的一个游戏的催化剂, 该游戏名为 “Frozen-Bubble”, 被公认为 Linux 社区最受欢迎的游戏。他编写了 [Hack #16]。

H. Merijn Brand 是 Perl 5 porter 以及 Amsterdam 的 Perl 推广者。他生于 1961 年 12 月 30 日, 已婚, 有两个小孩。他的工作从 1991 年 6 月开始就是荷兰的

PROCURA B.V. 的数据分析家。可以访问他的主页 <http://mirrors.develooper.com/hpux>。他编写了[Hack #21]。

Jesse Vincent 并不经常喝酒，但是却发现一杯或两杯酒有助于使 CPAN 更加引人注目。他编写了[Hack #37]。

Joe McMahon 是 Yahoo! 的测试自动化架构师，他的背景包括广博的开发经验：在 NASA 工作了 25 年，涉及从太空船地面通信系统到 Web 开发。他贡献了核心 Perl 测试支持，充实了调试器的内部文档并且支持一些 CPAN 模块，从 App::SimpleScan 以及 WWW::Mechanize::Pluggable 到 Date::PeriodParser 以及 GraphViz::Data::Structure。他编写了[Hack #59]。

Joshua ben Jore 因为必须为某政府活动编写 CGI 应用程序而使用 Perl，他曾经听闻 Perl 是完成这类工作的最佳选择。当他亲自体验了 CPAN 有多么惊人后就一直用 Perl 了。他的最新项目是得到 Lispy 宏以及 Perl 的懒惰性评价，并且知道了 Prolog 必须提供什么。他在 perlmonks.org 上众人皆知，被公认为 Morris 的舞者。他编写了[Hack #83]。

Michael Graham 是 Perl 程序员，住在 Toronto，他用 CGI::Application 和其丰富的插件套装编写 Web 应用程序。他为 Wilder & Associates 工作，喜欢音乐，喜欢在海滩上散步，也喜欢测试驱动的开发。他编写了[Hack #68]、[Hack #69]。

Philippe "BooK" Bruhat 是 HTTP::Proxy 和 Net::Proxy 的作者，与他的妻子还有猫生活在法国的 Lyon，是法国以及欧洲 Perl 社区的活跃成员。他在法国的 O'Reilly 致力于 Programming Perl, 3rd Edition 和 Perl Best Practices 的翻译，并且在 GNU/Linux Magazine France 上发表了 Perl 论文，协助组织了欧洲 Perl 研讨会和会议，也是名合格的演讲者。如果更多人使用他的 Acme::MetaSyntactic 模块（每周更新），他一定会相当开心。他编写了[Hack #18]、[Hack #97]。

Ricardo Signes 是追求完美的人，有很强的纪律性以及对称感。他编写了两本 O'Reilly 书籍：Perl and LWP 和 RTF Pocket Guide，为 CPAN 奉献了很多模块，并且是 Perl Journal 长达 5 年的正式专栏作家。（他的文章出现在多本 O'Reilly 的 Best of the Perl Journal 图书中，比如 Games、Diversions & Perl Culture volume。）曾经是个语言学家，现在住在 Alaska 东南的岛上，

在那里，他开发可以保护本机语言的工具。他的Web站点是狂热与幽默并存之处 <http://interglacial.com>。他编写了[Hack #11]、[Hack #6]、[Hack #3]。

Simon Wistow 正在整合 London.pm，这是所有 Perl Monger 组织中最重大且讨论最热烈的地方。在空余时间里，他编写了无数用途尚未确定的 CPAN 模块。他编写了[Hack #93]。

Stephen B. Jenkins（也叫 Erudil）是加拿大国家研究中心宇航研究所空气动力学实验室的资深程序员/分析家。他曾为 Dr. Dobbs' Journal、Linux Journal、The Perl Journal、USENIX 协会、美国航天航空研究所、IEEE 的航空宇宙及电子学系统社区以及计算机协会撰稿。更多信息见 <http://www.Erudil.com>。他编写了[Hack #101]。

Tim Allwine 是 O'Reilly Media, Inc 的高级软件工程师。他为市场调查组织开发软件，编写了多种用来从不同站点收集数据的工具，并且涉及 O'Reilly 的 Web 服务的开发。他编写了[Hack #71]。

Tom Moertel 是 Pittsburgh Perl Mongers 的长期贡献者。他创造了 Test::LectroTes（一个 Perl 的基于说明的测试系统）和 PXSL（“pixel”，一个便于编写 XML 文档的快捷方法）。他写了 The Coder's Guide to Coffee 这本书，这是针对程序员最容易误解的实用程序的简明向导。他住在 Pennsylvania 州的 Pittsburgh，在这里，他经营一家咨询公司，专攻解决疑难软件问题。从站点 <http://blog.moertel.com> 可以了解其项目和兴趣，包括浓咖啡以及功能性编程思想。他编写了[Hack #61]。

Brett Warden, BSEE，擅长 Perl 编程以及嵌入式/Telco 系统。他和妻儿一起生活在美国西北。可以在 <http://www.wgz.com/bwarden> 站点找到一系列奇特的项目。他在 Perl Monks 的用户名是 isotope。他编写了[Hack #42]。

致谢

笔者感谢编辑 Allison Randal，谢谢他领导整个出书过程，处理了所有编写之外的细节，并且全局协调 O'Reilly Media 的生产部、市场部、营业部，使得这本书最终由简单的纯文本变成了读者手中现在拿着的印刷精美的书籍。

同时感谢 Nathan Torkington 对于启动该项目所做出的帮助。

chromatic

首先感谢合作者，两个优秀的程序员，也是我的大学同学和朋友，以及所有为本书做出贡献者，没有你们，这本书就不可能像现在这样丰富、有趣。同时感谢我的朋友，家人和副手，尤其是 U of P 的全体工作人员：Aleatha、Esther、Eva、Kate、Katie、Laura 以及 Mel。感谢所有与我共同工作和分享爱好的同事，包括所有的程序员，正因为你们创造并优化了很多工具才使我们能够更好地工作。

最后，将这本书献给我的家人：Brett、Danielle、Jacob、Jack 和 Brad，希望我能成为你们值得骄傲的兄长、姐夫、叔叔和猫的守护者。

Damian Conway

谢谢 chromatic、Allison 和 Net 邀请我和他们一起构建这座奇妙的全新 Perl 丛林中的体育馆。谢谢 Larry Wall 创造出如此非凡的运动场。也谢谢所有来自 Perl 社区的朋友，正是你们的鼓励和支持才使得我坚持到最后。

最重要的是，谢谢我亲爱的 Linda，你的可爱、美丽和高雅照亮我生活的每一天。

Curtis "Ovid" Poe

多年前我初学 Perl 时，如果没有 Perl 社区耐心解答我那么多的愚蠢问题，那么今天我绝对不可能能够参与到这本书的编写中。他们不仅仅教会了我 Perl，而且教会了我如何成为一名程序员。非常感谢 chromatic 和 Allison 收录我的想法，我永远不会忘怀。我也要谢谢 Sean 和 Lil，没有你们，生活将不再美好。

技术审编

作者向技术审编们致以最真诚的谢意，正是他们发现了很多错误，提出了很多改动建议，并讨论决定重要的格式细节，这才使得本书更加清晰、精确而有趣。他们是 Ann Barcomb、Daniel Bosold、Brad Bowman、Philippe "Book" Bruhat、#cgiapp IRC channel、Chris Dolan、Michael Graham、Garick Hamlin、Joshua ben Jore、Yves Orton、Ricardo Signes 以及 David Wheeler。

前言

Perl是一门拥有丰富且极具表现力词汇的语言。自从1987年公开源码以来，它已经从“快餐”性的摘要及报告语言发展成Web编程、数据解析、GUI构建、自动粘合及成熟的应用程序开发语言。它是因特网的传送带和瑞士军刀。

就像传送带和瑞士军刀一样，Perl几乎可以完成所能想象并的确想做的任何事情。

如果只是想尽快完成工作，那么可以在编写出所知道的最简易的Perl代码后就继续其他工作。如果想构建大型应用程序，则可以在一些经验和少许规则下完成。如果不介意寻求帮助来解决问题，CPAN则时刻准备帮忙。

所有这一切都十分高效，并且这样的高效性能够一直保持至工作完成。另一方面，Perl也是有趣的。

想象有一窝小猫，它们在地板上翻着跟头，用牙齿、爪子、皮毛外加轻微的咆哮玩着球。显然，它们正在玩耍，但是它们同时也在练习着于残酷野生世界中生存所需的技能。它们小心翼翼不伤害到对方，其中更聪明小猫的战略和令人惊讶的发挥也将是其他小猫有价值的一课。

是什么成就Perl编程高手呢？部分是由于知识，但更主要的是那种与语言打交道、发现惊喜甚至发明属于自己的东西的好奇心。这正是本书编写过程中充满乐趣的原因所在。一些世上最好的Perl程序员总结出101个提示、窍门和技巧。其中有些是能立即见效的。有些是别无其他选择时的偏招。大部分

则具备两方面：所需立刻解决的问题和可适用于其他情况的深入细微的技巧。所有这些都值得我们去学习。

高效率是大家都喜闻乐见的。这是我们选用Perl编程的原因。再加上学习的乐趣，尤其是拥有这些花费Perl编程高手多年努力学习的课程，我们将足以应付一切。让朋友们刮目相看。让同事们大吃一惊。步入代码、规范的丛林，让自己拥有顾客所要求的解决任何问题的足够自信。

一定可以。

为什么学习 Perl Hack

Hack 这个术语在新闻中已经有个坏的名声。他们用它来指那些入侵系统或者用电脑作为武器大肆破坏的人。可是，在写代码的人当中，hack这个词是指一个问题的“快餐”解决方案或者做事的聪明方式。而且hacker这个词是一种非常高的赞誉，是指那些充满创造力的、拥有技术实力做事的人。O'Reilly 的 Hack 系列试图找回这个词的本意，说明hacking存在好的方式，传递hacker创造性和分享性的道德规范给那些刚刚入行的人。看别人怎么处理系统和问题通常是学习新技术最为快捷的方式。

而且这也十分有趣。

当然，没有一本书能完整说明可借助Perl达到的所有乐趣、创造性和强烈感觉。但是我们希望这本书可以让你正确思考并hack自己的疯狂想法。

如何使用本书

我们根据不同的话题来划分章节，而不是根据相对的难度。翻看浏览此书，你会发现吸引你注意力的有趣标题或段落，阅读它们！在可能的地方我们加入了与文中hack相关的参照。比如，如果你和同事正在一个慵懒的星期五下午享受忙碌后的闲暇时光，那么就从“与CPAN干杯”（Hack #37）开始吧。

本书组织结构

第1章：生产率Hack

接受它。当坐下来编程时，会有一些苦恼。它可能是重复的命令、文档