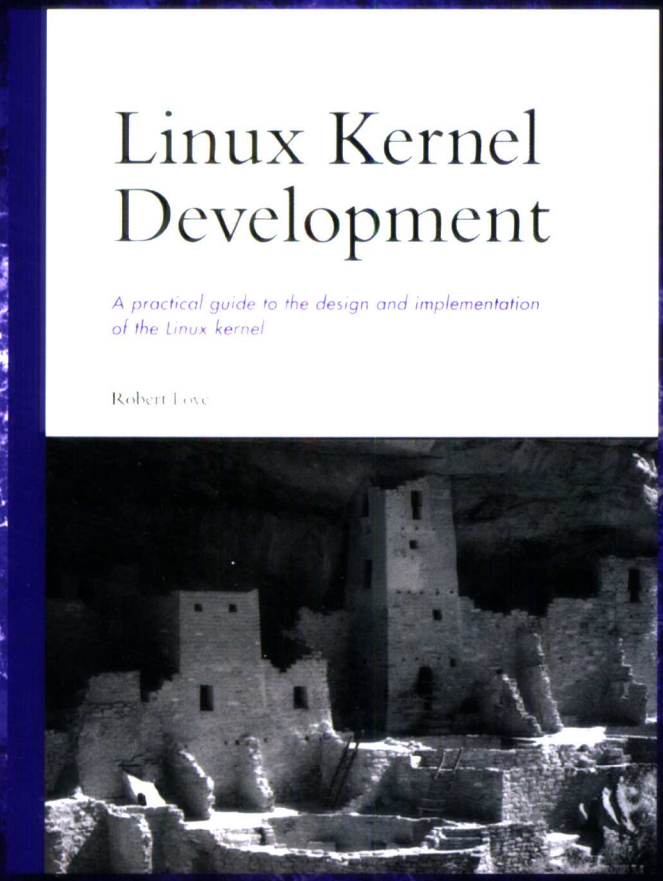
SAMS 计 算 机 科 学 丛 书

Linux内核设计与实现

(美) Robert Love 著 陈莉君 康华 张波 译



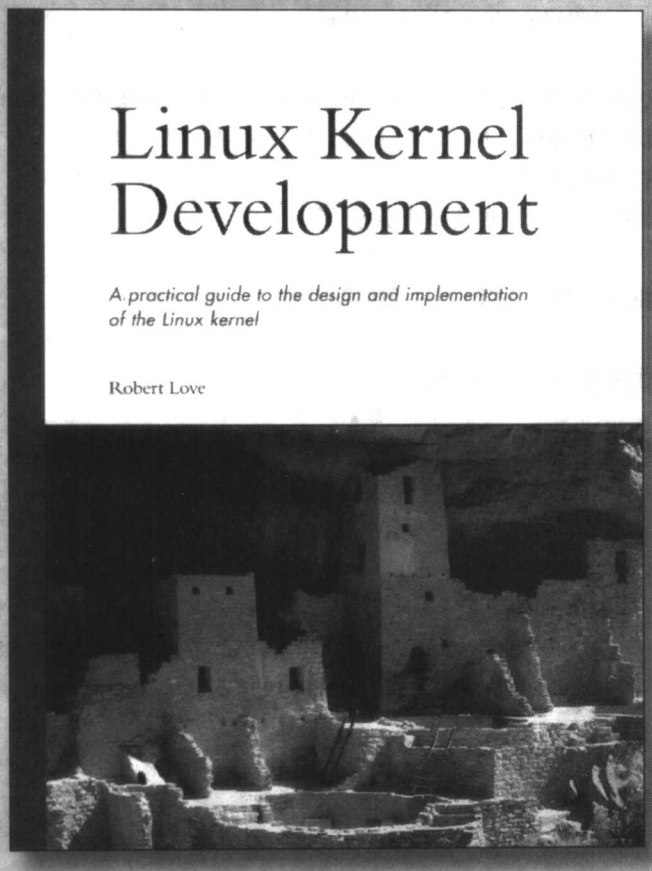
Linux Kernel Development

机械工业出版社
China Machine Press

计 算 机 科 学 丛 书

Linux内核设计与实现

(美) Robert Love 著 陈莉君 康华 张波 译



Linux Kernel Development

机械工业出版社
China Machine Press

本书提供Linux内核设计和实现的概述性信息，覆盖了从核心内核系统的应用到内核设计与实现等各方面内容，能够带领读者快速走进Linux内核世界。本书不但介绍了理论，而且也讨论了具体应用，可以满足不同读者的需要，适合于各类希望理解Linux内核软件开发的读者。

本书版权登记号：图字：01-2003-8103

图书在版编目（CIP）数据

Linux内核设计与实现/（美）勒伏（Love, R.）著；陈莉君等译.-北京：机械工业出版社，2004.11

（计算机科学丛书）

书名原文：Linux Kernel Development

ISBN 7-111-15241-7

Ⅰ.L… Ⅱ.①勒…②陈… Ⅲ.Linux操作系统-程序设计 Ⅳ.TP316.89

中国版本图书馆CIP数据核字（2004）第094927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：华　章

北京牛山世兴印刷厂印刷·新华书店北京发行所发行

2004年11月第1版第1次印刷

787mm×1092mm 1/16·17.5印张

印数：0 001-5000册

定价：35.00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

出版者的话

文艺复兴以降，源远流长的科学精神和逐步形成的学术规范，使西方国家在自然科学的各个领域取得了垄断性的优势；也正是这样的传统，使美国在信息技术发展的六十多年间名家辈出、独领风骚。在商业化的进程中，美国的产业界与教育界越来越紧密地结合，计算机学科中的许多泰山北斗同时身处科研和教学的最前线，由此而产生的经典科学著作，不仅奠划了研究的范畴，还揭橥了学术的源变，既遵循学术规范，又自有学者个性，其价值并不会因年月的流逝而减退。

近年，在全球信息化大潮的推动下，我国的计算机产业发展迅猛，对专业人才的需求日益迫切。这对计算机教育界和出版界都既是机遇，也是挑战；而专业教材的建设在教育战略上显得举足轻重。在我国信息技术发展时间较短、从业人员较少的现状下，美国等发达国家在其计算机科学发展的几十年间积淀的经典教材仍有许多值得借鉴之处。因此，引进一批国外优秀计算机教材将对我国计算机教育事业的发展起积极的推动作用，也是与世界接轨、建设真正的世界一流大学的必由之路。

机械工业出版社华章图文信息有限公司较早意识到“出版要为教育服务”。自1998年开始，华章公司就将工作重点放在了遴选、移译国外优秀教材上。经过几年的不懈努力，我们与Prentice Hall、Addison-Wesley、McGraw-Hill、Morgan Kaufmann等世界著名出版公司建立了良好的合作关系，从它们现有的数百种教材中甄选出Tanenbaum、Stroustrup、Kernighan、Jim Gray等大师名家的一批经典作品，以“计算机科学丛书”为总称出版，供读者学习、研究及收藏。大理石纹理的封面，也正体现了这套丛书的品位和格调。

“计算机科学丛书”的出版工作得到了国内外学者的鼎力襄助，国内的专家不仅提供了中肯的选题指导，还不辞劳苦地担任了翻译和审校的工作；而原书的作者也相当关注其作品在中国的传播，有的还专诚为其书的中译本作序。迄今，“计算机科学丛书”已经出版了近百个品种，这些书籍在读者中树立了良好的口碑，并被许多高校采用为正式教材和参考书籍，为进一步推广与发展打下了坚实的基础。

随着学科建设的初步完善和教材改革的逐渐深化，教育界对国外计算机教材的需求和应用都步入了一个新的阶段。为此，华章公司将加大引进教材的力度，在“华章教育”的总规划之下出版三个系列的计算机教材：除“计算机科学丛书”之外，对影印版的教材，则单独开辟出“经典原版书库”；同时，引进全美通行的教学辅导书“Schaum's Outlines”系列组成“全美经典学习指导系列”。为了保证这三套丛书的权威性，同时也为了更好地为学校和老师们服务，华章公司聘请了中国科学院、北京大学、清华大学、国防科技大学、复旦大学、上海交通大学、南京大学、浙江大学、中国科技大学、哈尔滨工业大学、西安交通大学、中国人民大学、北京航空航天大学、北京邮电大学、中山大学、解放军理工大学、郑州大学、湖北工学院、中国国家信息安全测评认证中心等国内重点大学和科研机构在计算机的各个领域的著名学者组成“专家指导委员会”，为我们提供选题意见和出版监督。

这三套丛书是响应教育部提出的使用外版教材的号召，为国内高校的计算机及相关专业

的教学度身订造的。其中许多教材均已为M. I. T., Stanford, U.C. Berkeley, C. M. U. 等世界名牌大学所采用。不仅涵盖了程序设计、数据结构、操作系统、计算机体系结构、数据库、编译原理、软件工程、图形学、通信与网络、离散数学等国内大学计算机专业普遍开设的核心课程，而且各具特色——有的出自语言设计者之手、有的历经三十年而不衰、有的已被全世界的几百所高校采用。在这些圆熟通博的名师大作的指引之下，读者必将在计算机科学的宫殿中由登堂而入室。

权威的作者、经典的教材、一流的译者、严格的审校、精细的编辑，这些因素使我们的图书有了质量的保证，但我们的目标是尽善尽美，而反馈的意见正是我们达到这一终极目标的重要帮助。教材的出版只是我们的后续服务的起点。华章公司欢迎老师和读者对我们的工作提出建议或给予指正，我们的联系方法如下：

电子邮件：hzedu@hzbook.com
联系电话：（010）68995264
联系地址：北京市西城区百万庄南街1号
邮政编码：100037

专家指导委员会

（按姓氏笔画顺序）

尤晋元	王　珊	冯博琴	史忠植	史美林
石教英	吕　建	孙玉芳	吴世忠	吴时霖
张立昂	李伟琴	李师贤	李建中	杨冬青
邵维忠	陆丽娜	陆鑫达	陈向群	周伯生
周立柱	周克定	周傲英	孟小峰	岳丽华
范　明	郑国梁	施伯乐	钟玉琢	唐世渭
袁崇义	高传善	梅　宏	程　旭	程时端
谢希仁	裘宗燕	戴　葵		

秘　书　组

武卫东　　温莉芳　　刘　江　　杨海玲

译 者 序

不知不觉涉足Linux内核已经几个年头了，与其他有志于此的朋友一样，我们也经历了学习-实用-追踪-再学习的过程，也就是说，我们也是从漫无边际到茫然无措，再到初窥门径，转而觉得心有戚戚焉这一路走下来的。其中甘苦，悠然在心。

Linux最为人称道的莫过于它的自由精神，所有源代码唾手可得。侯捷先生云：源码在前，了无秘密。是的，但是我们在面对它的时候，为什么却总是因为这种规模和层面所造就的陡峭学习曲线陷入困顿呢？很多朋友就此倒下，纵然Linux世界繁花似锦，纵然内核天空无比广阔。但是，眼前的迷雾重重，心中的阴霾又怎能被阳光驱散呢？纵有雄心壮志，拔剑四顾心茫然，脚下路在何方？

进入Linux内核确实不容易，它之所以难学，在于庞大的规模和涉及的层面。规模一大，就不易现出本来面目，浑然一体，自然不容易找到着手之处；层面一多，就会让人眼花缭乱，盘根错节，怎能让人提纲挈领？

“如果有这样一本书，既能提纲挈领，为我理顺思绪，指引方向，同时又能照顾小节，阐述细微，帮助我们更好更快地理解STL源码，那该有多好。”孟岩先生如此说，虽然针对的是C++，但道出的是研习源码的人们共同的心声。但研究Linux源码的方法却不大相同。这还是由于规模和层面决定的，比如说，在语言学习中，我们可以采取小步快跑的方法，通过一个个小程序和小尝试，就可以取得渐进的成果，就能从新技术中有所收获。而Linux呢？如果没有对整体的把握，即使你对某个局部的算法、技术或是代码再熟悉，也无法将其融入实用。其实，像内核这样的大规模的软件，正是编程技术施展身手的舞台（当然，目前的内核虽然包含了一些面向对象思想，但还不能让C++一展身手）。

面对Linux内核，往往是许多朋友跟大规模软件开发的第一次亲密接触。因此，就愈发不习惯这种自顶而下的学习方式。而很多程序员敲打键盘的速度甚至要比一个一个字地读大部头书快得多，让他们费尽心力去读Maurice J. Bach的《The Design of the UNIX Operating System》，简直是一种折磨。因为很多程序员都坚信：读程序固然能让人开拓视野，但写程序才能让人有真正的进步。

那么，我们能不能做出点儿什么，让学习Linux内核的过程更符合程序员的习惯呢？

Robert Love用本书回答了这个问题。Robert Love是一个狂热的内核爱好者，所以他的想法自然贴近程序员。是的，我们注定要在对所有核心的子系统有了全面认识之后，才能开始自己的实践，但却完全可以舍弃旁支末节，将行李压到最小，自然可以轻装快走，迅速进入动手阶段。这样做的同时，还给那些单纯想要了解一下内核大体实现的人们带来了福音，是的，既然不想动手，自然无需了解过多的细节。

因此，相对于Daniel P. Bovet和Marco Cesati的内核巨著《Understand Linux Kernel》，本书少了五分细节，相对于实践经典《Linux Device Driver》，它多了五分说理。可以说，本书填补了Linux内核理论和实践之间的鸿沟，这正是“一桥飞架南北，天堑变通途”。

就我们的经验而言，内核初学者（不是编程初学者）可以从本书着手，通过本书，可以

对内核各个核心子系统有个整体把握，包括它们提供什么样的服务，为什么要提供这样的服务，又是怎样实现的。而且，本书还包含了Linux内核开发者在开发时需要用到的很多信息，包括调试技术、编程风格、注意事项等等。在消化本书的基础上，如果你侧重于对内核的了解，可以进一步研究《Understand Linux Kernel》和源代码本身；如果你侧重于实际编程，可以研读《Linux Device Driver》，直接开始动手工作；如果读者想有一个轻松的内核学习和实践环，请访问我们的站点www.kerneltravel.net。

序　言

随着Linux内核和Linux应用程序越来越成熟，越来越多的系统软件工程师会涉足Linux开发和维护领域。他们中有些人纯粹是出于个人爱好，有些人是为Linux公司工作，有些是为硬件厂商做开发，还有一些是为内部项目工作的。

但是所有人都必须直面一个问题：内核的学习曲线变得越来越长，也越来越陡峭。系统规模不断扩大、复杂程度不断提高。长此以往，虽然现在这一拨内核开发者对内核的掌握越发炉火纯青，但却会造成新手无法跟上内核发展步伐，出现青黄不接的断层。

我认为这种新老鸿沟已经成为内核质量的一个隐患，而且问题将继续恶化。所以那些真正关心内核的人已经开始致力于扩大内核开发群体。

解决上述问题的一个方法是尽量保证代码简洁：接口定义合理、代码风格一致，“一次做一件事，做到完美”等等。这也就是Linus Torvalds倡导的解决办法。

我提倡的解决办法是对代码慷慨地加上注释：能够让读者立刻了解代码开发者意图的文字（识别意图和实现之间差异的工作称为调试。如果意图不明确显然调试难以进行）。

可是，即使有注解，也没办法清楚地展现内核的各个主要子系统的全景，说明它们到底要做什么。那么，这些开发者又该从何下手呢？

由文字材料来说明这些在起步阶段就该理解的材料，其实是最合适的。

Robert Love的贡献就在于此，有经验的开发者可以通过本书获得对内核子系统提供的服务的全面认识，同时还可以了解这些服务是怎么被实现的。对不少人来说，这些知识就已经足够了：那些好奇的人、那些应用程序开发者、那些想对内核的设计品头论足一番的人，都有足够的谈资了。

但是学习这本书同样可以作为那些有抱负的内核开发者更上一层楼的契机，可以帮他们更改内核代码以达到预定的目标。我建议有抱负的开发者能够亲身实践：理解内核某部分的捷径就是对它做些修改，这样能为开发者揭示仅仅通过看内核代码无法看到的深层机理。

一个严肃认真的内核开发者应该加入开发邮件列表，不断和其他开发者交流。这是内核开发者相互切磋和并肩前进的最好方法。而Robert在书中对内核生活中至关重要的文化和技巧都作了精彩介绍。

请学习和欣赏Robert的书吧。想必你也希望能精益求精，继续探索，成为内核开发社区中的一员，那么，首先你要清楚的是，社区欢迎你。我们评价和衡量一个人是根据他所作的贡献，当你投身于Linux时，你要明白，虽然你仅仅贡献了一小份力，但马上就会有数千万或上亿人受益。这是我们欢乐之源，也是我们责任之本。

Andrew Morton
Digeo Interactive, Palo Alto
July 2003

前　言

在我刚开始有把自己的内核开发经验集结成册、撰写一本书的念头时，我其实也觉得有点头绪繁多、不知道该从何下手。我实在不想落入传统内核书籍的窠臼、照猫画虎地再写这么一本。不错，前人著述备矣，但我终归是要写出点儿与众不同的东西来。说实话，这确实让人颇费思量。

后来，灵感终于浮现出来，我意识到自己可以从一个全新的视角看待这个主题。开发内核是我的工作、开发内核是我的嗜好，内核就是我的挚爱。这些年来，我不断搜集与内核有关的奇闻轶事、不断积攒关键的开发诀窍，依靠这些日积月累的材料，我可以写一本关于开发内核该做什么和不该做什么的书籍。本质上，这本书仍旧是描述Linux内核是如何设计和实现的，但是写法却另辟蹊径，所提供的信息更倾向于实用，通过本书，你就可以做一些内核开发的工作了——并且是使用正确的方法去做。

我希望读者可以从这本书中领略到更多Linux内核的精妙之处（写出来的和没写出来的），也希望读者敢于从阅读本书和读内核代码开始跨越到开始尝试开发可用、可靠、清晰的内核代码。当然如果你仅仅是兴致所至、读书自娱，那也希望你能从中找到乐趣。

不管你学习Linux的意图是什么，我都衷心地希望你能喜欢我的书。

作者的体会

开发Linux内核不需要天赋异秉，不需要有什么魔法，连Unix开发者普遍长着的络腮胡子都不一定要有。内核虽然有一些有趣并且独特的规则和要求，但是它和其他大型软件项目相比，并没有太大差别。像所有的大型软件开发一样，要学的东西确实不少，但是内核开发并不神秘，也不深奥，Linux内核开发者其实并不需要付出比其他开发者更多的努力。

认真阅读源码非常必要，Linux系统代码的开放性其实是弥足珍贵的，不要无动于衷地将它搁置一边，浪费了大好资源。实际上就是读了代码还远远不够呢，你应该钻研并尝试着动手改动一些代码。寻找一个bug然后去修改它，改进你的硬件设备的驱动程序，总之要有的放矢地做一些实际工作！只有动手写代码才能真正融会贯通。

读者范围

本书是写给那些有志于理解Linux内核的软件开发者的。本书并不逐行逐字地注解内核源代码，也不是指导开发驱动程序或是内核API的参考手册（如果存在标准的内核API的话）。本书的初衷是提供足够多的关于Linux内核设计和实现的信息，希望读过本书的程序员能够拥有较为完备的知识，可以真正开始开发内核代码。无论开发内核是为了兴趣还是为了赚钱，我都希望能够带领读者快速走进Linux内核世界。本书不但介绍了理论而且也讨论了具体应用，可以满足不同读者的不同需要。全书无处不围绕着理论联系实践，并非一味强调理论或是实践。无论你研究Linux内核的动机是什么，我都希望这本书都能将内核的设计和实现分析清楚，起到抛砖引玉的作用。

因此，本书覆盖了从核心内核系统的应用到内核设计与实现等各方面内容，我认为这点很重要。例如,第6章讨论的是下半部机制。其中分别讨论了内核下半部机制的设计和实现（核心内核开发者会感兴趣），随即便介绍了如何使用内核提供的接口实现你自己的下半部（这对设备驱动开发者很有用处）。其实，我认为上述两部分内容是相得益彰的，虽然核心内核开发者主要关注的问题是内核内部如何工作，但是也应该清楚如何使用接口；同样，如果设备驱动开发者了解了接口背后的实现机制，自然也会受益匪浅。

这好比学习某些库的API函数与研究该库的具体实现。初看，好像应用程序开发者仅仅需要理解API——我们被灌输的思想是，应该像看待黑盒子一样看待接口。而另一方面，库的开发者也只关心库的设计与实现。但是我认为双方都应该花时间相互学习。能深刻了解操作系统本质的应用程序开发者无疑可以更好地利用它。同样，库开发者也决不应该脱离基于此库的应用程序、蒙头开发。因此，我既讨论了内核子系统的设计，也讨论了它的用法，希望本书能对核心开发者和应用开发者都有用。

我假设读者已经掌握了C语言，而且对Linux比较熟悉。如果读者还具有与操作系统设计相关的经验和其他计算机科学的概念就更好不过了。当然，我也会尽可能多地解释这些概念，但如果你仍然不能理解这些知识的话，请看本书最后参考资料中给出的一些关于操作系统设计方面的经典书籍。

这本书很适合在大学中作为介绍操作系统的辅助教材，与介绍操作系统理论的书相搭配。对于大学高年级课程或者研究生课程来说，可直接使用本书作为教材。

本书的组织形式

本书循序渐进地揭示Linux内核的结构，也就是说，每一章都尽可能地为下一章做铺垫，因此本书是有序地组织各章内容的。但是，这并不是说读者就不能自由地查阅相关章节获取感兴趣部分的参考资料。恰恰相反，我尽力使得各章内容都相对独立，内容尽可能完备，那些对内核基本情况已经有所了解的读者，完全可以按照自己的实际需要直接阅读相关章节。

第1章介绍了操作系统、内核、Unix和本书的主角Linux。如果你读过有关Unix、Linux或其他操作系统的书籍，你一定会发现许多似曾相识的内容。第2章讨论了进程的内核抽象，以及进程如何被创建、销毁和管理。由于操作系统最终目的是让用户运行程序，所以这章是最基础的内容。第3章继续讨论进程的调度，在像Linux这样的抢占式多任务的操作系统中，内核负责调度进程的执行。这一章从进程调度的概念逐步深入到Linux进程调度程序的实现细节。第4章讨论系统调用，它是应用程序访问内核的标准机制。这一章详细讨论了系统调用的原理、系统调用处理程序的设计和如何实现一个新系统调用。第5章讨论了中断和中断处理程序。第6章讨论了下半部和其他推后执行机制。这些技术在编写设备驱动程序和管理设备时是必不可少的。接下来，我用两章篇幅着重介绍同步和锁的概念。第7章讨论了同步和并发，附带介绍了竞争条件和死锁概念，同时提出了对它们的解决方法，比如使用锁定机制。第8章介绍了内核为了维护同步性提供的实际接口，比如自旋锁和信号量。第8章在第7章的基础上对内核锁机制进行了进一步探讨。第9章讨论内核时间流，这一章讨论时间对操作系统的意义和表示形式，另外还介绍了内核如何管理系统时间。最后介绍了内核定时器的实现和使用方法。第10章讨论了内存管理和如何在内核中分配内存。第11章讨论虚拟文件系统（VFS），它为用户空间提供了通用文件接口，是粘合硬件、文件系统、用户空间的一个中间层。第12章讨论块I/O

层，它是内核为了管理硬盘等硬件设备提供的子系统。这层对内核提高块设备的性能至关重要。第13章讨论进程地址空间和进程的虚拟内存。第14章讨论了页高速缓存和如何执行页回写操作。页缓存是Linux内核的主要内存缓存——通过内存缓存可以减少磁盘访问频率，提高系统性能。第15章讨论调试内核的技术，调试内核被公认为是内核编程难于用户程序编程的主要内容，因为不能像对待用户空间的应用程序那样不加限制地对待内核。此章内容包含了调试内核的基本方法，还有一些可以利用的附加技术（这里并没有标准的调试器），以及一些可能带你走出困境的调试技巧。第16章讨论了可移植性和各种体系结构的特殊之处，它可以帮助你编写可以在任意体系结构上运行的优良代码。可移植性是一个有趣的话题，因为它能跨越体系结构的差异。另外由于Linux是个高度可移植性的操作系统，所以在编写高质量内核代码时必须考虑到这一点才能满足跨体系结构运行的要求。第17章讨论了生成补丁和给系统打补丁，以及如何参与Linux内核社区等问题。本书的最后一部分是4个附录和参考资料。附录分别包含了内核链表的实现、单处理器接口、内核随机数产生器和算法复杂度。参考资料列举了许多相关主题的阅读资料。

本书的相关网站

我维护了一个包含本书相关信息的网站：http://tech9.net/rml/kernel_book/。其中包括本书的勘误表、内容扩展和修改，同时也提供了未来重印和再版的信息。希望读者多到这个站点看看。

感谢

与其他作者一样，我决非是一个人躲在山洞里孤苦地写出这本书来的，我能最终完成本书原稿是与无数建议和关怀分不开的。仅仅一页纸无法容纳我的感激，但我还是要衷心地感谢所有给予我鼓励、给予我知识和给予我灵感的朋友和同事。

首先我要提到的人是我的编辑Kathryn Mohr女士，感谢她对我的支持和建议。是她鼓励我开始写作此书，并且从撰写提纲到截稿的整个过程中，都给予我无私的帮助和指导。除她以外，我还要感谢Sams出版社负责本书的小组的其他几个成员，感谢他们给予我的帮助，他们是Chip Gardner、Scott Meyers和George Nedeff。

我的技术编辑Zack Brown也是我需要倍加感谢的人，他认真、准确地校对我的书稿，并给了我许多宝贵的意见。Zack独到的洞察力是至关重要的。他的工作称得上完美，如果书中仍遗留有错误，那责任只由我自己承担，绝对和Zack先生无关。

许多内核开发者为我提供了大力支持，回答了许多问题，还有那些撰写代码的人，本书正是由于有了这些代码，才有了存在的意义。他们的名字是：Alan Cox、Greg Kroah Hartman、William Irwin、Daniel Phillips、David Miller、Partrick Mochel、Ingo Molnar、Andrew Morton、Rik van Riel和Linus Torvalds。

还有许多人在我写作期间不断鼓励我，在方方面面都给予我关怀。这本书也凝聚着他们的爱心。他们是：Paul Amici、Richard Erickson、Dustin Hall、Jack Handy、Joyce Hawkins、Doris Love、Jonathan Love、Linda Love、Dustin Mounce、Randy O' Dowd、Salvatore Ribaudo和他了不起的妈妈、Chirs Rivera、Larry Rivero、Jeremy VanDoren 和他的全家、Steve Weisberg、Helen Whisnant等。

我还要感谢我的雇主、MontaVista 软件公司以及那里的同事，特别是一直支持我的Scott Anderson和Mark Orvek。

感谢Marlena，没有她我不可能取得什么成就。

最后，我要感谢我的父母，感谢他们的爱。

内核黑客万岁。

Robert Love
Gainesville, Florida

◇ ◇ ◇ ◇ ◇

读者反馈方式

本书的所有读者都是我们尊敬的批评者和评论者，我们渴望获得你的意见，我们希望了解我们所做的哪些工作是正确的，哪些工作可以做得更好，你们希望我们出版什么内容的书，以及任何对我们有帮助建议。

你可以直接发email或写信给我，告诉我这本书哪里你喜欢，哪里你不喜欢——我们如何提高书的质量。

请注意我无法回答你们的关于书中主题的相关技术问题，我接收的邮件太多，我难以一一回答。

请在写信时注明书的作者和书中的主题，同时也请写清楚你的姓名、电话和email地址。我一定会详细地阅读你的评论，并且与书的作者和编辑交流。

Email： opensource@samspublishing.com

通信地址： Mark Taber
Associate Publisher
Sams Publishing
800 East 96th Street
Indianapolis, Indiana 46240 USA

如果希望了解该书的更多信息和Sams出版社的其他出版图书，请访问我们的网站www.samspublishing.com。可以在搜索框中输入ISBN（除去连字符）或书的标题来寻找你要找的书。

◇ ◇ ◇ ◇ ◇

关于作者

Robert Love很早就开始使用Linux，而且一直活跃于开源社区。最近，他受聘于MontaVista软件公司，作为软件工程师，继续做开发Linux内核的工作。

他的内核项目包括进程调度程序、抢占式内核，还有VM和多任务处理性能优化。他负责

维护的另外两个开源项目是schedutils和procps.。除此之外，他对内核有不少精彩评论，而且他还是*Linux Journal*杂志的特邀编辑。目前，他居住在佛罗里达州的Gainesville，爱好摄影以及美食。

关于技术编辑

Zack Brown是著名的时事通讯*Kernel Traffic*的发起人和作者之一，该杂志起源于1999年，它跟踪linux内核开发邮件列表的最新事件。他定期提供*Linux Journal*的“diff-u”服务。并为德国*Linux Magazin*杂志撰写“Zack's Kernel News”。他在旧金山生活和工作。

目　录