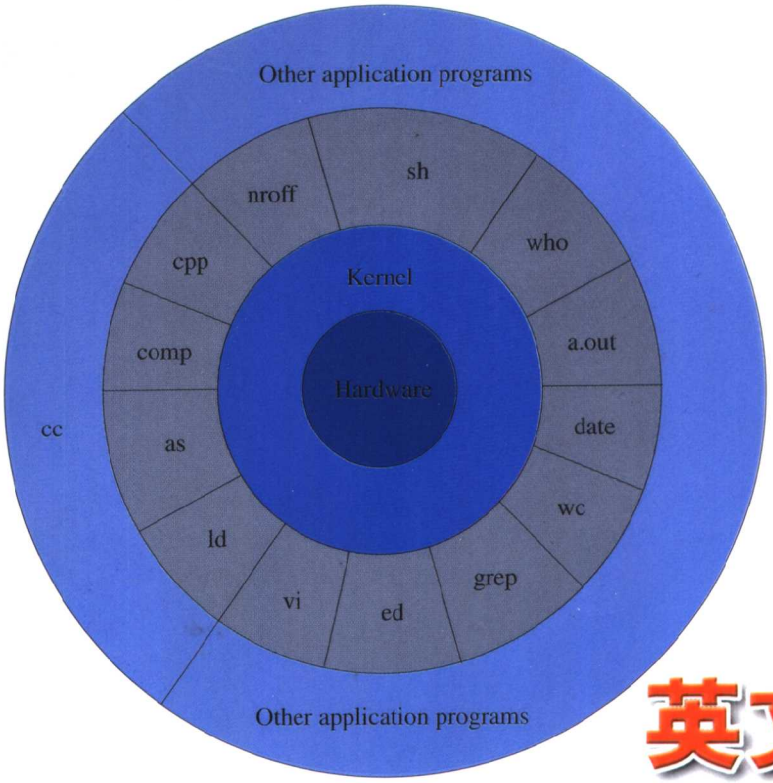
国外著名高等院校
信息科学与技术优秀教材

PH
PTR

UNIX操作系统设计

The Design of The UNIX Operating System

Maurice J. Bach



英文版

人民邮电出版社
POSTS & TELECOMMUNICATIONS PRESS

封面设计：胡平利

PH
PTR

国外著名高等院校信息科学与技术优秀教材

UNIX 操作系统设计

The Design of The
UNIX
Operating System

英文版

本书探讨了在计算机产业中流行的UNIX操作系统。作者描述了构成操作系统基础（内核）的内部算法和结构，以及它们同程序员所看到的编程接口的关系。

本书的主要特色包括：

- 描述内核体系结构的轮廓
- 介绍系统缓冲区的缓存机制
- 包括文件系统内部使用的数据结构和算法
- 涵盖提供文件系统用户接口的系统调用
- 定义进程的现场（context），研究控制进程现场的内部内核原语
- 介绍控制进程现场的系统调用
- 描述进程调度机制
- 讨论包括交换和调页系统在内的内存管理
- 概述一般性的驱动程序接口，特别讨论了磁盘驱动程序和终端驱动程序
- 介绍流的概念
- 介绍进程间通信和连网技术，包括 System V 消息、共享内存和信号量
- 阐述紧耦合多处理机 UNIX 系统
- 研究分布式 UNIX 系统

www.PearsonEd.com

ISBN 7-115-11246-0

9 787115 112460 >

ISBN7-115-11246-0/TP·3436
定价：45.00 元

Pearson Education

人民邮电出版社
http: // www.ptpress.com.cn

国外著名高等院校信息科学与技术优秀教材

UNIX 操作系统设计

(英文版)

The Design Of The UNIX Operating System

Maurice J. Bach

人 民 邮 电 出 版 社

图书在版编目（CIP）数据

UNIX 操作系统设计. 英文 /（美）巴赫（Bach, M. J. ）著. —北京：人民邮电出版社，2003.6
国外著名高等院校信息科学与技术优秀教材
ISBN 7-115-11246-0
I. U… II. 巴… III. UNIX 操作系统—程序设计—高等学校—教材—英文 IV. TP316.81
中国版本图书馆 CIP 数据核字（2003）第 031029 号

版 权 声 明

国外著名高等院校信息科学与技术优秀教材

UNIX 操作系统设计（英文版）

◆ Maurice J.Bach
责任编辑 李 际

◆ 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
读者热线 010-67132705
北京汉魂图文设计有限公司制作
北京朝阳展望印刷厂印刷
新华书店总店北京发行所经销

◆ 开本：800×1000 1/16
印张：30.5
字数：668 千字 2003 年 6 月第 1 版
印数：1-3 500 册 2003 年 6 月北京第 1 次印刷

著作权合同登记 图字：01-2003-0897 号

ISBN 7-115-11246-0/TP·3436

定价：45.00 元

本书如有印装质量问题，请与本社联系 电话：（010）67129223

内 容 提 要

本书以 UNIX 系统 V 为背景，全面、系统地介绍了 UNIX 操作系统内核的内部数据结构和算法。本书首先对系统内核结构做了简要介绍，然后分章节描述了文件系统、进程调度和存储管理，并在此基础上讨论了 UNIX 系统的高级问题，如驱动程序接口、进程间通信与网络等。在每章之后，还给出了大量富有启发性和实际意义的题目。

本书可作为大学计算机科学系高年级学生和研究生的教材或参考书。本书也为从事 UNIX 系统研究与实用程序开发人员提供了一本极有价值的参考资料。

出版说明

2001 年，教育部印发了《关于“十五”期间普通高等教育教材建设与改革的意见》。该文件明确指出，“九五”期间原国家教委在“抓好重点教材，全面提高质量”方针指导下，调动了各方面的积极性，产生了一大批具有改革特色的新教材。然而随着科学技术的飞速发展，目前高校教材建设工作仍滞后于教学改革的实践，一些教材内容陈旧，不能满足按新的专业目录修订的教学计划和课程设置的需要。为此该文件明确强调，要加强国外教材的引进工作。当前，引进的重点是信息科学与技术和生物科学与技术两大学科的教材。要根据专业（课程）建设的需要，通过深入调查、专家论证，引进国外优秀教材。要注意引进教材的系统配套，加强对引进教材的宣传，促进引进教材的使用和推广。

邓小平同志早在 1977 年就明确指出：“要引进外国教材，吸收外国教材中有益的东西。”随着我国加入 WTO，信息产业的国际竞争将日趋激烈，我们必须尽快培养出大批具有国际竞争能力的高水平信息技术人才。教材是一个很关键的问题，国外的一些优秀教材不但内容新，而且还提供了很多新的研究方法和思考方式。引进国外原版教材，可以促进我国教学水平的提高，提高学生的英语水平和学习能力，保证我们培养出的学生具有国际水准。

为了贯彻中央“科教兴国”的方针，配合国内高等教育教材建设的需要，人民邮电出版社约请有关专家反复论证，与国外知名的教材出版公司合作，陆续引进一些信息科学与技术优秀教材。第一批教材针对计算机专业的主干核心课程，是国外著名高等院校所采用的教材，教材的作者都是在相关领域享有盛名的专家教授。这些教材内容新，反映了计算机科学技术的最新发展，对全面提高我国信息科学与技术的教学水平必将起到巨大的推动作用。

出版国外著名高等院校信息科学与技术优秀教材的工作将是一个长期的、坚持不懈的过程，我社网站（www.ptpress.com.cn）上介绍了我们陆续推出的图书的详细情况，敬请关注。希望广大教师和学生将使用中的意见和建议及时反馈给我们，我们将根据您的反馈不断改进我们的工作，推出更多更好的引进版信息科学与技术教材。

人民邮电出版社

序　言

本书几乎是公认的第一本全面详尽介绍 UNIX System V 内核结构的经典书籍。Bach 在他的这本传世之作中深入分析了 UNIX 的内核算法、基本数据结构以及它们同上层 UNIX 编程接口的关系。

《The Design of The UNIX Operating System》一书写于 1986 年。此前 AT&T 已经将 UNIX 操作系统商业化，任何人要学习和使用 UNIX，再也不能像原来那样无偿地从 AT&T 获得源代码拷贝和许可。相反，AT&T 不但要对发放的 UNIX 源代码许可证收取 10 万美元，而且还要强迫需要 UNIX 的人与之签订苛刻的保密协议。一时间，对于普通人来说，获得有关 UNIX 原理和技术的资料一下子变得非常困难，UNIX 被蒙上一层厚重的面纱。

本书作者 Bach 当时在 AT&T 工作，因而有机会研究 UNIX 的源代码。虽然法律规定，他不能在自己的书里直接援引 C 源代码，但是聪明的 Bach 却利用伪代码的形式巧妙地绕过了这项限制。采用伪代码来介绍内核原理，不但叙述简洁，而且易于理解，因为伪代码可以让读者避免陷入大量微观细节而失去全局概念的情况。Bach 在宏观上阐述了构成内核的基本算法，而把微观的具体实现细节留给读者自己去构想。这样一来，就在事实上鼓励着富有智慧的程序员，在读罢本书之后尝试实现自己的同 UNIX 兼容的内核。

在本书为数众多的读者当中，就有芬兰赫尔辛基大学的研究生 Linus Torvalds。20 世纪 90 年代初，Linus 为了学习操作系统的内部原理，自己开始着手编写一个同 UNIX 兼容的操作系统，Bach 的书正好是从程序员角度来讲述 UNIX 的，这给予 Linus 很大的帮助。同时 Linus 还参考了荷兰教授 A. S. Tanenbaum 为教学而开发的 Minix。1991 年的夏天，22 岁的 Linus 在 Internet 上发布了他的操作系统内核 Linux kernel 0.01。一个崭新的时代到来了！

虽然本书已经问世将近 20 年，在这期间，UNIX 操作系统的发展突飞猛进，现代 UNIX 家族的成员可谓种类繁多，但是直到现在，世界上很多大学里讲授操作系统课程时，仍然会把本书列为标准参考书。因为本书虽然倾向于 System V，可是介绍的算法、数据结构却并没有特别针对任何一种特定的内核，所以时至今日，如果读者想要踏上学习 UNIX/Linux 内核的旅程，那么本书依然是最好的出

发点之一。

系统程序员通过阅读本书，可以更好地掌握 UNIX 内核的工作原理，而且还可以把 UNIX 内核中的算法和其他操作系统中的算法进行比较。UNIX 程序员通过阅读本书，可以更深入地理解他们编写的程序是怎样和操作系统本身发生交互关系的。这不但让程序员写代码的工作变得容易了，而且写出的代码能高效地和 UNIX 内核协调发挥作用。

如果说本书还有缺点的话，那就是在 1986 年的时候，现代 UNIX 系统普遍具有的 SMP 和 multithreading 等技术尚未成形，因此 Bach 的书中没有对这些新特性进行介绍，有兴趣的读者可以参考人民邮电出版社出版的《现代体系结构上的 UNIX 系统——内核程序员的 SMP 和 Caching 技术》和《UNIX 系统内幕：新的技术领域（英文版）》两本书。

梁 煜　博士
美国　明尼苏达大学
高性能计算研究中心

To my parents, for their patience and devotion,
to my daughters, Sarah and Rachel, for their laughter,
to my son, Joseph, who arrived after the first printing,
and to my wife, Debby, for her love and understanding.

CONTENTS

1

GENERAL OVERVIEW OF THE SYSTEM

The UNIX system has become quite popular since its inception in 1969, running on machines of varying processing power from microprocessors to mainframes and providing a common execution environment across them. The system is divided into two parts. The first part consists of programs and services that have made the UNIX system environment so popular; it is the part readily apparent to users, including such programs as the shell, mail, text processing packages, and source code control systems. The second part consists of the operating system that supports these programs and services. This book gives a detailed description of the operating system. It concentrates on a description of UNIX System V produced by AT&T but considers interesting features provided by other versions too. It examines the major data structures and algorithms used in the operating system that ultimately provide users with the standard user interface.

This chapter provides an introduction to the UNIX system. It reviews its history and outlines the overall system structure. The next chapter gives a more detailed introduction to the operating system.

1.1 HISTORY

In 1965, Bell Telephone Laboratories joined an effort with the General Electric Company and Project MAC of the Massachusetts Institute of Technology to

develop a new operating system called Multics [Organick 72]. The goals of the Multics system were to provide simultaneous computer access to a large community of users, to supply ample computation power and data storage, and to allow users to share their data easily, if desired. Many people who later took part in the early development of the UNIX system participated in the Multics work at Bell Laboratories. Although a primitive version of the Multics system was running on a GE 645 computer by 1969, it did not provide the general service computing for which it was intended, nor was it clear when its development goals would be met. Consequently, Bell Laboratories ended its participation in the project.

With the end of their work on the Multics project, members of the Computing Science Research Center at Bell Laboratories were left without a "convenient interactive computing service" [Ritchie 84a]. In an attempt to improve their programming environment, Ken Thompson, Dennis Ritchie, and others sketched a paper design of a file system that later evolved into an early version of the UNIX file system. Thompson wrote programs that simulated the behavior of the proposed file system and of programs in a demand-paging environment, and he even encoded a simple kernel for the GE 645 computer. At the same time, he wrote a game program, "Space Travel," in Fortran for a GECOS system (the Honeywell 635), but the program was unsatisfactory because it was difficult to control the "space ship" and the program was expensive to run. Thompson later found a little-used PDP-7 computer that provided good graphic display and cheap executing power. Programming "Space Travel" for the PDP-7 enabled Thompson to learn about the machine, but its environment for program development required cross-assembly of the program on the GECOS machine and carrying paper tape for input to the PDP-7. To create a better development environment, Thompson and Ritchie implemented their system design on the PDP-7, including an early version of the UNIX file system, the process subsystem, and a small set of utility programs. Eventually, the new system no longer needed the GECOS system as a development environment but could support itself. The new system was given the name UNIX, a pun on the name Multics coined by another member of the Computing Science Research Center, Brian Kernighan.

Although this early version of the UNIX system held much promise, it could not realize its potential until it was used in a real project. Thus, while providing a text processing system for the patent department at Bell Laboratories, the UNIX system was moved to a PDP-11 in 1971. The system was characterized by its small size: 16K bytes for the system, 8K bytes for user programs, a disk of 512K bytes, and a limit of 64K bytes per file. After its early success, Thompson set out to implement a Fortran compiler for the new system, but instead came up with the language B, influenced by BCPL [Richards 69]. B was an interpretive language with the performance drawbacks implied by such languages, so Ritchie developed it into one he called C, allowing generation of machine code, declaration of data types, and definition of data structures. In 1973, the operating system was rewritten in C, an unheard of step at the time, but one that was to have tremendous impact on its acceptance among outside users. The number of installations at Bell