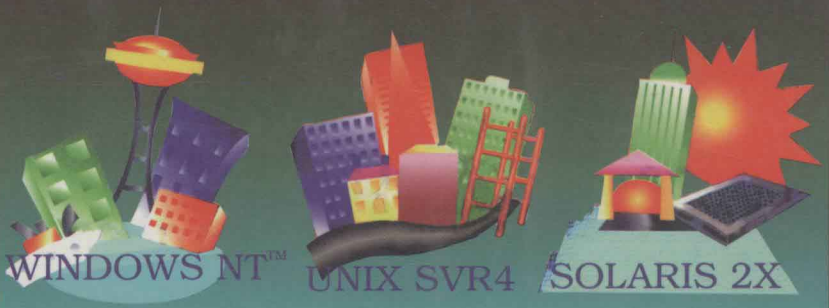
大学计算机教育丛书（影印版）

THIRD EDITION

OPERATING SYSTEMS

Internals and Design Principles



操作系统

精髓与设计原理

第 3 版

William Stalling

清华大学出版社 · PRENTICE HALL
http://www.tup.tsinghua.edu.cn

OPERATING SYSTEMS

Internals and Design Principles

THIRD EDITION

操作系统

精髓与设计原理

第3版

William Stallings

清华大学出版社

Prentice-Hall International, Inc.

(京) 新登字 158 号

Operating systems: internals and design principles 3rded. /William Stallings

图书在版编目(CIP)数据

操作系统: 精髓与原理 第 3 版: 英文/( )斯托林斯(Stallings, W.)著. —北京: 清华大学出版社, 1998.6
ISBN 7-302-02976-8
Ⅰ. 操… Ⅱ. 斯… Ⅲ. 操作系统(软件)-英文 Ⅳ. TP316

中国版本图书馆 CIP 数据核字(98)第 10958 号

出版者: 清华大学出版社(北京清华大学校内, 邮编 100084)
因特网地址: www.tup.tsinghua.edu.cn
印刷者: 清华大学印刷厂
发行者: 新华书店总店北京科技发行所
开 本: 850×1168 1/32 印张: 25.125
版 次: 1998 年 6 月第 1 版 1998 年 6 月第 1 次印刷
书 号: ISBN 7-302-02976-8/TP·1577
印 数: 0001～5000
定 价: 35.00 元

出版前言

我们的大学生、研究生毕业后，面临的将是一个国际化的信息时代。他们将需要随时查阅大量的外文资料；会有更多的机会参加国际性学术交流活动；接待外国学者；走上国际会议的讲坛。作为科技工作者，他们不仅应有与国外同行进行口头和书面交流的能力，更为重要的是，他们必须具备极强的查阅外文资料获取信息的能力。有鉴于此，在国家教委所颁布的“大学英语教学大纲”中有一条规定：专业阅读应作为必修课程开设。同时，在大纲中还规定了这门课程的学时和教学要求。有些高校除开设“专业阅读”课之外，还在某些专业课拟进行英语授课。但教、学双方都苦于没有一定数量的合适的英文原版教材作为教学参考书。为满足这方面的需要，我们挑选了 7 本计算机科学方面最新版本的教材，进行影印出版。首批影印出版的 6 本书受到广大读者的热情欢迎，我们深受鼓舞，今后还将陆续推出新书。希望读者继续给予大力支持。Prentice Hall 公司和清华大学出版社这次合作将国际先进水平的教材引入我国高等学校，为师生们提供了教学用书，相信会对高校教材改革产生积极的影响。

清华大学出版社

Prentice Hall 公司

1997.11

PREFACE

OBJECTIVES

This book is about the concepts, structure, and mechanisms of operating systems. Its purpose is to present, as clearly and completely as possible, the nature and characteristics of modern-day operating systems.

This task is a challenging one for several reasons. First, there is a tremendous range and variety of computer systems for which operating systems are designed. These include single-user workstations and personal computers, medium-sized shared systems, large mainframe and supercomputers, and specialized machines such as real-time systems. The variety is not just in the capacity and speed of machines, but in applications and system support requirements. Second, the rapid pace of change that has always characterized computer systems continues with no letup. A number of key areas in operating system design are of recent origin, and research into these and other new areas continues.

In spite of this variety and pace of change, certain fundamental concepts apply consistently throughout. To be sure, the application of these concepts depends on the current state of technology and the particular application requirements. The intent of this book is to provide a thorough discussion of the fundamentals of operating system design, and to relate these to contemporary design issues and to current directions in the development of operating systems.

The object is to provide the reader with a solid understanding of the key mechanisms of modern operating systems, the types of design tradeoffs and decisions involved in OS design, and the context within which the operating system functions (hardware, other system programs, application programs, interactive users).

EXAMPLE SYSTEMS

This text is intended to acquaint the reader with the design principles and implementation issues of contemporary operating systems. Accordingly, a purely conceptual or theoretical treatment would be inadequate. To illustrate the concepts and to tie them to real-world design choices that must be made, three operating systems have been chosen as running examples:

- **Windows NT:** A single-user, multitasking operating system for personal computers, workstations, and servers. As a new operating system, it incorporates in a clean fashion many of the latest developments in operating system technology. In addition, Windows NT is one of the first important commercial operating systems to rely heavily on object-oriented design principles.
- **UNIX:** A multiuser operating system, originally intended for minicomputers, but implemented on a wide range of machines from powerful microcomputers to supercomputers. The version emphasized in this book is SVR4, which incorporates many state-of-the-art features.
- **Solaris:** The most widely used commercial version of UNIX. Solaris includes multithreading and other features not found in SVR4 and most other UNIX variants.

These three systems were chosen because of their relevance and representativeness. As with the technique used in the author's *Computer Organization and Architecture,* the discussion of the example systems is distributed throughout the text rather than assembled as a single chapter or appendix. Thus, during the discussion of concurrency, the concurrency mechanisms of each example systems are described, and the motivation for the individual design choices is discussed. With this approach, the design concepts discussed in a given chapter are immediately reinforced with real-world examples.

INTENDED AUDIENCE

This book is intended for both an academic and a professional audience. As a textbook, it is intended as a one-semester undergraduate course in operating systems for computer science, computer engineering, and electrical engineering majors. It covers the topics in Subject Area 5 of the IEEE Model Program in Computer Science and Engineering and also covers the OS-related topics in CS 6 and CS10 of the ACM Recommendations for the Undergraduate Program in Computer Science. The book also serves as a basic reference volume and is suitable for self-study.

The book is organized in seven parts:

I. **Background:** Provides an overview of computer architecture and organization, with emphasis on topics that relate to operating system design, plus an overview of the OS topics in remainder of the book.

II. **Processes:** Presents a detailed analysis of processes, multithreading, symmetric multiprocessing (SMP), and microkernels. This part also examines the key aspects of concurrency on a single system, with emphasis on issues of mutual exclusion and deadlock.

III. **Memory:** Provides a comprehensive survey of techniques for memory management.

IV. **Scheduling:** Provides a comparative discussion of various approaches to process scheduling. Thread scheduling, SMP scheduling, and real-time scheduling are also examined.

V. **Input/Output and Files:** Examines the issues involved in OS control of the I/O function. Special attention is devoted to disk I/O, which is the key to system performance. Also provides an overview of file management.

VI. **Distributed Systems:** Examines the major trends in the networking of computer systems, including TCP/IP, client/server computing, and clusters. Also describes some of the key design areas in the development of distributed operating systems.

VII. **Security:** Provides a survey of threats and mechanisms for providing computer and network security.

A more detailed, chapter-by-chapter summary appears at the end of Chapter 2. In addition, the book includes an extensive glossary, a list of frequently used acronyms, and a bibliography. Each chapter includes problems and suggestions for further reading.

There is a Web page for this book that provides support for students and instructors. The page includes links to relevant sites, transparency masters of figures in the book in PDF (Adobe Acrobat) format, a set of lecture notes that can be used as slides or student handouts, and sign-up information for the book's internet mailing list. The Web page is at http://www.shore.net/~ws/OS3e.html. An Internet mailing list has been set up so that instructors using this book can exchange information, suggestions, and questions with each other and with the author. As soon as typos or other errors are discovered, an errata list for this book will be available at http://www.shore.net/~ws.

OPERATING SYSTEM PROJECTS

For many instructors, an important component of an OS course is a project or set of projects by which the student gets hands-on experience to reinforce concepts from the text. This book provides an unparalled degree of support for including a projects component to the course. Information is provided on three software packages that serve as frameworks for project implementation: OSP and Nachos for developing components of an OS, and BACI for studying concurrency mechanisms. In addition, the instructor's manual includes a series of small programming projects, each intended to take a week or two, that cover a broad range of topics and that can be implemented in any suitable language on any platform. See Appendix C for details.

WHAT'S NEW IN THE THIRD EDITION

In the three years since the second edition of *Operating Systems* was published, the field has seen continued innovations and improvements. In this new edition, I try to capture these changes, while maintaining a broad and comprehensive coverage of the entire field. To begin this process of revision, the second edition of this book was extensively reviewed by a number of professors who taught from that edition. The result is that, in many places, the narrative has been clarified and tightened, and illustrations have been improved. Also, a number of new "field-tested" problems have been added.

Beyond these refinements to improve pedagogy and user friendliness, there have been major substantive changes throughout the book. Highlights include the following:

- **Multithreading:** The coverage of multithreading has been greatly expanded, including treatment of user-level versus kernel-level threads, thread scheduling, and thread migration. Discussion of multithreading in Windows NT has been expanded and treatment of Solaris multithreading has been added.
- **Symmetric Multiprocessing (SMP):** This area has also been substantially expanded, covering process management and process/thread scheduling in greater detail.
- **Microkernels:** A section on microkernels has been added.
- **Clusters:** Clusters have become an important form of computer and OS organization in recent years, and a new section has been added to treat this topic. Discussions of both the Windows NT Wolfpack and Solaris MC approaches are also included.
- **Object-oriented Design:** The coverage of object-oriented design has been expanded, including greater detail in the example systems. The appendix now contains an introduction to CORBA.
- **Expanded Instructor Support:** As mentioned above, the book now provides extensive support for projects. Support provided by the book Web site has also been expanded.

ACKNOWLEDGMENTS

This new edition has benefited from review by a number of people, who gave generously of their time and expertise. The following people reviewed the second edition and made many helpful suggestions: John B. Connely (Cal Poly State Univ.–San Luis Obispo), Michael P. Deignan, John Graham (Univ. of Delaware), Jeff Jirka, Wayne Madison (Clemson), Matt W. Mutka (Michigan State Univ.), Bina Ramamurthy (SUNY–Buffalo), and Jaleh Rezaie (Easter Kentucky Univ.). Tracy Camp (U. of Alabama) reviewed the new material on microkernels. Ewan Grantham (Grantham & Associates), Bill Kuhn (Telecommunications Research Associates), Dave Porter, and Bill Potvin (STK/NGS/TerIS) reviewed the Windows NT material. Jerry Gulla (Sun Microsystems) and William Tepfenhart (AT&T) reviewed the UNIX material.

I would also like to thank Michael Kifer and Scott A. Smolka (SUNY–Stony Brook) for contributing Appendix D; Bill Bynum (College of William and Mary) and Tracy Camp (U. of Alabama) for contributing Appendix E; and Steve Taylor (Worcester Polytechnic Institute) for contributing the programming projects and reading/report assignments in the instructor's manual.

CONTENTS

APPENDICES

PART ONE Background

ISSUES FOR PART ONE

The purpose of Part One is to provide a background and context for the remainder of this book. The fundamental concepts of computer architecture and operating system internals are presented.

ROAD MAP FOR PART ONE

Chapter 1: Computer System Overview

An operating system mediates between application programs, utilities, and users on the one hand and the computer system hardware on the other. To appreciate the functionality of the operating system (OS) and the design issues involved, one must have some appreciation for computer organization and architecture. Chapter 1 provides a brief survey of the processor, memory, and input/output I/O elements of a computer system.

Chapter 2: Operating System Overview

The topic of operating system design covers a huge territory and it is easy to get lost in the details and lose the context of a discussion of a particular issue. Chapter 2 provides an overview to which the reader can return at any point in the book for context. We begin with a statement of the objectives and functions of an operating system. Then some historically important systems and OS functions are described. This discussion allows us to present some fundamental OS design principles in a simple environment so that the relationship among various OS functions is clear. The next chapter highlights important characteristics of modern operating systems. Throughout the book, as various topics are discussed, it is necessary to talk about both fundamental, well-established principles as well as more recent innovations in OS design. The discussion in this chapter alerts the reader to this blend of