THE DESIGN OF THE UNIX OPERATING SYSTEM

MAURICE J. BACH

THE DESIGN OF THE UNIX® OPERATING SYSTEM

Maurice J. Bach

Prentice-Hall of India Private Limited
New Delhi-110001
1997

CONTENTS

PREFACE xi

CHAPTER 1 GENERAL OVERVIEW OF THE SYSTEM 1

1.1 HISTORY 1

1.2 SYSTEM STRUCTURE 4

1.3 USER PERSPECTIVE 6

1.4 OPERATING SYSTEM SERVICES 14

1.5 ASSUMPTIONS ABOUT HARDWARE 15

1.6 SUMMARY 18

CHAPTER 2 INTRODUCTION TO THE KERNEL 19
2.1 ARCHITECTURE OF THE UNIX OPERATING SYSTEM 19
2.2 INTRODUCTION TO SYSTEM CONCEPTS 22
2.3 KERNEL DATA STRUCTURES 34
2.4 SYSTEM ADMINISTRATION 34
2.5 SUMMARY AND PREVIEW 36
2.6 EXERCISES 37

CHAPTER 3 THE BUFFER CACHE 38
3.1 BUFFER HEADERS 39
3.2 STRUCTURE OF THE BUFFER POOL 40
3.3 SCENARIOS FOR RETRIEVAL OF A BUFFER 42
3.4 READING AND WRITING DISK BLOCKS 53
3.5 ADVANTAGES AND DISADVANTAGES OF THE BUFFER CACHE 56
3.6 SUMMARY 57
3.7 EXERCISES 58

CHAPTER 4 INTERNAL REPRESENTATION OF FILES 60
4.1 INODES 61
4.2 STRUCTURE OF A REGULAR FILE 67
4.3 DIRECTORIES 73
4.4 CONVERSION OF A PATH NAME TO AN INODE . . . 74
4.5 SUPER BLOCK 76
4.6 INODE ASSIGNMENT TO A NEW FILE 77
4.7 ALLOCATION OF DISK BLOCKS 84
4.8 OTHER FILE TYPES 88
4.9 SUMMARY 88
4.10 EXERCISES 89

CHAPTER 5 SYSTEM CALLS FOR THE FILE SYSTEM 91
5.1 OPEN 92
5.2 READ 96
5.3 WRITE 101
5.4 FILE AND RECORD LOCKING 103
5.5 ADJUSTING THE POSITION OF FILE I/O – LSEEK . . 103
5.6 CLOSE 103
5.7 FILE CREATION 105
5.8 CREATION OF SPECIAL FILES 107
5.9 CHANGE DIRECTORY AND CHANGE ROOT . . . 109
5.10 CHANGE OWNER AND CHANGE MODE 110
5.11 STAT AND FSTAT 110
5.12 PIPES 111
5.13 DUP 117
5.14 MOUNTING AND UNMOUNTING FILE SYSTEMS . . 119
5.15 LINK 128
5.16 UNLINK 132
5.17 FILE SYSTEM ABSTRACTIONS 138
5.18 FILE SYSTEM MAINTENANCE 139
5.19 SUMMARY 140
5.20 EXERCISES 140

CHAPTER 6 THE STRUCTURE OF PROCESSES 146
6.1 PROCESS STATES AND TRANSITIONS 147
6.2 LAYOUT OF SYSTEM MEMORY 151
6.3 THE CONTEXT OF A PROCESS 159
6.4 SAVING THE CONTEXT OF A PROCESS 162
6.5 MANIPULATION OF THE PROCESS ADDRESS SPACE 171
6.6 SLEEP 182

6.7 SUMMARY 188
6.8 EXERCISES 189

CHAPTER 7 PROCESS CONTROL 191
7.1 PROCESS CREATION 192
7.2 SIGNALS 200
7.3 PROCESS TERMINATION 212
7.4 AWAITING PROCESS TERMINATION 213
7.5 INVOKING OTHER PROGRAMS 217
7.6 THE USER ID OF A PROCESS 227
7.7 CHANGING THE SIZE OF A PROCESS 229
7.8 THE SHELL 232
7.9 SYSTEM BOOT AND THE INIT PROCESS 235
7.10 SUMMARY 238
7.11 EXERCISES 239

CHAPTER 8 PROCESS SCHEDULING AND TIME 247
8.1 PROCESS SCHEDULING 248
8.2 SYSTEM CALLS FOR TIME 258
8.3 CLOCK 260
8.4 SUMMARY 268
8.5 EXERCISES 268

CHAPTER 9 MEMORY MANAGEMENT POLICIES 271
9.1 SWAPPING 272
9.2 DEMAND PAGING 285
9.3 A HYBRID SYSTEM WITH SWAPPING AND DEMAND PAGING 307
9.4 SUMMARY 307
9.5 EXERCISES 307

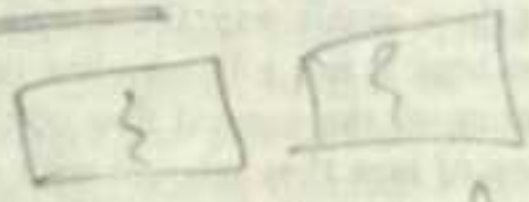
CHAPTER 10 THE I/O SUBSYSTEM 312
10.1 DRIVER INTERFACES 313
10.2 DISK DRIVERS 325
10.3 TERMINAL DRIVERS 329
10.4 STREAMS 344
10.5 SUMMARY 351
10.6 EXERCISES 352

CHAPTER 11 INTERPROCESS COMMUNICATION 355
11.1 PROCESS TRACING 356
11.2 SYSTEM V IPC 359
11.3 NETWORK COMMUNICATIONS 382
11.4 SOCKETS 383
11.5 SUMMARY 388
11.6 EXERCISES 389

CHAPTER 12 MULTIPROCESSOR SYSTEMS 391
12.1 PROBLEM OF MULTIPROCESSOR SYSTEMS 392
12.2 SOLUTION WITH MASTER AND SLAVE PROCESSORS 393
12.3 SOLUTION WITH SEMAPHORES 395
12.4 THE TUNIS SYSTEM 410
12.5 PERFORMANCE LIMITATIONS 410
12.6 EXERCISES 410

CHAPTER 13 DISTRIBUTED UNIX SYSTEMS 412
13.1 SATELLITE PROCESSORS 414
13.2 THE NEWCASTLE CONNECTION 422
13.3 TRANSPARENT DISTRIBUTED FILE SYSTEMS . . . 426
13.4 A TRANSPARENT DISTRIBUTED MODEL WITHOUT STUB

No concept of Multi-threading in Unix.



PREFACE

The UNIX system was first described in a 1974 paper in the Communications of the ACM [Thompson 74] by Ken Thompson and Dennis Ritchie. Since that time, it has become increasingly widespread and popular throughout the computer industry where more and more vendors are offering support for it on their machines. It is especially popular in universities where it is frequently used for operating systems research and case studies.

Many books and papers have described parts of the system, among them, two special issues of the Bell System Technical Journal in 1978 [BSTJ 78] and 1984 [BLTJ 84]. Many books describe the user level interface, particularly how to use electronic mail, how to prepare documents, or how to use the command interpreter called the shell; some books such as *The UNIX Programming Environment* [Kernighan 84] and *Advanced UNIX Programming* [Rochkind 85] describe the programming interface. This book describes the internal algorithms and structures that form the basis of the operating system (called the kernel) and their relationship to the programmer interface. It is thus applicable to several environments. First, it can be used as a textbook for an operating systems course at either the advanced undergraduate or first-year graduate level. It is most beneficial to reference the system source code when using the book, but the book can be read independently, too. Second, system programmers can use the book as a reference to gain better understanding of how the kernel works and to compare algorithms used in the UNIX system to algorithms used in other operating systems.