

Inhaltsübersicht

Vorwort	21
Kapitel 1 Einführung	27
Kapitel 2 Prozesse und Threads	125
Kapitel 3 Speicherverwaltung	237
Kapitel 4 Dateisysteme	329
Kapitel 5 Eingabe und Ausgabe	415
Kapitel 6 Deadlocks	531
Kapitel 7 Virtualisierung und die Cloud	573
Kapitel 8 Multiprozessorsysteme	627
Kapitel 9 IT-Sicherheit	715
Kapitel 10 Fallstudie 1: Linux	855
Kapitel 11 Fallstudie 2: Windows 8	1017
Kapitel 12 Entwurf von Betriebssystemen	1161
Kapitel 13 Bibliografie	1221
Namensregister	1265
Register	1267

Inhaltsverzeichnis

Vorwort	21
Kapitel 1 Einführung	27
1.1 Was ist ein Betriebssystem?	31
1.1.1 Das Betriebssystem als eine erweiterte Maschine	31
1.1.2 Das Betriebssystem als Ressourcenverwalter	33
1.2 Geschichte der Betriebssysteme	34
1.2.1 Die erste Generation (1945–1955) – auf Basis von Elektronenrohren.	35
1.2.2 Die zweite Generation (1955–1965) – Transistoren und Stapelverarbeitungssysteme	36
1.2.3 Die dritte Generation (1965–1980) – integrierte Schaltkreise und Multiprogrammierung	38
1.2.4 Die vierte Generation (1980 bis heute) – der PC	43
1.2.5 Die funfte Generation (1990 bis heute) – mobile Computer	48
1.3 Überblick über die Computerhardware	49
1.3.1 Prozessoren	50
1.3.2 Arbeitsspeicher	54
1.3.3 Festplatten	58
1.3.4 Ein-/Ausgabegeräte	59
1.3.5 Bussysteme	63
1.3.6 Hochfahren des Computers.	66
1.4 Die Betriebssystemfamilie	67
1.4.1 Betriebssysteme für Großrechner	67
1.4.2 Betriebssysteme für Server	67
1.4.3 Betriebssysteme für Multiprozessorsysteme	68
1.4.4 Betriebssysteme für PCs	68
1.4.5 Betriebssysteme für Handheld-Computer	68
1.4.6 Betriebssysteme für eingebettete Systeme	69
1.4.7 Betriebssysteme für Sensorknoten	69
1.4.8 Echtzeitbetriebssysteme	69
1.4.9 Betriebssysteme für Smartcards	70
1.5 Betriebssystemkonzepte	71
1.5.1 Prozesse	71
1.5.2 Adressräume	74
1.5.3 Dateien	74
1.5.4 Ein- und Ausgabe	78
1.5.5 Datenschutz und Datensicherheit.	78
1.5.6 Die Shell.	79
1.5.7 Die Ontogenese rekapituliert die Phylogenese.	80

1.6	Systemaufrufe	84
1.6.1	Systemaufrufe zur Prozessverwaltung	89
1.6.2	Systemaufrufe zur Dateiverwaltung	91
1.6.3	Systemaufrufe zur Verzeichnisverwaltung	92
1.6.4	Sonstige Systemaufrufe	94
1.6.5	Die Win32-Programmierschnittstelle (API) unter Windows ..	95
1.7	Betriebssystemstrukturen	98
1.7.1	Monolithische Systeme	99
1.7.2	Geschichtete Systeme	100
1.7.3	Mikrokerne	102
1.7.4	Das Client-Server-Modell	105
1.7.5	Virtuelle Maschinen	105
1.7.6	Exokerne	110
1.8	Die Welt aus der Sicht von C	111
1.8.1	Die Programmiersprache C	111
1.8.2	Header-Dateien	112
1.8.3	Große Programmierprojekte	113
1.8.4	Das Laufzeitmodell	114
1.9	Forschung im Bereich der Betriebssysteme	115
1.10	Überblick über das Buch	116
1.11	Metrische Einheiten	117
	Zusammenfassung	119
	Übungen	120
 Kapitel 2 Prozesse und Threads		125
2.1	Prozesse	126
2.1.1	Das Prozessmodell	127
2.1.2	Prozesserzeugung	129
2.1.3	Prozessbeendigung	132
2.1.4	Prozesshierarchien	133
2.1.5	Prozesszustände	133
2.1.6	Implementierung von Prozessen	136
2.1.7	Modellierung der Multiprogrammierung	138
2.2	Threads	139
2.2.1	Der Gebrauch von Threads	140
2.2.2	Das klassische Thread-Modell	146
2.2.3	POSIX-Threads	150
2.2.4	Implementierung von Threads im Benutzeradressraum	153
2.2.5	Implementierung von Threads im Kern	156
2.2.6	Hybride Implementierungen	157
2.2.7	Scheduler-Aktivierungen	158
2.2.8	Pop-up-Threads	160
2.2.9	Einfachthread-Code in Multithread-Code umwandeln	161
2.3	Interprozesskommunikation	165
2.3.1	Race Conditions	165
2.3.2	Kritische Regionen	167

2.3.3	Wechselseitiger Ausschluss mit aktivem Warten	168
2.3.4	Sleep und Wakeup	174
2.3.5	Semaphor	177
2.3.6	Mutex	179
2.3.7	Monitor	186
2.3.8	Nachrichtenaustausch	192
2.3.9	Barrieren	195
2.3.10	Sperren vermeiden: das Read-Copy-Update-Schema	197
2.4	Scheduling	199
2.4.1	Einführung in das Scheduling	199
2.4.2	Scheduling in Stapelverarbeitungssystemen	207
2.4.3	Scheduling in interaktiven Systemen	209
2.4.4	Scheduling in Echtzeitsystemen	215
2.4.5	Strategie versus Mechanismus	217
2.4.6	Thread-Scheduling	217
2.5	Klassische Probleme der Interprozesskommunikation	219
2.5.1	Das Philosophenproblem	219
2.5.2	Das Leser-Schreiber-Problem	223
2.6	Forschung zu Prozessen und Threads	224
	Zusammenfassung	226
	Übungen	227
 Kapitel 3 Speicherverwaltung		 237
3.1	Systeme ohne Speicherabstraktion	239
3.2	Speicherabstraktion: Adressraum	242
3.2.1	Das Konzept des Adressraums	243
3.2.2	Swapping	245
3.2.3	Verwaltung von freiem Speicher	248
3.3	Virtueller Speicher	251
3.3.1	Paging	253
3.3.2	Seitentabellen	257
3.3.3	Beschleunigung des Paging	259
3.3.4	Seitentabellen für große Speicherbereiche	263
3.4	Seitenerersatzalgorithmen	267
3.4.1	Der optimale Algorithmus zur Seitenersetzung	268
3.4.2	Der Not-Recently-Used-Algorithmus (NRU)	269
3.4.3	Der First-In-First-Out-Algorithmus (FIFO)	270
3.4.4	Der Second-Chance-Algorithmus	271
3.4.5	Der Clock-Algorithmus	272
3.4.6	Der Least-Recently-Used-Algorithmus (LRU)	272
3.4.7	Simulation von LRU durch Software	273
3.4.8	Der Working-Set-Algorithmus	275
3.4.9	Der WSClock-Algorithmus	279
3.4.10	Zusammenfassung der Seiteneretzungsstrategien	281

3.5	Entwurfskriterien für Paging-Systeme	282
3.5.1	Lokale versus globale Zuteilungsstrategien	282
3.5.2	Lastkontrolle.	285
3.5.3	Seitengröße.	286
3.5.4	Trennung von Befehls- und Datenräumen	287
3.5.5	Gemeinsame Seiten	288
3.5.6	Gemeinsame Bibliotheken	290
3.5.7	Memory-Mapped-Dateien	292
3.5.8	Bereinigungsstrategien.	293
3.5.9	Schnittstelle des virtuellen Speichersystems	294
3.6	Implementierungsaspekte	295
3.6.1	Aufgaben des Betriebssystems beim Paging	295
3.6.2	Behandlung von Seitenfehlern	296
3.6.3	Sicherung von unterbrochenen Befehlen	297
3.6.4	Sperren von Seiten im Speicher	298
3.6.5	Hintergrundspeicher	299
3.6.6	Trennung von Strategie und Mechanismus	301
3.7	Segmentierung.	302
3.7.1	Implementierung von Segmentierung	306
3.7.2	Segmentierung mit Paging: MULTICS	306
3.7.3	Segmentierung mit Paging: x86 von Intel	310
3.8	Forschung zur Speicherverwaltung	314
	Zusammenfassung	316
	Übungen.	317
Kapitel 4	Dateisysteme	329
4.1	Dateien	332
4.1.1	Benennung von Dateien	332
4.1.2	Dateistruktur.	334
4.1.3	Dateitypen	336
4.1.4	Dateizugriff.	338
4.1.5	Dateiattribute	338
4.1.6	Dateioperationen	340
4.1.7	Beispielprogramm mit Aufrufen zum Dateisystem	341
4.2	Verzeichnisse	344
4.2.1	Verzeichnissysteme mit einer Ebene.	345
4.2.2	Hierarchische Verzeichnissysteme	345
4.2.3	Pfadnamen	346
4.2.4	Operationen auf Verzeichnissen	348
4.3	Implementierung von Dateisystemen	350
4.3.1	Layout eines Dateisystems.	350
4.3.2	Implementierung von Dateien	351
4.3.3	Implementierung von Verzeichnissen	357
4.3.4	Gemeinsam benutzte Dateien	360
4.3.5	Log-basierte Dateisysteme	363

4.3.6	Journaling-Dateisysteme	365
4.3.7	Virtuelle Dateisysteme	367
4.4	Dateisystemverwaltung und -optimierung.	371
4.4.1	Plattenspeicherverwaltung	371
4.4.2	Sicherung von Dateisystemen.	379
4.4.3	Konsistenz eines Dateisystems	385
4.4.4	Performanz eines Dateisystems.	388
4.4.5	Defragmentierung von Plattenspeicher	393
4.5	Beispiele von Dateisystemen	394
4.5.1	Das MS-DOS-Dateisystem	395
4.5.2	Das UNIX-V7-Dateisystem	399
4.5.3	CD-ROM-Dateisysteme	401
4.6	Forschung zu Dateisystemen	407
	Zusammenfassung.	408
	Übungen	409
Kapitel 5	Eingabe und Ausgabe	415
5.1	Grundlagen der Ein-/Ausgabehardware.	417
5.1.1	Ein-/Ausgabegeräte	417
5.1.2	Controller.	419
5.1.3	Memory-Mapped-Ein-/Ausgabe	420
5.1.4	Direct Memory Access.	424
5.1.5	Interrupts	427
5.2	Grundlagen der Ein-/Ausgabesoftware	432
5.2.1	Ziele von Ein-/Ausgabesoftware.	432
5.2.2	Programmierte Ein-/Ausgabe	434
5.2.3	Interruptgesteuerte Ein-/Ausgabe.	435
5.2.4	Ein-/Ausgabe mit DMA	436
5.3	Schichten der Ein-/Ausgabesoftware.	437
5.3.1	Unterbrechungsroutinen.	438
5.3.2	Geratetreiber	439
5.3.3	Geräteunabhängige Ein-/Ausgabesoftware.	444
5.3.4	Ein-/Ausgabesoftware im Benutzeradressraum	450
5.4	Plattenspeicher	452
5.4.1	Hardware von Plattenspeichern	452
5.4.2	Formatierung von Plattenspeichern	460
5.4.3	Strategien zur Steuerung des Plattenarms	464
5.4.4	Fehlerbehandlung	468
5.4.5	Zuverlässiger Speicher	471
5.5	Uhren.	475
5.5.1	Hardwareuhren	475
5.5.2	Softwareuhren	477
5.5.3	Soft-Timer	480
5.6	Benutzungsschnittstellen: Tastatur, Maus, Bildschirm	482
5.6.1	Eingabesoftware.	482
5.6.2	Ausgabesoftware	488

5.7	Thin Clients	506
5.8	Energieverwaltung	508
5.8.1	Hardwareaspekte	509
5.8.2	Betriebssystemaspekte	511
5.8.3	Energieverwaltung und Anwendungsprogramme	517
5.9	Forschung im Bereich Ein-/Ausgabe	518
	Zusammenfassung	521
	Übungen	522

Kapitel 6 Deadlocks 531

6.1	Ressourcen	533
6.1.1	Unterbrechbare und nicht unterbrechbare Ressourcen	533
6.1.2	Ressourcenanforderung	534
6.2	Einführung in Deadlocks	536
	Definition: <i>Deadlock</i>	536
6.2.1	Voraussetzungen für Ressourcen-Deadlocks	537
6.2.2	Modellierung von Deadlocks	537
6.3	Der Vogel-Strauß-Algorithmus	541
6.4	Erkennen und Beheben von Deadlocks	541
6.4.1	Deadlock-Erkennung bei einer Ressource je Typ	541
6.4.2	Deadlock-Erkennung bei mehreren Ressourcen je Typ	544
6.4.3	Beheben von Deadlocks	547
6.5	Verhinderung von Deadlocks (Avoidance)	548
6.5.1	Ressourcenspuren	549
6.5.2	Sichere und unsichere Zustände	550
6.5.3	Der Bankier-Algorithmus für eine einzelne Ressource	551
6.5.4	Der Bankier-Algorithmus für mehrere Ressourcen	553
6.6	Vermeidung von Deadlocks (Prevention)	554
6.6.1	Unterlaufen der Bedingung des wechselseitigen Ausschlusses	554
6.6.2	Unterlaufen der Hold-and-Wait-Bedingung	555
6.6.3	Unterlaufen der Bedingung der Ununterbrechbarkeit	556
6.6.4	Unterlaufen der zyklischen Wartebedingung	556
6.7	Weitere Themen zu Deadlocks	558
6.7.1	Zwei-Phasen-Sperren	558
6.7.2	Kommunikationsdeadlocks	558
6.7.3	Livelock	560
6.7.4	Verhungern	563
6.8	Forschung zu Deadlocks	563
	Zusammenfassung	565
	Übungen	566

Kapitel 7 Virtualisierung und die Cloud 573

7.1	Geschichte der Virtualisierung	577
7.2	Anforderungen für die Virtualisierung	578
7.3	Typ-1- und Typ-2-Hypervisoren	581

7.4	Techniken für die effiziente Virtualisierung	582
7.4.1	Das Nichtvirtualisierbare virtualisieren	583
7.4.2	Kosten der Virtualisierung	586
7.5	Der Hypervisor: ein idealer Mikrokern?	587
7.6	Speichervirtualisierung	590
7.7	Ein-/Ausgabevirtualisierung	595
7.8	Virtual Appliances	599
7.9	Virtuelle Maschinen bei Mehrkernprozessoren.	600
7.10	Fragen bezüglich der Lizenzierung	601
7.11	Clouds	601
7.11.1	Clouds-as-a-Service	602
7.11.2	Migration von virtuellen Maschinen	603
7.11.3	Checkpointing	604
7.12	Fallstudie: VMware.	604
7.12.1	Die Anfänge von VMware	605
7.12.2	VMware Workstation.	607
7.12.3	Aufgaben bei der Virtualisierungseinführung im x86	608
7.12.4	VMware Workstation: Überblick über die Lösungen.	609
7.12.5	Die Weiterentwicklung von VMware Workstation	619
7.12.6	ESX Server: Typ-1-Hypervisor von VMware	620
7.13	Forschung zu Virtualisierung und der Cloud	622
	Übungen	624
 Kapitel 8 Multiprozessorsysteme		627
8.1	Multiprozessoren	631
8.1.1	Hardware von Multiprozessoren	631
8.1.2	Betriebssystemarten für Multiprozessoren.	643
8.1.3	Synchronisation in Multiprozessorsystemen.	647
8.1.4	Multiprozessor-Scheduling.	652
8.2	Multicomputer	659
8.2.1	Hardware von Multicomputern	660
8.2.2	Low-Level-Kommunikationssoftware.	664
8.2.3	Kommunikationssoftware auf Benutzerebene	668
8.2.4	Entfernter Prozeduraufruf (RPC).	671
8.2.5	Distributed Shared Memory	674
8.2.6	Multicomputer-Scheduling.	679
8.2.7	Lastausgleich	680
8.3	Verteilte Systeme.	683
8.3.1	Netzwerkhardware	686
8.3.2	Netzwerkdienste und -protokolle	689
8.3.3	Dokumentenbasierte Middleware.	693
8.3.4	Dateisystembasierte Middleware	695
8.3.5	Objektbasierte Middleware	700
8.3.6	Koordinationsbasierte Middleware	702
8.4	Forschung zu Multiprozessorsystemen	705
	Zusammenfassung.	707
	Übungen	708

Kapitel 9	IT-Sicherheit	715
9.1	Die Sicherheitsumgebung	719
9.1.1	Bedrohungen	719
9.1.2	Angreifer	723
9.2	Betriebssystemsicherheit	723
9.2.1	Können wir sichere Systeme bauen?	724
9.2.2	Trusted Computing Base	726
9.3	Zugriff auf Ressourcen steuern	727
9.3.1	Schutzdomänen	727
9.3.2	Zugriffskontrolllisten	730
9.3.3	Capabilities	733
9.4	Formale Modelle von sicheren Systemen	736
9.4.1	Multilevel-Sicherheit	738
9.4.2	Verdeckte Kanäle	741
9.5	Grundlagen der Kryptografie	746
9.5.1	Symmetrische Kryptografie	747
9.5.2	Public-Key-Kryptografie	748
9.5.3	Einwegfunktionen	749
9.5.4	Digitale Signaturen	750
9.5.5	Trusted Platform Module (TPM)	752
9.6	Authentifizierung	754
9.6.1	Authentifizierung durch Besitz	762
9.6.2	Biometrische Authentifizierung	765
9.7	Ausnutzen von Sicherheitslücken	768
9.7.1	Pufferüberlaufangriffe	770
9.7.2	Formatstring-Angriffe	780
9.7.3	Hangende Zeiger	783
9.7.4	NULL-Zeiger-Dereferenzierungsangriff	784
9.7.5	Angriffe durch Ganzzahlüberlauf	785
9.7.6	Angriffe durch Kommando-Injektion	786
9.7.7	Time-of-Check-to-Time-of-Use-Angriff	787
9.8	Insider-Angriffe	788
9.8.1	Logische Bomben	789
9.8.2	Hintertüren	789
9.8.3	Login-Spoofing	790
9.9	Malware	791
9.9.1	Trojanische Pferde	795
9.9.2	Viren	797
9.9.3	Würmer	808
9.9.4	Spyware	811
9.9.5	Rootkits	815
9.10	Abwehrmechanismen	820
9.10.1	Firewalls	820
9.10.2	Antiviren- und Anti-Antivirentechniken	823
9.10.3	Codesignierung	830
9.10.4	Jailing	832

9.10.5	Modellbasierte Angriffserkennung	833
9.10.6	Kapselung von mobilem Code	834
9.10.7	Java-Sicherheit	839
9.11	Forschung zum Thema IT-Sicherheit	842
	Zusammenfassung	844
	Übungen	845
Kapitel 10	Fallstudie 1: Linux	855
10.1	Die Geschichte von UNIX und Linux	857
10.1.1	UNICS	857
10.1.2	PDP-11-UNIX	858
10.1.3	Portable UNIX-Varianten	859
10.1.4	Berkeley-UNIX	860
10.1.5	Standard-UNIX	861
10.1.6	MINIX	862
10.1.7	Linux	864
10.2	Überblick über Linux	866
10.2.1	Ziele von Linux	867
10.2.2	Schnittstellen zu Linux	868
10.2.3	Die Shell	870
10.2.4	Hilfsprogramme unter Linux	873
10.2.5	Kernstruktur	875
10.3	Prozesse in Linux	878
10.3.1	Grundlegende Konzepte	878
10.3.2	Systemaufrufe zur Prozessverwaltung in Linux	881
10.3.3	Implementierung von Prozessen und Threads in Linux	886
10.3.4	Scheduling in Linux	893
10.3.5	Starten von Linux	898
10.4	Speicherverwaltung in Linux	901
10.4.1	Grundlegende Konzepte	902
10.4.2	Systemaufrufe zur Speicherverwaltung in Linux	905
10.4.3	Implementierung der Speicherverwaltung in Linux	906
10.4.4	Paging in Linux	913
10.5	Ein-/Ausgabe in Linux	917
10.5.1	Grundlegende Konzepte	917
10.5.2	Netzwerkimplementierung	919
10.5.3	Systemaufrufe zur Ein-/Ausgabe in Linux	920
10.5.4	Implementierung der Ein-/Ausgabe in Linux	921
10.5.5	Linux-Kernmodule	925
10.6	Das Linux-Dateisystem	926
10.6.1	Grundlegende Konzepte	926
10.6.2	Systemaufrufe zur Dateiverwaltung in Linux	931
10.6.3	Implementierung des Linux-Dateisystems	935
10.6.4	NFS – das Netzwerkdateisystem	945

10.7	Sicherheit in Linux	952
10.7.1	Grundlegende Konzepte	952
10.7.2	Systemaufrufe zu Sicherheitsfunktionen in Linux	954
10.7.3	Implementierung von Sicherheitsfunktionen in Linux	955
10.8	Android	956
10.8.1	Android und Google	957
10.8.2	Geschichte von Android	958
10.8.3	Entwurfsziele	962
10.8.4	Architektur von Android	963
10.8.5	Linux-Erweiterungen	965
10.8.6	Dalvik	969
10.8.7	Interprozesskommunikation mit Binder	971
10.8.8	Android-Anwendungen	980
10.8.9	Intents	992
10.8.10	Sandboxen in Anwendungen	994
10.8.11	Sicherheit	995
10.8.12	Prozessmodell	1001
	Zusammenfassung	1007
	Übungen	1009
Kapitel 11 Fallstudie 2: Windows 8		1017
11.1	Die Geschichte von Windows bis Windows 8.1	1018
11.1.1	Die 1980er: MS-DOS	1019
11.1.2	Die 1990er: MS-DOS-basiertes Windows	1020
11.1.3	Die 2000er: NT-basiertes Windows	1020
11.1.4	Windows Vista	1023
11.1.5	Die 2010er: Modern Windows	1025
11.2	Programmierung von Windows	1026
11.2.1	Die native NT-Programmierschnittstelle	1030
11.2.2	Die Win32-Programmierschnittstelle	1034
11.2.3	Die Windows-Registrierungsdatenbank	1039
11.3	Systemstruktur	1042
11.3.1	Betriebssystemstruktur	1042
11.3.2	Starten von Windows	1060
11.3.3	Implementierung des Objekt-Managers	1062
11.3.4	Subsysteme, DLLs und Dienste im Benutzermodus	1074
11.4	Prozesse und Threads in Windows	1077
11.4.1	Grundlegende Konzepte	1077
11.4.2	API-Aufrufe zur Job-, Prozess-, Thread- und Fiberverwaltung	1084
11.4.3	Implementierung von Prozessen und Threads	1091
11.5	Speicherverwaltung	1099
11.5.1	Grundlegende Konzepte	1099
11.5.2	Systemaufrufe zur Speicherverwaltung	1104
11.5.3	Implementierung der Speicherverwaltung	1105
11.6	Caching in Windows	1116

11.7	Ein-/Ausgabe in Windows	1118
11.7.1	Grundlegende Konzepte	1118
11.7.2	API-Aufrufe für die Ein-/Ausgabe	1120
11.7.3	Implementierung der Ein-/Ausgabe	1123
11.8	Das Windows-NT-Dateisystem	1128
11.8.1	Grundlegende Konzepte	1128
11.8.2	Implementierung des NT-Dateisystems	1129
11.9	Energieverwaltung in Windows	1141
11.10	IT-Sicherheit in Windows 8	1143
11.10.1	Grundlegende Konzepte	1144
11.10.2	API-Aufrufe zu Sicherheitsfunktionen	1147
11.10.3	Implementierung von Sicherheitsfunktionen	1148
11.10.4	Mitigation in der IT-Sicherheit	1150
	Zusammenfassung	1154
	Übungen	1156
 Kapitel 12 Entwurf von Betriebssystemen		1161
12.1	Das Problem des Entwurfs	1163
12.1.1	Ziele	1163
12.1.2	Warum ist es schwierig, ein Betriebssystem zu entwerfen?	1164
12.2	Schnittstellenentwurf	1166
12.2.1	Leitlinien	1167
12.2.2	Paradigmen	1169
12.2.3	Die Systemaufrufchnittstelle	1173
12.3	Implementierung	1176
12.3.1	Systemstruktur	1176
12.3.2	Mechanismus versus Strategie	1181
12.3.3	Orthogonalität	1182
12.3.4	Namensräume	1183
12.3.5	Zeitpunkt des Bindens	1185
12.3.6	Statische versus dynamische Strukturen	1186
12.3.7	Top-down- versus Bottom-up-Implementierung	1187
12.3.8	Synchrone versus asynchrone Kommunikation	1188
12.3.9	Nützliche Techniken	1190
12.4	Performanz	1196
12.4.1	Warum sind Betriebssysteme langsam?	1196
12.4.2	Was sollte verbessert werden?	1197
12.4.3	Der Zielkonflikt zwischen Laufzeit und Speicherplatz	1198
12.4.4	Caching	1201
12.4.5	Hints	1202
12.4.6	Ausnutzen der Lokalität	1203
12.4.7	Optimieren des Normalfalls	1203

12.5	Projektverwaltung	1204
12.5.1	Der Mythos vom Mann-Monat.	1204
12.5.2	Teamstruktur	1206
12.5.3	Die Bedeutung der Erfahrung	1208
12.5.4	No Silver Bullet	1209
12.6	Trends beim Entwurf von Betriebssystemen	1210
12.6.1	Virtualisierung und die Cloud.	1210
12.6.2	Vielkern-Prozessoren	1210
12.6.3	Betriebssysteme mit großem Adressraum	1211
12.6.4	Nahtloser Datenzugriff.	1212
12.6.5	Batteriebetriebene Computer.	1213
12.6.6	Eingebettete Systeme	1214
	Zusammenfassung	1215
	Übungen.	1216
 Kapitel 13 Bibliografie		 1221
13.1	Empfehlungen für weiterführende Literatur	1222
13.1.1	Einführung	1222
13.1.2	Prozesse und Threads	1223
13.1.3	Speicherverwaltung	1224
13.1.4	Dateisysteme.	1224
13.1.5	Ein- und Ausgabe	1225
13.1.6	Deadlocks	1225
13.1.7	Virtualisierung und die Cloud.	1226
13.1.8	Multiprozessorsysteme	1227
13.1.9	IT-Sicherheit.	1228
13.1.10	Fallstudie 1: UNIX, Linux und Android.	1230
13.1.11	Fallstudie 2: Windows 8	1230
13.1.12	Betriebssystementwurf.	1231
13.2	Alphabetische Literaturliste	1232
 Namensregister		 1265
 Register		 1267