

Inhaltsübersicht

Vorwort	23
Kapitel 1 Einführung	29
Kapitel 2 Prozesse und Threads	123
Kapitel 3 Speicherverwaltung	227
Kapitel 4 Dateisysteme	313
Kapitel 5 Eingabe und Ausgabe	395
Kapitel 6 Deadlocks	511
Kapitel 7 Multimedia-Betriebssysteme	549
Kapitel 8 Multiprozessorsysteme	611
Kapitel 9 IT-Sicherheit	709
Kapitel 10 Fallstudie 1: Linux	831
Kapitel 11 Fallstudie 2: Windows Vista	935
Kapitel 12 Fallstudie 3: Symbian OS	1065
Kapitel 13 Entwurf von Betriebssystemen	1099
Kapitel 14 Bibliografie	1153
Fachwörterverzeichnis	1199
Namensregister	1219
Register	1221

Inhaltsverzeichnis

Vorwort	23
Kapitel 1 Einführung	29
1.1 Was ist ein Betriebssystem?	33
1.1.1 Das Betriebssystem als eine erweiterte Maschine	33
1.1.2 Das Betriebssystem als Ressourcenverwalter	35
1.2 Geschichte der Betriebssysteme	37
1.2.1 Die erste Generation (1945–1955) – auf Basis von Elektronenröhren	38
1.2.2 Die zweite Generation (1955–1965) – Transistoren und Stapelverarbeitungssysteme	38
1.2.3 Die dritte Generation (1965–1980) – integrierte Schaltkreise und Multiprogrammierung	41
1.2.4 Die vierte Generation (1980 bis heute) – der Personalcomputer (PC)	46
1.3 Überblick über die Computer-Hardware	50
1.3.1 Prozessoren	51
1.3.2 Arbeitsspeicher	55
1.3.3 Festplatten	58
1.3.4 Magnetbänder	59
1.3.5 Ein-/Ausgabegeräte	60
1.3.6 Bussysteme	63
1.3.7 Hochfahren des Computers	66
1.4 Die Betriebssystemfamilie	67
1.4.1 Betriebssysteme für Großrechner	67
1.4.2 Betriebssysteme für Server	68
1.4.3 Betriebssysteme für Multiprozessorsysteme	68
1.4.4 Betriebssysteme für Personalcomputer	68
1.4.5 Betriebssysteme für Handheld-Computer	69
1.4.6 Betriebssysteme für eingebettete Systeme	69
1.4.7 Betriebssysteme für Sensorknoten	69
1.4.8 Echtzeitbetriebssysteme	70
1.4.9 Betriebssysteme für Smart Cards	71
1.5 Betriebssystemkonzepte	71
1.5.1 Prozesse	71
1.5.2 Adressräume	74
1.5.3 Dateien	75
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	98
1.7.2	Geschichtete Systeme	100
1.7.3	Mikrokerne	101
1.7.4	Das Client-Server-Modell	104
1.7.5	Virtuelle Maschinen	105
1.7.6	Exokerne	109
1.8	Die Welt aus der Sicht von C	109
1.8.1	Die Programmiersprache C	109
1.8.2	Header-Dateien	110
1.8.3	Große Programmierprojekte	111
1.8.4	Das Laufzeitmodell	113
1.9	Forschung im Bereich der Betriebssysteme	113
1.10	Überblick über das Buch	115
1.11	Metrische Einheiten	116
	Zusammenfassung	117
	Übungen	118

Kapitel 2 Prozesse und Threads 123

2.1	Prozesse	124
2.1.1	Das Prozessmodell	125
2.1.2	Prozesserzeugung	127
2.1.3	Prozessbeendigung	129
2.1.4	Prozesshierarchien	130
2.1.5	Prozesszustände	131
2.1.6	Implementierung von Prozessen	133
2.1.7	Modellierung der Multiprogrammierung	135
2.2	Threads	137
2.2.1	Der Gebrauch von Threads	137
2.2.2	Das klassische Thread-Modell	143
2.2.3	POSIX-Threads	147
2.2.4	Implementierung von Threads im Benutzeradressraum	149
2.2.5	Implementierung von Threads im Kern	153
2.2.6	Hybride Implementierungen	154
2.2.7	Scheduler-Aktivierungen	155
2.2.8	Pop-up-Threads	156
2.2.9	Einfach-Thread-Code in Mehrfach-Thread-Code umwandeln ..	157
2.3	Interprozesskommunikation	161
2.3.1	Race Conditions	161
2.3.2	Kritische Regionen	163
2.3.3	Wechselseitiger Ausschluss mit aktivem Warten	164

2.3.4	Sleep und Wakeup	170
2.3.5	Semaphor	173
2.3.6	Mutex	176
2.3.7	Monitor	181
2.3.8	Nachrichtenaustausch	187
2.3.9	Barrieren	190
2.4	Scheduling	192
2.4.1	Einführung in das Scheduling	192
2.4.2	Scheduling in Stapelverarbeitungssystemen	200
2.4.3	Scheduling in interaktiven Systemen	202
2.4.4	Scheduling in Echtzeitsystemen	208
2.4.5	Strategie versus Mechanismus	209
2.4.6	Thread-Scheduling	210
2.5	Klassische Probleme der Interprozesskommunikation	212
2.5.1	Das Philosophenproblem	212
2.5.2	Das Leser-Schreiber-Problem	215
2.6	Forschung zu Prozessen und Threads	217
	Zusammenfassung	218
	Übungen	219

Kapitel 3 Speicherverwaltung 227

3.1	Systeme ohne Speicherabstraktion	229
3.2	Speicherabstraktion: Adressräume	232
3.2.1	Das Konzept des Adressraumes	233
3.2.2	Swapping	235
3.2.3	Verwaltung freien Speichers	237
3.3	Virtueller Speicher	241
3.3.1	Paging	242
3.3.2	Seitentabellen	246
3.3.3	Beschleunigung des Paging	248
3.3.4	Seitentabellen für große Speicherbereiche	251
3.4	Seitenersetzungsalgorithmen	255
3.4.1	Der optimale Algorithmus zur Seitenersetzung	256
3.4.2	Der Not-Recently-Used-Algorithmus (NRU)	257
3.4.3	Der First-In-First-Out-Algorithmus (FIFO)	258
3.4.4	Der Second-Chance-Algorithmus	258
3.4.5	Der Clock-Algorithmus	259
3.4.6	Der Least-Recently-Used-Algorithmus (LRU)	260
3.4.7	Simulation von LRU durch Software	261
3.4.8	Der Working-Set-Algorithmus	263
3.4.9	Der WSClock-Algorithmus	267
3.4.10	Zusammenfassung der Seitenersetzungsstrategien	269
3.5	Entwurfskriterien für Paging-Systeme	270
3.5.1	Lokale versus globale Zuteilungsstrategien	270
3.5.2	Lastkontrolle	273
3.5.3	Seitengröße	273

3.5.4	Trennung von Befehls- und Datenräumen	275
3.5.5	Gemeinsame Seiten	276
3.5.6	Gemeinsame Bibliotheken	277
3.5.7	Memory-Mapped-Dateien	280
3.5.8	Bereinigungsstrategien	280
3.5.9	Schnittstelle des virtuellen Speichersystems	281
3.6	Implementierungsaspekte	282
3.6.1	Aufgaben des Betriebssystems beim Paging	282
3.6.2	Behandlung von Seitenfehlern	283
3.6.3	Sicherung von unterbrochenen Befehlen	284
3.6.4	Sperren von Seiten im Speicher	285
3.6.5	Hintergrundspeicher	286
3.6.6	Trennung von Strategie und Mechanismus	288
3.7	Segmentierung	289
3.7.1	Implementierung von Segmentierung	293
3.7.2	Segmentierung mit Paging: MULTICS	294
3.7.3	Segmentierung mit Paging: der Intel Pentium	297
3.8	Forschung zur Speicherverwaltung	302
	Zusammenfassung	303
	Übungen	304

Kapitel 4 Dateisysteme 313

4.1	Dateien	316
4.1.1	Benennung von Dateien	316
4.1.2	Dateistruktur	318
4.1.3	Dateitypen	319
4.1.4	Dateizugriff	321
4.1.5	Dateiattribute	322
4.1.6	Dateioperationen	324
4.1.7	Beispielprogramm mit Aufrufen zum Dateisystem	325
4.2	Verzeichnisse	328
4.2.1	Verzeichnissysteme mit einer Ebene	328
4.2.2	Hierarchische Verzeichnissysteme	329
4.2.3	Pfadnamen	329
4.2.4	Operationen auf Verzeichnissen	332
4.3	Implementierung von Dateisystemen	333
4.3.1	Layout eines Dateisystems	333
4.3.2	Implementierung von Dateien	334
4.3.3	Implementierung von Verzeichnissen	340
4.3.4	Gemeinsam benutzte Dateien	343
4.3.5	Log-basierte Dateisysteme	346
4.3.6	Journaling-Dateisysteme	348
4.3.7	Virtuelle Dateisysteme	350
4.4	Dateisystemverwaltung und -optimierung	353
4.4.1	Plattenspeicherverwaltung	354
4.4.2	Sicherung von Dateisystemen	361
4.4.3	Konsistenz eines Dateisystems	367

4.4.4	Performanz eines Dateisystems	371
4.4.5	Defragmentierung von Plattenspeicher	376
4.5	Beispiele von Dateisystemen	377
4.5.1	CD-ROM-Dateisysteme	377
4.5.2	Das MS-DOS-Dateisystem	383
4.5.3	Das UNIX-V7-Dateisystem	386
4.6	Forschung zu Dateisystemen	389
	Zusammenfassung.	390
	Übungen	390

Kapitel 5 Eingabe und Ausgabe 395

5.1	Grundlagen der Ein-/Ausgabe-Hardware	396
5.1.1	Ein-/Ausgabegeräte	396
5.1.2	Controller.	398
5.1.3	Memory-Mapped-Ein-/Ausgabe	399
5.1.4	Direct Memory Access (DMA)	403
5.1.5	Interrupts	406
5.2	Grundlagen der Ein-/Ausgabe-Software	411
5.2.1	Ziele von Ein-/Ausgabe-Software	411
5.2.2	Programmierte Ein-/Ausgabe	413
5.2.3	Interruptgesteuerte Ein-/Ausgabe	414
5.2.4	Ein-/Ausgabe mit DMA	415
5.3	Schichten der Ein-/Ausgabe-Software	416
5.3.1	Unterbrechungsrouinen	417
5.3.2	Gerätetreiber	418
5.3.3	Geräteunabhängige Ein-/Ausgabe-Software	422
5.3.4	Ein-/Ausgabe-Software im Benutzeradressraum	428
5.4	Plattenspeicher	430
5.4.1	Hardware von Plattenspeichern	430
5.4.2	Formatierung von Plattenspeichern	447
5.4.3	Strategien zur Steuerung des Plattenarms	451
5.4.4	Fehlerbehandlung	454
5.4.5	Zuverlässiger Speicher	457
5.5	Uhren.	461
5.5.1	Hardwareuhren	461
5.5.2	Softwareuhren	463
5.5.3	Soft-Timer	466
5.6	Benutzungsschnittstellen: Tastatur, Maus, Bildschirm	467
5.6.1	Eingabe-Software	468
5.6.2	Ausgabe-Software	473
5.7	Thin Clients.	490
5.8	Energieverwaltung	492
5.8.1	Hardwareaspekte	493
5.8.2	Betriebssystemaspekte	495
5.8.3	Energieverwaltung und Anwendungsprogramme	500

5.9	Forschung im Bereich Ein-/Ausgabe	502
	Zusammenfassung	503
	Übungen	504

Kapitel 6 Deadlocks 511

6.1	Ressourcen	513
6.1.1	Unterbrechbare und nicht unterbrechbare Ressourcen	513
6.1.2	Ressourcenanforderung	514
6.2	Einführung in Deadlocks	516
6.2.1	Voraussetzungen für Ressourcen-Deadlocks	517
6.2.2	Modellierung von Deadlocks	517
6.3	Der Vogel-Strauß-Algorithmus	520
6.4	Erkennen und Beheben von Deadlocks	521
6.4.1	Deadlock-Erkennung bei einer Ressource je Typ	521
6.4.2	Deadlock-Erkennung bei mehreren Ressourcen je Typ	523
6.4.3	Beheben von Deadlocks	526
6.5	Verhinderung von Deadlocks (Avoidance)	528
6.5.1	Ressourcenspuren	528
6.5.2	Sichere und unsichere Zustände	529
6.5.3	Der Bankier-Algorithmus für eine einzelne Ressource	531
6.5.4	Der Bankier-Algorithmus für mehrere Ressourcen	532
6.6	Vermeidung von Deadlocks (Prevention)	533
6.6.1	Unterlaufen der Bedingung des wechselseitigen Ausschlusses	534
6.6.2	Unterlaufen der Hold-and-Wait-Bedingung	534
6.6.3	Unterlaufen der Bedingung der Ununterbrechbarkeit	535
6.6.4	Unterlaufen der zyklischen Wartebedingung	535
6.7	Weitere Themen zu Deadlocks	536
6.7.1	Zwei-Phasen-Sperren	537
6.7.2	Kommunikationsdeadlocks	537
6.7.3	Livelock	539
6.7.4	Verhungern	541
6.8	Forschung zu Deadlocks	541
	Zusammenfassung	542
	Übungen	543

Kapitel 7 Multimedia-Betriebssysteme 549

7.1	Einführung in Multimedia	551
7.2	Multimedia-Dateien	555
7.2.1	Codierung von Videodaten	557
7.2.2	Codierung von Audiodaten	560
7.3	Videokompression	561
7.3.1	Der JPEG-Standard	562
7.3.2	Der MPEG-Standard	565
7.4	Audiokompression	568
7.5	Multimedia-Prozess-Scheduling	572
7.5.1	Scheduling von homogenen Prozessen	572
7.5.2	Allgemeines Echtzeit-Scheduling	573

7.5.3	Raten-monotones Scheduling	574
7.5.4	Earliest-Deadline-First-Scheduling	576
7.6	Modelle für Multimedia-Dateisysteme	578
7.6.1	Videorecorder-Steuerfunktionen	579
7.6.2	Near-Video-on-Demand	581
7.6.3	Near-Video-on-Demand mit Videorecorder-Steuerfunktionen	583
7.7	Dateiplatzierung	585
7.7.1	Platzierung einer Datei auf einer einzelnen Platte	585
7.7.2	Zwei alternative Strategien zur Dateioorganisation	586
7.7.3	Platzierung von Dateien für Near-Video-on-Demand	590
7.7.4	Platzierung mehrerer Dateien auf einer einzelnen Platte	592
7.7.5	Platzierung von Dateien auf mehreren Platten	594
7.8	Caching	596
7.8.1	Block-Caching	597
7.8.2	Datei-Caching	599
7.9	Plattenspeicher-Scheduling für Multimedia	599
7.9.1	Statisches Plattenspeicher-Scheduling	599
7.9.2	Dynamisches Plattenspeicher-Scheduling	601
7.10	Forschung im Bereich Multimedia	603
	Zusammenfassung	604
	Übungen	605

Kapitel 8	Multiprozessorsysteme	611
8.1	Multiprozessoren	615
8.1.1	Hardware von Multiprozessoren	615
8.1.2	Betriebssystemarten für Multiprozessoren	624
8.1.3	Synchronisation in Multiprozessorsystemen	628
8.1.4	Multiprozessor-Scheduling	633
8.2	Multicomputer	639
8.2.1	Hardware von Multicomputern	640
8.2.2	Low-Level-Kommunikationssoftware	644
8.2.3	Kommunikationssoftware auf Benutzerebene	646
8.2.4	Entfernter Prozeduraufruf (RPC)	650
8.2.5	Distributed Shared Memory	652
8.2.6	Multicomputer-Scheduling	657
8.2.7	Lastausgleich	658
8.3	Virtualisierung	661
8.3.1	Anforderungen für die Virtualisierung	663
8.3.2	Typ-1-Hypervisor	664
8.3.3	Typ-2-Hypervisor	665
8.3.4	Paravirtualisierung	667
8.3.5	Speichervirtualisierung	669
8.3.6	Ein-/Ausgabevirtualisierung	671
8.3.7	Virtual Appliances	672
8.3.8	Virtuelle Maschinen bei Mehrkernprozessoren	673
8.3.9	Fragen bezüglich der Lizenzierung	674

8.4	Verteilte Systeme	674
8.4.1	Netzwerkhardware	677
8.4.2	Netzwerkdienste und -protokolle	680
8.4.3	Dokumentenbasierte Middleware	684
8.4.4	Dateisystembasierte Middleware.	686
8.4.5	Objektbasierte Middleware	691
8.4.6	Koordinationsbasierte Middleware.	692
8.4.7	Grid-Systeme	698
8.5	Forschung zu Multiprozessorsystemen	699
	Zusammenfassung.	700
	Übungen.	701

Kapitel 9 IT-Sicherheit 709

9.1	Die Sicherheitsumgebung	712
9.1.1	Bedrohungen	712
9.1.2	Angreifer.	714
9.1.3	Unbeabsichtigter Datenverlust	715
9.2	Grundlagen der Kryptografie.	715
9.2.1	Symmetrische Kryptografie	717
9.2.2	Public-Key-Kryptografie	717
9.2.3	Einwegfunktionen	719
9.2.4	Digitale Signaturen.	719
9.2.5	Trusted Platform Module (TPM).	721
9.3	Schutzmechanismen	722
9.3.1	Schutzdomänen	722
9.3.2	Zugriffskontrolllisten.	724
9.3.3	Capabilities.	727
9.3.4	Vertrauenswürdige Systeme	730
9.3.5	Trusted Computing Base	732
9.3.6	Formale Modelle von sicheren Systemen.	733
9.3.7	Multilevel-Sicherheit.	735
9.3.8	Verdeckte Kanäle	737
9.4	Authentifizierung	742
9.4.1	Authentifizierung durch Passwörter	743
9.4.2	Authentifizierung durch Besitz.	753
9.4.3	Biometrische Authentifizierung	756
9.5	Insider-Angriffe	758
9.5.1	Logische Bomben	759
9.5.2	Falltüren	760
9.5.3	Login-Spoofing.	761
9.6	Das Ausnutzen von Programmierfehlern	762
9.6.1	Pufferüberlaufangriffe	763
9.6.2	Formatstring-Angriffe	765
9.6.3	Return-to-libc-Angriffe.	767
9.6.4	Angriffe durch Ganzzahlüberlauf	769
9.6.5	Angriffe durch Code-Injektion.	769
9.6.6	Privilege-Escalation-Angriff.	771

9.7	Malware.	771
9.7.1	Trojanische Pferde.	774
9.7.2	Viren.	777
9.7.3	Würmer	788
9.7.4	Spyware	791
9.7.5	Rootkits	795
9.8	Abwehrmechanismen.	800
9.8.1	Firewalls	800
9.8.2	Antiviren- und Anti-Antivirentechniken	802
9.8.3	Codesignierung	809
9.8.4	Jailing.	811
9.8.5	Modellbasierte Angriffserkennung.	812
9.8.6	Kapselung von mobilem Code	814
9.8.7	Java-Sicherheit.	818
9.9	Forschung zum Thema IT-Sicherheit.	821
	Zusammenfassung.	822
	Übungen	823
Kapitel 10 Fallstudie 1: Linux		831
10.1	Die Geschichte von UNIX und Linux.	833
10.1.1	UNICS	833
10.1.2	PDP-11-UNIX.	834
10.1.3	Portable UNIX-Varianten	835
10.1.4	Berkeley-UNIX.	836
10.1.5	Standard-UNIX	837
10.1.6	MINIX.	838
10.1.7	Linux	839
10.2	Überblick über Linux	842
10.2.1	Ziele von Linux	842
10.2.2	Schnittstellen zu Linux.	843
10.2.3	Die Shell.	845
10.2.4	Hilfsprogramme unter Linux	848
10.2.5	Kernstruktur.	851
10.3	Prozesse in Linux	853
10.3.1	Grundlegende Konzepte	854
10.3.2	Systemaufrufe zur Prozessverwaltung in Linux	857
10.3.3	Implementierung von Prozessen und Threads in Linux	861
10.3.4	Scheduling in Linux	868
10.3.5	Starten von Linux	871
10.4	Speicherverwaltung in Linux.	874
10.4.1	Grundlegende Konzepte	874
10.4.2	Systemaufrufe zur Speicherverwaltung in Linux	878
10.4.3	Implementierung der Speicherverwaltung in Linux.	879
10.4.4	Paging in Linux	885
10.5	Ein-/Ausgabe in Linux	889
10.5.1	Grundlegende Konzepte	889
10.5.2	Netzwerkimplementierung.	891

10.5.3	Systemaufrufe zur Ein-/Ausgabe in Linux	892
10.5.4	Implementierung der Ein-/Ausgabe in Linux	893
10.5.5	Linux-Kernmodule	897
10.6	Das Linux-Dateisystem	898
10.6.1	Grundlegende Konzepte	898
10.6.2	Systemaufrufe zur Dateiverwaltung in Linux	903
10.6.3	Implementierung des Linux-Dateisystems	907
10.6.4	NFS – das Netzwerkdateisystem	916
10.7	Sicherheit in Linux	923
10.7.1	Grundlegende Konzepte	923
10.7.2	Systemaufrufe zu Sicherheitsfunktionen in Linux	925
10.7.3	Implementierung von Sicherheitsfunktionen in Linux	926
	Zusammenfassung	927
	Übungen	929

Kapitel 11 Fallstudie 2: Windows Vista 935

11.1	Die Geschichte von Windows Vista	936
11.1.1	Die 1980er: MS-DOS	937
11.1.2	Die 1990er: MS-DOS-basiertes Windows	937
11.1.3	Die 2000er: NT-basiertes Windows	938
11.1.4	Windows Vista	941
11.2	Programmierung von Windows Vista	942
11.2.1	Die native NT-Programmierschnittstelle	945
11.2.2	Die Win32-Programmierschnittstelle	949
11.2.3	Die Windows-Registrierungsdatenbank	953
11.3	Systemstruktur	956
11.3.1	Betriebssystemstruktur	957
11.3.2	Starten von Windows Vista	973
11.3.3	Implementierung des Objekt-Managers	974
11.3.4	Subsysteme, DLLs und Dienste im Benutzermodus	986
11.4	Prozesse und Threads in Windows Vista	989
11.4.1	Grundlegende Konzepte	989
11.4.2	API-Aufrufe zur Job-, Prozess-, Thread- und Fiberverwaltung	994
11.4.3	Implementierung von Prozessen und Threads	1000
11.5	Speicherverwaltung	1009
11.5.1	Grundlegende Konzepte	1009
11.5.2	Systemaufrufe zur Speicherverwaltung	1014
11.5.3	Implementierung der Speicherverwaltung	1016
11.6	Caching in Windows Vista	1026
11.7	Ein-/Ausgabe in Windows Vista	1028
11.7.1	Grundlegende Konzepte	1028
11.7.2	API-Aufrufe für die Ein-/Ausgabe	1030
11.7.3	Implementierung der Ein-/Ausgabe	1033
11.8	Das Windows-NT-Dateisystem	1039
11.8.1	Grundlegende Konzepte	1040
11.8.2	Implementierung des NT-Dateisystems	1041

11.9	IT-Sicherheit in Windows Vista	1052
11.9.1	API-Aufrufe zu Sicherheitsfunktionen.	1056
11.9.2	Implementierung von Sicherheitsfunktionen	1057
	Zusammenfassung.	1060
	Übungen	1061
Kapitel 12 Fallstudie 3: Symbian OS		1065
12.1	Die Geschichte von Symbian OS	1067
12.1.1	Die Wurzeln von Symbian OS: Psion und EPOC.	1067
12.1.2	Symbian OS Version 6	1068
12.1.3	Symbian OS Version 7	1069
12.1.4	Symbian OS heute.	1069
12.2	Überblick über Symbian OS.	1070
12.2.1	Objektorientierung.	1070
12.2.2	Mikrokerndesign	1071
12.2.3	Der Nanokern von Symbian OS	1072
12.2.4	Client-Server-Ressourcenzugriff	1073
12.2.5	Merkmale eines größeren Betriebssystems.	1074
12.2.6	Kommunikation und Multimedia	1074
12.3	Prozesse und Threads in Symbian OS.	1075
12.3.1	Threads und Nanothreads.	1075
12.3.2	Prozesse	1077
12.3.3	Aktive Objekte	1077
12.3.4	Interprozesskommunikation	1078
12.4	Speicherverwaltung	1079
12.4.1	Systeme ohne virtuellen Speicher	1079
12.4.2	Wie Symbian OS den Speicher adressiert	1081
12.5	Eingabe und Ausgabe	1083
12.5.1	Gerätetreiber	1083
12.5.2	Kernerweiterungen	1084
12.5.3	Direct Memory Access.	1085
12.5.4	Spezialfall: Speichermedien.	1085
12.5.5	Blockieren der Ein-/Ausgabe	1086
12.5.6	Wechseldatenträger.	1086
12.6	Speichersysteme	1087
12.6.1	Dateisysteme für mobile Geräte	1087
12.6.2	Das Symbian-OS-Dateisystem.	1088
12.6.3	Dateisystemsicherheit und -schutz.	1088
12.7	IT-Sicherheit in Symbian OS	1089
12.8	Kommunikation in Symbian OS	1092
12.8.1	Basisinfrastruktur	1092
12.8.2	Ein genauerer Blick auf die Infrastruktur.	1093
	Zusammenfassung.	1096
	Übungen	1097

Kapitel 13 Entwurf von Betriebssystemen

1099

13.1	Das Problem des Entwurfs.	1101
13.1.1	Ziele	1101
13.1.2	Warum ist es schwierig, ein Betriebssystem zu entwerfen?	1102
13.2	Schnittstellenentwurf	1104
13.2.1	Leitlinien	1105
13.2.2	Paradigmen.	1106
13.2.3	Die Systemaufrufchnittstelle	1110
13.3	Implementierung	1113
13.3.1	Systemstruktur	1113
13.3.2	Mechanismus versus Strategie	1117
13.3.3	Orthogonalität	1118
13.3.4	Namensräume.	1119
13.3.5	Zeitpunkt des Bindens.	1121
13.3.6	Statische versus dynamische Strukturen	1122
13.3.7	Top-down- versus Bottom-up-Implementierung	1123
13.3.8	Nützliche Techniken	1124
13.4	Performanz.	1130
13.4.1	Warum sind Betriebssysteme langsam?	1130
13.4.2	Was sollte verbessert werden?	1131
13.4.3	Der Zielkonflikt zwischen Laufzeit und Speicherplatz	1132
13.4.4	Caching.	1135
13.4.5	Hints	1136
13.4.6	Ausnutzen der Lokalität	1137
13.4.7	Optimieren des Normalfalls	1137
13.5	Projektverwaltung	1138
13.5.1	Der Mythos vom Mann-Monat.	1138
13.5.2	Teamstruktur	1140
13.5.3	Die Bedeutung der Erfahrung	1142
13.5.4	No Silver Bullet	1143
13.6	Trends beim Entwurf von Betriebssystemen	1143
13.6.1	Virtualisierung	1143
13.6.2	Mehrkern-Prozessoren	1144
13.6.3	Betriebssysteme mit großem Adressraum.	1144
13.6.4	Netzwerkfähigkeiten	1145
13.6.5	Parallele und verteilte Systeme.	1146
13.6.6	Multimedia.	1146
13.6.7	Batteriebetriebene Computer.	1146
13.6.8	Eingebettete Systeme	1147
13.6.9	Sensorknoten	1148
	Zusammenfassung	1148
	Übungen.	1149

Kapitel 14	Bibliografie	1153
14.1	Empfehlungen für weiterführende Literatur	1154
14.1.1	Einführung und allgemeine Werke.	1154
14.1.2	Prozesse und Threads	1155
14.1.3	Speicherverwaltung.	1155
14.1.4	Ein- und Ausgabe	1156
14.1.5	Dateisysteme	1156
14.1.6	Deadlocks.	1156
14.1.7	Multimedia-Betriebssysteme.	1157
14.1.8	Multiprozessorsysteme	1157
14.1.9	IT-Sicherheit	1158
14.1.10	Linux	1160
14.1.11	Windows Vista.	1161
14.1.12	Symbian OS	1161
14.1.13	Entwurfsprinzipien	1162
14.2	Alphabetische Literaturliste	1163
Fachwörterverzeichnis		1199
Namensregister		1219
Register		1221