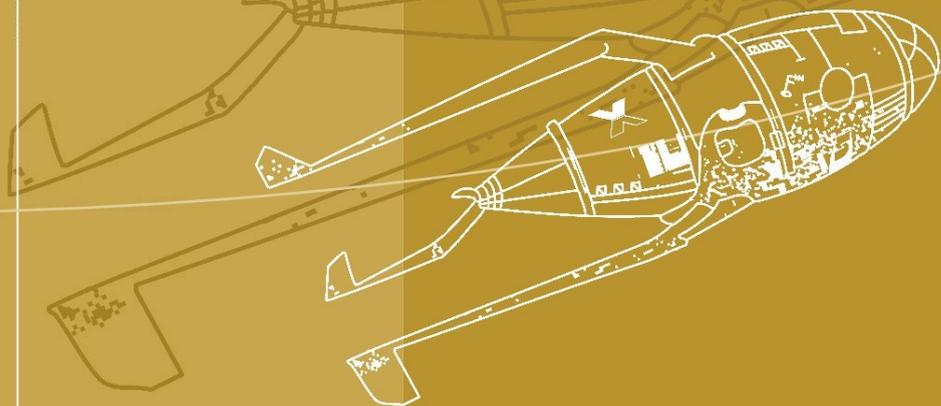


0° 10° 20° 30° 40° 50° 60° 70° 80° 90° 100° 110° 120° 130° 140°



X-systems.press



Wolfgang W. Osterhage

Performance- Optimierung

 Springer Vieweg

X . systems . press

Weitere Bände siehe:
<http://www.springer.com/series/5189>

X.systems.press ist eine praxisorientierte
Reihe zur Entwicklung und Administration von
Betriebssystemen, Netzwerken und Datenbanken.

Wolfgang W. Osterhage

Performance-Optimierung

Systeme, Anwendungen, Geschäftsprozesse

Wolfgang W. Osterhage
Wachtberg-Niederbachem
Deutschland

ISSN 1611-8618
ISBN 978-3-642-17189-5
DOI 10.1007/978-3-642-17190-1

ISBN 978-3-642-17190-1 (eBook)

Die Deutsche Nationalbibliothek verzeichnet diese Publikation in der Deutschen Nationalbibliografie; detaillierte bibliografische Daten sind im Internet über <http://dnb.d-nb.de> abrufbar.

Springer Vieweg

© Springer-Verlag Berlin Heidelberg 2012

Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung, die nicht ausdrücklich vom Urheberrechtsgesetz zugelassen ist, bedarf der vorherigen Zustimmung des Verlags. Das gilt insbesondere für Vervielfältigungen, Bearbeitungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Die Wiedergabe von Gebrauchsnamen, Handelsnamen, Warenbezeichnungen usw. in diesem Werk berechtigt auch ohne besondere Kennzeichnung nicht zu der Annahme, dass solche Namen im Sinne der Warenzeichen- und Markenschutz-Gesetzgebung als frei zu betrachten wären und daher von jedermann benutzt werden dürften.

Einbandentwurf: KuenkelLopka GmbH, Heidelberg

Gedruckt auf säurefreiem und chlorfrei gebleichtem Papier

Springer Vieweg ist eine Marke von Springer DE.

Springer DE ist Teil der Fachverlagsgruppe Springer Science+Business Media
www.springer-vieweg.de

Vorwort

Performance von Computerleistung hat zu Zeiten, als Hardware noch teuer war, eine große Rolle gespielt, weil diese optimal genutzt werden musste. Später war dieses Kriterium nicht mehr so wichtig, da Hardware erschwinglich wurde. Inzwischen hat man erkannt, dass trotz alledem Performance wieder eine Rolle spielt – eben auch wegen der großzügigen Handhabung von Ressourcen bei der Entwicklung komplexer Anwendungen. Dieses Buch differenziert zunächst die drei Ebenen der Performance-Optimierung: Systeme, Anwendungen und die unterstützten Geschäftsprozesse. Auf allen drei Ebenen lassen sich Optimierungen erzielen und Kostenpotenziale erschließen. Es werden die zugehörigen Theorien und Messmethoden sowie Lösungsvorschläge zur Umsetzung von Optimierungsmaßnahmen vorgestellt. Eine Bewertung von Netzwerkmonitoren und eine Checkliste runden das Werk ab.

Mein Dank gilt der Springer-Redaktion, insbesondere Herrn Engesser und Frau Glaunsinger, sowie dem Realisierungsteam für ihre geduldige Unterstützung dieses Vorhabens.

Februar 2012

Dr. Wolfgang W. Osterhage

Inhalt

1	Einführung	1
1.1	Begrifflichkeiten	1
1.2	Drei Ebenen	1
1.3	Die Struktur des Buches	4
2	Performance-Theorie	7
2.1	System-Performance	7
2.1.1	Hardware-Parameter	8
2.1.2	CPU	12
2.1.3	Hauptspeicher	17
2.1.4	Platten	21
2.1.5	I/O	23
2.1.6	Betriebssystemparameter	26
2.2	Anwendungsperformance	27
2.2.1	Anwendungsparameter	27
2.2.2	Datenhaltungskonzepte	29
2.2.3	Anwendungsumgebung	34
3	Performance-Messung	37
3.1	Allgemein	37
3.2	Messung	37
3.2.1	Dynamische Daten	37
3.2.2	Statische Daten	39
3.3	Voraussetzungen	40
3.3.1	Beispiele für Erkenntnisse aus Performance-Messungen	41
3.4	Die Datensammlung selbst	42
3.4.1	Testdaten	42
3.4.2	Andere wichtige Statistiken und Kenngrößen	42
3.4.3	Monitoring	44
3.5	Sonderfall Cloud-Anwendungen	46
3.5.1	Einführung	46
3.5.2	Cloud-Anwendungen	48

3.5.3	Vorgehensweise	49
3.5.4	Antwortzeitverhalten	49
3.5.5	End-User-Antwortzeiten	51
3.5.6	Testkonstrukte	51
3.5.7	Durchführung der Messungen	52
3.5.8	Analyse von Cloud-Messungen	54
3.5.9	Fazit	54
3.5.10	Checkliste	55
3.6	Automatisierung	55
4	Analyse	59
4.1	Allgemein	59
4.2	Datenauswertung	60
4.3	Erkenntnisse	61
4.4	Ergebnisbewertung	65
4.4.1	Durchschnittliche CPU-Auslastung	66
4.4.2	Zeitabhängige CPU-Auslastung	67
4.4.3	Relative CPU-Last	67
4.4.4	Durchschnittliche Plattenzugriffe	67
4.4.5	Zeitabhängige I/O-Waits	67
4.4.6	Memory Management	68
4.4.7	Durchschnittliche Memory Management-Aktivitäten	68
4.4.8	Zeitabhängige Memory Management-Aktivitäten	68
4.4.9	Paging I/Os	68
4.4.10	Zusammenfassende Erkenntnisse	69
5	Optimierung	71
5.1	Systemtuning	72
5.1.1	Verteilung von Dateien	73
5.1.2	Datei-Säuberungen	73
5.1.3	Dateien reorganisieren	73
5.1.4	System-Laufwerk	73
5.1.5	Paging Areas	74
5.1.6	Virtueller Speicher	74
5.1.7	Spool Area	74
5.1.8	Tabellen	74
5.1.9	Puffer	75
5.1.10	Prioritäten	75
5.1.11	File Sharing, Code Sharing	76
5.1.12	Versionen	76
5.1.13	Treiber, Protokolle	76
5.1.14	Registries	77
5.2	Anwendungstuning	77
5.2.1	Anwendungsarchitektur	77
5.2.2	Subsysteme	78

5.2.3	Funktionsaufrufe	78
5.2.4	Programm-Struktur	79
5.2.5	Häufige Aufrufe und Opens	79
5.2.6	Dateigrößen	80
5.2.7	File Handling	80
5.2.8	Sortierungen	80
5.2.9	Kopieren	81
5.2.10	GUI	81
5.2.11	Produktionsplan	81
5.2.12	Schulungen	81
5.3	Investitionen	82
5.3.1	Auslastung von Plattenspeichern	82
5.3.2	Datenhaltung	83
5.3.3	Zusammenfassung	84
6	Network Performance	87
6.1	Einführung	87
6.2	Besonderheiten	87
6.3	Evaluierung	88
6.4	Tools	88
6.5	Netzwerke	89
6.5.1	LAN	90
6.5.2	WLAN	91
7	Prozess-Performance	105
7.1	Ausgangslage	105
7.2	Identifizierung kritischer Unternehmensprozesse	105
7.2.1	Performance-relevante Prozesse	106
7.3	Prozessdokumentation	112
7.3.1	Dokumentationsarten	112
7.3.2	Vollständigkeit und Relevanz	115
7.3.3	Verantwortlichkeiten	116
7.3.4	Versionierung und Freigaben	116
7.4	Prozesstests bei neuen Anwendungen	117
7.4.1	Verfügbarkeit von Partnersystemen	119
7.4.2	Testskripte und Testdaten	119
7.4.3	Aufbau der Tests	121
8	Best Practice Beispiele	125
8.1	Hewlett&Packard: LoadRunner	125
8.1.1	Bedarfsbetriebslast	125
8.1.2	Umfassende Abdeckung technischer Umgebungen	126
8.1.3	Umsetzung	126
8.1.4	Service Level Objective (SLO)	126
8.1.5	Monitoring	127

- 8.1.6 Analyse 127
- 8.1.7 Schnittstelle zu HP Diagnostics 128
- 8.1.8 Skalierbarkeit 128
- 8.2 Siemens: SM2 und COSMOS 128
 - 8.2.1 SM2 128
 - 8.2.2 COSMOS 129
- 8.3 Microsoft: SQL Server 2008 130
 - 8.3.1 Allgemeines 130
 - 8.3.2 Der Profiler 131
 - 8.3.3 Leistungsindikatoren 132
 - 8.3.4 Sonstige Abfragen und Berichte 133
- 8.4 UNIX-Systeme 133
 - 8.4.1 Übersicht 133
 - 8.4.2 Mittlere Systemlast 134
 - 8.4.3 Prozessübersicht 134
 - 8.4.4 Weitere Berichte 135
- 8.5 Windows: TuneUp 136
 - 8.5.1 Übersicht über alle Module 137
 - 8.5.2 Leistung steigern 138
 - 8.5.3 Speicherplatz gewinnen 139
 - 8.5.4 Probleme lösen 140
 - 8.5.5 System warten 142
 - 8.5.6 Weitere Tools 143
 - 8.5.7 Windows anpassen 143
 - 8.5.8 TuneUp Utilities 144
- 9 Darüber hinaus 147**
- 10 Checklisten 151**
- 11 Anhang 157**
 - 11.1 Projektmanagement 157
 - 11.1.1 Technische und organisatorische Hilfsmittel 157
 - 11.1.2 Zielsetzung und Aufgaben 158
 - 11.1.3 PMS-Funktionalitäten 159
 - 11.2 Projekt-Controlling 164
 - 11.2.1 Besonderheiten 164
 - 11.2.2 Objekte 164
 - 11.2.3 Prozessuale Ansätze 165
 - 11.2.4 Berichtswesen 167
 - 11.2.5 Steuerung 168
- Literatur 171**
- Sachverzeichnis 173**

Abkürzungsverzeichnis

ANSI	American National Standard Institute
ARP	Address Resolution Protocol
BPSK	Binary Phase Shift Keying
BSI	Bundesamt für Sicherheit in der Informationstechnik
CCK	Complementary Code Keying
CD	Compact Disk
CODASYL	Conference on Data Systems Languages
CPU	Central Processing Unit
CRC	Cyclic Redundancy Check
CRM	Customer Relationship Management
CSMA	Carrier Sense Multiple Access
CSMA/CA	Carrier Sense Multiple Access with Collision Avoidance
CSMA/CD	Carrier Sense Multiple Access with Collision Detection
dBm	Decibel Milliwatt
DBMS	Data Base Management System
DSSS	Direct Sequence Spread Spectrum
E/A	Ein-/Ausgabe
EPK	Ereignisgesteuerte Prozesskette
ERP	Enterprise Resource Planning
FHSS	Frequency Hopping Spread Spectrum
GHz	Gigahertz
GUI	General User Interface
HAT	Hauptaufwandstreiber
HR/DSSS	High Rate/Direct Sequence Spread Spectrum
Hz	Hertz
ID	Identifier
IEEE	Institute of Electrical and Electronic Engineers
I/O	Input/Output
IP	Internet Protocol
ISAM	Index Sequential Access Method
ISM	Industrial, Scientific, Medical
IT	Information Technology

Kbit/s	Kilobits per second
LAN	Local Area Network
Ldev	Logical Device
MAC	Medium Access Control
MBit/s	Megabits per second
MHz	Megahertz
MIMO	Multiple Input Multiple Output
MPDU	MAC Protocol Data Units
MPL	Multi Programming Level
OFDM	Orthogonal Frequency Division Multiplexing
OSI	Open Systems Interconnection Model
PHY	Physical Layer
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase Shift Keying
RAID	Redundant Arrays of Independent Disks
RDBMS	Relational Data Base Management System
RPF	Relative Performance Factor
RTS/CTS	Request to Send/Clear to Send
SLO	Service Level Objective
SQL	Structured Query Language
TCP/IP	Transmission Control Protocol/Internet Protocol
TP	Transaction Processing
USB	Universal Serial Bus
WAN	Wide Area Network
WLAN	Wireless Local Area Network
XOR	eXclusive OR

Kapitel 1

Einführung

1.1 Begrifflichkeiten

Das Thema Performance-Optimierung gliedert sich in drei Hauptteile:

- System-Performance
- Anwendungs-Performance und
- Prozess-Performance.

Für alle drei Bereiche existieren wiederum (Abb. 1.1):

- Theorie
- Messung
- Analyse und
- Optimierung.

1.2 Drei Ebenen

Wenn von Performance die Rede ist, wird sehr häufig implizit nur die System-Performance gemeint – oder noch mehr vereinfacht: die Leistungsfähigkeit der Hardware, sprich Prozessor und Hauptspeicher. Das ist mit ein Grund dafür, dass das Thema Performance in den zurückliegenden Jahren vernachlässigt worden ist. Hardware wurde irgendwann so billig, dass sich programmtechnische Optimierungen nicht mehr zu lohnen schienen, da Manpower eben im Verhältnis zu teuer geworden war. Man kaufte Hardware und Erweiterungen dazu, und schon liefen die Systeme wieder schneller. Oder man war von vornherein so ausgestattet, dass Performance-Probleme einfach nicht auftreten sollten.

Die End-User-Erfahrungen jedoch sprachen immer schon eine andere Sprache. Nach wie vor spielt negativ empfundenes Antwortzeitverhalten eine nicht nur psychologisch wichtige Rolle, sondern auch bei der Bewältigung des Durchsatzes im Tagesgeschäft. Das Verhältnis von Hardware-Investitionen zu Optimierungen ist

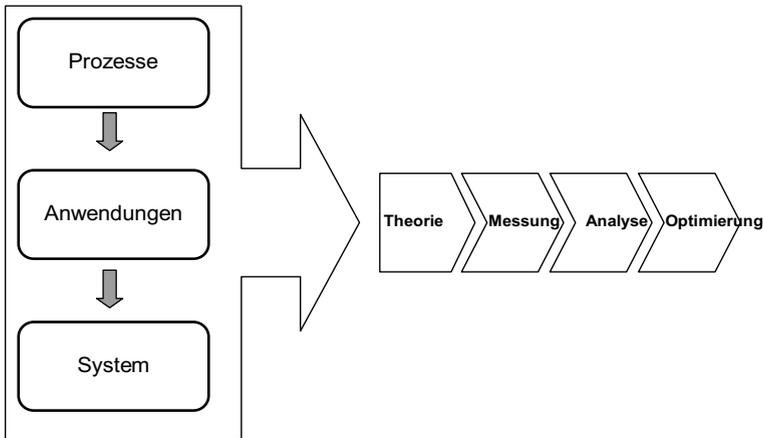


Abb. 1.1 Performance-Dimensionen

quasi immer konstant geblieben. Die Ursache liegt darin, dass großzügige Hardware-Ressourcen ebenso großzügig ausgebeutet werden.

Noch vor vierzig Jahren konnte man sich eine Speicherbelegung mit Leerzeichen oder binären Nullen nicht erlauben. Bereits auf der Ebene der Variablen-DeklARATIONEN und sukzessive bei der Adressierung musste bewusst jedes Byte ausgespart werden. Ansonsten wären Großanwendungen nicht ausführbar gewesen. Spätestens seit Einführung grafischer Oberflächen, mit C++, Java und deren Derivaten, war es mit der strukturierten Programmierung im klassischen Verständnis vorbei. Ansprüche an Bedienkomfort, End-User-Queries usw. haben das ihrige getan, um alte Flaschenhälse in neuem Gewand wiederauferstehen zu lassen. Somit ist die Performance-Debatte wieder aktuell geworden – und eben nicht nur auf Systeme und Hardware allein beschränkt. Obwohl zu Anfang (Abb. 1.1) die drei Ebenen

- System-Performance
- Anwendungs-Performance
- Prozess-Performance

angesprochen wurden, beschränken sich die nun folgenden Begriffsbestimmungen zunächst nur auf System- und Anwendungs-Performance.

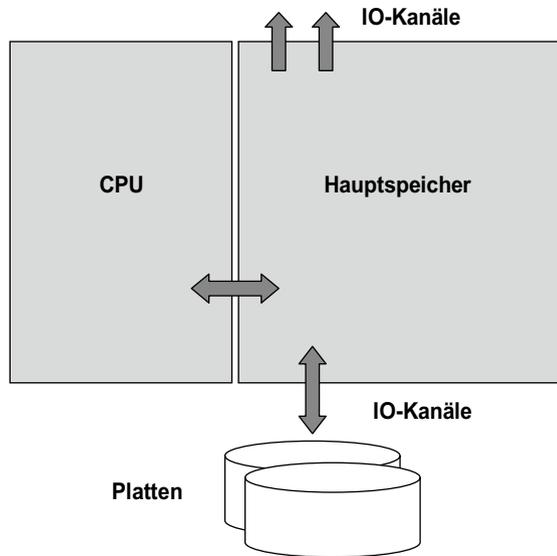
Die System-Performance schließt ein (Abb. 1.2):

- Hardware-Auslastung (Speicher, Prozessor)
- Konfiguration der Systemtabellen
- Ein-/Ausgabe

mit allen für das System-Management relevanten Vorgängen und Parametern.

Bei der Anwendungs-Performance und deren Analyse gibt es natürlich über den Aufruf von System-Ressourcen, die Datenspeicherung und Ein-Ausgabe-Vorgänge

Abb. 1.2 Untergliederung System-Performance



mehr oder weniger starke Verquickungen mit den System-Ressourcen im Detail. Insgesamt aber existiert der große Zusammenhang, dass zum Ausführen von Anwendungen eben Systeme benötigt werden. Abbildung 1.3 zeigt diesen Gesamtzusammenhang auf. Bei Performance-Betrachtungen können beide Ebenen letztendlich nicht als getrennt nebeneinander existierend betrachtet werden.

Die wesentlichen Elemente, die bei der Anwendungsanalyse eine Rolle spielen, sind:

- Programmstruktur (Gesamtanwendung und Module)
- Datenhaltungskonzept
- GUI (General User Interface)

Auf der obersten Ebene schließlich steht die Prozess-Performance. Damit ist nicht der systemische Prozessor gemeint, sondern diejenigen Unternehmensprozesse, die durch die zu untersuchenden Anwendungen auf ihren Systemen unterstützt werden. Insofern werden bei dem Thema die üblichen Managementberatungswerkzeuge wie Balanced Scorecard, Durchlaufzeiten usw. nicht ins Spiel gebracht. Die Bedeutung der Prozesse für die IT-seitige Gesamtpformance liegt darin, bei einer Optimierung gleichzeitig die prozessuale Notwendigkeit bestimmter Abläufe und deren mehr oder weniger ressourcenintensive Systemstützung zu hinterfragen:

- Welche Systemstützung ist wirklich erforderlich?
- Wie wird mein Durchsatz durch Anwendungsperformance beeinflusst?

Auf dieser Basis ist schließlich zu entscheiden, wo beim Tuning am ehesten angesetzt werden muss. An dieser Stelle kommt die Kosten-Nutzen-Frage zu ihrer wahren Bedeutung.

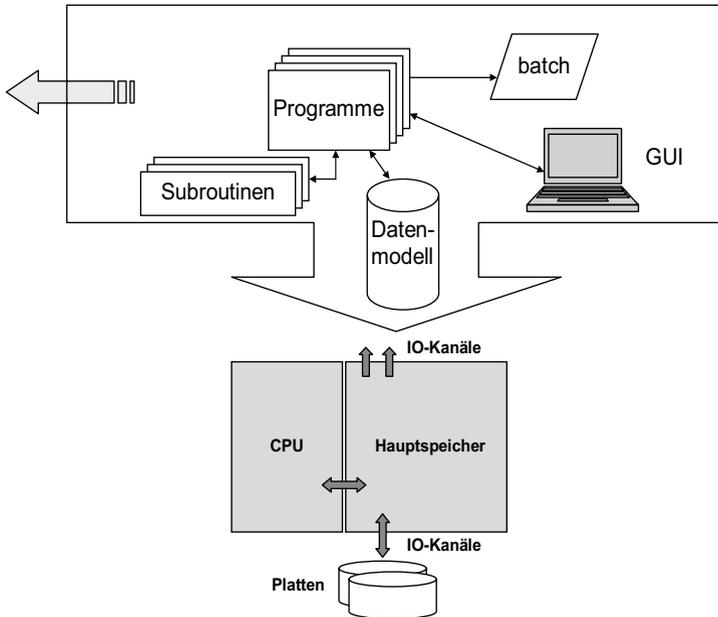


Abb. 1.3 Zusammenspiel zwischen Anwendungen und System

Zusammenfassend lässt sich sagen:

Performance-Probleme werden auf der Prozessebene zuerst erkannt, wenn das Tagesgeschäft leidet. An neuralgischen Punkten werden die kritischen Anwendungen identifiziert. Sie wiederum sind abhängig von den Systemen, auf denen sie laufen.

Performance-Messung startet auf der Systemebene und identifiziert Engpässe, die wiederum auf den Ressourcen-Verbrauch von Anwendungen zurückzuführen sind. Anwendungsanalyse führt letztendlich zur Infragestellung bestimmter Teilprozesse.

Das gesamte Tuningpaket schließlich beinhaltet einen Mix von Maßnahmen auf allen drei Ebenen. Deren Priorisierung zur Umsetzung hängt von dem zu erwartenden wirtschaftlichen Nutzen in einer Organisation gegenüber den zur Umsetzung erforderlichen Kosten ab.

1.3 Die Struktur des Buches

Nach dieser Einführung wird zunächst ein Kapitel den ganzen Komplex der Performance-Theorie behandeln – und zwar getrennt nach System und Anwendungen. Systemseitig werden die Hauptkomponenten Betriebssystem, CPU, Hauptspeicher

und Ein-/Ausgabe besprochen. Besonderes Augenmerk bei der Anwendungstheorie erfahren dabei die Datenhaltungskonzepte.

Im dann anschließenden Kapitel werden die Methoden und Möglichkeiten der Performance-Messung erörtert. Dabei wird unterschieden zwischen dynamischen und statischen Systemdaten und anderen Kenngrößen. Neben den Ausführungen über das Monitoring allgemein erhält die Performance-Problematik in der Cloud besondere Aufmerksamkeit.

Das Analysekapitel enthält eine Vielzahl von grafischen Ergebnissen aus der Aufbereitung von Performance-Rohdaten. Die Schlussfolgerungen aus diesen Erkenntnissen werden ausführlich diskutiert und untereinander in Beziehung gesetzt.

Als Ergebnis werden danach Möglichkeiten und Potenziale des Tunings zur Performance-Verbesserung vorgestellt. Auch hier wird zunächst nach System und Anwendungen differenziert, aber auch die wechselseitigen Abhängigkeiten werden erläutert. Insbesondere der beobachteten Tatsache einer Mischung von verschiedenen Problemen, bei der Lösungswege ohne Kompromisse selten möglich sind, wird Rechnung getragen.

Ein gesondertes Kapitel behandelt die Thematik der Netzwerk-Performance. Es werden zunächst die gängigsten Netzwerk-Performance-Monitore vorgestellt. Anschließend wird die Performance von WLAN-Anwendungen vertieft behandelt.

Eine ganz eigene Herausforderung stellt sich bei der Optimierung von Unternehmensprozessen dar. An dieser Stelle werden nicht die grundsätzlich bekannten Beratungsansätze aus Rationalisierungsprojekten weiterverfolgt, sondern die Möglichkeiten aufgezeigt, die sich im Rahmen der Optimierung von Systemen und IT-Anwendungen ergeben. Dabei geht es einmal um die Konsequenzen für die Prozessvielfalt aus der Anwendungsoptimierung selbst, andererseits um die Identifizierung Performance-sensibler Prozesse.

Schließlich werden in einem gesonderten Kapitel beispielhaft fünf Performance-Tools as best practice vorgestellt: HP LoadRunner, Siemens SM2 und Kosmos, SQL Server, Unix-Befehle und TuneUp.

Das Buch wird abgerundet durch ein Abkürzungsverzeichnis, ein Sachregister und eine umfangreiche Checkliste.

Kapitel 2

Performance-Theorie

Wie in den späteren Kapiteln auch, werden wir die Systemseite und die Anwendungsseite getrennt behandeln. Die Praxis lehrt zwar, dass beide Aspekte eigentlich untrennbar miteinander verwoben sind und Änderungen in der Parametrisierung des einen mit an Sicherheit grenzender Wahrscheinlichkeit Effekte auf der anderen Seite nach sich ziehen werden, aus mindestens zwei Gründen jedoch eine Abschichtung des Gesamtproblems durch eine solche Separatbetrachtung erleichtert wird:

- Auftrennung von Problembereichen nach wechselseitigen Schwerpunkten
- Identifizierung spezifischer Maßnahmen (ohne mögliche Wechselwirkungen aus den Augen zu verlieren).

Eine vermengte Gesamtbetrachtung der Performance-Komplexität erschwert ungenügend das Herausarbeiten von kritischen Einzelaspekten. In diesem Kapitel werden wir also die beiden Linien System und Anwendung nacheinander betrachten. Wegen ihrer unterschiedlichen Natur werden diese Abschnitte anders strukturiert sein.

2.1 System-Performance

Der Abschnitt Systemparameter teilt sich selbst noch einmal auf in

- Hardware-Parameter und
- Betriebssystemparameter.

Auch hier haben wir wiederum Verflechtungen, da beide sich bedingen. Beide Elemente machen aus, was man Systemarchitektur nennt, sollten aber, wenn möglich, getrennt betrachtet werden. Das Spektrum möglicher Betriebssystemparameter ist natürlich abhängig vom Hardwarehersteller. Auf der anderen Seite ist eine fast unendliche Kombinatorik von Betriebseinstellungen denkbar für eine gegebene Hardwarekonfiguration desselben Herstellers – nur begrenzt durch die Realitäten der laufenden Anwendungen, wobei wir wieder bei den oben genannten Randbedingungen wären. Umgekehrt kann dieselbe Version eines Betriebssystems auf unterschiedlichen Hardwareumgebungen zuhause sein. Wir werden zunächst auf die Hardware selbst eingehen.