

X.systems.press

Leistungsbewertung bei Computersystemen

Praktische Performance-Analyse von Rechnern und ihrer Kommunikation

Bearbeitet von
Hans Günther Kruse

1. Auflage 2009. Buch. vii, 201 S. Hardcover
ISBN 978 3 540 71053 0
Format (B x L): 15,5 x 23,5 cm

Weitere Fachgebiete > EDV, Informatik > Datenbanken, Informationssicherheit,
Geschäftssoftware > Informations- und Kodierungstheorie

schnell und portofrei erhältlich bei


DIE FACHBUCHHANDLUNG

Die Online-Fachbuchhandlung bech-shop.de ist spezialisiert auf Fachbücher, insbesondere Recht, Steuern und Wirtschaft. Im Sortiment finden Sie alle Medien (Bücher, Zeitschriften, CDs, eBooks, etc.) aller Verlage. Ergänzt wird das Programm durch Services wie Neuerscheinungsdienst oder Zusammenstellungen von Büchern zu Sonderpreisen. Der Shop führt mehr als 8 Millionen Produkte.

1 Vorbemerkungen zur Last mit der Leistung

Die Leistungsanalyse von Rechnern und Rechnernetzen ist im Prinzip aus der Notwendigkeit entstanden, vorhandene Systeme auf Schwachstellen hin zu untersuchen, um anschließend Verbesserungen konzipieren zu können.

Voraussetzung ist natürlich, dass man eine genaue Vorstellung von **Leistung** hat, und dies ist leider nicht immer gewährleistet. Im Rahmen eines nahe liegenden Ansatzes wäre es natürlich möglich, sich auf die Physik zu besinnen und Leistung als Arbeit pro Zeit zu definieren. Für viele Situationen und Systeme trifft diese einfache Vorgehensweise auch zu und wir werden sie auch später ausgiebig nutzen, jedoch immer die Grenzen aufzeigen.

Will man jedoch die Leistungsfähigkeit eines hochkomplexen Systems wie das eines heutigen Computers oder Rechnernetzes bewerten, so muss man sich genau überlegen, welche Größen bzw. Variablen die Leistung beschreiben.

Besteht die Leistung in einer möglichst schnellen Verarbeitung von Befehlen oder im Befehlsumfang, besteht sie aus der Größe der Speicherkapazitäten oder der Organisationsform wie SingleUser/SingleTask, MultiUser/MultiTask, Dialog- und Batch-Betrieb? Auch hier wird es auf die Gründe der Leistungsanalyse ankommen, welche letztlich die Auswahl der entsprechenden Variablen oder Messgrößen festlegen. So wird es einen großen Unterschied ausmachen, ob die Geschwindigkeit beim Übersetzen von Programmen oder die Verarbeitung einer großen Anzahl von Gleitkommaoperationen im Vordergrund steht. Beide Male sind Maschinenbefehle zu analysieren, wobei aber Organisation und Ablauf völlig verschieden sind. Bei Übersetzern/Compilern kommt es auf die effiziente Nutzung von Befehlen auf Zeichenketten und zum Lesen/Schreiben des Hauptspeichers an. Bei den Gleitkommaoperationen ist es häufig nur eine Addition, die gleichförmig über eine große Anzahl von Operanden abläuft.

Oder betrachten wir eine andere Anwendung, welche sehr stark vom Zugriff auf Datenbanken Gebrauch macht. Die dazugehörigen Transaktionen hängen von den Speicher- und I/O-Zugriffen ab, auch die Cache-Organisation spielt eine Rolle, numerische oder Zeichenkettenoperationen wie bei unseren beiden ersten Beispielen treten völlig in den Hintergrund. Natürlich kann man argumentieren, dass

die **Laufzeit z_{cpu} [sec]** aller drei Anwendungen die entscheidende Leistungsgröße ist – je kürzer, desto effizienter und leistungsfähiger –, die Ursache dafür sind jedoch so verschieden, dass kein Leistungsvergleich auf dieser Ebene einen Sinn macht.

Wir werden uns daher in den folgenden Ansätzen bemühen, solche Widersprüchlichkeiten zu vermeiden, und beginnen mit der Darstellung der grundsätzlichen Situation bei der Leistungsanalyse (s. Abb. 1.1).

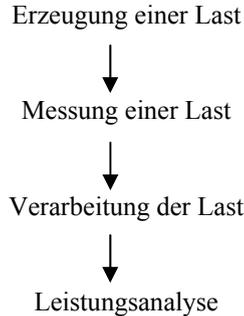


Abb. 1.1 Last/Leistungsmodell

Zentral sind hierbei die Begriffe **Leistung**, dessen Problematik schon skizziert wurde, und **Last**. Im einfachsten Fall wird dies – die **Last** – ein typisches Anwendungsprogramm sein, welches von einem Compiler in Maschineninstruktionen umgewandelt, in der Verarbeitungseinheit abläuft. Genauso denkbar ist aber ein Datenpaket mit definiertem Aufbau und Länge, welches in einem Router an eine bestimmte Adresse (forwarding) geschickt wird. In beiden Fällen besteht die Last aus einem Objekt, an welchem z. B. die **Ablaufzeit** (CPU-Zeit) bestimmt wird – man misst einen individuellen Auftrag.

Der weitaus schwierigere Fall besteht darin, die Last aus mehreren Programmen oder vielen Datenpaketen zu gestalten, die untereinander kein einheitliches Verhalten zeigen. So enthalten sie mal mehr, mal weniger I/O-Anweisungen oder Gleitkommaoperationen – auch können die Datenpakete im Routerbeispiel unterschiedlich lang sein, ähnlich ist es bei den Datenbanktransaktionen. Wenn es Variablen gibt, die eine gewisse statistische Gesetzmäßigkeit aufweisen, kann man diese zur Beschreibung der Gesamtlast heranziehen – allerdings sind dabei häufig gewisse vereinfachende Modellvorstellungen nicht zu vermeiden. Als Leistungskenngröße bietet sich hier der **Durchsatz** an, d. h. die Anzahl der bearbeiteten Aufträge oder Datenpakete pro Zeiteinheit. Interessanterweise kommen wir mit diesem Begriff „Durchsatz“ auch an die physikalische Definition Arbeit pro Zeit nahe heran.

Bei den vorausgegangenen Ausführungen war immer von den DV-Systemen als Ganzes ausgegangen und die entsprechenden Leistungsmaße wie Systemzeit oder Durchsatz definiert worden. Nun kann man die Komponenten wie E/A-Subsystem,

Bus, Caches oder Platten entsprechend behandeln und Leistungsmaße wie Bandbreite (Mbit/sec), Hitraten oder Umdrehungen pro Sekunde (rps) einführen – aber es wird sehr schwierig, Maße für das Zusammenwirken solcher Komponenten zu finden. Dies verkompliziert sich noch, wenn zur Hardware noch Software wie Betriebssysteme oder Netzprotokolle treten.

Aus diesem Grund verzichten wir in den folgenden Analysen und Diskussionen weitgehend auf die Berücksichtigung von Details und bleiben bei der bereits anfangs konzipierten „Black-Box“. Das bedeutet die Lastdefinition und Lastmessung „vor“ der Box und die Leistungsmessung „hinter“ der Box. Dabei beachten wir folgende Grundsätze:

1. Last- und Leistungsmaß müssen miteinander korrespondieren, z. B. wird man ein interaktives System anders als ein Batch-System ausmessen.
2. Sorgfältige Auswahl geeigneter statistischer Techniken, um die Messungen zu validieren und zu vergleichen.
3. Vergleich der gemessenen Werte mit entsprechenden Ergebnissen analytischer oder simulativer Modelle.

Der letzte Punkt mag vor dem Hintergrund der reinen Messung nicht ganz einseitig erscheinen, tatsächlich zeigt aber die praktische Erfahrung, wie wichtig gut strukturierte und validierte Modelle für das Verständnis von Systemleistung sind.

Die Leistungsanalyse und Modellierung haben ihren Ursprung in der Telefonie, erst nach dem Zweiten Weltkrieg und vor allem zu Beginn der 70er-Jahre des letzten Jahrhunderts wurden sie mit Erfolg auf dem Feld der Rechner und Netze angewandt. Da die analytischen Methoden für die immer komplexer werdenden Systeme bald nicht mehr ausreichten, entstanden umfangreiche und leistungsfähige numerisch orientierte Algorithmen und Programme. Vor allem die Entwicklung der Internettechnologie und der Druck, eine effiziente Steuerung der Mainframes/Großrechner, die damals die Hauptstütze der Datenverarbeitung darstellten, zur Verfügung zu haben, hatten einen entscheidenden Anteil daran.

In dieser Zeit entstand auch das klassische Werk von L. Kleinrock [1], welches bis in die heutige Zeit ein Standard hinsichtlich der grundsätzlichen analytischen Methodik geblieben ist. Wir werden uns in den Ausführungen der anschließenden Kapitel, vor allem bei der Modellierung häufig darauf beziehen. In den letzten 20 Jahren sind zahlreiche Lehrbücher zur Thematik „Leistungsanalyse“ erschienen, wir werden häufig Bezug nehmen auf die Werke von Raj Jain [2], von M. Haas und W. Zorn [3], die ausgezeichneten Werke von Tran-Gia [4] und G. Bolch [5]. Eine herausragende Rolle spielt das Lehrbuch von Hennessy & Patterson [6], dem viele moderne Anregungen und instruktive Beispiele entnommen wurden. Gerade dieses Buch zeigt, wie man analytisch-mathematische Methoden zum Verständnis sinnvoll nutzen kann, ohne die Methodik und Mathematik als Selbstzweck (*l'art pour l'art*) darzustellen. Ein ähnliches Ziel, allerdings ohne Berücksichtigung von Hardware-Details, ist hier auch ins Auge gefasst. Meinen Kollegen Dr. Erich

Strohmaier in Berkeley und Dr. Heinz Kredel in Mannheim verdanke ich ebenfalls in dieser Richtung viele wertvolle Diskussionsbeiträge sowie Herrn Prof. Effelsberg/Mannheim die günstigen Rahmenbedingungen.

Ich habe mich, zumindest in den ersten Kapiteln, bemüht, mit einer elementaren Mathematik auszukommen, bis auf einige „Ausrutscher“ bei Statistik-Problemen im Kap. 5 über Benchmarks sollte dies im Großen und Ganzen gelungen sein. In den folgenden Ausführungen des Kap. 6 über messbare Leistungsgesetze kommt die Technik der linearen Gleichungssysteme zum Einsatz, während ab Kap. 7 einfache Differenzialgleichungen 1. Ordnung in der Zeit eine Rolle spielen. Hier wurden einmal bei der Lösung einer speziellen Gleichung etwas ausführlicher als notwendig mathematische Details skizziert, diese können jedoch ohne Probleme übersprungen werden. Die Theorie der Markoff’schen Prozesse, welche der sogenannten stochastischen Modellierung zugrunde liegt, wird nur in Ansätzen aufgezeigt und darüber hinaus auf jede mathematische Strenge verzichtet – der Schwerpunkt liegt eindeutig auf der Nachvollziehbarkeit der Beispiele. Dies gilt auch für die Skizze der Chapman-Kolmogoroff-Gleichungen und ihrer Herleitung, welche mir nicht nur wegen der Vollständigkeit, sondern wegen ihrer praktischen Anwendung unverzichtbar erscheint.

Im Kap. 10 über Leistungsgesetze und Modellierung wird der Versuch unternommen, Messung und Theorie in Einklang zu bringen, die Beschränkung auf den Algorithmus der Mittelwertanalyse (MVA) ist vor dem Hintergrund eines Einstiegs in diese Art von Verfahren zu sehen. Der an mehr und detaillierteren Informationen interessierte Leser sei an dieser Stelle auf das ausgezeichnete Buch von G. Bolch [5] verwiesen.

Das folgende Kap. 11 beschäftigt sich mit einigen Ideen und Konzepten zur Leistungsanalyse von Kommunikationssystemen, wobei zunächst weitgehend auf bereits diskutierte Schemata zurückgegriffen wird. Da die stochastische Modellierung für Protokolle sehr viel an mathematischer Technik verlangt, wurde zunächst ein relativ simpler Ansatz gewählt, der anschließend verfeinert wird, und deshalb etwas aufwendiger ist. Auch hier können die mathematischen Details aber ohne Probleme übergangen werden.

Die beiden letzten Kap. 12 und 13 beschäftigen sich mit dem Spannungsverhältnis von Modell und Wirklichkeit sowie der Gültigkeit von zwei wichtigen Leistungsgesetzen bei modernen Clusterarchitekturen. Da es sich gezeigt hat, dass in vielen Fällen die diskutierten Modellierungstechniken nicht ausreichen, möchte ich hier Hinweise geben, wie man in diesen Fällen zu verfahren hat, aber auch nicht verschweigen, dass ab jetzt doch Einiges mehr an Mathematikkenntnissen notwendig ist.

An der schriftlichen Form dieser Ausarbeitung waren eine Reihe von Studierenden als Hilfskräfte beteiligt. Erwähnen möchte ich Florian Strasser, Julia Doll, Bernhard Kusch und besonders Sabrina Hauser – ihnen allen gilt mein Dank für ihre Geduld und ihre Mühe.