

Anton Hald
Wolf Nevermann

**Datenbank-Engineering
für Wirtschaftsinformatiker**

Rechnerarchitektur

von John L. Hennessy and David A. Patterson

Aufbau und Arbeitsweise von Rechenanlagen

von Wolfgang Coy

Mehr als nur Programmieren ...

Eine Einführung in die Informatik

von Rainer Gmehlich und Heinrich Rust

Interaktive Systeme

Software-Entwicklung und Software-Ergonomie

von Christian Stary

Modernes Software Engineering

von Reiner Dumke

Management von Softwareprojekten

von Fritz Peter Elzer

PASCAL für Wirtschaftswissenschaftler

von Uwe Schnorrenberg et al.

Datenbank-Engineering

für Wirtschaftsinformatiker

von Anton Hald und Wolf Nevermann

Konzepte und Praxis des Compilerbaus

von Volker Penner

Theoretische Informatik

Grundlagen und praktische Anwendungen

von Werner Brecht

Anton Hald
Wolf Nevermann

Datenbank-Engineering für Wirtschaftsinformatiker

Eine praxisorientierte Einführung



Das in diesem Buch enthaltene Programm-Material ist mit keiner Verpflichtung oder Garantie irgendeiner Art verbunden. Die Autoren und der Verlag übernehmen infolgedessen keine Verantwortung und werden keine daraus folgende oder sonstige Haftung übernehmen, die auf irgendeine Art aus der Benutzung dieses Programm-Materials oder Teilen davon entsteht.

Alle Rechte vorbehalten

© Friedr. Vieweg & Sohn Verlagsgesellschaft mbH, Braunschweig/Wiesbaden, 1995

Der Verlag Vieweg ist ein Unternehmen der Bertelsmann Fachinformation GmbH.



Das Werk einschließlich aller seiner Teile ist urheberrechtlich geschützt. Jede Verwertung außerhalb der engen Grenzen des Urheberrechtsgesetzes ist ohne Zustimmung des Verlags unzulässig und strafbar. Das gilt insbesondere für Vervielfältigungen, Übersetzungen, Mikroverfilmungen und die Einspeicherung und Verarbeitung in elektronischen Systemen.

Umschlaggestaltung: Klaus Birk, Wiesbaden

ISBN 978-3-528-05436-6

ISBN 978-3-322-92929-7 (eBook)

DOI 10.1007/978-3-322-92929-7

Vorwort

Datenbanken gehören zu den Kernanwendungen der EDV. Es ist daher nicht verwunderlich, daß kaum ein anderes Gebiet der Informatik so gründlich erforscht ist. Dabei überholen sich die Entwicklungen gegenseitig. Während viele Unternehmen die Umstellung von zentralen Datenhaltungssystemen zu dezentralisierten verteilten Systemen auf der Grundlage relationaler Datenbanken noch längst nicht abgeschlossen haben, drängen bereits objektorientierte Produkte auf den Markt.

Ein Lehrbuch zum Thema Datenbankengineering sollte sowohl einen Überblick zu den aktuellen „Trends“ (wie z.B. verteilte Systeme, Einsatz von CASE-Werkzeugen, objektorientierte Systemanalyse) als auch eine Darstellung der „klassischen“ Datenbanktheorie (Datenorganisation, Datenmodelle etc.) enthalten. Hierbei ist eine Schwerpunktsetzung erforderlich. Dieses Buch vermittelt über die theoretischen Grundlagen von Datenbanksystemen hinaus vor allem eine praxisorientierte Einführung in Anforderungen und Problemstellungen der Entwicklung von Anwendungen für relationalen Systeme.

Im ersten Kapitel wird zunächst eine allgemeiner Überblick zu den Leistungsmerkmalen und der Architektur von Datenbanksystemen gegeben. Das zweite Kapitel stellt die „klassischen“ Datenmodelle dar, auf denen Datenbanken basieren. Das Anfang der siebziger Jahre entwickelte relationale Modell spielt dabei insbesondere in der kommerziellen Datenverarbeitung eine herausragende Rolle. Das dritte Kapitel widmet sich daher vertieft den relationalen Datenbanksystemen. Neben der Darstellung der theoretischen Grundlagen des relationalen Datenmodells (Normalisierung, relationale Algebra) erfolgt eine Einführung in die weit verbreitete Datenbanksprache SQL.

Zunehmend spielen objektorientierte Analyse- und Designmethoden und objektorientierte Programmiersprachen eine wichtige Rolle in der Anwendungsentwicklung. Das vierte Kapitel stellt die wesentlichen Grundkonzepte objektorientierter Systeme dar und beschreibt Anforderungen und Leistungsmerkmale objektorientierter Datenbanken.

Die Effizienz von Datenbanksystemen wird wesentlich von der Art der physischen Organisation der Daten bestimmt. Im fünften Kapitel wird ein Überblick zu den verschiedenen physischen Organisationsformen von

Datenbanken gegeben. Das sechste Kapitel beschreibt die verschiedenen Aspekte und Problemstellungen bei der Gewährleistung der Integrität von Daten.

Zunehmend werden in Unternehmen zentrale Datenhaltungssysteme und Datenbanken durch verteilte Systeme ergänzt bzw. abgelöst. Diese haben den Vorteil, daß sie Daten dort bereitstellen und aufbereiten, wo sie benötigt werden. Das siebente Kapitel vermittelt hierzu einen Überblick.

Die letzten beiden Kapitel führen praxisorientiert in den Entwurf und die Entwicklung von relationalen Datenbanksystemen ein. In Kapitel acht werden die Schritte von der Daten- und Anwendungsmodellierung mit Hilfe von CASE-Tools, die Sicherstellung der Integrität und des Datenschutzes, das physische Design und die Implementierung einer relationalen Datenbank dargestellt. Die Beispiele hierzu beziehen sich auf das weit vorbereitete ORACLE-System. Das abschließende Kapitel neun beschäftigt sich mit der Programmierung der Anwendungsebene. Hierbei werden die Möglichkeiten aufgezeigt, Datenbankanwendungen mit Sprachen der dritten Generation (z.B.: C in Verbindung mit „embedded SQL“) und Werkzeugen der vierten Generation (z.B.: SQLForms, MS-ACCESS und SQLWindows) zu erstellen.

Das Buch hat seine Ursprünge in Vorlesungen und Kursen an der Fachrichtung Wirtschaftsinformatik der Berufsakademie Mannheim. Es ist sowohl für Studierende dieses Studiengangs als auch für Studentinnen und Studenten von Fachhochschulen und Universitäten oder für Berufspraktiker geeignet, die sich einen Überblick zu Theorie und Praxis des Datenbankengineering verschaffen wollen.

Die Autoren möchten sich insbesondere bei Peter Hohendorf und Kai Persch für die kritischen Anmerkungen, die zahlreichen Anregungen und die Verbesserungsvorschläge, die sie bei Durchsicht des Manuskriptes eingebracht haben, bedanken. Weiterhin danken wir Berit Carstens, Torsten Harder, Christian Krüger, Joachim Warntjen und Ralf Winterer für ihre Unterstützung bei der Korrektur des Manuskriptes. Nicht zuletzt danken wir dem Vieweg-Verlag für seine Hilfen bei der Gestaltung des Buches.

Mannheim, im Januar 1995

Anton Hald
Wolf Nevermann

Konventionen

In diesem Buch werden folgende Zeichen verwendet:

- beendet eine Definition
- beendet einen Hinweis
- beendet ein Beispiels
- ◆ beendet eine Aufgabe

Bei der allgemeinen Darstellung der Syntax von Befehlen werden die Schlüsselworte generell groß geschrieben; die variablen Begriffe erscheinen dagegen *kursiv* in Kleinschreibung.

Beispiel:

```
SELECT spaltenname FROM tabellenname;
```

Stehen Teile eines Befehls in eckigen Klammern ([...]), so ist die Eingabe nicht zwingend (optional).

Beispiel:

```
CREATE DATABASE db-name  
[ARCHIVELOG];
```

In den Beispielen zu den Befehlen werden Attribute generell klein geschrieben. Die Tabellennamen werden dagegen wie normale Substantive behandelt.

Beispiel:

```
SELECT personalnr.name FROM Mitarbeiter;
```

Inhaltsverzeichnis

1 EINFÜHRUNG	1
1.1 WOZU DATENBANKSYSTEME?.....	1
1.2 AUFBAU UND AUFGABEN VON DATENBANKSYSTEMEN	2
1.3 ARCHITEKTUR VON DATENBANKSYSTEMEN.....	5
1.3.1 Drei-Schichten-Architektur.....	5
1.3.2 Verteilte Architekturen	7
2 DATENMODELLE.....	10
2.1 ENTITY-RELATIONSHIP MODELL (ER-MODELL).....	10
2.1.1 Die Menge, das Entity-Set.....	10
2.1.2 Abbildungen.....	11
2.1.3 Beziehungen zwischen Entity-Sets	14
2.1.4 Generalisierung/Spezialisierung	19
2.1.5 Beispiel: Projektentwicklungssystem.....	21
2.2 RELATIONALES DATENMODELL	25
2.2.1 Attribute und ihre Domänen, Relationen.....	25
2.3 HIERARCHISCHES DATENMODELL.....	28
2.4 NETZWERK-MODELL.....	31
3 RELATIONALE DATENBANKSYSTEME.....	35
3.1 NORMALISIERUNG.....	35
3.1.1 1. Normalform	37
3.1.2 2. Normalform	38
3.1.3 3. Normalform	39
3.1.4 Weitere Normalformen.....	41
3.1.5 Die Normalisierung-eine Grundlage für das logische Datenmodell.....	42
3.2 RELATIONALE ALGEBRA	46
3.3 DIE RELATIONALE SPRACHE SQL	50
3.3.1 Abfrage von Daten.....	52
3.3.1.1 Auswahl von Tabellenspalten.....	53
3.3.1.2 Auswahl von Tabellen	55
3.3.1.3 Auswahl von Zeilen	56
3.3.1.4 Unterabfragen.....	58
3.3.1.5 Gruppierung.....	60
3.3.1.6 Sortierung	62
3.3.1.7 Tabellen verbinden mit Joins.....	63
3.3.2 Datenmanipulation.....	65

3.3.2.1 Einfügen von Zeilen mit INSERT	65
3.3.2.2 Ändern von Spaltenwerten mit UPDATE	66
3.3.2.3 Löschen von Zeilen	67
3.4 ANFORDERUNGEN AN RELATIONALE DATENBANKSYSTEME	68
3.5 DATA-DICTIONARY	71
4 OBJEKTORIENTIERTE DATENBANKEN.....	74
4.1 GRUNDLEGENDE BEGRIFFE UND KONZEPTE	74
4.1.1 Die Entwicklung zur objektorientierten Programmierung	75
4.1.2 Objektorientierte Abstraktion.....	76
4.2 OBJEKTORIENTIERTE SYSTEMMODELLIERUNG.....	83
4.3 VOM OBJEKTMODELL ZU OBJEKTORIENTIERTEN DATENBANKEN.....	87
4.4 RELATIONALE UND OBJEKTORIENTIERTE DATENBANKEN IM VERGLEICH	91
5 PHYSISCHE DATENBANKORGANISATION.....	94
5.1 SPEICHERMEDIEN	95
5.2 PRIMÄRORGANISATION	97
5.2.1 Sequentielle Organisationsform	97
5.2.2 Index-Sequentielle Organisationsform	98
5.2.2.1 ISAM Organisation.....	99
5.2.2.2 VSAM Organisation	101
5.2.3 Reine Index-Organisationsform	104
5.2.4 Direkte Organisationsform.....	106
5.3 SEKUNDÄRORGANISATION.....	108
5.3.1 Indizes	108
5.3.2 Invertierte Liste.....	113
5.3.3 Listenorganisation.....	116
6 DATENINTEGRITÄT	119
6.1 SEMANTISCHE INTEGRITÄT	120
6.2 OPERATIONALE INTEGRITÄT.....	122
6.2.1 Probleme bei parallelen Zugriffen auf eine Datenbank	123
6.2.2 Spermechanismen	130
6.3 RECOVERY	138
6.3.1 Methoden und Werkzeuge zur Wiederherstellung der Datenkonsistenz.....	139
7 VERTEILTE ARCHITEKTUREN.....	142
7.1 VERTEILTE DV-SYSTEME	143
7.2 OFFENE SYSTEME	144
7.2.1 Bedeutung offener Systeme und Standards	145
7.2.2 Open Database Connectivity	146
7.3 CLIENT-SERVER-ARCHITEKTUR.....	149
7.4 VERTEILTE DATENBANKEN	153
7.4.1 Die Fundamentalprinzipien	153
7.4.2 Transaktionsmanagement in VDBMS	155
7.4.3 Replikationen	156
7.4.4 Optimierung der Transaktionsverarbeitung	158

7.5	AUSBLICK	159
8 DESIGN UND IMPLEMENTIERUNG VON DATENBANKSYSTEMEN 161		
8.1	VON DER ANALYSE ZUM DESIGN	163
8.1.1	Analyse des konzeptionellen Datenmodells	164
8.1.2	Analyse der Hard- und Softwarearchitektur	165
8.1.3	Analyse der funktionalen Anforderungen	165
8.1.4	Übergang zur Designphase.....	166
8.2	DESIGN UND IMPLEMENTIERUNG DER TABELLEN.....	167
8.2.1	Definition der Tabellen.....	167
8.2.2	Definition der Tabellenbeziehungen.....	168
8.2.3	Implementierung der Tabellen in SQL.....	173
8.3	DESIGN UND IMPLEMENTIERUNG DER INTEGRITÄTSREGELN.....	177
8.3.1	Definition von deklarativen Integritätsregeln.....	177
8.3.2	Änderung von deklarativen Integritätsregeln.....	181
8.3.3	Probleme mit deklarativen Integritätsregeln	182
8.3.4	Datenbanktrigger.....	184
8.3.5	Definition von Integritätsregeln mit CASE-Tools	186
8.4	DESIGN UND IMPLEMENTIERUNG DER INTERNEN EBENE	187
8.4.1	Anlegen der Datenbank-Dateien	188
8.4.2	Tablespaces	189
8.4.3	Cluster.....	195
8.4.4	Indizes	196
8.4.5	Unterstützung durch CASE-Tools.....	198
8.5	ADMINISTRATION DES DATENBANKSYSTEMS	199
8.5.1	Datenbankrechte	200
8.5.2	Tabellenrechte.....	201
8.6	DESIGN UND IMPLEMENTIERUNG DER EXTERNEN EBENE	203
9 PROGRAMMIERUNG VON DATENBANKANWENDUNGEN..... 205		
9.1	PROGRAMMIERUNG MIT „EMBEDDED SQL“	205
9.1.1	Host-Variable	206
9.1.2	Indikator-Variable.....	207
9.1.3	Anmeldung und Abmeldung bei der Datenbank	207
9.1.4	Cursor-Operationen.....	209
9.1.5	Dynamische SQL-Anweisungen.....	210
9.2	PROGRAMMIERUNG MIT 4GL-SPRACHEN	213
9.2.1	Anwendungsentwicklung mit SQLForms	213
9.2.2	Anwendungsentwicklung mit MS-ACCESS.....	217
9.2.2.1	Formulare	217
9.2.2.2	Abfragen	221
9.2.2.3	Dialogsteuerung	222
9.2.2.4	Programmierung mit ACCESS-BASIC.....	223
9.2.3	Anwendungsentwicklung mit SQLWindows	227
9.2.3.1	QuickForm und QuickObjects.....	228
9.2.3.2	Objektorientierte Programmierung mit SAL	230
9.2.3.3	Team-Programmierung.....	231

LITERATURVERZEICHNIS.....	233
SACHWORTVERZEICHNIS	236