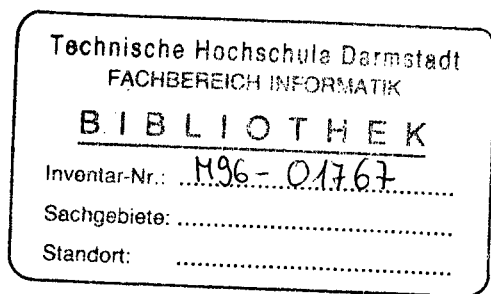

Datenbank- systeme

Eine Einführung

von
Prof. Alfons Kemper, Ph. D.
Dipl.-Inform. André Eickler
Universität Passau



R. Oldenbourg Verlag München Wien 1996

Inhaltsverzeichnis

Vorwort	13
1 Einleitung und Übersicht	15
1.1 Motivation für den Einsatz eines DBMS	15
1.2 Datenabstraktion	17
1.3 Datenunabhängigkeit	18
1.4 Datenmodelle	19
1.5 Datenbankschema und Datenbankausprägung	20
1.6 Einordnung der Datenmodelle nach Einsatzfeld	21
1.6.1 Modelle des konzeptuellen Entwurfs	21
1.6.2 Logische (Implementations-)Datenmodelle	23
1.7 Architekturübersicht eines DBMS	25
1.8 Übungen	27
1.9 Literatur	27
2 Datenbankentwurf	29
2.1 Abstraktionsebenen des Datenbankentwurfs	29
2.2 Allgemeine Entwurfsmethodik	30
2.3 Die Datenbankentwurfsschritte	31
2.4 Die Anforderungsanalyse	31
2.4.1 Informationsstrukturanforderungen	33
2.4.2 Datenverarbeitungsanforderungen	35
2.5 Grundlagen des Entity-Relationship-Modells	35
2.6 Schlüssel	37
2.7 Charakterisierung von Beziehungstypen	37
2.7.1 Funktionalitäten der Beziehungen	37
2.7.2 Erweiterung der Funktionalitätsangaben auf n -stellige Beziehungen	39
2.7.3 Die (min , max)-Notation	40
2.8 Existenzabhängige Entitytypen	44
2.9 Generalisierung	44
2.10 Aggregation	47
2.11 Kombination von Generalisierung und Aggregation	47
2.12 Konsolidierung	48
2.13 Objektorientierte Datenmodellierung	50
2.14 Übungen	50
2.15 Literatur	51

3	Das relationale Modell	53
3.1	Definition des relationalen Modells	53
3.1.1	Mathematischer Formalismus	53
3.1.2	Schema-Definition	54
3.2	Umsetzung eines konzeptuellen Schemas in ein relationales Schema	55
3.2.1	Relationale Darstellung von Entitytypen	55
3.2.2	Relationale Darstellung von Beziehungen	55
3.3	Verfeinerung des relationalen Schemas	59
3.3.1	1:N-Beziehungen	60
3.3.2	1:1-Beziehungen	61
3.3.3	Relationale Modellierung der Generalisierung	62
3.3.4	Beispielausprägung der Universitäts-Datenbank	63
3.4	Die relationale Algebra	64
3.4.1	Selektion	64
3.4.2	Projektion	66
3.4.3	Vereinigung	67
3.4.4	Mengendifferenz	67
3.4.5	Kartesisches Produkt (Kreuzprodukt)	67
3.4.6	Umbenennung von Relationen und Attributen	68
3.4.7	Definition der relationalen Algebra	69
3.4.8	Der relationale Verbund (Join)	70
3.4.9	Mengendurchschnitt	73
3.4.10	Die relationale Division	75
3.4.11	Operatorbaum-Darstellung	76
3.5	Der Relationenkalkül	76
3.5.1	Beispielanfrage im relationalen Tupelkalkül	77
3.5.2	Quantifizierung von Tupelvariablen	78
3.5.3	Formale Definition des Tupelkalküls	79
3.5.4	Sichere Ausdrücke des Tupelkalküls	80
3.5.5	Der relationale Domänenkalkül	80
3.5.6	Beispielanfragen im Domänenkalkül	81
3.5.7	Sichere Ausdrücke des Domänenkalküls	82
3.6	Ausdruckskraft der Anfragesprachen	83
3.7	Übungen	83
3.8	Literatur	86
4	Relationale Anfragesprachen	87
4.1	Geschichte	87
4.2	Datentypen	88
4.3	Schemadefinition	88
4.4	Schemaveränderung	89
4.5	Elementare Datenmanipulation: Einfügen von Tupeln	89
4.6	Einfache SQL-Anfragen	90
4.7	Anfragen über mehrere Relationen	91
4.8	Mengenoperationen und geschachtelte Anfragen	94
4.9	Der Existenzquantor <i>exists</i>	95
4.10	Aggregatfunktionen und Gruppierung	95

4.11	Spezielle Bedingungen	96
4.12	Joins in SQL-92	98
4.13	Rekursion	99
4.14	Veränderungen am Datenbestand	104
4.15	Sichten	105
4.16	Sichten zur Modellierung von Generalisierungen	106
4.17	Charakterisierung update-fähiger Sichten	108
4.18	Einbettung von SQL in Wirtssprachen	109
4.19	Anfragen in Anwendungsprogrammen	111
4.20	Query by Example	112
4.21	Übungen	115
4.22	Literatur	117
5	Datenintegrität	119
5.1	Referentielle Integrität	120
5.2	Gewährleistung referentieller Integrität	120
5.3	Referentielle Integrität in SQL	121
5.4	Überprüfung einfacher statischer Integritätsbedingungen	122
5.5	Das Universitätsschema mit Integritätsbedingungen	124
5.6	Behandlung von Nullwerten	126
5.7	Überprüfung komplexer Integritätsbedingungen	126
5.8	Übungen	127
5.9	Literatur	127
6	Relationale Entwurfstheorie	129
6.1	Funktionale Abhängigkeiten	129
6.1.1	Konventionen zur Notation	130
6.1.2	Einhaltung einer funktionalen Abhängigkeit	130
6.2	Schlüssel	131
6.3	Bestimmung funktionaler Abhängigkeiten	132
6.3.1	Kanonische Überdeckung	135
6.4	„Schlechte“ Relationenschemata	136
6.4.1	Die Updateanomalien	136
6.4.2	Einfügeanomalien	137
6.4.3	Löschanomalien	137
6.5	Zerlegung (Dekomposition) von Relationen	137
6.5.1	Verlustlosigkeit	138
6.5.2	Kriterien für die Verlustlosigkeit einer Zerlegung	140
6.5.3	Abhängigkeitsbewahrung	141
6.6	Erste Normalform	143
6.7	Zweite Normalform	144
6.8	Dritte Normalform	146
6.9	Boyce-Codd Normalform	149
6.10	Mehrwertige Abhängigkeiten	151
6.11	Vierte Normalform	153
6.12	Zusammenfassung	155
6.13	Übungen	155

6.14	Literatur	160
7	Physische Datenorganisation	161
7.1	Speichermedien	161
7.2	Der Datenbankpuffer	162
7.3	Abbildung von Relationen auf den Sekundärspeicher	164
7.4	Indexstrukturen	165
7.5	ISAM	167
7.6	B-Bäume	169
7.7	B ⁺ -Bäume	173
7.8	Präfix-B ⁺ -Bäume	174
7.9	Hashing	175
7.10	Erweiterbares Hashing	177
7.11	Ballung logisch verwandter Datensätze	179
7.12	Unterstützung eines Anwendungsverhaltens	182
7.13	Physische Datenorganisation in SQL	183
7.14	Übungen	183
7.15	Literatur	184
8	Anfragebearbeitung	187
8.1	Logische Optimierung	188
8.1.1	Äquivalenzen in der relationalen Algebra	189
8.1.2	Anwendung der Transformationsregeln	192
8.2	Physische Optimierung	196
8.2.1	Implementierung der Selektion	198
8.2.2	Implementierung von binären Zuordnungsoperatoren	199
8.2.3	Gruppierung und Duplikateliminierung	205
8.2.4	Projektion und Vereinigung	206
8.2.5	Zwischenspeicherung	206
8.2.6	Übersetzung der logischen Algebra	208
8.3	Kostenmodelle	211
8.3.1	Selektivitäten	212
8.3.2	Kostenabschätzung für die Selektion	214
8.3.3	Kostenabschätzung für den Join	215
8.3.4	Kostenabschätzung für die Sortierung	215
8.4	Übungen	216
8.5	Literatur	218
9	Transaktionsverwaltung	221
9.1	Begriffsbildung	221
9.2	Anforderungen an die Transaktionsverwaltung	222
9.3	Operationen auf Transaktions-Ebene	222
9.4	Abschluß einer Transaktion	223
9.5	Eigenschaften von Transaktionen	225
9.6	Transaktionsverwaltung in SQL	226
9.7	Zustandsübergänge einer Transaktion	227
9.8	Literatur	228

10 Fehlerbehandlung	229
10.1 Fehlerklassifikation	229
10.1.1 Lokaler Fehler einer Transaktion	229
10.1.2 Fehler mit Hauptspeicherverlust	230
10.1.3 Fehler mit Hintergrundspeicherverlust	231
10.2 Die Speicherhierarchie	231
10.2.1 Ersetzung von Puffer-Seiten	231
10.2.2 Einbringen von Änderungen einer abgeschlossenen Transaktion	232
10.2.3 Einbringstrategie	233
10.2.4 Hier zugrunde gelegte Systemkonfiguration	234
10.3 Protokollierung von Änderungsoperationen	234
10.3.1 Struktur der Log-Einträge	235
10.3.2 Beispiel einer Log-Datei	235
10.3.3 Logische oder physische Protokollierung	236
10.3.4 Schreiben der Log-Information	237
10.3.5 Das WAL-Prinzip	238
10.4 Wiederanlauf nach einem Fehler	239
10.4.1 Analyse des Logs	239
10.4.2 Redo-Phase	240
10.4.3 Undo-Phase	240
10.5 Fehlertoleranz (Idempotenz) des Wiederanlaufs	241
10.6 Lokales Zurücksetzen einer Transaktion	243
10.7 Partielles Zurücksetzen einer Transaktion	244
10.8 Sicherungspunkte	244
10.8.1 Transaktionskonsistente Sicherungspunkte	245
10.8.2 Aktionskonsistente Sicherungspunkte	246
10.8.3 Unschärfe (fuzzy) Sicherungspunkte	247
10.9 Recovery nach einem Verlust der materialisierten Datenbasis	248
10.10 Übungen	249
10.11 Literatur	251
11 Mehrbenutzersynchronisation	253
11.1 Fehler bei unkontrolliertem Mehrbenutzerbetrieb	254
11.1.1 Verlorengegangene Änderungen (<i>lost update</i>)	254
11.1.2 Abhängigkeit von nicht freigegebenen Änderungen	254
11.1.3 Phantomproblem	255
11.2 Serialisierbarkeit	255
11.2.1 Beispiele serialisierbarer Ausführungen (Historien)	256
11.2.2 Nicht serialisierbare Historie	257
11.3 Theorie der Serialisierbarkeit	259
11.3.1 Definition einer Transaktion	259
11.3.2 Historie (Schedule)	260
11.3.3 Äquivalenz zweier Historien	261
11.3.4 Serialisierbare Historien	262
11.3.5 Kriterien für Serialisierbarkeit	262
11.4 Eigenschaften von Historien bezüglich der Recovery	264
11.4.1 Rücksetzbare Historien	264

11.4.2	Historien ohne kaskadierendes Rücksetzen	264
11.4.3	Strikte Historien	265
11.4.4	Beziehungen zwischen den Klassen von Historien	265
11.5	Der Datenbank-Scheduler	266
11.6	Sperrbasierte Synchronisation	267
11.6.1	Zwei Sperrmodi	268
11.6.2	Zwei-Phasen-Sperrprotokoll	268
11.6.3	Kaskadierendes Rücksetzen (Schneeballeffekt)	269
11.7	Verklemmungen (Deadlocks)	270
11.7.1	Erkennung von Verklemmungen	270
11.7.2	Preclaiming zur Vermeidung von Verklemmungen	272
11.7.3	Verklemmungsvermeidung durch Zeitstempel	273
11.8	Hierarchische Sperrgranulate	274
11.9	Zeitstempel-basierende Synchronisation	278
11.10	Optimistische Synchronisation	280
11.11	Synchronisation von Indexstrukturen	281
11.12	Mehrbenutzersynchronisation in SQL-92	284
11.13	Übungen	286
11.14	Literatur	289
12	Sicherheitsaspekte	291
12.1	Discretionary Access Control	293
12.2	Zugriffskontrolle in SQL	293
12.2.1	Identifikation und Authentisierung	294
12.2.2	Autorisierung und Zugriffskontrolle	294
12.2.3	Sichten	295
12.2.4	Auditing	296
12.3	Verfeinerung des Autorisierungsmodells	296
12.3.1	Implizite Autorisierung von Subjekten	297
12.3.2	Implizite Autorisierung von Operationen	298
12.3.3	Implizite Autorisierung von Objekten	299
12.3.4	Implizite Autorisierung entlang einer Typhierarchie	299
12.4	Mandatory Access Control	301
12.5	Multilevel-Datenbanken	301
12.6	Kryptographie	305
12.6.1	Der Data Encryption Standard	305
12.6.2	Public-Key Kryptographie	306
12.7	Zusammenfassung	308
12.8	Übungen	309
12.9	Literatur	310
13	Objektorientierte Datenbanken	311
13.1	Bestandsaufnahme relationaler Datenbanksysteme	311
13.2	Vorteile der objektorientierten Datenmodellierung	315
13.3	Der ODMG-Standard	316
13.4	Eigenschaften von Objekten	317
13.4.1	Objektidentität	318

13.4.2	Typ eines Objekts	319
13.4.3	Wert eines Objekts	320
13.5	Definition von Objekttypen	320
13.5.1	Attribute	320
13.5.2	Beziehungen	321
13.5.3	Typeigenschaften: Extensionen und Schlüssel	327
13.6	Modellierung des Verhaltens: Operationen	328
13.7	Vererbung und Subtypisierung	330
13.7.1	Terminologie	331
13.7.2	Einfache und Mehrfachvererbung	332
13.8	Beispiel einer Typhierarchie	334
13.9	Verfeinerung (Spezialisierung) und spätes Binden von Operationen	336
13.10	Mehrfachvererbung	338
13.11	Die Anfragesprache OQL	340
13.11.1	Einfache Anfragen	340
13.11.2	Geschachtelte Anfragen und Partitionierung	340
13.11.3	Pfadausdrücke	341
13.11.4	Erzeugung von Objekten	342
13.11.5	Operationsaufruf	342
13.12	C++-Einbettung	343
13.12.1	Objektidentität	343
13.12.2	Objekterzeugung und Ballung	345
13.13	Transaktionen	345
13.13.1	Einbettung von Anfragen	346
13.14	Objektorientierte Modellierung	347
13.14.1	Modellierung der Struktur von Objekten	347
13.14.2	Modellierung des Verhaltens von Objekten	349
13.15	Übungen	351
13.16	Literatur	352
14	Deduktive Datenbanken	355
14.1	Terminologie	355
14.2	Datalog	355
14.3	Eigenschaften von Datalog-Programmen	359
14.3.1	Rekursivität	359
14.3.2	Sicherheit von Datalog-Regeln	359
14.4	Auswertung von nicht-rekursiven Datalog-Programmen	360
14.4.1	Auswertung eines Beispielsprogramms	360
14.4.2	Auswertungs-Algorithmus	363
14.5	Auswertung rekursiver Regeln	366
14.6	Inkrementelle (semi-naive) Auswertung rekursiver Regeln	367
14.7	Bottom-Up oder Top-Down Auswertung	371
14.8	Negation im Regelrumpf	374
14.8.1	Stratifizierte Datalog-Programme	374
14.8.2	Auswertung von Regeln mit Negation	374
14.8.3	Ein etwas komplexeres Beispiel	375
14.9	Ausdruckskraft von Datalog	376

14.10	Übungen	378
14.11	Literatur	381
15	Verteilte Datenbanken	383
15.1	Terminologie und Abgrenzung	383
15.2	Entwurf verteilter Datenbanken	385
15.3	Horizontale und vertikale Fragmentierung	387
15.3.1	Horizontale Fragmentierung	388
15.3.2	Abgeleitete horizontale Fragmentierung	390
15.3.3	Vertikale Fragmentierung	391
15.3.4	Kombinierte Fragmentierung	393
15.3.5	Allokation für unser Beispiel	394
15.4	Transparenz in verteilten Datenbanken	395
15.4.1	Fragmentierungstransparenz	395
15.4.2	Allokationstransparenz	396
15.4.3	Lokale Schema-Transparenz	396
15.5	Anfrageübersetzung und -optimierung in VDBMS	397
15.5.1	Anfragebearbeitung bei horizontaler Fragmentierung	397
15.5.2	Anfragebearbeitung bei vertikaler Fragmentierung	400
15.6	Join-Auswertung in VDBMS	401
15.6.1	Join-Auswertung ohne Filterung	402
15.6.2	Join-Auswertung mit Filterung	402
15.7	Transaktionskontrolle in VDBMS	405
15.8	Mehrbenutzersynchronisation in VDBMS	410
15.8.1	Serialisierbarkeit	410
15.8.2	Das Zwei-Phasen-Sperrprotokoll in VDBMS	411
15.9	Deadlocks in VDBMS	412
15.9.1	Erkennung von Deadlocks	412
15.9.2	Deadlock-Vermeidung	415
15.10	Synchronisation bei replizierten Daten	416
15.11	Übungen	418
15.12	Literatur	422
	Literaturverzeichnis	423
	Index	439