

Datenbanksysteme  
Eine Einführung  
10., aktualisierte und erweiterte Auflage<sup>1</sup>

Professor Alfons Kemper, Ph.D.  
Dr. André Eickler

6. Oktober 2015

<sup>1</sup>©2015 R. Oldenbourg Verlag GmbH. Dieser Auszug ist eine Vorabversion des gleichnamigen Buches. Er ist urheberrechtlich geschützt. Kein Teil darf in irgendeiner Form reproduziert oder in eine maschinenlesbare Sprache übertragen werden.

Intentionally left blank

# Impressum

Intentionally left blank

Intentionally left blank

# Inhaltsverzeichnis

<b>Vorwort</b>	<b>17</b>
<b>1 Einleitung und Übersicht</b>	<b>21</b>
1.1 Motivation für den Einsatz eines DBMS . . . . .	21
1.2 Datenabstraktion . . . . .	23
1.3 Datenunabhängigkeit . . . . .	24
1.4 Datenmodelle . . . . .	25
1.5 Datenbankschema und Ausprägung . . . . .	26
1.6 Einordnung der Datenmodelle . . . . .	26
1.6.1 Modelle des konzeptuellen Entwurfs . . . . .	26
1.6.2 Logische (Implementations-)Datenmodelle . . . . .	27
1.7 Architekturübersicht eines DBMS . . . . .	30
1.8 Übungen . . . . .	32
1.9 Literatur . . . . .	32
<b>2 Datenbankentwurf</b>	<b>33</b>
2.1 Abstraktionsebenen des Datenbankentwurfs . . . . .	33
2.2 Allgemeine Entwurfsmethodik . . . . .	34
2.3 Die Datenbankentwurfsschritte . . . . .	35
2.4 Die Anforderungsanalyse . . . . .	35
2.4.1 Informationsstrukturanforderungen . . . . .	37
2.4.2 Datenverarbeitungsanforderungen . . . . .	39
2.5 Grundlagen des Entity-Relationship-Modells . . . . .	39
2.6 Schlüssel . . . . .	41
2.7 Charakterisierung von Beziehungstypen . . . . .	41
2.7.1 Funktionalitäten der Beziehungen . . . . .	41
2.7.2 Funktionalitätsangaben bei $n$ -stelligen Beziehungen . . . . .	43
2.7.3 Die $(min, max)$ -Notation . . . . .	46
2.8 Existenzabhängige Entitytypen . . . . .	50
2.9 Generalisierung . . . . .	51
2.10 Aggregation . . . . .	52
2.11 Kombination von Generalisierung und Aggregation . . . . .	54
2.12 Konsolidierung, Sichtenintegration . . . . .	55
2.13 Konzeptuelle Modellierung mit UML . . . . .	61
2.13.1 UML-Klassen . . . . .	61
2.13.2 Assoziationen zwischen Klassen . . . . .	62
2.13.3 Aggregation in UML . . . . .	63
2.13.4 Anwendungsbeispiel: Begrenzungsflächendarstellung von Polyedern in UML . . . . .	64
2.13.5 Generalisierung in UML-Notation . . . . .	65
2.13.6 Die Modellierung der Universität in UML . . . . .	65
2.13.7 Verhaltensmodellierung in UML . . . . .	66
2.13.8 Anwendungsfall-Modellierung (use cases) . . . . .	66

2.13.9 Interaktionsdiagramme . . . . .	68
2.13.10 Interaktionsdiagramm zur Prüfungsdurchführung . . . . .	68
2.14 Übungen . . . . .	69
2.15 Literatur . . . . .	71
<b>3 Das relationale Modell</b>	<b>73</b>
3.1 Definition des relationalen Modells . . . . .	73
3.1.1 Mathematischer Formalismus . . . . .	73
3.1.2 Schema-Definition . . . . .	74
3.2 Umsetzung eines konzeptuellen Schemas in ein relationales Schema .	75
3.2.1 Relationale Darstellung von Entitytypen . . . . .	75
3.2.2 Relationale Darstellung von Beziehungen . . . . .	75
3.3 Verfeinerung des relationalen Schemas . . . . .	80
3.3.1 1:N-Beziehungen . . . . .	80
3.3.2 1:1-Beziehungen . . . . .	82
3.3.3 Vermeidung von Null-Werten . . . . .	83
3.3.4 Relationale Modellierung der Generalisierung . . . . .	84
3.3.5 Beispieldarstellung der Universitäts-Datenbank . . . . .	85
3.3.6 Relationale Modellierung schwacher Entitytypen . . . . .	87
3.4 Die relationale Algebra . . . . .	87
3.4.1 Selektion . . . . .	88
3.4.2 Projektion . . . . .	89
3.4.3 Vereinigung . . . . .	89
3.4.4 Mengendifferenz . . . . .	90
3.4.5 Kartesisches Produkt (Kreuzprodukt) . . . . .	90
3.4.6 Umbenennung von Relationen und Attributen . . . . .	91
3.4.7 Definition der relationalen Algebra . . . . .	92
3.4.8 Der relationale Verbund (Join) . . . . .	92
3.4.9 Mengendurchschnitt . . . . .	97
3.4.10 Die relationale Division . . . . .	98
3.4.11 Gruppierung und Aggregation . . . . .	99
3.4.12 Operatorbaum-Darstellung . . . . .	100
3.5 Der Relationenkalkül . . . . .	100
3.5.1 Beispieldarstellung im relationalen Tupelkalkül . . . . .	101
3.5.2 Quantifizierung von Tupelvariablen . . . . .	102
3.5.3 Formale Definition des Tupelkalküls . . . . .	103
3.5.4 Sichere Ausdrücke des Tupelkalküls . . . . .	104
3.5.5 Der relationale Domänenkalkül . . . . .	104
3.5.6 Beispieldarstellung im Domänenkalkül . . . . .	105
3.5.7 Sichere Ausdrücke des Domänenkalküls . . . . .	106
3.6 Ausdruckskraft der Anfragesprachen . . . . .	107
3.7 Übungen . . . . .	107
3.8 Literatur . . . . .	110
<b>4 Relationale Anfragesprachen</b>	<b>113</b>
4.1 Geschichte . . . . .	113
4.2 Datentypen . . . . .	114
4.3 Schemadefinition . . . . .	114
4.4 Schemaveränderung . . . . .	115
4.5 Elementare Datenmanipulation: Einfügen von Tupeln . . . . .	116

4.6	Einfache SQL-Anfragen . . . . .	116
4.7	Anfragen über mehrere Relationen . . . . .	117
4.8	Aggregatfunktionen und Gruppierung . . . . .	120
4.9	Geschachtelte Anfragen . . . . .	121
4.10	Modularisierung von SQL-Anfragen . . . . .	125
4.11	Mengen-Operatoren . . . . .	126
4.12	Quantifizierte Anfragen in SQL . . . . .	127
4.13	Nullwerte . . . . .	129
4.14	Spezielle Sprachkonstrukte . . . . .	130
4.15	Joins in SQL-92 . . . . .	131
4.16	Rekursion . . . . .	132
4.17	Veränderungen am Datenbestand . . . . .	137
4.18	Sichten . . . . .	139
4.19	Sichten zur Modellierung von Generalisierungen . . . . .	140
4.20	Charakterisierung update-fähiger Sichten . . . . .	142
4.21	Einbettung von SQL in Wirtssprachen . . . . .	143
4.22	Anfragen in Anwendungsprogrammen . . . . .	144
4.23	JDBC: Java Database Connectivity . . . . .	146
4.23.1	Verbindungsauflauf zu einer Datenbank . . . . .	148
4.23.2	ResultSet-Programmbeispiel . . . . .	150
4.23.3	Vorübersetzung von SQL-Ausdrücken . . . . .	150
4.24	SQLJ: Eine Einbettung von SQL in Java . . . . .	152
4.25	Query by Example . . . . .	153
4.26	Übungen . . . . .	157
4.27	Literatur . . . . .	160
<b>5</b>	<b>Datenintegrität und temporale Daten</b>	<b>163</b>
5.1	Referentielle Integrität . . . . .	164
5.2	Gewährleistung referentieller Integrität . . . . .	165
5.3	Referentielle Integrität in SQL . . . . .	165
5.4	Überprüfung statischer Integritätsbedingungen . . . . .	166
5.5	Das Universitätsschema mit Integritätsbedingungen . . . . .	168
5.6	Komplexere Integritätsbedingungen . . . . .	170
5.7	Trigger . . . . .	171
5.8	Temporale Daten . . . . .	173
5.8.1	System-versionierte Relationen . . . . .	173
5.8.2	Temporale Daten nach Anwendungszeit . . . . .	174
5.9	Übungen . . . . .	176
5.10	Literatur . . . . .	177
<b>6</b>	<b>Relationale Entwurfstheorie</b>	<b>179</b>
6.1	Funktionale Abhängigkeiten . . . . .	179
6.1.1	Konventionen zur Notation . . . . .	180
6.1.2	Einhaltung einer funktionalen Abhängigkeit . . . . .	180
6.2	Schlüssel . . . . .	181
6.3	Bestimmung funktionaler Abhängigkeiten . . . . .	182
6.3.1	Kanonische Überdeckung . . . . .	185
6.4	„Schlechte“ Relationenschemata . . . . .	186
6.4.1	Die Updateanomalien . . . . .	186
6.4.2	Einfügeanomalien . . . . .	187

6.4.3	Löschanomalien . . . . .	187
6.5	Zerlegung (Dekomposition) von Relationen . . . . .	187
6.5.1	Verlustlosigkeit . . . . .	188
6.5.2	Kriterien für die Verlustlosigkeit einer Zerlegung . . . . .	190
6.5.3	Abhängigkeitsbewahrung . . . . .	191
6.6	Erste Normalform . . . . .	193
6.7	Zweite Normalform . . . . .	194
6.8	Dritte Normalform . . . . .	196
6.9	Boyce-Codd Normalform . . . . .	198
6.10	Mehrwertige Abhängigkeiten . . . . .	201
6.11	Vierte Normalform . . . . .	203
6.12	Zusammenfassung . . . . .	205
6.13	Übungen . . . . .	206
6.14	Literatur . . . . .	210
<b>7</b>	<b>Physische Datenorganisation</b>	<b>211</b>
7.1	Speichermedien . . . . .	211
7.2	Speicherhierarchie . . . . .	212
7.3	Speicherarrays: RAID . . . . .	214
7.4	Der Datenbankpuffer . . . . .	218
7.5	Abbildung von Relationen auf den Sekundärspeicher . . . . .	219
7.6	Indexstrukturen . . . . .	221
7.7	ISAM . . . . .	222
7.8	B-Bäume . . . . .	224
7.9	B <sup>+</sup> -Bäume . . . . .	228
7.10	Präfix-B <sup>+</sup> -Bäume . . . . .	230
7.11	Hintergrundspeicher-Struktur der B-Bäume . . . . .	230
7.12	Hashing . . . . .	232
7.13	Erweiterbares Hashing . . . . .	234
7.14	Mehrdimensionale Indexstrukturen . . . . .	238
7.15	Ballung logisch verwandter Datensätze . . . . .	242
7.16	Unterstützung eines Anwendungsverhaltens . . . . .	245
7.17	Physische Datenorganisation in SQL . . . . .	246
7.18	Übungen . . . . .	247
7.19	Literatur . . . . .	249
<b>8</b>	<b>Anfragebearbeitung</b>	<b>251</b>
8.1	Logische Optimierung . . . . .	252
8.1.1	Äquivalenzen in der relationalen Algebra . . . . .	254
8.1.2	Anwendung der Transformationsregeln . . . . .	256
8.1.3	Optimierung durch Entschachtelung von Unteranfragen . . . . .	260
8.2	Physische Optimierung . . . . .	266
8.2.1	Implementierung der Selektion . . . . .	267
8.2.2	Implementierung von binären Zuordnungsoperatoren . . . . .	268
8.2.3	Gruppierung und Duplikateliminierung . . . . .	275
8.2.4	Projektion und Vereinigung . . . . .	275
8.2.5	Zwischenspeicherung . . . . .	276
8.2.6	Sortierung von Zwischenergebnissen . . . . .	276
8.2.7	Übersetzung der logischen Algebra . . . . .	279
8.3	Kostenmodelle . . . . .	283

8.3.1	Selektivitäten . . . . .	284
8.3.2	Kostenabschätzung für die Selektion . . . . .	286
8.3.3	Kostenabschätzung für den Join . . . . .	287
8.3.4	Kostenabschätzung für die Sortierung . . . . .	288
8.4	„Tuning“ von Datenbankanfragen . . . . .	288
8.5	Kostenbasierte Optimierer . . . . .	290
8.5.1	Suchraum für die Join-Optimierung . . . . .	290
8.5.2	Dynamische Programmierung . . . . .	292
8.6	Übungen . . . . .	296
8.7	Literatur . . . . .	298
<b>9</b>	<b>Transaktionsverwaltung</b>	<b>301</b>
9.1	Begriffsbildung . . . . .	301
9.2	Anforderungen an die Transaktionsverwaltung . . . . .	302
9.3	Operationen auf Transaktions-Ebene . . . . .	302
9.4	Abschluss einer Transaktion . . . . .	303
9.5	Eigenschaften von Transaktionen . . . . .	305
9.6	Transaktionsverwaltung in SQL . . . . .	306
9.7	Zustandsübergänge einer Transaktion . . . . .	307
9.8	Literatur . . . . .	308
<b>10</b>	<b>Fehlerbehandlung</b>	<b>309</b>
10.1	Fehlerklassifikation . . . . .	309
10.1.1	Lokaler Fehler einer Transaktion . . . . .	309
10.1.2	Fehler mit Hauptspeicherverlust . . . . .	310
10.1.3	Fehler mit Hintergrundspeicherverlust . . . . .	311
10.2	Die Speicherhierarchie . . . . .	311
10.2.1	Ersetzung von Puffer-Seiten . . . . .	311
10.2.2	Einbringen von Änderungen einer Transaktion . . . . .	312
10.2.3	Einbringstrategie . . . . .	313
10.2.4	Hier zugrunde gelegte Systemkonfiguration . . . . .	314
10.3	Protokollierung von Änderungsoperationen . . . . .	314
10.3.1	Struktur der Log-Einträge . . . . .	315
10.3.2	Beispiel einer Log-Datei . . . . .	315
10.3.3	Logische oder physische Protokollierung . . . . .	315
10.3.4	Schreiben der Log-Information . . . . .	316
10.3.5	Das WAL-Prinzip . . . . .	318
10.4	Wiederanlauf nach einem Fehler . . . . .	318
10.4.1	Analyse des Logs . . . . .	319
10.4.2	Redo-Phase . . . . .	320
10.4.3	Undo-Phase . . . . .	320
10.5	Fehlertoleranz des Wiederanlaufs . . . . .	320
10.6	Lokales Zurücksetzen einer Transaktion . . . . .	322
10.7	Partielles Zurücksetzen einer Transaktion . . . . .	323
10.8	Sicherungspunkte . . . . .	324
10.8.1	Transaktionskonsistente Sicherungspunkte . . . . .	324
10.8.2	Aktionskonsistente Sicherungspunkte . . . . .	325
10.8.3	Unscharfe (fuzzy) Sicherungspunkte . . . . .	327
10.9	Recovery nach einem Verlust der materialisierten Datenbasis . . . . .	328
10.10	Übungen . . . . .	329

---

10.11 Literatur . . . . .	330
<b>11 Mehrbenutzersynchronisation</b>	<b>331</b>
11.1 Fehler bei unkontrolliertem Mehrbenutzerbetrieb . . . . .	332
11.1.1 Verlorengegangene Änderungen ( <i>lost update</i> ) . . . . .	332
11.1.2 Abhängigkeit von nicht freigegebenen Änderungen . . . . .	332
11.1.3 Phantomproblem . . . . .	333
11.2 Serialisierbarkeit . . . . .	333
11.2.1 Beispiele serialisierbarer Ausführungen (Historien) . . . . .	334
11.2.2 Nicht serialisierbare Historie . . . . .	334
11.3 Theorie der Serialisierbarkeit . . . . .	337
11.3.1 Definition einer Transaktion . . . . .	337
11.3.2 Historie (Schedule) . . . . .	338
11.3.3 Äquivalenz zweier Historien . . . . .	339
11.3.4 Serialisierbare Historien . . . . .	340
11.3.5 Kriterien für Serialisierbarkeit . . . . .	340
11.4 Eigenschaften von Historien bezüglich der Recovery . . . . .	342
11.4.1 Rücksetzbare Historien . . . . .	342
11.4.2 Historien ohne kaskadierendes Rücksetzen . . . . .	342
11.4.3 Strikte Historien . . . . .	343
11.4.4 Beziehungen zwischen den Klassen von Historien . . . . .	343
11.5 Der Datenbank-Scheduler . . . . .	344
11.6 Sperrbasierte Synchronisation . . . . .	345
11.6.1 Zwei Sperrmodi . . . . .	345
11.6.2 Zwei-Phasen-Sperrprotokoll . . . . .	346
11.6.3 Kaskadierendes Rücksetzen (Schneeballeffekt) . . . . .	348
11.7 Verklemmungen (Deadlocks) . . . . .	348
11.7.1 Erkennung von Verklemmungen . . . . .	349
11.7.2 Preclaiming zur Vermeidung von Verklemmungen . . . . .	350
11.7.3 Verklemmungsvermeidung durch Zeitstempel . . . . .	351
11.8 Hierarchische Sperrgranulate . . . . .	352
11.9 Einfüge- und Löschoperationen, Phantome . . . . .	356
11.10 Zeitstempel-basierende Synchronisation . . . . .	357
11.11 Optimistische Synchronisation . . . . .	359
11.12 Snapshot Isolation . . . . .	360
11.13 Klassifizierung der Verfahren . . . . .	361
11.14 Synchronisation von Indexstrukturen . . . . .	361
11.15 Mehrbenutzersynchronisation in SQL-92 . . . . .	365
11.16 Übungen . . . . .	367
11.17 Literatur . . . . .	369
<b>12 Sicherheitsaspekte</b>	<b>371</b>
12.1 Discretionary Access Control . . . . .	373
12.2 Zugriffskontrolle in SQL . . . . .	373
12.2.1 Identifikation und Authentisierung . . . . .	374
12.2.2 Autorisierung und Zugriffskontrolle . . . . .	374
12.2.3 Sichten . . . . .	375
12.2.4 Individuelle Sicht für eine Benutzergruppe . . . . .	376
12.2.5 k-Anonymität . . . . .	377
12.2.6 Auditing . . . . .	377

12.3	Verfeinerung des Autorisierungsmodells . . . . .	378
12.3.1	Rollenbasierte Autorisierung: Implizite Autorisierung von Subjekten . . . . .	379
12.3.2	Implizite Autorisierung von Operationen . . . . .	380
12.3.3	Implizite Autorisierung von Objekten . . . . .	380
12.3.4	Implizite Autorisierung entlang einer Typ hierarchie . . . . .	381
12.4	Mandatory Access Control . . . . .	383
12.5	Multilevel-Datenbanken . . . . .	383
12.6	SQL-Injection . . . . .	386
12.6.1	Attacken . . . . .	387
12.6.2	Schutz vor SQL-Injection-Attacken . . . . .	388
12.7	Kryptographie . . . . .	390
12.7.1	Der Data Encryption Standard . . . . .	390
12.7.2	Der Advanced Encryption Standard (AES) . . . . .	392
12.7.3	Public-Key-Kryptographie . . . . .	393
12.7.4	Public-Key-Infrastruktur (PKI) . . . . .	395
12.8	Zusammenfassung . . . . .	398
12.9	Übungen . . . . .	398
12.10	Literatur . . . . .	399
<b>13</b>	<b>Objektorientierte Datenbanken</b>	<b>401</b>
13.1	Bestandsaufnahme relationaler Datenbanksysteme . . . . .	401
13.2	Vorteile der objektorientierten Datenmodellierung . . . . .	405
13.3	Der ODMG-Standard . . . . .	406
13.4	Eigenschaften von Objekten . . . . .	407
13.4.1	Objektidentität . . . . .	408
13.4.2	Typ eines Objekts . . . . .	409
13.4.3	Wert eines Objekts . . . . .	409
13.5	Definition von Objekttypen . . . . .	410
13.5.1	Attribute . . . . .	410
13.5.2	Beziehungen . . . . .	410
13.5.3	Typeigenschaften: Extensionen und Schlüssel . . . . .	417
13.6	Modellierung des Verhaltens: Operationen . . . . .	417
13.7	Vererbung und Subtypisierung . . . . .	420
13.7.1	Terminologie . . . . .	420
13.7.2	Einfache und Mehrfachvererbung . . . . .	421
13.8	Beispiel einer Typ hierarchie . . . . .	422
13.9	Verfeinerung (Spezialisierung) und spätes Binden von Operationen . . . . .	425
13.10	Mehrfachvererbung . . . . .	428
13.11	Die Anfragesprache OQL . . . . .	429
13.11.1	Einfache Anfragen . . . . .	429
13.11.2	Geschachtelte Anfragen und Partitionierung . . . . .	430
13.11.3	Pfadausdrücke . . . . .	431
13.11.4	Erzeugung von Objekten . . . . .	432
13.11.5	Operationsaufruf . . . . .	432
13.12	C++-Einbettung . . . . .	432
13.12.1	Objektidentität . . . . .	434
13.12.2	Objekterzeugung und Ballung . . . . .	435
13.12.3	Einbettung von Anfragen . . . . .	435

13.13 Übungen . . . . .	436
13.14 Literatur . . . . .	437
<b>14 Erweiterbare und objekt-relationale Datenbanken</b>	<b>439</b>
14.1 Übersicht über die objekt-relationalen Konzepte . . . . .	439
14.2 Large Objects (LOBs) . . . . .	440
14.3 Distinct Types: Einfache benutzerdefinierte Datentypen . . . . .	442
14.4 Table Functions . . . . .	446
14.4.1 Nutzung einer <i>Table Function</i> in Anfragen . . . . .	447
14.4.2 Implementierung einer <i>Table Function</i> . . . . .	447
14.5 Benutzerdefinierte strukturierte Objekttypen . . . . .	449
14.6 Geschachtelte Objekt-Relationen . . . . .	453
14.7 Vererbung von SQL-Objekttypen . . . . .	457
14.8 Komplexe Attribut-Typen . . . . .	460
14.9 Übungen . . . . .	461
14.10 Literatur . . . . .	462
<b>15 Deduktive Datenbanken</b>	<b>463</b>
15.1 Terminologie . . . . .	463
15.2 Datalog . . . . .	463
15.3 Eigenschaften von Datalog-Programmen . . . . .	467
15.3.1 Rekursivität . . . . .	467
15.3.2 Sicherheit von Datalog-Regeln . . . . .	467
15.4 Auswertung von nicht-rekursiven Datalog-Programmen . . . . .	468
15.4.1 Auswertung eines Beispielprogramms . . . . .	468
15.4.2 Auswertungs-Algorithmus . . . . .	471
15.5 Auswertung rekursiver Regeln . . . . .	473
15.6 Inkrementelle (semi-naive) Auswertung rekursiver Regeln . . . . .	475
15.7 Bottom-Up oder Top-Down Auswertung . . . . .	479
15.8 Negation im Regelrumpf . . . . .	481
15.8.1 Stratifizierte Datalog-Programme . . . . .	481
15.8.2 Auswertung von Regeln mit Negation . . . . .	482
15.8.3 Ein etwas komplexeres Beispiel . . . . .	483
15.9 Ausdruckskraft von Datalog . . . . .	483
15.10 Übungen . . . . .	485
15.11 Literatur . . . . .	489
<b>16 Verteilte Datenbanken</b>	<b>491</b>
16.1 Terminologie und Abgrenzung . . . . .	491
16.2 Entwurf verteilter Datenbanken . . . . .	493
16.3 Horizontale und vertikale Fragmentierung . . . . .	495
16.3.1 Horizontale Fragmentierung . . . . .	496
16.3.2 Abgeleitete horizontale Fragmentierung . . . . .	498
16.3.3 Vertikale Fragmentierung . . . . .	499
16.3.4 Kombinierte Fragmentierung . . . . .	501
16.3.5 Allokation für unser Beispiel . . . . .	502
16.4 Transparenz in verteilten Datenbanken . . . . .	503
16.4.1 Fragmentierungstransparenz . . . . .	503
16.4.2 Allokationstransparenz . . . . .	504
16.4.3 Lokale Schema-Transparenz . . . . .	504

16.5	Anfrageübersetzung und -optimierung in VDBMS . . . . .	505
16.5.1	Anfragebearbeitung bei horizontaler Fragmentierung . . . . .	505
16.5.2	Anfragebearbeitung bei vertikaler Fragmentierung . . . . .	507
16.6	Join-Auswertung in VDBMS . . . . .	509
16.6.1	Join-Auswertung ohne Filterung . . . . .	509
16.6.2	Join-Auswertung mit Semijoin-Filterung . . . . .	510
16.6.3	Join-Auswertung mit Bitmap-Filterung . . . . .	512
16.7	Transaktionskontrolle in VDBMS . . . . .	514
16.8	Mehrbenutzersynchronisation in VDBMS . . . . .	519
16.8.1	Serialisierbarkeit . . . . .	519
16.8.2	Das Zwei-Phasen-Sperrprotokoll in VDBMS . . . . .	519
16.9	Deadlocks in VDBMS . . . . .	520
16.9.1	Erkennung von Deadlocks . . . . .	520
16.9.2	Deadlock-Vermeidung . . . . .	523
16.10	Synchronisation bei replizierten Daten . . . . .	524
16.11	Übungen . . . . .	527
16.12	Literatur . . . . .	530
<b>17</b>	<b>Betriebliche Anwendungen: OLTP, Data Warehouse, Data Mining</b>	<b>533</b>
17.1	SAP ERP: Ein betriebswirtschaftliches Datenbankanwendungssystem	533
17.1.1	Architektur von SAP ERP . . . . .	533
17.1.2	Datenmodell und Schema von SAP ERP . . . . .	534
17.1.3	ABAP/4 . . . . .	535
17.1.4	Transaktionen in SAP ERP . . . . .	538
17.2	Data Warehouse, Decision-Support, OLAP . . . . .	539
17.2.1	Datenbankentwurf für das Data Warehouse . . . . .	540
17.2.2	Anfragen im Sternschema: Star Join . . . . .	543
17.2.3	Roll-Up/Drill-Down-Anfragen . . . . .	544
17.2.4	Flexible Auswertungsmethoden . . . . .	546
17.2.5	Materialisierung von Aggregaten . . . . .	546
17.2.6	Der <b>cube</b> -Operator . . . . .	548
17.2.7	Wiederverwendung materialisierter Aggregate . . . . .	548
17.2.8	Bitmap-Indices für OLAP-Anfragen . . . . .	551
17.2.9	Auswertungsalgorithmen für komplexe OLAP-Anfragen . . . . .	552
17.3	Window-Funktionen in SQL . . . . .	554
17.4	Bewertung (Ranking) von Objekten . . . . .	562
17.4.1	Top-k-Anfragen . . . . .	562
17.4.2	Skyline-Anfragen . . . . .	566
17.4.3	Data Warehouse-Architekturen . . . . .	568
17.5	Data Mining . . . . .	570
17.5.1	Klassifikation von Objekten . . . . .	570
17.5.2	Assoziationsregeln . . . . .	571
17.5.3	Der $\Delta$ Priori-Algorithmus . . . . .	572
17.5.4	Bestimmung der Assoziationsregeln . . . . .	574
17.5.5	Cluster-Bestimmung . . . . .	575
17.6	Übungen . . . . .	577
17.7	Literatur . . . . .	579

<b>18 Hauptspeicher-Datenbanken</b>	<b>583</b>
18.1 Hardware-Entwicklungen . . . . .	583
18.2 Einsatz von Hauptspeicher-Datenbanken . . . . .	585
18.3 Leistungsengpässe heutiger Disk-basierter Datenbanksysteme . . . . .	586
18.4 Column Stores: Attribut-basierte Speicherung . . . . .	588
18.5 Datenstrukturen einer Hauptspeicher-DB . . . . .	592
18.5.1 Row-Store-Format . . . . .	593
18.5.2 Column-Store-Format . . . . .	593
18.5.3 Hybrides Speichermodell . . . . .	595
18.6 Anwendungs-Operationen in der Datenbank: Stored Procedures . . . . .	597
18.7 Architektur-Varianten für hybride OLTP/OLAP-Datenbanken . . . . .	600
18.7.1 Update Staging . . . . .	600
18.7.2 Heterogene Workload-Verwaltung . . . . .	601
18.7.3 Kontinuierliche Datawarehouse-Auffrischung . . . . .	602
18.7.4 Versionierung der transaktionalen Daten . . . . .	602
18.7.5 Batch-Verarbeitung . . . . .	602
18.7.6 Das Schattenspeicher-Konzept . . . . .	603
18.7.7 Berechnete Snapshots . . . . .	604
18.7.8 Reduzierte Isolationsstufen . . . . .	605
18.8 Snapshots des virtuellen Speichers . . . . .	605
18.9 Kompaktifizierung der Datenbank . . . . .	608
18.10 Transaktionsverwaltung . . . . .	612
18.11 Langlaufende Transaktionen . . . . .	614
18.12 Mehrbenutzersynchronisation mit multiplen Versionen . . . . .	617
18.13 Hochverfügbarkeit und Scale-Out für OLAP . . . . .	623
18.14 Indexstrukturen für Hauptspeicher-DBs . . . . .	625
18.15 Join-Berechnung . . . . .	628
18.15.1 Massiv Paralleler Sort/Merge-Join (MPSM) . . . . .	629
18.15.2 Paralleler Radix-Hash-Join . . . . .	632
18.15.3 Paralleler Hash-Join ohne Partitionierung . . . . .	633
18.16 Feingranulare adaptive Parallelisierung der Anfragebearbeitung . . . . .	635
18.17 Übungen . . . . .	639
18.18 Literatur . . . . .	640
<b>19 Internet-Datenbankanbindungen</b>	<b>645</b>
19.1 HTML- und HTTP-Grundlagen . . . . .	645
19.1.1 HTML: Die Hypertext-Sprache des World Wide Web . . . . .	645
19.1.2 Adressierung von Web-Dokumenten . . . . .	646
19.1.3 Client/Server-Architektur des World Wide Web . . . . .	648
19.1.4 HTTP: Das HyperText Transfer Protokoll . . . . .	648
19.1.5 HTTPS . . . . .	649
19.2 Web-Datenbank-Anbindung via Servlets . . . . .	650
19.2.1 Beispiel-Servlet . . . . .	650
19.3 Java Server Pages / Active Server Pages . . . . .	656
19.3.1 JSP/HTML-Seite mit Java-Code . . . . .	657
19.3.2 HTML-Seite mit Java-Bean-Aufruf . . . . .	659
19.3.3 Die Java-Bean Komponente <i>VorlesungenBean</i> . . . . .	660
19.3.4 Sokrates' Homepage . . . . .	662
19.4 Datenbankanbindung via Java-Applets . . . . .	662

19.5 Übungen . . . . .	663
19.6 Literatur . . . . .	664
<b>20 XML-Datenmodellierung und Web-Services</b>	<b>665</b>
20.1 XML-Datenmodellierung . . . . .	665
20.1.1 Schema oder kein Schema . . . . .	666
20.1.2 Rekursive Schemata . . . . .	668
20.1.3 Universitätsinformation in XML-Format . . . . .	668
20.1.4 XML-Namensräume . . . . .	670
20.1.5 XML Schema: Eine Schemadefinitionssprache . . . . .	672
20.1.6 Verweise (Referenzen) in XML-Daten . . . . .	674
20.2 XQuery: Eine XML-Anfragesprache . . . . .	675
20.2.1 Pfadausdrücke . . . . .	675
20.2.2 Verkürzte XPath-Syntax . . . . .	680
20.2.3 Beispiel-Pfadausdrücke in verkürzter Syntax . . . . .	681
20.2.4 Anfragesyntax von XQuery . . . . .	682
20.2.5 Geschachtelte Anfragen . . . . .	684
20.2.6 Joins in XQuery . . . . .	684
20.2.7 Join-Prädikat im Pfadausdruck . . . . .	685
20.2.8 Das let-Konstrukt . . . . .	686
20.2.9 Dereferenzierung in FLWOR-Ausdrücken . . . . .	687
20.2.10 Das if–then–else–Konstrukt . . . . .	689
20.2.11 Rekursive Anfragen . . . . .	690
20.3 Zusammenspiel von relationalen Datenbanken und XML . . . . .	692
20.3.1 XML-Repräsentation gemäß Pre- und Postorder-Rängen . . . . .	698
20.3.2 Der neue Datentyp xml . . . . .	702
20.3.3 Änderungen der XML-Dokumente . . . . .	706
20.3.4 Publikation relationaler Daten als XML-Dokumente . . . . .	707
20.3.5 Fallstudie: XML-Unterstützung in IBM DB2 V9 . . . . .	711
20.4 Web-Services . . . . .	716
20.4.1 Erstellen und Nutzen eines Web-Services im Überblick . . . . .	718
20.4.2 Das Auffinden von Diensten . . . . .	720
20.4.3 Ein Beispiel-Web-Service . . . . .	722
20.4.4 Definition der Web-Service-Schnittstellen . . . . .	722
20.4.5 Nachrichtenformat für die Interaktion mit Web-Services . . . . .	725
20.4.6 Implementierung des Web-Services . . . . .	727
20.4.7 Aufruf des Web-Services . . . . .	728
20.5 Übungen . . . . .	730
20.6 Literatur . . . . .	733
<b>21 Big Data</b>	<b>737</b>
21.1 Datenbanken für das Semantic Web . . . . .	737
21.1.1 RDF: Resource Description Framework . . . . .	737
21.1.2 SPARQL: Die RDF Anfragesprache . . . . .	740
21.1.3 Implementierung einer RDF-Datenbank . . . . .	742
21.2 Datenströme . . . . .	746
21.3 Information Retrieval und Suchmaschinen . . . . .	751
21.3.1 TF-IDF: Dokument-Ranking basierend auf Begriffs-Häufigkeit	752
21.3.2 Invertierte Indexierung . . . . .	754
21.3.3 Page Rank . . . . .	754

21.3.4 Der HITS Algorithmus . . . . .	757
21.4 Graph-Exploration (Graph Mining) . . . . .	760
21.4.1 Darstellung von Graphen . . . . .	760
21.4.2 Zentralitätsmaße . . . . .	763
21.4.3 Verbindungs-Zentralität (Degree Centrality) . . . . .	763
21.4.4 Nähe-Zentralität (Closeness Centrality) . . . . .	764
21.4.5 Pfad-Zentralität (Betweenness Centrality) . . . . .	765
21.5 MapReduce: Massiv parallele Datenverarbeitung . . . . .	766
21.6 Peer-to-Peer-Informationssysteme . . . . .	770
21.6.1 P2P-Systeme für den Datenaustausch (File-Sharing) . . . . .	771
21.6.2 Verteilte Hashtabellen (Distributed Hash Tables DHTs) . . . . .	773
21.6.3 Mehrdimensionaler P2P-Datenraum . . . . .	777
21.7 No-SQL- und Key/Value-Datenbanksysteme . . . . .	778
21.8 Multi-Tenancy, Cloud Computing und Software as a Service . . . . .	780
21.9 Übungen . . . . .	786
21.10 Literatur . . . . .	789
<b>22 Leistungsbewertung</b>	<b>793</b>
22.1 Überblick über Datenbanksystem-Benchmarks . . . . .	793
22.2 Der TPC-C Benchmark . . . . .	793
22.3 Die TPC-H und TPC-R (früher TPC-D) Benchmarks . . . . .	796
22.4 Der OO7 Benchmark für oo-Datenbanken . . . . .	802
22.5 Hybrider OLTP&OLAP-Benchmark: CH-BenCHmark . . . . .	803
22.6 Der TPC-W Benchmark . . . . .	806
22.7 Neue TPC-Benchmarks . . . . .	808
22.7.1 TPC-E: Der neue OLTP-Benchmark . . . . .	808
22.7.2 TPC-App: der neue Webservice-Benchmark . . . . .	810
22.7.3 TPC-DS: der neue Decision Support Benchmark . . . . .	811
22.8 Übungen . . . . .	812
22.9 Literatur . . . . .	812
<b>Literaturverzeichnis</b>	<b>815</b>
<b>Index</b>	<b>861</b>

# Vorwort

Wir drohen derzeit von einer wahren Informationsflut (Stichwort *Big Data*) „überrollt“ zu werden und sind auf dem besten Weg in die Informationsgesellschaft. Datenbanksysteme spielen eine immer größere Rolle in Unternehmen, Behörden und anderen Organisationen. Ihre Bedeutung wird durch die zunehmende weltweite Vernetzung – Internet und World Wide Web – noch stärker wachsen. Gleichzeitig wird der systematische Einsatz von Datenbanksystemen wegen der zunehmenden Informationsmenge, der Verteilung der Information auf ein Netz von Datenbankservern, der steigenden Komplexität der Anwendungen und der erhöhten Leistungsanforderungen immer schwieriger – auch wenn sich die Datenbanksystemprodukte weiterentwickeln.

In diesem Buch zur Einführung in Datenbanksysteme haben wir die Lehrinhalte zusammengestellt, die nach unserer Meinung für alle Informatik-nahen Studiengänge an Universitäten oder Fachhochschulen – wie z.B. Informatik, Software-Engineering, Wirtschafts-Informatik, Bio-Informatik etc. – essenziell sind.

Im Vergleich zu anderen Datenbank-Lehrbüchern setzen wir folgende Akzente:

- Es wurde ein durchgängiges Beispiel aus dem Hochschulbereich gewählt, das den Datenbankeinsatz gut illustriert. Dieses Beispiel haben wir bewusst einfach gehalten, damit man es sich gut einprägen kann. Für SQL-Übungen stellen wir auch eine Webschnittstelle unseres an der TUM entwickelten Datenbanksystems HyPer zur Verfügung: [www.hyper-db.de](http://www.hyper-db.de).
- Das Buch eignet sich auch zum Selbststudium, da wir uns bemüht haben, alle Konzepte an gut verständlichen Beispielen zu veranschaulichen. Eine ideale Ergänzung bietet darüber hinaus das neue *Übungsbuch Datenbanksysteme* von Kemper und Wimmer (2012), das Lösungsvorschläge für die Übungsaufgaben und weitergehende (teilweise multimediale) Lernhilfen enthält.
- Das Buch behandelt nur „moderne“ Datenbanksysteme. Sehr ausführlich gehen wir auf das relationale Modell ein, da es derzeit die marktbeherrschende Rolle spielt. Es werden aber auch neuere Entwicklungen, wie Hauptspeicher-Datenbanken, Big Data-Technologien und -Anwendungen, XML und Multi-Tenancy für Cloud-Datenbanken behandelt. Ältere Datenmodelle (die sogenannten satzorientierten Modelle, zu denen das Netzwerkmodell und das hierarchische Modell zählen) haben wir ausgeklammert, da diese Datenbanksysteme in absehbarer Zeit wohl nur noch „historische“ Bedeutung haben werden.
- Das Buch behandelt auch Implementierungsaspekte – wie z.B. physische Strukturen für die Datenverwaltung, Realisierungskonzepte für die Mehrbenutzersynchronisation und die Recovery, Optimierungsmethoden zur Anfrageauswertung etc. Auch wenn die wenigsten Informatiker später ein Datenbanksystem „bauen“ werden, so meinen wir doch, dass ein tiefgehendes Wissen unabdingbar

ist, um ein Datenbanksystem in der „harten“ industriellen Praxis systematisch einsetzen und optimieren zu können.

- Das Buch betont die praktischen Aspekte des Datenbankbereichs – ohne jedoch die theoretischen Grundlagen zu vernachlässigen. Die zugrundeliegende Theorie wird eingeführt, auf Beweise haben wir aber bewusst verzichtet.
- UML wird als objekt-orientierte Datenmodellierungs-Alternative zum ER-Mo dell eingeführt und die objekt-orientierten und objekt-relationalen Datenbankkonzepte werden detailliert diskutiert.
- Der Einsatz von Datenbanken als Data Warehouse für Decision Support-Anfragen sowie für das Data Mining wird beschrieben.
- Die XML-Datenbanktechnologien werden ausführlich behandelt: XML-Datenmodell, XPath und XQuery als Anfragesprachen sowie XML-basierte Web Services. Weiterhin wird die XML-Unterstützung der kommerziellen relationalen Datenbanksysteme ausführlich diskutiert.
- In dieser zehnten Auflage wurden die Ausführungen aktualisiert und neuere Entwicklungen aufgegriffen. In einigen Kapiteln (insbesondere in den Kapiteln über die relationale Anfragesprache SQL und deren logische Optimierung) wurde die Darstellung vertieft, um neuesten Entwicklungen gerecht zu werden. Insbesondere wurden in Kapitel 17 die SQL Window-Funktionen (oft auch SQL OLAP Funktionen genannt) sowie das Data Mining vertiefend dargestellt.
- Die neue, insbesondere auch von der SAP propagierte Entwicklung der **Hauptspeicher-Datenbanken** wurde in dem dedizierten Kapitel 18 vertiefend behandelt. Diese Systeme machen sich die neuesten Hardwareentwicklungen im Bezug auf Multi-Core-Parallelisierung und auf TeraByte-Level skalierte Hauptspeichergrößen zunutze, um mit einer neuen Datenbank-Architektur dramatische Leistungssteigerungen im Vergleich zu traditionellen Sekundärspeicher-Datenbanken zu erzielen.
- In dem Kapitel **Big Data** wurden die Techniken für die Beherrschbarkeit der Informationsflut des Webs, wie NoSQL Key-Value-Speicher, RDF/SPARQL als Grundlage des *Semantic Web*, allgemeine Graph-Datenrepräsentationen sowie Graphexploration, Information Retrieval und Suchmaschinen-Grundlagen (u.a. PageRank), hochgradig verteilte Datenverarbeitung (MapReduce), Datenströme, und Cloud/Multi-Tenancy Datenbanken aktualisiert und vertieft.
- Zusätzliche Unterlagen zu diesem Buch findet man über unseren Webserver (<http://www-db.in.tum.de>).

Wir haben uns bemüht, die inhaltlichen Abhängigkeiten zwischen den Kapiteln gering zu halten. Deshalb ist es problemlos möglich (und wird von uns an der TU München auch praktiziert), eine schon im Grundstudium enthaltene Einführung in Datenbanksysteme – in der beispielsweise die Grundlagen der konzeptuellen Datenmodellierung, der physischen Datenorganisation, des relationalen Datenmodells und der Anfragesprache SQL, der Datenbanktheorie und der Transaktionsverwaltung

vermittelt werden – aus diesem Buch zu „extrahieren“, um das Themengebiet dann im Hauptstudium mit den übrigen Kapiteln zu vervollständigen. Es ist auch möglich, einige der weiterführenden Kapitel in einer Vorlesung zur Datenbankimplementierung oder in einer projektorientierten Datenbankeinsatz-Vorlesung zu „verwerten“.

**Danksagung** Dr. Reinhard Braumandl, Dr. Christian Wiesner, Dr. Jens Claußen, Dr. Carsten Gerlhof, Prof. Donald Kossmann, Dr. Natalija Krivokapić, Dr. Klaus Peithner und Dr. Michael Steinbrunn danken wir für ihre Hilfe bei früheren Auflagen. Dr. Stefan Seltzsam, Dr. Richard Kuntschke und Dr. Martin Wimmer haben wesentlich bei der Ausarbeitung zu den Web-Datenbankschnittstellen geholfen. Dr. Martina-Cezara Albutiu und Herr Stefan Kinauer haben bei der Korrektur geholfen. Meinem Kollegen, Prof. Thomas Neumann, sowie den Doktoranden am Lehrstuhl, die im HyPer-Projekt forschen bzw. geforscht haben (Dr. Martina-Cezara Albutiu, Dr. Stefan Aulbach, Robert Brunel, Jan Finis, Dr. Florian Funke, Harald Lang, Viktor Leis, Dr. Henrik Mühe, Tobias Mühlbauer, Dr. Angelika Reiser, Wolf Rödiger, Dr. Michael Seibold, Manuel Then) danke ich für die Zusammenarbeit – sie finden einige ihrer Forschungsergebnisse in den neueren Kapiteln dieser zehnten Auflage wieder.

Wir haben von etlichen „externen“ Lesern Anregungen bekommen. Besonders hilfreich waren die Hinweise von Prof. Stefan Brass, Prof. Sven Helmer, Prof. Volker Linnemann, Prof. Guido Moerkotte, Prof. Reinhard Pichler, Prof. Erhard Rahm, Prof. Stefanie Scherzinger, Frau Katrin Seyr, Prof. Bernhard Thalheim und Prof. Rainer Weber. Dr. Michael Ley danken wir weiterhin für seinen phantastischen Bibliographie-Server <http://dblp.uni-trier.de/>.

München, im August 2015

*Alfons Kemper*