

Inhaltsverzeichnis

Teil I Einführung

1	Einleitung	3
1.1	Von der zentralisierten zur verteilten Datenverarbeitung	5
1.2	Anforderungen an verteilte und parallele DMS	9
1.3	Maße der parallelen Datenverarbeitung: Speedup und Scaleup	16
1.4	Übungsaufgaben	19
2	Grundlagen zu Datenbanken und verteilten Systemen	23
2.1	Relationales Datenmodell	24
2.2	Aufbau von Datenbanksystemen	27
2.2.1	Schemaarchitektur	27
2.2.2	Funktioneller Aufbau relationaler Datenbanksysteme	28
2.3	Das Transaktionskonzept (ACID)	30
2.4	DBS-Einsatz: OLTP vs. OLAP	31
2.4.1	Online Transaction Processing	31
2.4.2	OLAP und Data Warehouses	32
2.5	NoSQL-Systeme	34
2.6	Rechnernetze und verteilte Systeme	37
2.6.1	Typen von Rechnernetzen	37
2.6.2	Kommunikationsprotokolle	38
2.6.3	Kommunikation in Clustern und Datacentern	39
2.7	Übungsaufgaben	40
3	Architekturen für verteiltes und paralleles Datenmanagement	43
3.1	Arten der Parallelverarbeitung	43
3.1.1	Transaktions-, Query- und Operatorparallelität	44
3.1.2	Daten- vs. Pipelineparallelität	46
3.1.3	Verarbeitungs- vs. E/A-Parallelität	47
3.2	Klassifikationskriterien	48
3.2.1	Daten-/Externspeicherzuordnung	50

3.2.2	Räumliche Anordnung	51
3.2.3	Rechnerkopplung	52
3.3	Parallele DBS-Architekturen	53
3.3.1	Shared Everything	53
3.3.2	Shared Disk vs. Shared Nothing	55
3.4	Architekturen mit funktionaler Spezialisierung	58
3.4.1	Client-Server-Systeme	59
3.4.2	Hardwareunterstützung zur DB-Verarbeitung	61
3.5	Alternativen zur Unterstützung heterogener DBS	62
3.6	Hadoop als Big-Data-Plattform	65
3.6.1	Das verteilte Dateisystem HDFS	66
3.6.2	Hadoop's MapReduce-Engine	69
3.6.3	Weitere Hadoop-Komponenten	71
3.7	Vergleich und qualitative Bewertung	72
3.7.1	Vergleich verteilter Architekturen zur Datenbankverarbeitung	72
3.7.2	Architekturen für Big Data	74
3.8	Übungsaufgaben	75

Teil II Katalogverwaltung und Datenverteilung

4	Schemaarchitektur und Katalogverwaltung	81
4.1	Schemaarchitektur von Verteilten DBS	81
4.2	Katalogverwaltung	82
4.3	Namensverwaltung	84
4.3.1	Objektnamen in Verteilten DBS	85
4.3.2	Nutzung globaler Namensräume	87
4.4	Übungsaufgaben	88
5	Grundlagen der Datenverteilung	91
5.1	Fragmentierung	92
5.1.1	Horizontale Fragmentierung	94
5.1.2	Vertikale Fragmentierung	97
5.1.3	Hybride Fragmentierung	98
5.1.4	Fragmentierungstransparenz	100
5.2	Allokation und Replikation	101
5.3	Bestimmung der Datenverteilung	102
5.4	Übungsaufgaben	104
6	Datenverteilung in Parallelen DBS	107
6.1	Bestimmung des Verteilgrades	108
6.2	Fragmentierung	110
6.2.1	Basisansätze	111

6.2.2	Verfeinerte Bereichsfragmentierung	113
6.2.3	Mehrdimensionale Fragmentierung	114
6.2.4	Mehrstufige Fragmentierung	117
6.3	Allokation	119
6.4	Replikation	120
6.4.1	Spiegelplatten/Disk-Arrays	121
6.4.2	Verstreuete Replikation	122
6.4.3	Verkettete Replikation	123
6.5	Datenverteilung für Shared Everything und Shared Disk	124
6.5.1	Verteil- und Parallelitätsgrad	124
6.5.2	Fragmentierung	125
6.5.3	Indexallokation	126
6.5.4	Replikation	126
6.6	Datenallokation in NoSQL-Systemen	127
6.6.1	Consistent Hashing	127
6.6.2	Auto-Sharding in MongoDB	129

Teil III Anfrageverarbeitung

7	Grundlagen der Anfrageverarbeitung	139
7.1	Phasen der Anfrageverarbeitung	139
7.2	Planoperatoren	141
7.2.1	Scan	141
7.2.2	Sortieren	142
7.2.3	Verbund	144
7.2.4	Pipelining	145
7.3	Kostenmodelle	146
7.4	Anfrageoptimierung	148
7.5	Übungsaufgaben	151
8	Verteilte Anfrageverarbeitung	153
8.1	Phasen der verteilten Anfrageverarbeitung	153
8.2	Datenlokalisierung und Anfragezerlegung	155
8.2.1	Datenlokalisierung bei primärer horizontaler Fragmentierung	155
8.2.2	Datenlokalisierung bei abgeleiteter horizontaler Fragmentierung	158
8.2.3	Datenlokalisierung bei vertikaler Fragmentierung	159
8.2.4	Datenlokalisierung bei hybrider Fragmentierung	160
8.3	Kostenmodelle für verteilte Anfragen	161
8.4	Verteilte Verbundberechnung	163
8.4.1	Ship Whole vs. Fetch Matches	163
8.4.2	Semiverbund	164

8.4.3	Weitere Verbundvarianten	168
8.5	Mehr-Wege-Verbunde	169
8.6	Übungsaufgaben	171
9	Parallele Anfrageverarbeitung	175
9.1	Parallele Realisierung unärer Anfrageoperatoren	175
9.2	Paralleles Sortieren	177
9.3	Parallele Verbundberechnung	180
9.3.1	Dynamische Replikation	181
9.3.2	Dynamische Partitionierung	182
9.3.3	Parallele Hash-Verbunde	184
9.3.4	Verbundberechnung für Multicore-Systeme	185
9.4	Der Exchange-Operator	190
9.5	Optimierung und Ressourcenallokation	191
9.5.1	Optimierung von Mehr-Wege-Verbunden	192
9.5.2	Prozessorallokation	194
9.5.3	Skew-Behandlung	196
9.6	Übungsaufgaben	200
10	Parallele Analyse großer Datenmengen mit MapReduce	203
10.1	Grundlagen	203
10.2	MapReduce als Programmiermodell	205
10.3	MapReduce-Programmierung mit Hadoop	207
10.4	Datenbankoperationen mit MapReduce	211
10.5	Datenflusssprachen für MapReduce	212
10.6	Apache Spark	217
10.7	Übungsaufgaben	221

Teil IV Konsistenzsicherung

11	Verteilte Transaktionsausführungen	227
11.1	Struktur verteilter Transaktionen	228
11.2	Konsensus-Protokolle	229
11.3	Das Zwei-Phasen-Commit	230
11.3.1	Das Zwei-Phasen-Commit-Protokoll	231
11.3.2	Probleme beim 2PC	234
11.4	Varianten des 2PC	234
11.4.1	Lineares 2PC	234
11.4.2	Verteiltes 2PC	235
11.4.3	Hierarchisches 2PC	236
11.5	Das Drei-Phasen-Commit-Protokoll	237

11.5.1	Phasen des Drei-Phasen-Commit-Protokolls	237
11.5.2	Probleme des Drei-Phasen-Commit-Protokolls	240
11.6	Das Paxos-Protokoll	241
11.7	Recovery	243
11.7.1	Klassisches Recovery	243
11.7.2	Recovery in verteilten DBMS	244
11.8	Strukturierte Transaktionen	245
11.8.1	Geschachtelte Transaktionen	245
11.8.2	Geschlossen geschachtelte Transaktionen	246
11.8.3	Offen geschachtelte Transaktionen	248
11.8.4	Eigenschaften geschachtelter Transaktionen	249
11.9	Übungsaufgaben	250
12	Synchronisationsverfahren	253
12.1	Grundlagen der Synchronisation	253
12.1.1	Serialisierbarkeitsbegriffe	254
12.1.2	Abhängigkeitsgraphen	255
12.1.3	Anforderungen für Verteilte DBS	256
12.2	Sperrverfahren in Verteilten DBS	256
12.2.1	Zentrale Sperrprotokolle	257
12.2.2	Verteilte Sperrverfahren	257
12.3	Zeitmarkenverfahren	258
12.3.1	Prinzip des Zeitmarkenverfahrens	258
12.3.2	Realisierung des Zeitmarkenverfahrens	258
12.4	Optimistische Synchronisation	260
12.4.1	Validierungsansätze	261
12.4.2	Zentrale Validierung	264
12.4.3	Verteilte Validierung	264
12.5	Mehrversionensynchronisation	266
12.5.1	Grundlagen	267
12.5.2	Mehrversionenansätze mit zentralisierter Kontrolle	269
12.5.3	Verteilte Mehrversionensynchronisation	270
12.6	Deadlock-Behandlung	272
12.6.1	Deadlock-Verhütung und -Vermeidung	273
12.6.2	Timeout-Verfahren	276
12.6.3	Deadlock-Erkennung	277
12.6.4	Hybride Strategien	280
12.7	Übungsaufgaben	281
13	Replikation	285
13.1	Replikation in verteilten Datenbanksystemen	285
13.2	Klassifikation von Replikationsansätzen	287

13.2.1	Synchrone Aktualisierung mit Primärkopie	288
13.2.2	Synchrone Aktualisierung mit verteilten Änderungen	288
13.2.3	Verzögerte Aktualisierung mit Primärkopie	289
13.2.4	Verzögerte Aktualisierung mit verteilten Änderungen	290
13.2.5	Klassifikation pessimistischer Replikationsstrategien	290
13.3	Write-All-Ansätze	291
13.4	Primary-Copy-Verfahren	292
13.5	Votingverfahren	294
13.5.1	Mehrheitsvotieren (Majority Consensus)	295
13.5.2	Gewichtetes Votieren (Quorum Consensus)	296
13.6	Schnappschussreplikation	297
13.7	Katastrophen-Recovery	302
13.7.1	Systemstruktur	302
13.7.2	Commit-Behandlung	304
13.7.3	Geo-Replikation	305
13.8	Übungsaufgaben	306
14	Transaktionsverarbeitung für Shared Disk	309
14.1	Systemeigenschaften und Herausforderungen	309
14.2	Synchronisation in Shared-Disk-DBS	311
14.2.1	Verfahrensüberblick	312
14.2.2	Globale Sperrverwaltung auf dedizierten Rechnern	314
14.2.3	Nutzung von Lese- und Schreibautorisierungen	315
14.2.4	Verteilte Sperrverfahren mit fester GLA-Zuordnung	317
14.2.5	Verteilte Sperrverfahren mit dynamischer GLA-Zuordnung	320
14.3	Kohärenzkontrolle	321
14.3.1	Behandlung von Pufferinvalidierungen	323
14.3.2	Update-Propagierung	325
14.3.3	On-Request-Invalidierung	329
14.3.4	Einsatz von Haltesperren	333
14.3.5	Zusammenfassende Bewertung	335
14.4	Logging und Recovery	337
14.5	Nah gekoppelte Shared-Disk-Systeme	339
14.5.1	Einsatzformen der nahen Kopplung	340
14.5.2	Realisierungsalternativen	341
14.5.3	IBM Parallel Sysplex und DB2 PureScale	343
14.6	Übungsaufgaben	347
15	Konsistenz in Cloud-Datenbanken	353
15.1	Konsistenzanforderungen in Cloud-Szenarien	353
15.1.1	Von ACID zu BASE	353
15.1.2	Das CAP-Theorem	354

15.1.3	Abgeschwächte Konsistenzmodelle	358
15.2	Replikation in Cloud-Datenbanken	359
15.2.1	Optimistische Replikation	359
15.2.2	Phasen der optimistischen Replikation	360
15.2.3	Konflikterkennung und Strategien der Konfliktauflösung	361
15.2.4	Vektoruhren	362
15.2.5	Epidemische Propagierung	364
15.3	Konsistenz in kommerziellen Cloud-Datenbanken	365
15.4	Übungsaufgaben	368
	Sachverzeichnis	371