

Verteilte Datenbanken

Systemkonzepte
und Produkte

Von Hermann Kudlich

Inhaltsverzeichnis

1	Verteilte Verarbeitung: Ein neues Konzept	11	4	Verteiltes Transaktions- management	69
1.1	Der Nutzen verteilter Systeme	12	4.1	Eigenschaften von Transaktionen	69
1.2	Kommunikationsformen verteilter Verarbeitung	15	4.2	Ziele des Transaktionsmanagements	71
1.3	Zwei unterschiedliche Verteilungs- formen: Verteilte Datenbanken – Verteilte Verarbeitung	19	4.3	Sperrmechanismen in zentralisierten Datenbanksystemen	73
2	Basisfunktionen verteilter Datenbanksysteme	25	4.4	Zwei-Phasen-Commit-Protokoll (2PC-Protokoll)	77
2.1	Referenzarchitektur	25	4.4.1	Funktionsweise	77
2.2	Relationale Operationen	27	4.4.2	Mögliche Fehlersituationen innerhalb des 2PC-Protokolls	80
2.3	Arten der Datenfragmentierung	30	4.4.3	Verbesserung des 2PC-Protokolls	81
2.4	Verteilungstransparenz	32	4.4.4	Zustandsdiagramm des 2PC-Protokolls	83
2.5	Kriterien für die Verteilung von Fragmenten	36	4.5	Drei-Phasen-Commit-Protokoll (3PC-Protokoll)	85
3	Optimierung von Zugriffsstrategien	38	4.6	Kommunikationsstrukturen für Ende-Protokolle	87
3.1	Optimierungsziele	38	4.7	Sperralgorithmen in verteilten Datenbanksystemen	89
3.2	Lokale Optimierungsstrategien	40	4.8	Deadlockerkennung in verteilten Datenbanksystemen	92
3.3	Zerlegung von Datenbankabfragen	44	4.8.1	Timeout-Verfahren	92
3.3.1	Äquivalenztransformation	46	4.8.2	Zentrale Deadlockerkennung	92
3.3.2	Kanonischer Ausdruck	47	4.8.3	Verteilte Deadlockerkennung	96
3.3.3	Vereinfachung bei horizontaler Fragmentierung	49	4.8.4	Verteilte Deadlockerkennung bei SESAM	98
3.3.4	Vereinfachung bei vertikaler Fragmentierung	51	4.8.5	Falsche Deadlocks	99
3.3.5	Semi-Join-Operation	52	4.9	Deadlockverhütung in verteilten Datenbanksystemen	100
3.3.6	Verteilung von Built-in-Funktionen	53	4.9.1	Aktive Deadlock- Verhütungsmethode	100
3.3.7	Dynamische Queries	56	4.9.2	Passive Deadlock- Verhütungsmethode	100
3.4	Globale Optimierungsstrategien	57	4.9.3	Zeitstempelmethode	102
3.4.1	Bewertungskriterien des Semi-Join-Programmes	60	4.10	Update-Probleme bei replizierten Datenbanken	106
3.4.2	Join-Strategien	61	4.10.1	Write-all-read-any-Verfahren	107
3.4.3	Zusammengesetzte Join- und Union-Queries	61	4.10.2	Majoritäts-Verfahren (Voting-Verfahren)	109
3.4.4	Eigenschaften eines globalen Optimizers	64	4.10.3	Primary-Copy-Verfahren	112

5	Architektur verteilter Datenbanksysteme	113	8	Das Datenbanksystem SESAM/SQL	177
5.1	Die Kommunikationsfunktion	114	8.1	Verwaltung linear und relational organisierter Datenstrukturen	177
5.2	Lokale Netze	116	8.2	Die Systemarchitektur von SESAM	179
5.2.1	Netztopologien	117	8.3	Die Arbeitsweise der Verteilungskomponente SESAM-DCN	182
5.2.2	Übertragungsmedien	119	8.4	Lokalitätstransparenz	184
5.2.3	Zugriffsverfahren	122	8.5	Das Transaktionskonzept von SESAM	185
5.2.4	Komponenten von lokalen Netzen	125	8.5.1	Sperrkonzept	185
5.3	Die Prozeßstruktur	126	8.5.2	Deadlockbehandlung	186
5.4	Die Verteilungskomponente	130	8.5.3	Longlock	187
5.5	Funktionskategorien verteilter Systeme	133	8.5.4	2PC-Protokoll	187
5.5.1	Verteilungsformen	134	8.5.5	Wiederanlauf	187
5.5.2	Client-Server-Architektur	137	8.5.6	Sicherheitskonzept	188
5.6	Erweiterte Client-Server-Strukturen: Gateways	139	8.6	Lokale und zentrale Administration	189
5.6.1	Funktionalität von Gateways	139	8.7	Verteilte Datenbankintegrität	190
5.6.2	Unterschiede in Semantik und Syntax der SQL	141	8.8	Zugriffsoptimierung	190
5.6.3	Probleme beim Einsatz von Gateways	144	9	Das Datenbanksystem INFORMIX	193
6	Anforderungen an verteilte Datenbanken	146	9.1	Die Systemarchitektur von INFORMIX	193
6.1	12 Regeln für verteilte Datenbanken	146	9.1.1	Prozeßstruktur	194
6.2	Probleme verteilter Datenbanken	153	9.1.2	Physisches und logisches Speicherkonzept	195
7	Das Datenbanksystem UDS	155	9.2	Verteilung mit den Komponenten INFORMIX-NET und INFORMIX-STAR	196
7.1	Das CODASYL-Modell	155	9.3	Lokalitätstransparenz	197
7.2	Die Systemarchitektur von UDS	156	9.4	Das Transaktionskonzept von INFORMIX	201
7.3	Die Arbeitsweise der Verteilungskomponente UDS-D	158	9.4.1	Transaktionssicherung	201
7.4	Lokalitätstransparenz	162	9.4.2	Restart	202
7.5	Das Transaktionskonzept	165	9.4.3	Sperrstrategie	202
7.5.1	Primäre und sekundäre Teiltransaktionen	165	9.4.4	Isolation gegenüber konkurrierendem Zugriff	202
7.5.2	Sichern der konfigurationsübergreifenden Konsistenz mit dem Zwei-Phasen-Commit-Protokoll (2PC-Protokoll)	167	9.4.5	Deadlockbehandlung	203
7.5.3	Der Zustand PTC (Prepared to Commit)	168	9.4.6	2PC-Protokoll	204
7.5.4	Zeitgesteuertes Überwachen	170	9.5	Lokale und zentrale Autonomie	204
7.5.5	Auswirkungen des Zustands PTC	171	9.6	Verteilte Datenbankintegrität	204
7.5.6	Sperrverhalten und Deadlockerkennung	173	9.7	Zugriffsoptimierung	204
7.6	Lokale und zentrale Administration	173	10	Das Datenbanksystem ORACLE	206
7.7	Verteilte Datenbankintegrität	174	10.1	Die Systemarchitektur von ORACLE	206
7.8	Zugriffsoptimierung	175	10.1.1	Prozeßstruktur	206
			10.1.2	Physisches und logisches Speicherkonzept	208
			10.2	Verteilung mit SQL*NET	210
			10.3	Lokalitätstransparenz	214

10.4	Das Transaktionskonzept von ORACLE	216	11.1.2	Physisches und logisches Speicherkonzept	224
10.4.1	Transaktionssicherung	216	11.2	Verteilung mit den Komponenten INGRES/NET und	
10.4.2	Restart	217		INGRES/STAR	226
10.4.3	Sperrstrategie	217	11.3	Lokalitätstransparenz	228
10.4.4	Parallelität lesender Transaktionen . .	218	11.4	Das Transaktionskonzept von INGRES	229
10.4.5	Deadlockbehandlung	218	11.5	Lokale Autonomie	231
10.4.6	2PC-Protokoll	219	11.6	Verteilte Datenbankintegrität	231
10.5	Lokale Autonomie	219			
10.6	Verteilte Datenbankintegrität	219			
10.7	Zugriffsoptimierung	220			
11	Das Datenbanksystem INGRES . . .	222		Literaturverzeichnis	234
11.1	Die Systemarchitektur von INGRES .	223			
11.1.1	Prozeßstruktur	223		Stichwortverzeichnis	235