

Inhaltsverzeichnis

1	Einleitung	11
1.1	Problemstellung und Zielsetzung.....	14
1.2	Aufbau der Arbeit	15
2	Verteilte Datenverwaltungssysteme	18
2.1	Mehrrechner-Datenbanksysteme	19
2.1.1	Architektur von Mehrrechner-Datenbanksystemen	19
2.1.2	Rechnernetze	22
2.1.3	Kommunikation.....	24
2.2	Traditionelle Anforderungen an verteilte DBVS	26
2.2.1	Transparenz	28
2.2.2	Autonomie.....	30
2.2.3	Konsistenz	33
2.3	Integrationsstufen der verteilten Datenverwaltung	37
2.4	Grundlagen der Datenverteilung	40
2.4.1	Fragmentierung und Allokation	42
2.4.2	Katalogverwaltung	44
2.5	Transaktionale Datenverarbeitung	46
2.5.1	Serialisierungstheorie	47
2.5.2	Synchronisation und Recovery	49
2.5.3	Verteilte Transaktionsverarbeitung.....	52
2.5.4	Handhabung von Verklemmungen	58
3	Grundlagen der Datenreplikation.....	60
3.1	Begriffsbestimmungen und logische Grundlagen	60
3.2	Zielsetzung	64
3.2.1	Verfügbarkeit und Zuverlässigkeit.....	64
3.2.2	Antwortzeitverhalten.....	69
3.2.3	Korrektheit	70
3.3	Verfahren zur Replikationskontrolle.....	76
3.3.1	ROWA (Read-One/Write-All).....	76
3.3.2	ROWAA (Read-One/Write-All-Available)	77
3.3.3	Votierungsverfahren.....	78
3.3.4	Primary-Copy Verfahren.....	82
3.3.5	Mischverfahren	85
3.3.6	Einfluß der Replikationskontrolle auf Verklemmungen	87
3.4	Bewertung von Replikationsprotokollen	88
3.4.1	Allgemeine Bewertungskriterien	90
3.4.2	Methoden zur Berechnung von Kennzahlen	91
3.4.3	Schlußbemerkung.....	94

4	Datenverwaltung aus Anwendungssicht.....	96
4.1	Ubiquität und Need-To-Know	96
4.2	Transaktionskorrektheit und Datenbankkonsistenz	99
4.3	Anforderungen zur adaptiven verteilten Datenhaltung	102
4.3.1	Zugriffslokalität	105
4.3.2	Aktualitätsanforderungen	107
4.3.3	Synchronisationsanforderungen	109
4.4	Wege zu einer adaptiven Datenverwaltung	110
4.4.1	Views.....	111
4.4.2	Snapshots.....	112
4.4.3	Quasi-Kopien	113
4.4.4	Identity Connections	114
4.4.5	Data Dependency Descriptors.....	115
4.4.6	DDMS und Propagierungsregeln	118
4.4.7	Epsilon-Serialisierbarkeit	124
4.4.8	Konvergenzsicherung bei asynchroner Propagierung von Änderungen	126
4.4.9	Replikation in kommerziellen Datenbanksystemen.....	130
4.5	Klassifikation und Bewertung.....	138
4.5.1	Ansätze ohne Konsistenzspezifikation.....	139
4.5.2	Datenorientierte vs. ablaforientierte Ansätze	139
4.5.3	Techniken zur Aktualisierung von Replikaten.....	142
4.5.4	Techniken zur Behandlung konfligierender Änderungen	143
4.5.5	Zusammenfassung.....	146
5	ASPECT - Konzeptioneller Entwurf einer Spezifikationsmethodologie für Konsistenzanforderungen.....	148
5.1	Referenzmodell eines verteilten Datenverwaltungssystems	148
5.2	Das Konzept der "virtuellen Primärkopie"	153
5.3	Dimensionen der Konsistenzspezifikation	155
5.4	Spezifikation von Kohärenzprädikaten	158
5.4.1	Spezifikation eines tolerierbaren Abstands	159
5.4.2	Synchronisationsanforderungen	162
5.5	Zeitabhängigkeit von Konsistenzanforderungen	165
5.5.1	Ereignisse	165
5.5.2	Spezifikation von Ereignissen	167
5.5.3	Lebensdauer und Gültigkeitsbereich von Ereignissen	169
5.6	Mögliche Zustände von Replikaten	173
5.7	Spezifikation von Integritätsbedingungen	174

6	Adaptive Replikationskontrolle	178
6.1	Das Konzept der Konsistenzinseln	178
6.1.1	Auffinden der Konsistenzinsel	181
6.1.2	Erweiterte Sperrkonzepte	184
6.2	Synchronisation innerhalb der Konsistenzinsel	188
6.2.1	Replikationskontrolle auf der Basis von Quorum Consensus	188
6.2.2	Replikationskontrolle auf der Basis von Virtual Partition	192
6.3	Ereignisorientierte Aktualisierung schwach konsistenter Replikate	197
6.3.1	Aktualisierungsnachrichten	198
6.3.2	Sicherstellung ereignisorientierter Konsistenzanforderungen	199
6.4	Synchrone Wartung kontinuierlicher Konsistenzanforderungen.....	202
6.4.1	Synchrone Sicherstellung eines maximalen Versionsabstands...	206
6.4.2	Synchrone Sicherstellung eines maximalen Wertabstands	211
6.4.3	Sicherstellung eines maximalen Zeitrückstands	213
6.5	Asynchrone Wartung kontinuierlicher Konsistenzanforderungen.....	216
6.5.1	Maßnahmen zur Einrichtung eines Alive-Senders in der Konsistenzinsel	217
6.5.2	Inhalt von Alive-Nachrichten.....	218
6.5.3	Alternativen zur asynchronen Änderungspropagierung.....	220
6.6	Sicherung der Transaktionskonsistenz.....	222
6.6.1	Vollständige Aktualisierung von Integritätsbereichen.....	223
6.6.2	Alternative Aktualisierungsstrategien.....	224
6.7	Bewertung adaptiver Replikationsprotokolle mit Konsistenzinseln	228
6.7.1	Änderungsverfügbarkeit unter Wahrung der One-Copy- Serialisierbarkeit	229
6.7.2	Vorteile durch schwach konsistente Replikate	232
7	Anpassung an das relationale Datenmodell	235
7.1	Probleme	236
7.2	Spezifikation von Allokationsbereichen und Konsistenzanforderungen ..	239
7.3	Definition disjunkter Verwaltungseinheiten	243
7.3.1	Grundlagen der Fragmentierung	244
7.3.2	Das Problem der Objektmigration	247
7.3.3	Abbildung von Extents auf Fragmente	247
7.3.4	Handhabung von Objektmigration auf Fragmentenebene	250
7.3.5	Optimierung des Fragmentierungsschemas	252
7.3.6	Ergänzungen zum Fragmentierungskonzept	254
7.4	Kompromißlösungen.....	254

8	Implementierungsaspekte	257
8.1	Konzeptionelle Überlegungen zur Systemarchitektur	257
8.1.1	Anforderungen	258
8.1.2	Aufbau einer DBVS-Instanz	259
8.1.3	Konzeptionelle Erweiterungen	266
8.2	Prototypische Implementierung	267
8.2.1	Zusatzebenenarchitektur	268
8.2.2	Integration von Datenbankanwendungen	269
8.2.3	Systemarchitektur von ATLAS	270
8.2.4	Prozesse in ATLAS	274
8.2.5	Kommunikation zwischen Prozessen	277
8.2.6	Unterstützte Konsistenzanforderungen	278
9	Zusammenfassung	280
	Literaturverzeichnis	285
	Stichwortverzeichnis	294