

Inhaltsverzeichnis

DB2-Theorie und Praxis - Band 1 Themenbereiche	1
Grafischer Wegweiser durch den Band 1	1
Einstiegs-Grafik in die Thematik für Band 1	2
1 Relationales Modell	3
1.1 Grafischer Wegweiser durch das Kapitel 1 - Relationales Modell	3
1.2 Allgemeine Entwicklungsgeschichte	4
1.2.1 Entwicklung innerhalb IBM	4
1.2.2 Entwicklung außerhalb IBM	4
1.2.3 Produkt-Kompatibilität	5
Kompatibilität zwischen den IBM-DB2-Produkten	5
Kompatibilität zwischen Fremdhersteller-Produkten und DB2-Produkten	5
1.2.4 Entwicklungsgeschichte von DB2 für MVS, OS/390 und z/OS	6
Grober Überblick der DB2-Versions- und Release-Entwicklungen	6
Die wesentlichen Komponenten der Release-Entwicklungen	7
Version 2 - Release 1	7
Version 2 - Release 2	7
Version 2 - Release 3	8
Version 3	9
Version 4	10
Version 5 - DB2 for OS/390	11
Version 6 - DB2 Universal Database (UDB) for OS/390	12
Version 6 nach GA - nach der generellen Verfügbarkeit von V6	14
Version 7 - DB2 Universal Database (UDB) for OS/390 and z/OS	15
Version 7 nach GA - nach der generellen Verfügbarkeit	17
Version 8 - DB2 Universal Database (UDB) for z/OS	18
Version 8 nach GA - nach der generellen Verfügbarkeit	22
Version 9 - DB2 Version 9.1 for z/OS	23
1.3 Terminologie	28
1.3.1 Informationsablage in Tabellenform	28
Tabelle (Table, Relation)	28
Spalte/Column (Attribut)	28
Zeile/Row (Tupel)	29
Datenwert/Value	29
Primary-Key (PK)	30
Foreign-Key (FK) und referenzielle Integrität (RI)	30
Abbildung der Primary- und Foreign-Key-Beziehungen (Beispiel-Tabellen des Buches und ihre logischen Beziehungen)	31
Informationelle referenzielle Integrität (Informational RI)	32
Index	33
Daten-Repräsentation und -Ablage	34
1.3.4 Informationsbeziehungen durch Wertehalte	34
1.3.5 Relationale Sprachschnittstelle	35
SQL	37
QUEL	38
QBE	39
1.4 Charakteristiken eines relationalen DBMS	40

1.4.1 Anforderungen an ein relationales DBMS	40
Codd'sches Anforderungsprofil - RM/V2	40
1.4.2 Zusammenfassung wesentlicher Zielsetzungen, Leistungen und Schwächen des Relationen-Modells	50
Zielsetzungen des Relationen-Modells	50
Standards im Bereich des Relationen-Modells	50
Schwächen des Relationen-Modells	51
1.5 Vergleich der Datenhaltungs- und Bereitstellungs-Konzepte	52
1.5.1 Gruppierung konventioneller Datenhaltungs- und Bereitstellungs- und DBMS-Typen	52
1.5.2 Terminologievergleich konventionelle Systeme - RDBMS	52
1.5.3 Problembereiche konventioneller Datenhaltungs- und DBMS-Typen	53
Strukturierte DBMS	53
Lineare DBMS (relational-orientierte Systeme)	54
1.5.5 Komplexe DBMS: Objektorientierte Datenbank-Systeme (OODBMS)	56
1.6 Speicherung und Verarbeitung von komplexen Objekten (LOBs / Large Objects)	59
1.6.1 Die Abbildung der Realität in der IT durch strukturierte und unstrukturierte Modelle	59
1.6.2 Ein komplexes Objekt ist immer zerlegbar (zumindest theoretisch - praktisch manchmal nicht empfehlenswert)	60
Der Einfluss der Zeit auf komplexe Objekte	61
1.6.3 Kategorisierung komplexer Objekte	62
1.6.4 Speicherung komplexer Objekte mit DB2: Large Objects (LOBs)	63
1.6.5 Problematiken beim Design und der Speicherung komplexer Objekte	64
1.7 Speicherung und Verarbeitung von XML-Dokumenten	65
1.7.1 XML-Terminologie	65
1.7.2 Ein XML-Dokument unterliegt einem strukturierten hierarchischen Modell	67
1.7.3 Zielsetzung und Aufgabenstellung von XML	68
1.7.4 Sind die Ziele des Relationen-Modells mit XML vergleichbar?	69
Eine völlig unterschiedliche Herangehensweise	69
Unterschiedliche Eignungen	70
1.8 Fazit	71
1.9 Standardisierung von Datenbank-Methoden und Techniken	73
1.9.1 Standardisierungs-Organisationen	73
1.9.2 Wichtige allgemeine Standards	73
Datenbank-Standards	74
ANSI/SPARC Architektur oder auch Drei-Schema-Architektur	75
SQL-Standards	76
SQL-Standard: SQL 89	78
SQL-Standard: SQL92 (SQL2)	78
SQL-Standard: 1999/SQL3 (grober Auszug)	80
SQL-Standard: 2003 (relevanter Auszug)	81
SQL-Standard: 2006 (relevanter Auszug)	83
SQL-Standard: 2008 (relevanter Auszug)	84
Die Auswirkungen der bisherigen SQL-Standardisierungsmaßnahmen	85
2 DB2-System-Aufbau	86
2.1 Grafischer Wegweiser durch das Kapitel 2 - DB2-System-Aufbau	86
2.2 DB2-System-Komponenten und Zusatz-Produkte	87
2.2.1 DB2-System-Komponenten	87
2.2.2 DB2-Zusatz-Produkte von IBM	88

Zusatz-Features	88
Objekt- bzw. Daten-Management-Tools	89
2.2.3 Sonstige Tools diverser Software-Hersteller	90
Entwicklungswerkzeuge für die PAEs	90
Verwaltungswerkzeuge für die DBADMs	90
2.3 Betriebssystem-Komponenten	92
2.3.1 System-Schichten von z/OS	92
2.3.2 DB2-Subsystem (Lokation oder Server)	93
2.3.3 DB2-Adressraum-Konzept	95
Sonstige Betriebssystem-Komponenten	96
Trägersysteme (Allied Agents)	96
DB2-System-Adressräume	97
System-Dienste (System Services - MSTR = Master)	97
Datenbank-Dienste (Database Services - DBM1 = Database Manager)	97
Internal Resource-Lock-Manager (IRLM)	97
Administrativer Task Scheduler (ADMT)	97
Distributed Data Facilities (DDF)	98
Adressräume für Routinen (Stored Procedures und/oder User-defined Functions)	99
2.4 Allied Agents: Ankoppelbare Trägersysteme	100
2.4.1 Attachment-Facility, Connection und Thread	100
2.4.2 Zugriff auf DB2-Ressourcen aus Anwendungen heraus	101
Schnittstellen zu den Trägersystem-Typen	101
TSO-Schnittstelle (DSNELI)	101
CICS-Schnittstelle (DSNCLI)	101
IMS-Schnittstelle (DFSLI000)	101
Batch-Schnittstelle mit CAF (DSNALI)	101
Recoverable Resource Manager-Schnittstelle (DSNRLI)	102
Dateiverarbeitungsmöglichkeiten der Trägersystem-Typen	102
2.5 Verteilung (distributing) und gemeinsame Nutzung (sharing) der DB2-Daten	103
2.5.1 Verteilung der DB2-Daten (distributed data)	103
Die Client-Server-Architektur (C/S)	104
DRDA - Distributed Relational Database Architecture	105
DRDA - Level	106
Remote-Zugriff über dreiteilige Objekt-Namen/ALIAS oder nach CONNECT	107
DB2-Privat-Protokoll und DRDA-Protokoll	108
Technologische Grenzen der Verteilung von Daten	108
2.5.2 Gemeinsame Nutzung der DB2-Daten (DS - Data Sharing)	109
Betrieb mehrerer z/OS-Systeme	109
Data-Sharing: Group und Member	109
Argumente für Data Sharing	110
Grundsätzliche Anforderungen an ein Data-Sharing-Verfahren	111
Technische Lösung des DB2 Data Sharings	112
Data Sharing benötigt einen Parallel Sysplex	112
Aufgaben-Abgrenzung zwischen Mitgliedern und dem Coupling Facility	113
Speicher-Strukturen einer Data Sharing Group	114
Die Konsistenzbewahrung bei Data Sharing (DS)	115
Sicher is sicher: Struktur-Duplizierung und Duplexed GBP	116
Einflüsse einer Data Sharing Group auf die System-Administration	117

Objekte und Namenskonventionen einer Data Sharing Group	118
Gruppen-Objekte und ihre Namen	118
Member-Objekte und ihre Namen	119
2.6 DB2-Ressource-Management durch den DB2 Administrativen Task Scheduler	120
2.6.1 Aufgabenstellung	120
2.6.2 Komponenten des Task Schedulers	120
System-Architektur	120
ADMT-DB2-System-Komponenten	121
Planbare Ressourcen	122
Architektur in einer Data Sharing Group	123
2.7 Synchronisation mehrerer Ressource-Manager: RRS - Resource Recovery Services	124
2.7.1 RRS-Aufgabenstellung	124
2.7.2 Programm-Typen für die RRS-Synchronisations-Aufgaben	125
2.8 DB2-Daten-Objekt-Typen	126
2.8.1 Meta-Objekt-Typen für die Speicherung von Daten	126
Verwaltung der Meta-Daten	127
Anlegen, Änderung und Löschung von Meta-Daten	127
CREATE - Anlegen von Meta-Daten	127
ALTER - Änderung von Meta-Daten	128
Online-Schema-Evolution: Realtime-Änderung	128
DROP - Löschung von Meta-Daten	128
Schema, Eigentümer und Ersteller von Daten-Objekt-Typen	129
Autorisierungs-Id, Schema, Ersteller (creator) und Eigentümer (owner)	129
Das Zusammenspiel der Spezialregister	130
Versionsführung für Meta-Objekte (Versionierung)	131
Wie kann man sich eine Versionsführung vorstellen?	131
Wie muss man sich die DB2-Versionenführung vorstellen?	132
Relevante Katalog-Informationen der Versionsführung	133
Test-Beispiele der DB2-Versionenführung	134
Im DB2 existieren vielfältige Versionen und Versionierungen	135
Domain-Konzept und Daten-Typen	136
Grundsätzliche Überlegungen und Ausgangssituation	136
Builtin Daten-Typen	136
LOB = Large Object	136
XML = XML-Dokument	137
Benutzerdefinierte Daten-Typen (UDT = User-defined Data-Types)	138
Definition der Daten-Typen	138
Automatisch generierte Cast-Funktionen	139
Nutzungsmöglichkeit der Distinct Data-Types	140
User-defined LOB-Data Types	140
User-defined XML-Data Types	140
Vergleichbarkeit und Konvertierbarkeit der Daten-Typen	141
Benutzer-orientierte Daten-Objekt-Typen	142
Tabelle/Table (Base Table)	142
Definition der Tabellen-Spalten	142
Die Rolle des PKs	144
Einrichtung referenzieller Beziehungen (referential constraint)	144
Einrichtung von Check-Constraints	146
Maschinelle Nummern-Vergabe: ROWID, IDENTITY, SEQUENCE	146
GENERATED-Klausel (GENERATED ALWAYS und GENERATED BY DEFAULT)	147
Zuordnung einer Tabelle zu einem Tablespace	149
Definition einer Partitioned Table (table-kontrollierte Partitionierung)	150

Änderung von Tabellen- und Spalten-Charakteristiken	152
Änderung des Daten-Typs einer Spalte	153
SQL-DDL-Unterstützung für die Basis-Tabelle	157
Clone Table	158
Aufgabenstellung und Funktionsumfang	158
Metadaten und Daten-Instanzen von Base Table und Clone Table	159
Metadaten von Base Table und Clone Table (Katalog-informationen)	159
Daten-Instanzen von Base Table und Clone Table	160
Daten-Definition von Base Table und Clone Table mittels SQL-DDL	161
Daten-Manipulationen und Utility-Untersützung einer Clone Table	162
EXCHANGE-Statement: Austausch der Daten-Instanzen zwischen Base Table und Clone Table	163
Löschen der Clone Table	163
Restriktionen der Base Table und der Clone Table	164
Fazit	164
Temporäre Tabelle	165
CTT - Created Global Temporary Table	166
DTT - Declared Global Temporary Table	166
Temporäre Result Table/Virtual Table	169
Temporäre Result-Table-Typen	169
Positionierung oder Materialisierung einer temporären Result Table	171
DB2-View (Virtual Table)	172
Ist die View-Result-Table read-only oder updateable?	174
View auf View - macht das Sinn?	176
Anwendungs-Unabhängigkeit von physischen Sichten	177
Individuelle Views oder generalisierte Views	178
Nachteile und Grenzen des View-Konzepts	179
Empfehlungen zum View-Konzept	179
MQT - Materialisierte Query Table (Materialisierter View bzw. Materialisierte Result Table)	180
Komponenten einer materialisierten Query Table (MQT)	181
Definition der MQT	182
System- oder benutzer-kontrollierte MQT	184
SYNONYM	185
ALIAS	186
Machinelle Vergabe einer lfd. Nr. durch DB2: SEQUENCE und IDENTITY	187
SEQUENCE	189
Systemorientierte DB2-Objekt-Typen	190
Index	190
Definition der Index-Spalten	191
Wann wird ein Index aufgebaut und wie wird er verwaltet?	191
Index-Typen	192
Basis-Index	194
Ein stinknormaler Index besteht aus	194
Der Clustering Index organisiert die Datenfolge	195
Partitioning Index und Partitioned Index	197
und nun alle Index-Typen auf einmal	200
DPSI: Data Partitioned Secondary Index	202
Extended Index	205
Scalar Expression Index (Index on Expressions)	206
Interne Organisation eines Indexes	208
Zuordnung eines Indexes zum Indexspace, den Page Sets und Datasets	208
B-Tree-Struktur eines Indexes	209
Änderung von Index-Charakteristiken	211
Löschung eines Indexes	211
Welche Vorteile bietet ein Index, welche Nachteile sind zu beachten?	212
Vorteile eines Indexes	212
Nachteile eines Indexes	213

Empfehlungen zum Index-Einsatz	214
Tablespace	215
Aufgabenstellung und relevante Zuordnungen	215
Die Page-Größe und ihre Einflüsse	217
Tablespace-Typen	218
Vor- und Nachteile der Tablespace-Typen	221
Besonderheiten des Partitioned Tablespaces	223
Physische Charakteristiken pro Partition	223
4096 Partitions und die Konsequenzen	224
Dynamische Partitions: Hinzufügen neue Partition	226
Dynamische Partitions: Rotation	228
Dynamische Partitions: Partition-Werte balancieren	230
Nachträgliche Änderungsmöglichkeiten eines Tablespaces	231
Indexspace	232
Aufgabenstellung und relevante Zuordnungen	232
Database	234
Storage Group	235
LOB-Objekte: Auxiliary Table, Auxiliary Index, LOB-Tablespace	236
XML-Objekte: XML Table, DocID- und NodeID-Index, XML-Index und XML-Tablespace	238
2.8.2 DB2-System-Ressourcen	240
DB2-Directory: Database DSNDB01	241
DB2-Katalog: Database DSNDB06	242
Datenspeicherung von System- und Benutzerdaten	242
Zwischenspeicher für Materialisierungen	243
Workfiles: Workfile Database (DSNDB07)	243
Utility-Arbeitsdateien	243
DB2-Log-Dateien	244
Bootstrap-Dateien	244
DB2-Bibliotheken	244
2.9 Routinen: Stored Procedures und User-defined Functions	245
2.9.1 Grafischer Wegweiser durch das Kapitel 2 - Routinen: Stored Procedures und User-defined Functions und Trigger	245
2.9.2 Terminologie und Übersicht	246
Routinen-Typen	246
Aufgabenbereiche und Komponenten von Routinen	247
Unterschiede zwischen Externen Routinen und SQL Routinen	248
2.9.3 Identifikation von Stored Procedures und Functions	249
Schema, SQL-Pfad, Specific Name und External Name	249
Function Overloading und Function Resolution	249
2.9.4 Der Einsatz von Stored Procedures	250
Grobe Zielsetzung einer Stored Procedure	250
Prozedur-Typen	251
Entwicklung einer Stored Procedure	252
Definition einer Stored Procedure	254
Namen der Stored Procedure	254
Parameter einer Stored Procedure	254
Übergabe der Parameterwerte	254
Parameter-Typen (Eingabe, Ausgabe oder beides)	256
NULL-Fähigkeit der Parameter	256
PARAMETER STYLE GENERAL, GENERAL WITH NULLS und SQL (Übergabe-Konventionen einer Stored Procedure)	257
Beispiel der Parameterübergabe zwischen PL/I- und COBOL	259
PARAMETER STYLE JAVA	260

Rückgabe einer Ergebnismenge (Query Result Set)	261
CALL-Aufruf-Varianten einer Stored Procedure	262
Statische CALL-Varianten	262
Dynamischer CALL (ODBC bzw. CLI)	262
External Procedure	263
Unterstützte Programmiersprachen	263
Entwicklungsschritte einer External Stored Procedure	263
Aufrufendes und aufgerufenes Modul	263
Compile und Linkage Editor	263
Definition der Stored Procedure im Katalog	264
SQL Procedure	265
Typen und unterstützte Funktionalitäten	265
SQL External Procedure	266
Unterstützte Programmiersprachen und Sprachmittel	266
Entwicklungsschritte einer SQL External Procedure	267
Data Studio oder Development Center	268
IBM-Stored Procedure DSNTPSMP	268
Explizite Definition mit der JCL-Prozedur DSNHSQL	269
Definition der SQL Stored Procedure im Katalog	269
SQL Native Procedure	270
Unterstützte Programmiersprachen und Sprachmittel	270
Versions-Konzept einer SQL External Procedure	271
Entwicklungsschritte einer SQL External Procedure	272
Definition der SQL Stored Procedure im Katalog	272
Ausführung einer Stored Procedure	273
Adressraum-Konzept	273
Starten und Stoppen von Stored Procedures	273
Aufruf einer remote Stored Procedure und Ablauf	274
Vor- und Nachteile von Stored Procedures	276
Liste der Standard-Stored Procedures (IBM-Supplied Stored Procedures)	277
Relative Performance-Gewichtung von Stored Procedures mit unterschiedlichen Programmiersprachen	280
2.9.5 Der Einsatz von benutzerdefinierten Funktionen (User-defined Functions)	281
Grobe Zielsetzung einer User-defined Function (UDF)	281
Funktions-Aufruf und -Ergebnis	281
Funktions-Typen	282
Entwicklung einer User-defined Function	284
Definition einer User-defined Function	286
Namen der User-defined Function	286
Parameter einer User-defined Function	286
Übergabe der Parameterwerte	286
Parameter-Typen (Eingabe oder Ausgabe)	287
NULL-Fähigkeit der Parameter	287
PARAMETER STYLE SQL oder JAVA (Übergabe-Konventionen für Funktionen)	288
Beispiel der Parameterübergabe zwischen COBOL und PL/I	290
Aufruf-Varianten einer User-defined Function	291
Externe Aufruf-Varianten der Funktion	291
Interne Aufruf-Typen (CALL-Typ bzw. FINAL CALL-Klausel)	291
Scratchpad: Zwischenspeicher für die internen Aufrufe	294
Thread- und TCB-Konzept für Funktions-Aufrufe	295
TCB-Nutzung beim Funktions-Aufruf	295
TCB-Konzept bei Parallel-Abwicklung	296
DETERMINISTIC oder NOT DETERMINISTIC - das ist hier die Frage	297
Sourced Function	298
Unterstützte Programmiersprachen	298
Entwicklungsschritte einer Sourced Function	298

Definition der Sourced Function im Katalog	298
External Functions	299
Unterstützte Programmiersprachen	299
Entwicklungsschritte einer External Function	299
Aufrufendes und aufgerufenes Funktions-Modul	299
Compile und Linkage Editor	299
External Scalar Function	300
Definition der Scalar Function im Katalog	300
External Table Function	301
Definition der Table Function im Katalog	301
SQL Scalar Function	302
Definition der SQL Scalar Function im Katalog	302
Ausführung einer User-defined Function	303
Adressraum-Konzept	303
Starten und Stoppen von User-defined Functions	303
Vor- und Nachteile von User-defined Functions	304
Vor- und Nachteile von External Scalar Functions	304
Vor- und Nachteile von SQL Scalar Functions	304
Vor- und Nachteile von Table Functions	305
2.9.6 Mögliche Ressource-Zugriffe	306
Unterstützte SQL-Funktionalität	306
Unterstützung von LOB-Spalten und -Werten	306
Unterstützung von XML-Spalten und -Werten	306
Zugriff auf externe Ressourcen	306
Remote DB2-Objekte	306
Andere relationale Datenbanken	306
CICS-Ressourcen	307
IMS-Ressourcen	307
WebSphere MQ-Ressourcen	307
Sonstige externe Ressourcen	307
2.9.7 Level-Konzept (Nested Routines)	308
Behandlung von Spezialregistern in Routinen	309
Fehlerbehandlung von Routinen	310
Auswahl relevanter Fehlercodes	311
Bewahrung der Daten-Integrität (COMMIT und ROLLBACK)	312
2.9.8 Erforderliche Privilegien	313
2.9.9 Package- und Plan-Behandlung	314
Behandlung der externen Module	314
Wechsel der Package	314
Nutzung verschiedener Programm-Versionen	314
2.9.10 Testmöglichkeiten von SQL-Routinen	315
Allgemeine Problematik	315
Test-Tools (Debugger)	315
Besonderheiten beim Test	315
Routinen-Versionen einer SQL Native Procedure	315
2.9.11 Zusammenfassung der relevanten Klauseln für die Definition von Routinen	316
Abwicklungs-Komponenten und Zuordnung der DDL-Parameter-Wirkung	316
Ausführungs-Steuerung (Optionen) von Routinen	317
Übersicht der Options-Argumente und Zuordnungen zu den Routinen-Typen	317

Besonderheiten und Konventionen für Java Applikationen	318
2.9.12 z/OS-Ausführungs-Umgebung	319
Überblick der beteiligten z/OS-System-Komponenten	319
RRS - Resource Recovery Services	320
Die Rolle im Rahmen von Routinen	320
RRSAF-Attachment: Statisches oder dynamisches Laden	321
Language Environment	322
LE-Konzept und relevante Komponenten	322
LE-Run-time-Options	322
MSGFILE	322
RPTOPTS	323
RPTSTG	324
Optionen zur Kontrolle des Speichers	325
TEST und NOTEST	325
Sonstige LE-Besonderheiten	325
Workload-managed Adressräume (WLM-established)	326
WLM-Terminologie	326
WLM-Aufgabenstellung	326
WLM-Komponenten	326
WLM-Subsystem-Typen	326
WLM-Service-Definitions	327
WLM-Operations-Modus	327
WLM-DB2-Environment	328
DB2-Installations-Parameter für WLM	328
Parametrisierung für WLM	328
WLM-JCL-Beispiel	328
NUMTCB = Anzahl paralleler Tasks	329
Steuerung des WLM-Adressraums	330
Kontrolle und Abbruch aktiver Routinen	330
Wieviele WLM-Environments sind sinnvoll?	331
Privilegien für die Nutzung des WLM-Environments	331
Modul-Eigenschaften in der WLM-Umgebung	332
Threadsafe Module (reentrant)	332
Residente Module	332
Haupt- oder Unter-Routine	332
Austausch der ausführbaren Version einer Routine	333
Testmöglichkeiten in der WLM-Umgebung	334
Allgemeine Problematik	334
Test-Tools (Debugger)	334
Besonderheiten beim Test	334
Vorteile des WLM-Environments	335
Nachteile des WLM-Environments	335
Performance-Aspekte bei externen Modulen	336
2.10 Der Einsatz von Triggern	337
2.10.1 Zielsetzungen eines Triggers	337
2.10.2 Trigger-Terminologie	338
Beispiel eines Triggers	338
Trigger-Charakteristiken	338
2.10.3 Übersicht der Trigger-Typen und ihre Charakteristiken	340
Komponenten eines Triggers	341
Aktivierungszeit: BEFORE	342

Allgemeine Regeln	342
Trigger-Beispiele: Before-Trigger	343
Abbildung des Einsatzes	343
Trigger mit Abweisung ungültiger Referenzen (RI-Erweiterung)	344
Trigger zur Einsetzung von Initialwerten bei Einfügungen	345
Trigger mit Übernahme redundanter Daten	345
Trigger mit Aufruf einer Funktion	345
Aktivierungszeit: AFTER	346
Allgemeine Regeln	346
Trigger-Beispiele: After-Trigger	347
Abbildung des Einsatzes	347
Verwaltung von Redundanzen	348
Trigger mit Übergabe einer Transition-Table an eine Funktion	349
Trigger mit Übergabe von Transition-Variablen an eine Funktion	349
Aktivierungszeit: INSTEAD OF	350
Allgemeine Regeln	350
Trigger-Beispiele: Instead of-Trigger	351
Abbildung des Einsatzes	351
Restriktionen eines Instead of Triggers	353
Empfehlungen für den Einsatz eines Instead of Triggers	354
Ereignis-Typen: INSERT, UPDATE und DELETE	355
Häufigkeit der Funktionsauslösung: ROW oder STATEMENT	355
2.10.4 Ablauf der Verarbeitung bei Einsatz eines Triggers	356
Trigger-Aktion	357
Trigger-Condition	357
Trigger-Body	357
Result Table des Triggers	358
Transition-Variablen	358
Transition-Tables	359
Cascading-Effekte und Trigger-Level bei After Triggern	360
2.10.5 Fehler-Behandlung	361
Fehler-Erkennung und -Signalisierung	361
Fehler bei der Ausführung der Trigger-Statements	362
2.10.6 Aufruf von Routinen aus einem Trigger heraus	363
Aufruf-Techniken	363
Behandlung von Spezialregistern	364
2.10.7 Trigger-Package	365
Erzeugen und Löschung einer Trigger-Package	365
REBIND einer Trigger-Package	365
Abhängigkeiten einer Trigger-Package	365
2.10.8 Trigger-Restriktionen	366
2.10.9 Vor- und Nachteile von Triggern	367
3 Methodische und technische Voraussetzungen für DB2	369
3.1 Grafischer Wegweiser durch das Kapitel 3 - Methodische und technische Voraussetzungen für DB2	369
3.2 Allgemeine Voraussetzungen für eine DB2-Einführung	370
3.2.1 Personelle Voraussetzungen	371
Objekt- (OA) bzw. Daten-Administration (DA)	371
Datenbank-Administration (DBA)	371

3.2.2 Organisatorische Voraussetzungen	372
Auswirkung auf die Anwendungs-Entwicklung	372
3.2.3 Technische Voraussetzungen für DB2 Version 9 for z/OS	374
Hardware	374
Betriebssystem-Software	375
Funktionsspezifische Abhängigkeiten	375
Betriebssystemnahe Standard-Software	375
Transaktions-Management	376
DRDA-Verbindungsmöglichkeiten	376
Web-Verbindungen	376
Anwendungs-Entwicklungs-Tools	377
Programmiersprachen	377
Data Warehouse-Unterstützung	377
Migrations-Pfad zur Version 9	378
3.2.4 Was ist neu im DB2?	379
3.2.5 Migration-Strategie ins DB2	381
3.3 DB2-Systemkonzept	382
3.3.1 Single-DBMS oder Multi-DBMS?	382
3.3.2 Abgrenzung Produktionssystem -Testsystem	383
System-Kategorien: Test, Übergabe und Produktion	383
System-Kategorie: Test	384
System-Kategorie: Übergabe in die Produktion (System-Test)	384
System-Kategorie: Produktion	385
Problematiken bei der Übernahme von Test in die Produktion	385
Isolierte Systeme haben trotzdem Kommunikations-Erfordernisse	385
Koordination der Objekte zwischen Systemen	386
Management der Versions-Änderungen (Change Management)	386
Spezielle DB2-Übernahme-Kriterien	386
3.3.3 Datenbank-Hierarchie und Nutzungsmöglichkeiten	387
3.3.4 Informations-Aktualität	388
3.3.5 Client-Server-Konzept (C/S)	389
Zentralisierung versus Dezentralisierung	389
C/S sollte in den Köpfen des Managements beginnen ...	389
C/S benötigt klare Konzepte	390
Argumente für eine dezentrale Organisationsform (Verteilung)	391
Argumente für eine zentrale Organisationsform	391
Fazit	391
Varianten der C/S-Verarbeitung	392
Verteilungsaspekte	393
Lokale Zuordnung der Ressourcen	393
Verfügbarkeit und Nutzbarkeit der Ressourcen	393
Konsequenzen einer Verteilung	394
3.4 Methodische Voraussetzungen	395
3.4.1 Abhängigkeiten zwischen Programmen und Daten	395
Probleme beim Einsatz konventioneller Datenbank-Systeme	395
Sind solche Probleme unter DB2 automatisch gelöst?	397
Abhängigkeit des SQL-Spracheinsatzes von DB2-Katalog-Informationen	397

Abhängigkeit bei fehlendem Schichten-Konzept	400
Lass die Datenbank loopen!	401
Auswirkung der Daten-Modellierung auf die Programm-Sicht	402
Sicht auf unnormalisierte Daten	403
Definition einzelner Spalten in Wiederhol-Elementen	403
Variable Zeilen mit VARCHAR- oder LOB-Definitionen	404
Speicherung von XML-Dokumenten	405
Sicht auf normalisierte Daten	406
Sicht auf Daten in der ersten Normalform (1 NF)	406
Sicht auf Daten in höheren Normalformen (2 NF, 3NF)	407
3.4.2 Abbau der Abhängigkeiten durch Methodik-Einsatz	408
Ebenen, Modell- und Ergebnis-Typen im Unternehmens-Modell	408
Auswirkungen auf das Anwendungs-Design	409
Schichten-Modell für Anwendungs-Programme	409
Separierung der Datenzugriffe von sonstigen Funktionen	410
Objektorientierte Verwaltung der Daten	411
Daten-Verwaltung immer über eine zuständige Funktion/Methode	412
Einsatzmöglichkeit von Standard-Zugriffs-Routinen	413
3.4.3 Abbau der Abhängigkeiten durch DB2-Techniken	414
Das Ebenen-Modell von DB2	414
Katalog-Konzept	415
Verlagerung von Programmlogik in den DB2-Katalog	415
Komponenten einer "aktiven Datenbank"	416
Nutzung von Stored Procedures, User-defined Functions und Triggern	417
Integritätsbewahrung als Aufgabenstellung für die Datenbank	417
Verlagerung sämtlicher datennaher Dienste in den DB2-Bereich	418
Argumente für und wider eine "aktive Datenbank"	419
3.4.4 Normungs-Effekte durch internationale Standards	420
Normungs-Effekt durch den SQL-Spracheinsatz	420
Open Database Connectivity (ODBC) und der CLI-Spracheinsatz	420
4 DB2-Sprachschnittstellen	421
4.1 Grafischer Wegweiser durch das Kapitel 4 - DB2-Sprachschnittstellen	421
4.2 Benutzergruppen und deren Anforderungen	422
4.3 DB2-Sprachschnittstellen für die Benutzergruppen	423
4.3.1 TSO-DSN-Commands: Übersicht	425
4.3.2 SQL-Statements	426
SQL-Statement- und Benutzer-Kategorien	426
DCL - Data Control Language	426
DDL - Data Definition Language	426
DML - Data Manipulation Language	427
SQL-Anwendungsformen und Sprach-Schnittstellen	429
Interactive SQL (DB2I)	429
Embedded SQL (EXEC-Level)	429
CALL-Level-Interface für C- und C++-Programme (CLI bzw. ODBC)	430
JDBC-API für Java-Anwendungen	430
SQLJ-API für Java-Anwendungen	430
Static SQL	431
Dynamic SQL	431
Übersicht der SQL-Statements und ihre Einsatzmöglichkeiten	432

4.3.3	DB2-Befehle/DB2-Commands	436
	DB2-System-Commands	436
	Admin-Scheduler-Commands	438
	CICS-Commands	438
	IMS-Commands	439
	IRLM-Commands	439
4.3.4	DB2-Hilfsprogramme/DB2-Utilities	440
	DB2-Online-Utilities	440
	Utility-Control-Statements	441
4.3.5	Standalone Utilities/Service Aids	441
4.3.6	TSO-CLISTS und TSO-DSN-Subcommands	442
4.4	DB2I: Masken der interaktiven DB2-Oberfläche	443
4.4.1	Einstiegs-Menü (Primary Option Menu)	443
4.4.2	D - DB2I-Defaults	444
	Default-Panel 1	444
	Default-Panel 2	445
4.4.3	1 - SPUFI: SQL Processor Using File Input	446
	SPUFI-Steuerungsmaske	446
	SPUFI-Defaults	448
	Default-Panel 1	448
	Default-Panel 2	450
	SPUFI-Edit-Maske	451
	SPUFI-Browse-Maske	452
4.4.4	2 - DCLGEN: Declarations Generator	453
	Eingabemaske	453
	Maske mit Zusatz-Optionen	454
	Beispiel generierter Ausgaben	455
	COBOL-Host-Variablen-Struktur	455
	SEMINAR-Tabelle	455
	Variante: Default-Struktur- und Spalten-Namen ohne Label	455
	Variante: Default-Struktur- und Spalten-Namen mit Label	456
	Variante: Individuelle-Struktur- und Spalten-Namen (Table-Name mit Numerierung)	457
	Variante: Individuelle-Struktur- und konkreten Spalten-Namen	458
	Alle Daten-Typen	459
	PL/I--Host-Variablen-Struktur	461
	SEMINAR-Tabelle	461
	Variante: Default-Struktur- und Spalten-Namen ohne Label	461
	Variante: Default-Struktur- und Spalten-Namen mit Label	462
	Variante: Individuelle-Struktur- und Spalten-Namen (Table-Name mit Numerierung)	463
	Variante: Individuelle-Struktur- und konkreten Spalten-Namen	464
	Alle Daten-Typen	465
	C--Programm-Host-Variablen-Struktur	467
	SEMINAR-Tabelle	467
	Variante: Default-Struktur- und Spalten-Namen ohne Label	467
	Variante: Default-Struktur- und Spalten-Namen mit Label	468
	Variante: Individuelle-Struktur- und Spalten-Namen (Table-Name mit Numerierung)	469
	Variante: Individuelle-Struktur- und konkreten Spalten-Namen	470
	Alle Daten-Typen	471
4.4.5	3 - Program Preparation	473
	Steuerungsmaske	473

Program Preparation 'Compile, Link and Run Panel'	474
4.4.6 4 - Precompile	475
4.4.7 5 - BIND / REBIND / FREE	476
1 - BIND PLAN	477
Steuerungsmaske	477
Default Panel BIND PLAN	478
Connection Types Panel	479
Package List Panel	480
Ergebnis einer erfolgreichen BIND PLAN-Ausführung	480
2 - REBIND PLAN	481
Steuerungsmaske	481
Default-Panel	482
3 - FREE PLAN	483
Steuerungsmaske	483
4 - BIND PACKAGE	484
Steuerungsmaske	484
Default-Panel	486
Connection Types Panel	487
Path List Panel	488
Ergebnis einer erfolgreichen BIND PACKAGE-Ausführung	488
5 - REBIND PACKAGE	489
Steuerungsmaske	489
Default-Panel	491
6 - REBIND TRIGGER PACKAGE	492
Steuerungsmaske	492
7 - FREE PACKAGE	493
Steuerungsmaske	493
4.4.8 6 - RUN - Aktivierung einer Programm-Ausführung im Foreground oder Batch	494
Anstoß über Panel	494
JCL-Beispiel eines Batch-Ausführungs-Jobs	495
4.4.9 7 - DB2 Commands	496
Vorgabe von DB2-Commands	496
Ergebnis einer DB2-Command-Ausführung	496
4.4.10 8 - Utilities (Online-Utilities)	497
Steuerungsmaske	497
Control Statement Data Set Names	498
Data Set Names	499
Ergebnisse der Job-Generierung	500
Generierungs-Informationen	500
Generierter Utility-Job	501
5 SQL-DDL-Data Definition Language	502
5.1 Grafischer Wegweiser durch das Kapitel 5 - SQL-DDL-Data Definition Language	502
5.2 Dynamische DB2-Objektverwaltung	503
5.3 Anlegen DB2-Objekt-Typen: CREATE	505
5.3.1 Defaults beim Anlegen von Objekten	505
5.3.2 Beispiele CREATE	508
Systemorientierte Daten-Objekt-Typen	508

STOGROUP, DATABASE und TABLESPACE	508
INDEX	510
Base-Index-Varianten	510
Auxiliary-Index, Extended Index und XML-Index	511
Benutzerorientierte Daten-Objekt-Typen	512
Distinct Type	512
Base-Table-Varianten	512
Auxiliary Table	517
XML Table	517
Global Temporary Table	518
MQT - Materialized Query Table	519
Clone Table	520
VIEW	521
SYNONYM und ALIAS	522
COMMENT und LABEL	523
SEQUENCE	524
Funktionsorientierte Objekt-Typen: Routinen	525
UDF - User-defined Function	525
STP - Stored Procedure	527
Funktionsorientierte Objekt-Typen: Trigger	529
Trigger für die Manipulation von Tabellen: Before und After Trigger	529
Trigger für die Manipulation von Views: Instead of Trigger	530
5.4 Ändern DB2-Objekt-Typen: ALTER und RENAME	531
5.4.1 Besonderheiten bei Struktur-Veränderungen und Erweiterungen	532
Wie verhalten sich die Daten bei Struktur-Veränderungen?	532
Wie verhalten sich die Daten bei Struktur-Erweiterungen?	532
Was tun, wenn kein ALTER existiert?	533
5.4.2 Beispiele ALTER	534
Systemorientierte Daten-Objekt-Typen	534
STOGROUP, DATABASE und TABLESPACE	534
INDEX	535
Base-Index	535
Auxiliary-Index, Extended Index und XML-Index	535
Benutzerorientierte Daten-Objekt-Typen	536
Distinct Type	536
Base-Table-Varianten	536
Auxiliary Table	538
XML Table	538
Global Temporary Table	538
MQT - Materialized Query Table	538
Clone Table	539
VIEW	540
SYNONYM und ALIAS	540
COMMENT und LABEL	540
SEQUENCE	541
Funktionsorientierte Objekt-Typen: Routinen	542
UDF - User-defined Function	542
STP - Stored Procedure	543
Funktionsorientierte Objekt-Typen: Trigger	544

5.4.3 Beispiele RENAME	544
5.5 Löschen DB2-Objekt-Typen: DROP	545
5.5.1 Die Auswirkung eines DROP-Kommandos auf andere DB2-Objekte (Abbildung mit referenziellen Effekten)	547
5.5.2 Beispiele DROP	548
Systemorientierte Daten-Objekt-Typen	548
STOGROUP, DATABASE und TABLESPACE	548
INDEX	549
Base-Index	549
Auxiliary-Index	549
Extended Index und XML-Index	549
Benutzerorientierte Daten-Objekt-Typen	550
Distinct Type	550
Base-Table	550
Auxiliary Table	550
XML Table	550
Global Temporary Table	551
MQT - Materialized Query Table	551
Clone Table	552
VIEW	552
SYNONYM und ALIAS	552
COMMENT und LABEL	552
SEQUENCE	552
Funktionsorientierte Objekt-Typen: Routinen	553
UDF - User-defined Function	553
STP - Stored Procedure	553
Funktionsorientierte Objekt-Typen: Trigger	554
Objekt-Typ: Package	554
6 SQL-DML-Data Manipulation Language	555
6.1 Grafischer Wegweiser durch das Kapitel 6 - SQL-DML-Data Manipulation Language	555
6.2 SQL-DML-Sprachspektrum	556
6.3 SQL-DML-Anwendungsformen und Sprach-Schnittstellen	557
6.4 SELECT - Datenabfragen	558
6.4.1 SELECT-Typen	558
Relevante Komponenten eines SELECT-Statements	558
Result Table eines SELECT-Statements	558
Syntaktische Grundformen des SELECT-Statements	559
Entwicklung und Veränderungen der SELECT-Klauseln in DB2-Version 8 und 9	561
6.4.2 Syntax des SELECT-Statements	562
Grundformat des SELECT-Statements	562
Die logische Abarbeitungsfolge der SELECT-Parameter	563
SELECT-Klausel	564
Bereitstellung von Daten aus einer einzelnen Tabelle	564
Verdichtung der Result Table (DISTINCT)	564
Bereitstellung von Daten aus mehreren Tabellen (Join)	564
FROM-Klausel	565
Bereitstellung von Daten aus Basis-Tabellen bzw. Views	565
Inner und Outer Join	565
Nested Table Expression	565

WHERE-Klausel	566
Einfache Vergleichsoperatoren	566
Subquery	566
Row Expression (Row-Value-Expression)	567
ORDER BY-Klausel	567
GROUP BY-Klausel	567
6.4.3 Tabelleninhalte der Beispieltabellen	568
6.4.4 Einfache SELECT-Abfragebeispiele aus einer Tabelle	569
Einfache Spaltenauswahl (PROJECT)	569
Konstante und arithmetisch errechnete Werte	569
Vergabe von Namen für Spalten der Result Table	570
Auswahl bestimmter Zeilen mit WHERE	570
Variable Auswahl bestimmter Zeilen mit WHERE und CASE	571
6.4.5 Boolsche Verknüpfungen und Prädikate	572
AND-Verknüpfung	572
OR-Verknüpfung	572
NOT-Verknüpfung	573
BETWEEN-Prädikat	573
IN-Prädikat	573
LIKE-Prädikat	573
NULL-Prädikat	575
DISTINCT-Prädikat	575
Mix von Prädikaten und Klammerung	576
6.4.6 Sortierung - ORDER BY	577
Auf- und absteigende Sortierung	577
Sortierung von abgeleiteten Spalten	578
Sortierung der Result Table eines Sub-Selects und der endgültigen Result Table	579
6.4.7 Funktion (Builtin-Funktion)	580
Aggregate Function	580
Scalar Function	582
Konvertierungs-Funktion	582
DateTime-Funktion	583
Sonstige Funktionen	585
Verschachtelung von Aggregate Functions und Scalar Functions	586
6.4.8 Gruppierung/ GROUP BY	587
HAVING	589
6.4.9 Set-Operatoren	592
UNION und UNION ALL	594
EXCEPT und EXCEPT ALL	597
INTERSECT und INTERSECT ALL	598
6.4.10 JOIN Selektion aus mehreren Tabellen	599
Join-Verarbeitungs-Regeln	599
Kartesisches Produkt	601
Equi-Join	602
Join einer Tabelle mit sich selbst	603
Inner Join und Outer Join	604
Übersicht der Ergebnis-Typen	604

Inner Join	606
Left Outer Join	606
Right Outer Join	607
Full Outer Join	608
Die Wirkung der ON-Klausel	609
Verknüpfung mehrerer Join-Typen	611
Kombination von Join und Table Expression	613
6.4.11 Scalar Full-Select	614
Möglichkeiten und Risiken	614
Scalar Full-Select in der WHERE-Bedingung	615
Scalar Full-Selects und Joins	616
Scalar Full-Select in der CASE-Expression	617
6.4.12 Table Expression	618
Nested Table Expression	618
Einfaches Beispiel	618
Nested Table Expression mit Outer Join	620
Nested Table Expression mit CASE (Statistik-Funktion)	621
Common Table Expression	622
Einsatzgebiet	622
Einfaches Beispiel	622
Einsatz mehrerer Common Table Expressions	623
Rekursives SQL	626
Die typische Hierarchie-Abbildung	626
Aufbau von Test- oder Hilfsdaten	630
Kalender mit Tagen, Wochen und Wochen-Tag	630
Kalender mit Jahr, Woche und Tag nach ISO (wann beginnt die erste Woche im Jahr?)	631
Common Table Expression und INSERT	632
6.4.13 Subquery (Sub-Query)	633
Allgemeine Syntax	633
Non-correlated Subquery	635
Subquery direkt nach einem Vergleichsoperator	635
Subquery nach IN	636
Subquery nach EXISTS	637
Subquery nach ANY bzw. SOME und ALL	638
Correlated Subquery	639
Verarbeitungsunterschiede zu non-correlated	639
Subquery direkt nach einem Vergleichsoperator	640
Subquery nach EXISTS	640
6.5 SQL-Manipulations-Statements	641
6.5.1 Übersicht der Manipulations-Typen	641
6.5.2 SELECT-Statement mit SQL Data Change Statement	642
6.5.3 INSERT - Daten-Einfügung	643
Der INSERT fordert eine updateable Result Table	643
INSERT-Formate	644
Verhalten bei Spalten mit Default-Maßnahmen	645
Beispiele für SQL-Dateneinfügungen	646
INSERT einer einzelnen Zeile	646
INSERT einer Zeilen-Gruppe (im Anwendungsprogramm)	646
INSERT mehrerer Zeilen mit SELECT	647

SELECT mit SQL Data Change Statement (hier: SELECT mit INSERT)	649
FINAL TABLE mit einer Zeile	650
Beispiel: SELECT FROM FINAL TABLE (INSERT ... mit INCLUDE)	651
FINAL TABLE mit mehr als einer Zeile	653
6.5.4 UPDATE - Daten-Veränderung	655
Der UPDATE fordert eine updateable Result Table	655
UPDATE-Formate	656
Beispiele für SQL-Datenveränderungen	657
Searched Update	657
Positioned Update	658
SELECT mit SQL Data Change Statement (hier: SELECT mit UPDATE)	659
6.5.5 DELETE - Daten-Löschung	660
Der DELETE fordert eine updateable Result Table	660
DELETE-Formate	661
Beispiele für SQL-Datenlöschungen	662
Searched Delete	662
Positioned Delete	663
SELECT mit SQL Data Change Statement (hier: SELECT mit DELETE)	664
6.5.6 MERGE - Daten einmischen	665
Der MERGE fordert eine updateable Result Table	665
MERGE-Formate	666
Beispiele für SQL-Daten-Mischvorgänge	667
Einfügung einer Zeile	667
Änderung einer Zeile	668
Änderung mehrerer Zeilen	669
SELECT mit SQL Data Change Statement (hier: SELECT mit MERGE)	670
SELECT mit MERGE in einem Cursor	673
7 SQL-DCL-Data Control Language	674
7.1 Grafischer Wegweiser durch das Kapitel 7 - SQL-DCL-Data Control Language	674
7.2 DB2-Zugriffsschutzkonzept	675
7.2.1 Überblick	675
Vorüberlegungen	675
Welche Ressourcen können von DB2 geschützt werden?	676
Konsequenzen bei einer vollständigen Auslagerung auf einen ESM	678
Aufteilung der Sicherheitsfunktionen zwischen DB2 und einem ESM	679
7.2.2 DB2-Privilegien-Konzept	680
7.2.3 DB2-Autorisierungs-Konzept	681
Übergabe der Autorisierungs-Identifikatoren	682
Primär-, Sekundär- und Current-Autorisierungs-Id	683
Zeitpunkt der Autorisierungsprüfung	684
Autorisierungsprüfung zur Bind-Zeit	684
Autorisierungsprüfung zur Ausführungs-Zeit	685
Verhaltensweisen bei Dynamic SQL (Statement Behavior)	686
Wirkung der DYNAMICRULES-Optionen BIND, DEFINEBIND, INVOKEBIND, RUN, DEFINERUN und INVOKERUN	686
Zusammenfassung der Verhaltensweisen (Statement-Behavior) und Konsequenzen der Autorisierungsprüfungen	687
7.2.4 Privilegien-Konzept	688
Privilegien-Struktur	688

Explizite Privilegien	689
Grant-Typen und Privilegienbereiche	690
DB2-Ressource-Typen und ihre Zuordnung der Einzel-Privilegien	691
Implizite Privilegien	693
Owner eines Objekts	693
Administrations-Gruppen	695
Hierarchie der ADM-Privilegien	697
7.2.5 Welche Privilegien werden wann benötigt?	698
Erforderliche Privilegien zur Verwaltung der DB2-Objekte	698
Die erforderlichen Privilegien eines Anwendungsentwicklers	699
Zusammenspiel von Autorisierungs-Id-Typen und Privilegien	700
Ausführung von Dynamic SQL-Statements und die erforderlichen Privilegien	700
Plan- und Package-Aktivitäten	701
7.3 GRANT - Erteilen DB2-Privilegien	702
7.3.1 Privilegien-Kategorien und GRANT-Varianten	702
7.3.2 Hierarchisches GRANT-Konzept	703
7.3.3 GRANT-Besonderheiten	704
7.3.4 Beispiele GRANT	705
GRANT-Szenario	705
7.4 REVOKE - Aufheben DB2-Privilegien	707
7.4.1 REVOKE-Varianten	707
7.4.2 REVOKE und die Cascading Effekte	708
7.4.3 REVOKE-Besonderheiten	710
7.4.4 Beispiele REVOKE	711
REVOKE-Szenario	711
7.5 Trusted Context und Rollen	712
7.5.1 Trusted Connection	712
Komponenten der Definition eines Trusted Contextes	713
Name des Trusted Contextes	713
System-Autorisierungs-Id	713
Default-Rolle und ROLE AS OBJECT OWNER	713
ENABLE oder DISABLE	713
Default-Security-Label	713
ATTRIBUTES	713
WITH USE FOR	714
7.5.2 Die Rolle (ROLE) im Rahmen eines Trusted Contextes	715
7.5.3 Aktivitäten eines anderen Benutzers mit dessen Objekt-Privilegien durchführen	716
7.5.4 SQL-Statements und Katalog-Informationen für Trusted Context und Rollen-Konzept	717
7.6 VIEW - Inhaltsbezogener Datenschutz	718
7.6.1 Besonderheiten bei der Anwendung in Programmen	719
7.7 MLS - Multi-Level Security	720
7.7.1 Probleme bisheriger Zugriffsschutz-Verfahren	720
7.7.2 MLS-Konzept und relevante Komponenten	721
Security Kategorie, Security Level und Security Label	722
7.7.3 MAC - Mandatory Access Control / Mandatory Access Checking	723
Dominanz (Dominate/Dominance)	723
Äquivalenz (Equivalent)	723

Nicht verarbeitbar	723
Inkompatibilität (disjoint oder disjunkt)	723
Write-down Control	723
7.7.4 MLS-Implementierungs-Varianten im DB2	724
MLS und MAC auf Objekt-Ebene	724
MLS auf Zeilen-Ebene (Row-Level Granularity)	725
Write-down Control bei Row-Level Granularity	725
Behandlung der betroffenen SQL-Statements	725
CREATE TABLE/ALTER TABLE	725
DROP TABLE	725
SELECT	725
INSERT	725
UPDATE	726
DELETE	726
Behandlung der betroffenen Utilities	727
LOAD	727
UNLOAD	727
REORG TABLESPACE	727
Behandlung in Triggern	727
Sonstige Besonderheiten	727
7.7.5 Zusammenfassung und Konsequenzen des MLS-Konzepts	728
Beispielhafte Ressource- und Benutzer-Definitionen	728
Security-Kategorien	728
Security-Level	728
Security-Label	728
Fazit	729
7.7.6 Vor- und Nachteile des MLS-Konzepts	730
Vorteile des MLS-Konzepts	730
Nachteile des MLS-Konzepts	730
7.8 Verschlüsselung von Daten (Encryption)	731
7.8.1 Komponenten für die Verschlüsselung	731
7.8.2 Welche Daten eignen sich für DB2-Verschlüsselungs-Techniken?	731
7.8.3 Definition der Meta-Daten für die Verschlüsselung	732
7.8.4 Die Rolle des Encryption Passwords und Password Hints	732
7.8.5 Verschlüsselungs-Konzepte	733
Verschlüsselung auf Spalten-Ebene	733
Verschlüsselung auf Daten-Wert-Ebene	733
7.8.6 Effekte der Verschlüsselung	734
Behandlung der Prädikate bei Vergleichen	734
Performance-Aspekte	734
Kann man auch PKs und FKs verschlüsseln?	735
Wie behandeln Utilities verschlüsselte Daten?	735
8 DB2-Katalog	736
8.1 Grafischer Wegweiser durch das Kapitel 8 - DB2-Katalog	736
8.2 Aufbau und Inhalte	737
8.2.1 Liste der Katalog-Tabellen	739
8.2.2 Zuordnung der Objekte zu Katalog-Tabellen	743

8.2.3	Referenzielle Beziehungen der wichtigsten Katalog-Tabellen	744
8.2.4	Katalog-Informationen zur Unterstützung der Objekt-Administration	746
	Katalog-Spalten mit Informationen zur Speicherorganisation	746
	Katalog-Spalten mit Informationen zur Zugriffspfad-Analyse durch den Optimizer	748
8.2.5	Statistiken und ihre Nutzungsmöglichkeit	756
	Überblick	756
	Was ist der Zweck der Statistiken?	756
	Übersicht der relevanten Statistik-Informationen (Generelle Statistik-Kategorien)	757
	DB2-Statistik-Kategorien und relevante Komponenten	758
	Speicherung der relevanten Index-Statistik-Informationen	760
	Bedeutung von LEAFDIST	761
	Bedeutung von LEAFNEAR und LEAFFAR	762
	Speicherung der relevanten Daten-Statistik-Informationen	763
	Bedeutung von NEAROFFPOSF und FAROFFPOSF	764
	Bedeutung von CLUSTERRATIOF	765
	Bedeutung von NEARINDREF und FARINDREF	766
	Speicherung der relevanten LOB-Statistik-Informationen	767
	Bedeutung von ORGRATIO	768
	Aktuelle Statistiken: RTS - Realtime-Statistiken	769
	Überblick	769
	RTS-Komponenten	770
	Sammeln und Aktualisieren der Statistikdaten	771
	Ausschreiben der Statistiken in die RTS-Tabellen (Externalisierung)	772
	Analyse der Statistiken	772
	Stored Procedure: DSNACCOX (vor DB2-Version 9 DSNACCOR) - Analyse Realtime Statistiken	772
	Welche RTS-Tatbestände führen zu einer Utility-Aktivierungs-Empfehlung?	773
	Welche generellen Katalog-Tatbestände führen zu einer Utility-Aktivierungs-Empfehlung?	774
8.3	Katalog-Abfragen	775
8.3.1	Katalog-Abfragen der Katalogbeschreibungen (Metadaten)	775
	Objekt-Abfragen für Anwendungs-Entwickler	777
8.3.2	Katalog-Abfragen zur Überwachung der optimalen Speicherorganisation	781
	Prüfung, ob RUNSTATS überhaupt noch nicht oder seit längerer Zeit nicht aktiviert wurde	781
8.3.3	Abfrage auf Reorganisations-Erfordernis	782
8.3.4	Nutzung von Tabellen und Indizes durch Packages	785
8.3.5	Katalog-Queries über referenzielle Beziehungen	786
8.3.6	Katalog-Queries zur Unterstützung und Kontrolle des Datenschutzes	788
8.4	Katalog-Veränderungen	789
9	DB2-Utilities	790
9.1	Grafischer Wegweiser durch das Kapitel 9 - DB2-Utilities	790
9.2	Überblick und relevante Komponenten	791
9.2.1	Abgrenzung IBM-DB2-Tools - DB2-Utilities	791
9.2.2	DB2-Online-Utilities und Standalone-Utilities	793
9.2.3	DB2-Online-Utilities	794
	Komponenten eines Utilities	794
	Aktivierungs-Varianten für Utilities	795
	Die Phasen eines Utilities	796
	Ergebnis- und Status-Kontrolle eines Utility-Laufs	797
	Status-Kontrolle während der Aktivierungs-Zeit eines Utility-Laufs	797
	Ergebnis-Kontrolle nach dem Abschluss eines Utility-Laufs	797

RESTART (CURRENT) und RESTART (PHASE)	798
Utility-Datasets	800
Vorgabe von Utility-Control-Statements	800
CCSID-Behandlung von Utility-Control Statements	800
Spezielle Utility-Control-Statements (SYSIN-DD-Statement)	800
Dataset-Charakteristiken	801
Übersicht der Datasets für Online-Utilities	802
Dataset-Dispositionen	803
Dataset-Größen	804
Vorgehensweisen, wenn Datasets zu klein sind	804
Generelle und spezielle Utility-Control Statements	805
Utility-Control Statement: EXEC SQL	806
Utility-Control Statement: LISTDEF	808
Utility-Control Statement: TEMPLATE	812
Utility-Control Statement: OPTIONS	813
Die Auswirkungen der Objekt-Versionierung für Utilities	815
Wie behandeln Utilities verschlüsselte Daten?	816
Wie behandeln Utilities MLS-geschützte Daten	816
Inline-Utility-Ausführungen	817
Besonderheiten der Inline-Kopien für Utilities	817
Inline-Kopien mit dem LOAD-Utility	817
Inline-Kopien mit dem REORG-Utility	821
Inline-Statistiken mit dem LOAD-, REORG- oder REBUILD-Utility	822
Sperren von Utilities und Kompatibilität zu anderen Utilities	823
Claims und Drains	823
Restriktiver Space-Status von Utilities	823
Kompatibilität der Utilities untereinander	824
Standalone-Utilities	825
9.2.4 Stored Procedures im Utility-Umfeld	825
9.3 DB2 Online Utilities	826
9.3.1 Übersicht der Kategorien und Utilities	826
9.3.2 Utilities zum Laden, Entladen und Reorganisieren von Daten und Indizes	827
LOAD-Utility	828
Überblick	828
Vorbereitungs-Aktivitäten	829
LOAD: Erforderliche Datasets und Objekte	830
LOAD: Utility-Ausführungs-Phasen	831
LOAD: Utility-Sperren und Abhängigkeiten zu anderen Utilities	832
Besonderheiten von Daten-Typen- und -Charakteristiken	833
Komprimierung der Daten	834
Performance-Faktoren beim Laden	834
Offline-Load	835
Überblick über die Komponenten und relevante Klauseln	835
Ein- und Ausgabe-Datasets	835
Laden der Daten von Partitioned Tablespaces	837
Paralleler Aufbau der Indizes beim LOAD	838
Abbruch und Restart des LOAD-Utilities	840
Online-Load	841
Überblick über die Komponenten und relevante Klauseln	841
Ein- und Ausgabe-Datasets	841
Unterscheidungsmerkmale von Offline- und Online-Load	842
Konsequenzen bei Einsatz des Online-Load-Utilities	842

Test-Ergebnisse Vergleich Offline-Online-Load	843
Empfehlungen	843
Nachbereitungs-Aktivitäten	844
Test-Beispiele	845
Erläuterung des Test-Beispiels	845
Job-Control und Utility-Control-Statements	845
Ausführungs-Protokolle	846
REBUILD INDEX-Utility	847
Überblick über die Komponenten und relevante Klauseln	847
Online-Rebuild (SHRLEVEL CHANGE)	848
Ein- und Ausgabe-Datasets	849
Vorbereitungs-Aktivitäten	849
REBUILD INDEX: Erforderliche Datasets und Objekte	850
REBUILD INDEX: Utility-Sperren und Abhängigkeiten zu anderen Utilities	850
REBUILD INDEX: Utility-Ausführungs-Phasen	851
Paralleler Aufbau der Indizes beim REBUILD INDEX	851
Abbruch und Restart des REBUILD INDEX-Utilities	851
Wann ist REBUILD INDEX zweckmäßig oder erforderlich?	852
Test-Beispiele	853
Job-Control	853
Utility-Control-Statements	853
Auszüge der Ausführungs-Protokolle	853
REORG INDEX-Utility	854
Überblick	854
Reorganisations-Methoden	854
Überblick über die Komponenten und relevante Klauseln	855
Ein- und Ausgabe-Datasets	855
Vorbereitungs-Aktivitäten	856
REORG INDEX: Erforderliche Datasets und Objekte	856
REORG INDEX: Utility-Sperren und Abhängigkeiten zu anderen Utilities	857
REORG INDEX: Utility-Ausführungs-Phasen	857
Paralleler Aufbau der Indizes beim REORG INDEX	858
Abbruch und Restart des REORG INDEX-Utilities	858
Wann ist ein REORG INDEX zweckmäßig oder erforderlich?	859
Test-Beispiele	860
Job-Control	860
Utility-Control-Statements	860
Auszüge der Ausführungs-Protokolle	860
REORG TABLESPACE-Utility	861
Überblick	861
Reorganisations-Methoden	861
Erweiterungen ab DB2-Version 9	862
Reorganisation der verschiedenen Tablespace-Typen	863
Reorganisation eines Segmented Tablespaces	863
Reorganisation eines Partitioned Tablespaces	863
Reorganisation von UTS-PBG-Tablespaces (Partition-by-Growth)	863
Reorganisation von XML-Tablespaces	863
Reorganisation von LOB-Tablespaces	864
Logging eines LOB-Tablespaces	865
Reorganisation von Katalog- und Directory-Tablespaces	865
Überblick über die Komponenten und relevante Klauseln	866
Ein- und Ausgabe-Datasets	866

Performance-Faktoren bei der Reorganisation	868
Parallele Reorganisation bei Partitioned Tablespaces	868
Parallele Verarbeitungsmöglichkeiten durch andere Prozesse	869
SHRLEVEL NONE - keine Parallelverarbeitung ab RELOAD erlaubt	869
SHRLEVEL REFERENCE - lesende Parallel-Verarbeitung erlaubt	870
SHRLEVEL CHANGE - Parallel-Verarbeitung erlaubt (Online-Reorg)	871
Rolle der Mapping Table	872
Besonderheiten beim Aufbau von Non-Partitioned Indizes (NPIs)	872
Nicht-Verfügbarkeitszeiten der Objekte für Parallel-Anwendungen	873
Konsequenzen bei Einsatz des Online-Reorg-Utilities	873
REBALANCE - Re-Balancieren ungleich großer Partitions	874
Vorbereitungs-Aktivitäten	875
REORG TABLESPACE: Erforderliche Datasets und Objekte	876
REORG TABLESPACE: Utility-Ausführungs-Phasen	878
REORG TABLESPACE: Utility-Sperren und Abhängigkeiten zu anderen Utilities	879
Paralleler Aufbau von Partitions beim REORG TABLESPACE	880
Paralleler Aufbau der Daten-Partitions	880
Paralleler Aufbau der Indizes	880
Abbruch und Restart des REORG TABLESPACE-Utilities	881
REORG TABLESPACE-Utility: Abbruch in Phasen und Pending-Status	881
Wann ist REORG TABLESPACE zweckmäßig oder erforderlich?	882
Nachbereitungs-Aktivitäten	883
Test-Beispiele	884
Job-Control mit Direkt-Vorgabe eines Tablespace-Namens	884
Utility-Control-Statements	884
Auszüge der Ausführungs-Protokolle	885
Job-Control mit Vorgabe von Objekt-Listen und Templates	886
Auszüge der Ausführungs-Protokolle	886
UNLOAD-Utility	889
Überblick	889
Besonderheiten beim Entladen der Daten	891
Entladen von Tablespaces, Tables und Image-Copies	891
Beeinflussung der Spalten- und Datenauswahl	892
Konvertierung der Daten	892
Vorbereitungs-Aktivitäten	892
UNLOAD: Erforderliche Datasets und Objekte	893
UNLOAD: Utility-Ausführungs-Phasen	893
UNLOAD: Utility-Sperren und Abhängigkeiten zu anderen Utilities	893
Abbruch und Restart des UNLOAD-Utilities	894
Test-Beispiele	894
Test-Beispiel: Entladen Daten eines Partitioned Tablespaces	894
Job-Control	894
Utility-Control-Statements	894
Auszüge des Ausführungs-Protokolls	894
Test-Beispiel: Entladen Daten DELIMITED	895
EBCDIC-Daten	895
Test-Beispiel: Entladen Daten mit der WHEN-Klausel und mit TEMPLATE in SYSTEMPL	896
Job-Control	896
Auszüge des Ausführungs-Protokolls	896
Ausdruck der Daten	897
Test-Beispiele: Entladen Daten mit CCSID EBCDIC und UNICODE	898
Test-Basis	898
Entladen beide mit Default-CCSID	898
Entlade-Bestände	899

Entladen beide mit CCSID EBCDIC	900
Entlade-Bestände	901
Entladen beide mit CCSID UNICODE	902
Entlade-Bestände	903
Entladen beide mit CCSID UNICODE DELIMITED	904
Entlade-Bestände	905
Test-Beispiel: Entladen LOB-Daten mit externer File Referenz und TEMPLATE	906
Test-Basis	906
UNLOAD-Job	906
UNLOAD-Ausführungs-Protokoll	907
9.3.3 Utilities zum Erzeugen von Statistik-Informationen aus Daten und Indizes	908
RUNSTATS-Utility	909
Überblick	909
Aktualisierung der Katalog-Statistiken	912
Welche Tabellen und Spalten aktualisiert RUNSTATS?	912
RUNSTATS: Wirkung der Parameter UPDATE und HISTORY	913
Woran erkennt man die Aktualität der Statistiken?	915
Darstellung der Statistikwerte im Katalog	915
Manuelle Veränderung der Statistikwerte im Katalog	916
Exaktheit der Statistikwerte	916
Ungleichförmige Verteilung von Datenwerten	917
Statische Erfassung von Einzelwerten (KEYCARD und FREQUVAL) und Spalten-Gruppen (COLGROUP)	917
Statische Erfassung der Verteilung von Spaltenwerten oder Wertegruppen (HISTOGRAM)	918
Vorbereitungs-Aktivitäten	920
RUNSTATS TABLESPACE: Erforderliche Datasets und Objekte	920
RUNSTATS TABLESPACE: Utility-Ausführungs-Phasen	920
RUNSTATS TABLESPACE: Utility-Sperren und Abhängigkeiten zu anderen Utilities	921
Abbruch und Restart des RUNSTATS-Utilities	921
Test-Beispiele	922
RUNSTATS TABLESPACE TABLE ALL und INDEX ALL	922
Utility-Control-Statements	922
Auszüge der Ausführungs-Protokolle	922
STOSPACE-Utility	925
Überblick	925
Vorbereitungs-Aktivitäten	926
STOSPACE: Erforderliche Datasets und Objekte	926
STOSPACE: Utility-Ausführungs-Phasen	926
STOSPACE: Utility-Sperren und Abhängigkeiten zu anderen Utilities	926
Abbruch und Restart des STOSPACE-Utilities	926
Test-Beispiele	927
Job-Control	927
Utility-Control-Statements	927
Ausführungs-Protokoll	927
Katalog-Statistiken	927
9.3.4 Utilities zum Prüfen der Konsistenz von Daten und Indizes	928
Überblick und Abgrenzung der Aufgaben	928
LOB- oder XML-Fehler-Typen: Orphan-, Missing-, Out-of-Synch- und Invalid-Werte	930
Entdeckung der Fehlerzustände und mögliche Ursachen	930
Bereinigungs-Möglichkeiten der Fehlerzustände	931
CHECK DATA-Utility	932
Überblick	932
Vorbereitungs-Aktivitäten	935
CHECK DATA: Erforderliche Datasets und Objekte	935

CHECK DATA: Utility-Ausführungs-Phasen	936
CHECK DATA: Utility-Sperren und Abhängigkeiten zu anderen Utilities	936
Abbruch und Restart des CHECK DATA-Utilities	937
Nachbereitungs-Aktivitäten	937
Test-Beispiele	938
Löschen fehlerhafter referenzieller Zeilen	938
Job-Control	938
Utility-Control-Statements	938
Auszüge der Ausführungs-Protokolle	938
LOB-Daten aufgrund eines PIT-Recoveries auf ungültig setzen	939
Utility-Control-Statements	939
Ausführungs-Protokoll	939
CHECK INDEX-Utility	940
Überblick	940
Vorbereitungs-Aktivitäten	941
CHECK INDEX: Erforderliche Datasets und Objekte	942
CHECK INDEX: Utility-Ausführungs-Phasen	942
CHECK INDEX: Utility-Sperren und Abhängigkeiten zu anderen Utilities	942
Abbruch und Restart des CHECK DATA-Utilities	942
Nachbereitungs-Aktivitäten	943
Test-Beispiele	944
Job-Control	944
Utility-Control-Statements	944
Auszüge der Ausführungs-Protokolle	944
CHECK LOB-Utility	945
Überblick	945
Vorbereitungs-Aktivitäten	946
CHECK LOB: Erforderliche Datasets und Objekte	946
CHECK LOB: Utility-Ausführungs-Phasen	946
CHECK LOB: Utility-Sperren und Abhängigkeiten zu anderen Utilities	947
Abbruch und Restart des CHECK LOB-Utilities	947
Nachbereitungs-Aktivitäten	947
Test-Beispiele	948
Job-Control	948
Utility-Control-Statements	948
Auszüge der Ausführungs-Protokolle	948
9.3.5 Utilities zum Sichern und Wiederherstellen von Daten und Indizes	949
BACKUP SYSTEM-Utility	950
Überblick	950
Vorbereitungs-Aktivitäten	952
BACKUP SYSTEM: Erforderliche Datasets und Objekte	952
BACKUP SYSTEM: Utility-Ausführungs-Phasen	952
BACKUP SYSTEM: Utility-Sperren und Abhängigkeiten zu anderen Utilities	952
Abbruch und Restart des BACKUP SYSTEM-Utilities	953
Nachbereitungs-Aktivitäten	953
Test-Beispiele	953
Job-Control	953
COPY-Utility	954
Überblick	954
Copy-Typen	956
Full Image Copy	956
Incremental Image Copy	956

Primär und Backup-Kopie	957
Kopien für das lokale und das Recovery-System	957
DFSMS Concurrent Copy	957
Parallele Ausführungsmöglichkeiten des Utilities	958
Parallele Sicherung von Objekten in einem Job	958
Parallele Sicherung von Objekten in mehreren Jobs	958
Erzeugen von Kopien für mehrere Objekte	958
Vorgabe einer Liste in einer Copy-Anweisung eines Jobs	958
Vorgabe mehrerer Copy-Anweisungen in einem Jobs	958
Wie kann die Konsistenz der Objekte sichergestellt werden?	959
Bei bestimmten 'Restriktiven Pending-Status' kann bzw. sollte keine Image Copy erzeugt werden	959
Vom Einsatz von -TERM UTIL wird beim COPY abgeraten	960
Vorbereitungs-Aktivitäten	960
COPY: Erforderliche Datasets und Objekte	960
COPY: Utility-Ausführungs-Phasen	961
COPY: Utility-Sperren und Abhängigkeiten zu anderen Utilities	961
Abbruch und Restart des COPY-Utilities	961
Nachbereitungs-Aktivitäten	961
Test-Beispiele	962
Job-Control	962
Utility-Control-Statements	962
Auszüge der Ausführungs-Protokolle	962
COPYTOCOPY-Utility	965
Überblick	965
Vorbereitungs-Aktivitäten	967
COPYTOCOPY: Erforderliche Datasets und Objekte	967
COPYTOCOPY: Utility-Ausführungs-Phasen	967
COPYTOCOPY: Utility-Sperren und Abhängigkeiten zu anderen Utilities	967
Abbruch und Restart des COPYTOCOPY-Utilities	968
Nachbereitungs-Aktivitäten	968
Test-Beispiele	969
Job-Control	969
Utility-Control-Statements	969
Auszüge der Ausführungs-Protokolle	969
MERGECOPY-Utility	970
Überblick	970
Vorbereitungs-Aktivitäten	971
MERGECOPY: Erforderliche Datasets und Objekte	971
MERGECOPY: Utility-Ausführungs-Phasen	972
MERGECOPY: Utility-Sperren und Abhängigkeiten zu anderen Utilities	972
Abbruch und Restart des MERGECOPY-Utilities	972
Nachbereitungs-Aktivitäten	972
Test-Beispiele	973
Job-Control	973
Utility-Control-Statements	973
Auszüge der Ausführungs-Protokolle	973
QUIESCE-Utility	974
Überblick	974
Vorbereitungs-Aktivitäten	976
QUIESCE: Erforderliche Datasets und Objekte	976
QUIESCE: Utility-Ausführungs-Phasen	976
QUIESCE: Utility-Sperren und Abhängigkeiten zu anderen Utilities	976

Abbruch und Restart des QUIESCE-Utilities	977
Nachbereitungs-Aktivitäten	977
Test-Beispiele	977
Job-Control	977
Utility-Control-Statements	977
Auszüge der Ausführungs-Protokolle	977
RECOVER-Utility	979
Überblick	979
Zurücksetzen von Datenzuständen	981
Zurücksetzen auf den aktuellen Status	982
Zurücksetzen auf einen früheren Zeitpunkt	982
Copy-Typen	983
RECOVER von Tablespaces mit verknüpften Objekten (Indizes, RI-Constraints, LOB- oder XML-Tablespaces sowie deren Indizes)	984
Pending Status	984
Ungültigkeits-Markierung (invalid) von LOB- oder XML-Werten	984
Wiederherstellen von Base-Tablespaces	984
Wiederherstellen von LOB- oder XML-Tablespaces	984
Wiederherstellen von Indexspaces	984
RECOVER von Tablespaces, die mit NOT LOGGED definiert sind	985
RECOVER von Katalog- und Directory-Objekten	985
Objekt-Hierarchie	985
Vorbereitungs-Aktivitäten	986
RECOVER: Erforderliche Datasets und Objekte	986
RECOVER: Utility-Ausführungs-Phasen	987
RECOVER: Utility-Sperren und Abhängigkeiten zu anderen Utilities	987
Abbruch und Restart des RECOVER-Utilities	988
Nachbereitungs-Aktivitäten	988
Test-Beispiele	988
Job-Control	988
Utility-Control-Statements	988
Auszüge der Ausführungs-Protokolle	988
RESTORE SYSTEM-Utility	990
Überblick	990
Vorbereitungs-Aktivitäten	991
RESTORE SYSTEM: Erforderliche Datasets und Objekte	992
RESTORE SYSTEM: Utility-Ausführungs-Phasen	992
RESTORE SYSTEM: Utility-Sperren und Abhängigkeiten zu anderen Utilities	992
Abbruch und Restart des RESTORE SYSTEM-Utilities	993
Nachbereitungs-Aktivitäten	993
Test-Beispiele	993
Job-Control	993
9.3.6 Utilities zum Löschen nicht mehr benötigter System-Informationen	994
MODIFY RECOVERY-Utility	995
Überblick	995
Vorbereitungs-Aktivitäten	996
MODIFY RECOVERY: Erforderliche Datasets und Objekte	997
MODIFY RECOVERY: Utility-Ausführungs-Phasen	997
MODIFY RECOVERY: Utility-Sperren und Abhängigkeiten zu anderen Utilities	997
Abbruch und Restart des MODIFY RECOVERY-Utilities	997
Nachbereitungs-Aktivitäten	997
Test-Beispiele	998

Job-Control	998
Utility-Control-Statements	998
Auszüge der Ausführungs-Protokolle	998
Szenario der Löschung aller Image Copies, außer der letzten Full Image Copy und ggf. nachfolgender Incremental Image Copies	999
Auszug aus SYSIBM.SYSCOPY vor MODIFY RECOVERY	999
MODIFY RECOVERY-Protokoll	999
Auszug aus SYSIBM.SYSCOPY nach MODIFY RECOVERY	1000
MODIFY STATISTICS-Utility	1001
Überblick	1001
Vorbereitungs-Aktivitäten	1002
MODIFY STATISTICS: Erforderliche Datasets und Objekte	1002
MODIFY STATISTICS: Utility-Ausführungs-Phasen	1002
MODIFY STATISTICS: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1002
Abbruch und Restart des MODIFY STATISTICS-Utilities	1002
Nachbereitungs-Aktivitäten	1002
Test-Beispiele	1003
Job-Control	1003
Utility-Control-Statements	1003
Auszüge der Ausführungs-Protokolle	1003
9.3.7 Utilities für Objekt-Informationen und zur Fehler-Analyse und Fehler-Bereinigung	1004
DIAGNOSE-Utility	1005
Überblick	1005
Vorbereitungs-Aktivitäten	1006
DIAGNOSE: Erforderliche Datasets und Objekte	1006
DIAGNOSE: Utility-Ausführungs-Phasen	1006
DIAGNOSE: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1006
Abbruch und Restart des DIAGNOSE-Utilities	1006
Nachbereitungs-Aktivitäten	1006
Test-Beispiele	1007
Job-Control	1007
Utility-Control-Statements	1007
Auszüge der Ausführungs-Protokolle	1007
REPAIR-Utility	1008
Überblick	1008
Vorbereitungs-Aktivitäten	1010
REPAIR: Erforderliche Datasets und Objekte	1010
REPAIR: Utility-Ausführungs-Phasen	1010
REPAIR: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1011
Abbruch und Restart des REPAIR-Utilities	1011
Nachbereitungs-Aktivitäten	1011
Test-Beispiele	1012
Job-Control	1012
Utility-Control-Statements	1012
Auszüge der Ausführungs-Protokolle	1012
REPORT-Utility	1013
Überblick	1013
Vorbereitungs-Aktivitäten	1014
REPORT: Erforderliche Datasets und Objekte	1014
REPORT: Utility-Ausführungs-Phasen	1014
REPORT: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1015
Nachbereitungs-Aktivitäten	1015
Test-Beispiele	1015

Job-Control	1015
Utility-Control-Statements	1015
Auszüge der Ausführungs-Protokolle	1015
9.4 Standalone-Utilities	1018
9.4.1 Übersicht	1018
9.4.2 DSNJCNVB	1020
Überblick	1020
Vorbereitungs-Aktivitäten	1020
DSNJCNVB: Erforderliche Datasets und Objekte	1021
Nachbereitungs-Aktivitäten	1021
Test-Beispiele	1021
Job-Control	1021
Auszüge der Ausführungs-Protokolle	1021
9.4.3 DSNJLOGF (Preformat Active Log)	1022
Überblick	1022
Vorbereitungs-Aktivitäten	1022
DSNJLOGF: Erforderliche Datasets und Objekte	1022
Nachbereitungs-Aktivitäten	1023
Test-Beispiele	1023
Job-Control	1023
Auszüge der Ausführungs-Protokolle	1023
9.4.4 DSNJU003 (Change Log Inventory)	1024
Überblick	1024
Vorbereitungs-Aktivitäten	1026
DSNJU003: Erforderliche Datasets und Objekte	1026
Nachbereitungs-Aktivitäten	1026
Test-Beispiele	1027
Job-Control	1027
Auszüge der Ausführungs-Protokolle	1027
9.4.5 DSNJU004 (Print Log Map)	1028
Überblick	1028
Vorbereitungs-Aktivitäten	1029
DSNJU004: Erforderliche Datasets und Objekte	1029
Nachbereitungs-Aktivitäten	1029
Test-Beispiele	1029
Job-Control	1029
Auszüge der Ausführungs-Protokolle (aufbereitete Form mit Erläuterungen)	1030
9.4.6 DSN1CHKR	1033
Überblick	1033
Vorbereitungs-Aktivitäten	1034
DSN1CHKR: Erforderliche Datasets und Objekte	1034
Nachbereitungs-Aktivitäten	1034
Test-Beispiele	1034
Job-Control	1034
9.4.7 DSN1COMP	1035
Überblick	1035
Vorbereitungs-Aktivitäten	1036
DSN1COMP: Erforderliche Datasets und Objekte	1036

Nachbereitungs-Aktivitäten	1036
Test-Beispiele	1036
Job-Control	1036
Auszüge der Ausführungs-Protokolle	1037
9.4.8 DSN1COPY	1038
Überblick	1038
Vorbereitungs-Aktivitäten	1040
DSN1COPY: Erforderliche Datasets und Objekte	1040
Test-Beispiele	1041
Job-Control	1041
Auszüge der Ausführungs-Protokolle	1042
9.4.9 DSN1LOGP	1043
Überblick	1043
Vorbereitungsaktivitäten	1044
DSN1LOGP: Erforderliche Datasets und Objekte	1044
Nachbereitungs-Aktivitäten	1044
Test-Beispiele	1045
Job-Control	1045
Auszüge der Ausführungs-Protokolle	1045
9.4.10 DSN1PRNT	1046
Überblick	1046
Vorbereitungs-Aktivitäten	1047
DSN1PRNT: Erforderliche Datasets und Objekte	1047
Nachbereitungs-Aktivitäten	1047
Test-Beispiele	1048
Job-Control	1048
9.4.11 DSN1SDMP	1053
Überblick	1053
Vorbereitungs-Aktivitäten	1054
DSN1SDMP: Erforderliche Datasets und Objekte	1054
Nachbereitungs-Aktivitäten	1054
Test-Beispiele	1054
Job-Control	1054
9.5 Automatisierung des Anstoßes von Utility-Läufen durch Statistik-Analyse	1055
9.5.1 Relevante, automatisierbare Utilities	1056
9.6 Pending Status (Advisory und Restricted Status)	1057
10 Datenentwurf unter DB2	1061
10.1 Grafischer Wegweiser durch das Kapitel 10 - Datenentwurf	1061
10.2 Rolle des ANSI-SPARC-Drei-Schemata-Ansatzes	1062
10.3 Vorgehens-Modell	1064
10.3.1 Entwicklungsgeschichte der relevanten Daten-Modelle	1066
10.3.2 ERM - Entity-Relationship-Modell	1067
10.3.3 Relationen-Modell (RM)	1069
10.3.4 Gemeinsamkeiten von ERM und RM	1071
10.4 Logisches Design	1075
10.4.1 Normalisierung	1076

Unnormalisierte Daten	1077
1NF - Atomic Attribute ohne Wiederhol-Elemente	1079
2NF - Volle funktionale Abhängigkeit zu allen PK-Bestandteilen	1081
3NF - Keine transitiven Abhängigkeiten der Nicht-PK-Attribute	1083
Kritische Analyse der erreichten 3NF - Struktur-Qualität	1084
BCNF - Volle Abhängigkeit zum Candidate Key	1085
4NF - Keine paarweisen mehrwertigen Abhängigkeiten	1086
5NF - Ultimative Normal Form - Project-Join-Normalform (PJ/NF)	1088
Abschlussüberlegungen zur Normalisierung	1091
10.4.2 Dokumentation der Daten-Design-Ergebnisse	1092
Bubbles/Blasendiagramme	1092
Relationale Notation	1093
Dependency Diagram	1094
Netzwerk Diagramm	1095
Objektorientiertes State Transition Diagram	1096
DB2-Unterstützungstools	1097
10.4.3 Festlegung der referenziellen Verarbeitungs-Regeln	1098
Darstellung der referenziellen Verarbeitungs-Maßnahmen (RI)	1100
10.4.4 Festlegung der Geschäfts-Regeln (Business-Rules)	1102
Zusammenfassende Definition der Business-Rules	1103
Festlegung der logischen Trigger für die Business-Rules	1104
10.4.5 Übersicht der kompletten Integritäts-Regeln und -Maßnahmen	1106
Strukturierung der Geschäfts-Regeln und -Aktionen	1108
Umsetzung der Geschäfts-Regeln und -Aktionen in SQL-Funktionen	1109
10.4.6 Festlegung der konzeptionellen Views	1110
Bilden von Schemata für spezielle Daten-Sichten	1110
OLAP	1111
Data Warehouse: Cubes und Dimensionen	1112
Fakten (Facts)	1112
Maßeinheiten (Measures)	1112
Dimensionen (Dimensions)	1113
OLAP-Meta-Modell: Objekt-Typen	1117
10.4.7 Checkliste für das logische Daten-Design	1118
Fachliches Daten-Modell	1118
Konzeptionelles Daten-Modell	1119
10.5 Physisches Design	1121
10.5.1 Aufgabenstellungen	1121
10.5.2 Ableiten der physischen Benutzerobjekte aus logischen Relationen	1122
Zielsetzungen eines physischen Daten-Modells	1122
Kandidaten für die physische Implementierung	1123
10.5.3 Bewertung der Auswirkungen logisch definierter Zugriffspfade	1123
Aufwand bei der Datenbeschaffung	1125
Individuelle Performance-Relevanz	1128
Auswirkungen der Sperr-Maßnahmen	1129
10.5.4 De-Normalisierungs-Maßnahmen	1130
Daten-Zusammenführung	1130
Trennung der Daten (Verteilung)	1132

Bilden zusätzlicher Redundanzen	1133
Bilden von Schemata für spezielle Datenhaltungs-Konzepte	1135
Data Warehouse	1135
Terminologie und Komponenten	1135
MOLAP und ROLAP	1137
Star-Schema	1139
Snowflake-Schema	1140
Struktur-Besonderheiten beim Einsatz von MQTs	1141
Konsequenzen der De-Normalisierung	1142
10.5.5 Indizes	1143
10.5.6 Unterstützung der referenziellen Integrität (RI) durch DB2	1144
Terminologie und Einrichtung unter DB2	1144
Definierbare Strukturen	1146
Zulässige Einzel-Struktur-Beziehungen	1146
Umsetzung der M : N - Struktur-Beziehungen	1147
Logische referenzielle Struktur	1148
Physische referenzielle Struktur (Tablespace-Set)	1154
Gesamt-Überblick über die DB2-RI-Unterstützung	1156
Anforderungen an PK und FK	1158
DB2-Primary-Key (PK) und Parent Key	1158
DB2-Foreign-Key (FK)	1159
Verwaltung der RI-Definitionen im Katalog	1160
Katalog-Informationen für RI	1160
RI-DDL-Beispiele	1161
Argumente für und wider den Einsatz von DB2-RI	1163
10.5.7 Unterstützung der Business Rules mit Funktionen und Triggern	1164
De-Normalisierungs-Auswirkungen	1164
Zuordnung der Regeln und Maßnahmen zu den physischen Strukturen	1164
Funktionale Konsequenzen bei Nicht-Nutzung von DB2-RI	1166
Funktionale Konsequenzen bei Struktur-De-Normalisierungen	1167
10.5.8 Festlegung der sonstigen DB2-Objekte mit ihren Ausprägungen	1169
10.5.9 Festlegung der sonstigen physischen Ressourcen	1169
10.5.10 DB2-Unterstützungstools	1169
10.5.11 Checkliste für das physische DB2-Design	1170
11 Datenspeicherung intern und extern	1172
11.1 Grafischer Wegweiser durch das Kapitel 11	1172
11.2 Übersicht der Speicherungsformen	1173
11.3 Space-Management	1175
11.3.1 Verwaltung und Adressierung der Daten innerhalb des Page Sets	1175
11.3.2 Page Sets	1177
Linear Page Sets	1177
Partitioned Page Sets	1178
Universal Page Sets	1179
LOB Page Sets	1180
XML Page Sets	1181
11.3.3 Page Typen des File Page Sets	1182
Header Page und System Pages	1183

Space Map Pages	1184
Daten Pages	1185
Adressierung der Daten-Zeilen	1187
Insert-Strategie innerhalb der Daten-Pages	1188
Update-Strategie bei variablen Zeilen	1189
Delete-Strategie innerhalb der Daten-Pages	1189
Freespace-Verwaltung der Daten-Pages	1190
Plattenplatz-Bedarfs-Ermittlung	1192
Plattenplatz-Bedarfs-Ermittlung für LOB-Werte	1194
PRIQTY	1195
SECQTY	1195
Kompression der Daten	1196
11.3.4 Page Typen des Index Page Sets	1198
Header Page und System Pages	1198
Space Map Pages	1198
Index Pages	1198
Non-Leaf Pages	1199
Leaf Pages	1200
Insert-Strategie innerhalb der Index-Pages	1201
Update-Strategie innerhalb der Index-Pages	1201
Delete-Strategie innerhalb der Index-Pages	1201
Freespace-Verwaltung der Index-Pages	1202
Verteilung der Non-Partitioned-Index Daten mit Hilfe von PIECESIZE	1202
Plattenplatz-Bedarfs-Ermittlung für Indizes	1203
11.3.5 Unterstützung von Page-Größen > 4-KB im Index	1205
11.3.6 Index-Komprimierung	1206
DSN1COMP-Protokoll	1207
11.4 Nutzung des virtuellen Adressraums	1208
11.4.1 Nutzung der z/OS-Architektur	1208
11.4.2 Datenpool-Konzept	1209
RID-Pool	1209
Sort-Pool	1210
EDM-Speicherbereiche	1211
Compressions-Dictionary	1212
Materialisierung von LOB-Werten	1212
IRLM-Locks	1212
LE-Tokens	1212
11.4.3 Bufferpool-Konzept	1213
Group Bufferpools (GBP)	1215
Virtuelles Bufferkonzept	1215
Zuordnung der Bufferpools zu Tablespaces und Indexspaces	1215
Unterstützung verschiedener Page-Größen	1216
Bufferpool-Page-Größe und VSAM-CI-Größe	1216
Fixierung von Buffern im Realspeicher	1216
Einbindung der Bufferpools in die System-Schichten	1217
Lese-Anforderungen	1218
Direktes Einlesen (Synchronous Read)	1218
Sequential Prefetch	1218
Dynamic Prefetch (Sequential Detection)	1219

List Prefetch (List Sequential Prefetch)	1220
Schreib-Anforderungen	1221
Zustandsformen der Bufferpool-Page-Typen	1222
Bufferpool-Thresholds	1223
Threshold-Parametrisierungs-Empfehlungen für LOB- und XML-Werte	1224
Mögliche Strategien der Bufferpool-Einrichtung	1225
Bufferpool-Hit Ratio und MUPA	1226
Bufferpool-Besonderheiten für LOB- und XML-Werte	1227
11.5 VSAM-Datasets	1228
11.5.1 Nutzung von DFSMS (Storage Management Subsystem)	1230
12 DB2-Datensicherheitseinrichtungen	1231
12.1 Grafischer Wegweiser durch das Kapitel 12 - Datensicherheitseinrichtungen	1231
12.2 DB2-Sicherheitsinstrumentarium	1232
12.3 Transaktionsabwicklung	1234
12.3.1 Terminologie und Transaktions-Definition	1234
12.3.2 Konsistenz-Koordination bei Einsatz mehrerer Ressource-Manager	1236
12.4 LUW-, UOW- und UOR-Konzept	1238
12.4.1 Vorgangsabwicklung unter TSO oder einem Client ohne UOW-Konzept	1240
12.4.2 Vorgangsabwicklung unter CICS oder IMS mit UOW-Konzept	1241
12.4.3 Die Wirkung von SAVEPOINT und ROLLBACK TO SAVEPOINT	1242
12.4.4 Commit-Konzepte	1243
Überblick über Read-only-, Single-Phase- und Two-Phase-Commit	1243
Two-Phase-Commit (2-Phasen-Commit)	1244
UOR-Status nach einem Fehler	1245
12.5 Probleme und Lösungswege der Konkurrenzverarbeitung	1246
12.5.1 Verlorener Update	1247
12.5.2 Zugriff zu Daten, deren UOW noch nicht abgeschlossen ist	1248
12.5.3 Wiederholtes Einlesen von Daten innerhalb einer UOW	1249
12.5.4 Lösungswege zum Abbau der Konkurrenz-Problematiken	1250
12.5.5 Deadlock	1252
Vermeidung von Deadlocks	1253
Erkennung von Timeouts und Deadlocks	1255
12.6 DB2-Sperrverfahren	1256
12.6.1 Überblick	1256
12.6.2 Transaction Lock	1257
Lock-Charakteristiken	1257
Lock-Objekt	1259
Implizite Lock-Auswirkungen auf die DB2-Systemressourcen	1259
Lock-Dauer	1261
Lock-Niveau	1262
Lock Eskalation	1264
SQL-Statement: LOCK TABLE	1264
Lock-Modus	1265
Lock-Modus auf Lock-Niveau: Tablespace, Table bzw. Partition	1265
Lock-Modus auf Lock-Niveau: Page bzw. Row	1267
Zusammenfassung: Aktivierung von Lock-Niveau und Lock-Modus	1268
DB2-Lock-Modus-Entscheidungen	1269

Isolation-Level	1271
RR - Repeatable Read	1271
RS - Read Stability	1271
CS - Cursor Stability	1272
UR - Uncommitted Read	1272
Lock Promotion	1272
Vorgabemöglichkeiten des Isolation-Levels und Wechselwirkung in Package und Plan	1273
Unterschiede zwischen Latch und Lock	1274
Lock Avoidance	1274
Lock-Besonderheiten beim Cursor-Einsatz	1276
Optimistic Concurrency Control (Optimistic Locking)	1276
LOCK-Besonderheiten bei LOB- und XML-Daten	1277
12.6.3 Drain Lock, Drains und Claims	1278
12.6.4 IRLM - Internal Resource Lock Manager	1279
12.6.5 Data Sharing (DS)	1280
Lock-Optimierungen bei Data Sharing	1280
Lock-Typen bei Data Sharing	1280
L-Locks und P-Locks	1281
12.6.6 Zusammenfassung der DB2-Lock-Mechanismen	1283
Welche relevanten Parameter steuern die Lock-Mechanismen?	1283
Checkliste für den Einsatz der DB2-Lock-Mechanismen	1284
Übersicht: Sperrniveau und Lock-Maßnahmen	1286
12.7 Konsequenzen der DB2-Sperrverfahren auf die Anwendungs-Entwicklung	1287
12.7.1 Zusammenspiel von Isolation-Level, Lock-Modus und -Dauer	1287
12.7.2 Verhinderung eines Parallel-Updates	1288
Einsatz des Cursor-Konzeptes	1288
Cursor-Konzept oder SELECT WITH RR bzw. RS?	1289
Optimistic Locking Strategie	1290
12.7.3 Parallelverarbeitung von Online- und Batch-Anwendungen	1291
12.8 Logging-Einrichtungen	1292
12.8.1 Steuerungsfunktionen für die Logging-Abwicklung	1294
Logging-Philosophie	1294
Logging-Hierarchie	1295
12.8.2 Führen und Ausschreiben von LOG-Informationen	1297
Maßnahmen zur Reduzierung des LOG-Aufwands	1298
12.8.3 Konsistenzsicherung innerhalb des DB2-Subsystems (REDO, UNDO)	1299
Manuelle, nicht durch DB2 überwachte Konsistenz-Maßnahmen	1300
12.8.4 Konsistenzsicherung mit Allied Agents	1300
12.8.5 Konsistenzsicherung bei DB2 Data Sharing (DS)	1301
12.9 Storage Server	1302
12.9.1 ESS - Enterprise Storage Server	1302
12.9.2 AUDIT TRACE	1303
12.10 Problembereiche der Datensicherheitseinrichtungen	1304
13 Anwendungsprogrammierung unter DB2	1306
13.1 Grafischer Wegweiser durch das Kapitel 13 - Anwendungsprogrammierung	1306
13.2 Anwendungsprogrammierung: Relevante Komponenten im Überblick	1307

13.3 Einsatzspektrum von DB2	1308
13.3.1 Die verschiedenen Programm-Typen	1309
Abgrenzung Dialog-, Online-, Batch- und Client/Server-Verarbeitung	1309
Was versteht man unter Batch-Verarbeitung?	1310
Brauchen wir noch die Batch-Verarbeitung?	1311
Abgrenzung der Online- und Batch-Charakteristiken	1313
Client/Server-Charakteristiken	1314
13.3.2 Wahl des geeigneten Trägersystems	1315
Dialog-Verarbeitung	1315
Foreground-Verarbeitung	1315
Background-Verarbeitung	1316
Batch-Verarbeitung	1317
Started Tasks (STC) und Message-Konzept	1318
13.3.3 Die verschiedenen Betriebsformen	1319
Normale Betriebszeiten	1319
Erweiterte Betriebszeiten (erweiterte Verfügbarkeit)	1319
24-Stunden-Betrieb (permanente Verfügbarkeit)	1320
Zielsetzungen eines Multi-User-Betriebs	1321
Ein Wort zur 'Permanenz'	1321
13.3.4 Programm-Design-Überlegungen	1322
Programmübergreifende Maßnahmen	1322
Der Einsatz von SQL: Möglichkeiten und Restriktionen	1324
13.4 Programmentwicklung unter DB2	1326
13.4.1 Unterstützte Programmiersprachen	1326
Embedded SQL	1326
Call Level Interface für C- und C++-Programme (CLI bzw. ODBC)	1326
Java-Sprachschnittstellen (JDBC und SQLJ)	1327
JDBC-API für Java-Anwendungen	1327
SQLJ-API für Java-Anwendungen	1327
pureQuery für Java-Anwendungen	1327
13.4.2 Entwicklung einer DB2-Anwendung	1328
Überblick der relevanten Entwicklungs-Komponenten	1328
Entwicklungs-Komponenten für Embedded SQL	1329
DB2I: DCLGEN-Generierung	1330
SQL-Statement-Coprozessor	1331
DB2-Precompiler	1332
Aufgaben des Precompilers	1332
Precompiler bzw. Coprocessor-Optionen	1333
Standard-SQL	1336
Modifikationen der Programm-Source durch den Precompiler	1338
Entwicklungs-Komponenten für Java-Anwendungen	1340
JDBC-API	1340
SQLJ-API	1340
Entwicklungs-Komponenten für CLI/ODBC-Anwendungen	1341
BIND-Prozess	1342
Aufgaben des BIND-Prozesses	1342
Objekte des Bind-Prozesses	1343
DBRM: Database Request Module	1343

Package	1343
Package-Zuordnung zu mehreren Collections	1345
Was ist eine invalidated Package?	1346
Plan	1347
Package-Liste im Plan: PKLIST	1349
Nutzung von remote Packages	1351
Packages von Routinen	1351
PLAN Management (müsste eigentlich Package Management heißen)	1352
Die Zuordnung eines Plans zu einem Programm	1353
Dynamische Plan-Zuordnung im CICS	1354
Was ist ein invalidated Plan?	1354
Die verschiedenen BIND- und REBIND-Typen	1355
STATIC BIND: BIND PACKAGE, BIND PLAN	1355
REBIND PACKAGE, REBIND PLAN	1356
FREE PACKAGE, FREE PLAN, DROP PACKAGE	1356
DYNAMIC BIND	1356
AUTOMATIC REBIND	1357
INCREMENTAL BIND	1357
DEFERRED BIND	1357
Relevante Katalog-Tabellen	1358
Binder (Linkage Editor)	1359
Aktivieren der Attachment-Facility-Schnittstellen	1359
13.4.3 Beispiel- und Hilfs-Programme: DSNTIAD, DSNTDP2/4 und DSNTIAUL	1360
Übersicht	1360
DSNTIAUL	1361
DSNTIAUL-Beispiel	1361
DSNTEP2/DSNTEP4	1364
DSNTEP2/DSNTEP4-Beispiel	1365
13.5 SQL-Spracheinsatz in Anwendungsprogrammen	1366
13.5.1 Übersicht der Sprachschnittstellen	1366
Interactive SQL, Embedded SQL, CLI, Static SQL und Dynamic SQL	1366
EXEC-Level-API: Embedded SQL	1367
Behandlung von Datenmengen	1369
CLI: Call-Level-Interface bzw. ODBC-Schnittstelle	1370
Java-Schnittstellen	1372
JDBC - Java Database Connectivity	1374
SQLJ - SQL for Java	1375
REXX-Schnittstellen	1378
13.5.2 Grundstruktur eines DB2-Programmes im EXEC-Level	1379
Relevante Komponenten	1379
Basis-Codier-Regeln für SQL-Statements	1380
Allgemeine Regeln	1380
Assembler-Besonderheiten	1381
C-Besonderheiten	1381
COBOL-Besonderheiten	1382
PL/I-Besonderheiten	1385
13.5.3 Definition der Variablen und Daten-Strukturen	1386
Definition der SQLCA (SQL Communication Area)	1387
Tabellen- und View-Deklarationen	1388
Host-Variablen und NULL-Indikatoren	1390
Typen: Host-Variablen und Host-Strukturen	1390

Einlesen von NULL-Values (NULL-Markierungen)	1392
Setzen von NULL-Werten (NULL-Markierungen)	1394
Einsatzspektrum von Host-Variablen und Indikatoren	1396
Host-Variablen-Arrays	1397
Unterstützte Daten-Typen (mit SQLTYPE)	1398
Konvertierungsregeln zwischen DB2-Spalten und Host-Variablen	1400
Fehlerbehandlung	1403
Überblick	1403
Returncode-Analyse	1404
SQLCA - SQL Communication Area	1404
Auszug einiger relevanter SQLCODEs und SQLSTATES	1405
WHenever	1406
DSNTIAR, DSNTIAC	1406
GET DIAGNOSTICS-Statement	1407
Behandlung schwerer Fehler	1409
13.5.4 Cursor-Konzept	1410
Übersicht der Sprachmittel	1410
Überblick über die Cursor-Typen	1413
Ist die Result Table des Cursors read-only oder updateable?	1414
Non-Scrollable Cursor	1416
Deklaration des Cursors: DECLARE CURSOR	1416
Eröffnen des Cursors: OPEN	1418
Bereitstellen einzelner Zeilen der Result Table: FETCH	1418
Positioned Update und Delete: WHERE CURRENT OF	1419
Schließen des Cursors: CLOSE	1420
Explizites Schließen des Cursors	1420
Implizites Schließen des Cursors	1420
Statische und dynamische Result Table und ableitbare Integritäts- und Performance-Probleme beim Cursor-Konzept	1421
Mengenverarbeitung auf Realtime-Daten: das ewige Rätsel	1423
Scrollable Cursor	1425
Positionierungs-Möglichkeiten in der Result Table	1425
Cursor- und FETCH-Typen und ihre Sichten auf die Daten der Result Table	1426
Statischer Scrollable Cursor (Statisches Cursor-Modell)	1427
Deklaration des Cursors: DECLARE CURSOR .. SCROLL ..	1427
Informationsbereitstellung eines Static Scrollable Cursors	1428
Manipulationen über die Result Table eines Static Scrollable Cursors	1430
Dynamischer Scrollable Cursor (Dynamisches Cursor-Modell)	1431
Deklaration des Cursors	1431
Verarbeitung einer Zeilen-Gruppe (Rowset)	1432
Einlesen eines Rowsets: FETCH FOR n ROWS	1432
COBOL-Programm-Beispiel eines Rowset-Cursors mit GET DIAGNOSTICS	1433
Scrollable Cursor: Positionierung mit Rowsets	1437
Einfügen eines Rowsets: INSERT .. FOR n ROWS	1438
Analyse der Rowset-Verarbeitungs-Ergebnisse mit GET DIAGNOSTICS	1438
Einfluss des UOW/UOR-Konzepts auf die Cursor-Verarbeitung	1439
Zusammenfassung der relevanten Cursor-Typen und der zu definierenden Klauseln	1441
13.6 Besondere Programmier-Techniken unter DB2	1442
13.6.1 Effiziente Bereitstellung einer Result Table	1442
Relevante WHERE- und ORDER BY-Komponenten	1443
Aufsetzen auf einen Composite Key	1444

Regeln für das Aufsetzen	1445
13.6.2 Online-Anwendungen	1446
Online-Parallel-Update	1446
Online-Browsing/Blättern	1449
13.6.3 Batch- und Multi-User-Anwendungen	1450
Langlaufende oder ressourcenintensive Anwendungen	1450
UOW und UOR	1451
Die Lebensdauer einer UOW	1451
Wirkungen beim Abschluss einer UOR	1451
Einsatz von COMMIT (explizit oder implizit)	1452
Probleme, wenn kein COMMIT eingesetzt wird	1452
Was sind Kandidaten für die COMMIT-Schreibung?	1453
Konsequenzen des COMMIT-Einsatzes	1454
Unterstützung von Rollback und Restart der Batch-Programme	1455
Empfehlungen zum Design von Batch-Langläufern	1456
13.6.4 Anwendungen in verteilten Datenbankumgebungen	1457
Zugriffstypen und Protokolle	1457
DB2-Privat-Protokoll-Zugriff und DRDA-Zugriff	1457
Two-Phase-Commit (2-Phasen) in einer verteilten Datenbank-Umgebung	1458
Syntax-Anforderungen unterschiedlicher Systeme	1458
Programmvorbereitungen, Pläne und Packages	1459
SQL-Statement-Einsatz in Programmen	1460
Vorteile und Beschränkungen der Nutzung von dreiteiligen Objektnamen	1462
Behandlung von LOB- und XML-Werten bei remote Zugriffen	1462
Performance- und Integritäts-Aspekte	1463
Block-Fetch	1464
Limited und Continuous Block-Fetch	1465
13.6.5 Behandlung von LOB- und XML-Werten	1466
Grundsätzliche Verarbeitungsmöglichkeiten für LOB-Werte	1466
LOB-Daten-Typen und ihre Host-Variablen-Repräsentation	1468
LOB-Materialisierung	1470
LOB-Referenzierung mittels LOB-Locator	1471
LOB-Verarbeitung mittels File-Referenz-Struktur	1472
Grundsätzliche Verarbeitungsmöglichkeiten für XML-Werte	1473
XML-Funktionen	1474
Bereitstellungs-Techniken für XML-Daten	1475
SQL-XML-Funktionen: Beispiele für Komposition und De-Komposition	1476
XML-Daten-Typen und ihre Host-Variablen-Repräsentation	1477
13.6.6 Entwicklung und Einsatz von Stored Procedures	1479
Programmtechnische Umsetzung	1479
Aufruf-Varianten einer Stored Procedure	1479
Parameter der Stored Procedure	1479
Codier-Beispiele: Aufruf und Parameterübergabe	1480
Stored Procedure ohne Result Sets	1480
13.6.7 Entwicklung und Einsatz von User-defined Functions	1482
Überblick der Nutzungsmöglichkeiten	1482
Aufruf-Varianten einer externen Funktion	1483
Programmtechnische Besonderheiten	1484

Scratchpad-Bereich	1484
Parallellauf-Besonderheiten	1484
Parameter der User-defined Function	1484
Codier-Beispiele: Aufruf und Parameterübergabe	1485
External Scalar Function	1485
External Table Function	1487
13.6.8 Entwicklung und Einsatz von Triggern	1489
Codier-Beispiele: Aufruf mit Übergabe einer Transition-Table	1489
13.6.9 MQ-Series-Funktionen	1490
13.6.10 Session-Variablen und Scalar-Funktion GETVARIABLE	1493
13.6.11 Behandlung unterschiedlicher Codier-Sets für Host Variablen	1495
13.7 CAF - Call Attachment Facility	1496
13.7.1 Vorteile der CAF-Nutzung	1496
13.7.2 Entwicklung eines CAF-Programms	1496
13.7.3 Nutzbare CAF-Funktionen	1497
Implizite Connection	1498
Explizite Connection	1498
Generelle Ausprägungen einer CAF-Connection	1498
13.8 RRSF - Recoverable Resource Manager Services Attachment Facility	1499
13.8.1 Vorteile der RRSF-Nutzung	1499
13.8.2 Entwicklung eines RRSF-Programms	1499
13.8.3 Nutzbare RRSF-Funktionen	1500
Abschluss einer UOW	1501
13.9 Dynamic SQL	1502
13.9.1 Abgrenzung zu Static SQL	1502
13.9.2 Verhalten von Dynamic SQL-Statements (Statement Behavior)	1503
13.9.3 Funktionsspektrum von Dynamic SQL	1504
Manipulationen und Informationsanforderungen	1504
Parameter Markers (?)	1505
SQLDA - SQL Descriptor Area	1506
Struktur-Komponenten der SQLDA	1507
Beispiele der vom Precompiler/Preprozessor generierten SQLDA-Host-Variablen-Strukturen	1508
13.9.4 Manipulationen (Non-SELECT)	1509
Feste Parameter	1509
Variable Parameter	1511
13.9.5 Informations-Anforderungen (SELECT)	1512
Feste Parameter: Fixed-List-SELECT	1512
Variable Parameter: Varying-List-SELECT	1512
13.9.6 Statement Caching	1515
Prepare-Typen und -Ausführungsarten	1516
Sperr-Dauer bei dynamischen SQL-Statements	1517
13.9.7 Resource Limit Facility (RLF) - DB2 Governor	1518
13.9.8 Vor- und Nachteile von Dynamic SQL	1519
13.10 Programmier-Empfehlungen	1520
14 SQL-Performance	1523

14.1 Grafischer Wegweiser durch das Kapitel 14 - SQL-Performance	1523
14.2 Grundlagen der SQL-Performance	1524
14.2.1 Einleitung, Ansätze der Optimierung	1524
Leistungsbeeinflussende Faktoren	1524
Optimierungs-Möglichkeiten und -Maßnahmen	1525
14.2.2 Komponenten der Datenbeschaffung (Stages)	1527
Aufgabenteilung von RDS, DM und BM	1527
14.2.3 DB2-Tools zur Unterstützung von Performance-Maßnahmen	1529
Übersicht	1529
Statistikdaten des Katalogs	1531
DB2-Trace	1532
Leitlinien für Traces	1532
Trace-Typ	1533
14.2.4 Thread-Management	1535
Installations-Parameter für das Thread-Management	1535
Die Rolle von Prozess, Enklave, Transaktion, Thread, TCB und SRB	1536
Allied Thread	1537
Thread-Lebenszyklus	1537
Thread-Wiedernutzung (reuse)	1538
DBAT - Database Access Thread	1538
14.2.5 Die relevanten Zeiten einer Statement-Abwicklung	1540
DB2-Accounting-Zeiten	1540
14.2.6 Parallel-Verarbeitung	1542
Parallel-Verarbeitungs-Typen	1542
Kandidaten für Parallel-Verarbeitung	1542
Query-Typ: CPU bound oder I/O bound	1544
Effekte der Parallel-Verarbeitung	1544
Aktivierung der Parallel-Verarbeitung	1544
Voraussetzungen für die Parallel-Verarbeitung	1545
Restriktionen bei der Nutzung der Parallel-Verarbeitung	1545
14.3 Zugriffspfadanalyse des DB2-Optimizers	1546
14.3.1 Aufgaben des Optimizers	1546
14.3.2 Parsing und Parse Tree	1547
14.3.3 Optimization	1548
Bedingte Verarbeitungsanweisung (Prädikat)	1548
Erläuterung	1548
Prädikat-Typen	1549
Prädikat-Kategorien	1549
Filterung	1550
Inhaltliche Verteilung von Datenwerten	1551
Filter-Faktor (FF)	1551
Aufgabenstellung und Wirkung	1551
Filter-Faktor-Typen	1552
Filter-Faktor für gleichmäßige Streuung: Einfache/Simple Prädikate	1553
Filter-Faktor für gleichmäßige Streuung: Interpolations-Formeln für Range-Prädikate	1553
Filter-Faktor für gleichmäßige Streuung: Default-Filter-Faktoren für Interpolation-Formeln (Input-Variablen)	1554
Filter-Faktor für ungleichmäßige Verteilung: Frequenz- (FREQUAL) und Werteverteilungs-Statistiken (KEYCARD) für Spalten oder Spalten-Gruppen	1555
Filter-Faktor für ungleichmäßige Verteilung: Häufigkeitsverteilung (HISTOGRAM) für Spalten oder Spalten-Gruppen	1555

STAGE1, STAGE2 und INDEXABLE	1556
Indexable Prädikate	1559
Komplexe Prädikatvorgaben	1560
Column-Correlation	1561
Reihenfolge der Prädikat-Bewertung	1562
Aufwands-Kalkulation der SQL-Anforderung	1563
Prozess-Kostenschätzung	1563
Auswahl des Zugriffspfades	1563
IO-Kostenermittlung	1564
Timeron	1565
14.4 DB2-Zugriffspfade auf die Daten	1566
14.4.1 Analysemöglichkeit der Zugriffspfade	1566
Rahmenbedingungen	1566
EXPLAIN und PLAN_TABLE	1567
Einsatz von Input-Variablen beim EXPLAIN	1568
Beeinflussung der Optimizer-Entscheidungen durch OPTHINT	1568
EXPLAIN und DSN_STATEMNT_TABLE	1569
14.4.2 I/O-Zugriffs-Typ	1570
Sequential Prefetch und Dynamic Prefetch	1571
List Prefetch	1572
14.4.3 Einfache Zugriffspfade	1574
Tablespace Scan (File Page Set Scan)	1576
Index-Zugriffe	1577
Matching Index Scan	1577
Non-Matching Index Scan	1579
Equal Unique Index Access	1580
IN-List Index Scan	1581
One-Fetch Access	1582
Index-Only Access	1583
Multiple Index Access	1584
Multiple DocID Index Access über XML-Indizes	1585
Direkt-Zugriff auf die Daten unter Umgehung von Indizes	1587
Direkter Zeilen-Zugriff über die ROWID	1587
Direkter Zeilen-Zugriff über die RID-Funktion	1588
14.4.4 Komplexe Zugriffspfade	1589
Übersicht	1589
Join	1590
Composite Table und New Table, Outer Table und Inner Table	1590
Ausweis von Joins in der PLAN_TABLE	1591
Join-Typ	1592
Inner Join	1592
Outer Join	1593
Historische Entwicklung der Outer Joins	1595
Joins im Bereich des Data Warehouses: Star Join oder Paarweiser Join für Star-Schema und Snowflake-Schema	1597
Wann wird der Star Join genutzt?	1599
Ausweis eines Star Joins in der PLAN_TABLE	1600
Wann wird der Paarweise Join genutzt?	1601
Ausweis eines Paarweisen Joins in der PLAN_TABLE	1602
Klassische Join-Methoden	1603
Nested Loop Join	1603

Merge Scan Join	1606
Hybrid Join	1608
Komplexe Query-Typen	1610
Correlated und Non-Correlated Query-Typen	1610
Query-Block	1611
Der Nummerierungs-Wahnsinn der QBLOCKNO	1611
Subquery	1612
Non-correlated Subquery	1613
Correlated Subquery	1615
Table Expression	1617
Common Table Expression	1617
Rekursives SQL	1618
Nested Table Expression	1619
Scalar Full-Select	1620
SELECT FROM FINAL/OLD TABLE mit Data-Change-Statement	1621
Set-Operator	1623
Trigger-Package	1624
Sortierung	1625
Einige Worte zum Scrollable Cursor	1626
Parallel-Verarbeitung	1627
Materialisierung	1628
14.4.5 Beeinflussung der Zugriffspfad-Entscheidungen	1629
Statement-Modifikationen durch den Optimizer	1629
Prädikat-Modifikationen und Einfügungen	1629
Materialisierungs-Entscheidung	1629
Modifikationen der Join-Verarbeitung	1630
Regeln der Transformation von Subqueries in Join	1631
AQR - Automatic Query Re-Optimization	1632
Nutzungs-Potential von MQT und AQR	1636
Vorgabe von Zugriffspfad-Empfehlungen für den Optimizer (OPTHINT)	1637
Möglichkeiten für Anwendungsentwickler	1640
Veränderung von Objekt-Strukturen	1640
Manipulation der Statistiken	1643
Umschreiben von SQL-Statements	1644
OPTIMIZE FOR n ROWS	1646
Die Wirkungen von Host-Variablen beim BIND	1647
BIND-Parameter REOPT und seine Auswirkungen	1648
14.5 Zusammenfassung der relevanten Performance-Komponenten	1649
14.5.1 DB2-interne Automatismen	1649
Zugriffspfad-Effizienz	1649
Mögliche Zugriffspfade	1652
14.5.2 Systemgenerierungs- und Installations-Optionen	1653
14.5.3 Objekt-Definitionen	1655
14.5.4 Anwendungs-Design und -Programmierung	1657
SQL-Query-Empfehlungen	1658
Was tun bei langlaufenden Statements?	1659
14.6 Testdaten für die Performance-Tests	1660
14.6.1 Physisches Daten-Modell für Performance-Tests	1660
14.6.2 Metadaten-Übersicht der Test-Objekte	1661

14.6.3	Beispiel-SQL-Statements zur Einfügung der Daten	1665
14.6.4	Beispiel-Ausdrucke der Tabellen	1666
14.6.5	Beispiel-SQL-Statement zur Analyse der PLAN_TABLE für die Performance-Tests	1668