

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
Relationales Modell	3
Grafischer Wegweiser durch das Kapitel 1 - Relationales Modell	3
Allgemeine Entwicklungsgeschichte	4
Entwicklung innerhalb IBM	4
Entwicklung außerhalb IBM	4
Produkt-Kompatibilität	5
Kompatibilität zwischen den IBM-DB2-Produkten	5
Kompatibilität zwischen Fremdhersteller-Produkten und DB2-Produkten	5
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
Version 9 nach GA - nach der generellen Verfügbarkeit	28
Version 10 - DB2 10 for z/OS	29
Version 10 nach GA - nach der generellen Verfügbarkeit	38
Version 11 - DB2 11 for z/OS	39
Version 11 nach GA - nach der generellen Verfügbarkeit	44
Version 12 - DB2 12 for z/OS	45
Basis-Release: Function Level 500	45
Function Level 501	64
Terminologie	65
Informationsablage in Tabellenform	65
Tabelle (Table, Relation)	65
Spalte/Column (Attribut)	65
Zeile/Row (Tupel)	66
Datenwert/Value	66
Primary-Key (PK)	67
Foreign-Key (FK) und referenzielle Integrität (RI)	67
Abbildung der Primary- und Foreign-Key-Beziehungen (Beispiel-Tabellen des Buches und ihre logischen Beziehungen)	68
Informationelle referenzielle Integrität (Informational RI)	69

Index	70
Daten-Repräsentation und -Ablage	71
Informationsbeziehungen durch Wertehalte	71
Relationale Sprachschnittstelle	72
SQL	74
QUEL	75
QBE	76
Charakteristiken eines relationalen DBMS	77
Anforderungen an ein relationales DBMS	77
Codd'sches Anforderungsprofil - RM/V2	77
Zusammenfassung wesentlicher Zielsetzungen, Leistungen und Schwächen des Relationen-Modells	87
Zielsetzungen des Relationen-Modells	87
Standards im Bereich des Relationen-Modells	87
Schwächen des Relationen-Modells	88
Vergleich der Datenhaltungs- und Bereitstellungs-Konzepte	89
Gruppierung konventioneller Datenhaltungs- und Bereitstellungs- und DBMS-Typen	89
Terminologievergleich konventionelle Systeme - RDBMS	89
Problembereiche konventioneller Datenhaltungs- und DBMS-Typen	90
Strukturierte DBMS	90
Lineare DBMS (relational-orientierte Systeme)	91
Komplexe DBMS: Objektorientierte Datenbank-Systeme (OODBMS)	93
Speicherung und Verarbeitung von komplexen Objekten (LOBs / Large Objects)	96
Die Abbildung der Realität in der IT durch strukturierte und unstrukturierte Modelle	96
Ein komplexes Objekt ist immer zerlegbar (zumindest theoretisch - praktisch manchmal nicht empfehlenswert)	97
Der Einfluss der Zeit auf komplexe Objekte	98
Kategorisierung komplexer Objekte	99
Speicherung komplexer Objekte mit DB2: Large Objects (LOBs)	100
Problematiken beim Design und der Speicherung komplexer Objekte	101
Speicherung und Verarbeitung von XML-Dokumenten	102
XML-Terminologie	102
Ein XML-Dokument unterliegt einem strukturierten hierarchischen Modell	104
Zielsetzung und Aufgabenstellung von XML	105
Sind die Ziele des Relationen-Modells mit XML vergleichbar?	106
Eine völlig unterschiedliche Herangehensweise	106
Unterschiedliche Eignungen	107
Fazit	108
Standardisierung von Datenbank-Methoden und Techniken	110
Standardisierungs-Organisationen	110
Wichtige allgemeine Standards	110
Datenbank-Standards	111
ANSI/SPARC Architektur oder auch Drei-Schema-Architektur	111
SQL-Standards	112
SQL-Standard: SQL 89	115
SQL-Standard: SQL92 (SQL2)	115
SQL-Standard: 1999/SQL3 (grober Auszug)	117

SQL-Standard: 2003 (relevanter Auszug)	118
SQL-Standard: 2006 (relevanter Auszug)	120
SQL-Standard: 2008 (relevanter Auszug)	121
SQL-Standard: 2011 (relevanter Auszug)	122
SQL-Standard: 2016 (relevanter Auszug)	123
Die Auswirkungen der bisherigen SQL-Standardisierungsmaßnahmen	124
DB2-System-Aufbau	125
Grafischer Wegweiser durch das Kapitel 2 - DB2-System-Aufbau	125
DB2-System-Komponenten und Zusatz-Produkte	126
DB2-System-Komponenten	126
DB2-Zusatz-Produkte von IBM	127
Zusatz-Features	127
Objekt- bzw. Daten-Management-Tools	128
Sonstige Tools diverser Software-Hersteller	129
Entwicklungswerkzeuge für die PAEs	129
Verwaltungswerkzeuge für die DBADMs	129
Betriebssystem-Komponenten	131
System-Schichten von z/OS	131
DB2-Subsystem (Lokation oder Server)	132
DB2-Adressraum-Konzept	134
Sonstige Betriebssystem-Komponenten	135
Trägersysteme (Allied Agents)	135
DB2-System-Adressräume	136
System-Dienste (System Services - MSTR = Master)	136
Datenbank-Dienste (Database Services - DBM1 = Database Manager)	136
Internal Resource-Lock-Manager (IRLM)	136
Administrativer Task Scheduler (ADMT)	136
Distributed Data Facilities (DDF)	137
Adressräume für Routinen (Stored Procedures und/oder User-defined Functions)	138
Allied Agents: Ankoppelbare Trägersysteme	139
Attachment-Facility, Connection und Thread	139
Zugriff auf DB2-Ressourcen aus Anwendungen heraus	140
Schnittstellen zu den Trägersystem-Typen	140
Batch-Schnittstelle mit CAF (DSNALI)	140
CICS-Schnittstelle (DSNCLI)	140
TSO-Schnittstelle (DSNELI)	140
Recoverable Resource Manager-Schnittstelle (DSNRLI)	140
Universelle Schnittstelle (DSNULI)	140
IMS-Schnittstelle (DFSLI000)	141
Dateiverarbeitungsmöglichkeiten der Trägersystem-Typen	141
Verteilung (distributing) und gemeinsame Nutzung (sharing) der DB2-Daten	142
Verteilung der DB2-Daten (distributed data)	142
Die Client-Server-Architektur (C/S)	143
DRDA - Distributed Relational Database Architecture	144
DRDA - Level	145
Remote-Zugriff über dreiteilige Objekt-Namen/ALIAS oder nach CONNECT	146
DB2-Privat-Protokoll und DRDA-Protokoll	147

Technologische Grenzen der Verteilung von Daten	147
Gemeinsame Nutzung der DB2-Daten (DS - Data Sharing)	148
Betrieb mehrerer z/OS-Systeme	148
Data-Sharing: Group und Member	148
Argumente für Data Sharing	149
Grundsätzliche Anforderungen an ein Data-Sharing-Verfahren	150
Technische Lösung des DB2 Data Sharings	151
Data Sharing benötigt einen Parallel Sysplex	151
Aufgaben-Abgrenzung zwischen Members und dem Coupling Facility	152
Speicher-Strukturen einer Data Sharing Group	153
Die Konsistenzbewahrung bei Data Sharing (DS)	154
Sicher is sicher: Struktur-Duplizierung und Duplexed GBP	155
Einflüsse einer Data Sharing Group auf die System-Administration	156
Objekte und Namenskonventionen einer Data Sharing Group	157
Gruppen-Objekte und ihre Namen	157
Member-Objekte und ihre Namen	158
DB2-Ressource-Management durch den DB2 Administrativen Task Scheduler	159
Aufgabenstellung	159
Komponenten des Task Schedulers	159
System-Architektur	159
ADMT-DB2-System-Komponenten	160
Automatisierung von Utilities im Rahmen der Administrations- Scheduler Tasks (Autonomic Stored Procedures)	161
Planbare Ressourcen	162
Architektur in einer Data Sharing Group	163
Synchronisation mehrerer Ressource-Manager: RRS - Resource Recovery Services	164
RRS-Aufgabenstellung	164
Programm-Typen für die RRS-Synchronisations-Aufgaben	165
DB2-Daten-Objekt-Typen	166
Meta-Objekt-Typen für die Speicherung von Daten	166
Verwaltung der Meta-Daten	167
Anlegen, Änderung und Löschung von Meta-Daten	167
CREATE - Anlegen von Meta-Daten	167
ALTER - Änderung von Meta-Daten	168
Online-Schema-Evolution: Realtime-Änderung	168
DROP - Löschung von Meta-Daten	168
Schema, Eigentümer und Ersteller von Daten-Objekt-Typen	169
Autorisierungs-Id, Schema, Ersteller (creator) und Eigentümer (owner)	169
Das Zusammenspiel der Spezialregister	170
Versionsführung für Meta-Objekte (Versionierung)	171
Wie kann man sich eine Versionsführung vorstellen?	171
Wie muss man sich die DB2-Versionenführung vorstellen?	172
Relevante Katalog-Informationen der Versionsführung	173
Test-Beispiele der DB2-Versionenführung	174
Im DB2 existieren vielfältige Versionen und Versionierungen	175
Katalog-Änderungen im Schwebezustand (Pending Changes)	176
DDL_MATERIALIZATION-Einflussnahme auf sofortigen Vollzug der Änderung oder Pending-Status (nur für Änderungen von Spalten-Charakteristiken)	179
Aspekte des sofortigen Vollzug der Änderung oder des Setzen eines Pending-Status (bei Änderungen von Spalten-Charakteristiken)	181
Domain-Konzept und Daten-Typen	182
Grundsätzliche Überlegungen und Ausgangssituation	182

Builtin Daten-Typen	182
LOB = Large Object	183
XML = XML-Dokument	183
Benutzerdefinierte Daten-Typen (UDT = User-defined Data-Types)	184
Definition der Daten-Typen	185
Array Type	186
Vergleich von Einfachen/Ordinary Arrays und Assoziativen Arrays	187
Überblick der Array-Komponenten	188
Distinct Type	189
Automatisch generierte Cast-Funktionen	189
Nutzungsmöglichkeit der Distinct Data-Types	191
User-defined LOB-Data Types	191
User-defined XML-Data Types	191
Vergleichbarkeit und Konvertierbarkeit der Daten-Typen	192
Konvertierungen zwischen verschiedenen Built-in-Daten-Typen	193
Variablen-Typen	194
Lokale Variablen	195
Globale Variablen	196
Benutzer-orientierte Daten-Objekt-Typen	197
Tabelle/Table (Base Table)	197
Definition der Tabellen-Spalten	199
Die Rolle des PKs	201
Einrichtung referenzieller Beziehungen (referential constraint)	201
Einrichtung von Check-Constraints	203
Maschinelle Vergabe von Werten durch DB2	204
Nummern-Vergabe: ROWID, IDENTITY, SEQUENCE	204
Verwaltung von technischen Gültigkeitszeiträumen: PERIOD-Klausel für SYSTEM_TIME	204
GENERATED-Klausel für die automatische Generierung von Werten (GENERATED ALWAYS und GENERATED BY DEFAULT)	205
Zuordnung einer Tabelle zu einem Tablespace	209
Definition einer Partitioned Table (table-kontrollierte Partitionierung)	210
Änderung von Tabellen- und Spalten-Charakteristiken	212
Änderung des Daten-Typs einer Spalte	213
SQL-DDL-Unterstützung für die Basis-Tabelle	217
Archiv-Tabelle (Archive Table)	220
Aufgabenstellung und Konzept	220
Vorteile des Konzeptes	221
Nachteile des Konzeptes	221
Funktionsumfang und Steuerung	221
Built-in Globale Variablen SYSIBMADM.GET_ARCHIVE und SYSIBMADM.MOVE_TO_ARCHIVE	221
Manipulations-Wirkungen aufgrund der Globalen Variablen SYSIBMADM.MOVE_TO_ARCHIVE	222
Daten-Bereitstellungs-Wirkungen aufgrund der Globalen Variablen SYSIBMADM.GET_ARCHIVE	222
Dokumentation des Archiv-Konzeptes im Katalog	223
Können auch LOB- und XML-Werte archiviert werden?	224
Beispiel-Szenario für die Archivierung	225
Restriktionen des Archiv-Konzeptes	230
Clone Table	231
Aufgabenstellung und Funktionsumfang	231
Metadaten und Daten-Instanzen von Base Table und Clone Table	232
Metadaten von Base Table und Clone Table (Katalog-informationen)	232
Daten-Instanzen von Base Table und Clone Table	233
Daten-Definition von Base Table und Clone Table mittels SQL-DDL	234
Daten-Manipulationen und Utility-Unterstützung einer Clone Table	235

EXCHANGE-Statement: Austausch der Daten-Instanzen zwischen Base Table und Clone Table		236
Löschen der Clone Table		236
Restriktionen der Base Table und der Clone Table		237
Fazit		237
Temporal Tables (Tabellen mit Gültigkeitszeiten)		238
Einleitung und Überblick		238
Gültigkeits-Zeitleisten-Behandlung Inklusiv oder Exklusiv: Das ist hier die Frage		239
Unterstützte Datenhaltungs-Konzepte		240
Konzeptioneller Überblick des Modells		241
Fachliche Gültigkeit (BUSINESS_TIME)		241
Manipulations-Regeln für Daten mit fachlichen Zeitleisten		243
Technische Gültigkeit (SYSTEM_TIME)		244
Technische Lösung		246
Technische Komponenten der Datenhaltung		246
Katalog-Metadaten für Temporal Tables		248
Konzept der Historisierung der Daten		249
Ausgabe von Zusatzinformationen bei Temporal Tables (Audit-Informationen) - ON DELETE ADD EXTRA ROW		250
Fachliches Zeitkonzept (PERIOD BUSINESS_TIME)		252
Grundsätzliche Aspekte	252	
DDL-Definitionen	253	
Meta-Informationen des Katalogs	254	
Datenzustände für die Test-Beispiele	255	
SQL-Beispiele: Spezialregister CURRENT TEMPORAL BUSINESS_TIME mit NULL-Wert		256
Manipulations-Beispiele	256	
Beispiele für Informations-Bereitstellungen (SELECT)	258	
SQL-Beispiele: Spezialregister CURRENT TEMPORAL BUSINESS_TIME mit konkretem Wert		261
Manipulations-Beispiele	261	
Der problematische Einsatz des Spezialregisters CURRENT TEMPORAL BUSINESS_TIME (Zusammenfassung)		264
Bi-Temporale Tabellen mit fachlichen Gültigkeiten - PERIOD BUSINESS_TIMES und technischen Gültigkeiten - PERIOD SYSTEM_TIME und Versionierung (History-Table)		269
Grundsätzliche Aspekte	269	
DDL-Definitionen	271	
Meta-Informationen des Katalogs	273	
Datenzustände für die Test-Beispiele	276	
SQL-Beispiele: Spezialregister CURRENT TEMPORAL SYSTEM_TIME mit NULL-Wert		278
Manipulations-Beispiele	278	
Beispiele für Informations-Bereitstellungen (SELECT)	281	
SQL-Beispiele: Spezialregister CURRENT TEMPORAL SYSTEM_TIME mit konkretem Wert		282
Manipulations-Beispiele	282	
Referential Integrity von Temporal Tables (Temporal Referential Constraint)		285
Einführung		285
DB2-Rules für RI von Temporal Tables		286
Komponenten für die Definition von Referential Integrity für Temporal Tables		288
Temporal RI-Beispiel-Szenario 1: Manipulationen nur in der Parent Table (Beispiel: Inklusiv-Inklusiv)		289
Detail-Ablauf	290	
Temporal RI-Beispiel-Szenario 2: Manipulationen in der Parent- und Child-Table (Beispiel: Inklusiv-Inklusiv)		291
DB2-Temporal Tables - Zusammenfassung		293
Grundsätzliche Konzepte und unterstützte Funktionalitäten		293
Zusammenfassende Manipulations-Regeln für temporale SQL-Statements, wenn die temporalen Spezialregister zu berücksichtigen sind		295
Positive Aspekte		296
Negative Aspekte		296
Fazit		297
Temporäre Tabelle (Temporary Table)		298
CGTT - Created Global Temporary Table		299
DGTT - Declared Global Temporary Table		300

Temporäre Result Table/Virtual Table	302
Temporäre Result-Table-Typen	302
Positionierung oder Materialisierung einer temporären Result Table	304
DB2-View (Virtual Table)	305
Ist die View-Result-Table read-only oder updateable?	307
View auf View - macht das Sinn?	309
Anwendungs-Unabhängigkeit von physischen Sichten	310
Individuelle Views oder generalisierte Views	311
Besonderheiten bei der Nutzung von Temporalen Tabellen durch DB2-Views	312
Nachteile und Grenzen des View-Konzepts	314
Empfehlungen zum View-Konzept	314
MQT - Materialisierte Query Table (Materialisierter View bzw. Materialisierte Result Table)	315
Komponenten einer materialisierten Query Table (MQT)	316
Definition der MQT	317
System- oder benutzer-kontrollierte MQT	319
Einrichtung einer Hash-Organisation für den Direkt-Zugriff mit Hash-Methode	320
Überblick	320
Hash-Space: Aufteilung in Fixed Hash-Space und Overflow-Space	321
Katalog-Metadaten für Hash-Komponenten	323
Definition der Hash-Zugriffs-Methode	325
Neue Tabelle anlegen mit Hash-Organisation	325
Bestehende Tabelle mit Hash-Organisation ausstatten	327
SYNONYM (auslaufend/deprecated)	329
ALIAS	330
Maschinelle Vergabe einer lfd. Nr. durch DB2: SEQUENCE und IDENTITY	331
SEQUENCE	333
Systemorientierte DB2-Objekt-Typen	334
Index	334
Definition der Index-Spalten	335
Wann wird ein Index aufgebaut und wie wird er verwaltet?	335
Index-Typen	336
Basis-Index	338
Ein stinknormaler Index besteht aus	338
Der Clustering Index organisiert die Datenfolge	339
Partitioning Index und Partitioned Index	341
und nun alle Index-Typen auf einmal	344
DPSI: Data Partitioned Secondary Index	346
Extended Index	350
Scalar Expression Index (Index on Expressions)	351
Interne Organisation eines Indexes	353
Zuordnung eines Indexes zum Indexspace, den Page Sets und Datasets	353
B-Tree-Struktur eines Indexes	354
Änderung von Index-Charakteristiken	356
Löschung eines Indexes	357
Welche Vorteile bietet ein Index, welche Nachteile sind zu beachten?	358
Vorteile eines Indexes	358
Nachteile eines Indexes	359
Empfehlungen zum Index-Einsatz	360
Tablespace	361
Aufgabenstellung und relevante Zuordnungen	361
Die Page-Größe und ihre Einflüsse	363
Tablespace-Typen	364

Tablespace-Typen für die Datenhaltung von Basis-Daten	364
Tablespace-Typen für die Datenhaltung spezieller Daten	365
Konvertierung eines Tablespace-Typs in einen anderen	368
Vor- und Nachteile der Tablespace-Typen	369
Besonderheiten des Partitioned Tablespaces	371
Physische Charakteristiken pro Partition	371
4096 Partitions und die Konsequenzen	372
Dynamische Partitions: Hinzufügen einer neuen Partition	374
Einfügen einer neuen Partition am Ende des Tablespaces	375
Einfügen einer neuen Partition an jeder beliebigen Stelle	376
Dynamische Partitions: Rotation	377
Dynamische Partitions: Partition-Werte balancieren	379
Nachträgliche Änderungsmöglichkeiten eines Tablespaces	380
Indexspace	382
Aufgabenstellung und relevante Zuordnungen	382
Database	384
Storage Group	385
LOB-Objekte: Auxiliary Table, Auxiliary Index, LOB-Tablespace	386
XML-Objekte: XML Table, DocID- und NodeID-Index, XML-Index und XML-Tablespace	388
Unterstützung von Temporal Tables mit XML-Dokumenten	390
XML-Erweiterungen der DB2-Version 10	391
DB2-System-Ressourcen	392
DB2-Directory: Database DSNDB01	393
DB2-Katalog: Database DSNDB06	394
Datenspeicherung von System- und Benutzerdaten	394
Zwischenspeicher für Materialisierungen	395
Workfiles: Workfile Database (DSNDB07)	395
Utility-Arbeitsdateien	396
DB2-Log-Dateien	396
Bootstrap-Dateien (BSDS)	396
DB2-Bibliotheken	396
Routinen: Stored Procedures und User-defined Functions	397
Grafischer Wegweiser durch das Kapitel 2 - Routinen: Stored Procedures und User-defined Functions und Trigger	397
Terminologie und Übersicht	398
Routinen-Typen	398
Aufgabenbereiche und Komponenten von Routinen	399
Unterschiede zwischen Externen Routinen und SQL Routinen	400
SQL-Control-Statements/SQL-PL (SQL-Procedure Language) - (SQL/PSM - SQL/Persistent Stored Modules) für SQL-Routinen und Trigger	401
Kommunikations-Strukturen und Informations-Austausch bei Routinen	402
Überblick	402
Vorüberlegungen bei der Nutzung von Ressourcen bei Routinen	403
Von Routinen nutzbare DB2-Ressource-Kategorien	403
Identifikation von Stored Procedures und Functions	404
Schema, SQL-Pfad, Specific Name und External Name	404
Function Overloading und Function Resolution	404
Bewahrung der Daten-Integrität - Konzept der UOW und UOR bei Einsatz von Routinen	405
Die grundsätzlichen Prinzipien der Daten-Integrität	405
Einsatz von COMMIT und ROLLBACK	407
Autonome UOW/UOR einer Stored Procedure	408

Konzept	408
Der Einsatz von Stored Procedures	410
Grobe Zielsetzung einer Stored Procedure	410
Prozedur-Typen	411
Entwicklung einer Stored Procedure	412
Definition einer Stored Procedure	414
Namen der Stored Procedure	414
Parameter einer Stored Procedure	414
Übergabe der Parameterwerte	414
Parameter-Typen (Eingabe, Ausgabe oder beides)	416
NULL-Fähigkeit der Parameter	416
PARAMETER STYLE GENERAL, GENERAL WITH NULLS und SQL (Übergabe-Konventionen einer Stored Procedure)	417
Beispiel der Parameterübergabe zwischen PL/I- und COBOL	419
PARAMETER STYLE JAVA	420
Rückgabe einer Ergebnismenge (Query Result Set)	421
CALL-Aufruf-Varianten einer Stored Procedure	422
Statische CALL-Varianten	422
Dynamischer CALL (ODBC bzw. CLI)	422
External Procedure	423
Unterstützte Programmiersprachen	423
Entwicklungsschritte einer External Stored Procedure	423
Aufrufendes und aufgerufenes Modul	423
Compile und Linkage Editor	423
Definition der Stored Procedure im Katalog	424
SQL Procedure	425
Typen und unterstützte Funktionalitäten	425
SQL External Procedure	426
Unterstützte Programmiersprachen und Sprachmittel	426
Entwicklungsschritte einer SQL External Procedure	427
Data Studio oder Development Center	428
IBM-Stored Procedure DSNTPSMP	428
Explizite Definition mit der JCL-Prozedur DSNHSQL	429
Definition der SQL Stored Procedure im Katalog	429
SQL Native Procedure	430
Unterstützte Programmiersprachen und Sprachmittel	430
Einfaches Beispiel einer Native Procedure	431
Versions-Konzept einer SQL External Procedure	432
Entwicklungsschritte einer SQL Native Procedure	433
Definition der SQL Stored Procedure im Katalog	433
Ausführung einer Stored Procedure	434
Adressraum-Konzept	434
Starten und Stoppen von Stored Procedures	434
Aufruf einer remote Stored Procedure und Ablauf	435
Vor- und Nachteile von Stored Procedures	437
Liste der Standard-Stored Procedures (IBM-Supplied Stored Procedures)	438
Relative Performance-Gewichtung von Stored Procedures mit unterschiedlichen Programmiersprachen	445
Der Einsatz von benutzerdefinierten Funktionen (User-defined Functions)	446
Grobe Zielsetzung einer User-defined Function (UDF)	446
Funktions-Aufruf und -Ergebnis	446
Funktions-Typen	447
Entwicklung einer User-defined Function	450

Definition einer User-defined Function	452
Namen der User-defined Function	452
Inlined- oder Non-Inlined-Function (Compiled Function)	452
Parameter einer User-defined Function	453
Übergabe der Parameterwerte	453
Parameter-Typen (Eingabe oder Ausgabe)	453
NULL-Fähigkeit der Parameter	453
PARAMETER STYLE SQL oder JAVA (Übergabe-Konventionen für Funktionen)	454
Beispiel der Parameterübergabe zwischen COBOL und PL/I	456
Aufruf-Varianten einer User-defined Function	457
Externe Aufruf-Varianten der Funktion	457
Interne Aufruf-Typen (CALL-Typ bzw. FINAL CALL-Klausel)	457
Scratchpad: Zwischenspeicher für die internen Aufrufe	460
Thread- und TCB-Konzept für Funktions-Aufrufe	461
TCB-Nutzung beim Funktions-Aufruf	461
TCB-Konzept bei Parallel-Abwicklung	462
DETERMINISTIC oder NOT DETERMINISTIC - das ist hier die Frage	463
Sourced Function	464
Unterstützte Programmiersprachen	464
Entwicklungsschritte einer Sourced Function	464
Definition der Sourced Function im Katalog	464
External Function	465
Unterstützte Programmiersprachen	465
Entwicklungsschritte einer External Function	465
Aufrufendes und aufgerufenes Funktions-Modul	465
Compile und Linkage Editor	465
External Scalar Function	466
Definition der Scalar Function im Katalog	466
External Table Function	467
Definition der Table Function im Katalog	467
SQL Function	468
SQL Scalar Function	468
Definition der SQL Scalar Function im Katalog	468
SQL Table Function	469
Definition der SQL Scalar Function im Katalog	469
Ausführung einer User-defined Function	470
Adressraum-Konzept	470
Starten und Stoppen von User-defined Functions	470
Vor- und Nachteile von User-defined Functions	471
Vor- und Nachteile von External Scalar Functions	471
Vor- und Nachteile von SQL Scalar Functions	471
Vor- und Nachteile von External Table Functions	472
Vor- und Nachteile von SQL Table Functions	472
Mögliche Ressource-Nutzung	473
Unterstützte SQL-Funktionalität	473
Zugriff auf DB2-Tabellen	473
Unterstützung von LOB-Spalten und -Werten	473
Unterstützung von XML-Spalten und -Werten	473
Die Nutzung von User-defined Globalen Variablen in Routinen	474
Beispiel mit Nutzung in Stored-Procedure und Client-Anwendung	474
Nutzung von Array-Daten-Typen und Array-Funktionen in Routinen	475

Array-Typen	475
Nutzungsmöglichkeit	475
Einfaches Beispiel mit ARRAY_AGG-Funktion und UNNEST-Klausel im SELECT	475
Array-Funktionalität	476
Überblick der Nutzung	476
Beispiel-Szenario	477
Ordinary Array	477
Assoziativer Array	480
Zugriff auf externe Ressourcen	483
Remote DB2-Objekte	483
Andere relationale Datenbanken	483
CICS-Ressourcen	483
IMS-Ressourcen	484
WebSphere MQ-Ressourcen	484
Sonstige externe Ressourcen	484
Level-Konzept (Nested Routines)	485
Behandlung von Spezialregistern in Routinen	486
Fehlerbehandlung von Routinen	487
Auswahl relevanter Fehlercodes	488
Erforderliche Privilegien	489
Package- und Plan-Behandlung	490
Behandlung der externen Module	490
Behandlung von implizit erzeugten Packages	490
Wechsel der Package	490
Nutzung verschiedener Programm-Versionen	490
Testmöglichkeiten von SQL-Routinen	491
Allgemeine Problematik	491
Test-Tools (Debugger)	491
Besonderheiten beim Test	491
Routinen-Versionen einer SQL Procedure	491
Zusammenfassung der relevanten Klauseln für die Definition von Routinen	492
Abwicklungs-Komponenten und Zuordnung der DDL-Parameter-Wirkung	492
Ausführungs-Steuerung (Optionen) von Routinen	493
Übersicht der Options-Argumente und Zuordnungen zu den Routinen-Typen	493
x = Argument tritt auf	494
Besonderheiten und Konventionen für Java Applikationen	495
z/OS-Ausführungs-Umgebung	496
Überblick der beteiligten z/OS-System-Komponenten	496
RRS - Resource Recovery Services	497
Die Rolle im Rahmen von Routinen	497
RRSAF-Attachment: Statisches oder dynamisches Laden	498
LE - Language Environment	499
LE-Konzept und relevante Komponenten	499
LE-Run-time-Options	499
MSGFILE	499
RPTOPTS	500
RPTSTG	501
Optionen zur Kontrolle des Speichers	502

TEST und NOTEST	502
Sonstige LE-Besonderheiten	502
Workload-managed Adressräume (WLM-established)	503
WLM-Terminologie	503
WLM-Aufgabenstellung	503
WLM-Komponenten	503
WLM-Subsystem-Typen	503
WLM-Service-Definitions	504
WLM-Operations-Modus	504
WLM-DB2-Environment	505
DB2-Installations-Parameter für WLM	505
Parametrisierung für WLM	505
WLM-JCL-Beispiel	505
NUMTCB = Anzahl paralleler Tasks	506
Steuerung des WLM-Adressraums	507
Kontrolle und Abbruch aktiver Routinen	507
Wieviele WLM-Environments sind sinnvoll?	508
Privilegien für die Nutzung des WLM-Environments	508
Modul-Eigenschaften in der WLM-Umgebung	509
Threadsafe Module (reentrant)	509
Residente Module	509
Haupt- oder Unter-Routine	509
Austausch der ausführbaren Version einer Routine	510
Testmöglichkeiten in der WLM-Umgebung	511
Allgemeine Problematik	511
Test-Tools (Debugger)	511
Besonderheiten beim Test	511
Vorteile des WLM-Environments	512
Nachteile des WLM-Environments	512
Performance-Aspekte bei externen Modulen	513
Der Einsatz von Triggern	514
Zielsetzungen eines Triggers	514
Trigger-Terminologie	515
Beispiel eines Triggers	515
Trigger-Charakteristiken	515
Übersicht der Trigger-Typen und ihre Charakteristiken	517
Komponenten eines Triggers	519
Basic- und Advanced Trigger	520
Charakteristiken der beiden Trigger-Typen	520
Gegenüberstellendes Beispiel eines Basic- und eines Advanced Triggers	521
Dokumentation von Basic- und Advanced Trigger im Katalog	522
Änderungs-Möglichkeiten und der Varianten von Triggern	522
Aktivierungszeit: BEFORE	523
Allgemeine Regeln	523
Trigger-Beispiele: Before-Trigger	524
Abbildung des Einsatzes	524
Trigger mit Abweisung ungültiger Referenzen (RI-Erweiterung)	525
Trigger zur Einsetzung von Initialwerten bei Einfügungen	526
Trigger mit Übernahme redundanter Daten	526

Trigger mit Aufruf einer Funktion	526
Aktivierungszeit: AFTER	527
Allgemeine Regeln	527
Trigger-Beispiele: After-Trigger	528
Abbildung des Einsatzes	528
Verwaltung von Redundanzen	529
Trigger mit Übergabe einer Transition-Table an eine Funktion	530
Trigger mit Übergabe von Transition-Variablen an eine Funktion	530
Aktivierungszeit: INSTEAD OF	531
Allgemeine Regeln	531
Trigger-Beispiele: Instead of-Trigger	532
Abbildung des Einsatzes	532
Restriktionen eines Instead of Triggers	534
Empfehlungen für den Einsatz eines Instead of Triggers	535
Ereignis-Typen: INSERT, UPDATE und DELETE	536
Häufigkeit der Funktionsauslösung: ROW oder STATEMENT	536
Ablauf der Verarbeitung bei Einsatz eines Triggers	537
Trigger-Aktion	538
Trigger-Condition	538
Trigger-Body	538
Result Table des Triggers	539
Transition-Variablen	539
Transition-Tables	540
Cascading-Effekte und Trigger-Level bei After Triggern	541
Fehler-Behandlung	542
Fehler-Erkennung und -Signalisierung	542
Fehler bei der Ausführung der Trigger-Statements	543
Aufruf von Routinen aus einem Trigger heraus	544
Aufruf-Techniken	544
Behandlung von Spezialregistern	545
Nutzung sonstiger DB2-Ressourcen	546
Die Nutzung von User-defined Globalen Variablen in Triggern	546
Die Nutzung von Array-Daten-Typen und Array-Funktionen in Triggern	546
Trigger-Package	547
Erzeugen und Löschung einer Trigger-Package	547
Abhängigkeiten einer Trigger-Package	547
Trigger-Package für einen Basic Trigger	547
REBIND einer Trigger-Package für einen Basic Trigger	547
Trigger-Package für einen Advanced Trigger	548
REBIND einer Trigger-Package für einen Advanced Trigger	548
Trigger-Restriktionen	549
Vor- und Nachteile von Triggern	550
Methodische und technische Voraussetzungen für DB2	552
Grafischer Wegweiser durch das Kapitel 3 - Methodische und technische Voraussetzungen für DB2	552
Allgemeine Voraussetzungen für eine DB2-Einführung	553
Personelle Voraussetzungen	554
Objekt- (OA) bzw. Daten-Administration (DA)	554

Datenbank-Administration (DBA)	554
Organisatorische Voraussetzungen	555
Auswirkung auf die Anwendungs-Entwicklung	555
Technische Voraussetzungen für DB2 Version 12 for z/OS	557
Hardware	557
Betriebssystem-Software	558
Funktionsspezifische Abhängigkeiten	558
Betriebssystemnahe Standard-Software	558
Transaktions-Management	559
DRDA-Verbindungsmöglichkeiten	559
Web-Verbindungen	559
IBM Tools für das Utility-Management	559
Anwendungs-Entwicklungs-Tools	560
Programmiersprachen	560
Migrations-Pfad zur Version 12	561
Was ist neu im DB2?	563
Migration-Strategie ins DB2	565
DB2-Systemkonzept	566
Single-DBMS oder Multi-DBMS?	566
Abgrenzung Produktionssystem - Testsystem	567
System-Kategorien: Test, Übergabe und Produktion	567
System-Kategorie: Test	568
System-Kategorie: Übergabe in die Produktion (System-Test)	568
System-Kategorie: Produktion	569
Problematiken bei der Übernahme von Test in die Produktion	569
Isolierte Systeme haben trotzdem Kommunikations-Erfordernisse	569
Koordination der Objekte zwischen Systemen	570
Management der Versions-Änderungen (Change Management)	570
Spezielle DB2-Übernahme-Kriterien	570
Datenbank-Hierarchie und Nutzungsmöglichkeiten	571
Informations-Aktualität	572
Client-Server-Konzept (C/S)	573
Zentralisierung versus Dezentralisierung	573
C/S sollte in den Köpfen des Managements beginnen ...	573
C/S benötigt klare Konzepte	574
Argumente für eine dezentrale Organisationsform (Verteilung)	575
Argumente für eine zentrale Organisationsform	575
Fazit	575
Varianten der C/S-Verarbeitung	576
Verteilungsaspekte	577
Lokale Zuordnung der Ressourcen	577
Verfügbarkeit und Nutzbarkeit der Ressourcen	577
Konsequenzen einer Verteilung	578
Methodische Voraussetzungen	579
Abhängigkeiten zwischen Programmen und Daten	579
Probleme beim Einsatz konventioneller Datenbank-Systeme	579
Sind solche Probleme unter DB2 automatisch gelöst?	581

Abhängigkeit des SQL-Spracheinsatzes von DB2-Katalog-Informationen	581
Abhängigkeit bei fehlendem Schichten-Konzept	584
Lass die Datenbank loopen!	585
Auswirkung der Daten-Modellierung auf die Programm-Sicht	586
Sicht auf unnormalisierte Daten	587
Definition einzelner Spalten in Wiederhol-Elementen	587
Variable Zeilen mit VARCHAR- oder LOB-Definitionen	588
Speicherung von XML-Dokumenten	589
Sicht auf normalisierte Daten	590
Sicht auf Daten in der ersten Normalform (1 NF)	590
Sicht auf Daten in höheren Normalformen (2 NF, 3NF)	591
Abbau der Abhängigkeiten durch Methodik-Einsatz	592
Ebenen, Modell- und Ergebnis-Typen im Unternehmens-Modell	592
Auswirkungen auf das Anwendungs-Design	593
Schichten-Modell für Anwendungs-Programme	593
Separierung der Datenzugriffe von sonstigen Funktionen	594
Objektorientierte Verwaltung der Daten	595
Daten-Verwaltung immer über eine zuständige Funktion/Methode	596
Einsatzmöglichkeit von Standard-Zugriffs-Routinen	597
Abbau der Abhängigkeiten durch DB2-Techniken	598
Das Ebenen-Modell von DB2	598
Katalog-Konzept	599
Verlagerung von Programmlogik in den DB2-Katalog	599
Komponenten einer "aktiven Datenbank"	600
Nutzung von Stored Procedures, User-defined Functions und Triggern	601
Integritätsbewahrung als Aufgabenstellung für die Datenbank	601
Verlagerung sämtlicher datennaher Dienste in den DB2-Bereich	602
Argumente für und wider eine "aktive Datenbank"	603
Normungs-Effekte durch internationale Standards	604
Normungs-Effekt durch den SQL-Spracheinsatz	604
Open Database Connectivity (ODBC) und der CLI-Spracheinsatz	604
DB2-Sprachschnittstellen	605
Grafischer Wegweiser durch das Kapitel 4 - DB2-Sprachschnittstellen	605
Benutzergruppen und deren Anforderungen	606
DB2-Sprachschnittstellen für die Benutzergruppen	607
TSO-DSN-Commands: Übersicht	609
SQL-Statements	610
SQL-Statement- und Benutzer-Kategorien	610
DCL - Data Control Language	610
DDL - Data Definition Language	610
DML - Data Manipulation Language	611
SQL-Anwendungsformen und Sprach-Schnittstellen	613
Interactive SQL (DB2I)	613
Embedded SQL (EXEC-Level)	613
CALL-Level-Interface für C- und C++-Programme (CLI bzw. ODBC)	614
JDBC-API für Java-Anwendungen	614
SQLJ-API für Java-Anwendungen	614
Static SQL	615

Dynamic SQL	615
Übersicht der SQL-Statements und ihre Einsatzmöglichkeiten	616
DB2-Befehle/DB2-Commands	621
DB2-System-Commands	621
Admin-Scheduler-Commands	623
CICS-Commands	623
IMS-Commands	624
IRLM-Commands	624
DB2-Hilfsprogramme/DB2-Utilities	625
DB2-Online-Utilities	625
Utility-Control-Statements	626
Standalone Utilities/Service Aids	626
TSO-CLISTS und TSO-DSN-Subcommands	627
DB2I: Masken der interaktiven DB2-Oberfläche	628
Einstiegs-Menü (Primary Option Menu)	628
D - DB2I-Defaults	629
Default-Panel 1	629
Default-Panel 2	630
1 - SPUFI: SQL Processor Using File Input	631
SPUFI-Steuerungsmaske	631
SPUFI-RLF-Fehler-Maske	633
SPUFI-RLF-Warnungs-Maske	633
SPUFI-Defaults	634
Default-Panel 1	634
Default-Panel 2	636
SPUFI-Edit-Maske	637
SPUFI-Browse-Maske	638
2 - DCLGEN: Declarations Generator	639
Eingabemaske	639
Maske mit Zusatz-Optionen	640
Beispiele generierter Ausgaben	641
COBOL-Host-Variablen-Struktur	641
SEMINAR-Tabelle	641
Variante: Default-Struktur- und Spalten-Namen ohne Label	641
Variante: Default-Struktur- und Spalten-Namen mit Label	642
Variante: Individuelle-Struktur- und Spalten-Namen (Table-Name mit Numerierung)	643
Variante: Individuelle-Struktur- und konkreten Spalten-Namen	644
Alle Daten-Typen	645
PL/I-Host-Variablen-Struktur	647
SEMINAR-Tabelle	647
Variante: Default-Struktur- und Spalten-Namen ohne Label	647
Variante: Default-Struktur- und Spalten-Namen mit Label	648
Variante: Individuelle-Struktur- und Spalten-Namen (Table-Name mit Numerierung)	649
Variante: Individuelle-Struktur- und konkreten Spalten-Namen	650
Alle Daten-Typen	651
C-Programm-Host-Variablen-Struktur	653
SEMINAR-Tabelle	653
Variante: Default-Struktur- und Spalten-Namen ohne Label	653

Variante: Default-Struktur- und Spalten-Namen mit Label	654
Variante: Individuelle-Struktur- und Spalten-Namen (Table-Name mit Numerierung)	655
Variante: Individuelle-Struktur- und konkreten Spalten-Namen	656
Alle Daten-Typen	657
3 - Program Preparation	659
Steuerungsmaske	659
Program Preparation 'Compile, Link and Run Panel'	660
4 - Precompile	661
5 - BIND / REBIND / FREE	662
1 - BIND PLAN	663
Steuerungsmaske	663
Default Panel BIND PLAN	664
Connection Types Panel	665
Package List Panel	666
Ergebnis einer erfolgreichen BIND PLAN-Ausführung	666
2 - REBIND PLAN	667
Steuerungsmaske	667
Default-Panel	668
3 - FREE PLAN	669
Steuerungsmaske	669
4 - BIND PACKAGE	670
Steuerungsmaske	670
Default-Panel	672
Connection Types Panel	674
Path List Panel	675
Ergebnis einer erfolgreichen BIND PACKAGE-Ausführung	675
5 - REBIND PACKAGE	676
Steuerungsmaske	676
Default-Panel	678
Ergebnis-Protokollierung REBIND PACKAGE	680
Ergebnis einer nicht erfolgreichen REBIND PACKAGE-Ausführung	680
Ergebnis einer erfolgreichen REBIND PACKAGE-Ausführung	681
6 - REBIND TRIGGER PACKAGE	682
Steuerungsmaske	682
7 - FREE PACKAGE	684
Steuerungsmaske	684
6 - RUN - Aktivierung einer Programm-Ausführung im Foreground oder Batch	685
Anstoß über Panel	685
JCL-Beispiel eines Batch-Ausführungs-Jobs	686
7 - DB2 Commands	687
Vorgabe von DB2-Commands	687
Ergebnis einer DB2-Command-Ausführung	687
8 - Utilities (Online-Utilities)	688
Steuerungsmaske	688
Control Statement Data Set Names	689
Data Set Names	690
Ergebnisse der Job-Generierung	691
Generierungs-Informationen	691

Generierter Utility-Job	692
SQL-DDL-Data Definition Language	693
Grafischer Wegweiser durch das Kapitel 5 - SQL-DDL-Data Definition Language	693
Dynamische DB2-Objektverwaltung	694
Anlegen DB2-Objekt-Typen: CREATE	696
Defaults beim Anlegen von Objekten	696
Beispiele CREATE	699
Systemorientierte Daten-Objekt-Typen	699
STOGROUP, DATABASE und TABLESPACE	699
INDEX	701
Base-Index-Varianten	701
Auxiliary-Index, Extended Index und XML-Index	702
Benutzerorientierte Daten-Objekt-Typen	703
Array-Type	703
Distinct Type	703
Benutzerdefinierte Globale Variable (User-defined global variable)	703
Base-Table-Varianten	704
Basis-Tabelle mit Spalten-Definitionen	704
Basis-Tabelle mit Definition über SELECT-Statement mit Spalten-Kombination aus mehreren Tabellen	704
Basis-Tabelle mit Kopie vorhandener Tabellen-Spalten-Definitionen (LIKE)	705
Basis-Tabelle mit Index-kontrollierter Partitionierung (veraltet)	706
Basis-Tabelle mit Table-kontrollierter Partitionierung	706
Basis-Tabelle mit Hash-Organisation	707
Basis-Tabelle mit LOB-Spalte	707
Auxiliary Table für jede LOB-Spalte einer Basis-Tabelle	708
Basis-Tabelle mit XML-Spalte	709
XML Table	709
Archiv Table	710
Definition der Basis-Tabelle (Archive-enabled Table)	710
Clone Table	711
Global Temporary Table	712
MQT - Materialized Query Table	713
Temporal Table und History Table	714
Fachliche Zeitbezüge Temporal Table mit PERIOD BUSINESS_TIME	714
Technische Zeitbezüge Temporal Table mit PERIOD SYSTEM_TIME - ohne Versionierung	714
Technische Zeitbezüge Temporal Table mit PERIOD SYSTEM_TIME - mit Versionierung	715
Bi-Temporale Zeitbezüge Temporal Table und History Table - mit Versionierung	716
VIEW	717
ALIAS und SYNONYM (auslaufend)	718
COMMENT und LABEL	719
SEQUENCE	720
Funktionsorientierte Objekt-Typen: Routinen	721
UDF - User-defined Function	721
STP - Stored Procedure	723
Funktionsorientierte Objekt-Typen: Trigger	725
Trigger für die Manipulation von Tabellen: Before und After Trigger	725
Trigger für die Manipulation von Views: Instead of Trigger	726
Ändern DB2-Objekt-Typen: ALTER und RENAME	727

Besonderheiten bei Struktur-Veränderungen und Erweiterungen	729
Wie verhalten sich die Daten bei Struktur-Veränderungen?	729
Wie verhalten sich die Daten bei Struktur-Erweiterungen?	729
Was tun, wenn kein ALTER existiert?	730
Beispiele ALTER	731
Systemorientierte Daten-Objekt-Typen	731
STOGROUP, DATABASE und TABLESPACE	731
Konvertierung eines Segmented Tablespaces in einen UTS und setzen Objekt in Pending-Change-Status	732
INDEX	733
Base-Index	733
Auxiliary-Index, Extended Index und XML-Index	733
Benutzerorientierte Daten-Objekt-Typen	734
Array Type	734
Distinct Type	734
Globale Variable	734
Base-Table-Varianten	734
Archive Table	736
Auxiliary Table	736
Clone Table	737
Global Temporary Table	738
MQT - Materialized Query Table	738
Temporal Table	739
Temporal Table nachträglich auf Basis-Tabelle definieren	739
XML Table	740
VIEW	740
ALIAS und SYNONYM	740
COMMENT und LABEL	741
SEQUENCE	741
Funktionsorientierte Objekt-Typen: Routinen	742
UDF - User-defined Function	742
STP - Stored Procedure	743
Funktionsorientierte Objekt-Typen: Trigger	744
Beispiele RENAME	745
Löschen DB2-Objekt-Typen: DROP	746
Die Auswirkung eines DROP-Kommandos auf andere DB2-Objekte (Abbildung mit referenziellen Effekten)	748
Beispiele DROP	749
Systemorientierte Daten-Objekt-Typen	749
STOGROUP, DATABASE und TABLESPACE	749
INDEX	750
Base-Index	750
Auxiliary-Index	750
Extended Index und XML-Index	750
Benutzerorientierte Daten-Objekt-Typen	751
Array Type	751
Distinct Type	751
Globale Variable	751
Base-Table	751
Auxiliary Table	752
Clone Table	752

Global Temporary Table	752
MQT - Materialized Query Table	753
XML Table	753
VIEW	754
ALIAS und SYNONYM	754
COMMENT und LABEL	754
SEQUENCE	754
Funktionsorientierte Objekt-Typen: Routinen	755
UDF - User-defined Function	755
STP - Stored Procedure	755
Funktionsorientierte Objekt-Typen: Trigger	756
Objekt-Typ: Package	756
SQL-DML-Data Manipulation Language	757
Grafischer Wegweiser durch das Kapitel 6 - SQL-DML-Data Manipulation Language	757
SQL-DML-Sprachspektrum	758
SQL-DML-Anwendungsformen und Sprach-Schnittstellen	759
SELECT - Datenabfragen	760
SELECT-Typen	760
Relevante Komponenten eines SELECT-Statements	760
Result Table eines SELECT-Statements	761
Syntaktische Grundformen des SELECT-Statements	762
Entwicklung und Veränderungen der SELECT-Klauseln in den letzten DB2-Versionen	764
Syntax des SELECT-Statements	765
Grundformat des SELECT-Statements	765
Die logische Abarbeitungsfolge der SELECT-Parameter	766
SELECT-Klausel	767
Bereitstellung von Daten aus einer einzelnen Tabelle	767
Verdichtung der Result Table (DISTINCT)	767
Bereitstellung von Daten aus mehreren Tabellen (Join)	767
FROM-Klausel	768
Bereitstellung von Daten aus Basis-Tabellen bzw. Views	768
Inner und Outer Join	768
Nested Table Expression	768
WHERE-Klausel	769
Einfache Vergleichsoperatoren	769
Subquery	769
Row Expression (Row-Value-Expression)	770
ORDER BY-Klausel	770
GROUP BY-Klausel	770
OFFSET- und FETCH-Klausel	771
Tabelleninhalte der Beispieltabellen	772
Einfache SELECT-Abfragebeispiele aus einer Tabelle	774
Einfache Spaltenauswahl (PROJECT)	774
Konstante und arithmetisch errechnete Werte	774
Vergabe von Namen für Spalten der Result Table	775
Auswahl bestimmter Zeilen mit WHERE	775
Variable Auswahl bestimmter Zeilen mit WHERE und CASE	777

Boolsche Verknüpfungen und Prädikate	778
AND-Verknüpfung	778
OR-Verknüpfung	778
NOT-Verknüpfung	779
BETWEEN-Prädikat	779
IN-Prädikat	779
LIKE-Prädikat	779
NULL-Prädikat	781
DISTINCT-Prädikat	781
Mix von Prädikaten und Klammerung	782
Funktion (Builtin-Function)	783
Aggregate Function	783
Scalar Function	786
Konvertierungs-Funktion	786
DateTime-Funktion	787
VARCHAR_FORMAT-Funktion zur Aufbereitung von Timestamp- und numerischen Werten	789
Sonstige Funktionen	790
Verschachtelung von Aggregate Functions und Scalar Functions	791
Sortierung - ORDER BY	792
Auf- und absteigende Sortierung	792
Sortierung von abgeleiteten Spalten	793
Sortierung der Result Table eines Sub-Selects und der endgültigen Result Table	794
Variable Sortierung der Result Table aufgrund konkreter Inhalte mittels CASE	795
Gruppierung/ GROUP BY	796
HAVING	798
Grouping Sets, Cube und Rollup (ab DB2-V11)	801
Lösung mit Grouping Sets, Cube und Rollup	801
OFFSET- und FETCH-Klausel	803
Moving Aggregates (Scalar Aggregate Functions)	804
Set-Operatoren	807
UNION und UNION ALL	809
EXCEPT und EXCEPT ALL	812
INTERSECT und INTERSECT ALL	813
JOIN Selektion aus mehreren Tabellen	814
Join-Verarbeitungs-Regeln	814
Kartesisches Produkt	816
Equi-Join	817
Join einer Tabelle mit sich selbst	818
Inner Join und Outer Join	819
Übersicht der Ergebnis-Typen	819
Inner Join	821
Left Outer Join	821
Right Outer Join	822
Full Outer Join	823
Die Wirkung der ON-Klausel	824
Verknüpfung mehrerer Join-Typen	826
Kombination von Join und Table Expression	828

Scalar Full-Select	829
Möglichkeiten und Risiken	829
Scalar Full-Select in der WHERE-Bedingung	830
Scalar Full-Selects und Joins	831
Scalar Full-Select in der CASE-Expression	832
Table Expression	833
Nested Table Expression	833
Einfaches Beispiel	833
Nested Table Expression mit Outer Join	835
Nested Table Expression mit CASE (Statistik-Funktion)	836
Common Table Expression	837
Einsatzgebiet	837
Einfaches Beispiel	837
Einsatz mehrerer Common Table Expressions	838
Rekursives SQL	841
Die typische Hierarchie-Abbildung	841
Aufbau von Test- oder Hilfsdaten	845
Kalender mit Tagen, Wochen und Wochen-Tag	845
Kalender mit Jahr, Woche und Tag nach ISO (wann beginnt die erste Woche im Jahr?)	846
Common Table Expression und INSERT	847
Subquery (Sub-Query)	848
Allgemeine Syntax	848
Non-correlated Subquery	850
Subquery direkt nach einem Vergleichsoperator	850
Subquery nach IN	851
Subquery nach EXISTS	852
Subquery nach ANY bzw. SOME und ALL	853
Correlated Subquery	854
Verarbeitungsunterschiede zu non-correlated	854
Subquery direkt nach einem Vergleichsoperator	855
Subquery nach EXISTS	855
SELECT-Unterstützung für Temporal Tables	856
Perioden-Spezifikation der FROM-Klausel	856
SELECT mit der FOR BUSINESS-Klausel	857
Zugriff auf eine einzelne Tabelle	857
Zugriff auf mehrere Tabellen (Join)	858
SELECT mit der FOR SYSTEM-Klausel	860
Zugriff auf eine einzelne Tabelle	860
Zugriff auf mehrere Tabellen (Join)	863
SQL-Manipulations-Statements	867
Was versteht man unter Manipulations-Statements?	867
Temporal Tables fordern eine Abkopplung der fachlichen Änderung von der technischen Manipulations-Vorgabe	868
Fachlich-Funktionale Konsequenzen für Daten mit Zeitbezug	868
Manipulations-Typen	868
Fachliche Manipulations-Festlegungen hinsichtlich der Gültigkeitszeitleisten	869
DB2-Temporal-Table-spezifische Manipulations-Festlegungen	871
DELETE- und UPDATE-Regeln einer DB2-Temporal-Table	873
BUSINESS_PERIOD-Beispiele: UPDATE mit DATE-Charakteristiken (Inklusiv-Exklusiv)	874
BUSINESS_PERIOD-Beispiele: DELETE mit DATE-Charakteristiken (Inklusiv-Exklusiv)	875

SYSTEM_PERIOD-Beispiele: UPDATE	876
BUSINESS_PERIOD-Beispiele: DELETE (Inklusiv-Exklusiv)	878
Übersicht der SQL-Manipulations-Statement-Typen	880
SELECT-Statement mit SQL Data Change Statement	881
INSERT - Daten-Einfügung	882
Der INSERT fordert eine updateable Result Table	882
INSERT-Formate	883
Verhalten bei Spalten mit Default-Maßnahmen	884
Beispiele für SQL-Dateneinfügungen	885
INSERT einer einzelnen Zeile	885
INSERT einer Zeilen-Gruppe (im Anwendungsprogramm)	885
INSERT mehrerer Zeilen mit SELECT	886
SELECT mit SQL Data Change Statement (hier: SELECT mit INSERT)	888
FINAL TABLE mit einer Zeile	889
Beispiel: SELECT FROM FINAL TABLE (INSERT ... mit INCLUDE)	890
FINAL TABLE mit mehr als einer Zeile	892
UPDATE - Daten-Veränderung	894
Der UPDATE fordert eine updateable Result Table	894
UPDATE-Formate	895
Beispiele für SQL-Datenveränderungen	896
Searched Update	896
Positioned Update	897
SELECT mit SQL Data Change Statement (hier: SELECT mit UPDATE)	898
DELETE - Daten-Löschung	899
Der DELETE fordert eine updateable Result Table	899
DELETE-Formate	900
Beispiele für SQL-Datenlöschungen	901
Searched Delete	901
Positioned Delete	902
SELECT mit SQL Data Change Statement (hier: SELECT mit DELETE)	903
MERGE - Daten einmischen	904
Der MERGE fordert eine updateable Result Table	905
MERGE-Formate	906
Beispiele für SQL-Daten-Mischvorgänge	907
MERGE Typ 1 Sources: USING source-values - Source-Values Single Row	907
Einfügung einer Zeile	907
Änderung einer Zeile	908
Änderung mehrerer Zeilen	909
SELECT mit SQL Data Change Statement (hier: SELECT mit MERGE)	910
MERGE Typ 1 Sources: USING source-values - Source-Values Multiple Rows	913
SELECT mit MERGE in einem Cursor	913
MERGE Typ 2 Tables: USING table-reference	914
SQL-DCL-Data Control Language	915
Grafischer Wegweiser durch das Kapitel 7 - SQL-DCL-Data Control Language	915
DB2-Zugriffsschutzkonzept	916
Überblick	916
Vorüberlegungen	916

Welche Ressourcen können von DB2 geschützt werden?	917
Konsequenzen bei einer vollständigen Auslagerung auf einen ESM	919
Aufteilung der Sicherheitsfunktionen zwischen DB2 und einem ESM	920
Besonderheiten der Autorisierungs-Prüfungen, wenn RACF als ESM wirkt	921
DB2-Privilegien-Konzept	922
DB2-Autorisierungs-Konzept	923
Übergabe der Autorisierungs-Identifikatoren	924
Primär-, Sekundär- und Current-Autorisierungs-Id	925
Zeitpunkt der Autorisierungsprüfung	926
Autorisierungsprüfung zur Bind-Zeit	926
Autorisierungsprüfung zur Ausführungs-Zeit	927
Verhaltensweisen bei Dynamic SQL (Statement Behavior)	928
Wirkung der DYNAMICRULES-Optionen BIND, DEFINEBIND, INVOKEBIND, RUN, DEFINERUN und INVOKERUN	928
Zusammenfassung der Verhaltensweisen (Statement-Behavior) und Konsequenzen der Autorisierungsprüfungen	929
Privilegien-Konzept	930
Privilegien-Struktur	930
Explizite Privilegien	931
Grant-Typen und Privilegienbereiche	932
DB2-Ressource-Typen und ihre Zuordnung der Einzel-Privilegien	933
Implizite Privilegien	936
Owner eines Objekts	936
Administrations-Gruppen	938
Hierarchie der ADM-Privilegien	941
Explizit zuordenbare ADM-Gruppen	941
Implizit wirkende Einzel-Privilegien ADM-Gruppen	942
Welche Privilegien werden wann benötigt?	943
Erforderliche Privilegien zur Verwaltung der DB2-Objekte	943
Die erforderlichen Privilegien eines Anwendungsentwicklers	945
Zusammenspiel von Autorisierungs-Id-Typen und Privilegien	946
Ausführung von Dynamic SQL-Statements und die erforderlichen Privilegien	946
Plan- und Package-Aktivitäten	947
GRANT - Erteilen DB2-Privilegien	948
Privilegien-Kategorien und GRANT-Varianten	948
Hierarchisches GRANT-Konzept	949
GRANT-Besonderheiten	950
Beispiele GRANT	951
GRANT-Szenario	951
REVOKE - Aufheben DB2-Privilegien	953
REVOKE-Varianten	953
REVOKE und die Cascading Effekte	954
REVOKE-Besonderheiten	957
Beispiele REVOKE	958
REVOKE-Szenario	958
DB2-Datenzugriffs-Kontrolle (ACCESS CONTROL)	959
Column Access Control (Column-Mask) und Row Access Control (Row-Permission)	959
Überblick	959

Komponenten zur Einrichtung "geschützter" Objekte	960
Column Access Control (Column-Mask) - Maskierung von Ergebniswerten	962
Row Access Control (Row-Permission) - Filterung der Datenzeilen	963
Trusted Context und Rollen	964
Trusted Connection	964
Komponenten der Definition eines Trusted Contextes	965
Name des Trusted Contextes	965
System-Autorisierungs-Id	965
Default-Rolle und ROLE AS OBJECT OWNER	965
ENABLE oder DISABLE	965
Default-Security-Label	965
ATTRIBUTES	965
WITH USE FOR	966
Die Rolle (ROLE) im Rahmen eines Trusted Contextes	967
Aktivitäten eines anderen Benutzers mit dessen Objekt-Privilegien durchführen	968
SQL-Statements und Katalog-Informationen für Trusted Context und Rollen-Konzept	969
VIEW - Inhaltsbezogener Datenschutz	970
Besonderheiten bei der Anwendung in Programmen	971
MLS - Multi-Level Security	972
Probleme bisheriger Zugriffsschutz-Verfahren	972
MLS-Konzept und relevante Komponenten	973
Security Kategorie, Security Level und Security Label	974
MAC - Mandatory Access Control / Mandatory Access Checking	975
Dominanz (Dominate/Dominance)	975
Äquivalenz (Equivalent)	975
Nicht verarbeitbar	975
Inkompatibilität (disjoint oder disjunkt)	975
Write-down Control	975
MLS-Implementierungs-Varianten im DB2	976
MLS und MAC auf Objekt-Ebene	976
MLS auf Zeilen-Ebene (Row-Level Granularity)	977
Write-down Control bei Row-Level Granularity	977
Behandlung der betroffenen SQL-Statements	977
CREATE TABLE/ALTER TABLE	977
DROP TABLE	977
SELECT	977
INSERT	977
UPDATE	978
DELETE	978
Behandlung der betroffenen Utilities	979
LOAD	979
UNLOAD	979
REORG TABLESPACE	979
Behandlung in Triggern	979
Sonstige Besonderheiten	979
Zusammenfassung und Konsequenzen des MLS-Konzepts	980
Beispielhafte Ressource- und Benutzer-Definitionen	980

Security-Kategorien	980
Security-Level	980
Security-Label	980
Fazit	981
Vor- und Nachteile des MLS-Konzepts	982
Vorteile des MLS-Konzepts	982
Nachteile des MLS-Konzepts	982
Verschlüsselung von Daten (Encryption)	983
Komponenten für die Verschlüsselung	983
Welche Daten eignen sich für DB2-Verschlüsselungs-Techniken?	983
Definition der Meta-Daten für die Verschlüsselung	984
Die Rolle des Encryption Passwords und Password Hints	984
Verschlüsselungs-Konzepte	985
Verschlüsselung auf Spalten-Ebene	985
Verschlüsselung auf Daten-Wert-Ebene	985
Effekte der Verschlüsselung	986
Behandlung der Prädikate bei Vergleichen	986
Performance-Aspekte	986
Kann man auch PKs und FKs verschlüsseln?	987
Wie behandeln Utilities verschlüsselte Daten?	987
DB2-Katalog	988
Grafischer Wegweiser durch das Kapitel 8 - DB2-Katalog	988
Aufbau und Inhalte	989
Liste der Katalog-Tabellen	991
Zuordnung relevanter Objekte zu Katalog-Tabellen	997
Referenzielle Beziehungen der wichtigsten Katalog-Tabellen	998
Katalog-Informationen zur Unterstützung der Objekt-Administration	1000
Katalog-Spalten mit Informationen zur Speicherorganisation	1000
Katalog-Spalten mit Informationen zur Zugriffspfad-Analyse durch den Optimizer	1002
Statistiken und ihre Nutzungsmöglichkeit	1010
Überblick	1010
Was ist der Zweck der Statistiken?	1010
Übersicht der relevanten Statistik-Informationen (Generelle Statistik-Kategorien)	1011
DB2-Statistik-Kategorien und relevante Komponenten	1012
Speicherung der relevanten Index-Statistik-Informationen	1014
Bedeutung von LEAFDIST	1015
Bedeutung von LEAFNEAR und LEAFFAR	1016
Speicherung der relevanten Daten-Statistik-Informationen	1017
Bedeutung von NEAROFFPOSF und FAROFFPOSF	1018
Bedeutung von CLUSTERRATIOF	1019
Bedeutung von DATAREPEATFACTORF (DRF)	1020
Bedeutung von NEARINDREF und FARINDREF	1021
Speicherung der relevanten Hash-Statistik-Informationen	1022
Kontrolle der Speicher-Organisation	1022
Bewertung der Nutzung der Hash-Organisation	1026
Speicherung der relevanten LOB-Statistik-Informationen	1028
Bedeutung von ORGRATIO	1029

Aktuelle Statistiken: RTS - Realtime-Statistiken	1030
Überblick	1030
RTS-Komponenten	1031
Sammeln und Aktualisieren der Statistikdaten	1032
Ausschreiben der Statistiken in die RTS-Tabellen (Externalisierung)	1033
Analyse der Statistiken	1033
Stored Procedure: DSNACCOX (vor DB2-Version 9 DSNACCOR) - Analyse Realtime Statistiken	1033
Welche RTS-Tatbestände führen zu einer Utility-Aktivierungs-Empfehlung?	1034
Welche generellen Katalog-Tatbestände führen zu einer Utility-Aktivierungs-Empfehlung?	1035
Katalog-Abfragen	1036
Katalog-Abfragen der Katalogbeschreibungen (Metadaten)	1036
Objekt-Abfragen für Anwendungs-Entwickler	1038
Katalog-Abfragen zur Überwachung der optimalen Speicherorganisation	1042
Prüfung, ob RUNSTATS überhaupt noch nicht oder seit längerer Zeit nicht aktiviert wurde	1042
Abfrage auf Reorganisations-Erfordernis	1043
Nutzung von Tabellen und Indizes durch Packages	1046
Katalog-Queries über referenzielle Beziehungen	1047
Katalog-Queries zur Unterstützung und Kontrolle des Datenschutzes	1049
Katalog-Veränderungen	1050
Utilities	1051
Grafischer Wegweiser durch das Kapitel 9 - DB2-Utilities	1051
Überblick und relevante Komponenten	1052
Abgrenzung IBM-DB2-Tools - DB2-Utilities	1052
DB2-Online-Utilities und Standalone-Utilities	1054
DB2-Online-Utilities: Überblick	1055
Komponenten eines Utilities	1055
Aktivierungs-Varianten für Utilities	1056
Die Phasen eines Utilities	1057
Ergebnis- und Status-Kontrolle eines Utility-Laufs	1058
Status-Kontrolle während der Aktivierungs-Zeit eines Utility-Laufs	1058
Ergebnis-Kontrolle nach dem Abschluss eines Utility-Laufs	1058
RESTART (CURRENT) und RESTART (PHASE)	1059
Utility-Datasets	1061
Vorgabe von Utility-Control-Statements	1061
CCSID-Behandlung von Utility-Control Statements	1061
Spezielle Utility-Control-Statements (SYSIN-DD-Statement)	1061
Dataset-Charakteristiken	1062
Übersicht der Datasets für Online-Utilities	1063
Dataset-Dispositionen	1064
Dataset-Größen	1065
Vorgehensweisen, wenn Datasets zu klein sind	1065
Unterstützung von FlashCopy	1066
FlashCopy-Datasets	1067
FlashCopy-Installations-Parameter	1068
Unterstützung von Copypools	1069
Generelle und spezielle Utility-Control Statements	1070
Utility-Control Statement: EXEC SQL	1071
Utility-Control Statement: LISTDEF	1073

Utility-Control Statement: TEMPLATE	1077
Utility-Control Statement: OPTIONS	1078
Die Auswirkungen der Objekt-Versionierung für Utilities	1080
Wie behandeln Utilities verschlüsselte Daten?	1081
Wie behandeln Utilities MLS-geschützte Daten	1081
Inline-Utility-Ausführungen	1082
Besonderheiten der Inline-Kopien für Utilities	1082
Inline-Kopien mit dem LOAD-Utility	1083
Inline-Kopien mit dem REORG-Utility	1087
LOAD- und REORG-Besonderheiten für Inline-Kopien bei LOB- und XML-Objekten	1087
Inline-Statistiken mit dem LOAD-, REORG- oder REBUILD-Utility	1088
Sperrungen von Utilities und Kompatibilität zu anderen Utilities	1089
Claims und Drains	1089
Restriktiver Space-Status von Utilities	1089
Kompatibilität der Utilities untereinander	1090
Standalone-Utilities	1091
Stored Procedures im Utility-Umfeld	1091
DB2 Online Utilities	1092
Übersicht der Kategorien und Utilities	1092
Utilities zum Laden, Entladen und Reorganisieren von Daten und Indizes	1093
LOAD-Utility	1094
Überblick	1094
Vorbereitungs-Aktivitäten	1097
LOAD: Erforderliche Datasets und Objekte	1098
LOAD: Utility-Sperrungen und Abhängigkeiten zu anderen Utilities	1099
LOAD: Utility-Ausführungs-Phasen	1100
Besonderheiten von Daten-Typen- und -Charakteristiken	1101
Besonderheiten beim Laden von Daten in Tabellen mit IDENTITY-Spalten	1102
Besonderheiten beim Laden von Daten mit GENERATED ALWAYS-Spalten	1103
Komprimierung der Daten	1104
Performance-Faktoren beim Laden	1104
Offline-Load	1105
Überblick über die Komponenten und relevante Klauseln	1105
Ein- und Ausgabe-Datasets	1105
SHRLEVEL NONE oder CHANGE	1107
REPLACE oder RESUME - Besonderheiten	1108
Laden der Daten von Partitioned Tablespaces	1109
Laden von Daten für Universal Tablespaces Partition-by-growth (UTS-PBG)	1110
Laden von LOB-Daten	1110
Laden von XML-Daten	1110
LOAD-Parallelverarbeitung	1111
Paralleler-Verarbeitungs-Varianten	1111
Paralleler Aufbau der Indizes beim LOAD	1113
Abbruch und Restart des LOAD-Utilities	1115
Online-Load	1116
Überblick über die Komponenten und relevante Klauseln	1116
Ein- und Ausgabe-Datasets	1116
Unterscheidungsmerkmale von Offline- und Online-Load	1117
Konsequenzen bei Einsatz des Online-Load-Utilities	1117
Test-Ergebnisse Vergleich Offline-Online-Load	1118

Empfehlungen	1119
Nachbereitungs-Aktivitäten	1120
Test-Beispiele	1121
Laden 1 Mio Zeilen in einen UTS-PBR	1121
Erläuterung des Test-Beispiels	1121
Utility-Aktivierung ohne Parallel-Verarbeitung (PARALLEL(1) als Default)	1121
Job-Control und Utility-Control-Statements ohne Parallel-Verarbeitung	1121
Ausführungs-Protokolle	1121
Utility-Aktivierung mit Parallel-Verarbeitung (PARALLEL)	1123
Job-Control und Utility-Control-Statements mit Parallel-Verarbeitung	1123
Ausführungs-Protokolle	1123
Laden von 6 Tabellen in einen Segmented Tablespace	1125
Erläuterung des Test-Beispiels	1125
Job-Control und Utility-Control-Statements	1125
Ausführungs-Protokolle	1126
Test-Beispiel: LOAD von XML-Daten einer Temporal Table mit SPANNED YES	1127
Job-Control und Utility-Control-Statements	1127
Ausführungs-Protokolle	1128
REBUILD INDEX-Utility	1129
Überblick über die Komponenten und relevante Klauseln	1129
Online-Rebuild (SHRLEVEL CHANGE)	1131
Ein- und Ausgabe-Datasets	1132
Vorbereitungs-Aktivitäten	1132
REBUILD INDEX: Erforderliche Datasets und Objekte	1133
REBUILD INDEX: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1134
REBUILD INDEX: Utility-Ausführungs-Phasen	1134
Paralleler Aufbau der Indizes beim REBUILD INDEX	1135
Abbruch und Restart des REBUILD INDEX-Utilities	1135
Wann ist REBUILD INDEX zweckmäßig oder erforderlich?	1136
Test-Beispiele	1137
Job-Control	1137
Utility-Control-Statements	1137
Auszüge der Ausführungs-Protokolle	1137
REORG INDEX-Utility	1138
Überblick	1138
Reorganisations-Methoden	1138
Überblick über die Komponenten und relevante Klauseln	1139
Ein- und Ausgabe-Datasets	1139
Abbruch parallel laufender Prozesse mittels FORCE-Klausel bei Verhinderung des REORG-Abschlusses	1140
Vorbereitungs-Aktivitäten	1141
REORG INDEX: Erforderliche Datasets und Objekte	1142
REORG INDEX: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1143
REORG INDEX: Utility-Ausführungs-Phasen	1143
Paralleler Aufbau der Indizes beim REORG INDEX	1144
Abbruch und Restart des REORG INDEX-Utilities	1144
Wann ist ein REORG INDEX zweckmäßig oder erforderlich?	1145
Test-Beispiele	1146
Job-Control	1146
Utility-Control-Statements	1146
Auszüge der Ausführungs-Protokolle	1146
REORG TABLESPACE-Utility	1147

Überblick	1147
Reorganisations-Methoden	1147
Erweiterungen ab DB2-Version 9	1148
Erweiterungen ab DB2-Version 10	1148
Erweiterungen ab DB2-Version 11	1149
Erweiterungen ab DB2-Version 12	1150
Besonderheiten bei einer Reorganisation mit der Materialisierung von Pending-Objekt-Änderungen	1151
Strategien beim Entladen der Daten: Clustering Index-Folge oder Sortierung (SORTDATA)	1152
Reorganisation der verschiedenen Tablespace-Typen	1153
Reorganisation eines Segmented Tablespaces	1153
Reorganisation eines Partitioned Tablespaces	1153
Parallele Reorganisation bei Partitioned Tablespaces	1153
SORTNPSI - Sollen NPI-Index-Daten komplett entladen werden oder nicht?	1154
Reorganisation von UTS-PBG-Tablespaces (Partition-by-Growth)	1156
Reorganisation von XML-Tablespaces	1156
Reorganisation von LOB-Tablespaces	1157
Logging eines LOB-Tablespaces	1158
Reorganisation von Katalog- und Directory-Tablespaces	1159
Überblick über die Komponenten und relevante Klauseln	1160
Ein- und Ausgabe-Datasets	1160
Performance-Faktoren bei der Reorganisation	1162
Parallele Verarbeitungsmöglichkeiten durch andere Prozesse	1163
SHRLEVEL NONE - keine Parallelverarbeitung ab RELOAD erlaubt	1163
SHRLEVEL REFERENCE - lesende Parallel-Verarbeitung erlaubt	1164
SHRLEVEL CHANGE - Parallel-Verarbeitung erlaubt (Online-Reorg)	1165
Rolle der Mapping Table	1166
Nicht-Verfügbarkeitszeiten der Objekte für Parallel-Anwendungen	1166
Abbruch parallel laufender Prozesse mittel FORCE-Klausel bei Verhinderung des REORG-Abschlusses	1167
Konsequenzen bei Einsatz des Online-Reorg-Utilities	1167
REBALANCE - Re-Balancieren ungleich großer Partitions	1168
Beispiel einer Online-Rebalance-Maßnahme	1169
Erläuterung des Beispiel	1169
Utility-Control-Statements	1169
Ausführungs-Protokolle	1169
Vorbereitungs-Aktivitäten	1171
REORG TABLESPACE: Erforderliche Datasets und Objekte	1173
REORG TABLESPACE: Utility-Ausführungs-Phasen	1175
REORG TABLESPACE: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1176
Abbruch und Restart des REORG TABLESPACE-Utilities	1177
REORG TABLESPACE-Utility: Abbruch in Phasen und Pending-Status	1177
Wann ist REORG TABLESPACE zweckmäßig oder erforderlich?	1178
Nachbereitungs-Aktivitäten	1179
Test-Beispiele	1180
Job-Control mit Direkt-Vorgabe eines Tablespace-Namens	1180
Utility-Control-Statements	1180
Auszüge der Ausführungs-Protokolle	1181
Job-Control mit Vorgabe von Objekt-Listen und Templates	1182
Auszüge der Ausführungs-Protokolle	1182
Beispiel der Reorganisation von LOB-Tablespaces mit SHRLEVEL CHANGE	1185
Utility-Control	1185
Ausführungs-Protokollierung	1185

Beispiel der Reorganisation eines Tablespaces mit Objekt-Änderungen im Schwebezustand (Pending Changes)	1186
Utility-Control	1186
Ausführungs-Protokollierung (relevanter Auszug)	1186
Beispiel der Reorganisation von UTS-PBG-Tablespaces mit Löschen nicht benötigter Partitions am Ende (DROP_PART YES)	1187
Utility-Control	1187
Ausführungs-Protokollierung	1187
Beispiel der Reorganisation eines Tablespaces mit Objekt-Änderungen im Schwebezustand (Pending Changes)	1188
Utility-Control	1188
Ausführungs-Protokollierung (relevanter Auszug)	1188
UNLOAD-Utility	1189
Überblick	1189
Besonderheiten beim Entladen der Daten	1191
Entladen von Tablespaces, Tables und Image-Copies	1191
Beeinflussung der Spalten- und Datenauswahl	1192
Konvertierung der Daten	1192
Vorbereitungs-Aktivitäten	1192
UNLOAD: Erforderliche Datasets und Objekte	1193
UNLOAD: Utility-Ausführungs-Phasen	1193
UNLOAD: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1193
Abbruch und Restart des UNLOAD-Utilities	1194
Test-Beispiele	1194
Test-Beispiel: Entladen Daten eines Partitioned Tablespaces	1194
Job-Control	1194
Utility-Control-Statements	1194
Auszüge des Ausführungs-Protokolls	1194
Test-Beispiel: Entladen Daten DELIMITED	1195
EBCDIC-Daten	1195
Test-Beispiel: Entladen Daten mit der WHEN-Klausel und mit TEMPLATE in SYSTEMPL	1196
Job-Control	1196
Auszüge des Ausführungs-Protokolls	1196
Ausdruck der Daten	1197
Test-Beispiele: Entladen Daten mit CCSID EBCDIC und UNICODE	1198
Test-Basis	1198
Entladen beide mit Default-CCSID	1198
Entlade-Bestände	1199
Entladen beide mit CCSID EBCDIC	1200
Entlade-Bestände	1201
Entladen beide mit CCSID UNICODE	1202
Entlade-Bestände	1203
Entladen beide mit CCSID UNICODE DELIMITED	1204
Entlade-Bestände	1205
Test-Beispiel: Entladen LOB-Daten mit externer File Referenz und TEMPLATE	1206
Test-Basis	1206
UNLOAD-Job	1206
UNLOAD-Ausführungs-Protokoll	1207
Test-Beispiel: UN LOAD von XML-Daten einer Temporal Table mit SPANNED YES	1208
Test-Basis und UNLOAD-Utility-Control	1208
Utilities zum Erzeugen von Statistik-Informationen aus Daten und Indizes	1209
RUNSTATS-Utility	1210
Überblick	1210
Aktualisierung der Katalog-Statistiken	1214
Welche Tabellen und Spalten aktualisiert RUNSTATS?	1214

RUNSTATS: Wirkung der Parameter UPDATE und HISTORY	1215
Woran erkennt man die Aktualität der Statistiken?	1217
Darstellung der Statistikwerte im Katalog	1217
Manuelle Veränderung der Statistikwerte im Katalog	1218
Exaktheit der Statistikwerte	1218
Ungleichförmige Verteilung von Datenwerten	1219
Statische Erfassung von Einzelwerten (KEYCARD und FREQUVAL) und Spalten-Gruppen (COLGROUP)	1219
Statische Erfassung der Verteilung von Spaltenwerten oder Wertegruppen (HISTOGRAM)	1220
Zurücksetzen der Katalog-Statistiken für die Zugriffspfad-Analyse (RESET ACCESSPATH)	1222
Beispiel eines RUNSTATS-Laufs mit Zurücksetzen des kompletten Statistik-Stands auf den Initialisierungszustand (RESET ACCESSPATH)	1223
Nutzung von Tabellen-Profilen zur Vereinheitlichung der RUNSTATS-Ausführungen (PROFILES)	1224
Vorbereitungs-Aktivitäten	1225
RUNSTATS TABLESPACE: Erforderliche Datasets und Objekte	1225
RUNSTATS TABLESPACE: Utility-Ausführungs-Phasen	1225
RUNSTATS TABLESPACE: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1226
Abbruch und Restart des RUNSTATS-Utilities	1226
Test-Beispiele	1227
RUNSTATS TABLESPACE TABLE ALL und INDEX ALL	1227
Utility-Control-Statements	1227
Auszüge der Ausführungs-Protokolle	1227
STOSPACE-Utility	1230
Überblick	1230
Vorbereitungs-Aktivitäten	1231
STOSPACE: Erforderliche Datasets und Objekte	1231
STOSPACE: Utility-Ausführungs-Phasen	1231
STOSPACE: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1231
Abbruch und Restart des STOSPACE-Utilities	1231
Test-Beispiele	1232
Job-Control	1232
Utility-Control-Statements	1232
Ausführungs-Protokoll	1232
Katalog-Statistiken	1232
Utilities zum Prüfen der Konsistenz von Daten und Indizes	1233
Überblick und Abgrenzung der Aufgaben	1233
Bereinigung erkannter Fehler-Situationen	1235
Löschen fehlerhafter Basis-Datenzeilen mittels DELETE YES	1235
LOB- oder XML-Fehler-Typen: Orphan-, Missing-, Out-of-Synch- und Invalid-Werte	1236
Entdeckung der Fehlerzustände und mögliche Ursachen	1236
Bereinigungs-Möglichkeiten der Fehlerzustände	1237
CHECK DATA-Utility	1238
Überblick	1238
Was wird denn nun wie und warum gesteuert?	1241
Behandlung des CHKP-Check-Pending Status ab DB2-Version 10	1241
Objekt-Auswahl: SCOPE-Klausel	1242
XML-Tablespace-Prüfungen	1243
xxxERROR-Klausel	1244
Vorbereitungs-Aktivitäten	1245
CHECK DATA: Erforderliche Datasets und Objekte	1245
CHECK DATA: Utility-Ausführungs-Phasen	1246
CHECK DATA: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1246
Abbruch und Restart des CHECK DATA-Utilities	1247

Nachbereitungs-Aktivitäten	1247
Test-Beispiele	1248
Löschen fehlerhafter referenzieller Zeilen	1248
Job-Control	1248
Utility-Control-Statements	1248
Auszüge der Ausführungs-Protokolle	1248
LOB-Daten aufgrund eines PIT-Recoveries auf ungültig setzen	1249
Utility-Control-Statements	1249
Ausführungs-Protokoll	1249
CHECK INDEX-Utility	1250
Überblick	1250
Vorbereitungs-Aktivitäten	1251
CHECK INDEX: Erforderliche Datasets und Objekte	1252
CHECK INDEX: Utility-Ausführungs-Phasen	1252
CHECK INDEX: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1252
Abbruch und Restart des CHECK DATA-Utilities	1252
Nachbereitungs-Aktivitäten	1253
Test-Beispiele	1254
Job-Control	1254
Utility-Control-Statements	1254
Auszüge der Ausführungs-Protokolle	1254
CHECK LOB-Utility	1255
Überblick	1255
Behandlung des CHKP-Check-Pending Status ab DB2-Version 10	1257
Vorbereitungs-Aktivitäten	1257
CHECK LOB: Erforderliche Datasets und Objekte	1258
CHECK LOB: Utility-Ausführungs-Phasen	1258
CHECK LOB: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1259
Abbruch und Restart des CHECK LOB-Utilities	1259
Nachbereitungs-Aktivitäten	1259
Test-Beispiele	1260
Job-Control	1260
Utility-Control-Statements	1260
Auszüge der Ausführungs-Protokolle	1260
Utilities zum Sichern und Wiederherstellen von Daten und Indizes	1261
BACKUP SYSTEM-Utility	1262
Überblick	1262
Vorbereitungs-Aktivitäten	1264
BACKUP SYSTEM: Erforderliche Datasets und Objekte	1264
BACKUP SYSTEM: Utility-Ausführungs-Phasen	1264
BACKUP SYSTEM: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1264
Abbruch und Restart des BACKUP SYSTEM-Utilities	1265
Nachbereitungs-Aktivitäten	1265
Test-Beispiele	1265
Job-Control	1265
COPY-Utility	1266
Überblick	1266
Copy-Typen	1268
Full Image Copy	1268
Incremental Image Copy	1268

Primär und Backup-Kopie	1269
Kopien für das lokale und das Recovery-System	1269
FlashCopy Kopien	1269
DFSMS Concurrent Copy	1270
Parallele Ausführungsmöglichkeiten des Utilities	1271
Parallele Sicherung von Objekten in einem Job	1271
Parallele Sicherung von Objekten in mehreren Jobs	1271
Erzeugen von Kopien für mehrere Objekte	1271
Vorgabe einer Liste in einer Copy-Anweisung eines Jobs	1271
Vorgabe mehrerer Copy-Anweisungen in einem Job	1271
Wie kann die Konsistenz der Objekte sichergestellt werden?	1272
Bei bestimmten 'Restriktiven Pending-Status' kann bzw. sollte keine Image Copy erzeugt werden	1272
Vom Einsatz von -TERM UTIL wird beim COPY abgeraten	1273
Vorbereitungs-Aktivitäten	1273
COPY: Erforderliche Datasets und Objekte	1273
COPY: Utility-Ausführungs-Phasen	1274
COPY: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1274
Abbruch und Restart des COPY-Utilities	1275
Nachbereitungs-Aktivitäten	1275
Test-Beispiele	1276
Job-Control	1276
Utility-Control-Statements	1276
Auszüge der Ausführungs-Protokolle	1276
COPYTOCOPY-Utility	1279
Überblick	1279
Vorbereitungs-Aktivitäten	1281
COPYTOCOPY: Erforderliche Datasets und Objekte	1281
COPYTOCOPY: Utility-Ausführungs-Phasen	1281
COPYTOCOPY: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1281
Abbruch und Restart des COPYTOCOPY-Utilities	1282
Nachbereitungs-Aktivitäten	1282
Test-Beispiele	1283
Job-Control	1283
Utility-Control-Statements	1283
Auszüge der Ausführungs-Protokolle	1283
MERGECOPY-Utility	1284
Überblick	1284
Vorbereitungs-Aktivitäten	1285
MERGECOPY: Erforderliche Datasets und Objekte	1285
MERGECOPY: Utility-Ausführungs-Phasen	1286
MERGECOPY: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1286
Abbruch und Restart des MERGECOPY-Utilities	1286
Nachbereitungs-Aktivitäten	1286
Test-Beispiele	1287
Job-Control	1287
Utility-Control-Statements	1287
Auszüge der Ausführungs-Protokolle	1287
QUIESCE-Utility	1288
Überblick	1288
Vorbereitungs-Aktivitäten	1290

QUIESCE: Erforderliche Datasets und Objekte	1290
QUIESCE: Utility-Ausführungs-Phasen	1290
QUIESCE: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1290
Abbruch und Restart des QUIESCE-Utilities	1291
Nachbereitungs-Aktivitäten	1291
Test-Beispiele	1291
Job-Control	1291
Utility-Control-Statements	1291
Auszüge der Ausführungs-Protokolle	1291
RECOVER-Utility	1293
Überblick	1293
Zurücksetzen von Datenzuständen	1298
Vorwärts- und Rückwärts-Recovery (forward/backward und BACKOUT YES)	1299
Zurücksetzen auf den aktuellen Status	1300
Zurücksetzen auf einen früheren Zeitpunkt	1300
Copy-Typen	1301
RECOVER-Szenario eines Partitioned Tablespaces	1302
COPY- und RECOVER-Klauseln für die verschiedenen Recover- und Objekt-Typen eines Partitioned Tablespaces	1303
PIT-RECOVER-Beispiele	1304
COPY und PIT-RECOVER kompletter Partitioned Tablespaces	1304
COPY und RECOVER aller oder einzelner Partitions eines Partitioned Tablespaces	1305
RECOVER von Tablespaces mit verknüpften Objekten (Indizes, RI-Constraints, LOB- oder XML-Tablespaces sowie deren Indizes)	1307
Pending Status	1307
Ungültigkeits-Markierung (invalid) von LOB- oder XML-Werten	1307
Wiederherstellen von Base-Tablespaces	1307
Wiederherstellen von LOB- oder XML-Tablespaces	1307
Wiederherstellen von Indexspaces	1307
RECOVER von Tablespaces, die mit NOT LOGGED definiert sind	1308
RECOVER von Katalog- und Directory-Objekten	1308
Objekt-Hierarchie	1308
Vorbereitungs-Aktivitäten	1309
RECOVER: Erforderliche Datasets und Objekte	1309
RECOVER: Utility-Ausführungs-Phasen	1310
RECOVER: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1311
Abbruch und Restart des RECOVER-Utilities	1311
Nachbereitungs-Aktivitäten	1311
Test-Beispiele	1312
Job-Control	1312
Utility-Control-Statements	1312
Auszüge der Ausführungs-Protokolle	1312
RESTORE SYSTEM-Utility	1314
Überblick	1314
Vorbereitungs-Aktivitäten	1315
RESTORE SYSTEM: Erforderliche Datasets und Objekte	1316
RESTORE SYSTEM: Utility-Ausführungs-Phasen	1316
RESTORE SYSTEM: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1316
Abbruch und Restart des RESTORE SYSTEM-Utilities	1317
Nachbereitungs-Aktivitäten	1317
Test-Beispiele	1317
Job-Control	1317

Utilities zum Löschen nicht mehr benötigter System-Informationen	1318
MODIFY RECOVERY-Utility	1319
Überblick	1319
Vorbereitungs-Aktivitäten	1321
MODIFY RECOVERY: Erforderliche Datasets und Objekte	1321
MODIFY RECOVERY: Utility-Ausführungs-Phasen	1321
MODIFY RECOVERY: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1321
Abbruch und Restart des MODIFY RECOVERY-Utilities	1322
Nachbereitungs-Aktivitäten	1322
Test-Beispiele	1323
Job-Control	1323
Utility-Control-Statements	1323
Auszüge der Ausführungs-Protokolle	1323
Szenario der Löschung aller Image Copies, außer der letzten Full Image Copy und ggf. nachfolgender Incremental Image Copies	1324
Auszug aus SYSIBM.SYSCOPY vor MODIFY RECOVERY	1324
MODIFY RECOVERY-Protokoll	1324
Auszug aus SYSIBM.SYSCOPY nach MODIFY RECOVERY	1325
MODIFY STATISTICS-Utility	1326
Überblick	1326
Vorbereitungs-Aktivitäten	1327
MODIFY STATISTICS: Erforderliche Datasets und Objekte	1327
MODIFY STATISTICS: Utility-Ausführungs-Phasen	1327
MODIFY STATISTICS: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1327
Abbruch und Restart des MODIFY STATISTICS-Utilities	1327
Nachbereitungs-Aktivitäten	1327
Test-Beispiele	1328
Job-Control	1328
Utility-Control-Statements	1328
Auszüge der Ausführungs-Protokolle	1328
Utilities für Objekt-Informationen und zur Fehler-Analyse und Fehler-Bereinigung	1329
DIAGNOSE-Utility	1330
Überblick	1330
Vorbereitungs-Aktivitäten	1331
DIAGNOSE: Erforderliche Datasets und Objekte	1331
DIAGNOSE: Utility-Ausführungs-Phasen	1331
DIAGNOSE: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1331
Abbruch und Restart des DIAGNOSE-Utilities	1331
Nachbereitungs-Aktivitäten	1331
Test-Beispiele	1332
Job-Control	1332
Utility-Control-Statements	1332
Auszüge der Ausführungs-Protokolle	1332
REPAIR-Utility	1333
Überblick	1333
Vorbereitungs-Aktivitäten	1335
REPAIR: Erforderliche Datasets und Objekte	1335
REPAIR: Utility-Ausführungs-Phasen	1335
REPAIR: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1336
Abbruch und Restart des REPAIR-Utilities	1336
Nachbereitungs-Aktivitäten	1336

Test-Beispiele	1337
Job-Control	1337
Utility-Control-Statements	1337
Auszüge der Ausführungs-Protokolle	1337
REPORT-Utility	1338
Überblick	1338
Vorbereitungs-Aktivitäten	1339
REPORT: Erforderliche Datasets und Objekte	1339
REPORT: Utility-Ausführungs-Phasen	1339
REPORT: Utility-Sperren und Abhängigkeiten zu anderen Utilities	1340
Abbruch und Restart des REPORT-Utilities	1340
Nachbereitungs-Aktivitäten	1340
Test-Beispiele	1341
Job-Control	1341
Utility-Control-Statements	1341
Auszüge der Ausführungs-Protokolle	1341
Standalone-Utilities	1348
Übersicht	1348
DSNJCNVB - Konvertieren BSDS Log-Datasets	1350
Überblick	1350
Vorbereitungs-Aktivitäten	1350
DSNJCNVB: Erforderliche Datasets und Objekte	1351
Nachbereitungs-Aktivitäten	1351
Test-Beispiele	1351
Job-Control	1351
Auszüge der Ausführungs-Protokolle	1351
DSNJCNVT - Konvertieren BSDS auf das 10-Byte-Extended RBA/LRSN-Format	1352
Überblick	1352
Vorbereitungs-Aktivitäten	1353
DSNJCNVB: Erforderliche Datasets und Objekte	1354
Nachbereitungs-Aktivitäten	1354
Test-Beispiele	1354
Job-Control	1354
Auszüge der Ausführungs-Protokolle	1354
DSNJLOGF (Preformat Active Log)	1355
Überblick	1355
Vorbereitungs-Aktivitäten	1355
DSNJLOGF: Erforderliche Datasets und Objekte	1356
Nachbereitungs-Aktivitäten	1356
Test-Beispiele	1356
Job-Control	1356
Auszüge der Ausführungs-Protokolle	1356
DSNJU003 (Change Log Inventory)	1357
Überblick	1357
Vorbereitungs-Aktivitäten	1359
DSNJU003: Erforderliche Datasets und Objekte	1359
Nachbereitungs-Aktivitäten	1359
Test-Beispiele	1360

Job-Control	1360
Auszüge der Ausführungs-Protokolle	1360
DSNJU004 (Print Log Map)	1361
Überblick	1361
Vorbereitungs-Aktivitäten	1362
DSNJU004: Erforderliche Datasets und Objekte	1362
Nachbereitungs-Aktivitäten	1362
Test-Beispiele	1362
Job-Control	1362
Auszüge der Ausführungs-Protokolle (aufbereitete Form mit Erläuterungen)	1363
DSN1CHKR	1366
Überblick	1366
Vorbereitungs-Aktivitäten	1367
DSN1CHKR: Erforderliche Datasets und Objekte	1367
Nachbereitungs-Aktivitäten	1367
Test-Beispiele	1367
Job-Control	1367
DSN1COMP	1368
Überblick	1368
Vorbereitungs-Aktivitäten	1369
DSN1COMP: Erforderliche Datasets und Objekte	1369
Nachbereitungs-Aktivitäten	1369
Test-Beispiele	1369
Job-Control	1369
Auszüge der Ausführungs-Protokolle	1370
DSN1COPY	1371
Überblick	1371
DSN1COPY: Erweiterte Konsistenzprüfungen ab DB2-Version 11	1373
Vorbereitungs-Aktivitäten	1374
DSN1COPY: Erforderliche Datasets und Objekte	1374
Test-Beispiele	1375
Job-Control	1375
Auszüge der Ausführungs-Protokolle	1376
DSN1LOGP	1377
Überblick	1377
Vorbereitungsaktivitäten	1378
DSN1LOGP: Erforderliche Datasets und Objekte	1378
Nachbereitungs-Aktivitäten	1378
Test-Beispiele	1379
Job-Control	1379
Auszüge der Ausführungs-Protokolle	1379
DSN1PRNT	1380
Überblick	1380
Vorbereitungs-Aktivitäten	1381
DSN1PRNT: Erforderliche Datasets und Objekte	1381
Nachbereitungs-Aktivitäten	1381
Test-Beispiele	1382
Job-Control	1382

Ausführungs-Protokolle	1382
DSN1SDMP	1386
Überblick	1386
Vorbereitungs-Aktivitäten	1387
DSN1SDMP: Erforderliche Datasets und Objekte	1387
Nachbereitungs-Aktivitäten	1387
Test-Beispiele	1387
Job-Control	1387
Automatisierung des Anstoßes von Utility-Läufen durch Statistik-Analyse (Autonomic Maintenance)	1388
Relevante, automatisierbare Utilities	1390
Autonomic Statistics: Automatisierung von RUNSTATS	1391
Automatisierte Erzeugung von RUNSTATS-Profilen	1392
Pending Status (Advisory und Restricted Status)	1393
Datenentwurf unter DB2	1397
Grafischer Wegweiser durch das Kapitel 10 - Datenentwurf	1397
Rolle des ANSI-SPARC-Drei-Schemata-Ansatzes	1398
Vorgehens-Modell	1400
Entwicklungsgeschichte der relevanten Daten-Modelle	1402
ERM - Entity-Relationship-Modell	1403
Relationen-Modell (RM)	1405
Gemeinsamkeiten von ERM und RM	1407
Logisches Design	1411
Normalisierung	1412
Unnormalisierte Daten	1413
1NF - Atomic Attribute ohne Wiederhol-Elemente	1415
2NF - Volle funktionale Abhängigkeit zu allen PK-Bestandteilen	1417
3NF - Keine transitiven Abhängigkeiten der Nicht-PK-Attribute	1419
Kritische Analyse der erreichten 3NF - Struktur-Qualität	1420
BCNF - Volle Abhängigkeit zum Candidate Key	1421
4NF - Keine paarweisen mehrwertigen Abhängigkeiten	1422
5NF - Ultimative Normal Form - Project-Join-Normalform (PJ/NF)	1424
Abschlussüberlegungen zur Normalisierung	1427
Dokumentation der Daten-Design-Ergebnisse	1428
Bubbles/Blasendiagramme	1428
Relationale Notation	1429
Dependency Diagram	1430
Netzwerk Diagramm	1431
Objektorientiertes State Transition Diagram	1432
DB2-Unterstützungstools	1433
Festlegung der referenziellen Verarbeitungs-Regeln	1434
Grundsätzliche Betrachtung von Referential Integrity für Tabellen ohne Zeitbezug	1434
Darstellung der referenziellen Verarbeitungs-Maßnahmen (RI)	1436
Referential Integrity von Tabellen unter Zeitbezug (Temporale Tabellen- Temporal Referential Constraint)	1438
Einleitende Gedanken	1438
Aufnahme von Zeit-Attributen und Konsequenzen	1439

Festlegung der Primary-Keys (PKs) und Foreign-Keys (FKs)	1440
Festlegung der Geschäfts-Regeln (Business-Rules)	1441
Zusammenfassende Definition der Business-Rules	1442
Festlegung der logischen Trigger für die Business-Rules	1443
Übersicht der kompletten Integritäts-Regeln und -Maßnahmen	1445
Strukturierung der Geschäfts-Regeln und -Aktionen	1447
Umsetzung der Geschäfts-Regeln und -Aktionen in SQL-Funktionen	1448
Konzeptionelle Modelle für Spezial-Konstrukte	1449
Einleitung	1449
Daten unter Zeitbezug - Temporal Tables und Relationen-Modell	1450
Fachliche Zeitaspekte	1450
Einleitung, Terminologie	1450
Komponenten eines Modells für Daten mit Zeitbezug	1451
Auswirkung auf fachliche Informationen und Informations-Zusammenhänge	1453
Aufnahme von Zeit-Attributen und Konsequenzen	1454
Fachliche Funktionsauswirkungen der Zeitunterstützung	1456
Manipulation und Wirkung	1456
Entwicklung der Beispiel-Daten unter Zeitaspekten	1457
Konsequenzen der Zeitunterstützung für Fachbereichs-Benutzer	1458
Konzeptionelle Umsetzung für Zeitaspekte	1459
PK-Festlegung: Objekt-Identifikator plus Zeit- bzw. Versions-Bezug	1459
FK-Festlegung: Objekt-Identifikator mit oder ohne Zeit- bzw. Versions-Bezug	1460
Unser konzeptionelles Modell steckt voller Redundanzen: Wie sieht ein redundanzfreies Modell aus?	1462
Zeit-Leiste für jeden Attribut-Zustand	1462
Datenhaltung für jede Daten-Version bei veränderten Attribut-Zuständen	1464
Funktionale Konsequenzen für Daten mit Zeitbezug	1465
Manipulations-Typen	1465
Manipulations-Festlegungen hinsichtlich der Gültigkeitszeitleisten	1466
Referenzielle Regeln für Daten mit Zeitbezug (RI-Bedingungen)	1468
Informations-Bereitstellungs-Typen	1470
Festlegung der konzeptionellen Views	1471
Bilden von Schemata für spezielle Daten-Sichten	1471
OLAP	1472
Data Warehouse: Cubes und Dimensionen	1473
Fakten (Facts)	1473
Maßeinheiten (Measures)	1473
Dimensionen (Dimensions)	1474
OLAP-Meta-Modell: Objekt-Typen	1478
Checkliste für das logische Daten-Design	1479
Fachliches Daten-Modell	1479
Konzeptionelles Daten-Modell	1480
Physisches Design	1482
Aufgabenstellungen	1482
Ableiten der physischen Benutzerobjekte aus logischen Relationen	1483
Zielsetzungen eines physischen Daten-Modells	1483
Kandidaten für die physische Implementierung	1484
Bewertung der Auswirkungen logisch definierter Zugriffspfade	1484
Aufwand bei der Datenbeschaffung	1486

Individuelle Performance-Relevanz	1489
Auswirkungen der Sperr-Maßnahmen	1490
De-Normalisierungs-Maßnahmen	1491
Daten-Zusammenführung	1491
Trennung der Daten (Verteilung)	1493
Trennung der Daten zwischen Base- und History-Table bei Daten-Versionierung von Temporal Tables	1494
Bilden zusätzlicher Redundanzen	1496
Bilden von Schemata für spezielle Datenhaltungs-Konzepte	1498
Data Warehouse	1498
Terminologie und Komponenten	1498
MOLAP, ROLAP und HOLAP	1500
Star-Schema	1502
Snowflake-Schema	1503
Struktur-Besonderheiten beim Einsatz von MQTs	1504
Konsequenzen der De-Normalisierung	1505
Indizes	1506
Unterstützung der referenziellen Integrität (RI) durch DB2	1507
Terminologie und Einrichtung unter DB2 (Basis-Tabellen, keine Temporal Tables)	1507
Definierbare Strukturen	1509
Zulässige Einzel-Struktur-Beziehungen	1509
Umsetzung der M : N - Struktur-Beziehungen	1510
Logische referenzielle Struktur	1511
Physische referenzielle Struktur (Tablespace-Set)	1517
Gesamt-Überblick über die DB2-RI-Unterstützung	1519
Anforderungen an PK und FK	1521
DB2-Primary-Key (PK) und Parent Key	1521
DB2-Foreign-Key (FK)	1522
Verwaltung der RI-Definitionen im Katalog	1523
Katalog-Informationen für RI	1523
RI-DDL-Beispiele	1524
Argumente für und wider den Einsatz von DB2-RI	1526
Besonderheiten der Referential Integrity von Temporalen Tabellen (Temporal Referential Constraint)	1527
Unterstützung der Business Rules mit Funktionen und Triggern	1528
Datenbank-Funktionen zur Integritäts-Gewährleistung und zur Verlagerung von Anwendungs-Funktionalität	1528
De-Normalisierungs-Auswirkungen	1529
Zuordnung der Regeln und Maßnahmen zu den physischen Strukturen	1529
Funktionale Konsequenzen bei Nicht-Nutzung von DB2-RI	1531
Funktionale Konsequenzen bei Struktur-De-Normalisierungen	1532
Festlegung der sonstigen DB2-Objekte mit ihren Ausprägungen	1534
Festlegung der sonstigen physischen Ressourcen	1534
DB2-Unterstützungstools	1534
Checkliste für das physische DB2-Design	1535
Datenspeicherung intern und extern	1537
Grafischer Wegweiser durch das Kapitel 11	1537
Übersicht der Speicherungsformen	1538
Space-Management	1540

Verwaltung und Adressierung der Daten innerhalb des Page Sets	1540
Page Sets	1542
Linear Page Sets	1542
Partitioned Page Sets	1543
Regular Partitioned Tablespace	1543
Universal Page Sets	1544
LOB Page Sets	1545
XML Page Sets	1546
Page-Nummerierungs-Konzept: Absolut (APN) oder Relativ (RPN)	1547
Page Typen des File Page Sets	1548
Header Page und System Pages	1549
Kontrolle der Umstellung des RBA/LRSN-Formats von Basic auf Extended in einer Header-Page	1550
Space Map Pages	1551
Daten Pages	1552
Satzformat: Basic-Record-Format (BRF) oder Reordered-Row-Format (RRF)	1553
Indikatoren für LOB- und XML-Werte in der Basis-Datenzeile	1554
Adressierung der Daten-Zeilen	1554
Speicherung und Adressierung bei Hash-organisierten Tabellen	1555
Insert-Strategie innerhalb der Daten-Pages	1557
Fast Un-clustered INSERT (INSERT-Algorithmus 2)	1558
Update-Strategie bei variablen Zeilen	1559
Delete-Strategie innerhalb der Daten-Pages	1559
Freespace-Verwaltung der Daten-Pages	1560
FREEPAGE und PCTFREE	1560
PCTFREE FOR UPDATE	1561
MAXROWS	1561
Plattenplatz-Bedarfs-Ermittlung	1562
Plattenplatz-Bedarfs-Ermittlung für LOB-Werte	1564
Besonderheiten bei Inline-LOBs	1565
PRIQTY	1566
SECQTY	1566
Kompression der Daten	1567
Page Typen des Index Page Sets	1569
Header Page und System Pages	1569
Space Map Pages	1569
Index Pages	1569
Non-Leaf Pages	1570
Leaf Pages	1571
Insert-Strategie innerhalb der Index-Pages	1572
Update-Strategie innerhalb der Index-Pages	1573
Delete-Strategie innerhalb der Index-Pages	1573
Freespace-Verwaltung der Index-Pages	1573
Verteilung der Non-Partitioned-Index Daten mit Hilfe von PIECESIZE	1573
Plattenplatz-Bedarfs-Ermittlung für Indizes	1574
Unterstützung von Page-Größen > 4-KB im Index	1576
Index-Komprimierung	1577
DSN1COMP-Protokoll	1578
Nutzung des virtuellen Adressraums	1579

Nutzung der z/OS-Architektur	1579
Erweiterte Nutzung der 64-Bit-Architektur	1580
Datenpool-Konzept	1582
RID-Pool	1583
Sort-Pool	1584
EDM-Speicherbereiche	1585
Fast Traverse Block (FTB) - In-Memory-Index-Optimierung - Fast Index Traversal	1586
Compressions-Dictionary	1587
Materialisierung von LOB-Werten	1587
IRLM-Locks	1587
LE-Tokens	1587
Bufferpool-Konzept	1588
Group Bufferpools (GBP)	1590
Virtuelles Bufferkonzept	1590
Zuordnung der Bufferpools zu Tablespaces und Indexspaces	1590
Unterstützung verschiedener Page-Größen	1591
Bufferpool-Page-Größe und VSAM-CI-Größe	1591
Fixierung von Buffern im Realspeicher	1591
Einbindung der Bufferpools in die System-Schichten	1592
Lese-Anforderungen	1593
Direktes Einlesen (Synchronous Read)	1593
Sequential Prefetch	1593
Dynamic Prefetch (Sequential Detection)	1594
Sequential Detection	1595
List Prefetch (List Sequential Prefetch)	1596
Schreib-Anforderungen	1597
Zustandsformen der Bufferpool-Page-Typen	1598
Bufferpool-Thresholds	1599
Threshold-Parametrisierungs-Empfehlungen für LOB- und XML-Werte	1600
Mögliche Strategien der Bufferpool-Einrichtung	1601
Bufferpool-Hit Ratio und MUPA	1602
Bufferpool-Besonderheiten für LOB- und XML-Werte	1603
In-Memory-Bereiche (Virtuelle Speicherbereiche für spezielle Aufgabenstellungen)	1604
VSAM-Datasets	1605
Nutzung von DFSMS (Storage Management Subsystem)	1607
DB2-Datensicherheitseinrichtungen	1608
Grafischer Wegweiser durch das Kapitel 12 - Datensicherheitseinrichtungen	1608
DB2-Sicherheitsinstrumentarium	1609
Transaktionsabwicklung	1611
Terminologie und Transaktions-Definition	1611
Konsistenz-Koordination bei Einsatz mehrerer Ressource-Manager	1613
LUW-, UOW- und UOR-Konzept	1615
Vorgangsabwicklung unter TSO oder einem Client ohne UOW-Konzept	1617
Vorgangsabwicklung unter CICS oder IMS mit UOW-Konzept	1618
Die Wirkung von SAVEPOINT und ROLLBACK TO SAVEPOINT	1619
Commit-Konzepte	1620

Überblick über Read-only-, Single-Phase- und Two-Phase-Commit	1620
Two-Phase-Commit (2-Phasen-Commit)	1621
UOR-Status nach einem Fehler	1622
Probleme und Lösungswege der Konkurrenzverarbeitung	1623
Verlorener Update	1624
Zugriff zu Daten, deren UOW noch nicht abgeschlossen ist	1625
Wiederholtes Einlesen von Daten innerhalb einer UOW	1626
Lösungswege zum Abbau der Konkurrenz-Problematiken	1627
Deadlock	1629
Vermeidung von Deadlocks	1630
Erkennung von Timeouts und Deadlocks	1632
DB2-Sperrverfahren	1633
Überblick	1633
Transaction Lock	1634
Lock-Charakteristiken	1634
Lock-Objekt	1636
Implizite Lock-Auswirkungen auf die DB2-Systemressourcen	1636
Lock-Dauer	1638
Lock-Niveau	1640
Lock Eskalation	1642
SQL-Statement: LOCK TABLE	1642
Lock-Modus	1643
Lock-Modus auf Lock-Niveau: Tablespace, Table bzw. Partition	1643
Lock-Modus auf Lock-Niveau: Page bzw. Row	1645
Zusammenfassung: Aktivierung von Lock-Niveau und Lock-Modus	1646
Lock Promotion	1647
DB2-Lock-Modus-Entscheidungen	1647
Isolation-Level	1649
RR - Repeatable Read	1649
RS - Read Stability	1650
CS - Cursor Stability	1650
UR - Uncommitted Read	1651
Vorgabemöglichkeiten des Isolation-Levels und Wechselwirkung in Package und Plan	1652
Zugriff auf aktuelle Daten trotz Parallel-Manipulation ("currently committed") oder auch CONCURRENT ACCESS RESOLUTION	1653
Unterschiede zwischen Latch und Lock	1655
Lock Avoidance	1655
Lock-Besonderheiten beim Cursor-Einsatz	1657
Optimistic Concurrency Control (Optimistic Locking)	1657
LOCK-Besonderheiten bei LOB- und XML-Daten	1658
Drain Lock, Drains und Claims	1659
IRLM - Internal Resource Lock Manager	1660
Data Sharing (DS)	1661
Lock-Typen bei Data Sharing	1661
Speicher-Strukturen einer Data Sharing Group	1662
Überblick	1662
Inter-DB2-Interesse im Data Sharing	1664
Übersicht der memberübergreifenden Entscheidungs-Maßnahmen	1664

IRLM-Lock-Konzept im Data Sharing	1665
L-Locks und P-Locks	1665
Parent und Child-Locks und das Zusammenspiel der L-Locks und P-Locks	1666
Lock-Optimierungen bei Data Sharing	1667
Zusammenfassung der DB2-Lock-Mechanismen	1668
Welche relevanten Parameter steuern die Lock-Mechanismen?	1668
Checkliste für den Einsatz der DB2-Lock-Mechanismen	1669
Übersicht: Sperrniveau und Lock-Maßnahmen	1671
Konsequenzen der DB2-Sperrverfahren auf die Anwendungs-Entwicklung	1672
Zusammenspiel von Isolation-Level, Lock-Modus und -Dauer	1672
Verhinderung eines Parallel-Updates	1673
Einsatz des Cursor-Konzeptes	1673
Cursor-Konzept oder SELECT WITH RR bzw. RS?	1674
Optimistic Locking Strategie	1675
Parallelverarbeitung von Online- und Batch-Anwendungen	1676
Logging-Einrichtungen	1677
Steuerungsfunktionen für die Logging-Abwicklung	1679
Logging-Philosophie	1679
Logging-Hierarchie	1680
Führen und Ausschreiben von LOG-Informationen	1682
Maßnahmen zur Reduzierung des LOG-Aufwands	1684
Konsistenzsicherung innerhalb des DB2-Subsystems (REDO, UNDO)	1685
Manuelle, nicht durch DB2 überwachte Konsistenz-Maßnahmen	1686
Konsistenzsicherung mit Allied Agents	1686
Konsistenzsicherung bei DB2 Data Sharing (DS)	1687
Storage Server	1688
ESS - Enterprise Storage Server	1688
AUDIT TRACE	1689
Regelbasierte Protokollierung sicherheitsrelevanter Ereignisse (Policy-based Audit Capability)	1690
Übersicht der Komponenten	1690
Audit-Kategorien	1691
Problembereiche der Datensicherheitseinrichtungen	1693
Anwendungsprogrammierung unter DB2	1695
Grafischer Wegweiser durch das Kapitel 13 - Anwendungsprogrammierung	1695
Anwendungsprogrammierung: Relevante Komponenten im Überblick	1696
Einsatzspektrum von DB2	1697
Die verschiedenen Programm-Typen	1698
Abgrenzung Dialog-, Online-, Batch- und Client/Server-Verarbeitung	1698
Was versteht man unter Batch-Verarbeitung?	1699
Brauchen wir noch die Batch-Verarbeitung?	1700
Abgrenzung der Online- und Batch-Charakteristiken	1702
Client/Server-Charakteristiken	1703
Wahl des geeigneten Trägersystems	1704
Dialog-Verarbeitung	1704
Foreground-Verarbeitung	1704

Background-Verarbeitung	1705
Batch-Verarbeitung	1706
Permanent laufende Anwendungen, Started Tasks (STC) und Message-Konzept	1707
Die verschiedenen Betriebsformen	1708
Normale Betriebszeiten	1708
Erweiterte Betriebszeiten (erweiterte Verfügbarkeit)	1708
24-Stunden-Betrieb (permanente Verfügbarkeit)	1709
Zielsetzungen eines Multi-User-Betriebs	1710
Ein Wort zur 'Permanenz'	1710
Programm-Design-Überlegungen	1711
Programmübergreifende Maßnahmen	1711
Der Einsatz von SQL: Möglichkeiten und Restriktionen	1713
Programmentwicklung unter DB2	1715
Unterstützte Programmiersprachen	1715
Embedded SQL	1715
Call Level Interface für C- und C++-Programme (CLI bzw. ODBC)	1715
Java-Sprachschnittstellen (JDBC und SQLJ)	1716
JDBC-API für Java-Anwendungen	1716
SQLJ-API für Java-Anwendungen	1716
pureQuery für Java-Anwendungen	1716
Entwicklung einer DB2-Anwendung	1717
Überblick der relevanten Entwicklungs-Komponenten	1717
Entwicklungs-Komponenten für Embedded SQL	1718
DB2I: DCLGEN-Generierung	1719
SQL-Preprozessor (SQL Statement-Coprozessor bzw. DB2-Coprozesor)	1720
DB2-Precompiler	1721
Aufgaben des Precompilers	1721
Precompiler bzw. Coprocessor-Optionen (Processing Options)	1722
Standard-SQL	1725
Modifikationen der Programm-Source durch den Precompiler	1727
Entwicklungs-Komponenten für Java-Anwendungen	1729
JDBC-API	1729
SQLJ-API	1729
Entwicklungs-Komponenten für CLI/ODBC-Anwendungen	1730
BIND-Prozess	1731
Aufgaben des BIND-Prozesses	1731
Objekte des Bind-Prozesses	1732
DBRM: Database Request Module	1732
Package	1732
Package-Zuordnung zu mehreren Collections	1734
Was ist eine invalidated Package?	1735
Plan	1736
Package-Liste im Plan: PKLIST	1738
Nutzung von remote Packages	1740
Packages von Routinen	1740
Die Zuordnung eines Plans zu einem Programm	1741
Dynamische Plan-Zuordnung im CICS	1742
Was ist ein invalidated Plan?	1742
Die verschiedenen BIND- und REBIND-Typen	1743

STATIC BIND: BIND PACKAGE, BIND PLAN	1743
REBIND PACKAGE, REBIND PLAN	1744
FREE PACKAGE, FREE PLAN, DROP PACKAGE	1744
DYNAMIC BIND	1744
AUTOMATIC REBIND	1745
INCREMENTAL BIND	1745
DEFERRED BIND	1745
Relevante Katalog-Tabellen	1746
Konvertierung der in Plänen eingebundenen DBRMs in Packages	1746
PLAN Management (müsste eigentlich Package Management heißen)	1747
Binder (Linkage Editor)	1748
Aktivieren der Attachment-Facility-Schnittstellen	1748
Beispiel- und Hilfs-Programme: DSNTIAD, DSNTDP2/4 und DSNTIAUL	1749
Übersicht	1749
DSNTIAUL	1750
DSNTIAUL-Return-Codes	1751
DSNTIAUL-Beispiel	1751
DSNTEP2/DSNTEP4	1753
Datasets für DSNTEP2/DSNTEP4	1753
DSNTEP2/DSNTEP4-Return-Codes	1753
Eingabe-Parameter für DSNTEP2 bzw. DSNTEP4	1754
DSNTEP2/DSNTEP4-Beispiel	1755
SQL-Spracheinsatz in Anwendungsprogrammen	1756
Übersicht der Sprachschnittstellen	1756
Interactive SQL, Embedded SQL, CLI, Static SQL und Dynamic SQL	1756
EXEC-Level-API: Embedded SQL	1757
Behandlung von Datenmengen	1759
CLI: Call-Level-Interface bzw. ODBC-Schnittstelle	1760
Java-Schnittstellen	1762
JDBC - Java Database Connectivity	1764
SQLJ - SQL for Java	1765
REXX-Schnittstellen	1768
Grundstruktur eines DB2-Programmes im EXEC-Level	1769
Relevante Komponenten	1769
Basis-Codier-Regeln für SQL-Statements	1770
Allgemeine Regeln	1770
Assembler-Besonderheiten	1771
C-Besonderheiten	1771
COBOL-Besonderheiten	1772
PL/I-Besonderheiten	1775
Definition der Variablen und Daten-Strukturen	1776
Definition der SQLCA (SQL Communication Area)	1777
Beschreibung der SQLCA-Struktur-Komponenten	1777
SQLCA-Definitions-Struktur: Beispiele COBOL und PL/I	1780
Tabellen- und View-Deklarationen	1781
Host-Variablen und NULL-Indikatoren	1783
Typen: Host-Variablen und Host-Strukturen	1783
Einlesen von NULL-Values (NULL-Markierungen)	1785

Setzen von NULL-Werten (NULL-Markierungen)	1787
Einsatzspektrum von Host-Variablen und Indikatoren	1789
Erweiterte NULL-Indikator-Variablen (Extended Indicator Variables)	1790
Host-Variablen-Arrays	1792
Unterstützte Daten-Typen (mit SQLTYPE)	1793
Konvertierungsregeln zwischen DB2-Spalten und Host-Variablen	1795
Fehlerbehandlung	1798
Überblick	1798
Returncode-Analyse	1799
SQLCA - SQL Communication Area	1799
Auszug einiger relevanter SQLCODEs und SQLSTATES	1800
WHENEVER	1801
DSNTIAR, DSNTIAC	1801
GET DIAGNOSTICS-Statement	1802
Behandlung schwerer Fehler	1804
Cursor-Konzept	1805
Übersicht der Sprachmittel	1805
Überblick über die Cursor-Typen	1808
Ist die Result Table des Cursors read-only oder updateable?	1809
Non-Scrollable Cursor	1811
Deklaration des Cursors: DECLARE CURSOR	1811
Eröffnen des Cursors: OPEN	1813
Bereitstellen einzelner Zeilen der Result Table: FETCH	1813
Positioned Update und Delete: WHERE CURRENT OF	1814
Schließen des Cursors: CLOSE	1815
Explizites Schließen des Cursors	1815
Implizites Schließen des Cursors	1815
Statische und dynamische Result Table und ableitbare Integritäts- und Performance-Probleme beim Cursor-Konzept	1816
Mengenverarbeitung auf Realtime-Daten: das ewige Rätsel	1818
Scrollable Cursor	1820
Positionierungs-Möglichkeiten in der Result Table	1820
Cursor- und FETCH-Typen und ihre Sichten auf die Daten der Result Table	1821
Statischer Scrollable Cursor (Statisches Cursor-Modell)	1822
Deklaration des Cursors: DECLARE CURSOR .. SCROLL ..	1822
Informationsbereitstellung eines Static Scrollable Cursors	1823
Manipulationen über die Result Table eines Static Scrollable Cursors	1825
Dynamischer Scrollable Cursor (Dynamisches Cursor-Modell)	1826
Deklaration des Cursors	1826
Verarbeitung einer Zeilen-Gruppe (Rowset)	1827
Einlesen eines Rowsets: FETCH FOR n ROWS	1827
COBOL-Programm-Beispiel eines Rowset-Cursors mit GET DIAGNOSTICS	1828
Scrollable Cursor: Positionierung mit Rowsets	1832
Einfügen eines Rowsets: INSERT .. FOR n ROWS	1833
Analyse der Rowset-Verarbeitungs-Ergebnisse mit GET DIAGNOSTICS	1833
Einfluss des UOW/UOR-Konzepts auf die Cursor-Verarbeitung	1834
Zusammenfassung der relevanten Cursor-Typen und der zu definierenden Klauseln	1836
Besondere Programmier-Techniken unter DB2	1837
Effiziente Bereitstellung einer Result Table	1837
Relevante WHERE- und ORDER BY-Komponenten	1838

Aufsetzen auf einen Composite Key	1839
Regeln für das Aufsetzen	1840
Online-Anwendungen	1841
Online-Parallel-Update	1841
Online-Browsing/Blättern	1844
Batch- und Multi-User-Anwendungen	1845
Langlaufende oder ressourcenintensive Anwendungen	1845
UOW und UOR	1846
Die Lebensdauer einer UOW	1846
Wirkungen beim Abschluss einer UOR	1846
Einsatz von COMMIT (explizit oder implizit)	1847
Probleme, wenn kein COMMIT eingesetzt wird	1847
Was sind Kandidaten für die COMMIT-Schreibung?	1848
Konsequenzen des COMMIT-Einsatzes	1849
Unterstützung von Rollback und Restart der Batch-Programme	1850
Empfehlungen zum Design von Batch-Langläufern	1851
Anwendungen in verteilten Datenbankumgebungen	1852
Zugriffstypen und Protokolle	1852
DB2-Privat-Protokoll-Zugriff und DRDA-Zugriff	1852
Two-Phase-Commit (2-Phasen) in einer verteilten Datenbank-Umgebung	1853
Syntax-Anforderungen unterschiedlicher Systeme	1853
Programmvorbereitungen, Pläne und Packages	1854
SQL-Statement-Einsatz in Programmen	1855
Vorteile und Beschränkungen der Nutzung von dreiteiligen Objektnamen	1857
Behandlung von LOB- und XML-Werten bei remote Zugriffen	1857
Performance- und Integritäts-Aspekte	1858
Block-Fetch	1859
Limited und Continuous Block-Fetch	1860
Überblick	1860
Behandlung von LOB- und XML-Werten	1862
Grundsätzliche Verarbeitungsmöglichkeiten für LOB-Werte	1862
LOB-Daten-Typen und ihre Host-Variablen-Repräsentation	1864
LOB-Materialisierung	1866
LOB-Referenzierung mittels LOB-Locator	1867
LOB-Verarbeitung mittels File-Referenz-Struktur	1868
Grundsätzliche Verarbeitungsmöglichkeiten für XML-Werte	1869
XML-Funktionen	1870
Bereitstellungs-Techniken für XML-Daten	1871
SQL-XML-Funktionen: Beispiele für Komposition und De-Komposition	1872
XML-Daten-Typen und ihre Host-Variablen-Repräsentation	1873
Entwicklung und Einsatz von Stored Procedures	1875
Programmtechnische Umsetzung	1875
Aufruf-Varianten einer Stored Procedure	1875
Parameter der Stored Procedure	1875
Codier-Beispiele: Aufruf und Parameterübergabe	1876
Stored Procedure ohne Result Sets	1876
Entwicklung und Einsatz von User-defined Functions	1878
Überblick der Nutzungsmöglichkeiten	1878

Aufruf-Varianten einer externen Funktion	1879
Programmtechnische Besonderheiten	1880
Scratchpad-Bereich	1880
Parallellauf-Besonderheiten	1880
Parameter der User-defined Function	1880
Codier-Beispiele: Aufruf und Parameterübergabe	1881
External Scalar Function	1881
External Table Function	1883
Entwicklung und Einsatz von Triggern	1885
Codier-Beispiele: Aufruf mit Übergabe einer Transition-Table	1885
MQ-Series-Funktionen	1886
Nutzung von Globalen Variablen	1889
Session-Variablen und Scalar-Funktion GETVARIABLE	1889
Behandlung unterschiedlicher Codier-Sets für Host-Variablen	1891
CAF - Call Attachment Facility	1892
Vorteile der CAF-Nutzung	1892
Entwicklung eines CAF-Programms	1892
Nutzbare CAF-Funktionen	1893
Implizite Connection	1894
Explizite Connection	1894
Generelle Ausprägungen einer CAF-Connection	1894
RRSAF - Recoverable Resource Manager Services Attachment Facility	1895
Vorteile der RRSAF-Nutzung	1895
Entwicklung eines RRSAF-Programms	1895
Nutzbare RRSAF-Funktionen	1896
Abschluss einer UOW	1897
Universal Language Interface - Universelle Schnittstelle (DSNULI)	1898
Dynamic SQL	1899
Abgrenzung zu Static SQL	1899
Verhalten von Dynamic SQL-Statements (Statement Behavior)	1900
Funktionsspektrum von Dynamic SQL	1901
Manipulationen und Informationsanforderungen	1901
Parameter Markers (?)	1902
SQLDA - SQL Descriptor Area	1903
Struktur-Komponenten der SQLDA	1904
Beispiele der vom Precompiler/Preprozessor generierten SQLDA-Host-Variablen-Strukturen	1905
Manipulationen (Non-SELECT)	1906
Feste Parameter	1906
Variable Parameter	1908
Informations-Anforderungen (SELECT)	1909
Feste Parameter: Fixed-List-SELECT	1909
Variable Parameter: Varying-List-SELECT	1909
Dynamic Statement Cache	1912
Prepare-Typen und -Ausführungsarten	1913
Ersetzen von Literalen durch generische &-Variablen (Literal-Replacement bzw. Statement Concentrator)	1914

Sperr-Dauer bei dynamischen SQL-Statements	1915
Resource Limit Facility (RLF) - DB2 Governor	1916
Vor- und Nachteile von Dynamic SQL	1917
Vorteile	1918
Nachteile	1919
Programmier-Empfehlungen	1920
SQL-Performance	1923
Grafischer Wegweiser durch das Kapitel 14 - SQL-Performance	1923
Grundlagen der SQL-Performance	1924
Einleitung, Ansätze der Optimierung	1924
Leistungsbeeinflussende Faktoren	1924
Optimierungs-Möglichkeiten und -Maßnahmen	1925
Komponenten der Datenbeschaffung (Stages)	1927
Aufgabenteilung von RDS, DM und BM	1927
DB2-Tools zur Unterstützung von Performance-Maßnahmen	1929
Übersicht	1929
DB2-Trace	1931
Leitlinien für Traces	1931
Trace-Typ	1932
Thread-Management	1934
Installations-Parameter für das Thread-Management	1934
Die Rolle von Prozess, Enklave, Transaktion, Thread, TCB und SRB	1935
Allied Thread	1936
Thread-Lebenszyklus	1936
Thread-Wiedernutzung (reuse)	1937
DBAT - Database Access Thread	1937
Die relevanten Zeiten einer Statement-Abwicklung	1939
DB2-Accounting-Zeiten	1939
Parallel-Verarbeitung	1941
Parallel-Verarbeitungs-Typen	1941
Kandidaten für Parallel-Verarbeitung	1941
Query-Typ: CPU bound oder I/O bound	1943
Effekte der Parallel-Verarbeitung	1943
Aktivierung der Parallel-Verarbeitung	1943
Voraussetzungen für die Parallel-Verarbeitung	1944
Restriktionen bei der Nutzung der Parallel-Verarbeitung	1944
Technische Abwicklungs-Modelle	1945
Dynamic Record-Range-Partitioning anstelle Static Keyrange-Partitioning	1945
Straw-Modell für eine effiziente Workload-Verteilung	1946
Bedeutung der Statistiken für Optimizer und Zugriffspfad-Analyse	1947
Überblick der DB2-Statistik-Komponenten	1947
Woher bezieht der Optimizer seine Informationen zur Zugriffspfad-Bestimmung?	1949
Statistik-Feedback: Dokumentieren und Ausschreiben fehlender oder kritischer Statistiken durch den Optimizer	1950
Zugriffspfad-Analyse des DB2-Optimizers	1952
Aufgaben des Optimizers	1952

Parsing und Parse Tree	1952
Praktisches Beispiel der Optimizer-Maßnahmen im Rahmen des Parsings	1953
Entscheidungs-Faktoren des Optimizers und Zugriffspfad-Varianten	1954
Optimization	1955
Bedingte Verarbeitungsanweisung (Prädikat)	1955
Erläuterung	1955
Prädikat-Typen	1956
Prädikat-Kategorien	1956
Einsatz von Literalen (Konstanten) oder Variablen (Input-Variablen)	1957
Filterung	1959
Inhaltliche Verteilung von Datenwerten	1960
Filter-Faktor (FF)	1960
Aufgabenstellung und Wirkung	1960
Filter-Faktor-Typen	1961
Filter-Faktor für gleichmäßige Streuung: Einfache/Simple Prädikate	1962
Filter-Faktor für gleichmäßige Streuung: Interpolations-Formeln für Range-Prädikate	1962
Filter-Faktor für gleichmäßige Streuung: Default-Filter-Faktoren für Interpolation-Formeln (Input-Variablen)	1963
Filter-Faktor für ungleichmäßige Verteilung: Frequenz- (FREQUAL) und Werteverteilungs-Statistiken (KEYCARD) für Spalten oder Spalten-Gruppen	1964
Filter-Faktor für ungleichmäßige Verteilung: Häufigkeitsverteilung (HISTOGRAM) für Spalten oder Spalten-Gruppen	1964
Manuelle Ermittlung des Filter-Faktors	1965
Index-Probing (Safe Query Optimization Technik)	1966
Adaptiver Index zur Ausführungszeit (Execution time adaptive index)	1967
STAGE1, STAGE2 und INDEXABLE	1968
Indexable Prädikate	1971
Komplexe Prädikatvorgaben	1972
Column-Correlation	1973
Reihenfolge der Prädikat-Bewertung	1974
Aufwands-Kalkulation der SQL-Anforderung	1975
Prozess-Kostenschätzung	1975
Auswahl des Zugriffspfades	1975
IO-Kostenermittlung	1976
Timeron	1977
DB2-Zugriffspfade auf die Daten	1978
Analysemöglichkeit der Zugriffspfade	1978
Rahmenbedingungen	1978
Komponenten der Zugriffspfad-Analyse und EXPLAIN_Tables	1979
EXPLAIN und EXPLAIN_TABLES	1980
EXPLAIN_Tables: Überblick	1980
EXPLAIN	1981
Einsatz von Input-Variablen beim EXPLAIN	1982
Beeinflussung der Optimizer-Entscheidungen durch OPTHINT oder Überschreibung von Selektivitäts-Statistiken	1982
EXPLAIN-Statement Beispiele	1983
EXPLAIN eines einfachen SQL-Statements mit Analyse der PLAN_TABLE	1983
DSN_STATEMNT_TABLE: Analyse-Möglichkeit	1984
DSN_STATEMENT_CACHE_TABL: Analyse von dynamischen SQL-Statements aus dem Dynamic Statement Cache	1985
EXPLAIN einer Package ohne REBIND	1987
DSN_PREDICATE_TABLE und DSN_PREDICATE_SELECTIVITY und DSN_STAT_FEEDBACK-Tabellen: Beispiel-Szenario	1988
Beispielhaftes Statement als Vorgabe mit EXPLAIN-Statement	1988
Inhalte der Tabelle DSN_PREDICAT_TABLE	1988
Inhalte der Tabelle DSN_FILTER_TABLE	1989

Inhalte der Tabelle DSN_PREDICATE_SELECTIVITY	1990
Inhalte der Tabelle DSN_STAT_FEEDBACK	1990
I/O-Zugriffs-Typ	1991
Direktes Einlesen (Synchronous Read)	1991
Sequential Prefetch und Dynamic Prefetch	1992
Dynamic Prefetch	1993
List Prefetch	1993
Nutzung dieser Technik	1993
List-Prefetch im Index	1993
List-Prefetch auf Daten-Pages	1994
Der Ausweis des List-Prefetches in der PLAN_TABLE	1995
Vor- und Nachteile des List-Prefetch-Verfahrens	1996
Einfache Zugriffspfade	1997
Tablespace Scan (File Page Set Scan)	1999
Index-Zugriffe	2000
Matching Index Scan	2000
Non-Matching Index Scan	2002
Equal Unique Index Access	2003
IN-List Index Scan	2004
Einzelne IN-Liste	2004
Mehrere IN-Listen	2005
One-Fetch Access	2006
Index-Only Access	2007
Range-List-Access (Aufsetzlogik, Inclusion-Methode, Blätter-Funktionalität, Pagination)	2008
Multiple Index Access	2010
Multiple DocID Index Access über XML-Indizes	2011
Direkt-Zugriff auf die Daten unter Umgehung von Indizes	2013
Einleitung und unterstützte Direkt-Zugriffs-Typen	2013
Direkter Zeilenzugriff über einen Unique Hash-Key	2014
Direkter Zeilen-Zugriff über die ROWID	2016
Direkter Zeilen-Zugriff über die RID-Funktion	2017
Komplexe Zugriffspfade	2018
Übersicht	2018
Join	2019
Composite Table und New Table, Outer Table und Inner Table	2019
Ausweis von Joins in der PLAN_TABLE	2020
Join-Typ	2021
Inner Join	2021
Outer Join	2022
Historische Entwicklung der Outer Joins	2024
Joins im Bereich des Data Warehouses: Star Join oder Paarweiser Join für Star-Schema und Snowflake-Schema	2026
Wann wird der Star Join genutzt?	2028
Ausweis eines Star Joins in der PLAN_TABLE	2029
Wann wird der Paarweise Join genutzt?	2030
Ausweis eines Paarweisen Joins in der PLAN_TABLE	2031
Klassische Join-Methoden	2032
Nested Loop Join	2032
Materialisierung/Sortierung bei einem Nested Loop Join	2035
Merge Scan Join	2036
Hybrid Join	2038

Beeinflussung der Outer-Table-Auswahl und der Join-Methode	2040
Komplexe Query-Typen	2041
Correlated und Non-Correlated Query-Typen	2041
Query-Block	2042
Der Nummerierungs-Wahnsinn der QBLOCKNO	2042
Subquery	2043
Non-correlated Subquery	2044
Correlated Subquery	2046
Table Expression	2048
Common Table Expression	2048
Rekursives SQL	2049
Nested Table Expression	2050
Scalar Full-Select	2051
SELECT FROM FINAL/OLD TABLE mit Data-Change-Statement	2052
Set-Operator	2054
Trigger-Package	2055
Sortierung	2056
Sparse Index (Hash-Type-Join)	2058
Einige Worte zum Scrollable Cursor	2059
Parallel-Verarbeitung	2060
Materialisierung	2061
Spezielle Zugriffspfade	2062
IBM DB2 Analytics Accelerator for z/OS (ehemals IBM Smart Analytics Optimizer)	2062
Überblick	2062
Technisches Konzept	2063
Integration ins DB2 z/OS	2064
AQR - Automatic Query Rewrite bei Einsatz von MQTs (Materialisierte Query Tables)	2065
Überblick	2065
Nutzungs-Potential von MQT und AQR	2069
Temporal Tables	2070
Einleitung	2070
Bi-Temporal Table	2071
DDL-Beispiele für SEMTYP, REFERENT und SEMINAR (als Auszug aus dem Seminar-Daten-Modell)	2071
Explain-Analysen	2074
Einfache SELECT-Statements	2074
Komplexe SELECT-Statements (Join)	2076
BUSINESS_TIME	2076
SYSTEM_TIME	2077
Manipulations-Statements	2080
Beeinflussung der Zugriffspfad-Entscheidungen	2081
Statement-Modifikationen durch den Optimizer	2082
Prädikat-Modifikationen und Einfügungen	2082
Modifikationen der Join-Verarbeitung	2084
Konvertierung von Subqueries in Joins	2085
Transformation von Non-Correlated Subqueries in Correlated und umgekehrt (De-Correlation)	2085
Materialisierungs-Entscheidung	2085
Prädikate, die bei der Evaluierung immer 'wahr' oder immer 'falsch' ergeben	2086
OR 0 = 1	2086
OR 0 <> 0	2086

Möglichkeiten für Anwendungsentwickler bei der Beeinflussung des Zugriffspfads	2087
Veränderung von Objekt-Strukturen	2087
Manipulation der Statistiken	2090
Umschreiben von SQL-Statements	2091
OPTIMIZE FOR n ROWS	2093
BIND-Parameter REOPT und seine Auswirkungen	2094
Extended Optimization: Beeinflussung der Zugriffspfad-Optimierung durch Vorgabe von Performance-Einfluss-Faktoren	2095
Überblick der Komponenten	2095
Selektivitäts-Statistiken: Überschreibung der vom Optimizer herangezogenen Statistikwerte (Statement-level selectivity overrides)	2097
Überblick der relevanten Komponenten	2097
Szenario zur Vorgabe der Statistik-Vorgaben	2098
DSN_USERQUERY_TABLE: Beispiel Selektivitäts-Statistiken	2099
DSN_PREDICATE_SELECTIVITY : Beispiel Selektivitäts-Statistiken	2100
BIND/REBIND mit Ergebnis-Protokollierung	2100
OPTHINT: Beeinflussung der Entscheidung des Optimizers mittels Optimization-Hints	2101
Komponenten	2101
Voraussetzungen zur Nutzung	2103
Prioritäten der Berücksichtigung von Optimization Hints	2103
Statement-Level Optimization Hint - Nutzung des Zugriffspfad-Repositories (Access Path Repository)	2104
Statement-Level Optimization Hint: Parameter-Spezifikationen (Parameter Hints)	2105
Statement-Level Optimization Hint: Zugriffspfad-Empfehlungen (Access Path Hints)	2107
BIND/REBIND mit Ergebnis-Protokollierung	2108
User-Level Optimization Hint: Vorgabe von Zugriffspfad-Empfehlungen für den Optimizer über die PLAN_TABLE (OPTHINT - Optimization Hints)	2109
Stabilität von Zugriffspfaden und Zugriffspfad-Repository (Access-Path-Repository)	2112
Komponenten des Zugriffspfad-Repositories (Access-Path-Repository)	2113
Static Plan Stability	2113
Static Plan Stability - Zugriffspfad-Stabilität: Bewahrung bzw. Kontrolle vorhandener Zugriffspfade	2114
Komponenten der Zugriffspfad-Stabilität	2114
Kontrolle der Zugriffspfad-Stabilität	2115
Was tun bei Änderung des Zugriffspfads	2116
Dynamic Plan Stability - Zugriffspfad-Stabilität: Stabilisierung bestimmter dynamischer SQL-Statements und ihrer Zugriffspfade	2117
Dynamic-Plan-Stability-Komponenten	2118
Zusammenfassung der relevanten Performance-Komponenten	2120
DB2-interne Automatismen	2120
Zugriffspfad-Effizienz	2120
Mögliche Zugriffspfade	2123
Systemgenerierungs- und Installations-Optionen	2124
Objekt-Definitionen	2126
Anwendungs-Design und -Programmierung	2128
SQL-Query-Empfehlungen	2129
Was tun bei langlaufenden Statements?	2130
Zusammenfassende Empfehlungen für Anwendungs-Entwickler	2131
Testdaten für die Performance-Tests	2132
Physisches Daten-Modell für Performance-Tests	2132
Metadaten-Übersicht der Test-Objekte	2132
DDL für die Performance-Tests	2135

Beispiel-SQL-Statements zur Einfügung der Daten	2138
Beispiel-Ausdrucke der Tabellen	2139
Beispiel-SQL-Statement zur Analyse der PLAN_TABLE für die Performance-Tests	2141