

Jo

Gunter Saake • Kai-Uwe Sattler

Datenbanken & Java

JDBC, SQLJ und ODMG

Technische Universität Darmstadt
FACHBEREICH INFORMATIK
BIBLIOTHEK

Inventurnr.: M07-00270

Sachgebiete: _____

Signatur: H.2 / Saak



Inhaltsverzeichnis

1	Einführung	1
1.1	Inhalt des Buches	2
1.2	Architekturen für Datenbankanwendungen	4
1.3	Beispielanwendung	7
1.4	Technische Hinweise	9
2	Java im Überblick	11
2.1	Einführung	11
2.2	Datentypen	13
2.3	Kontrollstrukturen und Operatoren	13
2.4	Klassen, Objekte und Methoden	14
2.5	Pakete	18
2.6	Ausnahmen	19
2.7	Übersetzung und Ausführung	20
2.8	Dateiarbeit mit Java	22
2.8.1	Ströme für Datenquellen und -senken	24
2.8.2	Filterströme	25
2.8.3	Datenströme	25
2.8.4	Objektströme	27
3	Grundlagen von relationalen Datenbanken	31
3.1	Einführung	32
3.1.1	Datenunabhängigkeit	33
3.1.2	Transaktionen	34
3.2	Das Relationenmodell	37
3.3	Relationaler Datenbankentwurf und Normalisierung	39
3.4	SQL – ein Überblick	41
3.4.1	Tabellendeklaration in SQL	41
3.4.2	Anfragen in SQL	43
3.4.3	Änderungsoperationen in SQL	45
3.4.4	Weitere Aspekte von SQL	46
3.4.5	Definition von Sichten in SQL	46
3.5	Anwendungsentwicklung für RDBMS	50

4	JDBC – das Datenbank-API für Java	55
4.1	Überblick	55
4.2	JDBC am Beispiel	59
4.3	JDBC im Detail	64
4.3.1	Treiber und Datenbankverbindung	64
4.3.2	Ausführung von SQL-Anfragen	73
4.3.3	ResultSet: Ergebnisse von Anfragen	83
4.3.4	Metadaten	86
4.3.5	Fehlerbehandlung	90
4.3.6	Abbildung von SQL-Typen in Java	92
4.4	JDBC 2.1	94
4.4.1	ResultSet-Erweiterungen	95
4.4.2	Batch-Updates	104
4.4.3	SQL99-Datentypen	106
4.4.4	Nutzung von JNDI	119
4.4.5	Connection Pooling	121
4.4.6	Verteilte Transaktionen	123
4.4.7	RowSets	124
4.4.8	Abschließende Bemerkungen	128
4.5	Beispielanwendung mit JDBC	128
5	SQLJ – die Verbindung von Java und SQL	137
5.1	Embedded SQL für Java	137
5.1.1	Überblick	137
5.1.2	Embedded SQL am Beispiel	139
5.1.3	SQLJ-Klauseln und Host-Variablen	143
5.1.4	Iteratoren	148
5.1.5	Kontexte	153
5.1.6	Der SQLJ-Translator und Profile	157
5.1.7	Interoperabilität mit JDBC	162
5.2	Gespeicherte Prozeduren	165
5.2.1	Überblick	166
5.2.2	Implementierung von gespeicherten Prozeduren	169
5.2.3	Installation und Registrierung	175
5.2.4	Aufruf und Anwendung	181
5.3	Java-Klassen als SQL-Datentypen	181
5.3.1	Implementierung von SQL-Typen	182
5.3.2	Installation und Registrierung	183
6	Java und Objektdatebanken	187
6.1	Einführung	187
6.2	Das Objektmodell der ODMG	192
6.3	Das Java-Binding der ODMG	200
6.3.1	Prinzip	201

6.3.2	Java-ODL	202
6.3.3	Java-OML	208
6.4	Anfragen in OQL	220
6.5	Beispielanwendung mit der ODMG-Anbindung	226
6.6	ODMG 3.0	233
7	Von Objekten zu Relationen	237
7.1	Grundlagen der objektrelationalen Abbildung	237
7.2	Aspekte der Laufzeitunterstützung	244
7.3	Java Blend	248
7.3.1	Überblick	248
7.3.2	Mapping-Werkzeug	249
7.3.3	Java-Blend-API	253
7.3.4	Anfragen	259
7.4	Zusammenfassung	264
8	Java und Transaktionen	265
8.1	Einführung in Transaktionsbegriffe	265
8.1.1	Der Scheduler	265
8.1.2	Synchronisation und Serialisierbarkeit	267
8.1.3	Operationen für die Transaktionssteuerung	268
8.1.4	Transaktionsmonitore	269
8.2	Verteilte Transaktionen und Transaktionsmanager	270
8.3	Java Transaction API	273
8.3.1	Schnittstelle JTA	274
8.3.2	Beispielprogramm	276
9	Orthogonal persistentes Java — eine Alternative? ...	279
9.1	Persistente Programmiersprachen	279
9.2	PJama – ein orthogonal persistentes Java	281
9.3	PJama: Programmierschnittstellen	282
9.4	PJama: Ein Beispiel	284
9.5	Anwendungsentwicklung mit PJama	287
10	Ausblick	289
A	Schemadefinition für die Beispielanwendung	293
A.1	SQL-Skript zur Definition des Beispielschemas	293
A.2	Relationale Repräsentation der Beispieldaten	295
A.3	Java-Klassen für die ODMG-Datenbank	297
	Literaturverzeichnis	307
	Index	311