



DATABASE MANAGEMENT SYSTEM

An Evolutionary Approach

Jagdish Chandra Patni
Hitesh Kumar Sharma
Ravi Tomar
Avita Katal



CRC Press
Taylor & Francis Group

A CHAPMAN & HALL BOOK

สำนักหอสมุด มหาวิทยาลัยเชียงใหม่

b16622480
012554285
122608825

Database Management System

An Evolutionary Approach



Jagdish Chandra Patni

Hitesh Kumar Sharma

Ravi Tomar

Avita Katal



CRC Press

Taylor & Francis Group

Boca Raton London New York

CRC Press is an imprint of the
Taylor & Francis Group, an Informa business
A CHAPMAN & HALL BOOK

Contents

Preface, xvii

Authors, xix

| | |
|---|----|
| CHAPTER 1 ■ Database Basics | 1 |
| 1.1 INTRODUCTION AND HISTORY OF DATABASES | 1 |
| 1.2 DATA AND INFORMATION | 4 |
| 1.2.1 Data | 4 |
| 1.2.2 Information | 5 |
| 1.3 DATABASE | 6 |
| 1.4 NEED FOR A DATABASE | 7 |
| 1.5 FILE-BASED DATA MANAGEMENT SYSTEM | 7 |
| 1.6 DATABASE SYSTEM | 9 |
| 1.6.1 Database System Components | 9 |
| 1.6.2 DBMS Services | 12 |
| 1.7 DATABASE CHARACTERISTICS | 14 |
| 1.8 ADVANTAGES OF A DATABASE | 16 |
| 1.8.1 Reducing Redundancy | 16 |
| 1.8.2 Preventing Unauthorized Access | 16 |
| 1.8.3 Faster Data Access | 17 |
| 1.8.4 Backup and Recovery | 17 |
| 1.8.5 Multiple User Interfaces | 17 |
| 1.8.6 Enforcing Integrity Constraints and Support Inferencing/Actions | 18 |
| 1.8.7 Persistent Storage | 18 |

| | | |
|-----------|--|----|
| 1.8.8 | Additional Benefits | 18 |
| 1.9 | LIMITATIONS OF DATABASES | 19 |
| 1.9.1 | Increased Cost | 19 |
| 1.9.2 | Complexity | 19 |
| 1.9.3 | Performance | 19 |
| 1.10 | SUMMARY | 21 |
| 1.11 | REVIEW QUESTIONS | 21 |
| | REFERENCES | 22 |
| <hr/> | | |
| CHAPTER 2 | ■ Data Models and Architecture of a DBMS | 23 |
| 2.1 | EVOLUTION OF DATA MODEL | 23 |
| 2.2 | HIERARCHICAL DATABASE MODEL | 25 |
| 2.2.1 | Advantages | 26 |
| 2.2.2 | Disadvantages | 27 |
| 2.3 | NETWORK DATA MODEL | 28 |
| 2.3.1 | Advantages | 28 |
| 2.3.2 | Disadvantages | 29 |
| 2.4 | RELATIONAL DATABASE MODEL | 30 |
| 2.4.1 | Concepts of RDBMS | 30 |
| 2.4.2 | Advantages | 32 |
| 2.4.3 | Disadvantages | 32 |
| 2.5 | OBJECT-ORIENTED DATA MODEL | 33 |
| 2.5.1 | Advantages | 34 |
| 2.5.2 | Disadvantages | 35 |
| 2.6 | OBJECT RELATIONAL DATA MODEL | 35 |
| 2.6.1 | Advantages | 36 |
| 2.6.2 | Disadvantages | 37 |
| 2.7 | THREE-LEVEL ARCHITECTURE OF A DATABASE | 38 |
| 2.7.1 | Data, Relationships, Constraints, and Schema | 38 |
| 2.7.2 | Internal Level | 39 |
| 2.7.3 | Conceptual Level | 39 |
| 2.7.4 | External Level | 40 |

| | | |
|--|---|-----------|
| 2.7.5 | Advantages | 41 |
| 2.8 | DATA INDEPENDENCE | 41 |
| 2.8.1 | Importance of Database Independence | 42 |
| 2.9 | DATABASE LANGUAGES | 43 |
| 2.10 | SUMMARY | 45 |
| 2.11 | REVIEW QUESTIONS | 45 |
| | REFERENCES | 46 |
| CHAPTER 3 ■ Relational Database Management System | | 47 |
| 3.1 | DIFFERENCE BETWEEN RDBMS AND DBMS | 47 |
| 3.2 | FEATURES OF RDBMS | 48 |
| 3.3 | ADVANTAGES OF RDBMS | 48 |
| 3.4 | DISADVANTAGES OF RDBMS | 49 |
| 3.5 | EXAMPLES OF RDBMS | 49 |
| 3.5.1 | MySQL | 49 |
| 3.5.1.1 | <i>Limitations for SQL Database</i> | 50 |
| 3.5.2 | NoSQL | 50 |
| 3.5.2.1 | <i>Benefits and Scope</i> | 50 |
| 3.5.2.2 | <i>Disadvantage and Limitations</i> | 51 |
| 3.5.3 | PostgreSQL | 51 |
| 3.5.4 | Microsoft SQL Server | 51 |
| 3.6 | RDBMS TERMINOLOGIES | 51 |
| 3.7 | KEYS IN DATABASE | 52 |
| 3.7.1 | Super Keys | 52 |
| 3.7.2 | Candidate Keys | 52 |
| 3.7.3 | Primary Keys | 53 |
| 3.7.4 | Composite Keys | 53 |
| 3.7.5 | Secondary or Alternative Key | 53 |
| 3.7.6 | Foreign Keys | 53 |
| 3.7.7 | Difference between Primary Key, Candidate Key, Super Key, and Foreign Key | 54 |
| 3.8 | INTEGRITY CONSTRAINTS IN DBMS | 55 |

| | | |
|----------|--|----|
| 3.8.1 | Integrity Constraints | 55 |
| 3.8.1.1 | <i>Entity Integrity Constraints</i> | 55 |
| 3.8.1.2 | <i>Referential Integrity Constraints</i> | 55 |
| 3.8.1.3 | <i>Domain Constraints</i> | 57 |
| 3.8.1.4 | <i>Not Null Constraint</i> | 57 |
| 3.8.1.5 | <i>Default Value Constraint</i> | 58 |
| 3.8.1.6 | <i>Check Clause</i> | 58 |
| 3.9 | RELATIONAL ALGEBRA | 59 |
| 3.9.1 | Unary Operators | 59 |
| 3.9.2 | Binary Operators | 59 |
| 3.9.2.1 | <i>SELECT Operator</i> | 59 |
| 3.9.2.2 | <i>PROJECT Operator</i> | 60 |
| 3.9.2.3 | <i>Union Operation</i> | 60 |
| 3.9.2.4 | <i>Set Difference</i> | 62 |
| 3.9.2.5 | <i>Cartesian Product(X)</i> | 62 |
| 3.9.2.6 | <i>Join Operations</i> | 63 |
| 3.9.3 | Left Outer Join | 65 |
| 3.9.4 | Right Outer Join | 65 |
| 3.9.5 | Full Outer Join | 65 |
| 3.10 | STRUCTURED QUERY LANGUAGE (SQL) | 66 |
| 3.10.1 | SELECT Statements | 66 |
| 3.11 | CODD'S 12 RULES OF RELATIONAL DATABASE | 66 |
| 3.12 | DATABASE DEVELOPMENT LIFE CYCLE | 68 |
| 3.12.1 | The Database Initial Study | 68 |
| 3.12.1.1 | <i>Analyze the Scenario</i> | 69 |
| 3.12.1.2 | <i>Define the Problems and Other Constraints</i> | 69 |
| 3.12.1.3 | <i>Objectives Scope and Boundaries</i> | 69 |
| 3.12.2 | Database Design | 69 |
| 3.12.3 | Implementation of Databases | 70 |
| 3.12.3.1 | <i>Install the DBMS</i> | 70 |
| 3.12.3.2 | <i>Database Creation</i> | 70 |

| | | |
|---------------------------------------|------------------------------------|----|
| 3.12.3.3 | <i>Populate the Data</i> | 70 |
| 3.12.4 | Testing and Evaluation | 70 |
| 3.12.4.1 | <i>Database Test</i> | 71 |
| 3.12.4.2 | <i>Tuning of Database</i> | 71 |
| 3.12.4.3 | <i>Database Evaluation</i> | 71 |
| 3.12.5 | Operation | 71 |
| 3.12.6 | Maintenance of Database System | 71 |
| 3.13 | SUMMARY | 72 |
| 3.14 | REVIEW QUESTIONS | 76 |
| | REFERENCES | 77 |
| CHAPTER 4 ■ Entity-Relationship Model | | 79 |
| 4.1 | ER DIAGRAM | 79 |
| 4.2 | COMPONENTS OF AN ER DIAGRAM | 80 |
| 4.2.1 | Entity Set | 80 |
| 4.2.1.1 | <i>Weak Entity Set</i> | 81 |
| 4.2.1.2 | <i>Strong Entity Set</i> | 81 |
| 4.2.2 | Attributes | 81 |
| 4.2.2.1 | <i>Key Attribute</i> | 81 |
| 4.2.2.2 | <i>Composite Attribute</i> | 82 |
| 4.2.2.3 | <i>Multivalued Attribute</i> | 82 |
| 4.2.2.4 | <i>Derived Attribute</i> | 83 |
| 4.2.3 | Relationship | 83 |
| 4.2.3.1 | <i>One-to-One Relationship</i> | 83 |
| 4.2.3.2 | <i>One-to-Many Relationship</i> | 83 |
| 4.2.3.3 | <i>Many-to-One Relationship</i> | 83 |
| 4.2.3.4 | <i>Many-to-Many Relationship</i> | 84 |
| 4.3 | PARTICIPATION CONSTRAINTS | 84 |
| 4.4 | STRONG AND WEAK RELATIONSHIP | 85 |
| 4.4.1 | Strong Entity Set | 85 |
| 4.4.2 | Weak Entity | 85 |
| 4.5 | HANDLING MANY-TO-MANY RELATIONSHIP | 86 |

| | | |
|---------|--|-----|
| 4.5.1 | Loss of Analytical Capability | 87 |
| 4.5.2 | Multiple Counting | 88 |
| 4.5.2.1 | <i>Method 1</i> | 89 |
| 4.5.2.2 | <i>Method 2</i> | 90 |
| 4.5.2.3 | <i>Method 3</i> | 90 |
| 4.6 | EXAMPLE OF ER MODEL | 91 |
| 4.6.1 | An ER Diagram of an Organization | 91 |
| 4.6.1.1 | <i>Initial Conceptual Design of the Company Database</i> | 91 |
| 4.6.1.2 | <i>Relations and Its Structure</i> | 91 |
| 4.6.1.3 | <i>Tables for Company Database</i> | 92 |
| 4.6.1.4 | <i>Relationship Types to Database Schema</i> | 92 |
| 4.7 | ENHANCED ER MODEL | 93 |
| 4.7.1 | Hierarchies | 93 |
| 4.7.2 | Properties of IsA Relationship Vehicle | 94 |
| 4.7.3 | Constraints in ER/EER Diagram | 95 |
| 4.7.4 | Specialization Abstraction | 95 |
| 4.7.5 | Generalization Abstraction | 96 |
| 4.7.6 | Union or Category | 96 |
| 4.8 | REDUCTION OF ER DIAGRAM TO RELATIONAL MODEL | 97 |
| 4.8.1 | Regular Table | 98 |
| 4.8.2 | Attributes for the Table | 98 |
| 4.8.3 | Key Attribute (Primary Key) | 98 |
| 4.8.4 | Separate Table for Multivalued Attribute | 99 |
| 4.8.5 | Derived Attributes | 99 |
| 4.8.6 | Mapping of Relationships in the Tables | 99 |
| 4.8.7 | Tables | 101 |
| 4.9 | SUMMARY | 102 |
| 4.10 | REVIEW QUESTIONS | 103 |
| | REFERENCES | 105 |

| | |
|--|-----|
| CHAPTER 5 ■ Normalization | 107 |
| <hr/> | |
| 5.1 INTRODUCTION TO NORMALIZATION – A BOTTOM-UP APPROACH | 107 |
| 5.2 NEED FOR NORMALIZATION | 107 |
| 5.3 TYPES OF DEPENDENCIES | 108 |
| 5.3.1 Trivial FD | 108 |
| 5.3.2 Nontrivial FD | 109 |
| 5.3.3 Armstrong's Axioms or Inference Rule of Functional Dependencies | 109 |
| 5.4 FIRST NORMAL FORM | 110 |
| 5.5 SECOND NORMAL FORM | 110 |
| 5.6 THIRD NORMAL FORM | 111 |
| 5.7 BOYCE-CODD NORMAL FORM | 112 |
| 5.8 MULTI-VALUED DEPENDENCY | 114 |
| 5.9 JOIN DEPENDENCY | 115 |
| 5.10 LOSSLESS AND LOSSY DECOMPOSITIONS | 115 |
| 5.11 NORMALIZING TABLES AND UNSOLVED EXAMPLES WITH CASE STUDIES | 116 |
| 5.12 SUMMARY | 116 |
| 5.13 REVIEW QUESTIONS | 117 |
| REFERENCES | 117 |
| <hr/> | |
| CHAPTER 6 ■ Managing Data Using Structured Query Language | 119 |
| <hr/> | |
| 6.1 INTRODUCTION TO STRUCTURED QUERY LANGUAGE | 119 |
| 6.1.1 Advantages of SQL | 119 |
| 6.1.2 Types of SQL Commands | 120 |
| 6.2 DATA DEFINITION COMMANDS | 120 |
| 6.3 DML | 122 |
| 6.4 DATA CONTROL LANGUAGE | 126 |
| 6.4.1 GRANT Command | 126 |

| | | |
|---|--|------------|
| 6.4.2 | REVOKE Command | 126 |
| 6.5 | TRANSACTION CONTROL LANGUAGE (TCL) | 127 |
| 6.5.1 | COMMIT Command | 127 |
| 6.5.2 | ROLLBACK Command | 128 |
| 6.6 | DATA QUERY LANGUAGE (DQL) | 128 |
| 6.7 | AGGREGATE FUNCTIONS | 129 |
| 6.8 | DATE AND TIME FUNCTIONS | 133 |
| 6.9 | STRING FUNCTIONS | 135 |
| 6.10 | CONVERSION FUNCTIONS | 137 |
| 6.11 | MATHEMATICAL FUNCTIONS | 139 |
| 6.12 | SPECIAL OPERATORS | 141 |
| 6.13 | TYPES OF CONSTRAINTS | 143 |
| 6.14 | SUBQUERY | 144 |
| 6.15 | SUMMARY | 146 |
| 6.16 | REVIEW QUESTIONS | 146 |
| | REFERENCES | 147 |
| CHAPTER 7 ■ Introduction to PL/SQL | | 149 |
| 7.1 | PL/SQL | 149 |
| 7.2 | VARIABLES AND CONSTANTS | 150 |
| 7.2.1 | Constant | 151 |
| 7.3 | DATA TYPES | 151 |
| 7.3.1 | PL/SQL Number Types | 151 |
| 7.3.2 | PL/SQL Character and String Types and PL/SQL National Character Types | 151 |
| 7.3.3 | “LONG and LONG RAW Datatypes” | 152 |
| 7.3.4 | PL/SQL Date, Time, and Interval Types | 152 |
| 7.3.5 | NULLs in PL/SQL | 153 |
| 7.4 | PL/SQL LITERALS | 153 |
| 7.4.1 | Example of These Different Types of Literals | 153 |
| 7.5 | CONTROL STATEMENTS | 153 |
| 7.5.1 | If Statement | 154 |

| | | |
|----------|--|-----|
| 7.5.1.1 | <i>IF-THEN Statement</i> | 154 |
| 7.5.1.2 | <i>IF-THEN-ELSE Statement</i> | 154 |
| 7.5.2 | Example of PL/SQL If Statement | 155 |
| 7.6 | PL/SQL CASE STATEMENT | 156 |
| 7.6.1 | Example of PL/SQL Case Statement | 157 |
| 7.7 | PL/SQL LOOP | 157 |
| 7.7.1 | Example of PL/SQL EXIT Loop with WHEN Condition | 158 |
| 7.8 | PL/SQL EXIT LOOP EXAMPLE WITH IF CONDITION | 159 |
| 7.8.1 | Example of PL/SQL While Loop | 160 |
| 7.8.2 | PL/SQL WHILE Loop Example 2 | 161 |
| 7.8.3 | PL/SQL for Loop Example 1 | 162 |
| 7.8.4 | PL/SQL for Loop REVERSE Example | 163 |
| 7.9 | PL/SQL CONTINUE STATEMENT | 164 |
| 7.9.1 | Example of PL/SQL Continue Statement | 164 |
| 7.10 | PL/SQL GOTO STATEMENT | 165 |
| 7.10.1 | Example of PL/SQL GOTO Statement | 165 |
| 7.11 | PL/SQL PROCEDURE AND FUNCTIONS | 166 |
| 7.11.1 | PL/SQL Procedure Syntax and Examples | 167 |
| 7.11.1.1 | <i>PL/SQL Procedure Example</i> | 167 |
| 7.11.2 | PL/SQL Function Syntax and Examples | 169 |
| 7.11.2.1 | <i>PL/SQL Function Example</i> | 169 |
| 7.11.2.2 | <i>PL/SQL Function Example 3</i> | 171 |
| 7.11.2.3 | <i>PL/SQL Function Example 4</i> | 173 |
| 7.12 | PL/SQL CURSOR | 174 |
| 7.12.1 | PL/SQL Implicit Cursors | 174 |
| 7.12.2 | PL/SQL Implicit Cursor Example | 175 |
| 7.12.3 | PL/SQL Explicit Cursors | 175 |
| 7.12.3.1 | <i>PL/SQL Explicit Cursor Example</i> | 176 |
| 7.13 | PL/SQL EXCEPTION | 177 |
| 7.14 | PL/SQL EXCEPTION HANDLING | 178 |

| | | |
|---|-----------------------------------|------------|
| 7.14.1 | System-Defined Exception Handling | 179 |
| 7.14.2 | System-Defined Exceptions | 180 |
| 7.14.3 | User-Defined Exceptions | 182 |
| 7.14.4 | Raising Exceptions | 182 |
| 7.15 | PL/SQL TRIGGER | 182 |
| 7.15.1 | Trigger Example | 184 |
| 7.16 | SUMMARY | 187 |
| 7.17 | REVIEW QUESTIONS | 188 |
| CHAPTER 8 ■ Transaction Management in a Database | | 191 |
| 8.1 | INTRODUCTION | 191 |
| 8.2 | DEFINITION OF TRANSACTION | 191 |
| 8.2.1 | Commit | 193 |
| 8.2.2 | Rollback | 193 |
| 8.3 | PROPERTIES OF TRANSACTION | 194 |
| 8.3.1 | Atomicity | 194 |
| 8.3.2 | Consistency | 194 |
| 8.3.3 | Isolation | 194 |
| 8.3.4 | Durability | 195 |
| 8.4 | STATES OF TRANSACTION | 195 |
| 8.4.1 | Active State | 195 |
| 8.4.2 | Partially Committed State | 196 |
| 8.4.3 | Committed State | 196 |
| 8.4.4 | Failed State | 196 |
| 8.4.5 | Aborted State | 196 |
| 8.4.6 | Terminated State | 196 |
| 8.5 | SCHEDULE | 197 |
| 8.5.1 | Types of Schedules | 197 |
| 8.5.1.1 | <i>Serial Schedule</i> | 197 |
| 8.5.1.2 | <i>Concurrent Schedule</i> | 198 |
| 8.5.1.3 | <i>Cascadeless Schedule</i> | 198 |
| 8.5.1.4 | <i>Recoverable Schedule</i> | 199 |

| | | |
|----------|--|-----|
| 8.6 | SERIALIZABILITY | 199 |
| 8.6.1 | Conflict-Serializable Schedule | 200 |
| 8.6.2 | Precedence Graph Method of Serializability | 201 |
| 8.6.3 | View-Serializable Schedule | 201 |
| 8.7 | RECOVERABLE SCHEDULE | 203 |
| 8.8 | CONCURRENCY CONTROL | 203 |
| 8.8.1 | Lost Update Problem | 203 |
| 8.8.2 | Dirty Read Problem | 204 |
| 8.8.3 | Inconsistent Retrieval or Unrepeatable Read Problem | 204 |
| 8.9 | CONCURRENCY CONTROL MECHANISM | 205 |
| 8.9.1 | Lock-Based Protocol | 205 |
| 8.9.1.1 | <i>Lock-Based Protocols</i> | 205 |
| 8.9.1.2 | <i>Deadlock Problem with Lock-Based Protocol</i> | 206 |
| 8.9.1.3 | <i>Two-Phase Locking Protocol</i> | 207 |
| 8.9.2 | Timestamp-Based Protocol | 208 |
| 8.9.2.1 | <i>Timestamp Ordering Protocol</i> | 208 |
| 8.9.2.2 | <i>Basic Timestamp Ordering</i> | 209 |
| 8.9.3 | Multiple Granularity-Based Protocols | 210 |
| 8.9.4 | Intention Lock Mode | 212 |
| 8.9.5 | Multiversion Schemes | 214 |
| 8.9.5.1 | <i>Multiversion Concurrency Control Techniques</i> | 214 |
| 8.9.5.2 | <i>Multiversion Techniques Based on Timestamp Ordering</i> | 215 |
| 8.10 | DATABASE BACKUP AND RECOVERY | 215 |
| 8.10.1 | Log-Based Recovery | 217 |
| 8.10.2 | Recovery with Concurrent Transactions | 218 |
| 8.10.2.1 | <i>Checkpoint</i> | 218 |
| 8.10.2.2 | <i>Recovery</i> | 218 |
| 8.11 | SECURITY, INTEGRITY, AND AUTHORIZATION | 219 |

| | |
|-----------------------|-----|
| 8.12 SUMMARY | 220 |
| 8.13 REVIEW QUESTIONS | 221 |
| REFERENCES | 223 |
| INDEX, 225 | |