

 Cengage

Carlos Coronel • Steven Morris

Database Systems

Design, Implementation, & Management

14TH Edition

Carlos Coronel • Steven Morris

Database Systems

● Design, Implementation, & Management

14TH Edition



Australia • Brazil • Canada • Mexico • Singapore • United Kingdom • United States

This is an electronic version of the print textbook. Due to electronic rights restrictions, some third party content may be suppressed. Editorial review has deemed that any suppressed content does not materially affect the overall learning experience. The publisher reserves the right to remove content from this title at any time if subsequent rights restrictions require it. For valuable information on pricing, previous editions, changes to current editions, and alternate formats, please visit www.cengage.com/highered to search by ISBN#, author, title, or keyword for materials in your areas of interest.

Important Notice: Media content referenced within the product description or the product text may not be available in the eBook version.

**Database Systems: Design, Implementation,
and Management, 14th Edition**
Carlos Coronel and Steven Morris

SVP, Higher Education Product Management: Erin Joyner

VP, Product Management, Learning Experiences:
Thais Alencar

Product Director: Mark Santee

Product Manager: Natalie Onderdonk

Product Assistant: Ethan Wheel

Learning Designers: Mary Clyne and Carolyn Mako

Sr. Content Managers: Tim Bailey and Kara DiCaterino

Digital Project Manager: Jim Vaughey

Technical Editor: Danielle Shaw

Developmental Editor: Lisa Ruffolo

VP, Product Marketing: Jason Sakos

Director, Product Marketing: Danaë April

Portfolio Marketing Manager: Mackenzie Paine

IP Analyst: Ann Hoffman

IP Project Manager: Anjali Kambli, Lumina Datamatics

IP Photo/Text Researcher: Manoranjini Boomiappan, Lumina Datamatics

Production Service: Straive

Sr. Designer: Erin Griffin

Cover Image Source:
ConnectVector/Shutterstock.com

© 2023, 2019, 2015 Cengage Learning, Inc. ALL RIGHTS RESERVED.

WCN: 02-300

No part of this work covered by the copyright herein may be reproduced or distributed in any form or by any means, except as permitted by U.S. copyright law, without the prior written permission of the copyright owner.

Unless otherwise noted, all content is Copyright © Cengage Learning, Inc.

All screenshots, unless otherwise noted, are used with permission from Microsoft Corporation. Microsoft® is a registered trademark of the Microsoft Corporation.

For product information and technology assistance, contact us at
Cengage Customer & Sales Support, 1-800-354-9706
or **support.cengage.com**.

For permission to use material from this text or product, submit all requests online at **www.copyright.com**.

Library of Congress Control Screenshots for this book were created using Microsoft Access® and Excel® and were used with permission from Microsoft. Microsoft and the Office logo are either registered trademarks or trademarks of Microsoft Corporation in the United States and/or other countries.

Oracle is a registered trademark, and Oracle12c and MySQL are trademarks of Oracle Corporation.

Library of Congress Control Number: 2022912918

ISBN: 978-0-357-67303-4

Loose Leaf Edition ISBN: 978-0-357-67307-2

Cengage

200 Pier 4 Boulevard
Boston, MA 02210
USA

Cengage is a leading provider of customized learning solutions with employees residing in nearly 40 different countries and sales in more than 125 countries around the world. Find your local representative at **www.cengage.com**.

To learn more about Cengage platforms and services, register or access your online learning solution, or purchase materials for your course, visit **www.cengage.com**.

Notice to the Reader

Publisher does not warrant or guarantee any of the products described herein or perform any independent analysis in connection with any of the product information contained herein. Publisher does not assume, and expressly disclaims, any obligation to obtain and include information other than that provided to it by the manufacturer. The reader is expressly warned to consider and adopt all safety precautions that might be indicated by the activities described herein and to avoid all potential hazards. By following the instructions contained herein, the reader willingly assumes all risks in connection with such instructions. The publisher makes no representations or warranties of any kind, including but not limited to, the warranties of fitness for particular purpose or merchantability, nor are any such representations implied with respect to the material set forth herein, and the publisher takes no responsibility with respect to such material. The publisher shall not be liable for any special, consequential, or exemplary damages resulting, in whole or part, from the readers' use of, or reliance upon, this material.

Printed in the United States of America

Print Number: 01

Print Year: 2022

Dedication

To the treasures in my life: To Victoria, for 31 wonderful years. Thank you for your unending support and for being my angel, my sweetie, and, most importantly, my best friend. To Carlos Anthony, who has become a remarkable man, pride of his father, and husband to our beautiful, sweet, and brilliant daughter-in-law, Jered. Thank you for your words of wisdom, hard-working attitude, and for giving us reasons to be happy. You are still young; your best times are still to come. To Cerila Reyan, our beautiful granddaughter, whose smiles give us so much hope and happiness. To Gabriela Victoria, who is the image of brilliance, beauty, and faithfulness. The way you give your time and talents in the service of others is an inspiration to all of us. Thank you for being my sunshine on cloudy days. Your future is bright and endless. To Christian Javier, who is smarter than all of us. Thank you for being the youthful reminder of life's simple beauties. Keep challenging yourself to new highs and keep working hard to achieve your dreams. To my parents, Sarah and Carlos, thank you for your sacrifice and example. To all of you, you are all my inspiration. "TQTATA."

Carlos Coronel

To Pamela, from high school sweetheart through nearly 30 years of marriage, you are the beautiful love of my life who has supported, encouraged, and inspired me. More than anyone else, you are responsible for whatever successes I have achieved. To my son, Alexander, your depth of character is without measure. You are my pride and joy. To my daughter, Lauren, your beauty and intensity take my breath away. You are my heart and soul. To my daughter-in-law, Blakley, whom I could not love more if you were my flesh and blood. To my granddaughter, Daphne, a bundle of joy brought into a world that does not deserve someone so precious. To my mother, Florence, and to the memory of my father, Alton, together they instilled in me the desire to learn and the passion to achieve. To my mother-in-law, Connie, and to the memory of my father-in-law, Wayne, they taught me to find joy in all things. To all of you, with all my love, I dedicate this book.

Steven Morris

For Peter

To longtime colleague and friend, Peter Rob: Your drive and dedication to your students started this book. Your depth of knowledge, attention to detail, and pursuit of excellence made it succeed. Your patience and guidance continue to light our path. It is our sincere hope that, as we move forward, we can continue to live up to your standard. Enjoy your retirement, my friend; you have surely earned it.

Carlos Coronel and Steven Morris

Brief Contents

Preface, xiv
Text Features, xx
Additional Features, xxii
Acknowledgments, xxiv

Part 1

Database Concepts 2

Chapter 1 Database Systems, 3
Chapter 2 Data Models, 33

Part 2

Design Concepts 64

Chapter 3 The Relational Database Model, 65
Chapter 4 Entity Relationship (ER) Modeling, 108
Chapter 5 Advanced Data Modeling, 162
Chapter 6 Normalization of Database Tables, 192

Part 3

Advanced Design and Implementation 238

Chapter 7 Introduction to Structured Query Language (SQL), 239
Chapter 8 Advanced SQL, 351
Chapter 9 Database Design, 431

Part 4

Advanced Database Concepts 472

Chapter 10 Transaction Management and Concurrency Control, 473
Chapter 11 Database Performance Tuning and Query Optimization, 506
Chapter 12 Distributed Database Management Systems, 539
Chapter 13 Business Intelligence and Data Warehouses, 573
Chapter 14 Big Data and NoSQL, 640

Part 5

Databases and the Internet 674

Chapter 15 Database Connectivity and Web Technologies, 675

Part 6

Database Administration 714

Chapter 16 Database Administration and Security, 715

Glossary, 760
Index, 775

The following appendices are included on the Instructor and Student Resource Sites at www.cengage.com.

- Appendix A:** Designing Databases with Lucidchart: A Tutorial
- Appendix B:** The University Lab: Conceptual Design
- Appendix C:** The University Lab: Conceptual Design Verification, Logical Design, and Implementation
- Appendix D:** Converting an ER Model into a Database Structure
- Appendix E:** Comparison of ER Modeling Notations
- Appendix F:** Client/Server Systems
- Appendix G:** Object-Oriented Databases
- Appendix H:** Unified Modeling Language (UML)
- Appendix I:** Databases in Electronic Commerce
- Appendix J:** Web Database Development with ColdFusion
- Appendix K:** The Hierarchical Database Model
- Appendix L:** The Network Database Model
- Appendix M:** MS Access Tutorial
- Appendix N:** Creating a New Database Using Oracle
- Appendix O:** Data Warehouse Implementation Factors
- Appendix P:** Working with MongoDB
- Appendix Q:** Working with Neo4j

Contents

Preface, xiv

Text Features, xx

Additional Features, xxii

Acknowledgments, xxiv

Part 1

Database Concepts

Chapter 1

Database Systems 3

1-1 Why Databases? 4

1-2 Data versus Information 5

1-3 Introducing the Database 8

1-3a Role and Advantages of the DBMS 8

1-3b Types of Databases 10

1-4 Why Database Design Is Important 12

1-5 Evolution of File System Data Processing 15

1-5a Manual File Systems 16

1-5b Computerized File Systems 16

1-5c File System Redux: Modern End-User Productivity Tools 18

1-6 Problems with File System Data Processing 18

1-6a Structural and Data Dependence 19

1-6b Data Redundancy 20

1-6c Data Anomalies 21

1-7 Database Systems 21

1-7a The Database System Environment 22

1-7b DBMS Functions 24

1-7c Managing the Database System: A Shift in Focus 27

1-8 Preparing for Your Database Professional Career 28

Summary 29

Key Terms 29

Review Questions 30

Problems 31

Chapter 2

Data Models 33

2-1 Data Modeling and Data Models 34

2-2 The Importance of Data Models 35

2-3 Data Model Basic Building Blocks 36

2-4 Business Rules 37

2-4a Discovering Business Rules 37

2-4b Translating Business Rules into Data Model Components 38

2-4c Naming Conventions 39

2-5 The Evolution of Data Models 39

2-5a Hierarchical and Network Models 39

2-5b The Relational Model 41

2-5c The Entity Relationship Model 43

2-5d The Object-Oriented Model 45

2-5e Object/Relational and XML 47

2-5f Emerging Data Models: Big Data and NoSQL 48

2-5g Data Models: A Summary 51

2-6 Degrees of Data Abstraction 52

2-6a The External Model 55

2-6b The Conceptual Model 56

2-6c The Internal Model 57

2-6d The Physical Model 58

Summary 59

Key Terms 60

Review Questions 60

Problems 61

Part 2

Design Concepts

Chapter 3

The Relational Database Model 65

3-1 A Logical View of Data 67

3-1a Tables and Their Characteristics 67

3-2 Keys 69

3-2a Dependencies 69

3-2b Types of Keys 70

3-3 Integrity Rules 73

3-4 Relational Algebra 75

3-4a Formal Definitions and Terminology 75

3-4b Relational Set Operators 76

3-5 The Data Dictionary and the System Catalog 84

3-6 Relationships within the Relational Database 86

3-6a The 1:M Relationship 86

3-6b The 1:1 Relationship 88

3-6c The M:N Relationship 90

3-7 Data Redundancy Revisited 94**3-8 Indexes 96****3-9 Codd's Relational Database Rules 98**

Summary 98

Key Terms 99

Review Questions 99

Problems 102

Chapter 4**Entity Relationship (ER) Modeling 108****4-1 The Entity Relationship Model 109**

4-1a Entities 110

4-1b Attributes 110

4-1c Relationships 116

4-1d Connectivity and Cardinality 116

4-1e Existence Dependence 119

4-1f Relationship Strength 119

4-1g Weak Entities 122

4-1h Relationship Participation 124

4-1i Relationship Degree 126

4-1j Recursive Relationships 128

4-1k Associative (Composite) Entities 132

4-2 Developing an ER Diagram 134**4-3 Database Design Challenges:
Conflicting Goals 142**

Summary 146

Key Terms 147

Review Questions 147

Problems 149

Cases 155

Chapter 5**Advanced Data Modeling 162****5-1 The Extended Entity Relationship Model 163**

5-1a Entity Supertypes and Subtypes 163

5-1b Specialization Hierarchy 164

5-1c Inheritance 165

5-1d Subtype Discriminator 167

5-1e Disjoint and Overlapping Constraints 167

5-1f Completeness Constraint 169

5-1g Specialization and Generalization 170

5-2 Entity Clustering 170**5-3 Entity Integrity: Selecting Primary Keys 171**

5-3a Natural Keys and Primary Keys 172

5-3b Primary Key Guidelines 172

5-3c When to Use Composite Primary Keys 172

5-3d When to Use Surrogate Primary Keys 174

5-4 Design Cases: Learning Flexible Database Design 1755-4a Design Case 1: Implementing 1:1
Relationships 1765-4b Design Case 2: Maintaining History of Time-Variant
Data 177

5-4c Design Case 3: Fan Traps 180

5-4d Design Case 4: Redundant Relationships 181

Summary 182

Key Terms 182

Review Questions 182

Problems 183

Cases 185

Chapter 6**Normalization of Database Tables 192****6-1 Database Tables and Normalization 193****6-2 The Need for Normalization 194****6-3 The Normalization Process 197**

6-3a Conversion to First Normal Form (1NF) 198

6-3b Conversion to Second Normal Form (2NF) 202

6-3c Conversion to Third Normal Form (3NF) 204

6-4 Improving the Design 206**6-5 Surrogate Key Considerations 210****6-6 Higher-Level Normal Forms 212**

6-6a The Boyce-Codd Normal Form 212

6-6b Fourth Normal Form (4NF) 215

6-6c Fifth Normal Form (5NF) 217

6-7 Normalization and Database Design 218**6-8 Denormalization 222****6-9 Data-Modeling Checklist 225**

Summary 227

Key Terms 227

Review Questions 228

Problems 229

Part 3

Advanced Design and Implementation

Chapter 7

Introduction to Structured Query Language (SQL) 239

7-1 SQL Basics 240

- 7-1a Data Types 241
- 7-1b SQL Queries 242
- 7-1c The Database Model 243

7-2 Basic SELECT Queries 244

7-3 SELECT Statement Options 245

- 7-3a Using Column Aliases 246
- 7-3b Using Computed Columns 247
- 7-3c Arithmetic Operators:
The Rule of Precedence 248
- 7-3d Date Arithmetic 249
- 7-3e Listing Unique Values 250

7-4 FROM Clause Options 251

7-5 ORDER BY Clause Options 252

7-6 WHERE Clause Options 254

- 7-6a Selecting Rows with Conditional Restrictions 255
- 7-6b Using Comparison Operators on Character Attributes 257
- 7-6c Using Comparison Operators on Dates 258
- 7-6d Logical Operators: AND, OR, and NOT 258
- 7-6e Special Operators 260

7-7 JOIN Operations 264

- 7-7a Natural Join 265
- 7-7b JOIN USING Syntax 267
- 7-7c JOIN ON Syntax 268
- 7-7d Common Attribute Names 269
- 7-7e Old-Style Joins 269
- 7-7f Outer Joins 271
- 7-7g Cross Join 274
- 7-7h Joining Tables with an Alias 275
- 7-7i Recursive Joins 275

7-8 Aggregate Processing 277

- 7-8a Aggregate Functions 277
- 7-8b Grouping Data 281
- 7-8c HAVING Clause 284

7-9 Subqueries 286

- 7-9a WHERE Subqueries 287
- 7-9b IN Subqueries 288
- 7-9c HAVING Subqueries 289

7-9d Multirow Subquery Operators: ALL and ANY 289

7-9e FROM Subqueries 290

7-9f Attribute List Subqueries 291

7-9g Correlated Subqueries 293

7-10 SQL Functions 296

- 7-10a Date and Time Functions 297
- 7-10b Numeric Functions 300
- 7-10c String Functions 300
- 7-10d Conversion Functions 302

7-11 Relational Set Operators 304

- 7-11a UNION 305
- 7-11b UNION ALL 306
- 7-11c INTERSECT 307
- 7-11d EXCEPT (MINUS) 308
- 7-11e Syntax Alternatives 309

7-12 Crafting SELECT Queries 310

- 7-12a Know Your Data 310
- 7-12b Know the Problem 310
- 7-12c Build One Clause at a Time 311

Summary 312

Key Terms 313

Review Questions 314

Problems 315

Chapter 8

Advanced SQL 351

8-1 Data Definition Commands 352

- 8-1a Starting Database Model 352
- 8-1b Creating the Database 354
- 8-1c The Database Schema 354
- 8-1d Data Types 355

8-2 Creating Table Structures 358

- 8-2a CREATE TABLE command 358
- 8-2b SQL Constraints 362
- 8-2c Creating a Table with a SELECT Statement 365
- 8-2d SQL Indexes 366

8-3 Altering Table Structures 367

- 8-3a Changing a Column's Data Type 368
- 8-3b Changing a Column's Data Characteristics 368
- 8-3c Adding a Column 368
- 8-3d Adding Primary Key, Foreign Key, and Check Constraints 369
- 8-3e Dropping a Column 369
- 8-3f Deleting a Table from the Database 370

8-4 Data Manipulation Commands 370

- 8-4a Adding Table Rows 370

- 8-4b Inserting Table Rows with a SELECT Subquery 372
- 8-4c Saving Table Changes 374
- 8-4d Updating Table Rows 374
- 8-4e Deleting Table Rows 377
- 8-4f Restoring Table Contents 378

8-5 Virtual Tables: Creating a View 378

- 8-5a Updatable Views 380

8-6 Auto Increment, Identity, and Sequences 381

8-7 Procedural SQL 387

- 8-7a Stored Procedures 389
- 8-7b Working with Variables 391
- 8-7c Conditional Execution 392
- 8-7d Iteration or Looping 393
- 8-7e SELECT Processing with Cursors 396
- 8-7f Stored Procedures with Parameters 399
- 8-7g Triggers 401
- 8-7h User Defined Functions 412

8-8 Embedded SQL 412

- Summary 417
- Key Terms 418
- Review Questions 418
- Problems 418
- Cases 425

Chapter 9

Database Design 431

9-1 The Information System 432

9-2 The Systems Development Life Cycle 434

- 9-2a Planning 434
- 9-2b Analysis 436
- 9-2c Detailed Systems Design 436
- 9-2d Implementation 437
- 9-2e Maintenance 437

9-3 The Database Life Cycle 437

- 9-3a The Database Initial Study 438
- 9-3b Database Design 442
- 9-3c Implementation and Loading 444
- 9-3d Testing and Evaluation 445
- 9-3e Operation 448
- 9-3f Maintenance and Evolution 448

9-4 Conceptual Design 448

- 9-4a Data Analysis and Requirements 450
- 9-4b Entity Relationship Modeling and Normalization 452
- 9-4c Data Model Verification 455
- 9-4d Distributed Database Design 458

9-5 DBMS Software Selection 458

9-6 Logical Design 459

- 9-6a Map the Conceptual Model to the Logical Model Components 459
- 9-6b Validate the Logical Model Using Normalization 461
- 9-6c Validate the Logical Model Integrity Constraints 461
- 9-6d Validate the Logical Model against User Requirements 462

9-7 Physical Design 462

- 9-7a Define Data Storage Organization 463
- 9-7b Define Integrity and Security Measures 463
- 9-7c Determine Performance Measurements 464

9-8 Database Design Strategies 464

9-9 Centralized versus Decentralized Design 465

- Summary 468
- Key Terms 468
- Review Questions 468
- Problems 469

Part 4

Advanced Database Concepts

Chapter 10

Transaction Management and Concurrency Control 473

10-1 What Is a Transaction? 474

- 10-1a Evaluating Transaction Results 476
- 10-1b Transaction Properties 478
- 10-1c Transaction Management with SQL 479
- 10-1d The Transaction Log 480

10-2 Concurrency Control 481

- 10-2a Lost Updates 481
- 10-2b Uncommitted Data 482
- 10-2c Inconsistent Retrievals 483
- 10-2d The Scheduler 484

10-3 Concurrency Control with Locking Methods 486

- 10-3a Lock Granularity 486
- 10-3b Lock Types 489
- 10-3c Two-Phase Locking to Ensure Serializability 490
- 10-3d Deadlocks 491

10-4 Concurrency Control with Time Stamping Methods 492

- 10-4a Wait/Die and Wound/Wait Schemes 493

10-5 Concurrency Control with Optimistic Methods 494

10-6 ANSI Levels of Transaction Isolation 494

10-7 Database Recovery Management 496

10-7a Transaction Recovery 497

Summary 501

Key Terms 501

Review Questions 502

Problems 502

Chapter 11

Database Performance Tuning and Query Optimization 506

11-1 Database Performance-Tuning Concepts 507

11-1a Performance Tuning: Client and Server 508

11-1b DBMS Architecture 509

11-1c Database Query Optimization Modes 511

11-1d Database Statistics 512

11-2 Query Processing 513

11-2a SQL Parsing Phase 514

11-2b SQL Execution Phase 515

11-2c SQL Fetching Phase 515

11-2d Query Processing Bottlenecks 515

11-3 Indexes and Query Optimization 516

11-4 Optimizer Choices 518

11-4a Using Hints to Affect Optimizer Choices 520

11-5 SQL Performance Tuning 521

11-5a Index Selectivity 521

11-5b Conditional Expressions 522

11-6 Query Formulation 524

11-7 DBMS Performance Tuning 525

11-8 Query Optimization Example 527

Summary 533

Key Terms 534

Review Questions 534

Problems 535

Chapter 12

Distributed Database Management Systems 539

12-1 The Evolution of Distributed Database Management Systems 540

12-2 DDBMS Advantages and Disadvantages 542

12-3 Distributed Processing and Distributed Databases 543

12-4 Characteristics of Distributed Database Management Systems 545

12-5 DDBMS Components 546

12-6 Levels of Data and Process Distribution 547

12-6a Single-Site Processing, Single-Site Data 547

12-6b Multiple-Site Processing, Single-Site Data 548

12-6c Multiple-Site Processing, Multiple-Site Data 549

12-7 Distributed Database Transparency Features 550

12-8 Distribution Transparency 551

12-9 Transaction Transparency 553

12-9a Distributed Requests and Distributed Transactions 553

12-9b Distributed Concurrency Control 556

12-9c Two-Phase Commit Protocol 557

12-10 Performance and Failure Transparency 558

12-11 Distributed Database Design 559

12-11a Data Fragmentation 559

12-11b Data Replication 563

12-11c Data Allocation 565

12-12 The CAP Theorem 565

12-13 C. J. Date's 12 Commandments for Distributed Databases 567

Summary 568

Key Terms 568

Review Questions 569

Problems 570

Chapter 13

Business Intelligence and Data Warehouses 573

13-1 The Need for Data Analysis 574

13-2 Business Intelligence 574

13-2a Business Intelligence Architecture 576

13-2b Business Intelligence Benefits 580

13-2c Business Intelligence Evolution 580

13-2d Business Intelligence Technology Trends 582

13-3 Decision Support Data 583

13-3a Operational Data versus Decision Support Data 583

13-3b Decision Support Database Requirements 586

13-4 The Data Warehouse 588

13-4a Data Marts 590

13-4b Twelve Rules That Define a Data Warehouse 591

13-5 Star Schemas 592

13-5a Facts 592

13-5b Dimensions 592

13-5c Attributes 593

13-5d Attribute Hierarchies 595

13-5e Star Schema Representation 596

13-5f Performance-Improving Techniques for the Star Schema 598

- 13-6 Online Analytical Processing 602**
 - 13-6a Multidimensional Data Analysis Techniques 602
 - 13-6b Advanced Database Support 604
 - 13-6c Easy-to-Use End-User Interfaces 604
 - 13-6d OLAP Architecture 604
 - 13-6e Relational OLAP 607
 - 13-6f Multidimensional OLAP 608
 - 13-6g Relational versus Multidimensional OLAP 609
- 13-7 Data Analytics 610**
 - 13-7a Data Mining 610
 - 13-7b Predictive Analytics 613
- 13-8 SQL Analytic Functions 614**
 - 13-8a The ROLLUP Extension 615
 - 13-8b The CUBE Extension 616
 - 13-8c Materialized Views 618
- 13-9 Data Visualization 621**
 - 13-9a The Need for Data Visualization 622
 - 13-9b The Science of Data Visualization 624
 - 13-9c Understanding the Data 626
- 13-10 Data Lake 627**
 - Summary 628
 - Key Terms 628
 - Review Questions 629
 - Problems 630

Chapter 14

Big Data and NoSQL 640

- 14-1 Big Data 641**
 - 14-1a Volume 643
 - 14-1b Velocity 644
 - 14-1c Variety 645
 - 14-1d Other Characteristics 646
- 14-2 Hadoop 647**
 - 14-2a HDFS 648
 - 14-2b MapReduce 650
 - 14-2c Hadoop Ecosystem 652
 - 14-2d Hadoop Pushback 654
- 14-3 NoSQL 654**
 - 14-3a Key-Value Databases 655
 - 14-3b Document Databases 656
 - 14-3c Column-Oriented Databases 657
 - 14-3d Graph Databases 660
 - 14-3e Aggregate Awareness 662
- 14-4 NewSQL Databases 662**
- 14-5 Working with Document Databases Using MongoDB 663**

- 14-5a Importing Documents in MongoDB 664
- 14-5b Example of a MongoDB Query Using find() 665

14-6 Working with Graph Databases Using Neo4j 666

- 14-6a Creating Nodes in Neo4j 667
- 14-6b Retrieving Node Data with MATCH and WHERE 668
- 14-6c Retrieving Relationship Data with MATCH and WHERE 669

Summary 670

Key Terms 672

Review Questions 673

Part 5

Databases and the Internet

Chapter 15

Database Connectivity and Web Technologies 675

15-1 Database Connectivity 676

- 15-1a Native SQL Connectivity 677
- 15-1b ODBC, DAO, and RDO 678
- 15-1c OLE-DB 680
- 15-1d ADO.NET 683
- 15-1e Java Database Connectivity (JDBC) 685

15-2 Database Internet Connectivity 686

- 15-2a Web-to-Database Middleware: Server-Side Extensions 687
- 15-2b Web Server Interfaces 689
- 15-2c The Web Browser 690
- 15-2d Client-Side Extensions 691
- 15-2e Web Application Servers 692
- 15-2f Web Database Development 692

15-3 Extensible Markup Language (XML) 696

- 15-3a Document Type Definitions (DTD) and XML Schemas 698
- 15-3b XML Presentation 700
- 15-3c XML Applications 702

15-4 Cloud Computing Services 703

- 15-4a Cloud Implementation Types 706
- 15-4b Characteristics of Cloud Services 706
- 15-4c Types of Cloud Services 707
- 15-4d Cloud Services: Advantages and Disadvantages 708
- 15-4e SQL Data Services 709

Summary 710

Key Terms 711

Review Questions 712

Problems 713

Part 6

Database Administration

Chapter 16

Database Administration and Security 715

- 16-1 Data as a Corporate Asset 716**
- 16-2 The Need for a Database and Its Role in an Organization 718**
- 16-3 Introduction of a Database: Special Considerations 719**
- 16-4 The Evolution of Database Administration 721**
- 16-5 The Database Environment's Human Component 724**
 - 16-5a The DBA's Managerial Role 726
 - 16-5b The DBA's Technical Role 731
- 16-6 Security 737**
 - 16-6a Security Policies 738
 - 16-6b Security Vulnerabilities 738
 - 16-6c Database Security 740

- 16-7 Database Administration Tools 741**
 - 16-7a The Data Dictionary 742
 - 16-7b Case Tools 744
- 16-8 Developing a Data Administration Strategy 746**
- 16-9 The DBA's Role in the Cloud 748**
- 16-10 The DBA at Work: Using Oracle for Database Administration 749**
 - 16-10a Oracle Database Administration Tools 750
 - 16-10b Ensuring That the RDBMS Starts Automatically 751
 - 16-10c Creating Tablespaces and Datafiles 751
 - 16-10d Managing Users and Establishing Security 753
 - 16-10e Customizing the Database Initialization Parameters 756
- Summary 757**
- Key Terms 758**
- Review Questions 758**

Glossary 760

Index 775

The following appendices are included on the Instructor and Student Resource Sites at www.cengage.com.

- Appendix A: Designing Databases with Lucidchart: A Tutorial**
- Appendix B: The University Lab: Conceptual Design**
- Appendix C: The University Lab: Conceptual Design Verification, Logical Design, and Implementation**
- Appendix D: Converting an ER Model into a Database Structure**
- Appendix E: Comparison of ER Modeling Notations**
- Appendix F: Client/Server Systems**
- Appendix G: Object-Oriented Databases**
- Appendix H: Unified Modeling Language (UML)**
- Appendix I: Databases in Electronic Commerce**
- Appendix J: Web Database Development with ColdFusion**
- Appendix K: The Hierarchical Database Model**
- Appendix L: The Network Database Model**
- Appendix M: MS Access Tutorial**
- Appendix N: Creating a New Database Using Oracle**
- Appendix O: Data Warehouse Implementation Factors**
- Appendix P: Working with MongoDB**
- Appendix Q: Working with Neo4j**

About the Authors

Carlos Coronel is currently the IT Resources Director for the Jones College of Business at Middle Tennessee State University. He has more than 30 years of experience in various fields as a Database Administrator, Network Administrator, Web Manager, and Technology Entrepreneur and Innovator. He has taught courses in web development, database design and development, and data communications at the undergraduate and graduate levels.

Steven Morris earned the Ph.D. in Management Information Systems from Auburn University. He is Professor of Information Systems and Analytics in the Jones College of Business at Middle Tennessee State University. He has over 20 years of experience working with and teaching database systems and is actively engaged with consulting and professional database training with businesses in the Middle Tennessee and Nashville areas. He primarily teaches courses at the graduate and undergraduate levels in database design, advanced database programming, and Big Data for analytics.

Preface

It is our great pleasure to present the fourteenth edition of *Database Systems*. We are grateful and humbled that so many of our colleagues around the world have chosen this text to support their classes. We wrote the first edition of this book because we wanted to explain the complexity of database systems in a language that was easy for students to understand. Over the years, we have maintained this emphasis on reaching out to students to explain complex concepts in a practical, approachable manner. This resource has been successful through thirteen editions because the authors, editors, and the publisher paid attention to the impact of technology and to adopters' questions and suggestions. We believe that the fourteenth edition successfully reflects the same attention to such factors.

The Approach: A Continued Emphasis on Design

As the title suggests, *Database Systems: Design, Implementation, and Management* covers three broad aspects of database systems. However, for several important reasons, special attention is given to database design.

- The availability of excellent database software enables people with little experience to create databases and database applications. Unfortunately, the “create without design” approach usually paves the road to a number of database disasters. In our experience, many database system failures are traceable to poor design and cannot be solved with the help of even the best programmers and managers. Nor is better DBMS software likely to overcome problems created or magnified by poor design. Even the best bricklayers and carpenters can't create a good building from a bad blueprint.
- Most vexing problems of database system management seem to be triggered by poorly designed databases. It hardly seems worthwhile to use scarce resources to develop excellent database management skills merely to use them on crises induced by poorly designed databases.
- Design provides an excellent means of communication. Clients are more likely to get what they need when database system design is approached carefully and thoughtfully. In fact, clients may discover how their organizations really function once a good database design is completed.
- Familiarity with database design techniques promotes understanding of current database technologies. For example, because data warehouses derive much of their data from operational databases, data warehouse concepts, structures, and procedures make more sense when the operational database's structure and implementation are understood.

Because the practical aspects of database design are stressed, we have covered design concepts and procedures in detail, making sure that the numerous end-of-chapter problems and cases are sufficiently challenging, so students can develop real and useful design skills. We also make sure that students understand the potential and actual conflicts between database design elegance, information requirements, and transaction processing speed. For example, it makes little sense to design databases that meet design elegance standards while they fail to meet end-user information requirements. Therefore, we explore the use of carefully defined trade-offs to ensure that the databases meet end-user requirements while conforming to high design standards.

Prerequisites

Students using these materials are expected to be familiar with basic system concepts and know the difference between hardware and software. Familiarity with basic productivity software such as MS Excel and MS Access is useful. Prior programming experience is not required.

Instructors can configure their classes as best suits their resources and pedagogical approach. Data modeling can be done with a number of computer resources, some of which are free to use, or drawn by hand. Coding problems will require a DBMS. The MindTap resources include hands-on programming activities with an embedded MySQL database sandbox for students to work in. Other instructors may choose to have each student run their own individual DBMS, while still others may choose an instructor-administered central DBMS to which all students connect to work. The text supports all of these approaches.

Target Market

The text is written in a comprehensive manner that allows usage at a wide range of academic levels. The skills covered in the text are in high demand and are applicable in programs from two-year degrees, four-year degrees, graduate degrees, and adult continuing education. They are even taught in some specialized high school programs. We have reports of the text being used at all of these types of academic venues. The text is written in an easily approachable style that starts with the most basic concepts and builds to advanced topics. The modular nature of the chapters makes it easy for instructors to adapt the coverage to the level of sophistication that is most appropriate for their students.

Changes to the Fourteenth Edition

In many respects, revising a resource like this is more difficult than writing it the first time. If the text is successful, as this one is, a major concern is that the updates, inserts, and deletions will adversely affect writing style and continuity of coverage. The combination of superb reviewers and editors, plus a wealth of feedback from adopters and students of the previous editions, helped make this new edition the best yet.

In this fourteenth edition, we have responded to the requests and suggestions of numerous adopters. The database arena is vast, wonderful, and in constant change. The role of data in society and industry is ever increasing, and database and data management technologies are constantly changing to address new challenges. To help you address changing learning needs as efficiently as possible, we've prioritized the following changes in this edition.

Streamlined Coverage

Scope creep is a real challenge because the ingenuity of our industry continually finds new ways to solve data problems. In this edition, we have streamlined many tangential topics that, while important, are not suitable for the depth of coverage previously provided.

Reorganized SQL

We have reorganized the SQL coverage to make sure basics are covered before introducing more advanced topics. This ensures the presentation is easier to follow and avoids overwhelming students. We start with simple SQL statements to familiarize students with the basic SQL syntax and environment. This approach helps students build the confidence to transition to the more advanced SQL features and commands. These changes provide a better flow of material.

MySQL Examples

We continue our tradition of supporting multiple DBMS products (MS Access, Oracle, SQL Server, and MySQL). In cases where code or concepts are the same in all of these products, we have shifted many of the illustrations and examples into MySQL to provide a more consistent appearance. In cases where the code is different across products, we have multiple notes and alternative examples to clarify the concepts for other DBMS products. It is our intention that faculty using any of these products, or none of them, can use this edition seamlessly in their

classes. All students will benefit from consistency in the presentation. Faculty that support the text by using the coding resources available in MindTap can also integrate the coding problems more easily with the text.

Note

If instructors want students to complete the hands-on data manipulation assignments provided in the text, they need to ensure that students have access to a DBMS product. Further, some exercises involving MS Excel or MS Access require a complete version of that product. The hands-on coding exercises for MongoDB and Neo4j also require students to have access to that software.

Coding Labs in MindTap

The fourteenth edition presents a major step forward in the integration of digital content with this resource by providing automatically graded coding labs through the MindTap available for this product. The labs allow students to write SQL code in an interactive environment that provides immediate feedback on problems.

Here is a summary of key changes in the fourteenth edition:

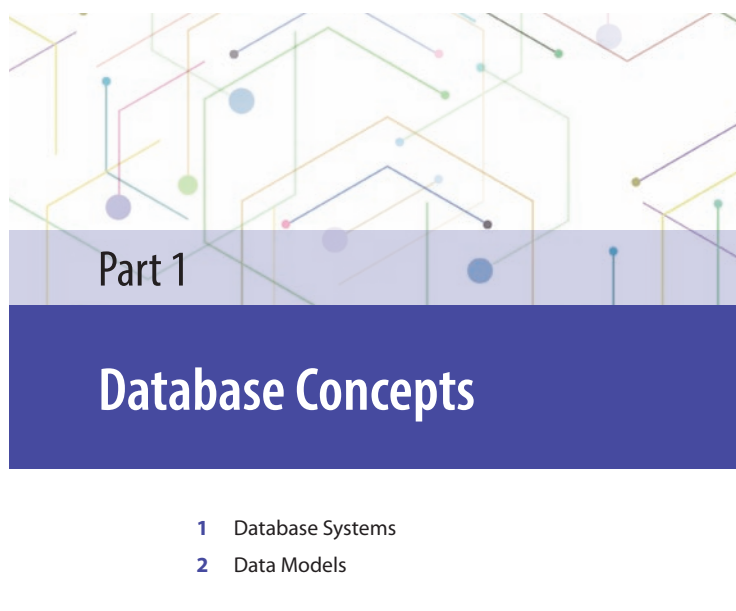
- Streamlined topic coverage, including Big Data technologies
- Reorganization of SQL to ensure basics are covered before more advanced topics
- Enhanced consistency and support for multiple DBMS products

Topical Organization

The fourteenth edition continues to provide a solid and practical foundation for the design, implementation, and management of database systems. This foundation is built on the notion that, while databases are very practical, their successful creation depends on understanding the important concepts that define them. It's not easy to come up with the proper mix of theory and practice, but the previously mentioned feedback suggests that we largely succeeded in our quest to maintain the proper balance.

The Systems View

The title for this text begins with *Database Systems*. Therefore, we examine the database and design concepts covered in Chapters 1–6 as part of a larger whole by placing them within the systems analysis framework of Chapter 9. Database designers who fail to understand that the database is part of a larger system are likely to overlook important design requirements. In fact, Chapter 9, Database Design, provides the map for the advanced database design developed in Appendices B and C. Within the larger systems framework, we can also explore issues such as transaction management and concurrency control (Chapter 10), distributed database management systems (Chapter 12), business intelligence and data warehouses (Chapter 13), new technologies for Big Data (Chapter 14), database connectivity and web technologies (Chapter 15), and database administration and security (Chapter 16).



Database Design

The first item in our subtitle is **Design**, and our examination of database design is comprehensive. For example, Chapters 1 and 2 examine the development and future of databases and data models and illustrate the need for design. Chapter 3 examines the details of the relational database model; Chapter 4 provides extensive, in-depth, and practical database design coverage; and Chapter 5 explores advanced database design topics. Chapter 6 is devoted to critical normalization issues that affect database efficiency and effectiveness. Chapter 9 examines database design within the systems framework and maps the activities required to successfully design and implement the complex, real-world database developed in Appendices B and C. Appendix A is a good introductory tutorial on designing databases with Lucidchart.

Because database design is affected by real-world transactions, the way data is distributed, and ever-increasing information requirements, we examine major database features that must be supported in current-generation databases and models. For example, Chapter 10, Transaction Management and Concurrency Control, focuses on the characteristics of database transactions and how they affect database integrity and consistency. Chapter 11, Database Performance Tuning and Query Optimization, illustrates the need for query efficiency in a world that routinely generates and uses terabyte-size databases and tables with millions of records. Chapter 12, Distributed Database Management Systems, focuses on data distribution, replication, and allocation. In Chapter 13, Business Intelligence and Data Warehouses, we explore the characteristics of databases that are used in decision support and online analytical processing, including coverage of data visualization and data analytics. Chapter 14, Big Data and NoSQL, explores the challenges of leveraging nonrelational databases to use vast global stores of unstructured data. Chapter 15, Database Connectivity and Web Technologies, covers the basic database connectivity issues in a web-based data world, development of web-based database front ends, and emerging cloud-based services.

Implementation

The second portion of the subtitle is **Implementation**. We use Structured Query Language (SQL) in Chapters 7 and 8 to show how relational databases are implemented and managed. Appendix M, MS Access Tutorial, provides a quick but comprehensive guide to implementing an MS Access database. Appendices B and C demonstrate the design of a database that was fully implemented; these appendices illustrate a wide range of implementation issues. We had to deal with conflicting design goals: design elegance, information requirements, and operational speed. Therefore, we carefully audited the initial design in Appendix B to check its ability to meet end-user needs and establish appropriate implementation protocols. The result of this audit yielded the final design developed in Appendix C. While relational databases are still the appropriate database technology to use in the vast majority

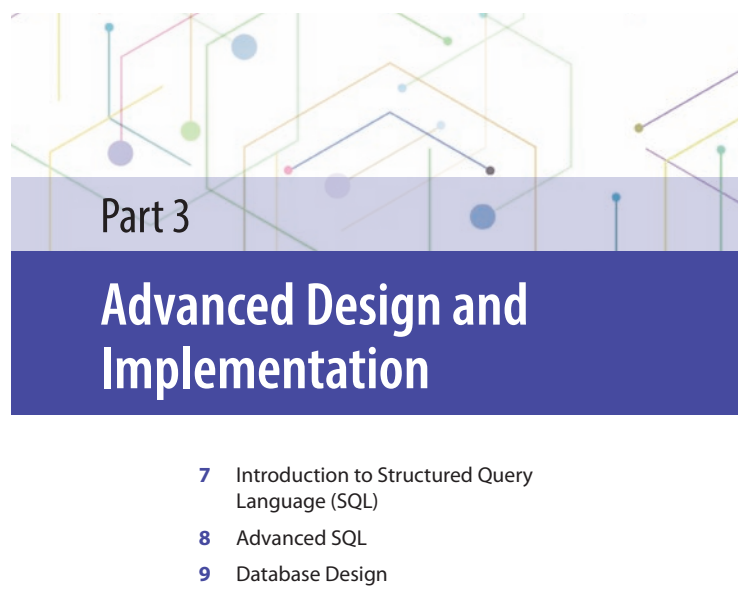


Chapter
9
Database Design

Learning Objectives
After completing this chapter, you will be able to:

- 9-1 Describe the role of database design as the foundation of a successful information system
- 9-2 Describe the five phases in the Systems Development Life Cycle (SDLC)
- 9-3 Design databases using the six phases in the Database Life Cycle (DBLC) framework
- 9-4 Conduct evaluation and revision within the SDLC and DBLC frameworks
- 9-5 Distinguish between top-down and bottom-up approaches in database design
- 9-6 Distinguish between centralized and decentralized conceptual database design

PREVIEW
Databases are a part of a larger picture called an information system. Database designs that fail to recognize this fact are not likely to be successful. Database designers must recognize that the database is a critical means to an end rather than an end in itself. Managers want the database to serve their management needs, but too many databases seem to force managers to alter their routines to fit the database requirements. Information systems don't just happen; they are the product of a carefully staged development process. Systems analysis is used to determine the need for an information system and to establish its limits. Within systems analysis, the actual



Part 3
Advanced Design and Implementation

- 7 Introduction to Structured Query Language (SQL)
- 8 Advanced SQL
- 9 Database Design



of situations, Big Data issues have created an environment in which special requirements can call for the use of new, nonrelational technologies. Chapter 14, Big Data and NoSQL, describes the types of data that are appropriate for these new technologies and the array of options available in these special cases. Appendix P, Working with MongoDB, and Appendix Q, Working with Neo4j, provide hands-on coverage of using MongoDB and Neo4j, some of the most popular NoSQL options. The special issues encountered in an Internet database environment are addressed in Chapter 15, Database Connectivity and Web Technologies, and in Appendix J, Web Database Development with ColdFusion.

Management

The final portion of the subtitle is **Management**. We deal with database management issues in Chapter

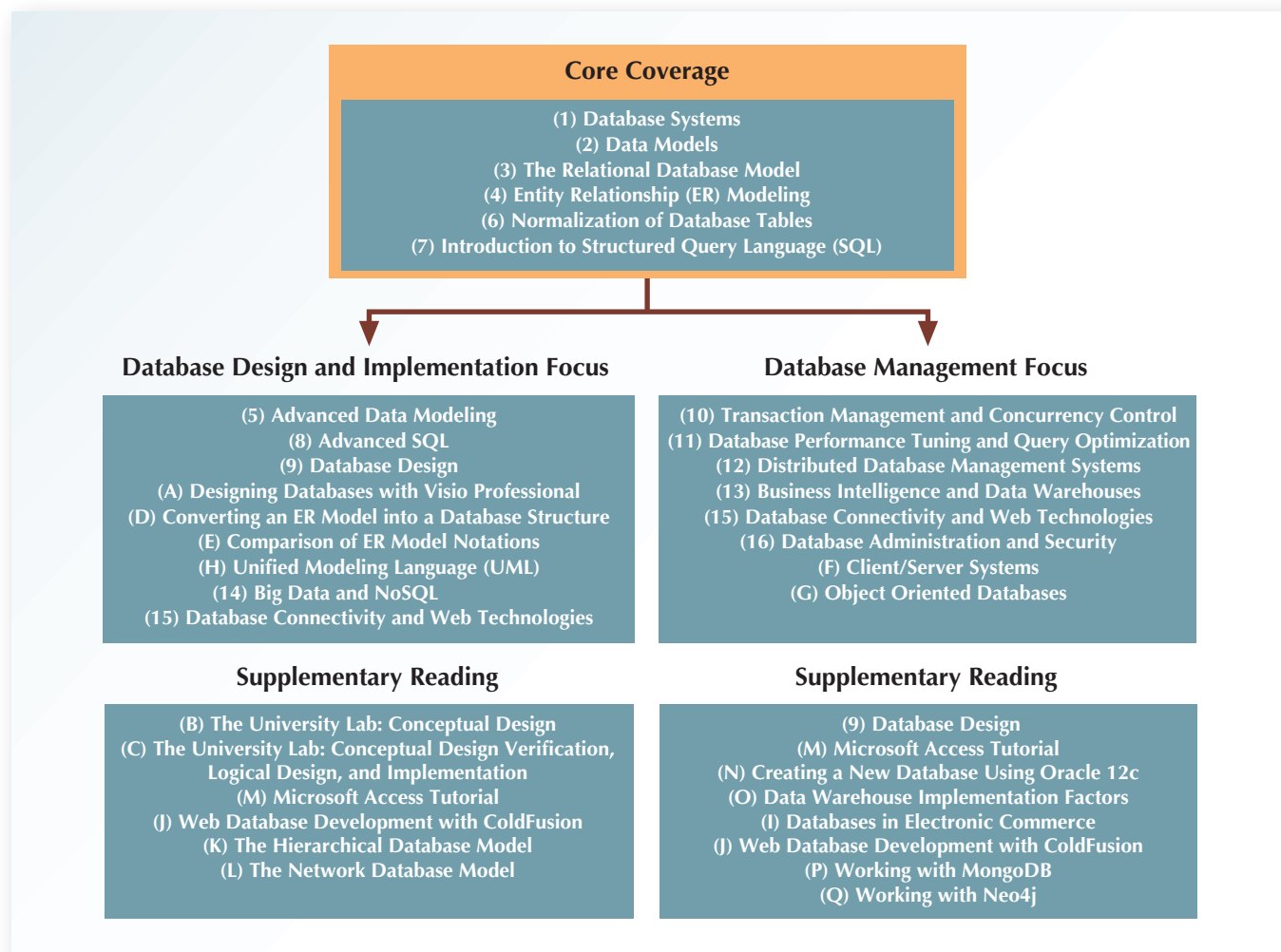
10, Transaction Management and Concurrency Control; Chapter 12, Distributed Database Management Systems; and Chapter 16, Database Administration and Security. Chapter 11, Database Performance Tuning and Query Optimization, is a valuable resource that illustrates how a DBMS manages data retrieval. In addition, Appendix N, Creating a New Database Using Oracle, walks you through the process of setting up a new database.

Teaching Database: A Matter of Focus

Given the wealth of detailed coverage, instructors can “mix and match” chapters to produce the desired coverage. Depending on where database courses fit into the curriculum, instructors may choose to emphasize database design or database management. (See Figure 1.)

The hands-on nature of database design lends itself particularly well to class projects in which students use instructor-selected software to prototype a system that they design for the end user. Several end-of-chapter problems are sufficiently complex to serve as projects, or an instructor may work with local businesses to give students hands-on experience. Note that some elements of the database design track are also found in the database management track, because it is difficult to manage database technologies that are not well understood.

The options shown in Figure 1 serve only as a starting point. Naturally, instructors will tailor their coverage based on their specific course requirements. For example, an instructor may decide to make Appendix I an outside reading assignment and make Appendix A a self-taught tutorial, and then use that time to cover client/server systems or object-oriented databases. The latter choice would serve as a gateway to UML coverage.

Figure 1

Text Features

Online Content boxes draw attention to material at www.cengage.com for this text and provide ideas for incorporating this content into the course.

Online Content

The file structures you see in this problem set are simulated in a Microsoft Access database named Ch01_Problems, which is available at www.cengage.com.

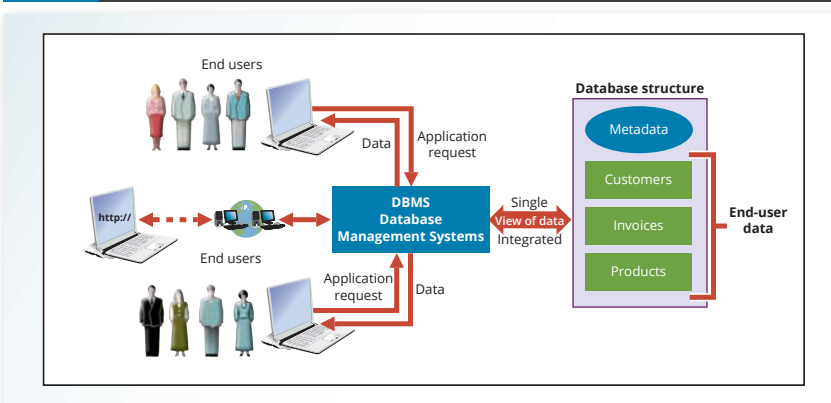
Notes highlight key facts about the concepts introduced in the chapter.

Note

The terms *data model* and *database model* are often used interchangeably. In this book, the term *database model* is used to refer to the implementation of a *data model* in a specific database system.

A variety of **four-color figures**, including ER models and implementations, tables, and illustrations, clearly illustrate difficult concepts.

Figure 1.4 The DBMS Manages the Interaction between the End User and the Database



A robust **Summary** at the end of each chapter ties together the major concepts and serves as a quick review for students.

Summary

- Data consists of raw facts. Information is the result of processing data to reveal its meaning. Accurate, relevant, and timely information is the key to good decision making, and good decision making is the key to organizational

files, each requiring its own data management program. Although this method of data management is largely outmoded, understanding its characteristics makes database design easier to comprehend.

An alphabetic list of **Key Terms** summarizes important terms.

Key Terms

ad hoc query	centralized database	data dictionary
analytical database	cloud database	data inconsistency
application programming interface (API)	data	data independence
business intelligence	data anomaly	data integrity
	data dependence	data management

Review Questions challenge students to apply the skills learned in each chapter.

Review Questions

- Define each of the following terms:
 - data
 - field
 - record
 - file
- What are the main components of a database?
- What is metadata?
- Explain why database design is important.
- What are the potential costs of implementing a database system?

Problems become progressively more complex as students draw on the lessons learned from the completion of preceding problems.

Problems

Given the file structure shown in Figure P1.1, answer Problems 1–4.

Figure P1.1 The File Structure for Problems 1–4

PROJECT_CODE	PROJECT_MANAGER	MANAGER_PHONE	MANAGER_ADDRESS	PROJECT_BID_PRICE
21-5Z	Holly B. Parker	904-338-3416	3334 Lee Rd., Gainesville, FL 37123	16833460.00
25-2D	Jane D. Grant	615-898-9909	218 Clark Blvd., Nashville, TN 36362	12500000.00
25-5A	George F. Dorts	615-227-1245	124 River Dr., Franklin, TN 29185	32512420.00
25-9T	Holly B. Parker	904-338-3416	3334 Lee Rd., Gainesville, FL 37123	21563234.00
27-4Q	George F. Dorts	615-227-1245	124 River Dr., Franklin, TN 29185	10314545.00
29-2D	Holly B. Parker	904-338-3416	3334 Lee Rd., Gainesville, FL 37123	25559999.00
31-7P	William K. Moor	904-445-2719	216 Morton Rd., Stetson, FL 30155	56850000.00

Additional Features

MindTap® for Database Systems, Fourteenth Edition

MindTap for *Database Systems, Fourteenth Edition* is an online learning solution designed to help students master the skills they need to thrive in today's workforce. Research shows employers need critical thinkers, troubleshooters, and creative problem-solvers to stay relevant in our fast-paced, technology-driven marketplace. MindTap helps prepare you for that marketplace with relevant assignments and activities including hands-on practice. Students are guided through assignments that progress from basic knowledge and understanding to more challenging problems. MindTap activities and assignments are tied to validated learning objectives.

Additional Resources for Students and Instructors

Instructor and student resources for this product are available online. Instructor assets include an instructor manual, data files, an educator's guide, PowerPoint® slides, an image gallery, and a test bank powered by Cengage®. Student assets include data files. Sign up or sign in at www.cengage.com to search for and access this product and its online resources.

For further detail about instructor resources, read on.

Appendices

Seventeen online appendices provide additional material on a variety of important areas, such as Lucidchart® and Microsoft Access, ER model notations, UML, object-oriented databases, databases and electronic commerce, Adobe® ColdFusion®, and working with newer NoSQL databases MongoDB and Neo4j.

Database, SQL Script, JSON Documents, and ColdFusion Files

The online materials for this resource include all of the database structures and table contents used in the text. For students using Oracle®, MySQL, and Microsoft SQL Server™, SQL scripts are included to help students create and load all tables used in the SQL chapters (7 and 8). Text documents for importing JSON-formatted documents into MongoDB and a script for creating a graph database in Neo4j (Appendices P and Q) are also included. In addition, all ColdFusion scripts used to develop the web interfaces in Appendix J are included.

Instructor Manual

The instructor manual that accompanies this course provides additional instructional material to assist in class preparation, including suggestions for classroom activities, discussion topics, and additional projects.

Solutions and Answer Guide

Answers to the Review Questions, Problems, Database for Life, and Reflection activities are provided. Lab solutions are provided separately.

SQL Script Files for Instructors

The authors have provided SQL script files to allow instructors to cut and paste the SQL code into the SQL windows. (Scripts are provided for Oracle, MySQL, and MS SQL Server.) The SQL scripts, which have all been tested by Cengage Learning, are a major convenience for instructors. You won't have to type in the SQL commands, and the use of the scripts eliminates typographical errors that are sometimes difficult to trace.

ColdFusion Files for Instructors

The ColdFusion web development solutions are provided. Instructors have access to a menu-driven system that allows them to show the code as well as its execution.

Databases

For many chapters, Microsoft Access instructor databases are available that include features not found in the student databases. For example, the databases that accompany Chapters 7 and 8 include many of the queries that produce the problem solutions. Other Access databases, such as the ones that accompany Chapters 3, 4, 5, and 6, include implementations of the design problem solutions to allow instructors to illustrate the effect of design decisions. In addition, instructors have access to all the script files for Oracle, MySQL, and MS SQL Server so that all the databases and their tables can be converted easily and precisely.

Cengage Testing Powered by Cognero

Cognero is a flexible, online system that allows you to:

- Author, edit, and manage test bank content from multiple Cengage solutions.
- Create multiple test versions in an instant.
- Deliver tests from your LMS, your classroom, or wherever you want.

PowerPoint® Presentations

This course comes with Microsoft PowerPoint slides for each module. These are included as a teaching aid for classroom presentation, to make available to students on the network for module review, or to be printed for classroom distribution. Instructors, please feel at liberty to add your own slides for additional topics you introduce to the class.

Figure Files

All of the figures in the course are reproduced on the Instructor Resource Site. Similar to the PowerPoint presentations, these are included as a teaching aid for classroom presentation, to make available to students for review, or to be printed for classroom distribution.

Acknowledgments

Regardless of how many editions of *Database Systems* are published, they will always rest on the solid foundation created by the first edition. We remain convinced that our work has become successful because that first edition was guided by Frank Ruggirello, a former Wadsworth senior editor and publisher. Aside from guiding the book's development, Frank also managed to solicit the great Peter Keen's evaluation (thankfully favorable) and subsequently convinced Peter Keen to write the foreword for the first edition. Although we sometimes found Frank to be an especially demanding task master, we also found him to be a superb professional and a fine friend. We suspect Frank will still see his fingerprints all over our current work. Many thanks.

A difficult task in revising this resource is deciding what new approaches, topical coverage, and changes to depth of coverage are appropriate for a product that has successfully weathered the test of the marketplace. The comments and suggestions made by adopters, students, and reviewers play a major role in deciding what coverage is desirable and how that coverage is to be treated.

Some adopters became extraordinary reviewers, providing incredibly detailed and well-reasoned critiques even as they praised the topic coverage and style. Dr. David Hatherly, a superb database professional who is a senior lecturer in the School of Information Technology, Charles Sturt University–Mitchell, Bathurst, Australia, made sure that we knew precisely what issues led to his critiques. Even better for us, he provided the suggestions that made it much easier for us to improve the topical coverage in earlier editions. All of his help was given freely and without prompting on our part. His efforts are much appreciated, and our thanks are heartfelt.

We also owe a debt of gratitude to Professor Emil T. Cipolla, who teaches at St. Mary College. Professor Cipolla's wealth of IBM experience turned out to be a valuable resource when we tackled the embedded SQL coverage in Chapter 8.

Every technical resource receives careful scrutiny by several groups of reviewers selected by the publisher. We were fortunate to benefit from the scrutiny of reviewers who were superbly qualified to offer their critiques, comments, and suggestions—many of which strengthened this edition. While holding them blameless for any remaining shortcomings, we owe these reviewers many thanks for their contributions:

Wael Jabr, Assistant Professor,
Smeal College of Business,
Pennsylvania State University

David Goldberg
San Diego State University

J. Ken. Corley II
Appalachian State University

In some respects, writing books resembles building construction: When 90 percent of the work seems done, 90 percent of the work remains to be done. Fortunately for us, we had a great team on our side.

- We are deeply indebted to Lisa Ruffolo for her help and guidance. Lisa has been a godsend. We write what we think, then Lisa helps us turn it into what we meant to say. As authors, the Development Editor is our closest point of contact with the publisher. Lisa was immediately in sync with what we were trying to do, our style of writing, and the voice for which we strive. Her precision and attention to detail were amazing, and we are deeply indebted to her for her invaluable contributions.
- After writing so many books and fourteen editions of *this* book, we know just how difficult it can be to transform the authors' work into an attractive product. The content and production teams, both at Cengage (Michele Stulga) and Straive (Arun Kumar Vasu), have done an excellent job.

- We also owe Michele Stulga, our content developer, special thanks for her ability to guide this book to a successful conclusion, and Danielle Shaw, our technical editor, deserves many thanks for making sure all code and technical references were accurate.

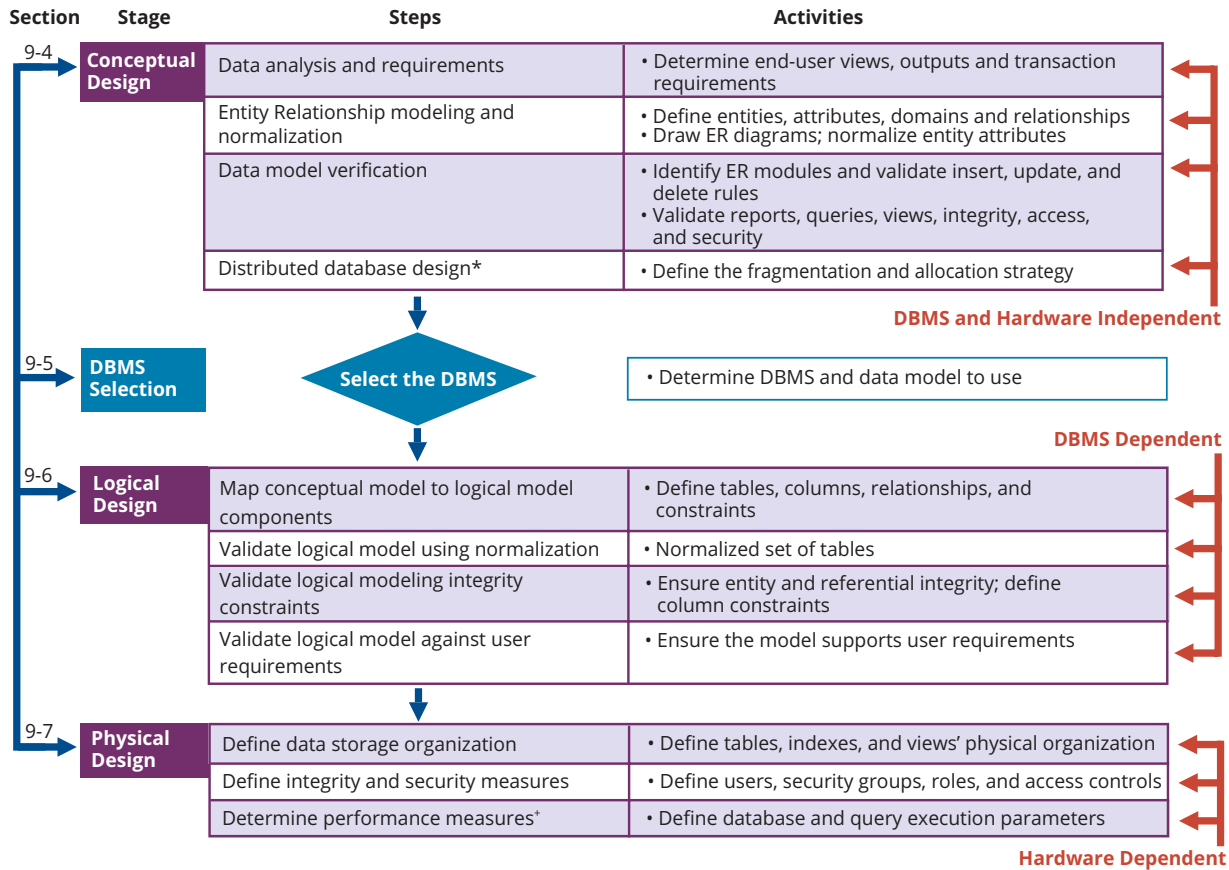
We also thank our students for their comments and suggestions. They are the reason for writing this book in the first place. One comment stands out in particular: “I majored in systems for four years, and I finally discovered why when I took your course.” And one of our favorite comments by a former student was triggered by a question about the challenges created by a real-world information systems job: “Doc, it’s just like class, only easier. You really prepared me well. Thanks!”

Special thanks go to a unique and charismatic gentleman—for over 20 years, Peter Rob has been the driving force behind the creation and evolution of this book. This book originated as a product of his drive and dedication to excellence. For over 22 years, he was the voice of *Database Systems* and the driving force behind its advancement. We wish him peace in his retirement, time with his loved ones, and luck on his many projects.

Last, and certainly not least, we thank our families for their solid support at home. They graciously accepted the fact that during more than a year’s worth of rewriting, there would be no free weekends, rare free nights, and even rarer free days. We owe you much, and the dedications we wrote are but a small reflection of the important space you occupy in our hearts.

Carlos Coronel and Steven Morris

Database Design Process



* See Chapter 12, Distributed Database Management Systems

* See Chapter 11, Database Performance Tuning and Query Optimization

Database Design Process

Business Rules

- Properly document and verify all business rules with the end users.
- Ensure that all business rules are written precisely, clearly, and simply. The business rules must help identify entities, attributes, relationships, and constraints.
- Identify the source of all business rules, and ensure that each business rule is justified, dated, and signed off by an approving authority.

Data Modeling

Naming Conventions: All names should be limited in length (database-dependent size).

Entity Names:

- Should be nouns that are familiar to business and should be short and meaningful
- Should document abbreviations, synonyms, and aliases for each entity
- Should be unique within the model
- For composite entities, may include a combination of abbreviated names of the entities linked through the composite entity

Attribute Names:

- Should be unique within the entity
- Should use the entity abbreviation as a prefix
- Should be descriptive of the characteristic
- Should use suffixes such as _ID, _NUM, or _CODE for the PK attribute
- Should not be a reserved word
- Should not contain spaces or special characters such as @, !, or &

Relationship Names:

- Should be active or passive verbs that clearly indicate the nature of the relationship

Entities:

- Each entity should represent a single subject.
- Each entity should represent a set of distinguishable entity instances.
- All entities should be in 3NF or higher. Any entities below 3NF should be justified.
- The granularity of the entity instance should be clearly defined.
- The PK is clearly defined and supports the selected data granularity.

Attributes:

- Should be simple and single-valued (atomic data)
- Should document default values, constraints, synonyms, and aliases
- Derived attributes should be clearly identified and include source(s)
- Should not be redundant unless they are justified for transaction accuracy, performance, or maintaining a history
- Nonkey attributes must be fully dependent on the PK attribute

Relationships:

- Should clearly identify relationship participants
- Should clearly define participation, connectivity, and document cardinality

ER Model:

- Should be validated against expected processes: inserts, updates, and deletes
- Should evaluate where, when, and how to maintain a history
- Should not contain redundant relationships except as required (see Attributes)
- Should minimize data redundancy to ensure single-place updates
- Should conform to the minimal data rule: "All that is needed is there and all that is there is needed."



Part 1

Database Concepts

- 1 Database Systems
- 2 Data Models

Chapter

1

Database Systems

Learning Objectives

After completing this chapter, you will be able to:

- 1-1** Define the difference between data and information
- 1-2** Describe what a database is, the various types of databases, and why they are valuable assets for decision making
- 1-3** Explain the importance of database design
- 1-4** Outline how modern databases evolved from file systems
- 1-5** Identify flaws in file system data management
- 1-6** Outline the main components of the database system
- 1-7** Describe the main functions of a database management system (DBMS)

PREVIEW

Organizations use data to keep track of their day-to-day operations. Such data is used to generate information, which in turn is the basis for good decisions. Data is likely to be managed most efficiently when it is stored in a database. Databases are involved in almost all facets and activities of our daily lives: from school to work, medical care, government, nonprofit organizations, and houses of worship. In this chapter, you will learn what a database is, what it does, and why it yields better results than other data management methods. You will also learn about various types of databases and why database design is so important.

Databases evolved from the need to manage large amounts of data in an organized and efficient manner. In the early days, computer file systems were used to organize such data. Although file system data management is now largely outmoded, understanding the characteristics of file systems is important because file systems are the source of serious data management limitations. In this chapter, you will also learn how the database system approach helps eliminate most of the shortcomings of file system data management.

Data Files and Available Formats

	MS Access	Oracle	MS SQL	MySQL
Ch01_Text	Yes	Yes	Yes	Yes
Ch01_Problems	Yes	Yes	Yes	Yes

Data Files available on cengage.com

1-1 Why Databases?


So, why do we need databases? In today's world, data is ubiquitous (abundant, global, everywhere) and pervasive (unescapable, prevalent, persistent). From birth to death, we generate and consume data. The trail of data starts with the birth certificate and continues all the way to a death certificate (and beyond!). In between, each individual produces and consumes enormous amounts of data. As you will see in this book, databases are the best way to store and manage data. Databases make data persistent and shareable in a secure way. As you look at Figure 1.1, can you identify some of the data generated by your own daily activities?

Figure 1.1 The Pervasive Nature of Databases


A Day in Susan's Life

See how many databases she interacts with each day


In the morning, Susan goes grocery shopping




Where are the product data stored?
Is the product quantity in stock updated at checkout?
Does she pay with a credit card?




Later, she picks up her prescription at the pharmacy




Where is the pharmacy inventory data stored?
What data about each product will be in the inventory data?
What data is kept about each customer and where is it stored?




In the afternoon, she orders some items online



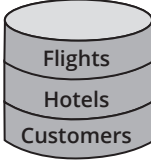
Where are the product and stock data stored?
Where does the system get the data to generate product "recommendations" to the customer?




At night, she plans for a trip and buys airline tickets and hotel reservations online



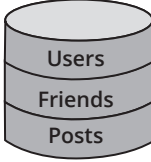
Where does the online travel website get the airline and hotel data from?
What customer data would be kept by the website?
Where would the customer data be stored?



Before going to bed, Susan checks her social media accounts



What particular customer data is kept by each website?
Where is the data about the friends and groups stored?
Where are the "likes" stored and what would they be used for?



Data is not only ubiquitous and pervasive; it is also essential for organizations to survive and prosper. Imagine trying to operate a business without knowing who your customers are, what products you are selling, who is working for you, who owes you money, and to whom you owe money. All businesses have to keep this type of data and much more. Just as important, they must have that data available to decision makers when necessary. It can be argued that the ultimate purpose of all business information systems is to help businesses use information as an organizational resource. At the heart of all of these systems are the collection, storage, aggregation, manipulation, dissemination, and management of data.

Depending on the type of information system and the characteristics of the business, this data could vary from a few megabytes on just one or two topics to petabytes covering hundreds of topics within the business's internal and external environment. Telecommunications companies, such as Sprint and AT&T, are known to have systems that keep data on trillions of phone calls, with new data being added to the system at speeds up to 70,000 calls per second! Not only do these companies have to store and manage immense collections of data, but they must be able to find any given fact in that data quickly. Consider the case of Internet search staple Google. While Google is reluctant to disclose many details about its data storage specifications, it is estimated that the company responds to over 91 million searches per day across a collection of data that is several terabytes in size. Impressively, the results of these searches are available almost instantly.

How can these businesses process this much data? How can they store it all, and then quickly retrieve just the facts that decision makers want to know, just when they want to know it? The answer is that they use databases. Databases, as explained in detail throughout this book, are specialized structures that allow computer-based systems to store, manage, and retrieve data very quickly. Virtually all modern business systems rely on databases. Therefore, a good understanding of how these structures are created and their proper use is vital for any information systems professional. Even if your career does not take you down the amazing path of database design and development, databases will be a key component of the systems that you use. In any case, you will probably make decisions in your career based on information generated from data. Thus, it is important that you know the difference between data and information.

1-2 Data versus Information

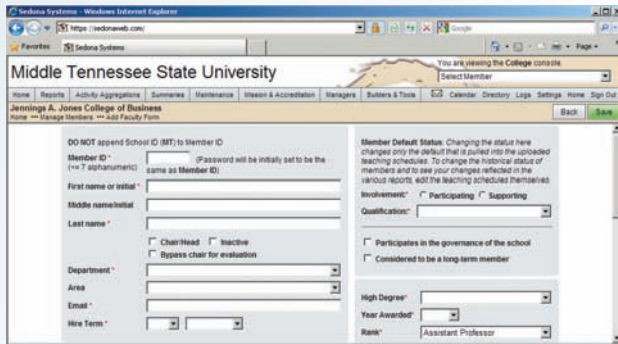
To understand what drives database design, you must understand the difference between data and information. **Data** consists of raw facts. The word *raw* indicates that the facts have not yet been processed to reveal their meaning. For example, suppose that a university tracks data on faculty members for reporting to accrediting bodies. To get the data for each faculty member into the database, you would provide a screen to allow for convenient data entry, complete with drop-down lists, combo boxes, option buttons, and other data-entry validation controls. Figure 1.2(a) shows a simple data-entry form from a software package named Sedona. When the data is entered into the form and saved, it is placed in the underlying database as raw data, as shown in Figure 1.2(b). Although you now have the facts in hand, they are not particularly useful in this format. Reading through hundreds of rows of data for faculty members does not provide much insight into the overall makeup of the faculty. Therefore, you transform the raw data into a data summary like the one shown in Figure 1.2(c). Now you can get quick answers to questions such as “What percentage of the faculty in the Information Systems (INFS) department are adjuncts?” In this case, you can quickly determine that 20 percent of the INFS faculty members are adjunct faculty. Because graphics can enhance your ability to quickly extract meaning from data, you show the data summary pie chart in Figure 1.2(d).

data

Raw facts, or facts that have not yet been processed to reveal their meaning to the end user.

Figure 1.2 Transforming Raw Data into Information

a) Data entry screen



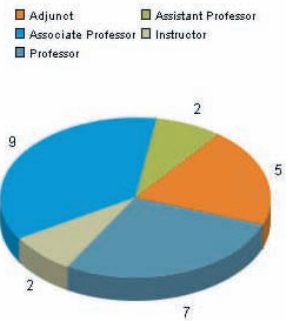
b) Raw data

ID	Last Name	McName	First Name	DeptCode	Office	Email	Rank	Hire Year	Degree
1	Washington	A.	George	MGMT	N125	gwashton@mtsu.edu	Professor	2001	Ph.D.
2	Adams	J.	John	FNH	N113	jadams@mtsu.edu	Professor	1984	Ph.D.
3	Jefferson	L.	Thomas	ECON	N126	tjefferson@mtsu.edu	Instructor	2002	M.B.A.
4	Madison	D.	James	FNH	N126	jmadison@mtsu.edu	Associate Professor	1994	Ph.D.
5	Monroe	N.	James	ACCT	N411	jmonroe@mtsu.edu	Assistant Professor	1995	Ph.D.
6	Adams	O.	John	ACCT	N418	jadams@mtsu.edu	Associate Professor	1989	Ph.D.
7	Jackson	C.	Andrew	ECON	N363	ajackson@mtsu.edu	Associate Professor	1999	Ph.D.
8	Van Buren	T.	Martin	FNH	N106	mvanbur@mtsu.edu	Professor	1988	Ph.D.
9	Hanson	R.	William	MGTG	N118	whanson@mtsu.edu	Professor	1984	Ph.D.
10	Tyler	M.	John	MGMT		jtyler@mtsu.edu	Assistant Professor	2009	Ph.D.
11	Polk	M.	Cheryl	MKTG	N143	cpolk@mtsu.edu	Associate Professor	2002	Ph.D.
12	Taylor	G.	Zachary	ACCT	N415	ztaylor@mtsu.edu	Assistant Professor	1998	Ph.D.
13	Filmore	M.	Mildred	JCB	N219	mfilmon@mtsu.edu	Professor	1992	Ph.D.
14	Pierce	A.	Franklin	MKTG	N293	fpierce@mtsu.edu	Instructor	2005	M.B.A.
15	Buchanan	T.	James	MGMT	N146	jbuchanan@mtsu.edu	Associate Professor	1998	D.B.A.
17	Lincoln	W.	Larry	MGMT	N150	lrobin@mtsu.edu	Associate Professor	1995	Ph.D.
18	Johnson		Andrew	ISYS	N160	ajohnso@mtsu.edu	Professor	1987	Ph.D.
19	Grant		Katie	MKTG	N120	kgrant@mtsu.edu	Assistant Professor	1989	D.B.A.
20	Rutherford		Hayes	ACCT	N488	hrutherford@mtsu.edu	Professor	1992	Ph.D.
21	Cornfield		Dennis	ACCT		dcornef@mtsu.edu	Assistant Professor	2018	Ph.D.
22	Arthur		Emily	ACCT	N413	emarthu@mtsu.edu	Associate Professor	2003	J.D.
23	Ovenland	G.	Robert	ACCT	N481	rovenlan@mtsu.edu	Associate Professor	1997	Ph.D.
24	Hanson	K.	Patricia	BLA	N466	phanson@mtsu.edu	Associate Professor	2001	J.D.
25	McKinley	B.	Priscilla	ISYS	N363	pmckinle@mtsu.edu	Adjunct	1984	M.S.
25	Ricevevell	F.	Hilary	MGMT	N164	hricevev@mtsu.edu	Associate Professor	2002	Ph.D.
27	Wilson	X.	Leann	BCEN	N448	lwilson@mtsu.edu	Professor	1982	Ph.D.
28	Harding	W.	Warren	MKTG	N114	wharding@mtsu.edu	Professor	1984	Ed.D.
29	Cookidge	L.	Calvin	ECON	N316	ccookidg@mtsu.edu	Professor	1975	Ph.D.
30	Hoover	L.	Lisa	MGMT		lhoover@mtsu.edu	Adjunct	1978	M.B.A.
31	Truman	B.	Betty	ACCT	N416	btruman@mtsu.edu	Professor	1971	Ed.D.
32	Johnson		Robert	BCEN	N448	rjohnson@mtsu.edu	Professor	2001	Ph.D.

c) Information in summary format

Rank	COUNT	%/INFS	TOT/COL	%/COL. TOT.	%/COL. FAC.
Adjunct	5	20.00%	23	21.74%	3.27%
Assistant Professor	2	8.00%	28	7.14%	1.31%
Associate Professor	9	36.00%	37	24.32%	5.88%
Instructor	2	8.00%	18	11.11%	1.31%
Professor	7	28.00%	47	14.89%	4.58%

d) Information in graphical format



information

The result of processing raw data to reveal its meaning. Information consists of transformed data and facilitates decision making.

knowledge

The body of information and facts about a specific subject. Knowledge implies familiarity, awareness, and understanding of information as it applies to an environment. A key characteristic is that new knowledge can be derived from old knowledge.

Information is the result of processing raw data to reveal its meaning. Data processing can be as simple as organizing data to reveal patterns or as complex as making forecasts or drawing inferences using statistical modeling. To reveal meaning, information requires *context*. For example, an average temperature reading of 105 degrees does not mean much unless you also know its context: Is this reading in degrees Fahrenheit or Celsius? Is this a machine temperature, a body temperature, or an outside air temperature? Information can be used as the foundation for decision making. For example, the data summary for the faculty can provide accrediting bodies with insights that are useful in determining whether to renew accreditation for the university.

Keep in mind that raw data must be properly *formatted* for storage, processing, and presentation. For example, dates might be stored in Julian calendar formats within the database, but displayed in a variety of formats, such as day-month-year or month/day/year, for different purposes. Respondents' yes/no responses might need to be converted to a Y/N or 0/1 format for data storage. More complex formatting is required when working with complex data types, such as sounds, videos, or images.

In this “information age,” production of accurate, relevant, and timely information is the key to good decision making. In turn, good decision making is the key to business survival in a global market. We are now said to be entering the “knowledge age.”¹

Data is the foundation of information, which is the bedrock of **knowledge**—that is, the body of information and facts about a specific subject. Knowledge implies familiarity, awareness, and understanding of information as it applies to an environment. A key characteristic of knowledge is that “new” knowledge can be derived from “old” knowledge.

¹Peter Drucker coined the phrase “knowledge worker” in 1959 in his book *Landmarks of Tomorrow*. In 1994, Esther Dyson, George Keyworth, and Dr. Alvin Toffler introduced the concept of the “knowledge age.”

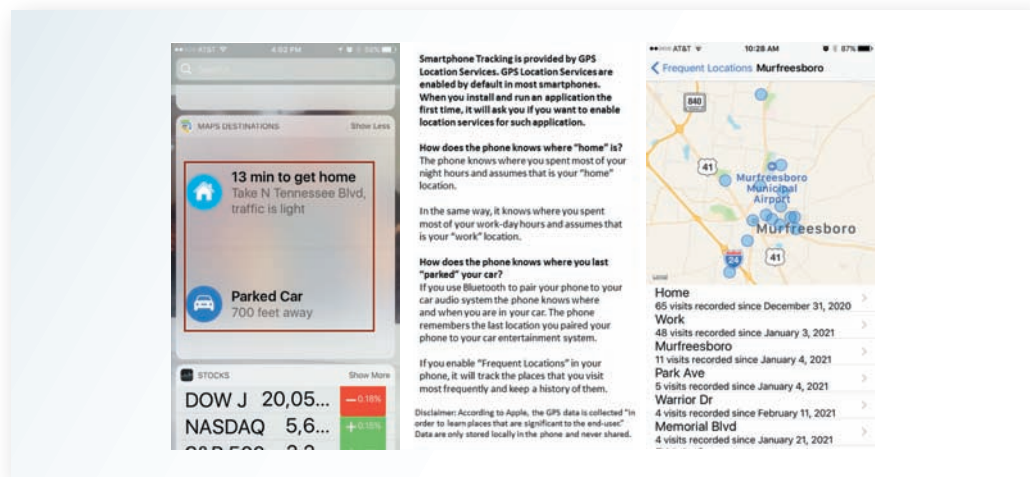
Let's summarize some key points:

- Data constitutes the building blocks of information.
- Information is produced by processing data.
- Information is used to reveal the meaning of data.
- Accurate, relevant, and timely information is the key to good decision making.
- Good decision making is the key to organizational survival in a global environment.

The previous paragraphs have explained the importance of data and how the processing of data is used to reveal information that in turn generates “actionable” knowledge. Let's explore a simple example of how this works in the real world.

In today's information-centric society, you use smartphones on a daily basis. These devices have advanced GPS functionality that constantly tracks your whereabouts. This data is stored and shared with various applications. When you get a new smartphone, you can use the map application to go places and to set up your home address (now the phone knows where you live!). The GPS feature in your phone tracks your daily locations. In some cases, the information generated is very helpful: it can help you navigate to various locations and even to find where you parked your car. Figure 1.3 shows screenshots from the smartphone of one of the authors. The phone “knows” that this is about the time he goes home and tells him how long it is going to take to get there. It also tells him where he parked his car; if he clicks the Parked Car icon, it will open a map so he can locate the car.

Figure 1.3 Smartphone Tracking



Source: Morris Coronel

Furthermore, in terms of privacy issues, your smartphone may know more about your activities than you imagine. For example, suppose that every Wednesday night you go to the gym and play indoor soccer with your friends. Next Wednesday night, 20 minutes before you leave home, your phone pops up a message saying “19 minutes to [gym address]. Traffic is light.” The phone has been storing GPS data on your movements to develop patterns based on days, times, and locations to generate this knowledge. It can then associate such knowledge as your daily activities provide more data points. Imagine that on Wednesday when you go to the Magic Box gym to play soccer, when you arrive you use Facebook on your phone to check in to the gym. Now, your phone also knows the name of the place where you go every Wednesday night.

As you can see from this example, knowledge and information require timely and accurate data. Such data must be properly generated and stored in a format that is easy to access and process. In addition, like any basic resource, the data environment must be managed carefully. **Data management** is a discipline that focuses on the proper generation, storage, and retrieval of data. Given the crucial role that data plays, it should not surprise you that data management is a core activity for any business, government agency, service organization, or charity.

data management

A process that focuses on data collection, storage, and retrieval. Common data management functions include addition, deletion, modification, and listing.

database

A shared, integrated computer structure that houses a collection of related data. A database contains two types of data: end-user data (raw facts) and metadata.

metadata

Data about data; that is, data about data characteristics and relationships. See also *data dictionary*.

database management system (DBMS)

The collection of programs that manages the database structure and controls access to the data stored in the database.

1-3 Introducing the Database

Efficient data management typically requires the use of a computer database. A **database** is a shared, integrated computer structure that stores a collection of the following:

- End-user data—that is, raw facts of interest to the end user
- **Metadata**, or data about data, through which the end-user data is integrated and managed

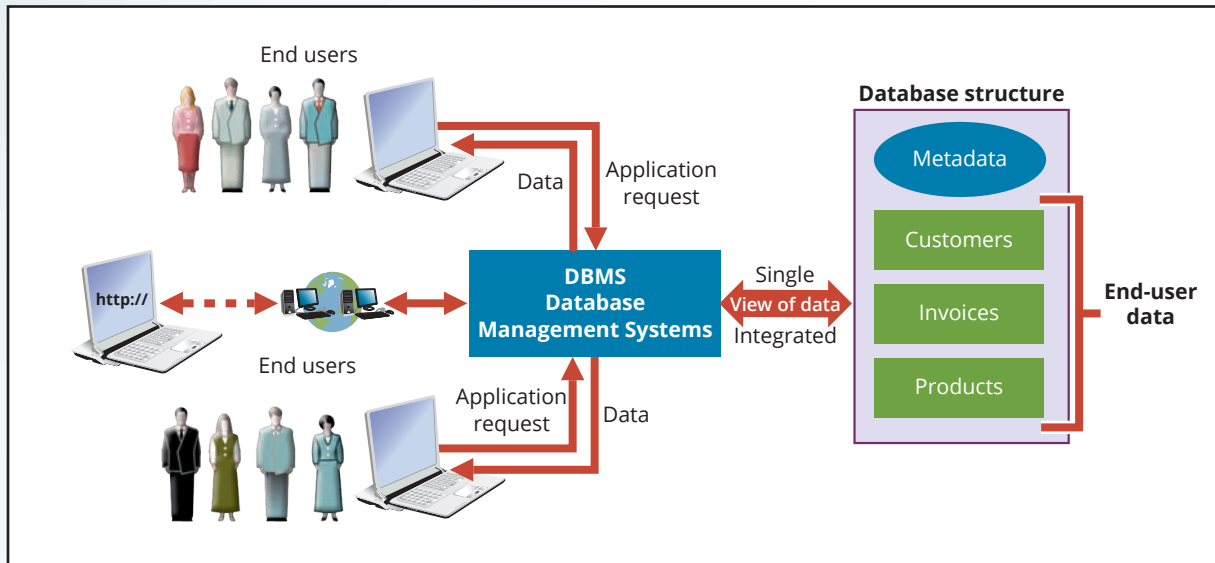
The metadata describes the data characteristics and the set of relationships that links the data found within the database. For example, the metadata component stores information such as the name of each data element, the type of values (numeric, dates, or text) stored on each data element, and whether the data element can be left empty. The metadata provides information that complements and expands the value and use of the data. In short, metadata presents a more complete picture of the data in the database. Given the characteristics of metadata, you might hear a database described as a “collection of *self-describing data*.”

A **database management system (DBMS)** is a collection of programs that manages the database structure and controls access to the data stored in the database. In a sense, a database resembles a very well-organized electronic filing cabinet in which powerful software (the DBMS) helps manage the cabinet’s contents.

1-3a Role and Advantages of the DBMS

The DBMS serves as the intermediary between the user and the database. The database structure itself is stored as a collection of files, and the only way to access the data in those files is through the DBMS. Figure 1.4 emphasizes the point that the DBMS presents the end user (or application program) with a single, integrated view of the data in the database. The DBMS receives all application requests and translates them into the complex operations required to fulfill those requests. The DBMS hides much of the database’s internal complexity from the

Figure 1.4 The DBMS Manages the Interaction between the End User and the Database



application programs and users. The application program might be written by a programmer using a programming language, such as Python, Java, or C#, or it might be created through a DBMS utility program.

Having a DBMS between the end user's applications and the database offers some important advantages. First, the DBMS enables the data in the database *to be shared* among multiple applications or users. Second, the DBMS *integrates* the many different users' views of the data into a single all-encompassing data repository.

Because data is the crucial raw material from which information is derived, you must have a good method to manage such data. As you will discover in this book, the DBMS helps make data management more efficient and effective. In particular, a DBMS provides these advantages:

- *Improved data sharing.* The DBMS helps create an environment in which end users have better access to more and better-managed data. Such access makes it possible for end users to respond quickly to changes in their environment.
- *Improved data security.* The more users access the data, the greater the risks of data security breaches. Corporations invest considerable amounts of time, effort, and money to ensure that corporate data is used properly. A DBMS provides a framework for better enforcement of data privacy and security policies.
- *Better data integration.* Wider access to well-managed data promotes an integrated view of the organization's operations and a clearer view of the big picture. It becomes much easier to see how actions in one segment of the company affect other segments.
- *Minimized data inconsistency.* **Data inconsistency** exists when different versions of the same data appear in different places. For example, data inconsistency exists when a company's sales department stores a sales representative's name as Bill Brown and the company's personnel department stores that same person's name as William G. Brown, or when the company's regional sales office shows the price of a product as \$45.95, and its national sales office shows the same product's price as \$43.95. The probability of data inconsistency is greatly reduced in a properly designed database.
- *Improved data access.* The DBMS makes it possible to produce quick answers to ad hoc queries. From a database perspective, a **query** is a specific request issued to the DBMS for data manipulation—for example, to read or update the data. Simply put, a query is a question, and an **ad hoc query** is a spur-of-the-moment question. The DBMS sends back an answer (called the **query result set**) to the application. For example, when dealing with large amounts of sales data, end users might want quick answers to questions (ad hoc queries). Some examples are the following:
 - What was the dollar volume of sales by product during the past six months?
 - What is the sales bonus figure for each of our salespeople during the past three months?
 - How many of our customers have credit balances of \$3,000 or more?
- *Improved decision making.* Better-managed data and improved data access make it possible to generate better-quality information, on which better decisions are based. The quality of the information generated depends on the quality of the underlying data. **Data quality** is a comprehensive approach to promoting the accuracy, validity, and timeliness of the data. While the DBMS does not guarantee data quality, it provides a framework to facilitate data quality initiatives. Data quality concepts will be covered in more detail in Chapter 16, Database Administration and Security.
- *Increased end-user productivity.* The availability of data, combined with the tools that transform data into usable information, empowers end users to make quick, informed decisions that can make the difference between success and failure in the global economy.

data inconsistency

A condition in which different versions of the same data yield different (inconsistent) results.

query

A question or task asked by an end user of a database in the form of SQL code. A specific request for data manipulation issued by the end user or the application to the DBMS.

ad hoc query

A "spur-of-the-moment" question.

query result set

The collection of data rows returned by a query.

data quality

A comprehensive approach to ensuring the accuracy, validity, and timeliness of data.