

# Systems Analysis and Design in a Changing World

## 系统分析 与设计

(英文版)

(美) John W. Satzinger  
Robert B. Jackson 著  
Stephen D. Burd



机械工业出版社  
China Machine Press



COURSE  
TECHNOLOGY

计算机科学丛书

# 系统分析与设计

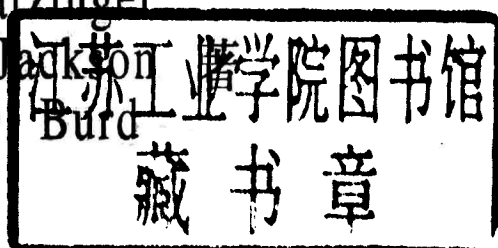
(英文版)

Systems Analysis and Design  
in a Changing World

John W. Satzinger

(美) Robert B. Jackson

Stephen D. Burd



机械工业出版社  
China Machine Press

John W.Satzinger, Robert B.Jackson and Stephen D.Burd: Systems Analysis and Design in a Changing World.

Original copyright © 2000 by Course Technology. All rights reserved.

First published by Course Technology, a division of Thomson Learning, United States of America.

Reprinted for People's Republic of China by Thomson Asia Pte Ltd and China Machine Press under the authorization of Thomson Learning. No part of this book may be reproduced in any form without the prior written permission of Thomson Learning and China Machine Press.

本书影印版由美国汤姆森公司授权机械工业出版社独家出版。未经出版者书面许可，不得以任何方式复制或抄袭本书内容。

版权所有，侵权必究。

本书版权登记号：图字：01-2000-1858

#### 图书在版编目（CIP）数据

系统分析与设计：英文 /（美）赛金格（Satzinger, J. W.），（美）杰克逊（Jackson, R. B.），（美）伯德（Burd, S. D.）著. -北京：机械工业出版社，2000.5

（计算机科学丛书）

书名原文：Systems Analysis and Design in a Changing World

ISBN 7-111-08037-8

I. 系… II. ①赛… ②杰… ③伯… III. ①系统分析-英文 ②系统设计-英文  
IV. V945

中国版本图书馆CIP数据核字（2000）第22478号

机械工业出版社（北京市西城区百万庄大街22号 邮政编码 100037）

责任编辑：廉一兵

北京第二外国语学院印刷厂印刷·新华书店北京发行所发行

2001年4月第1版第1次印刷

787mm × 1092mm 1/16 · 41印张

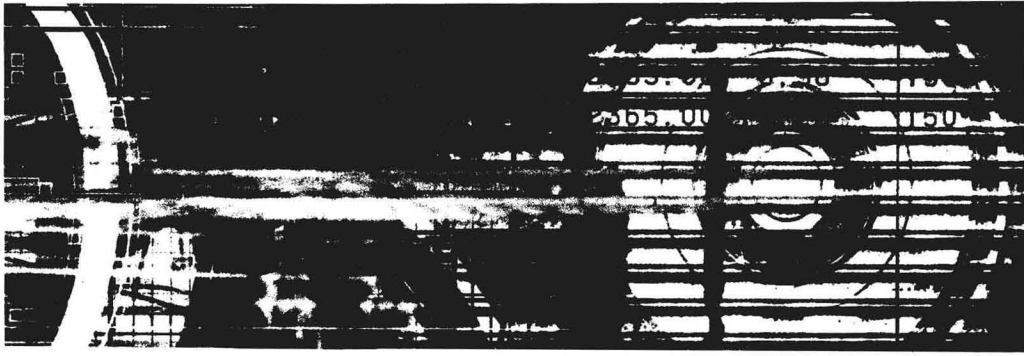
印数：0 001-3 000册

定价：60.00元

凡购本书，如有倒页、脱页、缺页，由本社发行部调换



# PREFACE



It is not easy to develop information systems in today's rapidly changing environment, but the satisfaction and rewards for a job well done are substantial. This text is designed for students and practitioners of analysis and design who are up to the challenge of trying to do it right.

Systems analysis and design is a practical field that relies on a core set of concepts and principles, as well as what sometimes seems an eclectic collection of rapidly evolving tools and techniques. Learning analysis and design today therefore requires an appreciation of the tried-and-true techniques widely embraced by experienced analysts plus the mastery of new and emerging tools and techniques that new graduates are increasingly expected to apply on the job. This text was developed for use in undergraduate and graduate courses in systems analysis and design where instructors and students do not want to ignore the past or hide from the future.

## **Objectives and Vision**

This text was developed by a team who shared a commitment to produce an analysis and design text that was different—a text that is flexible and streamlined, yet comprehensive and deep. We were guided by the belief that the text must be flexible enough to appeal to instructors emphasizing more traditional approaches to systems analysis and design as well as those emphasizing the latest object-oriented techniques. We did not want to oversimplify the problem of system development, yet we felt that it was time to reevaluate whether many of the topics and features included in competing analysis and design texts were still essential. At the same time, many new developments are affecting systems analysis and design, and we wanted to include key trends—package solutions, enterprise resource planning (ERP), components, the Internet, and so on.

We also wanted the text to teach the key concepts and techniques, not just describe them. Therefore, we focus on fundamentals of lasting value and then show how these fundamentals apply to all approaches to development. Then we explore traditional structured analysis and design and object-oriented analysis and design in depth. Flexible and streamlined? Comprehensive and deep? We think you will agree these objectives have been achieved with this text.



# DEDICATION



To JoAnn, Brian, and Kevin—JWS

To Anabel and my children for their continued support—RBJ

To Dee, Amelia, and Alex—SDB

## Innovations

This text is unique in its integration of key systems modeling concepts that apply to both the traditional structured approach and the newer object-oriented approach—events that trigger system activities and objects/entities that are part of the system's problem domain. We devote one chapter to event partitioning and modeling key objects/entities. After completing that chapter, instructors can emphasize structured analysis and design or object-oriented analysis and design, or both. The object-oriented approach is not added as an afterthought—it is assumed from the beginning that everyone should understand the key object-oriented concepts. The traditional approach is not discarded—it is assumed from the beginning that everyone should understand the key structured concepts.

The traditional approach presented in this text is based on modern structured analysis and design as refined by McMenamin and Palmer, Ed Yourdon, and Meilir Page-Jones. Modern structured analysis is an integrated, model-driven approach that includes event partitioning, data modeling with entity-relationship diagrams (ERD), and process modeling with data flow diagrams (DFDs). Modern structured design is also based on event partitioning and uses the structure chart for software design. Database design using a relational database management implementation is featured. Instructors who emphasize the structured approach to development will be satisfied by the presentation and depth of coverage in this text.

The object-oriented approach presented in this text is based on the Unified Modeling Language (UML) from the Object Management Group as originated by Booch, Rumbaugh, and Jacobson. A model-driven approach to analysis starts with use cases and scenarios and then defines classes of objects and object interactions. We include requirements modeling with use case diagrams, class diagrams, sequence diagrams, collaboration diagrams, and statechart diagrams. Design models are also discussed in detail, with particular attention to design class diagrams and statechart diagrams that are used to design methods. Our database design chapter covers two approaches to object persistence—a hybrid approach using relational database management and a pure approach using object-oriented database management systems (OODBMS). Instructors who emphasize the object-oriented approach will not be disappointed by the presentation and depth of coverage in this text.

Additional concepts and techniques are included in response to the realities of system development today. First, rapid application development and component-based development are covered in depth. Second, package solutions and enterprise resource planning (ERP) are described as alternatives to custom development.

## Features and Pedagogy

The text uses an *integrated case study* of moderate complexity—Rocky Mountain Outfitters—to illustrate key concepts and techniques on a specific system project. An overview of the strategic systems plan for Rocky Mountain Outfitters (RMO) is presented in Chapter 1 to place the system project in context. It is a planned project of strategic importance to RMO, and the need to integrate the new system with legacy systems and other planned systems is emphasized from the beginning. The planned system architecture provides for rich examples—a client-server Windows-based component as well as an e-commerce component with direct customer interaction via the Internet. Details about the case are integrated directly into each chapter to make a point or to illustrate a concept—just-in-time examples—rather than isolating the case study in separate sections of the chapters. The same system project is used to illustrate traditional and object-oriented models and solutions, so both approaches can be understood and directly compared.

Short stand-alone *opening case studies* describe a real-world situation relevant to the material in each chapter. A variety of companies and situations are included to provide the reader with a broad view of the problems and opportunities found in the real world.

Each chapter includes extensive *figures and illustrations* designed to clarify and summarize key points and to provide examples of models and other deliverables produced by an analyst. *Margin definitions* of key terms are placed in the text when the term is first used. End-of-chapter material includes a *detailed summary*, an *indexed list of key terms*, and a list of *additional resources* and references. Each chapter also includes ample *review questions*, problems and exercises to get the student *thinking critically*, a collection of *experiential exercises* involving additional research or problem solving, and end-of-chapter *case studies* that invite students to practice completing analysis and design tasks appropriate to the chapter. Some cases extend from chapter to chapter to build on earlier concepts and techniques.

### Organization and Use

The text includes fifteen chapters organized into four parts—far fewer chapters than competing texts. Additionally, many chapters are modular—sections can be skipped without loss of continuity based on the objectives for the course. Some chapters are entirely optional. The *Instructor's Manual* includes a discussion of different approaches to using the text in analysis and design courses, including suggested course outlines for instructors emphasizing the traditional structured approach or the object-oriented approach and for instructors teaching graduate courses on analysis and design.

#### Part 1: The Modern Systems Analyst

Chapter 1 discusses the work of a modern systems analyst, including a streamlined discussion of systems and the role of the systems analyst as a problem solver in a modern business organization. Chapter 2 moves right to the heart of the course—a systems development project—introduced while describing the system development life cycle (SDLC), project planning, feasibility assessment, and project management. Students are drawn quickly into the RMO project so that the material has a meaningful context. Chapter 3 then asks, Now that we have a project, what do we have to do to get this system built? That is, what are the methodologies, models, tools, and techniques that can be used to develop systems? We make it clear that there are a variety of approaches to system development and that today's analysts need to be familiar with all of them. Even if students specialize in one approach in their course or later in their job, they should be able to distinguish among the structured approaches, information engineering, and the object-oriented approach in a meaningful way.

#### Part 2: Systems Analysis Tasks

Part 2 moves ahead with systems analysis techniques. Chapter 4 covers investigating system requirements, including gathering information and interviewing system owners and users. Chapter 5 covers modeling system requirements—using our approach, which includes event partitioning and modeling objects/entities, as described earlier. Chapter 6 continues requirements modeling using the traditional approach, including data flow diagrams (DFDs), data flow definitions, process descriptions, and so on. A few additional models that an analyst might encounter are also presented. Chapter 7 continues the discussion from Chapter 5 using the object-oriented approach to requirements. Instructors can simply choose to emphasize Chapter 6 or Chapter 7 to focus the course on either the traditional or the object-oriented approach, or both. Chapter 8 presents an overview of technical environments that affect the generation of alternative system solutions. Then, a comprehensive guide to generating and evaluating alternatives is presented, including the reality that a package solution is always a viable option.

### Part 3: System Design Tasks

Chapter 9 covers system design, including both traditional structured design models and object-oriented models. The instructor can include either or both sections. Chapter 10 covers database design—relational, hybrid, and object-oriented databases. Again, the instructor can emphasize any or all approaches. Chapter 11 discusses system inputs and outputs, with particular attention to system controls. Chapter 12 covers human-computer interaction, and we include general principles and concepts of dialog design in addition to using UML diagrams to model the dialog.

### Part 4: Implementation and Support

System implementation is increasingly technology specific, and because of the diverse development environments in the real world, we decided to streamline the discussion of implementation. However, we did include two chapters on important alternative approaches to implementation. Although the text emphasizes iteration and prototyping throughout, we include a comprehensive discussion of rapid application development and component-based development in Chapter 13. Similarly, although package solutions are discussed as viable alternatives throughout, we include a detailed discussion of packages and enterprise resource planning (ERP) in Chapter 14, including specific examples from SAP. Chapter 15 provides an overview of implementation and support that addresses traditional technology and object technology.

### Available Support

The text includes a package of proven supplements for instructors and students. The *Instructor's Resource Kit* includes *Course Presenter*, *Course Test Manager*, *Electronic Instructor's Manual*, and *Figure Files*.

*Course Presenter* includes a PowerPoint lecture presentation in outline form with figures for each chapter. *Course Test Manager* can be used to create paper or LAN-based tests with multiple-choice, fill-in-the-blank, and/or essay questions. The *Electronic Instructor's Manual* includes suggestions and strategies for using the text, test bank questions, answers to review questions, and suggested solutions to chapter exercises and cases. *Figure Files* allow instructors to create their own presentations using figures from the text.

Many instructors like to include software for students to use for exercises and course projects, and this text offers many bundling possibilities. A popular option to use for drawing diagrams is Visio Professional 5.0. Some instructors like to emphasize CASE tools, and Course Technology can bundle several popular CASE tools with the text, including Oracle Designer, Visible Analyst Student Edition, and others. Contact Course Technology for the latest information.

### Credits and Acknowledgments

This project was launched following some initial brainstorming between publisher Kristen Duerr of Course Technology and author John Satzing. We agreed that an analysis and design text required a major commitment from the publisher and from a team of authors to be competitive. It was agreed that no one person or even pair of authors could complete a text that met the objectives—flexible and streamlined, yet comprehensive and deep. Therefore, Course Technology took an active role in assembling a team of authors who shared the vision. The senior editor brought in to manage the project was Jennifer Normandin, who had a major role in bringing the authors together and shaping the direction and final form of the text. This text truly could not have been completed without the commitment of Kristen Duerr and the leadership and energy of Jennifer Normandin. We thank them for everything.

Another essential member of the team is developmental editor Karen Hill of Elm Street Publishing Services. She collected and digested the comments and reactions of reviewers, provided guidance and design for the features and chapter pedagogy, suggested improvements and refinements to the organization and content, read each draft of each chapter from the perspective of the student to help us be consistent and clear, and (to the extent it is humanly possible) edited the chapters to provide a consistent style.

We also want to thank some other key people for their specific contributions—Richard A. Johnson of Southwest Missouri State University for writing Chapter 14 on packages and ERP, William Baker for contributing material on presentation techniques, and Mario Busjra for designing the RMO Logo and Web site. Many other colleagues and friends at SMSU, BYU, the University of New Mexico, and elsewhere contributed to and supported our work in one way or another. Special thanks also to Lavette Teague, Lorne Olfman, and Paul Gray for guidance and inspiration.

Many other people were involved in the production of this text. Amanda Young of Course Technology was always available and helpful. Debbie Masi, the production editor, came through for us to help sort out the subtleties of diagrams that probably appeared rather primitive to the artists. The production team went the extra mile to be true to the diagramming conventions and standards to the extent that we could define them.

Last, but certainly not least, we want to thank the reviewers who worked so hard for us, beginning with an initial proposal and continuing all the way through to the completion of the text. It is not easy to review a work in progress, and we were lucky enough to have reviewers with broad perspectives, in-depth knowledge, and diverse preferences. We listened very carefully, and the text is much better as a result of their input. The reviewers included:

Robert Beatty, *University of Wisconsin, Milwaukee*  
 Paul H. Cheney, *University of Central Florida*  
 Jon D. Clark, *Colorado State University*  
 David Little, *High Point University*  
 Ellen D. Hoadley, *Loyola College in Maryland*  
 Robert Keim, *Arizona State University*  
 Rebecca Koop, *Wright State University*  
 James E. LaBarre, *University of Wisconsin—Eau Claire*  
 George M. Marakas, *Indiana University*  
 Roger McHaney, *Kansas State University*  
 Bruce Neubauer, *Pittsburgh State University*  
 Mary Prescott, *University of South Florida*  
 Robert Saldarini, *Bergen Community College*

All of us involved in the development of this text wish you all the best as you take on the challenge of analysis and design in a changing world.

- John Satzinger
- Bob Jackson
- Steve Burd

## PART 1 | *The Modern Systems Analyst*

<b>CHAPTER 1</b>	<b>The World of the Modern Systems Analyst . . . . .</b>	<b>2</b>
	<i>A Systems Analyst at Rutherford Manufacturing . . . . .</i>	<i>3</i>
	The Analyst as a Business Problem Solver . . . . .	4
	Systems That Solve Business Problems . . . . .	6
	Information Systems . . . . .	6
	Types of Information Systems . . . . .	8
	Required Skills of the Systems Analyst . . . . .	10
	Technical Knowledge and Skills . . . . .	10
	Business Knowledge and Skills . . . . .	11
	People Knowledge and Skills . . . . .	11
	A Few Words about Integrity and Ethics . . . . .	12
	The Environment Surrounding the Analyst . . . . .	12
	Types of Technology Encountered . . . . .	12
	Typical Job Titles and Places of Employment . . . . .	13
	The Analyst's Role in Strategic Planning . . . . .	14
	Special Projects . . . . .	14
	Strategic Planning Processes . . . . .	14
	Information Systems Strategic Planning . . . . .	14
	Enterprise Resource Planning (ERP) . . . . .	15
	Strategic Systems Plan for Rocky Mountain Outfitters . . . . .	16
	Overview of Rocky Mountain Outfitters . . . . .	16
	Organization and Locations . . . . .	17
	Information Systems Department Organization . . . . .	18
	Existing Systems . . . . .	19
	The Information Systems Strategic Plan . . . . .	20
	Two Major New Systems . . . . .	21
	The Analyst as a System Developer (The Heart of the Course) . . . . .	22
	Part 1: The Modern Systems Analyst . . . . .	22
	Part 2: Systems Analysis Tasks . . . . .	23
	Part 3: System Design Tasks . . . . .	24
	Part 4: System Implementation and Support . . . . .	24
	Summary . . . . .	25
	Key Terms . . . . .	25
	Review Questions . . . . .	25
	Thinking Critically . . . . .	26
	Experiential Exercises . . . . .	26
	Case Study . . . . .	27
	<i>Association for Information Technology Professionals Meeting . . . . .</i>	<i>27</i>
	Further Resources . . . . .	27

<b>CHAPTER 2</b>	<b>The Analyst as a Project Manager . . . . .</b>	<b>28</b>
	<i>Blue Sky Family of Mutual Funds: Managing the IRA Project . . . . .</i>	<i>29</i>
	Systems Development Projects and the SDLC . . . . .	30
	The Systems Development Life Cycle . . . . .	30
	The First Four Phases of the Systems Development Life Cycle . . . . .	31
	Scheduling of Project Phases . . . . .	35
	The Project Team . . . . .	36
	Project Management . . . . .	38
	Planning and Organizing . . . . .	39
	Directing—Executing, Monitoring, and Controlling . . . . .	39
	Project Initiation . . . . .	40
	Project Initiation for Rocky Mountain Outfitters . . . . .	41
	The Project Planning Phase . . . . .	41
	Defining the Problem . . . . .	42
	Confirming Project Feasibility . . . . .	45
	Developing the Project Schedule . . . . .	52
	Staffing the Project . . . . .	56
	Launching the Project . . . . .	57
	Summary . . . . .	57
	Key Terms . . . . .	57
	Review Questions . . . . .	58
	Thinking Critically . . . . .	58
	Experiential Exercises . . . . .	59
	Case Study . . . . .	60
	<i>Custom Load Trucking . . . . .</i>	<i>60</i>
	Further Resources . . . . .	60
<b>CHAPTER 3</b>	<b>Approaches to System Development . . . . .</b>	<b>62</b>
	<i>Development Approaches at Ajax Corporation, Consolidated Concepts, and Pinnacle Manufacturing . . . . .</i>	<i>63</i>
	Methodologies, Models, Tools, and Techniques . . . . .	64
	Methodologies . . . . .	64
	Models . . . . .	64
	Tools . . . . .	65
	Techniques . . . . .	66
	Three Approaches to System Development . . . . .	67
	The Structured Approach . . . . .	67
	The Information Engineering Approach . . . . .	72
	The Object-Oriented Approach . . . . .	73
	System Development Life Cycle Variations . . . . .	75
	Variations of the Phases (Waterfall Models) . . . . .	76
	Variations Based on Iteration . . . . .	77
	Variations Based on an Emphasis on People . . . . .	78
	Variations Based on Speed of Development . . . . .	79
	Computer-Aided System Engineering (CASE) Tools . . . . .	80
	Analysis and Design Phases in More Detail . . . . .	83
	The Analysis Phase . . . . .	83
	The Design Phase . . . . .	87
	Summary . . . . .	90
	Key Terms . . . . .	91
	Review Questions . . . . .	91
	Thinking Critically . . . . .	91
	Experiential Exercises . . . . .	92
	Case Study . . . . .	92
	<i>Factory System Development Project . . . . .</i>	<i>92</i>
	Further Resources . . . . .	93

## PART 2 | Systems Analysis Tasks

<b>CHAPTER 4</b>	<b>Investigating System Requirements . . . . .</b>	<b>96</b>
	<i>Providing Customer Service—and Gasoline—at Convenience America. . . . .</i>	<i>97</i>
	Functional and Technical Requirements . . . . .	98
	Stakeholders—the Source of System Requirements . . . . .	99
	User Stakeholders. . . . .	100
	Client Stakeholders. . . . .	101
	Technical Stakeholders. . . . .	102
	The Stakeholders for Rocky Mountain Outfitters. . . . .	102
	Identifying System Requirements . . . . .	104
	Distribute and Collect Questionnaires . . . . .	106
	Review Existing Reports, Forms, and Procedure Descriptions. . . . .	107
	Conduct Interviews and Discussions with Users . . . . .	108
	Observe Business Processes and Workflows . . . . .	112
	Building Prototypes . . . . .	113
	Conduct Joint Application Design Sessions . . . . .	113
	Structured Walkthroughs . . . . .	116
	What and When . . . . .	117
	Who . . . . .	117
	How . . . . .	118
	Business Process Reengineering. . . . .	119
	Summary . . . . .	120
	Key Terms. . . . .	121
	Review Questions . . . . .	122
	Thinking Critically. . . . .	122
	Experiential Exercises . . . . .	122
	Case Study . . . . .	123
	<i>John and Jacob, Inc. On-Line Trading System. . . . .</i>	<i>123</i>
	Further Resources . . . . .	124
 <b>CHAPTER 5</b>	 <b>Modeling System Requirements: Events and Things. . . . .</b>	 <b>126</b>
	<i>Walters on Wheels: Computerized Delivery Tracking. . . . .</i>	<i>127</i>
	Models and Modeling . . . . .	128
	The Purpose of Models. . . . .	128
	Types of Models. . . . .	130
	Overview of Models used in Analysis and Design . . . . .	132
	Events and System Requirements . . . . .	133
	The Background of the Event Concept . . . . .	134
	Types of Events . . . . .	135
	Identifying Events. . . . .	136
	Events in the Rocky Mountain Outfitters Case . . . . .	139
	Looking at Each Event . . . . .	140
	Things and System Requirements. . . . .	142
	Types of Things . . . . .	143
	Relationships among Things. . . . .	144
	Attributes of Things. . . . .	145
	Data Entities and Objects . . . . .	146
	The Entity-Relationship Diagram . . . . .	147
	Examples of ERD Notation . . . . .	147
	The Rocky Mountain Outfitters Case ERD . . . . .	151
	The Class Diagram . . . . .	153
	More Complex Issues about Classes of Objects . . . . .	153
	Examples of Class Diagram Notation . . . . .	154
	The Rocky Mountain Outfitters Case Class Diagram. . . . .	157

**CHAPTER 6**

Where You Are Headed . . . . .	158
Summary . . . . .	160
Key Terms . . . . .	161
Review Questions . . . . .	161
Thinking Critically . . . . .	162
Experiential Exercises . . . . .	163
Case Studies . . . . .	164
<i>The Spring Breaks 'R' Us Travel Service Booking System</i> . . . . .	164
<i>The Real Estate Multiple Listing Service System</i> . . . . .	164
<i>The State Patrol Ticket Processing System</i> . . . . .	165
Further Resources . . . . .	166
<b>The Traditional Approach to Requirements . . . . .</b>	<b>168</b>
<i>Performance Auto Electric: Following the Data Flow</i> . . . . .	169
Traditional and Object-Oriented Views of Activities . . . . .	170
Data Flow Diagrams . . . . .	170
Data Flow Diagrams and Levels of Abstraction. . . . .	172
Context Diagrams . . . . .	173
DFD Fragments . . . . .	174
The Event-Partitioned System Model . . . . .	175
Decomposing Processes to See Detail of One Activity . . . . .	182
Physical and Logical DFDs. . . . .	184
Evaluating DFD Quality . . . . .	186
Documenting DFD Components. . . . .	190
Process Descriptions . . . . .	190
Data Flow Definitions . . . . .	195
Data Element Definitions . . . . .	196
Data Store Definitions . . . . .	197
DFD Summary . . . . .	197
Information Engineering Models. . . . .	198
The IE System Development Life Cycle . . . . .	198
IE and Structured Development Compared . . . . .	199
Process Decomposition and Dependency Models . . . . .	200
Considering Locations and Communication through Networks. . . . .	204
Workflow Modeling . . . . .	208
Summary . . . . .	211
Key Terms. . . . .	211
Review Questions . . . . .	212
Thinking Critically. . . . .	212
Experiential Exercises . . . . .	213
Case Studies . . . . .	213
<i>The Real Estate Multiple Listing Service System</i> . . . . .	213
<i>The State Patrol Ticket Processing System</i> . . . . .	213
<i>The Reliable Pharmaceutical Service</i> . . . . .	213
Further Resources . . . . .	215

**CHAPTER 7**

<b>The Object-Oriented Approach to Requirements. . . . .</b>	<b>216</b>
<i>Northwoods Kayaks: Identifying the Methods within the Objects</i> . . . . .	217
The Unified Modeling Language and the Object Management Group. . . . .	218
Object-Oriented Requirements . . . . .	218
The Class Diagram . . . . .	220
The System Activities: An Object-Oriented Use Case/Scenario View. . . . .	221
Use Cases and Actors . . . . .	221
Scenarios . . . . .	221
The Use Case Diagram . . . . .	223

Object Interactions: Collaboration and Sequence Diagrams . . . . .	227
Sequence Diagrams . . . . .	228
Collaboration Diagrams . . . . .	236
Object Behavior: States, State Transitions, and Statechart Diagrams . . . . .	238
Object States . . . . .	239
Object Transitions . . . . .	241
Messages, Transitions, and Actions . . . . .	243
Statechart Diagrams . . . . .	244
Concurrent Behavior . . . . .	247
Statechart Development . . . . .	250
Summary . . . . .	252
Key Terms . . . . .	253
Review Questions . . . . .	253
Thinking Critically . . . . .	254
Experiential Exercises . . . . .	256
Case Studies . . . . .	256
<i>The Real Estate Multiple Listing Service System</i> . . . . .	256
<i>The State Patrol Ticket Processing System</i> . . . . .	256
<i>The Downtown Videos Rental System</i> . . . . .	257
Further Resources . . . . .	257
 <b>CHAPTER 8</b>	
<b>Environments, Alternatives, and Decisions . . . . .</b>	<b>258</b>
<i>Tropic Fish Tales: Netting the Right System</i> . . . . .	259
Assessing the Target Processing Environment . . . . .	260
Centralized Systems . . . . .	261
Distributed Computing . . . . .	263
The Internet and Intranets . . . . .	266
Development and System Software Environments . . . . .	269
The Environment at Rocky Mountain Outfitters . . . . .	270
Deciding on Scope and Level of Automation . . . . .	273
Determining Scope . . . . .	273
Defining the Level of Automation . . . . .	275
Selecting Alternatives . . . . .	278
Generating Alternatives for Implementation . . . . .	282
Facilities Management . . . . .	283
Packaged and Turnkey Software Systems . . . . .	283
Custom Software Development . . . . .	284
In-House Development . . . . .	285
Choosing an Alternative for Implementation . . . . .	285
Identifying Criteria for Selection . . . . .	286
Making the Selection . . . . .	289
Considering Outsourced Solutions . . . . .	290
Generating a Request for Proposal . . . . .	290
Benchmarking and Choosing a Vendor . . . . .	291
Presenting the Results and Making the Decisions . . . . .	292
Summary . . . . .	293
Key Terms . . . . .	293
Review Questions . . . . .	294
Thinking Critically . . . . .	294
Experiential Exercises . . . . .	295
Case Study . . . . .	296
<i>Tropic Fish Tales' RFPs</i> . . . . .	296
Further Resources . . . . .	296

**PART 3 | System Design Tasks**

<b>CHAPTER 9</b>	<b>Moving to Design</b>	<b>298</b>
	<i>New Capital Bank: Object-Oriented Development of a Payroll System</i>	299
	Understanding the Elements of Design	300
	Inputs: Moving from Analysis to Design	300
	Major Components and Levels of Design	300
	Outputs: Structured Models and Object-Oriented Models	303
	Designing the Application Architecture: The Structured Approach	305
	The Automation System Boundary	306
	The System Flow Chart	308
	The Structure Chart	311
	Module Algorithm Design: Pseudocode	323
	Integration of the Structured Application Design with User-Interface Design, Database Design, and Network Design	325
	Designing the Application Architecture: The Object-Oriented Approach	325
	Object-Oriented Programs	326
	Object-Oriented Models	329
	Package Diagrams	329
	Design Class Diagrams	331
	Design Class Diagram Development	335
	Method Development and Pseudocode	339
	Inheritance, Overriding, and Polymorphism	340
	Integrating the Object-oriented Application Design with User Interface Design, Database Design, and Network Design	340
	Coordinating the Project	341
	Coordinating Project Teams	342
	Coordinating Information	342
	Summary	344
	Key Terms	344
	Review Questions	344
	Thinking Critically	345
	Experiential Exercises	351
	Case Studies	351
	<i>The Real Estate Multiple Listing Service System (Structured)</i>	351
	<i>The Reliable Pharmaceutical Service System</i>	351
	<i>The Real Estate Multiple Listing Service System (Object-Oriented)</i>	351
	<i>The DownTown Video Rental System</i>	351
	Further Resources	352
<b>CHAPTER 10</b>	<b>Designing Databases</b>	<b>354</b>
	<i>Nationwide Books: Designing a New Database</i>	355
	Databases and Database Management Systems	356
	Database Models	357
	Relational Databases	358
	Designing Relational Databases	360
	Representing Entities	361
	Representing Relationships	363
	Enforcing Referential Integrity	365
	Evaluating Schema Quality	365
	Object-Oriented Databases	372
	Designing Object Databases	372
	Representing Classes	372
	Representing Relationships	373
	Hybrid Object-Relational Database Design	378
	Classes and Attributes	379
	Relationships	381

	Data Types . . . . .	382
	Relational DBMS Data Types . . . . .	383
	Object DBMS Data Types . . . . .	383
	Distributed Databases . . . . .	384
	Distributed Database Architectures . . . . .	385
	The RMO Distributed Database Architecture . . . . .	390
	Summary . . . . .	393
	Key Terms . . . . .	393
	Review Questions . . . . .	394
	Thinking Critically . . . . .	395
	Experiential Exercises . . . . .	395
	Case Studies . . . . .	396
	<i>The Real Estate Multiple Listing Service System</i> . . . . .	396
	<i>The State Patrol Ticket Processing System</i> . . . . .	396
	<i>The Reliable Pharmaceutical Service</i> . . . . .	396
	Further Resources . . . . .	396
<b>CHAPTER 11</b>	<b>Designing Inputs, Outputs, and Controls . . . . .</b>	<b>398</b>
	<i>Customized Cars Clearinghouse: Moving into Electronic Commerce</i> . . . . .	399
	Integrity Controls . . . . .	400
	Objectives of Integrity Controls . . . . .	400
	System Access Controls . . . . .	401
	Input Integrity Controls . . . . .	404
	Output Integrity Controls . . . . .	405
	Design of System Inputs . . . . .	406
	Identifying Devices and Mechanisms . . . . .	407
	Developing the List of Inputs and Data Requirements of Each . . . . .	409
	Designing and Prototyping Input Forms . . . . .	417
	Design of System Outputs . . . . .	418
	Determining the Type of Output . . . . .	419
	Making a List of Specific Reports Based on the Application Design . . . . .	424
	Designing and Prototyping Reports . . . . .	426
	Summary . . . . .	427
	Key Terms . . . . .	427
	Review Questions . . . . .	428
	Thinking Critically . . . . .	428
	Experiential Exercises . . . . .	429
	Case Study . . . . .	429
	<i>All-Shop Superstores</i> . . . . .	429
	Further Resources . . . . .	431
<b>CHAPTER 12</b>	<b>Human-Computer Interaction . . . . .</b>	<b>432</b>
	<i>Interface Design at Aviation Electronics</i> . . . . .	433
	The User Interface . . . . .	434
	Physical Aspects of the User Interface . . . . .	434
	Perceptual Aspects of the User Interface . . . . .	435
	Conceptual Aspects of the User Interface . . . . .	435
	User-Centered Design . . . . .	435
	Human-Computer Interaction as a Field of Study . . . . .	436
	Metaphors for Human-Computer Interaction . . . . .	438
	Interface Design Guidelines . . . . .	444
	Visibility and Affordance . . . . .	444
	Eight Golden Rules . . . . .	444
	Documenting Dialog Designs . . . . .	448

Events, Subsystems, and Menu Hierarchy . . . . .	448
Dialogs and Storyboards . . . . .	449
Dialog Documentation with UML Diagrams . . . . .	452
Guidelines for Designing Windows Forms . . . . .	455
Form Layout and Formatting . . . . .	456
Data Keying and Entry for Standard Windows . . . . .	458
Navigation and Support Controls . . . . .	459
Data Keying and Entry for the Browser Interface . . . . .	459
Help Support . . . . .	459
Dialog Design for Rocky Mountain Outfitters . . . . .	459
Dialog Design for Web Sites . . . . .	466
Summary . . . . .	468
Key Terms . . . . .	469
Review Questions . . . . .	469
Thinking Critically . . . . .	470
Experiential Exercises . . . . .	470
Case Studies . . . . .	471
<i>The 21st Century University System</i> . . . . .	471
<i>The Downtown Videos Rental System</i> . . . . .	472
<i>The Waiters on Wheels System</i> . . . . .	472
<i>The State Patrol Ticket Processing System</i> . . . . .	473
<i>Another Alternative for Rocky Mountain Outfitters</i> . . . . .	473
Further Resources . . . . .	473

## **PART 4 | Implementation and Support**

### **CHAPTER 13**

#### **Rapid Application Development and Component**

<b>Based Development . . . . .</b>	<b>476</b>
<i>CDs Direct 2U: Speeding Development of a Customer Support System</i> . . . . .	477
Rapid Application Development . . . . .	478
Reasons for Slow Development . . . . .	478
What Is RAD? . . . . .	480
Rapid Development Approaches . . . . .	480
The Prototyping Approach to Development . . . . .	482
The Spiral Approach to Development . . . . .	487
Rapid Development Techniques . . . . .	490
Risk Management . . . . .	490
Joint Application Design . . . . .	494
Tool-Based Development . . . . .	495
Software Reuse . . . . .	497
Object Frameworks . . . . .	498
A Sample Object Framework . . . . .	498
Object Framework Types . . . . .	500
The Impact of Object Frameworks on Design and Implementation Tasks . . . . .	501
Pitfalls to Consider for Object Frameworks . . . . .	502
Components . . . . .	503
Component Standards and Infrastructure . . . . .	504
Components and the Development Life Cycle . . . . .	506
Summary . . . . .	509
Key Terms . . . . .	509
Review Questions . . . . .	510
Thinking Critically . . . . .	510
Experiential Exercises . . . . .	512
Case Study . . . . .	512
Further Resources . . . . .	515