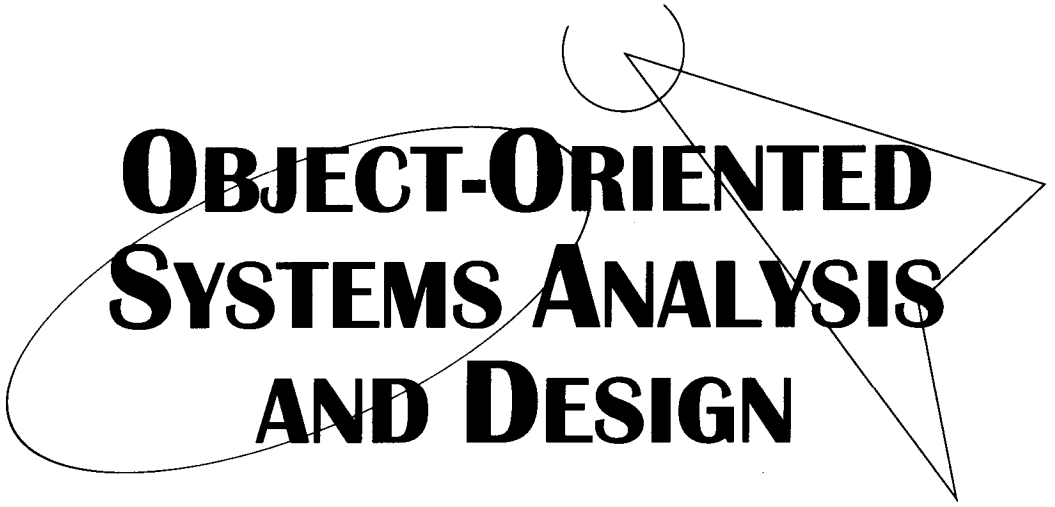
Prentice Hall Series in Information Management

OBJECT-ORIENTED SYSTEMS ANALYSIS AND DESIGN

RONALD J. NORMAN



OBJECT-ORIENTED SYSTEMS ANALYSIS AND DESIGN

Ronald J. Norman

San Diego State University

Prentice Hall · Upper Saddle River, NJ 07458

Library of Congress Cataloging-in-Publication Data

Norman, Ronald J.
Object-oriented systems analysis and design / Ronald J. Norman.
p. cm. — (Prentice Hall series in information management)
Includes bibliographical references and index.
ISBN 0-13-122946-X
1. Object-oriented programming (Computer science) 2. System analysis. 3. System design. I. Title. II. Series.
(William R. King, series editor)
QA76.64.N67 1996
005.1'2—dc20 95-36248
CIP

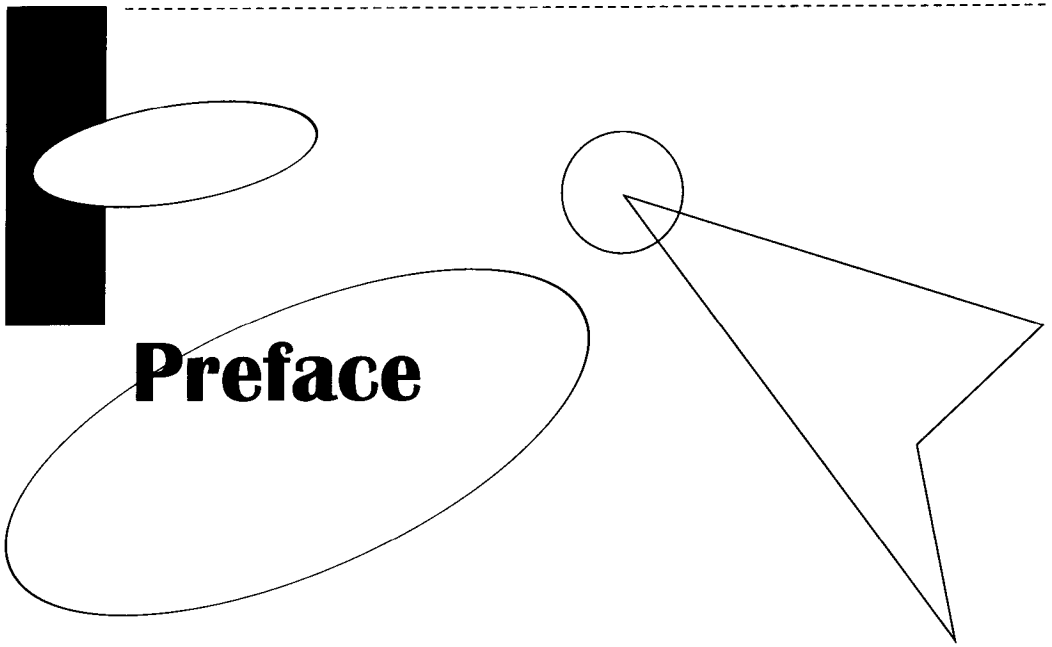
Editor-in-Chief: *Rich Wohl*
Managing Editor: *Nicholas Radhuber*
Project Manager: *Susan Rifkin*
Manufacturing Buyer: *Paul Smolenski*
Editorial Assistant: *Jane Avery*
Cover Designer: *Karen Salzbach*
Interior Design: *Lorraine Castellano*

Printed in the United States of America

10 9 8 7 6 5 4 3 2 1

ISBN: 0-13-122946-X

PRENTICE-HALL INTERNATIONAL (UK) LIMITED, *London*
PRENTICE-HALL OF AUSTRALIA PTY. LIMITED, *Sydney*
PRENTICE-HALL CANADA INC., *Toronto*
PRENTICE-HALL HISPANOAMERICANA, S A , *Mexico*
PRENTICE-HALL OF INDIA PRIVATE LIMITED, *New Delhi*
PRENTICE-HALL OF JAPAN, INC., *Tokyo*
SIMON & SCHUSTER ASIA PTE. LTD., *Singapore*
EDITORA PRENTICE-HALL DO BRASIL, LTDA., *Rio de Janeiro*



Preface

Information systems. They are everywhere. But you can't touch them, see them, or hold them in the same sense that you can touch a statue, see a sunset, or hold a hamburger in your hand. For the most part, information systems are invisible except for the human-computer interface portion such as a keyboard, joystick, monitor, mouse, or printer.

Business and government rely heavily on information systems to support their missions and goals. In fact it is almost impossible to conduct substantial business these days without "computers." Many people use the word "computers" as a surrogate for the information system on which they are relying. Keeping track of names and addresses, paying bills, playing games, creating term papers, and doing income tax returns are just a few of the information systems for which people use their computer.

The purpose of this book is to be an introductory resource for systems analysis, design, and implementation of information systems. Systems analysis, design, and implementation are, in the broadest sense, the processes used by professional men and women to create or maintain information systems.

THE INTENDED AUDIENCE FOR THIS BOOK

Object-Oriented Systems Analysis and Design is intended to be used in a single-term course or a two-term sequence of courses (e.g., semester or quarter) in information systems development. Four-year universities and colleges, community or junior colleges, and trade-technical/vocational schools that have one or more courses devoted to information systems development or software engineering are perhaps the best candidates for its use. Master's programs that also have one or more courses in

information systems development or software engineering would also benefit from this book although the writing style is admittedly geared primarily to undergraduate students.

I recommend that students have basic computer literacy skills prior to taking such a course. In addition, experience with a programming language such as COBOL, Visual (anything), or C/C++ would contribute to a better understanding and appreciation of the information systems development process.

WHY I WROTE THIS BOOK

My sole goal for writing this book is to provide my own students with a systems analysis, design, and implementation resource textbook that would effectively communicate the essence of these activities in the most meaningful way possible. The anonymous feedback that I have received on earlier drafts of this book from my students has encouraged me to pursue its formal publication so that other students may benefit from it as well.

The current generation of students taking computer-related courses has come to expect more than just theory in the classroom because many students have grown up with computers at school and in their homes. In almost every instance, they have had plenty of "hands- on" experiences with the computer and video games. Computer theory without practical hands-on application is a little like peanut butter without jelly or graham crackers without milk—some things are just made for each other.

I have used many good systems analysis and design textbooks over the years, none of which were perfect. This one isn't either. However, in this book, I have tried to address some of the perceived issues that my students consistently bring up as shortcomings in other textbooks. Chief among their complaints are books that:

- are intended to be introductory yet assume a high degree of related background knowledge.
- are too conceptual or theoretical.
- minimize concepts and theories and substitute anecdotal experiences.
- use examples that students have a difficult time relating to.
- claim to be presenting the best systems analysis and design methodology with the implication that everybody should be using it.

A FEW WORDS ABOUT INFORMATION SYSTEMS JARGON

Jargon is a way of life. Understanding of and use of jargon is equated with knowledge of the subject area. Misuse of jargon can distinguish the fake from the real. Doctors, attorneys, accountants, engineers, mechanics, and surfers all have their jargon. Even young people have their own jargon.

Information systems and computer technology have jargon also. Lots of it! The accepted name of the field itself has changed over the years. A few of the accepted (at

least at one time) names for this field are: data processing, management information systems, business information systems, information systems, software engineering, systems engineering, and information systems engineering. For purposes of this book these terms are synonymous. Information systems is the preferred one for use throughout this book. Is this the best term? Who knows! My goal is to be consistent throughout the book.

Systems analysis and design is known by a few other terms also, such as software engineering, systems engineering, information systems development, and information systems engineering. You probably noticed that some of these terms were included as accepted terms for this field of study. Not surprising for a field of study as young (about 50 years) and as diverse as this one. I have selected systems analysis and design as the preferred term for use in this book. On rare occasion, due to sentence structure, I may use one of the other terms.

The titles assigned to the people who do systems analysis and design are also diverse. Titles such as systems analyst, analyst/programmer, software engineer, and systems engineer are quite common. This book will use systems analyst as the preferred term.

Finally, there is the ongoing discussion of the most appropriate term used to classify the people who use information systems, such as knowledge worker, user, and customer. Knowledge worker appears to have a very small following these days. Perhaps the oldest term is the word "user." This term conjures up all kinds of thoughts. So, the term "customer" has become popular in recent years. This term has a positive connotation that is good. However, students tend to think of themselves as customers—customers of banks, restaurants, stores, super markets, and even the university. They, in turn, may even wait on customers in their role as employees of a bank, restaurant, store, and so on. Because of this potential confusion, I have chosen the term "user" for this book. When used, the term can refer to a single person or a group of people.

HOW TO USE THIS BOOK

Object-Oriented Systems Analysis and Design is divided into three parts. My personal experience with several other systems analysis and design books suggests that instructors can readily omit sections of chapters or entire chapters and modules to suit their own teaching style and content preferences. Rearranging of chapter and module order to meet instructional goals is also an option. Instructors can do the same with this book.

Part I, Systems Analysis and Conceptual Design, is made up of seven chapters. Chapter 1 presents an introduction to systems analysis and design that introduces the reader to the systems development process, most often referred to as the systems development life cycle. Foundational information related to the systems development process is covered along with additional foundational information related to systems analysts, the people who perform the systems development process. Chapter 2 introduces the reader to feasibility analysis and the requirements determination activities

within systems analysis. Chapter 3 presents an object-oriented methodology and model in its entirety. This shows the reader the "big picture" before moving on to necessary details. Chapters 4-7 present the details of the object-oriented methodology and model.

Part II, Physical Design and Implementation, consists of six chapters. Chapter 8 presents an overview of the design portion of systems analysis and design. Chapter 9 focuses on output design. Chapters 10 and 11 address input design and file and database design. Chapter 12 presents software construction and testing concepts followed by presentation of information systems implementation from both a technical and organizational behavior perspective in Chapter 13.

Part III, Miscellaneous Systems Analysis and Design Topics, presents eight supplementary and complementary topics that either enable systems analysis and design or are additional aspects of systems analysis and design. For example, Module A, Information Systems Planning, is an additional aspect of systems analysis and design, while Module G, Communication and Electronic Meetings, presents these topics which contribute in an enabling way to systems analysis and design. Each of the other six modules—Prototyping, CASE, Software Process Improvement, The Systems Development Challenge, Project Management, and Business Process Reengineering—presents topics that are important to the systems analysis and design process. These eight modules are intended to be used at the instructor's discretion and at the point in time during the course that best fits his or her course goals.

SUPPLEMENTS

As you, the instructor, knows one of the significant differences between textbooks and trade books is the supplemental materials that accompany the former. *Object-Oriented Systems Analysis and Design* has four supplements including:

- An instructor's guide (with sample object models)
- A test item file
- Classroom presentation materials
- Object Modeling software
- Electronic communication with the author

The instructor's guide contains suggestions for presenting each of the chapters and modules in the book along with the answers to the end of chapter questions. The test item file contains objective (e.g., true and false/multiple-choice) and subjective (e.g., short answers required) questions and answers. The classroom materials include overhead transparencies of most figures and other important chapter concepts that may not have figures in the book. In addition, a full-color electronic version of the classroom materials in Microsoft PowerPoint (IBM) format is available by contacting the publisher.

Another supplement is object-oriented modeling software. Object International, Inc. (OI) graciously makes its object-oriented modeling whiteboard software package called Playground™ available to both students and instructors. Playground™ is an

easy-to-use tool for building object models. It uses the same modeling notation presented in this book.

The "classroom and personal study only" version is available at no cost to a student or an instructor. Playground™ requires the Microsoft Windows operating system. Due to constant upgrades from both Microsoft and OI, specific system requirements are not listed here. Specific requirements are available at the time of downloading the Playground™ software directly from OI's world-wide web home page which is http://www.oi.com/business/object/. The file that is downloaded is a self-extracting file and at the time of this book's publication took between 1 and 2 megabytes of disk space to download.

Again, each student is allowed his or her own personal copy of Playground™. The instructor may choose to download Playground™ and then make copies available to students in a manner he or she chooses. The only difference between the "classroom and personal study only" version of Playground™ and the registered (fee) version is the "classroom and personal study only" watermark that appears at the bottom of the display screen and on printed output.

The final supplement is one that is not generally found in textbooks—electronic addresses to communicate directly with the author. Recognizing that change is a constant, these addresses are correct as of the publication date for this book; however there is no guarantee that these addresses will be correct in the future. If you, the instructor who adopts this text or is considering its adoption, send me electronic communication, I will do my best to reply back to you as soon as possible. I welcome your comments and questions, feedback and suggestions, as well as corrections for a future revision of the book.

My World Wide Web Home Page Universal Resource Locator (URL) address is: http://rohan.sdsu.edu/dept/cbaweb/IDS/RNorman.html (case sensitive)

My electronic mail address is: ronald.norman@sdsu.edu (not case sensitive)

San Diego State University's (SDSU) Home Page URL is: http://www.sdsu.edu/

SDSU's College of Business Administration (CBA) Home Page URL is: http://rohan.sdsu.edu/dept/cbaweb/

SDSU's CBA's Information and Decision Systems Department Home page URL is: http://rohan.sdsu/dept/cbaweb/IDS/index.html (case sensitive)

ACKNOWLEDGMENTS

I am indebted to many people for their contributions in the development of this book and would like to acknowledge them here.

I am grateful to many students over several semesters and years in my undergraduate and graduate systems analysis and design courses who used preliminary versions of this book. I appreciate not only their helpful suggestions but also their patience and support of this work. Two students in particular, Ron Dashwood and Vicky Smith, combined to create the first basic object-oriented primer for my students to use in 1992.

I am grateful to the following reviewers of preliminary versions or portions of this book. My only regret is that I could not include all of their suggestions for a variety of reasons:

Terrence K. Babbitt, Former CIO Foodmaker Corporation; John Borton, University of Southern Colorado; Peter Coad, Object International, Inc.; Gail F. Corbitt, California State University, Chico; Bill Hardgrave, University of Arkansas; Joseph Morrell, Metro State College of Denver; Stevan Mrdalj, Eastern Michigan University; Richard G. Ramirez, Iowa State University; Ramesh Subramanian, University of Alaska, Anchorage; Thaddeus Usowicz, San Francisco State University; and David C. Yen, Miami of Ohio University.

I am grateful for the help of Germaine Kang, a former graduate student, who assisted with some of the object-oriented database material in the file and database design chapter of the book. I am also grateful for the help of Jin-Jue Tsai, a former graduate student, for his diligent and persistent search for object technology research materials and resources.

Thanks to Elliot Chikofsky, Northeastern University and DMR Group, Inc., for his friendship and long-term support and assistance with my CASE and software development research. Many professional opportunities became possible because of his involvement.

Thanks also to Dr. Stephen Robbins for mentoring me during this writing process. His many years of experience as an author have been a source of inspiration to me when the going got tough.

Thanks to Jeff Whitten and Lonnie Bentley, both of Purdue University, for their work in systems analysis and design over the years. It is a privilege to know and work with them.

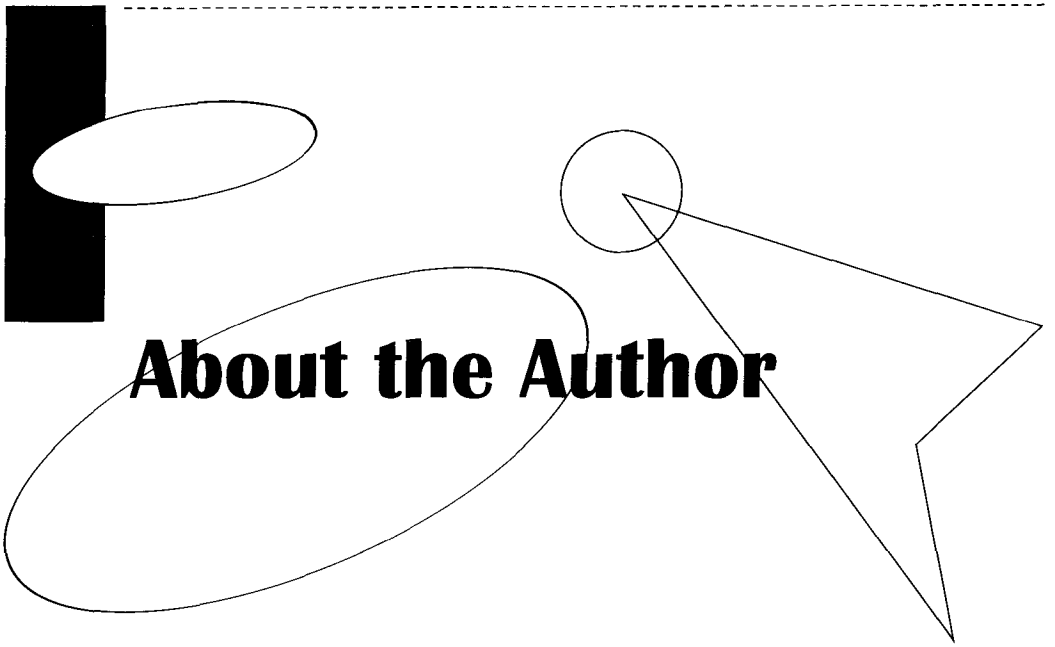
I would like to thank Dr. Gail Corbitt, Associate Professor at California State University, Chico, for her support and contributions during the years we worked together at San Diego State University. She crystallized a significant part of the organizational behavior portion of the implementation chapter for me.

My thanks also to the Prentice Hall staff—Rich Wohl, P.J. Boardman, Susan Rifkin, Paul Becker, and others—that assisted with the publication of this book.

I apologize to any contributing person I may have overlooked at this time.

Finally, I hope that you will derive value from this book. Any shortcomings, errors, omissions, or inconsistencies attributed to the book are the sole responsibility of me. Best wishes for your career.

Ronald J. Norman
1995



About the Author

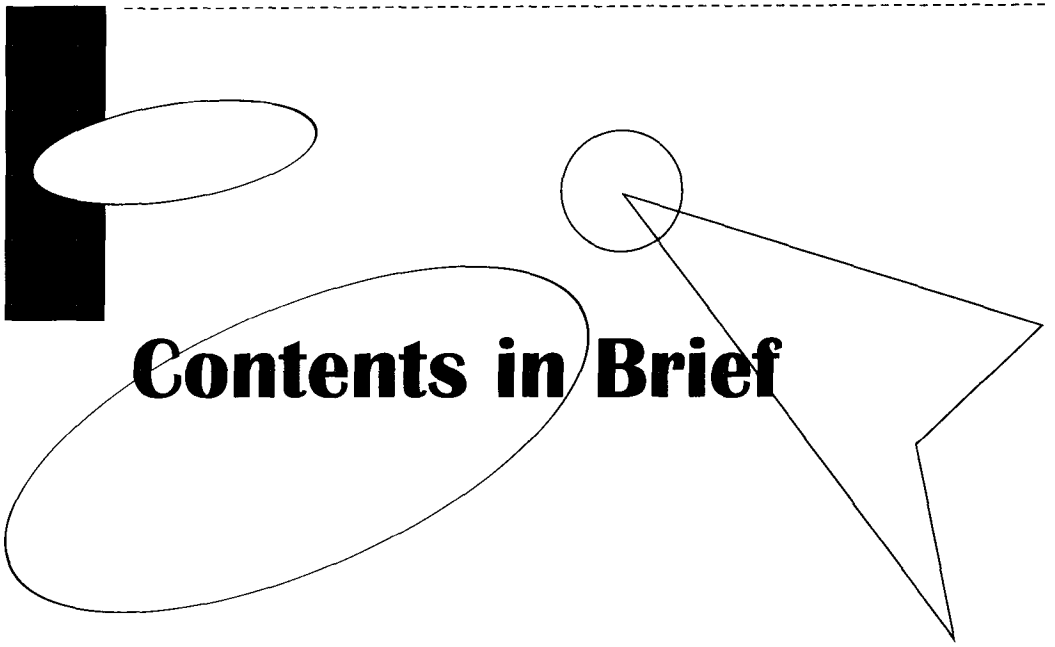
Ronald J. Norman is currently a Professor of Information and Decision Systems at San Diego State University. He teaches executive, graduate, and undergraduate level courses in information systems management, software engineering/systems analysis and design, and information as an organizational resource. He received a Ph.D. in Management Information Systems and Organizational Behavior from the University of Arizona in 1987. He is also a Certified Computer Professional (CCP) and has over 25 years of industry MIS experience including software development, consulting, and management.

Dr. Norman's research interests are object-oriented systems development, organizational behavior related to the introduction of technology, and CASE technology. He has served as facilities chairman for the International Conference on Information Systems and was the program committee chairman for CASE '88, the Second International Workshop on Computer-Aided Software Engineering. He was the General Chairman for CASE '90, co-sponsored by the IEEE Computer Society, a mini-track coordinator on IPSE/Software Development Environments, and a mini-track coordinator on CASE technology for the Hawaiian International Conference on Systems Sciences (HICSS-24) in 1991 and repeated the CASE mini-track at HICSS-25 in January 1992. He was guest editor for the March 1992 issue of IEEE Software focusing on integrated CASE and was also guest editor for the April 1992 issue of the Communications of the ACM on a similar topic. He served as program co-chair for CASE '95 in Toronto, Canada and was on the program committee for the European CASE conference in 1995 and again in 1996. He is also co-editor of a special CASE theme issue of the Automated Software Engineering Journal to appear in 1996.

Dr. Norman has published or co-published over two dozen articles which have

appeared in journals including the Communications of the ACM, IEEE Software, International Journal of Software Engineering and Knowledge Engineering, CIO JOURNAL, Information & Software Technology, Journal of Computer Information systems, and Journal of Systems Management.

Dr. Norman's industry work/consulting engagements include several international, national, and local organizations, such as Avco Finance, Unisys Corporation, Tymshare, Inc., NCR Corporation, BASF, US Navy, American Management Systems, Performance Research Corporation, Science Applications Inc, University of California Medical Center, San Diego Police Department, Solar Turbines International, General Dynamics Corporation, National Liberty Corporation, and INJOY.



Contents in Brief

	PREFACE	*xix*
Part I	**Systems Analysis and Conceptual Design**	
	1. INTRODUCTION	*1*
	2. FEASIBILITY ANALYSIS AND REQUIREMENTS DETERMINATION	*29*
	3. AN OBJECT-ORIENTED METHODOLOGY AND MODEL	*55*
	4. OBJECTS AND CLASSES	*86*
	5. OBJECT RESPONSIBILITIES: ATTRIBUTES	*102*
	6. OBJECT RESPONSIBILITIES: CLASS AND OBJECT CONNECTIONS	*116*
	7. OBJECT RESPONSIBILITIES: SERVICES AND SCENARIOS	*149*
Part II	**Physical Design & Implementation**	
	8. SYSTEMS DESIGN	*184*
	9. OUTPUT DESIGN	*206*
	10. INPUT DESIGN	*229*

11. FILE AND DATABASE DESIGN *249*

12. SOFTWARE CONSTRUCTION AND TESTING *286*

13. IMPLEMENTATION *310*

Part III Modules—Miscellaneous Systems Analysis and Design Topics

A. INFORMATION SYSTEMS PLANNING *329*

B. PROTOTYPING *337*

C. COMPUTER-AIDED SOFTWARE ENGINEERING (CASE) *346*

D. SOFTWARE PROCESS IMPROVEMENT *353*

E. THE SYSTEMS DEVELOPMENT CHALLENGE *361*

F. PROJECT MANAGEMENT *373*

G. COMMUNICATION AND ELECTRONIC MEETINGS *387*

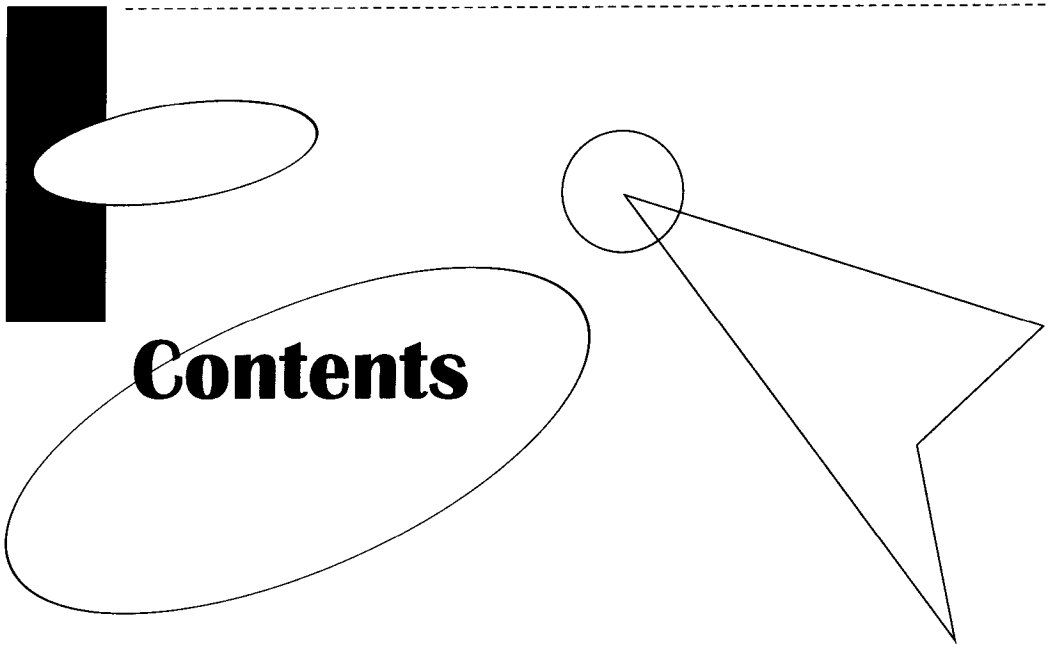
H. BUSINESS PROCESS REENGINEERING *396*

GLOSSARY *401*

RECOMMENDED READING TO GET STARTED—AUGUST 1995 *410*

BIBLIOGRAPHY—OBJECT-ORIENTED TECHNOLOGY *411*

INDEX *421*



Contents

PREFACE ***xix***

Part I Systems Analysis and Conceptual Design

1. INTRODUCTION ***1***

Chapter Objectives 1
Systems Analysis and Design Has Many Other Names 2
What is a System? 4
What is an Information System? 5
What is an Automated Information System? 7
What are the Basic Characteristics of an Information System? 8
What is Systems Analysis and Design? 10
What Makes Systems Analysis and Design such a Difficult Human Endeavor? 10
Stakeholders of an Information System 12
Systems Analysis and Design as a Career 13
What does a Systems Analyst Do? 14
What is a Systems Analyst Responsible For? 14
Systems Analysis and Design Skills and Activities 15
General Model of Systems Analysis and Design 17

The Detailed Activities of Analysis and Design 18
Systems Analysis and Design Projects 21
Where do Information Systems Analysis and Design Projects Come From? 22
Information Systems Requirements Specification 23
Information Systems Life Cycle and Information Systems Development Life Cycle (SDLC) 24
Principles to Guide Information Systems Analysis and Design 26
Summary 27
Questions 27
References 28

2. FEASIBILITY ANALYSIS AND REQUIREMENTS DETERMINATION 29

Chapter Objectives 29
Feasibility Analysis 30
Feasibility Types, 30
Requirements Determination 33
Problem Domain 35
Frameworks for Understanding and Doing Requirements Determination 37
Requirements Determination Sub-Activities, 38 *The Pieces Framework,* 38 *Kozar's Requirements Model,* 40 *Object-Oriented Requirements Determination Modeling Activities,* 44
Methods Used to Gather an Information System's Requirements 45
Feedback to the User, 48 *Requirements Ambiguity,* 49
Summary 51
Questions 52
References 53

3. AN OBJECT-ORIENTED METHODOLOGY AND MODEL 55

Chapter Objectives 55
Methodologies 55
The Traditional Methodology, 56 *Structured Analysis and Design Methodology,* 56 *Information Modeling Methodology,* 58 *Object-Oriented Methodology,* 59 *Key Characteristics of an Object-Oriented Methodology,* 60
Two Classic Problems Resolved with Object-Oriented Analysis and Design 66
Classification Theory 67
Coad's Object-Oriented Methodology 68

Coad's Object-Oriented Analysis and Design Methodology and Notation 68
Coad's Object Model Components, 72 *An Object-Oriented Model,* 73
Summary 82
Questions 82
References 83

4. OBJECTS AND CLASSES **86**

Chapter Objectives 86
Objects and Classes 87
Object and Class Rules and Guidelines, 88 *Class Attributes and Services Defined,* 89 *Comment on Object-Oriented Problem Solving Strategy,* 91
Finding Objects 92
Wirfs-Brock Noun Phrase Strategy, 92 *Wirfs-Brock CRC Strategy,* 94 *Conglomeration Strategy,* 94 *The Video Store Example - Finding Objects,* 96
A Future Enhancements Strategy 98
Summary 99
Questions 100
References 101

5. OBJECT RESPONSIBILITIES: ATTRIBUTES **102**

Chapter Objectives 102
Attributes 103
Determining Attributes, 106 *Attribute Types,* 107 *Object-Oriented Methodology Strategy for Different Attribute Types,* 109 *Object-Oriented Strategy for Multivalue Attributes,* 110 *The Video Store Example - Identifying Attributes,* 112
Summary 113
Questions 114
References 115

6. OBJECT RESPONSIBILITIES: CLASS AND OBJECT CONNECTIONS **116**

Chapter Objectives 116
"Who 'I' Know" Responsibility of an Object 117
Object Patterns 118
Generalization-Specialization Pattern, 118 *Generalization-Specialization Inheritance,* 125 *Whole-Part Object Connection Pattern,* 128 *Heuristics for Finding Whole-Part Patterns,* 134 *Object Connection Patterns,* 136

Video Store Example 144
Summary 147
Questions 147
References 148

7. OBJECT RESPONSIBILITIES: SERVICES AND SCENARIOS ***149***

Chapter Objectives 149
"What I Do" Responsibility of an Object 150
Business Objectives & Tactics, Information Systems Objectives and Tactics, and Policies and Procedures, 150 *Types of Services,* 151 *Basic Services,* 151 *Problem Domain Specific Services,* 154 *Finding and Identifying Services,* 159 *The Video Store Example - Identifying Services,* 161 *Other Techniques for Identifying Services,* 162
Service Details 164
Techniques for Documenting and Describing Service Details, 164 *Scenarios,* 165 *Structured English or Pseudocode,* 166 *Decision Tables and Decision Trees,* 168 *A Decision Table Example,* 171 *A Decision Tree Example,* 174 *State-Transition Diagrams,*176 *The Video Store Example - Assigning Services to Classes and Message Connections,* 177 *Transition from Systems Analysis to Systems Design,* 180
Summary 181
Questions 182
References 183

Part II Physical Design & Implementation

8. SYSTEMS DESIGN ***184***

Chapter Objectives 184
Information Systems Design 185
Historical Information Systems Design, 185 *An Object-Oriented Analysis and Design Methodology,* 187 *Information Systems Design Strategy Choices,* 187 *Object-Oriented Design,* 193
Alternative Object-Oriented Information Systems Development Strategies 202
Summary 203
Questions 204
References 205