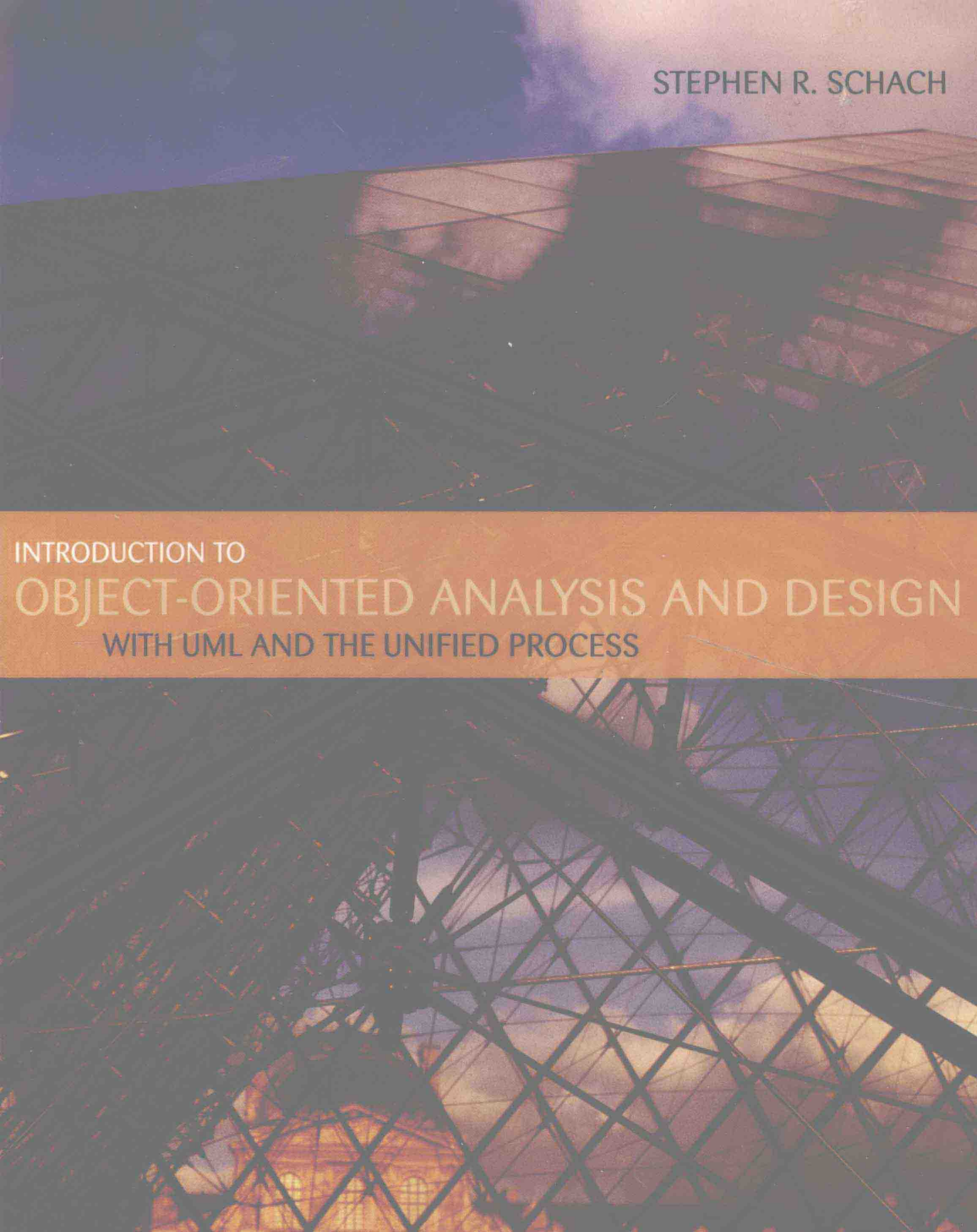

STEPHEN R. SCHACH

INTRODUCTION TO OBJECT-ORIENTED ANALYSIS AND DESIGN WITH UML AND THE UNIFIED PROCESS

An Introduction to Object-Oriented Systems Analysis and Design with UML and the Unified Process

Stephen R. Schach
Vanderbilt University

Mc Graw Hill Irwin

Boston Burr Ridge, IL Dubuque, IA Madison, WI New York San Francisco St. Louis
Bangkok Bogotá Caracas Kuala Lumpur Lisbon London Madrid Mexico City
Milan Montreal New Delhi Santiago Seoul Singapore Sydney Taipei Toronto

The McGraw·Hill Companies

Mc Graw Hill Irwin

INTRODUCTION TO OBJECT-ORIENTED ANALYSIS AND DESIGN
Published by McGraw-Hill/Irwin, a business unit of The McGraw-Hill Companies, Inc., 1221 Avenue of the Americas, New York, NY, 10020. Copyright © 2004 by The McGraw-Hill Companies, Inc.

This book is printed on acid-free paper.

domestic 1 2 3 4 5 6 7 8 9 0 VNH/VNH 0 9 8 7 6 5 4 3
international 1 2 3 4 5 6 7 8 9 0 VNH/VNH 0 9 8 7 6 5 4 3

ISBN 0-07-282646-0

Publisher: *Stewart Mattson*
Senior sponsoring editor: *Paul Ducham*
Developmental editor: *Kelly L. Delso*
Manager, Marketing and Sales: *Greta Kleinert*
Media producer: *Greg Bates*
Project manager: *Natalie J. Ruffatto*
Production supervisor: *Debra R. Sylvester*
Designer: *Adam Rooke*
Supplement producer: *Vicki Laird*
Senior digital content specialist: *Brian Nacik*
Cover design: *JoAnne Schopler*
Typeface: *10/12 Times Roman*
Compositor: *Interactive Composition Corporation*
Printer: *Von Hoffmann Corporation*

Library of Congress Cataloging-in-Publication Data

Schach, Stephen R.
An introduction to object-oriented systems analysis and design with UML and the unified process / Stephen R. Schach.
p. cm.
ISBN 0-07-282646-0 (alk. paper) — ISBN 0-07-121510-7 (international)
1. Object-oriented methods (Computer science) 2. UML (Computer science) 3. System analysis. 4. System design. I. Title.
QA76.9.O35S24 2004
005.1'17—dc21
2003042144

INTERNATIONAL EDITION ISBN 0-07-121510-7

www.mhhe.com

To Sharon, David, and Lauren

The following are registered trademarks:

Access
Ada
American Express
Ant
Apache
ArgoUML
Avis
Battlemap
C
C#
C++
COBOL
Coca-Cola
Communicator
Cutter Consortium
CVS
DB2
EDS
Excel
FORTRAN
Fortune 500
Foundation
General Motors
Hertz
IBM
Informix
Internet Explorer
Java
JBuilder
Linux
Lucent Technologies
Mac OS
Macintosh
MacProject
Mastercard
Microsoft
Netscape
New York Times
Objectory
OMT
OO-COBOL
Oracle
OS/360
OS/370
Palm Pilot
Peachtree
PeopleSoft
PL/I
Post-it note
Project
Prudential
PVCS
Rational
Requisite Pro
Rose
SAP
Software through Pictures
Solaris
SourceSafe
Standish Group
Sun
Sun Fire
Sybase
System Architect
System/360
Teamwork
TurboTax
UNIX
Visa
Visual Basic
Visual C++
Visual Java
Windows
Word
ZIP Code
zSeries

Preface

This book is a textbook for the first course in object-oriented systems analysis and design. The material conforms to the guidelines of IS'2001 (Course IS'01.7, Analysis and Logical Design); in particular, the student does not need to have any programming knowledge. The book uses the Unified Modeling Language (UML) to model information systems and the Unified Process as the methodology.

The primary objective of this book is to ensure that, by the end of the course, every student is able to perform object-oriented analysis and design. This objective is achieved in three different ways:

- Pedagogics: The steps of object-oriented analysis and design are carefully explained. After each step has been presented, it is illustrated by applying it to the first of the two running case studies. Then, when each step has been explained and illustrated separately, all the steps are applied to the second running case study.
- Organization: The material on object-oriented analysis and design (Part 2 of the book) is presented as early as possible by keeping Part 1 of the book as short as possible. The result is that the student has most of the semester or quarter to master the material on object-oriented analysis and design.
- Practice: The only way to become proficient in object-oriented analysis and design is by doing it repeatedly. Accordingly, there are different types of problems at the end of each chapter. First, there are exercises in object-oriented analysis and design. Second, there are problems based on the two running case studies, both of which are presented in the fullest detail. Third, there is a team-based term project in which the students perform systems analysis and design from scratch. The instructor has the option of assigning one or more problems of any type.

During the 1990s, there were three major figures in object-oriented technology, namely, Grady Booch (his approach is referred to as "Booch's method"), Ivar Jacobson ("Objectory"), and Jim Rumbaugh ("OMT"). Booch, Jacobson, and Rumbaugh then joined forces at Rational, Inc., and made two landmark contributions to information technology: the Unified Modeling Language (UML) and the Unified Process.

Within a few months of its introduction in 1997, UML version 1.0 literally swept the world. It is inconceivable that a textbook written today would attempt to represent an information system using any other modeling language. UML is now a product of the Object Management Group (OMG), a consortium of the leading software technology firms worldwide. This book uses version 1.4 of UML, the latest version at the time of writing.

Next, Booch, Jacobson, and Rumbaugh developed a methodology that integrated ("unified") their three separate (but similar) methodologies. This unified methodology was first known as the Unified Software Development Process (USDP) or the Rational Unified Process (RUP). Currently it is also called the Unified Process. Just as with UML, there is now no alternative—currently the best methodology for a twenty-first-century textbook on object-oriented analysis and design of information systems is the Unified Process. The other three methodologies are now no longer supported by their respective authors.

The book is divided into three parts. Part 1, as previously stated, has been kept as short as possible. It contains the minimum information needed to enable students to understand object-oriented analysis and design with the Unified Process and UML. Part 2 is on how to perform object-oriented analysis and design using the Unified Process with UML as the modeling language. Part 3 presents nine major topics that should be taught in a course on systems analysis and design, such as project management, planning, and team organization. That is, Part 2 teaches the necessary technical skills, whereas Part 3 stresses the managerial and interpersonal skills that every good systems analyst must have. In addition, further technical skills are taught in Part 3, such as testing and user-interface design. As with the rest of this book, there is an emphasis on the knowledge needed for information systems for the twenty-first century, including portability, reuse, interoperability, the World Wide Web, and process improvement, including CMM. Finally, if an instructor wishes to cover more technical concepts such as modularity and polymorphism and dynamic binding, the last chapter contains introductory-level material on these topics.

As previously mentioned, following the IS'2001 guidelines, programming is not a prerequisite for this book. However, if the class has programming experience, there are three additional problems that can be assigned. More specifically, the term project consists of a number of components that every student can complete, such as determining the requirements and performing the object-oriented analysis. However, the term project also includes three optional components, including a complete implementation, that the instructor may choose to assign if the class has the necessary programming experience.

PROBLEM SETS

There are review questions and problems at the end of each chapter. The review questions can all be answered from the relevant section of the textbook. The problems are of different types. As previously mentioned, there are exercises in object-oriented analysis and design, as well as problems based on the two running case studies. There are also essay-type problems, numerical problems, and problems that test how thoroughly the students understand what was taught in class. Finally, at the end of most chapters there are components of the term project.

The Instructor's Manual has solutions to every problem, including all the components of the term project. There are even implementations of the term project in C++ and Java. The instructor should contact his or her McGraw-Hill representative to obtain the Instructor's Manual.

SUPPLEMENTARY MATERIAL

A complete set of lecture notes in the form of PowerPoint slides are available at the website for this book, www.mhhe.com/schach. In addition, an Instructor CD-Rom is available to adopters. The Instructor CD-Rom contains the PowerPoint lecture notes, together with the following additional resources:

An Instructor's Manual containing suggestions for using the book, teaching suggestions, and answers to all the end of chapter material.

A testbank containing true-false, multiple choice, and fill-in-the-blank questions, as well as a computerized testbank with Brownstone Diploma software offering fully networkable LAN test administration. Tests also can be printed for standard paper delivery or posted to a website for student access.

Acknowledgments

I should like to thank the reviewers who have commented on the various earlier drafts of this book. They are

Thad Crews
Western Kentucky University

Kevin P. Duffy
The University of Texas at Arlington

Jeff Gray
University of Alabama at Birmingham

Kris Howell
University of Southern Colorado

Jon Jasperson
University of Oklahoma

Geoffrey Kennedy
University of Sydney

Sylnovie Merchant
California State University—Sacramento

Megan Murray
Kennesaw State University

Jeffrey Parsons
Memorial University of Newfoundland

David L. Paul
University of Denver

Elizabeth Perry
Binghamton University

Carl Scott
University of Houston

J. Michael Tarn
Western Michigan University

Amir Tomer
The Technion

Steve Walczak
Colorado University at Denver

Each of these reviewers, without exception, provided numerous helpful suggestions that have improved this book, and I am grateful to all of them. Nevertheless, I wish to single out Dr. Kris Howell, whose comments, suggestions, and criticisms were of immeasurable assistance to me.

A special thank you is extended to the following focus group participants:

Jeffrey Parsons
Memorial University of Newfoundland

Parag C. Pendharkar
Penn State University

Carl Scott
University of Houston

Ron Thompson
Wake Forest University

Steve Walczak
Colorado University at Denver

I am indebted to three individuals who also have made significant contributions to many of my earlier books. Dr. Jeff Gray was previously a co-author of four Instructor's Manuals. For this book he has served as a reviewer, and he also was responsible for Appendices B through E; that is, he performed the design and implementation of both the case studies. My son David and my daughter Lauren are two of the co-authors of the Instructor's Manual. Lauren also contributed to the PowerPoint lecture notes. I thank Jeff, David, and Lauren for their usual meticulous work.

Kris Irwin took over Jeff's previous role—she provided a complete solution to the term project, including implementing it in both Java and C++. It was a pleasure working with an information technology professional of Kris's caliber. I also thank Kris for her many suggestions that led to improvements in the term project.

The copyeditor was Betsy Blumenthal. I thank her for her superlative work and her many helpful suggestions.

I turn now to my publisher, McGraw-Hill. The aim of object-oriented analysis is to determine *what* the target information system should do, and the aim of object-oriented design is to determine *how* the information system should do it. From the start of this project, publisher George Werthman and senior editor Rick Williamson made it clear *what* book they wanted me to write, but they left *how* to write it entirely up to me. It has been an eye-opening education to work with George and Rick. It also has been tremendous fun.

I should like to single out project manager Natalie Ruffatto. Natalie was a tower of strength throughout the production process. I deeply appreciate all that she has done.

I am grateful to my development editor, Kelly Delso, for her help from start to finish. I should also like to thank other McGraw-Hill professionals: Paul Ducham, senior sponsoring editor; Greta Kleinert, marketing manager; Greg Bates, media producer; Debra R. Sylvester, production supervisor; Adam Rooke, designer; and Rose Range, supplement coordinator.

I am also grateful to Jade Moran, project manager, and the compositors at Interactive Composition Corporation for their outstanding work. They were always willing to go the extra mile—and more.

I would like to thank the numerous instructors from all over the world who have sent me e-mail about my earlier books. I have used some of their comments, criticisms, and suggestions in this book. I look forward to receiving instructors' feedback on this book, too. My e-mail address is srs@vuse.vanderbilt.edu.

I also want to thank Kai Chen, Fang Fang, David Gannaway, Yun Huang, Kris Irwin (again), Michael Kearney, Johann Klemmack, Tao Lu, Yolanda McMillian, Sarah Miracle, Melissa Nestor, and Glenn Stein, the students in a graduate course on the Unified Process that I recently taught at Vanderbilt University. Their comments and critical analyses have helped to shape my views on the Unified Process, and thereby enriched this book.

I am grateful for the questions and suggestions about earlier books that students have e-mailed to me, and I look forward to receiving student feedback on this book.

I should like to thank Howard Aksen. In 1988 he persuaded me to write my first book, *Software Engineering*. Without Howard's urging, I would never have had 15 years of sheer enjoyment writing textbooks.

Finally, as always, I would like to thank my family for their unfailing support. Once again, writing this book has been a family activity. For the 11th time, it is my privilege to dedicate this book to my wife, Sharon, and my children, David and Lauren, with love.

Stephen R. Schach

Brief Contents

PART ONE

PART TWO

PART THREE

APPENDIXES

Contents

Chapter 5
The Requirements Workflow II 75

Chapter 6
The Object-Oriented Analysis Workflow I 99

Chapter 7
The Object-Oriented Analysis Workflow II 129

Chapter 8
The Object-Oriented Design Workflow 161

Chapter 9
The Workflows and Phases of the Unified Process 177

Chapter 10
More on UML 189

PART THREE
MAJOR TOPICS IN SYSTEMS ANALYSIS AND DESIGN 205

Chapter 11
CASE 207

Chapter 12
Teams 231

Chapter 13
Testing 243

Chapter 14
Management Issues 259

Chapter 15
Planning and Estimating 287

Chapter 16
Maintenance 311

Chapter 17
User-Interface Design 325

Chapter 18
Introduction to Web-Based Information Systems 337

Chapter 19
Introduction to Database Management Systems 347

Part 1

Introduction to UML and the Unified Process

The first part of this book consists of three relatively short chapters. These chapters contain the background information needed to perform systems analysis and design or, more precisely, to perform object-oriented systems analysis and design using UML and the Unified Process. The three chapters explain what systems analysis and design is, what the phrase "object oriented" means, and what UML and the Unified Process are and why they are so important.

Chapter 1, *Introduction to Information Systems,* provides basic information about information systems. The major topic of this chapter is the phases of the traditional information system life cycle.

Chapter 2 is entitled *How Information Systems Are Developed.* Whereas Chapter 1 describes how information systems would be developed in an ideal world, Chapter 2 explains in detail why this does not happen in practice. Most of the chapter is devoted to the iterative and incremental life cycle, the way that information systems are usually developed in the real world.

The Object-Oriented Paradigm, UML, and the Unified Process is the title of Chapter 3. The traditional ("structured") approach to developing information systems has proved to be unsatisfactory for larger information systems. The object-oriented paradigm is the modern approach to developing information systems. It is explained in detail in Chapter 3, which also contains an introduction to UML and the Unified Process.