EDITION
9

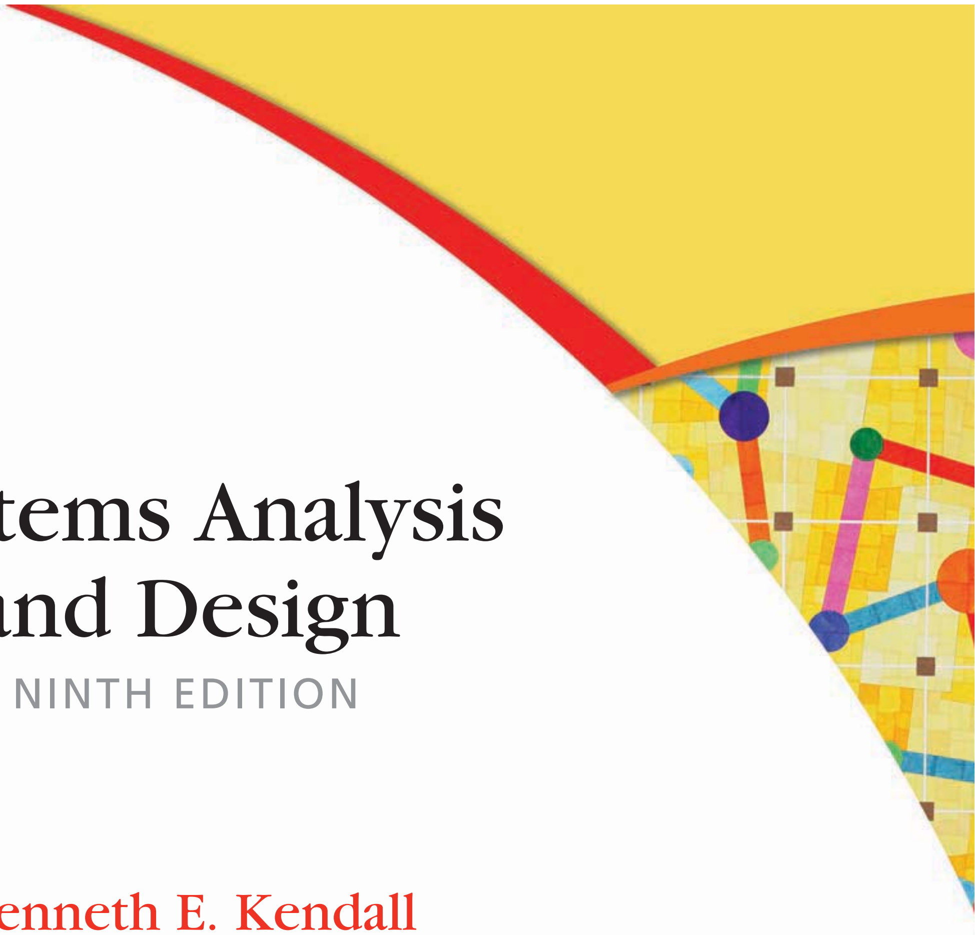
Systems Analysis and Design

KENDALL & KENDALL

CONSULTING OPPORTUNITIES

Systems Analysis and Design

This page intentionally left blank



Systems Analysis and Design

NINTH EDITION

Kenneth E. Kendall
RUTGERS UNIVERSITY
School of Business–Camden
Camden, New Jersey

Julie E. Kendall
RUTGERS UNIVERSITY
School of Business–Camden
Camden, New Jersey

PEARSON

Boston Columbus Indianapolis New York San Francisco Upper Saddle River
Amsterdam Cape Town Dubai London Madrid Milan Munich Paris Montréal Toronto
Delhi Mexico City São Paulo Sydney Hong Kong Seoul Singapore Taipei Tokyo

Editor in Chief: Stephanie Wall
Executive Editor: Bob Horan
Director of Editorial Services: Ashley Santora
Senior Editorial Project Manager: Kelly Loftus
Editorial Assistant: Ashlee Bradbury
Director of Marketing: Maggie Moylan
Executive Marketing Manager: Anne Fahlgren
Marketing Assistant: Gianna Sandri
Senior Managing Editor: Judy Leale
Production Project Manager: Ilene Kahn
Operations Specialist: Maura Zaldivar
Creative Director: Blair Brown
Art Director: Steve Frim
Interior Designer: Jill Lehan
Cover Designer: Jill Lehan
Cover Art: *P3 Vega* courtesy of Richard Kalina and Lennon, Weinberg Gallery, NYC
Full-Service Project Management and Composition: Element LLC
Printer/Binder: Courier/Kendallville
Cover Printer: Lehigh-Phoenix Color/Hagerstown
Text Font: 10/12 Times

Credits and acknowledgments borrowed from other sources and reproduced, with permission, in this textbook appear on the appropriate page within text.

Microsoft and/or its respective suppliers make no representations about the suitability of the information contained in the documents and related graphics published as part of the services for any purpose. All such documents and related graphics are provided "as is" without warranty of any kind. Microsoft and/or its respective suppliers hereby disclaim all warranties and conditions with regard to this information, including all warranties and conditions of merchantability, whether express, implied or statutory, fitness for a particular purpose, title and non-infringement. In no event shall Microsoft and/or its respective suppliers be liable for any special, indirect or consequential damages or any damages whatsoever resulting from loss of use, data or profits, whether in an action of contract, negligence or other tortious action, arising out of or in connection with the use or performance of information available from the services.

The documents and related graphics contained herein could include technical inaccuracies or typographical errors. Changes are periodically added to the information herein. Microsoft and/or its respective suppliers may make improvements and/or changes in the product(s) and/or the program(s) described herein at any time. Partial screen shots may be viewed in full within the software version specified.

Microsoft® and Windows® are registered trademarks of the Microsoft Corporation in the U.S.A. and other countries. This book is not sponsored or endorsed by or affiliated with the Microsoft Corporation.

Many of the designations by manufacturers and sellers to distinguish their products are claimed as trademarks. Where those designations appear in this book, and the publisher was aware of a trademark claim, the designations have been printed in initial caps or all caps.

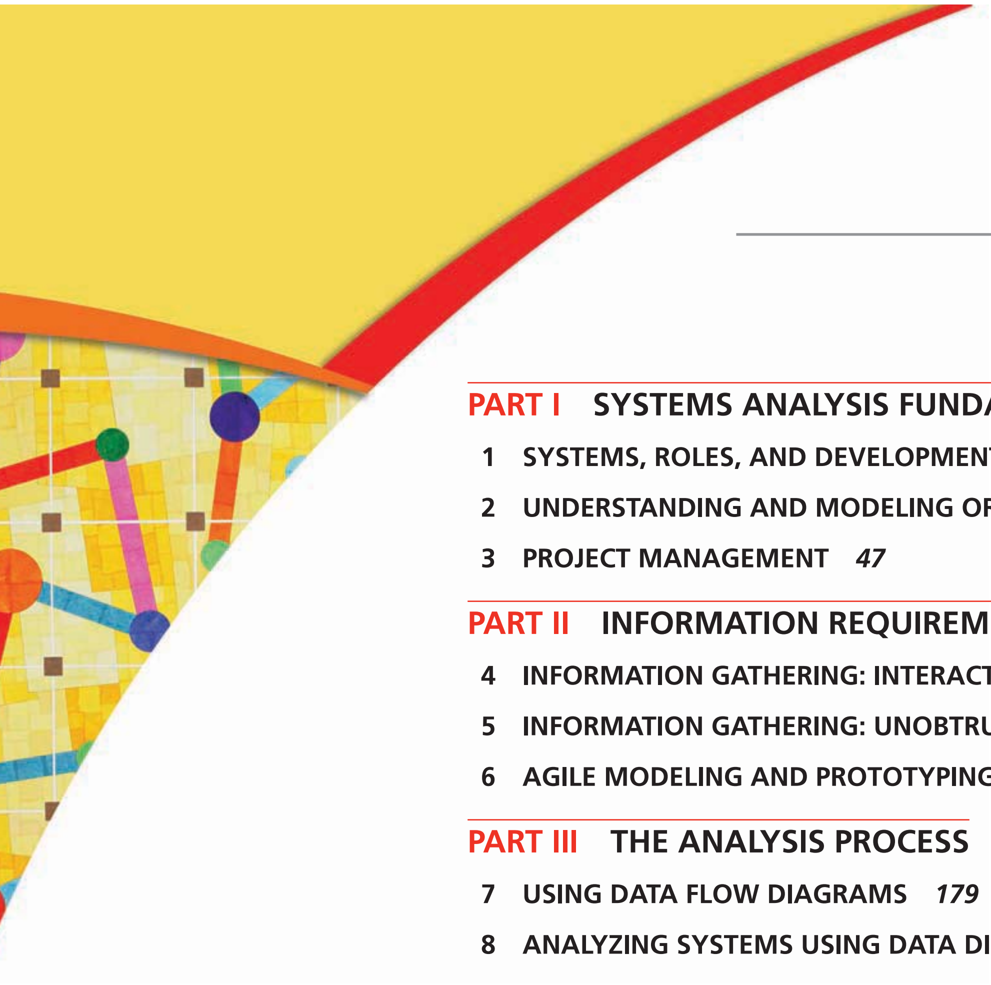
Library of Congress Cataloging-in-Publication Data

Library of Congress Cataloging-in-Publication Data is available upon request.

PEARSON

10 9 8 7 6 5 4 3 2 1
ISBN 10: 0-13-302344-3
ISBN 13: 978-0-13-302344-2

To the memory of Julia A. Kendall and the memory of Edward J. Kendall, whose lifelong example of working together will inspire us forever.



BRIEF CONTENTS

CONTENTS

PART III THE ANALYSIS PROCESS 179

7 USING DATA FLOW DIAGRAMS 179

8 ANALYZING SYSTEMS USING DATA DICTIONARIES 209

9 PROCESS SPECIFICATIONS AND STRUCTURED DECISIONS 234

10 OBJECT-ORIENTED SYSTEMS ANALYSIS AND DESIGN USING UML 253

This page intentionally left blank

PREFACE

NEW TO THIS EDITION

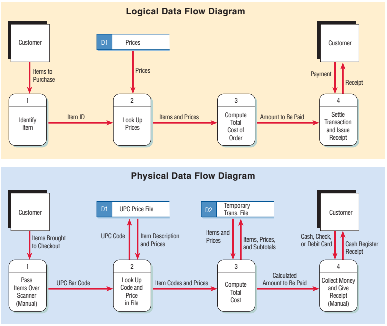
The ninth edition of *Systems Analysis and Design* includes extensive changes inspired by the swift transformations in the IS field over the past three years, and they are included as a response to the thoughtful input of our adopters, students, and reviewers. Many innovative and upgraded features are incorporated throughout this new edition. In particular:

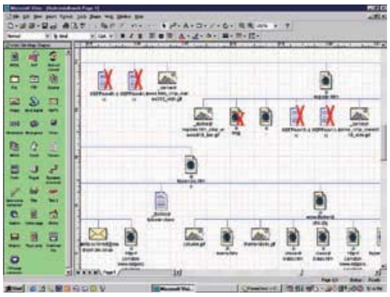
- New coverage of how systems analysts and organizations can participate in open source communities (Chapter 1)
- Expanded coverage of the analyst role in ERP (enterprise systems) (Chapter 2)
- New in-depth coverage of project management techniques (Chapter 3)
- Expanded coverage of when to use cloud services versus purchasing hardware and software (Chapter 3)
- New coverage of time estimation techniques for project management (Chapter 3)
- New coverage of the work breakdown structure (WBS) for project management (Chapter 3)
- New material on designing corporate and ecommerce sites to include Web 2.0 technologies and social media (Chapter 11)
- Innovative treatment of designing apps for smartphones and tablets (Chapter 11)
- Expanded coverage of designing input for intranets, the Web, smartphones, and tablets (Chapter 12)
- New material on the relationship of business intelligence to data warehouses, big data, business analytics, and text analytics (Chapter 13)
- Innovative coverage on designing gesture-based interfaces for smartphones and tablets (Chapter 14)
- Additional material on designing alerts, queries, and notices for smartphones and tablets (Chapter 14)
- Innovative handling of designing two-dimensional (2D) codes such as Microsoft Tags and QR codes for input (Chapter 15)
- New material on how service-oriented architecture and cloud computing are changing the nature of information systems design (Chapter 16)
- Expanded coverage of ERP systems and their relationship to cloud computing (Chapter 16)

DESIGN FEATURES

Figures have a stylized look in order to help students more easily grasp the subject matter.

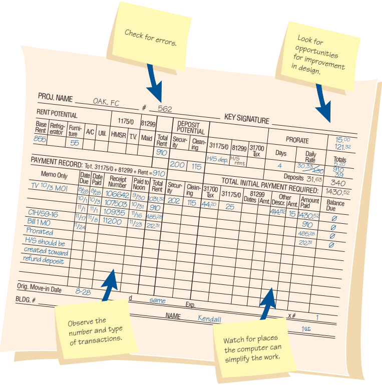
Conceptual diagrams are used to introduce the many tools that systems analysts have at their disposal. This example shows the differences between logical data flow diagrams and physical data flow diagrams. Conceptual diagrams are color coded so that students can distinguish easily among them, and their functions are clearly indicated. Many other important tools are illustrated, including use case diagrams, sequence diagrams, and class diagrams.


Logical Data Flow Diagram
Customer
D1 Prices
Customer
Items to Purchase
Prices
Payment
Receipt
1 Identify Item
Item ID
2 Look Up Prices
Items and Prices
3 Compute Total Cost of Order
Amount to Be Paid
4 Settle Transaction and Issue Receipt
Physical Data Flow Diagram
Customer
D1 UPC Price File
D2 Temporary Trans. File
Customer
Items Brought to Checkout
UPC Code
Item Description and Prices
Items and Prices
Items, Prices, and Subtotals
Cash, Check, or Debit Card
Cash Register Receipt
1 Pass Items Over Scanner (Manual)
UPC Bar Code
2 Look Up Code and Price in File
Item Codes and Prices
3 Compute Total Cost
Calculated Amount to Be Paid
4 Collect Money and Give Receipt (Manual)



Computer displays demonstrate important software features that are useful to the analyst. This example shows how a website can be evaluated for broken links by using a package such as Microsoft Visio. Actual screen shots show important aspects of design. Analysts are continuously seeking to improve the appearance of the screens and Web pages they design. Colorful examples help to illustrate why some screen designs are particularly effective.

Paper forms are used throughout to show input and output design as well as the design of questionnaires. Blue ink is always used to show writing or data input, thereby making it easier to identify what was filled in by users. Although most organizations have computerization of manual processes as their goal, much data capture is still done using paper forms. Improved form design enables analysts to ensure accurate and complete input and output. Better forms can also help streamline new internal workflows that result from newly automated business-to-consumer (B2C) applications for ecommerce on the Web.



Check for errors.

Look for opportunities for improvement in design.

PROJ. NAME OAK. FC # 562 KEY SIGNATURE

RENT POTENTIAL

Base Rent	Refrigerator	Furniture	A/C	Util.	1175/0 HMSR	81299 T.V.	Maid	Total Rent
855		55						910

DEPOSIT POTENTIAL

Security	Cleaning	31175/0	81299	31700 Tax
200	115	H/S dep.	H/S rent	

PRORATE

Days	Daily Rate	Totals
		15.00 121.32
4	30.33 ~~4.30~~	5.20 910 39
Deposits	31.63	340

TOTAL INITIAL PAYMENT REQUIRED: 1430.52

PAYMENT RECORD: Tot. 31175/0 + 81299 + Rent = 910

Memo Only	Date Due	Date Paid	Receipt Number	Paid to Noon	Total Rent	Security	Cleaning	31700 Tax	31175/0	81299 Dates	81299 Amt.	Other Descr.	Other Amt.	Amount Paid	Balance Due
TV 10/3 MO!	8/28	8/28	106642	9/30	1031.32	202	115	44.20	25			414.82	15	1430.52	Ø
	10/1	10/3	107503	10/31	910									910	Ø
	11/1	11/1	10935	11/16	485.28									485.28	Ø
C1H/S9-16	11/17	11/8	11200	11/23	212.31									212.31	Ø
Bill 1 MO	11/24														
Prorated															
H/S should be created toward refund deposit															

Orig. Move-in Date 8-28

BLDG. # d same Exp.

NAME Kendall x# 1

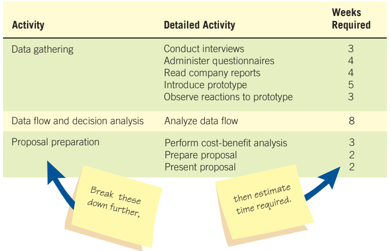
1st

Observe the number and type of transactions.

Watch for places the computer can simplify the work.

Tables are used when an important list needs special attention or when information needs to be organized or classified. In addition, tables are used to supplement the understanding of the reader in a way that departs from how material is organized in the narrative portion of the book. Most analysts find using tables a useful way to organize numbers and text into a meaningful "snapshot."

This example of a table from Chapter 3 shows how analysts can refine their activity plans for analysis by breaking them down into smaller tasks and then estimating how much time it will take to complete them. This book is built on the idea that systems analysis and design is a process that integrates the use of many tools with the unique talents of the systems analyst to systematically improve business through the implementation or modification of computerized information systems. Systems analysts can grow in their work: by taking on new IT challenges, whether they are posed by designing for multiple platforms, new types of users, or implementing cloud-based systems, and by keeping up to date in their profession through the application of new methods, software, and alternative tools.

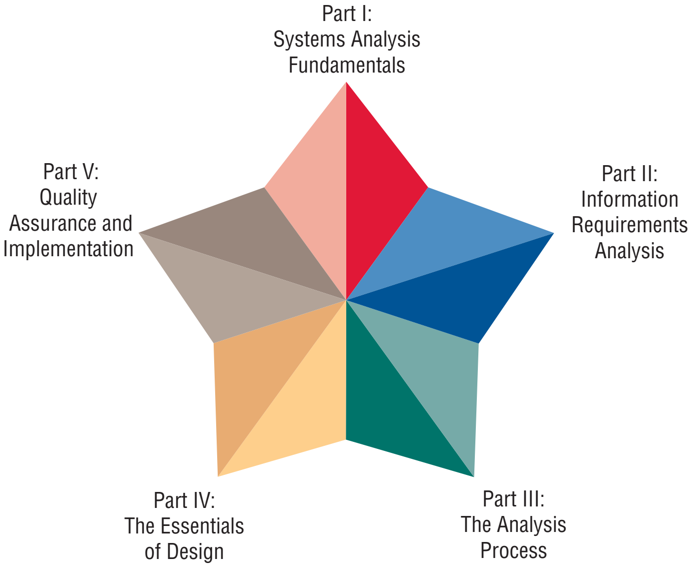


Activity	Detailed Activity	Weeks Required
Data gathering	Conduct interviews Administer questionnaires Read company reports Introduce prototype Observe reactions to prototype	3 4 4 5 3
Data flow and decision analysis	Analyze data flow	8
Proposal preparation	Perform cost-benefit analysis Prepare proposal Present proposal	3 2 2

Break these down further,

then estimate time required.

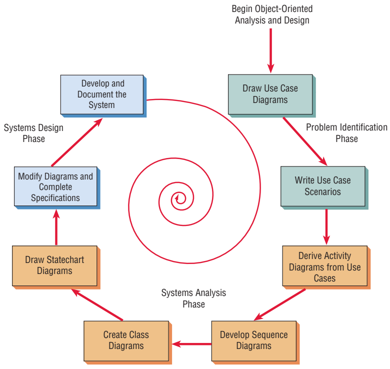
A BRIEF TOUR OF THE NINTH EDITION


Part I: Systems Analysis Fundamentals
Part II: Information Requirements Analysis
Part III: The Analysis Process
Part IV: The Essentials of Design
Part V: Quality Assurance and Implementation

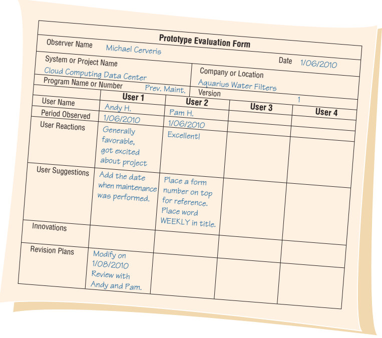
Systems analysis and design is typically taught in one or two semesters. This book may be used in either situation. The text is appropriate for undergraduate (junior or senior) curricula at a four-year university, graduate school, or community college. The level and length of the course can be varied and supplemented by using real-world projects, HyperCase, The CPU Case online, or other materials available on the Instructor Resource Center.

The text is divided into five major parts: Systems Analysis Fundamentals (Part I), Information Requirements Analysis (Part II), The Analysis Process (Part III), The Essentials of Design (Part IV), and Quality Assurance and Implementation (Part V).

Part I (Chapters 1–3) stresses the basics that students need to know about what an analyst does and introduces the three main methodologies of the systems development life cycle (SDLC), agile approaches, and object-oriented analysis with UML, along with reasons and situations for when to use them. Part I introduces the three roles of a systems analyst—consultant, supporting expert, and agent of change—along with ethical issues and professional guidelines for serving as a systems consultant. There is also material on virtual teams and virtual organizations, and the concept of human–computer interaction (HCI) is introduced. The use of open source software (OSS) and how analysts and organizations can participate in open source communities is also introduced. Chapter 2 includes how to initially approach an organization by drawing context-level data flow diagrams, using entity-relationship models, and developing use cases and use case scenarios. It views the organization as a system through the description of enterprise systems (ERP). Chapter 3 focuses on project management. It introduces new material on when to use cloud services versus purchasing hardware and software. Expanded coverage of project management techniques is also included, including new time estimation techniques for project management. Chapter 3 also includes new material to help students approach projects using the work breakdown structure (WBS). Creating a problem definition, developing a project charter, and determining feasibility are also covered. Chapter 3 guides students in professionally writing and presenting an effective systems proposal, one that incorporates figures and graphs to communicate with users.


Begin Object-Oriented Analysis and Design
Draw Use Case Diagrams
Problem Identification Phase
Write Use Case Scenarios
Derive Activity Diagrams from Use Cases
Develop Sequence Diagrams
Systems Analysis Phase
Create Class Diagrams
Draw Statechart Diagrams
Modify Diagrams and Complete Specifications
Systems Design Phase
Develop and Document the System

Part II (Chapters 4–6) emphasizes the use of systematic and structured methodologies for performing information requirements analysis. Attention to analysis helps analysts ensure that they are addressing the correct problem before designing a system. Chapter 4 introduces a group of interactive methods, including interviewing, Joint Application Design (JAD), listening to user stories, and constructing questionnaires. Chapter 5 introduces a group of unobtrusive methods for ascertaining information requirements of users. These methods include sampling, investigating hard and archival data, and observation of decision makers' behavior and their physical environment. Chapter 6 on agile modeling and prototyping is innovative in its treatment of prototyping as another data-gathering technique that enables the analyst to solve the right problem by getting users involved from the start. Since agile approaches have their roots in prototyping, this chapter begins with prototyping to provide a proper context for understanding, and then takes up the agile approach. The values and principles, activities, resources, practices, processes, and tools associated with agile methodologies are presented.


Prototype Evaluation Form

Observer Name Michael Cerveris

System or Project Name Cloud Computing Data Center

Date 1/06/2010

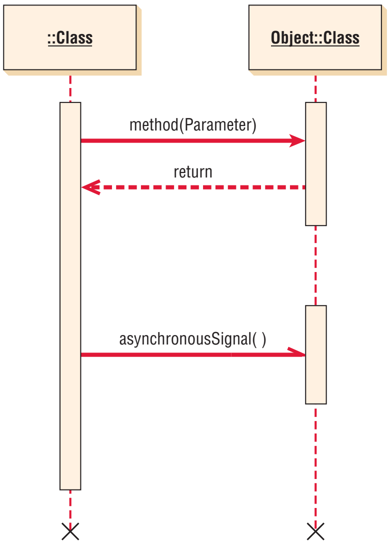
Company or Location Aquarius Water Filters

Program Name or Number Prev. Maint.

Version 1

	User 1	User 2	User 3	User 4
User Name	Andy H.	Pam H.		
Period Observed	1/06/2010	1/06/2010		
User Reactions	Generally favorable, got excited about project	Excellent!		
User Suggestions	Add the date when maintenance was performed.	Place a form number on top for reference. Place word WEEKLY in title.		
Innovations				
Revision Plans	Modify on 1/08/2010 Review with Andy and Pam.			

Part III (Chapters 7–10) details the analysis process. It builds on the previous two parts to move students into analysis of data flows as well as structured and semistructured decisions. It provides step-by-step details on how to use structured techniques to draw data flow diagrams (DFDs). Chapter 7 provides coverage of how to create child diagrams; how to develop both logical and physical data flow diagrams; and how to partition data flow diagrams. Chapter 8 features material on the data repository and vertical balancing of data flow diagrams. Chapter 8 also includes extensive coverage of Extensible Markup Language (XML) and demonstrates how to use



data dictionaries to create XML. Chapter 9 includes material on developing process specifications. A discussion of both logical and physical process specifications shows how to use process specifications for horizontal balancing. Chapter 9 also covers how to diagram structured decisions with the use of structured English, decision tables, and decision trees. In addition, the chapter covers how to choose an appropriate decision analysis method for analyzing structured decisions and creating process specifications.

Part III concludes with Chapter 10 on object-oriented systems analysis and design. This chapter includes an in-depth section on using Unified Modeling Language (UML). There is detailed coverage of the use case model, creating the class model diagram with UML, sequence diagrams, creating gen/spec diagrams, use case scenarios, and activity diagrams. Through several examples and Consulting Opportunities, this chapter demonstrates how to use an object-oriented approach. Consulting Opportunities, diagrams, and problems enable students to learn and use UML to model systems from an object-oriented perspective. Students learn the appropriate situations for using an object-oriented approach. This chapter helps students to decide whether to use the SDLC, the agile approach, or object-oriented systems analysis and design to develop a system.

Part IV (Chapters 11–14) covers the essentials of design. It begins with designing output because many practitioners believe systems to be output driven. The design of Web-based forms is covered in detail. Particular attention is paid to relating output method to content, the effect of output on users, and designing good forms and screens. Chapter 11 considers output, including Web displays, audio, and electronic output such as Web pages, email, and RSS feeds. Designing a website for ecommerce purposes is emphasized, and the importance of adding Web 2.0 technologies and social media to corporate and ecommerce websites is explored. Designing apps for smartphones and tablets is included, along with storyboarding, wireframing, and mockups. Output production and XML are covered.

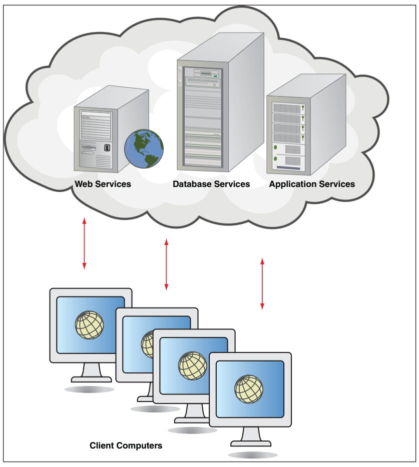
Chapter 12 includes innovative material on designing for smartphones and tablets as well as designing Web-based input forms and other electronic forms design. Also included is computer-assisted forms design. Chapter 12 also features in-depth coverage of website design, including guidelines on when designers should add video, audio, and animation to website designs. There is detailed consideration of how to create effective graphics for corporate websites and ways to design effective onscreen navigation for website users.

Coverage of intranet and extranet page design is also included. Consideration of database integrity constraints has been included as well, in addition to how the user interacts with the computer and how to design an appropriate interface. The importance of user feedback is also found in Part IV. How to design accurate data entry procedures that take full advantage of computer and human capabilities to assure entry of quality data is emphasized here.

Chapter 13 demonstrates how to use an entity-relationship diagram to determine record keys, as well as providing guidelines for file/database relation design. Students are shown the relevance of database design for the overall usefulness of the system, and how users actually use databases. The concepts of business intelligence (BI) and its relationships to big data, business analytics, and text analytics are also introduced in the context of data warehouses.

Chapter 14 emphasizes human–computer interaction (HCI), especially as it relates to interface design. It discusses the importance of HCI in designing systems that suit individuals and assisting them in achieving personal and organizational goals through their use of information technology. The concept of usability is introduced, so that systems analysis students can knowledgeably incorporate HCI practices into their designs. Chapter 14 introduces material on how to design gesture-based (multitouch) interfaces for smartphones and tablets, as well as designing alerts, notices and queries. Material on designing easy onscreen navigation for website visitors is also included. The chapter presents innovative approaches to searching on the Web, highlights material on GUI design, and provides innovative approaches to designing dialogs. Chapter 14 articulates specialized design considerations for ecommerce websites. Mashups, new applications created by combining two or more Web-based application programming interfaces, are also introduced. Chapter 14 also includes extensive coverage on how to formulate queries, all within the framework of HCI.

Part V (Chapters 15 and 16) concludes the book. Chapter 15 focuses on designing accurate data entry procedures and includes material on managing the supply chain through the effective design of business-to-business (B2B) ecommerce. It includes suggestions for incorporating two-dimensional codes, such as QR codes and Microsoft Tags, into data entry designs. It also considers the usefulness of RFID for automatic data collection. Chapter 16 emphasizes taking a total quality approach to improving software design and maintenance. In addition, material on system security and firewalls is included. Testing, auditing, and maintenance of systems are discussed in the context of total quality management. This chapter helps students understand how service-oriented architecture (SOA) and cloud computing combined with ERP are significantly altering the landscape of information systems design. In addition, students learn how to design appropriate training programs for users of the new system, how to recognize the differences among physical conversion strategies, and how to be able to recommend an appropriate one to a client. Chapter 16 also presents techniques for modeling networks, which can be done with popular tools such as Microsoft Visio.



Material on security and privacy in relation to designing ecommerce applications is included. Coverage of security, specifically firewalls, gateways, Public Key Infrastructure (PKI), Secure Electronic Transaction (SET), Secure Sockets Layer (SSL), virus protection software, URL filtering products, email filtering products, and virtual private networks (VPN) is included. In addition, current topics of interest to designers of ecommerce applications, including the development and posting of corporate privacy policies, are covered.

Important coverage of how the analyst can promote and monitor a corporate website is included in this section, which features Web activity monitoring, website promotion, Web traffic analysis, and audience profiling to ensure the effectiveness of new ecommerce systems. Techniques for evaluating the completed information systems project are covered systematically as well.

This ninth edition contains an updated **Glossary** of terms and a separate list of updated **Acronyms** used in the book and in the systems analysis and design field.

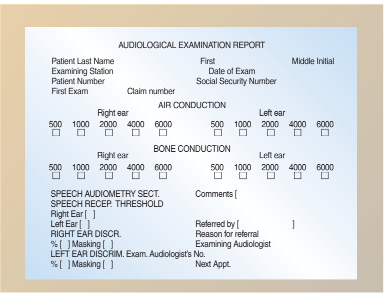
PEDAGOGICAL FEATURES

Chapters in this ninth edition contain:

- **Learning Objectives** at the beginning of each chapter
- **Summaries** that tie together the salient points of each chapter while providing an excellent source of review for exams
- **Keywords and Phrases**
- **Review Questions**
- **Problems**
- **Group Projects** that help students work together in a systems team to solve important problems that are best solved through group interaction
- **Consulting Opportunities,** now with more than 50 minicases throughout the book
- **Mac Appeal** columns that inform students on design software available on the Mac and iPhone
- **HyperCase Experiences**

CONSULTING OPPORTUNITIES

This ninth edition presents more than 50 Consulting Opportunities, and many of them address significant and emerging topics arising in information systems, including designing systems from an HCI perspective, ecommerce applications for the Web, cloud computing decisions, and using UML to model information systems from an object-oriented perspective. Consulting Opportunities can be used for motivating thoughtful in-class discussions or assigned as homework or take-home exam questions.



AUDIOLOGICAL EXAMINATION REPORT

Patient Last Name First Middle Initial
Examining Station Date of Exam
Patient Number Social Security Number
First Exam Claim number

AIR CONDUCTION
Right ear Left ear
500 □ 1000 □ 2000 □ 4000 □ 6000 □ 500 □ 1000 □ 2000 □ 4000 □ 6000 □

BONE CONDUCTION
Right ear Left ear
500 □ 1000 □ 2000 □ 4000 □ 6000 □ 500 □ 1000 □ 2000 □ 4000 □ 6000 □

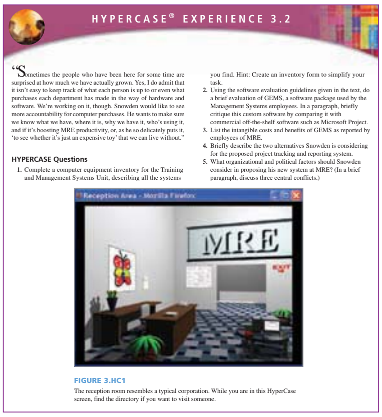
SPEECH AUDIOMETRY SECT. Comments [
SPEECH RECEP. THRESHOLD
Right Ear [ ]
Left Ear [ ] Referred by [ ]
RIGHT EAR DISCR. Reason for referral
% [ ] Masking [ ] Examining Audiologist
LEFT EAR DISCRIM. Exam. Audiologist's No.
% [ ] Masking [ ] Next Appt.

Because not all systems work demands extended two- or three-year projects, our book contains many Consulting Opportunities that can be solved in 20 to 30 minutes of group discussion or individual writing. These minicases, written in a humorous manner to enliven the material, require students to synthesize what they have learned up to that point in the course, ask students to mature in their professional and ethical judgment, and expect students to articulate the reasoning that led to their systems decisions.

HYPERCASE EXPERIENCES

HyperCase® Experiences that pose challenging student exercises are present in each chapter. HyperCase 2.9 has organizational problems featuring state-of-the-art technological systems. HyperCase represents an original virtual organization that allows students who access it to become immediately immersed in organizational life. Students will interview people, observe office environments, analyze their prototypes, and review the documentation of their existing systems.

HyperCase 2.9 is Web-based, interactive software that presents an organization called Maple Ridge Engineering (MRE) in a colorful, three-dimensional graphics environment. HyperCase permits professors to begin approaching a systems analysis and design class with exciting multimedia material. Carefully watching their use of time and managing multiple methods, students use the hypertext characteristics of HyperCase on the Web to create their own individual paths through the organization.



HYPERCASE® EXPERIENCE 3.2

"Sometimes the people who have been here for some time are surprised at how much we have actually grown. Yes, I do admit that it isn't easy to keep track of what each person is up to or even what purchases each department has made in the way of hardware and software. We're working on it, though. Snowden would like to see more accountability for computer purchases. He wants to make sure we know what we have, where it is, why we have it, who's using it, and if it's boosting MRE productivity, or, as he so delicately puts it, 'to see whether it's just an expensive toy' that we can live without."

HYPERCASE Questions

1. Complete a computer equipment inventory for the Training and Management Systems Unit, describing all the systems you find. Hint: Create an inventory form to simplify your task.
2. Using the software evaluation guidelines given in the text, do a brief evaluation of GEMS, a software package used by the Management Systems employees. In a paragraph, briefly critique this custom software by comparing it with commercial off-the-shelf software such as Microsoft Project.
3. List the intangible costs and benefits of GEMS as reported by employees of MRE.
4. Briefly describe the two alternatives Snowden is considering for the proposed project tracking and reporting system.
5. What organizational and political factors should Snowden consider in proposing his new system at MRE? (In a brief paragraph, discuss three central conflicts.)

FIGURE 3.HC1
The reception room resembles a typical corporation. While you are in this HyperCase screen, find the directory if you want to visit someone.

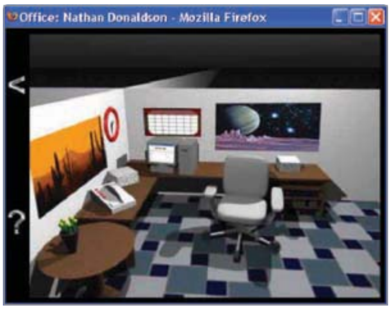
Maple Ridge Engineering is drawn from the actual consulting experiences of the authors of the original version (Raymond Barnes, Richard Baskerville, Julie E. Kendall, and Kenneth E. Kendall). Allen Schmidt joined the project for version 2.0 and has remained with it. Peter Schmidt was the HTML programmer, and Jason Reed created the images for the initial Web version.

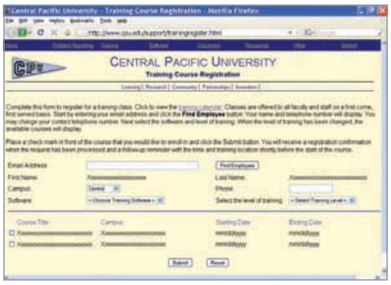
Each chapter contains HyperCase Experiences that include assignments (and even some clues) to help students solve difficult organizational problems including developing new systems, merging departments, hiring of employees, security, ecommerce, and disaster recovery planning they encounter at MRE. HyperCase has been fully tested in classrooms and was an award winner in the Decision Sciences Institute Innovative Instruction competition.

EXPANDED WEB SUPPORT

Systems Analysis and Design, ninth edition, features Web-based support for solid but lively pedagogical techniques in the information systems field:

- The website, located at **www.pearsonhighered.com/kendall**, contains a wealth of critical learning and support tools, which keep class discussions exciting.





Central Pacific University - Training Course Registration - Mozilla Firefox

http://www.cpu.edu/support/trainingregister.html

CENTRAL PACIFIC UNIVERSITY

Training Course Registration

Learning | Research | Community | Partnerships | Innovation |

Complete this form to register for a training class. Click to view the training calendar. Classes are offered to all faculty and staff on a first come, first served basis. Start by entering your email address and click the **Find Employee** button. Your name and telephone number will display. You may change your contact telephone number. Next select the software and level of training. When the level of training has been changed, the available courses will display.

Place a check mark in front of the course that you would like to enroll in and click the Submit button. You will receive a registration confirmation when the request has been processed and a follow-up reminder with the time and training location shortly before the start of the course.

Email Address		Find Employee	
First Name	Xxxxxxxxxxxxxxxxxxx	Last Name	Xxxxxxxxxxxxxxxxxxxxxxxxxx
Campus	Central	Phone	
Software	-Choose Training Software-	Select the level of training	-Select Training Level-

Course Title	Campus	Starting Date	Ending Date
□ Xxxxxxxxxxxxxxxxxxxxxxxx	Xxxxxxxxxxxxxxxxxxxxx	mm/dd/yyyy	mm/dd/yyyy
□ Xxxxxxxxxxxxxxxxxxxxxxxx	Xxxxxxxxxxxxxxxxxxxxx	mm/dd/yyyy	mm/dd/yyyy

- **HyperCase 2.9** is an award-winning, interactive organization game. Students are encouraged to interview people in the organization, analyze problems, drill down into and modify data flow diagrams and data dictionaries, react to prototypes, and design new input and output.
- **Entire Central Pacific University (CPU) case now online** In keeping with our belief that a variety of approaches are important, the entire Central Pacific University (CPU) case, accompanied by partially solved Student Exercises, is now available online. There is an episode to accompany each chapter of the ninth edition. The CPU case makes use of Microsoft Access, Microsoft Visio, and the popular CASE tool Visible Analyst by Visible Systems, Inc., for the sample screen shots and the student exercises. The CPU case takes students through all phases of the systems development life cycle. The CPU case has been fully tested in classrooms around the world with a variety of students over numerous terms. The case is detailed, rigorous, and rich enough to stand alone as a systems analysis and design project spanning one or two terms.
- **Student Exercises based on the online CPU case** This running case gives students an opportunity to solve problems on their own, using a variety of tools and data that users of the book can download from the Web containing Microsoft Visio, Microsoft Access, and Visible Analyst exercises specifically keyed to each chapter of the book. Partially solved problems and examples stored in Microsoft Access and Visible Analyst files allow students to develop a Web-based computer management system.

EXPANDED INSTRUCTOR SUPPLEMENTAL WEB SUPPORT

Extended support for instructors using this edition can be found at the official website located at **www.pearsonhighered.com/kendall**. Resources include:

- **Instructor's Manual**—The Instructor's Manual contains answers to problems, solutions to cases, and suggestions for approaching the subject matter.
- **Solutions to Student Exercises**—These exercises are based on the ongoing CPU case, with solutions and examples stored in Visible Analyst files and Microsoft Access files.
- **PowerPoint Presentations—**The PowerPoints feature lecture notes that highlight key text terms and concepts. Professors can customize the presentation by adding their own slides or by editing the existing ones.
- **Test Item File**—The Test Item File is an extensive set of multiple-choice, true/false, and essay-type questions for each chapter of the text. Questions are ranked according to difficulty level and referenced with page numbers from the text. The Test Item File is available in Microsoft Word format and as the computerized Prentice Hall TestGen software, with course management system conversions.
- **TestGen—**Pearson Education's test-generating software is available from **www.pearsonhighered.com/irc**. The software is PC/Mac compatible and preloaded with all the Test Item File questions. You can manually or randomly view test questions and drag and drop to create a test. You can add or modify test-bank questions as needed.
- **Image Library—**This collection of the text art is organized by chapter. This collection includes all the figures, tables, and screenshots from the book. These images can be used to enhance class lectures and PowerPoint slides.
- **CourseSmart eTextbooks Online—**CourseSmart eTextbooks were developed for students looking to save on required or recommended textbooks. Students simply select their eText by title or author and purchase immediate access to the content for the duration of the course using any major credit card. With a CourseSmart eText, students can search for specific keywords or page numbers, take notes online, print out reading assignments that incorporate lecture notes, and bookmark important passages for later review. For more information or to purchase a CourseSmart eTextbook, visit **www.coursesmart.com**

This page intentionally left blank

ACKNOWLEDGMENTS

The field of information systems was transforming astonishingly as we were writing the ninth edition of *Systems Analysis and Design*. We are thrilled that this edition is being published at the right time for us to capture many of these innovations in systems development.

One major change is the rapidly increasing use of the Web as a platform for information systems. Cloud computing will dramatically change the way that analysts approach designing systems solutions.

Another major change addressed in this edition is the emergence of smartphones and tablets as corporate platforms for IT. With the advent of BYOD (bring your own device) systems, analysts face new challenges in developing successful and secure systems that can easily traverse multiple platforms.

Throughout the book you will learn and apply numerous techniques, methods, tools, and approaches to help visually capture a system. But when the time comes to interpret what is happening in the organization and to develop meaningful information systems from the application of rules to your analysis, your training combines with creativity to produce a system that is in some ways a surprise: It is structured, yet intuitive, multilayered, and complex, in keeping with the character of the organization and uniquely reflective of you as a systems analyst and a human being.

The artist Richard Kalina, who created the colorful collage on the cover of the ninth edition, writes, "*P3 Vega* is inspired by the representation of scientific phenomena, ranging from astronomy, chemistry and physics to cybernetics and information theory. The painting is not a literal depiction, but rather an abstracted map or chart, a way of thinking and observing. *P3 Vega* is a set of interlocking connections, a network of circular nodes joined by colored lines. It feels stable but shifting—a static depiction of a changeable state. I am trying to find a visual corollary to the beauty that underlies logical systems, and to make something beautiful, hopefully, in the process." We hope that you as a student will also appreciate, through this book, the beauty that underlies logical systems.

It is, in fact, our own students who deserve recognition for this new edition because of their feedback and recommendations for improvements and requests for increased depth in certain topics. Students told us that they quickly put to use the new material on designing apps and interfaces for smartphones and tablets as well as the material on new project management techniques and cloud computing. We are indebted to their quest to continually improve their skills. We want to thank our coauthor, Allen Schmidt, who once again worked with us on the HyperCase 2.9 and *CPU Case Episodes* for all his dedication, insight, and humor during our collaboration. He is a superb human being. Our appreciation also goes to Peter Schmidt and Jason Reed for their improvements to the early HyperCase. In addition, we want to thank the other two original authors of HyperCase, Richard Baskerville and Raymond Barnes, who contributed so much.

We would like to thank our ninth edition production team, especially our executive editor, Bob Horan, whose intelligence and tranquil demeanor are always inspiring. We are also grateful to Kelly Loftus, who is our extremely capable senior project manager for MIS, for her composed competency and for her enthusiasm in keeping the project going. Ilene Kahn, our production project manager, also deserves thanks for helping us succeed in making this a robust, inclusive, and accurate revision. Their help and keen interest in our book facilitated the completion of this project in a smooth and timely manner.

We also appreciate the encouragement and support of the entire Rutgers community, including our chancellor, Wendell Prittchett, and our colleagues and staff in the School of Business–Camden and throughout all of Rutgers. They have been very enthusiastic about this edition as

Julie and Ken Kendall personally thank all of our friends in the theatre and the performing arts. Here are the Kendalls at the 2012 Tony Awards afterparty with Tony-Award winning Actor James Corden (*right*). Photo by Anita & Steve Shevett.

well as the many translations and versions of *Systems Analysis and Design* available in Spanish, Chinese, English for the Indian subcontinent, and Indonesian.

All the reviewers for the ninth edition deserve our thanks as well. Their thoughtful comments and suggestions helped to strengthen the book. They are:

Ron Davis, University of North Alabama
Chang-tseh Hsieh, University of Southern Mississippi
Sukgon Kim, Northern Illinois University
Angela Marsh, University of Arkansas–Monticello
Keng Siau, Missouri University of Science and Technology
Mead Bond Wetherbe, Jr., Texas Tech University

Many of our colleagues and friends have encouraged us through the process of writing this book. We thank them for their comments on our work. They include: Ayman Abu Hamdieh, Macedonio Alanis, Michel Avital, the Ciupeks, Roger T. Danforth, Gordon Davis, John Drozdal, EgoPo, Matt Germonprez, Nancy V. Gulick, Andy Hamingson, Blake Ives, Richard Kalina, Colleen Kelly-Lawler, Carol J. Latta, Ken and Jane Laudon, Josh Lawler, Lars Mathiassen, Joel and Bobbie Porter, Caryn Schmidt, Marc and Jill Schniederjans, Gabriel Shanks, Detmar W. Straub, Jr., the Vargos, Merrill Warkentin, Brian Warner, Jeff and Bonnie Weil, Arlene and Paul Wolfling, Brett Young, Ping Zhang and all of our friends and colleagues in The Drama League, The Actors Fund, the American Theatre Wing, The New York Marriott Marquis, the Association for Information Systems, the Decision Sciences Institute, IFIP Working Group 8.2, and all those involved in the PhD Project (founded by the KPMG Foundation), which serves minority doctoral students in information systems.

Our heartfelt thanks go to the memory of Julia A. Kendall and to the memory of Edward J. Kendall. Their belief that love, goals, and hard work are an unbeatable combination continues to infuse our every endeavor.

Systems Analysis and Design

This page intentionally left blank

PART I

Systems Analysis Fundamentals

CHAPTER 1

Systems, Roles, and Development Methodologies

Learning Objectives

Once you have mastered the material in this chapter you will be able to:

1. **Understand the need for systems analysis and design in organizations.**
2. **Realize what the many roles of a systems analyst are.**
3. **Comprehend the fundamentals of three development methodologies: SDLC, the agile approach, and object-oriented systems analysis and design.**

Organizations have long recognized the importance of managing key resources such as people and raw materials. Information has now moved to its rightful place as a key resource. Decision makers now understand that information is not just a by-product of conducting business; rather, it fuels business and can be the critical factor in determining the success or failure of a business.

To maximize the usefulness of information, a business must manage it correctly, just as it manages other resources. Managers need to understand that costs are associated with the production, distribution, security, storage, and retrieval of all information. Although information is all around us, it is not free, and its strategic use for positioning a business competitively should not be taken for granted.

The ready availability of networked computers, along with access to the Internet and the Web, has created an information explosion throughout society in general and business in particular. Managing computer-generated information differs in significant ways from handling manually produced data. Usually there is a greater quantity of computer information to administer. Costs of organizing and maintaining it can increase at alarming rates, and users often treat it less skeptically than information obtained in different ways. This chapter examines the fundamentals of different kinds of information systems, the varied roles of systems analysts, and the phases in the systems development life cycle (SDLC) as they relate to human–computer interaction (HCI) factors; it also introduces computer-aided software engineering (CASE) tools.

Need for Systems Analysis and Design

Systems analysis and design, as performed by systems analysts, seeks to understand what humans need to analyze data input or data flow systematically, process or transform data, store data, and output information in the context of a particular organization or enterprise. By doing thorough analysis, analysts seek to identify and solve the right problems. Furthermore, systems analysis and design is used to analyze, design, and implement improvements in the support of users and the functioning of businesses that can be accomplished through the use of computerized information systems.

Installing a system without proper planning leads to great user dissatisfaction and frequently causes the system to fall into disuse. Systems analysis and design lends structure to the analysis and design of information systems, a costly endeavor that might otherwise have been done in a haphazard way. It can be thought of as a series of processes systematically undertaken to improve a business through the use of computerized information systems. Systems analysis and design involves working with current and eventual users of information systems to support them in working with technologies in an organizational setting.

User involvement throughout a systems project is critical to the successful development of computerized information systems. Systems analysts, whose roles in the organization are discussed next, are the other essential component in developing useful information systems.

Users are moving to the forefront as software development teams become more international in their composition. This means that there is more emphasis on working with software users; on performing analysis of their business, problems, and objectives; and on communicating the analysis and design of the planned system to all involved.

New technologies also are driving the need for systems analysis. Ajax (Asynchronous JavaScript and XML) is not a new programming language but a technique that uses existing languages to make web pages function more like a traditional desktop application program. Systems analysts will increasingly need to build and redesign web pages that utilize Ajax technologies. New programming languages, such as the open source Web framework *Ruby on Rails* (*"Rails"* for short), which is a combination programming language and code generator for creating Web applications, will require more analysis.

Roles of a Systems Analyst

A systems analyst systematically assesses how users interact with technology and how businesses function by examining the inputting and processing of data and the outputting of information with the intent of improving organizational processes. Many improvements involve better support of users' work tasks and business functions through the use of computerized information systems. This definition emphasizes a systematic, methodical approach to analyzing—and potentially improving—what is occurring in the specific context experienced by users and created by a business.

Our definition of a systems analyst is necessarily broad. An analyst must be able to work with people of all descriptions and be experienced in working with computers. An analyst plays many roles, sometimes balancing several at the same time. The three primary roles of a systems analyst are consultant, supporting expert, and agent of change.

Systems Analyst as Consultant

A systems analyst frequently acts as a systems consultant to humans and their businesses and, thus, may be hired specifically to address information systems issues within a business. Such hiring can be an advantage because outside consultants can bring with them a fresh perspective that other people in an organization do not possess. It also means that outside analysts are at a disadvantage because an outsider can never know the true organizational culture. As an outside consultant, you will rely heavily on the systematic methods discussed throughout this text to analyze and design appropriate information systems for users working in a particular business. In addition, you will rely on information systems users to help you understand the organizational culture from others' viewpoints.

Systems Analyst as Supporting Expert

Another role that you may be required to play is that of supporting expert within a business for which you are regularly employed in some systems capacity. In this role, an analyst draws on professional expertise concerning computer hardware and software and their uses in the business.

CONSULTING OPPORTUNITY 1.1

Healthy Hiring: Ecommerce Help Wanted

"You'll be happy to know that we made a strong case to management that we should hire a new systems analyst to specialize in ecommerce development," says Al Falfa, a systems analyst for the multioutlet international chain Marathon Vitamin Shops. He is meeting with his large team of systems analysts to decide on the qualifications that their new team member should possess. Al continues, saying, "In fact, they were so excited by the possibility of our team helping to move Marathon into an ecommerce strategy that they've said we should start our search now and not wait until the fall."

Ginger Rute, another analyst, agrees, saying, "The demand for website developers is still outstripping the supply. We should move quickly. I think our new person should be knowledgeable in system modeling, JavaScript, C++, and Rational Rose and familiar with Ajax, just to name a few."

Al looks surprised at Ginger's long list of skills but then replies, "Well, that's certainly one way we could go. But I would also like to see a person with some business savvy. Most of the people coming out of school will have solid programming skills, but they should know about accounting, inventory, and distribution of goods and services, too."

The newest member of the systems analysis group, Vita Ming, finally breaks into the discussion. She says, "One of the reasons I chose to come to work with all of you was that I thought we all got along quite well together. Because I had some other opportunities, I looked very carefully at what the atmosphere was here. From what I've seen, we're a friendly group. Let's be sure to hire someone who has a good personality and who fits in well with us."

Al concurs, continuing, "Vita's right. The new person should be able to communicate well with us, and with business clients, too. We are always communicating in some way, through formal presentations, drawing diagrams, or interviewing users. If they understand decision making, it will make their job easier, too. Also, Marathon is interested in integrating ecommerce into the entire business. We need someone who at least grasps the strategic importance of the Web. Page design is such a small part of it."

Ginger interjects again with a healthy dose of practicality, saying, "Leave that to management. I still say the new person should be a good programmer." Then she ponders aloud, "I wonder how important UML will be?"

After listening patiently to everyone's wish list, one of the senior analysts, Cal Siem, speaks up, joking, "We'd better see if Superman is available!"

As the group shares a laugh, Al sees an opportunity to try for some consensus, saying, "We've had a chance to hear a number of different qualifications. Let's each take a moment and make a list of the qualifications we personally think are essential for the new ecommerce development person to possess. We'll share them and continue discussing until we can describe the person in enough detail to turn a description over to the human resources group for processing."

What qualifications should the systems analysis team be looking for when hiring their new ecommerce development team member? Is it more important to know specific languages or to have an aptitude for picking up languages and software packages quickly? How important is it that the person being hired has some basic business understanding? Should all team members possess identical competencies and skills? What personality or character traits are desirable in a systems analyst who will be working in ecommerce development?

This work is often not a full-blown systems project, but rather it entails a small modification or decision affecting a single department.

As the supporting expert, you are not managing the project; you are merely serving as a resource for those who are. If you are a systems analyst employed by a manufacturing or service organization, many of your daily activities may be encompassed by this role.

Systems Analyst as Agent of Change

The most comprehensive and responsible role that the systems analyst takes on is that of an agent of change, whether internal or external to the business. As an analyst, you are an agent of change whenever you perform any of the activities in the systems development life cycle (discussed in the next section) and are present and interacting with users and the business for an extended period (from two weeks to more than a year). An agent of change can be defined as a person who serves as a catalyst for change, develops a plan for change, and works with others in facilitating that change.

Your presence in the business changes it. As a systems analyst, you must recognize this fact and use it as a starting point for your analysis. Hence, you must interact with users and management (if they are not one and the same) from the very beginning of your project. Without their help, you cannot understand what they need to support their work in the organization, and real change cannot take place.

If change (that is, improvements to the business that can be realized through information systems) seems warranted after analysis, the next step is to develop a plan for change along with the people who must enact the change. Once a consensus is reached on the change that is to be made, you must constantly interact with those who are changing.

As a systems analyst acting as an agent of change, you advocate a particular avenue of change involving the use of information systems. You also teach users the process of change because changes in the information system do not occur independently; rather, they cause changes in the rest of the organization as well.

Qualities of a Systems Analyst

From the foregoing descriptions of the roles the systems analyst plays, it is easy to see that a successful systems analyst must possess a wide range of qualities. Many different kinds of people are systems analysts, so any description is destined to fall short in some way. There are some qualities, however, that most systems analysts seem to display.

Above all, an analyst is a problem solver. He or she is a person who views the analysis of problems as a challenge and who enjoys devising workable solutions. When necessary, an analyst must be able to systematically tackle the situation at hand through skillful application of tools, techniques, and experience. An analyst must also be a communicator capable of relating meaningfully to other people over extended periods of time. Systems analysts need to be able to understand humans' needs in interacting with technology, and they need enough computer experience to program, to understand the capabilities of computers, to glean information requirements from users, and to communicate what is needed to programmers. They also need to possess strong personal and professional ethics to help them shape their client relationships.

A systems analyst must be a self-disciplined, self-motivated individual who is able to manage and coordinate other people, as well as innumerable project resources. Systems analysis is a demanding career, but, in compensation, an ever-changing and always challenging one.

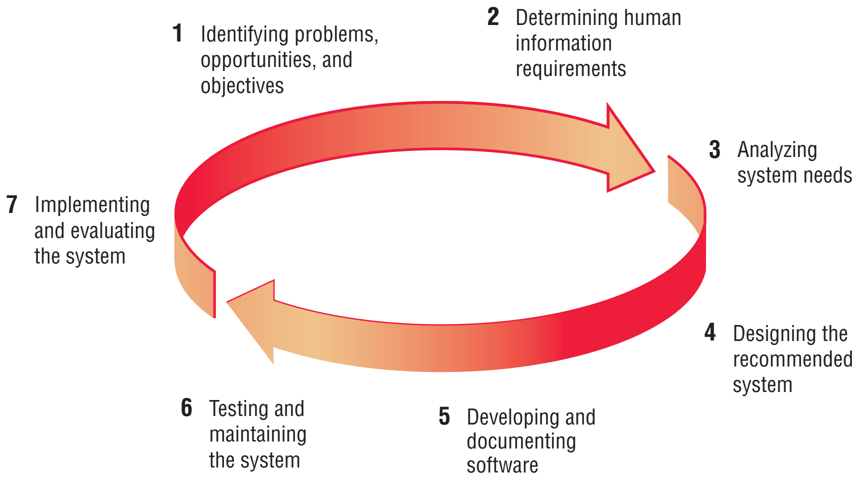
The Systems Development Life Cycle

Throughout this chapter we have referred to the systematic approach analysts take to the analysis and design of information systems. Much of this is embodied in what is called the systems development life cycle (SDLC). The SDLC is a phased approach to analysis and design which holds that systems are best developed through the use of a specific cycle of analyst and user activities.

Analysts disagree on exactly how many phases there are in the SDLC, but they generally laud its organized approach. Here we have divided the cycle into seven phases, as shown in Figure 1.1. Although each phase is presented discretely, it is never accomplished as a separate step. Instead, several activities can occur simultaneously, and activities may be repeated.

FIGURE 1.1

The seven phases of the systems development life cycle (SDLC).


1 Identifying problems, opportunities, and objectives
2 Determining human information requirements
3 Analyzing system needs
4 Designing the recommended system
5 Developing and documenting software
6 Testing and maintaining the system
7 Implementing and evaluating the system

Incorporating Human–Computer Interaction Considerations

In recent years, the study of human–computer interaction (HCI) has become increasingly important for systems analysts. Although the definition is still evolving, researchers characterize HCI as the "aspect of a computer that enables communications and interactions between humans and the computer. It is the layer of the computer that is between humans and the computer" (Zhang, Carey, Te'eni, & Tremaine, 2005, p. 518). Analysts using an HCI approach are emphasizing people rather than the work to be done or the IT that is involved. Their approach to a problem is multifaceted, looking at the "human ergonomic, cognitive, affective, and behavioral factors involved in user tasks, problem solving processes and interaction context" (Zhang, Carey, Te'eni, & Tremaine, 2005, p. 518). HCI moves away from focusing first on organizational and system needs and instead concentrates on human needs. Analysts adopting HCI principles examine a variety of user needs in the context of humans interacting with information technology to complete tasks and solve problems. These include taking into account physical or ergonomic factors; usability factors that are often labeled cognitive matters; the pleasing, aesthetic, and enjoyable aspects of using the system; and behavioral aspects that center on the usefulness of the system.

Another way to think about HCI is to think of it as a human-centered approach that puts people ahead of organizational structure or culture when creating new systems. When analysts employ HCI as a lens to filter the world, their work will possess a different quality than the work of those who do not possess this perspective.

Your career can benefit from a strong grounding in HCI. The demand for analysts who are capable of incorporating HCI into the systems development process keeps rising, as companies increasingly realize that the quality of systems and the quality of work life can both be improved by taking a human-centered approach at the outset of a project.

The application of human–computer interaction principles tries to uncover and address the frustrations that users voice over their use of information technology. These concerns include a suspicion that systems analysts misunderstand the work being done, the tasks involved, and how they can best be supported; a feeling of helplessness or lack of control when working with the system; intentional breaches of privacy; trouble navigating through system screens and menus; and a general mismatch between the system designed and the way users themselves think of their work processes.

Misjudgments and errors in design that cause users to neglect new systems or that cause systems to fall into disuse soon after their implementation can be eradicated or minimized when systems analysts adopt an HCI approach.

Researchers in HCI see advantages to the inclusion of HCI in every phase of the SDLC. This is a worthwhile approach, and we will try to mirror it by bringing human concerns explicitly into each phase of the SDLC. As a person who is learning systems analysis, you can also bring a fresh eye to the SDLC to identify opportunities for designers to address HCI concerns and ways for users to become more central to each phase of the SDLC. Chapter 14 is devoted to examining the role of the systems analyst in designing human-centered systems and interfaces from an HCI perspective.

Identifying Problems, Opportunities, and Objectives

In this first phase of the systems development life cycle, an analyst is concerned with correctly identifying problems, opportunities, and objectives. This stage is critical to the success of the rest of the project because no one wants to waste subsequent time addressing the wrong problem.

The first phase requires that the analyst look honestly at what is occurring in a business. Then, together with other organizational members, the analyst pinpoints problems. Often others will bring up these problems, and they are the reason the analyst was initially called in. Opportunities are situations that the analyst believes can be improved through the use of computerized information systems. Seizing opportunities may allow the business to gain a competitive edge or set an industry standard.

Identifying objectives is also an important component of the first phase. The analyst must first discover what the business is trying to do. Then the analyst will be able to see whether some aspect of information systems applications can help the business reach its objectives by addressing specific problems or opportunities.

The people involved in the first phase are the users, analysts, and systems managers coordinating the project. Activities in this phase consist of interviewing user management, summarizing the knowledge obtained, estimating the scope of the project, and documenting the results. The output of this phase is a feasibility report that contains a problem definition and summarizes the objectives. Management must then make a decision on whether to proceed with the proposed project. If the user group does not have sufficient funds in its budget or if it wishes to tackle unrelated problems, or if the problems do not require a computer system, a different solution may be recommended, and the systems project does not proceed any further.

Determining Human Information Requirements

The next phase the analyst enters involves determining the human needs of the users involved, using a variety of tools to understand how users interact in the work context with their current information systems. The analyst will use interactive methods such as interviewing, sampling and investigating hard data, and using questionnaires, along with unobtrusive methods, such as observing decision makers' behavior and their office environments, and all-encompassing methods, such as prototyping.

The analyst will use these methods to pose and answer many questions concerning HCI, including questions such as, "What are the users' physical strengths and limitations?" In other words, "What needs to be done to make the system audible, legible, and safe?" "How can the new system be designed to be easy to use, learn, and remember?" "How can the system be made pleasing or even fun to use?" "How can the system support a user's individual work tasks and make them more productive in new ways?"

In the information requirements phase of the SDLC, the analyst is striving to understand what information users need to perform their jobs. At this point, the analyst is examining how to make the system useful to the people involved. How can the system better support individual tasks that need to be done? What new tasks are enabled by the new system that users were unable to do without it? How can the new system be created to extend a user's capabilities beyond what the old system provided? How can the analyst create a system that is rewarding for workers to use?

The people involved in this phase are the analysts and users, typically operations managers and operations workers. The systems analyst needs to know the details of current system functions: the who (the people who are involved), what (the business activity), where (the environment in which the work takes place), when (the timing), and how (how the current procedures are performed) of the business under study. The analyst must then ask why the business uses the current system. There may be good reasons for doing business using the current methods, and these should be considered when designing any new system.

Agile development is an object-oriented approach (OOA) to systems development that includes a method of development (including generating information requirements) as well as software tools. In this text, it is paired with prototyping in Chapter 6. (There is more about object-oriented approaches in Chapter 10.)

If the reason for current operations is that "it's always been done that way," however, the analyst may wish to improve on the procedures. At the completion of this phase, the analyst should understand how users accomplish their work when interacting with a computer and begin to know how to make the new system more useful and usable. The analyst should also know how the business functions and have complete information on the people, goals, data, and procedures involved.

Analyzing System Needs

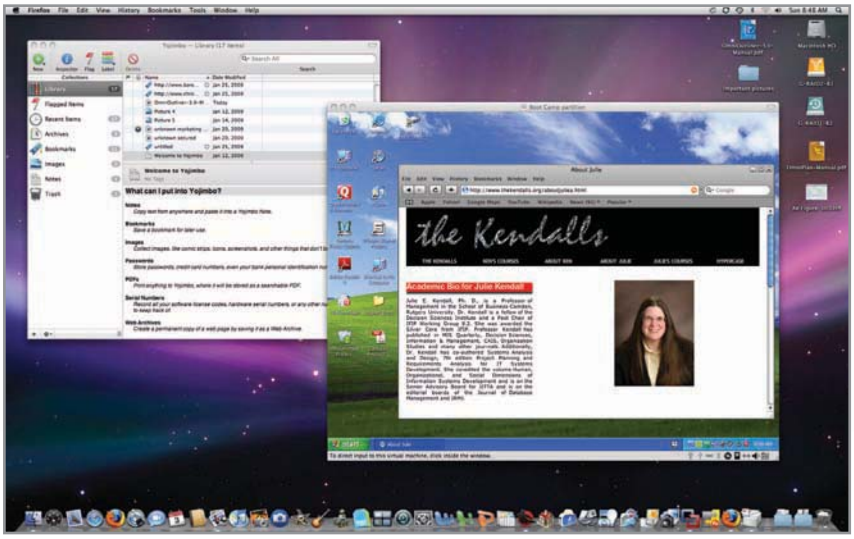
The next phase that the systems analyst undertakes involves analyzing system needs. Again, special tools and techniques help the analyst make requirement determinations. Tools such as data flow diagrams (DFDs) to chart the input, processes, and output of the business's functions, or activity diagrams or sequence diagrams to show the sequence of events, illustrate systems in a structured, graphical form. From data flow, sequence, or other diagrams, a data dictionary is developed that lists all the data items used in the system, as well as their specifications.

During this phase the systems analyst also analyzes the structured decisions made. Structured decisions are those for which the conditions, condition alternatives, actions, and action rules can be determined. There are three major tools for analyzing structured decisions: structured English, decision tables, and decision trees.

MAC APPEAL

At home and in our visits to university campuses and businesses around the world, we've noticed that students and organizations are increasingly showing an interest in Macs. Therefore, we thought it would add a little bit of interest to show some of the Mac options available to a systems designer. At the time we're writing this book, about one out of seven personal computers purchased in the United States is a Mac. Macs are quality Intel-based machines that run under a competent operating system and can also run Windows, so in effect, everything that can be done on a PC can be done on a Mac. One way to run Windows is to boot directly into Windows (once it's installed); another is to use virtualization, using software such as VM Fusion, which is shown in Figure 1.MAC.

Adopters of Macs have cited many reasons for using Macs, including better security built into the Mac operating system, intelligent backup using the built-in Time Machine, the multitude of applications already included, the reliability of setup and networking, and the ability to sync Macs with other Macs and iPhones. The most compelling reason, we think, is the design itself.



Julie E. Kendall, Ph. D., is a Professor of Management in the School of Business-Camden, Rutgers University. Dr. Kendall is a fellow of the Decision Sciences Institute and a Past Chair of IFIP Working Group 8.2. She was awarded the Silver Core from IFIP. Professor Kendall has published in MIS Quarterly, Decision Sciences, Information & Management, CAIS, Organization Studies and many other journals. Additionally, Dr. Kendall has co-authored Systems Analysis and Design, 7th edition Project Planning and Requirements Analysis for IT Systems Development. She co-edited the volume Human, Organizational, and Social Dimensions of Information Systems Development and is on the Senior Advisory Board for JITTA and is on the editorial boards of the Journal of Database Management and IRMJ.

FIGURE 1.MAC

Running Windows on a Mac using virtualization called VM Fusion. (Screenshot of Apple Desktop. Reprinted with permission of Apple Inc.; Screenshot from YOJIMBO. Copyright © 2012 by Bare Bones Software, Inc. Reprinted by permission; Screenshot from www.thekendalls.org. Copyright © by Kenneth and Julie Kendall. Reprinted with permission.)

At this point in the SDLC, the systems analyst prepares a systems proposal that summarizes what has been found out about the users, usability, and usefulness of current systems; provides cost–benefit analyses of alternatives; and makes recommendations on what (if anything) should be done. If one of the recommendations is acceptable to management, the analyst proceeds along that course. Each systems problem is unique, and there is never just one correct solution. The manner in which a recommendation or solution is formulated depends on the individual qualities and professional training of each analyst and the analyst's interaction with users in the context of their work environment.

Designing the Recommended System

In the design phase of the SDLC, the systems analyst uses the information collected earlier to accomplish the logical design of the information system. The analyst designs procedures for users to help them accurately enter data so that data going into the information system are correct. In addition, the analyst provides for users to complete effective input to the information system by using techniques of good form and web page or screen design.

Part of the logical design of the information system is devising the HCI. The interface connects the user with the system and is thus extremely important. The user interface is designed with the help of users to make sure that the system is audible, legible, and safe, as well as attractive and enjoyable to use. Examples of physical user interfaces include a keyboard (to type in questions and answers), onscreen menus (to elicit user commands), and a variety of graphical user interfaces (GUIs) that use a mouse or touch screen.

The design phase also includes designing databases that will store much of the data needed by decision makers in the organization. Users benefit from a well-organized database that is logical to them and corresponds to the way they view their work. In this phase the analyst also works with users to design output (either onscreen or printed) that meets their information needs.

Finally, the analyst must design controls and backup procedures to protect the system and the data, and to produce program specification packets for programmers. Each packet should contain input and output layouts, file specifications, and processing details; it may also include decision trees or tables, Unified Modeling Language (UML) or data flow diagrams, and the names and functions of any prewritten code that is either written in-house or using code or other class libraries.

Developing and Documenting Software

In the fifth phase of the SDLC, the analyst works with programmers to develop any original software that is needed. During this phase the analyst works with users to develop effective documentation for software, including procedure manuals, online help, and websites featuring frequently asked questions (FAQs) or Read Me files shipped with new software. Because users are involved from the beginning, phase documentation should address the questions they have raised and solved jointly with the analyst. Documentation tells users how to use software and what to do if software problems occur.

Programmers have a key role in this phase because they design, code, and remove syntactical errors from computer programs. To ensure quality, a programmer may conduct either a design or a code walkthrough, explaining complex portions of the program to a team of other programmers.

Testing and Maintaining the System

Before an information system can be used, it must be tested. It is much less costly to catch problems before the system is signed over to users than after. Some of the testing is completed by programmers alone, some of it by systems analysts in conjunction with programmers. A series of tests to pinpoint problems is run first with sample data and eventually with actual data from the current system. Often test plans are created early in the SDLC and are refined as the project progresses.

Maintenance of the system and its documentation begins in this phase and is carried out routinely throughout the life of the information system. Much of the programmer's routine work consists of maintenance, and businesses spend a great deal of money on maintenance. Some maintenance, such as program updates, can be done automatically via a vendor site on the Web. Many of the systematic procedures the analyst employs throughout the SDLC can help ensure that maintenance is kept to a minimum.

Implementing and Evaluating the System

In this last phase of systems development, the analyst helps implement the information system. This phase involves training users to handle the system. Vendors do some training, but oversight of training is the responsibility of the systems analyst. In addition, the analyst needs to plan for a smooth conversion from the old system to the new one. This process includes converting files from old formats to new ones or building a database, installing equipment, and bringing the new system into production.

Evaluation is included as part of this final phase of the SDLC mostly for the sake of discussion. Actually, evaluation takes place during every phase. A key criterion that must be satisfied is whether the intended users are indeed using the system.

It should be noted that systems work is often cyclical. When an analyst finishes one phase of systems development and proceeds to the next, the discovery of a problem may force the analyst to return to the previous phase and modify the work done there.

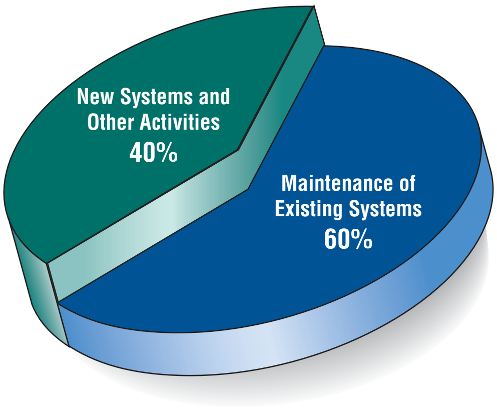


FIGURE 1.2

Some researchers estimate that the amount of time spent on system maintenance may be as much as 60 percent of the total time spent on systems projects.

The Impact of Maintenance

After the system is installed, it must be maintained, meaning that the computer programs must be modified and kept up to date. Figure 1.2 illustrates the average amount of time spent on maintenance at a typical MIS installation. Estimates of the time spent by departments on maintenance have ranged from 48 to 60 percent of the total time spent developing systems. Very little time remains for new systems development. As the number of programs written increases, so does the amount of maintenance they require.

Maintenance is performed for two reasons. The first of these is to correct software errors. No matter how thoroughly a system is tested, bugs or errors creep into computer programs. Bugs in commercial PC software are often documented as "known anomalies," and they are corrected when new versions of the software are released or in an interim release. In custom software (also called *bespoke software*), bugs must be corrected as they are detected.

The other reason for performing system maintenance is to enhance the software's capabilities in response to changing organizational needs, generally involving one of the following three situations:

1. Users often request additional features after they become familiar with the computer system and its capabilities.
2. The business changes over time.
3. Hardware and software are changing at an accelerated pace.

Figure 1.3 illustrates the amount of resources—usually time and money—spent on systems development and maintenance. The area under the curve represents the total dollar amount spent. You can see that over time, the total cost of maintenance is likely to exceed that of systems development. At a certain point it becomes more feasible to perform a new systems study because the cost of continued maintenance is clearly greater than the cost of creating an entirely new information system.

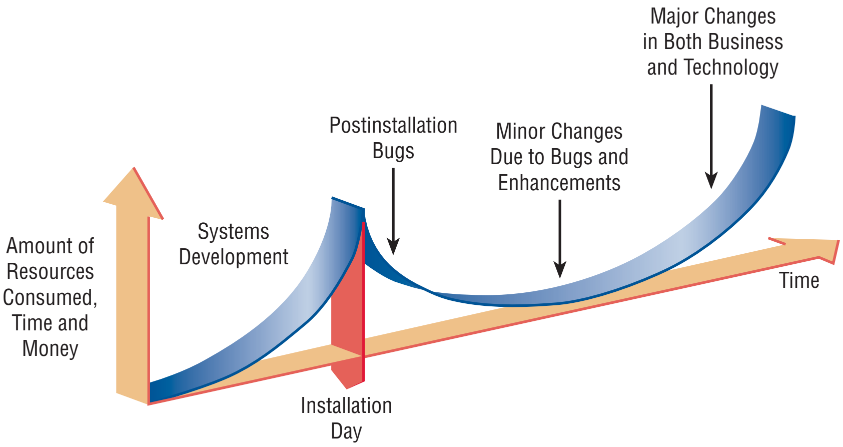

Major Changes in Both Business and Technology
Postinstallation Bugs
Minor Changes Due to Bugs and Enhancements
Systems Development
Amount of Resources Consumed, Time and Money
Time
Installation Day

FIGURE 1.3

Resource consumption over the system life.