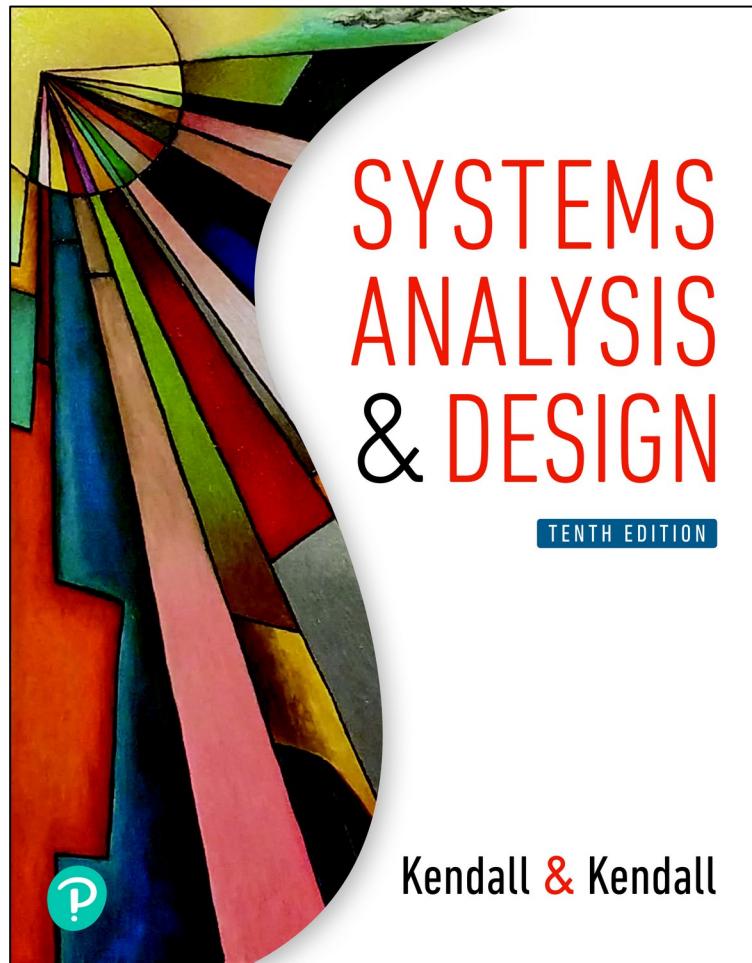


Systems Analysis & Design

Tenth Edition



Chapter 1

Systems, Roles, and
Development
Methodologies

Learning Objectives

- 1.1** Understand the need for systems analysis and design in organizations
- 1.2** Realize what the many roles of the systems analyst are
- 1.3** Comprehend the fundamentals of three development methodologies:
 - 1.3.1** The systems development life cycle (SDLC)
 - 1.3.2** The agile approach including Scrum
 - 1.3.3** Object-oriented systems analysis and design

Information – A Key Resource

- Fuels business and can be the critical factor in determining the success or failure of a business
- Needs to be managed correctly
- Managing computer-generated information differs from handling manually produced data

Major Topics

- Fundamentals of different kinds of information systems
- Roles of systems analysts
- Phases in the systems development life cycle as they relate to Human-Computer Interaction (HCI) factors
- Agile development
- Object-oriented Systems Analysis and Design

Need for Systems Analysis and Design

- 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
- User involvement throughout a systems project is critical to the successful development of computerized systems

Roles of the Systems Analyst

- The analyst must be able to work with people of all descriptions and be experienced in working with computers
- Three primary roles:
 - Consultant
 - Supporting Expert
 - Agent of change

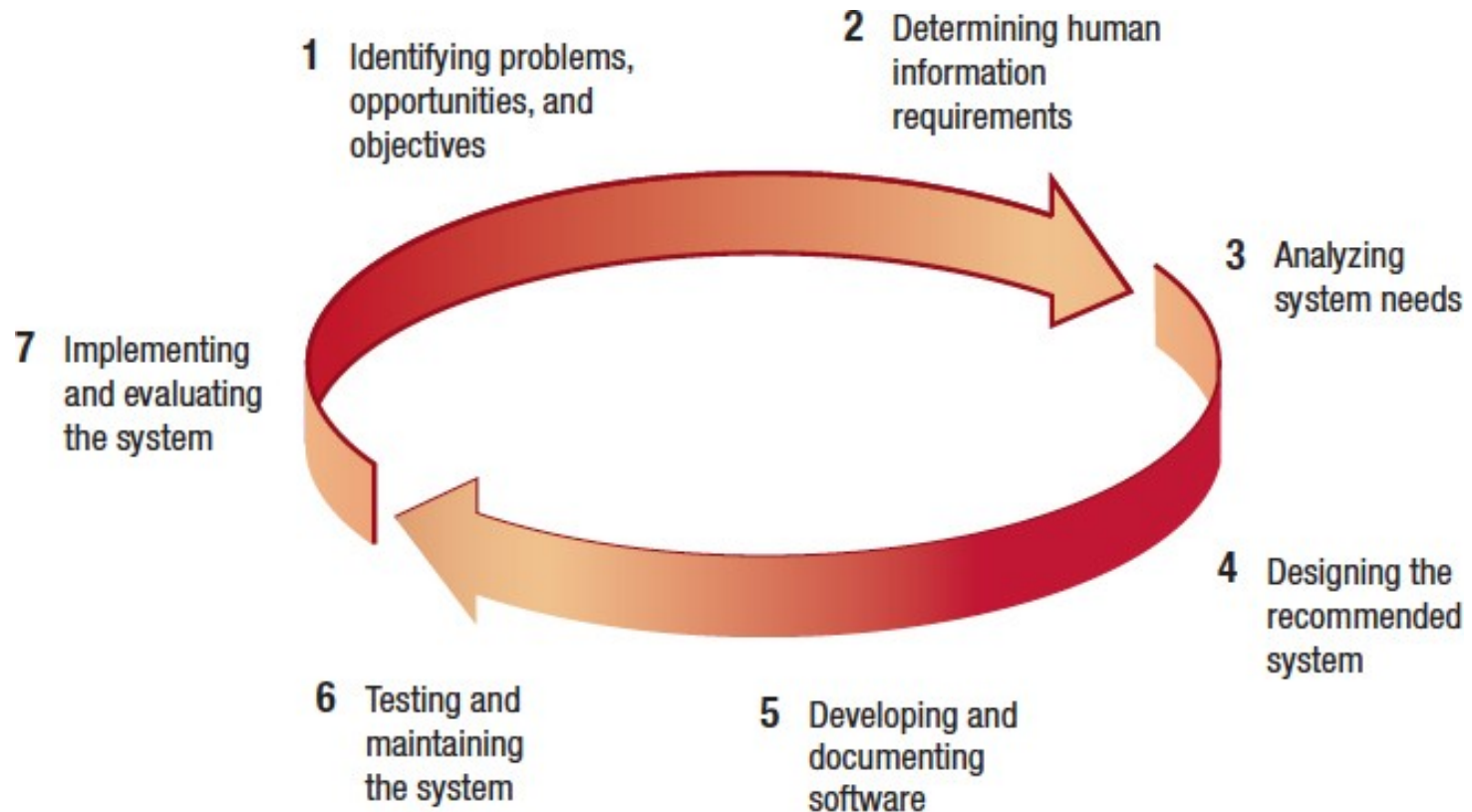
Qualities of the Systems Analyst

- Problem solver
- Communicator
- Strong personal and professional ethics
- Self-disciplined and self-motivated

Systems Development Life Cycle (SDLC)

- The systems development life cycle is a phased approach to solving business problems
- Developed through the use of a specific cycle of analyst and user activities
- Each phase has unique user activities

Figure 1.1 The Seven Phases of the Systems Development Life Cycle



Identifying Problems, Opportunities, and Objectives

- Activity:
 - Interviewing user management
 - Summarizing the knowledge obtained
 - Estimating the scope of the project
 - Documenting the results
- Output:
 - Feasibility report containing problem definition and objective summaries from which management can make a decision on whether to proceed with the proposed project

Determining Human Information Requirements (1 of 2)

- Activity:
 - Interviewing
 - Sampling and investigating hard data
 - Questionnaires
 - Observe the decision maker's behavior and environment
 - Prototyping
 - Learn the who, what, where, when, how, and why of the current system

Determining Human Information Requirements (2 of 2)

- Output:
 - Analyst understands how users accomplish their work when interacting with a computer
 - Begin to know how to make the new system more useful and usable
 - The analyst should also know the business functions
 - Have complete information on the people, goals, data and procedure involved

Analyzing System Needs

- Activity:
 - Create data flow diagrams
 - Complete the data dictionary
 - Analyze the structured decisions made
 - Prepare and present the system proposal
- Output:
 - Recommendation on what, if anything, should be done

Designing the Recommended System

- Activity:
 - Design procedures to accurately enter data
 - Design the human–computer interface (HCI)
 - Design files and/or database
 - Design backup procedures
- Output
 - Model of the actual system

Developing and Documenting Software

- Activity:
 - System analyst works with programmers to develop any original software
 - Works with users to develop effective documentation
 - Coders design, code, and remove syntactical errors from computer programs
 - Document software with procedure manuals, online help, Frequently Asked Questions (FAQ) and Read Me files
- Output:
 - Computer programs
 - System documentation

Testing and Maintaining the System

- Activity:
 - Test the information system
 - System maintenance
 - Maintenance documentation
- Output:
 - Problems, if any
 - Updated programs
 - Documentation

Implementing and Evaluating the System

- Activity:
 - Train users
 - Plans the conversion from old system to new system
 - Review and evaluate system
- Output:
 - Trained personnel
 - Installed system

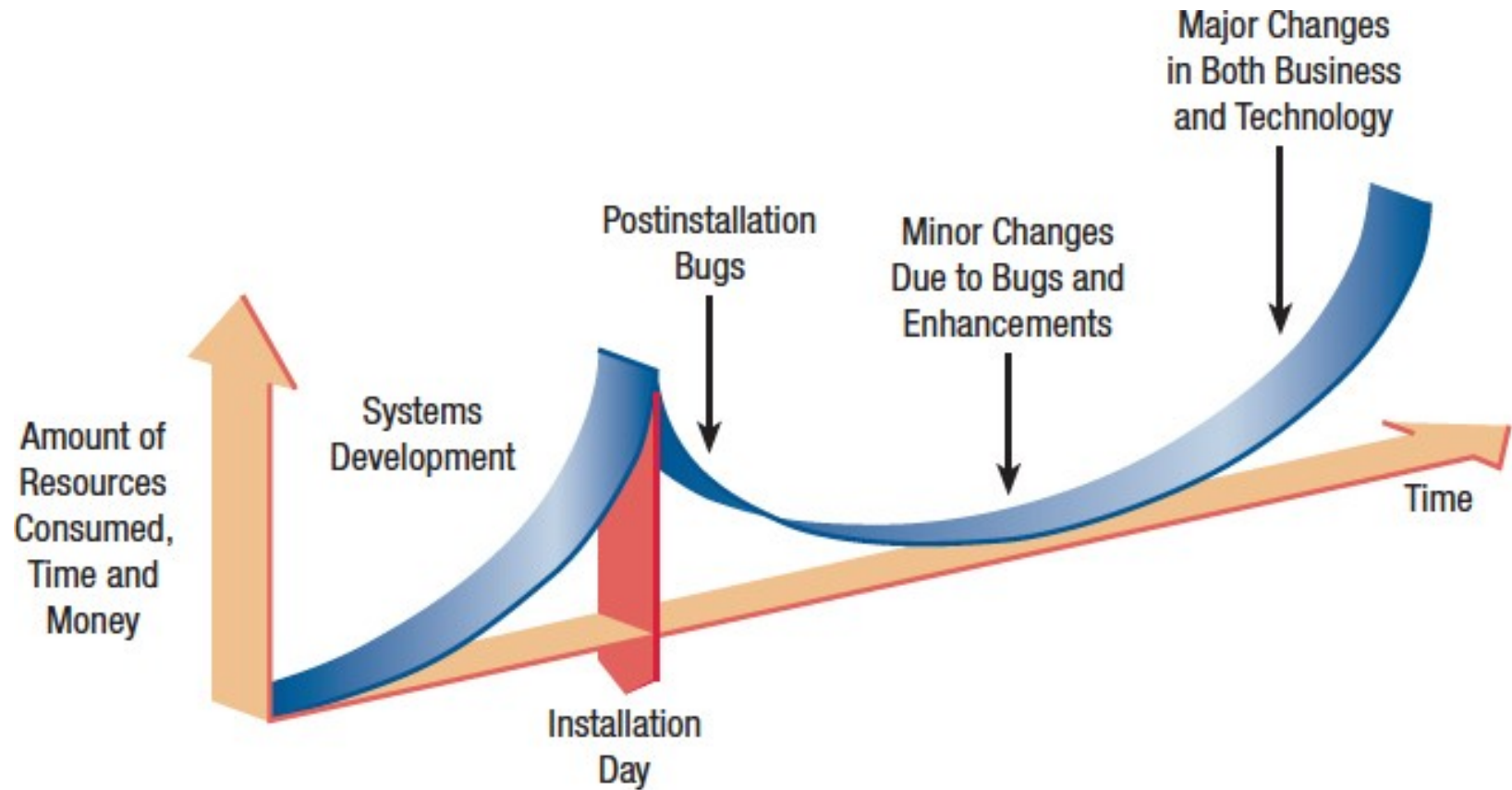
The Impact of Maintenance

- Maintenance is performed for two reasons
 - Remove software errors, and
 - Enhance existing software
- Enhance software for three reasons
 - Include additional features
 - Address business changes over time
 - Address hardware and software changes

Maintenance Impact

- Over time the cost of continued maintenance will be greater than that of creating an entirely new system
- At that point it becomes more feasible to perform a new systems study

Figure 1.2 Resource Consumption Over the System Life



Case Tools

- CASE (computer aided software engineering) tools are productivity tools for systems analysts that have been created explicitly to improve their routine work through the use of automated support
- Reasons for using CASE tools
 - Improving Analyst–User Communication
 - Help support modeling functional requirements
 - Assist in drawing project boundaries

The Agile Approach

- The agile approach is a software development approach based on
 - Values
 - Principles
 - Core practices

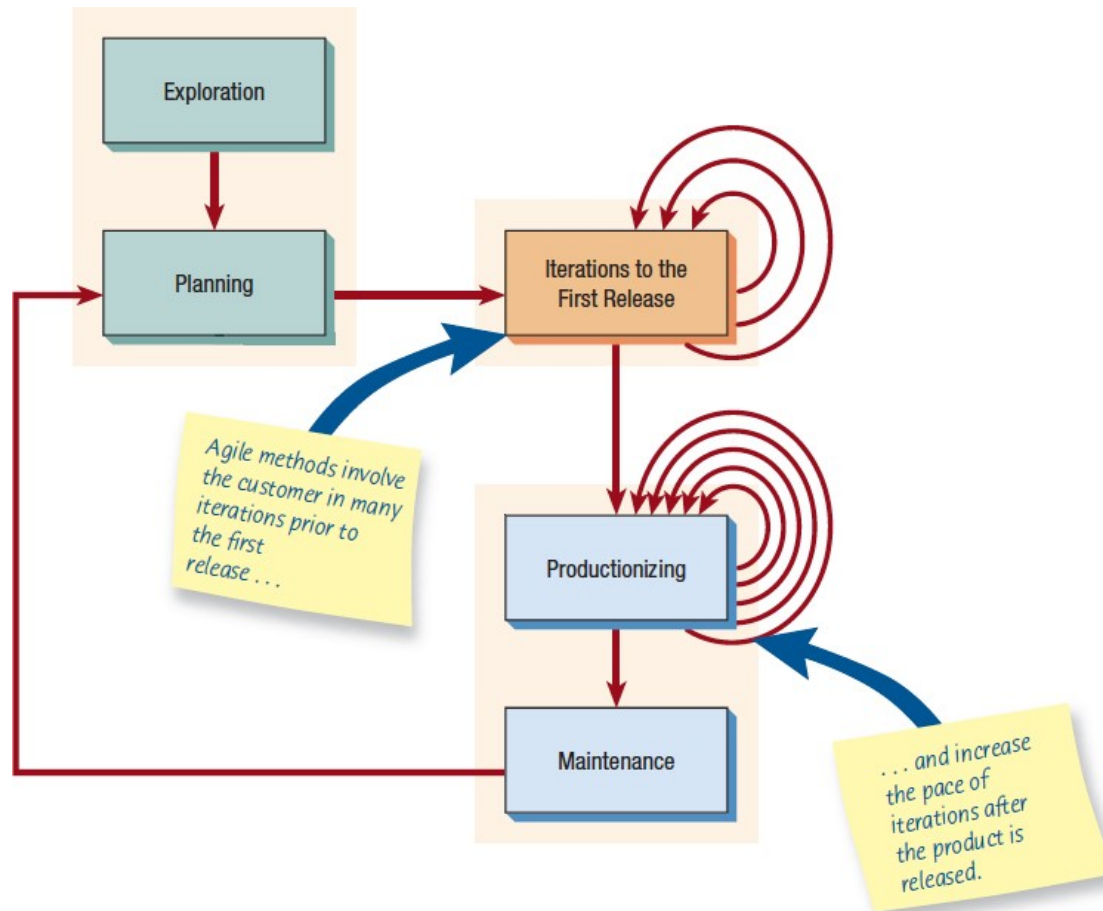
Agile Values

- The four values are
 - Communication
 - Simplicity
 - Feedback
 - Courage

Agile Approach

- Agile approach is
 - Interactive
 - Incremental
- Frequent iterations are essential for successful system development

Figure 1.4 The Five Stages of the Agile Modeling Development Process



Exploration

- Assemble team
- Assess skills
- Examine potential technologies
- Experiment with writing user stories
- Adopt a playful and curious attitude toward the work environment, its problems, technologies, and people

Planning

- Rules that can help formulate the agile development team's relationship with their business customers
- Maximize the value of the system produced by the agile team
- Main players are the development team and the business customer

Iterations to the First Release

- Iterations are cycles of
 - Testing
 - Feedback
 - Change
- One goal is to run customer-written function tests at the end of each iteration

Productionizing

- The product is released in this phase
- May be improved by adding other features

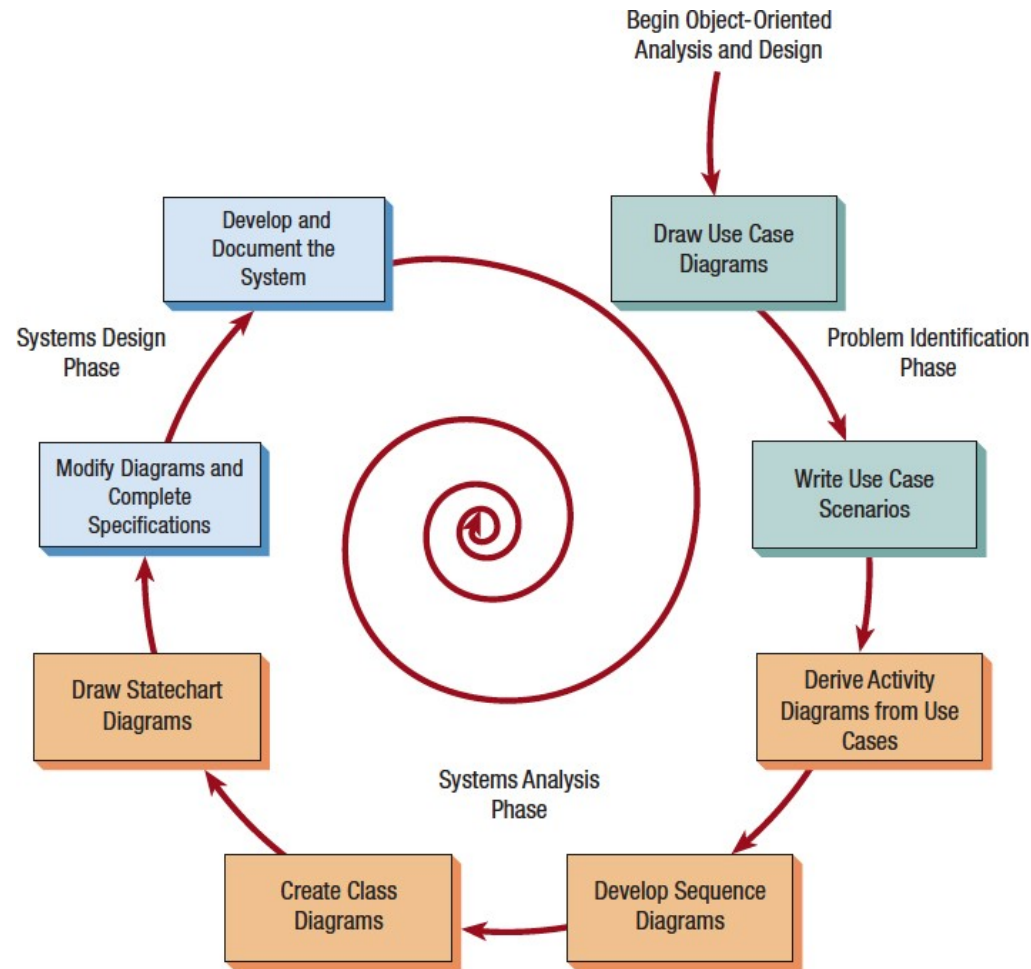
Maintenance

- New features may be added
- Riskier customer suggestions may be considered
- Team members may be rotated on or off the team

Object-Oriented Systems Analysis and Design

- Alternate approach to the structured approach of the SDLC that is intended to facilitate the development of systems that must change rapidly in response to dynamic business environments
- Use unified modeling language (UML) to model object-oriented systems
- Each object is a computer representation of some actual thing or event

Figure 1.5 The Steps in the UML Development Process



Define the Use Case Model

- Identify the actors and the major events initiated by the actors
- Draw a use case diagram
- A diagram with stick figures representing the actors and arrows showing how the actors relate

Begin Drawing UML Diagrams

- Draw activity diagrams, which illustrate all the major activities in the use case
- Create one or more sequence diagrams for each use case that show the sequence of activities and their timing
- Review the use cases, rethink them, and modify them if necessary

Analysis Phase

- Develop class diagrams
- Draw statechart diagrams

System Design

- Modifying the existing system
- Modifying the diagrams drawn in the previous phase
- Write class specifications for each class

Develop and Document the System

- The more complete the information you provide to the development team through documentation and UML diagrams, the faster the development and the more solid the final production system

Figure 1.6 How to Decide Which Development Method to Use

Choose	When
The Systems Development Life Cycle (SDLC) Approach	<ul style="list-style-type: none">systems have been developed and documented using SDLCit is important to document each step of the wayupper-level management feels more comfortable or safe using SDLCthere are adequate resources and time to complete the full SDLCcommunication of how new systems work is important
Agile Methodologies	<ul style="list-style-type: none">there is a project champion of agile methods in the organizationapplications need to be developed quickly in response to a dynamic environmenta rescue takes place (the system failed and there is no time to figure out what went wrong)the customer is satisfied with incremental improvementsexecutives and analysts agree with the principles of agile methodologies
Object-Oriented Methodologies	<ul style="list-style-type: none">the problems modeled lend themselves to classesan organization supports the UML learningsystems can be added gradually, one subsystem at a timereuse of previously written software is a possibilityit is acceptable to tackle the difficult problems first

Open Source Software (1 of 3)

- An alternative to traditional software development
- Many users and coders can study, share, and modify the code

Open Source Software (2 of 3)

- Categorized open source communities into four community types
 - Ad hoc
 - Standardized
 - Organized
 - Commercial

Open Source Software (3 of 3)

- Six different dimensions
 - General structure
 - Environment
 - Goals
 - Methods
 - User community
 - Licensing

Summary

- Systematic approach to identifying problems
- Systems analysts are required to take on many roles
- Analysts possess a wide range of skills
- The systems Development Life Cycle
- Agile approach
- Object-oriented analysis and design

Copyright



This work is protected by United States copyright laws and is provided solely for the use of instructors in teaching their courses and assessing student learning. Dissemination or sale of any part of this work (including on the World Wide Web) will destroy the integrity of the work and is not permitted. The work and materials from it should never be made available to students except by instructors using the accompanying text in their classes. All recipients of this work are expected to abide by these restrictions and to honor the intended pedagogical purposes and the needs of other instructors who rely on these materials.