PART 2

Systems Analysis Activities

Chapter **2**
Investigating System Requirements

Chapter **3**
Use Cases

Chapter **4**
Domain Modeling

Chapter **5**
Extending the Requirements Models

Optional Online Chapter **B**
The Traditional Approach to Requirements

2

Investigating System Requirements

Chapter Outline

- The RMO Consolidated Sales and Marketing System Project
- Systems Analysis Activities
- What Are Requirements?
- Models and Modeling
- Stakeholders
- Information-Gathering Techniques
- Documenting Workflows with Activity Diagrams

Learning Objectives

After reading this chapter, you should be able to:

- Describe the activities of systems analysis
- Explain the difference between functional and nonfunctional requirements
- Describe the role of models in systems analysis
- Identify and understand different kinds of stakeholders and their contributions to requirements definition
- Describe information-gathering techniques and determine when each is best applied
- Develop activity diagrams to model workflows

OPENING CASE

Mountain Vista Motorcycles

Amanda Lamy, president and majority stockholder of Mountain Vista Motorcycles (MVM), is an avid motorcycle enthusiast and businesswoman. MVM is headquartered in Denver and has locations throughout the western United States and Canada. Since the late-1990s, the market for motorcycles has grown tremendously. Amanda expects that the market will continue to be strong throughout the 2010s, although she is concerned about the "graying" of a significant portion of MVM's customer base.

The demographics of the motorcycle market are an interesting study in contrasts. At present, the majority of customers are over 50 years of age, male professionals or businesspeople, and partly or fully retired. They have substantial disposable income, lots of free time, and tend to own multiple expensive motorcycles from such manufacturers as Harley-Davidson, Honda, Ducati, and Moto Guzi. Older customers are generally comfortable with Internet and Web technology but are not significant users of social media technology. Although many own smartphones, they tend to use them primarily for voice, e-mail, and texting.

Male customers under 30 years of age tend to buy sport and dirt bikes, typically from such manufacturers as Suzuki and Kawasaki. They buy less expensive bikes than older customers and are more likely to buy parts and supplies from MVM to service their own bikes. Female customers under 30 years of age tend to buy motor scooters and smaller "commuter" motorcycles. Customers in the 30–50 age range include men and women who buy bikes of many types from many manufacturers. Comfort with and use of Internet technology, social media, and portable computing devices such as smartphones and iPads is very high with customers under 50 years of age, especially with customers under age 30.

Amanda is convinced that the key to long-term success in the motorcycle market is to build an active community of motorcycle enthusiasts at each MVM location that includes a broad spectrum of customers. In essence, each location needs to be seen as a hub of local motorcycle-related activity and information in physical and virtual terms. On the physical side, MVM has added activity and event-oriented pages to its Web sites, sponsored rallies and clubs, added meeting rooms and small coffee shops in some locations, and colocated with bars and restaurants that feature motorcycle-related themes and entertainment. These efforts have yielded good results with older customers but less so with younger customers.

Amanda is concerned about the lack of participation by younger customers and is sure that MVM's lack of presence in social media and virtual relationships is a significant factor. She and her senior staff, most of whom are older, are unsure how to attract younger customers. They have little knowledge of and no experience creating modern technology-based virtual communities.

MVM's chief information officer is starting to develop a project plan for a virtual community oriented toward younger customers. If the plan were for developing a traditional information system, she would use such standard approaches as interviewing internal users and managers and having her development staff write specifications, generate storyboards and screen layouts, and develop prototypes. But few of the intended virtual community users are MVM employees, and none of her staff members fully comprehends how to successfully use social media and other techniques for building virtual societies. Traditional methods of defining and refining requirements seem inadequate to the task.

Overview

In Chapter 1, you saw the System Development Life Cycle (SDLC) being employed by Ridgeline Mountain Outfitters (RMO) for a small information system application called the Tradeshow System. Development of that system followed the six core processes of the SDLC:

1. Identify the problem or need and obtain approval to proceed.
2. Plan and monitor the project—what to do, how to do it, and who does it.
3. Discover and understand the details of the problem or the need.
4. Design the system components that solve the problem or satisfy the need.
5. Build, test, and integrate system components.
6. Complete system tests and then deploy the solution.

In this chapter, we start expanding the scope and detail of the SDLC processes to cover a wider range of concepts, tools, and techniques. The extra depth and detail are needed to tackle larger and more complex projects. This chapter concentrates on systems analysis activities (the third core process listed), and the next few chapters follow up on that with detailed discussions of models developed during those systems analysis activities. Subsequent chapters expand the discussion of other core SDLC processes.

The RMO Consolidated Sales and Marketing System Project

Ridgeline Mountain Outfitters has an elaborate set of information systems applications developed over the years to support operations and management. However, there is a growing gap between customer expectations, modern technological capabilities, and existing RMO systems that support sales and customer interaction. This section reviews the existing system inventory and introduces the proposed Consolidated Sales and Marketing System that will update and enhance sales and marketing.

Existing RMO Information Systems and Architecture

RMO's Information Systems Department has always been forward looking. In past years, the department, in conjunction with corporate strategic plans, has developed five-year plans for development and deployment of new technology and information systems. The planning process has been an excellent tool to help the organization stay current with new trends and technology capabilities. However, the plans themselves have had mixed success. One of the complexities with long-range IT and software development plans is that technology changes rapidly and moves in unexpected directions. For example, the Tradeshow System described in Chapter 1 was made possible by the availability of powerful and flexible handheld devices and the widespread availability of Wi-Fi and Internet connections. Both technologies reached a fairly mature level in just a couple of years, which created an opportunity for RMO to optimize this important business process.

Historically, RMO has adopted new technology as soon as it became cost-effective. Past examples include adoption of smaller servers and desktop computing, interconnection of locations with dedicated telecommunications links, and such Web-based technologies as customer-oriented Web sites and browser-based user interfaces for internal systems. At present, RMO has a disparate collection of computers dispersed across home offices, retail stores, telephone centers, order fulfillment/shipping centers, and warehouses—everything connected by a complex set of local area networks (LANs), wide area networks (WANs), and virtual private networks (VPNs). The term **technology architecture** describes the set of computing hardware, network hardware and topology, and system software—such as operating and database management systems—employed by an organization. RMO's technology architecture is modern but not state of the art, which is consistent with its goal of adopting only cost-effective technology.

technology architecture a set of computing hardware, network hardware and topology, and system software employed by an organization

The term **application architecture** describes how software resources are organized and constructed to implement an organization's information systems. It describes the organization of software into modules and subsystems and includes supporting technologies (such as programming languages and development environments), architectural approaches (such as service-oriented architecture), and user-interface technologies (such as mobile computing displays, touch screen technology, and voice recognition).

application architecture the organization and construction of software resources to implement an organization's information systems

Currently, the major RMO systems consist of:

- Supply Chain Management (SCM)—This application was deployed five years ago as a client/server application using Java and Oracle. Currently, it supports inventory control, purchasing, and distribution, although integration of functions needs improvement. The new Tradeshow System will interface with this system.
- Phone/Mail Order System—A modest client/server application developed 12 years ago by using Visual Studio and Microsoft SQL Server as a quick

solution to customer demand for catalog phone and mail orders. It is integrated with the SCM and has reached capacity.

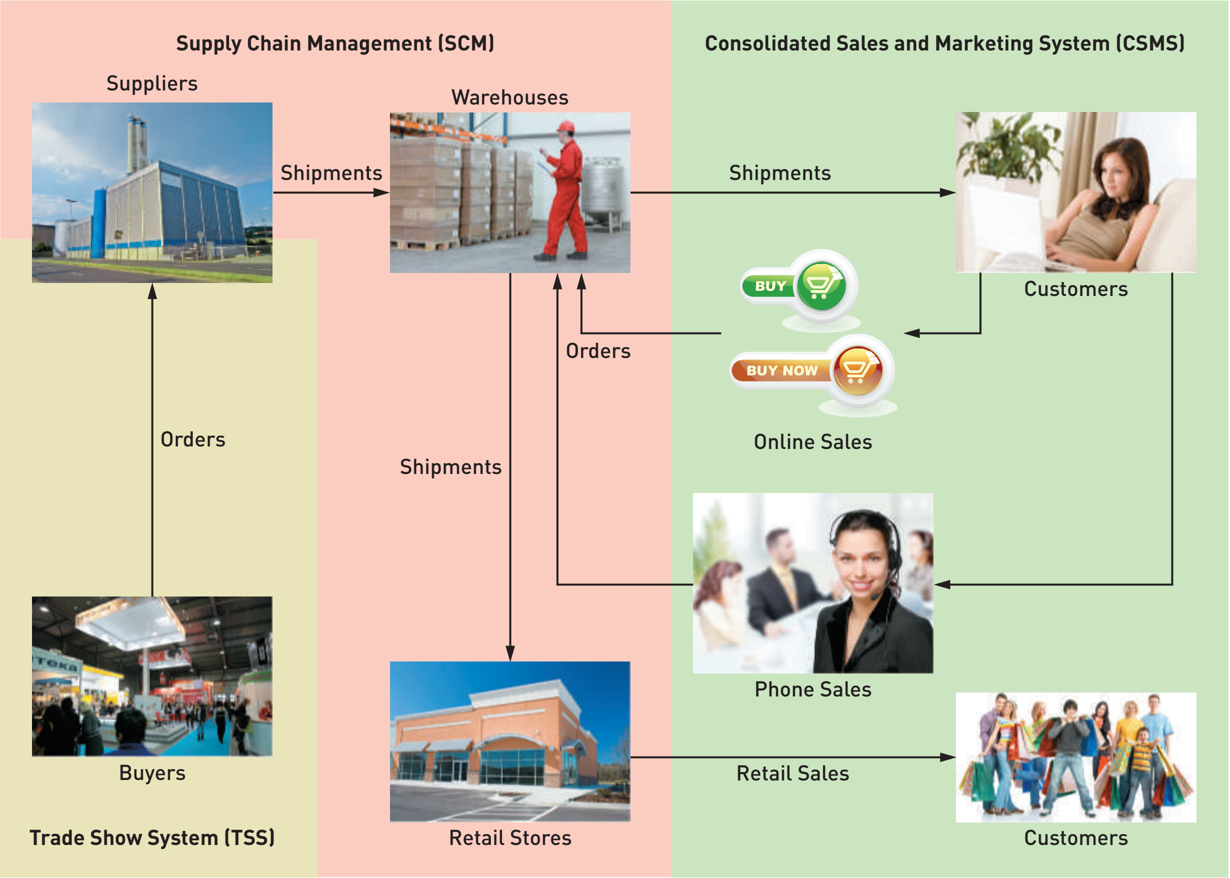
- Retail Store System (RSS)—A retail store package with point-of-sale processing. It was upgraded eight years ago from overnight batch to real-time inventory updates to/from the SCM.
- Customer Support System (CSS)—This system was first deployed 15 years ago as a Web-based catalog to support customer mail and phone orders. Four years later, it was upgraded to an Internet storefront, supporting customer inquiries, shopping cart, order tracking, shipping, back orders, and returns.

All organizations—including RMO—face a difficult challenge keeping all their information systems current and effective. Because development resources are limited, an organization's technology and application architecture and its information system inventory will include a mix of old and new. Older systems were often designed for outdated operational methods and typically lack modern technologies and features that some competitors have adopted to improve efficiency or competitiveness. Such is the case with RMO's existing customer-facing systems, which have several shortcomings, including:

- Treating phone, Web, and retail sales as separate systems rather than as an integrated whole
- Employing outdated Web-based storefront technology
- Not supporting modern technologies and customer interaction modes, including mobile computing devices and social networking

Rather than incrementally update the existing sales systems, RMO plans to replace them, as shown in **Figure 2-1**.

FIGURE 2-1 *Proposed application architecture for RMO*



The New Consolidated Sales and Marketing System

The goals of the Consolidated Sales and Marketing System (CSMS) are to modernize the technology and functionality of the CSS and to add more customer-oriented functionality. On the technology side, the CSMS will incorporate current Web standards and be built under the assumption of high-bandwidth customer Internet connections and high-resolution displays. Updating the technology will enable a higher degree of interactivity, richer graphics, and a streamlined interface.

Key additions to system functionality will be support for mobile computing devices, incorporation of customer feedback and comments into product information, and integration of social networking functions. Unlike the CSS, the CSMS will support smartphones and tablet computers with interfaces specifically designed for each platform and with downloadable apps. Customer feedback will be captured directly through the Internet storefront, from RMO-supported comment forums and blogs, and mined from Facebook and Twitter. RMO will develop a complete presence in each social networking venue and enable system users to share purchases, recommendations, coupons, and store credits using those venues.

The new CSMS will also have four subsystems:

- The Sales subsystem provides such basic functions as searching the online catalog and purchasing items and paying for them online. However, it has many new capabilities to assist the shopper making purchases. The system will provide specific suggestions about accessories that go with the purchased item. Images and videos of animated models will be available to help the customer see how various items and accessory packages will look together. The system will also provide information to shoppers about related purchases made by other shoppers. Customer ratings and comments are available for viewing. Finally, key social networking components will permit shoppers to network with their friends by sending messages to ask their opinions about particular merchandise items.
- The Order Fulfillment subsystem will perform all the normal tasks of shipping items and allowing customers to track the status of their orders as well as the shipments. In addition, as part of order fulfillment, customers can rate and make comments about particular merchandise and their overall shopping experience. They may also make suggestions directly to RMO about the attractiveness of the Web site and the quality of the service they received.
- The Customer Account subsystem provides all those services that enhance the customer experience. Customers can view and maintain their account information. They also can "link up" with friends who are also customers to share experiences and opinions on merchandise. The system will keep track of detailed shipping addresses as well as payment preferences and information. RMO also instituted a Mountain Bucks program wherein customers accumulate credits that can be used to redeem prizes as well as purchase merchandise. Customers may use these Mountain Bucks for themselves or they may transfer them to other people in their family/friends group. This is a great opportunity to combine accumulated bucks to obtain expensive merchandise.
- The Marketing subsystem is primarily for employees to set up the information and services for customers. In this subsystem, employees can enter information about all the merchandise offered by RMO. This subsystem is also fed by the SCM system to maintain accurate data on the inventory in stock and anticipated arrival dates of items on order. Employees also set up the various promotional packages and seasonal catalogs by using the functions of this subsystem. RMO is experimenting with a new idea to enhance customer satisfaction: It is building partner relationships with other retailers so that customers can earn "combined" points with RMO purchases or a partner retailer purchase. These points can be used at RMO or transferred and used at the partner. For example, because RMO sells outdoor and sporting clothing, it has partnered with various sporting goods stores.

That way, a person can buy sporting equipment at the sporting equipment store and get promotional discounts for clothing at RMO. The success of this new venture has yet to be proven, but RMO anticipates that it will enhance the shopping experience of all its customers.

In later chapters, more details will be given about the capabilities of the new CSMS system. Detailed explanations will also describe the information that must be maintained in the database to support these functions.

Systems Analysis Activities

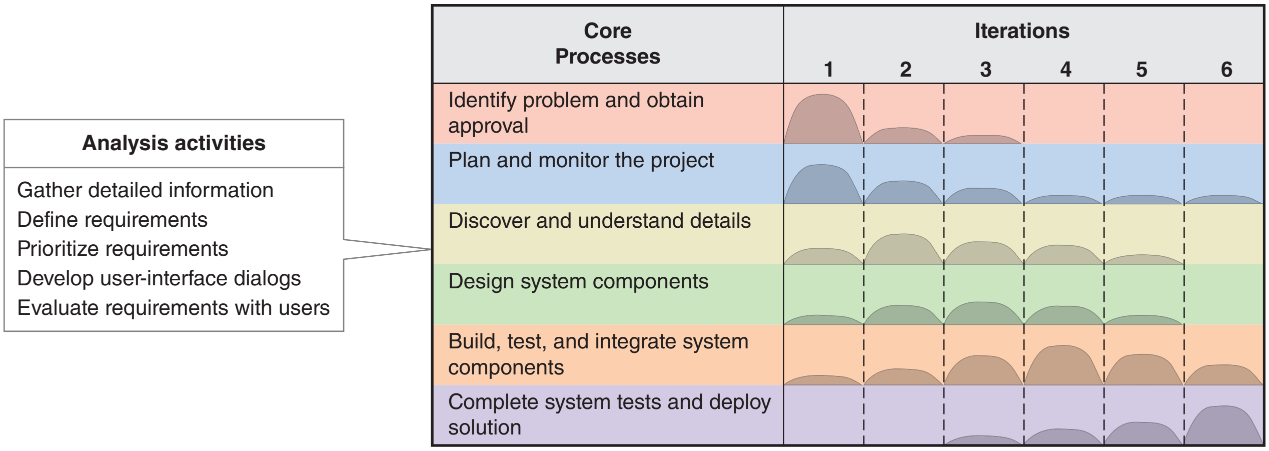
The callout on the left side of **Figure 2-2** lists the activities of the third core process, which is to discover and understand the details. This core process also goes by the name *systems analysis*. By completing these activities, the analyst defines in great detail what the information system needs to accomplish to provide the organization with the desired benefits. In essence, analysis activities are a second and more thorough pass at defining the problem and need. The first pass generates only enough detail to decide whether a new or upgraded system is warranted and feasible. The second (analysis) pass assumes that the organization is committed to the project. Thus, considerably more time and resources are invested to produce a much more detailed description of what the system will do.

Although we concentrate only on analysis activities in this chapter, keep in mind that they are usually intermixed with design, implementation, and other activities during the system development life cycle. For example, as shown in Figure 2-2, analysis activities are most intensive in the second iteration but occur in varying degrees during all project iterations except the last. This pattern is typical because an analyst will often concentrate on one part of a system, defining requirements only for that part and simultaneously designing and implementing software to satisfy those requirements. Organizing development activities in this iterative manner enables development to be broken into smaller steps and enables users to validate requirements by testing and observing a functional system. The following sections briefly describe analysis activities, and the remainder of this chapter expands the discussion of information gathering and defining system requirements.

Gather Detailed Information

Systems analysts obtain information from people who will be using the system, either by interviewing them or by watching them work. They obtain additional information by reviewing planning documents and policy statements. Analysts also study existing systems, including their documentation. They also frequently obtain additional information by looking at what other companies (particularly vendors) have done when faced with a similar business need. They try to

FIGURE 2-2 *Analysis activities*


Analysis activities
Gather detailed information
Define requirements
Prioritize requirements
Develop user-interface dialogs
Evaluate requirements with users
Core Processes
Iterations
1
2
3
4
5
6
Identify problem and obtain approval
Plan and monitor the project
Discover and understand details
Design system components
Build, test, and integrate system components
Complete system tests and deploy solution

understand an existing system by identifying and understanding the activities of all the current and future users and by identifying all the present and future locations where work occurs and all the system interfaces with other systems, both inside and outside the organization. In short, analysts need to talk to nearly everyone who will use the new system or has used similar systems, and they must read nearly everything available about the existing system. Later in this chapter, we discuss how to identify and extract information from all these people.

Beginning analysts often underestimate how much there is to learn about the work the user performs. The analyst must become an expert in the business area the system will support. For example, if you are implementing an order-entry system, you need to become an expert on the way orders are processed (including related accounting procedures). If you are implementing a loan-processing system, you need to become an expert on the rules used for approving credit. If you work for a bank, you need to think of yourself as a banker. The most successful analysts become experts in their organization's main business.

Define Requirements

The analyst uses information gathered from users and documents to define requirements for the new system. System requirements include the functions the system must perform (functional requirements) and such related issues as user-interface formats and requirements for reliability, performance, and security (nonfunctional requirements). We further discuss the distinction between functional and nonfunctional requirements a bit later in this chapter.

Defining requirements is not just a matter of writing down facts and figures. Instead, the analyst creates models to record requirements, reviews the models with users and others, and refines and expands the models to reflect new or updated information. Building and refining requirements models occupies much of the analyst's time. This chapter and the next two chapters explain in considerable depth how to create requirements models.

Prioritize Requirements

Once the system requirements are well understood, it is important to establish which requirements are most crucial for the system. Sometimes, users suggest additional system functions that are desirable but not essential. However, users and analysts need to ask themselves which functions are truly important and which are fairly important but not absolutely required. Again, an analyst who understands the organization and the work done by the users will have more insight toward answering these questions.

Why prioritize the functions requested by the users? Resources are always limited, and the analyst must always be prepared to justify the scope of the system. Therefore, it is important to know what is absolutely required. Unless the analyst carefully evaluates priorities, system requirements tend to expand as users make more suggestions (a phenomenon called *scope creep*). Requirements priorities also help to determine the number, composition, and ordering of project iterations. High-priority requirements are often incorporated into early project iterations so analysts and users have ample opportunity to refine those parts of the system. Also, a project with many high-priority requirements will typically have many iterations.

Develop User-Interface Dialogs

When a new system is replacing an old system that does similar work, users are usually quite sure about their requirements and the desired form of the user interface. But many system development projects break new ground by automating functions that were previously performed manually or by implementing functions that were not performed in the past. In either case, users tend to be uncertain about many aspects of system requirements. Such requirements models as use cases, activity diagrams, and interaction diagrams can be developed based on user input, but it is often difficult for users to interpret and validate such abstract models.

In comparison, user validation of an interface is much simpler and more reliable because the user can see and feel the system. To most users, the user interface is all that matters. Thus, developing user-interface dialogs is a powerful method of eliciting and documenting requirements. Analysts can develop user interfaces via abstract models, such as storyboards (covered in more detail in Chapter 7), or they can develop user-interface prototypes on the actual input/output devices that users will use (e.g., a computer monitor, iPad, or cell phone). A prototype interface can serve as a requirement and a starting point for developing a portion of the system. In other words, a user-interface prototype developed in an early iteration can be expanded in later iterations to become a fully functioning part of the system.

Evaluate Requirements with Users

Ideally, the activities of evaluating requirements with users and documenting the requirements are fully integrated. But in practice, users generally have other responsibilities besides developing a new system. Thus, analysts usually use an iterative process in which they elicit user input, work alone to model requirements, return to the user for additional input or validation, and then work alone to incorporate the new input and refine the models. Prototypes of user interfaces and other parts of the system may also be developed when "paper" models are inadequate or when users and analysts need to prove that chosen technologies will do what they are supposed to do. Also, if the system will include new or innovative technology, the users may need help visualizing the possibilities available from the new technology when defining what they require. Prototypes can fill that need. The processes of eliciting requirements, building models and prototypes, and evaluating them with users may repeat many times until requirements models and prototypes are complete and accurate.

What Are Requirements?

As you can see from the previous section, requirements and models that represent them are a key focus of analysis phase activities. Most of the analyst's time is devoted to requirements: gathering information about them, formalizing them by using models and prototypes, refining and expanding them, prioritizing them, and generating and evaluating alternatives. But to fully understand those activities, you need to answer a fundamental question: What are requirements?

system requirements the activities a system must perform or support and the constraints that the system must meet

functional requirements the activities that the system must perform

System requirements are all the activities the new system must perform or support and the constraints that the new system must meet. Generally, analysts divide system requirements into two categories: functional and nonfunctional requirements. **Functional requirements** are the activities that the system must perform (i.e., the business uses to which the system will be applied). For example, if you are developing a payroll system, the required business uses might include such functions as "generate electronic fund transfers," "calculate commission amounts," "calculate payroll taxes," "maintain employee-dependent information," and "report tax deductions to the IRS." The new system must handle all these functions. Identifying and describing all these business uses require a substantial amount of time and effort because the list of functions and their relationships can be very complex.

Functional requirements are based on the procedures and rules that the organization uses to run its business. Sometimes, they are well documented and easy to identify and describe. An example might be the following: "All new employees must fill out a W-4 form to enter information about their tax withholding in the payroll system." Other business rules might be more obtuse or difficult to find. An example from RMO might be the following: "Air shipping charges are reduced by 50 percent for orders over $200 that weigh less than two pounds." Discovering such rules is critical to the final design of the system. If this rule were not discovered, customers that had relied on it in the past might become angry. Modifying the system after customers start complaining would be much more difficult and expensive than building in the rule from the start.

FIGURE 2-3
FURPS+

Requirement categories	FURPS + categories	Example requirements
Functional	Functions	Business rules and processes
Nonfunctional	Usability	User interface, ease of use
	Reliability	Failure rate, recovery methods
	Performance	Response time, throughput
	Security	Access controls, encryption
	+ Design constraints	Hardware and support software
	Implementation	Development tools, protocols
	Interface	Data interchange formats
	Physical	Size, weight, power consumption
	Support	Installation and updates

nonfunctional requirements system characteristics other than the activities it must perform or support

FURPS a requirements classification framework (acronym stands for functionality, usability, reliability, performance, and security)

Nonfunctional requirements are characteristics of the system other than those activities it must perform or support. It is not always easy to distinguish functional from nonfunctional requirements. One way to do so is to use a framework for identifying and classifying requirements. There have been many such frameworks developed over time; the most widely used today is called FURPS+ (see **Figure 2-3**). **FURPS** is an acronym that stands for functionality, usability, reliability, performance, and security. The "F" in FURPS is equivalent to the functional requirements defined previously. The remaining FURPS categories describe nonfunctional requirements:

usability requirements operational characteristics related to users, such as the user interface, related work procedures, online help, and documentation

reliability requirements requirements that describe system dependability

performance requirements operational characteristics related to measures of workload, such as throughput and response time

security requirements requirements that describe how access to the application will be controlled and how data will be protected during storage and transmission

- **Usability requirements** describe operational characteristics related to users, such as the user interface, related work procedures, online help, and documentation. For example, the user interface for a smartphone app should behave similarly to other apps when responding to such gestures as two-finger slides, pinching, and expanding. Additional requirements might include menu format, color schemes, use of the organization's logo, and multingual support.
- **Reliability requirements** describe the dependability of a system—how often a system exhibits such behaviors as service outages and incorrect processing and how it detects and recovers from those problems.
- **Performance requirements** describe operational characteristics related to measures of workload, such as throughput and response time. For example, the client portion of a system might be required to have a one-half-second response time to all button presses, and the server might need to support 100 simultaneous client sessions (with the same response time).
- **Security requirements** describe how access to the application will be controlled and how data will be protected during storage and transmission. For example, the application might be password protected, encrypt locally stored data with 1024-bit keys, and use secure HTTP for communication among client and server nodes.

FURPS+ an extension of FURPS that includes design constraints as well as implementation, interface, physical, and supportability requirements

FURPS+ is an extension of FURPS that adds additional categories, including design constraints as well as implementation, interface, physical, and supportability requirements—all these additional categories summarized by the plus sign. Here are short descriptions of each category:

design constraints restrictions to which the hardware and software must adhere

- **Design constraints** describe restrictions to which the hardware and software must adhere. For example, a cell phone application might be required to use the Android operating system, consume no more than 30MB of flash memory storage, consume no more than 10MB of system memory while running, and operate on CPUs rated at 1 GHz or higher.

implementation requirements constraints such as required programming languages and tools, documentation method and level of detail, and a specific communication protocol for distributed components

- **Implementation requirements** describe constraints such as required programming languages and tools, documentation method and level of detail, and a specific communication protocol for distributed components.

interface requirements required interactions among systems

- **Interface requirements** describe interactions among systems. For example, a financial reporting system for a publicly traded company in the United States must generate data for the Securities and Exchange Commission (SEC) in a specific XML format. The system might also supply data directly to stock exchanges and bond rating agencies and automatically generate Twitter messages, RSS feeds, and Facebook updates.

physical requirements characteristics of hardware such as size, weight, power consumption, and operating conditions

- **Physical requirements** describe such characteristics of hardware as size, weight, power consumption, and operating conditions. For example, a system that supports battlefield communications might have such requirements as weighing less than 200 grams, being no larger than 5 centimeters cubed, and operating for 48 hours on a fully charged 1200 milliwatt lithium ion battery.

supportability requirements how a system is installed, configured, monitored, and updated

- **Supportability requirements** describe how a system is installed, configured, monitored, and updated. For example, requirements for a game installed on a home PC might include automatic configuration to maximize performance on existing hardware, error reporting, and download of updates from a support server.

As with any set of requirements categories, FURPS+ has some gray areas and some overlaps among its categories. For example, is a requirement that a battlefield communications device survive immersion in water and operate across a temperature range of –20°C to 50°C a performance or physical requirement? Is a restriction to use no more than 100 MB of memory a performance or design requirement? Is a requirement to secure communication between workstations and servers with 1024-bit encryption a performance, design, or implementation requirement? The answers to such questions are not important. What is important is that all requirements be identified and precisely stated early in the development process and that inconsistencies or trade-offs among them be resolved.

Models and Modeling

Modeling is an important part of analysis and design. Analysts build models to describe system requirements and use those models to communicate with users and designers. By developing a model and reviewing it with a user, an analyst demonstrates an understanding of the user's requirements. If the user spots errors or omissions, they are incorporated into the model before it becomes the basis for subsequent design and implementation activities. **Figure 2-4** summarizes the key reasons for building and using models.

Designers construct high-level and detailed models to describe system components and their interactions. Design models serve as a scratch pad for evaluating design alternatives and as a way to communicate the final design to programmers, vendors, and others who will build, acquire, and assemble components to create the final system. In general, models built during one SDLC activity are "consumed" during other activities.

model representation of some aspect of a system

A **model** is a representation of some aspect of the system being built. There are dozens of different models that an analyst or designer might develop and use (see **Figure 2-5**). Although this book emphasizes models and techniques for creating models, it is important to remember that system projects vary in the number of models required and in their formality. Smaller, simpler system projects will not need models showing every system detail, particularly when

FIGURE 2-4
Reasons for modeling

- Learning from the modeling process
- Reducing complexity by abstraction
- Remembering all the details
- Communicating with other development team members
- Communicating with a variety of users and stakeholders
- Documenting what was done for future maintenance/enhancement

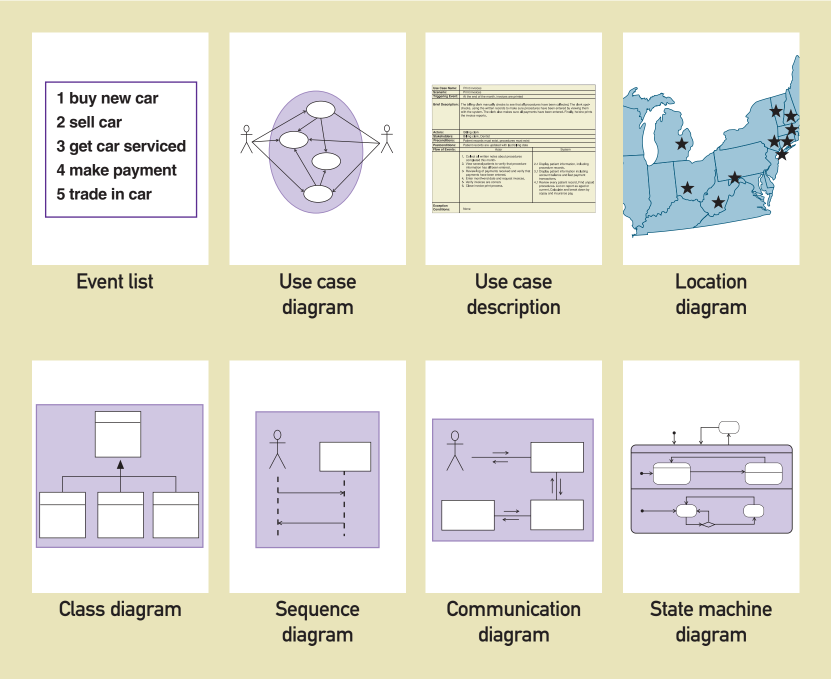
the project team has experience with the type of system being built. Sometimes, the key models are created informally in a few hours. Although models are often created by using specialized software tools, useful and important models are sometimes drawn quickly over lunch on a paper napkin or in an airport waiting room on the back of an envelope! As with any development activity, an iterative approach is used for creating models. The first draft of a model has some but not all details worked out. The next iteration might fill in more details or correct previous misconceptions.

Analysis and design models can be grouped into three generic types:

textual models text-based system models such as memos, reports, narratives, and lists

- **Textual models**—Analysts use such textual models as memos, reports, narratives, and lists to describe requirements that are detailed and are difficult to represent in other ways. The event list in Figure 2-5 is one example of a textual model. Narrative description is often the best way to initially record information gathered verbally from stakeholders, such as during an interview. In many cases, narratives and other textual models are later converted into a graphical format.

FIGURE 2-5
Some analysis and design models



graphical models system models that use pictures and other graphical elements

- **Graphical models**—Graphical models make it easier to understand complex relationships that are difficult to follow when described as a list or narrative. Recall the old saying that a picture is worth a thousand words. In system development, a carefully constructed graphical model might be worth a million words! Some graphical models actually look similar to a real-world part of the system, such as a screen design or a report layout design. However, the graphical models developed during analysis activities typically represent more abstract things, such as external agents, processes, data, objects, messages, and connections.

mathematical models system models that describes requirements numerically or as mathematical expressions

- **Mathematical models**—Mathematical models are one or more formulas that describe technical aspects of a system. Analysts often use mathematical models to represent functional requirements for scientific and engineering applications and occasionally use them to describe business system requirements in areas such as accounting and inventory control. Analysts and designers use mathematical models to describe requirements and operational parameters such as network throughput or database query response time.

Unified Modeling Language (UML) standard set of model constructs and notations defined by the Object Management Group

Many graphical models used in system development are drawn according to the notation specified by the **Unified Modeling Language (UML)**. In Figure 2-5, the use case diagram, class diagram, sequence diagram, communication diagram, and state machine diagram are UML graphical models. UML is the standard set of model constructs and notations defined by the Object Management Group (OMG), a standards organization for system development. By using UML, analysts and end users are able to depict and understand a variety of specific diagrams used in a system development project. Prior to UML, there was no standard, so diagrams could be confusing, and they varied from company to company (and from book to book).

We expand the discussion of models later in this chapter with a detailed look at one type of graphical model: the workflow diagram. In later chapters, you learn how to develop many of other types of analysis and design models.

Stakeholders

stakeholders persons who have an interest in the successful implementation of the system

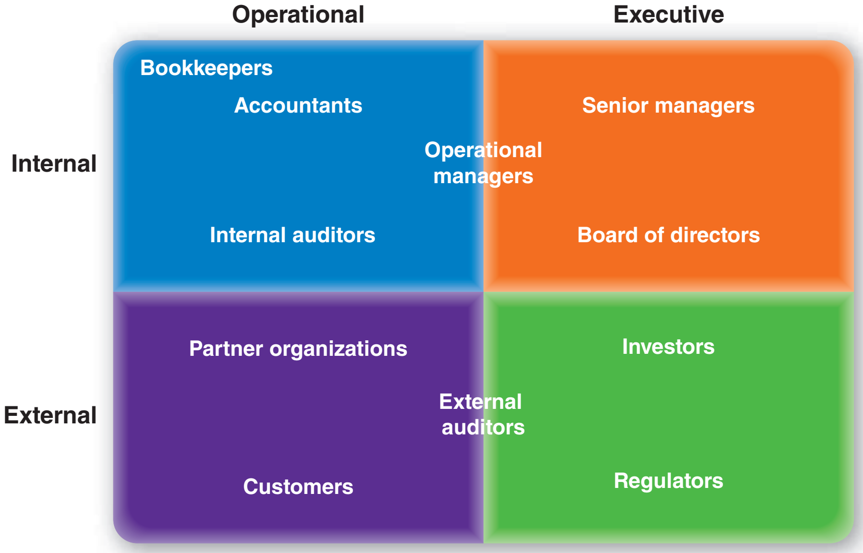
Stakeholders are your primary source of information for system requirements. **Stakeholders** are all the people who have an interest in the successful implementation of the system. Depending on the nature and scope of the system, this can be a small or a large, diverse group. For example, when implementing a comprehensive accounting system for a publicly traded corporation in the United States, the stakeholders include bookkeepers, accountants, managers and executives, customers, suppliers, auditors, investors, the SEC, and the Internal Revenue Service (IRS). Each stakeholder group interacts with the system in different ways, and each has a unique perspective on system requirements. Before gathering detailed information, the analyst identifies every type of stakeholder who has an interest in the new system and ensures that critical people from each stakeholder category are available to serve as the business experts.

internal stakeholders persons within the organization who interact with the system or have a significant interest in its operation or success

external stakeholders persons outside the organization's control and influence who interact with the system or have a significant interest in its operation or success

One useful way to help identify all the interested stakeholders is to consider two characteristics by which they vary: internal stakeholders versus external stakeholders and operational stakeholders versus executive stakeholders (see **Figure 2-6**). **Internal stakeholders** are those within the organization who interact with the system or have a significant interest in its operation or success. You may be tempted to define internal stakeholders as employees of an organization, but some organizations—such as nonprofits and educational institutions—have internal users (e.g., volunteers and students) who are not employees. **External stakeholders** are those outside the organization's control and influence, although this distinction can also be fuzzy, such as when an organization's strategic partners (e.g., suppliers and shipping companies) interact directly with internal systems.

FIGURE 2-6
Stakeholders of a comprehensive accounting system for a publicly traded company



operational stakeholders persons who regularly interact with a system in the course of their jobs or lives

executive stakeholders persons who don't interact directly with the system but who either use information produced by the system or have a significant financial or other interest in its operation and success

client person or group that provides the funding for a system development project

Operational stakeholders are those who regularly interact with a system in the course of their jobs or lives. Examples include bookkeepers interacting with an accounting or billing system, factory workers interacting with a production scheduling system, customers interacting with an Internet storefront, and patients who interact with a health care Web site, Facebook page, or Twitter newsfeed. Operational users are a key source of requirements information, especially as it pertains to user interfaces and related functionality. **Executive stakeholders** are those who do not interact directly with the system but who either use information produced by the system or have a significant financial or other interest in its operation and success. Examples include an organization's senior managers and board of directors, regulatory agencies, and taxing authorities.

Including such stakeholders in analysis activities is critical because the requirements information they possess may not be obvious or widely known in the organization. In addition, system requirements imposed by executive stakeholders, especially external ones, often have significant legal and financial implications. For example, consider the potential effects of IRS regulations on an accounting system or the effects of federal and state privacy laws on a social networking system.

Two other stakeholder groups that do not neatly fall into the categories just described deserve special attention. The **client** is the person or group that provides the funding for the project. In many cases, the client is senior management. However, clients may also be a separate group, such as a corporation's board of directors, executives in a parent company, or the board of regents of a university. The project team includes the client in its list of important stakeholders because the team must provide periodic status reviews to the client throughout development. The client or a direct representative on a steering or oversight committee also usually approves stages of the project and releases funds.

An organization's technical and support staff are also stakeholders in any system. The technical staff includes people who establish and maintain the computing environment of the organization. Support staff provide user training, troubleshooting, and related services. Both groups should provide guidance in such areas as programming language, computer platforms, network interfaces, and existing systems and their support issues. Any new system must fit within an organization's existing technology architecture, application architecture, and support environment. Thus, technical and support representatives are important stakeholders.

The Stakeholders for RMO

As a starting point for identifying CSMS stakeholders, it is helpful to develop a list of CSS stakeholders, which include:

- Phone/mail sales order clerks
- Warehouse and shipping personnel
- Marketing personnel who maintain online catalog information
- Marketing, sales, accounting, and financial managers
- Senior executives
- Customers
- External shippers (e.g., UPS and FedEx)

Because the CSMS will take over existing functions of the CSS, the list of CSMS stakeholders includes all the stakeholders in the CSS list. However, there are some subtle differences. For example, the inclusion of social networking functions in the CSMS and the planned ability to share Mountain Bucks expands the concept of who is a customer. While the old CSS was intended for use by potential customers visiting the Web site, the new system potentially interacts with a much larger group of external stakeholders, including friends and family of existing customers and potentially all users of popular social networking sites. In essence, the stakeholder group "Customers" is much larger, more diverse, and includes people who have not purchased from RMO. Ensuring that the viewpoints and requirements of this diverse group are represented during analysis activities will be a considerable challenge.

Expanded functionality for sales promotions with partner organizations creates an entirely new group of external stakeholders within those partner organizations. At this point, it is unclear whether that group will include operational stakeholders, executive stakeholders, or both and exactly how those stakeholders will interact with the system. Once again, ensuring adequate input from those stakeholders will be a significant challenge, especially because the portions of the system used by those stakeholders will not be based on an existing system.

RMO is a privately held company; John and Liz Blankens are the owners, and they hold two senior management positions. This is significant because the key operational systems of any publicly traded company "inherit" many external stakeholders due to the flow of information from those systems to the organization's financial reports. RMO is audited by an external accounting firm, primarily to ensure access to bank loans and other private financing.

As owners and senior managers, John and Liz are the primary clients, but so are other senior executives who form a collaborative decision-making body. In addition, existing technical and support staff are key stakeholders. **Figure 2-7** summarizes the internal managerial stakeholders in the form of an organization chart.

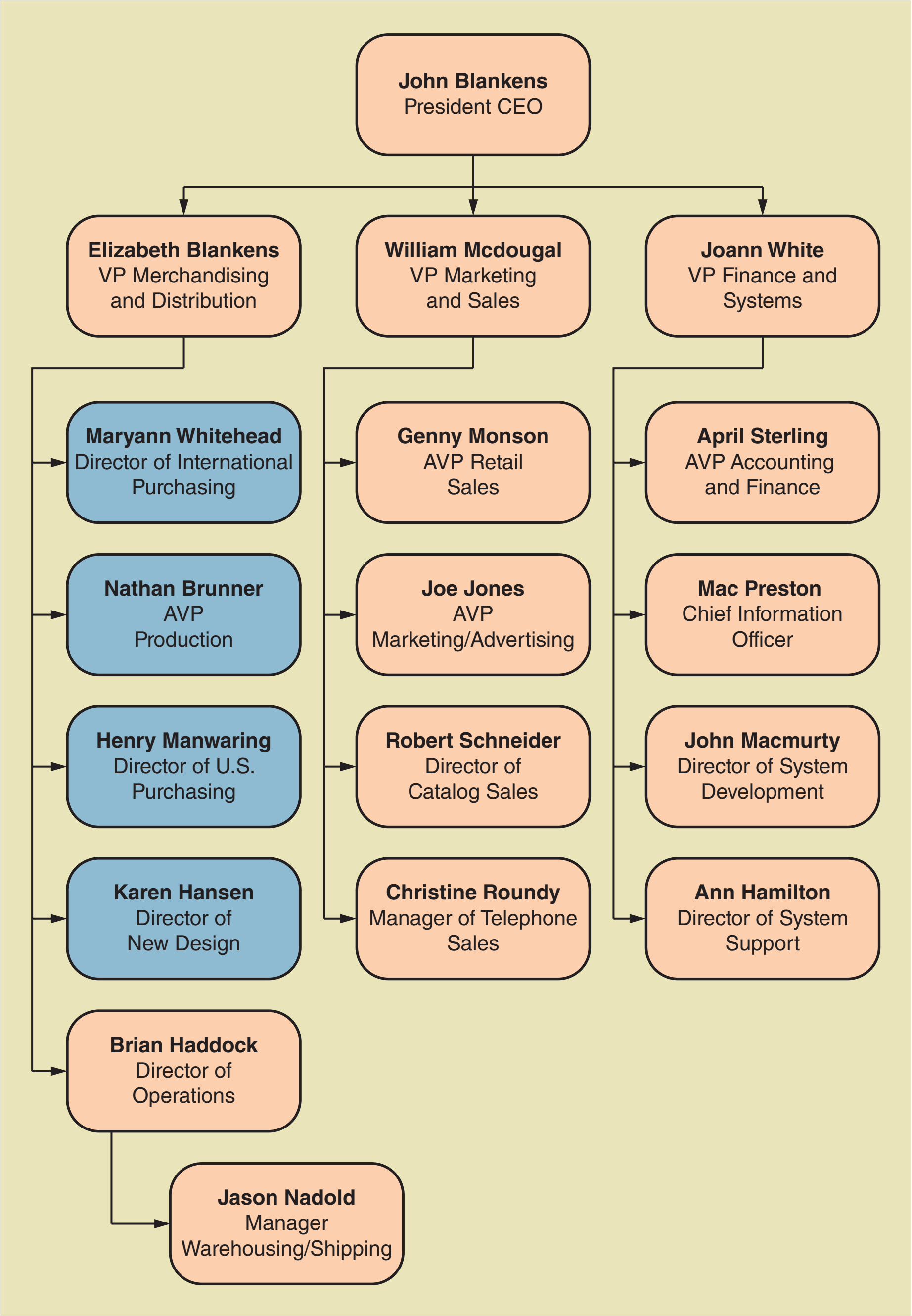
Information-Gathering Techniques

Techniques for gathering detailed requirements information include:

- Interviewing users and other stakeholders
- Distributing and collecting questionnaires
- Reviewing inputs, outputs, and documentation
- Observing and documenting business procedures
- Researching vendor solutions
- Collecting active user comments and suggestions

All these methods have proven to be effective, although some are more efficient than others. In most cases, analysts combine methods to increase their effectiveness and efficiency and to provide a comprehensive fact-finding approach.

FIGURE 2-7
RMO management stakeholders involved in the CSS requirements definition



Interview Users and Other Stakeholders

Interviewing users and other stakeholders is an effective way to understand business functions and business rules. Unfortunately, it is also the most time-consuming and resource-expensive option. In this method, systems analysts:

- Prepare detailed questions.
- Meet with individuals or groups of users.
- Obtain and discuss answers to the questions.
- Document the answers.
- Follow up as needed in future meetings or interviews.

Obviously, this process may take some time, so it usually requires multiple sessions with each of the users or user groups.

FIGURE 2-8
Themes for information-gathering questions

Theme	Questions to users
What are the business operations and processes?	What do you do?
How should those operations be performed?	How do you do it? What steps do you follow? How could they be done differently?
What information is needed to perform those operations?	What information do you use? What inputs do you use? What outputs do you produce?

Question Themes

Whether in informal meetings, formal interviews, or as part of a questionnaire or survey, analysts ask questions. But which questions should analysts ask? **Figure 2-8** shows three major themes that guide the analysts when they are asking questions to define system requirements; it also shows sample questions that arise from those themes.

What Are the Business Processes? The analyst must obtain a comprehensive list of all the business processes. In most cases, the users provide answers in terms of the current system, so the analyst must carefully discern which of those functions are fundamental (i.e., which will remain and which may possibly be eliminated with an improved system). For example, sales clerks might indicate that the first task they perform when a customer places an order is to check the customer's credit history. In the new system, sales clerks might never need to perform that function; the system might perform the check automatically. The function remains a system requirement, but the method of carrying out the function and its timing are changed.

How Are the Business Processes Performed? Again, the focus starts with the current system but gradually moves to the new system. The goal is to determine how the new system should support the function rather than how it supports it now. The analyst must be able to help the user visualize new and more efficient approaches to performing the business processes made possible by new technology or processes.

What Information Is Required? Some information inputs are formal; others are informal. When questioning the user, the analyst should specifically ask about exceptions or unusual situations in order to identify additional (nonroutine) information requirements. In this theme and the previous one, detail is the watchword. An analyst must understand the nitty-gritty detail to develop a correct solution.

Question Types

Questions can be roughly divided into two types:

open-ended questions questions that encourage discussion or explanation

closed-ended questions questions that elicit specific facts

- **Open-ended questions**—questions such as "How do you do this function?" that encourage discussion and explanation
- **Closed-ended questions**—questions such as "How many forms a day do you process?" that are used to get specific facts

Generally, open-ended questions help to start a discussion and enable a large number of requirements to be uncovered fairly quickly. Note that all the questions in the previous section are open ended. A discussion that starts with open-ended questions usually shifts gradually to closed-ended questions that elicit or confirm specific details of a business process.

Focus of Questions—Current System or New?

A significant question that faces all analysts is how much effort to expend studying and documenting the existing system (if one exists). Excess attention to an existing system can consume considerable time and can result in simply updating that system with newer technology. As a result, no matter how inefficient the current system is, system developers simply reimplement the procedures that are already in place. On the other hand, if a new system inherits many or all of the requirements of an existing system, then an analyst risks missing important requirements through insufficient study of the existing system.

To minimize both risks, analysts must balance the review of current business functions with discovery of new system requirements. It is still critical to have a complete correct set of system requirements, but in today's fast-paced world, there is no time or money to review all the old systems and document all the inefficient procedures. In fact, in today's development environment, one of the most valuable capabilities that a good system developer can bring is a new perspective to the problem.

Interview Preparation, Conduct, and Follow-Up

Figure 2-9 is a sample checklist that summarizes the major points to be covered; it is useful in preparing for, conducting, and following up an interview.

Preparing for the Interview Every successful interview requires preparation. The first and most important step in preparing for an interview is to establish its objective. In other words, what do you want to accomplish with this interview? Write down the objective so it is firmly established in your mind. The second step is to determine which stakeholders should be involved in the interview. A small number of interviewees is generally best when the objective is narrow or of a fact-finding nature. Larger groups are better if the objective is more open ended, such as when generating and evaluating new ideas. However, it can be difficult to manage a large group meeting to ensure high-quality input from all participants. If possible, have at least two analysts involved in every interview, and have them compare notes afterward to ensure accuracy.

The next step is to prepare detailed questions to be used in the interview. Write down a list of specific questions, and prepare notes based on the forms or reports received earlier. Usually, you should prepare a list of questions that are consistent with the objective of the interview. Open-ended questions and closed-ended questions are appropriate. Generally, open-ended questions help

FIGURE 2-9
Sample checklist to prepare for user interviews

Checklist for Conducting an Interview

Before

□ Establish the objective for the interview.
□ Determine correct user(s) to be involved.
□ Determine project team members to participate.
□ Build a list of questions and issues to be discussed.
□ Review related documents and materials.
□ Set the time and location.
□ Inform all participants of objective, time, and locations.

During

□ Arrive on time.
□ Look for exception and error conditions.
□ Probe for details.
□ Take thorough notes.
□ Identify and document unanswered items or open questions.

After

□ Review notes for accuracy, completeness, and understanding.
□ Transfer information to appropriate models and documents.
□ Identify areas needing further clarification.
□ Thank the participants.
□ Follow up on open and unanswered questions.

get the discussion started and encourage the user to explain all the details of the business process and the rules.

The last step is to make the final interview arrangements and then communicate those arrangements to all participants. A specific time and location should be established. If possible, a quiet location should be chosen, to avoid interruptions. Each participant should know the objective of the meeting and, when appropriate, should have a chance to preview the questions or materials to be used. Interviews consume a substantial amount of time, and they can be made more efficient if each participant knows beforehand what is to be accomplished.

Conducting the Interview The usual rules of workplace meetings apply during stakeholder interviews: plan ahead, arrive early, and ensure that the room is prepared and that needed resources are available. Limit the time of the interview for the benefit of the analyst(s) and stakeholder(s); stakeholders have other responsibilities, and the analysts can absorb only so much information at one time. It is better to have several shorter interviews than one long interview. A series of interviews provides an opportunity to absorb the material and then go back to get clarification later.

Look for exception and error conditions. Look for opportunities to ask "what if" questions. "What if it doesn't arrive? What if the signature is missing? What if the balance is incorrect? What if two orders are exactly the same?" The essence of good systems analysis is understanding all the "what ifs." Make a conscious effort to identify all the exception conditions and then ask about them. More than any other skill, the ability to think of the exceptions will help you discover the detailed business rules. It is a hard skill to teach from a textbook; experience will hone this skill. You will teach yourself this skill by conscientiously practicing it.

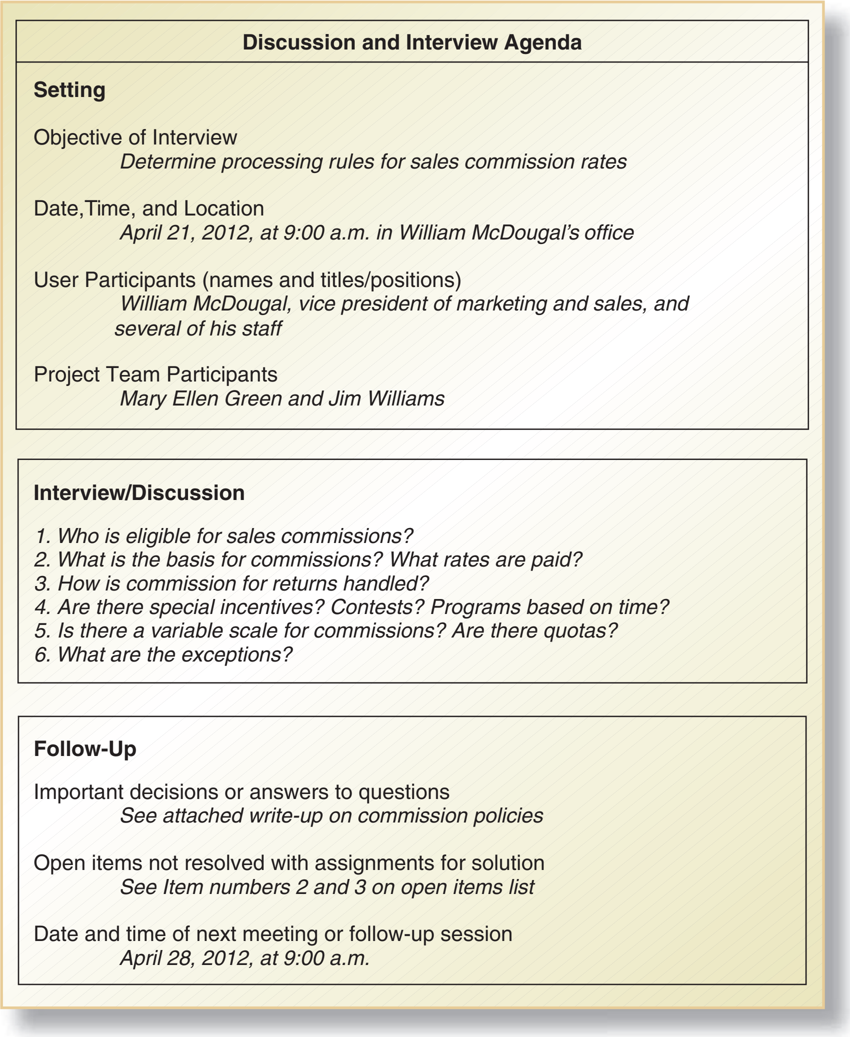
Probe for details. In addition to looking for exception conditions, the analyst must probe to ensure a complete understanding of all procedures and rules. One of the most difficult skills to learn as a new systems analyst is how to get enough details. Frequently, it is easy to get a general overview of how a process works. But do not be afraid to ask detailed questions until you thoroughly understand how the process works and what information is used. You cannot do effective systems analysis by glossing over the details.

Take careful notes. It is a good idea to take handwritten notes. Usually, tape recorders make users nervous. However, taking notes signals that you think the information you are obtaining is important, and the user is complimented. If two analysts conduct each interview, they can compare notes later. Identify and document in your notes any unanswered questions or outstanding issues that were not resolved. A good set of notes provides the basis for building the analysis models as well as establishing a basis for the next interview session.

Figure 2-10 is a sample agenda for an interview session. Obviously, you do not need to conform exactly to a particular agenda. However, as with the interview checklist shown in Figure 2-9, this figure will help prod your memory on issues and items that should be discussed in an interview. Make a copy and use it. As you develop your own style, you can modify the checklist to the way you like to work.

Following Up the Interview Follow-up is an important part of each interview. The first task is to absorb, understand, and document the information that was obtained. Generally, analysts document the details of the interview by constructing models of the business processes and writing textual descriptions of nonfunctional requirements. These tasks should be completed as soon after the interview as possible and the results distributed to the interview participants for validation. If the modeling methods are complex or unfamiliar to the users, the analyst should schedule follow-up meetings to explain and verify the models, as described in the last section of this chapter.

FIGURE 2-10 *Sample interview session agenda with follow-up information*



Discussion and Interview Agenda

Setting

Objective of Interview
Determine processing rules for sales commission rates

Date,Time, and Location
April 21, 2012, at 9:00 a.m. in William McDougal's office

User Participants (names and titles/positions)
William McDougal, vice president of marketing and sales, and several of his staff

Project Team Participants
Mary Ellen Green and Jim Williams

Interview/Discussion

1. Who is eligible for sales commissions?
2. What is the basis for commissions? What rates are paid?
3. How is commission for returns handled?
4. Are there special incentives? Contests? Programs based on time?
5. Is there a variable scale for commissions? Are there quotas?
6. What are the exceptions?

Follow-Up

Important decisions or answers to questions
See attached write-up on commission policies

Open items not resolved with assignments for solution
See Item numbers 2 and 3 on open items list

Date and time of next meeting or follow-up session
April 28, 2012, at 9:00 a.m.

During the interview, you probably asked some "what if" questions that the users could not answer. They are usually policy questions raised by the new system that management has not considered before. It is extremely important that these questions not get lost or forgotten. **Figure 2-11** shows a sample table for tracking outstanding or unresolved issues for RMO. The table includes questions posed by users or analysts and responsibilities assigned for resolving the issues. If several teams are working, a combined list can be maintained. Other columns that might be added to the list are an explanation of the problem's resolution and the date resolved.

FIGURE 2-11 *Sample open-items list*

ID	Issue title	Date identified	Target end date	Responsible project person	User contact	Comments
1	Partial shipments	6-12-2012	7-15-2012	Jim Williams	Jason Nadold	Ship partials or wait for full shipment?
2	Returns and commissions	7-01-2012	9-01-2012	Jim Williams	William McDougal	Are commissions recouped on returns?
3	Extra commissions	7-01-2012	8-01-2012	Mary Ellen Green	William McDougal	How to handle commissions on special promotions?

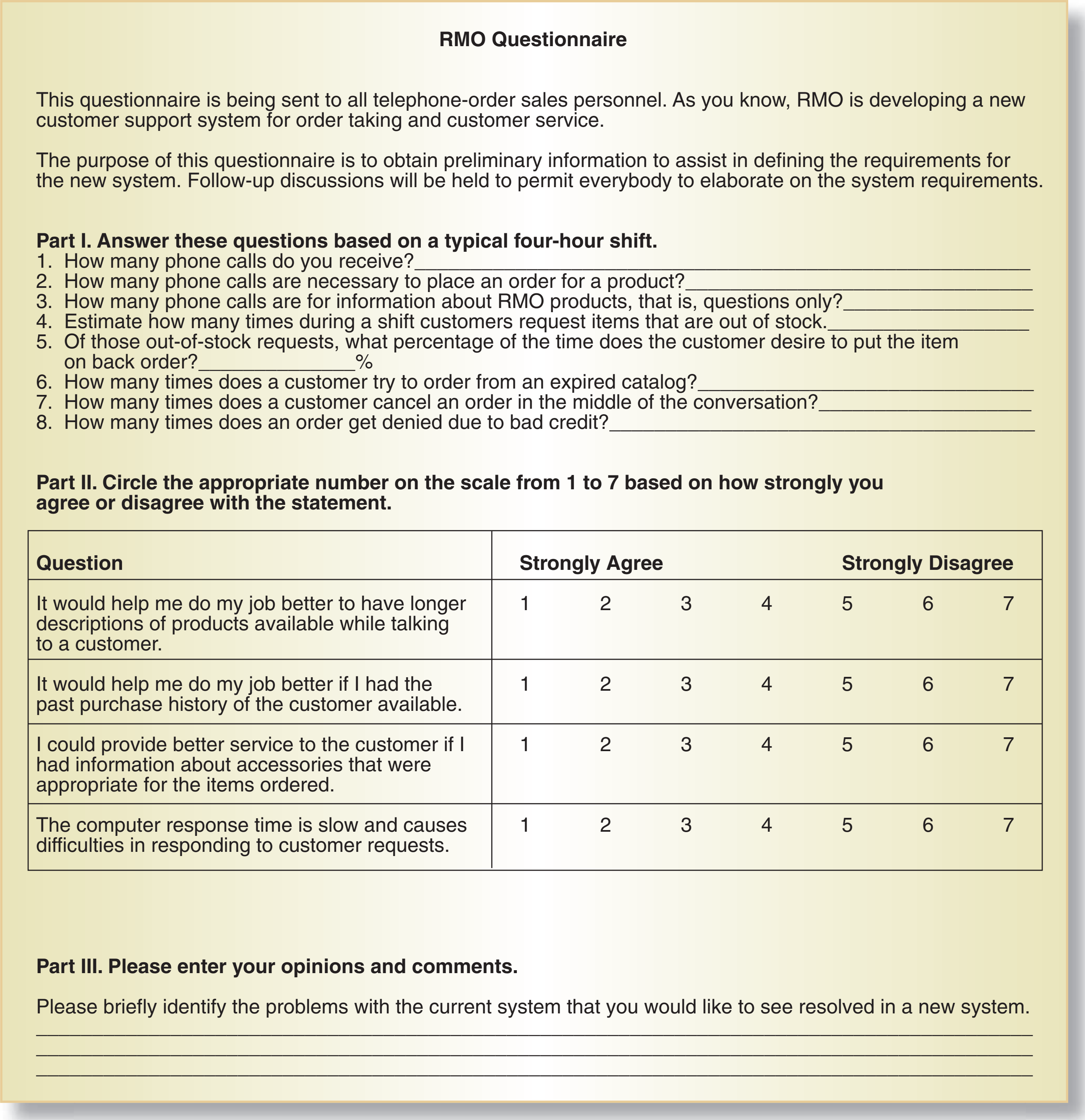
Finally, make a list of new questions based on areas that need further elaboration or that are missing information. This list will prepare you for the next interview.

Distribute and Collect Questionnaires

Questionnaires enable analysts to collect information from a large number of stakeholders. Even if the stakeholders are widely distributed geographically, they can still help define requirements through questionnaires. Questionnaires are often used to obtain preliminary insight into stakeholder information needs, which helps to determine areas that need further research by using other methods.

Figure 2-12 is a sample questionnaire showing three types of questions. The first part has closed-ended questions to determine quantitative information.

FIGURE 2-12 *Sample questionnaire*



RMO Questionnaire

This questionnaire is being sent to all telephone-order sales personnel. As you know, RMO is developing a new customer support system for order taking and customer service.

The purpose of this questionnaire is to obtain preliminary information to assist in defining the requirements for the new system. Follow-up discussions will be held to permit everybody to elaborate on the system requirements.

Part I. Answer these questions based on a typical four-hour shift.

1. How many phone calls do you receive?______________________________
2. How many phone calls are necessary to place an order for a product?____________________
3. How many phone calls are for information about RMO products, that is, questions only?__________
4. Estimate how many times during a shift customers request items that are out of stock.__________
5. Of those out-of-stock requests, what percentage of the time does the customer desire to put the item on back order?______________%
6. How many times does a customer try to order from an expired catalog?____________________
7. How many times does a customer cancel an order in the middle of the conversation?__________
8. How many times does an order get denied due to bad credit?______________________________

Part II. Circle the appropriate number on the scale from 1 to 7 based on how strongly you agree or disagree with the statement.

Question	**Strongly Agree**						**Strongly Disagree**
It would help me do my job better to have longer descriptions of products available while talking to a customer.	1	2	3	4	5	6	7
It would help me do my job better if I had the past purchase history of the customer available.	1	2	3	4	5	6	7
I could provide better service to the customer if I had information about accessories that were appropriate for the items ordered.	1	2	3	4	5	6	7
The computer response time is slow and causes difficulties in responding to customer requests.	1	2	3	4	5	6	7

Part III. Please enter your opinions and comments.

Please briefly identify the problems with the current system that you would like to see resolved in a new system.

____

____

____

The second part consists of opinion questions in which respondents are asked whether they agree or disagree with the statement. Both types of questions are useful for tabulating and determining quantitative averages. The third part requests an explanation of a procedure or problem. Questions such as these are good as a preliminary investigation to help direct further fact-finding activities.

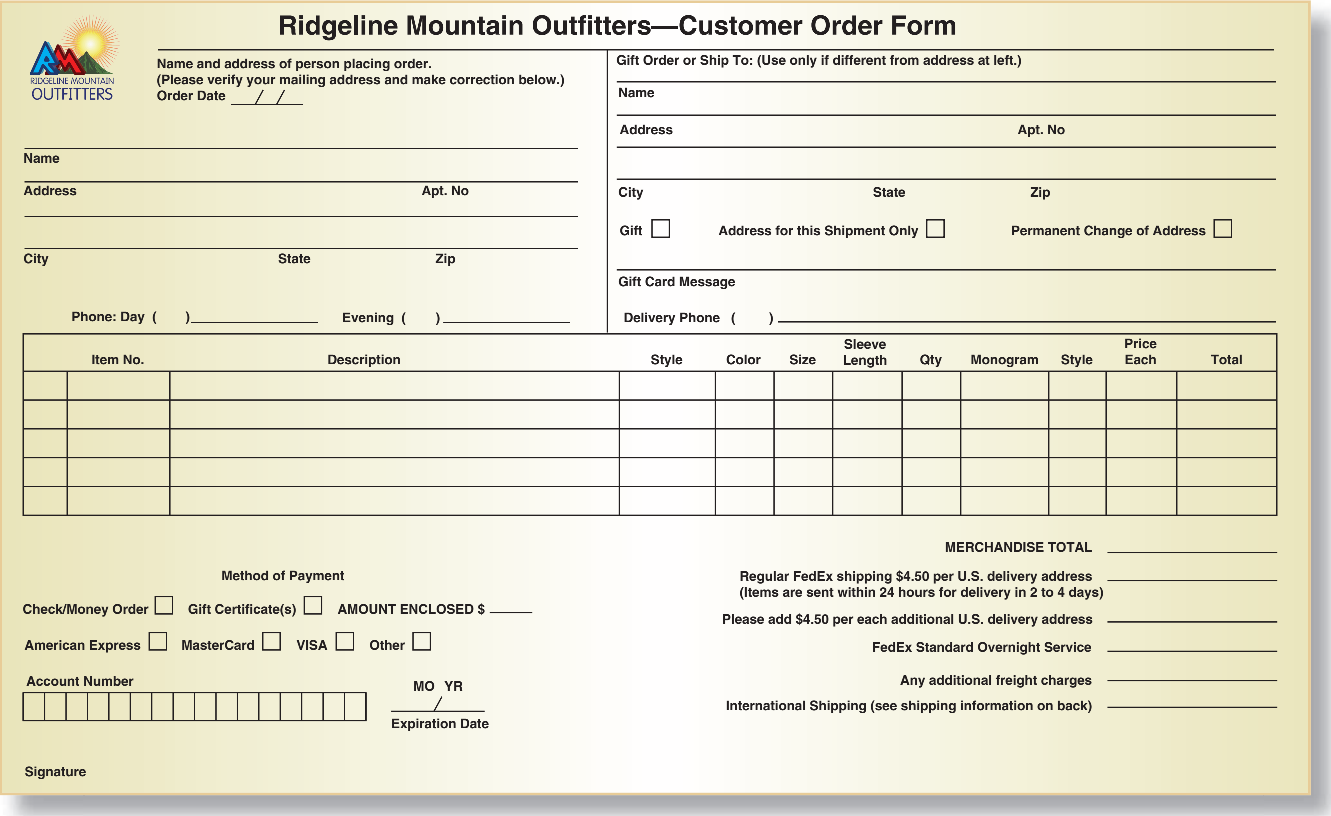
Questionnaires are not well suited to helping you learn about processes, workflows, or techniques. Open-ended questions such as "How do you do this process?" are best answered by using interviews or observation. Although a questionnaire can contain a very limited number of open-ended questions, stakeholders frequently do not return questionnaires that contain many open-ended questions.

Review Inputs, Outputs, and Procedures

There are two sources of information about inputs, outputs, and procedures. One source is external to the organization—industry-wide professional organizations and other companies. It may not be easy to obtain information from other companies, but they are a potential source of important information. Sometimes, industry journals and magazines report the findings of "best practices" studies. The project team would be negligent in its duties if its members were not familiar with best practice information.

The second source of inputs, outputs, and procedures is existing business documents and procedure descriptions within the organization. Reviewing internal documents and procedures serves two purposes. First, it is a good way to get a preliminary understanding of the processes. Second, existing inputs, outputs, and documents can serve as visual aids for the interview and as the working documents for discussion (see **Figure 2-13**). Discussion can focus on a specific input or output, its objective, its distribution, and its information content. The discussion should also include specific business events that initiate the use of an input or generation of an output. Several different business events

FIGURE 2-13 *RMO mail-order form used as a visual aid during an interview*



Ridgeline Mountain Outfitters—Customer Order Form

RIDGELINE MOUNTAIN OUTFITTERS

Name and address of person placing order.
(Please verify your mailing address and make correction below.)
Order Date ___/___/___

Name
Address Apt. No
City State Zip
Phone: Day ( )________ Evening ( )________

Gift Order or Ship To: (Use only if different from address at left.)
Name
Address Apt. No
City State Zip
Gift □ Address for this Shipment Only □ Permanent Change of Address □
Gift Card Message
Delivery Phone ( ) ________

	Item No.	Description	Style	Color	Size	Sleeve Length	Qty	Monogram	Style	Price Each	Total

Method of Payment
Check/Money Order □ Gift Certificate(s) □ AMOUNT ENCLOSED $ ____
American Express □ MasterCard □ VISA □ Other □
Account Number □□□□□□□□□□□□□□□□
MO YR
___/___
Expiration Date
Signature

MERCHANDISE TOTAL ________
Regular FedEx shipping $4.50 per U.S. delivery address ________
(Items are sent within 24 hours for delivery in 2 to 4 days)
Please add $4.50 per each additional U.S. delivery address ________
FedEx Standard Overnight Service ________
Any additional freight charges ________
International Shipping (see shipping information on back) ________

might require the same form, and specific information about the event and the business process is critical. It is always helpful to have screens and forms that have been filled out with real information to ensure that the analyst obtains a correct understanding of the data content.

Reviewing the documentation of existing procedures helps identify business rules that may not come up in the interviews. Analyzing formal procedure documentation also helps reveal discrepancies and redundancies in the business processes. However, procedure documents frequently are not kept up to date, and they commonly include errors. To ensure that the assumptions and business rules that derive from the existing documentation are correct, analysts should review them with the users.

Observe and Document Business Processes

Firsthand experience is invaluable to understand exactly what occurs within business processes. More than any other activity, observing a business process in action will help you understand the business functions. However, while observing existing processes, you must also be able to visualize the new system's associated business processes. That is, as you observe the current business processes to understand the fundamental business needs, you should never forget that the processes could and often should change to be made more efficient. Do not get locked into believing there is only one way of performing the process.

You can observe a business process in many ways, ranging from a quick walk-through of an office or plant to doing the work yourself. A quick walk-through gives a general understanding of the layout of the office, the need for and use of computer equipment, and the general workflow. Spending several hours observing users at their jobs helps you understand the details of using the computer system and carrying out business functions. Being trained as a user and actually doing the job enables you to discover the difficulties of learning new procedures, the importance of a system that is easy to use, and the stumbling blocks and bottlenecks of existing procedures and information sources.

It is not necessary to observe all processes at the same level of detail. A quick walk-through may be sufficient for one process, whereas a process that is critical or more difficult to understand might require an extended observation period. If you remember that the objective is a complete understanding of the business processes and rules, you can assess where to spend your time to gain that understanding. As with interviewing, it is usually better if two analysts combine their efforts in observing procedures.

Observation often makes the users nervous, so you need to be as unobtrusive as possible. You can put users at ease in several ways, such as by working alongside a user or observing several users at once. Common sense and sensitivity to the needs and feelings of the users will usually result in a positive experience.

Research Vendor Solutions

Many of the problems and opportunities that companies want to address with new information systems have already been solved by other companies. In addition, consulting firms often have experience with the same problems, and software firms may have already packaged solutions for a particular business need. Taking advantage of existing knowledge or solutions can avoid costly mistakes and save time and money.

There are three positive contributions and one danger in exploring existing solutions. First, researching existing solutions will frequently help users generate new ideas for how to better perform their business functions. Seeing how someone else solved a problem and applying that idea to the culture and structure of the existing organization will often provide viable alternative solutions for business needs.

Second, some of these solutions are excellent and state of the art. Without this research, the development team may create a system that is obsolete even before it is designed. Companies need solutions that not only solve basic business problems but that are up to date with competitive practices.

Third, it is often cheaper and less risky to buy a solution rather than to build it. If the solution meets the needs of the company and can be purchased, then that is usually a safer, quicker, and less expensive route.

The danger in exploring existing solutions is that the users and even the systems analysts may want to buy one of the alternatives immediately. But if a solution, such as a packaged software system, is purchased too early in the process, the company's needs may not be thoroughly investigated. Too many companies have purchased systems only to find out later that they only support half the functions that were needed. Do not rush into a purchase decision until requirements are fully defined and all viable alternatives have been thoroughly investigated.

Collect Active User Comments and Suggestions

As discussed in Chapter 1 and earlier in this chapter, system development normally proceeds with analysis, design, and other activities spread across multiple iterations. Portions of the system are constructed and tested during each iteration. Users and other stakeholders perform the initial testing of system functions during the iteration in which those functions are implemented. They also test and use those same functions during later iterations.

User feedback from initial and later testing is a valuable source of requirements information. Interviews, discussions, and model reviews are an imperfect way of eliciting complete and accurate requirements. The phrase "I'll know it when I see it" applies well to requirements definition. Users often cannot completely or accurately state their requirements until they can interact with a live system that implements those requirements. Based on those interactions, users can develop concrete suggestions for improvement and identify missing or poorly implemented requirements.

Documenting Workflows with Activity Diagrams

As you gather information about business processes, you will need to document your results. One effective way to capture this information is with diagrams. Eventually, you may want to use diagrams to describe the workflows of the new system, but for now, let us focus on how we would document the current business workflows.

workflow sequence of processing steps that completely handles one business transaction or customer request

activity diagram describes user (or system) activities, the person who does each activity, and the sequential flow of these activities

synchronization bar activity diagram component that either splits a control path into multiple concurrent paths or recombines concurrent paths

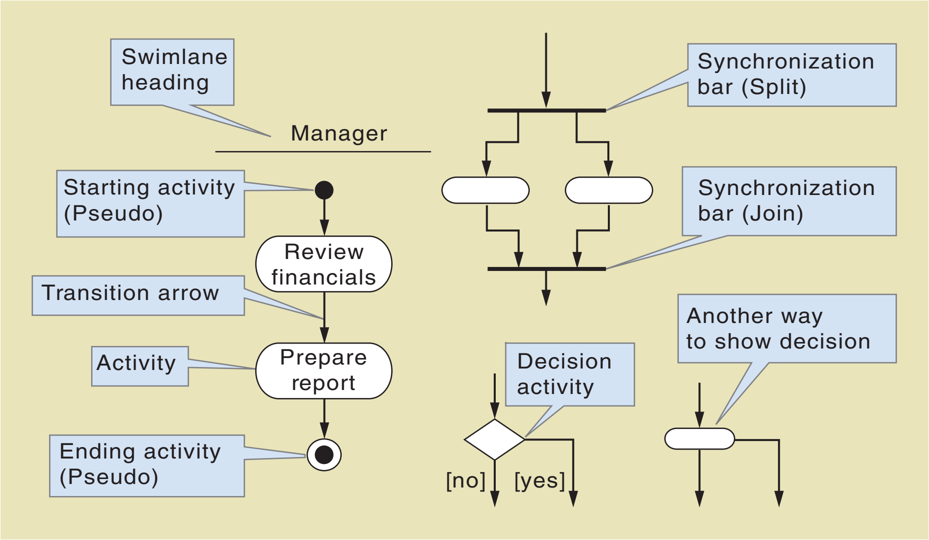
swimlane heading activity diagram column containing all activities for a single agent or organizational unit

A **workflow** is the sequence of processing steps that completely handles one business transaction or customer request. Workflows may be simple or complex. Complex workflows can be composed of dozens or hundreds of processing steps and may include participants from different parts of an organization.

An **activity diagram** describes the various user (or system) activities, the person who does each activity, and the sequential flow of these activities.

Figure 2-14 shows the basic symbols used in an activity diagram. The ovals represent the individual activities in a workflow. The connecting arrows represent the sequence between the activities. The black circles denote the beginning and the ending of the workflow. The diamond is a decision point at which the flow of the process will either follow one path or another. The heavy solid line is a **synchronization bar**, which either splits the path into multiple concurrent paths or recombines concurrent paths. The **swimlane heading** represents an agent who performs the activities. Because it is common in a workflow to have different agents (i.e., people) performing different steps of the workflow process,

FIGURE 2-14
Activity diagram symbols



the swimlane symbol divides the workflow activities into groups showing which agent performs which activity.

Figure 2-15 is an activity diagram that describes the order fulfillment process for the current RMO CSMS. Processing begins when the customer has completed the order checkout process. The diagram describes the back-and-forth flow of information and control between the Order subsystem, Inventory subsystem, warehouse(s), and shipper. The diagram is simplified because it omits many error-handling pathways, including what happens if enough item stock is on hand to fulfill part of an order.

Figure 2-16 illustrates another workflow diagram, which demonstrates some new concepts. In this example, a customer is ordering a product that has to be manufactured specifically to match customer specifications. The salesperson sends the order to Engineering, and the diagram uses a new symbol to emphasize the transmission of the document between Sales and Engineering. After Engineering develops the specifications, two concurrent activities happen: Purchasing orders the materials, and Production writes the program for the automated milling machines. These two activities are completely independent and can occur at the same time. Notice that one synchronization bar splits the path into two concurrent paths and that another synchronization bar reconnects them. Finally, Scheduling puts the order on the production schedule.

Creating activity diagrams to document workflows is straightforward. The first step is to identify the agents to create the appropriate swimlanes. Next, follow the various steps of the workflow and then make appropriate ovals for the activities. Connect the activity ovals with arrows to show the workflow. Here are a couple guidelines:

- Use a decision symbol to represent an either/or situation—one path or the other path but not both. As a shorthand notation, you can merge an activity (by using an oval) and a decision (by using a diamond) into a single oval with two exit arrows, as indicated on the right in Figure 2-14. This notation represents a decision (either/or) activity. Wherever you have an activity that reads “verify” or “check,” you will probably require a decision—one for the “accept” path and one for the “reject” path.
- Use synchronization bars for parallel paths—situations in which both paths are taken. Include a beginning and an ending synchronization bar. You can also use synchronization bars to represent a loop, such as a “do while” programming loop. Put the bar at the beginning of the loop and then describe it as “for every.” Put another synchronization bar at the end of the loop with the description “end for every.”

FIGURE 2-15 *Simple activity diagram for online checkout*

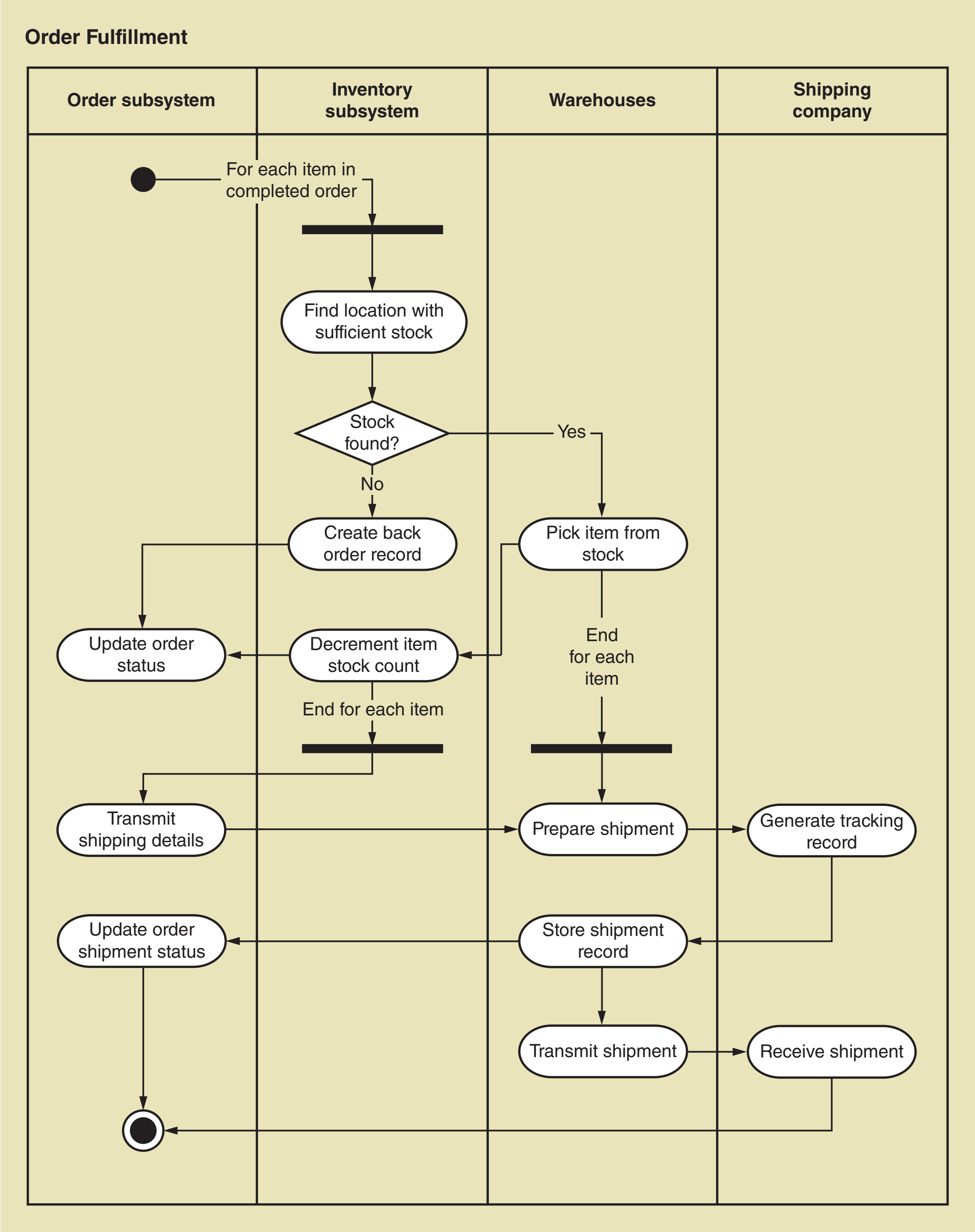
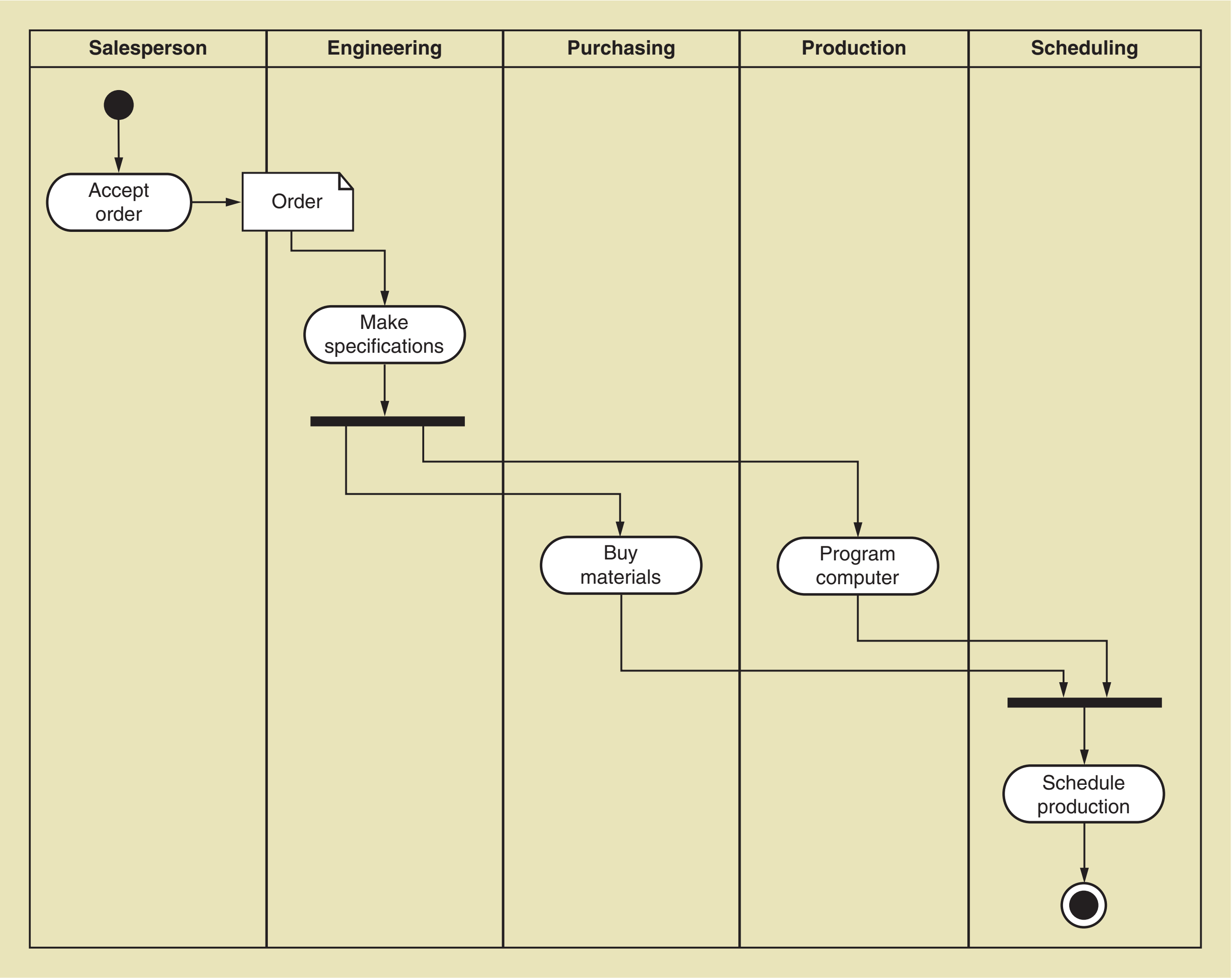


FIGURE 2-16 *Activity diagram showing concurrent paths*



Chapter Summary

There are five primary activities of systems analysis:

- Gather detailed information.
- Define requirements.
- Prioritize requirements.
- Develop user-interface dialogs.
- Evaluate requirements with users.

Functional requirements are those that explain the basic business functions that the new system must support. Nonfunctional requirements involve the system's objectives with regard to technology, performance, usability, reliability, and security.

Mathematical, descriptive, and graphical models are developed to document requirements and as an aid in evaluating requirements with users and other stakeholders. Stakeholders include internal and external users of the system and other persons or organizations that have a vested interest in the system.

Analysts use many techniques to gather information about requirements, including:

- Interviews
- Questionnaires
- Documentation, input, and output reviews
- Process observation and documentation
- Vendor solution research
- Active comments and suggestions from users

Workflow diagrams are a key modeling technique often used as an early requirements model. Workflow diagrams graphically model the steps of a business process and the participants who perform them. Other models and diagrams are covered in later chapters.

Key Terms

activity diagram	57
application architecture	37
client	47
closed-ended questions	50
design constraints	43
executive stakeholders	47
external stakeholders	46
functional requirements	42
FURPS	43
FURPS+	43
graphical models	46
implementation requirements	44
interface requirements	44
internal stakeholders	46
mathematical models	46
model	44
nonfunctional requirements	43
open-ended questions	50
operational stakeholders	47
performance requirements	43
physical requirements	44
reliability requirements	43
security requirements	43
stakeholders	46
supportability requirements	44
swimlane heading	57
synchronization bar	57
system requirements	42
technology architecture	37
textual models	45
unified modeling language (UML)	46
usability requirements	43
workflow	57

Review Questions

1. List and briefly describe the five activities of systems analysis.
2. What are three types of models?
3. What is the difference between functional requirements and nonfunctional requirements?
4. Describe the steps in preparing for, conducting, and following up an interview session.
5. What are the benefits of doing vendor research during information-gathering activities?
6. What types of stakeholders should you include in fact finding?
7. Describe the open-items list and then explain why it is important.
8. List and briefly describe the six information-gathering techniques.
9. What is the purpose of an activity diagram?
10. Draw and explain the symbols used on an activity diagram.

Problems and Exercises

1. Provide an example of each of the three types of models that might apply to designing a car, a house, and an office building.
2. One of the toughest problems in investigating system requirements is to make sure they are complete and comprehensive. How would you ensure that you get all the right information during an interview session?
3. One of the problems you will encounter during your investigation is “scope creep” (i.e., user requests for additional features and functions). Scope creep happens because users sometimes have many unsolved problems and the system investigation may be the first time anybody has listened to their needs. How do you keep the system from growing and including new functions that should not be part of the system?
4. What would you do if you got conflicting answers for the same procedure from two different people you interviewed? What would you do if one was a clerical person and the other was the department manager?
5. You have been assigned to resolve several issues on the open-items list, and you are having a hard time getting policy decisions from the user contact. How can you encourage the user to finalize these policies?
6. In the running case of RMO, assume that you have set up an interview with the manager of the

shipping department. Your objective is to determine how shipping works and what the information requirements for the new system will be. Make a list of questions—open ended and closed ended—that you would use. Include any questions or techniques you would use to ensure you find out about the exceptions.

7. Develop an activity diagram based on the following narrative. Note any ambiguities or questions that you have as you develop the model. If you need to make assumptions, also note them.

   The purchasing department handles purchase requests from other departments in the company. People in the company who initiate the original purchase request are the "customers" of the purchasing department. A case worker within the purchasing department receives the request and monitors it until it is ordered and received.

   Case workers process requests for the purchase of products under $1,500, write a purchase order, and then send it to the approved vendor. Purchase requests over $1,500 must first be sent out for bid from the vendor that supplies the product. When the bids return, the case worker selects one bid and then writes a purchase order and sends it to the vendor.

8. Develop an activity diagram based on the following narrative. Note any ambiguities or questions that you have as you develop the model. If you need to make assumptions, also note them.

   The shipping department receives all shipments on outstanding purchase orders. When the clerk in the shipping department receives a shipment, he or she finds the outstanding purchase order for those items. The clerk then sends multiple copies of the shipment packing slip. One copy goes to Purchasing, and the department updates its records to indicate that the purchase order has been fulfilled. Another copy goes to Accounting so a payment can be made. A third copy goes to the requesting in-house customer so he or she can receive the shipment.

   After payment is made, the accounting department sends a notification to Purchasing. After the customer receives and accepts the goods, he or she sends notification to Purchasing. When Purchasing receives these other verifications, it closes the purchase order as fulfilled and paid.

9. Conduct a fact-finding interview with someone involved in a procedure that is used in a business or organization. This person could be someone at the university, in a small business in your neighborhood, in the student volunteer office at the university, in a doctor's or dentist's office, or in a volunteer organization. Identify a process that is done, such as keeping student records, customer records, or member records. Make a list of questions and then conduct the interview. Remember, your objective is to understand that procedure thoroughly (i.e., to become an expert on that single procedure).

10. Using RMO and the CSMS as your guide, develop a list of all the procedures that may need to be researched. You may want to think about the exercise in the context of your experience with such retailers as L.L. Bean, Lands' End, or Amazon.com. Check out the Internet marketing done on the retailers' Web sites and then think about the underlying business procedures that are required to support those sales activities. List the procedures and then describe your understanding of each.

Case Study

John and Jacob, Inc.: Online Trading System

John and Jacob, Inc. is a regional brokerage firm that has been successful over the last several years. Competition for customers is intense in this industry. The large national firms have very deep pockets, with many services to offer clients. Severe competition also comes from discount and Internet trading companies. However, John and Jacob has been able to cultivate a substantial customer base from upper-middle-income clients in the northeastern United States. To maintain a competitive edge with its customers, John and Jacob is in the process of modernizing its online trading system. The modernization will add new features to the existing system and expand the range of interfaces beyond desktop and laptop computers to include tablet computers and smartphones. The system will add Twitter messaging in addition to continued support for traditional e-mail.

Edward Finnigan, the project manager, is in the process of identifying all the groups of people who should be included in the development of the system requirements. He is not quite sure exactly who should be included. Here are the issues he is considering:

- **Users:** The trading system will be used by customers and by staff in each of the company's 30 trading offices. Obviously, the brokers who are going to use the system need to have input, but how should this be done? Edward also is not sure what approach would be best to ensure that the requirements are complete yet not require tremendous amounts of time. Including all the offices would increase enthusiasm and support for the system, but it would take a lot of time. Involving more brokers would bring divergent opinions that would have to be reconciled.

- **Customers:** The trading system will also include trade order entry, investment analysis reports, trade confirmations, standard and customized reporting, and customer statements. Edward wonders how to involve John and Jacob customers in the development of system requirements. Edward is sensitive to this issue because many brokers have told him that many customers are unhappy with the current system, and customer complaints are sometimes posted to the public comments area of the current Web site. He would like to involve customers, but he does not know how.
- **Other stakeholders:** Edward knows he should involve other stakeholders to help define system requirements. He is not quite sure whom he should contact. Should he go to senior executives? Should he contact middle management? Should he include such back-office functions as accounting and investing? He is not quite sure how to get organized or how to decide who should be involved.

Answer the following questions:

1. What is the best method for Edward to involve the brokers (users) in the development of the updated online trading system? Should he use a questionnaire? Should he interview the brokers in each of the company's 30 offices or would one or two brokers representing the entire group be better? How can Edward ensure that the information about requirements is complete yet not lose too much time doing so?
2. Concerning customer input for the new system, how can Edward involve customers in the process? How can he interest them in participating? What methods can Edward use to ensure that the customers he involves are representative of John and Jacob's entire customer group?
3. As Edward considers what other stakeholders he should include, what are some criteria he should use? Develop some guidelines to help him build a list of people to include.

RUNNING CASES

Community Board of Realtors

The real estate business relies on an extensive amount of information used in the buying and selling of real property. Most communities of real estate agents and brokers have formed cooperative organizations to help consolidate and distribute information on the real estate profession, real estate trends, properties in the community, historical records of property sales, and current listings of properties for sale. These organizations are usually referred to as the Community Board of Realtors.

Research your local Community Board of Realtors to answer these questions:

1. Who are the stakeholders for the issues related to real estate in your community, and what are their main interests?
2. What types of information does the board collect and make available to its members and to the community?
3. Research the real estate industry in at least two countries other than the United States. For each of these countries, what are some of the cultural and legal issues that differ from those in the United States? If you were working on support for an international real estate cooperative system, in what ways would the information collection activity process be complicated?

The Spring Breaks 'R' Us Travel Service

Spring Breaks 'R' Us (SBRU) is an online travel service that books spring break trips to resorts for college students. Students have booked spring break trips for decades, but changes in technology have transformed the travel business in recent years. SBRU moved away from having campus reps with posted fliers and moved to the Web early on. The basic idea is to get a group of students to book a room at a resort for one of the traditional spring break weeks. SBRU contracts with dozens of resorts

(continued on page 64)

(continued from page 63)

in key spring break destinations in Florida, Texas, the Caribbean, and Mexico. Its Web site shows information on each resort and includes prices, available rooms, and special features. Students can research and book a room, enter contract information, and pay deposits and final payments through the system. SBRU provides updated booking information, resort information updates, and travel information for booked students when they log in to the site.

The resorts also need access to information from SBRU. They need to know about their bookings for each week, the room types that are booked, and so forth. Before the spring break booking season starts, they need to enter information on their resorts, including prices and special features. Resorts need to be paid by SBRU for the bookings, and they need to be able to report and collect for damages caused by springbreakers during their stay.

SBRU has recently decided to upgrade its system to provide social networking features for students. It is currently researching possibilities and collecting information from prospective customers about desirable features and functions. From the business standpoint, the idea is to increase bookings by enhancing the experience before, during, and after the trip.

1. Who are the stakeholders for SBRU? For each type of stakeholder, what aspects of the SBRU booking system are of particular interest?
2. What are the main functional requirements for the major subsystem areas (i.e., resort relations, student booking, accounting and finance, and social networking)?
3. Describe some usability requirements for students, booking interactions, and social networking interactions.
4. Assuming that social networking at the resorts will require wireless communication and connection to the Internet, what are some reliability requirements that resorts might be asked to maintain? What are some performance requirements? Is this a bigger issue because resorts are in international locations?
5. What are some security requirements? Is there any reason why students in Europe, Asia, or other locations could not book rooms through SBRU? What issues might be anticipated?
6. To collect information on functional requirements for the social networking subsystem, what are some techniques that might be used? Be specific and include some sample questions you might ask by using various techniques.

On the Spot Courier Services

As an employee of a large international courier and shipping service, Bill Wiley met almost every day with many companies that shipped and received packages. He was frequently asked if his company could deliver local packages on the same day. Over several months, he observed that there appeared to be a substantial need for courier services in the city in which he lived. He decided that he would form his own courier delivery company called On the Spot to fill this need.

Bill began by listing his mobile telephone number in the Yellow Pages. He also sent letters to all those companies that had requested same-day courier service that his prior company had not been able to serve. He hoped that, through good service and word-of-mouth advertising, his business would grow. He also began other advertising and marketing activities to promote his services.

At first, Bill received delivery requests on his business mobile phone. However, it was not long before his customers were asking if he had a Web site where they could place orders for shipments. He knew that if he could get a Web presence he could increase his exposure and help his business grow.

After he had been in business only a few short months, Bill discovered he needed to have additional help. He hired another person to help with the delivery and pickup of packages. It was good to see the business grow, but another person added to the complexity of coordinating pickups and deliveries. With the addition of a new person, he could no longer "warehouse" the packages out of his delivery van. He now needed a central warehouse where he could organize and distribute packages for delivery. He thought that if his business grew enough to add one more delivery person he would also need someone at the warehouse to coordinate the arrival and distribution of all the packages.

1. Who are the stakeholders for On the Spot? How involved should On the Spot's customers be in system definition? As the business grows, who else might be potential stakeholders and interested in system functions?
2. If you were commissioned to build a system for Bill, how would you determine the requirements? Be specific in your answer. Make a list of the questions you need answered.

(continued on page 65)

(continued from page 64)

3. What technology and communication requirements do you see? What are the hardware requirements, and what kind of equipment will provide viable options to the system? What would you recommend to Bill?
4. What are the primary functional requirements for the system as described so far in the case?

Sandia Medical Devices

Medical monitoring technology has advanced significantly in the last decade. Monitoring that once required a visit to a health-care facility can now be performed by devices located in a patient's home or carried or worn at all times. Examples include glucose level (blood sugar), pulse, blood pressure, and electrocardiogram (EKG). Measurements can be transmitted via telephone, Internet connection, and wireless data transmission standards, such as Bluetooth. A particularly powerful technology combination is a wearable device that records data periodically or continuously and transmits it via Bluetooth to a cell phone app. The cell phone app can inform the patient of problems and can automatically transmit data and alerts to a central monitoring application (see **Figure 2-17**).

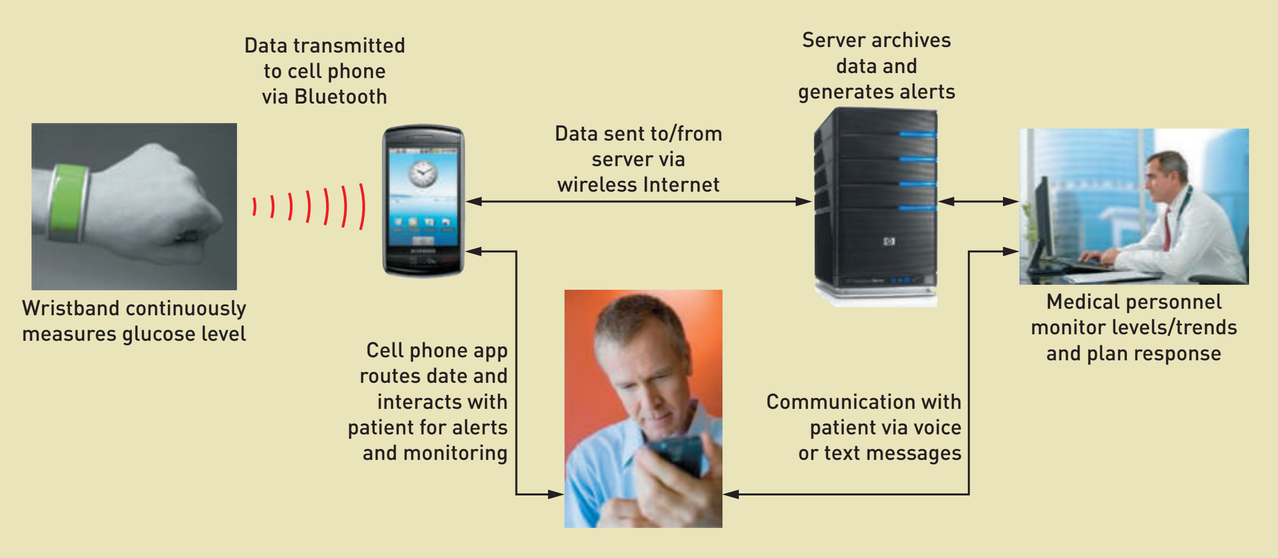
Health-care providers and patients incur significant costs when glucose levels are not maintained within acceptable tolerances. Short-term episodes of very high or very low glucose often result in expensive visits to urgent care clinics or hospitals. In addition, patients with frequent but less severe episodes of high or low glucose are more susceptible to such expensive long-term complications as vision, circulatory, and kidney problems.

Sandia Medical Devices (SMD), an Albuquerque manufacturer of portable and wearable medical monitoring devices, has developed a glucose monitor embedded in a wristband. The device is powered by body heat and senses glucose levels from minute quantities of perspiration. SMD is developing the Real-Time Glucose Monitoring (RTGM) device in partnership with New Mexico Health Systems (NMHS), a comprehensive health delivery service with patients throughout New Mexico, The system's vision statement reads as follows:

> *RTGM will enable patients and their healthcare providers to continuously monitor glucose levels, immediately identify short- and long-term medical dangers, and rapidly respond to those dangers in medically appropriate ways.*

SMD will develop the initial prototype software for smartphones with Bluetooth capability running the Google Android operating system. If successful,

FIGURE 2-17 *Data movement among devices and users*



(continued on page 66)

(continued from page 65)

NMHS and its patients will have free use of the software and SMD will resell the software to other health systems worldwide.

1. Who are RTGM's stakeholders? Should NMHS's patients be included in defining the system requirements? Why or why not? Should RTGM interact with medical professionals other than physicians? Why or why not?
2. If you were the lead analyst for RTGM, how would you determine the requirements? Be specific in your answer. List several questions you need answered.
3. What are the primary functional requirements for the system as described so far in the case?
4. Are the parameters for alerting patients and medical personnel the same for every patient? Can they vary over time for the same patient? What are the implications for the system's functional requirements?
5. Briefly describe some possible nonfunctional requirements for RTGM.

Further Resources

Soren Lauesen, *Software Requirements: Styles and Techniques*. Addison-Wesley, 2002.

Stan Magee, *Guide to Software Engineering Standards and Specifications*. Artech House, 1997.

Suzanne Robertson and James Robertson, *Mastering the Requirements Process*, Second Edition. Addison-Wesley, 2006.

Karl Wiegers, *Software Requirements*. Microsoft Press, 2003.

Karl Wiegers, *More About Software Requirements: Thorny Issues and Practical Advice*. Microsoft Press, 2006.

Ralph Young, *The Requirements Engineering Handbook*. Artech House, 2003.