

# CHAPTER 3

## REQUIREMENTS DETERMINATION

**D**uring the analysis phase, the analyst determines the functional requirements for the new system. This chapter begins by describing the analysis phase and its primary deliverable, the system proposal. The concept of a requirement is explained and several categories of requirements are defined. The purpose and structure of the requirements definition statement is outlined. Techniques to elicit requirements are discussed, including interviews, JAD sessions, questionnaires, document analysis, and observation. Finally, several requirements analysis strategies are described to help the analyst discover requirements.

### OBJECTIVES

- Explain the analysis phase of the SDLC.
- Describe the content and purpose of the requirements definition statement.
- Classify requirements correctly as business, user, functional, or nonfunctional requirements.
- Employ the requirement elicitation techniques of interviews, JAD sessions, questionnaires, document analysis, and observation.
- Define the role that each requirement elicitation technique plays in determining requirements.
- Describe several analysis strategies that can help the analyst discover requirements.

### CHAPTER OUTLINE

Introduction

The Analysis Phase

Requirements Determination

*What Is a Requirement?*

*The Process of Determining*

*Requirements*

*The Requirements Definition*

*Statement*

Requirements Elicitation Techniques

*Requirements Elicitation in Practice*

*Interviews*

*Joint Application Development (JAD)*

*Questionnaires*

*Document Analysis*

*Observation*

*Selecting the Appropriate Techniques*

Requirements Analysis Strategies

*Problem Analysis*

*Root Cause Analysis*

*Duration Analysis*

*Activity-Based Costing*

*Informal Benchmarking*

*Outcome Analysis*

*Technology Analysis*

*Activity Elimination*

*Comparing Analysis Strategies*

Applying the Concepts at Tune Source

*Eliciting and Analyzing*

*Requirements*

*Requirements Definition*

*System Proposal*

Summary

IMPLEMENTATION

## INTRODUCTION

---

Part 2 of this textbook focuses on the analysis phase of the SDLC. The work performed in the analysis phase involves expanding the vision described in the system request into a thorough, detailed understanding of exactly what the new system needs to do. As the detailed understanding of what the new system must do evolves, those details will be expressed and documented in several ways, including a detailed requirements definition statement (this chapter), use cases (Chapter 4), process models (Chapter 5), and data model (Chapter 6). Although the structure of a textbook requires that these topics are presented sequentially, in practice, the systems analyst uses all of the tools and techniques discussed in Chapters 3 through 6 throughout the analysis phase to define, clarify, and document the requirements for the new system.

## THE ANALYSIS PHASE

---

The analysis phase is so named because the term *analysis* refers to breaking a whole into its parts with the intent of understanding the parts' nature, function, and interrelationships. In the context of the SDLC, the outputs of the planning phase (the system request, feasibility study, and project plan), outline the business goals for the new system, define the project's scope, assess project feasibility, and provide the initial work plan. These planning phase deliverables are the key inputs into the analysis phase. In the analysis phase, the systems analyst works extensively with the business users of the new system to understand their needs from the new system. The basic process of *analysis* involves three steps:

- Understand the existing situation (the as-is system).
- Identify improvements.
- Define requirements for the new system (the to-be system).

Sometimes the first step (i.e., understanding the as-is system) is skipped or done in a limited manner. This happens when no current system exists, if the existing system and processes are irrelevant to the future system, or if the project team is using a RAD or agile development methodology in which the as-is system is not emphasized. Traditional methods such as waterfall and parallel development (see Chapter 2) typically allot significant time to understanding the as-is system and identifying improvements before moving to capture requirements for the to-be system. Newer RAD and agile methodologies, such as iterative development, system prototyping, throwaway prototyping, and extreme programming (see Chapter 2), focus almost exclusively on improvements and the to-be system requirements, and they devote little time for investigating the current as-is system. Experience shows that it is useful to study the current situation whenever possible. The insights gained from reviewing the existing system can be quite valuable to the project team.

To move the users “from here to there,” an analyst needs strong *critical thinking skills*. Critical thinking is the ability to recognize strengths and weaknesses and recast an idea in an improved form. These skills are needed in order for the analyst to understand issues and develop new and improved business processes that are supported by information system technologies. These skills are essential in examining the results of requirements discovery and translating those requirements into a concept for the new system.

As an example, let's say that a user states that the new system should "eliminate inventory stock-outs." While this might be a worthy project goal, the analyst needs to think about it critically in order to formulate the statement in terms of useful requirements. The analyst could first have the users think about circumstances leading to stock-outs (e.g., supplier orders are not placed in a timely way), and then describe the issues that lead to these circumstances (e.g., on-hand inventory levels are updated only once a week; delays occur in identifying the best supply source for the items; delays occur in receiving approval of the supply order, etc.). By focusing on these issues, the team is in a better position to develop new business processes that address these concerns. The new requirements will then be based on the issues that truly need to be fixed. In this case, the requirements might include, in part:

- The system shall update on-hand inventory levels twice per day.
- The system shall produce an out-of-stock notification immediately when an item quantity on hand reaches the item reorder point.
- The system shall include a recommended supplier with every out-of-stock notification.
- The system shall produce a supply purchase order that is sent to the appropriate manager for approval.
- The system shall send an approved supply purchase order to the supplier via secure electronic communication.

As this example demonstrates, the analyst cannot realistically expect that the true requirements for the new system are easily gathered following a few conversations with the stakeholders. The analyst must be prepared to dig into the situation and discover requirements. This is not often an easy process.

A number of techniques and tools can be used by the analyst to facilitate this process of discovering requirements. In this chapter, we will describe those techniques and tools so that you can learn how to use them during the analysis phase. We will also explain the critical role that requirements play in defining the new system. As mentioned above, the analyst also employs tools during this phase that are the subject of complete chapters: use cases (Chapter 4), process models (Chapter 5), and data models (Chapter 6).

The final deliverable of the analysis phase is the *system proposal*, which compiles the detailed requirements definition statement, use cases, process models, and data model together with a revised feasibility analysis and work plan. At the conclusion of the analysis phase, the system proposal is presented to the approval committee, usually in the form of a system *walk-through*. The goal of the walk-through is to explain the system in moderate detail so that the users, managers, and key decision makers clearly understand it, can identify any needed modifications, and are able to make a decision about whether the project should continue. Before moving into the design phase, the project should be reviewed to ensure that it continues to contribute business value to the organization. If approved, the system proposal components (requirements definition, use cases, process models, and data model) are used as inputs to the steps in the design phase, which further refine them and define in much more detail how the system will be built.

The line between the analysis and design phases is very blurry, because the deliverables created in the analysis phase are really the first step in the design of the new system. Many of the major design decisions for the new system are found in the analysis deliverables. In fact, a better name for the analysis phase would really

be “analysis and initial design,” but because this name is rather long and because most organizations simply call this phase “analysis,” we will, too. Nonetheless, it is important to remember that the deliverables from the analysis phase are really the first step in the design of the new system.

In many ways, determining requirements is the single most critical aspect of the entire SDLC. Although many factors contribute to the failure of systems development projects, failing to determine the correct requirements is a primary cause.<sup>1</sup> A 2008 study of Fortune 500 company software projects found just 37% of survey respondents felt the project met users’ needs.<sup>2</sup> Therefore, analysts should devote considerable attention to the work performed in the analysis phase. It is here that the major elements of the system first begin to emerge. If the requirements are later found to be incorrect or incomplete, significant rework may be needed, adding substantial time and cost to the project.

During requirements determination, the to-be system concept is easy to change because little work has been done yet. As the system moves through the subsequent SDLC phases (design and implementation), it becomes harder and harder to return to requirements determination and make major changes because of all of the rework that is involved. This is why the iterative approaches of many RAD and agile methodologies are so effective—small batches of requirements can be identified and implemented in incremental stages, allowing the overall system to change and evolve over time. Also, methodologies such as the V-model stress that tests for the system should be defined at the same time that the requirements are being defined. That way, testing is not just a last-minute, thrown-together process, but instead is based directly on the requirements of the system as they are being defined.

## REQUIREMENTS DETERMINATION

---

*Requirements determination* is performed to transform the system request’s high-level statement of business requirements into a more detailed, precise list of what the new system must do to provide the needed value to the business. This detailed list of requirements is supported, confirmed, and clarified by the other activities of the analysis phase: creating use cases, building process models, and building a data model. We first explain what a requirement is and discuss the process of creating a requirements definition statement.

### What Is a Requirement?

A *requirement* is simply a statement of what the system must do or what characteristics it needs to have. During a systems development project, requirements will be created that describe what the business needs (*business requirements*); what the users need to do (*user requirements*); what the software should do (*functional requirements*); characteristics the system should have (*nonfunctional requirements*); and how the system should be built (*system requirements*). Although this list of requirement

<sup>1</sup> For example, see Kweku Ewusi-Mensah, *Software Development Failures: Anatomy of Failed Projects*, MIT Press, 2003.

<sup>2</sup> Janet Mullaney, “Requirements gathering resources, practices lacking at Fortune 500 companies,” SearchSoftwareQuality.com, Aug. 20, 2008.

categories may seem intimidating at first, the categories merely reflect the purpose of the requirements and the stage in the SDLC in which they are defined.

We have already discussed the creation of the systems request in the planning phase of the SDLC. In the system request, there are statements that describe the reasons for proposing the systems development project. These statements reflect the business requirements that this system, if built, will fulfill. These business requirements help define the overall goals of the system and help clarify the contributions it will make to the organization's success. Examples of business requirements include: "Increase market share"; "Shorten order processing time"; "Reduce customer service costs"; "Lower inventory spoilage"; "Improve responsiveness to customer service requests"; and "Provide account access to mobile customers." When the systems development project is complete, success will be measured by evaluating whether the stated business requirements have actually been achieved; therefore, they provide the overall direction for the project.

During the analysis phase, requirements are written from the perspective of the business, and they focus on *what* the system needs to do in order to satisfy business user needs. A good starting place is to concentrate on what the user actually needs to accomplish with the system in order to fulfill a needed job or task. These user requirements describe tasks that the users perform as an integral part of the business' operations, such as: "Schedule a client appointment"; "Place a new customer order"; "Re-order inventory"; "Determine available credit"; and "Look up account balances." Use cases (discussed in Chapter 4) are tools used to clarify the steps involved in performing these user tasks. By understanding what the user needs to do in terms of tasks to perform, the analyst can then determine ways in which the new system can support the users' needs.

Determining ways in which the new system can support user needs leads to statements of the system's functional requirements. A functional requirement relates directly to a process the system has to perform as a part of supporting a user task and/or information it needs to provide as the user is performing a task. The International Institute of Business Analysis (IIBA) defines functional requirements as "the product capabilities, or things that a product must do for its users."<sup>3</sup> Functional requirements begin to define how the system will support the user in completing a task. For example, assume the user requirement is "Schedule a client appointment." The functional requirements associated with that task include: "Determine client availability," "Find available openings matching client availability," "Select desired appointment," "Record appointment," and "Confirm appointment." Notice how these functional requirements expand upon the user's task to describe capabilities and functions that the system will need to include, allowing the user to complete the task.

As the analyst works with the business users of the system to discover user and functional requirements, the user may reveal processes that will be needed or information that will be needed. For example, as shown in Figure 3-1, the user may state "The system must retain customer order history for three years" (an information need). The analyst should probe for the reasoning behind this statement, such as "The system should allow registered customers to review their own order history for the past three years" (a process need). Similarly, the user may state "The system should check incoming customer orders for inventory availability" (a process need). An alert analyst will recognize the related information need, "The system should

<sup>3</sup> International Institute of Business Analysis, *Guide to Business Analysis Body of Knowledge® (BABOK®)*, 2<sup>nd</sup> Ed.

Functional Requirement	Description	Examples
Process-oriented	A process the system must perform; a process the system must do	<ul style="list-style-type: none"> <li>■ The system must allow registered customers to review their own order history for the past three years.</li> <li>■ The system must check incoming customer orders for inventory availability.</li> <li>■ The system should allow students to view a course schedule while registering for classes.</li> </ul>
Information-oriented	Information the system must contain	<ul style="list-style-type: none"> <li>■ The system must retain customer order history for three years.</li> <li>■ The system must include realtime inventory levels at all warehouses.</li> <li>■ The system must include budgeted and actual sales and expense amounts for current year and three previous years.</li> </ul>

**FIGURE 3-1**  
Functional Requirements

maintain real-time inventory levels at all warehouses.” All of these requirements are necessary to fully understand the system that is being developed.

Process models (Chapter 5) are used to explain the relationship of functions/processes to the system users, how the functions/processes relate to each other, how data is entered and produced by functions/processes, and how functions/processes create and use stored data. Process models help clarify the software components that will be needed to accomplish the functional requirements. In addition, the functional requirements begin to define the data that must be kept track of in order to accomplish the user tasks. The data component of the system is defined in the data model (Chapter 6).

## YOUR

## T U R N

### 3-1 IDENTIFYING REQUIREMENTS

One of the most common mistakes made by new analysts is to confuse functional and non-functional requirements. Pretend that you received the following list of requirements for a sales system:

Requirements for Proposed System:  
The system should...

1. be accessible to Web users.
2. include the company standard logo and color scheme.
3. restrict access to profitability information.
4. include actual and budgeted cost information.
5. provide management reports.
6. include sales information that is updated at least daily.
7. have 2-second maximum response time for predefined queries and 10-minute maximum response time for ad hoc queries.
8. include information from all company subsidiaries.
9. print subsidiary reports in the primary language of the subsidiary.
10. provide monthly rankings of salesperson performance.

#### QUESTIONS:

1. Which requirements are functional business requirements? Provide two additional examples.
2. Which requirements are nonfunctional business requirements? What kind of nonfunctional requirements are they? Provide two additional examples.



User requirements and functional requirements defined in the analysis phase will flow into the design phase, where they evolve to become more technical, describing *how* the system will be implemented. Requirements in the design phase reflect the developer's perspective, and they usually are called *system requirements*. These requirements focus on describing how to create the software product that will be produced from the project. More will be said about system requirements in Part 3 of the textbook.

Before we continue, we want to stress that it can be difficult to draw a black-and-white dividing line between these categories of requirement—and, confusingly, some companies use the terms interchangeably. The important thing to remember is that a requirement is a statement of what the system must do, and the focus of requirements will change over time as the project moves from planning to analysis to design to implementation. Requirements evolve from broad statements of overall business needs from the system to detailed statements of the business capabilities that a system should support to detailed technical statements of the way in which the capabilities will be implemented in the new system.

The final category of requirements is *nonfunctional requirements*. The IIBA defines this group of requirements as “the quality attributes, design, and implementation constraints, and external interfaces which a product must have.”<sup>4</sup> Although the term “nonfunctional” is not very descriptive, this requirement category includes important behavioral properties that the system must have, such as performance and usability. The ability to access the system through a mobile device would be considered a nonfunctional requirement. Nonfunctional requirements are primarily used in the design phase when decisions are made about the user interface, the hardware and software, and the system's underlying architecture. Many of these requirements will be discovered during conversations with users in the analysis phase, however, and should be recorded as they are discovered.

Figure 3-2 lists different kinds of nonfunctional requirements and examples of each kind. Notice that the nonfunctional requirements describe a variety of system characteristics: operational, performance, security, and cultural and political. These characteristics do not describe business processes or information, but they are very important in understanding what the final system should be like. For example, the project team needs to know whether a system must be highly secure, requires sub-second response time, or has to reach a multilingual customer base. These requirements will affect design decisions that will be made in the design phase, particularly architecture design, so we will revisit them in detail in Chapter 8. The goal at this point is to identify any major issues. In addition, if the methodology in use includes developing test plans during analysis, then these requirements will be important in establishing testing benchmarks that will be needed later.

## The Process of Determining Requirements

Both business and IT perspectives are needed to determine requirements during the analysis phase. Systems analysts may not understand the true business needs of the users. A recent study by the Standish Group found that the lack of user involvement is the top reason for IT project failure.<sup>5</sup> On the other hand, the business users may

<sup>4</sup> Ibid.

<sup>5</sup> Frank Hayes, “Chaos is back,” *Computerworld*, November 8, 2004.

Nonfunctional Requirement	Description	Examples
Operational	The physical and technical environments in which the system will operate	<ul style="list-style-type: none"> <li>■ The system can run on handheld devices.</li> <li>■ The system should be able to integrate with the existing inventory system.</li> <li>■ The system should be able to work on any Web browser.</li> </ul>
Performance	The speed, capacity, and reliability of the system	<ul style="list-style-type: none"> <li>■ Any interaction between the user and the system should not exceed 2 seconds.</li> <li>■ The system downloads new status parameters within 5 minutes of a change.</li> <li>■ The system should be available for use 24 hours per day, 365 days per year.</li> <li>■ The system supports 300 simultaneous users from 9–11 A.M.; 150 simultaneous users at all other times.</li> </ul>
Security	Who has authorized access to the system under what circumstances	<ul style="list-style-type: none"> <li>■ Only direct managers can see personnel records of staff.</li> <li>■ Customers can see their order history only during business hours.</li> <li>■ The system includes all available safeguards from viruses, worms, Trojan horses, etc.</li> </ul>
Cultural and Political	Cultural and political factors and legal requirements that affect the system	<ul style="list-style-type: none"> <li>■ The system should be able to distinguish between U.S. currency and currency from other nations.</li> <li>■ Company policy is to buy computers only from Dell.</li> <li>■ Country managers are permitted to authorize custom user interfaces within their units.</li> <li>■ Personal information is protected in compliance with the Data Protection Act.</li> </ul>

Source: The Atlantic Systems Guild, <http://www.systemsguild.com>

**FIGURE 3-2**  
Nonfunctional Requirements

## CONCEPTS

### 3-A WHAT CAN HAPPEN IF YOU IGNORE NONFUNCTIONAL REQUIREMENTS

#### IN ACTION

I once worked on a consulting project in which my manager created a requirements definition without listing nonfunctional requirements. The project was then estimated based on the requirements definition and sold to the client for \$5,000. In my manager's mind, the system that we would build for the client would be a very simple stand-alone system running on current technology. It shouldn't take more than a week to analyze, design, and build.

Unfortunately, the client had other ideas. They wanted the system to be used by many people in three different departments, and they wanted the ability for any number of people to work on the system concurrently. The

technology they had in place was antiquated, but nonetheless they wanted the system to run effectively on the existing equipment. Because we didn't set the project scope properly by including our assumptions about nonfunctional requirements in the requirements definition, we basically had to do whatever they wanted.

The capabilities they wanted took weeks to design and program. The project ended up taking four months, and the final project cost was \$250,000. Our company had to pick up the tab for everything except the agreed upon \$5,000. This was by far the most frustrating project situation I ever experienced.

*Barbara Wixom*



not be aware of the opportunities that a new technology may offer. It is important that the team carefully considers the underlying business process and how best to support that business process with information system technology.

A good analogy is building a house or an apartment. We have all lived in a house or apartment, and most of us have some understanding of what we would like in our homes. If we were asked to design a dwelling from scratch, however, it would be a challenge because we lack appropriate design skills and technical engineering skills. Likewise, an architect acting alone would probably miss some of our unique requirements.

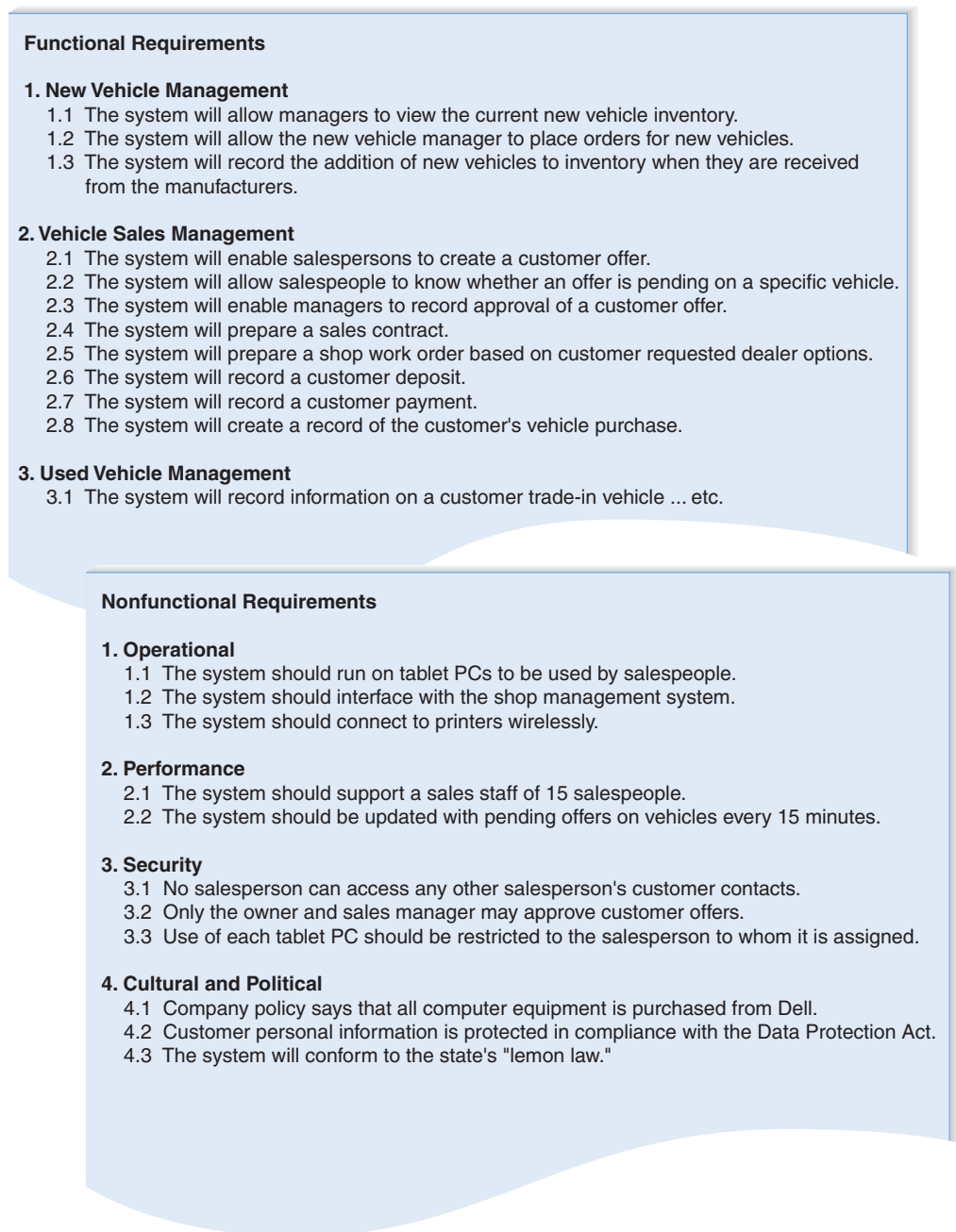
Therefore, the most effective approach is to have both businesspeople and analysts working together to determine requirements. In fact, the analysis phase involves significant interactions with people who have an interest in the new system (often called stakeholders). One of the first tasks for the analyst is to identify the primary sources of requirements, including the project sponsor, project champion(s), all users of the system (both direct and indirect), and possibly others. It is important that all user perspectives are included.

The analyst must also consider how best to elicit the requirements from the stakeholders. There are a variety of elicitation techniques that can be used to acquire information, including interviews, questionnaires, observation, joint application development (JAD), and document analysis. We will discuss these techniques in the next section. The information gathered by these techniques is critically analyzed and used to craft the requirements definition statement. The analyst works with the entire project team and the business users to verify, change, and complete the list of requirements and, if necessary, to prioritize the importance of the requirements that are identified. During this process, use cases, process models, and data models may be used to clarify and define the ideas for the new system. This process continues throughout the analysis phase, and the requirements definition evolves over time as new requirements are identified and as the project moves into later phases of the SDLC.

Beware: The evolution of the requirements definition must be carefully managed. Keeping the requirements list tight and focused is a key to project success. The project team cannot keep adding new items to the requirements definition or the system will keep growing and growing and never get finished. Instead, the project team carefully identifies requirements and evaluates which ones fit within the system scope. When a requirement reflects a real business need, but is not within the scope of the current system or current release, it should be evaluated in terms of its importance and impact on time and budget. It may be that the requirement is essential enough to add to the current project, along with appropriate adjustments to the project scope, budget, and time frame. We should not assume that the requirements for the project can never be changed. However, it is also possible that the requirement might be added to a list of future requirements or given a low priority. The management of requirements (and system scope) is one of the hardest parts of managing a project!

## The Requirements Definition Statement

The requirements definition statement—usually just called the *requirements definition*—is a straightforward text report that simply lists the functional and non-functional requirements in an outline format. Figure 3-3 shows a sample requirements definition for Holiday Travel Vehicles, a fictitious recreational vehicle dealership.



**FIGURE 3-3**  
Sample Requirements Definition

As shown in Figure 3-3, it is common to number the requirements in a legal or outline format so that each requirement is clearly identified. It is important that the requirements be identified with unique numbers so that each requirement can be easily tracked through the entire development process. For clarity, the requirements are typically grouped into functional and nonfunctional groupings. Then,

within each of those groups, they are classified further by the type of requirement or by business area.

Sometimes, requirements are prioritized on the requirements definition statement. They can be ranked as having “high,” “medium,” or “low” importance in the new system, or they can be labeled with the version of the system that will address the requirement (e.g., release 1, release 2, release 3). This practice is particularly important with RAD methodologies that deliver requirements in batches by developing incremental versions of the system.

The most *obvious* purpose of the requirements definition is to provide a clear statement of what the new system should do in order to achieve the system vision described in the system request. The use cases, process models, and data models provide additional explanatory content in different formats. A critically *important* purpose of the requirements definition, however, is to define the scope of the system. The document describes to the analysts exactly what the final system needs to do. In addition, it serves to establish the users’ expectations for the system. If and when discrepancies or misunderstandings arise, the document serves as a resource for clarification.

## REQUIREMENTS ELICITATION TECHNIQUES

---

An analyst is very much like a detective (and business users sometimes are like elusive suspects). He or she knows that there is a problem to be solved and therefore must look for clues that uncover the solution. Unfortunately, the clues are not always obvious (and often are missed), so the analyst needs to notice details, talk with witnesses, and follow leads, just as Sherlock Holmes would have done. The best analysts will thoroughly search for requirements using a variety of techniques and make sure that the current business processes and the needs for the new system are well understood before moving into design. You don’t want to discover later that you have key requirements wrong—surprises like this late in the SDLC can cause all kinds of problems.

### Requirements Elicitation in Practice

Before discussing the five requirements elicitation techniques in detail, a few practical tips are in order. First, the analyst should recognize that important side effects of the requirements definition process include building political support for the project and establishing trust and rapport between the project team and the ultimate users of the system. Every contact and interaction between the analyst and a potential business user or manager is an opportunity to generate interest, enthusiasm, and commitment to the project. Therefore, the analyst should be prepared to make good use of these opportunities as they arise during the requirements definition process.

Second, the analyst should carefully determine who is included in the requirements definition process. The choice to include (or exclude) someone is significant; involving someone in the process implies that the analyst views that person as an important resource and values his or her opinions. You *must* include all of the key *stakeholders* (the people who can affect the system or who will be affected by the system). This might include managers, employees, staff members,

and even some customers and suppliers. Also, be sensitive to the fact that some people may have significant influence within the organization even if they do not rank high in the formal organizational hierarchy. If you do not involve a key person, that individual may feel slighted, causing problems during implementation (e.g., saying “I could have told them this might happen, but they didn’t ask *me*!”).

Finally, do everything possible to respect the time commitment that you are asking the participants to make. The best way to do this is to be fully prepared and to make good use of all the types of requirements elicitation techniques. Although, as we will see, interviewing is the most commonly used technique, other indirect methods may help the analyst develop a basic understanding of the business domain so that the direct techniques are more productive. In general, a useful strategy for the analyst to employ is to begin requirements gathering by interviewing senior managers to gain an understanding of the project and get the “big picture.” These preliminary interviews can then be followed by document analysis and, possibly, observation of business processes to learn more about the business domain, the vocabulary, and the as-is system. More interviews may then follow to collect the rest of the information needed to understand the as-is system.

In our experience, identifying improvements is most commonly done through JAD sessions because these sessions enable the users and key stakeholders to work together and create a shared understanding of the possibilities for the to-be system. Occasionally, these JAD sessions are followed by questionnaires sent to a much larger group of users or potential users to get a broad range of opinions. The concept for the to-be system is frequently developed through interviews with senior managers, followed by JAD sessions with users of all levels, to make sure that the key requirements of the new system are well understood.

In this section, we focus on the five most commonly used requirements elicitation techniques: interviews, JAD sessions, questionnaires, document analysis, and observation.

## Interviews

The *interview* is the most commonly used requirements elicitation technique. After all, it is natural—usually, if you need to know something, you ask someone. In general, interviews are conducted one on one (one interviewer and one interviewee), but sometimes, due to time constraints, several people are interviewed at the same time. There are five basic steps to the interview process: selecting interviewees, designing interview questions, preparing for the interview, conducting the interview, and postinterview follow-up.<sup>6</sup>

**Selecting Interviewees** An *interview schedule* should be created, listing who will be interviewed, the purpose of the interview, and where and when it will take place. (See Figure 3-4.) The schedule can be an informal list that is used to help set up meeting times or a formal list that is incorporated into the work plan. The people who appear on the interview schedule are selected on the basis of the analyst’s information needs. The project sponsor, key business users, and other members of the project team can help the analyst determine who in the organization can best provide important information about requirements. These people are listed on the interview schedule in the order in which they should be interviewed.

<sup>6</sup> A good book on interviewing is Brian James, *The Systems Analysis Interview*, Manchester: NCC Blackwell, 1989.

**FIGURE 3-4**  
Sample Interview Schedule

Name	Position	Purpose of Interview	Meeting
Andria McClellan	Director, Accounting	Strategic vision for new accounting system	Mon, March 1 8:00–10:00 A.M.
Jennifer Draper	Manager, Accounts Receivable	Current problems with accounts receivable process; future goals	Mon, March 1 2:00–3:15 P.M.
Mark Goodin	Manager, Accounts Payable	Current problems with accounts payable process; future goals	Mon, March 1 4:00–5:15 P.M.
Anne Asher	Supervisor, Data Entry	Accounts receivable and payable processes	Wed, March 3 10:00–11:00 A.M.
Fernando Merce	Data Entry Clerk	Accounts receivable and payable processes	Wed, March 3 1:00–3:00 P.M.

People at different levels of the organization will have different viewpoints on the system, so it is important to include both managers who manage the processes and staff who actually perform the processes to gain both high-level and low-level perspectives on an issue. Also, the kinds of interview subjects that you need may change over time. For example, at the start of the project the analyst has a limited understanding of the as-is business process. It is common to begin by interviewing one or two senior managers to get a strategic view and then move to mid-level managers who can provide broad, overarching information about the business process and the expected role of the system being developed. Once the analyst has a good understanding of the big picture, lower-level managers and staff members can fill in the exact details of how the process works. Like most other things about systems analysis, this is an iterative process—starting with senior managers, moving to mid-level managers, then staff members, back to mid-level managers, and so on, depending upon what information is needed along the way.

It is quite common for the list of interviewees to grow, often by 50%–75%. As you interview people, you likely will identify more information that is needed and additional people who can provide the information.

## CONCEPTS

### 3-B SELECTING THE WRONG PEOPLE

#### IN ACTION

In 1990, I led a consulting team for a major development project for the U.S. Army. The goal was to replace eight existing systems used on virtually every Army base across the United States. The as-is process and data models for these systems had been built, and our job was to identify improvement opportunities and develop to-be process models for each of the eight systems.

For the first system, we selected a group of mid-level managers (captains and majors) recommended by their commanders as being the experts in the system under

construction. These individuals were the first and second line managers of the business function. The individuals were expert at managing the process, but did not know the exact details of how the process worked. The resulting to-be process model was very general and nonspecific. *Alan Dennis*

#### QUESTION:

Suppose you were in charge of the project. Create an interview schedule for the remaining seven projects.

**Designing Interview Questions** There are three types of interview questions: closed-ended questions, open-ended questions, and probing questions. *Closed-ended questions* require a specific answer. You can think of them as being similar to multiple choice or arithmetic questions on an exam. (See Figure 3-5.) Closed-ended questions are used when the analyst is looking for specific, precise information (e.g., how many credit card requests are received per day). In general, precise questions are best. For example, rather than asking “Do you handle a lot of requests?” it is better to ask “How many requests do you process per day?”

Closed-ended questions enable analysts to control the interview and obtain the information they need. However, these types of questions don’t uncover *why* the answer is the way it is, nor do they uncover information that the interviewer does not think to ask ahead of time.

*Open-ended questions* are those that leave room for elaboration on the part of the interviewee. They are similar in many ways to essay questions that you might find on an exam. (See Figure 3-5 for examples.) Open-ended questions are designed to gather rich information and give the interviewee more control over the information that is revealed during the interview. Sometimes the subjects the interviewee chooses to discuss uncover information that is just as important as the answer (e.g., if the interviewee talks only about other departments when asked for problems, it may suggest that he or she is reluctant to admit his or her own department’s problems).

The third type of question is the *probing question*. Probing questions follow up on what has just been discussed in order for the interviewer to learn more, and they often are used when the interviewer is unclear about an interviewee’s answer. They encourage the interviewee to expand on or to confirm information from a previous response, and they are a signal that the interviewer is listening and interested in the topic under discussion. Many beginning analysts are reluctant to use probing questions because they are afraid that the interviewee might be offended at being challenged or because they believe it shows that they didn’t understand what the interviewee said. When done politely, probing questions can be a powerful tool in requirements discovery.

In general, you should not ask questions about information that is readily available from other sources. For example, rather than asking what information is used to perform a task, it is simpler to show the interviewee a form or report (see document analysis later) and ask what information on it is used. This helps focus

Types of Questions	Examples
Closed-Ended Questions	<ul style="list-style-type: none"> <li>• How many telephone orders are received per day?</li> <li>• How do customers place orders?</li> <li>• What information is missing from the monthly sales report?</li> </ul>
Open-Ended Questions	<ul style="list-style-type: none"> <li>• What do you think about the way invoices are currently processed?</li> <li>• What are some of the problems you face on a daily basis?</li> <li>• What are some of the improvements you would like to see in the way invoices are processed?</li> </ul>
Probing Questions	<ul style="list-style-type: none"> <li>• Why?</li> <li>• Can you give me an example?</li> <li>• Can you explain that in a bit more detail?</li> </ul>

**FIGURE 3-5**  
Three Types of Questions

the interviewee on the task and saves time, because he or she does not need to describe the information in detail—he or she just needs to point it out on the form or report.

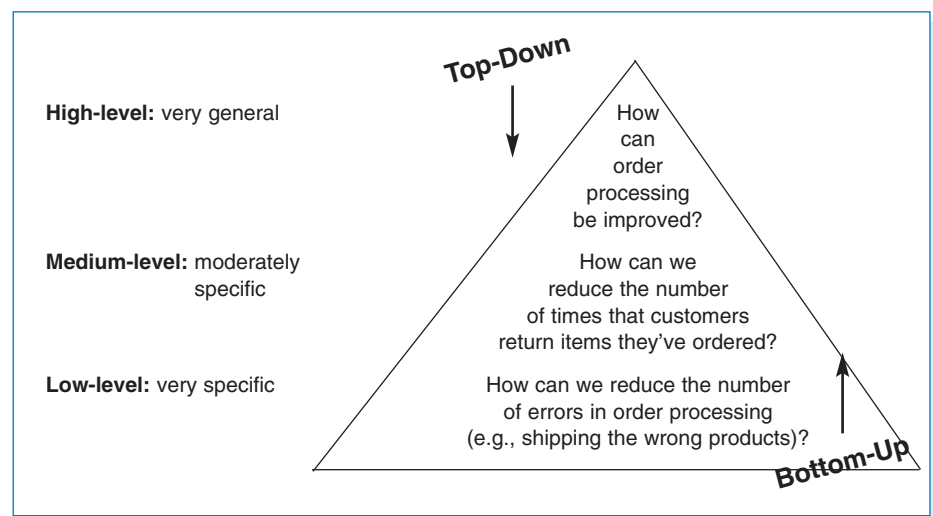
Your interview questions should anticipate the type of information the interviewee is likely to know. Managers are often somewhat removed from the details of daily business processes and so might be unable to answer questions about them, whereas lower-level staff members could readily respond. Conversely, lower-level employees may not be able to answer broad, policy-oriented questions, while managers could. Since no one wants to appear ignorant, avoid confounding your interviewees with questions outside their areas of knowledge.

No type of question is better than another, and usually a combination of questions is used during an interview. At the initial stage of an IS development project the as-is process can be unclear, so the interview process begins with *unstructured interviews*, interviews that seek a broad and roughly defined set of information. In this case, the interviewer has a general sense of the information needed, but few closed-ended questions to ask. These are the most challenging interviews to conduct because they require the interviewer to ask open-ended questions and probe for important information “on the fly.”

As the project progresses, the analyst comes to understand the business process much better, and he or she needs very specific information about how business processes are performed (e.g., exactly how a customer credit request is approved). At this time, the analyst conducts *structured interviews* in which specific sets of questions are developed prior to the interviews. There usually are more closed-ended questions in a structured interview than in the unstructured approach.

No matter what kind of interview is being conducted, interview questions must be organized into a logical sequence so that the interview flows well. For example, when trying to gather information about the current business process, the analyst will find it useful to move in logical order through the process or from the most important issues to the least important.

There are two fundamental approaches to organizing the interview questions: top-down or bottom-up; see Figure 3-6. With the *top-down interview*, the interviewer starts with broad, general issues and gradually works towards more specific



**FIGURE 3-6**  
Top-Down and Bottom-Up Questioning  
Strategies



ones. With the *bottom-up interview*, the interviewer starts with very specific questions and moves to broad questions. In practice, analysts mix the two approaches, starting with broad general issues, moving to specific questions, and then back to general issues.

The top-down approach is an appropriate strategy for most interviews. (It is certainly the most common approach.) The top-down approach enables the interviewee to become accustomed to the topic before he or she needs to provide specifics. It also enables the interviewer to understand the issues before moving to the details, because the interviewer may not have sufficient information at the start of the interview to ask very specific questions. Perhaps most importantly, the top-down approach enables the interviewee to raise a set of big-picture issues before becoming enmeshed in details, so the interviewer is less likely to miss important issues.

One case in which the bottom-up strategy may be preferred is when the analyst already has gathered a lot of information about issues and just needs to fill in some holes with details. Or, bottom-up may be appropriate if lower-level staff members feel threatened or are unable to answer high-level questions. For example, “How can we improve customer service?” may be too broad a question for a customer service clerk, whereas a specific question is readily answerable (e.g., “How can we speed up customer returns?”). In any event, all interviews should begin with non-controversial questions first and then gradually move into more contentious issues after the interviewer has developed some rapport with the interviewee.

**Preparing for the Interview** It is important to prepare for the interview in the same way that you would prepare to give a presentation. You should have a general interview plan which lists the questions that you will ask in the appropriate order; anticipates possible answers and provides how you will follow up with them; and identifies segues between related topics. Confirm the areas in which the interviewee has knowledge so you do not ask questions that he or she cannot answer. Review the topic areas, the questions, and the interview plan, and clearly decide which ones have the greatest priority in case you run out of time.

In general, structured interviews with closed-ended questions take more time to prepare than unstructured interviews. So, some beginning analysts prefer unstructured interviews, thinking that they can “wing it.” This is very dangerous and often counterproductive, because any information not gathered in the first interview would have to be obtained by follow-up efforts, and most people do not like to be interviewed repeatedly about the same issues.

Be sure to prepare the interviewee as well. When you schedule the interview, inform the interviewee of the reason for the interview and the areas you will be discussing far enough in advance so that he or she has time to think about the issues and organize his or her thoughts. This is particularly important when you are an outsider to the organization and for interviewing lower-level employees who often are not asked for their opinions and who may be uncertain about why you are interviewing them.

**Conducting the Interview** When you start the interview, the first goal is to build rapport with the interviewee so that he or she trusts you and is willing to tell you the whole truth, not just give the answers that he or she thinks you want. You should appear to be professional and an unbiased, independent seeker of information. The interview should start with an explanation of why you are there and why

you have chosen to interview the person, and then move into your planned interview questions.

It is critical to carefully record all the information that the interviewee provides. In our experience, the best approach is to take careful notes—write down *everything* the interviewee says, even if it does not appear immediately relevant. Don't be afraid to ask the person to slow down or to pause while you write, because this is a clear indication that the interviewee's information is important to you. One potentially controversial issue is whether or not to tape-record the interview. Recording ensures that you do not miss important points, but it can be intimidating for the interviewee. Most organizations have policies or generally accepted practices about the recording of interviews, so find out what they are before you start an interview. If you are worried about missing information and cannot tape the interview, then bring along a second person to take detailed notes.

As the interview progresses, it is important that you understand the issues that are discussed. If you do not understand something, be sure to ask. Don't be afraid to ask "dumb questions," because the only thing worse than appearing "dumb" is to *be* "dumb" by not understanding something that you could have cleared up by questioning. If you don't understand something during the interview, you certainly won't understand it afterward. Try to recognize and define jargon, and be sure to clarify jargon you do not understand. One good strategy to increase your understanding during an interview is to periodically summarize the key points that the interviewee is communicating. This avoids misunderstandings and also demonstrates that you are listening.

Finally, be sure to separate facts from opinion. The interviewee may say, for example, "We process too many credit card requests." This is an opinion, and it is useful to follow this up with a probing question requesting support for the statement (e.g., "Oh, how many do you process in a day?"). It is helpful to check the facts because any differences between the facts and the interviewee's opinions can point out key areas for improvement. Suppose that the interviewee complains about a high or increasing number of errors, but the logs show that errors have been decreasing. This suggests that errors are viewed as a very important problem that should be addressed by the new system, even if they are declining.

As the interview draws to a close, be sure to give the interviewee time to ask questions or provide information that he or she thinks is important but was not part of your interview plan. In most cases, the interviewee will have no additional concerns or information, but in some cases this will lead to unanticipated, but important information. Likewise, it can be useful to ask the interviewee if there are other people who should be interviewed. Make sure that the interview ends on time. (If necessary, omit some topics or plan to schedule another interview.)

As a last step in the interview, briefly explain what will happen next. (See the next section.) You don't want to prematurely promise certain features in the new system or a specific delivery date, but you do want to reassure the interviewee that his or her time was well spent and very helpful to the project.

Beginning systems analysts may naively think that conducting an interview is as easy as conversing with a friend. Unfortunately, this is almost never true. Interviewees often are not able or willing to hand over the needed information in a neat, organized fashion. In some cases, they may not want to share what they know at all. Analysts should hone their interpersonal skills to improve their interviewing success. (See Practical Tip 3-1.)

## PRACTICAL

## 3-1 DEVELOPING INTERPERSONAL SKILLS



## TIP

*Interpersonal skills* are those that enable you to develop rapport with others, and they are very important for interviewing. They help you to communicate with others effectively. Some people develop good interpersonal skills at an early age; they simply seem to know how to communicate and interact with others. Other people are less “lucky” and need to work hard to develop their skills.

Interpersonal skills, like most skills, can be learned. Here are some tips:

- **Don’t worry, be happy.** Happy people radiate confidence and project their feelings on others. Try interviewing someone while smiling and then interviewing someone else while frowning and see what happens!
- **Pay attention.** Pay attention to what the other person is saying (which is harder than you might think). See how many times you catch yourself with your mind on something other than the conversation at hand.
- **Summarize key points.** At the end of each major theme or idea that someone explains, you should repeat the

key points back to the speaker (e.g., “Let me make sure I understand. The key issues are ...”). This demonstrates that you consider the information important—and also forces you to pay attention. (You can’t repeat what you didn’t hear.)

- **Be succinct.** When you speak, be succinct. The goal in interviewing (and in much of life) is to learn, not to impress. The more you speak, the less time you give to others.
- **Be honest.** Answer all questions truthfully, and if you don’t know the answer, say so.
- **Watch body language (yours and theirs).** The way a person sits or stands conveys much information. In general, a person who is interested in what you are saying sits or leans forward, makes eye contact, and often touches his or her face. A person leaning away from you or with an arm over the back of a chair is disinterested. Crossed arms indicate defensiveness or uncertainty, while “steepling” (sitting with hands raised in front of the body with fingertips touching) indicates a feeling of superiority.

**Post-interview Follow-up** After the interview is over, the analyst needs to prepare an *interview report* that describes the information from the interview (Figure 3-7). The report contains *interview notes*, information that was collected over the course of the interview and is summarized in a useful format. In general, the interview report should be written within 48 hours of the interview, because the longer you wait, the more likely you are to forget information.

## CONCEPTS

## 3-C THE RELUCTANT INTERVIEWEE

## IN ACTION

Early in my consulting career I was sent to a client organization with the goal of interviewing the only person in the organization who knew how the accounts receivable system worked, and developing documentation for that system (nonexistent at the time). The interviewee was on time, polite, and told me absolutely nothing of value about the accounts receivable system, despite my best efforts over several interview sessions. Eventually, my manager called me off this

project, and our attempt to document this system was abandoned. *Roberta Roth*

## QUESTIONS:

1. Why do you suppose the interviewee was so uncooperative?
2. Can you think of any ways to avoid this failed outcome?

**FIGURE 3-7**  
Interview Report

Interview Notes Approved by: Linda Estey
<p><b>Person Interviewed: Linda Estey,</b> <b>Director, Human Resources</b></p> <p><b>Interviewer: Barbara Wixom</b></p> <p><b>Purpose of Interview:</b></p> <ul style="list-style-type: none"> <li>• Understand reports produced for Human Resources by the current system.</li> <li>• Determine information requirements for future system.</li> </ul> <p><b>Summary of Interview:</b></p> <ul style="list-style-type: none"> <li>• Sample reports of all current HR reports are attached to this report. The information that is not used and missing information are noted on the reports.</li> <li>• Two biggest problems with the current system are:             <ol style="list-style-type: none"> <li>1. The data are too old. (The HR Department needs information within 2 days of month end; currently information is provided to them after a 3-week delay.)</li> <li>2. The data are of poor quality. (Often, reports must be reconciled with the HR departmental database.)</li> </ol> </li> <li>• The most common data errors found in the current system include incorrect job-level information and missing salary information.</li> </ul> <p><b>Open Items:</b></p> <ul style="list-style-type: none"> <li>• Get current employee roster report from Mary Skudrna (extension 4355).</li> <li>• Verify calculations used to determine vacation time with Mary Skudrna.</li> <li>• Schedule interview with Jim VVack (extension 2337) regarding the reasons for data quality problems.</li> </ul> <p><b>Detailed Notes:</b> See attached transcript.</p>

Often, the interview report is sent to the interviewee with a request to read it and inform the analyst of clarifications or updates. Make sure the interviewee is convinced that you genuinely want his or her corrections to the report. Usually, there are few changes, but the need for any significant changes suggests that a second interview will be required. Never distribute someone's information without prior approval.

### Joint Application Development (JAD)

*Joint application development* (or *JAD* as it is more commonly known) is an information gathering technique that allows the project team, users, and management to work together to identify requirements for the system. IBM developed the JAD technique in the late 1970s, and it is often the most useful method for collecting information from users.<sup>7</sup> Capers Jones claims that JAD can reduce scope creep by 50%, and it prevents the requirements for a system from being too specific or too vague, both of which can cause trouble during later stages of the SDLC.<sup>8</sup> JAD is a structured process in which 10 to 20 users meet under the direction of a *facilitator* skilled in JAD techniques. The facilitator is a person who sets the meeting agenda and guides the discussion, but does not join in the discussion as a participant. He or she does not provide ideas or opinions on the topics under discussion and remains

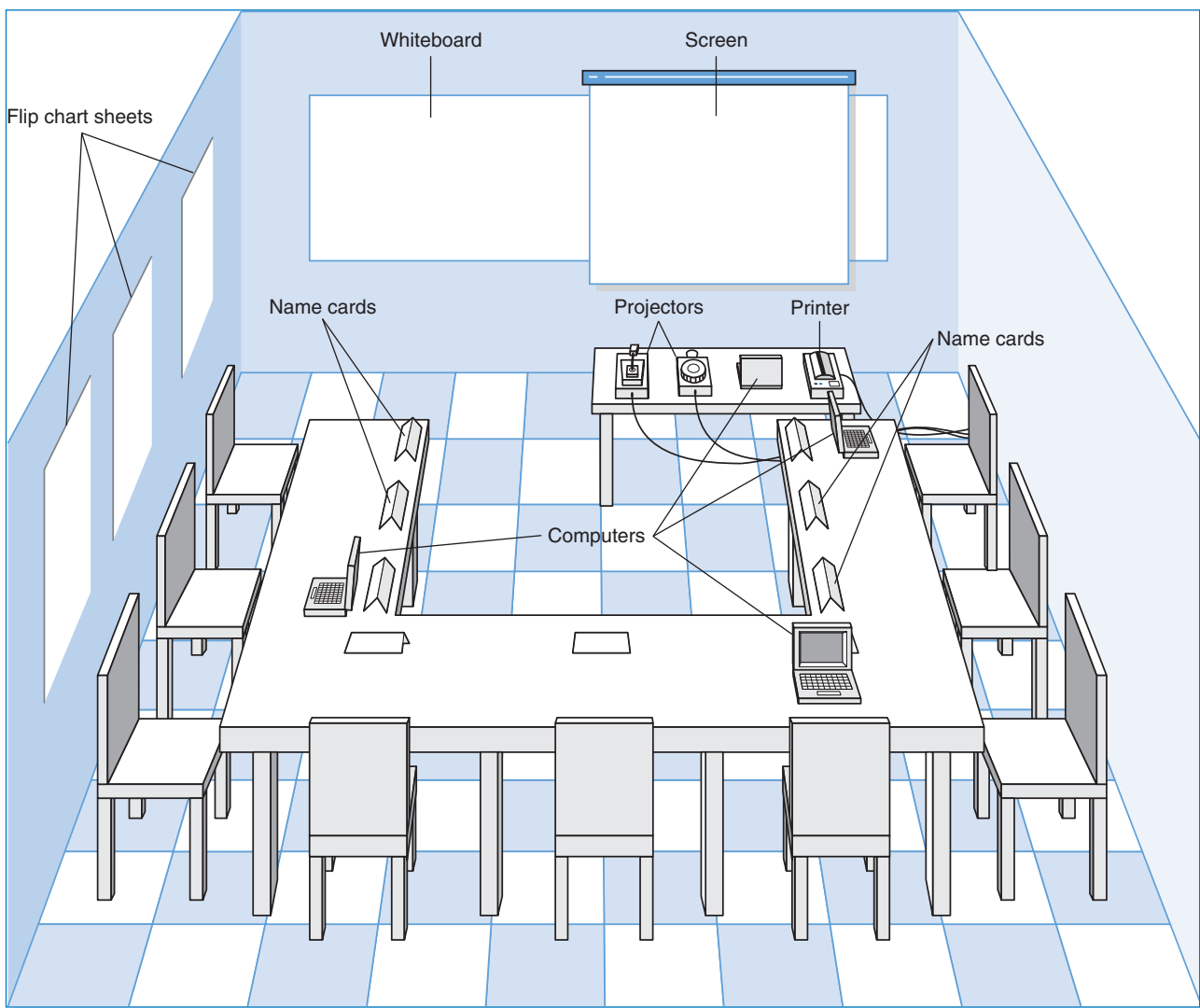
<sup>7</sup> More information on JAD can be found in J. Wood and D. Silver, *Joint Application Development*, New York: John Wiley & Sons, 1989; and Alan Cline, "Joint Application Development for Requirements Collection and Management," <http://www.carolla.com/wp-jad.htm>.

<sup>8</sup> See Kevin Strehlo, "Catching up with the Jones and 'Requirement' Creep," *InfoWorld*, July 29, 1996; and Kevin Strehlo, "The Makings of a Happy Customer: Specifying Project X," *Infoworld*, Nov 11, 1996.

neutral during the session. The facilitator must be an expert in both group process techniques and systems analysis and design techniques. One or two *scribes* assist the facilitator by recording notes, making copies, and so on. Often, the scribes will use computers and CASE tools to record information as the JAD session proceeds.

The JAD group meets for several hours, several days, or several weeks until all of the issues have been discussed and the needed information is collected. Most JAD sessions take place in a specially prepared meeting room, away from the participants' offices, so that they are not interrupted. The meeting room is usually arranged in a U shape so that all participants can easily see each other. (See Figure 3-8.) At the front of the room (the open part of the "U"), there is a whiteboard, flip chart and/or overhead projector for use by the facilitator, who leads the discussion.

One problem with JAD is that it suffers from the traditional problems associated with groups: Sometimes people are reluctant to challenge the opinions of others



**FIGURE 3-8**  
Joint Application Development Meeting Room

## YOUR

## TURN

## 3-2 INTERVIEW PRACTICE

Interviewing is not as simple as it first appears. Select two people from class to go to the front of the room to demonstrate an interview. (This also can be done in groups.) Have one person be the interviewer, and the other the interviewee. The interviewer should conduct a 5-minute interview regarding the school course registration system. Gather information about the existing system and how the system can be improved. If there is time, repeat with another pair.

**QUESTIONS:**

1. Describe the body language of the interview pair.
2. What kind of interview was conducted?
3. What kinds of questions were asked?
4. What was done well? How could the interview be improved?

(particularly their boss), a few people often dominate the discussion, and not everyone participates. In a 15-member group, for example, if everyone participates equally, then each person can talk for only 4 minutes each hour and must listen for the remaining 56 minutes—not a very efficient way to collect information.

*Electronic JAD*, or *e-JAD*, attempts to overcome these problems by the use of groupware. In an e-JAD meeting room, each participant uses special software on a networked computer to anonymously submit ideas, view all ideas generated by the group, and rate and rank ideas through voting. The facilitator uses the electronic tools of the e-JAD system to guide the group process, maintaining anonymity and enabling the group to focus on each idea's merits and not the power or rank of the person who contributed the idea. In this way, all participants can contribute at the same time, without fear of reprisal from people with differing opinions. Initial research suggests that e-JAD can reduce the time required to run JAD sessions by 50%–80%.<sup>9</sup>

**Selecting Participants** Selecting JAD participants is done in the same basic way as selecting interview participants. Participants are selected on the basis of information they can contribute, to provide a broad mix of organizational levels, and to build political support for the new system. The need for all JAD participants to be away from their offices at the same time can be a major problem. The office may need to be closed or run with a skeleton staff until the JAD sessions are complete.

Ideally, the participants who are released from regular duties to attend the JAD sessions should be the very best people in that business unit. However, without strong management support, JAD sessions can fail, because those selected to attend the JAD session are people who are less likely to be missed (i.e., the least competent people).

The facilitator should be someone who is an expert in JAD or e-JAD techniques and, ideally, someone who has experience with the business under discussion. In many cases, the JAD facilitator is a consultant external to the organization because the organization may not have a regular day-to-day need for JAD or e-JAD expertise. Developing and maintaining this expertise in-house can be expensive.

**Designing the JAD Session** JAD sessions can run from as little as a half day to several weeks, depending upon the size and scope of the project. In our experience, most JAD sessions tend to last 5 to 10 days spread over a 3-week period. Most e-JAD

<sup>9</sup> For more information on e-JAD, see A. R. Dennis, G. S. Hayes, and R. M. Daniels, "Business Process Modeling with Groupware," *Journal of Management Information Systems*, 1999, 15(4); 115–142.

sessions tend to last 1 to 4 days in a 1-week period. JAD and e-JAD sessions usually move beyond the collection of information into producing analysis deliverables. For example, the users and the analysts collectively can create use cases, process models, or the requirements definition.

As with interviewing, JAD success depends upon a careful plan. JAD sessions usually are designed and structured along the same principles as interviews. Most JAD sessions are designed to collect specific information from users, and this requires the development of a set of questions prior to the meeting. A difference between JAD and interviewing is that all JAD sessions are structured—they *must* be carefully planned. In general, closed-ended questions are seldom used, because they do not spark the open and frank discussion that is typical of JAD. In our experience, it is better to proceed top-down in JAD sessions when gathering information. Typically, 30 minutes is allocated to each separate agenda item, and frequent breaks are scheduled throughout the day because participants tire easily.

**Preparing for the JAD Session** As with interviewing, it is important to prepare the analysts and participants for the JAD session. Because the sessions can go beyond the depth of a typical interview and usually are conducted off-site, participants can be more concerned about how to prepare. It is important that the participants understand what is expected of them. If the goal of the JAD session, for example, is to develop an understanding of the current system, then participants can bring procedure manuals and documents with them. If the goal is to identify improvements for a system, then they can think about how they would improve the system prior to the JAD session.

**Conducting the JAD Session** Most JAD sessions try to follow a formal agenda, and most have formal *ground rules* that define appropriate behavior. Common ground rules include following the schedule, respecting others' opinions, accepting disagreement, and ensuring that only one person talks at a time.

The role of the JAD facilitator can be challenging. Many participants come to the JAD session with strong feelings about the system being discussed. Channeling these feelings so that the session moves forward in a positive direction and getting participants to recognize and accept—but not necessarily agree on—opinions and situations different from their own requires significant expertise in systems analysis and design, JAD, and interpersonal skills. Few systems analysts attempt to facilitate JAD sessions without being trained in JAD techniques, and most apprentice with a skilled JAD facilitator before they attempt to lead their first session.

The JAD facilitator performs three key functions. First, he or she ensures that the group sticks to the agenda. The only reason to digress from the agenda is when it becomes clear to the facilitator, project leader, and project sponsor that the JAD session has produced some new information that is unexpected and requires the JAD session (and perhaps the project) to move in a new direction. When participants attempt to divert the discussion away from the agenda, the facilitator must be firm, but polite, in leading the discussion back to the agenda and getting the group back on track.

Second, the facilitator must help the group understand the technical terms and jargon that surround the system development process and help the participants understand the specific analysis techniques used. Participants are experts in their business area, but they probably are not experts in systems analysis. The facilitator must therefore minimize the learning required and teach participants how to effectively provide the right information.

Third, the facilitator records the group's input on a public display area, which can be a whiteboard, flip chart, or computer display. He or she structures the information



that the group provides and helps the group recognize key issues and important solutions. Under no circumstance should the facilitator insert his or her opinions into the discussion. The facilitator *must* remain neutral at all times and simply help the group through the process. The moment the facilitator offers an opinion on an issue, the group will no longer see him or her as a neutral party, but rather as someone who could be attempting to sway the group into some predetermined solution.

However, this does not mean that the facilitator should not try to help the group resolve issues. For example, if two items appear to be the same to the facilitator, the facilitator should not say, “I think these may be similar.” Instead, the facilitator should ask, “Are these similar?” If the group decides that they are, the facilitator can combine them and move on. However, if the group decides that they are not similar (despite what the facilitator believes), the facilitator should accept the decision and move on. The group is *always* right, and the facilitator has no opinion.

It is common for the JAD participants to make use of a number of tools during the JAD session in order to fully define the new system. Use cases may be created to describe how the users will interact with the new system. Prototypes may be created to more fully understand the user interface or navigation through the system. Process models can be constructed to understand the software that will be developed, while a data model can be used to describe the data that will be captured and maintained. The facilitator and the analysts on the project team should use every tool at their disposal to help the participants clarify and define their needs for the new system.

**Post-JAD Follow-up** As with interviews, a JAD *post-session report* is prepared and circulated among session attendees. The post-session report is essentially the same as the interview report in Figure 3-7. Since the JAD sessions are longer and provide more information, it usually takes a week or two after the JAD session before the report is complete.

## Questionnaires

A questionnaire is a set of written questions for obtaining information from individuals. Questionnaires often are used when there is a large number of people from whom information and opinions are needed. In our experience, questionnaires are commonly used for systems intended for use outside of the organization (e.g., by customers or vendors) or for systems with business users spread across many geographic locations. Most people automatically think of paper when they think of questionnaires, but today more questionnaires are being distributed in electronic form, either via e-mail or on the Web. Electronic distribution can save a significant amount of money, compared with distributing paper questionnaires.

### YOUR

### 3-3 JAD PRACTICE

#### T U R N

**O**rganize yourselves into groups of four to seven people, and pick one person in each group to be the JAD facilitator. Using a blackboard, whiteboard, or flip chart, gather information about how the group performs some process (e.g., working on a class

assignment, making a sandwich, paying bills, getting to class). How did the JAD session go? Based on your experience, what are some pros and cons of using JAD in a real organization?

## PRACTICAL

## 3-2 MANAGING PROBLEMS IN JAD SESSIONS



## TIP

I have run more than a hundred JAD sessions and have learned several standard “facilitator tricks.” Here are some common problems and some ways to deal with them.

- **Reducing domination.** The facilitator should ensure that no one person dominates the group discussion. The only way to deal with someone who dominates is head on. During a break, approach the person, thank him or her for their insightful comments, and ask them to help you make sure that others also participate.
- **Encouraging noncontributors.** Drawing out people who have participated very little is challenging because you want to bring them into the conversation so that they will contribute again. The best approach is to ask a direct factual question that you are *certain* they can answer. And it helps to ask the question using some repetition to give them time to think. For example “Pat, I know you’ve worked shipping orders a long time. You’ve probably been in the Shipping Department longer than anyone else. Could you help us understand exactly what happens when an order is received in Shipping?”
- **Side discussions.** Sometimes participants engage in side conversations and fail to pay attention to the group. The easiest solution is simply to walk close to the people and continue to facilitate right in front of them. Few people will continue a side conversation when you are two feet from them and the entire group’s attention is on you and them.
- **Agenda merry-go-round.** The merry-go-round occurs when a group member keeps returning to the same issue every few minutes and won’t let go. One solution is to let the person have five minutes to ramble on about the issue while you carefully write down every point on a flip chart or computer file. This flip chart or file is then posted conspicuously on the wall. When the person brings up the issue again, you interrupt them, walk to the paper and ask them what to add. If they mention something already on the list, you quickly interrupt, point out that it is there, and ask what other information to add. Don’t let them repeat the same point, but write any new information.
- **Violent agreement.** Some of the worst disagreements occur when participants really agree on the issues but don’t realize that they agree because they are using different terms. An example is arguing whether a glass is half empty or half full; they agree on the facts, but can’t agree on the words. In this case, the facilitator has to translate the terms into different words and find common ground so the parties recognize that they really agree.
- **Unresolved conflict.** In some cases, participants don’t agree and can’t understand how to determine what alternatives are better. You can help by structuring the issue. Ask for criteria by which the group will identify a good alternative (e.g., “Suppose this idea really did improve customer service. How would I recognize the improved customer service?”). Then once you have a list of criteria, ask the group to assess the alternatives using them.
- **True conflict.** Sometimes, despite every attempt, participants just can’t agree on an issue. The solution is to postpone the discussion and move on. Document the issue as an “open issue” and list it prominently on a flip chart. Have the group return to the issue hours later. Often the issue will resolve itself by then and you haven’t wasted time on it. If the issue cannot be resolved later, move it to the list of issues to be decided by the project sponsor or some other more senior member of management.
- **Use humor.** Humor is one of the most powerful tools a facilitator has and thus must be used judiciously. The best JAD humor is always in context; never tell jokes but take the opportunity to find the humor in the situation.

*Alan Dennis*

**Selecting Participants** As with interviews and JAD sessions, the first step is to select the individuals to whom the questionnaire will be sent. However, it is not usual to select every person who could provide useful information. The standard approach is to select a *sample*, or subset, of people who are representative of the entire group. Sampling guidelines are discussed in most statistics books, and most business schools include courses that cover the topic, so we will not discuss it here. The important point in selecting a sample, however, is to realize that not everyone who receives a questionnaire will actually complete it. On average, only 30%–50% of paper and e-mail questionnaires are returned. Response rates for Web-based questionnaires tend to be significantly lower (often, only 5%–30%).

**FIGURE 3-9**  
Good Questionnaire Design

- Begin with nonthreatening and interesting questions.
- Group items into logically coherent sections.
- Do not put important items at the very end of the questionnaire.
- Do not crowd a page with too many items.
- Avoid abbreviations.
- Avoid biased or suggestive items or terms.
- Number questions to avoid confusion.
- Pretest the questionnaire to identify confusing questions.
- Provide anonymity to respondents.

**Designing the Questionnaire** Developing good questions is critical for questionnaires because the information on a questionnaire cannot be immediately clarified for a confused respondent. Questions on questionnaires must be very clearly written and must leave little room for misunderstanding; therefore, closed-ended questions tend to be most commonly used. Questions must enable the analyst to clearly separate facts from opinions. Opinion questions often ask the respondent the extent to which they agree or disagree (e.g., “Are network problems common?”), while factual questions seek more precise values (e.g., “How often does a network problem occur: once an hour, once a day, or once a week?”). See Figure 3-9 for guidelines on questionnaire design.

Perhaps the most obvious issue—but one that is sometimes overlooked—is to have a clear understanding of how the information collected from the questionnaire will be analyzed and used. You must address this issue before you distribute the questionnaire, because it is too late afterward.

Questions should be relatively consistent in style so that the respondent does not have to read instructions for each question before answering it. It is generally a good practice to group related questions together to make them simpler to answer. Some experts suggest that questionnaires should start with questions important to respondents, so that the questionnaire immediately grabs their interest and induces them to answer it. Perhaps the most important step is to have several colleagues review the questionnaire and then pretest it with a few people drawn from the groups to whom it will be sent. It is surprising how often seemingly simple questions can be misunderstood.

**Administering the Questionnaire** The key issue in administering the questionnaire is getting participants to complete the questionnaire and send it back. Dozens of marketing research books have been written about ways to improve response rates. Commonly used techniques include clearly explaining why the questionnaire is being conducted and why the respondent has been selected; stating a date by which the questionnaire is to be returned; offering an inducement to complete the questionnaire (e.g., a free pen); and offering to supply a summary of the questionnaire responses. Systems analysts have additional techniques to improve responses rates inside the organization, such as personally handing out the questionnaire and personally contacting those who have not returned them after a week or two, as well as requesting the respondents’ supervisors to administer the questionnaires in a group meeting.

**Questionnaire Follow-up** It is helpful to process the returned questionnaires and develop a questionnaire report soon after the questionnaire deadline. This ensures that the analysis process proceeds in a timely fashion and that respondents who requested copies of the results receive them promptly.

## YOUR

## 3-4 QUESTIONNAIRE PRACTICE

## TURN

Organize yourselves into small groups. Have each person develop a short questionnaire to collect information about the frequency in which group members perform some process (e.g., working on a class assignment, making a sandwich, paying bills, getting to class), how long it takes them, how they feel about the process, and opportunities for improving the process.

Once everyone has completed his or her questionnaire, ask each member to pass it to the right and then complete his or her neighbor's questionnaire.

Pass the questionnaire back to the creator when it is completed.

**QUESTIONS:**

1. How did the questionnaire you completed differ from the one you created?
2. What are the strengths of each questionnaire?
3. How would you analyze the survey results if you had received 50 responses?
4. What would you change about the questionnaire that you developed?

### Document Analysis

Project teams often use *document analysis* to understand the as-is system. Under ideal circumstances, the project team that developed the existing system will have produced documentation, which was then updated by all subsequent projects. In this case, the project team can start by reviewing the documentation and examining the system itself.

Unfortunately, most systems are not well documented, because project teams fail to document their projects along the way, and when the projects are over, there is no time to go back and document. Therefore, there may not be much technical documentation about the current system available, or it may not contain updated information about recent system changes. However, there are many helpful documents that do exist in the organization: paper reports, memorandums, policy manuals, user training manuals, organization charts, and forms. Problem reports filed by the system users can be another rich source of information about issues with the existing system.

But these documents (forms, reports, policy manuals, organization charts) only tell part of the story. They represent the *formal system* that the organization uses. Quite often, the “real,” or *informal system* differs from the formal one, and these differences, particularly large ones, give strong indications of what needs to be changed. For example, forms or reports that are never used likely should be eliminated. Likewise, boxes or questions on forms that are never filled in (or are used for other purposes) should be rethought. See Figure 3-10 for an example of how a document can be interpreted.

The most powerful indication that the system needs to be changed is when users create their own forms or add additional information to existing ones. Such changes clearly demonstrate the need for improvements to existing systems. Thus, it is useful to review both blank and completed forms to identify these deviations. Likewise, when users access multiple reports to satisfy their information needs, it is a clear sign that new information or new information formats are needed.

### Observation

*Observation*, the act of watching processes being performed, is a powerful tool to gain insight into the as-is system. Observation enables the analyst to see the reality of a situation, rather than listening to others describe it in interviews or JAD sessions.

**CENTRAL VETERINARY CLINIC**  
**Patient Information Card**

Name: ~~Buffy~~ Pat Smith

Pet's Name: Buffy Collie 7/6/07 Male

Address: 100 Central Court, Apartment 10  
Toronto, Ontario K7L 3N6

Phone Number: 416- 555-3400

Do you have insurance: yes

Insurance Company: Pet's Mutual

Policy Number: KA-5493243

The customer made a mistake. This should be labeled **Owner's Name** to prevent confusion.

The staff had to add additional information about the type of animal and the animal's date of birth and gender. This information should be added to the new form in the to-be system.

The customer did not include area code in the phone number. This should be made more clear.

**FIGURE 3-10**  
Performing a Document Analysis

Several research studies have shown that many managers really do not remember how they work and how they allocate their time. (Quick, how many hours did you spend last week on each of your courses?) Observation is a good way to check the validity of information gathered from other sources such as interviews and questionnaires.

In many ways, the analyst becomes an anthropologist as he or she walks through the organization and observes the business system as it functions. The goal is to keep a low profile, to not interrupt those working, and to not influence those being observed. Nonetheless, it is important to understand that what analysts observe may not be the normal day-to-day routine because people tend to be extremely careful in their behavior when they are being watched.<sup>10</sup> Even though

<sup>10</sup> This illustrates the Hawthorne effect: an increase in worker productivity produced by the psychological stimulus of being singled out and made to feel important. See R. H. Frank and J. D. Kaul, "The Hawthorne Experiments: First Statistical Interpretation," *American Sociological Review*, 1978, 43: 623–643.

## CONCEPTS

## 3-D PUBLIX CREDIT CARD FORMS

## IN ACTION

At my neighborhood Publix grocery store, the cashiers always handwrite the total amount of the charge on every credit card charge form, even though it is printed on the form. Why? Because the “back office” staff people who reconcile the cash in the cash drawers with the amount sold at the end of each shift find it hard to read the small print on the credit card forms. Writing in large print makes it easier for them to add the

values up. However, cashiers sometimes make mistakes and write the wrong amount on the forms, which causes problems. *Barbara Wixom*

## QUESTIONS:

1. What does the credit card charge form indicate about the existing system?
2. How can you make improvements with a new system?

normal practice may be to break formal organizational rules, the observer is unlikely to see this. (Remember how you drove the last time a police car followed you?) Thus, what you see may *not* be what you really want.

Observation is often used to supplement interview information. The location of a person’s office and its furnishings gives clues as to their power and influence in the organization, and such clues can be used to support or refute information given in an interview. For example, an analyst might become skeptical of someone who claims to use the existing computer system extensively if the computer is never turned on while the analyst visits. In most cases, observation will support the information that users provide in interviews. When it does not, it is an important signal that extra care must be taken in analyzing the business system.

### Selecting the Appropriate Techniques

Each of the requirements elicitation techniques just discussed has strengths and weaknesses. No one technique is always better than the others, and in practice most projects benefit from a combination of techniques. Thus, it is important to understand the strengths and weaknesses of each technique and when to use each. (See Figure 3-11.) One issue not discussed is that of the analysts’ experience. In general, document analysis and observation require the least amount of training, while JAD sessions are the most challenging.

**Type of Information** The first characteristic is type of information. Some techniques are more suited for use at different stages of the analysis process, whether understanding the as-is system, identifying improvements, or developing the to-be

## YOUR

## 3-5 OBSERVATION PRACTICE

## TURN

Visit the library at your college or university and observe how the book check-out process occurs. First, watch several students checking books out, and then check one out yourself. Prepare a brief summary report of your observations.

When you return to class, share your observations with others. You may notice that not all the reports present the same information. Why? How would the information be different had you used the interview or JAD technique?

	Interviews	Joint Application Design	Questionnaires	Document Analysis	Observation
Type of information	As-is, improvements, to-be	As-is, improvements, to-be	As-is, improvements	As-is	As-is
Depth of information	High	High	Medium	Low	Low
Breadth of information	Low	Medium	High	High	Low
Integration of information	Low	High	Low	Low	Low
User involvement	Medium	High	Low	Low	Low
Cost	Medium	Low–Medium	Low	Low	Low–Medium

**FIGURE 3-11**  
Comparison of Requirements Elicitation Techniques

system. Interviews and JAD are commonly used in all three stages. In contrast, document analysis and observation usually are most helpful for understanding the as-is system, although they occasionally provide information about improvements. Questionnaires are often used to gather information about the as-is system, as well as general information about improvements.

**Depth of Information** The depth of information refers to how rich and detailed the information is that the technique usually produces and the extent to which the technique is useful at obtaining not only facts and opinions, but also an understanding of *why* those facts and opinions exist. Interviews and JAD sessions are very useful at providing a good depth of rich and detailed information and helping the analyst to understand the reasons behind them. At the other extreme, document analysis and observation are useful for obtaining facts, but little beyond that. Questionnaires can provide a medium depth of information, soliciting both facts and opinions with little understanding of why.

**Breadth of Information** Breadth of information refers to the range of information and information sources that can be easily collected by that technique. Questionnaires and document analysis both are easily capable of soliciting a wide range of information from a large number of information sources. In contrast, interviews and observation require the analyst to visit each information source individually and, therefore, take more time. JAD sessions are in the middle because many information sources are brought together at the same time.

**Integration of Information** One of the most challenging aspects of requirements gathering is the integration of information from different sources. Simply put, different people can provide conflicting information. Combining this information and attempting to resolve differences in opinions or facts is usually very time consuming because it means contacting each information source in turn, explaining the discrepancy, and attempting to refine the information. In many cases, the individual wrongly perceives that the analyst is challenging his or her information, when in fact the source of conflict is another user in the organization. This can make the user defensive and make it hard to resolve the differences.

All techniques suffer integration problems to some degree, but JAD sessions are designed to improve integration because all information is integrated when it is collected, not afterward. If two users provide conflicting information, the conflict



becomes immediately obvious, as does the source of the conflict. The immediate integration of information is the single most important benefit of JAD that distinguishes it from other techniques, and this is why most organizations use JAD for important projects.

**User Involvement** User involvement refers to the amount of time and energy the intended users of the new system must devote to the analysis process. It is generally agreed that, as users become more involved in the analysis process, the chance of success increases. However, user involvement can have a significant cost, and not all users are willing to contribute valuable time and energy. Questionnaires, document analysis, and observation place the least burden on users, while JAD sessions require the greatest effort.

**Cost** Cost is always an important consideration. In general, questionnaires, document analysis, and observation are low-cost techniques (although observation can be quite time consuming). The low cost does not imply that they are more or less effective than the other techniques. We regard interviews and JAD sessions as having moderate costs. In general, JAD sessions are much more expensive initially, because they require many users to be absent from their offices for significant periods, and they often involve highly paid consultants. However, JAD sessions significantly reduce the time spent in information integration and thus cost less in the long term.

## REQUIREMENTS ANALYSIS STRATEGIES

---

The previous section discussed five essential techniques that analysts will use to interact with stakeholders in the system development project to elicit and define requirements. As we discussed earlier in the chapter, the analyst often must encourage the stakeholders to think critically about the needs for the new system and discover the true underlying requirements. In this section, we present several strategies that the analyst can employ with the stakeholders to accomplish this goal.

### Problem Analysis

The most straightforward (and probably the most commonly used) requirements analysis strategy is *problem analysis*. Problem analysis means asking the users and managers to identify problems with the as-is system and to describe how to solve them in the to-be system. Most users have a very good idea of the changes they would like to see, and most will be quite vocal about suggesting them. Most changes tend to solve problems rather than capitalize on opportunities, but this is possible, too. Improvements from problem analysis tend to be small and incremental (e.g., add a field to store the customer's cell phone number; provide a new report that currently does not exist).

This type of improvement often is very effective at improving a system's efficiency or ease of use. However, it often provides only minor improvements in business value—the new system is better than the old, but it may be hard to identify significant monetary benefits from the new system.

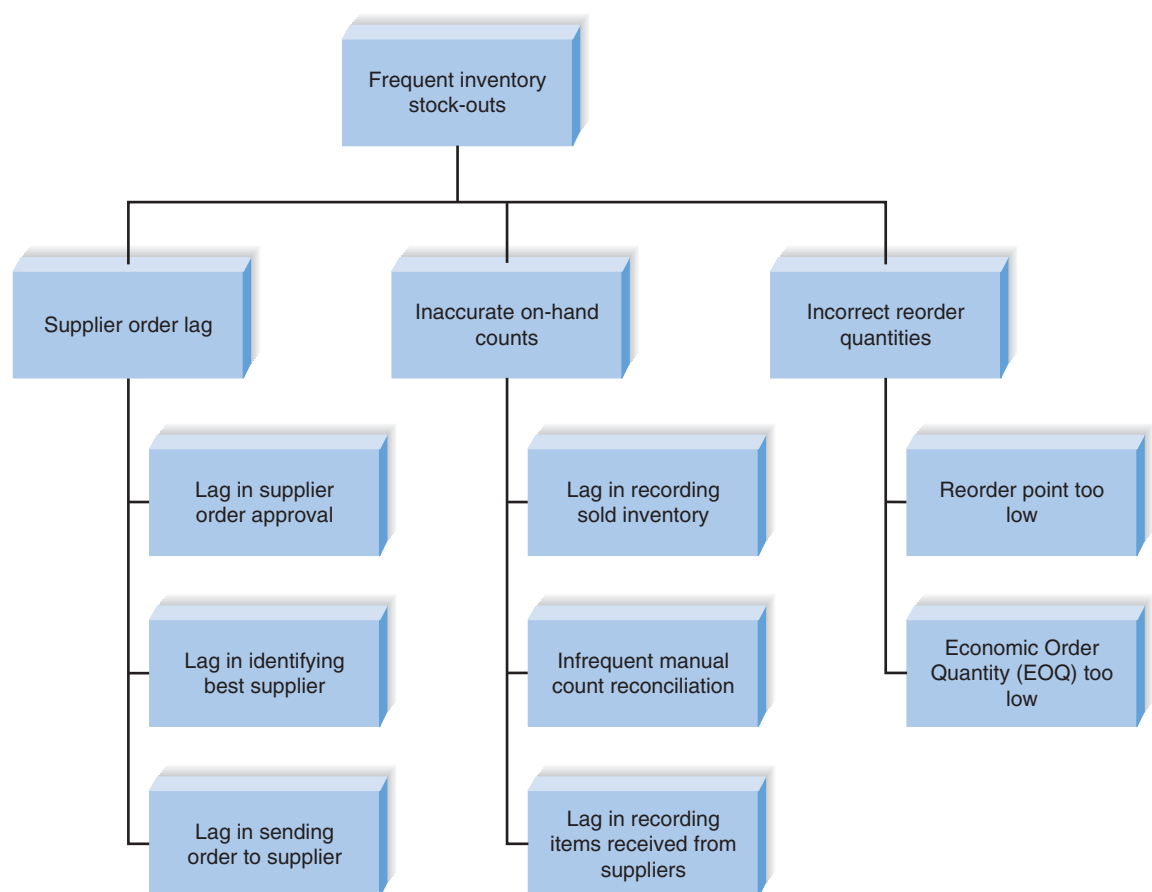
### Root Cause Analysis

The ideas produced by problem analysis tend to be *solutions* to problems. All solutions make assumptions about the nature of the problem, assumptions that may or may not be valid. In our experience, users (and most people in general) tend to jump quickly to solutions without fully considering the nature of the problem. Sometimes

the solutions are appropriate, but many times they address a *symptom* of the problem, not the true problem or *root cause* itself.<sup>11</sup>

For example, suppose that the users report that “inventory stock-outs happen frequently.” Inventory stock-outs are not good, of course, and one obvious way to reduce their occurrence is to increase the quantity of items kept in stock. This action incurs costs, however, so it is worthwhile to investigate the underlying cause of the frequent stock-outs instead of jumping to a quick-fix solution. The solutions that users propose (or systems that analysts consider) may address either symptoms or causes, but without careful analysis, it is difficult to tell which one. Finding out later that you’ve just spent millions of dollars and have not fixed the *true* underlying problem is a horrible feeling!

Root cause analysis focuses on problems first rather than solutions. The analyst starts by having the users generate a list of problems with the current system, then prioritizes the problems in order of importance. Starting with the most important, the users and/or analysts generate all possible root causes for the problem. As shown in Figure 3-12, the problem of “too frequent stock-outs” has several



**FIGURE 3-12**  
Root Cause Analysis for Inventory Stock Outs

<sup>11</sup> Two good books that discuss the problems in finding the root causes to problems are E. M. Goldratt and J. Cox, *The Goal*, Croton-on-Hudson, NY: North River Press, 1986; and E. M. Goldratt, *The Haystack Syndrome*, Croton-on-Hudson, NY: North River Press, 1990.

potential root causes (inaccurate on-hand counts; incorrect reorder points; lag in placing supplier orders). Each possible root cause is investigated and additional root causes are identified. As Figure 3-12 shows, it is sometimes useful to display the potential root causes in a tree-like hierarchy. Ultimately, the investigation process reveals the true root cause or causes of the problem, enabling the team to design the system to correct the problem with the right solution. The key point in root cause analysis is to always challenge the obvious and dig into the problem deeply enough that the true underlying cause(s) is revealed.

### Duration Analysis

*Duration analysis* requires a detailed examination of the amount of time it takes to perform each process in the current as-is system. The analysts begin by determining the total amount of time it takes, on average, to perform a set of business processes for a typical input. They then time each of the individual steps (or sub-processes) in the business process. The time to complete the basic steps are then totaled and compared with the total for the overall process. A significant difference between the two—and, in our experiences, the total time often can be 10 or even

## CONCEPTS

### 3-E SUCCESS FROM FAILURE

#### IN ACTION

Few niches crashed more spectacularly during Web 1.0 than the pet sector. In 2000, over just nine months, Pets.com managed to raise a jaw-dropping \$82.5 million in an IPO, air a \$1.2 million Super Bowl ad starring its sock puppet mascot, land funding from Amazon.com build a network of cavernous warehouses ... and go out of business without making a penny in profit. When Pets.com rolled over and died in November 2000, it presaged scores of dot-com disasters to follow and slammed the door on online pet businesses, seemingly for good.

So when San Francisco Web designer Ted Rheingold co-founded Dogster.com in January 2004 as a kind of canine version of Friendster, the news drew smirks from the few who bothered to notice. How could Dogster, a pet site cobbled together on weekends and launched on a shoestring budget, expect to succeed where lavishly funded pet sites had flamed out? The consensus on Dogster was unanimous: It would fail.

And indeed, it has failed. Over and over. But, alas, each knock has been a boost. Dogster has discovered ways to turn its mistakes into better features. With pretty much no promotion, Dogster (and sister site Catster.com) has evolved into a premier pet lover's social network. Membership exceeds 275,000; the site features 340,000 photos and profiles of dogs and cats, and a blue-chip advertising list that includes Disney, Holiday Inn, and Target. Dogster, come to find out, has a good profit sheet.

In many ways, the site is a prime example of how a Web deployment fails, but fails well by quick feature launch, seeing what works, and fixing things fast. According to Rheingold, "When we roll out a new feature, we know we're probably not going to get it right the first time." Dogster and similar companies have discovered that continually reviewing user data—most importantly, the discouraging events—provides important direction for enhancements. Says Rheingold, "Instead of working on a feature for months trying to get it perfect, we'll work on something for two weeks and then spend two or three days listening to users and fine-tuning it."

Source: "A Startup's Best Friend? Failure," Tom McNichol, *Business 2.0*. San Francisco: March 2007, vol. 8, iss. 2, p. 39–41.

#### QUESTIONS:

1. Do you agree with Dogster's view, or should companies aim for "zero-defect" operations? Why or why not? What implications does this business model have for systems analysts?
2. Startup companies like Dogster are not the only companies that are implementing the "fail fast" strategy. Large companies like Google have used it and are still using it—in Google's case, with the implementation of the Google Toolbar. Cite another company that has used this strategy. Has it been successful?

100 times longer than the sum of the parts—indicates that this part of the process is badly in need of a major overhaul.

For example, suppose that the analysts are working on a home mortgage system and discover that, on average, it takes 30 days for the bank to approve a mortgage. They then look at each of the basic steps in the process (e.g., data entry, credit check, title search, appraisal, etc.) and find that the total amount of time actually spent on each mortgage is about 8 hours. This is a strong indication that the overall process is badly broken, because it takes 30 days to perform 1 day's work.

These problems likely occur because the process is badly fragmented. Many different people must perform different activities before the process is complete. In the mortgage example, the application probably sits on many peoples' desks for long periods of time before it is processed. Processes in which many different people work on small parts of the inputs are prime candidates for *process integration* or *parallelization*. Process integration means changing the fundamental process so that fewer people work on the input, which often requires changing the processes and retraining staff to perform a wider range of duties. Process parallelization means changing the process so that all the individual steps are performed at the same time. For example in the mortgage application example, there is probably no reason that the credit check cannot be performed at the same time as the appraisal and title check.

### Activity-Based Costing

*Activity-based costing* is a similar analysis that examines the cost of each major process or step in a business process rather than the time taken.<sup>12</sup> The analysts identify the costs associated with each of the basic functional steps or processes, identify the most costly processes, and focus their improvement efforts on them.

Assigning costs is conceptually simple. You just examine the direct cost of labor and materials for each input. Materials costs are easily assigned in a manufacturing process, while labor costs are usually calculated on the basis of the amount of time spent on the input and the hourly cost of the staff. However, as you may recall from a managerial accounting course, there are indirect costs such as rent, depreciation, and so on that also can be included in activity costs.

### Informal Benchmarking

*Benchmarking* refers to studying how other organizations perform a business process in order to learn how your organization can do something better. Benchmarking helps the organization by introducing ideas that employees may never have considered, but that have the potential to add value.

*Informal benchmarking* is fairly common for “customer-facing” business processes (i.e., those processes that interact with the customer). With informal benchmarking, the managers and analysts think about other organizations, or visit them as customers to watch how the business process is performed. In many cases, the business studied may be a known leader in the industry or simply a related firm. For example, suppose that the team is developing a Web site for a car dealer. The

<sup>12</sup> Many books have been written on activity-based costing. Useful ones include K. B. Burk and D. W. Webster, *Activity-Based Costing*, Fairfax, VA: American Management Systems, 1994; and D. T. Hicks, *Activity-Based Costing: Making It Work for Small and Mid-Sized Companies*, New York: John Wiley, 1998. The two books by Eli Goldratt mentioned previously (*The Goal* and *The Haystack Syndrome*) also offer unique insights into costing.

## CONCEPTS

## 3-F A PROCESS IN NEED OF IMPROVEMENT

## IN ACTION

A group of executives from a Fortune 500 company used duration analysis to discuss their procurement process. Using a huge wall of Velcro and a handful of placards, a facilitator proceeded to map out the company's process for procuring a \$50 software upgrade. Having quantified the time it took to complete each step, she then assigned costs based on the salaries of the employees involved. The 15-minute exercise left the

group stunned. Their procurement process had gotten so convoluted that it took 18 days, countless hours of paperwork and nearly \$22,000 in people time to get the product ordered, received, and up and running on the requester's desktop.

Source: "For Good Measure," *CIO Magazine*, March 1, 1999, by Debby Young.

project sponsor, key managers, and key team members would likely visit the Web sites of competitors, those of others in the car industry (e.g., manufacturers, accessories suppliers), and those of other industries that have won awards for their Web sites.

### Outcome Analysis

*Outcome analysis* focuses on understanding the fundamental outcomes that provide value to customers. While these outcomes sound as though they should be obvious, they often aren't. For example, suppose that you are an insurance company and one of your customers has just had a car accident. What is the fundamental outcome from the *customer's* perspective? Traditionally, insurance companies have answered this question by assuming that the customer wants to receive the insurance payment quickly. To the customer, however, the payment is only a *means* to the real outcome: a repaired car. The insurance company might benefit by extending its view of the business process past its traditional boundaries to include, not simply paying for repairs, but performing the repairs or contracting with an authorized body shop to do them.

With this approach, the system analysts encourage the managers and project sponsor to pretend that they are customers and to think carefully about what the organization's products and services enable the customers to do—and what they *could* enable the customer to do.

### Technology Analysis

Many major changes in business over the past decade have been enabled by new technologies. *Technology analysis* therefore starts by having the analysts and managers develop a list of important and interesting technologies. Then the group systematically identifies how each and every technology could be applied to the business process and identifies how the business would benefit.

For example, one useful technology might be the Internet. A manufacturer could develop an extranet application for its suppliers. Rather than ordering parts for its products, the manufacturer makes its production schedule available electronically to its suppliers, who ship the needed parts so that they arrive at the plant just in time. This saves significant costs because it eliminates the need for people to monitor the production schedule and issue purchase orders.

**CONCEPTS****3-G “LIKE” YOUR LOCAL GOVERNMENT****IN ACTION**

Municipal, city, and county governments are seriously affected by the recent economic downturn and resulting budgetary pressure. Fewer employees mean it is harder to provide services in the same old way to citizens. Forward-thinking managers of governmental entities recognize that social media may be the wave of the future. Social media can allow citi-

zens to be a part of the government services, not just a recipient of government services. Citizens can report issues such as potholes in roads or abandoned vehicles when they see them and government services can repair them while out in the field. Citizens are part of the process and get immediate satisfaction by helping to solve problems.

**YOUR****3-6 IBM CREDIT****T U R N**

IBM Credit was a wholly owned subsidiary of IBM responsible for financing mainframe computers sold by IBM. While some customers bought mainframes outright or obtained financing from other sources, financing computers provided significant additional profit.

When an IBM sales representative made a sale, he or she would immediately call IBM Credit to obtain a financing quote. The call was received by a credit officer who would record the information on a request form. The form would then be sent to the credit department to check the customer's credit status. This information would be recorded on the form, which was then sent to the business practices department, which would write a contract (sometimes reflecting changes requested by the customer). The form and the contract would then go to the pricing department, which used the credit information to establish an interest rate and record it on the form. The form and contract was then sent to the clerical group, where an administrator would prepare a cover letter quoting the interest rate and send the letter and contract via Federal Express to the customer.

The problem at IBM Credit was a major one. Getting a financing quote took anywhere from four to eight days (six days, on average), giving the customer time to rethink the order or find financing elsewhere. While the quote was being prepared, sales representatives would often call to find out where the quote was in the process, so that they could tell the customer when to expect it. However, no one at IBM Credit could answer the question, because the paper forms could be in any department and

it was impossible to locate one without physically walking through the departments and going through the piles of forms on everyone's desk.

IBM Credit examined the process and changed it so that each credit request was logged into a computer system so that each department could record an application's status as soon as it was completed and sent it to the next department. In this way, sales representatives could call the credit office and quickly learn the status of each application. IBM used some sophisticated management science queuing theory analysis to balance workloads and staff across the different departments so that no applications would be overloaded. They also introduced performance standards for each department (e.g., the pricing decision had to be completed within one day after that department received an application).

However, process times got worse, even though each department was achieving almost 100 percent compliance on its performance goals. After some investigation, managers found that when people got busy, they conveniently found errors that forced them to return the credit request to the previous department for correction, thereby removing it from their time measurements.

**QUESTIONS:**

What techniques can you use to identify improvements? Choose one technique and apply it to this situation—what improvements did you identify?

Source: *Reengineering the Corporation*, New York: Harper Business, 1993, by M. Hammer and J. Champy.

### Activity Elimination

*Activity elimination* is exactly what it sounds like. The analysts and managers work together to identify how the organization could eliminate each and every activity in the business process, how the function could operate without it, and what effects are likely to occur. Initially, managers are reluctant to conclude that processes can be eliminated, but this is a “force-fit” exercise in that they must eliminate each activity. In some cases the results are silly; nonetheless, participants must address each and every activity in the business process.

For example, in the home mortgage approval process discussed earlier, the managers and analysts would start by eliminating the first activity, entering the data into the mortgage company’s computer. This leads to one of two obvious possibilities: (1) Eliminate the use of a computer system or (2) make someone else do the data entry (e.g., the customer, over the Web). They would then eliminate the next activity, the credit check. Silly, right? After all, making sure the applicant has good credit is critical in issuing a loan, isn’t it? Not really. The real answer depends upon how many times the credit check identifies bad applications. If all or almost all applicants have good credit and are seldom turned down by a credit check, then the cost of the credit check may not be worth the benefit of the few bad loans it prevents. Eliminating it may actually result in lower costs, even with the cost of bad loans, unless the number of applicants with poor credit greatly increases.

### Comparing Analysis Strategies

Each of the requirements analysis strategies discussed here has its own purpose. No one technique is inherently better than the others. Remember that an organization will likely have a wide range of projects in its portfolio; the requirements analysis strategy should be chosen to fit the nature of the project. Problem analysis and root cause analysis tend to be most useful in situations with a narrow focus where efficiency gains are sought. Duration analysis and activity-based costing strategies help the team find the most “broken” business processes so that those processes can be redesigned and improved. Outcome analysis, technology analysis, and informal benchmarking help the team think “outside the box” and are very useful when the team is trying to create completely new ways of accomplishing the business processes.

## APPLYING THE CONCEPTS AT TUNE SOURCE

---

Once the Tune Source approval committee approved the system request and feasibility analysis, the project team began performing analysis activities. These included gathering requirements by a variety of techniques and analyzing the requirements that were gathered. Some highlights of the project team’s activities are presented next.

### Eliciting and Analyzing Requirements

Jason believed that it would be important to understand the current Web-based sales processes and systems that already existed in the organization, because they would have to be closely integrated with the Digital Music Download system. Two requirements-gathering techniques proved to be helpful in understanding the current systems and processes—document analysis and interviews.

First, the project team collected existing reports (e.g., sales forms, screen shots of the online sales screens) and system documentation (data models, process



models) that shed light on the as-is system. They were able to gather a good amount of information about the existing order processes and systems in this way. When questions arose, they conducted short interviews with the person who provided the documentation, for clarification.

Next, Jason interviewed the senior analysts for the current sales systems to get a better understanding of how those systems worked. He asked whether they had any ideas for the new system, as well as whether there were any integration issues that would need to be addressed. Jason also interviewed a contact from the ISP and the IT person who supported Tune Source's current Web site—both provided information about the existing communications infrastructure at Tune Source and its Web capabilities.

Carly suggested that the project team conduct several JAD sessions with store managers, marketing analysts, and Web-savvy members of the IT staff. Together, the groups could brainstorm the features desired in the Digital Music Download system.

Jason facilitated three JAD sessions that were conducted over the course of a week. Jason's past facilitation experience helped the eight-person meetings run smoothly and stay on track. Because this project introduces a new business process, Jason used technology analysis and suggested several important Web technologies that could be used for the system. The JAD session generated ideas about how Tune Source could apply each of the technologies to the Digital Music Download system. Jason had the group categorize the ideas into three sets: "definite" ideas that would have a good probability of providing business value, "possible" ideas that might add business value, and "unlikely" ideas.

Next, Jason applied informal benchmarking by introducing the Web sites of several leading retailers and pointing out the features that they offered online. He selected some sites on the basis of their success with Internet sales, and others on the basis of their similarity to the vision for Tune Source's new system. The group discussed the features that were common across most retailers, versus unique functionality, and they created a list of suggested business requirements for the project team.

## Requirements Definition

Throughout all of these activities, the project team collected information and tried to identify the business requirements for the system from the information. As the project progressed, requirements were added to the requirements definition and grouped by requirements type. When questions arose, they worked with Carly and Jason to confirm that requirements were in scope. The requirements that fell outside of the scope of the current system were typed into a separate document that would be saved for future use.

At the end of the analysis phase, the requirements definition was distributed to Carly, two marketing employees who would work with the system on the business side, and several retail store managers. This group then met for a two-day JAD session to clarify, finalize, and prioritize the requirements and to create use cases (Chapter 4) to show how the system would be used.

The project team also spent time creating process models (Chapter 5) and data models (Chapter 6) that depicted the processes and data in the future system. Members of marketing and IT reviewed the documents during interviews with the project team. Figure 3-13 shows a portion of the final requirements definition.

## System Proposal

Jason reviewed the requirements definition and the other deliverables that the project team created during the analysis phase. Given Carly's desire to have the system in

**Functional Requirements:**

1. Search and Browse
  - 1.1 The system will allow customers to browse music choices by predefined categories.
  - 1.2 The system will allow customers to search for music choices by title, artist, and genre.
  - 1.3 The system will allow customers to listen to a short sample of a music selection.
  - 1.4 The system will enable the customer to add music selections to a "favorites" list.
2. Purchase
  - 2.1 The system will enable the customer to create a customer account (if desired) that will store customer data and payment information.
  - 2.2 The system will enable the customer to specify the music choice for download.
  - 2.3 The system will collect and verify payment information. Once payment is verified, the music selection download process will begin.
3. Promote
  - 3.1 The system will keep track of the customer's interests on the basis of samples selected for listening and will use this information to promote music selections during future visits to the Web site.
  - 3.2 Marketing department can create promotions and specials on the Web site.
  - 3.3 Based on customer's previous purchases, music choices can be targeted to the customer on future visits to the Web site. (Customers who like X will also like Y.)
  - 3.4 On the basis of customer interests, customers can be notified of special offers on CDs that can be purchased at the regular Tune Source Web site or in a Tune Source store.

**Nonfunctional Requirements:**

1. Operational
  - 1.1 The digital music database will be constructed to facilitate searches by title, artist, and genre.
  - 1.2 The system will run on any Web browser and on in-store kiosks.
  - 1.3 In the event of a failure during a download, the customer will be able to restart the download.
2. Performance
  - 2.1 Download speeds will be monitored and kept at an acceptable level.
3. Security
  - 3.1 Customer information will be secured.
  - 3.2 Payment information will be encrypted and secured.
4. Cultural and political
 

No special cultural and political requirements are expected.

**FIGURE 3-13**

Tune Source Requirements Definition

production as soon as possible, Jason decided to timebox the project. He had originally decided to approach the project in three versions (iterative development, see Chapter 2), and he is satisfied that this is a good way to structure the project. The first version, to be operational in the late spring, would implement a basic digital music download capability that will enable customers to download music on a fixed price per download basis. The second version, planned to be ready by midsummer, would incorporate a customer subscription program. The marketing department has yet to determine its preferred subscription program. It is considering a low fee, longer-term program or a higher fee, shorter-term program. By the time the project team is ready to begin version 2, however, the details should be nailed down. The third version, expected to be ready by late summer, will add the gift card option, entitling the gift card holder to a fixed number of downloads over a limited time.

<b>1. Table of Contents</b>	
<b>2. Executive Summary</b>	A summary of all the essential information in the proposal so that a busy executive can read it quickly and decide what parts of the plan to read in more depth.
<b>3. System Request</b>	The revised system request form. (See Chapter 1.)
<b>4. Work plan</b>	The original work plan, revised after having completed the analysis phase. (See Chapter 2.)
<b>5. Feasibility Analysis</b>	A revised feasibility analysis, using the information from the analysis phase. (See Chapter 1.)
<b>6. Requirements Definition</b>	A list of the functional and nonfunctional business requirements for the system (this chapter).
<b>7. Use Cases</b>	A set of use cases that illustrate the basic processes that the system needs to support. (See Chapter 4.)
<b>8. Process Model</b>	A set of process models and descriptions for the to-be system. (See Chapter 5.) This may include process models of the current as-is system that will be replaced.
<b>9. Data Model</b>	A set of data models and descriptions for the to-be system. (See Chapter 6.) This may include data models of the as-is system that will be replaced.
<b>Appendices</b>	These contain additional material relevant to the proposal, often used to support the recommended system. This might include results of a questionnaire survey or interviews, industry reports and statistics, etc.

**FIGURE 3-14**  
Outline of the Tune Source  
System Proposal

Jason reviewed the work plan and made some slight changes. He also conferred with Carly and the marketing department members to review the feasibility analysis. No major changes were made to it at this point; the project remains highly feasible overall. All of the deliverables from the project were then combined into a system proposal and submitted to the approval committee. Figure 3-14 shows the outline of the Tune Source system proposal. Carly and Jason met with the approval committee and presented the highlights of what was learned during the analysis phase and the final concept of the new system. On the basis of the proposal and presentation, the approval committee decided that it would continue to fund the Digital Music Download system.

## SUMMARY

### Analysis

Analysis focuses on capturing the business requirements for the system. Analysis identifies the “what” of the system, and it leads directly into the design phase, during which the “how” of the system is determined. Many deliverables are created during the analysis phase, including the requirements definition, use cases, process models, and a data model. At the end of analysis, all of these deliverables, along with revised planning and project management deliverables, are combined into a system proposal and submitted to the approval committee for a decision regarding whether or not to move ahead with the project.

### Requirements Determination

Requirements determination is the part of analysis in which the project team turns the very high level explanation of the business requirements stated in the system request into a more precise list of requirements. A requirement is simply a statement of what the system must do or what characteristic it needs to have. Business requirements describe the “what” of the system, and system requirements describe “how” the system will be implemented. A functional requirement relates directly to a process the system has to perform or information it needs to contain. Nonfunctional requirements refer to behavioral properties that the system must have, such as performance and usability. All of the functional and nonfunctional business requirements that fit within the scope of the system are written in the requirements definition.

### Requirements Elicitation Techniques

Five techniques can be used to elicit the business requirements for the proposed system: interviews, joint application development, questionnaires, document analysis, and observation. Interviews involve meeting one or more people and asking them questions. There are five basic steps to the interview process: selecting interviewees, designing interview questions, preparing for the interview, conducting the interview, and post-interview follow-up. Joint application development (JAD) allows the project team, users, and management to work together to identify requirements for the system. Electronic JAD attempts to overcome common problems associated with groups by using groupware. A questionnaire is a set of written questions developed for obtaining information from individuals. Questionnaires often are used when there is a large number of people from whom information and opinions are needed. Document analysis entails reviewing the existing documentation and examining the system itself. It can provide insights into the formal and informal system. Observation, the act of watching processes being performed, is a powerful tool for gathering information about the as-is system because it enables the analyst to see the reality of a situation firsthand.

### Requirements Analysis Strategies

Analysts often have to help the business users think critically about their new system requirements. Several strategies are helpful. Problem analysis and root cause analysis are two strategies that can assist the business users in understanding the problems and issues of the current system that require fixing. Duration analysis, activity-based costing, and informal benchmarking are three popular analysis strategies that help the team discover processes most in need of redesign. Finally, outcome analysis, technology analysis, and activity elimination are three strategies that can be used to “force” the business users to think about the business processes in new, novel ways, perhaps discovering completely new ways to accomplish the business processes.

## KEY TERMS

Activity elimination	Breadth of analysis	Electronic JAD (e-JAD)
Activity-based costing	Business requirement	Facilitator
Analysis	Closed-ended question	Formal system
As-is system	Critical thinking skills	Functional requirement
Benchmarking	Document analysis	Ground rule
Bottom-up interview	Duration analysis	Informal benchmarking

Informal system	Postsession report	Sample
Interpersonal skill	Potential business value	Scribe
Interview	Probing question	Stakeholder
Interview notes	Problem analysis	Structured interview
Interview report	Process integration	Symptom
Interview schedule	Project cost	System proposal
Joint application development (JAD)	Questionnaire	System requirement
Nonfunctional requirements	Requirement	Technology analysis
Observation	Requirements definition	To-be system
Open-ended question	Requirements determination	Top-down interview
Outcome analysis	Risk	Unstructured interview
Parallelization	Root cause	User requirements
	Root cause analysis	Walk-through

## QUESTIONS

1. What is the meaning of analysis? What is the purpose of the analysis phase of the SDLC?
2. What are the key elements of the system proposal?
3. A system development project may be approached in one of two ways: as a single, monolithic project in which all requirements are considered at once or as a series of smaller projects focusing on smaller sets of requirements. Which approach seems to be more successful? Why do you suppose that this is true?
4. Distinguish between business, user, and functional requirements.
5. Explain what is meant by a functional requirement. What are two types of functional requirements? Give two examples of each.
6. Explain what is meant by a nonfunctional requirement. What are the primary types of nonfunctional requirements? Give two examples of each. What role do nonfunctional requirements play in the project overall?
7. What is the value of producing a requirements definition and having the project sponsor and key users review and approve it?
8. What are the three basic steps of the analysis process? Is each step performed in every project? Why or why not?
9. Discuss the appropriate way to set up and conduct interviews to elicit requirements.
10. Give an example of a closed-ended question, an open-ended question, and a probing question. When would each type of question be used?
11. "Interviews should always be conducted as structured interviews." Do you agree with this statement? Why or why not?
12. Discuss the considerations that should be made when determining who to include in interviews and/or JAD sessions.
13. Is the primary purpose of requirements determination to discover facts or to discover opinions? Explain your answer.
14. Describe the five major steps in conducting JAD sessions.
15. Describe the primary roles involved in JAD sessions. What is the major contribution made by the person(s) fulfilling each role?
16. Discuss the reasons that question design for questionnaires is so difficult.
17. Why is document analysis useful? What insights into the organization can it provide?
18. Outline suggestions to make observation a useful, reliable requirements elicitation technique.
19. Describe a strategy for using the various requirements elicitation techniques in a project.
20. Discuss problem analysis as an analysis strategy. What are the strengths and limitations of this technique?
21. Discuss root cause analysis as an analysis strategy. What are the strengths and limitations of this technique?
22. Compare and contrast duration analysis and activity-based costing. What role do these activities play as analysis strategies?
23. How can informal benchmarking contribute to requirements determination?
24. Compare and contrast outcome analysis, technology analysis, and activity elimination. What general contribution do these strategies play in determining requirements?

## EXERCISES

- A. Review the Amazon.com Web site. Develop the requirements definition for the site. Create a list of functional business requirements that the system meets. What different kinds of nonfunctional business requirements does the system meet? Provide examples for each kind.
- B. Pretend that you are going to build a new system that automates or improves the interview process for the career services department of your school. Develop a requirements definition for the new system. Include both functional and nonfunctional system requirements. Pretend that you will release the system in three different versions. Prioritize the requirements accordingly.
- C. Describe in very general terms the as-is business process for registering for classes at your university. Collaborate with another student in your class and evaluate the process using problem analysis and root cause analysis. Based on your work, list some example improvements that you identified.
- D. Describe in very general terms the as-is business process for applying for admission at your university. Collaborate with another student in your class and evaluate the process using informal benchmarking. Based on your work, list some example improvements that you identified.
- E. Describe in very general terms the as-is business process for registering for classes at your university. Collaborate with another student in your class and evaluate the process using activity elimination. Based on your work, list some example improvements that you identified.
- F. Suppose that your university is having a dramatic increase in enrollment and is having difficulty finding enough seats in courses for students so that they can take courses required for graduation. Perform a technology analysis to identify new ways to help students complete their studies and graduate.
- G. Suppose that you are the analyst charged with developing a new system for the university bookstore with which students can order books online and have them delivered to their dorms and off-campus housing. What requirements-gathering techniques will you use? Describe in detail how you would apply the techniques.
- H. Suppose that you are the analyst charged with developing a new system to help senior managers make better strategic decisions. What requirements-gathering techniques will you use? Describe in detail how you would apply the techniques.
- I. Find a partner and interview each other about what tasks you/they did in the last job held (full-time, part-time, past, or current). If you haven't worked before, then assume that your job is being a student. Before you do this, develop a brief interview plan. After your partner interviews you, identify the type of interview, interview approach, and types of questions used.
- J. Find a group of students and run a 60-minute JAD session on improving alumni relations at your university. Develop a brief JAD plan, select two techniques that will help identify improvements, and then develop an agenda. Conduct the session, using the agenda, and write your post-session report.
- K. Find a questionnaire on the Web that has been created to capture customer information. Describe the purpose of the survey, the way questions are worded, and how the questions have been organized. How can the questionnaire be improved? How will the responses be analyzed?
- L. Develop a questionnaire that will help gather information regarding processes at a popular restaurant or the college cafeteria (e.g., ordering, customer service). Give the questionnaire to 10–15 students, analyze the responses, and write a brief report that describes the results.
- M. Contact the career services department at your university and find all the pertinent documents designed to help students find permanent and/or part-time jobs. Analyze the documents and write a brief report.

## MINICASES

1. The state firefighters' association has a membership of 15,000. The purpose of the organization is to provide some financial support to the families of deceased member firefighters and to organize a conference each year bringing together firefighters from all over the state. Annually, members are billed dues and calls. "Calls" are additional funds required to take care of payments made to the families of deceased members.



The bookkeeping work for the association is handled by the elected treasurer, Bob Smith, although it is widely known that his wife, Laura, does all of the work. Bob runs unopposed each year at the election, since no one wants to take over the tedious and time-consuming job of tracking memberships. Bob is paid a stipend of \$8000 per year, but his wife spends well over 20 hours per week on the job. The organization, however, is not happy with their performance.

A computer system is used to track the billing and receipt of funds. This system was developed in 1984 by a computer science student and his father. The system is a DOS-based system written in dBase 3. The most immediate problem facing the treasurer and his wife is the fact that the software package no longer exists, and there is no one around who knows how to maintain the system. One query in particular takes 17 hours to run. Over the years, they have just avoided running this query, although the information in it would be quite useful. Questions from members concerning their statements cannot easily be answered. Usually, Bob or Laura just jots down the inquiry and returns a call with the answer. Sometimes it takes 3 to 5 hours to find the information needed to answer the question. Often, they have to perform calculations manually, since the system was not programmed to handle certain types of queries. When member information is entered into the system, each field is presented one at a time. This makes it very difficult to return to a field and correct a value that was entered. Sometimes a new member is entered, but disappears from the records. The report of membership used in the conference materials does not alphabetize members by city. Only cities are listed in the correct order.

What requirements analysis technique or techniques would you recommend for this situation? Explain your answer.

2. Brian Callahan, IS project manager, is just about ready to depart for an urgent meeting called by Joe Campbell, manager of manufacturing operations. A major BPI project, sponsored by Joe, recently cleared the approval hurdle, and Brian helped bring the project through project initiation. Now that the approval committee has given the go-ahead, Brian has been working on the project's analysis plan.

One evening, while playing golf with a friend who works in the manufacturing operations department, Brian learned that Joe wants to push the project's time frame up from Brian's original estimate of 13 months. Brian's friend overheard Joe say, "I can't see why that

IS project team needs to spend all that time 'analyzing' things. They've got two weeks scheduled just to look at the existing system! That seems like a real waste. I want that team to get going on building my system."

Because Brian has a little inside knowledge about Joe's agenda for this meeting, he has been considering how to handle Joe. What do you suggest that Brian tell Joe?

3. Barry has recently been assigned to a project team that will be developing a new retail store management system for a chain of submarine sandwich shops. Barry has several years of experience in programming, but has not done much analysis in his career. He was a little nervous about the new work he would be doing, but was confident that he could handle any assignment he was given.

One of Barry's first assignments was to visit one of the submarine sandwich shops and prepare an observation report on how the store operates. Barry planned to arrive at the store around noon, but he chose a store in an area of town he was unfamiliar with, and due to traffic delays and difficulty in finding the store, he did not arrive until 1:30 P.M. The store manager was not expecting him and refused to let a stranger behind the counter until Barry had him contact the project sponsor (the director of store management) back at company headquarters to verify who he was and what his purpose was.

After finally securing permission to observe, Barry stationed himself prominently in the work area behind the counter so that he could see everything. The staff had to maneuver around him as they went about their tasks; however, there were only occasional minor collisions. Barry noticed that the store staff seemed to be going about their work very slowly and deliberately, but he supposed that was because the store wasn't very busy. At first, Barry questioned each worker about what he or she was doing, but the store manager eventually asked him not to interrupt their work so much—he was interfering with their service to the customers.

By 3:30, Barry was a little bored. He decided to leave, figuring that he could get back to the office and prepare his report before 5:00 P.M. that day. He was sure that his team leader would be pleased with his quick completion of his assignment. As he drove, he reflected, "There really won't be much to say in this report. All they do is take the order, make the sandwich, collect the payment, and hand over the order. It's really simple!" Barry's confidence in his analytical skills soared as he anticipated his team leader's praise.



Back at the store, the store manager shook his head, commenting to his staff, “He comes here at the slowest time of day on the slowest day of the week. He never even looked at all the work I was doing in the back room while he was here—summarizing yesterday’s sales, checking inventory on hand, making up resupply orders for the weekend ... plus he never even considered our store opening and closing procedures. I hate to think that the new store management system is going to be built by someone like that. I’d better contact Chuck (the director of store management) and let him know what went on here today.” Evaluate Barry’s conduct of the observation assignment.

4. Anne has been given the task of conducting a survey of sales clerks who will be using a new order entry system being developed for a household products catalog company. The goal of the survey is to identify the clerks’ opinions on the strengths and weaknesses of the current system. There are about 50 clerks who work in three different cities, so a survey seemed like an ideal way of gathering the needed information from the clerks.

Anne developed the questionnaire carefully and pretested it on several sales supervisors who were available at corporate headquarters. After revising it according to their suggestions, she sent a paper version of the questionnaire to each clerk, asking that it be returned within one week. After one week, she had only three completed questionnaires returned. After another week, Anne received just two more completed questionnaires. Feeling somewhat desperate, Anne then sent out an e-mail version of the questionnaire, again to all the clerks, asking them to respond to the questionnaire by e-mail as soon as possible. She received two e-mail questionnaires and three messages from clerks who had completed the paper version expressing annoyance at being bothered with the same questionnaire a second time. At this point, Anne has just a 14% response rate, which she is sure will not please her team leader. What suggestions do you have that could have improved Anne’s response rate to the questionnaire?