CHAPTER • 12 •

Systems Development: Investigation and Analysis

PRINCIPLES	LEARNING OBJECTIVES
▪ Effective systems development requires a team effort from stakeholders, users, managers, systems development specialists, and various support personnel, and it starts with careful planning.	▪ Identify the key participants in the systems development process and discuss their roles. ▪ Define the term *information systems planning* and list several reasons for initiating a systems project.
▪ Systems development often uses tools to select, implement, and monitor projects, including net present value (NPV), prototyping, rapid application development, CASE tools, and object-oriented development.	▪ Discuss the key features, advantages, and disadvantages of the traditional, prototyping, rapid application development, and end-user systems development life cycles. ▪ Identify several factors that influence the success or failure of a systems development project. ▪ Discuss the use of CASE tools and the object-oriented approach to systems development.
▪ Systems development starts with investigation and analysis of existing systems.	▪ State the purpose of systems investigation. ▪ Discuss the importance of performance and cost objectives. ▪ State the purpose of systems analysis and discuss some of the tools and techniques used in this phase of systems development.

Information Systems in the Global Economy GRUMA, Mexico

Systems Development South of the Border

GRUMA is a global business based in Monterrey, Nuevo Leòn, Mexico. GRUMA is the world leader in corn flour and tortilla production. It runs operations in the United States, Mexico, Central America, Venezuela, Europe, and China. About 19,000 employees work for GRUMA, and the company earns $3.2 billion in annual revenue.

GRUMA has only recently expanded its operation to Europe and China. One factor that allowed the expansion is a complete redesign of GRUMA's core information systems. The systems development process contributes to the success of the new systems.

GRUMA began operations in Mexico in 1949 based on progressive ideas. The company set goals to revolutionize the corn flour and tortilla industry through modern industrialization that was ecologically sound and efficient. Before long, the company grew to be the largest in the country and planned to expand beyond its borders. After establishing branches in the United States and South America, GRUMA faced challenges to further growth.

The company's information systems were designed to handle only one country's currency, taxes, and regulations. Different systems were designed for different countries—GRUMA in Mexico used one system, another system in the United States, and others in South American countries. To expand further, GRUMA needed a flexible centralized system that adapted to the economic requirements of many countries. Information systems are designed and implemented to meet the primary goals and strategic plans of a business. During the systems investigation phase, GRUMA discovered that its current system could not support its goal of becoming a global business. The company decided to conduct research into developing a new system that could help to achieve its goals.

Through a process called systems analysis, GRUMA studied their existing systems to discover what changes they needed. GRUMA's information systems team interviewed internal stakeholders—employees who interacted with the system and others who were otherwise impacted by the system. Through the interviews, the information systems team learned how the current system was used, what operations were effective, and what operations needed an overhaul. Realizing that significant changes were necessary, GRUMA decided to contract with an information systems firm to design the new system. It developed a request for proposals (RFP) to find a company to provide assistance at a reasonable cost.

After negotiations with several companies, GRUMA selected the information systems company SAP to develop an ERP system for the company. SAP dedicated systems analysts to work with GRUMA's information systems team to design a new system on which to base its operations. At this point, the systems development process progressed from investigation to analysis and then to design. Together, the two companies designed a system that accommodated "country-specific variances for taxes, product requirements, and currencies, languages, and cultural differences," according to SAP.

The team designed a template that GRUMA could distribute to international companies it acquired to standardize operations in all corporate facilities. The template would minimize work to be done setting up new facilities. The system also supported a variety of character sets so it could work with international languages, including Chinese.

During the systems implementation stage, members of the project team at GRUMA tested several prototypes before rolling out the new system for use. The team decided to use a phase-in approach to implement the system gradually. If problems occurred, business would not be interrupted. The new system was deployed at GRUMA's headquarters in Mexico for testing. After a successful launch, the system was deployed to GRUMA operations worldwide.

GRUMA management is thrilled with the new system, saying it allows them to better control growth and more quickly merge new acquisitions into in-house resources. The company can manage global operations in real time, reacting to market changes and new customer requirements as they arise.

Now in the operations and maintenance phase, GRUMA's new system is a huge success. GRUMA's information systems team is reviewing the system to measure its success and identify areas that can be improved. The team is looking to fine-tune features to better coordinate with systems managed by its business partners. It also plans to expand customer relationship management (CRM) tools. The work by GRUMA's systems team and SAP systems analysts has so impressed the executives at GRUMA that they now view the team as a value-creation unit with many more projects to investigate.

As you read this chapter, consider the following:

- What situations can arise in a business to trigger new systems development initiatives?
- What are the best methods for a business to use in approaching new systems development projects?

Why Learn About Systems Development?

Throughout this book, you have seen many examples of the use of information systems in a variety of careers. But where do you start to acquire these systems or have them developed? How can you work with IS personnel, such as systems analysts and computer programmers, to get what you need to succeed on the job or in your own business? This chapter, the first of two chapters on systems development, provides the answers to these questions. You will see how you can initiate the systems development process and analyze your needs with the help of IS personnel. Systems investigation and systems analysis are the first two steps of the systems development process. This chapter provides specific examples of how new or modified systems are initiated and analyzed in a number of industries. In this chapter, you will learn how your project can be planned, aligned with corporate goals, rapidly developed, and much more. We start with an overview of the systems development process.

When an organization needs to accomplish a new task or change a work process, how does it do so? It develops a new system or modifies an existing one. Systems development is the activity of creating new systems or modifying existing systems. It refers to all aspects of the process—from identifying problems to solve or opportunities to exploit to implementing and refining the chosen solution.

AN OVERVIEW OF SYSTEMS DEVELOPMENT

In today's businesses, managers and employees in all functional areas work together and use business information systems. As a result, they are helping with development and, in many cases, leading the way. Users might request that a systems development team determine whether they should purchase a few PCs or create an attractive Web site, using the tools discussed in Chapter 7. In another case, an entrepreneur might use systems development to build an Internet site to compete with large corporations.

This chapter and the next provide you with a deeper appreciation of the systems development process. Individuals can also use systems development to their advantage. Systems development skills and techniques discussed in this chapter and the next can help people launch their own businesses.[1] When Marc Mallow couldn't find off-the-shelf software to schedule workers, he took a few years to develop his own program. The software he created became the core of a New York-based company he founded. Corporations and nonprofit organizations use systems development to achieve their goals. First Health of the Carolinas, for example, upgraded its old imaging system to slash costs and provide better healthcare for patients. The nonprofit health organization reduced costs by more than 30 percent and offered doctors better radiological images to improve patient care.[2]

This chapter will also help you avoid systems development failures or projects that go over budget. In one example, a large $4 billion systems development effort to convert older, paper-based medical records to electronic records for a large healthcare company ran into trouble when it exceeded its budget.[3] In some cases, poorly executed systems development efforts can be costly. A tax system developed for the District of Columbia at a cost of $100 million didn't prevent tax fraud of about $20 million.[4] The fraud involved cashing refund checks sent to fictitious corporate accounts. In other cases, systems development failures can be deadly.[5] According to the CIO of Duke University's Health System, "Issues arising from badly designed and poorly integrated healthcare IT systems harm and kill more patients every year than do medications and medical devices."

To stay competitive in today's global economy, some cities and counties, including Chattanooga, Tennessee, are investing in high-speed fiber-optic cables that have the potential to deliver greater speed compared to existing cable and phone company offerings.[6] In the United States, less than 60 percent of the population has broadband Internet access, while some countries like Denmark and the Netherlands have more than 75 percent of their population with broadband Internet access. South Korea has over 90 percent of its citizens on broadband Internet.

Participants in Systems Development

Effective systems development requires a team effort. The team usually consists of stakeholders, users, managers, systems development specialists, and various support personnel. This team, called the *development team*, is responsible for determining the objectives of the information system and delivering a system that meets these objectives. Many development teams use a project manager to head the systems development effort combined with the project management approach to help coordinate the systems development process. A *project* is a planned collection of activities that achieves a goal, such as constructing a new manufacturing plant or developing a new decision support system. All projects have a defined starting point and ending point, normally expressed as dates such as August 4 and December 11. Most have a budget, such as $150,000. A *project manager* is responsible for coordinating all people and resources needed to complete a project on time. The project manager can make the difference between project success and failure. According to Tyrone Howard, founder of BizNova Consulting, "A project management system is just a tool. It is like this: A carpenter can buy a hammer, but the hammer won't build a house.... In IT, it's the people who do the building, not the technology."[7] In systems development, the project manager can be an IS person inside the organization or an external consultant hired to complete the project. Project managers need technical, business, and people skills. In addition to completing the project on time and within the specified budget, the project manager is usually responsible for controlling project quality, training personnel, facilitating communications, managing risks, and acquiring any necessary equipment, including office supplies and sophisticated computer systems. Research studies have shown that project management success factors include good leadership from executives and project managers, a high level of trust in the project and its potential benefits, and the commitment of the project team and organization to successfully complete the project and implement its results. Project escalation, where the size and scope of a new systems development effort greatly expands over time, is a major problem for project managers.[8] Project escalation often causes projects to go over budget and behind schedule.

In the context of systems development, **stakeholders** are people who, either themselves or through the area of the organization they represent, ultimately benefit from the systems development project. **Users** are people who will interact with the system regularly. They can be employees, managers, or suppliers. For large-scale systems development projects, where the investment in and value of a system can be high, it is common for senior-level managers,

stakeholders
People who, either themselves or through the organization they represent, ultimately benefit from the systems development project.

users
People who will interact with the system regularly.

including the functional vice presidents (of finance, marketing, and so on), to be part of the development team.

Because stakeholders ultimately benefit from the systems development project, they often work with others in developing a computer application.

(Source: © Reza Estakhrian/Getty Images.)

systems analyst
A professional who specializes in analyzing and designing business systems.

programmer
A specialist responsible for modifying or developing programs to satisfy user requirements.

Depending on the nature of the systems project, the development team might include systems analysts and programmers, among others. A **systems analyst** is a professional who specializes in analyzing and designing business systems. Systems analysts play various roles while interacting with the stakeholders and users, management, vendors and suppliers, external companies, programmers, and other IS support personnel (see Figure 12.1). Like an architect developing blueprints for a new building, a systems analyst develops detailed plans for the new or modified system. The **programmer** is responsible for modifying or developing programs to satisfy user requirements. Like a contractor constructing a new building or renovating an existing one, the programmer takes the plans from the systems analyst and builds or modifies the necessary software. The demand for systems analysts and computer programmers is expected to increase.[9] In Canada, the unemployment rate for IS professionals is about one-third the national average. According to the chairman of the Computer Science Department at the University of Toronto, "The numbers are quite stark. It's clear the demand in the workforce is there."

The other support personnel on the development team are mostly technical specialists, including database and telecommunications experts, hardware engineers, and supplier representatives. One or more of these roles might be outsourced to outside experts or consultants. Depending on the magnitude of the systems development project and the number of IS systems development specialists on the team, one or more IS managers might also belong to the team. The composition of a development team can vary over time and from project to project. For small businesses, the development team might consist of a systems analyst and the business owner as the primary stakeholder. For larger organizations, formal IS staff can include hundreds of people involved in a variety of activities, including systems development. Every development team should have a team leader. This person can be from the IS department, a manager from the company, or a consultant from outside the company. The team leader needs both technical and people skills.

Today, companies are using innovative ways to build new systems or modify existing ones without using in-house programmers. Outsourcing, which is discussed later in the chapter, is one approach.[10] Constellation Energy, a $19 billion utility company, is using another approach that asks programmers from around the world to get involved. The approach, called *crowd sourcing*, asks programmers to contribute code to the project.[11] Winning programmers that submit excellent code can be given from $500 to more than $2,000. Constellation is hoping to save time and money by using crowd sourcing, but neither result is guaranteed. In addition, the resulting programming code may not be consistent with what the company is expecting.

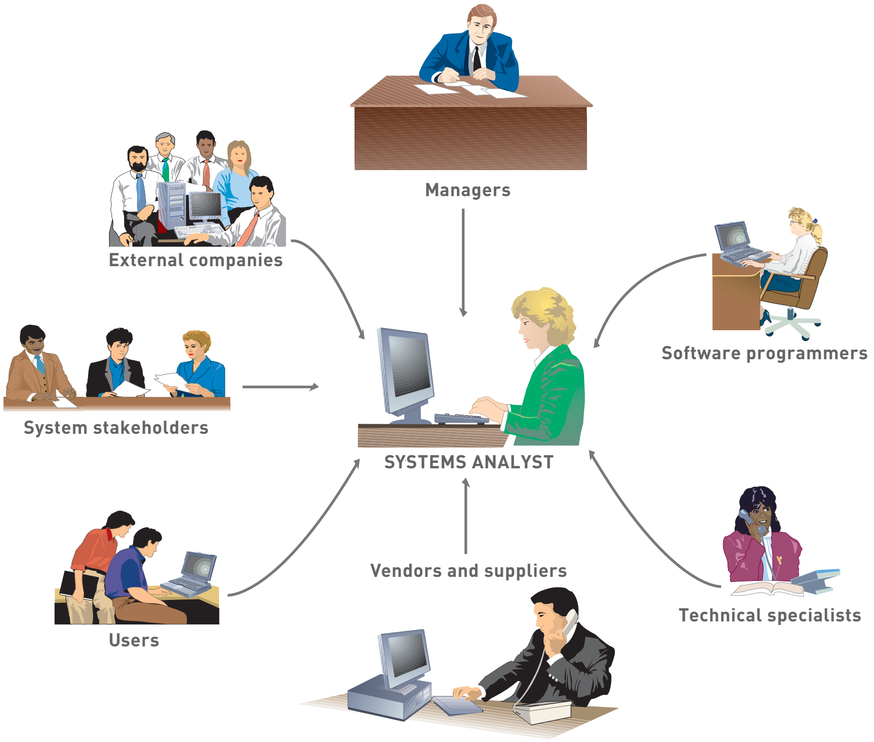


Figure 12.1

Role of the Systems Analyst

The systems analyst plays an important role in the development team and is often the only person who sees the system in its totality. The one-way arrows in this figure do not mean that there is no direct communication between other team members. These arrows just indicate the pivotal role of the systems analyst—a person who is often called on to be a facilitator, moderator, negotiator, and interpreter for development activities.

Regardless of the specific nature of a project, systems development creates or modifies systems, which ultimately means change. Managing this change effectively requires development team members to communicate well. Because you probably will participate in systems development during your career, you must learn communication skills. You might even be the individual who initiates systems development.

Initiating Systems Development

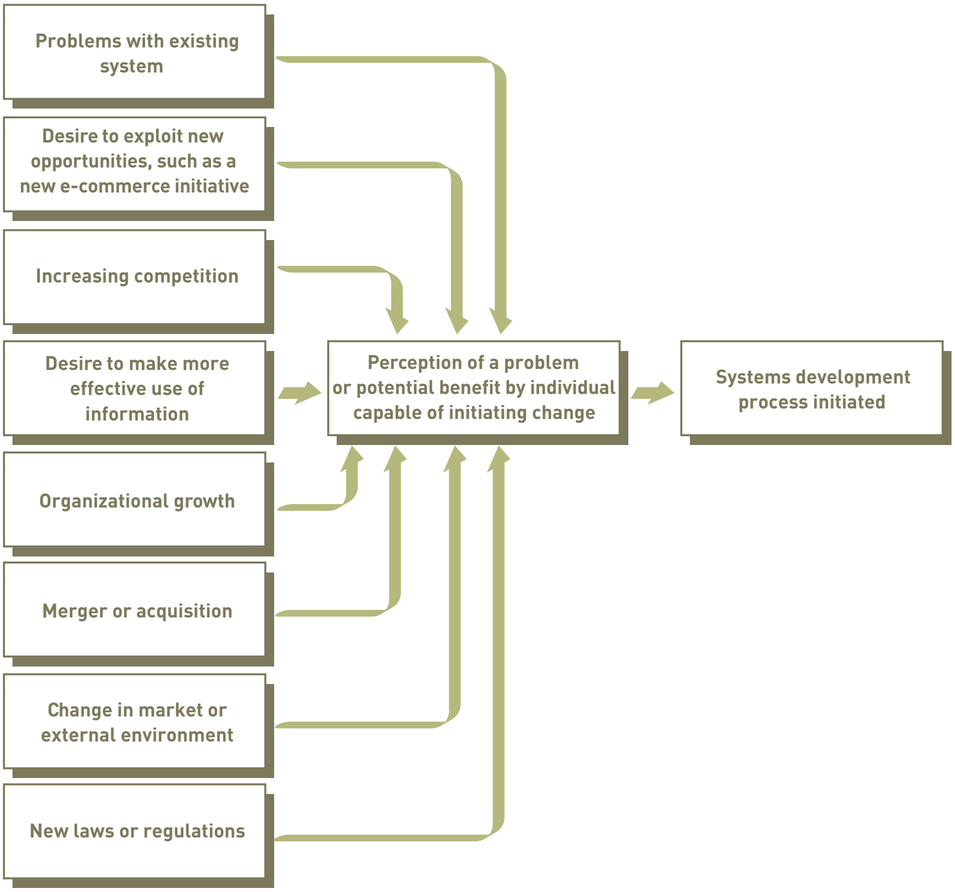
Systems development initiatives arise from all levels of an organization and are both planned and unplanned. Systems development projects are initiated for many reasons, as shown in Figure 12.2.

As shown in Figure 12.2, problems with the existing system can initiate systems development activity. Hannaford Brothers, a large grocer located in Maine, decided to upgrade its security system after millions of credit and debit card records were stolen from its computer system. The security upgrade is expected to cost millions of dollars.[12] The desire to exploit new opportunities is another cause of systems initiation. The increased use of the *cloud computing* approach, discussed in Chapter 7, has many IS professionals looking into using the Internet for applications, such as word processing and spreadsheet analysis, instead of putting these applications on desktop or laptop computers. According to Internet pioneer Marc Andreessen, “The cloud is a smart, complex, powerful computing system in the sky that people can just plug into.”[13] Mergers and acquisitions can trigger many systems development projects.[14] Because information systems often vary within a company, a large systems development effort is typically required to unify systems. Even with similar information systems, the procedures, culture, training, and management of the information systems are often different, requiring a realignment of the IS departments. In another case, Six Flags, one of the largest amusement park companies in the world with about $1 billion in annual sales, initiated a systems development project to build a sophisticated inventory control system to increase revenues.[15] According to CIO Michael Israel, “If a food stand is running low, we know at mid-day instead of the end of the day.”

A company’s customers or suppliers can trigger systems development. Daisy Brand, a dairy products company, was asked by one of its major customers, Wal-Mart, to start using special RFID tags.[16] Although the Wal-Mart RFID initiative was never fully implemented

Figure 12.2

Typical Reasons to Initiate a Systems Development Project


Problems with existing system
Desire to exploit new opportunities, such as a new e-commerce initiative
Increasing competition
Desire to make more effective use of information
Organizational growth
Merger or acquisition
Change in market or external environment
New laws or regulations
Perception of a problem or potential benefit by individual capable of initiating change
Systems development process initiated

by all of its customers, Dairy Products benefitted from the technology by streamlining its inventory processing. By putting RFID tags on every pallet of dairy products that it ships to customers, the company cut in half the time it used to take to load pallets onto delivery trucks.

Systems development can also be initiated when a vendor no longer supports an older system or older software. When this support is no longer available, companies are often forced to upgrade to new software and systems, which can be expensive and require additional training.[17] Major systems and application software companies, for example, often stop supporting their older software a few years after new software has been introduced. Some printer and computer vendors do the same. They stop providing support for their older systems after newer ones are introduced and sold in the market. This lack of support is a dilemma for many companies trying to keep older systems operational.

The federal government can foster new systems development projects in the private sector. As a result of some financial scandals, the government has instituted new corporate financial reporting rules under the Sarbanes-Oxley Act. These regulations have caused many U.S. companies to initiate systems development efforts. To comply with this law, companies can spend hundreds of thousands or millions of dollars in new systems development efforts.

Information Systems Planning and Aligning Corporate and IS Goals

Information systems planning and aligning corporate and IS goals are important aspects of any systems development project.[18] Achieving a competitive advantage is often the overall objective of systems development.

Information Systems Planning

The term **information systems planning** refers to translating strategic and organizational goals into systems development initiatives (see Figure 12.3).[19] Proper IS planning ensures that specific systems development objectives support organizational goals. Long-range planning can also be important and result in getting the most from a systems development effort. It can also align IS goals with corporate goals and culture, which is discussed next.[20] Hess Corporation, a large energy company with over 1,000 retail gasoline stations, uses long-range planning to determine what computer equipment they need and the IS personnel needed to run it.[21] According to Hess's CIO, "It became pretty clear that we needed to lay out a long-term strategy that would allow us to figure out how IT could support our business strategy over the next five years."

information systems planning
Translating strategic and organizational goals into systems development initiatives.

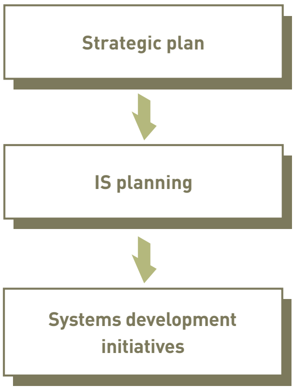


Figure 12.3

Information Systems Planning

Information systems planning transforms organizational goals outlined in the strategic plan into specific systems development activities.

Aligning Corporate and IS Goals

Aligning organizational goals and IS goals is critical for any successful systems development effort. Because information systems support other business activities, IS staff and people in other departments need to understand each other's responsibilities and tasks. Determining whether organizational and IS goals are aligned can be difficult, so researchers have increasingly tackled the problem. Most corporations, for example, have profits and return on investment (ROI), first introduced in Chapter 2, as primary goals. Procter & Gamble (P&G) uses ROI to measure the success of its projects and systems development efforts.[22] P&G produces Tide, Pringles, Pampers, and many other consumer products. The huge consumer-products company has a $76 billion annual supply chain. ROI calculations help companies like P&G prioritize systems development projects and align them with corporate goals. Providing outstanding service is another important corporate goal.[23] Coca-Cola Enterprises, which is Coca-Cola's largest bottler and distributor, decided to use online services from Microsoft and SharePoint to speed its systems development process.[24] According to the company CIO, "This is not a head-count reduction for us. Services are complementary to our IT strategy."

Specific systems development initiatives can spring from the IS plan, but the IS plan must also provide a broad framework for future success. The IS plan should guide development of the IS infrastructure over time. Another benefit of IS planning is that it ensures better use of IS resources—including funds, personnel, and time for scheduling specific projects. The steps of IS planning are shown in Figure 12.4.

Developing a Competitive Advantage

In today's business environment, many companies seek systems development projects that will provide them with a competitive advantage. Thinking competitively usually requires creative and critical analysis. By looking at problems in new or different ways and by introducing innovative methods to solve them, many organizations have gained significant competitive advantage.

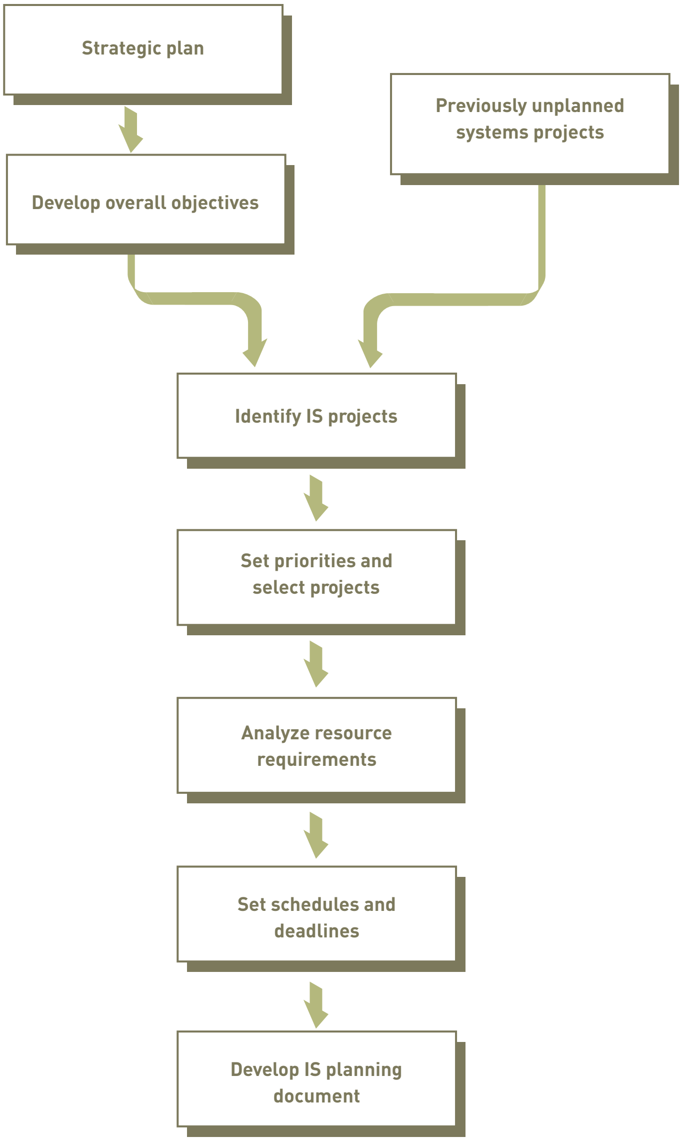
Creative analysis involves investigating new approaches to existing problems. By looking at problems in new or different ways and by introducing innovative methods to solve them, many firms have gained a competitive advantage. Typically, these new solutions are inspired

creative analysis
The investigation of new approaches to existing problems.

Figure 12.4

The Steps of IS Planning

Some projects are identified through overall IS objectives, whereas additional projects, called *unplanned projects*, are identified from other sources. All identified projects are then evaluated in terms of their organizational priority.


Strategic plan
Previously unplanned systems projects
Develop overall objectives
Identify IS projects
Set priorities and select projects
Analyze resource requirements
Set schedules and deadlines
Develop IS planning document

by people and events not directly related to the problem. Creative analysis can help organizations achieve their performance goals. According to Michael Hugos, principle at the Center for Systems Innovation and one of *Computerworld*'s 2006 Premier 100 IT Leaders, "Creativity is where we come up with ideas for combining available resources to create systems that could meet performance requirements."[25]

critical analysis
The unbiased and careful questioning of whether system elements are related in the most effective ways.

Critical analysis requires unbiased and careful questioning of whether system elements are related in the most effective ways. It involves considering the establishment of new or different relationships among system elements and perhaps introducing new elements into the system. Critical analysis in systems development involves the following actions:

- **Questioning statements and assumptions.** Questioning users about their needs and clarifying their initial responses can result in better systems and more accurate predictions. Too often, stakeholders and users specify certain system requirements because they assume that their needs can only be met that way.
- **Identifying and resolving objectives and orientations that conflict.** Each department in an organization can have different objectives and orientations. The buying department might want to minimize the cost of spare parts by always buying from the lowest-cost supplier, but engineering might want to buy more expensive, higher-quality spare parts to reduce the frequency of replacement. These differences must be identified and resolved before a new purchasing system is developed or an existing one modified.

INFORMATION SYSTEMS @ WORK

Investigating Conversion at Art.com

Art.com was an early Web pioneer, launching in 1995 with the purpose of selling all kinds of visual art online. Since then, Art.com has assisted over 4 million customers in decorating their walls by providing a virtual gallery of approximately 400,000 images. The company operates in both the United States and Europe and employs more than 500 worldwide.

Art.com has over 12 million visitors to its Web sites per month, most of whom visit the site without making a purchase. Art.com wanted to increase the percentage of visitors that make purchases—known as the conversion rate—by improving its Web site. With 12 million visitors, even a small improvement could mean a major increase in profits. The challenge was that the Web site had been continuously revised during the company's many years in business, and Art.com's management did not know what changes would improve the visitor's experience. They certainly didn't want to risk changes that might inadvertently turn visitors away.

With the goal clearly articulated to "increase the conversion rate by offering the best customer experience," Art.com systems analysts began to investigate what portions of the current system worked well and what portions could be improved.

The systems investigation proved to be no small task. Art.com draws thousands of images from product lines offered by many online properties. The company had been using a traditional Web analytics information system that recorded information such as number of visitors and which products were most popular; however, the system did not evaluate information on site obstacles that might be discouraging sales. Art.com needed a system that could provide more telling information such as key performance indicators (KPI) that suggested what customers did not like about its site.

The systems team found an off-the-shelf solution that performed more detailed Web analysis. The "online customer experience management solution" allowed systems investigators to view key performance indicators and then review the qualitative details of individual customer sessions on the site. Viewing the basic analytics allowed the investigators to quickly find trends in customer activity. Drilling down into those trends allowed the investigators to "play back" a customer's activities on the site to determine where the customer experienced problems or decided to leave the site. Rather than having to guess what was happening on the site, investigators could track the action in real time.

Using the new online customer experience management solution allowed Art.com to make several improvements to its Web site that contributed to a significant increase in conversion rate and prevented possible disasters.

One example of disaster recovery took place when Art.com sent sale coupons to many of its customers. Unfortunately, the coupon numbers were not entered into the back-end system, so when customers with coupons checked out, they received an "invalid coupon code" error message. Most abandoned their purchase at that point. Art.com's new Web analytics tool alerted management to the problem within hours. The coupon codes were added to the back-end system, and because user data was collected by the system, Art.com contacted those who were frustrated by the error and enticed them back. The quick correction of the problem probably saved Art.com $25,000 of revenue per day.

In another example, the checkout process at Art.com's French site was displaying error messages to customers using outdated browsers. Art.com's new system caught the problem when the alarm was raised and management corrected the problem within days. Again, customers who experienced difficulties were contacted and enticed to return.

In a third example, Art.com's new system showed investigators that up to 20,000 visitors referred by Web search engines were greeted with a page that informed them that "sorry, this product is no longer available." Web developers at Art.com changed the message to be less negative and more inviting by providing alternative products that might interest those visitors.

Systems analysts depend on tools to provide them with information on which portions of systems are working and which are not. Analyzing a Web site such as Art.com is like analyzing pedestrian traffic in a major city—it's impossible without appropriate tools. Using powerful Web analytics and an online customer experience management system, systems analysts can continuously review and investigate the effect that the system is having on Web site visitors, launching systems development projects as needed.

Discussion Questions

1. What was Art.com's biggest challenge in improving their customers' online experience?
2. How does the new online system allow Art.com to launch systems development projects that can improve sales?

Critical Thinking Questions

1. What are some useful functions of a good Web analytics and online customer experience management system?
2. During which stages of the systems development life cycle can Web analytics be useful, and why?

SOURCES: Tealeaf staff, "Art.com: Purveyor of the World's Largest Selection of Wall Décor," Computerworld/TeaLeaf, 2007, *http://zones.computerworld.com/tealeaf_customer_exp/registration.php?item=13&from= cw&src=cwlp*; Art.com Web site, *www.art.com*, accessed July 12, 2008; Tealeaf Web site, *www.tealeaf.com*, accessed July 12, 2008.

Establishing Objectives for Systems Development

The overall objective of systems development is to achieve business goals, not technical goals, by delivering the right information to the right person at the right time. The impact a particular system has on an organization's ability to meet its goals determines the true value of that system to the organization. Southern States, which sells farm equipment in over 20 states and is owned by about 300,000 farmers, decided to use Skyway Software, Inc.'s Visual Workplace to develop a new pricing application to help increase revenue.[26] The use of this service-oriented architecture (SOA) tool allowed Southern States to generate $1.4 million more in revenue the year after it was placed into operation.

Although all systems should support business goals, some systems are more pivotal in continued operations and goal attainment than others. These systems are called **mission-critical systems**. An order processing system, for example, is usually considered mission-critical. Without it, few organizations could continue daily activities, and they clearly would not meet set goals.

mission-critical systems
Systems that play a pivotal role in an organization's continued operations and goal attainment.

The goals defined for an organization also define the objectives that are set for a system. A manufacturing plant, for example, might determine that minimizing the total cost of owning and operating its equipment is critical to meet production and profit goals. **Critical success factors (CSFs)** are factors that are essential to the success of certain functional areas of an organization. The CSF for manufacturing—minimizing equipment maintenance and operating costs—would be converted into specific objectives for a proposed system. One specific objective might be to alert maintenance planners when a piece of equipment is due for routine preventative maintenance (e.g., cleaning and lubrication). Another objective might be to alert the maintenance planners when the necessary cleaning materials, lubrication oils, or spare parts inventory levels are below specified limits. These objectives could be accomplished either through automatic stock replenishment via electronic data interchange or through the use of exception reports.

critical success factors (CSFs)
Factors that are essential to the success of a functional area of an organization.

Regardless of the particular systems development effort, the development process should define a system with specific performance and cost objectives. The success or failure of the systems development effort will be measured against these objectives.

Performance Objectives

The extent to which a system performs as desired can be measured through its performance objectives. System performance is usually determined by factors such as the following:

- **The quality or usefulness of the output.** Is the system generating the right information for a value-added business process or by a goal-oriented decision maker?
- **The accuracy of the output.** Is the output accurate and does it reflect the true situation? As a result of the accounting scandals of the early 2000s, when some companies overstated revenues or understated expenses, accuracy is becoming more important, and top corporate officers are being held responsible for the accuracy of all corporate reports.
- **The speed at which output is generated.** Is the system generating output in time to meet organizational goals and operational objectives? Objectives such as customer response time, the time to determine product availability, and throughput time are examples. For Six Flags, speed is critical.[27] According to the CIO of Six Flags, "Speed per attendee is everything."
- **The scalability of the resulting system.** As mentioned in Chapter 4, s*calability* allows an information system to handle business growth and increased business volume. For example, if a midsized business realizes an annual 10 percent growth in sales for several years, an information system that is scalable will be able to efficiently handle the increase by adding processing, storage, software, database, telecommunications, and other information systems resources to handle the growth.
- **The risk of the system.** One important objective of many systems development projects is to reduce risk.[28] The BRE Bank in Poland (*www.brebank.pl/en*), for example, used systems development to create a model-based DSS to analyze and reduce loan risk and a variety of related risks associated with bank transactions. The project uses a mathematical algorithm, called FIRST (Financial Institutions Risk Scenario Trends), to reduce risk.

In some cases, the achievement of performance objectives can be easily measured (e.g., by tracking the time it takes to determine product availability). In other cases, it is sometimes more difficult to ascertain in the short term. For example, it might be difficult to determine how many customers are lost because of slow responses to customer inquiries regarding product availability. These outcomes, however, are often closely associated with corporate goals and are vital to the long-term success of the organization. Senior management usually dictates their attainment.

Cost Objectives

Organizations can spend more than is necessary during a systems development project. The benefits of achieving performance goals should be balanced with all costs associated with the system, including the following:

- **Development costs.** All costs required to get the system up and running should be included. Some computer vendors give cash rewards to companies using their systems to reduce costs and act as an incentive.
- **Costs related to the uniqueness of the system application.** A system's uniqueness has a profound effect on its cost. An expensive but reusable system might be preferable to a less costly system with limited use.
- **Fixed investments in hardware and related equipment.** Developers should consider costs of such items as computers, network-related equipment, and environmentally controlled data centers in which to operate the equipment.
- **Ongoing operating costs of the system.** Operating costs include costs for personnel, software, supplies, and resources such as the electricity required to run the system. Tridel Corporation (*www.tridel.com*) used systems development to build a new invoicing application, called Invoice Zero, to save over $20,000 in operating costs.[29] The new invoicing application, which consolidated invoices and sent them out once a month, cut the number of monthly invoices from 2,400 to just 17. Reducing costs was also an important factor for Cincinnati Bell. By switching from dedicated PCs to thin client computers and virtualization software, Cincinnati Bell expects to see a large reduction in help desk costs.[30] Some experts predict that help desk costs could be reduced by 70 percent or more. For many IS operations, ongoing operating costs are much higher than development or acquisition costs. According to a Gartner study, acquisition or development cost is only 20 percent of the total cost of a new information system.[31]

Balancing performance and cost objectives within the overall framework of organizational goals can be challenging. Setting objectives is important, however, because they allow an organization to allocate resources effectively and measure the success of a systems development effort. For PC manufacturers, for example, parts and components of a typical PC can cost under $500, which includes about $130 for the processor, $100 for a CD or DVD, $100 for memory, $45 for the Windows operating system, and the rest for other hardware parts and components. Some believe these low costs will eventually lead to lower costs for PCs.

Region: Cincinnati Edit

For detailed pricing and promotions, select the area where you live from the following list:

Consumer

Local & long distance, high-speed Internet, wireless phone & TV for your home.

Business

Small Office / Home Office

One-stop convenience for your small office or home office.

Small & Medium Business

Custom solutions to save your business time and money.

Enterprise

Integrated communication solutions for corporations.

My Account Access

My Account makes it easy for you to manage your services, your bills & your time.

Cincinnati Bell reduced help desk costs by switching from dedicated PCs to thin client computers and virtualization software.

SYSTEMS DEVELOPMENT LIFE CYCLES

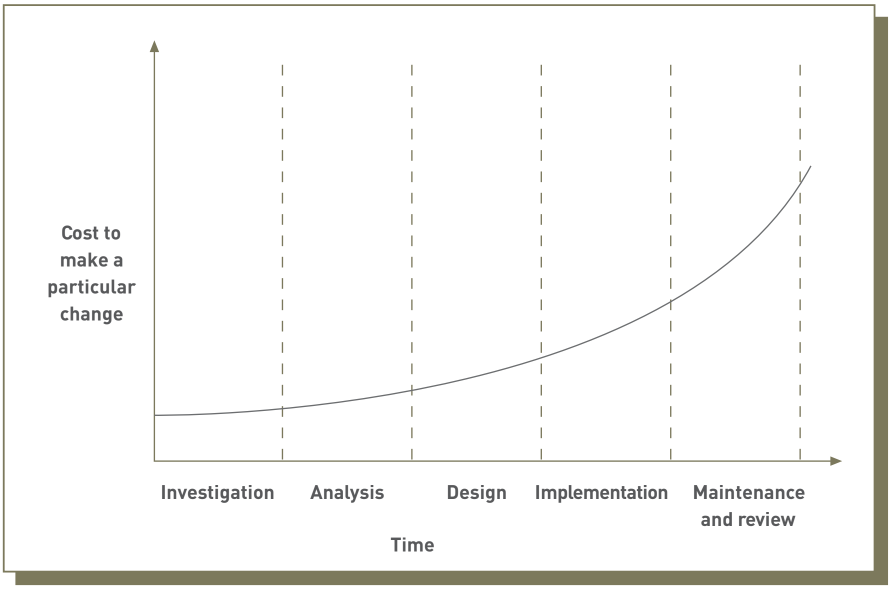
The systems development process is also called a *systems development life cycle (SDLC)* because the activities associated with it are ongoing. As each system is built, the project has timelines and deadlines, until at last the system is installed and accepted. The life of the system continues as it is maintained and reviewed. If the system needs significant improvement beyond the scope of maintenance, if it needs to be replaced because of a new generation of technology, or if the IS needs of the organization change significantly, a new project will be initiated and the cycle will start over.

A key fact of systems development is that the later in the SDLC an error is detected, the more expensive it is to correct (see Figure 12.5). One reason for the mounting costs is that, if an error is found in a later phase of the SDLC, the previous phases must be reworked to some extent. Another reason is that the errors found late in the SDLC affect more people. For example, an error found after a system is installed might require retraining users when a "work-around" to the problem has been found. Thus, experienced systems developers prefer an approach that will catch errors early in the project life cycle.

Figure 12.5

Relationship Between Timing of Errors and Costs

The later that system changes are made in the SDLC, the more expensive these changes become.



Several common systems development life cycles exist: traditional, prototyping, rapid application development (RAD), and end-user development. In addition, companies can outsource the systems development process. With some companies, these approaches are formalized and documented so that systems developers have a well-defined process to follow; other companies use less formalized approaches. Keep Figure 12.5 in mind as you are introduced to alternative SDLCs in the next section.

The Traditional Systems Development Life Cycle

Traditional systems development efforts can range from a small project, such as purchasing an inexpensive computer program, to a major undertaking. The steps of traditional systems development might vary from one company to the next, but most approaches have five common phases: investigation, analysis, design, implementation, and maintenance and review (see Figure 12.6).

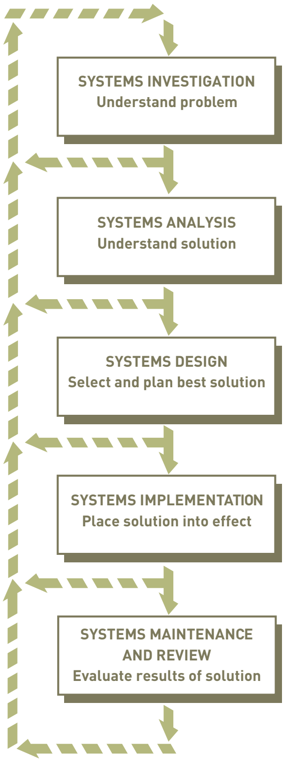

SYSTEMS INVESTIGATION
Understand problem
SYSTEMS ANALYSIS
Understand solution
SYSTEMS DESIGN
Select and plan best solution
SYSTEMS IMPLEMENTATION
Place solution into effect
SYSTEMS MAINTENANCE AND REVIEW
Evaluate results of solution

Figure 12.6

The Traditional Systems Development Life Cycle

Sometimes, information learned in a particular phase requires cycling back to a previous phase.

In the **systems investigation** phase, potential problems and opportunities are identified and considered in light of the goals of the business. Systems investigation attempts to answer the questions "What is the problem, and is it worth solving?" The primary result of this phase is a defined development project for which business problems or opportunity statements have been created, to which some organizational resources have been committed, and for which systems analysis is recommended. **Systems analysis** attempts to answer the question "What must the information system do to solve the problem?" This phase involves studying existing systems and work processes to identify strengths, weaknesses, and opportunities for improvement. The major outcome of systems analysis is a list of requirements and priorities. **Systems design** seeks to answer the question "How will the information system do what it must do to obtain the problem solution?" The primary result of this phase is a technical design that either describes the new system or describes how existing systems will be modified. The system design details system outputs, inputs, and user interfaces; specifies hardware,

systems investigation
The systems development phase during which problems and opportunities are identified and considered in light of the goals of the business.

systems analysis
The systems development phase that determines what the information system must do to solve the problem by studying existing systems and work processes to identify strengths, weaknesses, and opportunities for improvement.

systems design
The systems development phase that defines how the information system will do what it must do to obtain the problem solution.

software, database, telecommunications, personnel, and procedure components; and shows how these components are related. **Systems implementation** involves creating or acquiring the various system components detailed in the systems design, assembling them, and placing the new or modified system into operation.[32] An important task during this phase is to train the users. Systems implementation results in an installed, operational information system that meets the business needs for which it was developed. It can also involve phasing out or removing old systems, which can be difficult for existing users, especially when the systems are free. In 2005, Walt Disney developed the *Virtual Magic Kingdom (VMK)* game to celebrate the fiftieth anniversary of Disneyland.[33] The VMK game used Disney avatars and offered virtual rewards to game players. When Disney decided to remove or terminate the game, some players were outraged and protested outside Disney offices in California.

systems implementation
The systems development phase involving the creation or acquisition of various system components detailed in the systems design, assembling them, and placing the new or modified system into operation.

The purpose of **systems maintenance and review** is to ensure that the system operates and to modify the system so that it continues to meet changing business needs. As shown in Figure 12.6, a system under development moves from one phase of the traditional SDLC to the next.

systems maintenance and review
The systems development phase that ensures the system operates and modifies the system so that it continues to meet changing business needs.

The traditional SDLC allows for a large degree of management control. However, a major problem is that the user does not use the solution until the system is nearly complete. Table 12.1 lists advantages and disadvantages of the traditional SDLC.

Table 12.1

Advantages and Disadvantages of Traditional SDLC

Advantages	Disadvantages
Formal review at the end of each phase allows maximum management control.	Users get a system that meets the needs as understood by the developers; this might not be what is really needed.
This approach creates considerable system documentation.	Documentation is expensive and time consuming to create. It is also difficult to keep current.
Formal documentation ensures that system requirements can be traced back to stated business needs.	Often, user needs go unstated or are misunderstood.
It produces many intermediate products that can be reviewed to see whether they meet the users’ needs and conform to standards.	Users cannot easily review intermediate products and evaluate whether a particular product (e.g., data flow diagram) meets their business requirements.

Prototyping

Prototyping takes an iterative approach to the systems development process. During each iteration, requirements and alternative solutions to the problem are identified and analyzed, new solutions are designed, and a portion of the system is implemented.[34] Users are then encouraged to try the prototype and provide feedback (see Figure 12.7). Prototyping begins with creating a preliminary model of a major subsystem or a scaled-down version of the entire system. For example, a prototype might show sample report formats and input screens. After they are developed and refined, the prototypical reports and input screens are used as models for the actual system, which can be developed using an end-user programming language such as Visual Basic. The first preliminary model is refined to form the second- and third-generation models, and so on until the complete system is developed (see Figure 12.8).

Prototypes can be classified as operational or nonoperational. An *operational prototype* is a prototype that works—accesses real data files, edits input data, makes necessary computations and comparisons, and produces real output. A *nonoperational prototype* is a mock-up, or model, that includes output and input specifications and formats. The advantages and disadvantages of prototyping are summarized in Table 12.2.

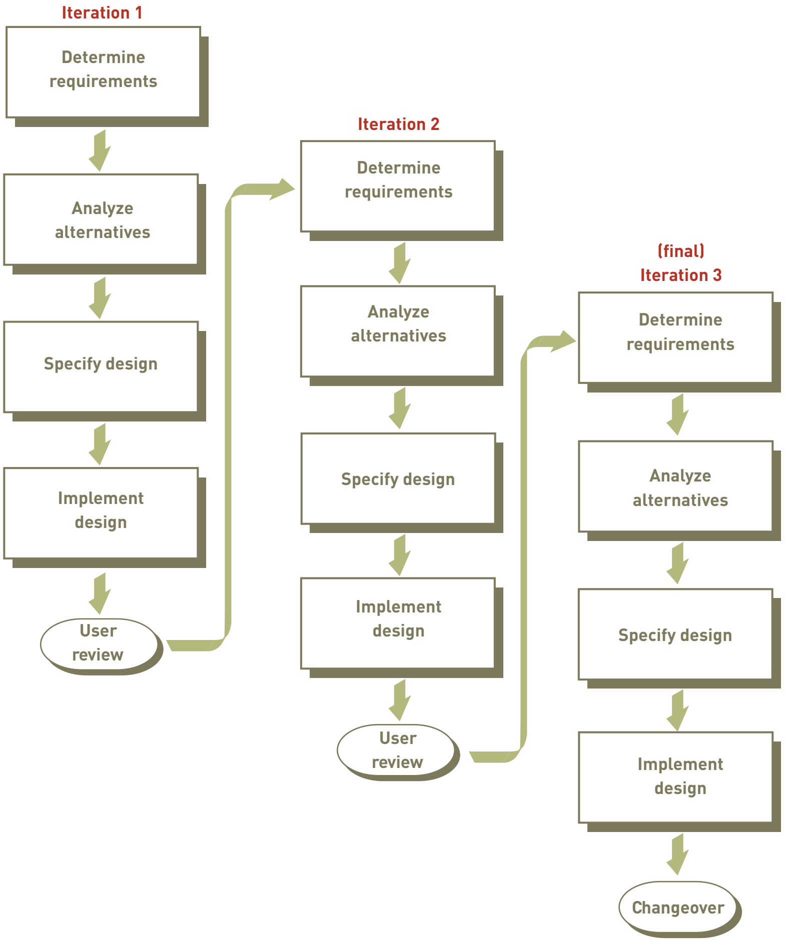


Figure 12.7

Prototyping

Prototyping is an iterative approach to systems development.

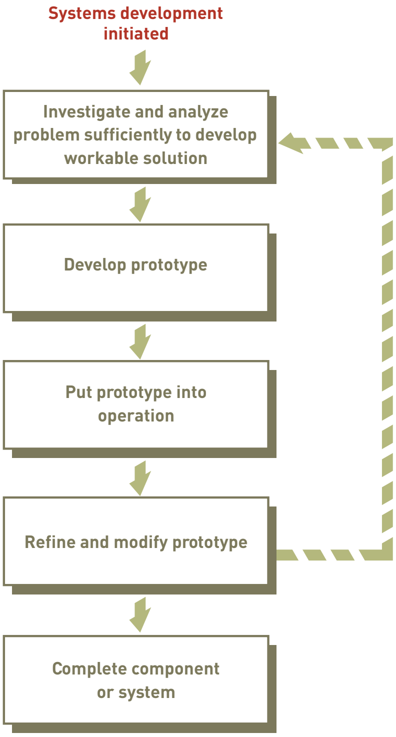

Systems development initiated
Investigate and analyze problem sufficiently to develop workable solution
Develop prototype
Put prototype into operation
Refine and modify prototype
Complete component or system

Figure 12.8

Refining During Prototyping

Each generation of prototype is a refinement of the previous generation based on user feedback.

Table 12.2

Advantages and Disadvantages of Prototyping

Advantages	Disadvantages
Users can try the system and provide constructive feedback during development.	Each iteration builds on the previous one. The final solution might be only incrementally better than the initial solution.
An operational prototype can be produced in weeks.	Formal end-of-phase reviews might not occur. Thus, it is very difficult to contain the scope of the prototype, and the project never seems to end.
As solutions emerge, users become more positive about the process and the results.	System documentation is often absent or incomplete because the primary focus is on development of the prototype.
Prototyping enables early detection of errors and omissions.	System backup and recovery, performance, and security issues can be overlooked in the haste to develop a prototype.

Rapid Application Development, Agile Development, Joint Application Development, and Other Systems Development Approaches

rapid application development (RAD)
A systems development approach that employs tools, techniques, and methodologies designed to speed application development.

Rapid application development (**RAD**) employs tools, techniques, and methodologies designed to speed application development. Vendors, such as Computer Associates International, IBM, and Oracle, market products targeting the RAD market. Rational Software, a division of IBM, has a RAD tool, called Rational Rapid Developer, to make developing large Java programs and applications easier and faster. Locus Systems, a program developer, used a RAD tool called OptimalJ to generate more than 60 percent of the computer code for three applications it developed. Royal Bank of Canada used OptimalJ to develop some customer-based applications. According to David Hewick, group manager of application architecture for the bank, "It was an opportunity to improve the development life cycle, reduce costs, and bring consistency." Advantage Gen, formerly known as COOL:Gen, is a RAD tool from Computer Associates International. It can be used to rapidly generate computer code from business models and specifications.[35]

Other approaches to rapid development, such as *agile development* or *extreme programming* (*XP*), allow the systems to change as they are being developed. Agile development requires frequent face-to-face meetings with the systems developers and users as they modify, refine, and test how the system meets users' needs and what its capabilities are. Microsoft, for example, has adopted a more agile development process in its server development division.[36] According to a Microsoft senior vice president, "We just realized that we're building products for customers, not just for technology's sake. So the sooner we could engage with our customers, the better we could make it from an architecture, feature, quality, and scalability perspective." BT Group, a large British telecommunications company, uses agile systems development to substantially reduce development time and increase customer satisfaction.[37] According to BT's managing director of service design, "BT's shift from traditional waterfall development techniques to an agile approach has led to significant productivity and business benefits, but it didn't happen overnight, nor was it easy for a company as massive and widespread as BT." Extreme programming (XP) uses pairs of programmers who work together to design, test, and code parts of the systems they develop.[38] The iterative nature of XP helps companies develop robust systems with fewer errors. Sabre Airline Solutions, a $2 billion computer company serving the airline travel industry, used XP to eliminate programming errors and shorten program development times.

joint application development (JAD)
A process for data collection and requirements analysis in which users, stakeholders, and IS professionals work together to analyze existing systems, propose possible solutions, and define the requirements of a new or modified system.

RAD makes extensive use of the **joint application development** (JAD) process for data collection and requirements analysis. Originally developed by IBM Canada in the 1970s, JAD involves group meetings in which users, stakeholders, and IS professionals work together to analyze existing systems, propose possible solutions, and define the requirements of a new or modified system. Today, JAD often uses *group support systems (GSS)* software to foster positive group interactions, while suppressing negative group behavior. Boeing, for example, used RAD and JAD to help develop software for its airplanes.[39] Group support systems were introduced in Chapter 10.

RAD should not be used on every software development project. In general, it is best suited for DSSs and MISs and less well suited for TPSs. During a RAD project, the level of participation of stakeholders and users is much higher than in other approaches. Table 12.3 lists advantages and disadvantages of RAD.

Table 12.3

Advantages and Disadvantages of RAD

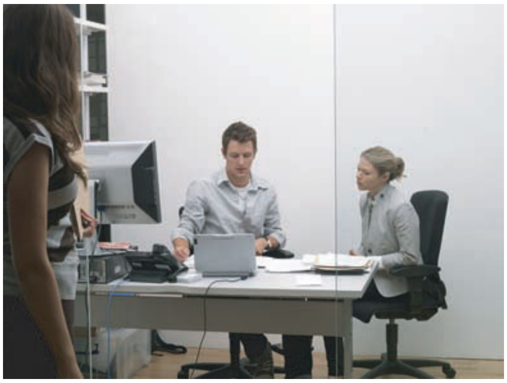
Advantages	Disadvantages
For appropriate projects, this approach puts an application into production sooner than any other approach.	This intense SDLC can burn out systems developers and other project participants.
Documentation is produced as a by-product of completing project tasks.	This approach requires systems analysts and users to be skilled in RAD systems development tools and RAD techniques.
RAD forces teamwork and lots of interaction between users and stakeholders.	RAD requires a larger percentage of stakeholders' and users' time than other approaches.

In addition to the systems development approaches discussed previously, a number of other systems development approaches are available, including adaptive software development, lean software development, Rational Unified Process (RUP), Feature-Driven Development (FDD), and dynamic systems development methods. Often created by computer vendors and authors of systems development books, these approaches all attempt to deliver better systems. The Ohio Casualty Corporation, for example, uses RUP from IBM and Rational Software. RUP uses an iterative approach to software development that concentrates on software quality as it is changed and updated over time.[40] Many other companies have also used RUP to their advantage.[41]

The End-User Systems Development

The term **end-user systems development** describes any systems development project in which business managers and users assume the primary effort. User-developed systems range from the very small (such as a software routine to merge form letters) to those of significant organizational value (such as customer contact databases for the Web). With end-user systems development, managers and other users can get the systems they want without having to wait for IS professionals to develop and deliver them.[42] End-user systems development, however, does have some disadvantages. Some end users don't have the training to effectively develop and test a system. Multimillion-dollar mistakes, for example, can be made using faulty spreadsheets that were never tested. Some end-user systems are also poorly documented. When these systems are updated, problems can be introduced that make the systems error-prone. In addition, some end users spend time and corporate resources developing systems that were already available.

end-user systems development
Any systems development project in which business managers and users assume the primary effort.



Many end users today are demonstrating their systems development capability by designing and implementing their own PC-based systems.

(Source: © Daniel Allan/Getty Images.)

Outsourcing and On-Demand Computing

Many companies hire an outside consulting firm or computer company that specializes in systems development to take over some or all of its development and operations activities.[43] Some companies, such as General Electric, have their own outsourcing subunits or have spun off their outsourcing subunits as separate companies. As mentioned in Chapter 2, *outsourcing* and *on-demand computing* are often used.[44] Table 12.4 describes the circumstances in which outsourcing is a good idea.

Reason	Example
When a company believes it can cut costs	PacifiCare outsourced its IS operations to IBM and Keane, Inc. PacifiCare hopes the outsourcing will save it about $400 million over ten years.
When a firm has limited opportunity to distinguish itself competitively through a particular IS operation or application	Kodak outsourced its IS operations, including mainframe processing, telecommunications, and personal computer support, because it had limited opportunity to distinguish itself through these IS operations. Kodak kept application development and support in-house because it thought that these activities had competitive value.
When outsourcing does not strip the company of technical know-how required for future IS innovation	Firms must ensure that their IS staffs remain technically up-to-date and have the expertise to develop future applications.
When the firm's existing IS capabilities are limited, ineffective, or technically inferior	A company might use outsourcing to help it make the transition from a centralized mainframe environment to a distributed client/server environment.
When a firm is downsizing	First Fidelity, a major bank, used outsourcing as part of a program to reduce the number of employees by 1,600 and slash expenses by $85 million.

Table 12.4

When to Use Outsourcing for Systems Development

Increasingly, small and medium-sized firms are using outsourcing to cut costs and acquire needed technical expertise that would be difficult to afford with in-house personnel. Millennium Partners Sports Club Management, for example, used Center Beam to outsource many of its IS functions, including its help desk operations. The Boston-based company plans to spend about $30,000 a month on outsourcing services, which it estimates to be less than it would have to pay in salaries for additional employees.[45] According to a company vice president, "If we hadn't outsourced, I couldn't focus 100 percent on things that can drive the company forward." The market for outsourcing services for small and medium-sized firms is expected to increase by 15 percent annually through 2010 and beyond.

Reducing costs, obtaining state-of-the-art technology, eliminating staffing and personnel problems, and increasing technological flexibility are reasons that companies have used the outsourcing and on-demand computing approaches.[46] Reducing costs is a primary reason for outsourcing. One American computer company, for example, estimated that a programmer with three to five years of experience in China would cost about $13 per hour, while a programmer with similar experience in the United States would cost about $56 per hour. U.S. companies also provide outsourcing services. Aelera Corporation spent about six months looking for the best outsourcing deal and determined that a company in Savannah, Georgia was the best. McKesson Corporation saved about $10 million by outsourcing jobs from San Francisco to Dubuque, Iowa. Mattel outsourced to rural Jonesboro, Arkansas. Increasingly, companies are looking to American outsourcing companies to reduce costs and increase services. Individuals, including students, are also outsourcing tasks they have to perform.

Companies often use several outsourcing services. GM, the large automotive company, has used six outsourcing companies since its outsourcing agreement with EDS expired.[47] Using more than one outsourcing company can increase competition and reduce outsourcing costs. According to one GM executive, "That's really just Economics 101."

A number of companies and nonprofit organizations offer outsourcing and on-demand computing services—from general systems development to specialized services.[48] IBM's Global Services, for example, is one of the largest full-service outsourcing and consulting services.[49] IBM has consultants located in offices around the world. In India, IBM has increased its employees from less than 10,000 people to more than 30,000.[50] The company looks for skilled and talented workers. According to Amitabh Ray, a vice president of consulting and application services at IBM in India, "We're not a body shop. We need the right kind of people." Electronic Data Systems (EDS) is another large company that specializes in consulting and outsourcing.[51] EDS has approximately 140,000 employees in almost 60 countries and more than 9,000 clients worldwide. Accenture is another company that specializes in consulting and outsourcing.[52] The company has more than 75,000 employees in 47 countries. Amazon, the large online retailer of books and other products, will offer on-demand computing to individuals and other companies of all sizes, allowing them to use Amazon's computer expertise and database capacity.[53] Individuals and companies will only pay for the computer services they use. See Figure 12.9.

Figure 12.9

With more than 75,000 employees in 47 countries, Accenture specializes in consulting and outsourcing.

(Source: © Namas Bhojani/ Bloomberg News/Landov.)

Outsourcing has some disadvantages, however. Some companies, such as J. Crew, are starting to reduce their use of outsourcing and bring systems development back in-house.[54] Internal expertise can be lost and loyalty can suffer under an outsourcing arrangement. When a company outsources, key IS personnel with expertise in technical and business functions are no longer needed. When these IS employees leave, their experience with the organization and expertise in information systems is lost. For some companies, it can be difficult to achieve a competitive advantage when competitors are using the same computer or consulting company. When the outsourcing or on-demand computing is done offshore or in a foreign country, some people have raised security concerns. How will important data and trade secrets be guarded? U.S. federal authorities often investigate defense contractors for improper outsourcing.[55] In one case, a company was fined about $100 million for violating a federal arms export control law.

FACTORS AFFECTING SYSTEMS DEVELOPMENT SUCCESS

Successful systems development means delivering a system that meets user and organizational needs—on time and within budget. Getting users and stakeholders involved in systems development is critical for most systems development projects. Having the support of top-level managers is also important. In addition to user involvement and top management support, other factors can contribute to successful systems development efforts—at a reasonable cost. These factors are discussed next.

Degree of Change

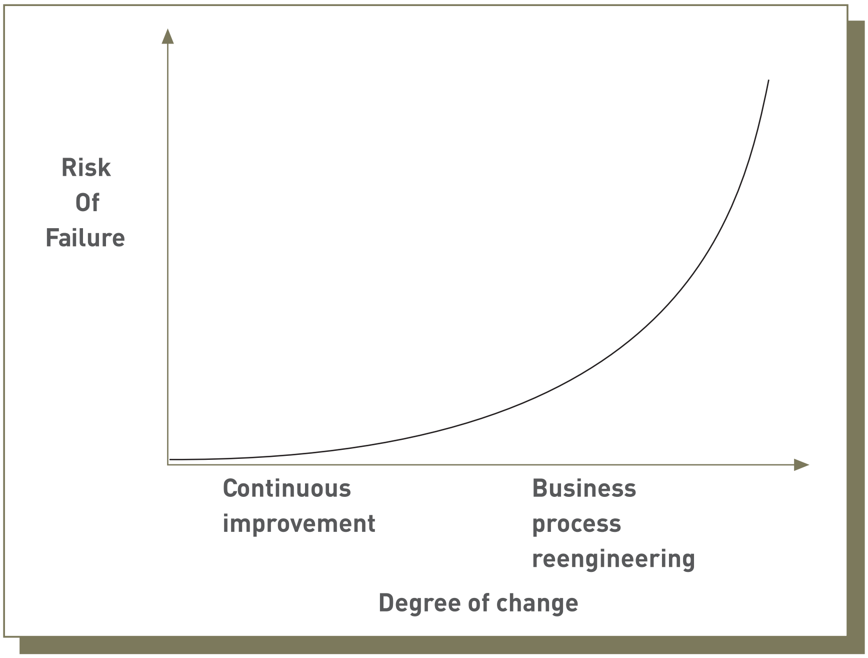
A major factor that affects the quality of systems development is the degree of change associated with the project. The scope can vary from enhancing an existing system to major reengineering. The project team needs to recognize where they are on this spectrum of change.

Continuous Improvement versus Reengineering

As discussed in Chapter 2, continuous improvement projects do not require significant business process or IS changes or retraining of people; thus, they have a high degree of success. Typically, because continuous improvements involve minor improvements, these projects also have relatively modest benefits. On the other hand, reengineering involves fundamental changes in how the organization conducts business and completes tasks. The factors associated with successful reengineering are similar to those of any development effort, including top management support, clearly defined corporate goals and systems development objectives, and careful management of change. Major reengineering projects tend to have a high degree of risk but also a high potential for major business benefits (see Figure 12.10).

Figure 12.10

The degree of change can greatly affect the probability of a project's success.



Managing Change

The ability to manage change is critical to the success of systems development. New systems inevitably cause change. For example, the work environment and habits of users are invariably affected by the development of a new information system. Unfortunately, not everyone adapts easily, and the increasing complexity of systems can multiply the problems. Managing change requires the ability to recognize existing or potential problems (particularly the concerns of users) and deal with them before they become a serious threat to the success of the new or modified system. Here are several of the most common problems that often need to be addressed as a result of new or modified systems:

- Fear that the employee will lose his job, power, or influence within the organization
- Belief that the proposed system will create more work than it eliminates
- Reluctance to work with “computer people”
- Anxiety that the proposed system will negatively alter the structure of the organization
- Belief that other problems are more pressing than those solved by the proposed system or that the system is being developed by people unfamiliar with “the way things need to get done”
- Unwillingness to learn new procedures or approaches

ETHICAL AND SOCIETAL ISSUES

When Systems Development Fails

Systems analysts and developers carry a significant amount of responsibility on their shoulders. Information systems play an important role in the success of today's businesses, and a faulty system can mean the end of a business or worse. When information systems mean life and death to clients, much more is at stake than a business's reputation.

Such was the case with a major healthcare organization, which for the sake of anonymity is called HCO in this article. In 2004, HCO decided it would be more economical to handle all of its kidney transplants itself rather than using a nearby university medical center. HCO built a new kidney transplant center to handle their kidney transplant patients and named a director. The director began to transfer all the patient records from the university medical center to the new center—over 1,500 patients in all.

However, rather than coordinating with the university medical center to transfer patients and their data from one information system to the other, HCO decided to forgo the usual systems development process and rush the transition.

The staff at the previous medical facility found themselves ill-equipped to process and transfer the large number of patient records to the new center in the necessary amount of time. They discovered that the data in many patient records was incorrect, and until they corrected it, the center's staff could not process the patient records. Managing kidney transplants is complex and time sensitive. Kidneys are in rare supply and those eligible for transplants spend time on a waiting list, hoping they will be called before their own kidneys give out. Due to the glitch in data transfer, hundreds of patient records were lost.

To make matters worse, the new transplant center was understaffed and underfunded. Because it did not have proper information systems, the staff at the new center maintained medical records primarily on paper. They did not have a system to determine if any patient records were lost in the transfer, nor could patients use a system to voice concern or lodge a complaint.

Over two years, patients whose records had failed to transfer to the new facility were still waiting for the call for a new kidney that would never come. Finally, based on a whistleblower's story, a local TV station and newspaper began pressing the new center to reveal why patients were waiting longer than usual for transplants. The investigation quickly led to formal litigation against HCO on a number of counts, not the least of which was HCO's failure to adhere to five state and 15 federal regulations mostly dealing with the management of patient records. The state Department of Managed Health Care (DMHC) has concluded that the problems experienced by HCO and its patients are due to "lack of effective planning" and that the absence of proper information management posed "potentially life-threatening delays in care."

In fact, "potential" appears to be "actual" as further investigation shows that in the first year of operations, twice the typical number of patient deaths were caused by an extended wait for kidneys. Professionals in the transplant business say that this is the worst problem the industry has ever seen.

Eventually HCO abandoned its plans for a new center and returned all of its kidney patients to their previous care. The organization has paid $2 million in fines to the state Department of Managed Health Care (DMHC) and volunteered another $3 million in contributions to a transplant education group. Meanwhile, over 50 patients and families of people who died waiting for kidneys are suing the organization in separate cases, mostly for negligence or wrongful death.

As investigators sort through this case seeking an explanation of exactly what went wrong, those involved are accepting some blame, but also pointing fingers at each other. One thing is clear: Had proper systems development practices been put into place, the new kidney transplant center would be operational, patients lives would have been saved, and the reputation of the previously well-respected HCO would still be sparkling.

Discussion Questions

1. What went wrong at HCO? Who paid the price?
2. How is HCO responding to its mistakes, and how might it further regain its good reputation?

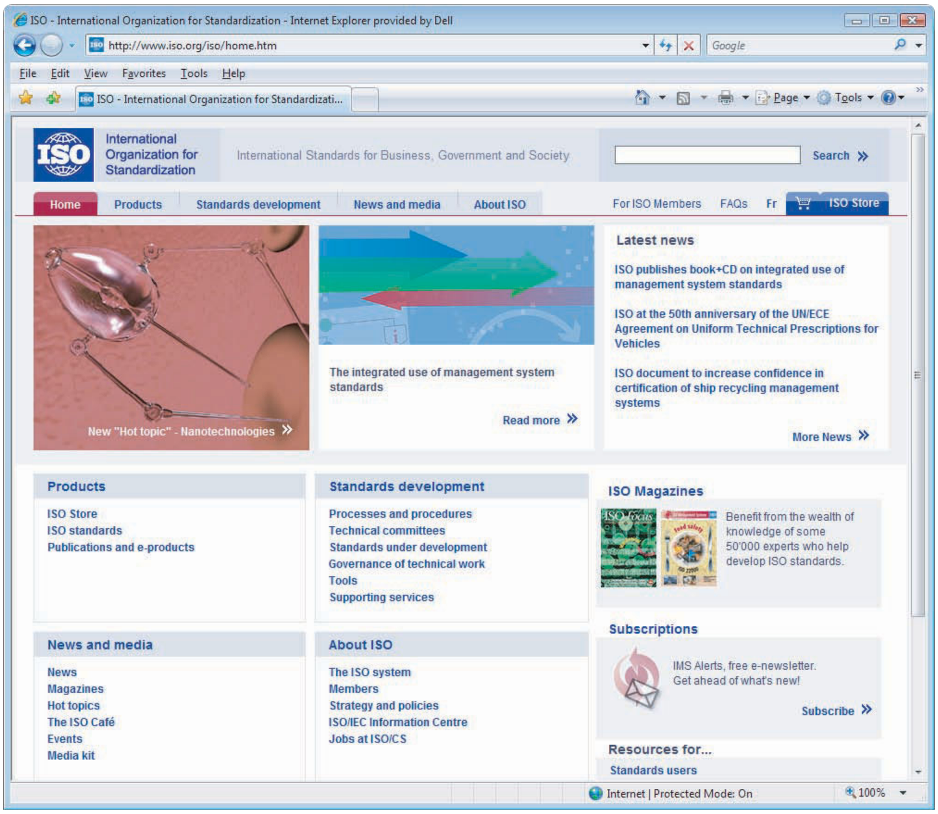
Critical Thinking Questions

1. What legitimate reasons might HCO's director provide for the failure of the new center? Is there any acceptable excuse? Who within HCO is ultimately to blame?
2. What other life-threatening or life-saving information systems are at risk of similar catastrophes?

SOURCES: Gage, Deborah, "We Really Did Screw Up," *CIO Insight*, May 14, 2008, *www.cioinsight.com/c/a/Past-News/QTEWe-Really-Did-Screw-UpQTE*; Kaiser Permanente Web site, *www.kaiserpermanente.org*, accessed July 12, 2008.

Quality and Standards

Quality and standards are other key success factors for systems development. Increasingly, corporations are expanding their standards to include many different computer platforms. While many companies try to standardize their operations on one operating system, others have multiple systems and platforms to take advantage of the strengths of each.[56] In these cases, many IS managers seek one tool to manage everything. According to a Clear Channel IS executive, “I don’t care what enterprise you walk into, they’re not going to be single-platform. I want to manage it all from one spot.” Today, many companies, including Microsoft, are developing software and systems that can be used to manage different operating systems and software products. In addition, organizations that do business around the globe may be required to meet certain international standards, such as ISO 9000, a set of international quality standards originally developed in Europe in 1987.


ISO - International Organization for Standardization - Internet Explorer provided by Dell
http://www.iso.org/iso/home.htm
Google
File Edit View Favorites Tools Help
ISO - International Organization for Standardizati...
Page Tools
ISO
International Organization for Standardization
International Standards for Business, Government and Society
Search
Home
Products
Standards development
News and media
About ISO
For ISO Members
FAQs
Fr
ISO Store
New "Hot topic" - Nanotechnologies
Read more
More News
Products
ISO Store
ISO standards
Publications and e-products
Standards development
Processes and procedures
Technical committees
Standards under development
Governance of technical work
Tools
Supporting services
ISO Magazines
News and media
News
Magazines
Hot topics
The ISO Café
Events
Media kit
About ISO
The ISO system
Members
Strategy and policies
ISO/IEC Information Centre
Jobs at ISO/CS
Subscriptions
Subscribe
Resources for...
Standards users
Internet | Protected Mode: On
100%

The integrated use of management system standards

Latest news

ISO publishes book+CD on integrated use of management system standards

ISO at the 50th anniversary of the UN/ECE Agreement on Uniform Technical Prescriptions for Vehicles

ISO document to increase confidence in certification of ship recycling management systems

Benefit from the wealth of knowledge of some 50'000 experts who help develop ISO standards.

IMS Alerts, free e-newsletter. Get ahead of what's new!

ISO 9000 is a set of international quality standards used by IS and other organizations to ensure the quality of products and services.

(Source: *www.iso.org*.)

The bigger the project, the more likely that poor planning will lead to significant problems. Many companies find that large systems projects fall behind schedule, go over budget, and do not meet expectations. Although proper planning cannot guarantee that these types of problems will be avoided, it can minimize the likelihood of their occurrence. Good systems development is not automatic. Certain factors contribute to the failure of systems development projects. These factors and the countermeasures to eliminate or alleviate the problem are summarized in Table 12.5.

Organizational experience with the systems development process is also an important factor for systems development success.[57] The *Capability Maturity Model (CMM)* is one way to measure this experience. It is based on research done at Carnegie Mellon University and work by the Software Engineering Institute (SEI). CMM is a measure of the maturity of the software development process in an organization. CMM grades an organization’s systems development maturity using five levels: initial, repeatable, defined, managed, and optimized.[58]

Table 12.5

Project Planning Issues Frequently Contributing to Project Failure

Factor	Countermeasure
Solving the wrong problem	Establish a clear connection between the project and organizational goals.
Poor problem definition and analysis	Follow a standard systems development approach.
Poor communication	Set up communications procedures and protocols.
Project is too ambitious	Narrow the project focus to address only the most important business opportunities.
Lack of top management support	Identify the senior manager who has the most to gain from the success of the project, and recruit this person to champion the project.
Lack of management and user involvement	Identify and recruit key stakeholders to be active participants in the project.
Inadequate or improper system design	Follow a standard systems development approach.
Lack of standards	Implement a standards system, such as ISO 9001.

Use of Project Management Tools

Project management involves planning, scheduling, directing, and controlling human, financial, and technological resources for a defined task whose result is achievement of specific goals and objectives.[59] Corporations and nonprofit organizations use these important tools and techniques. As an academic exercise, for example, Purdue University undertook a project to build a supercomputer using off-the-shelf PCs. The project was completed in a day and required more than 800 PCs.[60]

A **project schedule** is a detailed description of what is to be done. Each project activity, the use of personnel and other resources, and expected completion dates are described. A **project milestone** is a critical date for the completion of a major part of the project. The completion of program design, coding, testing, and release are examples of milestones for a programming project. The **project deadline** is the date the entire project is to be completed and operational—when the organization can expect to begin to reap the benefits of the project. One company offers a 20 percent refund if it doesn't meet a client's project deadline. In addition, any additional work done after the project deadline is performed free of charge.

project schedule
A detailed description of what is to be done.

project milestone
A critical date for the completion of a major part of the project.

project deadline
The date the entire project is to be completed and operational.

In systems development, each activity has an earliest start time, earliest finish time, and slack time, which is the amount of time an activity can be delayed without delaying the entire project. The **critical path** consists of all activities that, if delayed, would delay the entire project. These activities have zero slack time. Any problems with critical-path activities will cause problems for the entire project. To ensure that critical-path activities are completed in a timely fashion, formalized project management approaches have been developed. Tools such as Microsoft Project are available to help compute these critical project attributes.

critical path
Activities that, if delayed, would delay the entire project.

Although the steps of systems development seem straightforward, larger projects can become complex, requiring hundreds or thousands of separate activities. For these systems development efforts, formal project management methods and tools become essential. A formalized approach called **Program Evaluation and Review Technique** (PERT) creates three time estimates for an activity: shortest possible time, most likely time, and longest possible time. A formula is then applied to determine a single PERT time estimate. A **Gantt chart** is a graphical tool used for planning, monitoring, and coordinating projects; it is essentially a grid that lists activities and deadlines. Each time a task is completed, a marker such as a darkened line is placed in the proper grid cell to indicate the completion of a task (see Figure 12.11).

Program Evaluation and Review Technique (PERT)
A formalized approach for developing a project schedule that creates three time estimates for an activity.

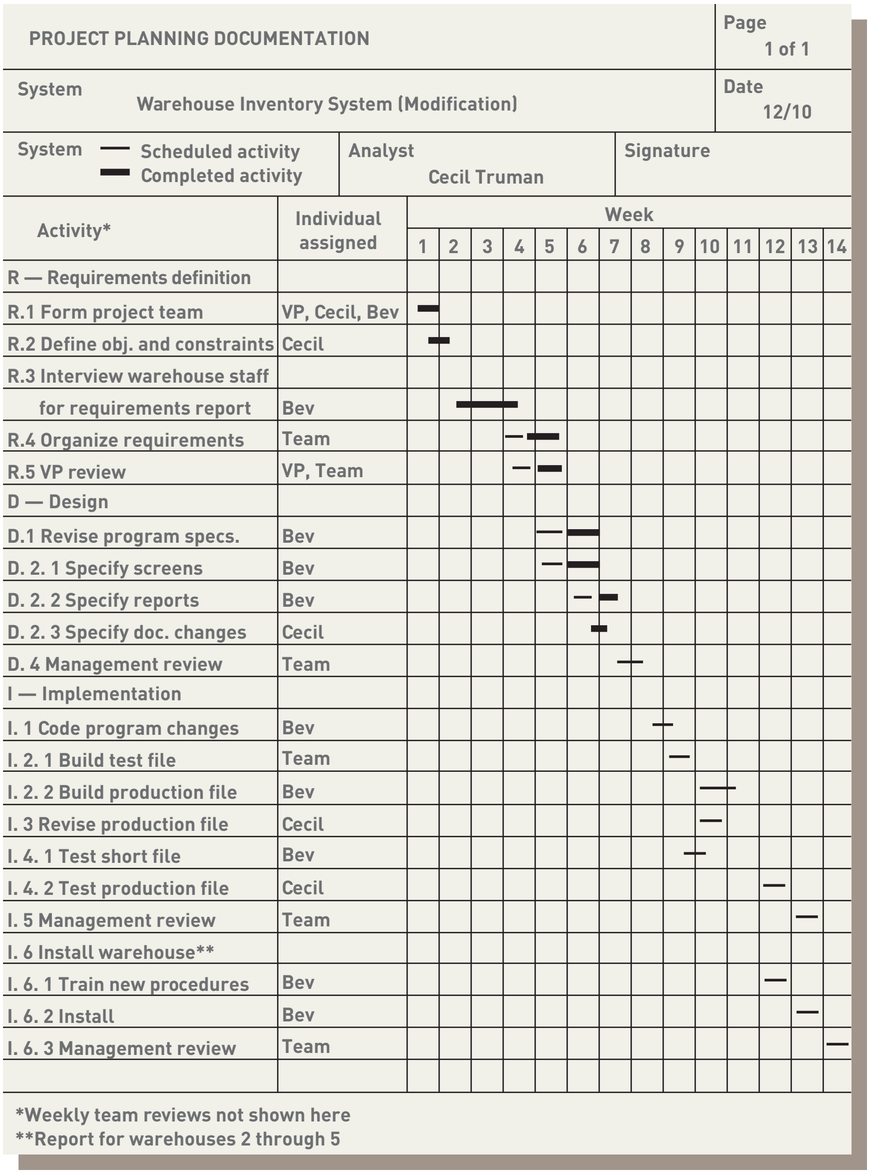
Gantt chart
A graphical tool used for planning, monitoring, and coordinating projects.

Both PERT and Gantt techniques can be automated using project management software. Several project management software packages are identified in Table 12.6. This software monitors all project activities and determines whether activities and the entire project are on time and within budget. Project management software also has workgroup capabilities to handle multiple projects and to allow a team to interact with the same software. Project management software helps managers determine the best way to reduce project completion

Figure 12.11

Sample Gantt Chart

A Gantt chart shows progress through systems development activities by putting a bar through appropriate cells.



PROJECT PLANNING DOCUMENTATION

Page 1 of 1

System Warehouse Inventory System (Modification)

Date 12/10

System — Scheduled activity ▬ Completed activity

Analyst Cecil Truman

Signature

Activity*	Individual assigned	1	2	3	4	5	6	7	8	9	10	11	12	13	14
R — Requirements definition															
R.1 Form project team	VP, Cecil, Bev	▬													
R.2 Define obj. and constraints	Cecil	▬	▬												
R.3 Interview warehouse staff for requirements report	Bev		▬	▬	▬										
R.4 Organize requirements	Team				—	▬									
R.5 VP review	VP, Team				—	▬									
D — Design															
D.1 Revise program specs.	Bev					—	▬								
D. 2. 1 Specify screens	Bev					—	▬								
D. 2. 2 Specify reports	Bev						—	▬							
D. 2. 3 Specify doc. changes	Cecil						▬	▬							
D. 4 Management review	Team							—	—						
I — Implementation															
I. 1 Code program changes	Bev								—	—					
I. 2. 1 Build test file	Team									—					
I. 2. 2 Build production file	Bev										—	—			
I. 3 Revise production file	Cecil										—				
I. 4. 1 Test short file	Bev									—	—				
I. 4. 2 Test production file	Cecil												—		
I. 5 Management review	Team													—	
I. 6 Install warehouse**															
I. 6. 1 Train new procedures	Bev												—		
I. 6. 2 Install	Bev													—	
I. 6. 3 Management review	Team														—

*Weekly team reviews not shown here

**Report for warehouses 2 through 5

Table 12.6

Selected Project Management Software

Software	Vendor
AboutTime	NetSQL Partners *(www.netsql.com)*
OpenPlan	Welcom *(www.welcom.com)*
Microsoft Project	Microsoft *(www.microsoft.com)*
Unifier	Skire *(www.skire.com)*
Project Scheduler	Scitor *(www.scitor.com)*

time at the least cost. In what some people believe is the largest private construction project in the United States, MGM Mirage and others used project management software to help them embark on an ambitious $8 billion construction project on 76 acres with over 4,000 hotel rooms, retail spaces, and other developments.[61] To complete the project, managers selected Skire's Unifier (*www.skire.com*), a powerful and flexible project management software package. The project management software should save the developer a substantial amount of money. According to a company spokesperson, "Any incremental improvement can go right to the bottom line when you're building on 76 acres of prime Las Vegas Strip real estate valued at $30 million an acre."

Use of Computer-Aided Software Engineering (CASE) Tools

Computer-aided software engineering (**CASE**) tools automate many of the tasks required in a systems development effort and encourage adherence to the SDLC, thus instilling a high degree of rigor and standardization to the entire systems development process. Prover Technology has developed a CASE tool that searches for programming bugs. The CASE tool searches for all possible design scenarios to make sure that the program is error free. Other CASE tools include Visible Systems (*www.visible.com*), Popkin Software (*www.popkin.com*), Rational Rose (part of IBM), and Visio, a charting and graphics program from Microsoft. Companies that produce CASE tools include Accenture, Microsoft, and Oracle. Oracle Designer and Developer CASE tools, for example, can help systems analysts automate and simplify the development process for database systems. See Table 12.7 for a list of CASE tools and their providers. The advantages and disadvantages of CASE tools are listed in Table 12.8. CASE tools that focus on activities associated with the early stages of systems development are often called *upper-CASE* tools. These packages provide automated tools to assist with systems investigation, analysis, and design activities. Other CASE packages, called *lower-CASE* tools, focus on the later implementation stage of systems development, and can automatically generate structured program code.

computer-aided software engineering (CASE)
Tools that automate many of the tasks required in a systems development effort and encourage adherence to the SDLC.

Table 12.7

Typical CASE Tools

CASE Tool	Vendor
Oracle Designer	Oracle Corporation *www.oracle.com*
Visible Analyst	Visible Systems Corporation *www.visible.com*
Rational Rose	Rational Software *www.ibm.com*
Embarcadero Describe	Embarcadero Describe *www.embarcadero.com*

Table 12.8

Advantages and Disadvantages of CASE Tools

Advantages	Disadvantages
Produce systems with a longer effective operational life	Increase the initial costs of building and maintaining systems
Produce systems that more closely meet user needs and requirements	Require more extensive and accurate definition of user needs and requirements
Produce systems with excellent documentation	Can be difficult to customize
Produce systems that need less systems support	Require more training of maintenance staff
Produce more flexible systems	Can be difficult to use with existing systems

Object-Oriented Systems Development

The success of a systems development effort can depend on the specific programming tools and approaches used. As mentioned in Chapter 4, object-oriented (OO) programming languages allow the interaction of programming objects—that is, an object consists of both data and the actions that can be performed on the data. So, an object could be data about an employee and all the operations (such as payroll, benefits, and tax calculations) that might be performed on the data.

Developing programs and applications using OO programming languages involves constructing modules and parts that can be reused in other programming projects. DTE Energy, a $7 billion Detroit-based energy company, has set up a library of software components that can be reused by its programmers. Systems developers from the company reuse and contribute to software components in the library. DTE's developers meet frequently to discuss ideas, problems, and opportunities of using the library of reusable software components.

Chapter 4 discussed a number of programming languages that use the object-oriented approach, including Visual Basic, C++, and Java. These languages allow systems developers to take the OO approach, making program development faster and more efficient, resulting in lower costs. Modules can be developed internally or obtained from an external source. After a company has the programming modules, programmers and systems analysts can modify them and integrate them with other modules to form new programs.

object-oriented systems development (OOSD)
An approach to systems development that combines the logic of the systems development life cycle with the power of object-oriented modeling and programming.

Object-oriented systems development (OOSD) combines the logic of the systems development life cycle with the power of object-oriented modeling and programming. OOSD follows a defined systems development life cycle, much like the SDLC. The life cycle phases are usually completed with many iterations. Object-oriented systems development typically involves the following tasks:

- **Identifying potential problems and opportunities within the organization that would be appropriate for the OO approach.** This process is similar to traditional systems investigation. Ideally, these problems or opportunities should lend themselves to the development of programs that can be built by modifying existing programming modules.
- **Defining what kind of system users require.** This analysis means defining all the objects that are part of the user's work environment (object-oriented analysis). The OO team must study the business and build a model of the objects that are part of the business (such as a customer, an order, or a payment). Many of the CASE tools discussed in the previous section can be used, starting with this step of OOSD.
- **Designing the system.** This process defines all the objects in the system and the ways they interact (object-oriented design). Design involves developing logical and physical models of the new system by adding details to the object model started in analysis.
- **Programming or modifying modules.** This implementation step takes the object model begun during analysis and completed during design and turns it into a set of interacting objects in a system. Object-oriented programming languages are designed to allow the programmer to create classes of objects in the computer system that correspond to the objects in the actual business process. Objects such as customer, order, and payment are redefined as computer system objects—a customer screen, an order entry menu, or a dollar sign icon. Programmers then write new modules or modify existing ones to produce the desired programs.
- **Evaluation by users.** The initial implementation is evaluated by users and improved. Additional scenarios and objects are added, and the cycle repeats. Finally, a complete, tested, and approved system is available for use.
- **Periodic review and modification.** The completed and operational system is reviewed at regular intervals and modified as necessary.

SYSTEMS INVESTIGATION

As discussed earlier in the chapter, systems investigation is the first phase in the traditional SDLC of a new or modified business information system. The purpose is to identify potential problems and opportunities and consider them in light of the goals of the company. In general, systems investigation attempts to uncover answers to the following questions:

- What primary problems might a new or enhanced system solve?
- What opportunities might a new or enhanced system provide?
- What new hardware, software, databases, telecommunications, personnel, or procedures will improve an existing system or are required in a new system?
- What are the potential costs (variable and fixed)?
- What are the associated risks?

Initiating Systems Investigation

Because systems development requests can require considerable time and effort to implement, many organizations have adopted a formal procedure for initiating systems development, beginning with systems investigation. The **systems request form** is a document that is filled out by someone who wants the IS department to initiate systems investigation. This form typically includes the following information:

- Problems in or opportunities for the system
- Objectives of systems investigation
- Overview of the proposed system
- Expected costs and benefits of the proposed system

systems request form
A document filled out by someone who wants the IS department to initiate systems investigation.

The information in the systems request form helps to rationalize and prioritize the activities of the IS department. Based on the overall IS plan, the organization's needs and goals, and the estimated value and priority of the proposed projects, managers make decisions regarding the initiation of each systems investigation for such projects.

Participants in Systems Investigation

After a decision has been made to initiate systems investigation, the first step is to determine what members of the development team should participate in the investigation phase of the project. Members of the development team change from phase to phase (see Figure 12.12). The systems investigation team can be diverse, with members located around the world. When Nokia decided to develop a new cell phone, its investigation team members were from England, Finland, and the United States.[62] Cooperation and collaboration are keys to successful investigation teams.

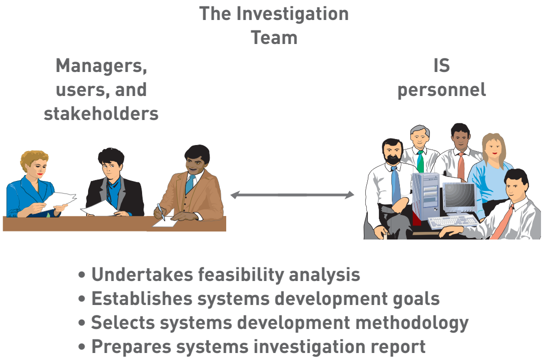

The Investigation Team
Managers, users, and stakeholders
IS personnel
• Undertakes feasibility analysis
• Establishes systems development goals
• Selects systems development methodology
• Prepares systems investigation report

Figure 12.12

The Systems Investigation Team

The team consists of upper- and middle-level managers, a project manager, IS personnel, users, and stakeholders.

In some cases, the participants in systems investigation are asked to step into big problems and fix them. When a new CIO at a major university was asked to investigate a security breach, he was a little apprehensive.[63] "I was literally walking into a river of alligators, but that's not always a bad thing. It can be a character building thing," he said.

Ideally, functional managers are heavily involved during the investigation phase. Other members could include users or stakeholders outside management, such as an employee who helped initiate systems development. The technical and financial expertise of others participating in investigation would help the team determine whether the problem is worth solving. The members of the development team who participate in investigation are then responsible for gathering and analyzing data, preparing a report justifying systems development, and presenting the results to top-level managers.

Feasibility Analysis

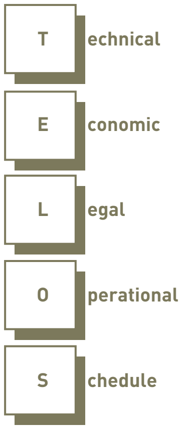
A key step of the systems investigation phase is **feasibility analysis**, which assesses technical, economic, legal, operational, and schedule feasibility (see Figure 12.13). **Technical feasibility** is concerned with whether the hardware, software, and other system components can be acquired or developed to solve the problem.

feasibility analysis
Assessment of the technical, economic, legal, operational, and schedule feasibility of a project.

technical feasibility
Assessment of whether the hardware, software, and other system components can be acquired or developed to solve the problem.

Figure 12.13

Technical, Economic, Legal, Operational, and Schedule Feasibility



economic feasibility
The determination of whether the project makes financial sense and whether predicted benefits offset the cost and time needed to obtain them.

Economic feasibility determines whether the project makes financial sense and whether predicted benefits offset the cost and time needed to obtain them. A securities company, for example, investigated the economic feasibility of sending research reports electronically instead of through the mail. Economic analysis revealed that the new approach could save the company up to $500,000 per year. Economic feasibility can involve cash flow analysis such as that done in net present value or internal rate of return (IRR) calculations.

net present value
The net amount by which project savings exceed project expenses after allowing for the cost of capital and the passage of time.

Net present value is an often-used approach for ranking competing projects and for determining economic feasibility. The net present value represents the net amount by which project savings exceed project expenses, after allowing for the cost of capital and the passage of time. The cost of capital is the average cost of funds used to finance the operations of the business. Net present value takes into account that a dollar returned at a later date is not worth as much as one received today, because the dollar in hand can be invested to earn profits or interest in the interim. Spreadsheet programs, such as Lotus and Microsoft Excel, have built-in functions to compute the net present value and internal rate of return.

legal feasibility
The determination of whether laws or regulations may prevent or limit a systems development project.

Legal feasibility determines whether laws or regulations can prevent or limit a systems development project. For example, a Web site that allowed users to share music without paying musicians or music producers was sued. Legal feasibility should have identified this vulnerability during the Web site development project. Legal feasibility involves an analysis of existing and future laws to determine the likelihood of legal action against the systems development project and the possible consequences.

operational feasibility
The measure of whether the project can be put into action or operation.

Operational feasibility is a measure of whether the project can be put into action or operation. It can include logistical and motivational (acceptance of change) considerations. Motivational considerations are important because new systems affect people and data flows and can have unintended consequences. As a result, power and politics might come into play, and some people might resist the new system. On the other hand, recall that a new system can help avoid major problems. For example, because of deadly hospital errors, a healthcare consortium looked into the operational feasibility of developing a new computerized physician order-entry system to require that all prescriptions and every order a doctor gives to staff be entered into the computer. The computer then checks for drug allergies and interactions between drugs. If operationally feasible, the new system could save lives and help avoid lawsuits.

schedule feasibility
The determination of whether the project can be completed in a reasonable amount of time.

Schedule feasibility determines whether the project can be completed in a reasonable amount of time—a process that involves balancing the time and resource requirements of the project with other projects.

Object-Oriented Systems Investigation

The object-oriented approach can be used during all phases of systems development, from investigation to maintenance and review. Consider a kayak rental business in Maui, Hawaii, where the owner wants to computerize its operations, including renting kayaks to customers and adding new kayaks into the rental program (see Figure 12.14). As you can see, the kayak rental clerk rents kayaks to customers and adds new kayaks to the current inventory available for rent. The stick figure is an example of an *actor*, and the ovals each represent an event, called a *use case*. In our example, the actor (the kayak rental clerk) interacts with two use cases (rent kayaks to customers and add new kayaks to inventory). The use case diagram is part of the Unified Modeling Language (UML) that is used in object-oriented systems development.

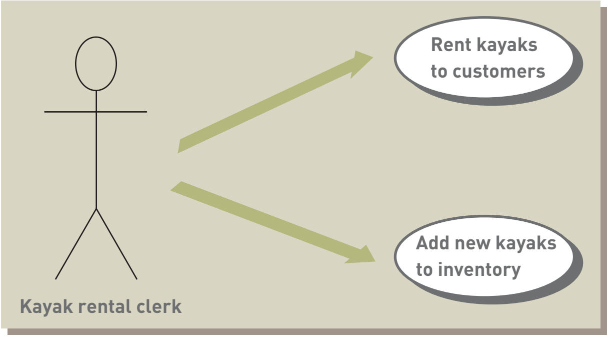


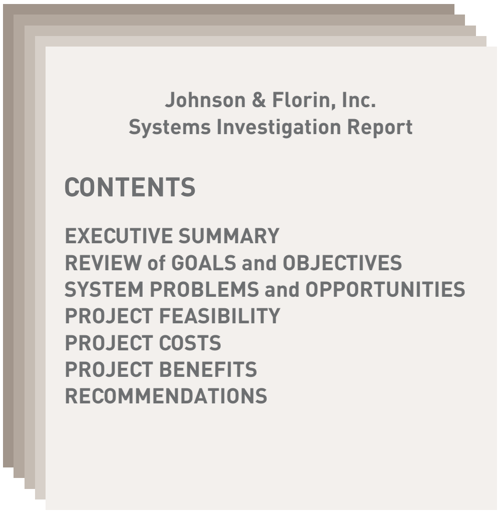
Figure 12.14

Use Case Diagram for a Kayak Rental Application

The Systems Investigation Report

The primary outcome of systems investigation is a **systems investigation report**, also called a *feasibility study*. This report summarizes the results of systems investigation and the process of feasibility analysis and recommends a course of action: continue on into systems analysis, modify the project in some manner, or drop it. A typical table of contents for the systems investigation report is shown in Figure 12.15.

systems investigation report
A summary of the results of the systems investigation and the process of feasibility analysis and recommendation of a course of action.



Johnson & Florin, Inc.
Systems Investigation Report

CONTENTS

EXECUTIVE SUMMARY
REVIEW of GOALS and OBJECTIVES
SYSTEM PROBLEMS and OPPORTUNITIES
PROJECT FEASIBILITY
PROJECT COSTS
PROJECT BENEFITS
RECOMMENDATIONS

Figure 12.15

A Typical Table of Contents for a Systems Investigation Report

The systems investigation report is reviewed by senior management, often organized as an advisory committee, or **steering committee**, consisting of senior management and users from the IS department and other functional areas. These people help IS personnel with their decisions about the use of information systems in the business and give authorization to pursue further systems development activities. After review, the steering committee might agree with the recommendation of the systems development team or suggest a change in project focus to concentrate more directly on meeting a specific company objective. Another alternative is that everyone might decide that the project is not feasible and cancel the project.

steering committee
An advisory group consisting of senior management and users from the IS department and other functional areas.

SYSTEMS ANALYSIS

After a project has been approved for further study, the next step is to answer the question "What must the information system do to solve the problem?" The process needs to go beyond mere computerization of existing systems. The entire system, and the business process with which it is associated, should be evaluated. Often, a firm can make great gains if it restructures both business activities and the related information system simultaneously. The overall emphasis of analysis is gathering data on the existing system, determining the requirements for the new system, considering alternatives within these constraints, and investigating the feasibility of the solutions. The primary outcome of systems analysis is a prioritized list of systems requirements. During its systems analysis phase, Mobius Management Systems (*www.mobius.com*), a company that manages databases and data resources for other companies, determined that the physical size of its data centers was an important systems requirement. Its current data centers were simply too large.[64] According to one IS administrator, "We were taking over what formerly were people's offices and making them data centers." The company analyzed the impact of replacing more than 100 of its hardware servers for software virtualization that allowed multiple applications to run on a single server, saving a tremendous amount of space.

General Considerations

Systems analysis starts by clarifying the overall goals of the organization and determining how the existing or proposed information system helps meet them. A manufacturing company, for example, might want to reduce the number of equipment breakdowns. This goal can be translated into one or more informational needs. One need might be to create and maintain an accurate list of each piece of equipment and a schedule for preventative maintenance. Another need might be a list of equipment failures and their causes.

Analysis of a small company's information system can be fairly straightforward. On the other hand, evaluating an existing information system for a large company can be a long, tedious process. As a result, large organizations evaluating a major information system normally follow a formalized analysis procedure, involving these steps:

1. Assembling the participants for systems analysis
2. Collecting appropriate data and requirements
3. Analyzing the data and requirements
4. Preparing a report on the existing system, new system requirements, and project priorities

Participants in Systems Analysis

The first step in formal analysis is to assemble a team to study the existing system. This group includes members of the original investigation team—from users and stakeholders to IS personnel and management. Most organizations usually allow key members of the development team not only to analyze the condition of the existing system but also to perform other aspects of systems development, such as design and implementation.

After the participants in systems analysis are assembled, this group develops a list of specific objectives and activities. A schedule for meeting the objectives and completing the specific activities is also devised, along with deadlines for each stage and a statement of the resources required at each stage, such as clerical personnel, supplies, and so forth. Major milestones are normally established to help the team monitor progress and determine whether problems or delays occur in performing systems analysis.

Data Collection

The purpose of data collection is to seek additional information about the problems or needs identified in the systems investigation report. During this process, the strengths and weaknesses of the existing system are emphasized.

Identifying Sources of Data

Data collection begins by identifying and locating the various sources of data, including both internal and external sources (see Figure 12.16).

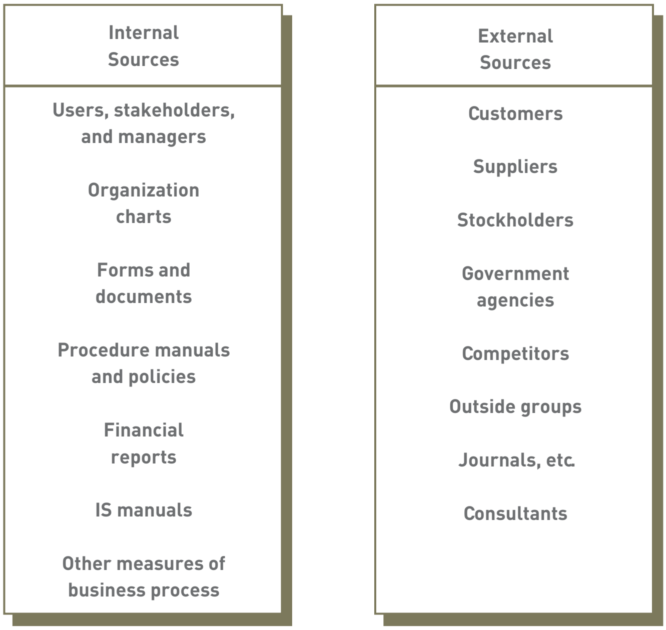


Figure 12.16

Internal and External Sources of Data for Systems Analysis

Collecting Data

After data sources have been identified, data collection begins. Figure 12.17 shows the steps involved. Data collection might require a number of tools and techniques, such as interviews, direct observation, and questionnaires.

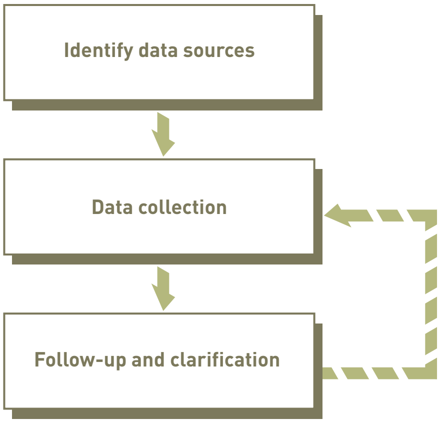


Figure 12.17

The Steps in Data Collection

Interviews can either be structured or unstructured. In a **structured interview**, the questions are written in advance. In an **unstructured interview**, the questions are not written in advance; the interviewer relies on experience in asking the best questions to uncover the inherent problems of the existing system. An advantage of the unstructured interview is that it allows the interviewer to ask follow-up or clarifying questions immediately.

With **direct observation**, one or more members of the analysis team directly observe the existing system in action. One of the best ways to understand how the existing system functions is to work with the users to discover how data flows in certain business tasks. Determining the data flow entails direct observation of users' work procedures, their reports, current screens (if automated already), and so on. From this observation, members of the analysis team determine which forms and procedures are adequate and which are inadequate

structured interview
An interview where the questions are written in advance.

unstructured interview
An interview where the questions are not written in advance.

direct observation
Watching the existing system in action by one or more members of the analysis team.

and need improvement. Direct observation requires a certain amount of skill. The observer must be able to see what is really happening and not be influenced by attitudes or feelings. This approach can reveal important problems and opportunities that would be difficult to obtain using other data collection methods. An example would be observing the work procedures, reports, and computer screens associated with an accounts payable system being considered for replacement.

Direct observation is a method of data collection. One or more members of the analysis team directly observe the existing system in action.

(Source: © Kriss Russell / iStockphoto.)

questionnaires
A method of gathering data when the data sources are spread over a wide geographic area.

When many data sources are spread over a wide geographic area, **questionnaires** might be the best method. Like interviews, questionnaires can be either structured or unstructured. In most cases, a pilot study is conducted to fine-tune the questionnaire. A follow-up questionnaire can also capture the opinions of those who do not respond to the original questionnaire.

statistical sampling
Selecting a random sample of data and applying the characteristics of the sample to the whole group.

Other data collection techniques can also be employed. In some cases, telephone calls are an excellent method. Activities can also be simulated to see how the existing system reacts. Thus, fake sales orders, stockouts, customer complaints, and data-flow bottlenecks can be created to see how the existing system responds to these situations. **Statistical sampling**, which involves taking a random sample of data, is another technique. For example, suppose that you want to collect data that describes 10,000 sales orders received over the last few years. Because it is too time consuming to analyze each of the sales orders, you can collect a random sample of 100 to 200 sales orders from the entire batch. You can assume that the characteristics of this sample apply to all 10,000 orders.

Data Analysis

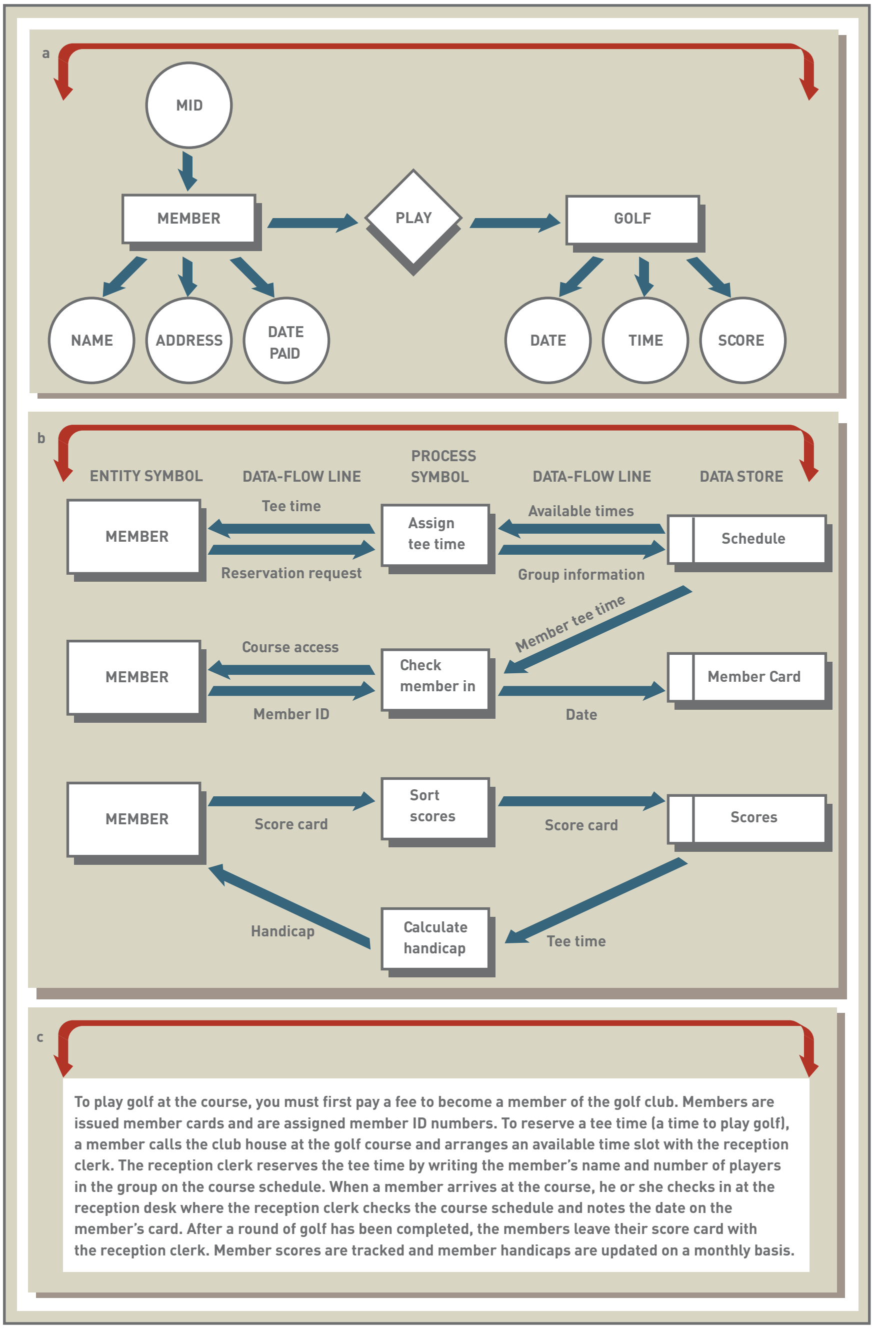
data analysis
The manipulation of collected data so that the development team members who are participating in systems analysis can use the data.

The data collected in its raw form is usually not adequate to determine the effectiveness of the existing system or the requirements for the new system. The next step is to manipulate the collected data so that the development team members who are participating in systems analysis can use the data. This manipulation is called **data analysis**. Data and activity modeling and using data-flow diagrams and entity-relationship diagrams are useful during data analysis to show data flows and the relationships among various objects, associations, and activities. Other common tools and techniques for data analysis include application flowcharts, grid charts, CASE tools, and the object-oriented approach.

Data Modeling

Data modeling, first introduced in Chapter 5, is a commonly accepted approach to modeling organizational objects and associations that employ both text and graphics. How data modeling is employed, however, is governed by the specific systems development methodology.

Data modeling is most often accomplished through the use of entity-relationship (ER) diagrams. Recall from Chapter 5 that an entity is a generalized representation of an object type—such as a class of people (employee), events (sales), things (desks), or places (city)—and that entities possess certain attributes. Objects can be related to other objects in many ways. An entity-relationship diagram, such as the one shown in Figure 12.18a, describes a number of objects and the ways they are associated. An ER diagram (or any other modeling tool) cannot by itself fully describe a business problem or solution because it lacks descriptions of the related activities. It is, however, a good place to start because it describes object types and attributes about which data might need to be collected for processing.



To play golf at the course, you must first pay a fee to become a member of the golf club. Members are issued member cards and are assigned member ID numbers. To reserve a tee time (a time to play golf), a member calls the club house at the golf course and arranges an available time slot with the reception clerk. The reception clerk reserves the tee time by writing the member's name and number of players in the group on the course schedule. When a member arrives at the course, he or she checks in at the reception desk where the reception clerk checks the course schedule and notes the date on the member's card. After a round of golf has been completed, the members leave their score card with the reception clerk. Member scores are tracked and member handicaps are updated on a monthly basis.

Figure 12.18

Data and Activity Modeling

(a) An entity-relationship diagram. (b) A data-flow diagram. (c) A semantic description of the business process.

(Source: G. Lawrence Sanders, *Data Modeling*, Boyd & Fraser Publishing, Danvers, MA: 1995.)

Activity Modeling

To fully describe a business problem or solution, the related objects, associations, and activities must be described. Activities in this sense are events or items that are necessary to fulfill the business relationship or that can be associated with the business relationship in a meaningful way.

data-flow diagram (DFD)
A model of objects, associations, and activities that describes how data can flow between and around various objects.

Activity modeling is often accomplished through the use of data-flow diagrams. A **data-flow diagram (DFD)** models objects, associations, and activities by describing how data can flow between and around various objects. DFDs work on the premise that every activity involves some communication, transference, or flow that can be described as a data element. DFDs describe the activities that fulfill a business relationship or accomplish a business task, not how these activities are to be performed. That is, DFDs show the logical sequence of associations and activities, not the physical processes. A system modeled with a DFD could operate manually or could be computer based; if computer based, the system could operate with a variety of technologies.

DFDs are easy to develop and easily understood by nontechnical people. Data-flow diagrams use four primary symbols, as illustrated in Figure 12.18b.

data-flow line
Arrows that show the direction of data element movement.

process symbol
Representation of a function that is performed.

entity symbol
Representation of either a source or destination of a data element.

data store
Representation of a storage location for data.

- **Data flow.** The **data-flow line** includes arrows that show the direction of data element movement.
- **Process symbol.** The **process symbol** reveals a function that is performed. Computing gross pay, entering a sales order, delivering merchandise, and printing a report are examples of functions that can be represented with a process symbol.
- **Entity symbol.** The **entity symbol** shows either the source or destination of the data element. An entity can be, for example, a customer who initiates a sales order, an employee who receives a paycheck, or a manager who receives a financial report.
- **Data store.** A **data store** reveals a storage location for data. A data store is any computerized or manual data storage location, including magnetic tape, disks, a filing cabinet, or a desk.

Comparing entity-relationship diagrams with data-flow diagrams provides insight into the concept of top-down design. Figure 12.18a and b show an entity-relationship diagram and a data-flow diagram for the same business relationship—namely, a member of a golf club playing golf. Figure 12.18c provides a brief description of the business relationship for clarification.

Application Flowcharts

application flowcharts
Diagrams that show relationships among applications or systems.

Application flowcharts show the relationships among applications or systems. Assume that a small business has collected data about its order processing, inventory control, invoicing, and marketing analysis applications. Management is thinking of modifying the inventory control application. The raw facts collected, however, do not help in determining how the applications are related to each other and the databases required for each. These relationships are established through data analysis with an application flowchart (see Figure 12.19). Using this tool for data analysis makes clear the relationships among the order processing functions.

In the simplified application flowchart in Figure 12.19, you can see that the telephone order clerk provides important data to the system about items such as versions, quantities, and prices. The system calculates sales tax and order totals. Any changes made to this order processing system could affect the company's other systems, such as inventory control and marketing.

Grid Charts

grid chart
A table that shows relationships among the various aspects of a systems development effort.

A **grid chart** is a table that shows relationships among various aspects of a systems development effort. For example, a grid chart can reveal the databases used by the various applications (see Figure 12.20).

The simplified grid chart in Figure 12.20 shows that the customer database is used by the order processing, marketing analysis, and invoicing applications. The inventory database is used by the order processing, inventory control, and marketing analysis applications. The supplier database is used by the inventory control application, and the accounts receivable database is used by the invoicing application. This grid chart shows which applications use common databases and reveals that, for example, any changes to the inventory control application must investigate the inventory and supplier databases.

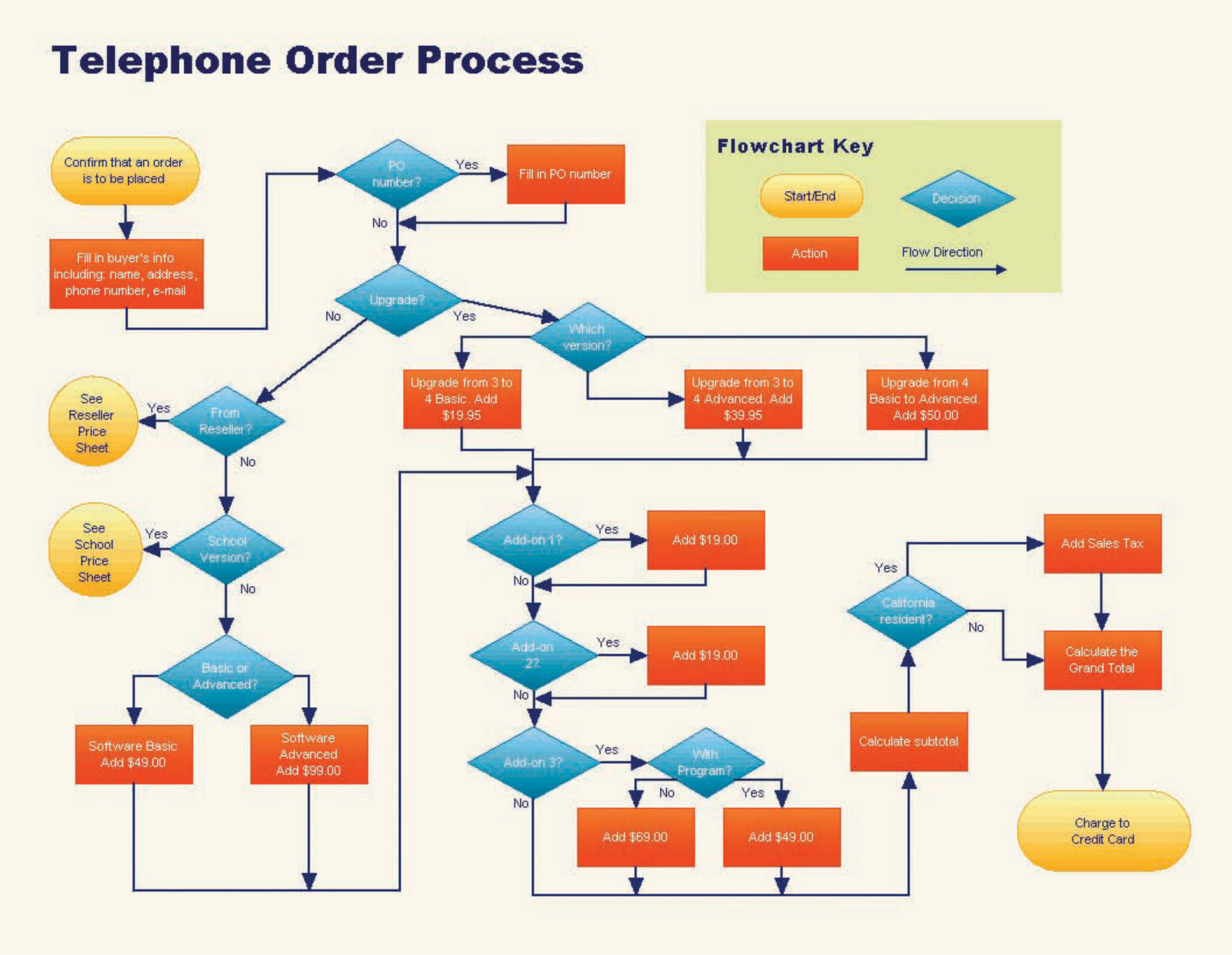


Figure 12.19

A Telephone Order Process Application Flowchart

The flowchart shows the relationships among various processes.

(Source: Courtesy of SmartDraw.com.)

Databases → Applications ↓	Customer database	Inventory database	Supplier database	Accounts receivable database
Order processing application	X	X		
Inventory control application		X	X	
Marketing analysis application	X	X		
Invoicing application	X			X

Figure 12.20

A Grid Chart

The chart shows the relationships among applications and databases.

CASE Tools

As discussed earlier, many systems development projects use CASE tools to complete analysis tasks. Most computer-aided software engineering tools have generalized graphics programs that can generate a variety of diagrams and figures. Entity-relationship diagrams, data-flow diagrams, application flowcharts, and other diagrams can be developed using CASE graphics programs to help describe the existing system. During the analysis phase, a **CASE repository**—a database of system descriptions, parameters, and objectives—will be developed.

CASE repository
A database of system descriptions, parameters, and objectives.

Requirements Analysis

requirements analysis
The determination of user, stakeholder, and organizational needs.

The overall purpose of **requirements analysis** is to determine user, stakeholder, and organizational needs.[65] For an accounts payable application, the stakeholders could include suppliers and members of the purchasing department. Questions that should be asked during requirements analysis include the following:

- Are these stakeholders satisfied with the current accounts payable application?
- What improvements could be made to satisfy suppliers and help the purchasing department?

One of the most difficult procedures in systems analysis is confirming user or systems requirements. In some cases, communications problems can interfere with determining these requirements. For example, an accounts payable manager might want a better procedure for tracking the amount owed by customers. Specifically, the manager wants a weekly report that shows all customers who owe more than $1,000 and are more than 90 days past due on their account. A financial manager might need a report that summarizes total amount owed by customers to consider whether to loosen or tighten credit limits. A sales manager might want to review the amount owed by a key customer relative to sales to that same customer. The purpose of requirements analysis is to capture these requests in detail. Numerous tools and techniques can be used to capture systems requirements. Often, various techniques are used in the context of a joint application development session.

Asking Directly

asking directly
An approach to gather data that asks users, stakeholders, and other managers about what they want and expect from the new or modified system.

One the most basic techniques used in requirements analysis is asking directly. **Asking directly** is an approach that asks users, stakeholders, and other managers about what they want and expect from the new or modified system. This approach works best for stable systems in which stakeholders and users clearly understand the system's functions. The role of the systems analyst during the analysis phase is to critically and creatively evaluate needs and define them clearly so that the systems can best meet them.

Critical Success Factors

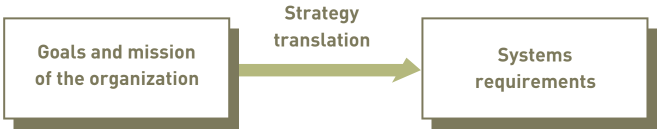
Another approach uses critical success factors (CSFs). As discussed earlier, managers and decision makers are asked to list only the factors that are critical to the success of their area of the organization. A CSF for a production manager might be adequate raw materials from suppliers; a CSF for a sales representative could be a list of customers currently buying a certain type of product. Starting from these CSFs, the system inputs, outputs, performance, and other specific requirements can be determined.

The IS Plan

As we have seen, the IS plan translates strategic and organizational goals into systems development initiatives. The IS planning process often generates strategic planning documents that can be used to define system requirements. Working from these documents ensures that requirements analysis will address the goals set by top-level managers and decision makers (see Figure 12.21). There are unique benefits to applying the IS plan to define systems requirements. Because the IS plan takes a long-range approach to using information technology within the organization, the requirements for a system analyzed in terms of the IS plan are more likely to be compatible with future systems development initiatives.

Figure 12.21

Converting Organizational Goals into Systems Requirements

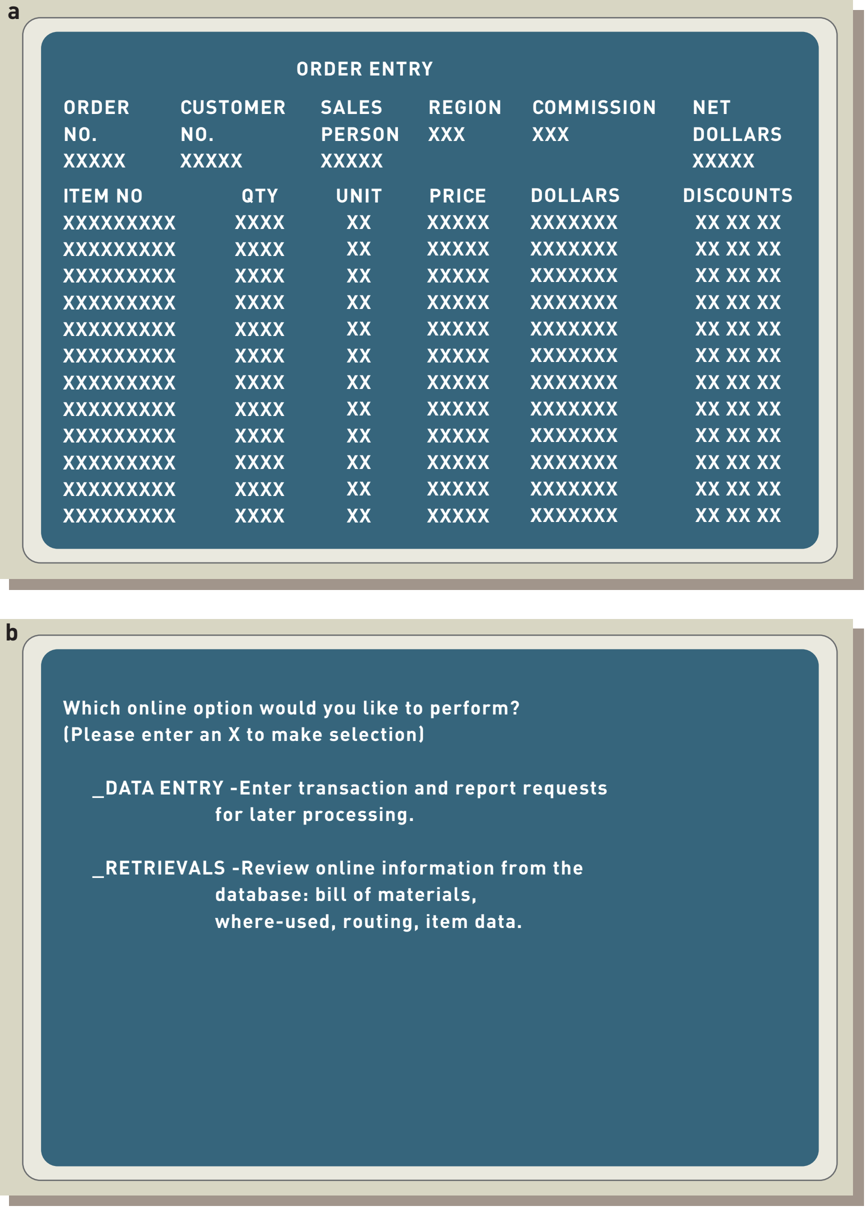


Screen and Report Layout

Developing formats for printed reports and screens to capture data and display information are some of the common tasks associated with developing systems. Screens and reports relating to systems output are specified first to verify that the desired solution is being delivered. Manual or computerized screen and report layout facilities are used to capture both output and input requirements.

Using a **screen layout**, a designer can quickly and efficiently design the features, layout, and format of a display screen. In general, users who interact with the screen frequently can be presented with more data and less descriptive information; infrequent users should have more descriptive information presented to explain the data that they are viewing (see Figure 12.22).

screen layout
A technique that allows a designer to quickly and efficiently design the features, layout, and format of a display screen.



```
                         ORDER ENTRY

ORDER       CUSTOMER    SALES    REGION   COMMISSION     NET
NO.         NO.         PERSON   XXX      XXX            DOLLARS
XXXXX       XXXXX       XXXXX                            XXXXX

ITEM NO          QTY      UNIT     PRICE    DOLLARS        DISCOUNTS
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
XXXXXXXXX        XXXX     XX       XXXXX    XXXXXXX        XX XX XX
```

```
Which online option would you like to perform?
(Please enter an X to make selection)

   _DATA ENTRY -Enter transaction and report requests
                for later processing.

   _RETRIEVALS -Review online information from the
                database: bill of materials,
                where-used, routing, item data.
```

Figure 12.22

Screen Layouts

(a) A screen layout chart for frequent users who require little descriptive information.

(b) A screen layout chart for infrequent users who require more descriptive information.

Report layout allows designers to diagram and format printed reports. Reports can contain data, graphs, or both. Graphic presentations allow managers and executives to quickly view trends and take appropriate action, if necessary.

report layout
A technique that allows designers to diagram and format printed reports.

Screen layout diagrams can document the screens users desire for the new or modified application. Report layout charts reveal the format and content of various reports that the application will prepare. Other diagrams and charts can be developed to reveal the relationship between the application and outputs from the application.

Requirements Analysis Tools

A number of tools can be used to document requirements analysis, including CASE tools. As requirements are developed and agreed on, entity-relationship diagrams, data-flow diagrams, screen and report layout forms, and other types of documentation are stored in the CASE repository. These requirements might also be used later as a reference during the rest of systems development or for a different systems development project.

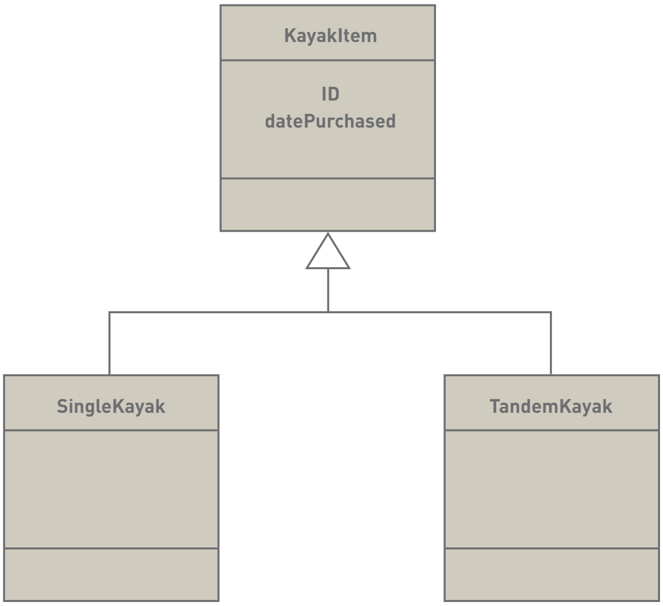
Object-Oriented Systems Analysis

The object-oriented approach can also be used during systems analysis. Like traditional analysis, problems or potential opportunities are identified during object-oriented analysis. Identifying key participants and collecting data is still performed. But instead of analyzing the existing system using data-flow diagrams and flowcharts, an object-oriented approach is used.

The section "Object-Oriented Systems Investigation" introduced a kayak rental example. A more detailed analysis of that business reveals that there are two classes of kayaks: single kayaks for one person and tandem kayaks that can accommodate two people. With the OO approach, a class is used to describe different types of objects, such as single and tandem kayaks. The classes of kayaks can be shown in a generalization/specialization hierarchy diagram (see Figure 12.23). KayakItem is an object that will store the kayak identification number (ID) and the date the kayak was purchased (datePurchased).

Figure 12.23

Generalization/Specialization Hierarchy Diagram for Single and Tandem Kayak Classes



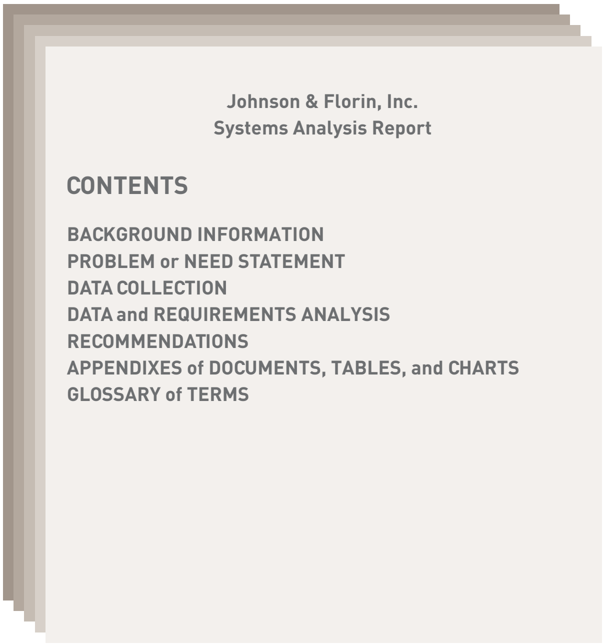
Of course, there could be subclasses of customers, life vests, paddles, and other items in the system. For example, price discounts for kayak rentals could be given to seniors (people over 65 years) and students. Thus, the Customer class could be divided into regular, senior, and student customer subclasses.

The Systems Analysis Report

Systems analysis concludes with a formal systems analysis report. It should cover the following elements:

- The strengths and weaknesses of the existing system from a stakeholder's perspective
- The user/stakeholder requirements for the new system (also called the *functional requirements*)
- The organizational requirements for the new system
- A description of what the new information system should do to solve the problem

Suppose analysis reveals that a marketing manager thinks a weakness of the existing system is its inability to provide accurate reports on product availability. These requirements and a preliminary list of the corporate objectives for the new system will be in the systems analysis report. Particular attention is placed on areas of the existing system that could be improved to meet user requirements. The table of contents for a typical report is shown in Figure 12.24.



Johnson & Florin, Inc.
Systems Analysis Report

CONTENTS

BACKGROUND INFORMATION
PROBLEM or NEED STATEMENT
DATA COLLECTION
DATA and REQUIREMENTS ANALYSIS
RECOMMENDATIONS
APPENDIXES of DOCUMENTS, TABLES, and CHARTS
GLOSSARY of TERMS

Figure 12.24

A Typical Table of Contents for a Report on an Existing System

The systems analysis report gives managers a good understanding of the problems and strengths of the existing system. If the existing system is operating better than expected or the necessary changes are too expensive relative to the benefits of a new or modified system, the systems development process can be stopped at this stage. If the report shows that changes to another part of the system might be the best solution, the development process might start over, beginning again with systems investigation. Or, if the systems analysis report shows that it will be beneficial to develop one or more new systems or to make changes to existing ones, systems design, which is discussed in Chapter 13, begins.

SUMMARY

Principle

Effective systems development requires a team effort from stakeholders, users, managers, systems development specialists, and various support personnel, and it starts with careful planning.

The systems development team consists of stakeholders, users, managers, systems development specialists, and various support personnel. The development team determines the objectives of the information system and delivers to the organization a system that meets its objectives.

Stakeholders are people who, either themselves or through the area of the organization they represent, ultimately benefit from the systems development project. Users are people who will interact with the system regularly. They can be employees, managers, customers, or suppliers. Managers on development teams are typically representative of stakeholders or can be stakeholders themselves. In addition, managers are most capable of initiating and maintaining change. For large-scale systems development projects, where the investment in and value of a system can be quite high, it is common to have senior-level managers be part of the development team.

A systems analyst is a professional who specializes in analyzing and designing business systems. The programmer is responsible for modifying or developing programs to satisfy user requirements. Other support personnel on the development team include technical specialists, either IS department employees or outside consultants. Depending on the magnitude of the systems development project and the number of IS development specialists on the team, the team might also include one or more IS managers. At some point in your career, you will likely be a participant in systems development. You could be involved in a systems development team—as a user, as a manager of a business area or project team, as a member of the IS department, or maybe even as a CIO.

Systems development projects are initiated for many reasons, including the need to solve problems with an existing system, to exploit opportunities to gain competitive advantage, to increase competition, to make use of effective information, to spur organizational growth, to settle a merger or corporate acquisition, and to address a change in the market or external environment. External pressures, such as potential lawsuits or terrorist attacks, can also prompt an organization to initiate systems development.

Information systems planning refers to the translation of strategic and organizational goals into systems development initiatives. Benefits of IS planning include a long-range view of information technology use and better use of IS resources. Planning requires developing overall IS objectives; identifying IS projects; setting priorities and selecting projects; analyzing resource requirements; setting schedules, milestones, and deadlines; and developing the IS planning document. IS planning can result in a competitive advantage through creative and critical analysis.

Establishing objectives for systems development is a key aspect of any successful development project. Critical success factors (CSFs) can identify important objectives. Systems development objectives can include performance goals (quality and usefulness of the output and the speed at which output is generated) and cost objectives (development costs, fixed costs, and ongoing investment costs).

Principle

Systems development often uses tools to select, implement, and monitor projects, including net present value (NPV), prototyping, rapid application development, CASE tools, and object-oriented development.

The five phases of the traditional SDLC are investigation, analysis, design, implementation, and maintenance and review. Systems investigation identifies potential problems and opportunities and considers them in light of organizational goals. Systems analysis seeks a general understanding of the solution required to solve the problem; the existing system is studied in detail and weaknesses are identified. Systems design creates new or modifies existing system requirements. Systems implementation encompasses programming, testing, training, conversion, and operation of the system. Systems maintenance and review entails monitoring the system and performing enhancements or repairs.

Advantages of the traditional SDLC include the following: It provides for maximum management control, creates considerable system documentation, ensures that system requirements can be traced back to stated business needs, and produces many intermediate products for review. Its disadvantages include the following: Users may get a system that meets the needs as understood by the developers, the documentation is expensive and difficult to maintain, users' needs go unstated or might not be met, and users cannot easily review the many intermediate products produced.

Prototyping is an iterative approach that involves defining the problem, building the initial version, having users work with and evaluate the initial version, providing feedback, and incorporating suggestions into the second version. Prototypes can be fully operational or nonoperational, depending on how critical the system under development is and how much time and money the organization has to spend on prototyping.

Rapid application development (RAD) uses tools and techniques designed to speed application development. Its use reduces paper-based documentation, automates program source code generation, and facilitates user participation in development activities. RAD can use newer programming techniques, such as agile development or extreme programming. RAD makes extensive use of the joint application development (JAD) process to gather data and perform requirements analysis. JAD involves group meetings in which users, stakeholders, and IS professionals work together to analyze existing systems, propose possible solutions, and define the requirements for a new or modified system.

The term *end-user systems development* describes any systems development project in which the primary effort is undertaken by a combination of business managers and users.

Many companies hire an outside consulting firm that specializes in systems development to take over some or all of its systems development activities. This approach is called *outsourcing*. Reasons for outsourcing include companies' belief that they can cut costs, achieve a competitive advantage without having the necessary IS personnel in-house, obtain state-of-the-art technology, increase their technological flexibility, and proceed with development despite downsizing. Many companies offer outsourcing services, including computer vendors and specialized consulting companies.

A number of factors affect systems development success. The degree of change introduced by the project, continuous improvement and reengineering, the use of quality programs and standards, organizational experience with systems development, the use of project management tools, and the use of CASE tools and the objected-oriented approach are all factors that affect the success of a project. The greater the amount of change a system will endure, the greater the degree of risk and often the amount of reward. Continuous improvement projects do not require significant business process or IS changes, while reengineering involves fundamental changes in how the organization conducts business and completes tasks. Successful systems development projects often involve such factors as support from top management, strong user involvement, use of a proven methodology, clear project goals and objectives, concentration on key problems and straightforward designs, staying on schedule and within budget, good user training, and solid review and maintenance programs. Quality standards, such as ISO 9001, can also be used during the systems development process.

The use of automated project management tools enables detailed development, tracking, and control of the project schedule. Effective use of a quality assurance process enables the project manager to deliver a high-quality system and to make intelligent trade-offs among cost, schedule, and quality. CASE tools automate many of the systems development tasks, thus reducing an analyst's time and effort while ensuring good documentation. Object-oriented systems development can also be an important success factor. With the object-oriented systems development (OOSD) approach, a project can be broken down into a group of objects that interact. Instead of requiring thousands or millions of lines of detailed computer instructions or code, the systems development project might require a few dozen or maybe a hundred objects.

Principle

Systems development starts with investigation and analysis of existing systems.

In most organizations, a systems request form initiates the investigation process. Participants in systems investigation can include stakeholders, users, managers, employees, analysts, and programmers. The systems investigation is designed to assess the feasibility of implementing solutions for business problems, including technical, economic, legal, operational, and schedule feasibility. Net present value analysis is often used to help determine a project's economic feasibility. An investigation team follows up on the request and performs a feasibility analysis that addresses technical, economic, legal, operational, and schedule feasibility.

If the project under investigation is feasible, major goals are set for the system's development, including performance, cost, managerial goals, and procedural goals. Many companies choose a popular methodology so that new IS employees, outside specialists, and vendors will be familiar with the systems development tasks set forth in the approach. A systems development methodology must be selected. Object-oriented systems investigation is being used to a greater extent today. The use case diagram is part of the Unified Modeling Language that is used to document object-oriented systems development. As a final step in the investigation process, a systems investigation report should be prepared to document relevant findings.

Systems analysis is the examination of existing systems, which begins after a team receives approval for further study from management. Additional study of a selected system allows those involved to further understand the system's weaknesses and potential areas for improvement. An analysis team is assembled to collect and analyze data on the existing system.

Data collection methods include observation, interviews, questionnaires, and statistical sampling. Data analysis manipulates the collected data to provide information. The analysis includes grid charts, application flowcharts, and CASE tools. The overall purpose of requirements analysis is to determine user and organizational needs.

Data analysis and modeling is used to model organizational objects and associations using text and graphical diagrams. It is most often accomplished through the use of entity-relationship (ER) diagrams. Activity modeling often employs data-flow diagrams (DFDs), which model objects, associations, and activities by describing how data can flow between and around various objects. DFDs use symbols for data flows, processing, entities, and data stores. Application flowcharts, grid charts, and CASE tools are also used during systems analysis.

Requirements analysis determines the needs of users, stakeholders, and the organization in general. Asking directly, using critical success factors, and determining requirements from the IS plan can be used. Often, screen and report layout charts are used to document requirements during systems analysis.

Like traditional analysis, problems or potential opportunities are identified during object-oriented analysis. Object-oriented systems analysis can involve using diagramming techniques, such as a generalization/specialization hierarchy diagram.

CHAPTER 12: SELF-ASSESSMENT TEST

Effective systems development requires a team effort from stakeholders, users, managers, systems development specialists, and various support personnel, and it starts with careful planning.

1. ___________________ is the activity of creating or modifying existing business systems. It refers to all aspects of the process—from identifying problems to be solved or opportunities to be exploited to the implementation and refinement of the chosen solution.
2. Which of the following people ultimately benefit from a systems development project?
   a. computer programmers
   b. systems analysts
   c. stakeholders
   d. senior-level manager
3. ________________ requires unbiased and careful questioning of whether systems elements are related in the most effective or efficient ways.
4. Like a contractor constructing a new building or renovating an existing one, the programmer takes the plans from the systems analyst and builds or modifies the necessary software. True or False?
5. The term ___________________ refers to the translation of strategic and organizational goals into systems development initiatives.
6. What involves investigating new approaches to existing problems?
   a. critical success factors
   b. systems analysis factors
   c. creative analysis
   d. critical analysis

Systems development often uses tools to select, implement, and monitor projects, including net present value (NPV), prototyping, rapid application development, CASE tools, and object-oriented development.

7. What employs tools, techniques, and methodologies designed to speed application development?
   a. rapid application development
   b. joint optimization
   c. prototyping
   d. extended application development
8. System performance is usually determined by factors such as fixed investments in hardware and related equipment. True or False?
9. _____________________ takes an iterative approach to the systems development process. During each iteration, requirements and alternative solutions to the problem are identified and analyzed, new solutions are designed, and a portion of the system is implemented.
10. Joint application development (JAD) employs tools, techniques, and methodologies designed to speed application development. True or False?
11. What consists of all activities that, if delayed, would delay the entire project?
    a. deadline activities
    b. slack activities
    c. RAD tasks
    d. the critical path

Systems development starts with investigation and analysis of existing systems.

12. The systems request form is a document that is filled out during systems analysis. True or False?
13. Feasibility analysis is typically done during which systems development stage?
    a. investigation
    b. analysis
    c. design
    d. implementation
14. Data modeling is most often accomplished through the use of ________________, whereas activity modeling is often accomplished through the use of __________________.
15. The overall purpose of requirements analysis is to determine user, stakeholder, and organizational needs. True or False?

CHAPTER 12: SELF-ASSESSMENT TEST ANSWERS

(1) Systems development (2) c (3) Critical analysis (4) True (5) information systems planning (6) c (7) a (8) False (9) Prototyping (10) False (11) d (12) False (13) a (14) entity-relationship (ER) diagrams, data-flow diagrams (15) True