CHAPTER

11

Automated Decision Systems and Expert Systems

LEARNING OBJECTIVES

- Understand the concept and applications of automated rule-based decision systems
- Understand the importance of knowledge in decision support
- Describe the concept and evolution of rule-based expert systems (ES)
- Understand the architecture of rule-based ES
- Learn the knowledge engineering process used to build ES
- Explain the benefits and limitations of rule-based systems for decision support
- Identify proper applications of ES
- Learn about tools and technologies for developing rule-based DSS

This chapter addresses two issues. First, how do some of the analytics technologies including predictive and optimization models get used in practice? In many cases, results of predictive models or even optimization models get simplified as rules that are then implemented in other applications. We call these automated decision systems. Second, in addition to the use of data and mathematical models, some managerial decisions require qualitative information and the judgmental knowledge that resides in the minds of human experts. Therefore, it is necessary to find effective ways to incorporate such information and knowledge into decision support systems (DSS). A system that integrates knowledge from experts is commonly called a knowledge-based decision support system (KBDSS) or an intelligent decision support system (IDSS). A KBDSS can enhance the capabilities of decision support not only by supplying a tool that directly supports a decision maker, but also by enhancing various computerized DSS environments. The foundation for building such systems is the techniques and tools that have been developed in the area of artificial intelligence—rule-based expert systems being the primary one. This chapter introduces the essentials of automated decision systems and provides a detailed description of expert systems.

11.1 OPENING VIGNETTE: InterContinental Hotel Group Uses Decision Rules for Optimal Hotel Room Rates

With 4,437 hotels and 647,161 rooms, InterContinental Hotel Group (IHG) is the world's largest hotel group in terms of number of rooms. About 85 percent of its hotels are franchised, 14 percent are managed, and 1 percent are owned directly by the InterContinental Hotel group. Some of the hotel brands that belong to this group are Holiday Inn, Holiday Express, Staybridge Suites, and Crowne Plaza. Revenue generated from their rooms amounts to around $20 billion. Before the optimization model was implemented, pricing decisions were made based on a complex myriad of variables, some of which were day of the week, seasonality, occupancy level, competition, and customer feedback. These decisions were made without the use of analytics. Price decisions were made with the assumption that demand was independent of the price charged for a room. This fundamental flaw worked well in normal economic conditions. However, when the hospitality industry suffered a decline in revenue, with the challenge posed by the widespread use of the Internet, which introduced multiple distribution channels, IHG started considering and exploring alternative and effective revenue generation methods. The main aim was to increase the revenue per available room (RevPAR).

METHODOLOGY/SOLUTION

IHG rolled out their retail price optimization system to help increase their RevPAR. The large number of hotels was a big challenge to this task. Pricing decisions numbered over 273 million (or 76,000 per hotel) per day. The project resulted in a change of their fundamental business flow. The final model included a demand forecast model, market response model, competitor rates model, and an optimization price model. For each hotel, a price response is calculated by the market response model based on historical data. Price and competitor rates were used to estimate the demand for rooms. The objective function used in this computation turned out to be nonlinear. The input data for the competitor rates model were derived from third-party sources. Decision variables used to determine the best rates for each hotel were based on factors like estimated demand, hotel capacity, current bookings, and prices being charged by competitors. IHG's price optimization system is packaged in a Web application called PERFORM^{sm}.

RESULTS/BENEFITS

There has been widespread adoption of the retail price optimization model by hotel managers globally. PERFORM is used by over 4,000 users worldwide. The retail price optimization model was tested in a couple of IHG's hotels and the results were compared with hotels where the model had not been implemented yet. It was recognized that there was a 2.7 percent increase in RevPAR for hotels where the optimization model had been implemented.

QUESTIONS FOR THE OPENING VIGNETTE

1. Describe the challenges faced by IHG during development of their retail price optimization system.
2. Besides the hotel business in the hospitality industry, explain at least three other areas where an optimization model could be used.
3. What other methods could be used to solve IHG's price optimization problem?

WHAT WE CAN LEARN FROM THIS VIGNETTE

IHG has been doing business using manual price optimization methods for a long time and it seems to have worked for them. However, sometimes business environments change, which renders existing methods of running a business obsolete. IHG used data analytics and mathematical optimization methods to revolutionize revenue management. The price optimization model was a combination of different operations research methods. What is also important is that such decisions are eventually implemented using a decision system that is available to each client hotel. They do not have to know anything about the underlying decision methods to be able to use the recommendations made by the system.

Source: Dev Koushik, Jon A. Higbie, and Craig Eister, "Retail Price Optimization at InterContinental Hotels Group." *Interfaces,* Vol. 42, No. 1, 2012, pp. 45–57.

11.2 AUTOMATED DECISION SYSTEMS

A relatively new approach to supporting decision making is called **automated decision systems (ADS)**, sometimes also known as **decision automation systems** (DAS; see Davenport and Harris, 2005). An ADS is a rule-based system that provides a solution, usually in one functional area (e.g., finance, manufacturing), to a specific repetitive managerial problem, usually in one industry (e.g., to approve or not to approve a request for a loan, to determine the price of an item in a store).

Application Case 11.1 shows an example of applying automated decision systems to a problem that every organization faces—how to price its products or services. In contrast with management science approaches, which provide a model-based solution to generic structured problems (e.g., resource allocation, inventory level determination), ADS provide rule-based solutions. The following are examples of business rules: "If only 70 percent of the seats on a flight from Los Angeles to New York are sold 3 days prior to departure, offer a discount of x to nonbusiness travelers," "If an applicant owns a house and makes over $100,000 a year, offer a $10,000 credit line," and "If an item costs more than $2,000, and if your company buys it only once a year, the purchasing agent does not need special approval." Such rules, which are based on experience or derived through data mining, can be combined with mathematical models to form solutions that can be automatically and instantly applied to problems (e.g., "Based on the information provided and subject to verification, you will be admitted to our university"), or they can be provided to a human, who will make the final decision (see Figure 11.1). ADS attempt to automate highly repetitive decisions (in order to justify the computerization cost), based on business rules. ADS are mostly suitable for frontline employees who can see

Application Case 11.1

Giant Food Stores Prices the Entire Store

Giant Food Stores, LLC, a regional U.S. supermarket chain based in Carlisle, Pennsylvania, had a narrow Every Day Low Price strategy that it applied to most of the products in its stores. The company had a 30-year-old pricing and promotion system that was very labor intensive and that could no longer keep up with the pricing decisions required in the fast-paced grocery market. The system also limited the company's ability to execute more sophisticated pricing strategies.

Giant was interested in executing its pricing strategy more consistently based on a definitive set of pricing rules (pricing rules in retail might include relationships between national brands and private-label brands, relationships between sizes, ending digits such as "9," etc.). In the past, many of the rules were kept on paper, others were kept in people's heads, and some were not documented well enough for others to understand and ensure continuity. The company also had no means of reliably forecasting the impact of rule changes before prices hit the store shelves.

Giant Foods worked with DemandTec to deploy a system for its pricing decisions. The system is able to handle massive amounts of point-of-sale and competitive data to model and forecast consumer demand, as well as automate and streamline complex rules-based pricing schemes. It can handle large numbers of price changes, and it can do so without increasing staff. The system allows Giant Foods to codify pricing rules with "natural language" sentences rather than having to go through a technician. The system also has forecasting capabilities. These capabilities allow Giant Foods to predict the impact of pricing changes and new promotions before they hit the shelves. Giant Foods decided to implement the system for the entire store chain.

The system has allowed Giant Foods to become more agile in its pricing. It is now able to react to competitive pricing changes or vendor cost changes on a weekly basis rather than when resources become available. Giant's productivity has doubled because it no longer has to increase staff for pricing changes. Giant now focuses on "maintaining profitability while satisfying its customer and maintaining its price image."

Source: "Giant Food Stores Prices the Entire Store with DemandTec," DemandTec, **https://mydt.demandtec.com/mydemandtec/c/document_library/get_file?uuid=3151a5e4-f3e1-413e-9cd7-333289eeb3d5&groupId=264319** (accessed February 2013).

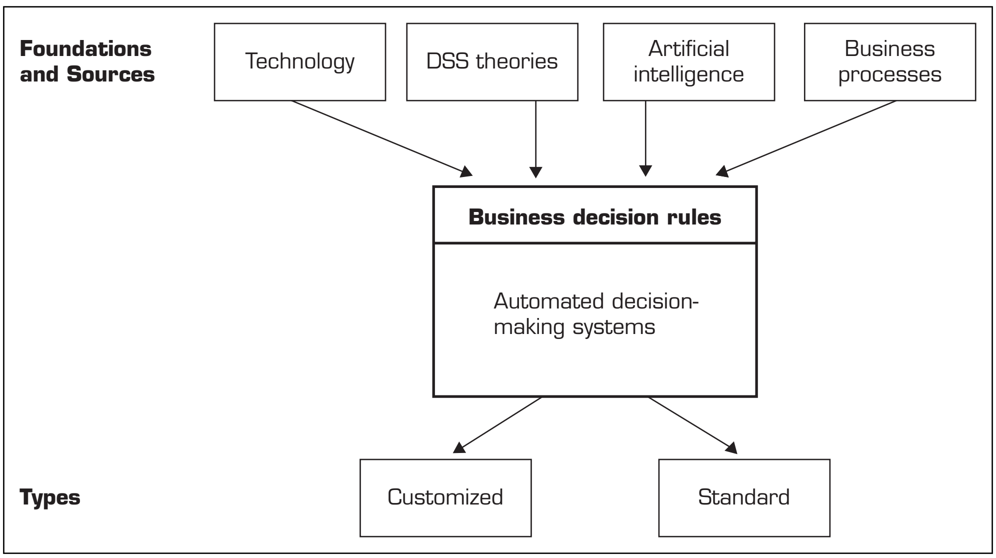

FIGURE 11.1 General Architecture of Automated Decision Systems.

the customer information online and frequently must make quick decisions. Davenport and Harris (2005) provide a good introduction to such systems. Dan Power started a Web site **decisionautomation.com** to compile information on such systems. He argues that decision automation systems are really not decision support systems. These types of systems make the decisions in real time or near–real time. Systems that provide credit approval decisions for loan approvals, or quote fares for the next airline flight reservation for a particular flight request, or deal with any other pricing issues are the most common examples of such systems. Application Case 11.1 illustrates use of such systems at many retailers that use such technologies. **Demandtec.com** is now an IBM Company.

ADS initially appeared in the airline industry, where they were called *revenue* (or *yield*) *management* (or revenue optimization) systems. Airlines use these systems to dynamically price tickets based on actual demand. Today, many service industries use similar pricing models.

The building blocks of such intelligent systems are business rules. A business rule can be as simple as saying—"Offer a discount if average sales drop by 10 percent." Once a set of rules are in place governing a decision, a model is built and implemented that is capable of making decisions autonomously. This removes the need for human intervention in making decisions. There might be some external factors or parameters that can cause the model to fail. However, advances in artificial intelligence have led to the creation of adaptable models, capable of adjusting to changes in external parameters.

To put this in perspective, almost all airlines have automated decision systems to assign dynamic prices based on demand. If it were left to a human being to analyze trends and travel patterns to generate prices, it would probably take a very long time, and the price would not cater to the trend.

All airlines have three major information systems closely integrated together: (1) pricing and accounting systems, (2) aircraft scheduling systems, and (3) inventory management systems. To manage this whole ecosystem, skilled individuals are hired possessing skillsets varying from operations management to business analytics to data warehousing. These individuals are assigned one of the most important tasks in running a successful airline—revenue management.

How does an airline go from processing inventory data, airport schedules, and customer demand to aircraft selection/scheduling and boarding? Revenue management planners (RM Planners) play a key role in running this whole process of keeping the airline profitable, while offering the best possible prices and customer service. The most important entity in this process is the customer. In an ideal situation, prices and schedules are driven by customer demand. However, forecasting customer demand is an extremely complex process involving thousands of possibilities. As a general practice, past demand is used to predict future demand, which is not always accurate. With the growth of low-cost carriers (LCCs), booking patterns change drastically from time to time. Customers are able to purchase tickets at low prices, within a week of departure, which creates a forecasting nightmare. The airlines use average demand levels to decide prices and make adjustments whenever necessary.

Recognizing the nature of a customer is extremely important to an airline. Leisure passengers are price sensitive but are willing to adapt to a flexible schedule for a lower price. These passengers are willing to commit to a reservation in advance. Business passengers, on the other hand, are extremely time sensitive and are willing to pay a higher price to get to the destination on time. Also, they are not likely to commit to a reservation in advance, and they prefer last-minute availability. Moreover, business passengers do not make purchases themselves, and they have their company pay for the trip. Both these types of customers have different needs and preferences. Catering to their specific preferences separately is extremely important.

Recognizing customer segments is extremely useful in determining the total revenue generated by a flight. Fares for leisure and business passengers are considerably different. As an example, if fares for business passengers are in the \$200–\$350 range, fares for leisure passengers are in the \$90–\$150 range. This distinction creates a need to balance the seats sold for both these classes for a certain flight to be profitable. Generally, revenue maximization is done over a network of airports operated by the airline, and not for a single flight. The objective is to generate overall revenues exceeding the minimum standards set by the airline. Empty seats on a flight are equivalent to lost revenue. Seats on a flight may be left empty due to a variety of reasons: last-minute cancellations, late arrival, and multiple bookings due to uncertainty of travel plans. As a precaution, most flights are overbooked, based on historical demand economics and human behavior. This is done to minimize lost revenue and allows more passengers to book their preferred flight. But overbooking can also lead to passengers being denied boarding, which creates ill-will. Airlines often try to compensate with attractive incentives to customers in exchange for moving to an alternate flight.

All of this process is done with the help of the revenue management system with inputs from RM Planners. Figure 11.2 presents a general architecture of such airline **revenue management systems**. This figure has been developed by Dr. Mukund Shankar, an airline revenue management specialist who has worked for/with several airlines in developing such systems. The pricing and accounting system handles ticket data, published fares, and pricing rules. The aircraft scheduling system handles flight schedules based on customer demand; finally, the inventory management system handles bookings, cancellations, and changes in departure data.

Customer demand is estimated at various price levels before prices are published. This accommodates seasonal- and weekday-based changes in customer demand. However, prices can be adjusted on-the-fly to reactively cater to changes in demand. Simultaneously, a change in the demand forecast calls for changes in aircraft scheduling

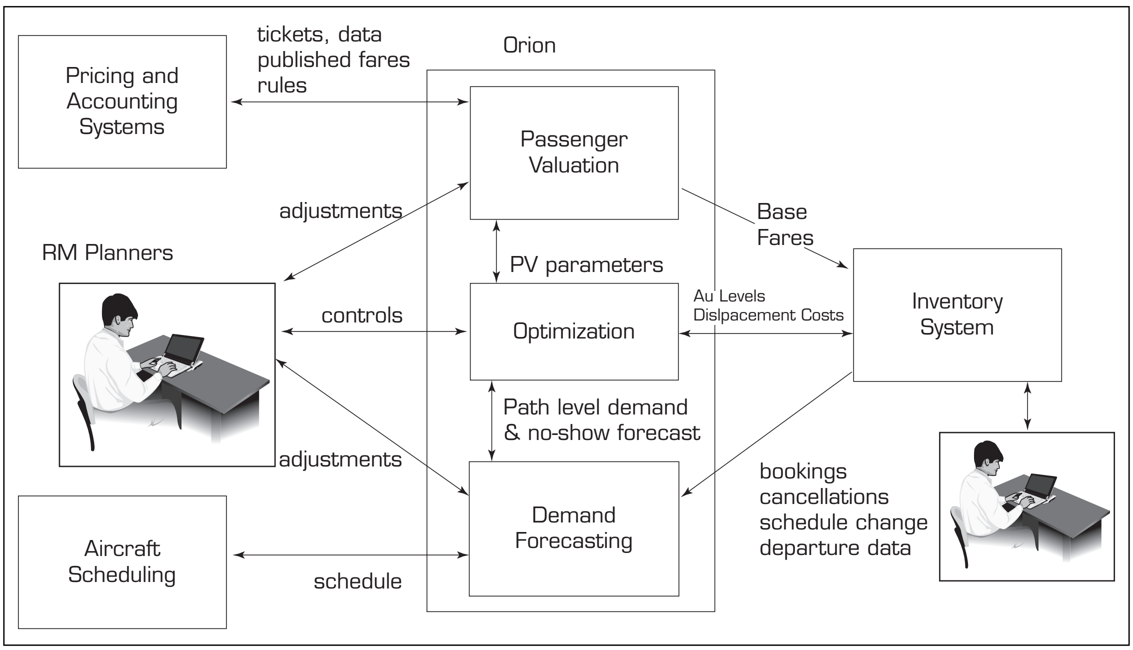

FIGURE 11.2 Architecture of Airline Revenue Management Systems. Courtesy: Mukund Shankar.

as empty or overbooked flights are a challenge to handle, and come at an expense of lost revenue. Moreover, optimization is done at each price level to operate flights at the lowest possible cost. This means that the whole information ecosystem of the airline needs to be extremely versatile, if it is to maximize profits. However, there is a limitation to how flexible this system can get as any airline has a limited inventory of flights at their disposal. Extreme demand cannot be accommodated with a limited fleet and often leads to losing customers to other airlines.

Because of the complexity involved in managing airlines, they invest heavily in building better predictive models for forecasting demand; analyzing the customer base more carefully to generate better segments of customers; operations research for optimizing flight routes, demand, and scheduling; and extremely fast hardware to process all this information as fast as possible. Virtually every major airline uses such automated decision systems. What is important to recognize is that similar systems also exist and are in use in many other industries. Although our examples are from business decision making, similar systems exist in engineering applications as well. For example, smart grid depends upon automated decisions to trigger specific activities whenever supply and demand of electricity demands switching of electricity generation or distributions. As we will discuss in the last chapter, we believe that these types of systems present a major future entrepreneurial opportunity in the consumer sector.

We next turn our attention to another class of **rule-based systems** that have been popular and in use since the mid-1980s. These systems have their roots in artificial intelligence, so we will first do an extremely quick overview of artificial intelligence.

SECTION 11.2 REVIEW QUESTIONS

1. Define *decision automation systems.*
2. What are the key components of a decision automation system?
3. Which industries are big users of decision automation systems?
4. How could decision automation systems assist consumers?

11.3 THE ARTIFICIAL INTELLIGENCE FIELD

Artificial intelligence (AI) is a collection of concepts and ideas that are related to the development of intelligent systems. These concepts and ideas may be developed in different areas and be applied to different domains. In order to understand the scope of AI, therefore, we need to see a group of areas that may be called the AI family. Figure 11.3 shows the major branches of AI applications. These applications are built on the foundation of many disciplines and technologies, including computer science, philosophy, electrical engineering, management science, psychology, and linguistics. Artificial intelligence (AI) is an area of computer science. Even though the term has many different definitions, most experts agree that AI is concerned with two basic ideas: (1) the study of human thought processes (to understand what intelligence is) and (2) the representation and duplication of those thought processes in machines (e.g., computers, robots).

One well-publicized, classic definition of AI is "behavior by a machine that, if performed by a human being, would be called intelligent." Rich and Knight (1991) provided a thought-provoking definition: "Artificial intelligence is the study of how to make computers do things at which, at the moment, people are better." To understand what artificial intelligence is, we need to examine those abilities that are considered to be signs of intelligence:

- Learning or understanding from experience
- Making sense out of ambiguous or contradictory messages

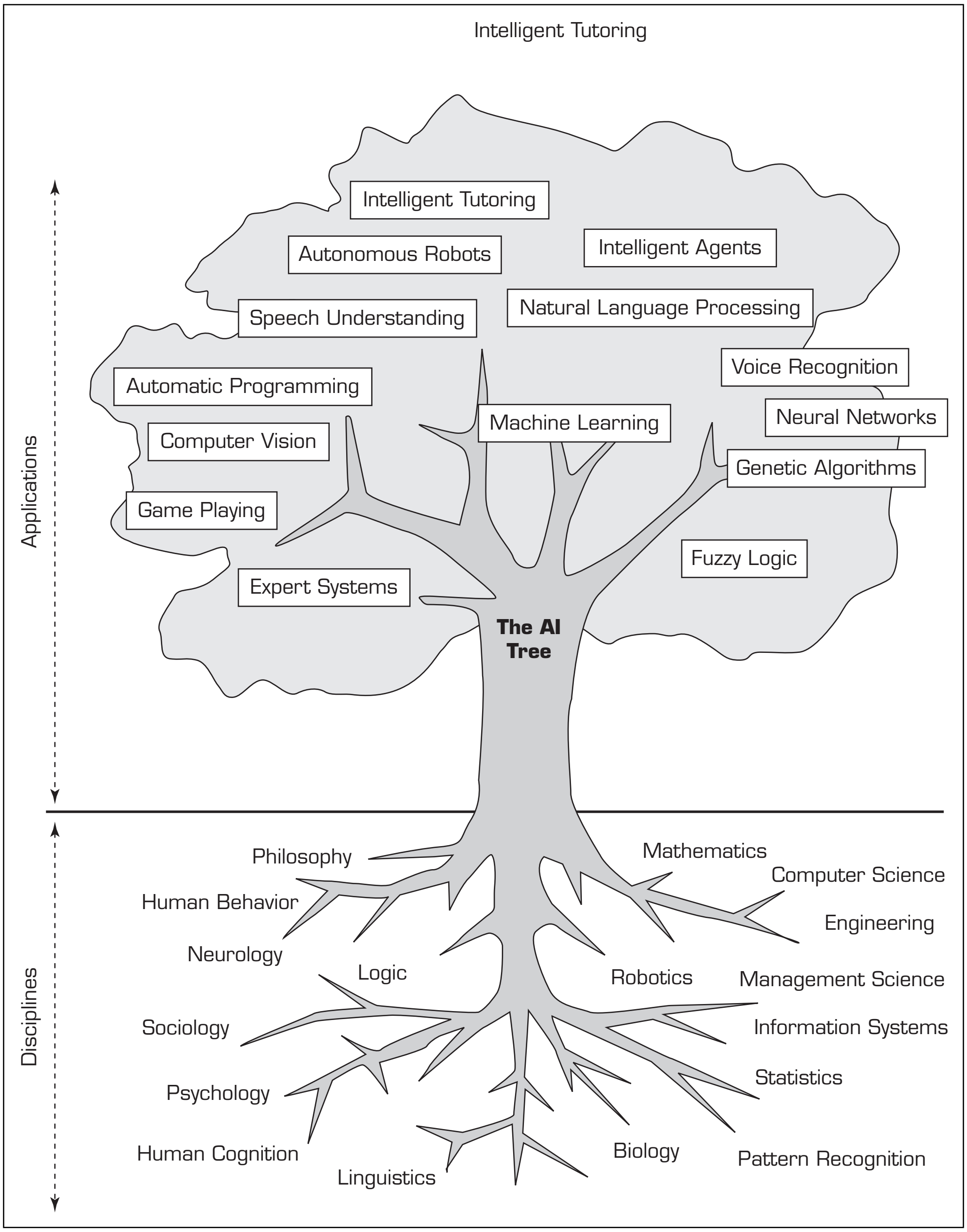

FIGURE 11.3 The Disciplines and Applications of AI.

- Responding quickly and successfully to a new situation (i.e., different responses, flexibility)
- Using reasoning in solving problems and directing conduct effectively
- Dealing with perplexing situations
- Understanding and inferring in a rational way
- Applying knowledge to manipulate the environment
- Thinking and reasoning
- Recognizing and judging the relative importance of different elements in a situation

Alan Turing designed an interesting test to determine whether a computer exhibits intelligent behavior; the test is called the *Turing test*. According to this test, a computer can be considered smart only when a human interviewer cannot identify the computer while conversing with both an unseen human being and an unseen computer.

As Figure 11.3 shows, there are many application areas of artificial intelligence. We will only focus on the expert systems because these relate to the automated rule-based decision systems described in the previous section.

SECTION 11.3 REVIEW QUESTIONS

1. What is the definition of artificial intelligence?
2. What are some of the major applications areas of artificial intelligence?
3. Identify some key characteristics of AI.

11.4 BASIC CONCEPTS OF EXPERT SYSTEMS

Expert systems (ES) are computer-based information systems that use expert knowledge to attain high-level decision performance in a narrowly defined problem domain. MYCIN, developed at Stanford University in the early 1980s for medical diagnosis, is the most well-known ES application. ES has also been used in taxation, credit analysis, equipment maintenance, help desk automation, environmental monitoring, and fault diagnosis. ES have been popular in large and medium-sized organizations as a sophisticated tool for improving productivity and quality.

The basic concepts of ES include how to determine who experts are, the definition of expertise, how expertise can be extracted and transferred from a person to a computer, and how the expert system should mimic the reasoning process of human experts. We describe these concepts in the following sections.

Experts

An **expert** is a person who has the special knowledge, judgment, experience, and skills to put his or her knowledge in action to provide sound advice and to solve complex problems in a narrowly defined area. It is an expert's job to provide knowledge about how he or she performs a task that a KBS will perform. An expert knows which facts are important and also understands and explains the dependency relationships among those facts. In diagnosing a problem with an automobile's electrical system, for example, an expert mechanic knows that a broken fan belt can be the cause for the battery to discharge.

There is no standard definition of *expert*, but decision performance and the level of knowledge a person has are typical criteria used to determine whether a particular person is an expert. Typically, experts must be able to solve a problem and achieve a performance level that is significantly better than average. In addition, experts are relative (and not absolute). An expert at a time or in a region may not be an expert in another time or region. For example, an attorney in New York may not be a legal expert in Beijing, China. A medical student may be an expert compared to the general public but may not be considered an expert in brain surgery. Experts have expertise that can help solve problems and explain certain obscure phenomena within a specific problem domain. Typically, human experts are capable of doing the following:

- Recognizing and formulating a problem
- Solving a problem quickly and correctly
- Explaining a solution

- Learning from experience
- Restructuring knowledge
- Breaking rules (i.e., going outside the general norms), if necessary
- Determining relevance and associations
- Declining gracefully (i.e., being aware of one's limitations)

Expertise

Expertise is the extensive, task-specific knowledge that experts possess. The level of expertise determines the performance of a decision. Expertise is often acquired through training, reading, and experience in practice. It includes explicit knowledge, such as theories learned from a textbook or in a classroom, and implicit knowledge, gained from experience. The following is a list of possible knowledge types:

- Theories about the problem domain
- Rules and procedures regarding the general problem domain
- Heuristics about what to do in a given problem situation
- Global strategies for solving these types of problems
- Metaknowledge (i.e., knowledge about knowledge)
- Facts about the problem area

These types of knowledge enable experts to make better and faster decisions than nonexperts when solving complex problems.

Expertise often includes the following characteristics:

- Expertise is usually associated with a high degree of intelligence, but it is not always associated with the smartest person.
- Expertise is usually associated with a vast quantity of knowledge.
- Expertise is based on learning from past successes and mistakes.
- Expertise is based on knowledge that is well stored, organized, and quickly retrievable from an expert who has excellent recall of patterns from previous experiences.

Features of ES

ES must have the following features:

- ***Expertise.*** As described in the previous section, experts differ in their level of expertise. An ES must possess expertise that enables it to make expert-level decisions. The system must exhibit expert performance with adequate robustness.
- ***Symbolic reasoning.*** The basic rationale of artificial intelligence is to use symbolic reasoning rather than mathematical calculation. This is also true for ES. That is, knowledge must be represented symbolically, and the primary reasoning mechanism must be symbolic. Typical symbolic reasoning mechanisms include backward chaining and forward chaining, which are described later in this chapter.
- ***Deep knowledge.*** Deep knowledge concerns the level of expertise in a knowledge base. The knowledge base must contain complex knowledge not easily found among nonexperts.
- ***Self-knowledge.*** ES must be able to examine their own reasoning and provide proper explanations as to why a particular conclusion was reached. Most experts have very strong learning capabilities to update their knowledge constantly. ES also need to be able to learn from their successes and failures as well as from other knowledge sources.

The development of ES is divided into two generations. Most first-generation ES use if-then rules to represent and store their knowledge. The second-generation ES are more flexible in adopting multiple knowledge representation and reasoning methods. They may integrate fuzzy logic, neural networks, or genetic algorithms with rule-based inference to achieve a higher level of decision performance. A comparison between conventional systems and ES is given in Table 11.1. Application Case 11.2 illustrates an application of such systems in the sports industry. We will review several applications in the next section.

TABLE 11.1 Comparison of Conventional Systems and Expert Systems

Conventional Systems	Expert Systems
Information and its processing are usually combined in one sequential program.	The knowledge base is clearly separated from the processing (inference) mechanism (i.e.,knowledge rules are separated from the control).
The program does not make mistakes (programmers or users do).	The program may make mistakes.
Conventional systems do not (usually) explain why input data are needed or how conclusions are drawn.	Explanation is a part of most ES.
Conventional systems require all input data. They may not function properly with missing data unless planned for.	ES do not require all initial facts. ES can typically arrive at reasonable conclusions with missing facts.
Changes in the program are tedious (except in DSS).	Changes in the rules are easy to make.
The system operates only when it is completed.	The system can operate with only a few rules (as the first prototype).
Execution is done on a step-by-step (algorithmic) basis.	Execution is done by using heuristics and logic.
Large databases can be effectively manipulated.	Large knowledge bases can be effectively manipulated.
Conventional systems represent and use data.	ES represent and use knowledge.
Efficiency is usually a major goal.	
Effectiveness is important only for DSS.	Effectiveness is the major goal.
Conventional systems easily deal with quantitative data.	ES easily deal with qualitative data.
Conventional systems use numeric data representations.	ES use symbolic and numeric knowledge representations.
Conventional systems capture, magnify, and distribute access to numeric data or information.	ES capture, magnify, and distribute access to judgment and knowledge.

Application Case 11.2

Expert System Helps in Identifying Sport Talents

In the world of sports, recruiters are constantly looking for new talent and parents want to identify the sport that is the most appropriate for their child. Identifying the most plausible match between a person (characterized by a large number of unique qualities and limitations) and a specific sport is anything but a trivial task. Such a matching process requires adequate information about the specific person (i.e., values of certain characteristics), as well as the deep knowledge of what this information should include (i.e., the types of characteristics). In other words, expert knowledge is what is needed in order to accurately predict the right sport (with the highest success possibility) for a specific individual.

It is very hard (if not impossible) to find the true experts for this difficult matchmaking problem. Because the domain of the specific knowledge is divided into various types of sports, the experts have in-depth knowledge of the relevant factors only for a specific sport (that they are an expert of), and beyond the limits of that sport they are not any better than an average spectator. In an ideal case, you would need experts from a wide range of sports brought together into a single room to collectively create a matchmaking decision. Because such a setting is not feasible in the real world, one might consider creating it in the computer world using expert systems. Because expert systems are known to incorporate knowledge from multiple experts, this situation seems to fit well with an expert system–type solution.

In a recent publication Papic et al. (2009) reported on an expert system application for the identification of sports talents. Tapping into the knowledge of a large number of sports experts, they have built a knowledge base of a comprehensive set of rules that maps the expert-driven factors (e.g., physical and cardiovascular measurement, performance test, skill assessments) to different sports. Taking advantage of the inexact representation capabilities of fuzzy logic, they managed to incorporate the exact natural reasoning of the expert knowledge into their advising system.

The system was built as a Web-based DSS using the ASP.NET development platform. Once the system development was completed, it was tested for verification and validation purposes. The system's prediction results were evaluated by experts using real cases collected from the past several years. Comparison was done between the sport proposed by the expert system and the actual outcome of the person's sports career. Additionally, the expert system output and the human expert suggestions were compared using a large number of test cases. All tests showed high reliability and accuracy of the developed system.

Source: V. Papic, N. Rogulj, and V. Pletina, "Identification of Sport Talents Using a Web-Oriented Expert System with a Fuzzy Module," *Expert Systems with Applications,* Vol. 36, 2009, pp. 8830–8838.

SECTION 11.4 REVIEW QUESTIONS

1. What is an ES?
2. Explain why we need ES.
3. What are the major features of ES?
4. What is expertise? Provide an example.
5. Define *deep knowledge* and give an example of it.

11.5 APPLICATIONS OF EXPERT SYSTEMS

ES have been applied to many business and technological areas to support decision making. Application Case 11.3 shows a recent real-world application of ES. Table 11.2 shows some representative ES and their application domains.

Application Case 11.3

Expert System Aids in Identification of Chemical, Biological, and Radiological Agents

Terrorist attacks using chemical, biological, or radiological agents (CBR) are of great concern due to the potential for widespread loss of life. The United States and other nations have spent billions of dollars on plans and protocols in defense against acts of terrorism that could involve CBR. However, CBR covers a wide range of agents with many specific chemicals and biological organisms that could be used in multiple subcategories. Timely response requires rapid identification of the agent involved. This can be a difficult process involving different methods and instruments.

The U.S. Environmental Protection Agency (EPA) along with Dr. Lawrence H. Keith, president of Instant Reference Sources Inc., and others from an extensive team incorporated their knowledge, experience, and expertise, plus information in publicly available EPA documents, to develop the CBR Advisor using Exsys Inc.'s Corvid® software.

One of the most important parts of the CBR Advisor is advice in logical step-by-step procedures to determine the identity of a toxic agent when little or no information is available, which is typical at the beginning of a terrorism incident. The systems help response staff proceed according to a well-established action plan—even in the highly stressful environment of a terrorist attack. The system's dual screens present three levels of information: (1) a top/executive level with brief answers, (2) an educational level with in-depth information, and (3) a research level with links to other documents, slide shows, forms, and Internet sites. Content includes:

- How to classify threat warnings
- How to conduct initial threat evaluation
- Immediate response actions
- How to perform site characterization
- Initial site evaluation and safe entry
- Where and how to best collect samples
- How to package and ship samples for analysis

Restricted content includes CBR agents and methods for analyzing them. The CBR Advisor can be used for incident response and/or training. It has two different menus, one for emergency response and another longer menu for training. The CBR Advisor is a restricted software program and is not publicly available.

Questions for Discussion

1. How can CBR Advisor assist in making quick decisions?
2. What characteristics of CBR Advisor make it an expert system?
3. What could be other situations where such expert systems can be employed?

What We Can Learn from This Application Case

Expert systems are now widely being used in high-pressure situations where the human decision makers often struggle to take quick actions involving both the subjective as well as the objective perspectives in responding to the situations.

Source: **www.exsys.com** "Identification of Chemical, Biological and Radiological Agents," **http://www.exsyssoftware.com/CaseStudySelector/casestudies.html** accessed February 2013.

Classical Applications of ES

Early ES applications, such as DENDRAL for molecular structure identification and MYCIN for medical diagnosis, were primarily in the science domain. XCON for configuration of the VAX computer system at Digital Equipment Corp. (a major producer of minicomputers around 1990 that was later taken over by Compaq) was a successful example in business.

DENDRAL The DENDRAL project was initiated by Edward Feigenbaum in 1965. It used a set of knowledge- or rule-based reasoning commands to deduce the likely molecular structure of organic chemical compounds from known chemical analyses and mass spectrometry data.

TABLE 11.2 Sample Applications of Expert Systems

Expert System	Organization	Application Domain
Classical Applications		
MYCIN	Stanford University	Medical diagnosis
XCON	DEC	System configuration
Expert Tax	Coopers & Lybrand	Tax planning
Loan Probe	Peat Marwick	Loan evaluation
La-Courtier	Cognitive Systems	Financial planning
LMOS	Pacific Bell	Network management
PROSPECTOR	Stanford Research Institute	Discovery of new mineral deposits
Reported Applications		
Fish-Expert	North China	Disease diagnosis in fish
HelpDeskIQ	BMC Remedy	Help desk management
Authorete	Haley	Business rule automation
eCare	CIGNA	Insurance claims
SONAR	NSAD	Stock market monitoring

DENDRAL proved to be fundamentally important in demonstrating how rule-based reasoning could be developed into powerful knowledge engineering tools and led to the development of other rule-based reasoning programs at the Stanford Artificial Intelligence Laboratory (SAIL). The most important of those programs was MYCIN.

MYCIN MYCIN is a rule-based ES that diagnoses bacterial infections of the blood. It was developed by a group of researchers at Stanford University in the 1970s. By asking questions and backward chaining through a rule base of about 500 rules, MYCIN can recognize approximately 100 causes of bacterial infections, which allows the system to recommend effective drug prescriptions. In a controlled test, its performance was rated to be equal that of human specialists. The reasoning and uncertainty processing methods used in MYCIN are pioneers in the area and have generated long-term impact in ES development.

XCON XCON, a rule-based system developed at Digital Equipment Corp., used rules to help determine the optimal system configuration that fit customer requirements. The system was able to handle a customer request within 1 minute that typically took the sales team 20 to 30 minutes. With the ES, service accuracy increased to 98 percent, from a manual approach with an accuracy of 65 percent, saving millions of dollars every year.

Newer Applications of ES

More recent applications of ES include risk management, pension fund advising, business rule automation, automated market surveillance, and homeland security. There are literally thousands of publications reporting applications of expert systems. We mention just a few here.

CREDIT ANALYSIS SYSTEMS ES have been developed to support the needs of commercial lending institutions. ES can help a lender analyze a customer's credit record and determine a proper credit line. Rules in the knowledge base can also help assess risk and risk-management policies. These kinds of systems are used in over one-third of the top 100 commercial banks in the United States and Canada.

PENSION FUND ADVISORS Nestlé Foods Corporation has developed an ES that provides information on an employee's pension fund status. The system maintains an up-to-date knowledge base to give participants advice concerning the impact of regulation changes and conformance with new standards. A system offered on the Internet at the Pingtung Teacher's College in Taiwan has functions that allow participants to plan their retirement through a what-if analysis that calculates their pension benefits under different scenarios.

AUTOMATED HELP DESKS BMC Remedy (**remedy.com**) offers HelpDeskIQ, a rule-based help desk solution for small businesses. This browser-based tool enables small businesses to deal with customer requests more efficiently. Incoming e-mails automatically pass into HelpDeskIQ's business rule engine. The messages are sent to the proper technician, based on defined priority and status. The solution assists help desk technicians in resolving problems and tracking issues more effectively.

Areas for ES Applications

As indicated in the preceding examples, ES have been applied commercially in a number of areas, including the following:

- ***Finance.*** Finance ES include insurance evaluation, credit analysis, tax planning, fraud prevention, financial report analysis, financial planning, and performance evaluation.
- ***Data processing.*** Data processing ES include system planning, equipment selection, equipment maintenance, vendor evaluation, and network management.
- ***Marketing.*** Marketing ES include customer relationship management, market analysis, product planning, and market planning.
- ***Human resources.*** Examples of human resources ES are human resources planning, performance evaluation, staff scheduling, pension management, and legal advising.
- ***Manufacturing.*** Manufacturing ES include production planning, quality management, product design, plant site selection, and equipment maintenance and repair.
- ***Homeland security.*** Homeland security ES include terrorist threat assessment and terrorist finance detection.
- ***Business process automation.*** ES have been developed for help desk automation, call center management, and regulation enforcement.
- ***Healthcare management.*** ES have been developed for bioinformatics and other healthcare management issues.

Now that you are familiar with a variety of different ES applications, it is time to look at the internal structure of an ES and how the goals of the ES are achieved.

SECTION 11.5 REVIEW QUESTIONS

1. What is MYCIN's problem domain?
2. Name two applications of ES in finance and describe their benefits.
3. Name two applications of ES in marketing and describe their benefits.
4. Name two applications of ES in homeland security and describe their benefits.

11.6 STRUCTURE OF EXPERT SYSTEMS

ES can be viewed as having two environments: the development environment and the consultation environment (see Figure 11.4). An ES builder uses the **development environment** to build the necessary components of the ES and to populate the knowledge base with appropriate representation of the expert knowledge. A nonexpert uses the **consultation environment** to obtain advice and to solve problems using the expert knowledge embedded into the system. These two environments can be separated at the end of the system development process.

The three major components that appear in virtually every ES are the knowledge base, the inference engine, and the user interface. In general, though, an ES that interacts with the user can contain the following additional components:

- Knowledge acquisition subsystem
- Blackboard (workplace)
- Explanation subsystem (justifier)
- Knowledge-refining system

Currently, most ES do not contain the knowledge refinement component. A brief description of each of these components follows.

Knowledge Acquisition Subsystem

Knowledge acquisition is the accumulation, transfer, and transformation of problem-solving expertise from experts or documented knowledge sources to a computer

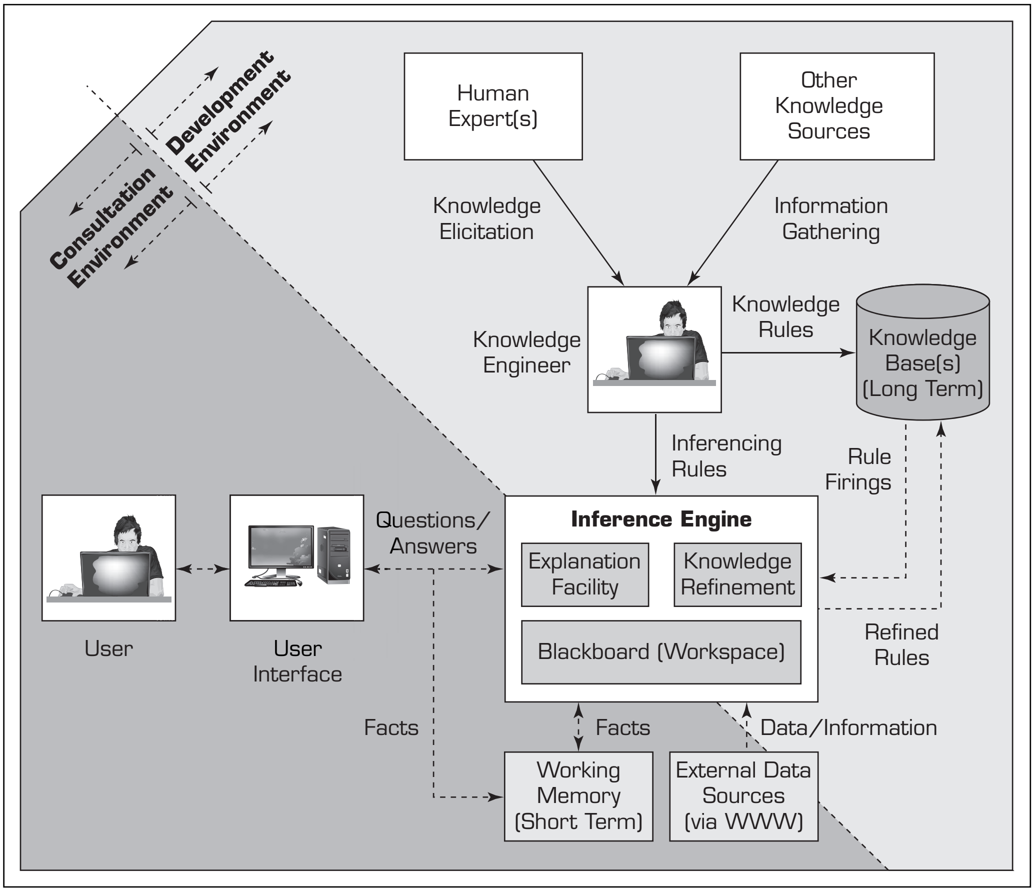

FIGURE 11.4 Structure/Architecture of an Expert System.

program for constructing or expanding the knowledge base. Potential sources of knowledge include human experts, textbooks, multimedia documents, databases (public and private), special research reports, and information available on the Web.

Currently, most organizations have collected a large volume of data, but the organization and management of organizational knowledge are limited. Knowledge acquisition deals with issues such as making tacit knowledge explicit and integrating knowledge from multiple sources.

Acquiring knowledge from experts is a complex task that often creates a bottleneck in ES construction. In building large systems, a knowledge engineer, or knowledge elicitation expert, needs to interact with one or more human experts in building the knowledge base. Typically, the **knowledge engineer** helps the expert structure the problem area by interpreting and integrating human answers to questions, drawing analogies, posing counterexamples, and bringing conceptual difficulties to light.

Knowledge Base

The **knowledge base** is the foundation of an ES. It contains the relevant knowledge necessary for understanding, formulating, and solving problems. A typical knowledge base may include two basic elements: (1) facts that describe the characteristics of a specific problem situation (or *fact base*) and the theory of the problem area and (2) special heuristics or rules (or *knowledge nuggets*) that represent the deep expert knowledge to solve specific problems in a particular domain. Additionally, the inference engine can include general-purpose problem-solving and decision-making rules (or *meta-rules*—rules about how to process production rules).

It is important to differentiate between the knowledge base of an ES and the knowledge base of an organization. The knowledge stored in the knowledge base of an ES is often represented in a special format so that it can be used by a software program (i.e., an expert system shell) to help users solve a particular problem. The organizational knowledge base, however, contains various kinds of knowledge in different formats (most of which is represented in a way that it can be consumed by people) and may be stored in different places. The knowledge base of an ES is a special case and only a very small subset of an organization's knowledge base.

Inference Engine

The "brain" of an ES is the inference engine, also known as the *control structure* or the *rule interpreter* (in rule-based ES). This component is essentially a computer program that provides a methodology for reasoning about information in the knowledge base and on the blackboard to formulate appropriate conclusions. The inference engine provides directions about how to use the system's knowledge by developing the agenda that organizes and controls the steps taken to solve problems whenever a consultation takes place. It is further discussed in Section 11.7.

User Interface

An ES contains a language processor for friendly, problem-oriented communication between the user and the computer, known as the **user interface**. This communication can best be carried out in a natural language. Due to technological constraints, most existing systems use the graphical or textual question-and-answer approach to interact with the user.

Blackboard (Workplace)

The **blackboard** is an area of working memory set aside as a database for description of the current problem, as characterized by the input data. It is also used for recording intermediate results, hypotheses, and decisions. Three types of decisions can be recorded on the blackboard: a plan (i.e., how to attack the problem), an agenda (i.e., potential actions awaiting execution), and a solution (i.e., candidate hypotheses and alternative courses of action that the system has generated thus far).

Consider this example. When your car fails to start, you can enter the symptoms of the failure into a computer for storage in the blackboard. As the result of an intermediate hypothesis developed in the blackboard, the computer may then suggest that you do some additional checks (e.g., see whether your battery is connected properly) and ask you to report the results. This information is also recorded in the blackboard. Such an iterative process of populating the blackboard with values of hypotheses and facts continues until the reason for the failure is identified.

Explanation Subsystem (Justifier)

The ability to trace responsibility for conclusions to their sources is crucial both in the transfer of expertise and in problem solving. The **explanation subsystem** can trace such responsibility and explain the ES behavior by interactively answering questions such as these:

- Why was a certain question asked by the ES?
- How was a certain conclusion reached?
- Why was a certain alternative rejected?
- What is the complete plan of decisions to be made in reaching the conclusion? For example, what remains to be known before a final diagnosis can be determined?

In most ES, the first two questions (why and how) are answered by showing the rule that required asking a specific question and showing the sequence of rules that were used (fired) to derive the specific recommendations, respectively.

Knowledge-Refining System

Human experts have a **knowledge-refining system**; that is, they can analyze their own knowledge and its effectiveness, learn from it, and improve on it for future consultations. Similarly, such evaluation is necessary in expert systems so that a program can analyze the reasons for its success or failure, which could lead to improvements resulting in a more accurate knowledge base and more effective reasoning.

The critical component of a knowledge refinement system is the self-learning mechanism that allows it to adjust its knowledge base and its processing of knowledge based on the evaluation of its recent past performances. Such an intelligent component is not yet mature enough to appear in many commercial ES tools. Application Case 11.4 illustrates another application of expert systems in healthcare.

Application Case 11.4

Diagnosing Heart Diseases by Signal Processing

Auscultation is the science of listening to the sounds of internal body organs, in this case the heart. Skilled experts can make diagnoses using this technique. It is a noninvasive screening method of providing valuable information about the conditions of the heart and its valves, but it is highly subjective and depends on the skills and experience of the listener. Researchers from the Department of Electrical & Electronic Engineering at Universiti Teknologi Petronas have developed an Exsys Corvid expert system, SIPMES (Signal Processing Module Integrated Expert System) to analyze digitally processed heart sound.

The system utilizes digitized heart sound algorithms to diagnose various conditions of the heart. Heart sounds are effectively acquired using a digital electronic stethoscope. The heart sounds were collected from the Institut Jantung Negara (National Heart Institute) in Kuala Lumpur and the Fatimah Ipoh Hospital in Malaysia. A total of 40 patients age 16 to 79 years old with various pathologies were used as the control group, and to test the validity of the system using their abnormal heart sound samples and other patient medical data.

The heart sounds are transmitted using a wireless link to a nearby workstation that hosts the Signal Processing Module (SPM). The SPM has the capability to segment the stored heart sounds into individual cycles and identifies the important cardiac events.

The SPM data was then integrated with the Exsys Corvid knowledge automation expert system. The rules in the system use expert physician reasoning knowledge, combined with information acquired from medical journals, medical textbooks, and other noted publications on cardiovascular diseases (CVD). The system provides the diagnosis and generates a list of diseases arranged in descending order of their probability of occurrence.

SIPMES was designed to diagnose all types of cardiovascular heart diseases. The system can help general physicians diagnose heart diseases at the earliest possible stages under emergency situations where expert cardiologists and advanced medical facilities are not readily available.

The diagnosis made by the system has been counterchecked by senior cardiologists, and the results coincide with these heart experts. A high coincidence factor of 74 percent has been achieved using SIPMES.

Questions for Discussion

1. List the major components involved in building SIPMES and briefly comment on them.
2. Do expert systems like SIPMES eliminate the need for human decision making?
3. How often do you think that the existing expert systems, once built, should be changed?

What We Can Learn from This Application Case

Many expert systems are prominently being used in the field of medicine. Many traditional diagnostic procedures are now being built into logical rule-based systems, which can readily assist the medical staff in quickly diagnosing the patient's condition of disease. These expert systems can help in saving the valuable time of the medical staff and increase the number of patients being served.

Source: **www.exsys.com,** "Diagnosing Heart Diseases,"**exsys http://www.exsyssoftware.com/CaseStudySelector/casestudies.html** (accessed February 2013).

SECTION 11.6 REVIEW QUESTIONS

1. Describe the ES development environment.
2. List and define the major components of an ES.
3. What are the major activities performed in the ES blackboard (workplace)?
4. What are the major roles of the explanation subsystem?
5. Describe the difference between a knowledge base of an ES and an organizational knowledge base.

11.7 KNOWLEDGE ENGINEERING

The collection of intensive activities encompassing the acquisition of knowledge from human experts (and other information sources) and conversion of this knowledge into a repository (commonly called a *knowledge base*) are called **knowledge engineering**. The term *knowledge engineering* was first defined in the pioneering work of Feigenbaum and McCorduck (1983) as the art of bringing the principles and tools of artificial intelligence research to bear on difficult application problems requiring the knowledge of experts for their solutions. Knowledge engineering requires cooperation and close communication between the human experts and the knowledge engineer to successfully codify and explicitly represent the rules (or other knowledge-based procedures) that a human expert uses to solve problems within a specific application domain. The knowledge possessed by human experts is often unstructured and not explicitly expressed. A major goal of knowledge engineering is to help experts articulate *how they do what they do* and to document this knowledge in a reusable form.

Knowledge engineering can be viewed from two perspectives: narrow and broad. According to the narrow perspective, knowledge engineering deals with the steps necessary to build expert systems (i.e., knowledge acquisition, knowledge representation, knowledge validation, inferencing, and explanation/justification). Alternatively, according to the broad perspective, the term describes the entire process of developing and maintaining any intelligent systems. In this book, we use the narrow definition. Following are the five major activities in knowledge engineering:

- ***Knowledge acquisition.*** Knowledge acquisition involves the acquisition of knowledge from human experts, books, documents, sensors, or computer files. The knowledge may be specific to the problem domain or to the problem-solving procedures, it may be general knowledge (e.g., knowledge about business), or it may be metaknowledge (knowledge about knowledge). (By *metaknowledge,* we mean information about how experts use their knowledge to solve problems and about problem-solving procedures in general.)
- ***Knowledge representation.*** Acquired knowledge is organized so that it will be ready for use, in an activity called knowledge representation. This activity involves preparation of a knowledge map and encoding of the knowledge in the knowledge base.
- ***Knowledge validation.*** Knowledge validation (or verification) involves validating and verifying the knowledge (e.g., by using test cases) until its quality is acceptable. Test results are usually shown to a domain expert to verify the accuracy of the ES.
- ***Explanation and justification.*** This step involves the design and programming of an explanation capability (e.g., programming the ability to answer questions such as why a specific piece of information is needed by the computer or how a certain conclusion was derived by the computer).

Figure 11.5 shows the process of knowledge engineering and the relationships among the knowledge engineering activities. Knowledge engineers interact with human experts or collect documented knowledge from other sources in the knowledge acquisition stage. The acquired knowledge is then coded into a representation scheme to create a knowledge base. The knowledge engineer can collaborate with human experts or use test cases to verify and validate the knowledge base. The validated knowledge can be used in a **knowledge-based system** to solve new problems via machine inference and to explain the generated recommendation. Details of these activities are discussed in the following sections.

Knowledge Acquisition

Knowledge is a collection of specialized facts, procedures, and judgment usually expressed as rules. Knowledge can come from one or from many sources, such as books, films, computer databases, pictures, maps, stories, news articles, and sensors, as well as from human experts. Acquisition of knowledge from human experts (often called *knowledge elicitation*) is arguably the most valuable and most challenging task in knowledge acquisition. Technology Insights 11.1 lists some of the difficulties of knowledge acquisition. The classical knowledge elicitation methods, which are also called *manual methods*, include interviewing (i.e., structured, semistructured, unstructured), tracking the reasoning process, and observing. Because these manual methods are slow, expensive, and sometimes inaccurate, the ES community has been developing semiautomated and fully automated means to acquire knowledge. These techniques, which rely on computers

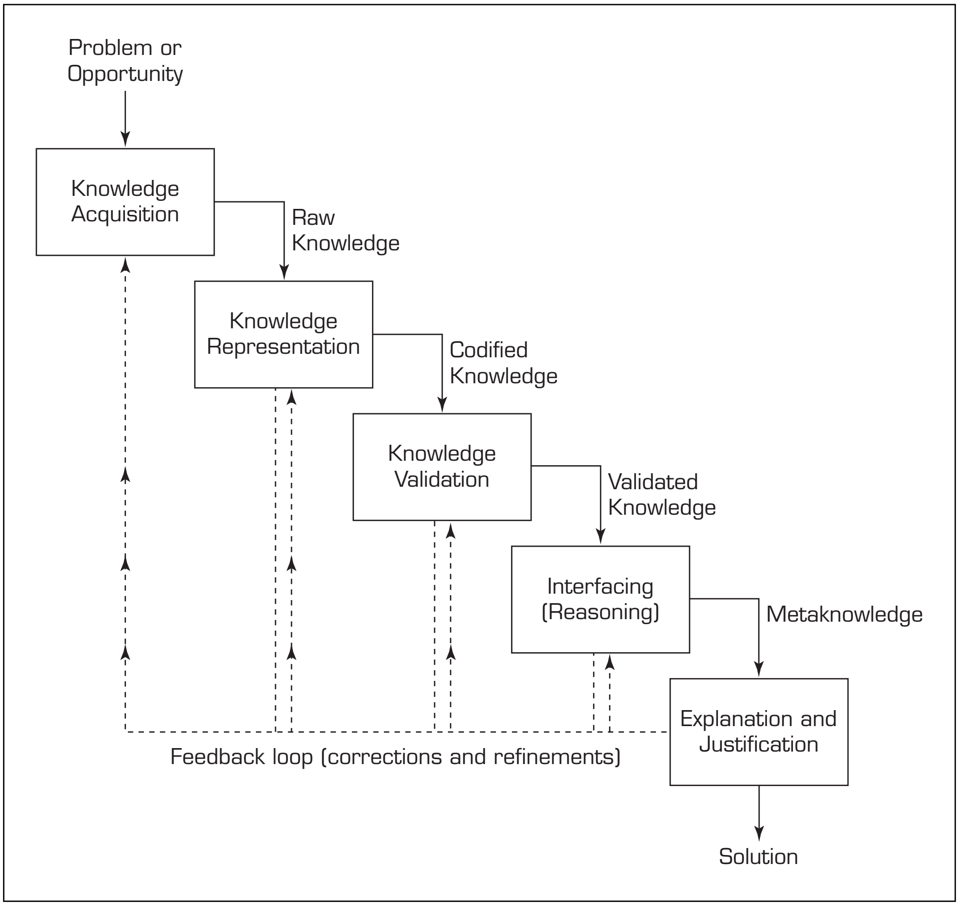

FIGURE 11.5 The Process of Knowledge Engineering.

and AI techniques, aim to minimize the involvement of the knowledge engineer and the human experts in the process. Despite its disadvantages, in real-world ES projects the traditional knowledge elicitation techniques still dominate.

TECHNOLOGY INSIGHTS 11.1 Difficulties in Knowledge Acquisition

Acquiring knowledge from experts is not an easy task. The following are some factors that add to the complexity of knowledge acquisition from experts and its transfer to a computer:

- Experts may not know how to articulate their knowledge or may be unable to do so.
- Experts may lack time or may be unwilling to cooperate.
- Testing and refining knowledge are complicated.
- Methods for knowledge elicitation may be poorly defined.
- System builders tend to collect knowledge from one source, but the relevant knowledge may be scattered across several sources.
- System builders may attempt to collect documented knowledge rather than use experts. The knowledge collected may be incomplete.
- It is difficult to recognize specific knowledge when it is mixed up with irrelevant data.
- Experts may change their behavior when they are observed or interviewed.
- Problematic interpersonal communication factors may affect the knowledge engineer and the expert.

A critical element in the development of an ES is the identification of experts. The usual approach to mitigate this problem is to build ES for a very narrow application domain in which expertise is more clearly defined. Even then, there is a very good chance that one might find more than one expert with different (sometime conflicting) expertise. In such situations, one might choose to use multiple experts in the knowledge elicitation process.

Knowledge Verification and Validation

Knowledge acquired from experts needs to be evaluated for quality, including evaluation, validation, and verification. These terms are often used interchangeably. We use the definitions provided by O'Keefe et al. (1987):

- *Evaluation* is a broad concept. Its objective is to assess an ES's overall value. In addition to assessing acceptable performance levels, it analyzes whether the system would be usable, efficient, and cost-effective.
- *Validation* is the part of evaluation that deals with the performance of the system (e.g., as it compares to the expert's). Simply stated, validation is building the right system (i.e., substantiating that a system performs with an acceptable level of accuracy).
- *Verification* is building the system right or substantiating that the system is correctly implemented to its specifications.

In the realm of ES, these activities are dynamic because they must be repeated each time the prototype is changed. In terms of the knowledge base, it is necessary to ensure that the right knowledge base (i.e., that the knowledge is valid) is used. It is also essential to ensure that the knowledge base has been constructed properly (i.e., verification).

Knowledge Representation

Once validated, the knowledge acquired from experts or induced from a set of data must be represented in a format that is both understandable by humans and executable on computers. A variety of knowledge representation methods is available: production rules, semantic networks, frames, objects, decision tables, decision trees, and predicate logic. Next, we explain the most popular method—production rules.

PRODUCTION RULES **Production rules** are the most popular form of knowledge representation for expert systems. Knowledge is represented in the form of condition/action pairs: IF this condition (or premise or antecedent) occurs, THEN some action (or result or conclusion or consequence) will (or should) occur. Consider the following two examples:

- If the stop light is red AND you have stopped, THEN a right turn is okay.
- If the client uses purchase requisition forms AND the purchase orders are approved and purchasing is separate from receiving AND accounts payable AND inventory records, THEN there is strongly suggestive evidence (90 percent probability) that controls to prevent unauthorized purchases are adequate. (This example from an internal control procedure includes a probability.)

Each production rule in a knowledge base implements an autonomous chunk of expertise that can be developed and modified independently of other rules. When combined and fed to the inference engine, the set of rules behaves synergistically, yielding better results than the sum of the results of the individual rules. In some sense, rules can be viewed as a simulation of the cognitive behavior of human experts. According to this view, rules are not just a neat formalism to represent knowledge in a computer; rather, they represent a model of actual human behavior.

KNOWLEDGE AND INFERENCE RULES Two types of rules are common in artificial intelligence: knowledge and inference. **Knowledge rules**, or *declarative rules,* state all the facts and relationships about a problem. **Inference rules**, or *procedural rules,* offer advice on how to solve a problem, given that certain facts are known. The knowledge engineer separates the two types of rules: Knowledge rules go to the knowledge base, whereas inference rules become part of the inference engine that was introduced earlier as a component of an expert system. For example, assume that you are in the business of buying and selling gold. The knowledge rules might look like this:

Rule 1: IF an international conflict begins, THEN the price of gold goes up.

Rule 2: IF the inflation rate declines, THEN the price of gold goes down.

Rule 3: IF the international conflict lasts more than 7 days and IF it is in the Middle East, THEN buy gold.

Inference rules contain rules about rules and thus are also called meta-rules. They pertain to other rules (or even to themselves). Inference (procedural) rules may look like this:

Rule 1: IF the data needed are not in the system, THEN request them from the user.

Rule 2: IF more than one rule applies, THEN deactivate any rules that add no new data.

Inferencing

Inferencing (or reasoning) is the process of using the rules in the knowledge base along with the known facts to draw conclusions. Inferencing requires some logic embedded in a computer program to access and manipulate the stored knowledge. This program is an algorithm that, with the guidance of the inferencing rules, controls the reasoning process and is usually called the **inference engine**. In rule-based systems, it is also called the *rule interpreter.*

The inference engine directs the search through the collection of rules in the knowledge base, a process commonly called *pattern matching*. In inferencing, when all of the hypotheses (the "IF" parts) of a rule are satisfied, the rule is said to be fired. Once a rule is fired, the new knowledge generated by the rule (the conclusion or the validation of the THEN part) is inserted into the memory as a new fact. The inference engine checks every rule in the knowledge base to identify those that can be fired based on what is known at that point in time (the collection of known facts), and keeps doing so until the goal is achieved. The most popular inferencing mechanisms for rule-based systems are forward and backward chaining:

- **Backward chaining** is a goal-driven approach in which you start from an expectation of what is going to happen (i.e., hypothesis) and then seek evidence that supports (or contradicts) your expectation. Often, this entails formulating and testing intermediate hypotheses (or subhypotheses).
- **Forward chaining** is a data-driven approach. We start from available information as it becomes available or from a basic idea, and then we try to draw conclusions. The ES analyzes the problem by looking for the facts that match the IF part of its IF-THEN rules. For example, if a certain machine is not working, the computer checks the electricity flow to the machine. As each rule is tested, the program works its way toward one or more conclusions.

FORWARD AND BACKWARD CHAINING EXAMPLE Here we discuss an example involving an investment decision about whether to invest in IBM stock. The following variables are used:

A = Have $10,000

B = Younger than 30

C = Education at college level

D = Annual income of at least $40,000

E = Invest in securities

F = Invest in growth stocks

G = Invest in IBM stock (the potential goal)

Each of these variables can be answered as true (yes) or false (no).

We assume that an investor has $10,000 (i.e., that A is true) and that she is 25 years old (i.e., that B is true). She would like advice on investing in IBM stock (yes or no for the goal).

Our knowledge base includes the following five rules:

R1: IF a person has $10,000 to invest and she has a college degree,
THEN she should invest in securities.

R2: IF a person's annual income is at least $40,000 and she has a college degree,
THEN she should invest in growth stocks.

R3: IF a person is younger than 30 and she is investing in securities,
THEN she should invest in growth stocks.

R4: IF a person is younger than 30,
THEN she has a college degree.

R5: IF a person wants to invest in a growth stock,
THEN the stock should be IBM.

These rules can be written as follows:

R1: IF A and C, THEN E.

R2: IF D and C, THEN F.

R3: IF B and E, THEN F.

R4: IF B, THEN C.

R5: IF F, THEN G.

Backward Chaining Our goal is to determine whether to invest in IBM stock. With backward chaining, we start by looking for a rule that includes the goal (G) in its conclusion (THEN) part. Because R5 is the only one that qualifies, we start with it. If several rules contain G, then the inference engine dictates a procedure for handling the situation. This is what we do:

1. Try to accept or reject G. The ES goes to the assertion base to see whether G is there. At present, all we have in the assertion base is A is true. B is true. Therefore, the ES proceeds to step 2.
2. R5 says that if it is true that we invest in growth stocks (F), then we should invest in IBM (G). If we can conclude that the premise of R5 is either true or false, then we have solved the problem. However, we do not know whether F is true. What shall we do now? Note that F, which is the premise of R5, is also the conclusion of R2 and R3. Therefore, to find out whether F is true, we must check either of these two rules.
3. We try R2 first (arbitrarily); if both D and C are true, then F is true. Now we have a problem. D is not a conclusion of any rule, nor is it a fact. The computer can either move to another rule or try to find out whether D is true by asking the investor for whom the consultation is given if her annual income is above $40,000. What the ES does depends on the search procedures used by the inference engine. Usually, a user is asked for additional information only if the information is not available or

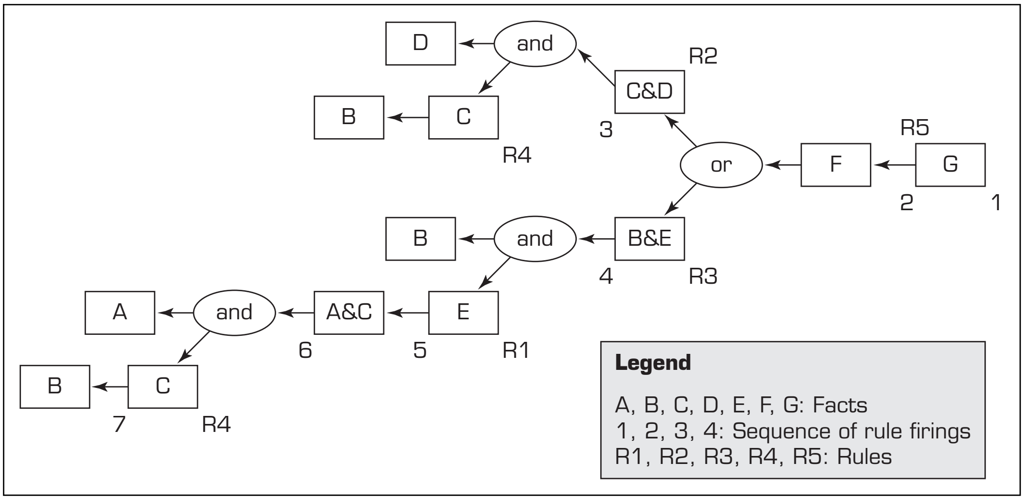

FIGURE 11.6 A Graphical Depiction of Backward Chaining.

cannot be deduced. We abandon R2 and return to the other rule, R3. This action is called *backtracking* (i.e., knowing that we are at a dead end, we try something else; the computer must be preprogrammed to handle backtracking).

4. Go to R3; test B and E. We know that B is true because it is a given fact. To prove E, we go to R1, where E is the conclusion.
5. Examine R1. It is necessary to determine whether A and C are true.
6. A is true because it is a given fact. To test C, it is necessary to test R4 (where C is the conclusion).
7. R4 tells us that C is true (because B is true). Therefore, C becomes a fact (and is added to the assertion base). Now E is true, which validates F, which validates our goal (i.e., the advice is to invest in IBM).

Note that during the search, the ES moved from the THEN part to the IF part, back to the THEN part, and so on (see Figure 11.6 for a graphical depiction of the backward chaining).

Forward Chaining Let us use the same example we examined in backward chaining to illustrate the process of forward chaining. In forward chaining, we start with known facts and derive new facts by using rules having known facts on the IF side. The specific steps that forward chaining would follow in this example are as follows (also see Figure 11.7 for a graphical depiction of this process):

1. Because it is known that A and B are true, the ES starts deriving new facts by using rules that have A and B on the IF side. Using R4, the ES derives a new fact C and adds it to the assertion base as true.
2. R1 fires (because A and C are true) and asserts E as true in the assertion base.
3. Because B and E are both known to be true (they are in the assertion base), R3 fires and establishes F as true in the assertion base.
4. R5 fires (because F is on its IF side), which establishes G as true. So the ES recommends an investment in IBM stock. If there is more than one conclusion, more rules may fire, depending on the inferencing procedure.

INFERENCING WITH UNCERTAINTY Although uncertainty is widespread in the real world, its treatment in the practical world of artificial intelligence is very limited. One could argue that because the knowledge provided by experts is often inexact an ES that mimics

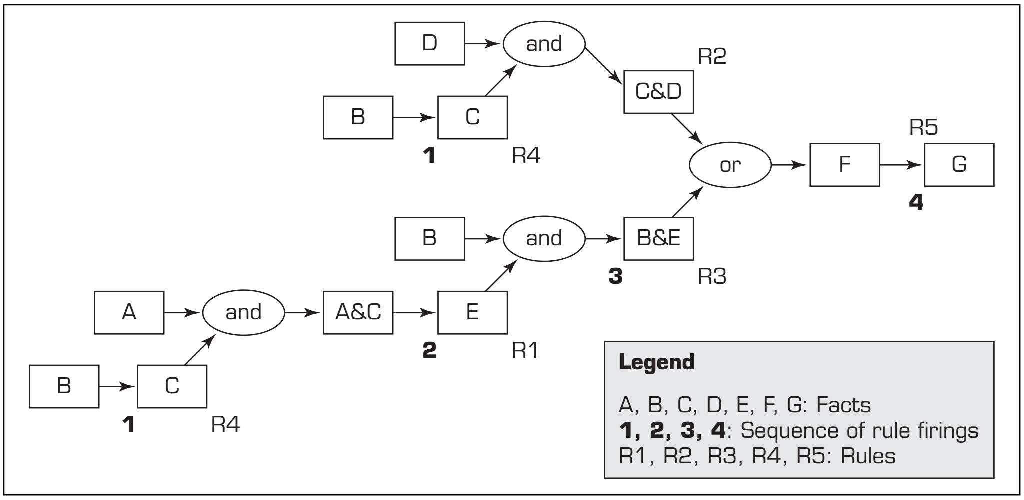

FIGURE 11.7 A Graphical Depiction of Forward Chaining.

the reasoning process of experts should represent such uncertainty. ES researchers have proposed several methods to incorporate uncertainty into the reasoning process, including probability ratios, the Bayesian approach, fuzzy logic, the Dempster–Shafer theory of evidence, and the theory of certainty factors. Following is a brief description of the theory of certainty factors, which is the most commonly used method to accommodate uncertainty in ES.

The **theory of certainty factors** is based on the concepts of belief and disbelief. The standard statistical methods are based on the assumption that an uncertainty is the probability that an event (or fact) is true or false, whereas certainty theory is based on the *degrees of belief* (not the calculated probability) that an event (or fact) is true or false.

Certainty theory relies on the use of certainty factors. **Certainty factors (CF)** express belief in an event (or a fact or a hypothesis) based on the expert's assessment. Certainty factors can be represented by values ranging from 0 to 100; the smaller the value, the lower the probability that the event (or fact) is true or false. Because certainty factors are not probabilities, when we say that there is a certainty value of 90 for rain, we do not mean (or imply) any opinion about no rain (which is not necessarily 10). Thus, certainty factors do not have to sum up to 100.

Combining Certainty Factors Certainty factors can be used to combine estimates by different experts in several ways. Before using any ES shell, you need to make sure that you understand how certainty factors are combined. The most acceptable way of combining them in rule-based systems is the method used in EMYCIN. In this approach, we distinguish between two cases, described next.

Combining Several Certainty Factors in One Rule Consider the following rule with an AND operator:

IF inflation is high, CF = 50 (A)

AND unemployment rate is above 7 percent, CF = 70 (B)

AND bond prices decline, CF = 100 (C),

THEN stock prices decline.

For this type of rule, all IFs must be true for the conclusion to be true. However, in some cases, there is uncertainty as to what is happening. Then the CF of the conclusion is the minimum CF on the IF side:

$$\text{CF(A, B, C)} = \text{minimum [CF(A), CF(B), CF(C)]}$$

Thus, in our case, the CF for stock prices to decline is 50 percent. In other words, the chain is as strong as its weakest link.

Now look at this rule with an OR operator:

IF inflation is low, CF = 70 percent

OR bond prices are high, CF = 85,

THEN stock prices will be high.

In this case, it is sufficient that only one of the IFs is true for the conclusion to be true. Thus, if both IFs are believed to be true (at their certainty factor), then the conclusion will have a CF with the maximum of the two:

$$CF\ (A \text{ or } B) = \text{maximum}\ [CF\ (A),\ CF\ (B)]$$

In our case, CF must be 85 for stock prices to be high. Note that both cases hold for any number of IFs.

Combining Two or More Rules Why might rules be combined? There may be several ways to reach the same goal, each with different certainty factors for a given set of facts. When we have a knowledge-based system with several interrelated rules, each of which makes the same conclusion but with a different certainty factor, each rule can be viewed as a piece of evidence that supports the joint conclusion. To calculate the certainty factor (or the confidence) of the conclusion, it is necessary to combine the evidence. For example, let us assume that there are two rules:

R1: IF the inflation rate is less than 5 percent,
THEN stock market prices go up (CF = 0.7).

R2: IF the unemployment level is less than 7 percent,
THEN stock market prices go up (CF = 0.6).

Now let us assume a prediction that during the next year, the inflation rate will be 4 percent and the unemployment level will be 6.5 percent (i.e., we assume that the premises of the two rules are true). The combined effect is computed as follows:

$$\begin{aligned} CF(R1, R2) &= CF(R1) + CF(R2) \times [1 - CF(R1)] \\ &= CF(R1) + CF(R2) - [CF(R1) \times CF(R2)] \end{aligned}$$

In this example, given CF(R1) = 0.7 and CF(R2) = 0.6

$$CF(R1, R2) = 0.7 + 0.6 - [(0.7) \times (0.6)] = 0.88$$

If we add a third rule, we can use the following formula:

$$\begin{aligned} CF(R1, R2, R3) &= CF(R1, R2) + CF(R3) \times [1 - CF(R1, R2)] \\ &= CF(R1, R2) + CF(R3) - [CF(R1, R2) \times CF(R3)] \end{aligned}$$

In our example:

R3: IF bond price increases,
THEN stock prices go up (CF = 0.85)

$$CF(R1, R2, R3) = 0.88 + 0.85 - [(0.88) \times (0.85)] = 0.982$$

Note that CF(R1,R2) was computed earlier as 0.88. For a situation with more rules, we can apply the same formula incrementally.

Explanation and Justification

A final feature of expert systems is their interactivity with users and their capacity to provide an explanation consisting of the sequence of inferences that were made by the system in arriving at a conclusion. This feature offers a means of evaluating the integrity of the system when it is to be used by the experts themselves. Two basic types of explanations are the why and the how. Metaknowledge is knowledge about knowledge. It is a structure within the system using the domain knowledge to accomplish the system's problem-solving strategy. This section deals with different methods used in ES for generating explanations.

Human experts are often asked to explain their views, recommendations, or decisions. If ES are to mimic humans in performing highly specialized tasks, they, too, need to justify and explain their actions. An explanation is an attempt by an ES to clarify its reasoning, recommendations, or other actions (e.g., asking a question). The part of an ES that provides explanations is called an *explanation facility* (or *justifier*). The explanation facility has several purposes:

- Make the system more intelligible to the user.
- Uncover the shortcomings of the rules and knowledge base (i.e., debugging of the systems by the knowledge engineer).
- Explain situations that were unanticipated by the user.
- Satisfy psychological and social needs by helping the user feel more assured about the actions of the ES.
- Clarify the assumptions underlying the system's operations to both the user and the builder.
- Conduct sensitivity analyses. (Using the explanation facility as a guide, the user can predict and test the effects of changes on the system.)

Explanation in rule-based ES is usually associated with a way of tracing the rules that are fired during the course of a problem-solving session. This is about the closest to a real explanation that today's systems come, given that their knowledge is usually represented almost exclusively as rules that do not include basic principles necessary for a human-type explanation. Though many explanations have been proposed, in the following section we provide brief descriptions for the two most common types: the why and the how explanations.

WHY EXPLANATIONS A user poses a typical "why?" question to a computer after the computer asks the user to provide some information. For example, using our earlier investment example, we might have the following dialogue:

COMPUTER: What is your annual income?

CLIENT: Why? (Why do you need to know?)

COMPUTER: R2; meaning "In checking R2, I need to know whether your income is above $40,000. If this is true, I will conclude that because you have a college degree, you should invest in growth stocks."

HOW EXPLANATIONS A user poses a typical "how?" question when he or she would like to know how a certain conclusion or recommendation was reached. Simple systems are limited to the final conclusion. More complex systems can handle intermediate conclusions as well. The system can explain why a certain rule was fired; that is, it shows the chain of rules used to reach the conclusion. In the investment problem, the how explanation might look like this:

COMPUTER: Invest in IBM stocks.

CLIENT: How? (How was the conclusion reached?)

COMPUTER: R5–R3–R7–R4; meaning "Given that you have $10,000 to invest and you are younger than 30, then according to R4, you have a college degree. If this is the case, then according to R1, you should invest in securities. For a young investor like you, according to R3, you should invest in growth stocks if you are going to invest at all. Finally, according to R5, if you need to invest in growth stocks, then IBM is your best bet."

The why and how explanations often show the rules as they were programmed and not in a natural language. However, some systems have the capability to present these rules in natural language.

SECTION 11.7 REVIEW QUESTIONS

1. State two production rules that can represent the knowledge of repairing your car.
2. Describe how ES perform inference.
3. Describe the reasoning procedures of forward chaining and backward chaining.
4. List the three most popular methods to deal with uncertainty in ES.
5. Why do we need to incorporate uncertainty in ES solutions?
6. What are the ways by which ES justify their knowledge?

11.8 PROBLEM AREAS SUITABLE FOR EXPERT SYSTEMS

ES can be classified in several ways. One way is by the general problem areas they address. For example, diagnosis can be defined as "inferring system malfunctions from observations." *Diagnosis* is a generic activity performed in medicine, organizational studies, computer operations, and so on. The generic categories of ES are listed in Table 11.3. Some ES belong to two or more of these categories. A brief description of each category follows:

- ***Interpretation systems.*** Systems that infer situation descriptions from observations. This category includes surveillance, speech understanding, image analysis, signal interpretation, and many kinds of intelligence analyses. An interpretation system explains observed data by assigning them symbolic meanings that describe the situation.

TABLE 11.3 Generic Categories of Expert Systems

Category	Problem Addressed
Interpretation	Inferring situation descriptions from observations
Prediction	Inferring likely consequences of given situations
Diagnosis	Inferring system malfunctions from observations
Design	Configuring objects under constraints
Planning	Developing plans to achieve goals
Monitoring	Comparing observations to plans and flagging exceptions
Debugging	Prescribing remedies for malfunctions
Repair	Executing a plan to administer a prescribed remedy
Instruction	Diagnosing, debugging, and correcting student performance
Control	Interpreting, predicting, repairing, and monitoring system behaviors

- ***Prediction systems.*** These systems include weather forecasting; demographic predictions; economic forecasting; traffic predictions; crop estimates; and military, marketing, and financial forecasting.
- ***Diagnostic systems.*** These systems include medical, electronic, mechanical, and software diagnoses. Diagnostic systems typically relate observed behavioral irregularities to underlying causes.
- ***Design systems.*** These systems develop configurations of objects that satisfy the constraints of the design problem. Such problems include circuit layout, building design, and plant layout. Design systems construct descriptions of objects in various relationships with one another and verify that these configurations conform to stated constraints.
- ***Planning systems.*** These systems specialize in planning problems, such as automatic programming. They also deal with short- and long-term planning in areas such as project management, routing, communications, product development, military applications, and financial planning.
- ***Monitoring systems.*** These systems compare observations of system behavior with standards that seem crucial for successful goal attainment. These crucial features correspond to potential flaws in the plan. There are many computer-aided monitoring systems for topics ranging from air traffic control to fiscal management tasks.
- ***Debugging systems.*** These systems rely on planning, design, and prediction capabilities for creating specifications or recommendations to correct a diagnosed problem.
- ***Repair systems.*** These systems develop and execute plans to administer a remedy for certain diagnosed problems. Such systems incorporate debugging, planning, and execution capabilities.
- ***Instruction systems.*** Systems that incorporate diagnosis and debugging subsystems that specifically address students' needs. Typically, these systems begin by constructing a hypothetical description of the student's knowledge that interprets her or his behavior. They then diagnose weaknesses in the student's knowledge and identify appropriate remedies to overcome the deficiencies. Finally, they plan a tutorial interaction intended to deliver remedial knowledge to the student.
- ***Control systems.*** Systems that adaptively govern the overall behavior of a system. To do this, a control system must repeatedly interpret the current situation, predict the future, diagnose the causes of anticipated problems, formulate a remedial plan, and monitor its execution to ensure success.

Not all the tasks usually found in each of these categories are suitable for ES. However, thousands of decisions do fit into these categories.

SECTION 11.8 REVIEW QUESTIONS

1. Describe a sample ES application for prediction.
2. Describe a sample ES application for diagnosis.
3. Describe a sample ES application for the rest of the generic ES categories.

11.9 DEVELOPMENT OF EXPERT SYSTEMS

The development of ES is a tedious process and typically includes defining the nature and scope of the problem, identifying proper experts, acquiring knowledge, selecting the building tools, coding the system, and evaluating the system.

Defining the Nature and Scope of the Problem

The first step in developing an ES is to identify the nature of the problem and to define its scope. Some domains may not be appropriate for the application of ES. For example, a problem that can be solved by using mathematical optimization algorithms is often inappropriate for ES. In general, rule-based ES are appropriate when the nature of the problem is qualitative, knowledge is explicit, and experts are available to solve the problem effectively and provide their knowledge.

Another important factor is to define a feasible scope. The current technology is still very limited and is capable of solving relatively simple problems. Therefore, the scope of the problem should be specific and reasonably narrow. For example, it may be possible to develop an ES for detecting abnormal trading behavior and possible money laundering, but it is not possible to use an ES to determine whether a particular transaction is criminal.

Identifying Proper Experts

After the nature and scope of the problem have been clearly defined, the next step is to find proper experts who have the knowledge and are willing to assist in developing the knowledge base. No ES can be designed without the strong support of knowledgeable and supportive experts. A project may identify one expert or a group of experts. A proper expert should have a thorough understanding of problem-solving knowledge, the role of ES and decision support technology, and good communication skills.

Acquiring Knowledge

After identifying helpful experts, it is necessary to start acquiring decision knowledge from them. The process of eliciting knowledge is called *knowledge engineering*. The person who is interacting with experts to document the knowledge is called a *knowledge engineer*.

Knowledge acquisition is a time-consuming and risky process. Experts may be unwilling to provide their knowledge for various reasons. First, their knowledge may be proprietary and valuable. Experts may not be willing to share their knowledge without a reasonable payoff. Second, even though an expert is willing to share, certain knowledge is tacit, and the expert may not have the skill to clearly dictate the decision rules and considerations. Third, experts may be too busy to have enough time to communicate with the knowledge engineer. Fourth, certain knowledge may be confusing or contradictory in nature. Finally, the knowledge engineer may misunderstand the expert and inaccurately document knowledge.

The result of knowledge acquisition is a knowledge base that can be represented in different formats. The most popular one is if-then rules. The knowledge may also be represented as decision trees or decision tables. The knowledge in the knowledge base must be evaluated for its consistency and applicability.

Selecting the Building Tools

After the knowledge base is built, the next step is to choose a proper tool for implementing the system. There are three different kinds of development tools, as described in the following sections.

GENERAL-PURPOSE DEVELOPMENT ENVIRONMENT The first type of tool is general-purpose computer languages, such as C++, Prolog, and LISP. Most computer programming languages support the if-then statement. Therefore, it is possible to use C++ to develop an ES for a particular problem domain (e.g., disease diagnosis). Because these programming languages do not have built-in inference capabilities, using them in this way is often very costly and time-consuming. Prolog and LISP are two languages for developing intelligent systems. It is easier to use them than to use C++, but they are still specifically designed for professional programmers and are not very friendly. For recent Web-based applications, Java and computer languages that support Web services (such as the Microsoft .NET platform) are also useful. Companies such as Logic Programming Associates (**www.lpa.co.uk**) offer Prolog-based tools.

ES SHELLS The second type of development tool, the **expert system (ES) shell**, is specifically designed for ES development. An ES shell has built-in inference capabilities and a user interface, but the knowledge base is empty. System development is therefore a process of feeding the knowledge base with rules elicited from the expert.

A popular ES shell is the Corvid system developed by Exsys (**exsys.com**). The system is an object-oriented development platform that is composed of three types of operations: variables, logic blocks, and command blocks. Variables define the major factors considered in problem solving. Logic blocks are the decision rules acquired from experts. Command blocks determine how the system interacts with the user, including the order of execution and the user interface. Figure 11.8 shows a screenshot of a logic block that shows the decision rules under Exys Corvid. More products are available from business rules management vendors, such as LPA's VisiRule (**www.lpa.co.uk/vsr.htm**), which is based on a general-purpose tool called Micro-Prolog.

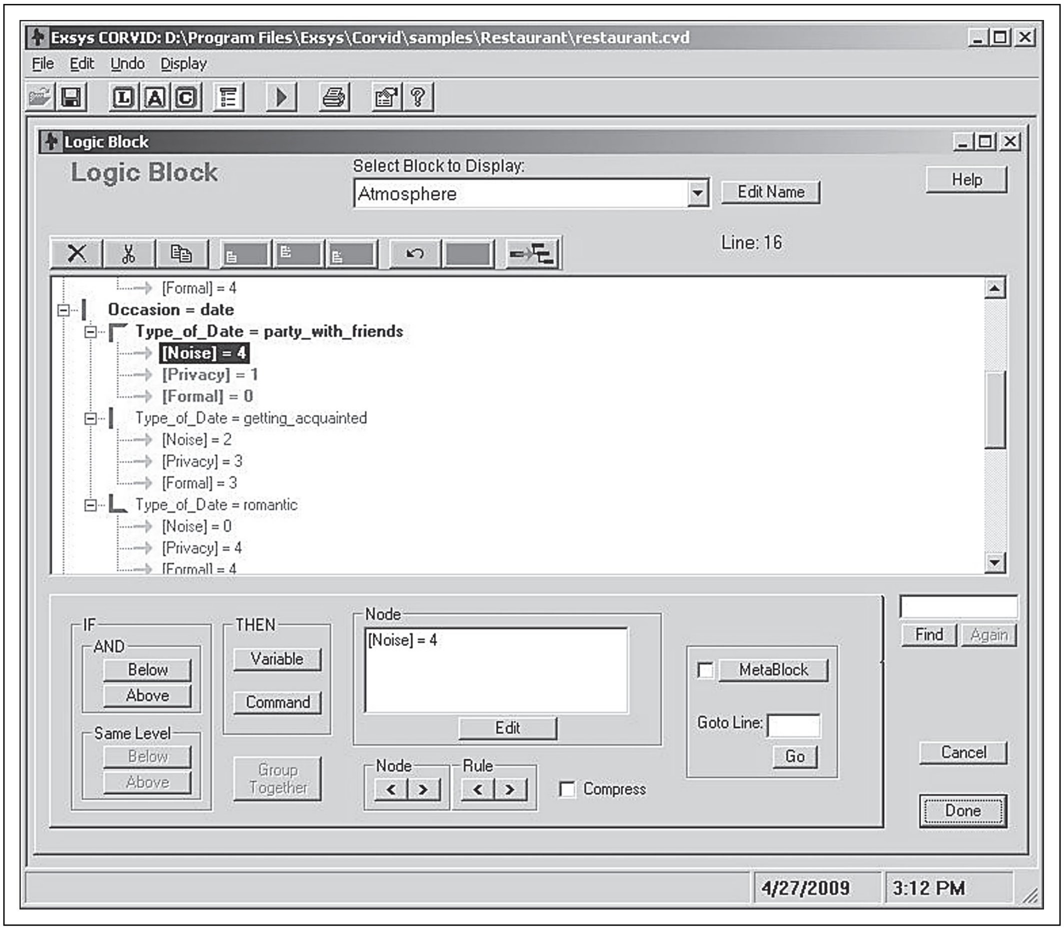

FIGURE 11.8 A Screenshot from Corvid Expert System Shell.

TAILORED TURN-KEY SOLUTIONS The third tool, a tailored turn-key tool, is tailored to a specific domain and can be adapted to a similar application very quickly. Basically, a tailored turn-key tool contains specific features often required for developing applications in a particular domain. This tool must adjust or modify the base system by tailoring the user interface or a relatively small portion of the system to meet the unique needs of an organization.

CHOOSING AN ES DEVELOPMENT TOOL Choosing among these tools for ES development depends on a few criteria. First, you need to consider the cost benefits. Tailored turn-key solutions are the most expensive option. However, you need to consider the total cost, not just the cost of the tool. Second, you need to consider the technical functionality and flexibility of the tool; that is, you need to determine whether the tool provides the function you need and how easily it allows the development team to make necessary changes. Third, you need to consider the tool's compatibility with the existing information infrastructure in the organization. Most organizations have many existing applications, and the tool must be compatible with those applications and needs to be able to be integrated as part of the entire information infrastructure. Finally, you need to consider the reliability of the tool and vendor support. The vendor's experiences in similar domains and training programs are critical to the success of an ES project.

Coding the System

After choosing a proper tool, the development team can focus on coding the knowledge based on the tool's syntactic requirements. The major concern at this stage is whether the coding process is efficient and properly managed to avoid errors. Skilled programmers are helpful and important.

Evaluating the System

After an ES system is built, it must be evaluated. Evaluation includes both verification and validation. Verification ensures that the resulting knowledge base contains knowledge exactly the same as that acquired from the expert. In other words, verification ensures that no error occurred at the coding stage. Validation ensures that the system can solve the problem correctly. In other words, validation checks whether the knowledge acquired from the expert can indeed solve the problem effectively. Application Case 11.5 illustrates a case where evaluation played a major role.

Application Case 11.5

Clinical Decision Support System for Tendon Injuries

Flexor tendon injuries in the hand continue to be one of the greatest challenges in hand surgery and hand therapy. Despite the advances in surgical techniques, better understanding is needed of the tendon anatomy, healing process and suture strength, edema, scarring, and stiffness. The Clinical Decision Support System (CDSS) system focuses on flexor tendon injuries in Zone II, which is technically the most demanding in both surgical and rehabilitation areas. This zone is considered a "No Man's Land" in which not many surgeons feel comfortable repairing. It is very difficult and time-consuming for both the hand surgeon and hand therapist working with tendon injury patients to keep up with the ongoing advances in this field. However, it is essential to be aware of all the information that would be

(*Continued*)

Application Case 11.5 (Continued)

potentially useful in making optimal clinical judgments. Major functions of the system include supporting clinical diagnosis, treatment plan processes, promoting the use of best practices, condition-specific guidelines, and population-based management. In the medical field, knowledge automation expert systems are a widely used type of CDSS.

The Clinical Decision Support System, developed using Exsys Knowledge Automation Software for Zone II flexor tendon injuries, encompasses the continuum from injury to complete rehabilitation of the tendon. The system architecture uses rule-based logic blocks to create a decision support system, which takes the user (hand surgeon, hand therapist, and other medical personnel) through a series of questions. Based on the users' input, the system's analysis will make recommendations for repair and rehabilitation for each particular situation. The CDSS takes the user through the entire process of a person's hand injury involving the flexor tendon from the emergency room encounter to full recovery, including rehabilitation of the tendon.

With combined experience of 45 years, this system was tested by hand therapists, and plastic hand and orthopedic surgeons. Two out of three concurred that this system works well, encompassing the entire continuum of the flexor tendon injury and rehabilitation. They also agreed that the system is a good support tool making good decisions, and can be used by nonexpert healthcare personnel.

Questions for Discussion

1. Research other expert systems in other domains and list a few of them.
2. Why is important to evaluate the expert systems before they are put into use?

What We Can Learn from This Application Case

The need for expert systems is continuously increasing in various fields. Many individuals are now imparting their knowledge and useful experience for future generations by building the expert systems. Also, expert systems having interactive capabilities and abilities to provide ready information based on the user inputs provide an ideal platform for expediting human decision making.

Source: **www.exsys.com,** "Advanced Clinical Advice for Tendon Injuries," **http://www.exsyssoftware.com/CaseStudySelector/casestudies.html** (accessed February 2013).

SECTION 11.9 REVIEW QUESTIONS

1. Describe the major steps in developing rule-based ES.

2. What are the necessary conditions for a good expert?

3. Compare three different types of tools for developing ES.

4. List the criteria for choosing a development tool.

5. What is the difference between verification and validation of an ES?

11.10 CONCLUDING REMARKS

Rule-based systems have become extremely useful and important in practice. As seen in this chapter, there are systems that implement the results of other analytical models through a rule-based system, and then there are systems that are built to take advantage of the subjective knowledge of human experts. Collectively, these technologies enable an organization to pull its expertise together and use it in decision making. Sometimes, it is customer-facing; at other times, it plays an internal decision-making role. But these technologies are now quite practicable and in use in organizations. Confluence of artificial intelligence, Web development technologies, and analytical methods makes it possible for systems to be deployed that collectively give an organization a major competitive advantage or allow for better social welfare.

Chapter Highlights

- Artificial intelligence (AI) is a discipline that investigates how to build computer systems to perform tasks that can be characterized as intelligent.
- The major characteristics of AI are symbolic processing, the use of heuristics instead of algorithms, and the application of inference techniques.
- Knowledge, rather than data or information, is the major focus of AI.
- Major areas of AI include expert systems, natural language processing, speech understanding, intelligent robotics, computer vision, fuzzy logic, intelligent agents, intelligent computer-aided instruction, automatic programming, neural computing, game playing, and language translation.
- Expert systems (ES) are the most often applied AI technology. ES attempt to imitate the work of experts. They capture human expertise and apply it to problem solving.
- For an ES to be effective, it must be applied to a narrow domain, and the knowledge must include qualitative factors.
- The power of an ES is derived from the specific knowledge it possesses, not from the particular knowledge representation and inference schemes it uses.
- Expertise is task-specific knowledge acquired through training, reading, and experience.
- ES technology can transfer knowledge from experts and documented sources to the computer and make it available for use by nonexperts.
- The major components of an ES are the knowledge acquisition subsystem, knowledge base, inference engine, user interface, blackboard, explanation subsystem, and knowledge-refinement subsystem.
- The inference engine provides reasoning capability for an ES.
- ES inference can be done by using forward chaining or backward chaining.
- Knowledge engineers are professionals who know how to capture the knowledge from an expert and structure it in a form that can be processed by the computer-based ES.
- ES development process includes defining the nature and scope of the problem, identifying proper experts, acquiring knowledge, selecting the building tools, coding the system, and evaluating the system.
- ES are popular in a number of generic categories: interpretation, prediction, diagnosis, design, planning, monitoring, debugging, repair, instruction, and control.
- The ES shell is an ES development tool that has the inference engine and building blocks for the knowledge base and the user interface. Knowledge engineers can easily develop a prototype system by entering rules into the knowledge base.

Key Terms

artificial intelligence (AI)
automated decision systems (ADS)
backward chaining
blackboard
certainty factors (CF)
consultation environment
decision automation systems
development environment
expert
expert system (ES)
expert system (ES) shell
expertise
explanation subsystem
forward chaining
inference engine
inference rules
knowledge acquisition
knowledge base
knowledge engineer
knowledge engineering
knowledge rules
knowledge-based system (KBS)
knowledge-refining system
production rules
revenue management systems
rule-based systems
theory of certainty factors
user interface

Questions for Discussion

1. Why are automated decision systems so important for business applications?
2. It is said that powerful computers, inference capabilities, and problem-solving heuristics are necessary but not sufficient for solving real problems. Explain.
3. Explain the relationship between the development environment and the consultation (i.e., runtime) environment.
4. Explain the difference between forward chaining and backward chaining and describe when each is most appropriate.
5. What kinds of mistakes might ES make and why? Why is it easier to correct mistakes in ES than in conventional computer programs?
6. An ES for stock investment is developed and licensed for $1,000 per year. The system can help identify the most undervalued securities on the market and the best timing for buying and selling the securities. Will you order a copy as your investment advisor? Explain why or why not.

Exercises

Teradata UNIVERSITY NETWORK (TUN) and Other Hands-on Exercises

1. Go to **teradatauniversitynetwork.com** and search for stories about Chinatrust Commercial Bank's (CTCB's) use of the Teradata Relationship Manager and its reported benefits. Study the functional demo of the Teradata Relationship Manager to answer the following questions:
   a. What functions in the Teradata Relationship Manager are useful for supporting the automation of business rules? In CTCB's case, identify a potential application that can be supported by rule-based ES and solicit potential business rules in the knowledge base.
   b. Access Haley and compare the Teradata Relationship Manager and Haley's Business Rule Management System. Which tool is more suitable for the application identified in the previous question?
2. We list 10 categories of ES applications in the chapter. Find 20 sample applications, 2 in each category, from the various functional areas in an organization (i.e., accounting, finance, production, marketing, and human resources).
3. Download Exsys' Corvid tool for evaluation. Identify an expert (or use one of your teammates) in an area where experience-based knowledge is needed to solve problems, such as buying a used car, selecting a school and major, selecting a job from many offers, buying a computer, diagnosing and fixing computer problems, etc. Go through the knowledge-engineering process to acquire the necessary knowledge. Using the evaluation version of the Corvid tool, develop a simple expert system application on the expertise area of your choice. Report on your experiences in a written document; use screenshots from the software as necessary.
4. Search to find applications of artificial intelligence and ES. Identify an organization with which at least one member of your group has a good contact who has a decision-making problem that requires some expertise (but is not too complicated). Understand the nature of its business and identify the problems that are supported or can potentially be supported by rule-based systems. Some examples include selection of suppliers, selection of a new employee, job assignment, computer selection, market contact method selection, and determination of admission into graduate school.
5. Identify and interview an expert who knows the domain of your choice. Ask the expert to write down his or her knowledge. Choose an ES shell and build a prototype system to see how it works.
6. Go to **exsys.com** to play with the restaurant selection example in its demo systems. Analyze the variables and rules contained in the example's knowledge base.
7. Access the Web site of the American Association for Artificial Intelligence (**aaai.org**). Examine the workshops it has offered over the past year and list the major topics related to intelligent systems.

End-of-Chapter Application Case

Tax Collections Optimization for New York State

Introduction

Tax collection in the State of New York is under the mandate of the New York State Department of Taxation and Finance's Collections and Civil Enforcement Division (CCED). Between 1995 and 2005, CCED changed and improved on its operations in order to make tax collection more efficient. Even though the division's staff strength decreased from over 1,000 employees in 1995 to about 700 employees, its tax collection revenue increased from $500 million to over $1 billion within the same period as a result of the improved systems and procedures they used.

Presentation of Problem

The State of New York found it a challenge to reverse its growing budget deficit, partly due to the unfavorable economic conditions prior to 2009. A key part of the state's budget is revenue from tax collection, which forms about 40 percent of their yearly revenue. Tax collection mechanism was therefore seen as one key area that would help decrease the state's budget deficit if improved. The goal was to optimize tax collection in a very efficient way. The existing rigid and manual rules took too long to implement and also required too many personnel and resources to be used. This was not going to be feasible any longer because the resources allocated to the CCED for tax collection were in line to be reduced. This meant the tax collection division had to find ways of doing more with fewer resources.

Methodology/Solution

Out of all the improvements CCED made to their work process between 1995 and 2005, one area that remained unchanged was the process of collection of delinquent taxes. The existing method for tax collection employed a linear approach to identify, initiate, and collect delinquent taxes. This approach emphasized what should be done, rather than what could be done, by tax collection officers. A "one-size-fits-all" procedure for data collection was used within the constraints of allowable laws. However, the challenge of a complex legal tax system, and the less than optimal results produced by their existing scoring system, made the approach deficient. When 70 percent of delinquent cases relate to individuals and 30 percent relate to business, it is difficult to operate at an optimal level by taking on delinquent cases based on whether it is allowable or not. Better processes that would allow smarter decisions about which delinquent cases to pursue had to be developed within a constrained Markov Decision Process (MDP) framework.

Analytics and optimization processes were coupled with a Constrained Reinforcement Learning (C-RL) method. This method helped develop rules for tax collection based on taxpayer characteristics. That is, they determined that the past behavior of a taxpayer was a major predictor of a taxpayer's future behavior, and this discovery was leveraged by the method used. Basically, data analytics and optimization process were performed based on the following inputs: a list of business rules for collecting taxes, the state of the tax collection process, and resources available. These inputs produced rules for allocating actions to be taken in each tax delinquency situation.

Results/Benefits

The new system, implemented in 2009, enabled the tax agency to only collect delinquent tax when needed as opposed to when allowed within the constraints of the law. The year-to-year increase in revenue between 2007 and 2010 was 8.22 percent ($83 million). As a result of more efficient tax collection rules, fewer personnel were needed both at their contact center and on the field. The average age of cases, even with fewer employees, dropped by 9.3 percent; however, the amount of dollars collected per field agent increased by about 15 percent. Overall, there was a 7 percent increase in revenue from 2009 to 2010. As a result, more revenue was generated to support state programs.

Questions for the End-of-Chapter Application Case

1. What is the key difference between the former tax collection system and the new system?
2. List at least three benefits that were derived from implementing the new system.
3. In what ways do analytics and optimization support the generation of an efficient tax collection system?
4. Why was tax collection a target for decreasing the budget deficit in the State of New York?

What We Can Learn from This End-of-Chapter Application Case

This case presents a scenario that depicts the dual use of predictive analytics and optimization in solving real-world problems. Predictive analytics was used to determine the future tax behavior of taxpayers based on their past behavior. Based on the delinquency status of the individual's or corporate body's tax situation, different courses of action are followed based on established rules. Hence, the tax agency was able to sidestep the "one-size-fits-all" policy and initiate tax collection procedures based on what they should do to increase tax revenue, and not just what they could lawfully do. The rule-based system was implemented using the information derived from optimization models.

Source: Gerard Miller, Melissa Weatherwax, Timothy Gardinier, Naoki Abe, Prem Melville, Cezar Pendus, David Jensen, et al., "Tax Collections Optimization for New York State." *Interfaces,* Vol. 42, No. 1, 2012, pp. 74–84.

References

DemandTec. "Giant Food Stores Prices the Entire Store with DemandTec." **https://mydt.demandtec.com/mydemandtec/c/document_library/get_file?uuid=3151a5e4-f3e1-413e-9cd7-333289eeb3d5&groupId=264319** (accessed February 2013).

Exsyssoftware.com. "Advanced Clinical Advice for Tendon Injuries." **exsyssoftware.com/CaseStudySelector/**

AppDetailPages/ExsysCaseStudiesMedical.html (accessed February 2013).

Exsyssoftware.com. "Diagnosing Heart Diseases." **exsyssoftware.com/CaseStudySelector/AppDetailPages/ExsysCaseStudiesMedical.html** (accessed February 2013).

Exsyssoftware.com. "Identification of Chemical, Biological and Radiological Agents." **exsyssoftware.com/CaseStudySelector/AppDetailPages/ExsysCaseStudiesMedical.html** (accessed February 2013).

Koushik, Dev, Jon A. Higbie, and Craig Eister. (2012). "Retail Price Optimization at InterContinental Hotels Group." *Interfaces,* Vol. 42, No. 1, pp. 45–57.

Miller, G., M. Weatherwax, T. Gardinier, N. Abe, P. Melville, C. Pendus, D. Jensen, et al. (2012). "Tax Collections Optimization for New York State." *Interfaces,* Vol. 42, No. 1, pp. 74–84.

Papic, V., N. Rogulj, and V. Pleština. (2009). "Identification of Sport Talents Using a Web-Oriented Expert System with a Fuzzy Module." *Expert Systems with Applications,* Vol. 36, pp. 8830–8838.

CHAPTER

12

Knowledge Management and Collaborative Systems

LEARNING OBJECTIVES

- Define knowledge and describe the different types of knowledge
- Describe the characteristics of knowledge management
- Describe the knowledge management cycle
- Describe the technologies that can be used in a knowledge management system (KMS)
- Describe different approaches to knowledge management
- Understand the basic concepts and processes of groupwork, communication, and collaboration
- Describe how computer systems facilitate communication and collaboration in an enterprise
- Explain the concepts and importance of the time/place framework
- Explain the underlying principles and capabilities of groupware, such as group support systems (GSS)
- Understand how the Web enables collaborative computing and group support of virtual meetings
- Describe the role of emerging technologies in supporting collaboration

In this chapter, we study two major IT initiatives related to decision support. First, we describe the characteristics and concepts of knowledge management. We explain how firms use information technology (IT) to implement knowledge management (KM) systems and how these systems are transforming modern organizations. Knowledge management, although conceptually ancient, is a relatively new business philosophy. The goal of knowledge management is to identify, capture, store, maintain, and deliver useful knowledge in a meaningful form to anyone who needs it, anyplace and anytime, within an organization. Knowledge management is about sharing and collaborating at the organization level. People work together, and groups make most of the complex decisions in organizations. The increase in organizational decision-making complexity increases the need for meetings and groupwork. Supporting groupwork, where team members may be in different locations and working at different times, emphasizes the important aspects of communications, computer-mediated collaboration, and work methodologies.

Group support is a critical aspect of decision support systems (DSS). Effective computer-supported group support systems have evolved to increase gains and decrease losses in task performance and underlying processes. So this chapter covers both knowledge management and collaborative systems. It consists of the following sections:

12.1 OPENING VIGNETTE: Expertise Transfer System to Train Future Army Personnel

A major problem for organizations implementing knowledge management systems such as lessons-learned capabilities is the lack of success of such systems or poor service of the systems to their intended goal of promoting knowledge reuse and sharing. Lessons-learned systems are part of the broad organizational and knowledge management systems that have been well studied by IS researchers. The objective of lessons-learned systems is to support the capture, codification, presentation, and application of expertise in organizations. Lesson-learned systems have been a failure mainly for two reasons—inadequate representation and lack of integration into an organization's decision-making process.

The expertise transfer system (ETS) is a knowledge transfer system developed by the Spears School of Business at Oklahoma State University as a prototype for the Defense Ammunition Center (DAC) in McAlester, Oklahoma, for use in Army ammunition career fields. The ETS is designed to capture the knowledge of experienced ammunition personnel leaving the Army (i.e., retirements, separations, etc.) and those who have been recently deployed to the field. This knowledge is captured on video, converted into units of actionable knowledge called "nuggets," and presented to the user in a number of learning-friendly views.

ETS begins with an audio/video–recorded (A/V) interview between an interviewee and a "knowledge harvester." Typically, the recording lasts between 60 and 90 minutes. Faculty from the Oklahoma State University College of Education trained DAC knowledge harvesters on effective interviewing techniques, methods of eliciting tacit information from the interviewees, and ways to improve recorded audio quality in the interview process. Once the videos have been recorded, the meat of the ETS process takes place, as depicted in Figure 12.1. First, the digital A/V files are converted to text. Currently, this is accomplished with human transcriptionists, but we have had promising results using voice recognition (VR) technologies for transcription and foresee a day when most of the transcription will be automated. Second, the transcriptions are parsed into small units and organized into knowledge nuggets (KN). Simply put, a knowledge nugget is a significant experience the interviewee had during his/her career that is worth sharing. Then these

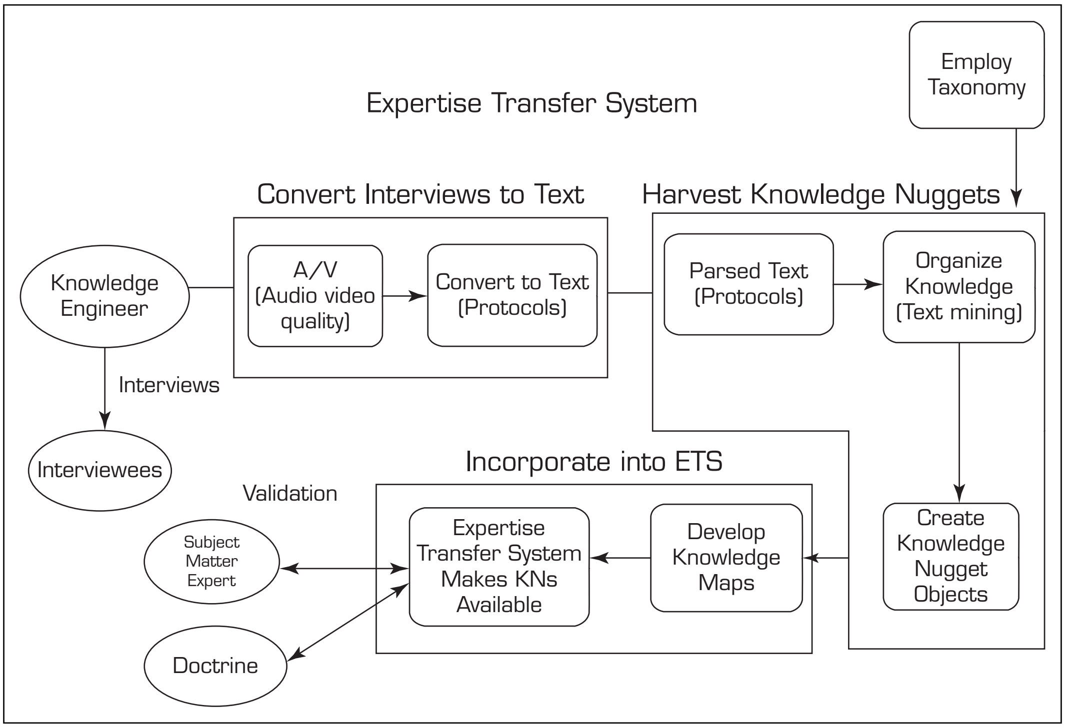

FIGURE 12.1 Development Process for Expertise Transfer System.

KNs are incorporated into the expertise transfer system. Finally, additional features are added to the KNs to make them easy to find, more user friendly, and more effective in the classroom.

KNOWLEDGE NUGGETS

We chose to call the harvested knowledge assets *knowledge nuggets* (KN). Of the many definitions or explanations provided by a thesaurus for *nugget*, two explanations stand out: (1) a lump of precious metal, and (2) anything of great value or significance. A knowledge nugget assumes even more importance because knowledge already is of great value. A KN can be just one piece of knowledge like a video or text. However, a KN can also be a combination of video, text, documents, figures, maps, and so forth. The tools used to transfer knowledge have a central theme, which is the knowledge itself. In our DAC repository, we have a combination of knowledge statements, videos, corresponding transcripts, causal maps, and photographs. The knowledge nugget is a specific lesson learned on a particular topic that has been developed for future use. It consists of several components. Figure 12.2 displays a sample knowledge nugget. A summary page provides the user with the title or "punchline" of the KN, the name and deployment information of the interviewee, and a bulleted summary of the KN. Clicking on the video link will bring the users to the KN video clip, whereas clicking on the transcript link will provide them with a complete transcript of the nugget. The KN text is linked back to the portion of the A/V interview from which it was harvested. The result is a searchable 30- to 60-second video clip (with captions) of the KN. A causal map function gives the user an opportunity to see and understand the thought process of the interviewee as they describe the situation captured by the nugget. The related links feature provides users with a list of regulatory guidance associated with the KN, and the related nuggets link lists all KNs within the same knowledge domain. Also provided is information about the interviewee, recognized subject matter experts (SMEs) in the KN domain, and supporting images related to the nugget.

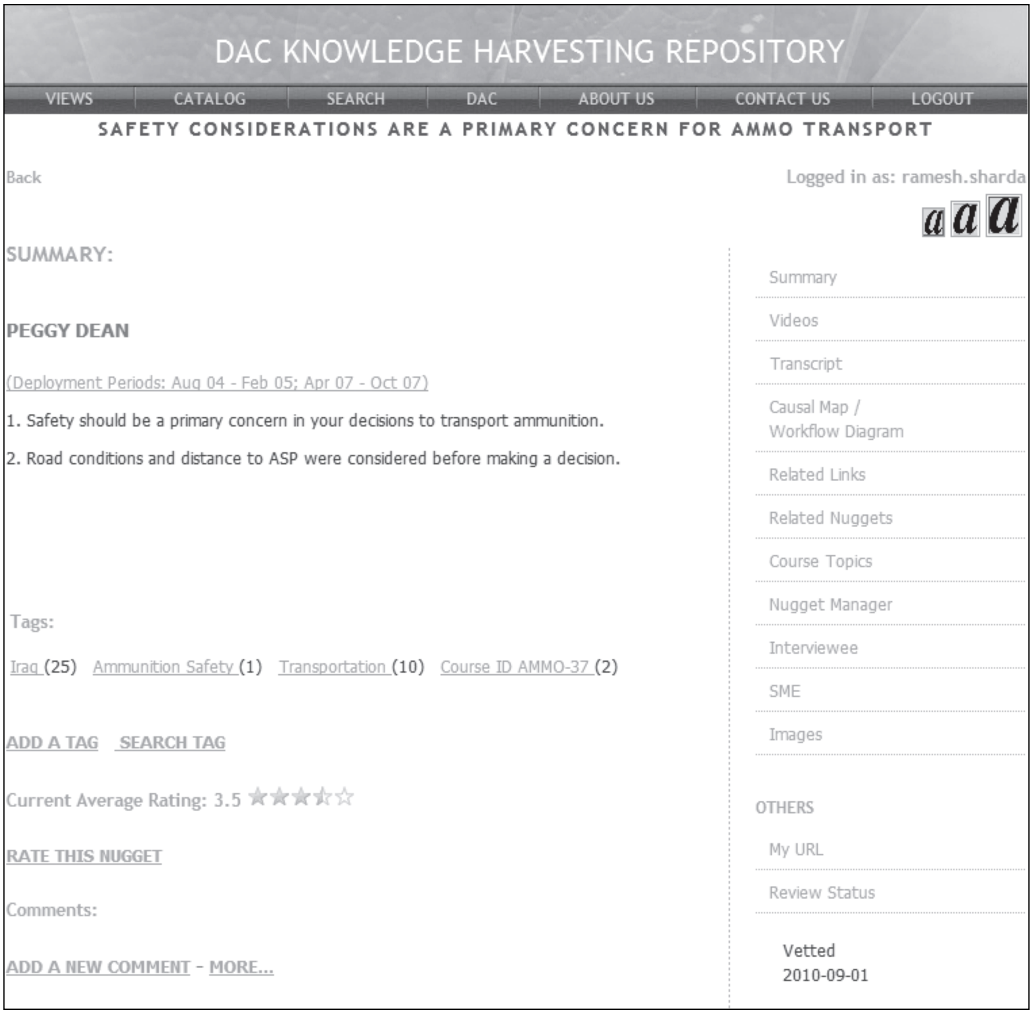

FIGURE 12.2 A Sample Knowledge Nugget.

One of the primary objectives of the ETS is to quickly capture knowledge from the field and incorporate it into the training curriculum. This is accomplished with the My URL feature. This function allows course developers and instructors to use ETS to identify a specific nugget for sharing, and then generate a URL that can be passed directly into a course curriculum and lesson plans. When an instructor clicks on the URL, it brings him/her directly to the KN. As such, the "war story" captured in the nugget becomes the course instructor's war story and provides a real-world decision-making or problem-solving scenario right in the classroom.

The summary page also includes capabilities for users to rate the KN and make any comments about its accuracy. This makes the knowledge nugget a live and continously updated piece of knoweldge. These nuggets can then be sorted on the basis of higher ratings, if so desired. Each nugget intially includes keywords created by the nugget developer. These are presented as tags. A user can also suggest their own tags. These user-specified tags make future searching faster and easier. This brings Web 2.0 concepts of user participation to knowledge management.

In its initial conceptualization, the ETS was supposed to capture the "lesson learned" of the interviews. However, we quickly learned that the ETS process often captures "lessons to be learned." That is, the interviewees often found themselves in situations where they had to improvise and be innovative while deployed. Many of their approaches and solutions are quite admirable, but sometimes they may not be appropriate or suitable for everyone. In light of that finding, a vetting process was developed for the KNs. Each KN is reviewed by recognized subject matter experts (SME). If the SMEs find the approach acceptable, it is noted as "vetted." If guidance for the KN situation already exists, it is identified and added to the related links. The KN is then noted as "doctrine." If the KN has yet to be reviewed, it is noted as "not reviewed." In this way, the user always has an idea of the quality of each KN viewed. Additionally, if the site must be brought down for any reason, the alerts feature is used to relay that information.

The ETS is designed for two primary types of users: DAC instructors and ammunition personnel. As such, a "push/pull" capability was developed. Tech training instructors do not have the time to search the ETS to find those KNs that are related to the courses they teach. To provide some relief to instructors, the KNs are linked to DAC courses and topics, and can be *pushed* to instructors' e-mail accounts as the KNs come online. Instructors can opt-in or opt-out of courses and topics at will, and they can arrange for new KNs to be pushed as often as they like. Ammunition personnel are the other primary users of the ETS. These users need the ability to quickly locate and *pull* KNs related to their immediate knowledge needs. To aid them, the ETS organizes the nuggets in various views and has a robust search engine. These views include courses and topics; interviewee names; chronological; and by user-created tags. The goal of the ETS is to provide the user with the KNs they need in 5 minutes or less.

KNOWLEDGE HARVESTING PROCESS

The knowledge harvesting process began with videotaping interviews with DAC employees regarding their deployment experience. Speech in the interviews, in some cases, was converted manually to text. In other cases, the knowledge harvesting team (hereinafter referred to as the "team") employed voice recognition technologies to convert the speech to text. The text was checked for accuracy and then passed through the text mining division of the team. The text mining group read through the transcript and employed text mining software to extract some preliminary knowledge from the transcript. The text mining process provided a one-sentence summary for the knowledge nugget, which became the knowledge statement, commonly known among the team as the "punchline." The punchline created from the transcripts along with the excerpts, relevant video from the interview, and causal maps make up the entire knowledge nugget. The knowledge nugget is further refined by checking for quality of general appearance, errors in text, and so on.

IMPLEMENTATION AND RESULTS

The ETS system was built as a prototype for demonstration of its potential use at the Defense Ammunition Center. It was built using a MySQL database for the collection of knowledge nuggets and the related content, and PHP and JavaScript as the Web language platform. The system also incorporated necessary security and access control precautions. It was made available to several groups of trainees who really liked using this type of tacit knowledge presentation. The feedback was very positive. However, some internal issues as well as the challenge of having the tacit knowledge be shared as official knowledge resulted in the system being discontinued. However, the application was developed to be more of a general knowledge–sharing system as opposed to just this specific use. The authors are exploring other potential users for this platform.

QUESTIONS FOR THE OPENING VIGNETTE

1. What are the key impediments to the use of knowledge in a knowledge management system?
2. What features are incorporated in a knowledge nugget in this implementation?
3. Where else could such a system be implemented?

WHAT WE CAN LEARN FROM THIS VIGNETTE

Knowledge management initiatives in many organizations have not succeeded. Although many studies have been conducted on this issue and we will learn more about this topic in future sections, two major issues seem to be critical. Compilation of a lot of user-generated information in a large Web compilation by itself does not present the needed information in the right format to the user. Nor does it make it easy to find the right knowledge at the right time. So developing a friendly knowledge presentation format that includes audio, video, text summary, and Web 2.0 features such as tagging, sharing, comments, and ratings makes it more likely that users will actually use the KM content. Second, organizing the knowledge to be visible in specific taxonomies as well as search and enabling the users to tag the content enable this knowledge to be more easily discovered within a knowledge management system.

Sources: Based on our own documents and S. Iyer, R. Sharda, D. Biros, J. Lucca, and U. Shimp, "Organization of Lessons Learned Knowledge: A Taxonomy of Implementation," *International Journal of Knowledge Management*, Vol. 5, No. 3 (2009).

12.2 INTRODUCTION TO KNOWLEDGE MANAGEMENT

Humans learn effectively through stories, analogies, and examples. Davenport and Prusak (1998) argue that knowledge is communicated effectively when it is conveyed with a convincing narrative. Family-run businesses transfer the secrets of business learned through experience to the next generation. Knowledge through experience does not necessarily reside in any business textbook, but the transfer of such knowledge facilitates its profitable use. Nonaka (1991) used the term *tacit knowledge* for the knowledge that exists in the head but not on paper. Tacit knowledge is difficult to capture, manage, and share. He also observes that organizations that use tacit knowledge as a strategic weapon are innovators and leaders in their respective business domains. There is no substitute for the substantial value that tacit knowledge can provide. Therefore, it is necessary to capture and codify tacit knowledge to the greatest extent possible.

In the 2000s knowledge management was considered to be one of the cornerstones of business success. Despite spending billions of dollars on knowledge management both by industry and government, success has been mixed. Usually it is the successful projects that see the limelight. Much research has focused on successful knowledge management initiatives as well as factors that could lead to a successful knowledge management project (Davenport et al., 1998). But a few researchers have presented case studies of knowledge management failures (Chua and Lam, 2005). One of the causes for such failures is that the prospective users of such knowledge cannot easily locate relevant information. Knowledge compiled in a knowledge management system is no good to the organization if it cannot be easily found by the likely end users. On the other hand, although their worth is difficult to measure, organizations recognize the value of their intellectual assets. Fierce global competition drives companies to better use their intellectual assets by transforming themselves into organizations that foster the development and sharing of knowledge. In the next few sections we cover the basic concepts of knowledge management.

Knowledge Management Concepts and Definitions

With roots in **organizational learning** and innovation, the idea of KM is not new (see Ponzi, 2004; and Schwartz, 2006). However, the application of IT tools to facilitate the creation, storage, transfer, and application of previously uncodifiable organizational knowledge is a new and major initiative in many organizations. Successful managers have long used intellectual assets and recognized their value. But these efforts were not systematic, nor did they ensure that knowledge gained was shared and dispersed appropriately for maximum organizational benefit. Knowledge management is a process that helps organizations identify, select, organize, disseminate, and transfer important information and expertise that are part of the organization's memory and that typically reside within the organization in an unstructured manner. **Knowledge management (KM)** is the systematic and active management of ideas, information, and knowledge residing in an organization's employees. The structuring of knowledge enables effective and efficient problem solving, dynamic learning, strategic planning, and decision making. KM initiatives focus on identifying knowledge, explicating it in such a way that it can be shared in a formal manner, and leveraging its value through reuse. The information technologies that make KM available throughout an organization are referred to as *KM systems*.

Through a supportive organizational climate and modern IT, an organization can bring its entire organizational memory and knowledge to bear on any problem, anywhere in the world, and at any time (see Bock et al., 2005). For organizational success, knowledge, as a form of capital, must be exchangeable among persons, and it must be able to grow. Knowledge about how problems are solved can be captured so that KM can promote organizational learning, leading to further knowledge creation.

Knowledge

Knowledge is very distinct from data and information (see Figure 12.3). Data are facts, measurements, and statistics; information is organized or processed data that is timely (i.e., inferences from the data are drawn within the time frame of applicability) and accurate (i.e., with regard to the original data) (Kankanhalli et al., 2005). **Knowledge** is information that is contextual, relevant, and actionable. For example, a map that gives detailed driving directions from one location to another could be considered data. An up-to-the-minute traffic bulletin along the freeway that indicates a traffic slowdown due to construction several miles ahead could be considered information. Awareness of an alternative, back-road route could be considered knowledge. In this case, the map is considered data because it does not contain current relevant information that affects the driving time and conditions from one location to the other. However, having the current conditions as information is useful only if you have knowledge that enables you to avert the construction zone. The implication is that knowledge has strong experiential and reflective elements that distinguish it from information in a given context.

Having knowledge implies that it can be exercised to solve a problem, whereas having information does not carry the same connotation. An ability to act is an integral part of being knowledgeable. For example, two people in the same context with the same information may not have the same ability to use the information to the same degree of success. Hence, there is a difference in the human capability to add value. The differences in ability may be due to different experiences, different training, different perspectives, and other factors. Whereas data, information, and knowledge may all be viewed as assets of an organization, knowledge provides a higher level of meaning about data and information. It conveys meaning and hence tends to be much more valuable, yet more ephemeral.

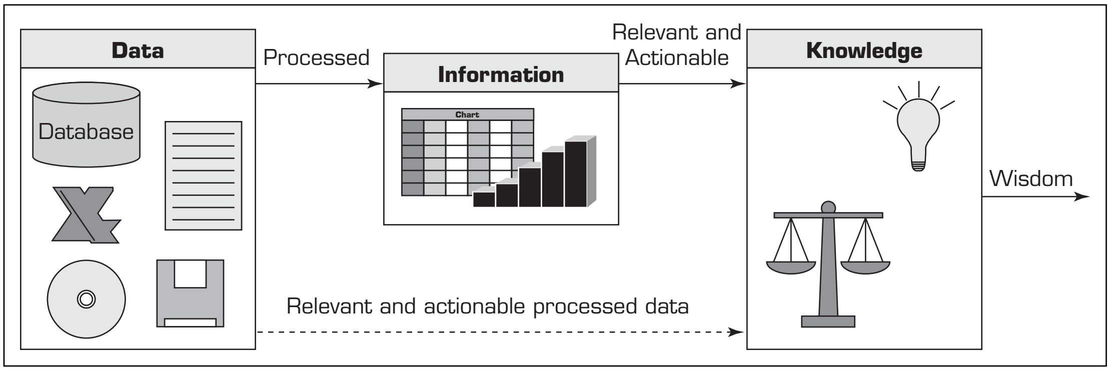

FIGURE 12.3 Relationship Among Data, Information, and Knowledge.

Unlike other organizational assets, knowledge has the following characteristics (see Gray, 1999):

- ***Extraordinary leverage and increasing returns.*** Knowledge is not subject to diminishing returns. When it is used, it is not decreased (or depleted); rather, it is increased (or improved). Its consumers can add to it, thus increasing its value.
- ***Fragmentation, leakage, and the need to refresh.*** As knowledge grows, it branches and fragments. Knowledge is dynamic; it is *information in action*. Thus, an organization must continually refresh its knowledge base to maintain it as a source of competitive advantage.
- ***Uncertain value.*** It is difficult to estimate the impact of an investment in knowledge. There are too many intangible aspects that cannot be easily quantified.
- ***Value of sharing.*** It is difficult to estimate the value of sharing one's knowledge or even who will benefit most from it.

Over the past few decades, the industrialized economy has been going through a transformation from being based on natural resources to being based on intellectual assets (see Alavi, 2000; and Tseng and Goo, 2005). The **knowledge-based economy** is a reality (see Godin, 2006). Rapid changes in the business environment cannot be handled in traditional ways. Firms are much larger today than they used to be, and, in some areas, turnover is extremely high, fueling the need for better tools for collaboration, communication, and knowledge sharing. Firms must develop strategies to sustain competitive advantage by leveraging their intellectual assets for optimal performance. Competing in the globalized economy and markets requires quick response to customer needs and problems. To provide service, managing knowledge is critical for consulting firms spread out over wide geographical areas and for virtual organizations.

There is a vast amount of literature about what knowledge and knowing mean in epistemology (i.e., the study of the nature of knowledge), the social sciences, philosophy, and psychology. Although there is no single definition of what knowledge and KM specifically mean, the business perspective on them is fairly pragmatic. Information as a resource is not always valuable (i.e., information overload can distract from what is important); knowledge as a resource is valuable because it focuses attention back toward what is important (see Carlucci and Schiuma, 2006; and Hoffer et al., 2002). Knowledge implies an implicit understanding and experience that can discriminate between its use and misuse. Over time, information accumulates and decays, whereas knowledge evolves. Knowledge is dynamic in nature. This implies, though, that today's knowledge may well become tomorrow's ignorance if an individual or organization fails to update knowledge as environmental conditions change.

Knowledge evolves over time with experience, which puts connections among new situations and events in context. Given the breadth of the types and applications of knowledge, we adopt the simple and elegant definition that knowledge is information in action.

Explicit and Tacit Knowledge

Polanyi (1958) first conceptualized the difference between an organization's explicit and tacit knowledge. **Explicit knowledge** deals with more objective, rational, and technical knowledge (e.g., data, policies, procedures, software, documents). **Tacit knowledge** is usually in the domain of subjective, cognitive, and experiential learning; it is highly personal and difficult to formalize. Alavi and Leidner (2001) provided a taxonomy (see Table 12.1), where they defined a spectrum of different types of knowledge, going beyond the simple binary classification of explicit versus tacit. However, most KM research has been (and still is) debating over the dichotomous classification of knowledge.

Explicit knowledge comprises the policies, procedural guides, white papers, reports, designs, products, strategies, goals, mission, and core competencies of an enterprise and its IT infrastructure. It is the knowledge that has been codified (i.e., documented) in a form that can be distributed to others or transformed into a process or strategy without requiring interpersonal interaction. For example, a description of how to process a job application would be documented in a firm's human resources policy manual. Explicit knowledge has also been called **leaky knowledge** because of the ease with which it can leave an individual, a document, or an organization due to the fact that it can be readily and accurately documented (see Alavi, 2000).

TABLE 12.1 Taxonomy of Knowledge

Knowledge Type	Definition	Example
Tacit	Knowledge is rooted in actions, experience, and involvement in specific context	Best means of dealing with a specific customer
Cognitive tacit:	Mental models	Individual's belief on cause-effect relationships
Technical tacit:	Know-how applicable to specific work	Surgery skills
Explicit	Articulated, generalized knowledge	Knowledge of major customers in a region
Individual	Created by and inherent in the individual	Insights gained from completed project
Social	Created by and inherent in collective actions of a group	Norms for intergroup communication
Declarative	Know-about	What drug is appropriate for an illness
Procedural	Know-how	How to administer a particular drug
Causal	Know-why	Understanding why the drug works
Conditional	Know-when	Understanding when to prescribe the drug
Relational	Know-with	Understanding how the drug interacts with other drugs
Pragmatic	Useful knowledge for an organization	Best practices, treatment protocols, case analyses, postmortems

Tacit knowledge is the cumulative store of the experiences, mental maps, insights, acumen, expertise, know-how, trade secrets, skillsets, understanding, and learning that an organization has, as well as the **organizational culture** that has embedded in it the past and present experiences of the organization's people, processes, and values. Tacit knowledge, also referred to as *embedded knowledge* (see Tuggle and Goldfinger, 2004), is usually either localized within the brain of an individual or embedded in the group interactions within a department or a branch office. Tacit knowledge typically involves expertise or high skill levels.

Sometimes tacit knowledge could easily be documented but has remained tacit simply because the individual housing the knowledge does not recognize its potential value to other individuals. Other times, tacit knowledge is unstructured, without tangible form, and therefore difficult to codify. It is difficult to put some tacit knowledge into words. For example, an explanation of how to ride a bicycle would be difficult to document explicitly and thus is tacit. Successful transfer or sharing of tacit knowledge usually takes place through associations, internships, apprenticeship, conversations, other means of social and interpersonal interactions, or even simulations (see Robin, 2000). Nonaka and Takeuchi (1995) claimed that intangibles such as insights, intuitions, hunches, gut feelings, values, images, metaphors, and analogies are the often-overlooked assets of organizations. Harvesting these intangible assets can be critical to a firm's bottom line and its ability to meet its goals. Tacit knowledge sharing requires a certain context or situation in order to be facilitated because it is less commonly shared under normal circumstances (see Shariq and Vendelø, 2006).

Historically, management information systems (MIS) departments have focused on capturing, storing, managing, and reporting explicit knowledge. Organizations now recognize the need to integrate both types of knowledge in formal information systems. For centuries, the mentor–apprentice relationship, because of its experiential nature, has been a slow but reliable means of transferring tacit knowledge from individual to individual. When people leave an organization, they take their knowledge with them. One critical goal of knowledge management is to retain the valuable know-how that can so easily and quickly leave an organization. **Knowledge management systems (KMS)** refer to the use of modern IT (e.g., the Internet, intranets, extranets, Lotus Notes, software filters, agents, data warehouses, Web 2.0) to systematize, enhance, and expedite intra- and interfirm KM.

KM systems are intended to help an organization cope with turnover, rapid change, and downsizing by making the expertise of the organization's human capital widely accessible. They are being built, in part, because of the increasing pressure to maintain a well-informed, productive workforce. Moreover, they are built to help large organizations provide a consistent level of customer service.

SECTION 12.2 REVIEW QUESTIONS

1. Define *knowledge management* and describe its purposes.
2. Distinguish between knowledge and data.
3. Describe the knowledge-based economy.
4. Define *tacit knowledge* and *explicit knowledge*.
5. Define *KMS* and describe the capabilities of KMS.

12.3 APPROACHES TO KNOWLEDGE MANAGEMENT

The two fundamental approaches to knowledge management are the process approach and the practice approach (see Table 12.2). We next describe these two approaches as well as hybrid approaches.

TABLE 12.2 The Process and Practice Approaches to Knowledge Management

	Process Approach	Practice Approach
Type of knowledge supported	Explicit knowledge—codified in rules, tools, and processes	Mostly tacit knowledge—unarticulated knowledge not easily captured or codified
Means of transmission	Formal controls, procedures, and standard operating procedures, with heavy emphasis on information technologies to support knowledge creation, codification, and transfer of knowledge	Informal social groups that engage in storytelling and improvisation
Benefits	Provides structure to harness generated ideas and knowledge Achieves scale in knowledge reuse Provides spark for fresh ideas and responsiveness to changing environment	Provides an environment to generate and transfer high-value tacit knowledge
Disadvantages	Fails to tap into tacit knowledge May limit innovation and forces participants into fixed patterns of thinking	Can result in inefficiency Abundance of ideas with no structure to implement them
Role of information technology (IT)	Requires heavy investment in IT to connect people with reusable codified knowledge	Requires moderate investment in IT to facilitate conversations and transfer of tacit knowledge

Source: Compiled from M. Alavi, T. R. Kayworth, and D. E. Leidner, "An Empirical Examination of the Influence of Organizational Culture on Knowledge Management Practices," *Journal of Management Information Systems,* Vol. 22, No. 3, 2006, pp. 191–224.

The Process Approach to Knowledge Management

The **process approach** to knowledge management attempts to codify organizational knowledge through formalized controls, processes, and technologies (see Hansen et al., 1999). Organizations that adopt the process approach may implement explicit policies governing how knowledge is to be collected, stored, and disseminated throughout the organization. The process approach frequently involves the use of IT, such as intranets, data warehousing, knowledge repositories, decision support tools, and groupware to enhance the quality and speed of knowledge creation and distribution in the organization. The main criticisms of the process approach are that it fails to capture much of the tacit knowledge embedded in firms and it forces individuals into fixed patterns of thinking (see Kiaraka and Manning, 2005). This approach is favored by firms that sell relatively standardized products that fill common needs. Most of the valuable knowledge in these firms is fairly explicit because of the standardized nature of the products and services. For example, a kazoo manufacturer has minimal product changes or service needs over the years, and yet there is steady demand and a need to produce the item. In these cases, the knowledge may be typically static in nature.

The Practice Approach to Knowledge Management

In contrast to the process approach, the **practice approach** to knowledge management assumes that a great deal of organizational knowledge is tacit in nature and that formal controls, processes, and technologies are not suitable for transmitting this type of understanding. Rather than build formal systems to manage knowledge, the focus of this approach is to build the social environments or communities of practice necessary to facilitate the sharing of tacit understanding (see Hansen et al., 1999; Leidner et al., 2006; and Wenger and Snyder, 2000). These communities are informal social groups that meet regularly to share ideas, insights, and best practices. This approach is typically adopted by companies that provide highly customized solutions to unique problems. For these firms, knowledge is shared mostly through person-to-person contact. Collaborative computing methods (e.g., group support systems [GSS], e-mail) help people communicate. The valuable knowledge for these firms is tacit in nature, which is difficult to express, capture, and manage. In this case, the environment and the nature of the problems being encountered are extremely dynamic. Because tacit knowledge is difficult to extract, store, and manage, the explicit knowledge that points to how to find the appropriate tacit knowledge (i.e., people contacts, consulting reports) is made available to an appropriate set of individuals who might need it. Consulting firms generally fall into this category. Firms adopting the codification strategy implicitly adopt the network storage model in their initial KMS (see Alavi, 2000).

Hybrid Approaches to Knowledge Management

Many organizations use a hybrid of the process and practice approaches. Early in the development process, when it may not be clear how to extract tacit knowledge from its sources, the practice approach is used so that a repository stores only explicit knowledge that is relatively easy to document. The tacit knowledge initially stored in the repository is contact information about experts and their areas of expertise. Such information is listed so that people in the organization can find sources of expertise (e.g., the process approach). From this start, best practices can eventually be captured and managed so that the **knowledge repository** will contain an increasing amount of tacit knowledge over time. Eventually, a true process approach may be attained. But if the environment changes rapidly, only some of the best practices will prove useful. Regardless of the type of KMS developed, a storage location for the knowledge (i.e., a knowledge repository) of some kind is needed.

Certain highly skilled, research-oriented industries exhibit traits that require nearly equal efforts with both approaches. For example, Koenig (2001) argued that the pharmaceutical firms in which he has worked require about a 50/50 split. We suspect that industries that require both a lot of engineering effort (i.e., how to create products) and heavy-duty research effort (where a large percentage of research is unusable) would fit the 50/50 hybrid category. Ultimately, any knowledge that is stored in a knowledge repository must be reevaluated; otherwise, the repository will become a knowledge landfill.

Knowledge Repositories

A knowledge repository is neither a database nor a knowledge base in the strictest sense of the terms. Rather, a knowledge repository stores knowledge that is often text based and has very different characteristics. It is also referred to as an organizational knowledge base. Do not confuse a knowledge repository with the knowledge base of an expert system. They are very different mechanisms: A knowledge base of an expert system contains knowledge for solving a specific problem. An organizational knowledge base contains all the organizational knowledge.

Capturing and storing knowledge are the goals for a knowledge repository. The structure of the repository is highly dependent on the types of knowledge it stores. The repository can range from simply a list of frequently asked (and obscure) questions and solutions, to a listing of individuals with their expertise and contact information, to detailed best practices for a large organization. Figure 12.4 shows a comprehensive KM architecture designed around an all-inclusive knowledge repository (Delen and

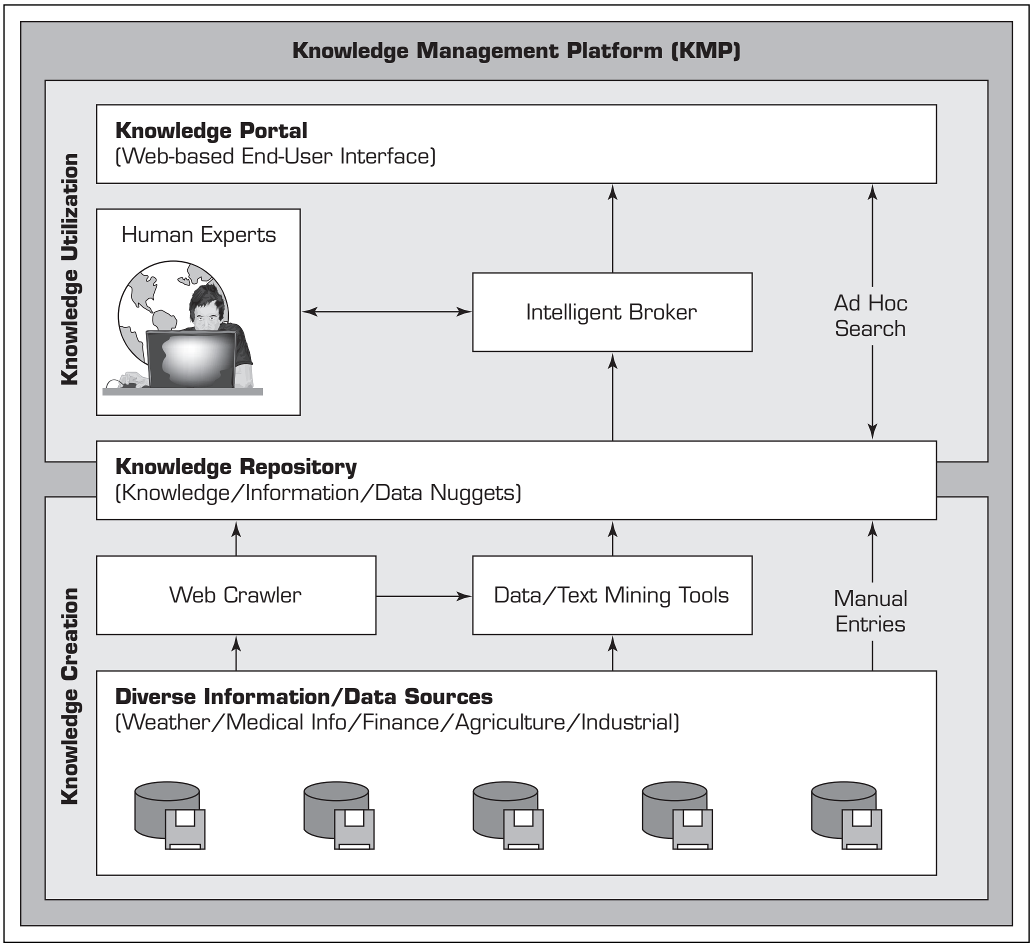

FIGURE 12.4 A Comprehensive View of a Knowledge Repository. *Source:* D. Delen and S. S. Hawamdeh, "A Holistic Framework for Knowledge Discovery and Management," *Communications of the ACM*, Vol. 52, No. 6, 2009, pp. 141–145.

Hawamdeh, 2009). Most knowledge repositories are developed using several different storage mechanisms, depending on the types and amount of knowledge to be maintained and used. Each has strengths and weaknesses when used for different purposes within a KMS. Developing a knowledge repository is not an easy task. The most important aspects and difficult issues are making the contribution of knowledge relatively easy for the contributor and determining a good method for cataloging the knowledge. The users should not be involved in running the storage and retrieval mechanisms of the knowledge repository. Typical development approaches include developing a large-scale Internet-based system or purchasing a formal electronic document management system or a knowledge management suite. The structure and development of the knowledge repository are a function of the specific technology used for the KMS.

SECTION 12.3 REVIEW QUESTIONS

1. Describe the process approach to knowledge management.
2. Describe the practice approach to knowledge management.
3. Why is a hybrid approach to KM desirable?
4. Define *knowledge repository* and describe how to create one.

12.4 INFORMATION TECHNOLOGY (IT) IN KNOWLEDGE MANAGEMENT

The two primary functions of IT in knowledge management are retrieval and communication. IT also extends the reach and range of knowledge use and enhances the speed of knowledge transfer. Networks facilitate collaboration in KM.

The KMS Cycle

A functioning KMS follows six steps in a cycle (see Figure 12.5). The reason for the cycle is that knowledge is dynamically refined over time. The knowledge in a good KMS is never finished because the environment changes over time, and the knowledge must be updated to reflect the changes. The cycle works as follows:

1. ***Create knowledge.*** Knowledge is created as people determine new ways of doing things or develop know-how. Sometimes external knowledge is brought in. Some of these new ways may become best practices.
2. ***Capture knowledge.*** New knowledge must be identified as valuable and be represented in a reasonable way.
3. ***Refine knowledge.*** New knowledge must be placed in context so that it is actionable. This is where human insights (i.e., tacit qualities) must be captured along with explicit facts.
4. ***Store knowledge.*** Useful knowledge must be stored in a reasonable format in a knowledge repository so that others in the organization can access it.
5. ***Manage knowledge.*** Like a library, a repository must be kept current. It must be reviewed to verify that it is relevant and accurate.
6. ***Disseminate knowledge.*** Knowledge must be made available in a useful format to anyone in the organization who needs it, anywhere and anytime.

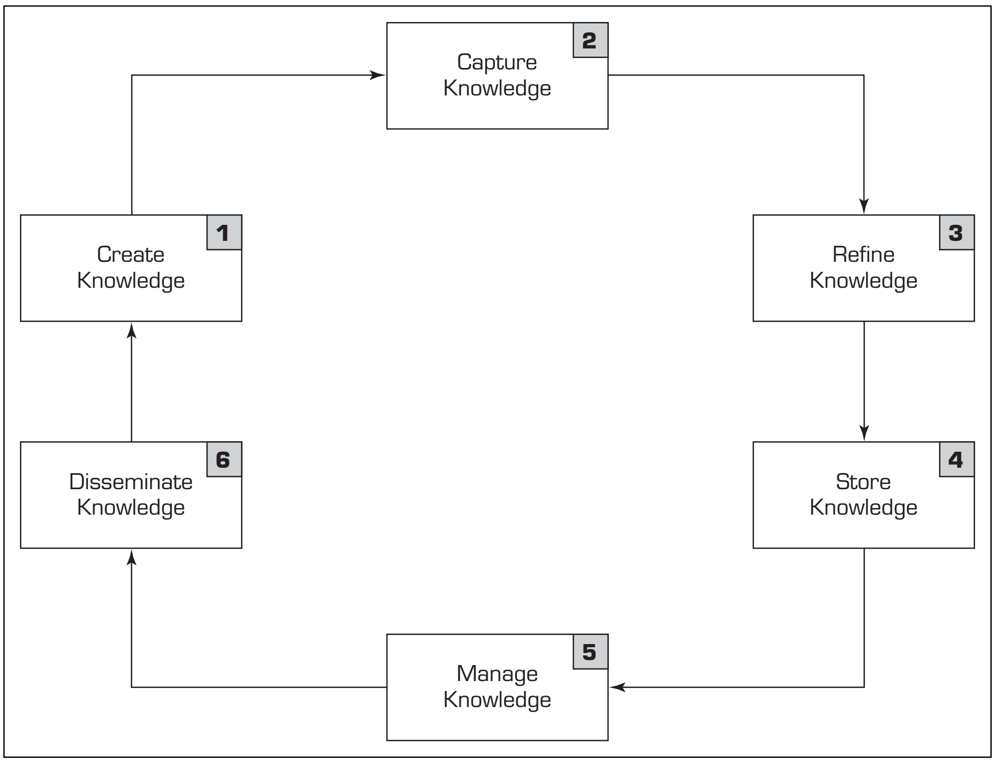

FIGURE 12.5 The Knowledge Management Cycle.

Components of KMS

Knowledge management is more a methodology applied to business practices than a technology or a product. Nevertheless, IT is crucial to the success of every KMS. IT enables knowledge management by providing the enterprise architecture on which it is built. KMS are developed using three sets of technologies: communication, collaboration, and storage and retrieval.

Communication technologies allow users to access needed knowledge and to communicate with each other—especially with experts. E-mail, the Internet, corporate intranets, and other Web-based tools provide communication capabilities. Even fax machines and telephones are used for communication, especially when the practice approach to knowledge management is adopted.

Collaboration technologies (next several sections) provide the means to perform groupwork. Groups can work together on common documents at the same time (i.e., synchronous) or at different times (i.e., asynchronous); they can work in the same place or in different places. Collaboration technologies are especially important for members of a **community of practice** working on knowledge contributions. Other collaborative computing capabilities, such as electronic brainstorming, enhance groupwork, especially for knowledge contribution. Additional forms of groupwork involve experts working with individuals trying to apply their knowledge; this requires collaboration at a fairly high level. Other collaborative computing systems allow an organization to create a virtual space so that individuals can work online anywhere and at any time (see Van de Van, 2005).

Storage and retrieval technologies originally meant using a database management system (DBMS) to store and manage knowledge. This worked reasonably well in the early days for storing and managing most explicit knowledge—and even explicit knowledge about tacit knowledge. However, capturing, storing, and managing tacit knowledge usually requires a different set of tools. Electronic document management systems and specialized storage systems that are part of collaborative computing systems fill this void. These storage systems have come to be known as knowledge repositories.

We describe the relationship between these knowledge management technologies and the Web in Table 12.3.

Technologies That Support Knowledge Management

Several technologies have contributed to significant advances in knowledge management tools. Artificial intelligence, intelligent agents, knowledge discovery in databases, eXtensible Markup Language (XML), and Web 2.0 are examples of technologies that enable advanced functionality of modern KMS and form the basis for future innovations in the knowledge management field. Following is a brief description of how these technologies are used in support of KMS.

ARTIFICIAL INTELLIGENCE In the definition of knowledge management, artificial intelligence (AI) is rarely mentioned. However, practically speaking, AI methods and tools are embedded in a number of KMS, either by vendors or by system developers. AI methods can assist in identifying expertise, eliciting knowledge automatically and semiautomatically, interfacing through natural language processing, and intelligently searching through intelligent agents. AI methods—notably expert systems, neural networks, fuzzy logic, and intelligent agents—are used in KMS to do the following:

- Assist in and enhance searching knowledge (e.g., intelligent agents in Web searches)
- Help establish knowledge profiles of individuals and groups

TABLE 12.3 Knowledge Management Technologies and Web Impacts

Knowledge Management	Web Impacts	Impacts on the Web
Communication	Consistent, friendly graphical user interface (GUI) for client units Improved communication tools Convenient, fast access to knowledge and knowledgeable individuals Direct access to knowledge on servers	Knowledge captured and shared is used in improving communication, communication management, and communication technologies.
Collaboration	Improved collaboration tools Enables anywhere/anytime collaboration Enables collaboration between companies, customers, and vendors Enables document sharing Improved, fast collaboration and links to knowledge sources Makes audio- and videoconferencing a reality, especially for individuals not using a local area network	Knowledge captured and shared is used in improving collaboration, collaboration management, and collaboration technologies (i.e., SharePoint, wiki, GSS).
Storage and retrieval	Consistent, friendly GUI for clients Servers provide for efficient and effective storage and retrieval of knowledge	Knowledge captured and shared is utilized in improving data storage and retrieval systems, database management/ knowledge repository management, and database and knowledge repository technologies.

- Help determine the relative importance of knowledge when it is contributed to and accessed from the knowledge repository
- Scan e-mail, documents, and databases to perform knowledge discovery, determine meaningful relationships, glean knowledge, or induce rules for expert systems
- Identify patterns in data (usually through neural networks)
- Forecast future results by using existing knowledge
- Provide advice directly from knowledge by using neural networks or expert systems
- Provide a natural language or voice command–driven user interface for a KMS

WEB 2.0 The Web has evolved from a tool for disseminating information and conducting business to a platform for facilitating new ways of information sharing, collaboration, and communication in the digital age. A new vocabulary has emerged, as mashups, social networks, media-sharing sites, RSS, blogs, and wikis have come to characterize the genre of interactive applications collectively known as Web 2.0. These technologies have given knowledge management a strong boost by making it easy and natural for everyone to share knowledge. In some ways this has occurred to the point of perhaps making the term *knowledge management* almost redundant. Indeed, Davenport (2008) characterized Web 2.0 (and its reflection to the enterprise world, Enterprise 2.0) as "new, new knowledge management." One of the bottlenecks for knowledge management practices has been the difficulty for nontechnical people to natively share their knowledge. Therefore, the ultimate value of Web 2.0 is its ability to foster greater responsiveness, better knowledge capture and sharing, and ultimately, more effective collective intelligence.

SECTION 12.4 REVIEW QUESTIONS

1. Describe the KMS cycle.
2. List and describe the components of KMS.
3. Describe how AI and intelligent agents support knowledge management.
4. Relate Web 2.0 to knowledge management

Web 2.0 also engenders collaborative inputs. Whether these collaborations are for knowledge management activities or other organizational decision making, the overall principles are the same. We study some basic collaborative mechanisms and systems in the next several sections.

12.5 MAKING DECISIONS IN GROUPS: CHARACTERISTICS, PROCESS, BENEFITS, AND DYSFUNCTIONS

Managers and other knowledge workers continuously make decisions, design and manufacture products, develop policies and strategies, create software systems, and so on. When people work in groups (i.e., teams), they perform groupwork (i.e., teamwork). **Groupwork** refers to work done by two or more people together.

Characteristics of Groupwork

The following are some of the functions and characteristics of groupwork:

- A group performs a task (sometimes decision making, sometimes not).
- Group members may be located in different places.
- Group members may work at different times.
- Group members may work for the same organization or for different organizations.
- A group can be permanent or temporary.
- A group can be at one managerial level or span several levels.
- It can create synergy (leading to process and task gains) or conflict.
- It can generate productivity gains and/or losses.
- The task may have to be accomplished very quickly.
- It may be impossible or too expensive for all the team members to meet in one place, especially when the group is called for emergency purposes.
- Some of the needed data, information, or knowledge may be located in many sources, some of which may be external to the organization.
- The expertise of no team members may be needed.
- Groups perform many tasks; however, groups of managers and analysts frequently concentrate on decision making.
- The decisions made by a group are easier to implement if supported by all (or at least most) members.

The Group Decision-Making Process

Even in hierarchical organizations, decision making is usually a shared process. A group may be involved in a decision or in a decision-related task, such as creating a short list of acceptable alternatives or choosing criteria for evaluating alternatives and prioritizing them. The following activities and processes characterize meetings:

- The decision situation is important, so it is advisable to make it in a group in a meeting.
- A meeting is a joint activity engaged in by a group of people typically of equal or nearly equal status.
- The outcome of a meeting depends partly on the knowledge, opinions, and judgments of its participants and the support they give to the outcome.
- The outcome of a meeting depends on the composition of the group and on the decision-making process the group uses.
- Differences in opinions are settled either by the ranking person present or, often, through negotiation or arbitration.
- The members of a group can be in one place, meeting face-to-face, or they can be a **virtual team**, in which case they are in different places while in a meeting.
- The process of group decision making can create benefits as well as dysfunctions.

The Benefits and Limitations of Groupwork

Some people endure meetings (the most common form of groupwork) as a necessity; others find them to be a waste of time. Many things can go wrong in a meeting. Participants may not clearly understand their goals, they may lack focus, or they may have hidden agendas. Many participants may be afraid to speak up, while a few may dominate the discussion. Misunderstandings occur through different interpretations of language, gesture, or expression. Table 12.4 provides a comprehensive list of factors that can hinder the effectiveness of a meeting (Nunamaker, 1997). Besides being challenging, teamwork is also expensive. A meeting of several managers or executives may cost thousands of dollars per hour in salary costs alone.

Groupwork may have both potential benefits (process gains) and potential drawbacks (process losses). **Process gains** are the benefits of working in groups. The unfortunate dysfunctions that may occur when people work in groups are called **process losses**. Examples of each are listed in Technology Insights 12.1.

TABLE 12.4 Difficulties Associated with Groupwork

• Waiting to speak	• Wrong composition of people
• Dominating the discussion	• Groupthink
• Fear of speaking	• Poor grasp of problem
• Fear of being misunderstood	• Ignored alternatives
• Inattention	• Lack of consensus
• Lack of focus	• Poor planning
• Inadequate criteria	• Hidden agendas
• Premature decisions	• Conflicts of interest
• Missing information	• Inadequate resources
• Distractions	• Poorly defined goals
• Digressions	

TECHNOLOGY INSIGHTS 12.1 Benefits of Working in Groups and Dysfunctions of the Group Process

Benefits of Working in Groups (Process Gains)	Dysfunctions of the Group Process (Process Losses)
• It provides learning. Groups are better than individuals at understanding problems.	• Social pressures of conformity may result in **groupthink** (i.e., people begin to think alike and do not tolerate new ideas; they yield to *conformance pressure*).
• People readily take ownership of problems and their solutions. They take responsibility.	• It is a time-consuming, slow process (i.e., only one member can speak at a time).
• Group members have their egos embedded in the decision, so they are committed to the solution.	• There can be lack of coordination of the meeting and poor meeting planning.
• Groups are better than individuals at catching errors.	• Inappropriate influences (e.g., domination of time, topic, or opinion by one or few individuals; fear of contributing because of the possibility of *flaming*).
• A group has more *information* (i.e., knowledge) than any one member. Group members can combine their knowledge to create new knowledge. More and more creative alternatives for problem solving can be generated, and better solutions can be derived (e.g., through *stimulation*).	• There can be a tendency for group members to either dominate the agenda or rely on others to do most of the work (free-riding).
• A group may produce *synergy* during problem solving. The effectiveness and/or quality of groupwork can be greater than the sum of what is produced by independent individuals.	• Some members may be afraid to speak up.
• Working in a group may stimulate the creativity of the participants and the process.	• There can be a tendency to produce compromised solutions of poor quality.
• A group may have better and more precise communication working together.	• There is often nonproductive time (e.g., socializing, preparing, waiting for latecomers; i.e., *air-time fragmentation*).
• Risk propensity is balanced. Groups moderate high-risk takers and encourage conservatives.	• There can be a tendency to repeat what has already been said (because of failure to remember or process).
	• Meeting costs can be high (e.g., travel, participation time spent).
	• There can be incomplete or inappropriate use of information.
	• There can be too much information (i.e., information overload).
	• There can be incomplete or incorrect task analysis.

(*Continued*)

Benefits of Working in Groups (Process Gains)	Dysfunctions of the Group Process (Process Losses)
	• There can be inappropriate or incomplete representation in the group. • There can be attention blocking. • There can be concentration blocking.

SECTION 12.5 REVIEW QUESTIONS

1. Define *groupwork.*
2. List five characteristics of groupwork.
3. Describe the process of a group meeting for decision making.

12.6 SUPPORTING GROUPWORK WITH COMPUTERIZED SYSTEMS

When people work in teams, especially when the members are in different locations and may be working at different times, they need to communicate, collaborate, and access a diverse set of information sources in multiple formats. This makes meetings, especially virtual ones, complex, with a greater chance for process losses. It is important to follow a certain process for conducting meetings.

Groupwork may require different levels of coordination (Nunamaker, 1997). Sometimes a group may operate at the individual work level, with members making individual efforts that require no coordination. As with a team of sprinters representing a country participating in a 100-meter dash, group productivity is simply the best of the individual results. Other times group members may interact at the coordinated work level. At this level, as with a team in a relay race, the work requires careful coordination between otherwise independent individual efforts. Sometimes a team may operate at the concerted work level. As in a rowing race, teams working at this level must make a continuous concerted effort to be successful. Different mechanisms support groupwork at different levels of coordination.

It is almost trite to say that all organizations, small and large, are using some computer-based communication and collaboration methods and tools to support people working in teams or groups. From e-mails to mobile phones and SMS as well as conferencing technologies, such tools are an indispensable part of one's work life today. We next highlight some related technologies and applications.

An Overview of Group Support Systems (GSS)

For groups to collaborate effectively, appropriate communication methods and technologies are needed. The Internet and its derivatives (i.e., intranets and extranets) are the infrastructures on which much communication for collaboration occurs. The Web supports intra- and interorganizational collaborative decision making through collaboration tools and access to data, information, and knowledge from inside and outside the organization.

Intra-organizational networked decision support can be effectively supported by an intranet. People within an organization can work with Internet tools and procedures through enterprise information portals. Specific applications can include important internal documents and procedures, corporate address lists, e-mail, tool access, and software distribution.

An *extranet* links people in different organizations. For example, **covisint.com** focuses on providing such collaborative mechanisms in diverse industries such as manufacturing, healthcare, and energy. Other extranets are used to link teams together to design products when several different suppliers must collaborate on design and manufacturing techniques.

Computers have been used for several decades to facilitate groupwork and group decision making. Lately, collaborative tools have received even greater attention due to their increased capabilities and ability to save money (e.g., on travel cost) as well as their ability to expedite decision making. Such computerized tools are called groupware.

Groupware

Many computerized tools have been developed to provide group support. These tools are called **groupware** because their primary objective is to support groupwork. Groupware tools can support decision making directly or indirectly, and they are described in the remainder of this chapter. For example, generating creative solutions to problems is a direct support. Some e-mail programs, chat rooms, instant messaging (IM), and teleconferencing provide indirect support.

Groupware provides a mechanism for team members to share opinions, data, information, knowledge, and other resources. Different computing technologies support groupwork in different ways, depending on the purpose of the group, the task, and the time/place category in which the work occurs.

Time/Place Framework

The effectiveness of a collaborative computing technology depends on the location of the group members and on the time that shared information is sent and received. DeSanctis and Gallupe (1987) proposed a framework for classifying IT communication support technologies. In this framework, communication is divided into four cells, which are shown together with representative computerized support technologies in Figure 12.6. The four cells are organized along two dimensions—time and place.

When information is sent and received almost simultaneously, the communication is **synchronous (real time)**. Telephones, IM, and face-to-face meetings are examples of synchronous communication. **Asynchronous** communication occurs when the receiver

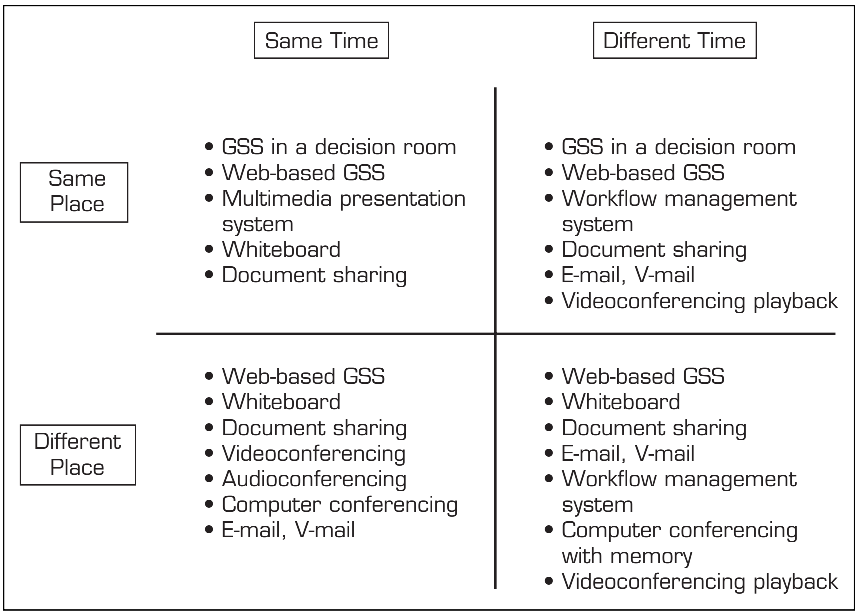

FIGURE 12.6 The Time/Place Framework for Groupwork.

gets the information at a different time than it was sent, such as in e-mail. The senders and the receivers can be in the same room or in different places.

As shown in Figure 12.6, time and place combinations can be viewed as a four-cell matrix, or framework. The four cells of the framework are as follows:

- ***Same time/same place.*** Participants meet face-to-face in one place at the same time, as in a traditional meeting or decision room. This is still an important way to meet, even when Web-based support is used, because it is sometimes critical for participants to leave the office to eliminate distractions.
- ***Same time/different place.*** Participants are in different places, but they communicate at the same time (e.g., with videoconferencing).
- ***Different time/same place.*** People work in shifts. One shift leaves information for the next shift.
- ***Different time/different place (any time, any place).*** Participants are in different places, and they also send and receive information at different times. This occurs when team members are traveling, have conflicting schedules, or work in different time zones.

Groups and groupwork (also known as *teams* and *teamwork*) in organizations are proliferating. Consequently, groupware continues to evolve to support effective groupwork, mostly for communication and collaboration.

SECTION 12.6 REVIEW QUESTIONS

1. Why do companies use computers to support groupwork?
2. Describe the components of the time/place framework.

12.7 TOOLS FOR INDIRECT SUPPORT OF DECISION MAKING

A large number of tools and methodologies are available to facilitate e-collaboration, communication, and decision support. The following sections present the major tools that support decision making indirectly.

Groupware Tools

Groupware products provide a way for groups to share resources and opinions. Groupware implies the use of networks to connect people, even if they are in the same room. Many groupware products are available on the Internet or an intranet to enhance the collaboration of a large number of people. The features of groupware products that support commutation, collaboration, and coordination are listed in Table 12.5. What follows are brief definitions of some of those features.

SYNCHRONOUS VERSUS ASYNCHRONOUS PRODUCTS Notice that the features in Table 12.5 may be synchronous, meaning that communication and collaboration are done in real time, or asynchronous, meaning that communication and collaboration are done by the participants at different times. Web conferencing and IM as well as Voice over IP (VoIP) are associated with synchronous mode. Methods that are associated with asynchronous modes include e-mail, wikilogs, and online workspaces, where participants can collaborate, for example, on joint designs or projects, but work at different times. Google Drive (**drive.google.com**) and Microsoft SharePoint (**http://office.microsoft.com/en-us/SharePoint/collaboration-software-SharePoint-FX103479517.aspx**) allow users to set up online workspaces for storing, sharing, and collaboratively working on different types of documents.

TABLE 12.5 Groupware Products and Features

General (Can Be Either Synchronous or Asynchronous)

- Built-in e-mail, messaging system
- Browser interface
- Joint Web-page creation
- Sharing of active hyperlinks
- File sharing (graphics, video, audio, or other)
- Built-in search functions (by topic or keyword)
- Workflow tools
- Use of corporate portals for communication, collaboration, and search
- Shared screens
- Electronic decision rooms
- Peer-to-peer networks

Synchronous (Same Time)

- Instant messaging (IM)
- Videoconferencing, multimedia conferencing
- Audioconferencing
- Shared whiteboard, smart whiteboard
- Instant video
- Brainstorming
- Polling (voting), and other decision support (consensus builder, scheduler)

Asynchronous (Different Times)

- Workspaces
- Threaded discussions
- Users can receive/send e-mail, SMS
- Users can receive activity notification alerts via e-mail or SMS
- Users can collapse/expand discussion threads
- Users can sort messages (by date, author, or read/unread)
- Auto responder
- Chat session logs
- Bulletin boards, discussion groups
- Use of blogs, wikis, and wikilogs
- Collaborative planning and/or design tools
- Use of bulletin boards

Companies such as **Dropbox.com** provide an easy way to share documents. Of course, similar systems are evolving for consumer and home use such as photo sharing (e.g., Picasa, Flicker, Facebook).

Groupware products are either stand-alone products that support one task (such as videoconferencing) or integrated kits that include several tools. In general, groupware technology products are fairly inexpensive and can easily be incorporated into existing information systems.

VIRTUAL MEETING SYSTEMS The advancement of Web-based systems opens the door for improved, electronically supported **virtual meetings**, where members are in different locations and even in different countries. For example, online meetings and presentation tools are provided by **webex.com, gotomeeting.com, Adobe.com, Skype.com**, and many others. Microsoft Office also includes a built-in virtual meeting capability. These systems feature Web seminars (popularly called Webinars), screen sharing, audioconferencing, videoconferencing, polling, question–answer sessions, and so on. Even mobile phones now have sufficient interaction capabilities to allow live meetings through applications such as Facetime.

Groupware

Although many of the technologies that enable group decision support are merging in common office productivity software tools such as Microsoft Office, it is instructive to learn about one specific software that illustrates some unique capabilities of groupware. GroupSystems (**groupsystems.com**) MeetingRoom was one of the first comprehensive same time/same place electronic meeting packages. The follow-up product, GroupSystems OnLine, offered similar capabilities, and it ran in asynchronous mode (anytime/anyplace) over the Web (MeetingRoom ran only over a local area network [LAN]). GroupSystems' latest product is ThinkTank, which is a suite of tools that significantly shortens cycle time for brainstorming, strategic planning, product development, problem solving, requirements gathering, risk assessments, team decision makings, and other collaborations. ThinkTank moves face-to-face or virtual teams through customizable processes toward their goals faster and more effectively than its predecessors. ThinkTank offers the following capabilities:

- ThinkTank builds in the discipline of an agenda, efficient participation, workflow, prioritization, and decision analysis.
- ThinkTank's anonymous brainstorming for ideas and comments is an ideal way to capture the participants' creativity and experience.
- ThinkTank Web 2.0's enhanced user interface ensures that participants do not need prior training to join, so they can focus 100 percent on solving problems and making decisions.
- With ThinkTank, all of the knowledge shared by participants is captured and saved in documents and spreadsheets and automatically converted to the meeting minutes and made available to all participants at the end of the session.

Another specialized product is eRoom (now owned by EMC/Documentum at **http://www.emc.com/enterprise-content-management/centerstage.htm**). This comprehensive Web-based suite of tools can support a variety of collaboration scenarios. Yet another product is Team Expert Choice (Comparion), which is an add-on product for Expert Choice (**expertchoice.com**). It has limited decision support capabilities, mainly supporting one-room meetings, but focuses on developing a model and process for decision making using the analytic hierarchy process that was covered in Chapter 9.

Collaborative Workflow

Collaborative workflow refers to software products that address project-oriented and collaborative types of processes. They are administered centrally yet are capable of being accessed and used by workers from different departments and even from different physical locations. The goal of collaborative workflow tools is to empower knowledge workers. The focus of an enterprise solution for collaborative workflow is on allowing workers to communicate, negotiate, and collaborate within an integrated environment. Some leading vendors of collaborative workflow applications are Lotus, EpicData, FileNet, and Action Technologies.

Web 2.0

The term *Web 2.0* refers to what is perceived to be the second generation of Web development and Web design. It is characterized as facilitating communication, information sharing, interoperability, user-centered design, and collaboration on the World Wide Web. It has led to the development and evolution of Web-based communities, hosted services, and novel Web applications. Example Web 2.0 applications include social-networking sites (e.g., LinkedIn, Facebook), video-sharing sites (e.g., YouTube, Flickr, Vimeo), wikis, blogs, mashups, and folksonomies.

Web 2.0 sites typically include the following features/techniques, identified by the acronym SLATES:

- ***Search.*** The ease of finding information through keyword search.
- ***Links.*** Ad hoc guides to other relevant information.
- ***Authoring.*** The ability to create content that is constantly updated by multiple users. In wikis, the content is updated in the sense that users undo and redo each other's work. In blogs, content is updated in that posts and comments of individuals are accumulated over time.
- ***Tags.*** Categorization of content by creating tags. Tags are simple, one-word, user-determined descriptions to facilitate searching and avoid rigid, premade categories.
- ***Extensions.*** Powerful algorithms leverage the Web as an application platform as well as a document server.
- ***Signals.*** RSS technology is used to rapidly notify users of content changes.

Wikis

A **wiki** is a piece of server software available at a Web site that allows users to freely create and edit Web page content through a Web browser. (The term *wiki* means "quick" or "to hasten" in the Hawaiian language; e.g., "Wiki Wiki" is the name of the shuttle bus at Honolulu International Airport.) A wiki supports hyperlinks and has a simple text syntax for creating new pages and cross-links between internal pages on-the-fly. It is especially suited for collaborative writing.

Wikis are unusual among group communication mechanisms in that they allow the organization of the contributions to be edited as well as the content itself. The term *wiki* also refers to the collaborative software that facilitates the operation of a wiki Web site.

A wiki enables documents to be written collectively in a very simple markup, using a Web browser. A single page in a wiki is referred to as a "wiki page," and the entire body of pages, which are usually highly interconnected via hyperlinks, is "the wiki"; in effect, it is a very simple, easy-to-use database. For further details, see **en.wikipedia.org/wiki/Wiki** and **wiki.org**.

Collaborative Networks

Traditionally, collaboration took place among supply chain members, frequently those that were close to each other (e.g., a manufacturer and its distributor, a distributor and a retailer). Even if more partners were involved, the focus was on the optimization of information and product flow between existing nodes in the traditional supply chain. Advanced approaches, such as collaborative planning, forecasting, and replenishment, do not change this basic structure.

Traditional collaboration results in a vertically integrated supply chain. However, Web technologies can fundamentally change the shape of the supply chain, the number of players in it, and their individual roles. In a collaborative network, partners at any point in the network can interact with each other, bypassing traditional partners. Interaction may occur among several manufacturers or distributors, as well as with new players, such as software agents that act as aggregators, business-to-business (B2B) exchanges, or logistics providers.

SECTION 12.7 REVIEW QUESTIONS

1. List the major groupware tools and divide them into synchronous and asynchronous types.
2. Identify specific tools for Web conferencing and their capabilities.
3. Define *wiki* and *wikilog*.
4. Define *collaborative hub*.

12.8 DIRECT COMPUTERIZED SUPPORT FOR DECISION MAKING: FROM GROUP DECISION SUPPORT SYSTEMS TO GROUP SUPPORT SYSTEMS

Decisions are made at many meetings, some of which are called in order to make one specific decision. For example, the federal government meets periodically to decide on the short-term interest rate. Directors may be elected at shareholder meetings, organizations allocate budgets in meetings, a company decides on which candidate to hire, and so on. Although some of these decisions are complex, others can be controversial, as in resource allocation by a city government. Process gains and dysfunctions can be significantly large in such situations; therefore, computerized support has often been suggested to mitigate these complexities. These computer-based support systems have appeared in the literature under different names, including *group decision support systems* (GDSS), *group support systems* (GSS), *computer-supported collaborative work* (CSCW), and *electronic meeting systems* (EMS). These systems are the subject of this section.

Group Decision Support Systems (GDSS)

During the 1980s, researchers realized that computerized support to managerial decision making needed to be expanded to groups because major organizational decisions are made by groups such as executive committees, special task forces, and departments. The result was the creation of group decision support systems (see Powell et al., 2004).

A **group decision support system (GDSS)** is an interactive computer-based system that facilitates the solution of semistructured or unstructured problems by a group of decision makers. The goal of GDSS is to improve the productivity of decision-making meetings by speeding up the decision-making process and/or by improving the quality of the resulting decisions.

The following are the major characteristics of a GDSS:

- Its goal is to support the process of group decision makers by providing automation of subprocesses, using information technology tools.
- It is a specially designed information system, not merely a configuration of already-existing system components. It can be designed to address one type of problem or a variety of group-level organizational decisions.
- It encourages generation of ideas, resolution of conflicts, and freedom of expression. It contains built-in mechanisms that discourage development of negative group behaviors, such as destructive conflict, miscommunication, and groupthink.

The first generation of GDSS was designed to support face-to-face meetings in a decision room. Today, support is provided mostly over the Web to virtual groups. The group can meet at the same time or at different times by using e-mail, sending documents, and reading transaction logs. GDSS is especially useful when controversial decisions have to be made (e.g., resource allocation, determining which individuals to lay off). GDSS applications require a facilitator when done in one room or a coordinator or leader when done using virtual meetings.

GDSS can improve the decision-making process in various ways. For one, GDSS generally provide structure to the planning process, which keeps the group on track, although some applications permit the group to use unstructured techniques and methods for **idea generation**. In addition, GDSS offer rapid and easy access to external and stored information needed for decision making. GDSS also support parallel processing of information and idea generation by participants and allow asynchronous computer discussion. They make possible larger meetings that would otherwise be unmanageable;

having a larger group means that more complete information, knowledge, and skills will be represented in the meeting. Finally, voting can be anonymous, with instant results, and all information that passes through the system can be recorded for future analysis (producing *organizational memory*).

Initially, GDSS were limited to face-to-face meetings. To provide the necessary technology, a special facility (i.e., room) was created. Also, groups usually had a clearly defined, narrow task, such as allocation of scarce resources or prioritization of goals in a long-range plan.

Over time, it became clear that support teams' needs were broader than that supported by GDSS. Furthermore, it became clear that what was really needed was support for virtual teams, both in different place/same time and different place/different time situations. Also, it became clear that teams needed indirect support in most decision-making cases (e.g., help in searching for information or collaboration) rather than direct support for the decision making. Although GDSS expanded to virtual team support, they were unable to meet all the other needs. Thus, a broader term, *GSS*, was created. We use the terms interchangeably in this book.

Group Support Systems

A **group support system (GSS)** is any combination of hardware and software that enhances groupwork either in direct or indirect support of decision making. GSS is a generic term that includes all forms of collaborative computing. GSS evolved after information technology researchers recognized that technology could be developed to support the many activities normally occurring at face-to-face meetings (e.g., idea generation, consensus building, anonymous ranking).

A complete GSS is still considered a specially designed information system, but since the mid-1990s many of the special capabilities of GSS have been embedded in standard productivity tools. For example, Microsoft Office can embed the Lync tool for Web conferences. Most GSS are easy to use because they have a Windows-based graphical user interface (GUI) or a Web browser interface. Most GSS are fairly general and provide support for activities such as idea generation, conflict resolution, and voting. Also, many commercial products have been developed to support only one or two aspects of teamwork (e.g., videoconferencing, idea generation, screen sharing, wikis).

GSS settings range from a group meeting at a single location for solving a specific problem to virtual meetings conducted in multiple locations and held via telecommunication channels for the purpose of addressing a variety of problem types. Continuously adopting new and improved methods, GSS are building up their capabilities to effectively operate in asynchronous as well as synchronous modes.

How GDSS (or GSS) Improve Groupwork

The goal of GSS is to provide support to meeting participants to improve the productivity and effectiveness of meetings by streamlining and speeding up the decision-making process (i.e., efficiency) or by improving the quality of the results (i.e., effectiveness). GSS attempts to increase process and task gains and decrease process and task losses. Overall, GSS have been successful in doing just that (see Holt, 2002); however, some process and task gains may decrease, and some process and task losses may increase. Improvement is achieved by providing support to group members for the generation and exchange of ideas, opinions, and preferences. Specific features such as **parallelism** (i.e., the ability of participants in a group to work simultaneously on a task, such as brainstorming or voting) and anonymity produce this improvement. The following are some specific GDSS support activities:

- GDSS support parallel processing of information and idea generation (parallelism).
- GDSS enable the participation of larger groups with more complete information, knowledge, and skills.
- GDSS permit the group to use structured or unstructured techniques and methods.
- GDSS offer rapid, easy access to external information.
- GDSS allow parallel computer discussions.
- GDSS help participants frame the big picture.
- Anonymity allows shy people to contribute to the meeting (i.e., get up and do what needs to be done).
- Anonymity helps prevent aggressive individuals from driving a meeting.
- GDSS provide for multiple ways to participate in instant, anonymous voting.
- GDSS provide structure for the planning process to keep the group on track.
- GDSS enable several users to interact simultaneously (i.e., conferencing).
- GDSS record all information presented at a meeting (i.e., **organizational memory**).

For GSS success stories, look for sample cases at vendors' Web sites. As you will see, in many of these cases, collaborative computing led to dramatic process improvements and cost savings.

Facilities for GDSS

There are three options for deploying GDSS/GSS technology: (1) as a special-purpose decision room, (2) as a multiple-use facility, and (3) as Internet- or intranet-based groupware, with clients running wherever the group members are.

DECISION ROOMS The earliest GDSS were installed in expensive, customized, special-purpose facilities called **decision rooms** (or electronic meeting rooms) with PCs and large public screens at the front of each room. The original idea was that only executives and high-level managers would use the facility. The software in a special-purpose electronic meeting room usually runs over a LAN, and these rooms are fairly plush in their furnishings. Electronic meeting rooms can be constructed in different shapes and sizes. A common design includes a room equipped with 12 to 30 networked PCs, usually recessed into the desktop (for better participant viewing). A server PC is attached to a large-screen projection system and connected to the network to display the work at individual workstations and aggregated information from the facilitator's workstation. Breakout rooms equipped with PCs connected to the server, where small subgroups can consult, are sometimes located adjacent to the decision room. The output from the subgroups can also be displayed on the large public screen.

INTERNET-/INTRANET-BASED SYSTEMS Since the late 1990s, the most common approach to GSS facilities has been to use Web- or intranet-based groupware that allows group members to work from any location at any time (e.g., WebEx, GotoMeeting, Adobe Connect, Microsoft Lync, GroupSystems). This groupware often includes audioconferencing and videoconferencing. The availability of relatively inexpensive groupware (for purchase or for subscription), combined with the power and low cost of computers and mobile devices, makes this type of system very attractive.

SECTION 12.8 REVIEW QUESTIONS

1. Define *GDSS* and list the limitations of the initial GDSS software.
2. Define *GSS* and list its benefits.

3. List process gain improvements made by GSS.
4. Define *decision room*.
5. Describe Web-based GSS.

This chapter has served to provide a relatively quick overview of knowledge management and collaborative systems, two movements that were really prominent in the past 20 years but have now been subsumed by other technologies for information sharing and decision making. It helps to see where the roots of many of the technologies today might have come from, although the names may have changed.

Chapter Highlights

- Knowledge is different from information and data. Knowledge is information that is contextual, relevant, and actionable.
- Knowledge is dynamic in nature. It is information in action.
- Tacit (i.e., unstructured, sticky) knowledge is usually in the domain of subjective, cognitive, and experiential learning; explicit (i.e., structured, leaky) knowledge deals with more objective, rational, and technical knowledge, and it is highly personal and difficult to formalize.
- Organizational learning is the development of new knowledge and insights that have the potential to influence behavior.
- The ability of an organization to learn, develop memory, and share knowledge is dependent on its culture. Culture is a pattern of shared basic assumptions.
- Knowledge management is a process that helps organizations identify, select, organize, disseminate, and transfer important information and expertise that typically reside within the organization in an unstructured manner.
- The knowledge management model involves the following cyclical steps: create, capture, refine, store, manage, and disseminate knowledge.
- Two knowledge management approaches are the process approach and the practice approach.
- Standard knowledge management initiatives involve the creation of knowledge bases, active process management, knowledge centers, collaborative technologies, and knowledge webs.
- A KMS is generally developed using three sets of technologies: communication, collaboration, and storage.
- A variety of technologies can make up a KMS, including the Internet, intranets, data warehousing, decision support tools, and groupware. Intranets are the primary vehicles for displaying and distributing knowledge in organizations.
- People collaborate in their work (called *groupwork*). Groupware (i.e., collaborative computing software) supports groupwork.
- Group members may be in the same organization or may span organizations; they may be in the same location or in different locations; they may work at the same time or at different times.
- The time/place framework is a convenient way to describe the communication and collaboration patterns of groupwork. Different technologies can support different time/place settings.
- Working in groups may result in many benefits, including improved decision making.
- Communication can be synchronous (i.e., same time) or asynchronous (i.e., sent and received in different times).
- Groupware refers to software products that provide collaborative support to groups (including conducting meetings).
- Groupware can support decision making/problem solving directly or can provide indirect support by improving communication between team members.
- The Internet (Web), intranets, and extranets support decision making through collaboration tools and access to data, information, and knowledge.
- Groupware for direct support such as GDSS typically contains capabilities for electronic brainstorming, electronic conferencing or meeting, group scheduling, calendaring, planning, conflict resolution, model building, videoconferencing, electronic document sharing, stakeholder identification, topic commentator, voting, policy formulation, and enterprise analysis.
- Groupware can support anytime/anyplace groupwork.

- A GSS is any combination of hardware and software that facilitates meetings. Its predecessor, GDSS, provided direct support to decision meetings, usually in a face-to-face setting.
- GDSS attempt to increase process and task gains and reduce process and task losses of groupwork.
- Parallelism and anonymity provide several GDSS gains.
- GDSS may be assessed in terms of the common group activities of information retrieval, information sharing, and information use.
- GDSS can be deployed in an electronic decision room environment, in a multipurpose computer lab, or over the Web.
- Web-based groupware is the norm for anytime/anyplace collaboration.

Key Terms

asynchronous
community of practice
decision room
explicit knowledge
group decision support system (GDSS)
group support system (GSS)
groupthink
groupware
groupwork
idea generation
knowledge
knowledge-based economy
knowledge management (KM)
knowledge management system (KMS)
knowledge repository
leaky knowledge
organizational culture
organizational learning
organizational memory
parallelism
practice approach
process approach
process gain
process loss
synchronous (real-time)
tacit knowledge
virtual meeting
virtual team
wiki

Questions for Discussion

1. Why is the term *knowledge* so difficult to define?
2. Describe and relate the different characteristics of knowledge to one another.
3. Explain why it is important to capture and manage knowledge.
4. Compare and contrast tacit knowledge and explicit knowledge.
5. Explain why organizational culture must sometimes change before knowledge management is introduced.
6. How does knowledge management attain its primary objective?
7. How can employees be motivated to contribute to and use KMS?
8. What is the role of a knowledge repository in knowledge management?
9. Explain the importance of communication and collaboration technologies to the processes of knowledge management.
10. List the three top technologies most frequently used for implementing KMS and explain their importance.
11. Explain why it is useful to describe groupwork in terms of the time/place framework.
12. Describe the kinds of support that groupware can provide to decision makers.
13. Explain why most groupware is deployed today over the Web.
14. Explain why meetings can be so inefficient. Given this, explain how effective meetings can be run.
15. Explain how GDSS can increase some of the benefits of collaboration and decision making in groups and eliminate or reduce some of the losses.
16. The original term for group support system (GSS) was group decision support system (GDSS). Why was the word *decision* dropped? Does this make sense? Why or why not?

Exercises

Teradata UNIVERSITY NETWORK (TUN) and Other Hands-on Exercises

1. Make a list of all the knowledge management methods you use during your day (work and personal). Which are the most effective? Which are the least effective? What kinds of work or activities does each knowledge management method enable?
2. Describe how to ride a bicycle, drive a car, or make a peanut butter and jelly sandwich. Now have someone else try to do it based solely on your explanation. How can you best convert this knowledge from tacit to explicit (or can't you)?
3. Examine the top five reasons that firms initiate KMS and investigate why they are important in a modern enterprise.

4. Read *How the Irish Saved Civilization* by Thomas Cahill (New York: Anchor, 1996) and describe how Ireland became a knowledge repository for Western Europe just before the fall of the Roman Empire. Explain in detail why this was important for Western civilization and history.
5. Examine your university, college, or company and describe the roles that the faculty, administration, support staff, and students have in the creation, storage, and dissemination of knowledge. Explain how the process works. Explain how technology is currently used and how it could potentially be used.
6. Search the Internet for knowledge management products and systems and create categories for them. Assign one vendor to each team. Describe the categories you created and justify them.
7. Consider a decision-making project in industry for this course or from another class or from work. Examine some typical decisions in the project. How would you extract the knowledge you need? Can you use that knowledge in practice? Why or why not?
8. How does knowledge management support decision making? Identify products or systems on the Web that help organizations accomplish knowledge management. Start with **brint.com** and **knowledgemanagement.com**. Try one out and report your findings to the class.
9. Search the Internet to identify sites that deal with knowledge management. Start with **google.com, kmworld.com, kmmag.com**, and **km-forum.org**. How many did you find? Categorize the sites based on whether they are academic, consulting firms, vendors, and so on. Sample one of each and describe the main focus of the site.
10. Make a list of all the communications methods (both work and personal) you use during your day. Which are the most effective? Which are the least effective? What kind of work or activity does each communications method enable?
11. Investigate the impact of turning off every communication system in a firm (i.e., telephone, fax, television, radio, all computer systems). How effective and efficient would the following types of firms be: airline, bank, insurance company, travel agency, department store, grocery store? What would happen? Do customers expect 100 percent uptime? (When was the last time a major airline's reservation system was down?) How long would it be before each type of firm would not be functioning at all? Investigate what organizations are doing to prevent this situation from occurring.
12. Investigate how researchers are trying to develop collaborative computer systems that portray or display nonverbal communication factors.
13. For each of the following software packages, check the trade literature and the Web for details and explain how computerized collaborative support system capabilities are included: Lync, GroupSystems, and WebEx.
14. Compare Simon's four-phase decision-making model to the steps in using GDSS.
15. A major claim in favor of wikis is that they can replace e-mail, eliminating its disadvantages (e.g., spam). Go to **socialtext.com** and review such claims. Find other supporters of switching to wikis. Then find counterarguments and conduct a debate on the topic.
16. Search the Internet to identify sites that describe methods for improving meetings. Investigate ways that meetings can be made more effective and efficient.
17. Go to **groupsystems.com** and identify its current GSS products. List the major capabilities of those products.
18. Go to the Expert Choice Web site (**expertchoice.com**) and find information about the company's group support products and capabilities. Team Expert Choice is related to the concept of the AHP described. Evaluate this product in terms of decision support. Do you think that keypad use provides process gains or process losses? How and why? Also prepare a list of the product analytical capabilities. Examine the free trial. How can it support groupwork?

END-OF-CHAPTER APPLICATION CASE

Solving Crimes by Sharing Digital Forensic Knowledge

Digital forensics has become an indispensable tool for law enforcement. This science is not only applied to cases of crime committed with or against digital assets, but is used in many physical crimes to gather evidence of intent or proof of prior relationships. The volume of digital devices that might be explored by a forensic analysis, however, is staggering, including anything from a home computer to a videogame console, to an engine module from a getaway vehicle. New hardware, software, and applications are being released into public use daily, and analysts must create new methods to deal with each of them.

Many law enforcement agencies have widely varying capabilities to do forensics, sometimes enlisting the aid of other agencies or outside consultants to perform analyses. As new techniques are developed, internally tested, and ultimately scrutinized by the legal system, new forensic hypotheses are born and proven. When the same techniques are applied to other cases, the new proceeding is strengthened by the precedent of a prior case. Acceptance of a methodology in multiple proceedings makes it more acceptable for future cases.

Unfortunately, new forensic discoveries are rarely formally shared—sometimes even among analysts within the same agency. Briefings may be given to other analysts within the same agency, although caseloads often dictate immediately moving on to the next case. Even less is shared between different agencies, or even between different offices of some federal law enforcement communities. The result of this lack of sharing is duplication of significant effort to re-discover the same or similar approaches to prior cases and a failure to take consistent advantage of precedent rulings that may strengthen the admission of a certain process.

The Center for Telecommunications and Network Security (CTANS), a center of excellence that includes faculty from Oklahoma State University's Management Science and Information Systems Department, has developed, hosted, and is continuously evolving Web-based software to support law enforcement digital forensics investigators (LEDFI) via access to forensics resources and communication channels for the past 6 years. The cornerstone of this initiative has been the National Repository of Digital Forensics Information (NRDFI), a collaborative effort with the Defense Cyber Crime Center (DC3), which has evolved into the Digital Forensics Investigator Link (DFILink) over the past 2 years.

Solution

The development of the NRDFI was guided by the theory of the egocentric group and how these groups share knowledge and resources among one another in a community of practice (Jarvenpaa & Majchrzak, 2005). Within an egocentric community of practice, experts are identified through interaction, knowledge remains primarily tacit, and informal communication mechanisms are used to transfer this knowledge from one participant to the other. The informality of knowledge transfer in this context can lead to local pockets of expertise as well as redundancy of effort across the broader community as a whole. For example, a digital forensics (DF) investigator in Washington, DC, may spend 6 hours to develop a process to extract data hidden in slack space in the sectors of a hard drive. The process may be shared among his local colleagues, but other DF professionals in other cities and regions will have to develop the process on their own.

In response to these weaknesses, the NRDFI was developed as a hub for knowledge transfer between local law enforcement communities. The NRDFI site was locked down so that only members of law enforcement were able to access content, and members were provided the ability to upload knowledge documents and tools that may have developed locally within their community, so that the broader law enforcement community of practice could utilize their contributions and reduce redundancy of efforts. The Defense Cyber Crime Center, a co-sponsor of the NRDFI initiative, provided a wealth of knowledge documents and tools in order to seed the system with content (see Figure 12.7).

Results

Response from the LEDFI community was positive, and membership to the NRDFI site quickly jumped to over 1,000 users. However, the usage pattern for these members was almost exclusively unidirectional. LEDFI members would periodically log on, download a batch of tools and knowledge documents, and then not log on again until the knowledge content on the site was extensively refreshed. The mechanisms in place for local LEDFI communities to share their own knowledge and tools sat largely unused. From here, CTANS began to explore the literature with regard to motivating knowledge sharing, and began a redesign of NRDFI

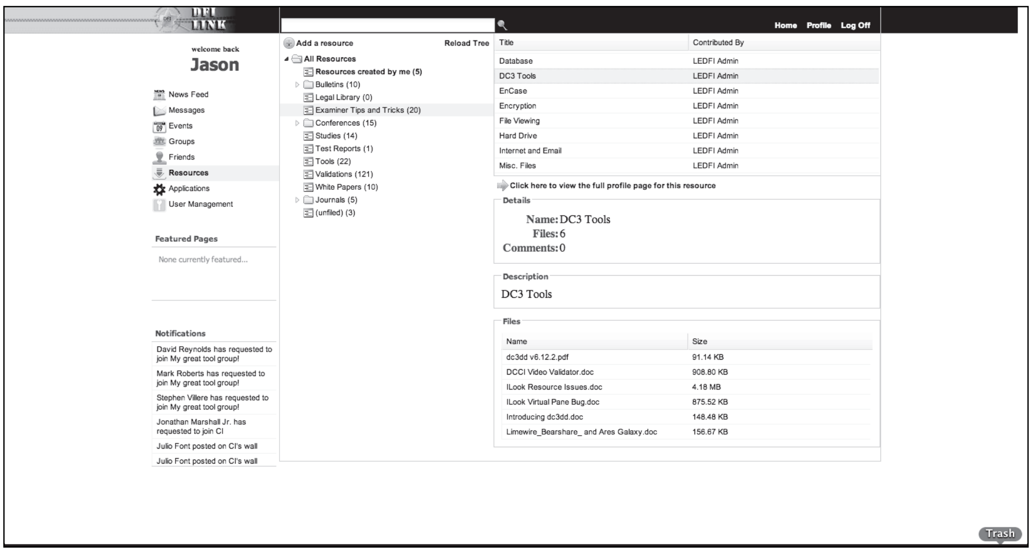

FIGURE 12.7 **DFI-Link Resources.**

driven by the extant literature; they focused on promoting sharing within the LEDFI community through the NRDFI.

Some additional capabilities include new applications such as a "Hash Link," which can provide DFI Link members with a repository of hash values that they would otherwise need to develop on their own and a directory to make it easier to contact colleagues in other departments and jurisdictions. A calendar of events and a newsfeed page were integrated into the DFI Link in response to requests from the users. Increasingly, commercial software is also being hosted. Some were licensed through grants and others were provided by vendors, but all are free to vetted users of the law enforcement community.

The DFI Link has been a positive first step toward getting LEDFI to better communicate and share knowledge with colleagues in other departments. Ongoing research is helping to shape the DFI Link to better meet the needs of its customers and promote even greater knowledge, sharing. Many LEDFI are inhibited from sharing such knowledge, as policies and culture in the law enforcement domain often promote the protection of information at the cost of knowledge sharing. However, by working with DC3 and the law enforcement community, researchers are beginning to knock down these barriers and create a more productive knowledge sharing environment.

Questions for the End-of-Chapter Application Case

1. Why should digital forensics information be shared among law enforcement communities?
2. What does egocentric theory suggest about knowledge sharing?
3. What behavior did the developers of NRDFI observe in terms of use of the system?
4. What additional features might enhance the use and value of such a KMS?

Sources: Harrison et al., "A Lessons Learned Repository for Computer Forensics," *International Journal of Digital Evidence*, Vol. 1, No. 3, 2002; S. Jarvenpaa and A. Majchrzak, *Developing Individuals' Transactive Memories of their Ego-Centric Networks to Mitigate Risks of Knowledge Sharing: The Case of Professionals Protecting CyberSecurity*. Paper presented at the Proceedings of the Twenty-Sixth International Conference on Information Systems, 2005; J. Nichols, D. P. Biros, and M. Weiser, "Toward Alignment Between Communities of Practice and Knowledge-Based Decision Support," *Journal of Digital Forensics, Security, and Law,* Vol. 7, No. 2, 2012; M. Weiser, D. P. Biros, and G. Mosier, "Building a National Forensics Case Repository for Forensic Intelligence," *Journal of Digital Forensics, Security, and Law,* Vol. 1, No. 2, May 2006 (This case was contributed by David Biros, Jason Nichols, and Mark Weiser).

References

Alavi, M. (2000). "Managing Organizational Knowledge." Chapter 2 in W. R. Zmud (ed.). *Framing the Domains of IT Management: Projecting the Future*. Cincinnati, OH: Pinnaflex Educational Resources.

Alavi, M., T. Kayworth, and D. Leidner. (2005/2006). "An Empirical Examination of the Influence of Organizational Culture on Knowledge Management Practice." *Journal of Management Information Systems,* Vol. 22, No. 3.

Alavi, M., and D. Leidner. (2001). "Knowledge Management and Knowledge Management Systems: Conceptual Foundations and Research Issues." *MIS Quarterly,* Vol. 25, No. 1, pp. 107–136.

Bock, G.-W., R. Zmud, Y. Kim, and J. Lee. (2005). "Behavioural Intention Formation in Knowledge Sharing: Examining the Roles of Extrinsic Motivators, Social Psychological Forces and Organizational Climate." *MIS Quarterly Journal,* Vol. 29, No. 1.

Carlucci, D., and G. Schiuma. (2006). "Knowledge Asset Value Spiral: Linking Knowledge Assets to Company's Performance." *Knowledge and Process Management,* Vol. 13, No. 1.

Chua, A., and W. Lam. (2005). "Why KM Projects Fail: A Multi-Case Analysis." *Journal of Knowledge Management,* Vol. 9, No. 3, pp. 6–17.

Davenport, D. (2008). "Enterprise 2.0: The New, New Knowledge Management?" **http://blogs.hbr.org/davenport/2008/02/enterprise_20_the_new_new_know.html** (accessed Sept. 2013).

Davenport, T., D. W. DeLong, and M. C. Beers. (1998, Winter). "Successful Knowledge Management Projects." *Sloan Management Review,* Vol. 39, No. 2.

Davenport, T., and L. Prusak. (1998). "How Organizations Manage What They Know." Boston: Harvard Business School Press.

Delen, D., and S. S. Hawamdeh. (2009). "A Holistic Framework for Knowledge Discovery and Management." *Communications of the ACM,* Vol. 52, No. 6, pp. 141–145.

DeSanctis, G., and R. B. Gallupe. (1987). "A Foundation for the Study of Group Decision Support Systems." *Management Science,* Vol. 33, No. 5.

Godin, B. (2006). "Knowledge-Based Economy: Conceptual Framework or Buzzword." *The Journal of Technology Transfer,* Vol. 31, No. 1.

Gray, P. (1999). "Tutorial on Knowledge Management." *Proceedings of the Americas Conference of the Association for Information Systems*, Milwaukee.

Hall, M. (2002, July 1). "Decision Support Systems." *Computerworld,* Vol. 36, No. 27.