

DECISION SUPPORT SYSTEMS: AN OVERVIEW

LEARNING OBJECTIVES

- Understand possible DSS configurations
- Describe DSS characteristics and capabilities
- Understand DSS components and how they integrate
- Describe the components and structure of each DSS component: the data-management subsystem, the model-management subsystem, the user-interface (dialog) subsystem, the knowledge-based management subsystem, and the user
- Explain how the World Wide Web has affected DSS, and vice versa
- Explain the unique role of the user in DSS versus MIS
- Describe the DSS hardware platforms
- Understand the important DSS classifications

In Chapter 1 we introduced DSS and stressed its support in the solution of complex managerial problems. The methodology of decision-making was presented in Chapter 2. In this chapter we show how DSS superiority is achieved by examining its capabilities, structure, and classifications in the following sections:

- 3.1 Opening Vignette: Southwest Airlines Flies in the Face of Competition through DSS
- 3.2 DSS Configurations
- 3.3 What Is a DSS?
- 3.4 Characteristics and Capabilities of DSS
- 3.5 Components of DSS
- 3.6 The Data Management Subsystem
- 3.7 The Model Management Subsystem
- 3.8 The User Interface (Dialog) Subsystem
- 3.9 The Knowledge-Based Management Subsystem
- 3.10 The User
- 3.11 DSS Hardware
- 3.12 DSS Classifications
- 3.13 Summary

3.1 OPENING VIGNETTE: SOUTHWEST AIRLINES FLIES IN THE FACE OF COMPETITION THROUGH OSS¹

INTRODUCTION

About a year after the September 11, 2001 disaster and the resulting plunge in airline revenues, Southwest Airlines was so pleased with the performance of its *business Intelligence/DSS applications* for financial management that it expanded deployment to include flight operations and maintenance. In the middle of a crisis, Southwest Airlines successfully deployed its Hyperion Solutions Corp. Essbase online analytical processing (OLAP) application and Pillar budgeting software. Southwest can accurately make financial forecasts in facing the severe market downturn. Southwest has exploited its business intelligence applications successfully.

PROBLEMS MANY COMPANIES FACE

Most companies do not adequately tie their financial applications into an OLAP system, analyze their data, and then meaningfully present it to business personnel. Southwest's success resulted from its ability to tie its enterprise resource planning applications to its OLAP software and then present relevant financial data and scenarios to its decision-makers.

THE SITUATION

Right after the terrorist attacks, the airline was operating "in a world of complete uncertainty," said Mike Van de Ven, vice president of financial planning and analysis at Southwest. "We were asked to give some sort of financial insight for a variety of decisions the company might make."

Prior to the roughly \$1 million installation of Essbase from Hyperion (Sunnyvale, California) in 1999, Southwest analysis personnel wrote queries by hand, spending about a half hour running them, and then put the figures into spreadsheets for additional analysis. The total time could take up to four hours.

RESULTS

Essbase has cut the analysis time to as little as two minutes, leading to massive savings. After running worst- and best-case scenarios and creating forecasts, Southwest developed an action plan to stabilize its finances. It helped answer questions like, "How fast would we burn through our cash?" As of July 2002, the forecasts have been within 2 percent of actual values.

¹Sources: Based on Marc L. Songini, "Southwest Expands Business Tools' Role," *ComputerWorld*, V61. 36, No. 29, July 15, 2002, p. 6; Trebor Banstetter, "Southwest Airlines Posts \$75 Million Fourth-Quarter Profit," *Knight Ridder Tribune Business News*, October 18, 2002; Trebor Banstetter, "Southwest Airlines Sees Passenger Traffic Rise," *Knight Ridder Tribune Business News*, October 3, 2002; and Bill Hensel, Jr., "Dallas-Based Southwest Airlines Adds Flights, Drops Fares," *Knight Ridder Tribune Business News*, August 28, 2002.

Analysts can access both operational and financial data to analyze and identify the impact of one set on the other. Relationships can be found to improve forecasting. Overall, the application has paid for itself just through the savings from automating the data-collection processes.

Southwest has better control over its cost structure than the network carriers do. It is the largest airline that has remained profitable since the travel industry began to slump in 2001. The airlines overall lost \$7 billion in 2001 and was expected to lose at least as much in 2002. Southwest Airlines may have been one of the only carriers to make a profit in 2002. Southwest Airlines is still growing (though cautiously), despite the massive market downturn and showed a \$75 million profit in the fourth quarter of 2002. Southwest's new business intelligence tools help decision-makers accurately predict their markets and decide which ones to expand into.

• **QUESTIONS FOR THE OPENING VIGNETTE**

1. What kinds of models do you suppose Southwest Airlines used in its OLAP?
2. How can business intelligence like that utilized by Southwest Airlines lead to higher profits and a more competitive position in the marketplace?
3. Explain how the benefits were obtained.
4. Why don't most companies do what Southwest Airlines did?
5. Explain how these ideas could be used in other industry segments (e.g., retail, insurance, oil and gas, universities).

3.2 DSS CONFIGURATIONS

The opening vignette illustrates the versatility of a **DSS/business intelligence system**. Specifically, it shows a support system with the following characteristics:

- It supports individual members and an entire team.
- It is used repeatedly and constantly.
- It has three major components: data, models, and a user interface.
- It uses subjective, personal, and objective data.
- It is used in the private sector.
- It helps the user to make faster, smarter, better decisions.

Though not mentioned, the user interface and database access were, no doubt, implemented with Web/Internet technologies.

This vignette demonstrates some of the potential diversification of DSS. Decision support can be provided in many different configurations. These configurations depend on the nature of the management-decision situation and the specific technologies used for support. These technologies are assembled from four basic components (each with several variations): *data*, *models*, *user interface*, and (optionally) *knowledge*, often deployed over the Web. Each of these components is managed by software that either is commercially available or must be programmed for the specific task. The manner in

which these components are assembled defines their major capabilities and the nature of the support provided. For example, models are emphasized in a model-oriented DSS. Such models can be customized with a spreadsheet or a programming language or can be provided by standard algorithm-based tools that include linear programming. Similarly, in a data-oriented DSS, a database and its management play the major roles. In the situation in the opening vignette, both approaches were used. In this chapter we will explore all of these and related topics, but first we revisit the definitions of a DSS.

3.3 WHAT IS A DSS?

The early definitions of a DSS identified it as a system intended to support managerial decision-makers in semistructured decision situations. DSS were meant to be an adjunct to decision-makers to extend their capabilities but not to replace their judgment. They were aimed at decisions where judgment was required or at decisions that could not be completely supported by algorithms. Not specifically stated, but implied in the early definitions, was the notion that the system would be computer-based, would operate interactively online, and preferably would have graphical output capabilities. The early definitions were open to several interpretations. Soon several other definitions appeared that caused considerable disagreement as to what a DSS really is. We discuss these definitions next.

DSS DEFINITIONS

Little (1970) defines DSS as a "model-based set of procedures for processing data and judgments to assist a manager in his decision-making." He argues that to be successful, such a system must be simple, robust, easy to control, adaptive, complete on important issues, and easy to communicate with. Alter (1980) defines DSS by contrasting them with traditional electronic data processing (EDP) systems on five dimensions, as shown in Table 3.1.

Moore and Chang (1980) argue that the structuredness concept, so much a part of early DSS definitions (i.e., that DSS can handle semistructured and unstructured situations), is not meaningful in general; a problem can be described as structured or unstructured only with respect to a particular decision-maker or a specific situation (i.e., structured decisions are structured because we choose to treat them that way). Thus, they define DSS as extendible systems capable of supporting ad hoc data analysis and decision modeling, oriented toward future planning, and used at irregular, unplanned intervals.

TABLE 3.1 DSS Versus HDP

<i>Dimension</i>	<i>DSS</i>	<i>EDP</i>
Use	Active	Passive
User	Line and staff management	Clerical
Goal	Effectiveness	Mechanical efficiency
Time horizon	Present and future	Past
Objective	Flexibility	Consistency

Source: Based on Alter (1980).

Bonczek et al. (1980) define a DSS as a computer-based system consisting of three interacting components: a language system (a mechanism to provide communication between the user and other components of the DSS), a knowledge system (a repository of problem domain knowledge embodied in DSS as either data or procedures), and a problem-processing system (a link between the other two components, containing one or more of the general problem-manipulation capabilities required for decision-making). The concepts provided by this definition are important for understanding the relationship between DSS and knowledge.

Finally, Keen (1980) applies the term DSS "to situations where a 'final' system can be developed only through an adaptive process of learning and evolution." Thus, he defines a DSS as the product of a developmental process in which the DSS user, the DSS builder, and the DSS itself are all capable of influencing one another, resulting in system evolution and patterns of use.

These definitions are compared and contrasted by examining the various concepts used to define DSS (Table 3.2). It seems that the basis for defining DSS has been developed from the perceptions of what a DSS does (e.g., support decision-making in unstructured problems) and from ideas about how the DSS's objective can be accomplished (e.g., components required, appropriate usage pattern, necessary development processes).

Unfortunately, the formal definitions of DSS do not provide a consistent focus because each tries to narrow the population differently. Furthermore, they collectively ignore the central purpose of DSS, that is, *to support and improve decision-making*. In later DSS definitions, the focus seems to be on inputs rather than outputs. A very likely reason for this change in emphasis is the difficulty of measuring the outputs of a DSS (e.g., decision quality or confidence in the decision made).

A DSS APPLICATION

A DSS is usually built to support the solution of a certain problem or to evaluate an opportunity. As such it is called a DSS application. In DSS in Focus 3.1 we provide a working definition that includes a range from a basic to an ideal DSS application. Later in this chapter the various configurations of DSS are explored. However, it is beneficial first to deal with the characteristics and capabilities of DSS, which we present next.

We show a typical Web-based DSS architecture in Figure 3.1. This DSS structure utilizes models in business intelligence work. Processing is distributed across several servers in solving large analytical problems. This multitiered architecture uses a Web browser to run programs on an application server. The server accesses data to construct one or more models. Data may also be provided by a data server that optionally extracts data from a data warehouse or a legacy mainframe system. When the user requires that the model be optimized, the model, populated with the data, is transferred to an optimization server. The optimization server may access additional data

TABLE 3.2 Concepts Underlying DSS Definitions

<i>Source</i>	<i>DSS Defined in Terms of</i>
Gorry and Scott-Morton (1971)	Problem type, system function (support)
Little (1970)	System function, interface characteristics
Alter (1980)	Usage pattern, system objectives
Moore and Chang (1980)	Usage pattern, system capabilities
Bonczek et al. (1989)	System components
<u>Keen (1980)</u>	Development process

DSS IN FOCUS 3.1

WHAT IS A DSS APPLICATION?

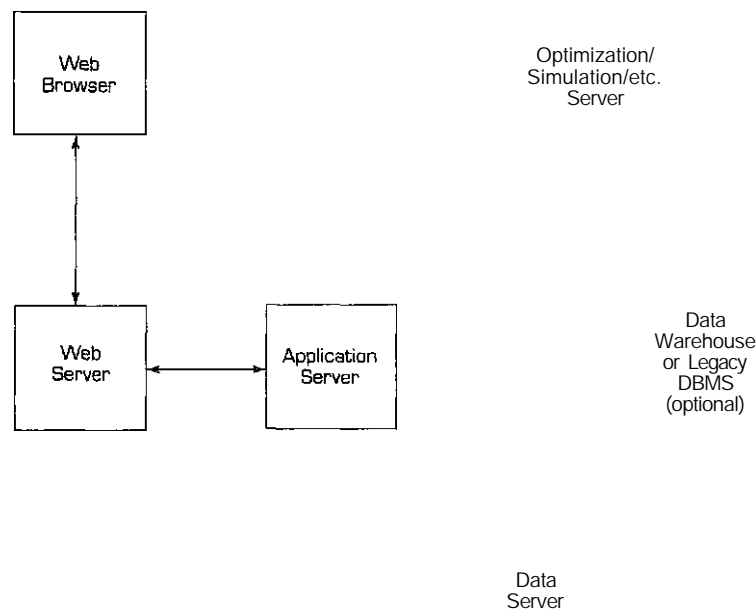
A DSS is an approach (or methodology) for supporting decision-making. It uses an interactive, flexible, adaptable CBIS especially developed for supporting the solution to a specific nonstructured management problem. It uses data, provides an easy user interface, and can incorporate the decision-maker's own insights.

In addition, a DSS usually uses models and is built (often by end-users) by an interactive and iterative process. It supports all phases of decision-making and may include a knowledge component.

Finally, a DSS can be used by a single user on a PC or can be Web-based for use by many people at several locations.

from the data server, solves the problem, and provides the solution directly to the user's Web browser. Generated solution reports, which the application server may massage to make them readable by managers, may be sent directly to appropriate parties via e-mail or may be made available through another Web portal as part of this enterprise information system. The Web-based DSS described in Case Application 3.2 is structured along these lines, as is the application described in DSS in Action 3.2. See Cohen, Kelly, and Medaglia (2001) for further examples of several Web-based applications that utilize this type of architecture. Similar architectures are described by Dong, Sundaram, and Srinivasan (2002), Gachet (2002), and Forgie et al. (2002).

FIGURE 3.1 MULTITIERED ARCHITECTURE FOR INCORPORATING OPTIMIZATION, SIMULATION, AND OTHER MODELS INTO WEB-BASED DSS



(Source: Adapted from Cohen, Kelly, and Medaglia, 2001.)

OSS IN ACTION 3.2

CAMERON AND BARKLEY COMPANY'S
WEB-BASED DSS REDUCES INVENTORIES
AND IMPROVES PERFORMANCE

Cameron and Barkley (Cambar) Company (Charleston, South Carolina) distributes industrial, electrical, and electronic supplies throughout the United States. Nearly one-half million products comprise Cambar's inventory. Cambar needed to reduce its inventory without sacrificing its level of customer service. These two goals are contradictory, yet occur often in practice. The company needed to manage and improve its product inventory and improve the accuracy of demand forecasts—the key to inventory reduction. By analyzing demand data, several good ordering rules were identified. Next a prototype inventory-planning and management system was developed, tested, and deployed. The buyers, who managed the inventory, had the goal of maintaining high enough inventory levels to meet strict levels of customer service. But too high invokes inventory carrying costs; capital is tied up in inventory, and there are costs of maintaining it. Buyers used judgmental and simple demand forecasts to determine these lev-

els. They had a tendency to overstock and far missed the goal of a minimum of four inventory turns per year.

Then Cambar developed the Inventory-Replenishment Planner (IRP). This model utilizes the architecture shown in Figure 3.1. A Web interface captures user interactions, saves the business information, and builds a model utilizing data from a data server on the application server. The model approximates lead-time demand and minimizes ordering and fixed costs subject to required service levels. The model is solved on the optimization/simulation server with two heuristics. The effects are evaluated on the optimization/simulation server by simulating the effect of policies to evaluate their effectiveness. Results are captured by the application server and handed off to the Web server, which generates meaningful reports to determine what and when to order.

Source: Adapted from Cohen, Kelly, and Medagli (2001).

3.4 CHARACTERISTICS AND CAPABILITIES OF DSS

Because there is no consensus on exactly what a DSS is, there is obviously no agreement on the standard characteristics and capabilities of DSS. The capabilities in Figure 3.2 constitute an ideal set, some members of which were described in the definitions as well as in the opening vignette. The term *business intelligence* is synonymous with DSS but has become tightly aligned with Web implementations (see DSS in Focus 3.3; also see Callaghan, 2002; Hall, 2002a,b; Harreld, 2002; d Ocken, 2002). **Business analytics** is another term that implies the use of models and data to improve an organization's performance or competitive posture. In business analytics, the focus is on the use of models, even if they are deeply buried inside the system. In fact, PricewaterhouseCoopers (PwC) estimates that only 10 to 20 percent of users access DSS tools. To reach the rest, business analytics must be embedded in core IT solutions (see Hall, 2000b). Hall (2002a) describes *Web analytics*, an approach to using business analytics tools on real-time Web information to assist in decision-making. Most of these applications are related to e-commerce, while others are being initiated in product development and supply chain management.

The key DSS characteristics and capabilities (Figure 3.2) are:

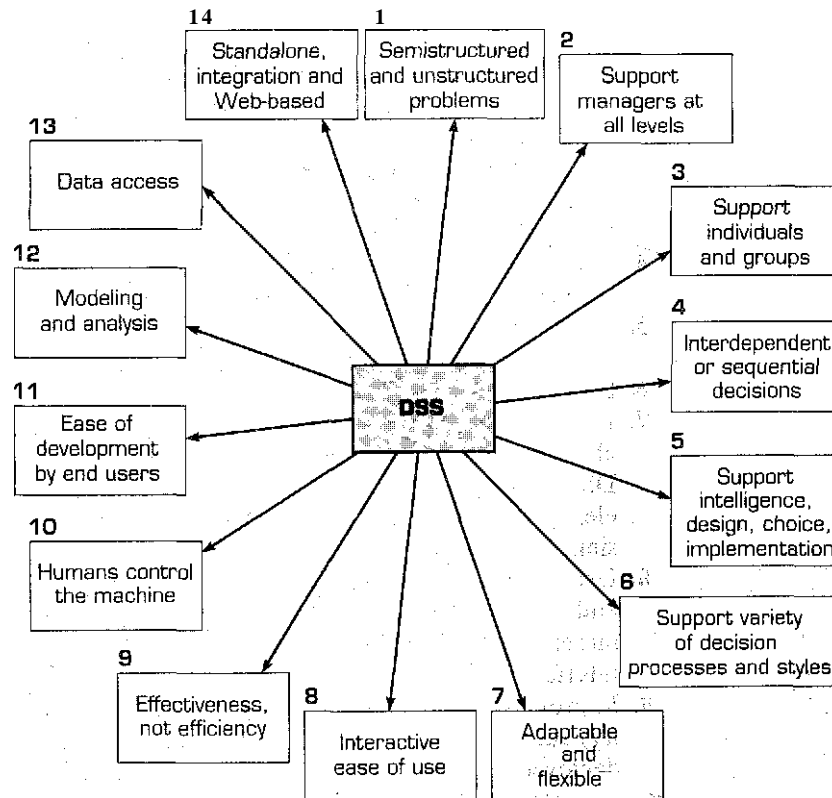


FIGURE 3.2 KEY CHARACTERISTICS AND CAPABILITIES OF DSS

DSS IN FOCUS 3.3

WHAT IS BUSINESS INTELLIGENCE?

Business intelligence (BI) is a collection of technical and process innovations across the data warehousing and business intelligence space. Proactive BI focuses on decision-making acceleration by leveraging existing BI infrastructure to identify, calculate, and distribute up-to-the-moment, mission-critical information. Through the application of these techniques and technologies, the reach and value of data warehouse and BI systems can be increased by one or more orders of magnitude. Business success today requires intelligent data use.

Proactive BI has five components: real-time warehousing, automated anomaly and exception detection,

proactive alerting with automatic recipient determination, seamless follow-through workflow, and automatic learning and refinement. Wireless technologies have a key role to play in increasing the value and efficiency of several of these Components.

Business analytics implies the use of models in business intelligence. These models may be manual, as in OLAP, or automatic, as in data mining.

Sources: Some material adapted from Langseth and Vivatrat, 2002;;also see Ocken, 2002; Rataock (2002).

1. Support for decision-makers, mainly in semistructured and unstructured situations, by bringing together human judgment and computerized systems. Problems that cannot be solved (or cannot be solved conveniently) by other computerized systems or by standard quantitative methods or tools.
2. Support for all managerial levels, ranging from top executives to line managers.
3. Support for individuals as well as to groups. Less-structured problems often require the involvement of individuals from different departments and organizational levels or even from different organizations. DSS support virtual teams through collaborative Web tools.
4. Support for interdependent and/or sequential decisions. The decisions may be made once, several times, or repeatedly.
5. Support in all phases of the decision-making process: intelligence* design, choice, and implementation.
6. Support in a variety of decision-making processes and styles.
7. Adaptivity over time. The decision-maker should be reactive, able to confront changing conditions quickly, and able to adapt the DSS to meet these changes. DSS are flexible, and so users can add, delete, combine, change, or rearrange basic elements. They are also flexible in that they can be readily modified to solve other, similar problems.
8. User feeling of at-homeness. User-friendliness, strong graphical capabilities, and a natural language interactive human-machine interface can greatly increase the effectiveness of DSS. Most new DSS applications use Web-based interfaces.
9. Improvement of the effectiveness of decision-making (accuracy, timeliness, quality) rather than its efficiency (the cost of making decisions). When DSS are deployed, decision-making often takes longer, but the decisions are better.
10. Complete control by the decision-maker over all steps of the decision-making process in solving a problem. A DSS specifically aims to support and not to replace the decision-maker.
11. End-users are able to develop and modify simple systems by themselves. Larger systems can be built with assistance from information system (IS) specialists. OLAP (online analytical processing) software in conjunction with data warehouses allows users to build fairly large, complex DSS.
12. Models are generally utilized to analyze decision-making situations. The modeling capability enables experimenting with different strategies under different configurations. In fact, the *models make a DSS different from most MIS*.
13. Access is provided to a variety of data sources, formats, and types, ranging from geographic information systems (GIS) to object-oriented ones.
14. Can be employed as a standalone tool used by an individual decision-maker in one location or distributed throughout an organization and in several organizations along the supply chain. It can be integrated with other DSS and/or applications, and can be distributed internally and externally, using networking and Web technologies.

These key DSS characteristics and capabilities allow decision-makers to make better, more consistent decisions in a timely manner, and they are provided by the major DSS components, which we describe next.

3.5 COMPONENTS OF DSS

A DSS application can be composed of the subsystems shown in Figure 3.3.

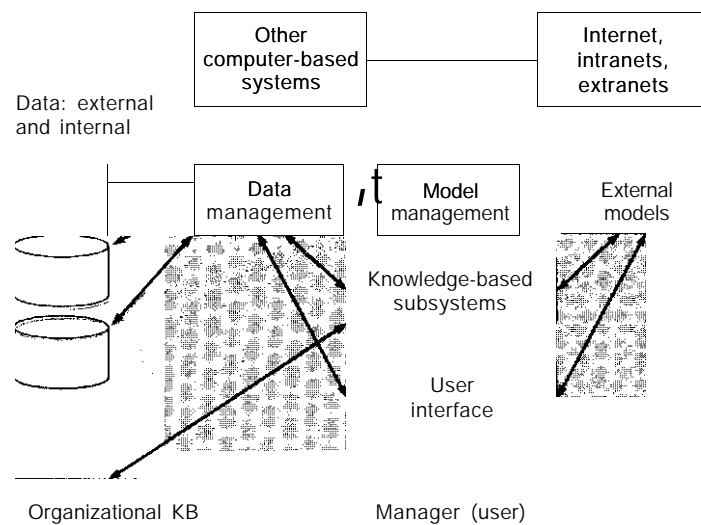
Data-management subsystem. The data management subsystem includes a database that obtains relevant data for the situation and is managed by software called the **database management system (DBMS)**.² The data management subsystem can be interconnected with the corporate **data warehouse**, a repository for corporate relevant decision-making data. Usually the data are stored or accessed via a database Web server.

Model management subsystem. This is a software package that includes financial, statistical, management science, or other quantitative models that provide the system's analytical capabilities and appropriate software management. Modeling languages for building custom models are also included. This software is often called a **model base management system (MBMS)**. This component can be connected to corporate or external storage of models. Model solution methods and management systems are implemented in Web development systems (like Java) to run on application servers.

User interface subsystem. The user communicates with and commands the DSS through this subsystem. The user is considered part of the system. Researchers assert that some of the unique contributions of DSS are derived from the intensive interaction between the computer and the decision-maker. The Web browser provides a familiar, consistent graphical user interface structure for most DSS.

Knowledge-based management subsystem. This subsystem can support any of the other subsystems or act as an independent component. It provides intelligence to

FIGURE 3.3 A SCHEMATIC VIEW OF DSS



²DBMS is both singular and plural (system and systems), as are many acronyms in this text.

augment the decision-maker's own. It can be interconnected with the organization's knowledge repository (part of a knowledge management system), which is sometimes called the **organizational knowledge base**. Knowledge may be provided via Web servers. Many artificial intelligence methods have been implemented in Web development systems like Java, and are easy to integrate into the other DSS components.

By definition, a DSS must include the three major components of a DBMS, MBMS, and user interface. The knowledge-based management subsystem is optional, but can provide many benefits by providing intelligence in and to the three major components. As in any management information system, the user may be considered a component of DSS.

These components form the DSS application system, which can be connected to the corporate intranet, to an extranet, or to the Internet. Typically the components communicate via Internet technology. Web browsers typically provide the user interface. The schematic view of a DSS and the above components shown in Figure 3.2 provides a basic understanding of the general structure of a DSS. In Table 3.3, we provide a sampling of the impacts of the Web on DSS components, and vice versa. These impacts have been substantial, because improvements in what began as the Internet have had a major effect on how we access, use, and think of DSS. Next, we present a more detailed look at each component; we provide details in Chapters 4-9.

3.6 THE DATA MANAGEMENT SUBSYSTEM

The data-management subsystem is composed of the following elements:

- DSS database
- Database management system
- Data directory
- Query facility.

These elements are shown schematically in Figure 3.4 (in the shaded area). The figure also shows the interaction of the data management subsystem with the other parts of the DSS, as well as its interaction with several data sources. A brief discussion of these elements and their function follows; further discussion is provided in Chapter 5. In DSS in Action 3.4, the primary focus of the DSS is on the database.

THE DATABASE

A **database** is a collection of interrelated data organized to meet the needs and structure of an organization and can be used by more than one person for more than one application. There are several possible configurations for a database. In many DSS instances, data are ported from the data warehouse or a legacy mainframe database system through a database Web server (see DSS in Action 3.2 and 3.4). For other DSS applications, a special database is constructed as needed. Several databases can be used in one DSS application, depending on the data sources. Generally users expect to utilize a Web browser for access, and database Web servers deliver the data regardless of the source. For examples, see DSS in Action 3.2 and 3.4.

The data in the DSS database, as shown in Figure 3.4, are extracted from internal and external data sources, as well as from personal data belonging to one or more users. The extraction results go to the specific application's database or to the corpo-

TA 3 LE 3.3 DSS Components and Web Impacts

<i>Phase</i>	<i>Web Impacts</i>	<i>Impacts on the Web</i>
Database Management System (DBMS)	<p>Consistent, friendly, graphical user interface</p> <p>Provides for a direct mechanism to query databases</p> <p>Provides a consistent communication channel for data, information, and knowledge</p> <p>Data access through m-commerce devices</p> <p>Intranets and extranets</p> <p>Web-based development tools</p> <p>New programming languages and systems</p> <p>Proliferation of database use throughout organizations-made enterprise-wide systems feasible</p> <p>Access to information about databases</p>	<p>A means to conduct e-commerce (transactions must be stored and acted upon)</p> <p>Database Web servers</p> <p>Stores data about the Web for analysis using models to determine effectiveness and efficiency</p>
Model Base Management System (MBMS)	<p>Access to models and solution methods implemented as Java applets and other Web development systems</p> <p>Use of models by untrained managers because they are so easy to use</p> <p>Access to Web-based AI tools to suggest models and solution methods in DSS</p> <p>Access to information about models</p>	<p>Improved infrastructure design and updates</p> <p>Models and solutions of Web infrastructure issues</p> <p>Models of Web message routing improves performance</p> <p>Forecasting models predict viability of hardware and software choices</p>
User Interface Dialog (UI) System	<p>Web browsers provide a flexible, Consistent, and familiar DSS graphical user interface</p> <p>Access to information about user interfaces</p> <p>Experimental user interfaces are tested and distributed via the Web</p>	<p>Initial graphical user interfaces and the computer mouse helped define how a Web browser should work</p> <p>Speech recognition and generation are deployed over the Web</p>
Knowledge-base Management System (KBMS)	<p>Access to AI methods</p> <p>Access to information about AI methods</p> <p>Access to knowledge</p> <p>Web-based AI tools are deployed as Java applets or as other Web development system tools</p>	<p>AI methods readily handle network design issues and message routing</p> <p>Expert systems diagnose problems and workarounds for failures in the Internet</p> <p>Expert systems diagnose hardware problems and recommend specific repairs</p> <p>Intelligent search engines learn user patterns</p>

rate data warehouse (Chapter 5), if it exists. In the latter case, it can be used for other applications.

Internal data come mainly from the organization's transaction processing system. A typical example of such data is the monthly payroll. Depending on the needs of the DSS, operational data from functional areas, such as marketing (e.g., Web transactions from e-commerce), might be included. Other examples of internal data are machine maintenance scheduling and budget allocations, forecasts of future sales, costs of out-of-stock items, and future hiring plans. Internal data can be made available through Web browsers over an **intranet**, an internal Web-based system.

External data include industry data, marketing research data, census data, regional employment data, government regulations, tax rate schedules, and national economic data. These data can come from government agencies, trade associations, market research firms, econometric forecasting firms, and the organization's own efforts to col-

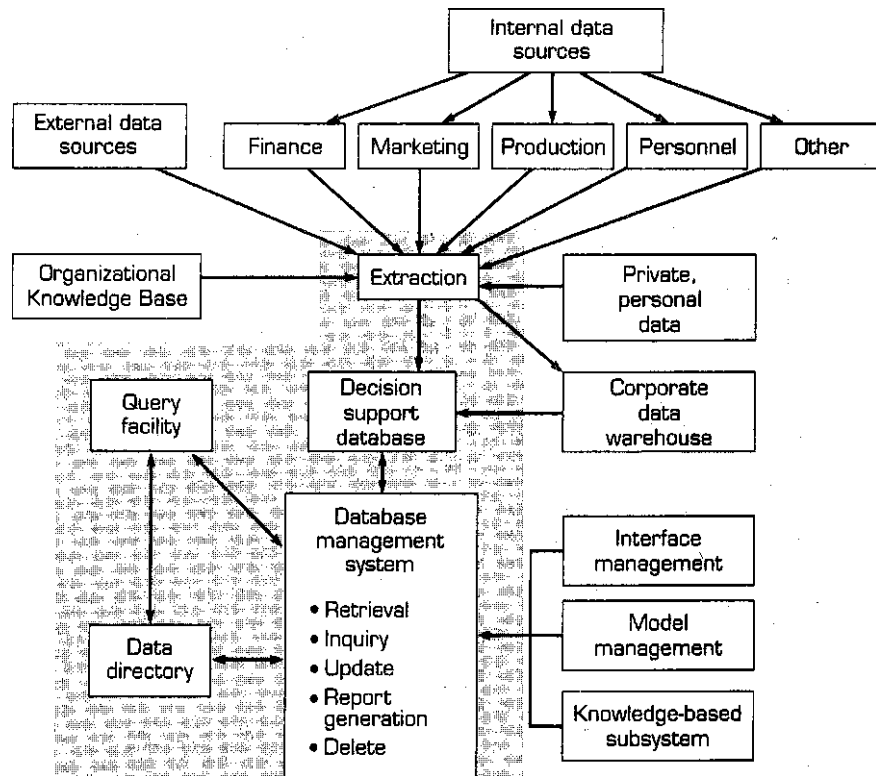


FIGURE 3.4 THE STRUCTURE OF THE DATA MANAGEMENT SUBSYSTEM

DSS IN ACTION 3.4

ROADWAY DRIVES LEGACY
APPLICATIONS ONTO THE WEB

It was time for Roadway Express Inc. (Akron, Ohio) to move from a mainframe *green screen* to the more popular Web browser interface. The existing system could handle data requests and updates, but it looked old and did not present customers with a good impression of the company. So Roadway, rather than reinvent the wheel, polished its surface instead. Roadway's Web design group developed a Web server front-end for access to mainframe scheduling and tracking data. The Janus Web Server (Sirius Software Inc., Cambridge, Massachusetts) front-ends the mainframe, allowing Roadway to

reuse its existing transportation-management and administrative systems. The link between the two systems is so seamless that users don't realize they are using 14-year-old technology. Customers can access the system and generate reports on their own shipments. Roadway's Web site is one of the most sophisticated and capable on the market.

Source: Adapted from Linda Rosencrance, "Roadway Drives Legacy Apps onto the Web," *ComputerWorld*, April 9, 2002, p. 48.

lect external data. Like internal data, external data can be maintained in the DSS database or accessed directly when the DSS is used. External data are provided, in many cases, over the Internet (e.g., from computerized online services or as picked up by search engines). As we mentioned in Chapter 2, Hansen (2002) provides a list of many Web sites with global macroeconomic and business data.

Private data can include guidelines used by specific decision-makers and assessments of specific data and/or situations.

DATA ORGANIZATION

Should a DSS have a standalone database? It depends. In small, ad hoc DSS, data can be entered directly into models, sometimes extracted directly from larger databases. In large organizations that use extensive amounts of data, such as Wal-Mart, AT&T, and American Air Lines, data are organized in a data warehouse and used when needed (Agosta, 2002; Inmon, 2002; Inmon et al. 2000,2001,2002; Marakas, 2003). Some large DSS have their own fully integrated, multiple-source DSS databases. A separate DSS database need not be physically separate from the corporate database. They can be stored together physically for economic reasons. Some OLAP systems extract data, whereas others manipulate the data in the external database directly. Later, in DSS in Action 3.8, we describe a spreadsheet-oriented DSS for production planning and scheduling (see Respicio, Captivo, and Rodrigues, 2002). The DSS has a separate database, essentially in an Excel spreadsheet, that is populated with data extracted from a legacy database. Updates to the legacy database based on the DSS solutions are uploaded back.

A DSS database can also share a DBMS with other systems. A DSS database can include multimedia objects (e.g., pictures, maps, sounds) (Castelli and Bergman, 2002). Object-oriented databases in XML have been developed and used in DSS. These are becoming more important as m-commerce applications are deployed, because XML is becoming the standard, consistent data translation method for m-commerce devices (e.g., PDAs, cell telephones, notebook computers, tablet computer). The XML format is also used for standard Web browser access to data.

EXTRACTION

To create a DSS database or a data warehouse, it is often necessary to capture data from several sources. This operation is called extraction. It basically consists of importing of files, summarization, standardization filtration, and condensation of data. Extraction also occurs when the user produces reports from data in the DSS database. As will be shown in Chapter 5, the data for the warehouse are extracted from internal and external sources. The extraction process is frequently managed by a *DBMS*. *This extraction process is not trivial!* MIS professionals generally structure this process so that users need not deal with the complicated details. Much effort is required to structure the extraction process properly. To extract data, an exact query must be made to several related data tables that may span several independent databases. The pieces to be extracted must be "reconnected" so that a useful DSS database results. OLAP software like Temtec's Executive Viewer requires these actions before the OLAP may be used.

DATABASE MANAGEMENT SYSTEM

A database is created, accessed, and updated by a DBMS. Most DSS are built with a standard commercial relational DBMS that provides capabilities (see DSS in Focus 3.5).

THE CAPABILITIES OF A RELATIONAL DBMS IN A DSS C L ^

- Captures or extracts data for inclusion in a DSS database
- Updates (adds, deletes, edits, changes) data records and files
- Interrelates data from different sources
- Retrieves data from the database for queries and reports (e.g., using SQL via the Web)
- Provides comprehensive data security (e.g., protection from unauthorized access and recovery capabilities)
- Handles personal and unofficial data so that users can experiment with alternative solutions based on their own judgment
- Performs complex data manipulation tasks based on queries
- Tracks data use within the DSS
- Manages data through a data dictionary

An effective database and its management can support many managerial activities; general navigation among records, support for creating and maintaining a diverse set of data relationships, and report generation are typical examples. However, the real power of a DSS occurs when data are integrated with its models. (See DSS in Actions 3.2 and 3.8.)

THE QUERY FACILITY

In building and using DSS, it is often necessary to access, manipulate, and query data. The **query facility** performs these tasks. It accepts requests for data from other DSS components (Figure 3.4), determines how the requests can be filled (consulting the data directory if necessary), formulates the detailed requests, and returns the results to the issuer of the request. The query facility includes a special query language (e.g., SQL). Important functions of a DSS query system are selection and manipulation operations (e.g., the ability to follow a computer instruction, such as "Search for all sales in the Southeast Region during June 2004 and summarize sales by salesperson"). Though transparent to the user, this is a very important activity. All the user may see is a screen with a simple request for data, and following the click of a button, the user gets the results neatly formatted in a table in a dynamic HTML (or other Web-structured) page displayed on the screen.

THE DIRECTORY

The data **directory** is a catalog of all the data in the database. It contains data definitions, and its main function is to answer questions about the availability of data items, their source, and their exact meaning. The directory is especially appropriate for supporting the intelligence phase of the decision-making process by helping to scan data and identify problem areas or opportunities. The directory, like any other catalog, supports the addition of new entries, deletion of entries, and retrieval of information on specific objects.

All of the database elements have been implemented on database Web servers that respond to Web browser screens. The Web has dramatically changed the way we access, use, and store data.

3.7 THE MODEL MANAGEMENT SUBSYSTEM

The model management subsystem of the DSS is composed of the following elements:

- Model base
- Model base management system
- Modeling language
- Model directory
- Model execution, integration, and command processor.

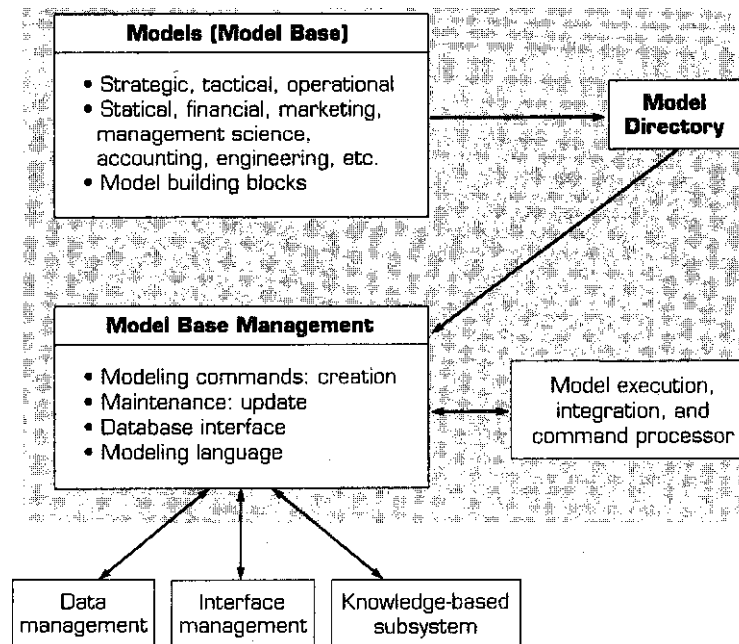
These elements and their interfaces with other DSS components are shown in Figure 3.5. The definition and function of each of these elements are described next.

MODEL BASE

A **model base** contains routine and special statistical, financial, forecasting, management science, and other quantitative models that provide the analysis capabilities in a DSS. The ability to invoke, run, change, combine, and inspect models is a *key DSS capability that differentiates it from other CBIS*. The models in the model base can be divided into four major categories: strategic, tactical, operational, and analytical. In addition, there are model building blocks and routines.

Strategic models are used to support top management's strategic planning responsibilities. Potential applications include devising an e-commerce venture, developing corporate objectives, planning for mergers and acquisitions, plant location selection,

FIGURE 3.5 THE STRUCTURE OF THE MODEL MANAGEMENT SUBSYSTEM



environmental impact analysis, and nonroutine capital budgeting. One example of a DSS strategic model is that of Southwest Airlines in the Opening Vignette. Southwest used the system to create accurate financial forecasts so that it could identify strategic opportunities. Another is described in the IMERYYS Case Applications at the end of Chapters 2, 4, and 6. The large-scale linear programming model is at the heart of the POP DSS that allows executives of the company to plan large, expensive equipment needs as many years ahead as needed.

Tactical models are used mainly by middle management to assist in allocating and controlling the organization's resources. Examples of tactical models include selecting a Web server, labor requirement planning, sales promotion planning, plant-layout determination, and routine capital budgeting. Tactical models are usually applicable only to an organizational subsystem, such as the accounting department. Their time horizons vary from one month to less than two years. Some external data are needed, but the greatest requirements are for internal data. When the IMERYYS POP DSS is used by managers in three- month to one-year time horizons, it is used as a tactical tool that determines how much clay they can produce to meet predicted market demand.

Operational models are used to support the day-to-day working activities of the organization. Typical decisions involve e-commerce transaction acceptance (purchases, etc.), approval of personal loans by a bank, production scheduling, inventory control, maintenance planning and scheduling, and quality control. Operational models mainly support first-line managers' decision-making with a daily to monthly time horizon. These models normally use only internal data. An excellent example of an operational model is the one developed by a large U.S. national bank with hundreds of branches (the officers of the bank wish it to remain anonymous). The bank developed an artificial neural network model to determine whether or not specific loan applicants should be given loans. The accurate predictions of the system allowed the bank to hold back on hiring additional loan officers, saving the bank some \$200,000 in its first year of operation for a development cost of about \$300,000. The POP DSS at IMERYYS is used operationally to determine exactly which clays to produce when over a two-week time horizon, over which the demand is known from actual contracted sales.

Analytical models are used to perform some analysis on the data. They include statistical models, management science models, data mining algorithms (see Chapter 4, and Hand, Mannila, and Smyth, 2001; Han and Kamber, 2000), financial models, and more. Sometimes they are integrated with other models, such as strategic planning models. The foundations of *business analytics* (the term was coined in the early 2000s) encompass all these analytical models. Typically, business analytics tools are Web-based, and that is why the term *Web analytics* was coined. These tools may readily be applied to Web systems; one example of their use is for administering and monitoring e-commerce. Business analytics software is generally easy to use. It includes OLAP, which is designed for use by managers or executives, as opposed to analysts, and data mining (see Hall, 2002a, 2002b; Langseth and Vivatrat, 2002).

The models in the model base can also be classified by functional areas (e.g., financial models, production control models) or by discipline (e.g., statistical models, management science allocation models). The number of models in a DSS can vary from a few to several hundred. Examples of DSS with several integrated models are described in DSS in Actions 3.2,3.8, and the Web Chapter on Procter & Gamble's redesign of its distribution system. Models in DSS are basically mathematical; that is, they are expressed by formulas. These formulas can be preprogrammed in DSS development tools such as Excel. They can be written in a spreadsheet and stored for future use, or they can be programmed for only one use.

MODEL BUILDING BLOCKS AND ROUTINES

In addition to strategic, tactical, and operational models, the model base can contain **model building blocks** and routines. Examples include a random number generator routine, a curve- or line-fitting routine, a present-value computational routine, and regression analysis. Such building blocks can be used in several ways. They can be employed on their own for such applications as data analysis. They can also be used as components of larger models. For example, a present-value component can be part of a make-or-buy model. Some of these building blocks are used to determine the values of variables and parameters in a model, as in the use of regression analysis to create trend lines in a forecasting model. Such building blocks are available in DSS commercial development software, such as the functions and add-ins of Excel, and in the general modeling structures of OLAP and data mining software. Since model solution methods have been implemented directly in Java and other Web development systems, access and integration of models has been simplified.

MODELING TOOLS

Because DSS deal with semistructured or unstructured problems, it is often necessary to customize models using programming tools and languages. Some examples of these are C++ and Java. OLAP software may also be used to work with models in data analysis. A Web-based system that uses a cluster analysis model for recommending movies is described in DSS in Action 3.6. For small and medium-sized DSS or for less complex ones, a spreadsheet (e.g., Excel) is usually used. We will use Excel for many key examples.

THE MODEL BASE MANAGEMENT SYSTEM

The functions of model base management system (MBMS) software are *model creation* using programming languages, DSS tools and/or subroutines, and other building blocks; *generation of new routines* and reports; *model updating* and *changing*-, and

DSS IN ACTION 3.6

A WEB-BASED DSS CLUSTER ANALYSIS METHOD MATCHES UP MOVIES AND THEATER-GOERS -

NetFlix.com (Los Gatos, California) provides movie recommendations to its 500,000 subscribers. The recommendations are provided by the subscribers themselves. But how do you go about identifying which movies are similar, so that you can make recommendations ("Customers who liked movie X also liked movie Y")? Canned software cannot evaluate the many subjective, on-the-fly reviews provided by tens of thousands of critics. NetFlix needed to do this to remain competitive. Enter *cluster analysis*. Mathematicians encoded cluster analysis software to define movie clusters, relate opinions to the clusters, evaluate thousands of ratings per second, and factor in current

Web site behavior to deliver a specially configured Web page before a customer can click again. The real-time analytics can also tell marketing managers what Web page design is working best for a given promotion. They can then change the Web page design immediately, based on the dynamic feedback. Cluster analysis is a very effective modeling tool that is used in customer relationship management systems (CRM) when trying to determine which products appeal to which customers.

Source: Adapted from Mark Hall, "Web Analytics: Get Real," *ComputerWorld*, Vol 36, No. 14, April 1, 2002, pp. 42-43.

DSS IN FOCUS 3.7

MAJOR FUNCTIONS OF THE MBMS

Creates models easily and quickly, either from scratch or from existing models or from the building blocks

Allows users to manipulate models so that they can conduct experiments and sensitivity analyses ranging from what-if to goal-seeking

Stores, retrieves, and manages a wide variety of different types of models in a logical and integrated manner

Accesses and integrates the model building blocks

Catalogs and displays the directory of models for use by several individuals in the organization

Tracks model data and application use

Interrelates models with appropriate linkages with the database and integrates them within the DSS

Manages and maintains the model base with management functions analogous to database management: store, access, run, update, link, catalog, and query

Uses multiple models to support problem solving

model data manipulation. The MBMS is capable of interrelating models with the appropriate linkages through a database (see DSS in Focus 3.7.)

THE MODEL DIRECTORY

The role of the model directory is similar to that of a database directory. It is a catalog of all the models and other software in the model base. It contains model definitions, and its main function is to answer questions about the availability and capability of the models.

MODEL EXECUTION, INTEGRATION, AND COMMAND

The following activities are usually controlled by model management. *Model execution* is the process of controlling the actual running of the model. *Model integration* involves combining the operations of several models when needed (e.g., directing the output of one model, say forecasting, to be processed by another one, say a linear programming planning model; see the IMERYS Case Applications 2.2 and 4.1, and DSS in Actions 3.2 and 3.8) or integrating the DSS with other applications. Portucel Industrial (a major Portuguese paper producer) uses a DSS that contains six integrated models: three capacity planning and scheduling models, two cutting plan models, and one demand forecasting model. (Respicio, Captivo, and Rodrigues, 2002; see DSS in Action 3.8).

A *model command processor* is used to accept and interpret modeling instructions from the user interface component and route them to MBMS, model execution, or integration functions.

An interesting question for a DSS might be: Which models should be used for what situations? Such model selection cannot be done by the MBMS because it requires expertise and therefore is done manually. This is a potential automation area for a *knowledge component* to assist the MBMS.

Another interesting, more subtle question is: What method should be used to solve a particular problem in a specific model class? For example, an assignment problem (say assigning 10 jobs to 10 people) is a type of transportation problem, which is a type of network flow problem, which is a type of linear programming problem, which is a type of mathematical optimization problem. Special solution methods are generally more efficient when dealing with more specialized structures. In other words, special methods for

DSS IN ACTION 3.8

PORTUCEL INDUSTRIAL S SPREADSHEET-BASED
DSS FOR PRODUCTION PLANNING
AND SCHEDULING IN THE PAPER INDUSTRY

Paper production planning and scheduling on a global level is a difficult problem. The tools necessary for solving it are typically quite difficult to understand and handle, and are rarely integrated in practice. Portucel Industrial (Portugal) developed a PC-based, spreadsheet DSS that utilizes six integrated models for paper production and scheduling. The system interacts with a human decision-maker who provides judgments as to the feasibility of plans. An exponential smoothing forecasting model (1) predicts product demands. Charts are produced for human interpretation. Three models perform capacity planning and scheduling. One model (2) assigns stock to client orders; a second (3) determines the acceptability of an order through effective capacity/aggregate demand ratio analysis; while the third (4) decomposes the problem into two subproblems to perform capacity planning and the actual scheduling. The next two models are used to determine how to cut the rolls of paper. The first (5) solves a cutting stock problem to determine the actual widths of the rolls to cut to meet all the orders. The second model (6) assigns the items to client orders in an attempt to minimize order spread (limit the waste). As these problems are solved,

the user may perform what-if analyses. These models are integrated in a PC-based DSS that exchanges data with the company's information system. Data are extracted daily from the IS into files that the spreadsheet-based system can import. The DSS generates local files for cutting plans, assignment of stock to client orders, and changes on active orders or proposed orders. The DSS exports these files to the IS, which updates the main database accordingly. Information about the cuts is automatically sent to the cutting machine on the shop floor.

The DSS provides many benefits. It is an easy-to-use tool that quickly generates and evaluates alternative solutions. The decision-maker can match these solutions against his or her knowledge and expertise. More rational and therefore better production decisions are made. Overall, costs are down and information is better organized. Production planning is better coordinated, leading to reduced lead times and an improvement in customer service.

Source: Adapted from Respicio, Captivo, and Rodrigues (2002).

solving an assignment problem should work better than applying transportation problem algorithms to it, and so on. But this is not always true. And to complicate matters, there may be many ways to solve a specific problem depending upon its characteristics. Again, there is potential for the knowledge component to assist in selecting an appropriate solution method. In the late 1990s, the elements of the model base management system migrated to Web-based systems, deployed as Java applets or modules of other Web development systems (see Fourer and Goux, 2002; Geoffrion and Krishnan, 2001).

3.8 THE USER INTERFACE (DIALOG) SUBSYSTEM

The term **user interface** covers all aspects of communication between a user and the DSS or any MSS. It includes not only the hardware and software but also factors that deal with ease of use, accessibility, and human-machine interactions. Some MSS experts feel that the user interface is the most important component because it is the source of many of the power, flexibility, and ease-of-use characteristics of MSS (Sprague and Watson, 1996a). Others state that the user interface is the system from the user's standpoint because it is the only part of the system that the user sees (Whitten, Bentley, and Dittman 2001). A difficult user interface is one of the major rea-

sons why managers do not use computers and quantitative analyses as much as they could, given the availability of these technologies. The Web browser has been recognized as an effective DSS graphical user interface because it is flexible, user friendly, and a gateway to almost all sources of necessary information and data (see Meredith, 2002). For a historical perspective and gallery of the graphical user interface, see Nathan Lineback's *Toasty Technology Web Page* (toastytech.com/guis/). For advances in interface research, see the PARC Inc. *User Interface @PARC Web Page* (www2.parc.com/istl/projects/uir/).

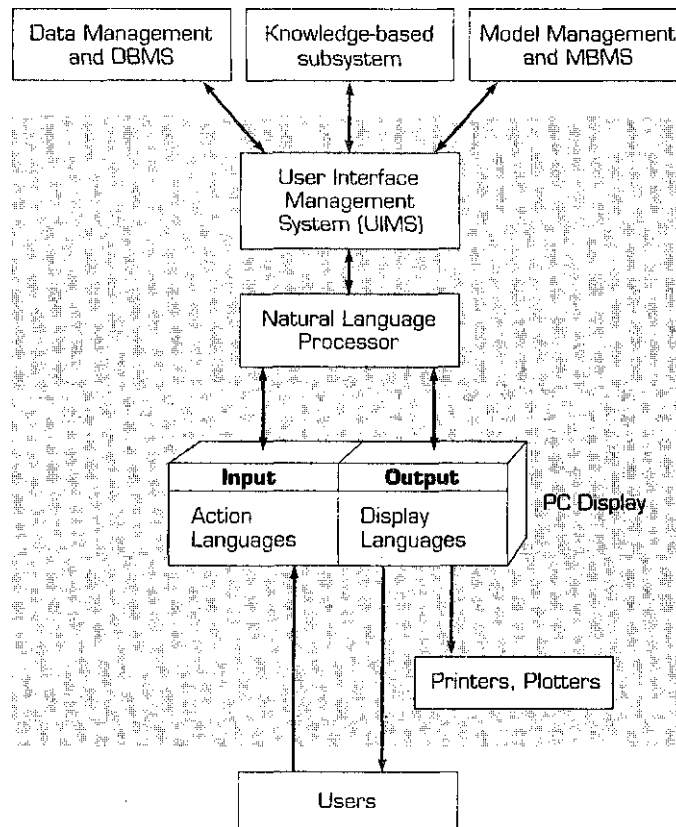
MANAGEMENT OF THE USER INTERFACE SUBSYSTEM

The user interface subsystem is managed by software called the **user interface management system** (UIMS). The UIMS is composed of several programs that provide the capabilities listed in DSS in Focus 3.10. The UIMS is also known as the dialog generation and management system.

THE USER INTERFACE PROCESS

The user interface process for an MSS is shown schematically in Figure 3.6. The user interacts with the computer via an action language processed by the UIMS. In

FIGURE 3.6 SCHEMATIC VIEW OF THE USER INTERFACE SYSTEM



DSS IN FOCUS 3.9

MAJOR CAPABILITIES OF THE UIMS

- | | |
|---|---|
| <ul style="list-style-type: none"> • Provides a graphical user interface, frequently using a Web browser • Accommodates the user with a variety of input devices • Presents data with a variety of formats and output devices • Gives users help capabilities, prompting, diagnostic, and suggestion routines, or any other flexible support • Provides interactions with the database and the model base ⁸ Stores input and output data • Provides color graphics, three-dimensional graphics, and data plotting | <ul style="list-style-type: none"> Has windows that allow multiple functions to be displayed concurrently Can support communication among and between users and builders of MSS Provides training by example (guiding users through the input and modeling processes) Provides flexibility and adaptiveness so the MSS can accommodate different problems and technologies Interacts in multiple, different dialog styles Captures, stores, and analyzes dialog usage (tracking) to improve the dialog system; tracking by user is also available |
|---|---|

advanced systems, the user interface component includes a natural language processor or can use standard **objects** (e.g., pull-down menus, buttons, Internet browser) through a **graphical user interface (GUI)**. The UIMS provides the capabilities listed in DSS in Focus 3.9 and enables the user to interact with the model-management and data management subsystems. A DSS user interface can be accessed from a cell telephone via either voice or the display panel. New, mobile DSS are being deployed directly on personal digital assistants (PDAs) that have a large memory, a quality graphical display, and wireless links through a built-in cell telephone or a direct Internet connection. PDAs can readily recognize a modified form of handwriting (e.g., Graffiti for Palm Pilots, palm.com). Advances in speech recognition technology create DSS interface opportunities (see DSS in Action 3.10). For example, Adomo provides a Mobile Communication Server that accesses Microsoft applications directly by voice over the telephone. These types of systems allow employees access to corporate applications directly over any telephone (Cohn, 2002). See Friley (2002) and Waters (2002) for more on speech recognition and associated technologies.

DSS IN ACTION 3.10

GIVING VOICE TO DSS APPLICATIONS

There are many reasons to build speech recognition and voice generation into DSS. One, of course, is for access of DSS via telephone. Another is accurate language translation—both verbal and textual. A third and most important one is that speech is a very natural way for humans to interact with one other. However, most computers do not *understand* the fuzzy nuances of human speech. For a computer to *interpret* the words of speech

is relatively easy, but understanding the *meaning* is fairly difficult. Artificial intelligence methods are often used. The good news is that speech recognition technologies have come a long way in the last decade (e.g., watch the captioning on CNN when a live story is broadcast). The bad news, however, is that they still have a long way to go before they can be used seamlessly in applications.

NEW USER INTERFACE DEVELOPMENTS

We have already mentioned voice/speech and handwriting recognition for its use for input, as well as direct translation of text into voice (which can even include gestures by a face on the screen—see annanova.com for an artificial newscaster). There are a number of new user-interface developments, mostly in laboratories, that may have an effect on how we use computers in decision-making and other tasks. For example, scientists are developing automatic, real-time, natural language translation (this requires speech recognition and generation), something that has challenged scientists and linguists for decades. As this book went to print, Sphinx (speech recognizer) and Carnival (speech synthesizer), developed at Carnegie Mellon's Language Technologies Institute, are making such language translation a reality. The quality and size of visual output displays are physically limited by the size of molecules. Even so, displays are getting better and better. Even PDAs and picture cell telephones have crisp images. Holographic displays that require neither specialized hardware nor glasses are just leaving the labs. LCD panels developed at Phillips Research have this capability. Scientists have experimented with helmets that detect brainwaves. Such a device could allow a quadriplegic the ability to interface with a computer. Tactile interfaces have been a bit of a problem. Immersion Corp.'s Cyberforce System includes a spandex glove that simulates the tactile sense that doctors get when performing surgery. This *haptic* interface allows surgeons to simulate their work before actually performing a real operation. In this way, medical students can experience virtual operations that feel so real that they have essentially performed the real thing. For videoconferencing, Microsoft has developed RingCam, an omnidirectional video camera that allows off-site meeting goers to view the entire room as if they were really at the meeting. It utilizes eight microphones and five small cameras. Finally, see DSS in Action 3.11 for a description of a *gesture interface* that utilizes holographic displays. See PC Magazine (2002) and Rhey (2002) for information on some of these developments.

NEW DEVELOPMENTS IN DECISION SUPPORT SYSTEMS

We conclude the sections on the three major DSS components with some recent technology and methodology developments that affect DSS and decision-making. In the

DSS IN ACTION 3.11

GESTURES IN THE AIR FOR INPUT

Spice (2002) reports on a human-automobile interface being developed at Carnegie Mellon University (Pittsburgh). Hand gestures (pointing, waving, etc.) toward icons projected onto the windshield are captured by cameras in the car and translated directly into instructions for adjusting the radio, putting someone on hold on your cell phone, or programming the on-board navigation system. This gesture interface can assist drivers in getting past the distractions caused by many electronic devices, whether a part of the car or brought on board. The goal is to increase safety. However, this new interface has implications for com-

puter interfaces in general. The next generation of PC interfaces may well be holographic in nature (see the "New Developments in DSS" subsection) or simply projected, and would be programmable. Gestures could be detected, instead of using hardware like a mouse or keyboard. There would be no moving parts, and the user would be able to use a set of preprogrammed gestures or could customize the system accordingly. In virtual reality settings, the "glove" that detects motion might become a relic. In addition to DSS, video games should benefit from the gesture interface technology.

preceding subsection, we described new technologies for the interface. Many developments in DSS components are the result of new developments in data warehousing, data mining, online analytical processing (OLAP), and World Wide Web technologies. Most DSS today access data from a data warehouse, and use models from OLAP or data mining tools. Data warehouses can provide petabytes of sales data for a retail organization. Data mining and OLAP systems provide integration with the data warehouse, the models, and often a very friendly user interface for DSS. Web communication technologies (Internet, intranets, extranets) provide links among the components, especially for accessing data sources and knowledge. Web browsers or Web-like user interfaces link users to the DSS. Web technologies enable virtual teams to collaborate, and provide access to integrated data, models, and knowledge components. For example, see DSS in Action 3.12 and the virtual environment of Andrienko, Andrienko, and Jankowski (2002). The Web has become the center of activity in developing DSS. Web-based DSS have reduced technological barriers and made it easier and less costly to make decision-relevant information and model-driven DSS available to managers and staff users in geographically distributed locations, especially through mobile devices. See Andrienko, Andrienko, and Jankowski (2002), Dong, Sundaram, and Srinivasan (2002), Eom (2002), Cachet (2002), Gregg (2002), Shim et al. (2002). We discuss some of these developments in Chapter 5.

There is also a clear link between hardware and software capabilities and improvements in DSS. Hardware continues to shrink in size while increasing in speed and other capabilities. However, we are reaching some physical limitations as to size and speed. Quantum computing (based on subatomic particle motion and charges) promises to break this barrier. By the end of 2002, a quantum system was capable of factoring the number 15. Though this seems to be a simple problem, it demonstrates the possibilities that quantum computing offers—very tiny, powerful computers. Artificial intelligence (see the next section) is making inroads in improving DSS. Faster, intelligent search engines are an obvious outcome. There are many others. For example, Desouza (2001) surveys applications of intelligent agents for competitive intelligence.

A fresh look at DSS evaluation was proposed by Phillips-Wren and Forgionne (2002). They developed an Analytical Hierarchy Process approach (see Chapter 2) toward evaluating Web-based real-time decision support systems in terms of criteria based on data, time, and effectiveness.

DSS IN ACTION 3.12

BLACKBOARD: A DSS FOR E-LEARNING

Blackboard Inc. (www.blackboard.com) offers a complete Web-based suite of enterprise software products and services that power a total "e-Education Infrastructure" for schools, colleges, universities, and other education providers. Blackboard solutions deliver the promise of the Internet for online teaching and learning, campus communities, auxiliary services, and integration of Web-enabled student services and back office systems. Blackboard provides a means of communication, collaboration, access to course materials (text,

data, software, etc.) and course tools (gradebook/grade reporting, e-mail, etc.), and so on. Essentially, Blackboard is a DSS for a course instructor and students. Blackboard is a course *portal* in the same sense as an enterprise information portal.

Sources: Blackboard, Inc. Web Site, www.blackboard.com, blackboard, Inc., Washington, DC, and Jay E. Aronson's personal experience using Blackboard for course management and delivery.

Some DSS in the future may include emotions, mood, tacit values, and other soft factors. This may be extremely important in dealing with health care choices, when the DSS is utilized by doctors, nurses, other caregivers, and patients. Though some of these factors were incorporated into the second generation of executive information systems, their importance is often overlooked.

Meredith (2002) proposed developing a multimedia, Internet-based DSS of this kind.

See Mora, Forgionne, and Gupta (2002) for a look at the future of DSS, and PC Magazine (2002) and Rhey (2002) for information on some technology developments.

3.9 THE KNOWLEDGE-BASED MANAGEMENT SUBSYSTEM

Many unstructured and even semistructured problems are so complex that their solutions require expertise. This can be provided by an expert system or other intelligent system. Therefore, more advanced DSS are equipped with a component called a *knowledge-based management subsystem*. This component can supply the required expertise for solving some aspects of the problem and provide knowledge that can enhance the operation of other DSS components.

Silverman (1995) suggests three ways to integrate knowledge-based expert systems (ES) with mathematical modeling: knowledge-based decision aids that support the steps of the decision process not addressed by mathematics (e.g., selecting a model class or a solution methodology); intelligent decision modeling systems that help users build, apply, and manage libraries of models; and decision analytic expert systems that integrate theoretically rigorous methods of uncertainty into expert system knowledge bases.

The knowledge component consists of one or more intelligent systems. Like database and model management software, knowledge-base management software provides the necessary execution and integration of the intelligent system. *Caution:* a knowledge management system is typically a text-oriented DSS; not a knowledge-based management system.

A decision support system that includes such a component is called an intelligent DSS, a DSS/ES, an expert-support system, active DSS, or knowledge-based DSS (see DSS in Action 3.13 for an example that includes both an expert system and an artificial neural network in a Web-based package written in Java). Most data mining applications include intelligent systems, such as artificial neural networks and rule induction methods for expert systems, to search for potentially profitable patterns in data. Many OLAP systems include artificial neural networks and data induction tools that extract rules for expert systems.

3.10 THE USER

The person faced with the decision that an MSS is designed to support is called the *user*, the *manager*, or the *decision-maker*. However, these terms fail to reflect the heterogeneity that exists among the users and usage patterns of MSS (Alter, 1980). There are differences in the positions that users occupy, their cognitive preferences and abilities, and their ways of arriving at a decision (decision styles). The user can be an individual or a group, depending upon who is responsible for the decision. The user, though not listed as a major component of DSS, by definition provides the *human intellect*. The

IAP SYSTEM'S INTELLIGENT DSS DETERMINES
THE SUCCESS OF OVERSEAS ASSIGNMENTS
AND LEARNS FROM THE EXPERIENCE

Overseas assignments for managers and executives can be an exciting adventure for the entire family; or a disaster. If an assignment is a failure, the cost of replacing the manager, and the impact on his or her family, can cost well over a quarter of a million dollars. Many companies (e.g., Coca-Cola) require employees to have overseas assignments before they can move into high executive positions. The critical issue is to be able to predict whether or not a specific assignment will be a good or bad experience for the manager and his or her family.

Enter Intelligent DSS. The International Assignment Profile (IAP) is a new, state-of-the-art method for use in ex-pat preparation (or selection) that collects key, comprehensive information about the family and compares their answers to known conditions in the anticipated international location.

IAP increases the human and business success of international assignments by spotting key issues and pinpointing the weak links or problems that could compromise an international relocation or assignment while there is still time to plan and prevent problems.

IAP's goals include:

- Better preparation for transfer
- Faster adjustment to international locations

- Significant reduction in compromised assignments
- No failed assignments

IAP is written in Exsys Corvid, a Web-based expert system shell (www.exsys.com). Through feedback from past assignments, artificial neural networks learn emerging patterns. IAP uses modern technology and *artificial intelligence* to assist companies in making more accurate, less stressful foreign placements and international relocations. The employee *and* his or her spouse complete the IAP interview process on the Web or on their computer. The system analyzes the information, detects and isolates critical patterns that might jeopardize the business purpose of the relocation, and produces a report for planning and problem prevention.

IAP produces a detailed list of exactly what issues need to be resolved and what planning needs to be done to ensure success. When the entire family is happy, the assignment succeeds. For a large firm, using IAP can readily save millions of dollars per year.

Source: Adapted from the International Assignment Profile Systems, Inc., Houston, TX Web site iapsystems.com, November 2002.

user, as the person or people primarily responsible for making the decision, provides expertise in guiding the development and use of a DSS. This intellectual capability is critical to the system's success and proper use. If the main user of a DSS is replaced by another, less knowledgeable user (in terms of the decision-making problem and environment), the DSS will generally be less effective.

An MSS has two broad classes of users: managers and staff specialists. Staff specialists, such as financial analysts, production planners, and marketing researchers, outnumber managers by about three to two, and use computers by a much larger ratio. When designing an MSS, it is important to know who will actually have hands-on use of it. In general, managers expect systems to be more user-friendly than do staff specialists. Staff specialists tend to be more detail-oriented, are more willing to use complex systems in their day-to-day work, and are interested in the computational capabilities of the MSS. That is why the first users of OLAP were staff specialists. Often, staff analysts are the intermediaries between management and the MSS.

An **intermediary** allows the manager to benefit from the decision support system without actually having to use the keyboard. Several types of intermediaries reflect different support for the manager:

1. Staff assistants have specialized knowledge about management principles and some experience with decision support technology
2. Expert tool users are skilled in the application of one or more types of specialized problem-solving tools. An expert tool user performs tasks that the problem solver does not have the skill or training to perform.
3. Business (system) analysts have a general knowledge of the application area, a formal business administration education (not in computer science), and considerable skill in using DSS construction tools.
4. Facilitators in a group support system control and coordinate the use of software to support the work of people working in groups, the facilitator also is responsible for the conduct of work-group sessions.

Within the categories of managers and staff specialists, there are important sub-categories that influence MSS design. For example, managers differ by organizational level, functional area, educational background, and need for analytic support. Staff specialists differ respect to education, functional area, and relationship to management.

Today's users are typically very hands-on oriented both in creating and using DSS (say through an OLAP), though they may need help from analysts in initially setting up access to needed data.

3.11 DSS HARDWARE

Decision support systems have evolved simultaneously with advances in computer hardware and software technologies. Hardware affects the functionality and usability of the MSS. The choice of hardware can be made before, during, or after the design of the MSS software, but it is often determined by what is already available in the organization. Typically, MSS run on standard hardware. The major hardware options are the organization's servers, mainframe computers with legacy database-management systems, a workstation, a personal computer, or a client/server system. Distributed DSS runs on various types of networks, including the Internet, intranets, and extranets (see Gachet, 2002; Gregg et al., 2002). Access may be provided for a number of mobile devices, including notebook PCs, tablet PCs, PDAs, and cell telephones. This portability has become critical for deploying decision-making capability (business intelligence) in the field, especially for salespersons (see Rothrock, 2002). A de facto hardware standard is that of a Web server through which the database management system provides data accessed from existing databases on the server, data warehouses, or legacy databases. Users access the DSS by client PCs (or other mobile devices) on which Web browsers are running. Models are provided directly through packages running on either the server, the mainframe, or some other external system, or even on the client PC. See Figure 3.1 for the architecture of what has become the typical DSS/business intelligence hardware configuration.

3.12 DSS CLASSIFICATIONS

There are several ways to classify DSS applications. The design process, as well as the operation and implementation of DSS, depends in many cases on the type of DSS involved. However, remember that not every DSS fits neatly into one category. We present representative classification schemes next.

ALTER'S OUTPUT CLASSIFICATION

Alter's (1980) classification is based on the "degree of action implication of system outputs" or the extent to which system outputs can directly support (or determine) the decision. According to this classification, there are seven categories of DSS (Table 3.4). The first two types are *data-oriented*, performing data retrieval or analysis; the third deals both with data and models. The remaining four are *model-oriented*, providing simulation capabilities, optimization, or computations that suggest an answer.

HOLSAPPLE AND WHINSTON'S CLASSIFICATION

Holsapple and Whinston (1996) classify DSS into the following six frameworks: text-oriented DSS, database-oriented DSS, spreadsheet-oriented DSS, solver-oriented DSS, rule-oriented DSS, and compound DSS.

J TEXT-ORIENTED DSS

Information (including data and knowledge) is often stored in a textual format and must be accessed by decision-makers. Therefore, it is necessary to represent and process text documents and fragments effectively and efficiently. A text-oriented DSS supports a decision-maker by electronically keeping track of textually represented information that could have a bearing on decisions. It allows documents to be electronically created, revised, and viewed as needed. Information technologies such as Web-based document imaging, hypertext, and intelligent agents can be incorporated into text-oriented DSS applications. There are many text-oriented DSS applications. Among them are electronic document management systems, knowledge-management, content management, and business rules systems. Content management systems (CMS) are used to manage the material posted on Web sites. Consistency, version control, accuracy, and proper navigation are handled directly by the system. See DSS in Action 3.14. Many freight and shipping companies (e.g., FedEx and UPS) use text-based DSS to coordinate shipping, help customers determine the best means to ship, and help customers and the company to track packages (see DSS in Action 3.4). In fact, FedEx has deployed a wireless handheld PC version of its system from which it expects to save \$20 million in annual costs (see Brewin, 2002).

J DATABASE-ORIENTED DSS

In this type of DSS, the database organization plays a major role in the DSS structure. Early generations of database-oriented DSS mainly used the *relational* database configuration. The information handled by relational databases tends to be voluminous, descriptive, and rigidly structured. A database-oriented DSS features strong report generation and query capabilities. Hendricks (2002) describes how the government of The Netherlands provides Web-based property management for intelligent decision-making. The system is primarily data-oriented and assists the government agency through standard and GIS databases in the effective use of its large portfolio of prop-

TABLE 3.4 Characteristics of Different Classes of Decision Support Systems

<i>Orientation</i>	<i>Category</i>	<i>Type of Operation</i>	<i>Type of Task</i>	<i>User</i>	<i>Usage Pattern</i>	<i>Time</i>
Data	File drawer systems	Access data items	Operational	Nonmanagerial line personnel	Simple inquiries	Irregular
	Data analysis systems	Ad hoc analysis of data files	Operational analysis	Staff analyst or managerial line personnel	Manipulation and display of data	Irregular or periodic
Data or Models	Analysis information systems	Ad hoc analysis involving multiple databases and small models	Analysis, planning	Staff analyst	Programming special reports, developing small models	Irregular, on request
Models	Accounting models	Standard calculations that estimate future results on the basis of accounting definitions	Planning, budgeting	Staff analyst or manager	Input estimates of activity; receive estimated monetary results as output	Periodic (e.g., weekly, monthly, yearly)
	Representational models	Estimating consequences of particular actions	Planning, budgeting	Staff analyst	Input possible decision; receive estimated results as output	Periodic or irregular (ad hoc analysis)
	Optimization models	Calculating an optimal solution to a combinatorial problem	Planning, resource allocation	Staff analyst	Input constraints and objectives; receive answer	Periodic or irregular (ad hoc analysis)
	Suggestion models	Performing calculations that generate a suggested decision	Operational	Nonmanagerial line personnel	Input a structured description of the decision situation; receive a suggested decision as output	Daily or periodic

Source: Condensed from Alter (1980), pp. 90-91.

erties. (Also see the Government Buildings Agency Property Management Web site: www.rijksgebouwendienst.nl.) DSS in Action 3.15 contains another example.

SPREADSHEET - ORIENTED DSS

A spreadsheet is a modeling system that allows the user to develop models to execute DSS analysis. These models not only create, view, and modify procedural knowledge³

³Procedural knowledge is generic knowledge regarding problem-solving procedures. In contrast, descriptive or declarative knowledge relates to the specific knowledge domain of the problem to be solved.

DSS IN ACTION 3.14

NOVANT HEALTH'S CONTENT MANAGEMENT
SYSTEM CREATES HEALTHY DOCUMENTS

At Novant Health (NC), a nonprofit health care system, 13,000 employees were generating and accumulating tons of documents on policy and procedures manual, patient education materials, and administrative and regulatory documents that needed to be posted to the organization's Web site by homemade tools. Ultimately, they implemented the Interwoven

TeamSite enterprise CMS. Since then, the IT department has been transformed to one that manages corporate information rather than maintain individual pages.

Source: Adapted from John Clymon, "From Chaos to Control," PC Magazine, September 17, 2002, pp. 125-133.

but also instruct the system to execute their self-contained instructions (macros). Spreadsheets are widely used in end-user developed DSS. (For examples, see Buehlmann, Ragsdale, and Gfeller, 2000; LeBlanc, Randalls, and Swann, 2000; Respicio, Captivo, and Rodrigues, 2002 [summarized in DSS in Action 3.8].) The most popular end user tool for developing DSS is Microsoft Excel. Excel includes dozens of statistical packages, a linear programming package (solver), and many financial and management science models,

Because packages such as Excel can include a rudimentary DBMS or can readily interface with one, they can handle some properties of a database-oriented DSS, especially the manipulation of descriptive knowledge. Some spreadsheet development tools include what-if analysis and goal-seeking capabilities, and these are revisited in Chapter 4. A spreadsheet-oriented DSS is a special case of a solver-oriented DSS.

DSS IN ACTION 3.15

DATABASE-ORIENTED DSS: GLAXO
WELLCOME ACCESSES LIFE-SAVING DATA

When Glaxo Wellcome revealed that a combination of two of its drugs, Epivir and Retrovir, was effective in treating AIDS, doctors began writing prescriptions en masse almost overnight. Such a tidal wave of demand could have resulted in lower inventories to pharmaceutical wholesalers and shortages.

Thanks to a data warehouse application, however, market analysts at Glaxo Wellcome were able to track the size and sources of demand and generate reports within hours, even minutes. The result: Wholesalers around the world never ran out of Epivir and Retrovir,

Called GWIS (Glaxo Wellcome Information System), the data warehouse application was built with MicroStrategy Inc.'s DSS relational online analytical processing (ROLAP) technology. GWIS works directly with data stored in a relational database-management

system, integrating internal data with data from external sources,

The application was rolled out in June 1996 to 150 employees in Glaxo Wellcome's marketing analysis department. Users can analyze sales, inventory, and prescription data for drugs on the fly, helping Glaxo Wellcome streamline its distribution process and cut operational costs. An additional IS benefit is that users can access information from various databases and computers. They no longer create local databases on their PCs which ultimately interfere with data integrity or require IT support. GWIS helps the IT organization design and manage the disparate data sources.

Source: Condensed from B. Fryer, "Fast Data Relief," InformationWeek, December 2, 1996, pp. 133-136; and www.microstrategy.com/customersuccesses, January 2000.

SOLVER - ORIENTED DSS

A solver is an algorithm or procedure written as a computer program for performing certain computations for solving a particular problem type. Examples of a solver can be an economic order quantity procedure for calculating an optimal ordering quantity or a linear regression routine for calculating a trend. A solver can be commercially programmed in development software. For example, Excel, includes several powerful solvers—*functions* and *procedures*—that solve a number of standard business problems. The DSS builder can incorporate the solvers in creating the DSS application. Solvers can be written in a programming language such as C++; they can be written directly on or can be an *add-in* tool in a spreadsheet, or they can be embedded in a specialized modeling language, such as Lingo. More complicated solvers, such as linear programming, used for optimization, are commercially available and can be incorporated in a DSS. For examples, see the Case Applications and examples in Chapter 4.

RULE - ORIENTED DSS

The knowledge component of DSS described earlier includes both procedural and inferential (reasoning) rules, often in an expert system format. These rules can be qualitative or quantitative, and such a component can replace quantitative models or can be integrated with them. For example, Bishop (1991) describes the integration of an assignment algorithm implementation (a form of linear programming) (Chapter 4) with that of an expert system for redirecting in-flight airplanes, flight crews, and passengers in the event that a major hub airport is knocked out of commission. Also see DSS in Action 3.17.

COMPOUND DSS

A compound DSS is a hybrid system that includes two or more of the five basic structures described earlier. See DSS in Action 3.16 for an example of a compound DSS.

DSS IN ACTION 3.16**COMPOUND DSS: FINANCIAL REPORTING, DECISION SUPPORT, AND EIS HELP T&N PREDICT THE FUTURE**

T&N is a leading world supplier of high-quality automotive components, as well as engineering and industrial materials. The company has an annual turnover of more than \$4.1 billion and employs 43,000 people throughout the world. The company formed an independent finance advisory division to improve company performance.

Operating units wanted detailed information at the product level; product groups wanted broader detail; management wanted strategic high-level summary and exception information (requiring three systems: financial reporting, decision support, and executive information), but all data had to be consistent.

A comprehensive MSS was initiated in the mid-1990s. Data are transmitted by e-mail to the Manchester (England) headquarters for the production of group accounts. This includes all accounting data, profit and

loss, analysis of expenditure, cash flow, and balance sheets.

T&N also stores explanation text in the database. The DSS is installed at all main consolidation points in the group, allowing rapid collection and aggregation of the data. The data are not seen simply as historical information, however. They are increasingly being used to help predict the future. T&N uses financial models and such techniques as simulation, stochastic forecasting, and statistical analysis of variance based on accurate information. This enables the firm to track resources more directly. The success of the DSS led to the completed implementation of an enterprise information system.

Source: Based on material at Comshare's Web site, comshare.com.

INTELLIGENT DSS

The **so-called** intelligent or knowledge-based DSS has attracted a lot of attention. The rule-oriented DSS that we described above can be divided into six types: descriptive, procedural, reasoning, linguistic, presentation, and assimilative. The first three are termed "primary" types, and the remainder are derived from them. Intelligent DSS are discussed in Parts IV and V of this book.

OTHER CLASSIFICATIONS OF DSS

There are several other classifications of DSS, such as the following.

INSTITUTIONAL AND AD HOC DSS

Institutional DSS (Donovan and Madnick, 1977) deal with decisions of a recurring nature. A typical example is a portfolio management system (PMS), which has been used by several large banks for supporting investment decisions. An institutionalized DSS can be developed and refined as it evolves over a number of years because the DSS is used repeatedly to solve identical or similar problems. It is important to

DSS IN ACTION 3.17

r.-cj

INSTITUTIONAL DSS: THE UNIVERSITY OF GEORGIA USES A WEB-BASED DSS FOR THE COURSE APPROVAL PROCESS

When the University of Georgia moved from the quarter to the semester system in 1998, there was a need to revamp the entire curriculum. Every course had to go through the entire course approval process, involving a lengthy paper trail with approve/modify/reject decisions made at every step. The workflow clearly needed to be automated, and decision-making embedded in the process. The Course Approval Process Automatic (CAPA) system was developed to support semester conversion issues with a work coordination and automation solution that used specific technology. Its objectives were to coordinate a decision-making process that involved multiple committees, dean's offices, departmental offices, the graduate school, and the vice president of academic affairs.

CAPA is a Web-based (intranet) system. It uses a two-tiered architecture. The Web server provides information to users, and the SQL database runs on another system in the background. Comments, approval, denial, or more work decisions are made every step of the way, and the results are recorded in the database.

The reason for using a Web server was so that the university could freely provide Web browsers for clients (access software for PCs on the various local area networks on campus). No additional hardware or software costs would be incurred by individual colleges and departments.

The principal benefits of CAPA are as follows:

- CAPA saves time and is cost-effective, especially for users.
- CAPA is flexible enough to support various related applications and is extensible, to support additional requirements.
- CAPA requires little or no user training and no new hardware or software.
- CAPA addresses long-term maintenance, management, and upgrade issues.

Appropriate information on courses can be accessed from the database to assist decision-makers at the departmental, college, and university levels. Information is current, and decisions on the courses are based on current information. Since the semester conversion, the CAPA system is the only course approval process at The University; no paper is used. And, the officially recognized courses and programs of study are those in CAPA, not in the annually printed bulletin. The University has since moved a number of other institutional systems to the Web, including registration and mid-semester course withdrawal.

remember that an institutional DSS may not be used by everyone in an organization; it is the recurring nature of the decision-making problem that determines whether a DSS is institutional versus ad hoc. See DSS in Action 3.17 for a description of an institutional DSS.

Ad hoc DSS deal with specific problems that are usually neither anticipated nor recurring. Ad hoc decisions often involve strategic planning issues and sometimes management control problems. Justifying³ DSS that will be used only once or twice is a major issue in DSS development. Countless ad hoc DSS applications have evolved into institutional DSS. Either the problem recurs and the system is reused, or others in the organization have similar needs that can be handled by the former ad hoc DSS. See DSS in Action 3.18 for a description of an ad hoc DSS that evolved into an institutional DSS.

PERSONAL, GROUP, AND ORGANIZATIONAL SUPPORT

The support given by DSS can be separated into three distinct, interrelated categories (Hackathorn and Keen, 1981):

Personal Support Here the focus is on an individual user performing an activity in a discrete task or decision. The task is fairly independent of other tasks. The situation described in DSS in Action 3.18 started with the development of a personal support DSS.

DSS IN ACTION 3.18

AD HOC VISUAL BASIC DSS HELPS CLOSE THE DEAL AND BECOMES AN INSTITUTIONAL DSS

No one needs to convince real estate agent Jim Rauschkolb about the value of information technology. A bad math error in 1980 turned him into a computer programmer and forever changed the way he sells property. In 1980 he sold a family's home and calculated, with a pencil and paper at their dining room table, what they were going to net. At the closing, he discovered that his calculations were off by \$1,800, which came out of his pocket. When this happened, Rauschkolb set out to develop a computer system that would remember every line item that needed to be calculated, do the math, and manage the increasingly complex interdependencies between details, such as the net gain from the sale of a home and the down payment on a new property. He learned how to program and built an ad hoc DSS. Using the software, he found that it was much easier to get people to sign a contract. He could show customers all the financial details up front in an easy-to-understand fashion, including whether they qualified for a mortgage. Furthermore, the calculations were done quickly and accurately. Rauschkolb, now a vice president in the

Century 21 office in Orinda, California, continued to use DSS and computers. In the late 1990s he needed to integrate applications and port them to a client/server development environment. Rauschkolb has ported three of his applications to Visual Basic (VB), with more on the way. One copyrighted application calculates the accurate cost of buying a house by analyzing sale price, equity, down payment, monthly mortgage, interest, income, and other factors. This process takes some agents hours to complete; Rauschkolb's program delivers accurate results in minutes. Having ported his applications to VB, Rauschkolb is now making the next logical move: distributing his programs to other agents for their PCs. He has packaged several of his applications with Web Capabilities, What starts as revenge against a math error ended as an outstanding Web DSS application—and the ad hoc application became an institutional DSS. And now it can be ported to .Net and to the Web.

Source: Condensed from R. Levin, "Visual Basic Helps Close the Deal," *InformationWeek*, November 4, 1996, pp. 16A-J7A.

Group Support The focus here is on a group of people, all of whom are engaged in separate but highly interrelated tasks. An example is a typical finance department in which one DSS can serve several employees all working on the preparation of a budget. If the use of an ad hoc DSS spreads, it becomes a group support DSS. Note that this is *not the same as* a group support system that provides collaboration and communication capabilities to a group working together.

Organizational Support Here the focus is on organizational tasks or activities involving a sequence of operations, different functional areas, possibly different locations, and massive resources. The Web-based CAPA system described in DSS in Action 3.17 at The University of Georgia provides organizational support for faculty, staff, and students.

INDIVIDUAL DSS VERSUS A GROUP SUPPORT SYSTEM (GSS)

Several DSS researchers and practitioners (such as Keen, 1980) point out that the fundamental model of a DSS—the lonely decision-maker striding down the hall at high noon to make a decision—is true only for minor decisions. In most organizations, be they public, private, Japanese, European, or American, most major decisions are made collectively. Working in a group can be a complicated process, and it can be supported by computers in what is called a **group support system (GSS)**. The Blackboard distance-learning system (DSS in Action 3.12; blackboard.com) provides support to all individuals and groups involved in a course. As a content management system it provides support to the group of students taking the course: it stores and distributes course materials. It supports the individual instructor through an online gradebook and a number of other tools that faculty need in course management. And, it functions as a GSS through its discussion lists, e-mail feature, and virtual classroom.

Note: The term *group support* introduced earlier should not be confused with the concept of group support system (GSS). In group support, the decisions are made by individuals whose tasks are interrelated. Therefore, they check the impact of their decision on others but do not necessarily make decisions as a group. In GSS, each decision (sometimes only one decision) is made by a group. Blackboard, just mentioned, is exceptional in that it does both.

CUSTOM-MADE SYSTEMS VERSUS READY-MADE SYSTEMS

Many DSS are custom-made for individual users and organizations (e.g., the Opening Vignette and the real estate DSS application in DSS in Action 3.18). However, a comparable problem may exist in similar organizations. For example, hospitals, banks, and universities share many similar problems. Similarly, certain nonroutine problems in a functional area (e.g., finance or accounting) can repeat themselves in the same functional area of different organizations. Therefore, it makes sense to build generic DSS that can be used (sometimes with modifications) in several organizations. Such DSS are called ready-made and are sold by various vendors (e.g., Cognos, Temtec, Teradata). Essentially, the database, models, interface, and other support features are built in: just add an organization's data and logo. For example, the IAP Systems application described in DSS in Action 3.13 is a ready-made DSS. The real estate applications described in DSS in Action 3.18 can also be viewed as a ready-made DSS, as can Blackboard. Recently, the number of ready-made DSS has been increasing because of their flexibility and low cost to develop them using Internet technologies for database access and communications, and Web browsers for interfaces (see DSS in Action 3.13).

One complication in terminology results when an organization develops an institutional system but, because of its structure, uses it in an ad hoc manner. An organization can build a large data warehouse but then use OLAP tools to query it and perform ad hoc analysis to solve nonrecurring problems. The DSS exhibits the traits of ad hoc and institutional systems, and also of custom and ready-made systems. We describe such a Web-based system in Case Application 3.2. Several ERP, CRM, KM, and SCM companies offer DSS applications online. These kinds of systems can be viewed as ready-made, though typically they require modifications (sometimes major) before they can be used effectively. See Chapter 8.

DSS AND THE WEB

Two recent developments in computer technology provide fertile ground for new or enhanced DSS applications. The first is Web technologies (Internet, intranet, and extranets), and the second is enterprise software, such as KM, ERP, CRM, and SCM (see Chapter 8). The power and capabilities of the World Wide Web are having a dramatic impact on DSS development, application, and use patterns. The link between the Web and DSS may be considered in two main categories: DSS development (Chapter 6) and DSS use.

DSS DEVELOPMENT

The Web can be used for collecting both external and internal (intranet) data for the DSS database. The Web can be used for communication and collaboration among DSS builders, users, and management. In addition, the Web can be used to download DSS software, use DSS applications provided by the company, or buy online from application service providers (ASPs). For example, see Fourer and Goux (2002) and Geoffrion and Krishnan (2001).

All major database vendors (e.g., IBM, Microsoft, Oracle, Sybase) provide Web capabilities by running directly on Web servers. Data warehouses, and even legacy systems running on mainframes or ported to small RISC workstations can be accessed through Web technologies. Typically models are solved on fast machines, but lately they have been ported to Web servers, either running in the background or accessed from other systems, such as mainframes. Optimization, simulation, statistics systems, and expert systems have been programmed to run in Java (see Fourer and Goux, 2002). These developments simplify access to data, models, and knowledge, and simplify their integration. Enterprise information systems/portals and OLAP systems provide powerful tools with which to develop DSS applications, generally via Web tools.

New software development tools, such as Java, PHP, and .Net, provide powerful on-screen objects (buttons, textboxes, etc.) for interfacing with databases and models. These readily open up direct access to the Web for the DSS developer. In many ways this simplifies the developer's tasks, especially by providing common development tools and a common interface structure through Web browser technologies.

DSS USE

The standard DSS interface is now the Web browser, or at least a similar-looking screen. Web browser technologies have changed our expectations of how software should look and feel. Many DSS provide drill-down capabilities (to look into data for the source of problems) and a traffic light display (green = OK, red = problems, yellow = problem brewing; see TemTec's Executive Viewer software). DSS is used on

FIGURE 3.7 SUMMARY OF DSS CAPABILITIES

Overall Capabilities		
Create variety of DSS Applications [specific DSS] quickly and easily Facilitate iterative design process		
General Capabilities		
Easy to use For routine use, modification, and construction of DSS	Access to a variety of data sources, types, and formats for a variety of problems and contexts	Access to a variety of analysis capabilities with some suggestion or guidance available
Component Capabilities		
User Interface	Data	Models
<ol style="list-style-type: none"> 1. Variety of output formats and devices 2. Variety of user input devices 3. Variety of dialog styles and ability to shift 4. Support communication among users and with builder 5. Support knowledge of users (documentation) B. Capture, store, analyze dialogs (tracking of dialogs) 7. Flexible and adaptive dialog support 	<ol style="list-style-type: none"> 1. Variety of data forms and types 2. Extraction, capture, and integration 3. Data access function <ol style="list-style-type: none"> a. retrieval/query b. report/display c. user/efficient data handling 4. Database management function 5. Variety of logical data views available B. Data documentation 7. Tracking of data usage 8. Flexible and adaptive data support 	<ol style="list-style-type: none"> 1. Library of models to constitute a model base <ol style="list-style-type: none"> a. many types b. maintain, catalog, integrate c. canned (preprogrammed) library 2. Model-building facility 3. Model-manipulation and use facility 4. Model base management functions 5. Model documentation 6. Tracking of model usage 7. Flexible and adaptive model support

Source: Based on Ralph Sprague and Eric Carlson, *Building Effective Decision Support Systems*, 1982, p. 313. Reprinted by permission of Prentice-Hall, Inc.

the Web in several ways. First, users can go on the intranet and activate ready-made DSS applications. All they need do is to enter some data, or specify dates and other information. The DSS is then run and they can see the results. For example, see Stihl's Chain Saw Assistant (www.stihlusa.com), which helps you select a chain saw (there are many product selection guides online). Second, they can get online advice and help on how to use the DSS applications. Third, they communicate with others regarding the interpretation of the DSS results. Finally, they can collaborate in implementing solutions generated by the DSS model. Web tools provide communication and collaboration capabilities for GSS and KMS, as well as for content management systems, EIS, CRM, and SCM.

3.13 SUMMARY

We have introduced the fundamentals of DSS. We began the chapter with a discussion of the Southwest Airlines vignette. We then covered the key DSS characteristics and capabilities. We summarize the major capabilities of DSS components (excluding the knowledge component) in Figure 3.7. For further details, see Daniel Power's DSS Web tour at dss.cba.uni.edu/tour/dsstour.html.

• CHAPTER HIGHLIGHTS

- There are several definitions of DSS.
- A DSS, also known as a business intelligence system, is designed to support complex managerial problems that other computerized techniques cannot. DSS is user-oriented, uses data, and models.
- DSS can provide support in all phases of the decision-making process and to all managerial levels for individuals, groups, and organizations.
- DSS is a user-oriented tool. Many applications can be constructed by end users.
- DSS can improve the effectiveness of decision-making, decrease the need for training, improve management control, facilitate communication, save effort by the user, reduce costs, and allow for more objective decision-making.
- The major components of a DSS are a database and its management, a model base and its management, and a user-friendly interface. An intelligent (knowledge-based) component can also be included. The user is also considered to be a component.
- The components of DSS are typically interconnected via Internet technologies. Web browsers are typically used as user interfaces.
- Data warehouses, data mining, and online analytical processing (OLAP) have made it possible to develop DSS quickly and easily.
- The data management subsystem usually includes a DSS database, a DBMS, a data directory, and a query facility.
- Data are extracted from several sources, internal and external.
- The DBMS provides many capabilities to the DSS, ranging from storage and retrieval to report generation.
- The model base includes standard models and models specifically written for the DSS.
- Custom-made models can be written in third- and fourth-generation languages, in special modeling languages, and in Web-based development systems (Java, etc.).
- The user interface (or dialog) is of utmost importance. It is managed by software that provides the needed capabilities. Web browsers often provide a friendly, consistent, and common DSS graphical user interface.
- The DSS is supplemented by the user's intellectual capabilities. The user is knowledgeable about the problem being solved.
- DSS can be used directly by managers (and analysts), or it can be used via intermediaries.
- DSS applications can be delivered and run on the Web. It is convenient to distribute them to remote locations.

• KEY WORDS

- ad hoc DSS
- business (system) analyst
- business analytics
- business intelligence
- data warehouse
- database
- database management system (DBMS)
- directory
- DSS application
- expert tool user
- extraction
- facilitator (in GSS)
- graphical user interface (GUI)
- group support system (GSS)
- institutionalized DSS
- intermediary
- Internet
- intranet
- model base
- model base management system (MBMS)
- model building blocks
- object
- operational models
- organizational knowledge base
- query facility
- staff assistant
- strategic models
- tactical models
- user interface
- user interface management system (UIMS)

• QUESTIONS FOR REVIEW

1. Provide two definitions of DSS. What do they have in common? What features differentiate them?
2. Why do people attempt to narrow the definition of DSS?
3. Give your own definition of DSS. Compare it to the definitions in Question 1.
4. List the major components of DSS and briefly define each of them.
5. What are the major functions (capabilities) of DBMS?
6. What is extraction?
7. What is the function of a query facility?
8. What is the function of a directory?
9. Models are classified as strategic, tactical, or operational. What is the purpose of such a classification? Give an example of each.
10. List some of the major functions of an MBMS.
11. Compare the features and structure of the MBMS to those of the DBMS.
12. Why is model selection for DSS difficult?
13. Define a text-oriented DSS.
14. What is the major purpose of a user interface system?
15. What are the major functions of a dialog (interface) management system?
16. List and describe the major classes of DSS users.
17. What types of support are provided by DSS?
18. Define the term ready-made DSS.
19. Compare a custom-made DSS with a ready-made DSS. List the advantages and disadvantages of each.
20. Search for a ready-made DSS. What type of industry is its market? Why is it a ready-made DSS?

• QUESTIONS FOR DISCUSSION

1. Review the major characteristics and capabilities of DSS. Relate each of them to the major components of DSS.
2. List some internal data and external data that could be found in a DSS for selecting a portfolio of stocks for an investor.
3. List some internal and external data in a DSS that would be constructed for a decision regarding a hospital expansion.
4. Provide a list of possible strategic, tactical, and operational models for a university, a restaurant, and a chemical plant.
5. Show the similarities between DBMS and MBMS. What is common to both and why? What are the differences and why?
6. Explain why DSS was the first MIS ever defined as requiring a computer.
7. Explain why a DSS needs a database management system, a model-management system, and a user interface, but *not* a knowledge-base management system.
6. Compare an individual DSS to a group DSS.
7. What are the benefits and the limitations of Holsapple and Whinston's classification approach?
8. Why do managers use intermediaries? Will they continue to use them in the future? Why or why not?
9. Explain why the user may be considered a component of the DSS.
10. Discuss the potential benefits that a DSS application can derive from the Web in terms of both developers and users.
11. Explain how the Web has impacted the components of DSS, and vice versa.

• EXERCISES

1. Susan Lopez has been made director of the transportation department at a medium-sized university. She controls the following vehicles: 17 sedans, 15 vans, and 3 trucks. The previous director was fired because there were too many complaints about vehicles not being available when needed. Susan has been told not to expect any increase in the budget for the next two years (meaning no replacement or additional Vehicles). Susan's major job is to schedule vehicles for employees and to schedule the maintenance and repair of these vehicles. All this was done manually by her predecessor. Your job is to consult with Susan regarding the possibility of using a DSS to improve the situation. Susan has a top-end PC and the newest version of Microsoft Office, but she uses the computer only as a word processor. She has access to the university's intranet and to the Internet. Answer the following questions:
 - a. Can the development and use of a DSS be justified? (That is, what can the DSS do to support Susan's job?)
 - b. What will be included in the data management, model management, and interface?
 - c. What type of support do you expect this DSS to render?
 - d. How would you classify this DSS?
 - e. Does it make sense to have a knowledge component?
 - f. Should the DSS be built, or should one be rented online? Why?
 - g. Should Susan disseminate the DSS to others on the intranet? Why or why not?
2. Consider the following two banking situations. A bank's marketing staff realizes that check-processing data which banks too often purge after a short period (60-90 days) could yield valuable information about customers' loan payment patterns and preferences.

The bank starts to retain these data using information discovery tools running on an advanced parallel-processing system to sort through checking account activity data to identify homeowner customers who pay mortgages by check on the fifth, sixth, or seventh day of the month. The bank targets these customers with a special home equity loan to consolidate debts, with automatic payment for the loan and the mortgage on the first of the month. The bank uses data mining tools to study levels of activity by affluent users over time in multiple channels: branches, automated teller machines (ATMs), telephone centers, and point-of-sale systems throughout all the regions the bank serves. It then takes the analysis to a second level: determining the profitability per transaction in each channel. Based on this initiative, the bank undertakes a comprehensive reengineering effort. Discovering that ATM and telephone banking are increasingly active and profitable, the management decides to focus resources and marketing efforts in expanding these channels. It decides to close full-service branches with low activity but replaces some with standalone ATM machines to continue providing customer service. Because some branches are still highly profitable and heavily used, management decides to expand the services offered at these locations. In both situations, identify the DSS applications that are used. Classify them according to the Alter scheme and according to the Holsapple and Winston scheme.

3. Find literature about an actual DSS application (use professional journals, ABI Inform, customer success stories on DSS vendors' sites, or the Internet for your search). In this application, identify the reasons for the DSS, the major components, the classification (type) of the DSS, the content of the model, and the development process and cost.

• GROUP PROJECT

1. Design and implement a-DSS for either the problem described in Exercise 1 or a similar real-world problem. Clearly identify data sources and model types,

and document the problems your group encountered while developing the DSS.

• INTERNET EXERCISES

1. Search the Internet for literature about DSS/ business intelligence/business analytics.
2. Identify a DSS/business intelligence/business analytics software vendor. Obtain information about its products. Write up your findings about its products in a report.

3. On the World Wide Web, find a DSS/ business intelligence/business analytics software vendor with downloadable demo software. Download the software, install it, and test it. Report your findings to the class and demonstrate the software's capabilities.

4. On the World Wide Web, identify a course syllabus and materials for a DSS/ business intelligence/ business analytics course at another college or university. Compare the course description to your own course. Repeat this assignment using a DSS/ business intelligence/business analytics course syllabus from a university in another country. Use www.isworld.org.
5. On the Web, identify several product selection guides that recommend specific products for you. Use five to ten of these, examine their positive and negative points, and describe their features and use in a report.
6. Explore the teradatauniversity.com site. In a report, describe at least three interesting DSS applications and three interesting DSS areas (CRM, SCM, etc.) that you have discovered there.

THE ADVANTAGE OF PETROVANTAGE: BUSINESS INTELLIGENCE/DSS CREATES AN E-MARKETPLACE

BACKGROUND

Aspen Tech supplies software to the process industries, and has carved out an important niche that in 2001 led to annual revenues of about \$380 million. With the release of PetroVantage, the 21-year-old company plans to streamline the processes for potentially lucrative industry petroleum. "The opportunity for companies to extract value using PetroVantage, from well head to gas pump, is substantial," said David McQuillin, Aspen Tech's chief operating officer and chief executive-elect. "The key part of this application is the trading and logistics capabilities." PetroVantage can save companies hundreds of thousands of dollars per day.

Industry analysts say the logistics of delivering petroleum from the wellhead to the consumer are among the most complicated of any industry. There are 500 types of crude oil, each with different characteristics; each refinery is unique, concentrating on different blends and end-product uses. Deciding what oil to buy and how to transport it involves an arcane process in which 20 to 25 worldwide traders make decisions that affect the international distribution process. These traders must integrate information on type, bulk, docking, refining, and delivery. They must know how much oil is coming out of the earth, where the ships are to transport it, what refineries can process the product, and what ports can accept the cargo. Then decisions must be made on how to transport the refined product to distributors. Critical analysis of these worldwide systems can be flawed, resulting in delays and losses. PetroVantage officials see an opportunity to launch a Web-based solution to modernize this immense process. "The world does more with petroleum than any other substance except water," said Chuck Moore, vice president of the petroleum business group at Aspen Tech. "We think there's a big opportunity here, especially because we will be leveraging some of the strengths of Aspen Tech."

THE DSS

PetroVantage is a suite of applications that enable a company to determine the best place to buy crude oil or any elements that make up different fuel mixtures, where to refine it, how to ship it, and how to distribute it to the retail sites. Engineers use parts of the application for every aspect of refinery or plant operations, including the design, building, cost, training, infrastructure and equipment, and maintenance of a facility. IBM provides the hardware, software, Web-hosting, and implementation infrastructure.

PetroVantage has developed online models that incorporate the attributes of about 600 of the world's 700 oil refineries. These attributes include production capacity and products produced. The marketplace provides an online platform for negotiating crude oil and oil products sales, evaluating deals, managing logistics, and linking key participants in complex crude oil trades. Traders use the system to buy, sell, and swap the physical barrel of crude oil and crude oil products, such as gasoline or jet fuel. The site's advantage lies in its ability to manage so many functions.

Decision support functions are what differentiate PetroVantage from other oil industry e-marketplaces, such as HoustonStreet.com and Altra Energy Technologies. The platform is unique because these kinds of decision support tools need to be based on some very complex models of what you can do with the crude oil.

The suite of applications falls into four main categories: end-to-end supply planning, refining solutions, fuel marketing, and, recently added, exploration and production. Perhaps the greatest return would be seen by those most familiar with procuring, trading, transporting, and storing oil and fuel. Moore said that 70 percent of fuel and crude oil distribution in the United States is handled by Aspen Tech's systems.

Developed from Anonymous, "PetroVantage Launches Commercial Software," *National Petroleum News*, Jan 2002, Vol. 94, No. 1, p. 54; William Copeland, "Accurate Inventory Tracking Means Opportunities Gained," *World Refining*, Vol. 11, No. 9, November 2001, p. 48; Matthew French, "Aspen Tech Fuels Up Its PetroVantage Product at Citgo," *Mass High Tech*, Vol. 20, No. 35, September 2, 2002, p. 8; Dyke Hendrickson, "Online Oil Exchange Heats Up," *Mass High Tech*, Vol. 18, No. 39, September 25, 2000, p. 1; Lewis, David, "Oil Exchange Lassos Big User—Occidental Joins Nine Other Customers in Pilot Test of E-marketplace," *InternetWeek*, Special Issue 872, August 6, 2001, p. 42.

PETROVANTAGE DEVELOPMENT AND PILOT TESTING

PetroVantage was pilot-tested by Citgo from early to mid-2002. The Tulsa, Oklahoma, company announced in September 2002 that it would deploy the PetroVantage across its entire enterprise to figure out cost-cutting measures and meet customer demands at 14,000 retail locations in 47 states. While the application isn't actually used at the retail level, all of the decision-making that takes place in the chain of command prior to that could rely on PetroVantage.

Williams R&M signed up with PetroVantage in the spring of 2001, and subsequently joined the PetroVantage Foundation Client Program. Williams R&M operates a refinery in Memphis with a capacity of 165,000 barrels per day. It pilot-tested PetroVantage to optimize its processing decisions, as well as crude oil logistics and refined-products distribution.

Occidental's marketing subsidiary, Occidental Energy Marketing Inc., joined another nine oil companies—including the \$11.6 billion Williams Cos. and spin-off Williams Energy Partners, as well as Midwest independent oil refiner Premcor—in testing PetroVantage in 2002. Occidental thinks the marketplace can help it wring better profit margins from crude oil trading. Occidental sells crude oil to wholesalers and brokers, as well as directly to refineries run by companies such as ExxonMobil.

The pilot program went well throughout 2001 and 2002, but the platform's long-term viability depends on other factors, especially the participation of the largest oil companies. PetroVantage is working to include futures and options, which some trading firms use to hedge against fluctuations in the price of oil.

RESULTS

The platform went live in September 2002. Its first commercial customers were Citgo, Premcor, Enron, and Williams Energy Partners. PetroVantage officials predict that their platform has the potential to achieve \$20 billion to \$30 billion in annual savings from the oil industry's logistics and trading costs of \$150 billion per year through its collaborative software solution.

Moore says, "If a company deals in a million barrels a day and you can save them even a few cents on each barrel, you're talking about a return of hundreds of thousands of dollars per day saved. Citgo deals in a *million barrels per day and 7 billion gallons of fuels per year.*"

Michael Cimino used to work in the trading and procurement of space, and now ensures its usability and mar-

ketability. "A trader using this software has a tremendous advantage over one who isn't," Cimino said. "A buyer can find, across the global market, a number of sellers and be able to determine within minutes what would be the best investment, based on what he or she already has and what they need."

Moore said that even market anomalies can be better dealt with using PetroVantage. The test came shortly after September 11, "When the airplanes around the country were grounded, oil and fuel companies were swimming in jet fuel, and had nowhere to unload it." Moore explained, "Using our solution, [a customer] was able to find the right deals to mix the fuel they had and turn it into diesel and home heating oil, and get it out of their hands ____ [A] process that normally takes several weeks was reduced to several days."

The Williams Cos. uses PetroVantage to simplify the trading process. Without PetroVantage, crude traders today might buy and sell on "several different electronic platforms, with a telephone in each ear to several brokers and the fax and e-mail going back and forth." "We like being able to go to a single site and pull everything together: what it costs to buy the components that make gasoline, the cost of arranging a barge or a ship, what kind of [storage] tankerage is available once it gets to port and what that's going to cost," said Bill Copeland, manager of terminal services for Williams Energy Partners. Williams also optimizes the scheduling of storage tanks to boost its terminals' profits. Occidental Marketing uses the e-marketplace to seek the best refineries to buy its oil at the most favorable price at a given time.

The PetroVantage collaborative software solution replaces the time-consuming data-gathering tasks and multiple approximations used in many of today's key trading and logistics decisions with fast and accurate optimization tools integrated with continuously updated data. It enables companies to identify costly deviations in operations, logistics, and deal margins. At the same time, it provides a means for faster and better coordination of responses from the many individuals across multiple companies and locations that are required to drive higher profitability in critical operations.

PetroVantage represents the next generation of digital marketplaces. "The company will offer a collaborative workflow environment that enables the petroleum industry to integrate state-of-the-art decision-support technology with an intuitive transaction platform, a feature no other petroleum industry marketplace currently offers" (PetroVantage Literature).

CASE QUESTIONS

1. How did the DSS/business intelligence tools provided by PetroVantage create and then assist decision-makers in the electronic marketplace?
2. Why was it important to perform pilot-testing with PetroVantage for almost two years?
3. How are customer supply chains integrated into PetroVantage?
4. What other features should be included in PetroVantage, and why?
5. Discuss the kinds of problems that can occur if the largest oil companies opt not to become customers of PetroVantage.
6. How could such a system provide benefits in other industries? Which are natural fits, which are not?

(AS* APPLICATION 1.1

FEDEX TRACKS CUSTOMERS ALONG WITH PACKAGES

INTRODUCTION

Federal Express Corp. is well-known for keeping track of its ever-moving overnight packages. It's one of the most important things the company does. In fact, there's only one thing that's more important for FedEx to track—its customer base. Until recently, FedEx wasn't doing a great job of quickly getting its business managers the information they needed to keep up with the company's fast-moving customers.

FedEx maintains a network of 46,000 U.S. drop-off points. But the company was not always sure that those points were in exactly the right (optimal) locations. New customers appear, old customers disappear, and some customers relocate. As businesses move from urban centers to suburban business parks, and as more and more individuals telecommute, FedEx wants its drop points, from large service centers to drop boxes, to be conveniently located for customers. But until recently FedEx managers did not have easy access to traffic information about its drop locations.

FedEx has a proprietary, mainframe-based Cosmos tracking and billing application that collects massive amounts of operational data, including where packages are picked up. But FedEx analysts could not easily access the data. Analysts submitted requests for custom reports (ad hoc use) to a staff of eight programmers, then waited for up to two weeks for a report. FedEx was using a mainframe version of Information Builders' FOCUS decision-support database to produce the reports. The old system did not support quick decision-making.

THE SOLUTION

FedEx decided to give analysts direct access to information. In June, the company deployed a Web-based version of the FOCUS database, WebFOCUS. The new system runs on the company's intranet and has a self-service data warehouse to help company executives make up-to-the-minute decisions about where it should locate the service centers and drop boxes that customers use every day. Data are downloaded from the Cosmos mainframe system to the WebFOCUS server running Windows NT. Analysts can query the data either by using a set of preconfigured reports (institutional use/ready-made DSS) or by creating their own ad hoc queries (ad hoc use/custom-made DSS).

FedEx evaluated several Web-based decision-support systems. It selected WebFOCUS primarily because the company already had programmers with FOCUS experience. That helped FedEx get an initial release of the intranet-based application deployed in just three weeks.

RESULTS

The self-service, intranet-based decision support system application makes it easier to get a more complete view of population shifts and other customer trends by combining the company's own drop point usage data with demographic data purchased from vendors. Programmers who had previously been developing reports from FedEx's mainframe FOCUS database have integrated external

data with the WebFOCUS data to allow analysts to anticipate and more accurately track customer trends.

Being able to anticipate customer trends is increasingly critical not only to FedEx also but to other companies in the distribution and logistics business. As companies such as FedEx try to link their distribution services directly into the supply chain operations of their large corporate customers, they need to make sure they have the support centers, trucks, and people in the right place at the right time.

FedEx expanded the system in several ways. First, the WebFOCUS database was extended to store 25 months of data instead of the original three months of historical shipment information. That increased the data warehouse's capacity from 21 million records to 260 million records, requiring a hardware upgrade.

FedEx is also improving the system's reporting capabilities. The company is rolling out the managed reporting features of WebFOCUS to allow analysts to schedule and create more predefined reports. FedEx is also deploying new applications in Information Builders' Cactus development tool to allow analysts to update and enhance drop-point data in the WebFOCUS database, not just read

it. With the new self-service data warehouse and planned enhancements, FedEx will have a better handle on tracking its fast-moving customers.

Redeploying the decision-support application on the intranet has already paid off in quicker access to information and quicker decisions. Analysts using WebFOCUS can tap directly into up-to-the-minute drop site usage data from any PC running a Web browser and get reports on their screens seconds instead of weeks. FedEx can more actively manage the location of its service centers and drop points as populations shift and customer habits change. The payoff is better customer service and lower operating costs.

In addition to more accurately tracking drop point usage, FedEx analysts can get fresh information on the profitability of each service center and drop box. Doing a better job of placing them will help cut costs and increase revenue. Have a look at the FedEx video on Information Builders' Web site (informationbuilders.com).

CASE QUESTIONS

1. Describe the benefits of the FedEx system. What other benefits might FedEx obtain with other features?
2. Why is it important for a company like FedEx to manage its drop locations effectively?
3. Describe the benefits of switching from FOCUS to WebFOCUS. Do you think this was the right approach? Why or why not?
4. How can the FedEx approach taken in this case be applied to other industries?