Chapter 10

Building Knowledge-Driven DSS and Mining Data

We can organize knowledge and make it available for business uses.

Introduction

Some people claim "knowledge leads to power". Even if that claim is true companies only win when knowledge is shared among employees and other stakeholders. Today sharing knowledge when making decisions is more important than most people recognize. One way to share knowledge is to build computerized systems that can store and retrieve knowledge codified as probabilities, rules and relationships. Specialized software can process this knowledge and assist managers in making decisions. Specialized decision support and artificial intelligence (AI) tools can also help create knowledge. An umbrella term that describes these systems is Knowledge-Driven Decision Support Systems. These DSS provide suggestions to managers and the dominant component is a "knowledge" capture and storage mechanism. Knowledge and suggestions are the two major themes that link these different knowledge tasks.

Knowledge-Driven DSS, Suggestion DSS, Rule-Based DSS and Intelligent DSS are overlapping terms for management support systems built using artificial intelligence technologies. We usually use expert systems development shells and data mining tools to create these systems. Business analysts identify relationships in very large databases using data mining or knowledge discovery tools. When a manager or knowledge worker uses a DSS with a data mining tool the results from an analysis may suggest relationships and new knowledge.

This chapter is an introduction and overview of Knowledge-Driven DSS technologies and applications. The first part of the chapter emphasizes expert system technologies and the second part emphasizes data mining techniques and tools. The overall thrust is to provide a foundation for building Knowledge-Driven DSS with specialized artificial intelligence tools. These technologies have been “hyped” by some vendors as solutions to a wide variety of problems, but artificial intelligence technologies are still “leading edge” capabilities for most businesses. At some point in the future all managers and knowledge workers may be using Knowledge-Driven DSS and mining data, but that future is over the horizon waiting to be implemented.

So the focus is on examining how we can use software to store and process knowledge for business decision-making and to find and derive knowledge for business decision-making. The following sections emphasize: Defining Key Terms and Concepts; Identifying Characteristics of Knowledge-Driven DSS; Managing Knowledge-Driven DSS Projects; Knowledge-Driven DSS Examples; Understanding Data Mining; Examples of Data Mining; and Evaluating Development Packages.

Key Terms and Concepts

Holsapple and Whinston (1996) discuss artificially intelligent DSS that “make use of computer-based mechanisms from the field of artificial intelligence”. These reasoning systems provide suggestions for business decision-makers and have the same general architecture as any other DSS. When a DSS uses artificial intelligence technologies including expert system technologies and some data mining tools to assist business decision-makers we will call the system a Knowledge-Driven Decision Support System. Every application of these technologies should not be called a decision support system. This section discusses key terms associated with Knowledge-Driven Decision Support Systems.

Knowledge-Driven DSS and Management Expert Systems

Knowledge-Driven Decision Support Systems store and apply knowledge for a variety of specific business problems. These problems include classification and configuration tasks such as Loan Credit Scoring, Fraud Detection and Investment Optimization.

Until recently, human experts had to perform this type of knowledge intensive task. Most of us identify a human expert as someone who is very knowledgeable in a particular area or subject. This human expert knows the appropriate questions to ask in order to draw a particular conclusion. In a similar way, one major type of expert system is a computer program that asks questions and reasons with the knowledge stored as part of the program about a narrow, specialized subject. This type of program attempts to solve a problem or give advice.

In general, expert systems are programs with specialized problem-solving expertise. The "expertise" consists of three components: 1) knowledge of symptoms related to a particular topic or domain, 2) understanding of the relations among symptoms, problems and solutions within that domain, and 3) "skill" or methods for solving some of the problems. An expert system is a knowledge-intensive program that captures the expertise of a human in a limited domain of knowledge and experience. It assists decision-makers by asking relevant questions in a problem domain and recommending actions and explaining reasons for adopting an action.

An expert system can explain the reasoning behind a conclusion it has reached. This explanation capability is extremely important in auditing and validating the results from a

Knowledge-Driven DSS. It also helps ensure the system is in compliance with applicable policies, regulations or legal requirements.

Knowledge-Driven DSS and management expert systems have a number of benefits. Such systems can improve consistency in decision-making, enforce policies and regulations, distribute expertise to non-expert staff and retain valuable expertise for a company when experts retire or resign.

Data Mining and Knowledge Discovery

Data mining and knowledge discovery are “hot” topics in the Information Systems and Marketing trade press. For many years companies have been storing large amounts of data and more recently companies have built large data warehouses. Now managers want to take advantage of the data they have collected by analyzing it using statistical and artificial intelligence tools. Data mining techniques can help managers discover hidden relationships and patterns in data. Some analysts feel data mining can help a company gain a competitive advantage. Data mining tools can be used for both hypothesis testing and knowledge discovery. When vendors discuss data mining, they may be selling a set of end-user tools or a decision support capability or both. Managers and business analysts can perform data mining activities. Target users of these tools include financial analysts, statisticians and marketing researchers. People who use these DSS and tools should have experience interpreting data.

Other Important Terms

Some artificially intelligent decision aids act for managers and are called agents. An **agent** is a self-contained program that runs in the background on a client or server and performs useful functions for a specific owner. Agents may monitor exceptions based on criteria or execute automated tasks. For example, once an event occurs an agent performs a pre-defined action and then it returns to a monitoring state. An agent might monitor sales goals or levels of product defects. Agents can be incorporated in a variety of DSS. The term is important to know, but how an agents works is beyond the purpose of this book.

A **development environment** is used by a Knowledge-Driven DSS designer and builder. A development environment typically includes software for creating and maintaining a knowledge base and software called an inference engine. An **inference engine** reasons with a set of rules created by a developer.

A **domain expert** is a key person in a Knowledge-Driven DSS development project. A domain expert is the person who has expertise in the domain in which a specific system is being developed. A domain expert works closely with a **knowledge engineer** to capture the expert's knowledge in a knowledge base. This process is used especially for capturing rule and relationship information in a computer readable format.

Knowledge acquisition is the extraction and formulation of knowledge derived from various sources, especially from experts. A **knowledge base** is a collection of organized facts, rules, and procedures. A knowledge base has a description of the elements in the process along with their characteristics, functions, and relationships. It also contains rules about the actions to implement as a result of certain events. A knowledge base can also obtain its information from external programs and databases. When dealing with a particular task or problem, a Knowledge-Driven DSS constructs a number of hypotheses based on the external information supplied, its own knowledge, and the rules in its knowledge base.

As the above paragraphs amply demonstrate, artificial intelligence researchers have an extensive technical jargon that managers and MIS professionals must have some familiarity with if they want to build Knowledge-Driven DSS. The key to success is learning some of the jargon, and staying focused on the broader objective of building decision support systems that use software with “reasoning” capabilities.

Characteristics of Knowledge-Driven DSS

We can identify a number of characteristics that are common to Knowledge-Driven DSS. First, this category of software aids managers in problem solving. Second, the systems use knowledge stored as rules, frames, or likelihood information. Third, people interact with a program when they are performing a task. Fourth, Knowledge-Driven DSS base recommendations on human knowledge and assist in performing very limited tasks. Fifth, Knowledge-Driven DSS and expert systems do **NOT** "think".

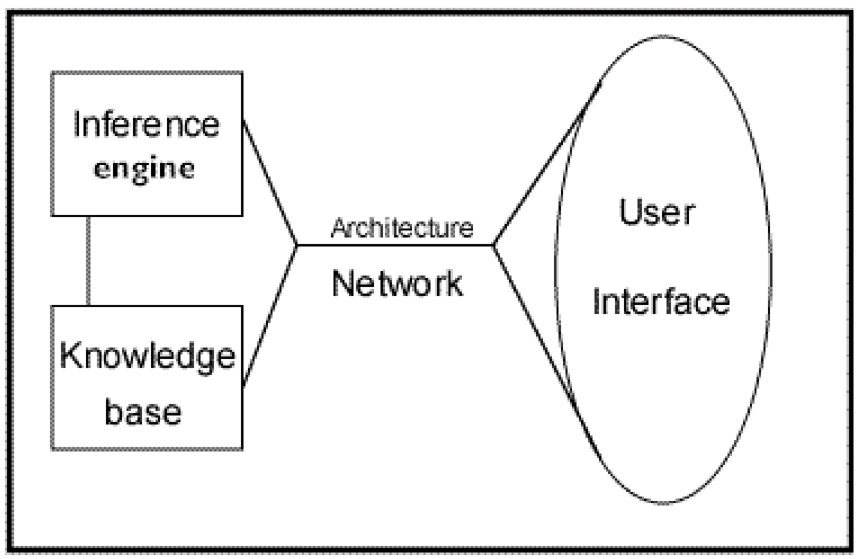

Figure 10.1 Knowledge-Driven DSS Components.

A Knowledge-Driven DSS differs from a more conventional Model-Driven DSS in the way knowledge is presented and processed. This difference exists because most expert systems attempt to simulate human reasoning processes. A Model-Driven DSS has a sequence of predefined instructions for responding to an event. In contrast, a Knowledge-Driven DSS based on expert system technologies attempts to reason about a response to an event using its knowledge base and logical rules for problem solving. Expert system technologies use representations of human knowledge. These representations are expressed in a special purpose language such as OPS5, PROLOG or Lisp. Expert systems can also perform standard numerical calculations or data retrieval. An expert system development environment uses heuristic methods to obtain a recommendation. A heuristic is an approximate method that identifies varying amounts of uncertainty in conclusions. A conventional Model-Driven DSS uses mathematical and statistical methods to obtain a more precise solution.

Figure 10.1 shows the components of a Knowledge-Driven DSS. The inference engine is the software that actually performs the reasoning function. In small systems, this is sometimes called the shell of the expert system, though the shell can be considered to be everything except the knowledge base itself. The inference engine is the software that uses the knowledge represented in the knowledge base to draw its conclusions. The design of the inference engine may limit the ways in which knowledge can be represented in the knowledge base so that certain shells are only suitable for particular types of applications.

In comparing Knowledge-Driven DSS and Model-Driven DSS, we should remember that:

Knowledge-Driven DSS = Knowledge Base + Inference Engine

Model-Driven DSS = Data + Quantitative Models

Managing Knowledge-Driven DSS Projects

Knowledge-Driven DSS should be initiated with a decision-oriented diagnosis and if the feasibility analysis is positive, then a small project team should complete a rapid prototyping development process. Many Knowledge-Driven DSS are built using rules and an expert system shell development environment. A knowledge engineer works with a domain expert to elicit rules and relationships. The testing and validation of the system may involve using prior examples and cases from the domain.

Several general rapid prototyping approaches for developing Expert Systems and Knowledge-Driven DSS have been proposed. Waterman (1986) proposed the following widely accepted approach: 1. Identification of a domain; 2. Conceptualization; 3. Formalization; 4. Implementation; and 5. Testing. These five stages are highly interrelated and interdependent. An iterative process continues until the Knowledge-Driven DSS consistently performs at an acceptable level.

Choosing a Knowledge-Driven DSS Project

If a business decision problem cannot readily be solved and supported using traditional methods it may be appropriate to try an expert system solution. How do we choose an appropriate Knowledge-Driven DSS project? In general, the "telephone test" can be used to help determine if a task can be supported with a Knowledge-Driven DSS nuilt using expert systems technologies. What is the “telephone test”? To apply the test, we ask “Can a domain expert solve the problem and support decision-making using a telephone exchange with a decision-maker?” Sometimes it is even helpful to ask the domain expert to interact with a potential user of a new DSS over the telephone and record the interaction that occurs. The domain expert should be told to ask structured rather than open-ended questions. If the answer is **YES** the telephone exchange works, then a Knowledge-Driven DSS based on expert systems technologies can be developed to support the decision-maker. On the other hand, if the decision-maker is unable to describe the problem verbally, or if the expert is unable to consistently recommend a reasonable solution, then development of a Knowledge-Driven DSS will likely be unsatisfactory. The telephone test assures that the expert is not gaining additional information about a problem from other senses and insures that the user is able to adequately describe the problem in words.

Using Rules for Knowledge-Driven DSS

Managers and developers should be familiar with the concept of a rule. A rule-based expert system has a large number of interconnected and nested IF-THEN statements or "rules" that are the basis for storing the knowledge in the system. Many expert system development environments store knowledge as rules.

The following is an example of a rule:

IF INCOME > $45,000 (condition)

AND IF SEX = "M" (condition)

THEN ADD to Target list (action)

A rule is a formal way of specifying a recommendation, directive, or strategy, expressed in an **IF** premise **THEN** conclusion structure. Rules are one way of expressing declarative knowledge. For example, IF a car won't start, AND the lights are dim, THEN the car may have a dead battery. Thus, rules are relationships rather than instructions. Note this structure is different than the IF-THEN structure used by procedural programming languages.

There are two ways an inference engine can manipulate rules. The first is forward chaining, where an inference engine starts from known facts and looks at the left-hand IF side of the rules to find any matches and proceeds to find further rules that apply to the user's responses. The second method is known as backward chaining. This technique involves starting the inference engine with a hypothesized solution by looking at the right-hand THEN statements, then working backwards to find the starting conditions that are necessary to arrive at that solution and see how they match with the user's responses.

Let's examine the two approaches from a different perspective. Suppose you want to fly from Waterloo, IA to Beijing, China. To find a "chain" of connecting flights you can search in one of two ways:

1. Start with flights that arrive in Beijing and work backwards to eventually find a chain to Waterloo, IA. This is a goal-driven, backward chaining search, or,

2. Start by listing all flights leaving Waterloo and mark intermediary cities. Look for flights out of intermediaries until you find all paths to Beijing. You are working forward toward your goal. This is a data-driven, forward chaining search.

Advantages and Limitations of Rules

Using an inference engine with rules is the most common development environment for Knowledge-Driven DSS. Rules are easy to understand. Also, explanations are easy to provide when you store knowledge as rules. From a developer's perspective, modification and maintenance of the knowledge base is relatively easy. A developer can also easily combine uncertainty knowledge with rules. There are, however, a number of major limitations of using this development approach. First, and most important, complex knowledge is difficult to represent using rules. Also, when rules are used the knowledge represented tends to be superficial. Knowledge-Driven DSS builders usually like developing systems based on rules, but using rules will not work for all applications.

Knowledge-Driven DSS Examples

Two classic examples of business expert systems are TAXAdvisor and XCON. More recent examples include a Scheduling System for the Tomakomai Paper Mill, a Customer Support System at Compaq Computer, and an Insurance Plan Selection System for Meiji Mutual Life Insurance Company.

TAXADVISOR was a Knowledge-Driven DSS designed to assist an attorney with tax and estate planning for clients with large estates (greater than $175,000). The system collected client data and inferred actions the clients need to take to settle their financial profile, including insurance purchases, retirement actions, transfer of wealth, and modifications to gift and will provisions. TAXADVISOR used knowledge about estate planning based on attorneys' experiences and strategies as well as more generally accepted knowledge from textbooks. The system used a rule-based knowledge representation scheme controlled by backward chaining. TAXADVISOR was implemented in EMYCIN. It was developed at the University of Illinois, Champaign-Urbana, as a PhD dissertation and it reached the stage of a research prototype.

XCON (e**X**pert **CON**figurer of VAX 11/780 computer systems) was developed to configure computer systems. Based upon a customer's order it decided what components needed to be included to produce a complete operational system and it determined the spatial relationships among all of the components. XCON was implemented in an expert system shell called OPS5 and was developed through a collaboration between researchers at Carnegie-Mellon University and Digital Equipment Corporation (now Compaq). This commercial expert system configured VAX computers on a daily basis and was for many years the largest and most mature rule-based expert system in operation. XCON was not actually a DSS because it made decisions rather than supporting managerial decision-making.

Scheduling systems and control systems are needed in the paper production industry to ensure that all plants in the mill operate correctly. At Tomakomai Mill an expert system is used to schedule the paper production machines. The Tomakomai Mill consists of ten paper making machines, energy supply plants and pulp supply plants. Two hundred paper products are produced per month. Each product has a specified production volume and due date, and requires a specified machine to produce. In the Tomakomai Mill, a millwide production management system exists. The system has a planning level and a control/operation level. The scheduling system is situated on the planning level for papermaking. It receives product orders from the headquarters office, makes a schedule and delivers it to the other planning systems. Each system schedules and optimizes its operations based on the papermaking schedule. The paper production scheduling system consists of an expert system for automated scheduling, and a data management system. The expert system consists of three subsystems: product group scheduling system, individual scheduling system, and a balancing scheduling system. The scheduling systems are implemented with an expert shell utility, ASIREX. This scheduling system has been in practical use since January 1989. The greatest advantage of this system is that it speeds up scheduling. The scheduling time for a monthly schedule was reduced from 3 days to 2 hours.

Compaq Computer Corporation created and implemented a very successful **Customer Support Intelligent System** to provide computer users expert diagnosis and recommendations about problems with hardware, software, network and other problems (cf., Dhar and Stein, 1997).

The Meiji Mutual Life Insurance Company is one of the oldest life insurance companies in Japan with assets of around $74 billion. Meiji offers a wide range of insurance and pension products. In addition, the company is aggressively involved in developing and introducing new products. However, with the increasing number of products, the company was finding it difficult to ensure that all the insurance sales staff had the expertise and the latest knowledge required to provide the best advice and service to its customers. To overcome this problem, Meiji used XpertRule to develop the **Life Insurance Plan Selection Expert System**. The system can select the most suitable product, along with a reason for the choice, from Meiji's range of 37 individual oriented products. Meiji began research into expert systems in 1986. Before using XpertRule, the company had completed a Lisp-based insurance plan selection system. This system, however, had a high delivery and maintenance cost and was not suited for distribution to all branches. Meiji adopted XpertRule because it allows for easy knowledge base construction. The knowledge base contained 47 decision tasks. The rules for selecting each plan were developed as a separate task. The system was structured so that when the details of a customer are entered, the system assesses the suitability of all the plans and report on the best five. The system only takes 3 to 4 seconds to make suitable selections (cf., http://www.attar.co.uk/pages/case_ml.htm).

Data Mining and Creating Knowledge

In the 1970s companies employed business analysts who used statistical packages like SAS and SPSS to perform trend analyses and cluster analyses on data. As it became possible and affordable to store large amounts of data, managers wanted to access and analyze transaction data like that generated at a retail store cash register. Bar coding and the World Wide Web have also made it possible for companies to collect large amounts of new data.

Database marketing has also benefited from mining data. The information incorporated in the database marketing process is the historical database of previous mailings and the features associated with the (potential) customers, such as age, zip code, their response in the past. Data mining software uses this information to build a model of customer behavior that can be used to predict which customers are most likely to respond to a new catalog. By using this information a marketing manager can target the customers likely to respond (cf., Thearling http://www3.shore.net/~kht/index.htm).

For many years companies had statisticians study company data. When a statistician looks at the data, he or she makes a hypothesis about a relationship, then performs a query on a database and uses statistical techniques to prove or disprove the hypothesis. This has been called the "verification mode" (IBM, 1998). Data mining software works in a "discovery mode." Data mining software looks for patterns. No hypothesis is established before the data analyzed.

There are two main kinds of models in data mining: predictive and descriptive. Predictive models can be used to forecast explicit values, based on patterns determined from known results. For example, from a database of customers who have already responded to a particular offer, a model can be built that predicts which prospects are likeliest to respond to the same offer. The predictive model is then used in a DSS. Descriptive models describe patterns in existing data, and are generally used to create meaningful subgroups such as demographic clusters. Once a descriptive model is identified it may be used for target marketing or other decision support tasks

Data Mining Techniques and Tools

There are a wide variety of tools for data mining. The decision about which technique to use depends on the type of data and the type of questions that managers want answered by that data. Many commercial data mining software packages include more than one data mining tool. A summer 2000 Kdnuggets.com poll indicated SPSS's (spss.com) Clementine is the most used data mining software package. It is targeted to business users and it is a visual rapid modeling environment. Advanced sources of information on data mining tools include Berry and Linoff (1997) and Dhar and Stein (1997). This section examines five common categories for data mining tools: Case-Based Reasoning, Data Visualization, Fuzzy Query and Analysis, Genetic Algorithms, and Neural Networks (cf., Greenfield, 1999).

Case-Based Reasoning

Case-based tools find records in a database that are similar to specified records. A user specifies how strong a relationship should be before a new case is brought to her attention. This category of tools is also called memory-based reasoning. Software tries to measure the "distance" based on a measure of one record to other records and cluster records by similarity. This technique has been successful in analyzing relationships in free-form text. The Web site www.ai-cbr.org is a resource for the artificial intelligence and case based reasoning technology fields. At the site there is a large list of links to Case-Based Reasoning tool vendors and consultants.

Case-based tools are used with a five-step problem-solving process:

Presentation: a description of the current problem is input to the system.

Retrieval: the system retrieves the closest-matching cases stored in a database of cases.

Adaptation: the system uses the current problem and closest-matching cases to generate a solution to the current problem.

Validation: the solution is validated through feedback from the user of the environment.

Update: if appropriate, the validated solution is added to the case base for use in future problem solving (cf., Allen, 1994).

Data Visualization

These tools graphically display complex relationships in multi-dimensional data from different perspectives. Visualization is the graphical presentation of information, with the goal of providing the viewer with a qualitative understanding of the information contents. Data visualization tools are data mining tools that translate complex formulas, mathematical relationships or data warehouse information into graphs or other easily understood models. Statistical tools like cluster analysis or classification and regression trees (CART) are often part of data visualization tools. Analysts can visualize the clusters or examine a binary tree created by classifying records. In a marketing, an analyst may create “co-occurrence” tables or charts of products that are purchased together. A good visualization is easy to understand and interpret and it is a reasonably accurate representation of the underlying data.

Fuzzy Query and Analysis

Fuzzy data mining tools allow users to look at results that are "close" to specified criteria. The user can vary what the definition of "close" is to help determine the significance and number of results that will be returned. This category of data mining tools is based on a branch of mathematics called fuzzy logic. The logic of uncertainty and “fuzziness” provides a framework for finding, scoring, and ranking the results of queries. Fuzzy Tech, a company that develops Fuzzy query software, has a Web site with excellent information on this tool at http://www.fuzzytech.com/index.htm.

Genetic Algorithms:

Genetic algorithms are optimization programs similar to the linear programming models discussed in Chapter 9. Genetic algorithm software conducts random experiments with new solutions while keeping the “good” interim results. A example problem would be to find the best subset of 20 variables to predict the stock market. To create a genetic model, the 20 variables would be identified as “genes” that have at least 2 possible values. The software would then select genes and their values randomly in an attempt to maximize of minimize a performance or fitness function. The performance function would provide a value for the fitness of the specific genetic model. Genetic optimization software also includes operators to combine and mutate genes. This quantitative model is used to find patterns, like other data mining techniques.

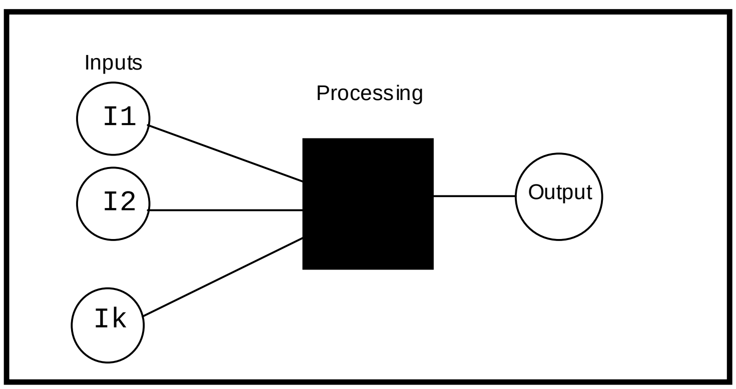

Figure 10.2 Neural Network Example.

Neural Networks:

Neural network tools are used to predict future information by learning patterns and then applying them to predict future relationships. According to Berry and Linoff (1997), neural networks are the most common type of data mining technique. Some people even think that using a neural network is the only type of data mining. Vendors make many claims for neural networks. One claim that is especially questionable is that neural networks can compensate for a lower quality of data. Neural networks attempt to learn patterns from data directly by repeatedly examining the data to identify relationships and build a model. They build models by trial and error. The network guesses a value that it compares to the actual number. If the guess is wrong, the model is adjusted. This process involves three iterative steps: predict, compare, and adjust. Neural networks are commonly used in a DSS to classify data and, as noted, to make predictions. Figure 10.2 shows that various inputs (from I_1 to I_k) are transformed by a network of simple processors. The processors combine and weight the inputs and produce an output value.

Data Mining Process

Data mining and knowledge discovery attempt to identify predictive relationships and provide managers with descriptive information about the subject of a database. There are a number of prescribed data mining processes. To make the best use of data mining, you must first make a clear statement of your objectives. Researchers at IBM have described data mining as a three-phase process of data preparation, mining operations, and presentation. Analysts at the Gartner Group describes it as a five-stage process:

1. Select and prepare the data to be mined.

2. Qualify the data via cluster and feature analysis.

3. Select one or more data mining tools.

4. Apply the data mining tool.

5. Apply the knowledge discovered to the company's specific line of business to achieve a business goal (Gerber, 1996).

These processes are similar. The first step is to select and prepare the data to be mined. Some data mining software packages include data preparation tools that can handle at least some of the preparation that needs to be done to the data. The second step is qualifying the data using cluster and feature analysis software. This step takes some business knowledge about the question that one is trying to answer. This is the step where bias in the data should be detected and removed (IBM, 1998). In the third step an appropriate data mining tool is selected and used. Finally, the results are presented to decision-makers and if they are considered useful they should be used to help achieve business goals.

Data Mining Examples

Let’s briefly review some examples of data mining applications. Some applications include: predicting which customers are likely to buy which products and when; improving credit/loan/mortgage risk analysis; identifying new untapped market segments that might be profitable; predicting which securities to buy/sell when; improving customer service, support, satisfaction and loyalty; understanding which factors affect profit and productivity; and detecting fraud earlier to avoid losses.

One example is identifying characteristics of users of ATM cards at points of sale. Some people never use their ATM cards at points of sale, others use their cards only a couple of times per month, and some use their cards quite frequently. Frequent users generate the most revenue for the financial institution that issues the card. Genetic data mining was used to evolve prediction models for several levels of card usage, based on parameters such as customer age, average checking account balance, and average number of checks written per month. Using these models of frequent users, the financial institution is able to target people matching the frequent-user profile for promotional campaigns (cf., http://www.ultragem.com/sample.htm).

Firstar Bank used data mining to determine which customers are likely to be interested in a new service. Data mining allowed Firstar to do target mailings saving the company time and money compared to broad mailings to all customers. As a result of the targeted mailing the response rate to the mailings increased by a factor of four (Freeman, 1997).

Siemens uses a DSS built using case-based reasoning to aid technical customer support services staff. The program uses the results of previous customer inquires to help quickly answer the questions from current inquires.

As the result of a data mining project done at ShopKo, managers discovered that the sale of film does not cause the sale of a camera, however the sale of a camera generally causes the sale of film. Data mining may find relationships that managers already knew existed. One hopes new knowledge and relationships are also discovered.

American Century used data mining to find information to help them cross sell financial products to existing customers. Developers shared a number of lessons they learned from this project. One lesson is that senior executive support, as well as IT support is necessary for success. Another lesson is that business issues must drive project development. If the project will not benefit the company, resources should not be allocated to it. They found that data mining often yields specific results rather than general rules. The quality of the data had a direct effect on the usefulness of the results. Finally, they found that data mining requires statistical skills, business skills, and analytical skills in order for the company to get the most benefit from the tools.

Evaluating Development Packages

The following five criteria should be carefully considered when evaluating vendor software for either mining data or building Knowledge-Driven DSS.

Cost. With the significant costs of technology and the rapid advancement of new technologies, companies want affordable packages. A development environment with multiple tools is often better than purchasing a more specialized development package. MIS staff want to learn software that can be applied to a wide variety of problems

Scalability. Companies need development software that will easily integrate with existing software applications and hardware platforms. Many Knowledge-Driven DSS need to be distributed to users so Web technologies are often appropriate. Some observers want more managers and analysts to have data mining tools so a distributed, scalable solution is also an issue in statistical analysis and knowledge discovery.

Security. With the increase in shared data, there is an increasing concern regarding the security of DSS knowledge and large databases. Both rule bases and behavioral data that will be mined need to be protected. Security is easily overlooked in developing knowledge applications.

Development features. Knowledge-Driven DSS are not usually standard "off-the-shelf" packages. It is important that packages allow for easy development of customized capabilities, rule input and maintenance. If uncertainties, frames or other capabilities are part of the development environment, then the package needs to help ensure that features and capabilities are used appropriately.

Ease of installation and use. Managers and MIS staff want software packages that are easy to install and require minimal training. This criterion is especially important with end-user data mining tools.

Conclusions and Commentary

Knowledge-Driven DSS and mining data are at the decision support frontier in organizations. During the 1980's, unrealistic expectations were created for expert systems and the recent hyperbole about data mining has also created some skepticism. Managers and IS staff need to investigate how these technologies might solve real business problems, but caution should be used in selling Knowledge-Driven DSS and data mining projects.

Data mining techniques and tools are **NOT** fundamentally different from the older quantitative model-building techniques. The methods used in data mining are extensions and generalizations of analytical methods known for decades. Neural networks are a special case of what is called projection pursuit regression, a method developed in the 1940s. Classification and regression tree (CART) methods were used by social scientists in the 1960s (cf., http://www.twocrows.com/iwk9701.htm). The computing technology used to implement these underlying methods has however greatly improved.

For the foreseeable future, modest Knowledge-Driven DSS projects can provide some benefits and can help companies develop experience using expert system and data mining tools. It is important for large companies to have projects in this category of Decision Support Systems, but only modest resources should be committed in many companies. The list of possible applications in this chapter should guide the selection of new projects.

Audit Questions

1. Has your company implemented any Knowledge-Driven DSS or business expert systems?
2. Have you used a data mining tool for analysis of business data?
3. Does the IS/IT group have any expert system or data mining development projects underway?

Questions for Review

1. What is a rule?
2. What is the "telephone test"?
3. What are 5 characteristics Knowledge-Driven DSS?
4. How should we develop a Knowledge-Driven DSS?
5. What are the two ways an inference engine can manipulate results?
6. What are the general steps in data mining?
7. What are the common categories of data mining tools?
8. What are the major components of a Knowledge-Driven DSS?

Questions for Further Thought

1. Why do managers need the support provided by Knowledge-Driven DSS?
2. How does a Knowledge-Driven DSS differ from data mining?
3. What benefits are there for deploying a Knowledge-Driven DSS on the World-Wide Web?

Internet Exercises

1. Find an example of a Knowledge-Driven DSS at a Web site. Most applications can be deployed on the Internet. The sites listed below have some KDSS examples that can be viewed on the Web.

   http://restrictedstock.com RestrictedStock.com -- Developed and maintained by MultiLogic, this site demonstrates the use of a graphical interface available with MultiLogic software.

   http://www.dol.gov:/elaws Elaws -- The Department of Labor has developed an interactive compliance assistance tool using MultiLogic's software. Elaws is comprised of individual advisors designed to help users understand their rights as employees and employers.

   http://www.osha.gov/Wren/csa.html OSHA Confined Spaces Advisor. -- OSHA used MultiLogic's software to create their Confined Space Advisor. This expert system provides guidance to help employers protect workers from the hazards of entry into permit-required confined areas.

2. Software developers use expert system shells to create expert systems and Knowledge-Driven DSS. Most shells are commercial software packages, but some shells are available as shareware or freeware. Find an expert system shell at a Web site.

Case Study - For Underwriting, NC Blue Turns to an Expert

Joint effort with PLATINUM technology to increase accuracy and productivity

By Christy Tauhert April 2, 1998

Expert systems are quite a draw for health insurers that want to automate and improve the accuracy and efficiency of their underwriting process, especially when mistakes affect the bottom line. This was a concern at Blue Cross Blue Shield of North Carolina (BCBSNC, Durham, NC, nearly $1 billion in assets), which embarked on an underwriting expert systems project for medical applications with PLATINUM technology (Oakbrook Terrace, IL) that not only increased the accuracy of underwriting decisions, but cut the underwriting process in its small group health unit from five days to just minutes.

"We saw the need to get more consistency in our underwriting decisions and increase productivity in the underwriting process," says John Friesen, vice president of BCBSNC's actuarial and underwriting services unit. The NC Blue, which offers managed care, traditional indemnity, group and individual health insurance, had such concerns in mind when it came across PLATINUM in mid-1995. PLATINUM's programming expertise, combined with BCBSNC's underwriting expertise, proved to be a good fit for a joint project to develop an automated underwriting system, says Friesen.

PLATINUM consultants worked with the insurer's underwriting and IT employees to co-develop an automated underwriting expert system framework using PLATINUM's Aion application development environment. Setting up the core system involved gathering and creating rules for the system. Business rules were created according to 157 ICD-9 codes (International Classification of Diseases medical condition codes). The automated system asks for the patient's conditions, medications and treatments. When it has enough information, the system makes a decision accordingly, says Friesen.

Once rules had been created in the Aion rule base, a Powerbuilder graphical user interface was built to collect enrollment data on Windows 95-based client PCs (used by 10 to 20 underwriters) and pass it to Aion. They then process the data and feed the results to an IMS database server that is supported by an MVS-based IBM mainframe. Sybase NetGateway, a database interface, integrates the rules base in Aion with the IMS database, client PCs, a Unix server, the mainframe and BCBSNC's proprietary rate quoting, risk management and policy writing systems. A second database, Oracle (running a Unix server), receives enrollment data for management reporting.

For the pilot, BCBSNC used a few medical conditions to make sure the underwriting decisions were being made accurately. A complete implementation process began in April 1997 to create the rest of the rules, and the system went live in September 1997.

The solution has saved the NC Blue money in terms of the type of employees it needs to facilitate the underwriting process, which now only requires analysts or enrollment entry people to enter key information into the system. As a result, BCBSNC was able to eliminate two full-time employees and change responsibilities for three others.

The solution also saves time. "Before, the process took anywhere from one to five days to complete. Now, it takes just as much time as it does to enter information into the computer — it's a matter of minutes," notes Friesen.

BCBSNC's goal was to use the system for 60 to 70 percent of its applications, although the number has grown to 85 percent. Friesen says he's hesitant to take that number higher, because "there are complex medical conditions I wouldn't want the machine to handle."

While Friesen would not discuss costs, return on investment will take less than a year, because accuracy of underwriting decisions has been improved by well more than one percent, he says.

Questions for Discussion:

1. Is this underwriting expert system a Knowledge-Driven DSS? Why or why not?
2. What are the benefits of this expert system?
3. Why is the expert system not used to make all underwriting decisions?
4. What is the architecture for the system? Draw a flow chart.

References

Allen, Bradley P. Case-based reasoning: business applications. Communication of the ACM, March 1994 v37 n3 p.40(3)

Byte Magazine, Data Mining State of The Art, October 1995, at URL http://www.byte.com/art/9510/sec8/sec8.htm.

Berry, Michael J. A. and Gordon Linoff. *Data Mining Techniques for Marketing, Sales, and Customer Support*. New York: Wiley Computer Publishing, 1997.

Darling, Charles B. Data Mining for the Masses. *Datamation*. February 1997. (URL http://www.datamation.com/PlugIn/issues/1997/feb/02mine.html)

Data Intelligence Group. An Overview of Data Mining at Dun and Bradstreet. September 1995. (URL http://www3.shore.net/~kht/text/wp9501/wp9501.htm)

Dhar, V. and R. Stein. *Intelligent Decision Support Methods: The Science of Knowledge Work.* Upper Saddle River, NJ: Prentice Hall, 1997.

Edelstein, H. Data Mining -- Let's Get Practical. *DB2 Magazine*, Summer, 1998 http://www.db2mag.com/98smEdel.htm.

Eom, S. A Survey of Operational Expert Systems in Business (1980-1993). *Interfaces*, vol.26, no. 5, 1996.

Ferrier, K. A Characterization of Data Mining Technologies and Processes, *Journal of Data Warehousing*, December 1997.

Freeman, Eva. Data Mining Unearths Dollars from Data. *Datamation*, July 1997. (URL http://www.datamation.com/PlugIn/issues/1997/july/07mine.html)

Gerber, Cheryl. Excavate Your Data. *Datamation*, May 1, 1996. (URL http://www.datamation.com/PlugIn/issues/1996/may1/05asoft3.html)

Gill, T. G. Early Expert Systems: Where Are They Now? *MIS Quarterly*, vol. 19, no. 1, 1995.

Greenfield, Larry. Data Mining. LGI Systems, Inc. January 12, 2000. (URL http://www.dwinfocenter.org/datamine.html)

Guimaraes. T., Y. Yoon, and A. Clevenson. Factors Important to Expert Systems Success: A Field Test. *Information and Management*, vol. 30, no. 3, June 1996.

Holsapple, C.W. and A. B. Whinston. *Decision Support Systems: A Knowledge-based Approach*, Minneapolis, MN: West Publishing Co., 1996.

IBM. Data Mining: Extending the Information Warehouse Framework. (URL http://www.almaden.ibm.com/cs/quest/papers/whitepaper.html)

Information Discovery, Inc. Perspective on Data Mining: Reaping Benefits from Your Data. (URL http://www.datamining.com/datamine/dm-ka.htm)

Kay, Emily. The democratization of data mining. *Datamation*. June 1998. (URL http://www.datamation.com/PlugIn/issues/1998/june/06mine.html)

Kantrowitz, Mark (editor). FAQ: Expert System Shells 1/1, 1997 (see URL: http://www.faqs.org/faqs/ai-faq/expert/part1/).

Lee, Bill. Data Mining: Understanding the Basics. (see URL http://fiat.gslis.utexas.edu/~palmquis/courses/project/dm_basic.htm)

Saarenvirta, G. Mining Customer Data. DB2 Magazine. 1998. (URL http://www.db2mag.com/98fsaar.html).

Sarle, W.S., (editor), Neural Network FAQ, part 1 of 7: Introduction, 1999, (URL: ftp://ftp.sas.com/pub/neural/FAQ.html).

Small, R. D. Debunking Data Mining Myths: Don't let contradictory claims about data mining keep you from improving your business. *Information Week*, January 20, 1997, CMP Media, Inc. http://www.twocrows.com/iwk9701.htm.

Thearling, Kurt. Data Mining, Decision Support and Database Marketing. (URL http://www3.shore.net/~kht/index.htm)

Thearling, Kurt. Data Mining and Advanced DSS Technology, an on-line Data Mining Tutorial. (URL http://www3.shore.net/~kht/dmintro/dmintro.htm).

Turban, E. *Decision Support and Expert Systems: Management Support Systems*. (Fourth Edition) Englewood Cliffs, NJ: Prentice Hall, Inc, 1995.

Turban, E. and J. Aronson. *Decision Support Systems and Intelligent Systems.* (Fifth Edition) Englewood Cliffs, NJ: Prentice-Hall, Inc, 1998.

Yoon, Y., T. Guimaraes, and Q. O'Neal. Exploring the Factors Associated with Expert Systems Success. *MIS Quarterly*, vol. 19, no. 1, March 1995.

Web Links

http://www.aaai.org/ American Association for Artificial Intelligence Web Site.

AI – Case-Based Reasoning, http://www.ai-cbr.org

A Comparison of Leading Data Mining Tools. A presentation by John F. Elder IV and Dean W. Abbott at the Fourth International Conference on Knowledge Discovery and Data Mining. (URL http://dataminninglab.com/) (PDF format).

The Data Mine http://www.cs.bham.ac.uk/~anp/TheDataMine.html.

fuzzyTECH, http://www.fuzzytech.com/

Information Discovery, Inc. is the leading provider of large scale data mining and knowledge discovery oriented decision support software and solutions. http://www.datamining.com/

Knowledge Discovery Nuggets (aka KDD) http://www.kdnuggets.com.

National Center for Data Mining, University of Illinois at Chicago, at URL http://www.ncdm.uic.edu/.

SPSS Inc., http://www.spss.com

Visual Techniques for Exploring Databases. Professor Daniel Keim, University of Halle-Wittenberg. (URL http://www.informatik.uni-halle.de/~keim/) (PDF format).

http://www.zaptron.com check the Fuzzy Logic application called "Intelligent Data Mining Software Suite". It uses neural network, fuzzy logic, non-linear programming, and time series analysis methodologies.