T-SMC/18/6//23128

A Prototype DSS for Structuring and Diagnosing Managerial Problems

Nassar H. Ata Mohammed
James F. Courtney, Jr.
David B. Paradice

Reprinted from
IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS
Vol. 18, No. 6, 1988

A Prototype DSS for Structuring and Diagnosing Managerial Problems

NASSAR H. ATA MOHAMMED, JAMES F. COURTNEY, JR., AND DAVID B. PARADICE

Abstract —A methodology for managerial problem recognition, structuring, and diagnosis is presented. The system integrates diagnostic concepts from artificial intelligence with database management and structural modeling techniques. Given an organizational database, a user specifies control bounds for selected data items and causal relationships between variables in the database. Structural modeling techniques are used to represent knowledge of causal relations. If a control variable's value is out of bounds, structural models are used in an attempt to diagnose (explain) the discrepancy. If the discrepancy is explained, a diagnosis is at hand. If not, the user's model is deficient somehow and "what if" analysis may be used to construct an improved model. This methodology has been implemented in a prototype decision support system (DSS) that is described, along with an example of its use.

I. Introduction

ALMOST all existing decision-support technology assumes that the decision problem has already been recognized, diagnosed and structured. Once modeled, mathematical techniques and data from a data base may then be used to analyze the problem. Recently, researchers have recognized the need to provide support for earlier stages of the decisionmaking process. For example, work on model management systems has the objective of supporting problem structuring and the dynamic construction of models for problem analysis [1].

This paper presents the basic concepts underlying a system that supports even earlier, and perhaps more important, steps in the decisionmaking process—problem finding and problem diagnosis. The focus of this paper is on problem diagnosis because this is a critical, but neglected, aspect of the decisionmaking process. In a survey of decision support system (DSS) users, for example, Meador, Guyote and Keen [2] found that needs assessment and problem diagnosis were rated as the most important factors in DSS development. The same survey revealed these factors were among the lowest rated in terms of system performance. As Mintzberg *et al.* [3, p. 274] put it,

> Diagnosis is probably the single most important routine (i.e., process), since it determines in large part, however implicitly, the subsequent course of action. Yet researchers have paid almost no attention to diagnosis, preferring instead to concentrate on the selection routines...

Problem finding is based on Pounds [4] notion of deviation from standard or expectations. Diagnosis refers to a search for cause–effect relationships between variables in the problem domain [5]. A "problem" occurs when a variable's value is outside a predetermined acceptable range. This definition is employed in the widely used technique of variance analysis in accounting systems, so the approach is applicable to a multitude of management problems in a wide variety of organizations. Expert system techniques are combined with structural modeling concepts and mathematical modeling in a problem processor to support problem structuring and diagnosis. The diagnostic search is conducted upon a knowledge base consisting of user assertions about cause–effect relationships between pairs of variables in the problem domain.

Briefly, problem diagnosis consists of searching the knowledge base for variables asserted to have an impact on the variable violating its bounds. The problem processor uses the information on variables impacting the problematic variable to dynamically construct "causation trees" [6], the branches of which represent various paths to the problem's potential causes. Mathematical procedures are applied to data in the data base to test the validity of these hypotheses.

If the validity of an hypothesis is confirmed, then the problem's causes have apparently been explained. If not, the user's mental model (captured in the knowledge base of variable relationships) is apparently deficient, and a more extensive search for potential problem causes may be directed by the user. This search may involve modifying the current model and applying the revised model to the problem. When appropriate, new assertions may be added to the knowledge base. In this manner, the user and the system learn about the problem domain together as assertions are continually proposed, tested, and refined [7]. The remainder of the paper describes recent research related to problem formulation and presents the conceptual foundations of a prototype system that supports problem formulation processes. A scenario is presented to illustrate the approach.

II. The Problem and Related Work

Simon [8] described managerial decision making as a three-phased process consisting of intelligence, design, and choice activity. Intelligence is used in the military sense to include searching the environment for problems (i.e., conditions requiring a decision); design involves developing

Manuscript received June 1, 1987; revised September 19, 1987 and June 25, 1988.
N. H. A. Mohammed is with King Fahd University of Petroleum, Dahran, Saudia Arabia.
J. F. Courtney and D. B. Paradice are with the Business Analysis and Research Department, College of Business Administration, Texas A&M University, College Station, TX 77843-4217.
IEEE Log Number 8823128.

alternative courses of action; choice involves evaluating, selecting, and implementing one alternative.

Mintzberg, Raisinghani, and Theoret [3] analyzed 25 strategic decision situations in the field and slightly redefined Simon's three original phases as identification, development, and selection, respectively. They also identified seven "decision routines" occurring within the major phases. Some routines may or may not be invoked, depending upon circumstances surrounding the decision process. Routines in the identification phase include problem recognition and problem diagnosis. The development phase may consist of a search routine in which the decisionmaker searches for "canned" solutions to a problem and/or a design routine in which customized solutions are developed. Finally, the selection phase consists of screening out irrelevant solutions, choosing a solution to implement, and getting authorization to implement the chosen alternative. These routines are summarized in Table I, along with three groups of "supporting routines" that may be invoked at essentially any point in the process.

Most work in management science and DDS has concentrated on the selection phase of the decisionmaking process, focusing almost exclusively on analytical techniques for execution within the evaluation–choice routine [3, p. 274]. In other words, existing techniques tend to provide adequate support for problem analysis, but provide very limited support for problem finding and model formulation [9]. This situation is summarized in Table II.

The fascination with problem analysis is not only unfortunate, but somewhat surprising. Lasswell [10], [11] noted some decades ago that problem-oriented inquiry frequently requires not new quantifications of existing models, but new models of processes. In his development of "policy science," Lasswell emphasized the construction of models capable of combining quantitative and qualitative observations, models whose elaboration is sufficient to enable investigators to deal with complex situations [10, p. 9]. Management scientists have recognized for quite some time that inadequate problem formulation may result in "correct" solutions, but to incorrectly formulated problems [12, p. 264]. This phenomenon is now known as an "error of the third kind," or a "Type III error" [12], [13]. Certainly, solving the wrong problem may lead to the selection of inappropriate "solutions" with disastrous consequences for the organization. Recent research in DSS has been directed toward the support of design activity.

TABLE I
MANAGERIAL DECISION ROUTINES [3]

I. IDENTIFICATION PHASE

A. Decision Recognition Routine
Realization that an opportunity, problem, or crisis exists, and the need to formulate a response.

B. Diagnosis Routine
Preliminary analysis of the opportunity, problem, or crisis, including attempts to clarify and define issues and cause–effect relationships.

II. DEVELOPMENT PHASE

A. Search Routine
Search for ready-made solutions.

B. Design Routine
Modify a ready-made solution or custom-design a solution.

III. SELECTION PHASE

A. Screen Routine
Eliminate irrelevant solutions.

B. Evaluation-Choice Routine
Evaluate alternative solutions and choose one.

C. Authorization Routine
Invoke procedures to authorize implementation of the selected solution.

IV. SUPPORTING ROUTINES

A. Decision Control Routine
Management of the overall decisionmaking process.

B. Decision Communication Routine
Communication among individuals and groups involved in the decisionmaking process.

C. Political Routine
Bargaining among individuals and groups affected by the decision.

To attack such problems, Will [42] proposed the development of model management systems (MMS) that use artificial intelligence (AI) techniques to facilitate problem structuring and the use of management science models [1], [14], [15], [43]. Specifically, structure inheritance networks (SINs) are used to represent knowledge about organizational models and their elements and to dynamically construct models in response to problem situations.

Bonczek, Holsapple, and Whinston [16]–[20] have published a series of works that integrate data base management and AI techniques in a DSS context. Konsynski and Dolk [21] use knowledge abstractions and frames rather than SINs to represent modeling knowledge. Minch and Burns [22] provide tools for linking and documenting model building blocks, and provide facilities for designing user interfaces. Wang and Courtney [23], [24] integrate data base management and modeling via analyst-developed "macros" that may perform both data retrieval and modeling in one composite query. Blanning [25], [26] em-

TABLE II
SUPPORT FOR DECISIONMAKING STEPS

Decisionmaking Step	Automated Tool	Status
I. Identification		
A. Problem recognition	Exception reporting	Well-developed
B. Problem diagnosis		
II. Development		
C. Search		
D. Design	DSS-model management	Recent-concept
III. Selection		
E. Screen	DSS and MS/OR	Well-developed
F. Evaluation-choice	DSS and MS/OR	Well-developed
G. Authorization		

phasizes the need to represent cause–effect relationships in business expert systems and has investigated an entity-relationship approach to model management.

Although Ackoff [27], [28] describes a system with a diagnostic function in general terms, the only major empirical work addressing *managerial* problem diagnosis is that of Bouwman [6]. Managerial problem diagnosis differs from problem diagnosis in other domains because the managerial problem domain is not as structured as many other problem domains. Unlike problem domains such as engineering, mathematics, or electrical circuit design, the managerial problem domain is not governed by well formulated relationships. Although specific areas of management may be structured, for example, certain areas of the tax code, general management problems are acutely ill-structured [29], [30]. (The AI literature contains many examples of problem diagnosis in other domains, as reviewed in [31]). Bouwman used protocol analysis to study how human subjects diagnose managerial problem situations by analyzing financial statements. Bouwman's system is qualitative in nature and quite complex. It provides an excellent bassis for the development of a system to support a quantitative approach directed at automatically detecting and diagnosing management problems.

The next section presents a general conceptual model of the process of managerial problem recognition and diagnosis. After that, a system supporting this process is described. The system extends Bouwman's approach by including a data base, knowledge base and model base to provide a quantitative approach to problem diagnosis. The inclusion of these components also permits statistical validation of user models, which assists in minimizing the effect of user biases and incorrectly formulated models [32].

III. The Conceptual Basis for Problem Recognition, Structuring, and Diagnosis

As illustrated in Table II, existing systems provide support for limited aspects of the decisionmaking process. Virtually no tools are available to support automatic problem diagnosis. In addition, the concept of supporting the design process (problem structuring and model formulation) is recent, and no well-developed design tools yet exist.

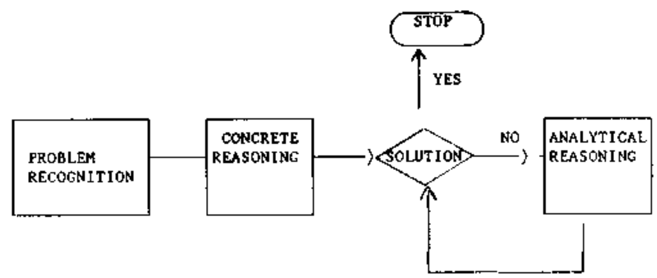


Fig. 1. Decisionmaking model underlying system.

The system described in this paper supports the first two decisionmaking phases: intelligence and design. The architecture of the system is based on a problem-solving model that is synthesized from the works of Pounds [4], Piaget [33], and Sage [34]. This model is illustrated in Fig. 1.

The findings of Pounds are used to specify the approach to problem recognition. If the problem is recognized as one already familiar to the decisionmaker, then "concrete operational thought" based primarily on manipulating symbols and organizing facts may be used to solve the problem. In this case, the decisionmaker has encountered the problem before and knows its solution. This mode of thought is preferred by most people because it is less stressful and does not strain cognitive processing capabilities.

If the problem is unfamiliar to the decisionmaker, then "formal analytical thought" is required for problem analysis. Formal analytical thought requires the capacity to reason hypothetically and to consider the effect of different variables or possibilities outside of personal experience [34, p. 656]. This process is invoked only when concrete operational thought fails.

The full architecture of the system supports both formal and concrete thought processes by including a module for storing the results ("experiences") of formal diagnostic analyses in an experiential knowledge base. The experiential knowledge base can subsequently simulate concrete reasoning (Has this problem been encountered before? If so, what was the diagnosis?). A discussion of the experiential knowledge base is given elsewhere [31].

The basic architecture of the system is illustrated in Fig. 2, and includes the following elements: 1) a user interface; 2) a monitor that looks for problems; 3) a problem processor that searches for a problem's causes; 4) a knowledge manager that maintains user assertions about cause-effect

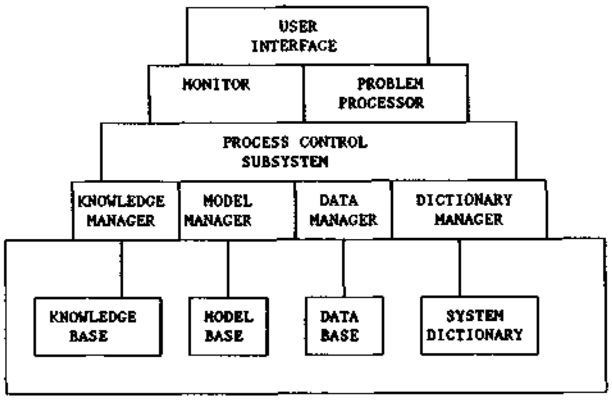


Fig. 2. Basic system architecture.

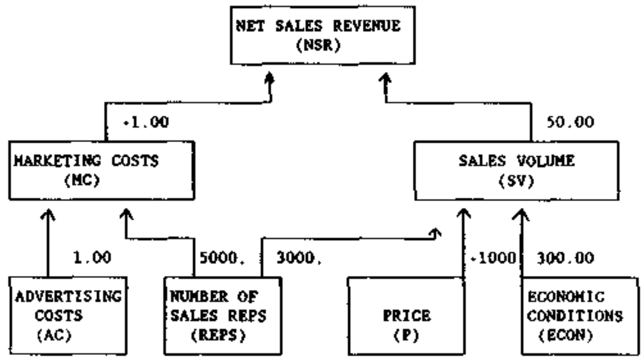


Fig. 3. Structural model for marketing data in table III.

TABLE III
KNOWLEDGE BASE FOR VARIABLES IN A SIMPLE MARKETING PROBLEM

Variables (Data Item)	Impacts Upon	Direction (+/−) and Magnitude
MCs	NSR	−1.00
NSR	MC	+5000.00
NSR	SV	+3000.00
AC	MC	+1.00
Price	SV	−1000.00
ECON	SV	+300.00
SV	NSR	+50.00

relationships between pairs of variables in the problem space; 5) a model manager that maintains a model base of statistical procedures; 6) a data manager that supplies data to the rest of the system as needed; 7) a dictionary manager that maintains information on the location, format, and other information about variables and data items in the system; 8) a process control subsystem that links various elements of the system together and controls the flow of information. This paper is primarily concerned with the monitor, the problem processor, the knowledge base, and the model base, which are described in some detail. Other elements of the system are described briefly.

A. Support for Intelligence Activity: Problem Finding

Support for intelligence activity consists of problem finding and is based on Pounds [4] conclusion that managers define a problem as an unexpected deviation from some mental standard. As Mintzberg *et al.* [3, p. 253] describe it,

> The need for a decision is identified as a difference between information on some actual situation and some expected standard. In a study of these differences, Pounds [4] found that these standards were based on past trends, projected trends, standards in some comparable organization, the expectations of other people, and theoretical models.

Intelligence activity is performed by the monitor that solicits information from the user on variables whose values are to be watched. Monitored variables may include traditional performance variables (such as sales volume, profit, or market share), "critical success factors" (such as return on investment or inventory turnover ratio), and factors from the organization's environment (such as prime interest rate or competitor's advertising expenditures). Values for monitored variables are maintained by the data manager along with values for other variables throughout the system.

In order to perform the monitoring function, the system must have information on standards for each monitored variable and how these standards are to be applied. Standards can be based on constant upper and lower bounds on the variable, on percent deviation from the value for a base period, on deviation from a trend line, or other similar measures.

The system uses this information on monitored variables and their standards in its quest to locate problem situations. Either concurrent with data base modification, or on demand by the user, the monitor applies the procedures for computing standards for each monitored variable. It then compares the current value for each variable to its computed standards, and whenever these standards are violated, the monitor places the corresponding variable on a list of "problem symptoms." At the end of the monitor's processing the list of problem symptoms is displayed to the user and passed to the problem processor for diagnosis.

Since organizational systems are highly complex and interrelated, one underlying cause may trigger several symptoms. The problem processor must be intelligent enough to account for this phenomenon, and avoid excessive searches when one or a few causes have led to the discovery of several problem symptoms. The manner in which the problem processor efficiently handles the list of problem symptoms is described next.

Support for problem structuring and diagnosis is based on a synthesis of ideas from expert systems, structural modeling, and statistical analysis. Pracht [9] has developed interactive graphics software based on concepts of structural modeling that is helpful in problem structuring [35]. Structural modeling is based on the theory of directed graphs and represents relationships between pairs of variables either in graphical (Fig. 3), tabular (Table III) or matrix form (by forming a row and column for each variable and inserting the coefficient for the impact of row varible i on column variable j in cell i, j). The graphical form is useful for displaying the structure of a problem,

TABLE IV
OBSERVED DATA VALUES

Variable (Abbreviation)	Current Period	Last Period	Change
NSR	1355000	1500000	−145000
MC	95000	90000	+5000
AC	50000	40000	+10000
REPS	9	10	−1
Price	50.00	50.00	0
ECON	105	104	1
SV	27300	30000	−2700

Monitor Report

List of Problem Symptoms

Item	Observed Value	Comparison Value	Variance
----	-------------	---------------	--------
Sales Volume	27,300	30,000	-9.00%
Net Sales Income	1,355,000	1,500,000	-9.67%

Fig. 4. Screen display showing example monitor report.

and the matrix form is useful in mathematical analysis of structural models. (See Lendaris [44] for a thorough discussion.)

The software solicits user assertions concerning cause–effect relationships between pairs of variables in the problem space, and stores these assertions in a knowledge base using a semantic network type of representation. Assertions are of the form: "A one unit change in variable i causes variable j to change by Cij. The Cij's will be referred to as "impact coefficients."

By storing this underlying causal structure, the system contains a foundation for the development of explanatory or advisory features. Such features would be able to provide advice to a user regarding how to attain specific goals, such as a specific level of sales. Clearly, advisory action must be more than a mere description of the problem structure, it must incorporate ultimate goals [36], [37]. Extensions in this regard are being pursued [38].

These binary assertions may be interpreted as the user's "mental model" or "cognitive map" of the problem space. Pracht's system combines binary assertions and displays them graphically in the form of structural models or directed graphs. Fig. 3 shows the structural model derived from the assertions in Table III. Structural models are also useful in depicting the process of problem diagnosis, as illustrated next. This structural model is also the causation tree for net sales revenue.

IV. The Prototype System and An Example

In the prototype system, monitoring and generation of the list of problem symptoms is based simply on changes in observed values from one period to the next. However, the problem structuring and diagnosis modules have been implemented in full. The diagnostic process is under user control via menus, rather than being strictly phase-wise. The structural model in Fig. 3 and the corresponding data in Tables III and IV will be used to illustrate the diagnostic process. A full-screen editor was used to develop screens for entering data as in Table III.

The data in Table IV gives the values used for each variable in periods P and $P-1$. The current value of net sales revenue and sales volume are outside of their predetermined bounds (not given in the table), so they appear on the monitor report in Fig. 4.

Since these data items have exceeded their bounds, they constitute the set of problem symptoms. Since there are significant deviations, the system displays the menu for interactive diagnosis shown in the Appendix, Section III.

Suppose the user chooses option C to display the causation tree. The system then asks the user to identify the variable for which the causation tree is to be displayed. If the user indicates net sales revenue (NSR), then the diagram in Fig. 3 is displayed.

If option X ("examine" a node) is chosen and the user again indicates net sales revenue, then the screen in the Appendix, Section IV is generated. This screen is based on the user model in the knowledge base and data values retrieved from the database. It corresponds to a breadth-first analysis and examines the direct descendants (children) of the selected node. Specifically the computed values are: NSR $= -1.0$ (marketing costs (MC)) $+ 50.0$ (sales volume (SV)); NSR $= -1.0$ $(5000) + 50$ (-2700); and NSR $= -140000$. Thus the model predicts a drop of \$140000 in net sales revenue that compares favorably to an actual drop of \$145000 (a 3.4-percent error).

The H option (Appendix, Section III) generates hypotheses about the problem's causes by searching paths through the hierarchy from the selected node to terminal nodes. This is a depth-first analysis. Again assuming net sales revenue is chosen, information on the five paths from net sales revenue to terminal nodes is displayed as shown in the Appendix, Section III. This information is sorted on the basis of each path's contribution to explaining the problem. In this case net sales revenue has declined, so the paths with the most negative values are displayed first. Paths with positive contributions have actually offset the problem.

In the example, it is estimated that \$150000 in revenue has been lost because a sales representative was lost and each representative sells 3000 units on average. This is the path from NSR to SV to sales representatives (REPS). This path is the highest ranked hypothesis. The path from net sales revenue (NSR) to MCs to advertising costs (AC) contributes \$10000 to the decline because advertising costs increased \$10000. Since price was unchanged, it had no impact on sales volume or net sales revenue. The fact that the firm saved money (\$5000) because it didn't have to pay one sales representative is reflected in the path from NSR to MC to REPS. This path and the path from NSR to SV to economic conditions (ECON) have mitigated the problem somewhat. The economy increased slightly and raised sales volume slightly.

This information should clearly be sufficient to explain the base cause of the problem—that a sales representative was lost and the revenue ordinarily generated by that representative was also lost. It is then up to the manager to evaluate this diagnosis and take the appropriate action.

V. "What If" Scenarios

The model described in the previous example provided a fairly faithful prediction of the observed changes in net sales revenue. The elements (variables) and parameters (impact coefficients) in this model are presumed to be asserted by users in the knowledge acquisition process. Unfortunately, users are notoriously biased in their beliefs about the structures of problem domains [32]. Models may be in error in several ways: 1) relevant variables have been omitted; 2) irrelevant variables have been included; 3) links between variables have been omitted; 4) irrelevant links between variables have been included; 5) parameter values have been incorrectly specified; 6) or any combination of the above.

Suppose, for example, that economic conditions as measured by the economic indicator were unchanged, but all other observed values were the same. The model still predicts the same change in net sales revenue, because its computed value is based on market costs and sales volume that are unchanged for the previous example.

If the node for sales volume is examined, however, its computed value has changed. The old model precisely predicted the actual drop of 2700 units. The new predicts a drop of 3000 units because the sales representative was lost, but price and economic conditions were unchanged. To explain this discrepancy, suppose the user inspects the model in Fig. 3 and concludes that a link should exist from advertising costs to sales volume (else, why advertise?). It is estimated that an additional 30 units of product are sold for each $1000 spent on advertising. This would correspond to adding a link from AC to SV in the model with a coefficient of 0.03. It is a simple matter to use the full screen editor to add this link by adding a line to Table III of the form

AC SV 0.03

Now the model again predicts the precise drop of 2700 units, because the increase in advertising has offset the loss of the sales representative. Obviously, it would be rare to have models as precise as this in practice. Nevertheless, this simple example illustrates the manner in which the user may experiment with the model to improve it and to search for errors. This process should lead to a better understanding of the problem domain, more accurate models and ultimately better decisions.

VI. Conclusion

To summarize, the system described in this paper supports aspects of the decisionmaking process from problem finding to problem structuring and diagnosis. This process is summarized in Table V.

TABLE V
Problem Finding/Diagnosis Process

I) Specify variables to be monitored.
II) Specify monitoring procedures and limits.
III) Specify variables in problem domain and structural interrelationships.
IV) Maintain values for all variables in data base.
V) Apply monitoring procedures.
VI) If a monitored variable is out of bounds:
  A) Search knowledge base for variables impacting the offender;
  B) Construct causation trees;
  C) Analyze causation trees for explanatory power;
  D) Report results to user;
  E) Explain results, if requested to do so;
  F) If causation trees do not explain problem
    1) Employ "what if..." techniques to improve mode; and
    2) If tenable, store in knowledge base.

When triggered by the monitor, the problem processor initiates a search for relevant assertions from the knowledge base (via the process control subsystem) and dynamically clusters them into causation trees representing potential explanatory problem hypotheses. To test the validity of these hypotheses, the system constructs a mathematical representation of each branch and retrieves data from the data base to compute values based on the model. If the resulting values are within bounds, the user's mental model is validated, and the problem's causes have been located.

If the model fails to explain the problem, then the user's model is apparently deficient, and the system may be used to alter and retest the model. If a better model is found, the knowledge base is updated and used in future diagnoses. In this manner, the user and the system learn together about the underlying structure of the problem domain, as models are continually tested and refined.

Preliminary tests of the system with models developed by Pracht's human subjects operating in the business management laboratory/systems laboratory for information management (BML/SLIM) [41] environment reinforced the notion that this system (and indeed any system embodying a model) is only as good as the knowledge representing relationships in the problem domain. Where the system has accurate knowledge, it is very helpful in diagnosis. It is also useful in spotting areas where additional knowledge is needed by noting situations in which a diagnosis cannot be produced. This highlights the importance of the validation of "expert" knowledge in knowledge-based systems in general. Experts may be biased in their beliefs about relationships in the problem domain. These biases, if incorporated into an expert system, lead to incorrect diagnoses or the inability to diagnose at all. Thus the area of knowledge validation is a critical one for the success of knowledge-based systems in general, and one that DSS researchers should address very carefully. This is especially important when one considers that the system is not intended to be the sole support for diagnosis, but rather is to be augmented with the user's judgement.

In developing the prototype, several lessons were learned about possible implementation of a full-scale system. First, use of a screen generator greatly facilitated implementation of the user interface for entering relationships as in Table III. Thus, this aspect of the system was easier to build than other parts, which had to be developed from scratch. Second, when a few users outside the research project tried to use the system with built-in models, at least 2 or 3 h of training is necessary in getting across the concept of diagnosis. Learning to use the screens and menus was easy, but interpreting the results did not come so easily. Thus a tutorial would be a helpful addition. finally, interactive graphics would be very useful in making the displays of structural models more readable and would allow changes directly to the graphical form of the model, which many of Pracht's [35] subjects seemed to prefer.

The system is, of course, not without limitations. The current prototype is limited to acyclic graphs and linear, quantitative relationships among variables. Even with these restrictions, the approach is applicable to one ubiquitous class of problems facing managers, that being variance analysis in accounting systems (see [31]). However, strictly quantitative data is not a requirement of this methodology. Certainly, the binary change coefficients could be replaced with only a sign indicating the direction of the change (+,0, −), or with a qualitative description of the direction of the change (increasing, decreasing, or steady). The methodology could still construct the causation trees and provide an indication of the impact of a change in one variable on another. Regardless of the scope of the quantification, this process should lead to a better understanding of the problem domain, more accurate models, and ultimately better decisions.

Our future efforts will be directed to three activities. First, formal techniques of causal modeling or path analysis [40] will be incorporated into the system to test the validity of user-asserted relationships and to search for missing links in the model. These techniques are designed for use with real data and will be used to search for instances where an indirect path from a causal variable to an effect variable should be replaced with a direct link from cause to effect. Second, attempts will be made to incorporate other kinds of relationships in the knowledge base, such as "A is upper bound for B." Finally, efforts will be made to use these relationships to automatically generate models that will support choice activity, as well as intelligence and design.

Appendix
Example Terminal Session

I) Enter the appropriate letter to choose an option:
A—Apply monitoring and diagnostic procedures;
U—Update data dictionary or variable bounds;
C—Create data dictionary; and
E—Exit.

A
Monitor Report

List of Problem Symptoms

Item	Current Value	Last Period Value	Percent Deviation
NETINC	−581217.50	−536911.00	−8.25
SV	27300.00	30000.00	−9.00
NSR	1355000.00	1500000.00	−9.67

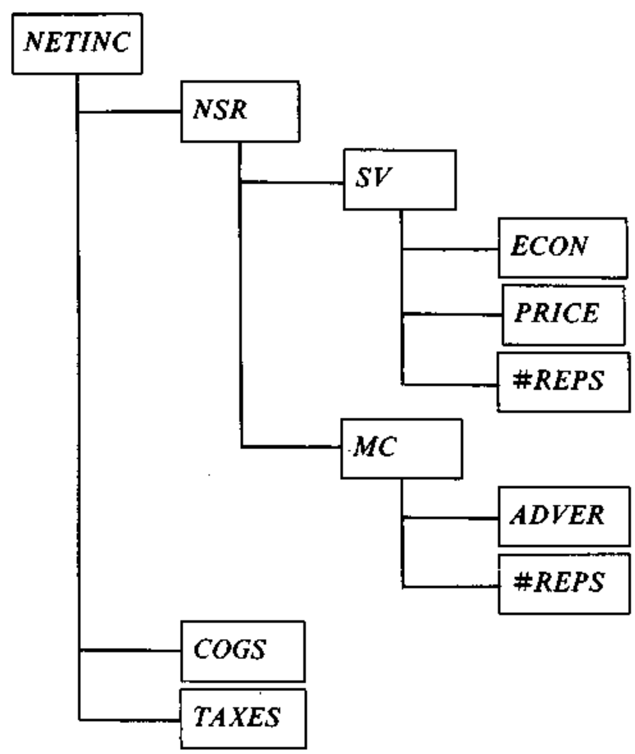
Number of significant findings = 3.

* * *END OF MONITOR REPORT* * *

II) Enter the appropriate letter to choose an option:
C—Causation tree;
H—Generate hypotheses;
X—Examine a node; and
E—Exit.

C
Causation tree for which node (variable)?
NETINC

KNOWLEDGE BASE TREE



III) Enter the appropriate letter to choose an option.
C—Causation tree;
H—Generate hypotheses;
X—Examine a node; and
E—Exit.

H

Generate hypotheses for which node (variable)?

NSR (Explanatory hypotheses (paths) for: NSR)

CONTRIBUTION OF PATHS

Rank	Hypothesis (path)			Contribution
1	NSR	← SV	← #REPS	−150000.00
2	NSR	← SV	← ADVER	−10000.00
3	NSR	← SV	← PRICE	0.00
4	NSR	← MC	← #REPS	5000.00
5	NSR	← SV	← ECON	15000.00

IV) Enter the appropriate letter to choose an option.

C—Causation tree;

H—Generate hypotheses;

X—Examine a node; and

E—Exit.

X

Which variable would you like to examine?

NSR

PARENT NODE: NSR

THIS PERIOD VALUE: 1355000.00

LAST PERIOD VALUE: 1500000.00

OBSERVED NET CHANGE: −145000.00

COMPUTED NET CHANGE: −140000.00

Child	This Period	Amount Changed	Coef	Coef * Change
MC	95000.00	5000.00	−1.00	−5000.00
SV	27300.00	−2700.00	50.00	−135000.00

References

[1] J. J. Elam, J. C. Henderson, and L. W. Miller, "Model management systems: An approach to decision support in complex organizations," First Int. Conf. Inform. Syst., Philadelphia, PA, 1980.

[2] C. L. Meador, M. J. Guyote, and P. G. W. Keen, "Setting priorities for DSS development," *MIS Quart.*, vol. 8, no. 2, pp. 117–129, June 1984.

[3] H. Mintzberg, D. Raisinghani, and A. Theoret, "The structure of 'unstructured' decision processes," *Administrative Sci. Quart.*, vol. 21, no. 6, pp. 246–275, June 1976.

[4] W. F. Pounds, "The process of problem finding," *Indust. Management Rev.*, vol. 11, no. 1, pp. 1–19, Fall 1969.

[5] K. Nakamura, S. Iwai, and T. Sawaragi, "Decision support using causation knowledge base," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-12, no. 6, pp. 765–777, Nov./Dec. 1982.

[6] M. J. Bouwman, "Human diagnostic reasoning by computer: An illustration from financial analysis," *Management Sci.*, vol. 29, no. 6, pp. 653–672, June 1983.

[7] K. Niwa, "A knowledge-based human–computer cooperative system for ill-structured management domains," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-16, no. 3, pp. 335–342, May/June 1986.

[8] H. A. Simon, *The New Science of Management Decisions*. New York: Harper and Row, 1960.

[9] W. E. Pracht, "A graphical interactive structural modeling aid for decision support systems," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-16, no. 2, pp. 265–270, Mar./Apr. 1986.

[10] D. Lerner and H. D. Lasswell, *The Policy Sciences*. Stanford, CA: Stanford Univ. Press, 1951.

[11] H. D. Lasswell, *A Pre-View of Policy Sciences*. New York: American Elsevier, 1971.

[12] H. Raiffa, *Decision Analysis*. Reading, MA: Addison Wesley, 1968.

[13] I. I. Mitroff and T. R. Featheringham, "On systemic problem solving and the error of the third kind," *Beh. Sci.*, vol. 19, no. 6, pp. 383–393, June 1974.

[14] J. J. Elam, "Model management systems: A framework for development," Tech. Rep. 79-02-04, The University of Pennsylvania, Philadelphia, PA, 1979.

[15] J. J. Elam and J. C. Henderson, "Knowledge engineering concepts for decision support system design and implementation," *Inform. and Management*, vol. 6, no. 2, pp. 109–114, June 1983.

[16] R. H. Bonczek, C. W. Holsapple, and A. B. Whinston, "Computer based support of organizational decision making," *Decision Sciences*, vol. 10, no. 2, pp. 268–291, Apr. 1979.

[17] ——, "The integration of network data base management and problem resolution," *Inform. Syst.*, vol. 4, no. 2, pp. 143–154, 1979.

[18] ——, "The evolving roles of models in decision support systems," *Decision Sci.*, vol. 11, no. 2, pp. 337–356, Apr. 1980.

[19] ——, "Future directions for developing decision support systems," *Decision Sci.*, vol. 11, no 4, pp. 616–631, Oct. 1980.

[20] ——, *Foundations of Decision Support Systems*. New York, NY: Academic Press, 1981.

[21] B. Konsynski, and D. Dolk, "Knowledge abstractions in model management," *DSS-82 Trans.*, San Francisco, CA, June 1982, pp. 187–202.

[22] R. P. Minch, and J. R. Burns, "Conceptual design for decision support systems utilizing management science models," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-13, no. 4, pp. 549–557, July/Aug. 1983.

[23] M. S. Wang and J. F. Courtney, "Design and implementation of the MAGIC/ROC decision support systems generator," *DSS-82 Trans., San Francisco, CA, June* 1982, *pp.* 37–48.

[24] M. S. Wang, and J. F. Courtney, "A conceptual architecture for generalized DSS software," *IEEE Trans. Syst. Man Cybern.*, vol. 24, no. 5, pp. 701–710, 1984.

[25] R. W. Blanning, "Management applications of expert systems," *Inform. and Management*, vol. 7, no. 6, pp. 311–316, Dec. 1984.

[26] R. W. Blanning, "An entity-relationship approach to model management," presented at the Model Management Workshop-University of Texas–Austin, May 16, 1985.

[27] R. L. Ackoff, "Beyond prediction and preparation," *J. Management Stud.*, vol. 20, no. 1, pp. 59–69, June 1983.

[28] R. L. Ackoff, *Creating the Corporate Future*. New York: John Wiley, 1981.

[29] A. Basden, "On the application of expert systems," *Int. J. Man–Machine Stud.*, vol. 19, no. 5, pp. 461–477, Nov. 1983.

[30] D. B. Paradice, "Combining databases and knowledge bases in managerial expert systems," *Artificial Intelligence--Implications for Computer Integrated Manufacturing*, A. Kusiak, Ed. Bedford, England: IFS Publications, Ltd., 1988, pp. 405–432.

[31] J. F. Courtney, D. B. Paradice, and N. Ata, "A knowledge-based DSS for managerial problem diagnosis," Special Issue of *Decision Sciences* on Expert Systems and Decision Support Systems, vol. 18, no. 3, pp. 373–399, Summer 1987.

[32] D. B. Paradice and J. F. Courtney, "Controlling bias in user assertions in expert decision support systems for problem formulation," *J. Management Inform. Syst.*, vol. 3, no. 1, pp. 52–64, Summer 1986.

[33] J. Piaget, *Memory and Intelligence*. New York: Basic Books, 1973.

[34] A. P. Sage, "Behavioral and organizational considerations in the design of information systems and processes for planning and decision support," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-11, no. 9, pp. 640–678, Sept. 1981.

[35] W. E. Pracht and J. F. Courtney, "The effects of an interactive graphics-based DSS to support problem structuring," *Decision Sci.*, vol. 19, no. 3, pp. 598–621, Summer 1988.

[36] P. Jackson and P. Lefrere, "On the application of rule-based techniques to the design of advice-giving systems," *Int. J. Man–Machine Stud.*, vol. 20, no. 1, pp. 63–86, Jan. 1984.

[37] D. Hasling, W. Clancey, and G. Rennels, "Strategic explanations for a diagnostic consultation system," *Int. J. Man–Machine Stud.*, vol. 20, no. 1, pp. 3–19, Jan. 1984.

[38] D. B. Paradice and J. F. Courtney, "Causal and non-causal relationships and dynamic model construction in a managerial advisory system," *J. Management Inform. Syst.*, vol. 3, no. 4, pp. 39–53, Spring 1987.

[39] B. Kosko, "Fuzzy Cognitive Maps," *Int. J. Man–Machine Stud.*, vol. 24, no. 1, pp. 65–76, Jan. 1986.

[40] H. B. Asher, *Causal Modeling*. Beverly Hills, CA: Sage Publications, Inc., 1983.
[41] J. F. Courtney, G. R. DeSanctis, and G. M. Kasper, "Continuity in MIS/DSS research: The case for a common gaming simulator," *Decision Sci.*, vol. 14, no. 3, pp. 419–439, July 1983.
[42] H. J. Will, "Model management systems," in *Information Systems and Organization Structure*, Erwin Grochla and Norbet Szyperski, Eds. New York: Walter de Gruyter and Co., 1975.
[43] R. Sprague and H. Watson, "A decision support system for banks," *Omega*, vol. 4, no. 6, pp. 657–671, 1976.
[44] G. G. Lendaris, "Structural modeling: A tutorial guide," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-10, no. 12, pp. 807–840, Dec. 1980.

Nassar H. Ata Mohammed received his B.S. and M.S. degrees in chemical engineering in 1970 and 1972, respectively, from the University of Texas and received a Ph.D. in business administration (MIS) in 1985 from Texas Tech University, Lubbock, TX. He is on the faculty of King Fahd University of Petroleum, Dharhan, Saudi Arabia. He previously was a manager of computing consulting services at the King Fahd University of Petroleum and has worked as an information systems consultant for several organizations in Saudi Arabia and the United States. He was a speaker at the Third National Computer conference in Jeddah. His research interests are in knowledge-based decision support systems. Aspects of his research have appeared in *Decision Sciences*.

James F. Courtney, Jr. received his Ph.D. in business administration (management science) from the University of Texas at Austin in 1974.

He is Tenneco Professor of Business Administration in the Business Analysis and Research Department at Texas A&M University. His academic experience includes faculty positions at the Georgia Institute of Technology, Texas Tech University and the State University of New York at Buffalo. Other experience includes positions as Database Analyst at MRI Systems Corporation (presently INTEL Corporation's Austin division), and Visiting Research Scientist at the NASA Johnson Space Center. He has published papers in several journals, including *Management Science*, *Communications of the ACM*, IEEE TRANSACTIONS ON SYSTEMS, MAN AND CYBERNETICS, *MIS Quarterly*, *Decision Sciences*, *Decision Support Systems*, the *Journal of Management Information Systems*, *Database*, *Interfaces*, the *Journal of Applied Systems Analysis*, the *Journal of Experiential Learning and Simulation*. He is the codeveloper of the Systems Laboratory for Information Management, a software package to support research and education in decision support systems, and co-author of *Database Systems for Management*.

Dr. Courtney's present research interests are knowledge-based decision support systems, intelligent organizational systems and management information systems.

David Paradice received his B.S. in information and computer sciences in 1978 and an M.S. in industrial management in 1979, both from Georgia Institute of Technology, Atlanta, GA and the Ph.D. in business administration (MIS) in 1986 from Texas Tech University, Lubbock, TX.

He is Assistant Professor of MIS in the Department of Business Analysis and Research at Texas A&M University. His primary research interests are in investigating means for applying artificial intelligence techniques to managerial information systems. Related research includes investigating methods of dynamic model construction in managerial advisory systems and developing methods of utilizing qualitative models of organizational systems. Aspects of his research have appeared in *Journal of Management Information Systems*, *Decision Sciences*, and *Annals of Operations Research*.