AU: Please disregard spacing problems and figures out of order. They will be adjusted before the time of publication.

SIMON: An Object-Oriented Information System for Coordinating Strategies and Operations

David B. Paradice

Abstract—**Successful future organizations must possess information processing capabilities oriented toward actively supporting organizational strategies. Such an information system would support organizationally intelligent behavior, since the system would facilitate purposeful, goal-oriented operations. An information system needs at least three characteristics to map organizational strategies into operational transaction processing: it must model various organizational levels, it must support information processing across organizational levels, and it must support information processing across functional areas. A computer-based information system designed with these characteristics has the potential to take an active role in supporting organizational decision-making activities. The concepts of an intelligent organization and the information system required to support it are presented. An object-oriented prototype is described that implements key components for supporting organizationally intelligent behavior. The advantages of this design are discussed, along with issues still needing to be resolved.**

I. Introduction

THE COMPLEXITY of modern organizations places new demands on organizational information systems. Successful future organizations will be those that demonstrate the most effective and efficient information processing [1]. Huber [2] has observed that future organizations will be faced with an environment characterized by more and increasing amounts of knowledge, complexity, and turbulence. These trends will require greater processing capabilities of the information systems that are expected to provide support for organizational decision-making processes. At least three characteristics of organizations will require enhanced information systems support in the future.

First, organizational layers must be explicitly reflected in the information system architecture. Anthony [3] observed that organizations function at three levels: strategic, control, and operational. Each level has different implications for information processing [4]–[7]. Global organizational strategies are defined at the strategic level. Strategies are ideals for which the organization strives but can never attain [8]. An ideal, for example, might be striving for the most profitable utilization of the organization's resources.

Strategies are translated into operational realities at the control level. Operational realities are organizational objectives. Unlike ideals, objectives are realizable, at least in the long run. An objective to reduce overhead expenses by three per cent may be verified through periodic measurement, for example.

Below the organization's control level is the operational level. Operations represent a factorization of objectives into goals that are achieved in the short run by means of various organizational transactions. These transactions implement the day-to-day activities of the organization.

Few, if any, information systems architectures reflect these layers inherent in complex organizations. In fact, the terminology currently used in the information systems industry makes difficult conceptualizing systems this way. Instead, there are executive information systems (EIS) and executive support systems (ESS) for the strategic level [9], decision support systems (DSS) and management information systems (MIS) for the control level [5], [10], and electronic data processing (EDP) and transaction processing (TP) systems for the operational level. While specific implementations may cross organizational levels, the terminology accurately reflects how these architectures have become associated with specific organizational levels.

Second, the impact of organizational layers on decision-making effectiveness must be considered in information system design. As organizations become more complex, a delay occurs between the time strategy is formulated and when a transaction implementing the strategy actually occurs. For example, the decision to develop a new product may be made months before the new product is actually developed. Environmental conditions may change between the time that a strategy is developed and when it is implemented. Thus, strategy implementation may result in one or more undesirable consequences including inefficient allocation of resources, lost market position, and financial hardship. Additionally, environmental changes may cause operations that once implemented strategies to become detrimental. Organizational complexity inherent in multiple organizational levels delays the communication and monitoring of operations vis-à-vis the organization's strategies. Although delay is inherent in complex organizations, few information system designs specifically mitigate its impact in the information management process. Of particular importance is a methodology that will focus managerial attention on changes in the organization's environment that impact organizational strategies [11].

Third, increased complexity within an organizational level must be addressed by the organizational information system. Another consequence of organizational complexity is that more specialized organizational units within an organizational level may be called upon to address a particular issue or

Manuscript received January 20, 1989; revised October 6, 1990.
The author is with the Business Analysis and Research Department, College of Business Administration, 401 Blocker Building—BANA, Texas A&M University, College Station, TX 77843-4217.
IEEE Log Number 9106503.

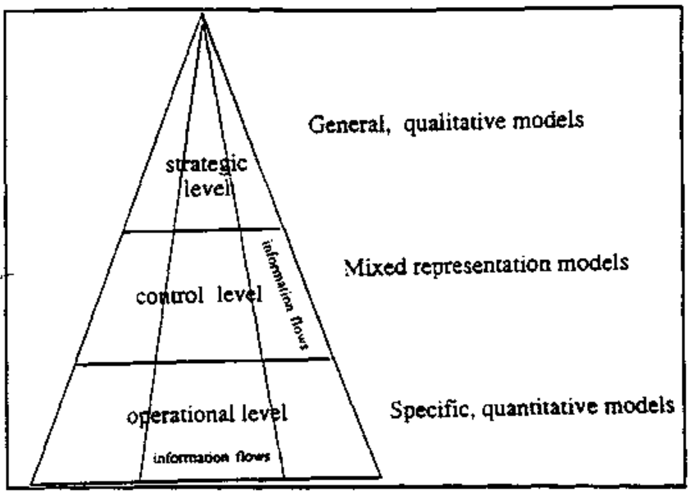

Fig. 1 Organizational information systems must support different model types and information flows.

problem. Consider, for example, a client service function in a typical retailing/mail order business. A customer's request for assistance regarding an order may require the customer service representative to access accounting information to verify payment, marketing information to verify applicability of a sales promotion, and inventory information to verify product availability. Integrated database systems address this issue, but one can expect that increased information system support within any specific organizational level will be required as organizational complexity increases [12].

In summary, three characteristics are required to attain the information processing capabilities of the future (Fig. 1). First, information systems architectures must reflect multiple organizational levels. Second, the design must support vertical processing across organizational levels. Third, the design must support horizontal processing across functional areas at any given level.

II. Modeling Organizational Levels—Conceptual Design

Ideally, the information system design would reflect naturally the characteristics of Anthony's trilevel pyramid. A hierarchy would exist, with fewer organizational entities at the strategic end of the pyramid and greater numbers of entities at the operational end. An object-oriented design has these characteristics.

An object-oriented design approach focuses on the entities that participate in a system [13]. Object-oriented approaches have been explored elsewhere in MIS research, notably in systems analysis and design [14] and organizational modeling [15]–[17]. Chen and Nunamaker [16] noted a need to integrate organization and information system modeling, and observed that the object-oriented paradigm could support this integration. Rajkumar and Yadav [18] observed that an object-oriented approach would support different levels of organizational models.

Generally speaking, objects have private features that might be manipulated via a public interface. Features are object attributes and thus all of an object's characteristics are encapsulated in the object's description. The public interface consists of processes, generally called methods, which may change the state of an object's attribute.

Objects may be defined in a hierarchy of relationships. Objects lower in the hierarchy are specializations of their hierarchical parents. These objects inherit the features and methods of their parents, although all of the parent's methods are not necessarily included in the specialized object's public interface. Additionally, the specialization object may have unique features and methods.

Applying the object-oriented paradigm to organizational information systems is not trivial, however. The information system must be designed to cope with fundamentally different representations of the organization's mission; strategic-level models differ from operational-level models. The systems envisioned for the future will require an ability to translate problem representations from one organizational level to the next. Without this ability, the system will not be able to address the impact of the delay in communicating from the operational level to the strategic level.

An ability to perform a representation shift among problem descriptions appears necessary in managerial problem solving. Skilled decision makers have an ability to "perceive patterns from the complex reality and to grasp layers of event flow at a variety of levels" [19]. Each managerial functional area contains numerous abstract models: funds flow models, accounting models, inventory models, product mix models and so forth. Managers must frequently begin with a problem description, identify an abstract model that addresses the problem, and convert the problem description to an internally solvable (although not always necessarily quantitative) representation. Successful problem-solving activity may require a series of representation shifts in either direction.

Managerial expert systems should support an ability to move between model representations, as each representation may have distinct advantages. However, many expert systems function by considering only a single problem representation and analyzing observed data. Often, a highly problem-specific model is developed and executed without regard to the inherent interrelationships between models in a managerial problem domain. For example, domain specific approaches are used in systems developed in accounting [20]–[23], financial planning [24]–[27], manufacturing [28], [29], and management [30], [31]. A potential flaw in this approach, however, is that recommendations may be made by production departments (for example) to expand capacity only to have financial departments veto the plan due to insufficient funding. Each group basing its judgments on its expert system.

Problem-specific models, however, make sense. Due to managerial domain complexity, observing enough decision-making variables to describe a comprehensive input data set from which to derive valid conclusions at a macro-organizational level may be impossible. Problem-specific models provide focus in an environment that may preclude adequate description of a general, managerial domain knowledge base. Hierarchical maintenance of multiple model representations could be one approach to utilizing problem-specific models more effectively. First, problem-specific model representations could be integrated within levels of the hierarchy, allowing some integration. Second, the hierarchy itself, via model representation shifts, could provide a focussing mechanism that would help decision makers cope with managerial domain complexity.

Representation shifts must incorporate an ability to shift attention between high-level, descriptive problem models (i.e., strategic models) to knowledge in a lower, more definitive model (i.e., operational model). This underlying model is frequently built from "causal" (cause-and-effect) relationships, and is said to embody "deep knowledge" [32]. Causal models are used to focus attention on specific data, thus providing a method for coping with complexity [33] by reducing the number of potential inputs to process and implicitly constraining the decision problem. Causal models are applicable in many domains, and are frequently sought by managers [34]. These models are attractive to managers because they provide structure to an otherwise ill-structured domain [35], [36]. Managers are better able to cope with the domain complexity through this structure.

Causal models are frequently used to diagnose managerial problems [34], [37]. Kasper [38], Pracht [39], and Loy [40] have demonstrated the effectiveness of supporting the construction of causal models in managerial problem situations. Courtney and his associates [41]–[45] have shown viable systems frameworks for capturing, organizing, and manipulating management-oriented causal models.

Davis [46], [47] has argued that causal models in diagnostic systems are superior to empirically based diagnostic systems. Causal models attempt to capture relationships and basic principles of expert knowledge [48]–[51]. They can deduce behavior from structure [52], [53] and may be able to build predictive models of future behavior [54].

Chandrasekaran and Mittal [51] have observed that the straightforward approach taken by expert systems employing condition-action production rules is not feasible in problem domains of any significant size. Hence, deep (i.e., causal) models will be necessary in large problem domains. Management is certainly one such domain.

III. Modeling Organizational Levels—An Implementation

A prototype system has been developed to investigate the feasibility of the design described previously. The system has been named SIMON, from the term Strategic Information Monitoring system. SIMON is implemented on a SUN 3/160 workstation using Quintus Prolog's object-oriented Prolog [55]. A window and icon interface is used, which allows users to simply "point and click" to verify strategies, hypothesize trends, investigate financial ratios, process transactions, and construct financial statements. Each of these activities is described in more detail in the following subsections..

A. Strategic Level: Modeling Qualitative Concerns

Since the organizational model at the strategic level is general and typically used to manipulate data expressed in imprecise terms, qualitative causal modeling concepts are utilized at this level. Qualitative causal modeling works with strictly qualitative descriptions of system structure.

Qualitative causal modeling has evolved from artificial intelligence (AI) approaches to modeling mechanisms and physical systems. Rieger and Grinberg [56] first used qualitative causal links to relate events, tendencies, states, and state changes in mechanisms. De Kleer and Brown [57], [58] developed a methodology for reasoning about mechanisms based on the relationship between structure and behavior. They pioneered the envisioning approach, wherein future states of a system become non-deterministic. When there is not enough qualitative information to determine a future state, external (nonqualitative) sources of information are used. De Kleer and Brown also developed an algebra for propagating incremental qualitative values.

Forbus [59] extended de Kleer's approach to include signs and magnitudes of values. These extensions were necessary to handle many complicated domains, especially domains involving moving, interacting objects. Kuipers [52] extended the techniques of qualitative causal modeling in another manner. His work developed a general approach for determining new, previously unknown equilibrium points in a system. Prior work established qualitative equilibria at states identified to the model by the modelers. Kuipers models' infer these equilibria themselves.

A distinguishing characteristic of the AI work is that the models have been applied to well-defined, mechanical systems. Hart, Barzilay, and Duda [60] have suggested extending the AI approaches to business domains, but no such extensions have been reported. Additionally, Iwasaki and Simon [61] argued that the work in qualitative causal modeling could be made more rigorous by applying formal characterizations of causality [62] and the method of comparative statics [63] from economics.

Examples of qualitative descriptions of managerial strategies are "increase revenues," "decrease expenses," and "improve market share." Such descriptions, though terse, are meaningful to senior managers. The SIMON system uses knowledge of terms such as increase and decrease to interpret strategies like these. Strategies may be defined in terms of variable classes: currently revenue, expense, market, and production variable classes are defined. SIMON's object-oriented structure allows access to specific corporate database variables in each class. Thus, trends may be calculated to determine whether strategies are being implemented.

Qualitative causal modeling is further supported by object attributes that assume qualitative values and methods that manipulate qualitative values. For example, a method may be defined to change the value of a variable named "dividend" from "steady" to "decreasing" when an attribute named "profit" changes from "steady" to "decreasing" or when an attribute named "operating expense" changes from "steady" to "increasing."

SIMON manipulates qualitative values at the strategic level. For example, the system can take as an input that sales are increasing and determine the impact on revenues. The system translates "increasing" into a one-unit change in the value of a variable, then propagates that change throughout lower-level models. The change in revenues is then translated into a qualitative response. Of course, a user could query the prototype in quantitative terms as well.

SIMON contains facilities for verifying strategy adherence and for hypothesizing expected behaviors. For example, a strategy such as "increase revenues" may be verified by traversing the object hierarchy and determining the status of all revenue class variables. The system determines whether the class is increasing. SIMON can also traverse causal links at various organizational levels and hypothesize whether variables affected by others are increasing, decreasing, or steady. Once hypothesized, the system can investigate the hypothesis in the same manner that it verifies strategies.

B. Control Level: Explanation Facilities and Financial Ratio Analysis

Middle managers need organizational models that are more detailed, yet narrower in scope than those used by managers at the strategic level. Since the middle manager's role focuses on control, models that are more quantitative in nature than the type used at higher organizational levels are frequently desired. The quantitative aspects provide means for measurable evaluations. A model incorporating financial ratio analysis can be used at this level, because financial ratios capture a degree of summarizing detail in a quantitative form.

The SIMON system models financial ratios as an object class. Ratios are defined in terms of the subclasses of the strategic level class variables. For example, the current ratio is modeled as having short-term assets (a subclass of the revenue variables class) in the numerator and short-term liabilities (a subclass of the expense variables class) in the denominator. Instances of the short-term assets subclass are operational level variables such as cash, short-term investments, and finished goods inventory). The ratio can be "connected" to the strategic level by specifying a strategy such as "steady current ratio." Thus, the SIMON system can move from a (relatively) high level representation embodied in the ratio itself to a more detailed (deeper) problem representation embodied in the mathematical combination of the variables. In this manner, the deep knowledge embodied, but not explicitly represented, in the surface model can be connected to the deep knowledge reflected in an income statement or a balance sheet.

Because a current ratio is ultimately modeled as having cash, sales, and inventories variables in the numerator and accounts payable and credit expense variables (operational level variables) in the denominator, the deep knowledge embodied in the terms of the financial ratio contains some indication of how middle managers can control organizational processes. The SIMON system contains basic knowledge of mathematical relationships that are used to explain why financial ratios are moving as they are. Since the components of the ratio are specified in terms of organizational variables, the system can examine the organization's database to determine which of the components are influencing the ratio.

This explanation facility is derived from the structure of the ratio and the variable relationships, not from rules specific to either. Thus, the system can manipulate any financial ratio incorporating any organizational data item without further enhancement. This approach is attractive in the business domain since the breadth and complexity of the domain may make an expert system based on comprehensive rule specification impossible. The SIMON system requires only specification of the ratio and the relationships between the organizational variables.

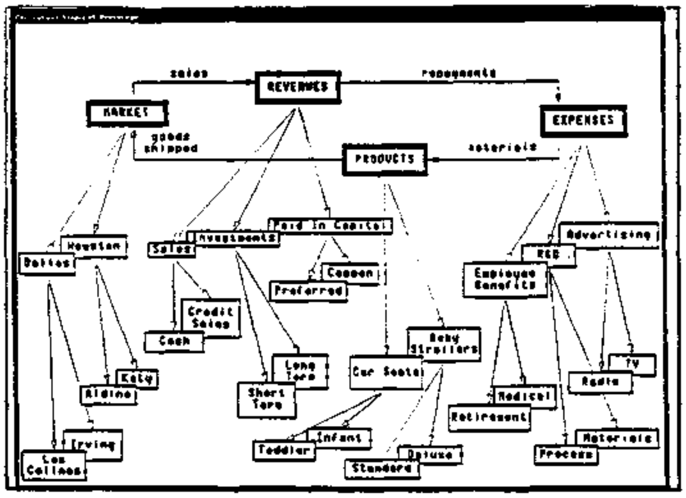

Fig. 2. Hierarchical nature of strategic qualitative concepts to specific operational data items.

SIMON uses knowledge of the organizational structure in a second manner. It can trace the links implied in lower-level causal models to recommend specific alternative actions that can be executed in order to implement qualitative strategies such as "decrease expenses." As shown in Fig. 2, the object orientation provides the system with knowledge (via the object hierarchy) to determine that "advertising expenses" and "R&D" are expense class variables. The causal relationships between these expenses and other variables at the operational level can then be investigated to determine the impact of manipulating these expenses.

C. Operational Level: Transactions and Financial Statement Construction

The organization's operational level is characterized by a focus on internal organizational concerns that are frequently embodied in transaction processing. Ideally, the organizational information system would monitor transactions as they occur. It could then evaluate each transaction's impact on the organizational model to determine whether the organization's actions are deviating from established strategies [64].

As mentioned previously, the primary means for connecting the operational and strategic organizational levels is through the object hierarchy. The control level is also connected to the operational level by specifying financial ratios in terms of operational level variables.

At the operational level, organizational models need to have their closest relationship with current, accurate data. Because financial statements are one form of operational level model, SIMON's operational level can automatically construct assorted financial statements. Financial statements are also modeled as objects, each with one attribute indicating a list of organizational variables that it contains.

A balance sheets represent a "snapshot" of an organization's financial condition at a specific time. It can be constructed at any time given knowledge of an initial financial position and any subsequent financial transactions. Similarly, the income statement shows the result of an organization's operations for a specific period. Like the balance sheet, it can also be constructed at any time. The SIMON system has been built to monitor transactions and instantly update the balance sheet and income statement. Similarly, since financial ratios are specified in terms of operational variables, financial ratios are also updated dynamically as the system monitors transactions.

IV. Advantages of the SIMON Architecture

The multilevel, causal structure architecture developed for the SIMON system yields a number of distinct advantages over other knowledge-based systems designs.

A. Problem-Solving Efficiencies

An ability to translate problem representations across organizational levels provides many economies in problem-solving behavior. The ability to communicate in very abstract terms allows problem descriptions to be made without detailed explanations of all implications. For example, consider the top level funds flow model in Fig. 2. This abstract model allows one to describe rather complex business phenomena in quite simple terms. The model represents the occurrence of a market that accepts goods from the organization and returns revenues to the organization. The revenues pay expenses that provide the resources to generate more goods.

Among knowledgeable, communicating agents, high-level (or surface) descriptions such as this suffice. This representation is adequate, however, only because underlying, more detailed models of domain structures exist. The importance of the deeper model is evident when consequences of changes in the surface model are considered. For example, consider the proposition that suddenly the link between the market and revenues fails to function as hoped. Concluding, quite properly, that this failure could lead to personnel layoffs can only be attributed to a deeper understanding of the system and a corresponding shift in model representation required to support the deep analysis.

The multilayered approach also provides efficient means for providing explanations. The implication that a revenue reduction will lead to layoffs is much more economically stated than describing the rates at which the employees will be dismissed and the corresponding savings that will be gained by the organization.

Finally, the SIMON architecture can increase the efficiency with which a system maintains action agendas. For example, a plan to increase cash can be understood to involve several actions such as selling fixed assets, decreasing accounts receivable collection periods, and foregoing dividend payments. Again, SIMON's ability to understand "increasing cash" comes from translating that strategy into deeper causal relationships.

B. Hypothesis Generation

This system supports dynamic hypothesis generation in two ways. First, the causal relationships stored in the organizational model could be interpreted as hypotheses. A corporate database can be accessed automatically to verify these relationships. Where the relationships are "proven" by the empirical analysis, the organizational model is "validated." Where the hypotheses are not verified, the system signals that the organizational model components reflected in these hypotheses need review.

Second, the financial ratios can be examined automatically by the system. When a ratio is found to have changed by a "significant" amount (where significant may be measured by deviation from a historical average or deviation from industry mean) the components of the ratio are examined individually to detect which one(s) explain the change. Here, the components of the ratio combined with simple knowledge about arithmetic ratios is used to form the hypotheses. The system queries the database to determine which variables have changed significantly and again signals that these variables may need managerial attention.

In each of these cases the system has taken a proactive role in identifying "interesting" situations that may require management's review. This approach makes the information system an active partner in the management of the organization. Rather than being a passive reservoir of data, the system becomes an active participant in the daily activities of the firm.

C. Information Anticipation

When viewed as components of an information anticipation system, the prototype design may reasonably be used in a more active role yet. The information system can recognize when the organization is not making progress toward specified strategies. It may now selectively screen communication over the organization in anticipation of information related to these strategies.

Huber [12] observed that delays are inherent in information systems and that delays may operate in ways that make the information system more effective. Delays may be influenced by organizational directives. Therefore, the duration of the delay associated with communicating the desired strategy to the organizational unit capable of implementing the strategy may be modified by an organizational preference function. By ordering the strategy information based on the organization's preference for strategy realization, a basis for the selection of alternative operations relevant to strategies may be made.

Suppose there are three strategies specified: produce a new product, advertise the new product, and increase research and development output. Suppose also that all strategies are simultaneously desired (i.e., there is no management directed delay in strategy communication) and that operations are occurring only to produce the new product. All that remains is to decide whether available funds will be spent on advertising the new product or in increasing research and development. If producing the new product represents a major undertaking (i.e., a long lead time exists before the new product can be effectively marketed), then an increase in research and development expenditures may be the best use of financial assets. On the other hand, if a competitor announces a plan to market a similar product, then advertising the new product may represent a better allocation of the limited funds.

Heuristics could be added that would take hypothesized conditions into consideration. For example, the heuristics in this scenario could be stated "if there is no competition, then delay advertising until production is nearly complete, otherwise begin advertising the new product." At the very least, a range of alternative courses of action could be generated for managerial review.

These scenarios demonstrate how an information system with extended recognition of strategies and operations assists the organization in becoming more adaptable to its environment, a necessary characteristic for long-term survival. Huber [2] has noted that future organizations operating in the greater turbulence of the post-industrial environment will require the ability to assimilate information faster, more frequently, and in the context of greater complexity than has been required of past organizations. The incorporation of strategies and descriptions of the interrelationships of the operations that implement strategies provides a methodology for directing the environmental scanning activity required by post-industrial organizations to ensure their survival.

D. Support for "Virtual" Organizational Teams

Many transactions that enter an organizational system cross functional area boundaries. Typical client service operations, for example, may involve accounting, inventory, marketing, and training personnel. The causal structure built into the system allows the system to accept a transaction that enters as a "client service" transaction and automatically generate messages to other related functional areas. Conceptually, staff personnel in the related areas would be notified that their services are required and resources in their area could be allocated to service the request. This dynamic allocation of resources produces a "virtual" team formed by organizational members from different functional areas on an as-needed basis.

E. Knowledge from Structure

The managerial problem domain is too wide and too semantically rich to hope to develop strictly rule-based expert systems for general organizational support. The SIMON architecture derives some of its functionality from the knowledge of organizational structure embedded in the system. Rather than having rules that indicate specific objects that should be notified to process transactions, the system has rules to send messages to "relevant" objects as described via the causal or compositional structure of objects.

For example, one must only specify that an object appears on the balance sheet. The system then monitors all changes to the balance sheet and sends messages to the object accordingly. Or, one must only specify that a variable is an expense variable, and the system will monitor that variable in light of specified strategies regarding expenses.

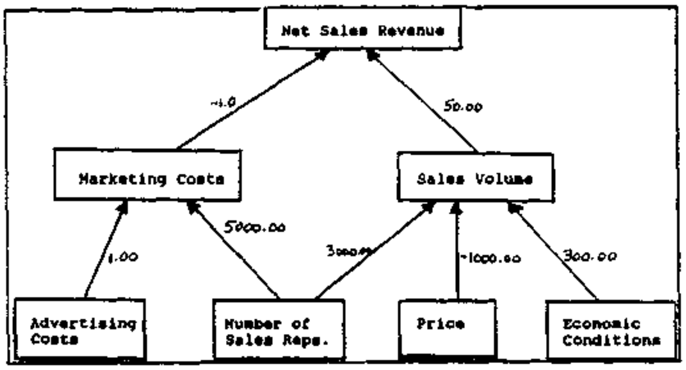

Fig. 3 Structural model of relationships between specific data items.

V. A Sample Session with the SIMON System

The example used in Ata-Mohammed *et al.* [42] is used to demonstrate system use. The structural model of relationships between variables is reproduced in Fig. 3.

Fig. 4 shows the initial SIMON interface for a small database. The view window provides information on the current state of database items. As items are selected (illustrated below), the current status of the item is displayed in the view window. The status window provides information regarding SIMON processes. Error messages or confirmation messages are displayed here. The console window is used to present reports to users, such as the balance sheet and income statement pro formas available in this illustration. Users may rearrange the screen at will if a different arrangement is preferred. The SIMON system monitors their location automatically. (The window labeled *cmdtool - /bin/csh* appears only to enable capturing the screen contents for this paper. It would not normally appear otherwise.)

Also shown in Fig. 4 is a small complement of the methods that are implemented in the SIMON system. In the upper right corner of the screen are system methods that provide various functions. Files may be read, transactions may be initiated, and the database contents may be "posted" to create a database synchronization point. Other methods available correspond to the items listed in each box for each organizational level. Thus, for strategies, one may either print a listing of a selected strategy description or one may check whether a selected strategy is being followed. Strategies are selected by moving the mouse cursor to the appropriate box and clicking a mouse button. This action places a check in the box, indicating the strategy has been selected.

Fig. 5 illustrates the SIMON interface when net_revenue has been selected by the user. (Database items are selected by simply moving the mouse pointer to the item and clicking on the item.) The view window indicates the current status of the net_revenue item. Figs. 6 and 7 illustrate examining the impact of adding a sales representative to the sales force. Fig. 6 shows the interface after selecting no_sales_reps from the Operations Level window and impact from the O_Methods window. The transaction is adds a value of 1 to the number of sales representatives and calculates the direct impact on other variables. Fig. 7 shows the system response that adding one sales representative will increase marketing costs $5000.00 and

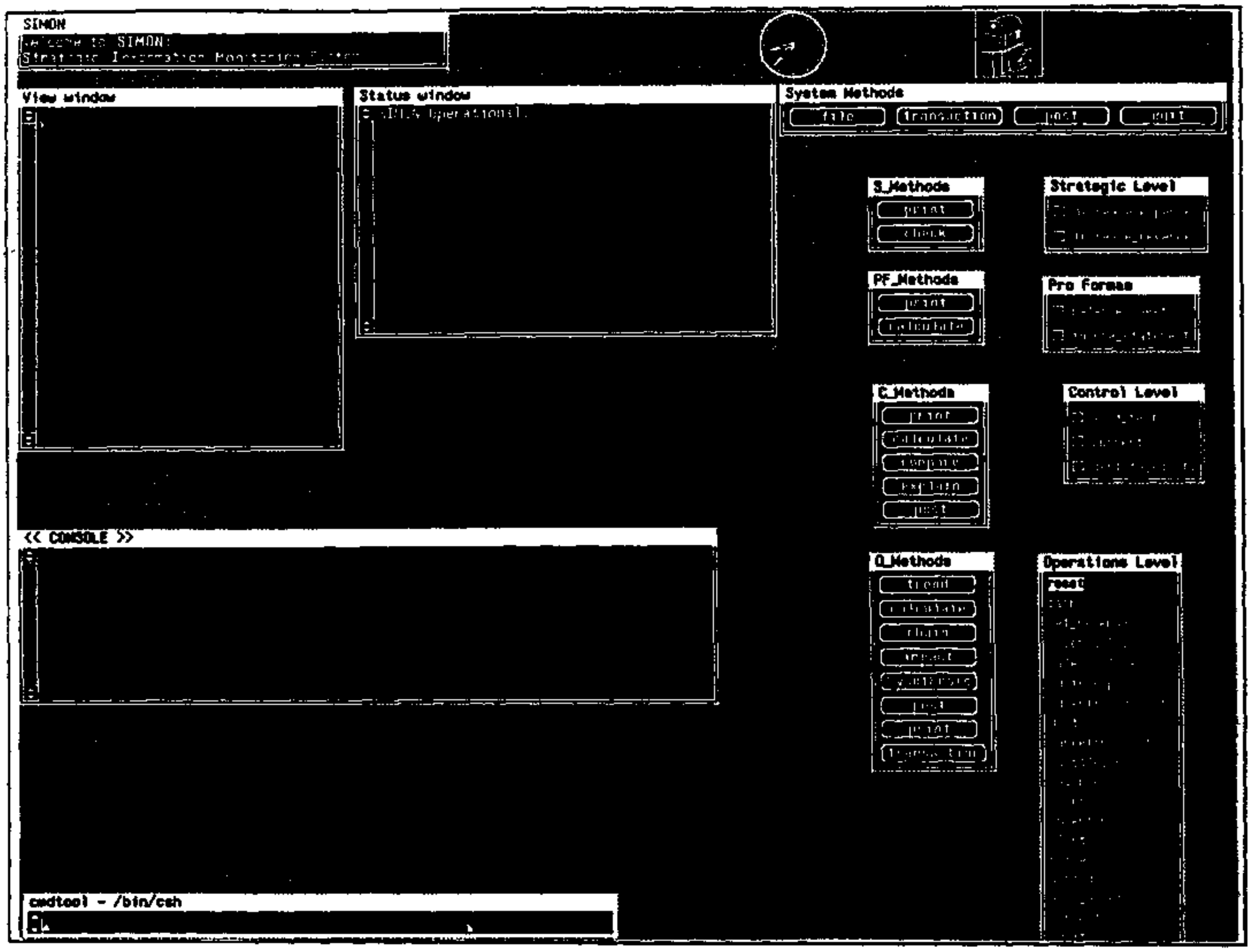

Fig. 4 The initial SIMON interface.

increase sales volume by 3000 units.

Fig. 8 shows the indirect impact of the same transaction to add one sales representative. This action was created by selecting no_sales_reps from the Operations Level menu and chain from the O_Methods window. (The transaction creates a "chain reaction" of impacts.) The console window is used to communicate that the proposed increase results in a $5000 reduction in net revenue due to increased marketing costs and a $150000 increase in net revenue through increased sales.

Figs. 9–11 illustrate the dynamic nature of the pro forma calculations. Fig. 9 shows the current status of the income statement in the console window. Fig. 10 illustrates a transaction to increase cash by 12000. This was effected by selecting transaction from the O_Methods window. Fig. 11 shows the new income statement, calculated dynamically from the database contents. In this manner, a manager could have information as current as the database.

VI. Areas of Future Research

Proper financial ratio analysis requires identification of 1) the purpose of the analysis and 2) organizational and environmental factors not evident from the financial statements. Financial statement analysis may be driven by several purposes. One may be interested in the organization's profitability for investment purposes, or one may be interested in the organization's liquidity in evaluating a loan request.

The purpose of analysis has immediate impact on the interpretation of financial ratios calculated. If a loan manager is reviewing an organization's financial statements with regard to a short-term loan, a high current ratio may be attractive since it could be interpreted as an ability to repay the loan. On the other hand, a manager in the organization may view the high current ratio as an inefficient use of the organization's liquid assets.

Additionally, the assumptions implicit in each ratio's calculation needs to be examined. In some cases, organizational factors may not be adequately represented in the financial statements. For example, in interpreting a return on assets calculation, one must be aware of whether the organization's assets will in fact appear on the balance sheet. A primary asset of a software development firm, for example, is its personnel. A high return on assets calculation for this type of firm could be misinterpreted as being better than it really is, since typically only fixed assets are included in the calculation.

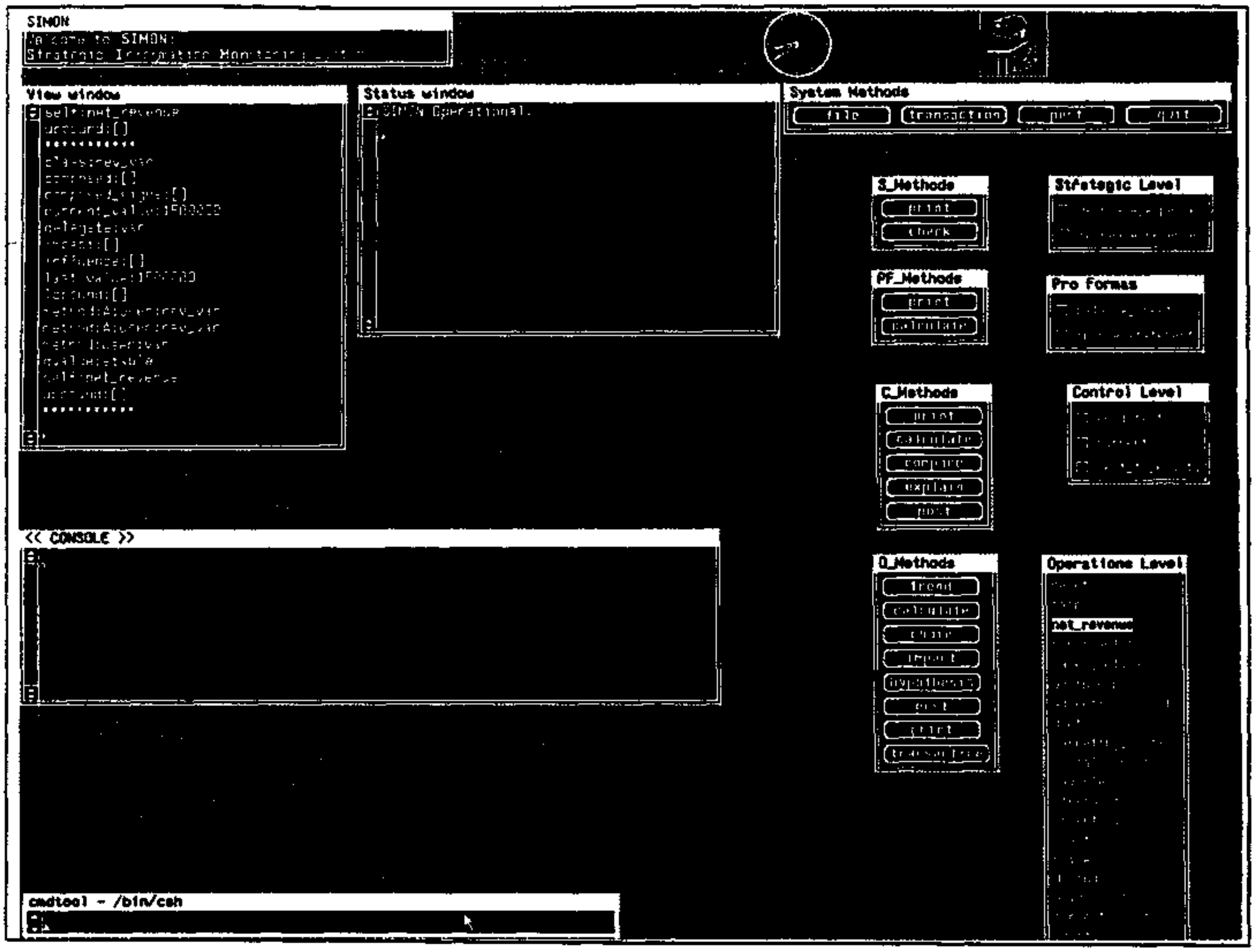

Fig. 5. SIMON description of the "net_revenue" data item.

These situations reflect the need to develop structures and rules addressing each condition. The analysis purpose and the characteristics of the organization that could influence the financial ratio interpretation could be included as facts. Of course, industrial averages could be included for use in the analysis.

To illustrate, the following simple rule for evaluating the liquidity of an organization requesting a short term loan can be included in the system.

The liquidity for a short-term loan decision is good if the current ratio is greater than 1, and the acid test ratio is greater than 1, and the receivables collection period is less than the payables purchases period.

As stated previously, a detailed analysis would require examination of each ratio in light of the characteristics of the organization being analyzed, since these characteristics could influence the interpretation of the ratios. However, this example illustrates a general approach to the analysis.

The capability to instantly update balance sheets and income statements provides a means for these organizational models to become active tools in the management process, rather than passive reports. An ultimate goal of the SIMON design is to link strategy specification with financial statement construction, so that the financial statements may also be monitored automatically.

VII. Conclusion

In order to take an active role in management, organizational information systems must become more intelligent. One area in which many information systems need improvement is in the explicit recognition of the multilevel structure of complex organizations. Organizational information systems must recognize that strategies are formalized at the strategic level while transactions that implement strategies occur at the operational level. Since a delay may exist between the formulation of strategies and the occurrence of transactions that implement them, information systems must monitor strategies and transactions.

Intelligent behavior can frequently be described in terms of selection of a course of action that enables goal satisfaction. The proper course of action should be directed by the goals of the system, not generated randomly until progress toward a goal is made. The methodology presented here provides goal

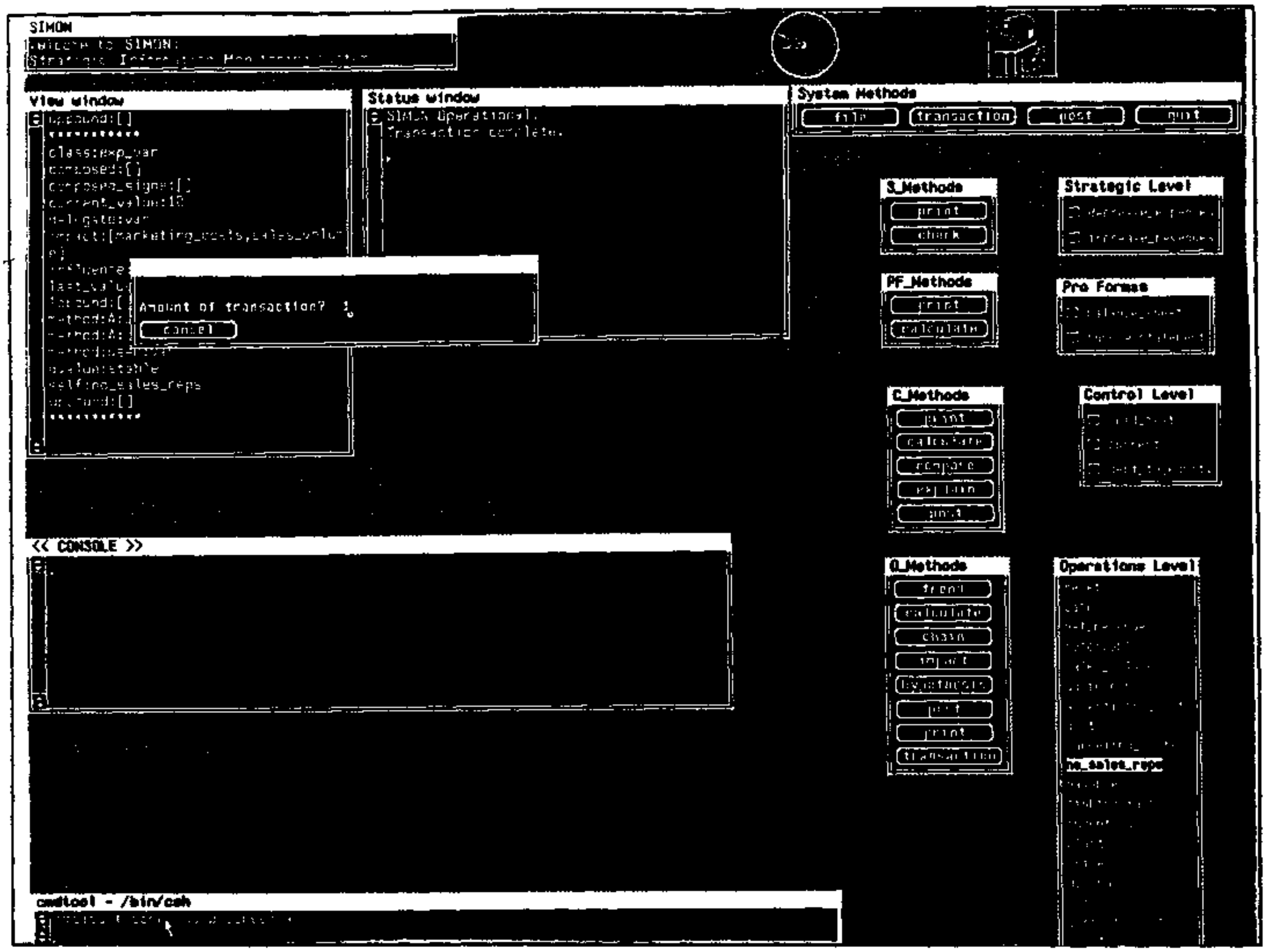

Fig. 6. Initial process of determining impact of adding one sales representative.

directed selection capabilities where the goal is implementation of a desired strategy.

This system design described here has many advantages. It supports hypothesis generation derived from a description of the system's structure, rather than derived totally from a knowledge base of specific rules. It supports the concept of information anticipation, since the system can deduce from its internal structure that organizational transactions are relevant to strategies. The design also supports "virtual" organizational teams, because the system automatically sends messages to objects in different functional areas.

Conceivably, the system could be "data driven" so that a "discovery module" could first determine which variables were changing substantially and thus invoke a model at any level. A natural source for data to examine is the corporate database. Once discovered, the qualitative causal model could be invoked with the states of variables set per the discovery process. Then, the impact of these trends, reflected in the model, could be evaluated by management.

In any case, corporate information systems must take an active role in managing the complexity inherent in the managerial domain. Only those that do will adequately meet the information processing needs of future organizations.

References

[1] I. F. Jackson, *Corporate Information Management*. Englewood Cliffs: Prentice-Hall, 1987.

[2] G. Huber, "The nature and design of post-industrial organizations," *Management Sci.*, vol. 30, no. 8, pp. 928–951, Aug. 1984.

[3] R. A. Anthony, "Planning and control systems: A framework for analysis," AUTHOR: IN WHAT KIND OF PUBLICATION DID THIS APPEAR?? Division of Research, Graduate School of Business Administration, Harvard University, Boston, MA, 1965.

[4] G. A. Gorry and M. S. Scott Morton, "A framework for management information systems" *Sloan Management Rev.*, vol. 13, no. 1, pp. 55–70, Fall 1971.

[5] P. G. W. Keen and M. S. Scott Morton, *Decision Support Systems: An Organizational Perspective*. Reading: Addison-Wesley, 1978.

[6] R. H. Sprague and E. D. Carlson, *Building Effective Decision Support Systems*. Englewood Cliffs: Prentice-Hall, 1982.

[7] J. F. Rockart, "The line takes the leadership—IS management in a wired society," *Sloan Management Rev.*, pp. 57–64, Summer 1988.

[8] R. Ackoff, "Towards a system of systems concepts," *Management Sci.*, vol. 17, no. 11, pp. 661–671, July 1971.

[9] B. C. McNurlin, Ed., *EDP Analyzer*, vol. 25, no. 4, Apr. 1987.

[10] R. W. Blanning, "The functions of a decision support system." Information and Management, vol. 2, no. 3, pp. 87–93, Sept. 1979.

[11] R. O. Mason and E. B. Swanson. "Measurement as an MIS Foundation." Data Base, vol. 14, no. 1, pp. 27–29, 1982.

[12] G. Huber, "Organizational Information Systems: Determinants of Their Performance and Behavior." Management Science, vol. 28, no. 2, pp. 138–155, Feb. 1982.

[13] P. Wegner, "The Object-Oriented Classification Paradigm." Research Directions in Object-Oriented Programming, 1988.

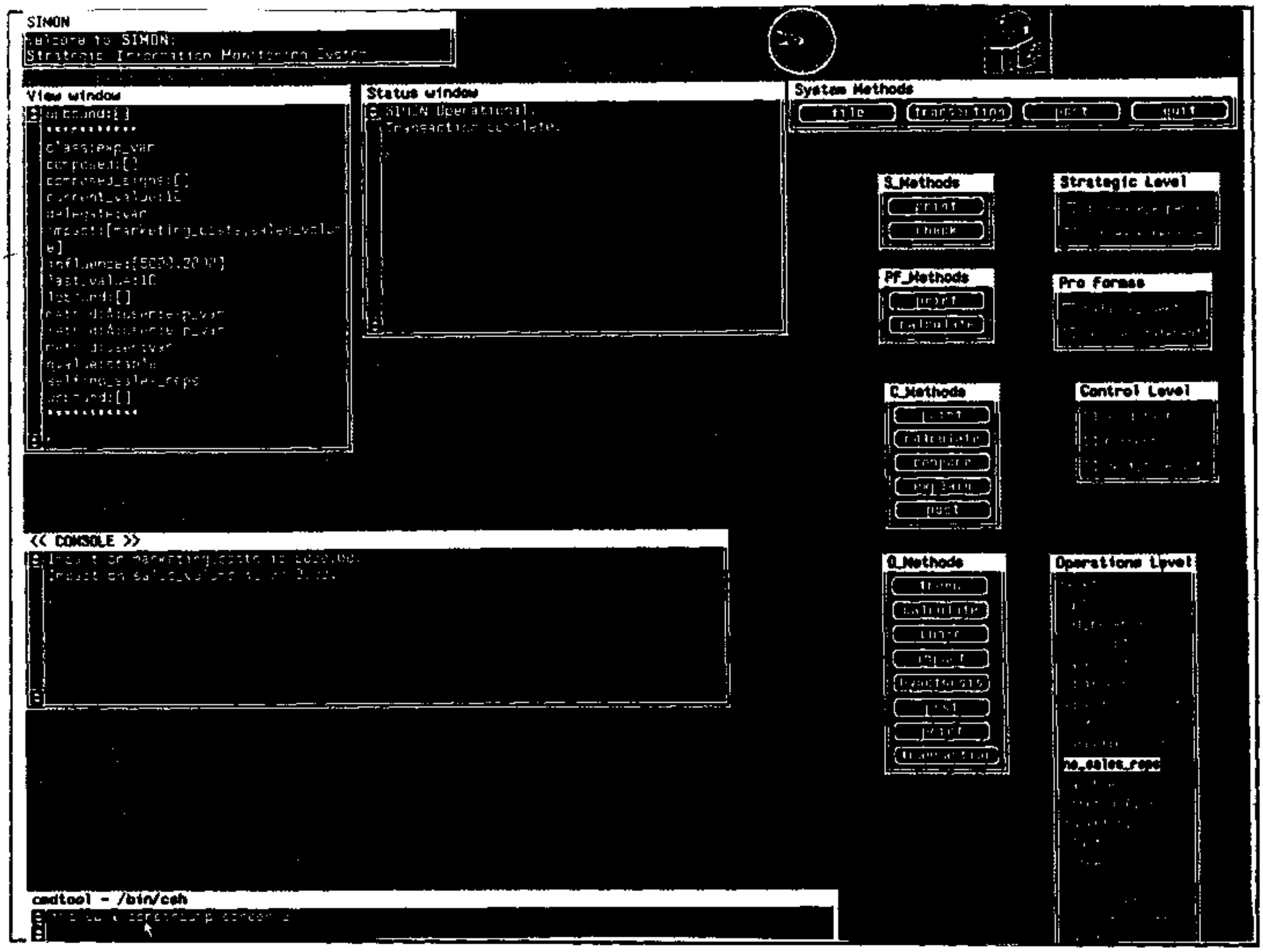

Fig. 7. Impact (determined by SIMON) of adding one sales representative.

[14] S. McIntyre and L. Higgins, "Object-Oriented Systems Analysis and Design: Methodology and Application," J. Management Information Systems, vol. 5, no. 1, pp. 25–35, 1988.

[15] R. W. Blanning, "Sensitivity Analysis in Logic-Based Models," Proc. Twentieth Annu. Hawaii Conf. Syst. Sci., pp. 665–674, 1987.

[16] M. Chen and J. F. Nunamaker, "Integration of Organization and Information System Modeling," Proc. Twenty-Second Annu. Hawaii Int. Conf. Syst. Sci., vol. 3, pp. 70–79, 1989.

[17] S. McIntyre and L. Higgins, "Embedding Stakeholder Analysis in Object-Oriented Organizational Modeling," Proc. Twenty-Second Annu. Hawaii Conf. System Sciences, pp. 80–86, Jan. 1989.

[18] T. J. Rajkumar and S. B. Yadav, "Requirements for an Extensible GDSS: An Object-Oriented Architecture," Proc. Twenty-First Annu. Hawaii Conf. System Sciences, pp. 125–130, Jan. 1988.

[19] T. Sawaragi, S. Iwai, and O. Katai, "A Human-Friendly Interface System for Decision Support Based on Self-Organized Multi-Layered Knowledge Structures," Toward Interactive and Intelligent Decision Support Systems; Proc. Seventh Int. Conf. Multiple Decision Making, Kyoto, Japan, August 18–22, 1986, edited by Sawaragi, Y., Inoue, K, and Nakayama, H. New York, Springer-Verlag, 1987, pp. 30–39.

[20] C. W. Dungan and J. S. Chandler, "Auditor: A microcomputer-based expert system to support auditors in the field," Expert Systems, vol. 2, no. 4, pp. 210–221, Oct. 1985. [21] J. V. Hansen and W. F. Messier, Jr., "A Preliminary Investigation of EDP-XPERT" Auditing: A Journal of Practice and Theory, vol. 6, no. 1, pp. 109–123, Fall 1986.

[21] D. Shpilberg and L. E. Graham, "Developing ExperTAX: An Expert System for Corporate Tax Accrual and Planning," Auditing: A Journal of Practice and Theory, vol. 6, no. 1, pp. 75–94, Fall 1986.

[22] P. J. Steinbart, "The Construction of a Rule-Based Expert System as a Method for Studying Materiality Judgments," Accounting Review, vol. 62, no. 1, pp. 97–116, Jan. 1987.

[23] J. Kastner, C. Apte, J. Griesmer, S. J. Hong, M. Karnaugh, E. Mays, and Y. Tozawa, "A Knowledge-Based Consultant for Financial Marketing," AI Magazine, vol. 7, no. 5, pp. 71–79, Winter 1986.

[24] H. Braun and J. S. Chandler, "Predicting Stock Market Behavior Through Rule Induction: An Application of the Learning-from-Example Approach," Decision Sciences, vol. 18, no. 2, pp. 415–429, 1987.

[25] C. Mui and W. E. McCarthy, "FSA: Applying AI Techniques to the Familiarization Phase of Financial Decision Making," IEEE Expert, vol. 2, no. 3, pp. 33–41, Fall 1987. [27] K. Umesa, "On-Line Underwriter," Best's Review, vol. 88, pp. 32–36, May 1987.

[26] G. Bruno, A. Elia, and P. Laface, "A Rule-Based System to Schedule Production," IEEE Computer, pp. 32–40, July 1986.

[27] C. J. Malmborg, M. H. Agee, G. R. Simons, and J. V. Choudhry, "A Prototype Expert System for Industrial Truck Type Selection," Industrial Engineering, vol. 19, pp. 58–64, Mar. 1987.

[28] H. Lemmon, "Comax: An Expert System for Cotton Crop Management," Science, vol. 233, pp. 29–33, July 1986.

[29] A. Sathi, T. E. Morton, and S. F. Roth, "Callisto: An Intelligent Project Management System," AI Magazine, vol. 7, no. 5, pp. 34–52, Winter 1986.

[30] D. Gentner and A. L. Stevens, (eds.) Mental Models. Hillsdale, NJ: Lawrence Erlbaum, 1983.

[31] A. P. Sage, "Behavioral and Organizational Considerations in the Design of Information Systems Planning and Decision Support," IEEE Trans. Syst., Man, Cybern., vol. 11, pp. 640–678, 1981. [34] R. W. Blanning, "Management Applications of Expert Systems," Information and Management, vol. 7, pp. 311–316, 1984.

[32] A. Basden, "On the Application of Expert Systems," Int. J. Man-Machine Studies, vol. 19, no. 5, pp. 461–477, Nov. 1983.

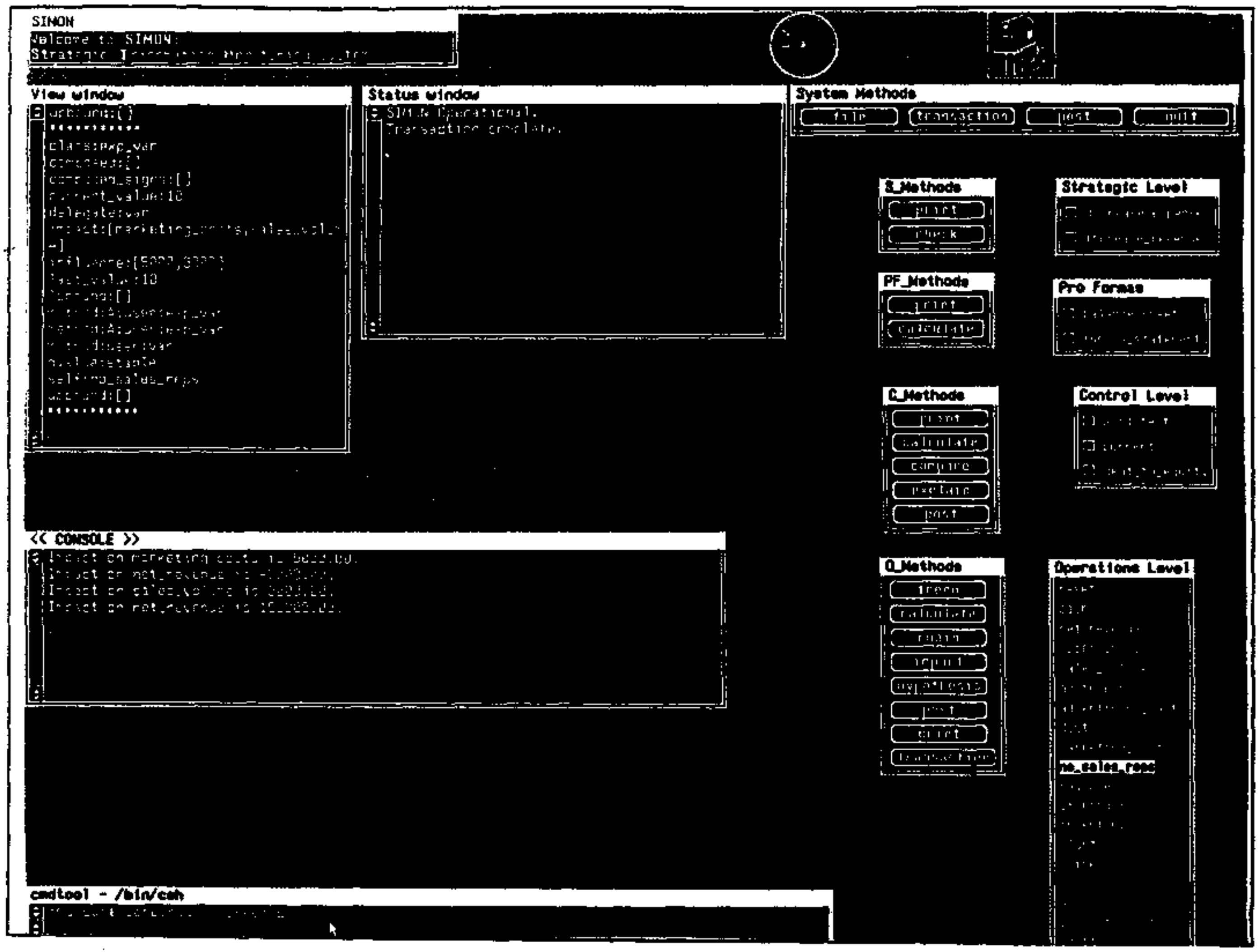

Fig. 8. Indirect impact (determined by SIMON) of adding one sales representative.

[33] B. Kosko, "Fuzzy Cognitive Maps." Int. J. Man-Machine Studies, vol. 24, no. 1, pp. 65–76, 1986.

[34] R. L. Ackoff and M. W. Sasieni, Fundamentals of Operations Research. New York: Wiley, 1968.

[35] G. M. Kasper, "The Effect of User-Developed DSS Applications on Forecasting Decision-Making Performance." J. Management Information Systems, vol. 2, no. 2, pp. 26–39, 1985.

[36] W. E. Pracht, "GISMO: A Visual Problem Structuring and Knowledge-Organization Tool." IEEE Trans. Syst., Man, Cybern., vol. 16, pp. 265–280, 1986.

[37] S. L. Loy, An Experimental Investigation of a Graphical Problem Structuring Aid and Nominal Group Technique for a Group Decision Support System. Unpublished Ph.D. Dissertation, Texas Tech University, 1986.

[38] J. F. Courtney, D. B. Paradice, and N. Ata Mohammed. "A Knowledge-Based DSS for Managerial Problem Diagnosis." Decision Sciences, vol. 18, no. 3, pp. 373–399, Summer 1987.

[39] N. Ata-Mohammed, J. F. Courtney, and D. B. Paradice, IEEE Trans. Syst., Man, Cybern., vol. 18, no. 6, pp. 899–907, 1988.

[40] D. B. Paradice and J. F. Courtney, "Causal and Non-causal Relationships and Dynamic Model Construction in a Managerial Advisory System." J. Management Information Systems, vol. 3, no. 4, pp. 39–53, 1987.

[41] D. B. Paradice and J. F. Courtney, "Controlling Bias in User Assertions in Expert Decision Support Systems for Problem Formulation." J. Management Information Systems, vol. 3, no. 1, pp. 52–64, 1986.

[42] W. E. Pracht and J. F. Courtney, "The Effects of an Interactive Graphics-Based DSS to Support Problem Structuring." Decision Sciences, pp. 598–621, 1988.

[43] R. Davis, H. E. Shrobe, W. Hamscher, K. Wickert, M. Shirley, and S. Polit, "Diagnosis Based on Description of Structure and Function." Proc. Int. Joint Conf. Artificial Intelligence, Menlo Park, CA: AAAI, 1982.

[44] R. Davis, "Reasoning from first principles in electronic troubleshooting," *Int. J. Man-Machine Studies*, vol. 19, pp. 403–423, 1983.

[45] P. E. Hart, "Direction of AI in the eighties," SIGART Newsletter, no. 79, pp. 11–16, 1982.

[46] D. Michie, "High road and low road programs," *AI Magazine*, vol. 3, no. 1, pp. 21–22, 1982.

[47] B. Chandrasekaran, "Towards a taxonomy of problem-solving types," *AI Magazine*, vol. 4, no. 1, pp. 9–17, 1983.

[48] B. Chandrasekaran and S. Mittal, "Deep versus Compiled Knowledge Approaches to Diagnostic Problem Solving." Int. J. Man-Machine Studies, vol. 19, pp. 425–436, 1983.

[49] B. Kuipers, "Commonsense Reasoning about Causality: Deriving behavior from structure," *Artificial Intell.*, vol. 24, no. 1-3, pp. 169–203, 1984.

[50] B. Kuipers, "Qualitative simulation," *Artificial Intell.*, vol. 29, pp. 289–338, 1986.

[51] S. M. Weiss, C. A. Kulikowski, S. Amarel, and S. Safir, "A model-based method for computer-aided medical decision making," *Artificial Intell.*, vol. 11, no. 2, pp. 145–172, 1978.

[52] Quintus. Quintus Prolog Release 2.4.2. Quintus Computer Systems, Inc., Mountain View, CA, 1988.

[53] C. Reiger and M. Grinberg, "The declarative representation and procedural simulation od?? causality in physical mechanisms," in *Proc. Fifth Int. Joint Conf. Artificial Intell.*, Cambridge, MA, August, 1977.

[54] J. de Kleer and J. S. Brown, "Mental Models of Physical Mechanisms." Cognitive Skills and Their Acquisition. J. R. Anderson, Ed. Hillsdale, NJ: Erlbaum, 1981.

[55] J. de Kleer and J. S. Brown, "Theories of causal ordering," *Artificial Intell.*, vol. 29, no. 1, pp. 33–61, 1986.

[56] K. Forbus, "Modeling motion with qualitative process theory," in *Proc. Nat. Conf. Artificial Intell.*, Menlo Park, CA: AAAI, 1982.

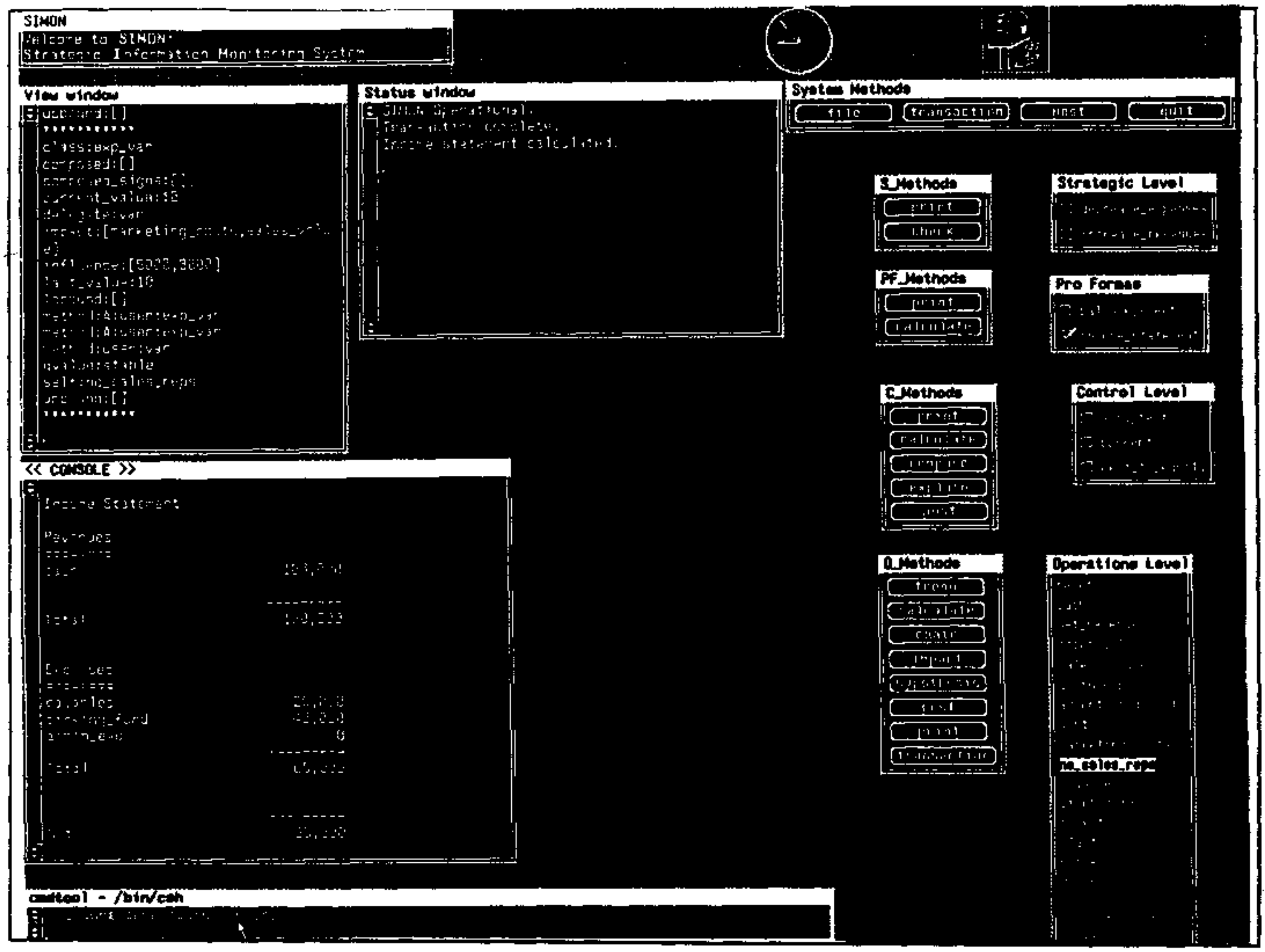

Fig. 9. Initial calculation of income statement from database items.

[57] P. E. Hart, A. Barzilay, and R. O. Duda, "Qualitative reasoning for financial assessments," *AI Mag.*, vol. 7, no. 1, pp. 62–68, 1986.

[58] Y. Iwasaki and H. A. Simon, "Causality in device behavior" and "Theories of causal ordering: Reply to de Kleer and Brown," *Artificial Intell.*, vol. 29, no. 1, pp. 3–32 and pp. 63–72, 1986.

[59] H. A. Simon, "On the definition of the causal relation," *J. Philosophy*, vol. 49, pp. 517–528, 1952.

[60] P. A. Samuelson, *Foundations of Economic Analysis*. Harvard University Press, Cambridge MA, 1947.

[61] D. B. Paradice, "The role of memory in intelligent information systems." *Proc. Twenty-First Annu. Hawaii Conf. System Sciences*, vol. 3, B. Konsynski, Ed., pp. 2–9, 1988.

David Paradice was born in Atlanta, GA in 1956. He received the B.S. degree in information and computer science and the M.S. degree in industrial management from Georgia Institute of Technology, Atlanta, GA, and the Ph.D. degree in business administration (MIS) 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 investigating means for applying artificial intelligence techniques in managerial domains and studying the impact of technology on society. His research includes investigating methods of dynamic model construction in managerial advisory systems and utilizing qualitative models of organizational systems, as well as investigating the influence of computer-based information systems on ethical decision-making processes. Prior to joining Texas A&M, he was a programmer analyst and consultant with Energy Management Associates, Atlanta, GA.

Dr. Paradice is a member of the American Association for Artificial Intelligence and The Institute of Management Scientists. Aspects of his research have appeared in the *Journal of MIS*, *Annals of Operations Research*, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, *Decision Sciences*, Information and Management, *Journal of Business Ethics*, *Information Resources Management Journal*, and *Journal of Education in Business*.

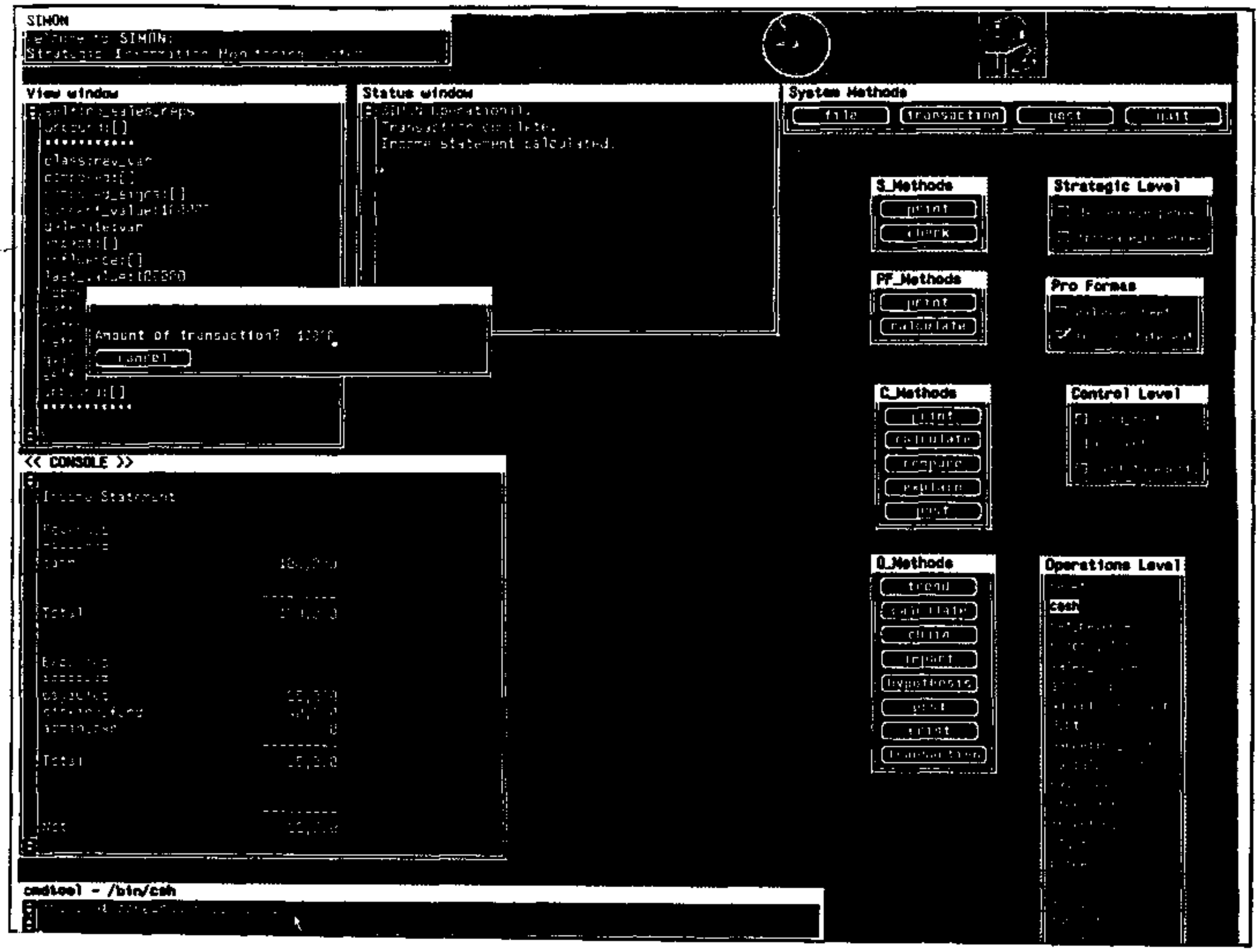

Fig. 10. Additional cash is added to demonstrate dynamic calculation of income statement.

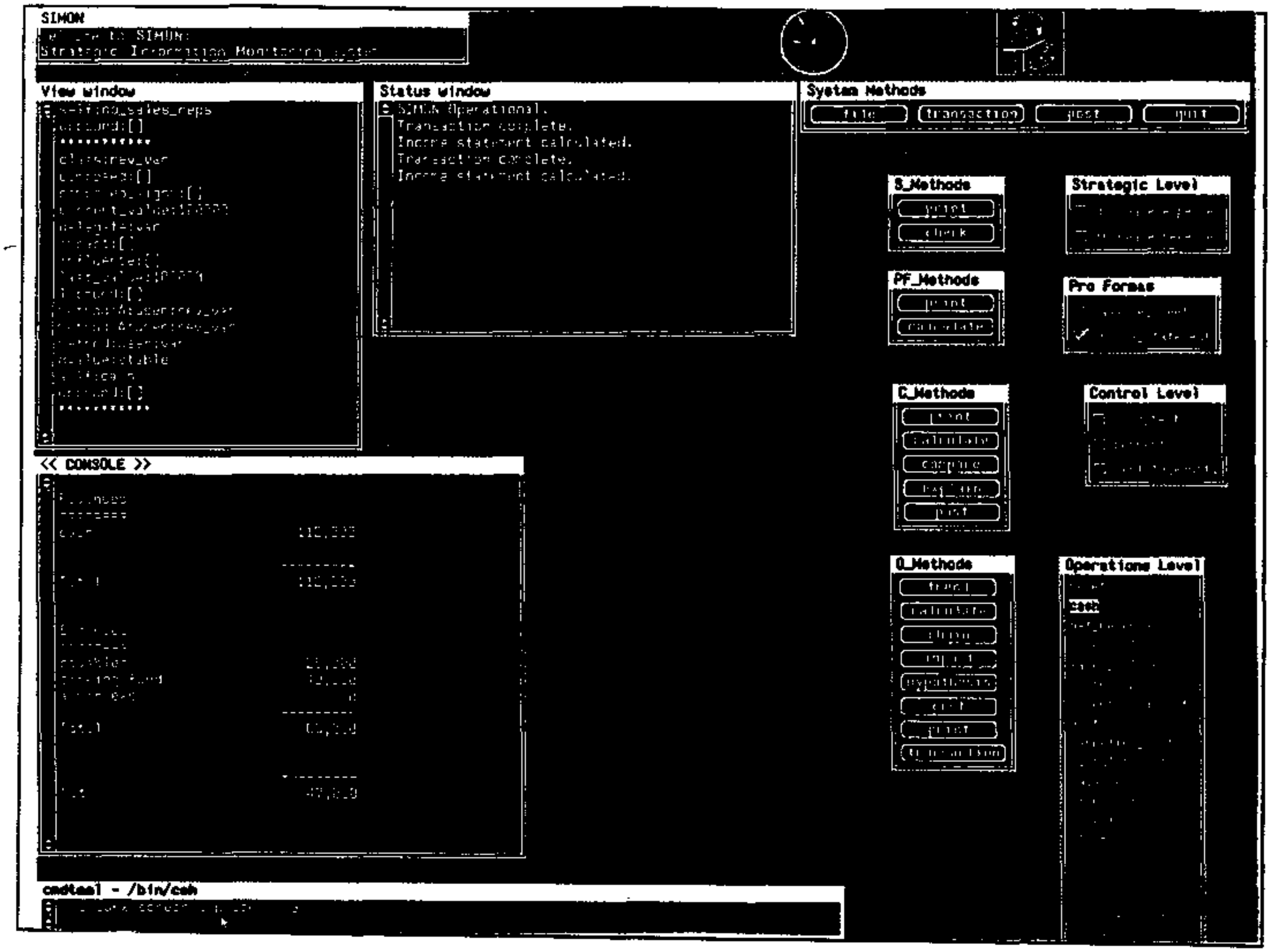

Fig. 11. Updated calculation of income statement from database items.