Open access • Journal Article • DOI:10.2307/249241

Effective design and use of computer decision models — Source link

William L. Fuerst, Merle P. Martin

Institutions: College of Business Administration

Published on: 01 Mar 1984 - Management Information Systems Quarterly (American Marketing Association)

Topics: Decision analysis, Evidential reasoning approach, Decision model and Verification and validation of computer simulation models

Effective Design And Use Of Computer Decision Models

By: William L. Fuerst
Merle P. Martin
Department of Business Analysis and Research
College of Business Administration
Texas A&M University
College Station, Texas 77843

Abstract

Computer decison models often provide useful results as management planning tools. However, these tools are frequently limited to firms with staffs of specialists who can assimilate the technical nature of the models. For other firms, the success of decision models such as simulation have not been demonstrated. This paper looks at recent literature regarding decision model deficiencies, evaluates selected financial simulation model packages, and suggests design needs for expanding the use of decision models to a broader range of firms.

Keywords: Decision models, MIS design, simulation
ACM Categories: D.3, H.1

Introduction

One of the main problems with today's computer decision models is that, for the most part, they are written in computer languages which only computer programmers can readily understand. Managers have no idea of the details of the computer program. In some organizations this deficiency is alleviated by the collection of a technical staff who, while not programmers, have enough technical knowledge to learn, use, and interpret complex models. However, managers of organizations without such technical experts are completely dependent on computer programmers, whether such programmers are employed by the user organization or by the firm marketing a packaged model.

The purpose of this article is to provide both designers and potential users of computer decision models with criteria for development and evaluation of those packages. This is done by presenting a review of the important user considerations which affect the success of model usage, by presenting an evaluation of selected financial simulation models, and by offering suggestions for possible enhancments that will serve to expand the use of computer decisions models.

Factors of Successful Decision Models

The user of a computer decision model is affected by several factors as presented in Figure 1. These factors determine the extent of use and relative success of the models. The problem environment includes considerations such as response time and accuracy required to make a decision. The environment of the organization dictates the acceptance of model results and the dedication of required resources to use the model effectively. As will be stressed in this article, the technical background and expertise of the user's staff affects the extent of use of a decision model, as does the user's experience with computerized models. Additionally, users of a decision model will be constrained by hardware considerations. For example, some organizations may use computers with a memory so limited that use of certain decision models may be impractical.

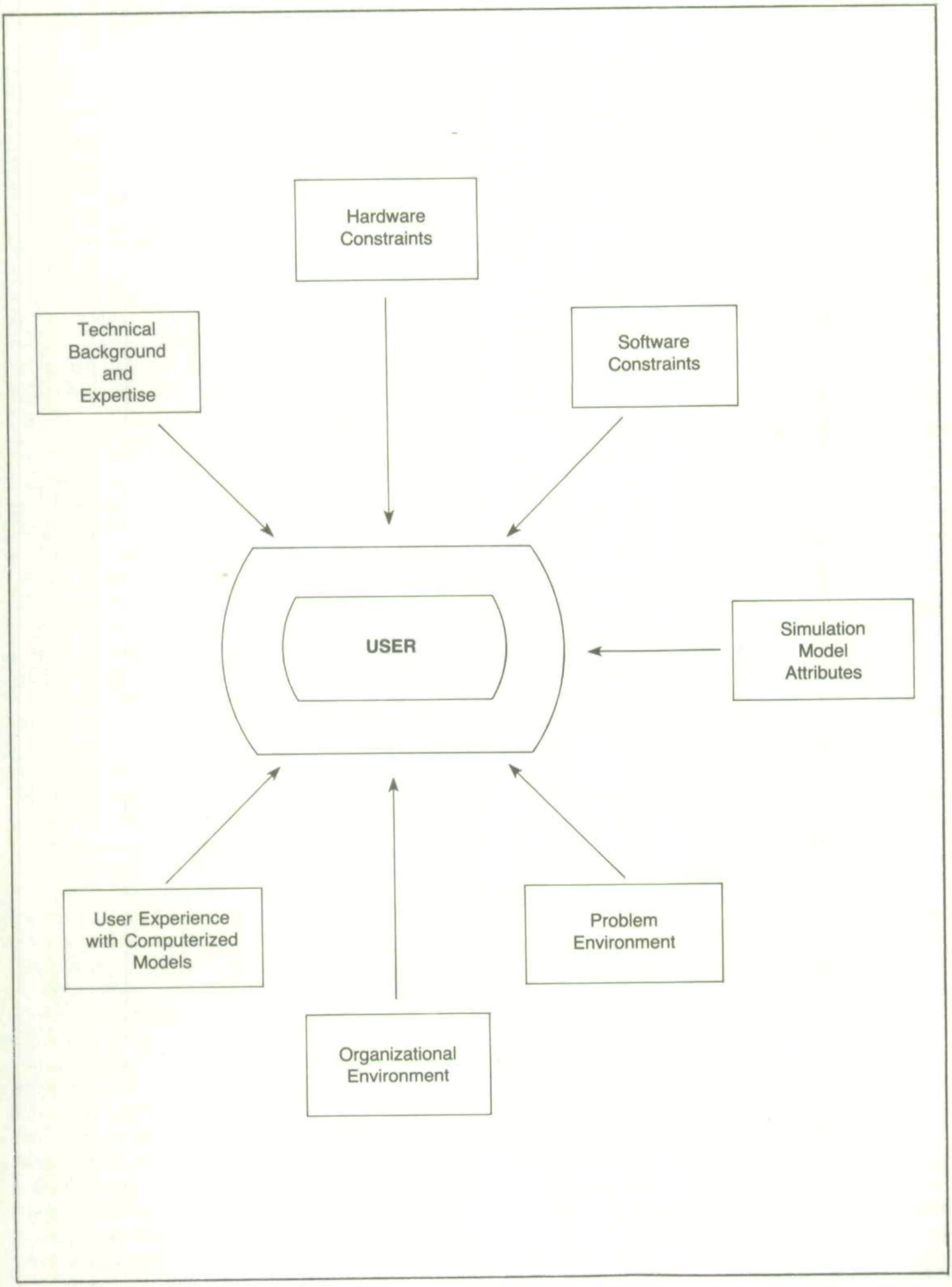
Hardware Constraints
Technical Background and Expertise
Software Constraints
USER
Simulation Model Attributes
User Experience with Computerized Models
Problem Environment
Organizational Environment

Figure 1. The User's Environment

The user may also be constrained by the language in which the decision model is programmed. Organizations that use microcomputers or minicomputers may not find it cost effective to support the language in which the model is written. Programs written in languages other than BASIC or perhaps PASCAL will probably preclude the use of such programs from most organizations that use only microcomputers.

Finally, the specific attributes of the decision model will have an effect on the extent and success of a user's involvement with that model. This is particularly true when the model's attributes conflict with other user factors such as technical background/expertise and the problem environment. Discussion of model attributes and their conflicts with user factors will be the primary focus of this article.

Problems of Computer Models

If computer modeling is to be more widely used as a decision support tool, then the modeling language must be useful to casual users of computer systems, rather than only to programmers or technicians. Without the assistance of technical staff, firms will face a modeling environment where managers are essentially the only users, and only when their time allows. Casual users may not be able or willing to recall complex details of a language.

For these casual users the system must provide support to help them write or use a computer model. Extensive error routines must be built into the model to guide the casual user in building that program [9]. The language must allow a programmer or management modeler to describe the system in terms familiar to the operation being modeled [8]. This creates a need for new, higher level computer languages [1, 25] that allow the computer system, through complex subroutines, to deal with much of the detail work. Such a system would free the designer to deal with the complexities of the system to be modeled rather than to concentrate on the details of coding. It would also allow the designer more time to ensure that the user is getting the information desired. Quite often, decision models are less useful because the solution presented to the user is a bewildering volume of statistics to be interpreted [6].

Another problem area concerns the user's confidence in the validity of the decision model. In many cases, this concern is justified; many complex models are inadequately validated because too much work is required due to the detail of the model and its output. An unverified model may not be an adequate representation of the firm's reality [4]. Lack of model validation, of communication between the user and designer, and of user understanding of the model results in a confidence trauma. The user's lack of confidence in the model means the model will not likely be used as a tool for solving problems or supporting decisions [19].

An accepted precept of systems design is that systems have a higher probability of succeeding if users are involved in development. User participation in model development ensures the opening and continuation of a communications channel which should lead to shared understanding. The developer of a model may not adequately understand the real world process in general, or the firm's process in particular. In turn, the user does not understand the nuances of the model and programmer logic and constraints. While participation is desirable, overcoming this communication problem is time-consuming and inhibits the use of models [19]. The problem is compounded by the use of packaged computer modeling systems where the user and the vendor programmer are geographically separated and do not share common organizational goals.

Evaluation of Simulation Packages

The problems highlighted in the previous section limit the use of decision model packages as decision making tools, particularly as the number of staff technicians decreases. In an attempt to illustrate these problems, several financial simulation packages were evaluated. The evaluation characteristics used in the analysis were aimed at the ease and effectiveness of use by decision makers and are supported in the literature as important criteria. Since the emphasis of this study was to evaluate *use* of decision model packages, cost was not included as a factor. Cost is a more pertinent measure in acquisition decisions and machine-time performance evaluations. The evaluation here consists of determining whether the following characteristics of the financial simulation packages were present.

- Interactive
- Conversational
- A natural language
- Nonprocedural
- Programmable by the user
- Modular
- Self-documenting
- Database oriented
- Graphics oriented
- Designed with user assistance

Evaluation criteria

Simulation packages should allow for human input during the processing cycle [3, 7, 10, 16], because more organizations, especially those with minicomputers and microcomputers, are using *interactive* computing. Interactive packages are not necessarily *conversational* [15, 22]. Frustrated users continually ask "why — why can't the computer do this, why can't it do that?" This is basically a communications problem caused by differences between computer capabilities and human skills. Computers have very fast processing capabilities, but must do so in a predetermined fashion. Humans, on the other hand, are slow processors but are capable of being creative and making decisions from incomplete information. Since humans often think and provide input to a decision making process in an unorganized manner, systems must allow for input instructions to be entered in an unstructured manner. The system must then translate those inputs into meaningful queries which the computer can process. Unfortunately, such systems are not well-developed to date, resulting in a large gap between man and machine. Although systems analysts, consultants, simulation specialists, and programmers function to close that gap, many organizations do not have such specialists permanently involved. Conversational languages allow the user and computer to communicate, and permit the user to fulfill the procedural requirements of the model without soliciting the specialized knowledge of analysts or programmers.

One of the newer techniques to help close the gap is the use of *natural languages*, which allow users to provide data and instructions in nonprocedural, English-like commands. Natural languages reflect the application rather than an artificial structure based on logical principles [10, 13]. These languages are further enhanced if they can preempt the computer's need to work in a predetermined fashion. This can be accomplished by allowing instructions to be entered in any order in a *nonprocedural* fashion [20]. A language translater then has the task of converting the nonprocedural statements into a correct procedural program for computer execution.

Design of a simulation language with the user and his/her skill limitations in mind would lead to a model that is *programmable by the user.* This allows packaged programs to be specifically tailored to the firm without the need for hiring expensive staff specialists.

To control complexity, *modular design* can be used to organize the system into a set of increasingly complex modules [14, 18, 23]. Languages can be used that are *self-documenting*; that is, they are comprehensive to the user without additional information [5, 11, 21]. This feature is a further enhancement of a simulation language, for it allows the user the opportunity to become well-versed in the language with a minimum amount of training time. Access to an *external database* is of value to some users [17]. *Graphical display* of model variables as they evolve allows the user an opportunity to see how a problem develops over time. Providing flexibility during execution to allow the user to interact with change in the model as displayed graphically is also an aid to building, verifying, and understanding that model [12]. Finally, a model that can be *designed with user assistance* enhances the communication between developer and user, and results in increased user confidence and understanding [19]. While the inclusion of such user assisted design can hardly be expected in the initial programming of commercial simulation packages, it can be provided through development of user alteration modules where report formats and process flow alternatives are selected by the user.

A look at selected packages

Financial packages were chosen in this evaluation because (1) they are most often the first type of "advanced" packages, beyond transaction processing packages, to be implemented, and (2) they are capable of cross-industry use. The six packages chosen, IFPS, FCS, Plan Master, RAL, Plancode, and IMPS were selected because they are commonly used in industry and are well-documented. Figure 2 presents evaluation results of these six financial packages concerning the criteria outlined in the preceding subsection. Although it is not desirable in this paper to discuss each package individually, it is possible to draw some general conclusions based on this evaluation.

All of the packages evaluated are interactive, but none is conversational; no system prompts the user for specific inputs. Users of these packages must possess prior knowledge of input procedures. The capability of submitting instructions in a nonprocedural fashion varies among the packages, but none support a natural language approach. However, of the packages examined, half allow programming by the users and provide self-documenting aids. Most packages provide for modular design to reduce complexity. Half of these have the capability of accessing an external database, whereas only one provides graphical displays. Finally, none of the packages allows the user to initially assist in, or to extensively modify, the design of the model.

Designing Computer Simulation Models

The preceding section should not be interpreted to imply that financial simulation packages are not effective. Indeed, one survey of users of financial simulation planning models indicated that such models were successfully implemented by middle managers [25]. However, this survey was conducted in large firms with extensive technical staffs possessing such titles as Financial Analyst, Planning Analyst, Project Engineer, and Cor-

Characteristics	IFPS	FCS	PLAN MASTER	RAL	IBM PLANCODE	IMPS
Interactive	yes	yes	yes	yes	yes	yes
Conversational	no	no	no	no	no	no
Natural Language	no	no	no	no	no	no
Nonprocedural	yes	yes	yes	no	no	no
Programmed by User	yes	yes	yes	no	no	no
Modular	yes	yes	yes	yes	no	no
Self-Documenting	yes	yes	yes	no	no	no
External Database	yes	yes	yes	no	no	no
Graphical I/O	yes	yes	yes	no	no	no
Designed with User Assistance	no	no	no	no	no	no

Figure 2. Characteristics of Selected Financial Packages

porate Researcher. If software packages such as simulation models are to be more widely used, as has been computer hardware, these packages must be designed to allow managers, rather than technical staff, to effectively develop and use the models required.

It is toward this end of allowing managers to develop and use models that the remainder of this article is directed. The next section presents the necessary characteristics for (1) enhanced use of simulation models in organizations with technical staffs and (2) increased use in other organizations where technical expertise is at a premium. It is recognized that development of such ideally user-oriented simulation models may be so costly today as to effectively exclude usage by the firms with predominantly casual users. However, given the potential size of the commerical market for such user-oriented models, it is anticipated that user costs for acquisition should decrease significantly over time.

Goal of Simulation Models

Figure 3 presents the user-oriented goal, criteria, and factors of simulation models. Statement and measurement of a goal are problems when dealing with the impact of a simulation model on making successful decisions. No optimal solution is given to the user as is the case with optimization models. Rather, inputs to the simulation model are manipulated to determine relative effects on the outcome, providing the manager with increased insight into, rather than a solution to, the problem at hand. Measurement of such increased insight is not well-defined. Thus, the most appropriate goal concerns the effective use of the model as a decision-making aid.

This goal can be broken down into three criteria: user understanding, ease of use, and model responsiveness. Each of these criteria can be further subdivided into specific factors of the goal of a user-oriented simulation model.

User understanding

The model must be understood by users who do not possess technical familiarity, skills, or staff. Modular program design has been shown to be beneficial. In addition, communication between the user and the model must be based on methods and techniques that best impart knowledge of complex systems. To achieve a true self-documenting system, the most effective media of documentation must be selected. Further experimental research must be conducted as to what is the most effective media. Care must be taken in research, and in practice, to differentiate between programmer and user documentation. Finally, it is generally accepted that user involvement in systems design increases understanding and confidence in the system being implemented. This can be developed by providing for a development module that allows the user to define, in his own terms, specialized process sequences, specific report formats, and user/model protocol. User involvement in the design process can be enhanced by bringing together the model developer and prospective model user to:

- give the developer a better understanding of the user's specific environment,
- allow the user to better understand and thus have more confidence in the model to be implemented, and
- jointly tailor the generalized simulation package to meet the specific requirements of the user.

For many organizations, providing such a development module is a practical alternative to a model that is programmable by the user.

Ease of use

The simulation model must also be easy to use. In many organizations, the user is likely to be technically unsophisticated. This simulation model must be interactive because the unsophisticated user must be put at ease when using and supplying data for the model. Thus, the language should be conversational, allowing for ongoing communication between the model and user. This communication should:

- provide continuous feedback so that a user knows that an input entry has had some effect,
- be consistent from one point of the model's process to the next,

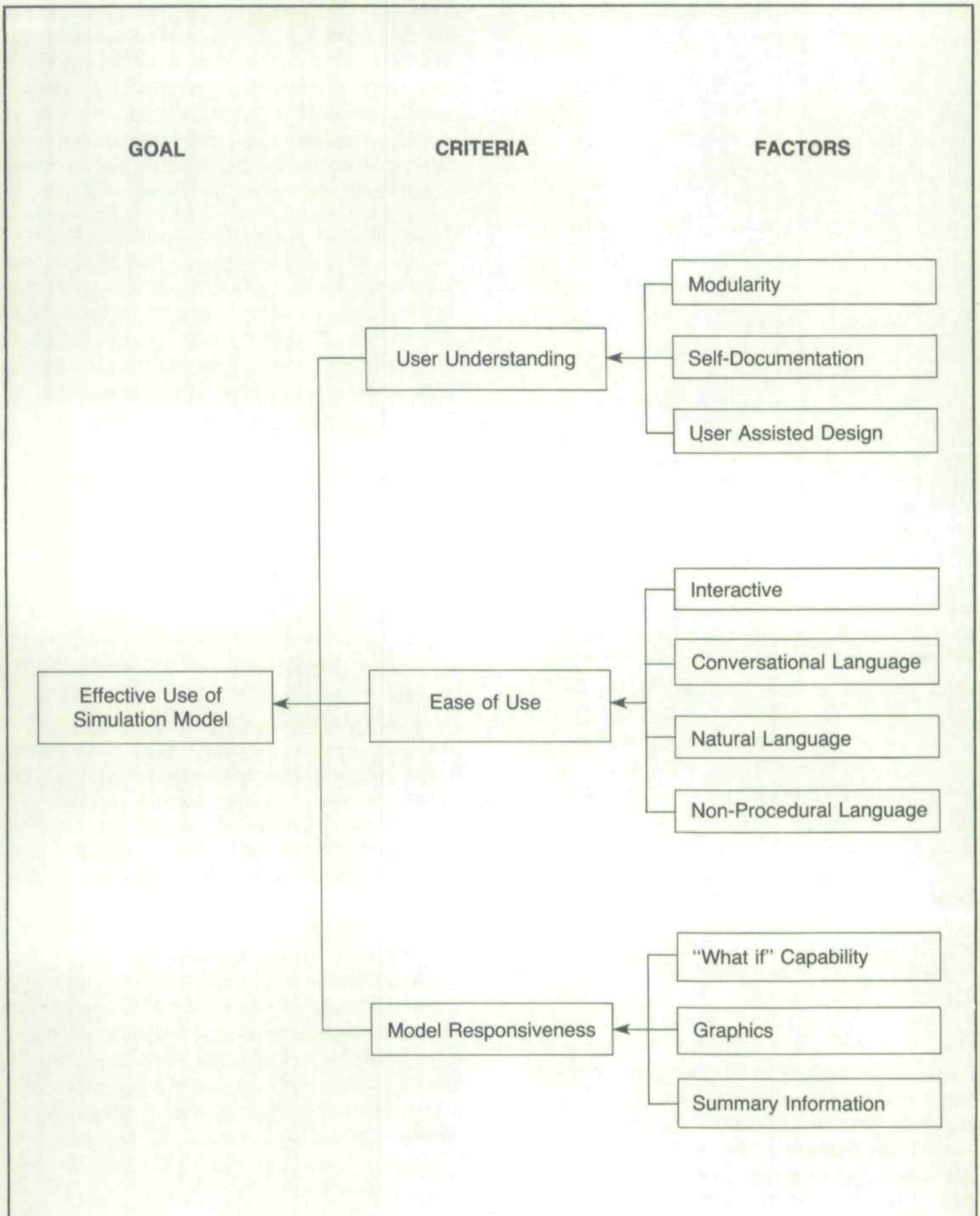

Figure 3. Effective Use of Simulation Models

- minimize requirements for generally unreliable human memory,
- be specifically designed, in terms of language and context, for that special audience of casual users,
- sustain user orientation to the extent that the user always knows at what stage the simulation model is operating [26].

Communication between the user and the simulation model must be biased toward the user. That communication should be natural (as close as possible to the user's language) and nonprocedural (entry of instruction in nonspecified sequence). The objective is to allow the user to communicate to the simulation model in a manner consistent with the user's natural thinking processes. The burden then shifts to the model programmer. This programmer must explain the user's language to the computer rather than forcing computer convenient languages on the user. This shift in responsibility does not preclude construction of the model with procedural programming languages such as FORTRAN, BASIC, SLAM or GPSS. However, the use of such languages must be transparent to the user. The user need not know nor care what language was used to construct the model that communicates with him in natural terms.

Model responsiveness

Obviously, a computer simulation model should be responsive to its user. However, the charge of this responsiveness changes for organizations without technical staffs. These organizations don't have persons trained in both solving and tolerating inscrutable and complex computer models. For simulation models, a "what if" capability is the essence of the models' usefulness. However, relatively unsophisticated users do not know that such a capability is possible. The model programmer cannot be content to wait on the user's request for sensitivity analysis, but must actively persuade the user to seek results to alternative scenarios. This may necessitate linkings to an external database whether or not such linking is requested by the unsophisticated user.

The same situation is true for graphics. The unsophisticated user may not be aware of the system's graphics capability and how graphics can help in representing data. For those users, a certain amount of graphics should be automatically provided. The increasingly sophisticated user should have the option to suppress automatically generated graphics.

Finally, there is the need for summary rather than detailed information. The content of summary information is a subject needing increased research efforts. Should there be medians rather than means? Should there be ranges rather than standard deviations? These are among a myraid of questions which need to be answered.

Summary

The use of computer decision models in general, and simulation models in particular, can provide managers with a valuable tool for decision making. Unfortunately, the use of these models has been limited, especially in organizations with small or no technical staffs. In order to expand the use of models there must be greater attention to developing designs that are truly non-technical and user-oriented. A combination of conversational, natural, and nonprocedural communication interfaces seems appropriate. In addition, the model must be fully understood and accepted by the user if it is to be effectively used. Research into media that contributes to such understanding and acceptance is essential. Other techniques such as modular and user-assisted design will certainly narrow the gap between the technologically designed computer decision model and the technologically uncertain managers. Finally, graphics, access to an external database, and summary statistics would be useful enhancements once language and understanding characteristics have been satisfied.

References

[1] Annino, J.S. and Russel, C. "The Ten Most Frequent Causes of Simulation Analysis Failure and How to Avoid Them," *Simula-*

tion, Volume 32, Number 6, June 1979, pp. 137-138.

[2] Ballard, B.W. and Bierman, A.W. "Programming in Natural Language; 'NLC' as a Prototype," *Proceedings of the 1979 Annual Conference of the Association for Computer Machinery*, New York, New York, 1979, pp. 228-237.

[3] Beguin, F.M. and Probst, A.R. "Interactive Simulation Models With User Guidance for Strategical Planning," *Proceedings of the Summer Computer Simulation Conference*, Society of Computer Simulation, Chicago, Illinois, July, 1977.

[4] Bekey, G.A. "Models and Reality: Some Reflections on the Art and Science of Simulation," *Simulation*, Volume 29, Number 5, November 1977, p. 161.

[5] Bowker, D. "An Interactive Aid to Financial Planning," *Journal of the Operational Research Society*, Volume 31, Number 10, October 1980, pp. 883-887.

[6] Bruggere, T.H. "Business Graphics: Making Computers Work For Management," *Proceedings IEEE Computer Software and Applications Conference*, 1978, p. 744.

[7] Card, S.K., Moron, T.P. and Newell, A. "The Keystroke-Level Model for User Performance Time With Interactive Systems," *Communications of the ACM*, Volume 23, Number 7, July 1980, pp. 396-410.

[8] Craig, G.D. "A Simulation System for Corporate Planning," *Long Range Planning*, Volume 13, Number 5, October 1980, pp. 43-56.

[9] Cuff, R.N. "On Casual Users," *International Journal of Man-Machine Studies*, Volume 12, Number 2, February 1980, pp. 63-187.

[10] *EDP Solutions*, Datapro Research Corporation, Delran, New Jersey, 1979, pp. E98-100, 146.

[11] Frankowski, E.N. and Franta, W.R. "A Process Oriented Simulation Model Specification and Documentation Language," *Software Practice and Experience*, Volume 10, Number 9, September 1980, pp. 721-742.

[12] Hurrion, R.O., Secker, R.J.R. "Visual Interactive Simulation: An Aid to Decision Making," *OMEGA*, Volume 6, Number 5, 1978, pp. 419-426.

[13] Kaplan, S.J. "On the Difference Between Natural Language and High Level Query Languages," *Proceedings of the 1978 Annual Conference of the Association for Computing Machinery*, New York, New York, 1978, pp. 27-38.

[14] Kiviat, P.J. "Requirements for an Interactive Modeling and Simulating System," in *Multi Access Computing*, Rosenthal, P.H. and Mish, R.K., ed., Hayden Book Co. Inc., Rochelle Park, New Jersey, 1974, pp. 53-61.

[15] Kupka, I. & Wilsing, N. *Conversational Languages*, John Wiley & Sons, New York, New York, 1980, pp. 98-107.

[16] Ledgard, H., Whiteside, J.A., Singer, A. and Seymour, W. "The Natural Language of Interactive Systems," *Communications of the ACM*, Volume 23, Number 10, October 1980, pp. 556-563.

[17] Lewis, J.M. "A Comparative Analysis of Financial Modeling Languages," *Proceedings of the 1978 Winter Simulation Conferences*, New York, New York, 1978, pp. 45-53.

[18] Linger, R.C. "The Management of Software Engineering Part III: Software Design Practices," *IBM Systems Journal*, Volume 19, Number 4, 1980, pp. 432-450.

[19] Lyneis, J.M. "Achieving Organizational Change Through Corporate Modeling," *Proceedings of the 1977 Summer Computer Simulation Conference*, La Jolla, California, 1977, pp. 596-601.

[20] Miller, L.A. "Natural Language Programming: Styles, Strategies, and Contrasts," *IBM Systems Journal*, Volume 20, Number 2, 1981, pp. 184-215.

[21] Oren, T.I. and Zeigler, B.P. "Concepts for Advanced Simulation Methodologies," *Simulation*, Volume 32, Number 3, March 1979, p. 69.

[22] Sheppard, S. "Control Issues in the Development of a Conversational Simulation Language," unpublished Ph.D. dissertation, University of Pittsburgh, Pittsburgh, Pennsylvania, 1977.

[23] Schneiderman, B. and Mayer, R. "Syntactic/Semantic Interactions in Programmer Behavior: A Model and Experimental Result," *International Journal of Computer and Information Sciences*, Volume 8, Number 3, June 1979, pp. 219-238.

[24] Wagner, G.R. "A Survey of Uses and Users of a Contemporary Planning System," EXECUCOM Systems Corporation, Austin, Texas, 1977.

[25] Winograd, T. "Beyond Programming Languages," *Communications of the ACM*, Volume 22, Number 7, July 1979, pp. 391-401.

[26] Simpson, H. "A Human-Factors Style Guide for Program Design," *Byte*, Volume 7, Number 4, April 1982, pp. 108-132.

About the Authors

***William L. Fuerst** is an Assistant Professor of Business Computing Science in the College of Business Administration at Texas A&M University. He received his D.B.A. from Texas Tech University in management information systems. His major areas of interest include design, implementation, and use of computerized systems, behavioral factors affecting MIS success, end-user computing, and computer simulation. He has consulted with both large and small organizations regarding a variety of computer-related matters. He is a member of AIDS and DPMA, and has published in* MIS Quarterly *and* Decision Sciences, *among others.*

***Merle P. Martin** is a Lecturer in the College of Business Administration at Texas A&M University. He received his MS from Stanford University in statistics. He has been Manager of Technical Operations for the Alaska Court System, Associate Professor of Quantitative Systems for the University of Alaska, Anchorage and a Lt. Colonel in the United States Air Force. His major areas of interest include design, implementation, and use of computerized systems, end-user computing and computer simulation. He is a member of DPMA and has published in the* Journal of Systems Management.