DSS SUCCESS: MEASUREMENT AND FACILITATION

Michael J. Ginzberg

CENTER FOR RESEARCH ON INFORMATION SYSTEMS
Computer Applications and Information Systems Area
Graduate School of Business Administration
New York University

Working Paper Series

CRIS #20

GBA #81-28(CR)

Presented to the NATO Advanced Study Institute on Data Base and Decision Support Systems, Estoril, Portugal, June 1-14, 1981.

Support for this paper was provided by the Center for Econometrics and Decision Sciences, College of Business Administration, University of Florida, Gainesville.

DSS Success: Measurement and Facilitation*

Michael J. Ginzberg
New York University
New York, NY USA

Decision Support Systems (DSS) represent an ever increasing portion of the investment in computer-based systems in organizations. Unlike earlier systems which aimed to replace existing clerical processes with faster, more efficient clerical processes, DSS attempt to extend and expand the capabilities of organizational decision makers. This fundamental difference in purpose between DSS and clerical systems causes our existing notions about system success to be inadequate. This paper explores the issue of DSS success, asking what it is, how it can be measured, and what can be done to facilitate it.

----

*Support for this paper was provided by the Center for Econometrics and Decision Sciences, College of Business Administration, University of Florida.

DSS Success: Measurement and Facilitation

I. Defining DSS Success

The continuing shift in emphasis in computer-based system development from clerical and transaction processing systems to decision support systems (DSS) raises many new issues for system designers, users, and researchers. Among those issues is the measurement of system success.

Transaction processing systems (TPS) aim to replace an existing clerical process with a more mechanized version of that process. They do nothing fundamentally new. As such, the measures of their success are straightforward:

1. Does the system accomplish the process it was designed to accomplish (i.e., the existing clerical task)?

2. Does the system accomplish the process with greater speed and accuracy than did the system (manual or automed) it replaced?

3. Does the system accomplish the process at lower cost (including both development and operating costs) than did the system it replaced?

Assessing success of this type of system is relatively easy because the prior system provides a benchmark against which the new system can be measured.

DSS seldom have an existing benchmark system against which their performance can be measured. Most DSS attempt to extend existing processes or to provide support for new processes [Ginzberg, 1978a; Keen and Morton, 1978]. Unlike TPS which aim at efficiency -- doing the same things better -- DSS aim at effectiveness -- doing better things. Thus, the ultimate aim of a DSS is to improve the quality of decision making, often by changing the decisions to which system users attend.

Unfortunately, measuring the quality of decisions is not an easy task. Some efforts to do so have been made in controlled environments such as laboratory experiments (e.g., Mock, [1971]) and business games (e.g., Marcotte, [1974]). However, even in those controlled environments, where objective measures of decision quality exist and there are no exogenous influences, measuring decision quality has proven to be quite difficult. In "real world" situations, where there are seldom objective measures of decision quality and numerous exogenous variables interact with the decision to determine ultimate outcomes, measuring decision quality is well nigh impossible.

As a result of these measurement problems, some surrogate for decision quality must be used to measure the success of a DSS. Three candidates have been suggested in the literature -- decision process (e.g., Stabell, [1974]), system usage (e.g., Lucas, [1978]; Wynne, [1977]), and user satisfaction (e.g., Ginzberg, [1978b]).

In theory, measuring the decision process as a surrogate for decision quality makes a great deal of sense. Models of decision making processes exist, and these models can be used to suggest characteristics of a "good" decision process -- e.g., the number of alternatives considered, the number of dimensions used to evaluate alterantives. These process characteristics can be measured both before and after the system is implemented to determine whether the system leads to an improvement in the decision process. There are two problems with this approach. Firstly, the link between characteristics of the decision process and quality of the resulting decision is not firmly established. Thus, it is not clear whether evaluating more alteratives, for example, will result in better decisions or not. Stabell [1977] has argued that much more research is needed in order to better understand the decision making process and the way it impacts decision quality. The second problem with this approach is that it is time consuming and potentially quite expensive. The decision maker must be tracked throughout the entire decision making process. Parts of the process might be captured by automatically logging his use of the DSS. But, observation and interviewing are likely to be necessary to capture other aspects of the process. Even then, it is not clear that the entire process can be captured, that there are not aspects of the process that neither can be verbalized by the decision maker nor leave a trace in the system log.

Both system usage and user satisfaction are more easily measured than is the decision process. Further, given the lack of certainty about the linkage between decision process and decision quality, a strong case can be made that usage and satisfaction are not inferior to process as surrogates for decision quality. Those who favor system usage as the measure of system sucess argue that unless the system is used, it cannot impact decisions. Hence, the degree of system usage is likely to correlate strongly with the degree of impact on the decision making process and ultimately with the quality of decisions made. They argue further that since most DSS are voluntary systems -- i.e., users have the option to use the system or not to -- usage is more likely to reflect the user's true evaluation of a system than is an expressed attitude (i.e., satisfaction).

Those who favor user satisfaction as the measure of DSS success argue that usage, even of voluntary systems, does not always reflect true satisfaction. Some DSS are used because they are the best system available to the user, although they do not, in fact, meet the user's needs. In at least one large organization, DSS usage was encouraged by eliminating other, competing sources of information. While this tactic certainly increased system usage, it did not increase satisfaction, nor is it clear that it resulted in better decisions. The argument to be made here is that usage will lead to improved decision making only if it is motivated by satisfaction with the system. Usage is still important, even critical, to ultimate system success; but, usage by itself is not enough.

(Insert Exhibit 1 about here.)

The entire question of whether usage or satisfaction is a better measure of DSS success would be irrelevant if it could be shown that these two measures tend to be highly correlated. The available data show this not to be the case. Exhibit 1 presents results from several research studies which have investigated usage of and satisfaction with DSS. It is quite apparent that the results are mixed. Whatever these two variables are measuring, they are not measuring the same thing. Ives and Olson [1981] point out that this is a common problem in attempts to correlate attitudes with behavior, and may result from poor specification of the attitude to be measured. Another possible explanation was suggested above.

Where does this leave us in our effort to define a measure of DSS success? We can eliminate both decision quality and decision process as having problems which are, at present, too large to overcome. Neither system usage nor user satisfasction alone is a completely adequate measure. Satisfaction without usage has been observed in a number of studies. It implies "users" who are satisfied with the system as long as they do not have to use it. Clearly, in such cases there is likely to be little impact on decision making. Usage, then, would appear to be the more important of the two measures, since without usage there can be no impact on decision quality. But, we must look at usage in the light of user satisfaction with the system. Usage without satisfaction may indicate a system which does not really meet user needs, will not have the desired impact on decision making, and should not be considered a success.

The remainder of this paper will focus on DSS usage -- what it means and how it can be measured and encouraged.

II. Measuring DSS Usage in Research and Practice

Before discussing how DSS usage can best be measured, it is appropriate to ask why we want to do so. Both researchers and practitioners have an interest in measuring DSS usage, and to some extent these interests overlap. The unique interests of these two groups, however, might suggest that they will be concerned with somewhat different measures.

Practitioners seem most concerned with assuring that the systems they develop are indeed used. They are likely to focus on questions such as:

- What types of systems are most readily accepted and used by decision makers?
- What charactersitics of the system interface are most important in assuring system acceptance and use?
- What development strategies and tactics most consistently lead to system use?
- How can use of an existing system be encouraged?

All of these questions focus on use as a generic, undifferentiated concept. A perhaps oversimplified statement of the view underlying these questions is that (1) some use will lead to acceptance of the system; (2) acceptance will lead to continued use; (3) continued use will lead to learning; and (4) learning will lead to good use, the kind of use which will improve decision making. Thus, there is no need to worry about the details of usage, just whether it is occurring or not. For this, gross measures of the amount of system usage are adequate.

Many researchers are concerned with questions similar to those posed above. But equally as many are concerned with other questions as well. For example:

- In what parts of the decision process (e.g., intelligence, design, choice) is the system used?
- In what ways is the system used at the various stages of decision making?
- How does the decision process change as a result of system use?
- How does system usage evolve over time?

Answering these types of questions requires a much more fine-grained measurement of usage than is the case for the questions posed earlier. It requires detailed measurement of the individual functions performed by the system, the sequences of function execution, and perhaps their timing in relation to events external to the system. Thus, in the discussion that follows, we should recognize that the needs of practitioners and of resarchers in measruing DSS usage will not always converge. Since the concern in this paper is measuring system success, and this is more directly the concern of practitioners than of researchers, our discussion will bend towards the practitioner's perspective.

Before system usage can be measured, it must be defined. The decision maker, the person whose actions are to be influenced by the DSS, can "use" the system in at least three fundamentally different wasy. The first is direct, "hands-on" usage. In this mode, the decision maker physically operates the terminal (or prepares input requests if the system does not operate on-line) and receives directly the outputs from the system. He (or she) assumes direct control of the system.

Many decision makers, however, use DSS through an intermediary. In this case, the decision maker does not take direct control of the system, but rather communicates with it through another individual. Both the inputs (or commands) to and ouputs from the DSS are screened by the intermediary.

In both hands-on and intermediated usage, there is physical use of the DSS. In the third type of usage, conceptual usage, there is <u>no</u> physical use of the DSS, either directly or through an intermediary. This type of usage occurs when the decision maker accepts the concepts embedded in the DSS and incorporates them into his decision making process, but does not use the computer-based system itself as an adjunct to that process.

(Insert Exhibit 2 about here)

Since our ultimate concern in measuring DSS usage is assessing the impact on decision making, all three types of use are relevant. However, they pose markedly different measurement problems. Hands-on usage is the most easily measured, and numerous possible measures exist. At the gross level (i.e., measuring only quantity of usage), these include terminal connect time, CPU time used, number of functions executed, and number of reports requested. At the micro level (i.e., measuring the type of usage), the number of different functions executed, the balance in usage across different functions, and the timing and sequence of function executions can all be measured.

While all of the variables suggested above could also be measured for systems used through an intermediary, the meaning of such measurements would be very different in the latter case. Our concern is understandng the impact on the decision maker, but in this case direct measures of physical system usage can only tell us about the intermediary. Thus, we need to measure what passes between the decision maker and the intermediary, the hands-on user. This could be the number (and nature) of requests for information made by the decision maker to the intermediary, or the number of reports based on system output received by the decision maker. Another possibility in this case is to measure the decision maker's perception of the degree to which he uses the system.

In the case of conceptual usage, there are no physical events or activities which can be measured as indicators of usage, so we can only rely on perceptual measures. One such measure is the decision maker's perception of the degree of influence the system -- or more likely, the process of exploring and developing the system -- has on his decision making process. Another is the degree to which the decision maker's conceptual structure changes to reflect the concepts embodied by the system (Stabell [1974] has used this type of measure in an attempt to understand hands-on usage at the micro level).

These various measures of DSS usage are summarized in Exhibit 2. The remainder of this section will examine some of the problems which arise in trying to apply these measures.

Usage Measurement Problems

As is indicated in the discussion above, both the general pattern of system usage (e.g., hands-on, intermediated, or conceptual) and the reason for usage measurement (e.g., assessment of overall success or research) suggest which measures of usage might be most appropriate. There are also a number of measurement problems which should be considered in choosing a usage measure. Generally, these problems are not specific to any one usage pattern or measurement purpose. Three problems to be considered here are (1) objective vs. subjective measures, (2) time period of measurement, and (3) single variable vs. multi variable measures.

It is often possible to assess usage with both objective (i.e., physical) and subjective (i.e., perceptual) measures. While these two types of measures normally are correlated, the correlation is far from perfect (See Exhibit 3); some users tend to overestimate their usage, while others underestimate it. Thus, if objective measures are available, they are in general preferable to subjective measures. This is especially true in the case of research which requires micro level usage measurement; reliable subjective measures of timing, balance, and sequence are likely to be especially difficult to obtain.

(Insert Exhibit 3 about here.)

DSS usage occurs over a period of time, and in most cases the pattern of use is not uniform. This raises two important questions. First, when should usage be measured? And second, over what period should it be measured? The question of when to measure usage must be answered in reference to the purpose of the measurement. Unless the purpose is to understand the dynamics of system adoption, it does not make sense to measure usage immediately after the system is installed. Normally, several months or longer are required before usage will reach an initial steady state.

(Insert Exhibit 4 about here.)

For most DSS users, the amount of usage is not uniform over time. Light users may use a system heavily over short periods, and heavy users may have periods of very little usage. Measuring usage over too short a period could produce a very unrealistic picture of the amount of system usage. Exhibit 4 presents data from a recent study [Ginzberg, 1981] which show how individual DSS users' usage levels varied from month to month. In order to minimize the impact of short term variations, a relatively long measurement period (5 months) was used in that study. Another possibility would be to sample usage for a number of short periods within a fairly long overall period. Obviously, the appropriate length of the measurement period will vary with the frequency and regularity of use of the DSS being studied.

(Insert Exhibit 5 about here.)

A final question in selecting a measure of DSS usage is whether to use a single variable measure, multiple single variable measures, or a multi-variable index. To answer this we must ask again what we are trying to measure. The answer is the degree to which the DSS is being used to support decision making. Unfortunately, we cannot measure this directly, but must measure instead the degree to which the DSS appears to be used to support decision making. But which is a better measure of this -- terminal connect time, CPU time, number of reports requested, or something else? That these variables are not the same and will not present identical pictures of usage is shown by the data in Exhibit 5. Different system users are likely to have different patterns of usage. Some wil1 spend little time at the terminal, but will execute many functions and produce many reports. Others will spend considerable terminal time reviewing only a few reports. Since it is seldom clear which of these patterns, if either, represents more usage, it is probably best to use a composite measure of usage (an index).

In summary, there are numerous possible indicators of DSS usage. The one or ones that are appropriate in a given situation depend on the general pattern of system usage and the purpose to which usage measures are to be put. Further, selection of usage measures must be tempered by an understanding of the measurement possibilities and problems which exist in each situation.

III. Encouraging DSS Usage

It was stated earlier that key concerns of many DSS practitioners are assuring that the DSS they build will be used and encouraging further use of those DSS that have already been implemented. This section will review those technical and organizational factors which appear to have an impact on DSS usage. Special attention will be paid to those factors which practitioners are most likely to be able to manipulate. Since the needs for hands-on use often differ from those for intermediated use, the discussion will indicate which type of use each factor is likely to encourage. Conceptual use wll not be discussed further in this paper, as this type of usage is seldom the goal of the system builder.

Technical Factors

The technical factors that can affect DSS usaage are the characteristics of the system itself, including both the model (system content) and the delivery vehicle (user interface). Many of these factors are derived from Little's [1970] statement of the requirements for a "decision calculus." The results of research on a number of the interface factors are discussed in Schneiderman [1980].

Seven characteristics of the model seem particularly important. These are (1) overall model quality, (2) simplicity, (3) robustness, (4) ease of control, (5) flexibility, (6) data availability, and (7) ease of communication.

Overall model quality concerns the technical accuracy of the system. Is it a good representation of what it purports to represent? Is it free from errors or bugs? A system which contains errors is likely to produce results which are obviously wrong. This will destroy user confidence in the system, and users will not use a system they do not trust. A model of high technical quality is of paramount importance whether usage is hands-on or intermediated.

Simplicity refers to the ease of model understandability. A simple model -- including relatively few variables, relationships, and phenomena -- will be more easily comprehended by the user than will a complex model. Since most managers are reluctant to base decisions on something they do not understand, the importance of this factor should be quite apparent. Care must be taken, however, to assure that the model is not too simple. This can result in obviously incorrect answers which will discourage further use. Model simplicity is more important for hands-on usage than for intermediated usage. In the latter case, a more complex model can be explained to the decision maker by an intermediary with substantial technical expertise.

Little [1970] describes a robust model as one which cannot be caused to give a silly answer. He cites the example of a model relating sales revenue to promotional expenditure. If this relationship is represented as being linear, then the model would suggest that the optimal promotional spending level is either zero or infinite. Since neither answer makes sense, the user would likely lose confidence in the model and stop using it. Like simplicity, robustness is most important in the case of hands-on usage. In intermediated usage, system outputs can be screened to assure that they are sensible before being passed on to the decision maker.

Ease of control refers to the degree to which the user can force the model to produce the results he wants by manipulating the inputs. Since DSS are meant to be an adjunct to managerial judgement, not a replacement for it, it is important that the system produce results consistent with the user's judgement. Counter-intuitive results are not likely to be believed, and will result in little use of the system. As with the previous two factors, ease of control is most important for hands-on use.

A flexible model is one which can be readily adapted to changing situations or problems. Flexibility is important because the problems a manager faces change. Further, his understanding of problems changes, and with this so must the problem representation embedded in the DSS. Without flexibility, a DSS will soon lose its usefulness because it will no longer correspond to reality. Flexibility is equally important in both hands-on and intermediated usage.

While no computer-based system is likely to be used if the data it is to operate on are unavailable, data availability takes on a special meaning in the case of DSS. Many decision problems that a manager deals with are short-lived. A DSS can be used to work on that problem only during its short lifetime. For problems with substantial data requirements, this often means that the data must exist in machine readable form, accessible to the DSS before the problem arises; if not, the problem will disappear before the data can be made available to the system. Data availability is equally critical to the usage of data-intensive DSS both in hands-on mode and through an intermediary.

The final system content characteristic, ease of communication, concerns the understandability of system inputs and outputs. Central to this is the use of operationally meaningful terms to express inputs and outputs. The use of terms which are consistent with those the manager normally uses in deaing with his environment provides a needed link between the model and the decisions it is designed to support. This link is most important for hands-on usage. In the case of intermediated usage, the intermediary can, if necessary, make the translation between the decision maker's language and the model's.

All seven system content characteristics described above can be manipulated by DSS builders. While, clearly, the importance of individual characteristics will vary from system to system, attention to these characteristics during system development should result in a high quality, usable model. Good system content by itself, however, is not enough. A high quality delivery mechanism or interface is also important. Six key characteristics of the interface are: (1) "fast" response, (2) "painless" input, (3) flexible control language, (4) meaningful error messages, (5) graphics output, and (6) terminal aesthetics.

System response time must be fast enough not to interfere with the user's progress in working on a problem. In hands-on use of an on-line system, the user expects the system to respond within a few seconds (though longer response times will be acceptable if the user believes he has asked the system to perform a complex task). Long waits for system resonse will force the user to interleave other activities with his use of the DSS, and may discourage hands-on use. Fast response is also important in intermediated usage if the decision maker is physically present while the system is being used. If, however, the intermediary uses the system in a location that is physically separated from the decision maker, response time is less critical.

The mode of input can have a serious impact on DSS usage. Many managers either cannot or do not like to type. Thus, usage of a DSS which requires a high volume of typed input may be limited. Light pens, menu selection, and structured dialog can be used to limit the amount of typing required. Typing is usally not a problem with systems used through an intermediary, since many intermediaries are proficient typists.

An unforgiving language for communicating with the DSS can frustrate the user and lead to his avoiding use of the system. A number of possible problems need to be considered in designing a control language. Novice or infrequent users need prompting, coaching, and explanations, both of what the system can do and of what inputs it requires. Frequent users, however, will quickly tire of long explanations and prompts from the system, and need only brief reminders from the system. On solution is to have multiple modes of interaction (e.g., Novice or Expert), and to allow the user to select the mode he wants. Another problem occurs because users make errors and change their minds. There should be some mechanism for gracefully backing out of a command sequence or changing a part of the sequence.

Still another problem occurs in systems which use key words. Users may not remember whether the proper command is RUN, EXECUTE, CALCULATE, or DO, and they need some mechanism to recognize synonyms or to aid their recall of the correct key word. A control language which handles these types of problems decreases the likelihood that users will become frustrated and abandon the system. While a flexible control language is valuable for any type of DSS usage, it is most important for hands-on usage. It should be noted that research evidence to date does not show "natural" language interfaces to be superior to well designed structured language interfaces. (e.g., Schneiderman [1980])

System error messages are another important attribute of the interface. While messages like IEBO392C or ERROR AT OCTAL LOCATION 07603 might make sense to a computer expert, they are of no help to a manager who is trying to use a DSS to solve a problem. Meaningful error messages are especially critical for systems used directly by decision makers.

Some discussions of DSS have stressed the need for graphics (as opposed to tabular) output. The research evidence in this area is far from conclusive; some studiess suggest that graphics is superior to tabular output, but others suggest the reverse. Perhaps the answer is that it depends largely on the cognitive style of the system user. This would suggest that flexibility in output format is the best way to assure DSS usage, as this would allow the user to choose the format most comfortable to him.

A final interface characteristic is terminal aesthetics, or how obtrusive is the terminal? This includes such factors as noise, glare, and physical size. Hands-on usage by a manager is most likely if the terminal is in his office, near his desk. Many managers object to having large, ugly, noisy terminals in their offices. Though this seems a rather pedestrian issue, it can impact the likelihood of system usage. Terminal aesthetics is considerably less important in cases of intermediated usage.

The impact these six interface characteristics will have on DSS usage is secondary to that of the model characteristics. That is, unless the model is adequate, usage is unlikely no matter how good the interface. However, a good model can go unused because of a poor interface. Clearly, a well designed interface is more important for hands-on usage than for usage through an intermediary. In the latter case, the greater the intermediary's computer expertise, the less important the interface becomes.

Organizational Factors

The technical factors tell only half the story about any DSS. These systems are used within some organizational context, and that context can have at least as great an impact on system usage as can the attributes of the system itself. The organizational factors tend to be broader and are less easy to neatly characterize than the technical factors. Four organizational factors which seem to be particularly important are (1) organizational support, (2) perceived value, (3) development climate, and (4) system fit.

Management support has long been recognized as an important factor in assuring system success (e.g., Garrity [1963]), and all of the research evidence shows it is equally important for DSS success. The concept of organizational support for a system goes beyond that of management support, and also helps to define in operational terms what constitutes adequate management support. Management's saying that it supports the use of a DSS is seldom enough. Adequate resources to support and promote the system must also be provided. One issue here is system availability: Are there enough terminals to meet the demands of users? Does the system receive a high enough priority to provide good response? In the case of intermediated usage, are there enough trained intermediaries? Training is another issue: Are mechanisms for formal and informal training provided, or are users simply given a manual and told to learn by themselves? Do training sessions show users how to use the system in the context of their jobs, or just how to operate the system? Perhaps the overriding issue is whether management makes it clear that they want users to use the system (e.g., by requiring system outputs as documentation for decisions) or not. A DSS which clearly has this type of support from management is likely to be used; one that does not is likely to be viewed as a distraction from work that must get done.

Perceived system value, though it has received much less attention than management support as a determinant of system success, is of equal importance. Perceived value has two components -- the perceived importance of the problem addressed by the DSS and the perceived usefulness of the DSS in addressing the problem. Importance is organizationally determined: Does the DSS deal with a problem or issue that people in the organization care about and pay attention to? Perceived usefulness of the DSS is partially determined by its technical characteristics, but equally important is the belief by members of the organization that previously existing mechanisms for dealing with the focal problem are inadequate. Organizational theorists refer to this as the existence of a "felt need." A DSS that is more effective than other available systems in dealing with an important problem will be seen as valuable and is very likely to be used.

Beyond management support, the one factor consistently suggested as a prescription for system success is user participation in the system development process. Alter [1978] has argued, however, that an equally important characteristic of the development process is initiation -- that is, was the project initiated by the user or not. His data indicate that both user initiation of and user participation in system development are related to the likelihood of system usage, though the relationship is far from perfect. More recently Ives and Olson [1981], in a comprehensive survey of the literature on user involvement, conclude that the impact of involvement on system usage is unclear; the split is roughly equal between studies showing a positive relationship and studies showing none.

The fourth organizational factor likely to affect DSS usage is sytem fit. System fit has two dimensions. The first is fit to the organization, what Schultz and Slevin [1975] have called "organizational validity." Organizations differ along many dimensions, including centralization, inter-unit integration, power distribution, norms of cooperation, etc. DSS, too, differ in the degree to which they fit with certain configurations of these variables. For example, a DSS which requires the close cooperation of marketing and production does not fit well with an organization in which these two departments are highly differentiated and do not have norms of cooperation. The likelihood of that system's being used in that organization is indeed small. The second dimension of system fit is fit to the individual user. Individuals have different problem solving styles (see McKenney and Keen, [1974]), and each style has its own preferred mode of gathering, manipulating, and presenting data relevant to a decision. Unless the DSS's capabilities are consistent with the user's decision style, the likelihood of his using it is small.

This discussion of organizational factors has been brief, but that should not be taken to mean that they are not important. The past decade has witnessed substantial research effort devoted to this area in an effort to understand why technically good systems were turning out to be failures in practice, i.e., were not used. Attention to the organizational factors is critical to DSS success, and these factors are of equal importance for both hands-on and intermediated usage. Like the technical factors, these organizational factors can be manipulated by DSS builders, though often not quite so easily or directly.

(Insert Exhibit 6 about here.)

IV. Summary and Conclusions

Exhibit 6 presents a summary of the technical and organizational factors which are likely to impact DSS usage. Many of the factors which will affect user satisfaction with the system are the same as those which affect usage. Indeed, one would expect to find some amount of feedback between usage and satisfaction: initial usage of a "good" system leads to satisfaction, which leads to further usage, continued satisfaction, etc. That one often finds low correlations between these two variables, even in the case of "good" systems, may largely be due to poor specification of the attitude being measured. For example, "general satisfaction" with a DSS should not necessarily be expected to correlate with usage at a particular time for a specific problem (Ajzen and Fishbein [1977] discuss the attitude-behavior correlation problem in general, and Ives and Olson [1981] provide an excellent discussion of this problem in the context of information systems). Perhaps the most we can say at this time is that "general satisfaction" should, over the long run, lead to greater usage of a DSS than "general dissatisfaction." Thus, we should be concerned with taking those actions necessary to promote satisfaciion, and along with it we should get usage.

One final comment seems worthwhile. The problems associated with measuring and encouraging success of DSS are essentially the same as those we encounter with other emerging types of computer-based systems in organizations, notably data-base systems, inquiry systems, and office automation. Thus, if we are able to master these problems in the context of DSS, we will have solved them for other system types as well.

REFERENCES

1. Ajzen, I. and M. Fishbein, "Attitude-Behavior Relations: A Theoretical Analysis and Review of Empirical Research," Psychological Bulletin, Vol. 84, No. 5 (1977), pp. 888 - 918.

2. Alter, S., "Development Patterns for Decision Support Sustems," MIS Quarterly, Vol. 2, No. 3 (1978), pp. 33 - 42.

3. Garrity, J.T., "Top Management and Computer Profits," Harvard Business Review, Vol. 41, No. 4 (1963), pp. 6 - 13.

4. Ginzberg, M.J., "Redesign of Managerial Tasks: A Requisite for Successful Decision Support Systems," MIS Quarterly, Vol. 2, No. 1 (1978a), pp. 39 - 52.

5. ______, "Finding an Adequate Measure of OR/MS Effectiveness," Interfaces, Vol. 8, No. 4 (1978b), pp. 59 - 62.

6. ______, "Early Diagnosis of Implemetation Failure: Promising Results and Unanswered Questions," Management Science (1981, forthcoming).

7. Ives, B. and M.H. Olson, "User Involvement in Information Systems: A Critical Review of the Empirical Literature," Working Paper No. 15, Center for Research on Information Systems, New York University, 1981.

8. Keen, P.G.W. and M.S. Scott Morton, Decision Support Systems: An Organizational Perspective, Reading, Mass.: Addison-Wesley, 1978.

9. Little, J.D.C., "Models and Managers: The Concept of A Decision Calculus," Management Science, Vol. 16, No. 8 (1970), pp. B466-B485.

10. Lucas, H.C., Jr., The Implementation of Computer-Based Models, New York: National Association of Accountants, 1976.

11. ______, "The Use of Interactive Information Storage and Retrieval System in Medical Research," Communications of the ACM, Vol. 21, No. 3 (1978), pp. 197 - 205.

12. Marcotte, A., unpublished Ph.D. dissertation, M.I.T., 1974.

13. McKenney, J.L. and P.G.W. Keen, "How Managers' Minds Work," Harvard Business Review, Vol. 52, No. 3 (1974), pp. 79 - 90.

14. Mock, T.J., "Concepts of Information Value and Accounting," The Accounting Review, Vol. 46, No. 4 (1971), pp. 765 - 778.

15. Schultz, R.L. and D.P. Slevin, "A Program of Research on Implemetation," in R.L. Schultz and D.P. Slevin (eds.), Implementing Operations Research/Management Science, New York: American Elsevier, (1975), pp. 31 - 52.

16. Schneiderman, B., Software Pschology: Human Factors in Computer and Information Systems, Cambridge, Mass.: Winthrop, 1980.

17. Stabell, C.B., "Individual Differences in Managerial Decision Making Processes: A Study of Conversational Computer Usage," unpublished Ph.D. dissertation, M.I.T., 1974.

18. ______, "On Defining and Improving Decision Making Effectiveness," Research Paper #287, Stanford University, 1977.

19. Wynne, B., "Measuring the Immeasurable or Credibility in the Public Sector," Interfaces, Vol. 8, No. 1 (1977), pp. 106 - 109.

DSS USAGE VS. SATISFACTION

Study	Correlation	Significance	N
Ginzberg [1981]	.30	p<.05	32
Lucas [1976]	NA	NS	39
Lucás [1978]	Negative for 3 of 8 usage measures	p<.05	58

NA = data not available

NS = not significant

Exhibit 1.

MEASURES OF DSS USAGE

Hands-on

- terminal connect time
- CPU time
- number of functions or reports requested
- balance, timing, and sequence of reports

Intermediated

- requests to intermediary
- reports received
- perceived use

Conceptual

- change in conceptual structure
- perceived influence

Exhibit 2.

ACTUAL VS. PERCEIVED USAGE CORRELATIONS

Study	Correlation	Significance	N
Ginzberg [1981]	0.48	p<.005	32
Lucas [1976]	0.61	p<.01	39

Exhibit 3.

MONTH TO MONTH USAGE FLUCTUATIONS

5-MONTH AVERAGE USAGE QUINTILE	AVERAGE MONTHLY USAGE QUINTILE				
	JULY	AUGUST	SEPTEMBER	OCTOBER	NOVEMBER
FIRST (n=7)	1.29 (1-3)*	1.29 (1-3)	1.57 (1-3)	1.14 (1-2)	1.29 (1-2)
SECOND (n=8)	2.00 (1-3)	2.25 (1-4)	1.75 (1-3)	2.50 (1-4)	2.13 (1-4)
THIRD (n=8)	2.75 (2-4)	3.13 (2-5)	2.88 (2-4)	2.63 (2-3)	3.38 (2-4)
FOURTH (n=8)	4.25 (3-5)	3.38 (2-5)	4.00 (3-5)	3.88 (3-5)	3.25 (2-4)
FIFTH (n=8)	4.38 (2-5)	4.50 (4-5)	4.88 (4-5)	4.88 (4-5)	5.00 (5)

*Range of quintiles covered by this group (5-month average quintile) during this month.

Exhibit 4.

USAGE MEASURE INTERCORRELATIONS*

	USAGE** INDEX	CONNECT TIME	TERMINAL SESSIONS
CONNECT TIME	.85	--	--
TERMINAL SESSIONS	.81	.44	--
FUNCTIONS EXECUTED	.94	.69	.78

*Pearson correlations; n=39 system users.

**Average of normalized values of connect time, terminal sessions, and functions executed.

Exhibit 5.

FACTORS IMPACTING DSS USAGE

	Hands-on	Intermediated
Technical Factors		
System content (model) characteristics:		
1) Overall quality	++	++
2) Simplicity	++	+
3) Robustness	++	+
4) Ease of control	++	+
5) Flexibility	++	++
6) Data availability	++	++
7) Ease of communication	++	+
Interface characteristics:		
1) "Fast" response	++	+
2) "Painless" input	+	0
3) Flexible control language	++	+
4) Meaningful error messages	++	+
5) Graphics output	?	?
6) Terminal aesthetics	+	0
Organizational Factors		
1) Organizational support	++	++
2) Perceived value	++	++
3) Development climate - involvement	+	+
4) System fit	++	++

? = unknown impact on usage
++= likely to have a strong impact on usage
+ = likely to have an impact on usage
0 = likely to have minimal impact on usage

Exhibit 6.