

LIBRARY  
OF THE  
MASSACHUSETTS INSTITUTE  
OF TECHNOLOGY







D28  
M414  
O. 439-70  
2

WORKING PAPER  
ALFRED P. SLOAN SCHOOL OF MANAGEMENT

STRATEGY FOR THE DESIGN AND EVALUATION  
OF AN INTERACTIVE DISPLAY SYSTEM  
FOR MANAGEMENT PLANNING\*

M. S. Scott Morton

439-70

MASSACHUSETTS  
INSTITUTE OF TECHNOLOGY  
50 MEMORIAL DRIVE  
CAMBRIDGE, MASSACHUSETTS 02139



JUL 20 1970

DEWEY LIBRARY

STRATEGY FOR THE DESIGN AND EVALUATION  
OF AN INTERACTIVE DISPLAY SYSTEM  
FOR MANAGEMENT PLANNING<sup>\*</sup>

M. S. Scott Morton

439-70

---

<sup>\*</sup>This paper was presented at the 1968 Conference at Carnegie-Mellon University on "New Directions in Management Information Systems."

RECEIVED  
AUG 21 1970  
M. I. LIBRARIES



Strategy for the Design and Evaluation  
of an Interactive Display System  
for Management Planning

Introduction

This paper discusses some experience with design, development and evaluation of an interactive display system used to support management planning. There has been discussion for some time on the potential that computer driven interactive displays may have for management planning but there have been very few experiments conducted thus far that have provided any real evidence. We have conducted lengthy experiments at Westinghouse and MIT in an attempt to identify the conditions under which interactive terminal systems are useful and to determine the nature and magnitude of their impact on the decision making process. One of the experiments involved building an interactive terminal system to provide decision-making support for line managers in the Westinghouse Electric Corporation. The managers involved use the system for support in dealing with a complex and significant monthly planning process. These experiments have been discussed elsewhere (6, 7, 9) and are only referred to briefly here. The purpose of this paper is to identify the strategy for development and evaluation of such systems and to comment on the software implementation problems.

The paper is in five parts, a general discussion of the role of terminals in a management setting, a presentation of a design strategy for such a system, some discussion of the evaluation process, software specifications for management decision systems and finally some conclusions



and comments on the future.

Before dealing with the basic topic there are some points that should be made with regard to the place of interactive terminals in the management sphere of activities.

The impact of computers on information systems and management decision making has varied widely. The conflicting views of professionals in the field, such people as Carroll (2), Dearden (4), and Diebold (5) can be traced in part to the different frameworks they use to describe management decision making. For useful discussion to take place it seems helpful to have an explicit framework in which one can discuss the issues. Since interactive terminal systems, like computers themselves, are no panacea for all managerial problems, one of the real issues in discussing them at this early stage in their development is to understand where their relative advantage lies. That is, in the full spectrum of managerial problems where are interactive terminal systems likely to be most useful?

For the purposes of this discussion the framework of managerial problem types developed in Figure 1 seems appropriate. This is a simple combination<sup>1</sup> of parts of the frameworks developed by Simon (12) and Anthony (1). In the cells of this framework are listed an assessment of the impact computers have had to date on management decision making. While this is a relatively arbitrary assignment of impact it is quite clear that to date computers have had virtually no direct impact on unstructured managerial decision making in a business environment. That is, the problem-

---

<sup>1</sup>This framework is discussed in more detail in (8).



Figure 1

Current Impact of Computers on Management Decision Making

Operational Control		Management Control	Strategic Planning
<p>High</p> <p>Accounting Payroll Accounts Receivable</p> <p>Inventory Control Bills of Material</p> <p>High</p>	<p>High</p> <p>Historical variance analysis and budget reporting - financial statements</p> <p>(Size is main variable--not many that are complexed and used)</p> <p>Low</p>	<p>High (where applicable)</p> <p>-Linear Programming -Dynamic Programming, etc. -Certain classes of simulation models -etc.  (Use is high where applicable (e.g. Oil Industry) but severe limitations on range of applicable problems) High (where applicable)</p>	
<p>Low</p> <p>-Interactive production scheduling</p> <p>-computer based information</p>	<p>Low</p> <p>-computer based information -cash flow forecasts -complex variance analysis</p> <p>Terminal costing (Research only)</p>	<p>Low</p> <p>-Warehouse location  (Research only)</p>	<p>1 2 1</p>





solving approach, the models used and the manager's involvement have not changed significantly because of the computers presence. Obviously batch processing computers generate data for the managers, and for solving structured problems this can be effective. Unfortunately many really significant problems are unstructured and these have not been changed much by computers. This stems in part from our lack of decent concepts and theories but in part from the fundamental characteristics of batch processing computers. Their inflexibility makes it too difficult to request new and different types or styles of information; their response time is normally hours and sometimes days; their structure encourages small, local and incompatible data bases and they normally have to be talked to through an intermediary programmer or systems analyst.

Interactive visual display systems<sup>2</sup> have the basic features (11) necessary to surmount these bottlenecks. Their remote location provides convenient access; their interactive features of light-pen or Rand Tablet permit the user to retrieve data or reports and to have access to a model bank or computational power; their on-line features permit fast response to questions on issues of strategy; as well as permitting data-bases to be maintained on-line, often with more global content, and in real-time in the few instances where this is necessary. All of these characteristics suggest that interactive terminal systems may be particularly useful in solving unstructured problems.

---

<sup>2</sup>The components of such a terminal system are discussed in (6) but are: graphical terminal, telephone connection to main computer, multiple access computer (not necessarily time-shared), and an on-line data-base for the problem being worked on (not necessarily maintained in real-time).



It should be stressed, perhaps, that the difference between "interactive terminals" and "interactive terminal systems" is most significant. The terminal unit is, quite obviously, a tool. It takes on significance when applied to a particular problem and driven by specific software giving it access to relevant models. Meaningful experiments then, seek to answer the question of when and under what conditions different system configurations are useful.

These assertions and the comments made in the balance of this paper are based on the experience with an experimental system at the Westinghouse Electric Corporation, where an interactive visual display system was used for solving an unstructured problem. This system has been in use with senior line managers for about two years as an aid to them in an unstructured planning problem. We have also run some experiments in another company with a terminal costing system (7) where the manager had to develop some cost implications, given his assumptions on several strategies, before settling on a pricing policy. It is quite clear from these experiences that batch processing computers have fundamental characteristics which makes it difficult to get them to provide much useful support to managers in these unstructured problem solving situations. Providing them with an improved quicker batch operation would not seem to change the situation significantly. The managers require something that is both conceptually and operationally different from the kinds of MIS support we have been used to providing in the past. If the problem is unstructured then this implies that the decision maker is vitally concerned with problem finding. That is, the decision maker's first problem is to browse through the data and identify the prob-





lems themselves. The second stage in this process, having found any given problem, is to identify and design a solution, and to do this in an environment where the criteria are hard to pin down and the data incomplete. In such an environment batch processing computers are of limited help but an interactive terminal system can provide a flexible enough interface between the decision maker and the computational power, model bank and data base that he requires to solve his unstructured problem. Our experience has shown that such systems can be built and used successfully to allow him to specify the data he is concerned with, the criteria and models he wishes to use at any given point for the particular problems that he feels he has.

All of the above implies that as technology changes, the design and use of management information systems should also change. Now that we have interactive visual display terminals we can construct information and decision systems that utilize the strengths of these systems. As has been pointed out such terminal-based information and decision systems are useful for quite different classes of decisions than the previous batch processing computer systems. Thus it is being asserted that batch processing computers have characteristics that make it most appropriate for them to support structured types of decision making. That is, situations where the problem is known, repetitive, and algorithms exist for its solution. On the other hand, interactive terminal systems turn out to be a relevant support tool when the decision maker has to deal with unstructured, ill-formed, problems.

To design, build and evaluate such systems requires a different approach and methodology than those appropriate for building batch processing support for structured decision making. To differentiate these systems from tradi-



tional MIS work, they are referred to in this paper as Management Decision Systems, MDS. Systems that provide managers with decision-making support. This paper is concerned with identifying the strategy involved in building such systems and commenting on the experience we have gained in the installations that have been implemented thus far.



### Design Strategy for a Management Decision System

By and large most management information systems developed thus far have tended to be functionally oriented. That is, they have concerned themselves with functional tasks that the organization has to deal with and have remained basically the same as new managers come and go. In contrast a Management Decision System (MDS) should be designed to support a particular decision, in other words the focus with a MDS is on a particular decision maker with a particular decision to handle. For example in the past, functional systems such as order-entry, production control, budgeting and cost accounting have been built by the company as general management information systems. However, in the case of a terminal system we are concerned with a particular production manager's problems with production control or the division manager's concern with the budget setting or performance evaluation process. It may very well be that other managers also have budget setting problems, but from the development and design standpoint, it seems to be important to focus on one manager and one set of decision problems at a time. The point of view of the systems designer, therefore, has to be that of providing support to the manager as the manager designs his system for his most critical problems.

With several years of systems design work implemented by the various designers active in the field of management information systems, it is clear that different levels and types of decisions require different types of systems. This is true both from a hardware and a software standpoint. However obvious this point may be now, it was not so obvious in the early days of systems design. The literature of only a year or two ago, would





describe an operational, or planned, production control system and then go on to draw analogies between the assistance this provided some low level inventory manager and the wonderful world in store for the president of the company. Until quite recently there was inadequate recognition of the need to provide different kinds of systems for different levels of decision making.

In a similar vein, MDS are not equally appropriate at all levels or for all problems. On the basis of our experiments to date, interactive visual display systems seem to be appropriate for problems with the following characteristics:

A. Large Data Base and/or Manipulation

If the data base is large or the manipulation required to derive signals from the data base is high, then a terminal system may be useful. Frequently with unstructured problems the data base is sufficiently large that it cannot be searched without interrupting the natural decision making process. Similarly the data may have to be passed through models in order to derive meaningful signals for the manager to interpret. The absolute level at which these aspects seriously impede the manager varies with the two factors, B and C, below, and the capacity of the manager himself.

B. Judgment

If the problem finding and problem solving processes require



"judgment" as to what constitutes a problem or a satisfactory strategy for a solution, then a terminal system is likely to be desirable. The visual terminal, and its interactive features can provide a flexible interface for the manager to input his judgments on values of the relevant parameters, as well as letting him use his own criteria on problem finding and his own notions of relevant data or manipulation.

C. Multidimensionality

If the problem has several dimensions along which the decision maker can measure performance and if it is not possible to specify a weighting function for these beforehand, then a terminal may be useful. That is, if the criterion function cannot be specified, then one has to have some form of flexible device with which to specify or develop weights for a solution. By permitting the user to look at his data along dimensions that seem relevant to him at problem solution time he can partially overcome this limitation.

In designing an interactive terminal system, then, it is useful to focus on a particular decision maker and a particular decision. Focusing on a decision problem, where the problem has the sorts of characteristics mentioned above, is the first step. We have no evidence to suggest that this decision problem cannot be in any functional field and it will almost certainly rely upon the usual functional models that have been developed





in each of the areas. The particular models or algorithms that the manager needs for problem solution come from the usual operations research base. The terminal system does permit more sophisticated models to be used as the managers, in our experiments at least, indicated considerable interest in using adequate models. In addition such fields as Bayesian decision theory begin to look operationally practical as the interface provided by the display is powerful enough to collect the managers' judgment and portray the results in a form understandable to him.

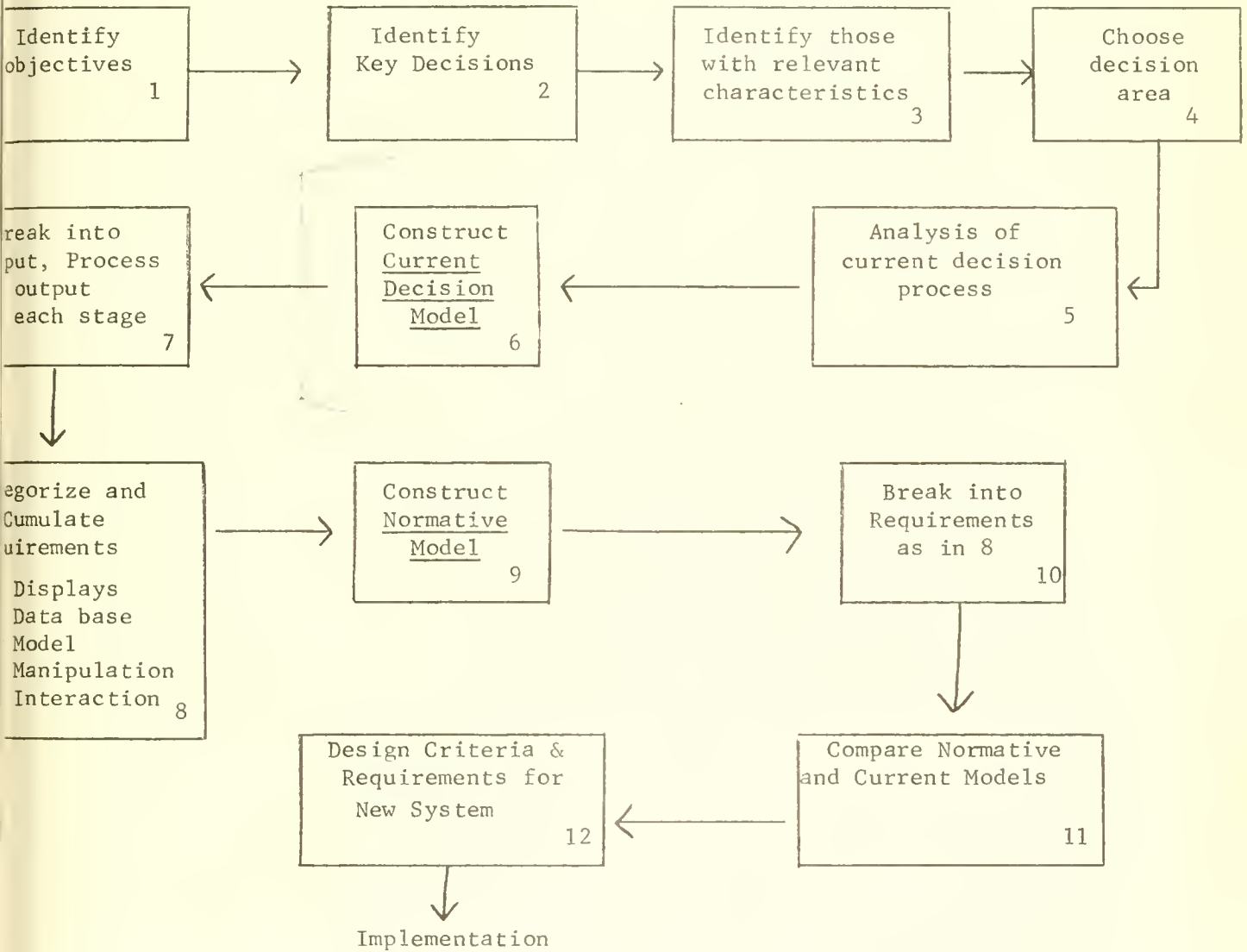
There seem to be five major steps necessary in developing an interactive terminal system. These five steps are certainly not unique nor indeed are they the only way of attacking the design and development problem. However, they have been found useful in the experiments thus far, and indeed they are really a reflection of the basic problem-solving process itself, whether this process is being applied to specific problems in the real world, or to analyzing those problems.

1. With any given manager, the first step (see Figure 2) is to identify the key decisions that he is making or should be making. This should involve an explicit identification of job objectives as the manager perceives them. In practical terms, it is also necessary for him to identify that particular problem about which he is most concerned. If the problem is significant and time consuming then there is a better chance that he will be willing to devote the time necessary to help develop a system to satisfy the problem. Developing terminal-based system is virtually



Figure 2

Design Process for a Management Decision System





impossible without the active support of the manager involved. The goal after all is an active support system for the manager's decision-making process.

2. The second stage is to analyze the decision-making process that the manager or managers are using presently. This involves taking protocols of the subject as he engages in the process, and generally probing and analyzing as deeply as possible with a view to developing a model of the process the decision-maker is using. The goal at this point is to develop an empirical model of the decision process. It is frequently useful to compare this with the general purpose models developed by the professionals in the field, such as Simon, with a view to clarifying the steps involved. The construction of this decision model was not a trivial task in the experiments we have run. For example it took three or four observations of the complete decision-making cycle before the major structure became clear.
3. Having developed the general model, the next step is to take each component or step of the process and look at it in detail. This involves identifying the inputs that are necessary for that step in the decision process and the computation that must take place to transform that input into information that is useful for the next step in the process. That is, each step has to be split into the input required, the processing required and the output that is required for



the following step.

From this detailed step by step analysis it is possible to develop the requirements for the data base, the nature and types of the models required and an identification of the form of manipulation that is necessary. Similarly, this step can be used to analyze the type of information display (graphical, tabular, etc.) that is relevant for the manager at each stage of the decision-making process.

4. Doing this analysis for all the steps in the decision making process generates a list of the various requirements involved in the current decision making process. The next step is to turn this from an empirical to a normative model. That is, given the objectives of the decision maker in this particular process, it is possible to ascertain from a normative standpoint the kinds of information processing and manipulation that the decision maker should require.

This normative model should be mapped out and its requirements identified in the same manner as the empirical model. When this stage is complete the requirements involved can be contrasted with the empirical model. The requirements can then be matched up with the capabilities of the decision system used by the decision maker at present. This acts as a check to make sure that the characteristics of an interactive terminal system are, in fact, necessary for this particular decision making process. It may very well be that





by restructuring the process or moving toward the normative model, an interactive system is not necessary. Geoffrey Clarkson's (3) experience with the trust officer identified that there were some algorithms that were useful for part of the process. Similarly, it may be that the analysis discussed above will identify the fact that there are some algorithms that take care of a large part of the process and the balance of the decision making task can be handled perfectly well by the decision maker without the aid of an interactive system. However, it may also be clear at this point that the kinds of computation involved, the requirements for a global data base and so forth, all imply a capability more than that possessed by man alone. If this is the case then the next step is the implementation of the terminal system.

5. The implementation process is simply one of providing, via the terminal system, the data base, computational power and interactive software support suggested in the normative model. This implementation process has been described in some detail (6) elsewhere and is not repeated here.

What is being suggested is that the best strategy in designing an interactive terminal system is to take an incremental approach, focus on a decision maker and within the problem set with which he deals, find a significant, complex, unstructured problem and develop a decision support system for him to use in that problem solving area. Having taken care of



that particular problem through the process described above, the hope is that the manager has some freedom to develop other areas and adapt the technology to those. Similarly, the systems and software development that has gone on in solving the first problem, can then be applied to other problems with the same decision maker or other decision makers. There is certainly a risk that the software developed for one person has become so specific that it is not usable in other situations but the experience described above with the experiments we have conducted so far indicates that the basic decision making process is reasonably constant between decisions and decision makers and therefore the software architecture is valid across problems and people. This point is developed further below but certainly more evidence will be required on this point before firm conclusions can be drawn.



### Evaluation Strategy

One lesson that must be learned from past experience with management information systems is that we have to be particularly careful and consistent in the evaluation process. We have to be concerned not only with cost effectiveness but with establishing what we have learned from each step in the decision system development. If we want to move incrementally towards improved systems, we have to take special care to evaluate experience at each step of the process. Clearly a straight statement that such interactive terminal systems led to better decision making would be the simplest form of evaluation. However, this immediately leads to the question as to what constitutes "better." In an effort to avoid this, a more useful strategy seems to be to identify the impact the management decision system has had on the decision making process.

In the experiments we have conducted thus far, the process has first been modeled, then the interactive terminal system has been introduced, and then the decision making process modeled again. Building an empirical model of the decision process with the managers using the MDS permits a comparison to be made between this model, the original empirical model of previous decision process, and the normative model developed in the design stage. This three way comparison process provides a basic tool for both evaluation and learning. At the points where the models are different, one can concentrate on developing some cost effectiveness data. This might involve the time that the manager spent in the decision making cycle, perhaps the inventory levels before and after the use of the system or whatever other quantitative measures are relevant to the particular decision process.



Where it is not possible to get quantitative data, one can force the manager to look at the differences in the process as shown by the models and evaluate these on a subjective basis. That is, the comparison identifies those aspects of the decision process that are different and the manager himself can evaluate the specific differences in light of his estimate of the savings involved.

For example, in the Westinghouse experiment the managers developed only one solution under the old process. That is, the first time they arrived at a solution which was satisfactory to all concerned, that marked the end of the decision making cycle. Because the decision cycle time has been sharply reduced with the use of the MDS, and computational power is available, the managers involved now develop a series of solutions. They then look at each of these explicitly and try to decide which of the several alternatives is the best. They are not using optimizing techniques and in fact, at this stage there is no certainty that they have, in fact, picked the best solution. However, in their view it is clearly better to be consciously evaluating three or four alternatives than operating in a totally "satisficing" fashion by selecting the first solution that meets the minimum requirements.

Similarly, by contrasting the model of the decision making process after the interactive terminal system has been introduced, with the normative model developed as the guide during the design phases, it is possible to identify the gaps in the implemented solution. By the same token the implementation process has identified some shortcomings in the original design. This iterative comparison allows the development of the design





model for the next stage. This evaluation process is neither perfect nor foolproof. However, by going through the process, it does exercise some discipline and forces both the designer and the manager to recognize the evaluation process.

To be more specific about this evaluation process, consider a particular example. In the Westinghouse experiment, the pre-terminal decision making process involved three phases. In summary form these can be thought of as Simon's "intelligence, design and choice." The decision makers proceeded sequentially through these phases, first of all endeavoring to find all the problems they thought they should deal with. Then, with this list, endeavoring to design particular solutions to each of these problems in turn, and finally, seeing if they were able to come up with a particular choice from the designed solutions that met their minimum criteria. Each step was quite separate and spread linearly over time. With the MDS, the managers operated in a quite different fashion. The intelligence design and choice phases merged into one and they would oscillate rapidly among the three sub-phases. We had not designed displays that made it particularly easy to look at this data in combined form on one display. Therefore they had to do some pointing with the light-pen and wait for seven seconds. If this is done ten times in a row it is easy to become frustrated. By redesigning that portion of the system, adding a module to permit a new form of display, we were able to bring the system into line with their new decision-making process. Similarly the managers when confronted with the evidence of their actual steps in solving the problem began to verbalize some of their criteria which, when formalized into decision



rules, reduced the necessity for some of the oscillation. Both of these changes involved specific evaluation of the pre and post terminal, empirical and normative, models.

There is one final evaluation test which has not been available to the designers and implementors of the traditional management information systems. By focusing on a particular decision maker and a particular decision, it is easy to establish whether or not he uses the system. In the final analysis, one always has the acid test of actual management use. If he is an independent line manager, and has a particular problem to solve, then one can probably conclude that if he uses it, he is finding it useful and in that sense the system has been a success!



### Software Specifications

Successful use of a MDS is heavily dependent on adequate software. This involves matching the software characteristics to the problem requirements as well as having a sound software architectural scheme. This latter issue is fairly technical and has been dealt with elsewhere (10), the former however is of more general interest and is discussed below.

The normative model of the decision process suggested that certain computational power and models were necessary for the decision maker to solve the problem he was faced with. Similarly the problem finding process may require an interactive browsing capability if it is to be effective with a large data-base and an unstructured problem. A graphical<sup>1</sup> visual display terminal can provide a flexible interface between this data-base, computational power, models and the decision maker. However, this interface can only be provided by good software support, to permit the interaction necessary. Obviously the terminal itself is merely a piece of hardware, and as such cannot provide any support, success or failure in its use for management decision making depends very largely on the design and implementation of the software. Clearly the design of such software is reasonably complex and a lack of comparable experience in other fields does not make this process any simpler.

To be successful in the problem environment discussed earlier, the software for a management terminal system should meet the following goals:

---

<sup>1</sup>The graphical feature is absolutely essential in a management setting if large quantities of information are to be assimilated efficiently. This point is argued at some length in (6).



- (a) simplicity of use
- (b) ease of interaction
- (c) general purpose characteristics
- (d) modular in design
- (e) fail-safe capability
- (f) hardware independence

A. Simplicity of use

To be effective a management decision system has to be simple enough so that a busy manager with no computer knowledge is able to use it comfortably. It also has to be a powerful system--where power implies many machine functions implemented for one user action. In previous interactive (typewriter) systems, there has always been an attempt to strive for the many/one relationship, frequently very successfully (such as the MIT-OPS system). However, this attempt has usually been in the context of compilers, where the many/one translations did not take place until the user had created some unambiguous source language coding. The system in the present instance has to respond to general purpose commands with specific instructions to create a particular display.

For the manager to find the system easy to use, it has to permit communication in the natural language of the manager, and must have aids to provide him with help and instruction. ✓

This was achieved with some success in the experimental systems as the part visible to the decision makers was worded in their terms. That is the terminology, manipulation and structure of the system, as they saw it, was developed from their former processes with their help. They reacted to initial designs and worked with the systems designers so that





they understood what was involved. The underlying architecture, of course, was designed to match the model of the decision-making process.

B. Ease of interaction

It was felt that the system had to respond readily to user requests. This involves both adequate response time from the system, and a meaningful response. Adequate response time is, of course, situationally dependent. That is, users expect the system's response time to be a function of the complexity of the task to be done. Unfortunately, apparently trivial tasks from the user's standpoint can take a large amount of machine time, and vice versa. In addition to this problem, the system must allow a wide range of action to the user. In fact, the range of action ought to be broad enough to encompass all useful procedures he can take in his particular problem solving process.

The graphical capability is of considerable help here. The system is aware of what is on the screen so that by merely pointing to an item with the light-pen or Rand Tablet the user conveys a great deal of information to the system. The software is based on a table driven assembler so that by pointing to items with the light pen entries are made in the assembler specifications table. When the user is ready for execution he hits "proceed" and the system monitor takes over and executes the necessary routines to build the display or responses required by the user. For example, on Exhibit 1 if the user were to hit, Cumulative Graph, Tumblers, Seasonal, Jan, 1967, September, 1967 he would get the display seen in Exhibit 2.

Similarly on Exhibit 2 if the user wanted to see the implications for inventory of increasing sales by 15% in June and July, he could iden-



Exhibit 1

GRAPH SPECIFICATION

<u>Presentation</u>	<u>Models</u>	<u>Axis</u>	<u>Time Span</u>	
			<u>From</u>	<u>To</u>
Cumulative Graph	Washers	Seasonal	Jan	Jan
Non-Cumulative Graph	Tumblers	Normal	Feb	Feb
Bar Chart	Agitators	Units	Mar	Mar
Tables	LAH-1	Dollars	Apr	1965
-	LAH-2	-	May	66
-	-	-	Jun	67
-	-	-	Jul	68
-	-	-	Aug	69
			Sep	70
			Oct	71
			Nov	72
			Dec	

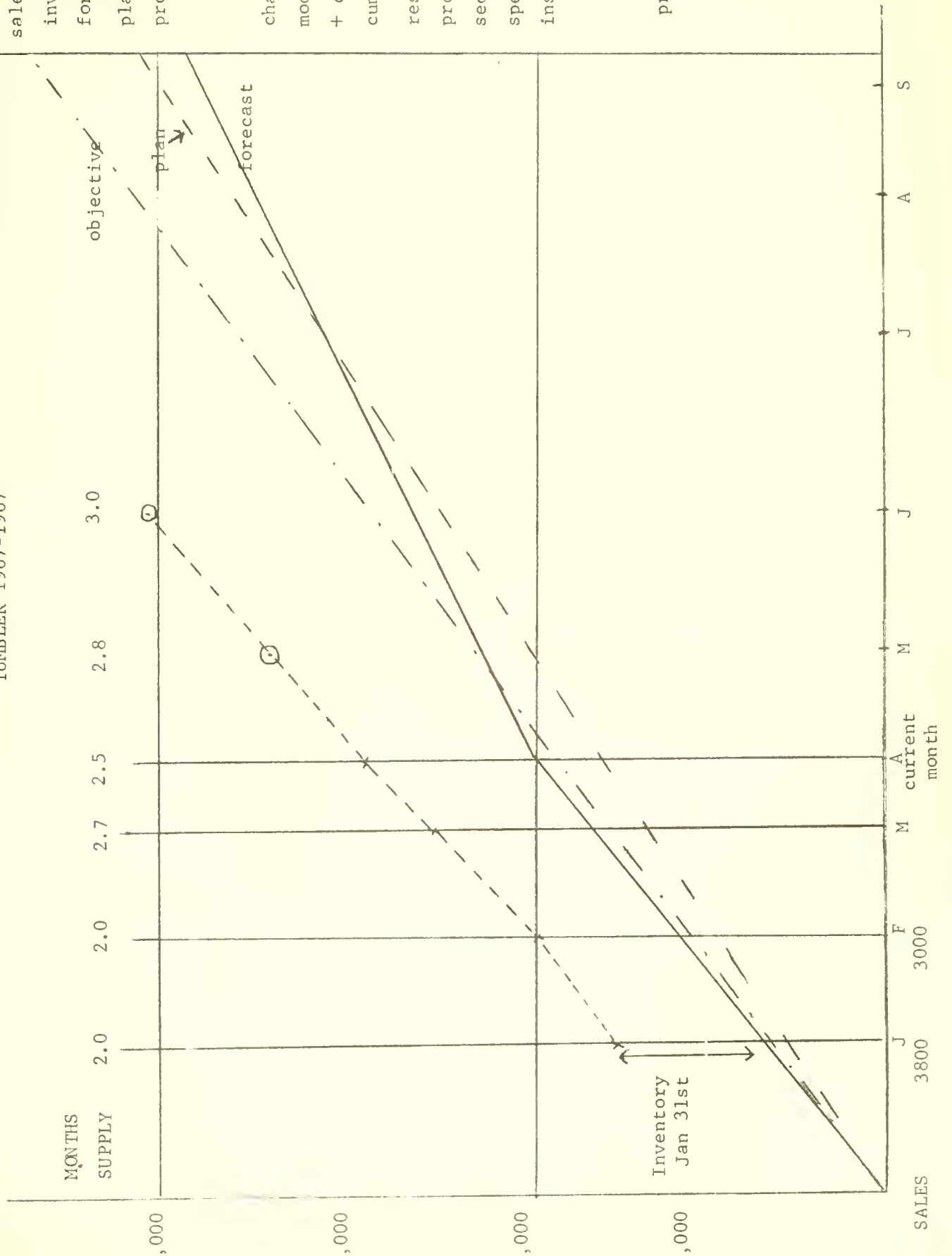
Proceed



Exhibit 2

TUMBLER 1967-1967

- objective
- sales-ly
- inventory
- forecast
- plan
- production
- change points
- model forecast
- + or - line
- cum/non-cum
- rescale
- project
- sequence
- specifications
- insert
- proceed





tify the points concerned with the light pen, insert +15% on the keyboard, hit 'change points' with the light pen and then hit 'proceed'. He would then get back a new display with that change implemented and, of course, his new inventory status.

This goal of ease of interaction is partially dependent on software design and partially on hardware components. A light-pen or Rand Tablet is absolutely essential.

#### C. General purpose characteristics

The software must be designed with an attempt to be appropriate for most applications within the broad framework of unstructured management problem solving. A clear distinction has to be drawn here between the structure of the software design and the specific details of implementation. The words that appear on the display in any one application are clearly a function of that particular application. The basic architecture of the system, however, should be common across functional areas and levels in the management hierarchy. We have not had enough experience in our experiments to date to claim that we have been successful with this goal. The software architecture (see 10) has in fact proved valid in other application areas that we have begun to test. For example the financial accounting and cost accounting problem areas seem to be compatible with our structure. Further testing will be required before we can generalize from this experience.

#### D. Modular in design

It was felt that the system had to be able to absorb changes in software to permit the addition of different types of displays, functions and





the use of different forms of data. The insertion or deletion of one set of subroutines or application packages should not affect the others in any way. The software description (see 10) establishes the way in which we accomplished this. Experience has reinforced the importance of this objective. Since the terminal systems support individual managers and since they involve in-depth analysis of the problem area they are extremely volatile. Change is constant as user understanding of his decision-making process grows. Coupled with this is the changing nature of the problems and models involved. Modularity is critical if the system is to remain useful while undergoing continual evolution.

#### E. Fail-safe capability

The system has to protect both itself and the user from any accidental or deliberate sabotage. For example, the system should be able to detect and respond to obvious user errors. These may be system errors, where the user misunderstands the appropriate response, or application errors, where he provides incorrect or inconsistent input data. Secrecy of data and the ability to protect one's private files from other user's writing or looking are obvious corollaries of this particular goal. We achieved some moderate success with this goal by simply applying the Project MAC time-sharing precautions. However we have not had to be innovative on this issue as we are yet to support more than one visual terminal at a time. Similarly we have a small and well-educated user group so that user error is very low. Intuitively the goal seems relevant and its solution difficult but we have had little experience thus far.



#### F. Hardware independence

The basic structure and logic of the system should not be critically dependent on the hardware being used at any one point in time. It was felt that considerable flexibility could be gained if the executive system was programmed in a higher level language. It was found that the loss in efficiency was offset by the ability to change the structure relatively easily. In a management setting it is most unlikely that any problem environment will remain stable enough, long enough, to justify the little additional benefit from the efficiency of the machine language coding. The importance of this goal is underscored by the rapid changes in technology. The terminal purchased two years ago from Information Displays Inc. for one of the projects cost roughly \$125,000. It has a light-pen, displays 140 characters across the screen, is connected by telephone to the central computer and completes a display in about 7 seconds. At the Sloan School, MIT, we have been using a terminal which has many of the characteristics necessary for a management terminal and cost \$15,000. Most predictions of the technology indicate that we can expect this order of magnitude reduction to continue in the coming years.

With these goals we developed a software system to support interactive management problem solving in an unstructured environment. Item 10 in the bibliography contains a general functional layout of this software and a description of its components. Our experience with the system in the two years of operation and expansion leads us to assert that these particular goals are meaningful in a management context. They are also necessary, as a failure along any one of the dimensions can cripple the project.



## Conclusions

Other material (6,11) discusses the impact of a management decision system on managerial problem solving. In the two cases mentioned in this discussion the impact has been considerable. As a result of their continuing use we have evidence from an on-going management situation that such systems can be built, and be cost/effective. Managers do find them to be useful support systems for certain classes of problems and in fact they can have a dramatic impact on the decision process. There was some evidence to suggest that their decision process was more effective, and it was clear that they had an improved sense of perspective of their problem and its relationship to their environment. This, coupled with the level of insight and the commitment exhibited by the managers indicated a powerful impact by the system.

Despite interactive terminal system's operational utility when used on unstructured problems, it is tempting to argue that their biggest benefit is from a research standpoint. The modeling of the decision making process and the manager's use of the system during his decision making activities, results in a great deal of understanding of that particular decision process. Over time, having done this in a number of different situations, it is obviously possible to collect a good deal of data about the way managers make decisions and about the way specific classes of decisions are handled.

Even more useful, perhaps, is the fact that if the terminal system is designed properly, the managers will be using it throughout the decision making process and, of course, the computer can maintain a trace



of exactly what they do. The system can be programmed to monitor all transactions that take place and the researcher then has a time trace of what has transpired. This can be analyzed and more data collection and statistical gathering routines designed and built in to further monitor the decision making process. From this interactive process, one can develop further decision rules and eventually these can be built into the system so that it can learn from itself. Even at the beginning of this cycle we can define rules to identify patterns, suggest alternatives, and generally act as an active participant in the decision making process. In the first prototype system that we constructed, we only had simple checks for consistency on the part of the managers. For example, if managers made inconsistent suggestions, the system would point these out to him. As we develop our concepts and understanding of the decision making process in general and the particular decision that the manager is faced with, it becomes possible to structure more and more of the decision making process.

This boot-strapping operation to provide the systems with "intelligence" is the most exciting aspect of the research to date and offers some interesting long run potential. As we work on these longer run considerations we can provide useful tools to line and staff management to help in their decision making. The design and evaluation strategy discussed here is one approach that has been taken to these problems and has proved to be reasonably effective. Much more work has to be done before anyone can claim to have a really good methodology; active discussion of what has been tried may be one useful way to improve the quality of future research.





References

1. Anthony, R. N., Planning and Control Systems: A Framework for Analysis, Division of Research, Graduate School of Business Administration, Harvard University, 1965.
2. Carroll, D. C., "Implications of On-Line, Real-Time Systems for Managerial Decision-Making," in Myers, C. A. (Ed), The Impact of Computers on Management, The M.I.T. Press, 1967.
3. Clarkson, G. P. E., A Simulation of Trust Investment, Prentice-Hall, 1961.
4. Dearden, J., "Myth of Real-Time Management Information," Harvard Business Review, May-June, 1966.
5. Diebold, J., "What's Ahead in Information Technology," Harvard Business Review, September-October 1965.
6. Morton, M.S.S., "Computer-Driven Visual Display Devices--Their Impact on the Management Decision-Making Process," Doctoral Thesis, Harvard Business School, 1967.
7. Morton, M.S.S., "Terminal Costing for Better Decisions" (with Andrew McCosh), Harvard Business Review, May-June 1968.
8. Morton, M.S.S., "Strategic Planning and Management Information Systems--Some Implications of Present Research," MIT Working Paper No. 338-68, July 1968.
9. Morton, M.S.S., "The Impact of Interactive Visual Display Systems on the Management Planning Process" (with James Stephens), Proceedings of the IFIP Congress 68, Edinburgh, Scotland, August 1968. (Also MIT Working Paper No. 356-68.)
10. Morton, M.S.S., "Software Architecture for Interactive Decision Systems," Forthcoming MIT Working Paper.
11. Morton, M.S.S., Management Decision Systems, Harvard Business School, Division of Research, (forthcoming).
12. Simon, H. A., The Shape of Automation for Men and Managers, Harper & Row 1965.

200-20

200-1

200-78

200-200

200-5

200-200

200-200

200-200

200-200

200-200

200-5

200-200



RAFFM Date Due

DEC 05 '75	<del>████████</del>	
FEB 05 '78	OCT 12 1988	
FEB 23 '78		
<del>FEB 17 '77</del>	FEB 14 1989	
SEP 24 '77		
Nov. 18		
<del>SEP 23 '77</del>		
MAY 17 '78		
AUG 21 '80		
W		
██████████		
██████████		
██████████		

Lib-26-67

MIT LIBRARIES



430-69

3 9080 003 906 689

MIT LIBRARIES



431-69

3 9080 003 906 705

HD28

34558

loan  
it

MIT LIBRARIES



432-69

3 9080 003 906 721

MIT LIBRARIES



433-69

3 9080 003 906 739

MIT LIBRARIES



434-69

3 9080 003 875 769

MIT LIBRARIES



435-69

3 9080 003 875 918

MIT LIBRARIES



436-70

3 9080 003 875 793

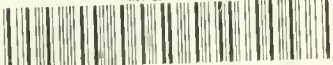
MIT LIBRARIES



438-70

3 9080 003 875 827

MIT LIBRARIES



439-70

3 9080 003 875 835

MIT LIBRARIES



440-70

3 9080 003 906 762



