DECISION SUPPORT SYSTEMS
AND
MANAGERIAL PRODUCTIVITY ANALYSIS

Peter G. W. Keen

October 1980

CISR No. 60
Sloan WP No. 1156-80

Center for Information Systems Research
Massachusetts Institute of Technology
Alfred P. Sloan School of Management
50 Memorial Drive
Cambridge, Massachusetts, 02139
617 253-1000

DECISION SUPPORT SYSTEMS

AND

MANAGERIAL PRODUCTIVITY ANALYSIS

Peter G. W. Keen

October 1980

CISR No. 60

Sloan WP No. 1156-80

This is an expanded version of a paper presented at the American Productivity Council Conference on Productivity Research, April 1980.

CONTENTS:

FIGURES:

1. Introduction

This paper describes one successful approach to using computer technology to improve managerial productivity. Decision Support Systems (DSS) are small-scale, interactive systems designed to provide managers with flexible, responsive tools that act in effect as a staff assistant, to whom they can delegate more routine parts of their job. DSS support, rather than replace, managers' judgment. They do not impose solutions and methods but provide access to information, models and reports and help extend the managers' scope of analysis.

A main stimulus to the development of the DSS field has been the clear lack of success the Management Information Systems and Management Science disciplines had in the 1960's and early 1970's in providing managers with useful and usable tools. Somehow, the impeccable analytic logic of MIS/MS methods rarely could be translated into a vehicle managers found helpful. It was easy to argue that the managers were at fault, either because they resisted new methods, feared computers or lacked adequate education. MIS/MS largely presented a prescriptive view of decision making; the concept of an "optimal" solution or the term management science indicates the rationalistic approach that underlay efforts to apply computers and analytic methods to decision making.

Decision Support - the end for which a DSS is the means - is far more descriptive. It begins by asking how managers actually make decisions. Support involves helping. Much of the skill in DSS design consists of tailoring the technical system to a nonanalytic manager's vocabulary, mode of thinking and perceptions of his or her job. At the same time, there has to be some rationale for investing time and money in building such aids. They must improve the effectiveness of the user.

While the DSS field has been pragmatic and primarily concerned with building systems, this need for descriptive and prescriptive mapping of decision making has resulted in a focus on two questions central to any general discussion of managerial productivity:

(1) What does it mean to "improve" a manager's decision making? (prescription)
(2) What does one have to know about the existing decision process before one can try to improve it? (description)

The conceptual and empirical work on DSS provides some practical answers to these questions that directly address issues ignored in many discussions of productivity. Too often, productivity is equated with efficiency: reducing unit cost, improving output per hour or eliminating errors. At the extreme, this approach suggests that we can improve the productivity of a strategic planning group by increasing the number of plans it generates per month. Managerial work, however, does not produce well-defined outputs from well-defined inputs. The <u>quality</u> of the plan is more important than volume. Performance is measured in terms of effectiveness as well as efficiency. Decision making is a process of problem-solving, not a tidy set of standard operating procedures.

For many managers - and white-collar and blue collar workers - much of the traditional literature on productivity is irrelevant and even objectionable. So, too, are the articles now prevalent in news and business publications, with titles like "Can American solve the productivity crisis?" From the worker's viewpoint, "productivity" implies efforts to impose more procedures and controls and squeeze more work from them. The message is "work harder". The concept of Decision Support is far more sympathetic to their view of their jobs, and not some outside economist's: "work more effectively". In fact, in many cases the explicit aim for a DSS is to relieve the managers of part of the workload: "work more effectively and less hard".

The output of any DSS project is a piece of software. In many instances this is well behind the technical state-of-the-art and is small in scale. A DSS is not a specific technology. Its quality and impact depend on the designer's understanding of the decision process. Because of this, Decision Support can provide the base for a generalized approach to productivity analysis, by looking at the task from the inside - the manager's view - to the outside - the accountant/economist's evaluation. Both perspectives are important, but if one focuses only on the outside, it will surely be hard to interest managers in "improving" productivity.

This paper discusses Decision Support from this perspective. Technical aspects of DSS development are included only where they illustrate how methods of productivity analysis can be used to specify tools for productivity improvement. Figure 1 shows typical DSS applications described in published case studies.[1]

FIGURE 1 Examples of DSS Applications

DSS		APPLICATIONS
GADS	Geodata Analysis Display System	Geographical resource allocation and analysis; applications include sales force territories, police beat redesign, designing school boundaries
PMS	Portfolio Management System	Portfolio investment management
IRIS	Industrial Relations Information System	Ad hoc access to employee data for analysis of productivity and resource allocation
PROJECTOR		Strategic financial planning
IFPS	Interactive Financial Planning System	Financial modelling, including mergers and acquisitions, new product analysis, facilities planning and pricing analysis
ISSPA	Interactive Support System for Policy Analysts	Policy analysis in state government; simulations, reporting and ad hoc modelling
BRANDAD		Marketing planning, setting prices and budgets for advertising, sales force, promotion, etc.
IMS	Interactive Marketing System	Media analysis of large consumer database; plan strategies for advertising
CAUSE	Computer-Assisted Underwriting System at Equitable	Calculate and track group insurance policy and renewals
AAIMS	An Analytical Information Management System	Analysis of time series data on airline industry operations (database contains airlines' reports to Civil Aeronautical Board)

2. Productivity in "Semi-structured" Tasks

Gorry and Morton (1972) identified "semi-structured" tasks as the target of opportunity for DSS. Structured tasks are programmables. They have clear goals, inputs, and rules or standard operating proceeding for arriving at the desired outputs. Figure 2 lists examples, several of which used to be unstructured; consumer credit scoring, for instance, once required the skilled judgment of a loan officer.

For a structured task, productivity is largely equivalent to efficiency. For credit scoring, "better" performance comes from:

(1) reducing the cost of processing each application
(2) improving standard operating procedures
(3) reducing turnaround time
(4) improving accuracy.

There is no need for judgment in such a task. A loan officer may, of course, override the "decision" of the scoring algorithm in special cases. However, the standard operating procedures generate a solution independent of the manager's judgment.

MIS/MS largely views tasks as structurable. For example, linear programming replaces ad hoc, intuitive or qualitative trade-offs with an explicit and normative methodology:

(1) define the (single) objective
(2) specify all relevant constraints and coefficients
(3) run the LP model
(4) implement the solution

One's definition of "productivity" and hence of "useful" decision aids largely depends on the degree to which one feels managerial activities can or should be structured. The traditional response of managers to MIS/MS zealots is that their own job involves situational factors and qualitative trade-offs. There is always some degree of judgment needed. They must work with incomplete data and poorly-defined problems. For example, lending officers in an international bank can point out that a credit scoring model is inappropriate, however useful it may be for small consumer loans. For $5,000 personal loans, there is a large actuarial data base available from which statistical rules can be devised (e.g. give the applicant 2 points for having a telephone and 4 for being in his or her current job for at least 2 years). The lender's risk is small and incorporated into the credit scoring model which is designed to hold loan defaults to an average of X%.

The situation is very different for a $15 million, eight-year loan to a Brazilian syndicate to take over an import-export firm. The factors listed below are only a few of the ones the lending officer has to take into account:

(1) country risk; Brazilian inflation, economic growth and the bank's current exposure in the country.
(2) customer relationship; is this loan desirable as a means of getting the business of clients whose needs will grow over the next few years?
(3) quality of the import-export firm; is this a sound investment?
(4) credit-worthiness and assets of the syndicate.
(5) type of loan; security, profitability, length.

The lending officers' view of what makes a "good" loan focuses on creativity, experience and process. While there are constraints and rules, these must be subordinated to judgment. The key document is the "credit file" for the loan. This contains a mass of qualitative information, often in letters.

The decision process is highly situational and there is no doubt that a credit scoring or LP model is inappropriate. In general, as in this example, interesting managerial tasks are not structured. If they are, they can - probably should - be delegated; "decision making" implies dilemmas and judgment.

Unfortunately, managers - and analysts - often view the use of MIS/MS techniques as "all-or-nothing". The manager states that since the task is not structured, computers cannot handle it and hence have nothing to offer. The analyst points out that is is not unstructured either. The Brazilian loan involves forecasting, calculating yields and payments, reporting and analyzing financial statements. The lending officer is required to obtain a spread (the difference between the bank's effective borrowing and lending rates) of at least 1.5%. Of course, systematic analysis is essential and of course a computer can be used to improve its quality.

Decision Support replaces all-or-nothing with some-of-each. It accepts more of the manager's view than the MIS/MS purists. Decision making is indeed a situational, ad hoc process in which effectiveness often depends on the manager's creativity, experience and judgment. It makes no sense to impose a method or structure on that process. At the same time, the management purist overlooks the fact that few if any tasks are purely judgmental. One can identify ways in which some form of computerized staff assistant could help the lending officer be more efficient and effective:

(1) handling "spread sheet" calculations, e.g. computing quarterly cash flow, profit and repayment schedules given such inputs as "revenues = $1.3 million for 1981, estimated annual growth = 11%", "Now adjust for inflation of 60%".

(2) quickly answering "what if?" questions: "what if the growth is 9% instead of 11?"

(3) fine-tuning the loan package: "say we reduce the interest by 1/4% and take a commitment fee of $150,000 in year 1?" "What about changing the draw-down rate from year 3 on?"

(4) working backwards: instead of computing the loan yield from the interest rate, ask "What rate do we need to get a 1.8% yield?"

(5) providing fast, ad hoc access to available data: "What do we have on trends in Brazil's trade in industry codes A, B and C?" "Check the credit file for anything on commitments from outside investors."

The concept of a staff assistant is key here. Handling the spread sheet projections and "what if" questions involves some simple simulation capability. Its usefulness depends on technical quality and the validity of the assumptions built into it. In this case, the rules are very straightforward. The usability of the system is at least as important to a manager as its usefulness. The dialog with the staff assistant needs to be natural; the manager should not have to learn awkward methods or have to adapt what he wants to do to the staff assistant's mode of operation. Flexibility, responsiveness and ease of use are essential.

Most computer decision aids lack these features. Models are too often arcane, cumbersome and above all, inflexible. From the manager's viewpoint, the marginal economics of effort are unfavorable. The investment of effort needed to use the system is greater than the perceived gains. After all, no one would use a staff assistant who speaks only in Serbo-Croat (albeit with a Boston accent), demands numbers in hexadecimal (to the base 16 - much more "efficient"), insists on a fixed dialog ("answer each question: (1) what is the estimated first year sales? (2) what is . . . .? (80) do you want to make another set of projections? o.k. (1)what is the estimated first year sales?. . . . etc.) and provides no explanations ("invalid input: try again"). Far too many computer systems are not really usable; it is rational then for a busy manager who is making fairly good decisions without their help to reject them even if they are potentially useful.

The more structured the task, the less flexibility and responsiveness is needed. In the consumer credit scoring case, the manager in effect says "income $23,600, renter, new job, 5 years in previous location. . . . is this above the risk threshold?" If there are 20 loans a day to process, the manager or the clerk to whom the task is delegated will willingly learn a shorthand mode of input "23600, R, 0, 5. . . . "

Conversely, the more judgment a task involves, the less predictable and structurable the problem-solving dialog. In addition, though, the greater the payoff can be from providing a computerized staff support function. Managers are not as productive as they would like to be. The "all-or-nothing" view of computers usually ends up in nothing. The loan officer does most projections inefficiently, by hand, approximation and use of a calculator. His effectiveness is limited; since it may take a week to obtain extra information on industry trends in Brazil, he does not ask for it. Since it takes 3 hours to project the impact of lowering the interest rate 1/4% and taking a commitment fee, he tries to limit the range of alternatives to be examined and avoids "what if?" questions. If a designer can make the DSS usable, there is plenty of opportunity to be useful.

FIGURE 2 Structured Tasks

TASK	PROCEDURE
consumer credit scoring	application of statistical weights derived from actuarial data
oil refinery scheduling	linear programming model
tic-tac-toe	if player 1 does A, do B
inventory ordering	EOQ (economic order quantity) algorhythm; when stock falls below border point, place order
airline reservations	check if customers request feasible; if not, try nearest time slot or alternate route

3. Improving Decision Making

"Improving productivity" is equivalent to "doing a better job". For managerial tasks this includes being more efficient, but clearly, the main concern is effectiveness. The lending officer's career depends on making "good" loans (or at least avoiding bad ones).

The contrast between efficiency and effectiveness has been captured as: efficiency involves knowing how to do a job and effectiveness as knowing what job to do. The more important the job, the more likely one is to make effectiveness a priority. Efficient purpose of an ineffective goal is a sure path to failure.

In trying to improve either effectiveness or efficiency, one must begin by establishing the baseline. It may be intellectually interesting to aim for "optimal" decisions, but more helpful to ask:

(1) how is the job currently done?
(2) where can performance be improved?

Only then can one ask the questions relevant to building a computer system:

(3) what has to happen for the manager to use a DSS that contributes to the improvement?
(4) what is the cost of providing it?

These will be discussed in more detail in sections 4 and 5. It is worth mentioning here that one reason why there is often little incentive to use computers to improve managerial efficiency is that the cost is disproportionate. For example, if it costs $9,000 to build a small-scale DSS (not an unrealistic figure; see section 6), one may need huge efficiency gains to justify the investment. Increasing the yield on a $15,000,000 8-year loan by 1/16 % is worth $75,000. It makes sense to commit time, resources and attention only to high-payoff areas, which is partly why busy managers neglect opportunities to be more efficient.

MIS/MS analysts often point to the limitations of managers. They feel that their own prescriptive strategy is essential and there is little sense in building on managers' existing, inadequate methods. For instance, several commentators have pointed to "satisficing" (Simon, Cyert and March, Ackoff and Sasieni). Decision makers do not search for the optimal solution, but stop when they find one that is good enough. The lending officer who sets an"aspiration level" of 1.5 % net spread may fine-tune a solution which yields 1.53 % but will not go on and actively look for the alternative that offers 1.84 %. Ackoff and Sasieni comment:

> "Satisficing is usually defended with the argument that it is better to produce a feasible plan that is not optimal than an optimal plan that is not feasible. This argument is only superficially compelling. Reflection reveals that it overlooks the possibility of obtaining the best feasible plan. Optimality can (and should) be defined so as to take feasibility into account, and the effort to do so forces us to examine the criteria of feasibility that are seldom made explicit in the satisficing process. Furthermore, the approximate attainment of an optimal plan may be more desirable than exact attainment of an inferior one. Not surprising, this type of planning seldom produces a significant break with the past. . . . It appeals to planners who are not willing to stick their necks out." 2

Apart from satisficing, most managers indeed have many limitations:

(1) they rely on heuristics: rules of thumb that give answers that are good enough most of the time, e.g. "set advertising equal to 3 % of the sales forecast" (surely, though, advertising is meant to influence the sales forecast)

(2) simplistic concepts (see, for examplè, Stabell; portfolio investment managers who have complex ways of analyzing stocks, think of portfolios only as "big v. little" or "income v. growth")

(3) overreliance on intuition (see Ross' discussion of the failings of "the intuitive psychologist")

(4) faulty interference (Tversky and Kahneman)

Taylor summarizes management decision making strategies as aimed at simplifying a problem to make it soluble, even if that oversimplifies or distorts it. A general principle in problem-solving is to reduce cognitive strain and strike a balance between performance and effort. Satisficing, heuristics and intuition based on experience provide very low cost strategies. Instead of having to commit substantial resources of time and intellect to fundamental analysis, managers can apply a few quick and simple principles, and get acceptable decisions. "Set advertising equal to 3% of the sales forecast" is clearly not optimal, but it may represent the best available compromise between performance and effort.

The "marginal economics of effort" seems central to any dis-[3] cussion of productivity that begins from the manager's perspective. MIS/MS stresses the potential improvement in decision quality that analytic methods provide, but generally overlooks the substantial increase in cognitive strain they impose. (see Figure 3) It is not at all unreasonable for managers to feel that the increase in quality is not worth the increase in effort, if their current performance is at a

FIGURE 3 Marginal Economics of Effort

CATEGORIES OF COGNITIVE EFFORT	CONSEQUENCES FOR MANAGEMENT DECISION MAKING
1. Computation arithmetic calculation enumeration	use of approximation or electronic calculator
2. Specification conscious articulation of weights, priorities, preferences point estimates of probabilities or utility functions	incremental approach used to avoid specification; compare choice A vs. status quo -- "do I prefer this to this?"; eliminates need for prior statement of objectives
3. Search scanning creating alternatives	use satisficing and heuristics to limit alternatives to be considered
4. Inference deriving conclusions from data statistical reasoning generalizing from specific instances	many logical errors, over-reaction to salient or recent data
5. Assimilation responding to numerical, graphical or verbal information assessing results of analysis "making sense" of data	rely on verbal information, avoid moods and reports that provide too much information
6. Explanation justifying solutions explaining conclusions and/or methodology that leads to them	stick to methods of analysis consistent with organizational norms and procedures

satisfactory level.

Decision Support exploits the marginal economics of effort. There are several ways the computerized staff assistant can contribute to managers' jobs.

It can help them:

(1) carry out existing activities with less effort
(2) replace easy-to-use heuristics with easy-to-use analytic methods
(3) stimulate broader search and analysis by reducing the effort required to look at one more alternative ("what if. . . ?")
(4) encourage fundamental analysis by making useful models usable.

Clearly, such aid cannot be imposed. If we accept that productivity means "doing a better job" and that in trying to do so managers balance quality and effort, then they themselves must be able to pick and choose from the DSS facilities. They will find their own ways of exploiting the system. Section 5 discusses the impact of DSS, drawing on published case studies. The general experience is that managers use the same DSS in very individualized ways; there is no typical pattern. Interestingly, the actual uses are often very different from the suspected or intended ones (see Figure 4).[4] Once managers trust the system and find it of personal value, they actively find new ways to improve their decision making - "improve" from their own perspective - for very little effort.

FIGURE 4 Unanticipated Uses of DSS

INTENDED USE	UNANTICIPATED USE
investment decisions	communicating with customers
portfolio analysis	explaining rationals of decisions; "problem-finding;" marketing tool
analyzing financial data to answer pre-planned questions	educational; alerting user to new issues and unplanned questions
budgeting and forecasting	"problem-finding;" where DSS projections differ from actual experience, users need to decide if their mental model is incomplete or if some environmental change has occurred

The marginal economics of effort is central to Decision Support. It highlights opportunities for and constraints on any strategy for productivity improvement. Managers use simple methods for analysis and decision making. This is:

(1) a constraint on introducing new methods and tools; they are very sensitive to costs, and indeed may choose to forego potential benefits because of this.
(2) an opportunity: using rules of thumb and limited scope of analysis is clearly not the best strategy a manager could choose. Simon comments that no one in his right way will choose to satisfice when he could just as well optimize. "Just as well" means "with no disproportionate increase in effort".

The view of decision making presented here is fairly simple and the goals of Decision Support are modest. It looks at managers in their jobs and does not try to define optimal strategies or worry too much about efficiency. The goal of productivity analysis should be to identify constraints and opportunities. The individual manager is the best judge of how to use the staff assistant. In the end, productivity improvement comes from the manager's own actions and learning.

4. DSS and Productivity Analysis

It is difficult to select a "typical" DSS given the range of successful applications and the custom tailoring of the systems to specific uses in specific tasks. IFPS (Interactive Financial Planning System) is a DSS generator used in a large number of organizations. There are several case studies and surveys of IFPS that highlight the features managers want in tools for Decision Support. IFPS is thus a useful example of the DSS approach.

A DSS generator is a language which allows models and reports to be built quickly by non-programmers. Figure 5 shows a brief example of the IFPS language. IFPS is extremely simple and English-like. (Interestingly, a general problem for DSS designers is that if they do their job well and build a flexible, easy-to-use system, technicians are often remarkably disdainful: "its too easy, lacks sophisication, not elegant" - but that is exactly what is should be, and it may take hard thinking and creative design to make it look easy.)

IFPS is one of several effective "end-user" languages that dramatically reduces the time needed for building systems and allows a (human) staff assistant to give managers responsive service. The average IFPS model takes 5 days to implement; most applications address some aspect of

FIGURE 5 <u>IFPS</u>

(1) Sample model statement

```
MODEL INCOME VERSION OF  05/08/79  13:11  .

COLUMNS 1-5
*
*      INCOME STATEMENT
*
VOLUME = VOLUME ESTIMATE, PREVIOUS VOLUME * 1.045
SELLING PRICE = PRICE ESTIMATE, PREVIOUS SELLING PRICE * 1.06
     SALES = VOLUME  *  SELLING PRICE
UNIT COST = .85
VARIABLE COST = VOLUME * UNIT COST
DIVISION OVERHEAD = 15% * VARIABLE COST
STLINE DEPR(INVESTMENT,SALVAGE,LIFE,DEPRECIATION)
     COST OF GOODS SOLD = VARIABLE COST & DIVISION OVERHEAD &
                          DEPRECIATION
GROSS MARGIN = SALES - COST OF GOODS SOLD
OPERATING EXPENSE = .02 * SALES
INTEREST EXPENSE = 15742,21522,21147,24905,21311
*
NET BEFORE TAX = GROSS MARGIN - OPERATING EXPENSE - INTEREST EXPENSE
TAXES = TAX RATE * NET BEFORE TAX
     NET AFTER TAX = NET BEFORE TAX - TAXES
*
INVESTMENT = 100000,125000,0,100000,0
*
RATE OF RETURN - IRR(NET AFTER TAX & DEPRECIATION, INVESTMENT)
*
*   DATA ESTIMATES
TAX RATE = .46
VOLUME ESTIMATE = 100000
PRICE ESTIMATE = 2.25
SALVAGE = 0
LIFE = 10
END OF MODEL
```

of strategic financial planning or forecasting.

The discussion below is based on a survey of 300 IFPS applications in 42 organizations (1979).[6] 75% of the reported applications replace manual procedures, for an additional 90% no similar analysis was done at all, prior to introducing DSS capability. Given that most of the companies are in the Fortune 100, this is astonishing. Financial planning and modelling are obvious applications for computers. They involve myriads of calculations and there are literally hundreds of cheap, accessible software packages available. This seems to be an example of the consequences of all-or-nothing. Clearly, existing systems are not seen as usable, even if useful, and managers and staff in major organizations still prefer to do things by hand.

The 300 IFPS models are used an average of 48 times a year by an average of 25 individuals. The DSS are built by staff analysts, not programmers; less than 1% were built by data processing personnel, 81% by planning analysts and managers, and 19% by line managers.

In only 16% of the cases were clear design specifications developed. In 66% of the applications, someone simply responded to a manager's request and got something started as quickly as possible. While virtually all respondents expressed full satisfaction with IFPS and felt that it had made a major contribution, 70% had no objective data to support its value.

The IFPS features respondents view as contibuting most to the success of the applications are:

1. speed of reponse
2. ease of use
3. package features (curve fitting, monte carlo, what-if, etc.,)
4. sensitivity analysis
5. time savings

The perceived benefits were:

1. better decisions
2. faster decisions
3. saving analyst time
4. better understanding of factors affecting the business
5. more creative thinking
6. saving management time
7. leveraging management skills

The overall findings of the survey are consistent with those of DSS case studies. Managers will use computer systems that mesh with their natural mode of operation.

The need for flexibility, ease of use and responsiveness in DSS has gradually led to a fairly standard design strategy that in itself provides a starting point for productivity analysis. This "adaptive design" methodology is based on:

(1) prototypes

(2) programming languages that allow fast development, modification and extension (generally at the cost of machine efficiency)

(3) a "verb-based" structure

ISSPA (Interactive Support Sustem for Policy Analyst) is a DSS built for use by state government policy makers and analysts. It is equally well-suited to human resource management, labor negotiations and any other application which involves analysis and reporting where the data are in the form of a matrix (eg. employee name (row) x career, salary, and training information (columns) or school district x financial and enrollment data). It was built explicitly to test and illustrate the effectiveness of the adaptive design approach.

One of the difficulties in improving performance in complex tasks is that they <u>are</u> complex. The basic activities shown in the international loan example are easy to identify, but by defination such a "semi-structured" task does not allow tidy routinization. The manager determines the sequence in which the activities are put together. Because of the situational nature of the process, it is almost impossible:

(a) for the analyst to understand it fully enough to lay out the specifications for a DSS

(b) for the manager to explain it fully enough to do so

David Ness uses the term "middle-out" design to capture the way in which a prototype can resolve this dilemma. (Middle-out is in contrast to "top-down" where the whole structure is laid out in advance and "bottom-up" where individual pieces of the system are built and then later integrated.) He argues (and the point is well-supported by most DSS case studies) that the DSS builder needs to put together as quickly as possible a concrete, usable system to which the manager can react. The DSS can then evolve through use, modification and extension.

This approach is almost opposite of traditional data-processing methods, which rely on gathering"functional" specifications in order to build an "operational" system on which the client "signs-off." It is no coincidence that data-processing has had most impact on efficiency in structured tasks.

A low-cost prototype can generally be built in under two weeks, assuming there are no complex technical or organizational problems in data management. It represents a first cut at a solution and will generally support only a few of the manager's activities. For example, the initial version of the ISSPA included just over twenty commands (see figure 6). The protoptype must be a real system, not an approximation nor equivalent to a feasibility study.

FIGURE 6 Examples of ISSPA Commands

A. ISSPA Commands

1. Commands in Initial version (22)

CORRELATE*, COUNTIF, CROSSTAB*,
DEFINE, DESCRIBE, FREQUENCIES*,
HISTO*, LIST, NTILES, RANK, REGRESS*, SCATTER*,
TOP, BOTTOM

* taken from APL public library

2. Added when initial version released for wider use (16)

ADD, DATABASE, FORMAT, PARTIAL CORR, RANGE, SCALE, WAVERAGE

3. Added at user request (9)

SESSION COST, DISPLAY...FOR UNITS, SYNONYM, YEARS, * SAMPLE

4. "Evolved" commands (6)

WTILES, BOXPLOT, CONDENSE, EQUITY

B. Examples from other "command-driven" systems

1. PMS (Portfolio Management System)

GRAPH, GROUPS, HISTO, MANAGER, MARKET, SCAN, SCATTER, SUMMARY,
TABLE, ADD, CREATE, FILTER, BUY, SELL

2. AAIMS

DISPLAY, PLOT, CHANGE

The prototype provides something concrete for the user to react to. Ackoff, in "Management Misinformation Systems", long ago pointed out that managers rarely know what they need in terms of reports. In a complex job involving judgement and experience, the designer asking "What do you need" is meaningless. The best answer may be "What have you got?" In middle-out design, the DSS builder first sketches out the user-system dialog -builds the staff assistant-and then identifies priority routines. In effect, the prototype represents the builders understanding of the task and instead of asking the user "What do you need", he or she presents the initial version as "How do you like this?" The user, by working with the DSS helps design it.

Obviously, any strategy based on prototypes and evolution must use tools that allow quick and easy development. If it takes 4 months to build the initial version and 3 months to modify in in response to the user's reactions, all momentum will be lost. There is a rapidly growing trend in organizations towards special-purpose languages for this reason. IFPS allows a model to be developed in an average of 5 days. APL, the language used for building ISSPA, is especially suitable for DSS. It has three main advanatages:

(1) whole routines can be built and tested in hours instead of weeks

(2) it can be learnt by anyone with mathematical and analytical aptitude in a month; this means that DSS can be and increasingly are built by nonprofessional programmers who are experts in the applications area (of the IFPS survey; 81% of the models were built by planners.)

(3) major additions to or reconstructing of the system can be made with minimal effort; the "bread-boarded" prototype then provides a basic structure for growing the DSS

APL has limitations. It is comparatively expensive to run, is arcane and at times results in unavoidable costs However, if the priorities are responsiveness and service to a manager - Decision Support - and not operational efficiency, the benefits of APL greatly outweigh the costs.

ISSPA is a "command-based" DSS. It consists of a set of routines invoked by the manager in any sequence. For example, a policy analyst can answer the following questions from a legislator through the simple dialogue shown below:

(1) Questions - "How many districts had revenues over $1,500 per pupil? What was their average aid? Can you give me the district names and calculate how much they would get under the Senate's new proposal? What would be their average level of aid? Which 5 districts would get the largest percentage increase?"

(2) ISSPA Dialogue - (slightly abridged; report headings and routines for DEFINE command are excluded).

SELECT IF Total revenue = 1500

15 Units Selected

Do You Wish To See The Units Currently Selected? YES

1. Bradley

2. Capitol

⋮

3. Lawrence

AVERAGE STATE AIDE

Average = $72,615

DEFINE (followed by Senate bill's formula; variable given label "Senate 81")

AVERAGE SENATE 81

Average = $86,812

DEFINE 100 (Senate 81 State Aid) ÷ State Aid (Variable given label "INC%")

TOP 5 INC%

1.	Zellerbach	103.6
2.	Conroy	89.3
⋮		
5.	Bradley	84.0

Each of the commands is equivalent to saying "do this" (verb) "to this" (noun). The verbs are usually commands to a staff assistant. The nouns are data items. The verbs are respresented in the DSS by a corresponding computer routine (APL is especially convenient in this respect.)

Identifying the user's verbs is a key step in productivity analysis. If the function in a DSS (models, retrieval and reporting routines) do not correspond fairly directly to a manager's verbs, it is hard to see how he or she can use them. Any computer aid must provide managers with familiar representations.[8] The building blocks in the manager's decision process are discrete substeps such as Forecast, Graph, List, Compute, Discount, Compare, etc. "Judgement" in effect determines the implications of each output from the previous steps and selects the next one. The DSS needs to be at the same level of operation.[9] Models are often too aggregate in nature and do not allow the manager to choose and respond to each substep. They thus impose a structure and impede flexibility.

Identifying user verbs in order to define DSS commands in no way results in a complete definition of the task. It merely classifies key activities; the manager links the activities together. (Again, in a structured task, we can get a full definition, since there is a set of procedures for linking the activities independent of human judgement). This strategy for analyzing a managerial task rests on a simple view of decision making.

There are two main categories of user verbs:

(1) Generic - These are used in decision making tasks. Every DSS, for example, needs a GRAPH and TABLE command, corresponding to the user verbs commands "Give me a graph of..." and "Get a listing/report showing...".

(2) Special Purpose - Specific to a given application

In general, decision makers seem to have only a few "priority" special-purpose verbs. If indeed managers are sensitive to the marginal economics of effort and try to simplify a problem to make it soluble, we can expect that they will usually rely on a few key heuristics. If they satisifice - look for good enough solutions - they will use a strategy that gets them close to a decision quickly.

Paul Berry illustrates both the simplicity of most decision maker's verbs and ease with which DSS can be built once they are found. He describes working with an economist whose job involved commoditys futures. The economist explained how complex the task is, the need for experience, etc., etc. He had to forecast the commodity prices, which fluctuated rapidly. While there was some value to analyzing historical data, he had to weigh recent trends more heavily.

Berry, one of the original developers of APL and among the most creative commentators on how to exploit it, concluded that the economist's activities relied on the verb SMOOTH: fit an exponential smoothing curve to historical data. The APL function is SMOOTH and the data items are simple vectors (time series) such as PRICES. Of course, such a simple DSS addresses only part of the task but:

(1) it is the key part

(2) the DSS provides a valuable tool at low cost

(3) is supportive, in that it meshes exactly with what the economist is now doing.

The DSS also does it better: more efficiently, in that the computation is done quickly, and painlessly, and more effectively since the economist can now try out various ways of fitting the data to a curve and look at outliers. The verb is SMOOTH; the DSS implementation of that command can at the same time support and extend the economist's activity. It supports by linking into his existing problem-solving strategy.

It can be extended by improving that strategy.Using the initial version of SMOOTH is likely to alert the economist to the fact that there are several types of possible trends and curves. Exponential smoothing is conceptually and empirically more valid than a linear fit, and in some instances Box-Jenkins or adaptive forecasting techniques may be most suitable.

DSS are a vehicle for learning. A recurrent theme is case studies is the unanticipated and creative ways managers extend either the DSS itself of their use of the current version. Figure 4 gives some examples. A DSS is evolved by adding new commands. The manager learns by adding new concepts or techniques. The point made easily needs reinforcing here: if the new commands do not correspond to verbs - "do this" - and vice versa, it will be hard for individuals to invoke the DSS functions and mesh what is in their heads into the DSS dialog.

The use of verbs as a base for productivity analysis and DSS design was the explicit basis for developing ISSPA. The "middle-out" version of ISSPA took 70 hours to build and contained 22 commands. It was developed by this author (Peter G.W. Keen), T.J. Gambino, and D.G. Clark. Keen and Clark had spent a year interviewing policy anaylsts and examining the use of computer systems and models as part of a research project aimed at improving the quality of analytic methods in school finance policy making. This prior study was invaluable since knowledge of the application environment is central to adaptive design. One cannot improve something one does not understand.

We identified some obvious general-purpose verbs for ISSPA: eg., LIST, RANK, DESCRIBE (descriptive statistics), HISTO (gram). Most of these are used in almost every DSS. We therefore checked to see if we could take the APL routine directly from a public library. While we generally have to modify the dialog to make such borrowed functions usable, we have found that virtually any graphical or statistical routine (repression, analysis of variance, cross tabulations) can be found in a public library and integrated with ISSPA in two to eight hours. Similarly, there is no need to write most functions for a DSS for financial analysis; they already exist (lease/buy comparision, net present value, internal rate of return, compound interest, etc.,).

We identified 30 special-purpose verbs for potential inclusion in the initial version of ISSPA. They were of two types:

(1) for policy analysis - eg., COUNTIF (how many units meet a specified condition?), REGRESS (multiple linear regression, the main technique used for forecasting and evaluation), NTILES (divide a distribution of values into n equal groups; this is particularly important in assessing redistribution effort of legislation)

(2) for school finance - eg., EQUITY (equity measures), GINI, LORENZ (measures of income distribution).

We put priorities on the commands. Some were essential to the user, either to support existing key activities or to extend them. For example:

(i) RANK-(prepare a report with units shown in ascending or descending order, ranked on a key variable.) This supports the analyst's current activities and also improves efficiency. The analysts rely on rankings in evaluating the existing and proposed legislation and had to rely on manual procedures or cumbersome inflexible computer programs.

(2) NTILES and WTILES(weighted NTILES) - 5 NTILES gives quintiles, 10 NTILES deciles, etc. This supports and improves effectiveness. Legislators and judges (almost every state has been involved in major school finance cases in the past decade) often compare, say, the top 10% and bottom 10% of the school districts. These simple commands extend the sophisication and range of analysis by making it easy for example to add weighting factors (WTILES).

(3) EQUITY - extends and increases efficiency. This command provides 11 measures of equity, which previously could be obtained only via special purpose and/or hand calculations. The marginal economics of effort now encourage the analysts to extend their analysis from simple rankings and NTILES to systematic evaluation.

(4) BOXPLOT, STEMLEAF, CONDENSE - extend and improve effectiveness. These are routines for exploratory data analysis (Tukey) that represent new strategies and techniques to those trained in classical statistics.

The rule we used and still recommend for selecting verbs to implement is: Support First, Extend Later. In practice this means that any verb, such as RANK or NTILES which the analyst either relies on now or views as a useful addition should be given first priority. At the very least, doing so improves efficiency. RANK does nothing new but substitutes a single command and a 2 second wait for what may be hours of work.

More often, the support aids effectiveness by directly improving efficiency. Analysts would often like to calculate additional variables and then look at the distribution effects but simply do not have the time to do so. NTILES, which took ten minutes to program, allows them time and thus facilitates a broader analysis.

If the initial version of a DSS is supportive, individuals will use it. We found that the analysts themselves soon asked for new facilities in ISSPA. Two commands, which significantly add to effectiveness were initiated and entirely designed by users. Of the 53 commands currently in the system:

(1) 9 were added at the user's request

(2) An additional 2 were defined by users

(3) 4 others were added by the designers specifically to extend the quality and scope of analysis

(4) 6 of the initial commands were enhanced in response to user's reactions.

This pattern seems typical of DSS usage and evaluation. It has some clear implications for efforts to improve productivity:

(1) the need to support first seems key; potential users are very concerned with usability and any system whose logic, style and dialog is too far from their experience is unlikely to be accepted.

(2) users are interested in imporving their performance and they

themselves will play an active role; the initial DSS is a catalyst and a framework for this.

(3) the verb command link interprets productivity analysis and productivity improvement.

5. The Impact of DSS

The traditional efficiency-based concept of productivity implies quantifiable methods of measuring gains. The view presented here does not. How do we measure better planning? What is the ability to do ad hoc analysis worth? What quantifiable savings does the use of WTILES or EQUITY result in?

The goals of Decision Support are limited. It assumes that most managers want to do the best they can, subject to constraints of time and effort. They make adequate decisions in most instances and are the best judges of how to get better ones. They naturally resist unfamiliar methods. Their jobs are complex and no DSS is likely to eliminate the need for judgement, imagination and hard thinking.

If these assumptions are accurate, and better productivity means doing a better job, we are unlikely to find quantifiable benefits. 70% of the IFPS users surveyed have no "hard and objective" evidence of its value. Traditional cost-benefit analyses seem rarely used for DSS ventures.

Instead, managers seem more interested in "value". Rather than comparing benefits and costs, sponsors of DSS first look at the qualitative benefits and <u>then</u> at cost. Until value is established, any cost is disproportionate. An advantage of middle-out prototypes is that they make DSS development a low cost R & D venture and not a capital investment. The initial system costs less than $10,000. If it turns out to be of no value, it can be written off. If it is seen as useful, management has a better sense of what must be spent to extend it.

Organizations, innovations are generally value-driven, not cost-driven. There is ample evidence that new technologies are adopted because they solve a visible problem and that cost-benefit trade-offs are irrelevant.[10] This point is important for DSS since the traditional, efficiency-based view of productivity shares many of the assumptions of cost-benefit analysis, Decision Support shares those of R & D innovation.

That said, even if DSS do not necessarily result in quantifiable, tangible benefits, they do provide identifiable ones. The following list is derived from about 30 DSS studies. Only benefits mentioned in at least five case studies are included. A few typical illustrations or quotes are given for each category on the list:

(1) <u>increase in the number of alternatives examined</u>:

- sensitivity analysis takes 10% of the time needed previously
- 8 detailed solutions generated versus 1 in previous study
- previously took weeks to evaluate a plan; now takes minutes, so much broader analysis

- users could imagine solutions and use DSS to test out hypotheses
- "no one had bothered to try price/profit option before"

(2) <u>better understanding of the business</u>

- president made major changes in company's overall plan, after using DSS to analyze single acquisition proposal
- DSS alerted managers to fact that an apparently successful marketing venture would be in trouble in six month's time
- DSS used to train managers; gives them a clear overall picture
- "now able to see relationships among variables"

(3) <u>fast response to unexpected situations</u>

- a marketing manager faced with unexpected budget cut used the DSS to show that this would have a severe impact later
- helped develop legal case to remove tariff on petroleum in New England states
- model revised in 20 minutes, adding risk analysis; led to reversal of major decision made 1 hour earlier

(4) ability to carry out ad hoc analysis

- 50% increase in planning group's throughput in 3 years
- "the governor's bill was published at noon and by 4 p.m. I had it fully costed out"
- "I can now do QAD's - quick-and-dirties"
- system successfully used to challenge legislator's statements within a few hours

(5) new insights and learning

- quickened management's awareness of branch bank problems
- gives a much better sense of true costs
- identified underutilized resources already at analyst's disposal
- allows a more elegant breakdown of data into categories heretofore impractical
- stimulated new approaches to evaluating investment proposals

(6) improved communication

- used in "switch presentations" by advertising agencies to reveal shortcomings in customer's present agency
- can explain rationale for decision to investment clients
- improved customer relations
- "analysis was easier to understand and explain. Management had confidence in the results."

- "it makes it a lot easier to sell [customers] on an idea"

(7) control

- permits better tracking of cases
- plans are more consistent and management can spot discrepancies
- can "get a fix on the overall expense picture"
- standardized calculation procedures
- improved frequency and quality of annual account reviews
- better monitoring of trends in airline's fuel consumption

(8) cost savings

- reduced clerical work
- eliminated overtime
- stay of patients shortened
- reduced turnover of underwriters

(9) better decisions

- "he was forced to think about issues he would not have considered otherwise"
- analysis of personnel data allowed management to identify for the first time where productivity gains could be obtained by investing in office automation
- increased depth and sophistication of analysis
- analysts became decision-makers instead of form preparers

(10) more effective team work

- allowed parents and administrators to work together exploring ideas
- reduced conflict: managers could quickly look at proposal without prior argument

(11) time savings

- planning cycle reduced from 6 man-days spread over 20 elapsed days to 1/2 a day spread over 2 days
- "substantial reduction in manhours" for planning studies
- [my] time-effectiveness improved by a factor of 20"

(12) making better use of data resource

- experimental engineers more ready to collect date since they knew it would be entered into a usable system
- "more cost-effective than any other system [we] implemented in capitalizing on the neglected and wasted resource of data
- allows quick browsing
- "puts tremendous amount of data at manager's disposal in form and combinations never possible at this speed"

Figure 7 summarizes these categories. They add up to a concept of productivity. It is these often qualitative aspects of effectiveness that managers value. The operating assumption of Decision Support is that improving communication, flexibility, learning and responsiveness leads to better decision making.

FIGURE 7 DSS Benefits

1. increase in number of alternatives examined
2. better understanding of the business
3. fast response to unexpected situations
4. ability to carry out ad hoc analysis
5. new insights and learning
6. improved communication
7. control
8. cost savings
9. better decisions
10. more effective teamwork
11. time savings
12. making better use of data resource

computers, since the technology is used to routinize, standardize and control. It ignores the fact that computers can be used in the opposite way, to increase flexibility, to allow ad hoc response, and to augment jobs.

The balance between efficiency and effectiveness is complex. For managerial work, effectiveness is the main issue. What Decision Support offers to productivity research, albeit in a fragmented and informal fashion, is a broader conception of "productivity," some simple and practical ways to look at complex jobs (productivity analysis) and evidence that computer systems can be built, at relatively low cost, that managers will find helpful and valuable (productivity improvement).

One cannot improve what one does not understand. Improving the productivity of managers can only begin by looking at their work from the inside. We need a descriptive perspective on managerial work as well as a prescriptive one. It is surprising how little emphasis there is in the literature on productivity on managers and professionals. Experts want to increase the output of a labor force they admit is often alienated and uninterested in its work. Managers, as a whole, are motivated and fairly responsive to challenges. They are a scarce resource and their performance can be given leverage by the application of suitable tools. Decision Support provides some of that leverage.

FOOTNOTES

1. Keen & Scott Morton & Alter provide detailed case studies of PMS, GADS, PROJECTOR, BRANDAD, IMS, CAUSE and AAIMS. ISSPA is described in Keen & Gambino.

2. Ackoff & Sasieni, p. 443.

3. see Keen, Decision Support Systems and the Marginal Economics of Effort for a more detailed discussion.

4. see Keen, Decision Support Systems: a Research Perspective.

5. IFPS is a proprietary product marketed by Execucom, Austin, Texas.

6. see Wagner.

7. see Keen & Gambino.

8. Some DSS researchers recommend a "table-driven" system. The latter uses menus to remind the users of the range of options available.

   The command-driven approach is a little more flexible but the table-driven DSS requires minimal knowledge and memorizing by the user.

   The "verb-based" strategy is as applicable to table-driven DSS as to command-driven ones.

9. see Keen, Decision Support Systems: A Research Perspective.

10. see Keen, Decision Support Systems & Value Analysis.

REFERENCES

Ackoff, R.L., "Management Misinformation Systems." Management Science, vol. 14, no. 4. pp. B 147-156, Dec. 1967.

______, and M.W. Sasieni. Fundamentals of Operations Research. New York: Wiley, 1968.

Alter, Steven, Decision Support Systems: Current Practice and Continuing Challenges. Reading, MA: Addison-Wesley, 1980.

Berry, P., "The Democratization of Computing." Paper presented at Eleventh Symposium Nacional de Systemas Computacionales, Monterrey, Mexico, March 15-18, 1977.

Cyert, R.M., and J.G. March. A Behavioral Theory of the Firm. Englewood Cliffs, N.J.: Prentice-Hall, 1963.

Gorry, G.A., and M.S. Scott Morton. "A Framework for Management Information Systems." Sloan Management Review, vol. 13, no. 1, pp. 55-70, Fall, 1971.

Keen, Peter G.W., "Decision Support Systems & Value Analysis." CISR Working Paper forthcoming.

______, "Decision Support Systems: A Research Perspective." CISR Working Paper No. 54, Cambridge, MA: Sloan School of Management, M.I.T., March, 1980.

______, "Decision Support Systems and the Marginal Economics of Effort." CISR Working Paper No. 48, Cambridge, MA: Sloan School of Management, M.I.T.

______,and T.J. Gambino. "Building a Decision Support System: The Mythical Man-Month Revisited." Building Support Systems. Reading, MA: Addison-Wesley forthcoming.

______, and M.S. Scott Morton. Decision Support Systems: An Organizational Perspective. Reading, MA: Addison-Wesley, 1978.

Mason, R., and E. Swanson. Management Measurement. Reading, MA: Addison-Wesley forthcoming.

Ness, D. "Interactive Systems: Theo[illegible]s of Design." Joint Wharton/ONR Conference - Interactive Inform[illegible]ion and DSS, Dept. of Decision Sciences, The Wharton School, University of Pennsylvania, Nov. 1975.

Ross. The Intuitive Psychologist. McGraw-Hill, 1980.

Simon, H.A. The New Science of Management Decision. New York: Harper & Row, 1960.

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

Taylor, R.N., "Psychological Determinants of Bounded Rationality: Implications for Decision-Making Strategies." Decision Sciences 6 (19), pp. 409-429.

Tversky, A. and D. Kahneman, "Judgement Under Uncertainty: Heuristic and Biases." Science 185, September, 1974.

Wagner, G.R. "Realizing DSS Benefits With The IFPS Planning Language." Paper Presented at Hawaii International Conference on System Sciences, January 3/4, 1980, Honolulu.