1.5.1.5. DSS VIS-A-VIS EDP, MIS, AOSS, AND MSS

Systems to support decision making include many tools, of which computer-based tools identify but one type of container for tools.

The functional groups of tools, singly or together, can be manual, automated, or computer-based. Standard definitions for concepts like management information systems include an embodiment in a computer. In contrast, I consider these concepts broadly enough to include manual embodiments and place the concepts within the framework of the Management System Model (MSM) so you can get a fixed, repeatable scope of the meaning of each concept. Figure 1.4.2.6.3. illustrates my view of what a decision support system (DSS) is, a view in terms of supporting decisions rather than a level or generation of hardware or software for a computer system. In Figure 1.4.2.6.3., the five functional groups of tools comprise a DSS when they are interrelated so they work together based on the same data and leading toward the same objective. I view other similar concepts (e.g., management information system, electronic data processing, and management support system) from the same perspective—tools, not necessarily computer-based, that we use to manage with.

The phrase "decision support" first appeared in a landmark paper by Gorry and Scott Morton in 1971. (*A Framework for Management Information Systems,* Sloan Management Review, Fall 1971, pp. 55-70.) Since then the term has come to be a "warmed over" and more palatable term for management information system, all of which has come to inherently include the concept of computer based. I agree generally with John D.C. Little in his TIMS Letter from the President in the April 1985 where he said "A certain amount of confusion has occasionally occurred about whether DSS is just OR/MS [Operations Research/Management Science]. Who cares? The business of management science is decision support. The computer is our chief delivery vehicle." I don't agree that the computer is now the chief delivery vehicle, but it will be.

I draw attention to how each such concept fits distinctly within the Management System Model; that is , the role each plays in helping us manage. Figures 1.5.1.5.1. and 1.5.1.5.2. show five such concepts in two diagrams, both representing an overlay of the concepts in the MSM.

Figure 1.5.1.5.1. includes data-information-related concepts and compares electronic data processing (EDP), management information system (MIS), and automated office support systems (AOSS). Figure 1.5.1.5.2 includes decision-related concepts and compares decision support systems (DSS) and management support systems (MSS). Within both diagrams, all concepts are defined and contrasted. My definitions are top-down, management-related concepts. They don't agree with computer-oriented definitions but are naturally distinguishable one from the other.

EDP Deals Only with Data.
Electronic data processing (EDP) is the first of several links of the data-to-information chain, shown in Figure 1.5.1.3.7. With EDP we merely access, store, retrieve, and manipulate data, a function well suited to clerical-level and operational-level endeavors. Before computers, we did the data processing function well using hand-generated spreadsheets for manipulation and notebooks or file cabinets for storage. Computers are able to duplicate

this function faster and more consistently. One disadvantage is that since we now can do data processing so much faster, we can do so much more of it. So we produce the same data, manipulated into dozens or hundreds of different spreadsheets or tables.

I consider narrative to be data. Therefore, word processors are computer-based EDP devices. When you think about it, state-of-the-art networking, spreadsheeting, and windowing practices are nothing more than manipulating data, since moving data from place to place is a form of manipulation. EDP is important to consider because that is most of what we are really doing under the guise of MIS or DSS.

MIS Is More Than Manipulating Data.
When I look at the technological advances we are so excited about today, I see us manipulating data, not making information. Networking is moving data from place to place, windowing is displaying data, and spreadsheeting is aggregating data. Either by moving it, showing it, or tallying it, we are getting better and better at just manipulating raw data, not enhancing its value or truly supporting management.

The management information system (MIS) was coined to represent a more-useful, higher-level form of management support using information rather than just data. Unfortunately, when MIS was required to do much more than EDP (as I define it), MIS failed. That is, MIS may have been a new term, but it did little else than raise managers' expectations and sell a lot of computers. (Those of us in the information business will pay for that.)

MIS, as shown in Figure 1.5.1.5.1., is the entire data-to-information chain and includes not only the EDP links, but the links for forming and presenting information. MIS never has adequately addressed the measurement/ data and the information portrayal/information perception interfaces. In terms of the individual links in the chain, we've developed the hardware and software specialties far beyond the ability or need of most of us to fully use them. We've recognized the garbage-in/ garbage-out syndrome at the data-to-measurement interface, but the operational aspects of this interface (e.g., data administration and corporate data sharing) are still quite primitive.

I believe the area for greatest contribution lies at the information-portrayal-to-information-perception interface and learning how to portray information with purpose rather than at random. Since information can be portrayed in four different formats (table, graphic, checklist, and text), the right format can be selected depending on the data and information structure and the desired conclusion. In very few cases have we ever achieved a successful MIS as defined here.

An MIS is much more than a computer. As we become involved with microcomputers we learn how true this is. In our work, we focus on getting hardware and software and, as shown in Figure 1.5.1.5.3., celebrate its arrival. Then we realize the need for training, documentation, back up, security and the host of other operational needs that together make up an MIS.

AOSS Includes Part of the What Is Managed Component.
In that an automated office support system (AOSS) is generally considered to include office tools (part of what is managed), my concept of AOSS includes MIS, most if not all of the methods, and the office tools. As opposed to including only word processing, scheduling, and teleconferencing, many of us see AOSS as the automation of all office-related efforts typically under the jurisdiction of a single office environment. There's no need to quibble about whether rolodexes or lighted slide cabinets are part of automation,

but these tools along with models, paperwork, and the like and the office's MIS all support office decisions based on information.

The DSS Comprises all Management Tools through their Interrelatedness.

I believe decision support systems (DSS) arose as a response to bad feelings about MIS. Given that MIS hasn't progressed much from EDP, why should DSS suddenly be able to accomplish more than what was originally expected of MIS? What is the difference between information for managers and support for decisions in the real world where we are doing neither? By my definition, DSS is quite different from MIS and includes all the tools of the what is used to manage component; and DSS does address the measurement-to-data and the information-portrayal-to-information-perception interfaces.

The key to DSS is the synergism that results from the tools working well together. Thus, DSS focuses on the interrelatedness of the tools. The methods category of management tools should affect the plans in the guides and rules category. The plans should be used as much as the data-to-information chain because these two tools should be used hand-in-glove in formulating the reference points so we can execute against them using our MIS. Often, in an effort to "obtain computerized decision support," managers gain such a good understanding of these interactions, the need for computerization (or automation) is lessened because the manager has systematized what he uses to manage.

The term DSS was first used by Gorry and Scott Morton in 1971. DSS was to support Gorry and Scott Morton's semi-structured and unstructured decisions, which are described in Module 1.4.5.5.1.

Gorry and Scott Morton say, "We shall call the information systems that support [decisions that are largely structured] 'Structured Decision Systems' (SDS). Decisions [that] are largely unstructured, and their supporting information systems are 'Decision Support Systems' (DSS). The SDS area encompasses almost all of what *has* been called Management Information System (MIS) in the literature—an area that has had almost nothing to do with real managers or information but has been largely routine data processing. We exclude from consideration here all of the *information handling* activities in an organization. A large percentage of computer time in many organizations is spent on straightforward data handling with no decisions, however structured, involved. The payroll application, for example, is a data handling operation." (p. 61.) In terms of semi-structured and unstructured decisions, we haven't come very far in 15 years.

MSS Includes The Who Manages Component.

My definition of management support systems (MSS), a concept introduced by Katzan in 1984, is different from Katzan's. Katzan's definition of MSS and my definition of DSS are nearly equivalent. I define MSS as a system that combines the who manages and what is used to manage components of the MSM, thereby including everything from the measurement-to-data interface to the decision-to-action interface.

In short, the DSS supports who manages, or the decision maker, and MSS supports what is managed, or the operation. Since who manages is considered to be a human decision-maker, with his or her cognitive style, experience, and capability and such human traits as emotions, ambitions, unpredictability, and sense of humor, an MSS, by definition is not easily *fully* automated. My view of an MSS explicitly recognizes that a manager is more than a decision-maker; he or she is a person, above all.

Consider, for example, a system where the entire process, from measurement to data to information portrayal and perception, and, finally, decision-making and action is automated. Computer-based systems such as these exist—for example, we have systems which keep track of inventories and if the level falls below a prescribed minimum, automatically place an order for more items. Should we consider such a system to be a fully automated MSS? No. Note that the computer doesn't replace the "who manages;" it only automates routine actions permitting the who manages to attend to the non-routine tasks. Therefore, the system is a management tool. If anything were to upset the routine, the system described above would need the intervention of a human manager. At least until computers can perform a broad range of routine and non-routine decision-making and be endowed with human characteristics, we'll stick with my definition.

Whether we use the above definitions or the somewhat blurred definitions commonly used, EDP, MIS, AOSS, DSS, and MSS are all portions of a management system. The MSM includes all components and interfaces to close the loop.

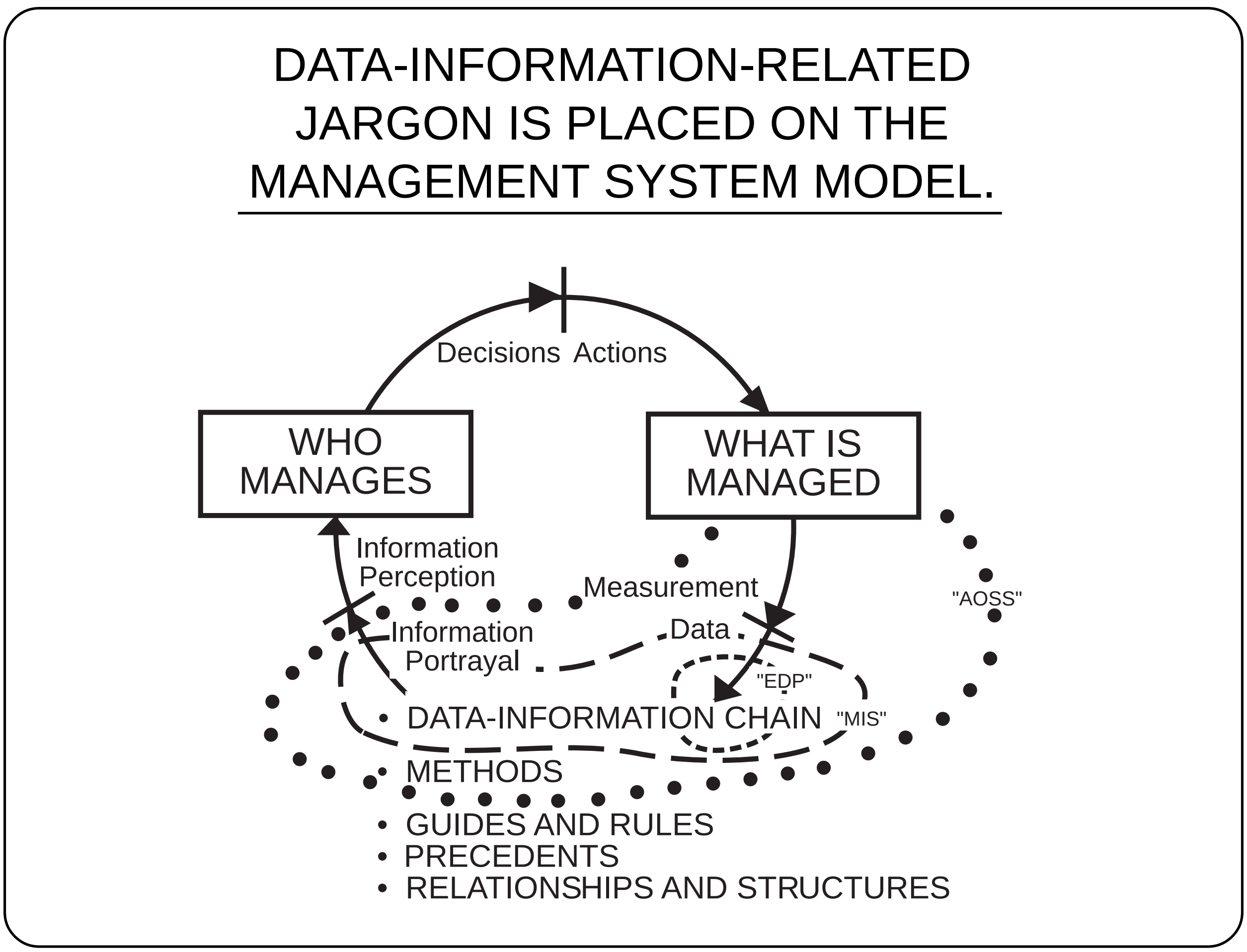

Figure 1.5.1.5.1. *Data-and-information-related concepts are easily defined using the Management System Model.*

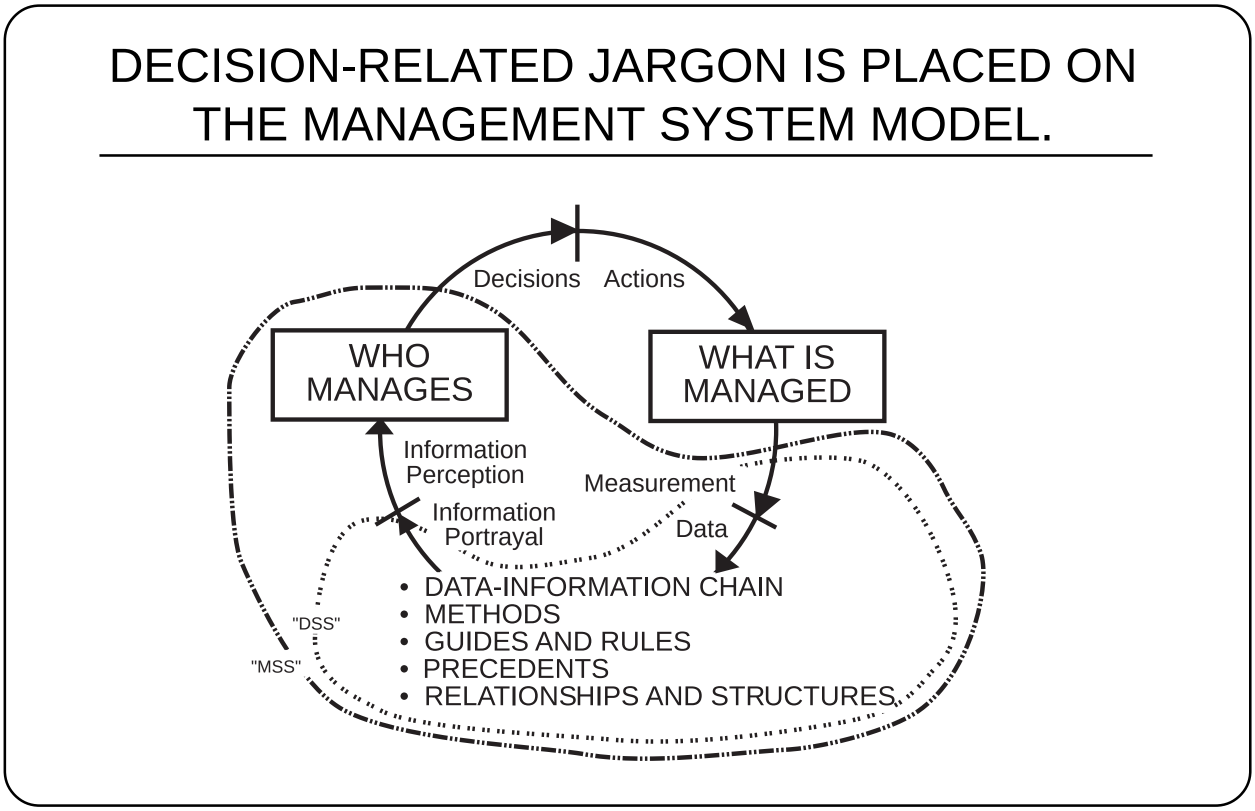

Figure 1.5.1.5.2. *Decision-related concepts are easily defined using the Management System Model.*

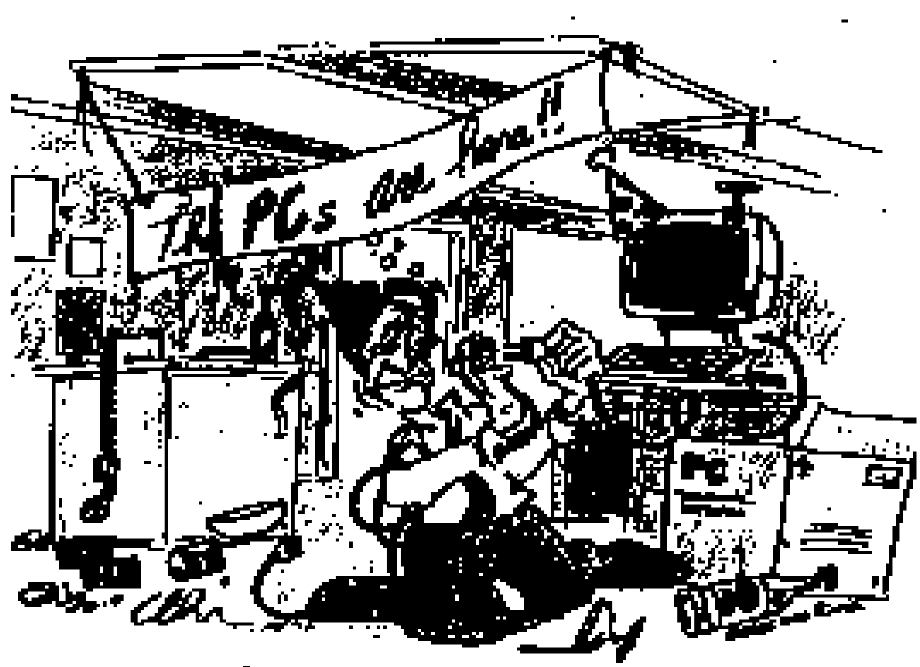

Figure 1.5.1.5.3. *"The PC's have come, the PC's have come."*

1.5.1.6. What Types of MIS Really Help You?

Make sure you know what an MIS gives you in supporting your decision making and takes from you in time and effort.

With office automation, a professional secretary working for a manager with mostly strategic-level or tactical-level endeavors is set free from clerical chores to do more judgmental work, and the resource pool isn't needed. At judgmental levels, automation makes the job more demanding. At the perfunctory levels, automation makes the job less demanding. If your work is more judgmental, you will end up with a new kind of office assistant. This person will be able to use office automation tools more deftly and will provide you the information you need when you need it, much as your secretary does now, only the new office assistant will require much more expertise.

Many of you may think that as you integrate automation into your offices, one or more secretaries will leave. That is an illusion.

At Virginia Tech we have a university-wide information system designed and developed by accountants for accountants. That system is useless to people like department heads and principal investigators in research because currency is sacrificed for preciseness and the outputs are intelligible only to accountants. Therefore, we decided to plan a system to satisfy everyone’s needs. What a wonderful system it is supposed to be.

In a committee meeting after hearing about the depth and the breadth of the new system, an associate dean asked the question, “Do we have any idea how much this is going to cost?” Among the startled people who had been dreaming of the promised land, one said, “However, we must balance the cost by the savings this system will bring.” I laughed because I knew in the university the savings that count are in reducing people, space, or equipment and that ultimately the system would require more (but different) people, space, and equipment.

Has the copy machine reduced the work for the secretary or increased it? Automation doesn’t mean you do the same things with fewer people—it means you do more or different things. Given a resource pool with a word processor, you write more letters. In your offices, you probably won’t replace people—you'll replace functions.

I’ve often said that strategic-level managers won't have computers on their desks. I say this for emphasis. What I mean is strategic-level managers will not program computers (including LOTUS 123) or manipulate programs. They'll bring up displays, review them, input notes or comments, and do simple “what-if” analyses. Without the new kind of office assistant, this will be very difficult.

As stated earlier, a successful computer-based management information system (MIS) can help satisfy the need for balance in the management system model (MSM). Computer-based MIS failures include any combination of those that aren't finished in time, involve turnover of key participants, exceed reasonable cost before any return, lack follow-through, become obsolete through obsolete requirements or obsolete technology, have an improper fit, or have not had their performance evaluated. I claim that we’re only 30% successful.

I helped design and develop a milestone tracking system for one of our government sponsors. A rousing success, the computer-based package provided reports used monthly for milestone meetings and annually for performance evaluation. The package was quick, easy, and comprehensive. The manager had exactly what he liked and his organization grew to depend on the package. After a year, the manager was on the package. After a another year, the manager was transferred across the country, his deputy took his place, and the organization happily continued with the familiar practices.

A new person was brought into the organization to replace the deputy and he was given responsibility for milestones. The "successful" system didn't suit his preferences. From the management system perspective, that tool failed. It didn't do what it was supposed to do for as long as the user expected the package to do it. Successful software, hardware, and experience doesn't make a successful system. A generic package would not suit the first who manages, the original manager. The custom-tailored package did not suit the third who manages, the second deputy manager. Both ultimately failed in fit.

We designed and developed a research project financial management system for one of our department heads in engineering. His department, with 40-50 research contracts, had long been suffering from overruns due to late or faulty information from the university system. His new system worked well. His secretary, who was afraid of computers, had no trouble using this computer-based system, and overruns were eliminated for four years. Then one day I got a call from the department head. He had overruns everywhere. What had happened to this "successful" package?

After four years, the computing center had made a change in the operating system that affected our package. When the secretary tried to update her database, the terminal screen went "into bozo-land." Discretion being the better part of valor, she decided that she would worry about the problem after her vacation, which began in a couple of weeks. Of course, her vacation didn't improve the package and she had the same problem when she returned. However, this time she could put off the inevitable until after she returned from her surgery, which would take place in a few weeks. Again, upon her return she found no improvement in the package. She finally called my system designer, who would fix the problem very soon. But—he was hired away by another contract sponsor and didn't transfer information to co-workers either in the problem or its solution.

To show the length of the long story, the problem was nine months old when the department head called me. Even the data in the database were so old we no longer had an updating problem but a reinitiation problem on top of the simple programming solution. A classic failure. No follow-through. Good hardware, software, training, and documentation. But no procedures to identify, notify and resolve a simple operational problem.

Some MIS's Do Succeed.

By hiding from our failures, we aren't able to discover reasons for our successes. If an MIS fails because the new concept of the management systems isn't understood and the system components aren't balanced, then why do the few MIS's succeed?

An examination of successful MIS's reveals three commonly used development strategies that balance the MSM components. I'll also discuss a fourth strategy, which results in illusory success.

First, a computer-based MIS can be successfully implemented by merely automating an existing successful manual system. In developing and using the manual system, the manager has intuitively balanced the management system components. The manual system generates the information he or she wants for making decisions affecting what he or she manages. However, manual systems limit the manager's ability to get and frequently change large amounts of data. Automating the manual system removes its limitation, thereby improving the management system balance. Most MIS successes are of this type.

Second, a manager can get so excited about MIS development that he or she devotes inordinate amounts of his or her time and energy working with the systems designers to insure a custom-fit between the MIS and his or her needs. In this case, the manager demands what he or she likes and persists until the management system components come into balance. This strategy is effective, but inefficient because managers usually have limited knowledge of the features and benefits of automation techniques.

Third, for some common specific needs involving structured decisions, a range of computer-based MIS packages has been developed. These packages have broad application and markets. Accounts receivable packages are an example. Often, a package can be found which fits the manager's specific need well enough that the MSM components are *nearly* balanced. However, failure is almost certain if, lured by the apparent potential for a perfect fit, a package with near-fit is obtained and attempts are made to customize the package.

Sometimes, in an effort to get aboard the computer bandwagon or to shore up management inadequacies, some managers will force the components of their management system to fit an improperly developed MIS or an off-the-shelf MIS package. The new management system, forced to fit the MIS, may appear to be balanced, giving an illusion of success. However, because of the forced fit, the new management system probably no longer addresses the original management system goals and objectives. The computer-based MIS may appear successful while the entire management system fails. Such an illusory success portends dire long-term consequences for real MIS success.

As shown in Figure 1.5.1.6., automated airline reservation systems provide an interesting example of force-fitting who manages into a computer-based MIS, that may (or may not) fit what is managed. The reservationists are forced onto the system, and they either adjust or terminate their employment. Often, poor fits causes poor performance.

At a minimum, people with management tools that don't fit are frustrated. In Figure 1.5.1.6., I've shown two of the three reservationists as upset. By chance the automated airline reservation system fits the experience, ability, and cognitive style of one of the reservationists. The figure reflects my proposition that 70% of all MIS's fail. Check my proposition out the next time you're in an airport.

Airlines have determined that the relative costs warrant the trade-off between improved speed or responsiveness to the customer, and loss in personnel. They haven't learned about quality. This trade-off may be considered reasonable for reservationists performing clerical, or at best operational, endeavors. But we certainly cannot make similar trade-offs where tactical-level or strategic-level managers are concerned. You won't successfully tell your boss to adjust to the system or leave.

The what is managed component of management systems changes mostly in terms of maturity, either with new technology or through improved decisions and actions. Therefore, the operation changes relatively less frequently.

The who manages component changes more often since frequent personnel changes are common in most organizations.

Given the low MIS success rate and little inclination to reduce the frequency of personnel changes, can we expect the MIS success rate to improve? Not soon! If we confine our efforts to automating manual systems, we may slightly improve the success rate. But this solution is impractical in light of the ever-pressing need for managers to deal with more rapidly changing data.

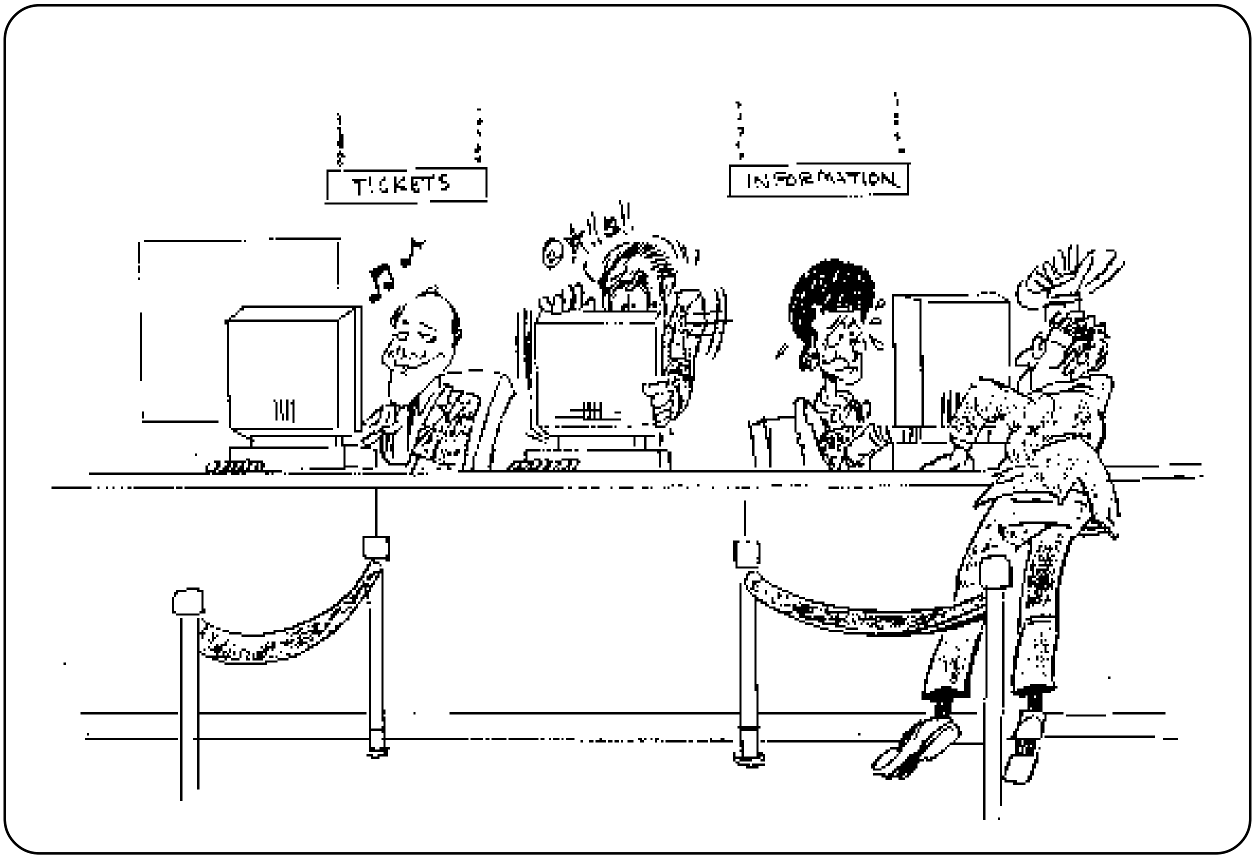

Figure 1.5.1.6. *"I have some reservations about this new computer system."*

1.5.1.7. We Need Responsive Systems.

We need systems that can understand what we need and give us just that.

I want to present a thought-provoking concern. Personnel changes are more than likely going to bring an existing MIS out of balance and cause it to fail. Therefore, to significantly increase the rate of MIS success, we must develop *adaptive* systems. To understand the profundity of this thought, I'll categorize successful systems as adaptive, adaptable, flexible, and, of course, custom-fit systems.

Adaptive Systems Are Self-Molding.
As in the bean-bag chair in Figure 1.5.1.7.1., *adaptive* systems are self-molding and adjust to who manages without being told. They will sense the user characteristics and adjust automatically. When a user touches the keyboard or just approaches in the vicinity of the management tool, it senses who is there and what he or she wants and then instinctively adapts to provide exactly what is needed (a very futuristic idea). The first glimmer of adaptiveness will be systems that adjust their menus and help-routines based on the user's knowledge or experience as monitored or sensed by the system.

Adaptable Systems Adjust When They're Told.
As in the electric car-seat in Figure 1.5.1.7.2., *adaptable* systems are less sophisticated; they adjust to who manages when they're told. System designers will determine what to tell the system and how to measure needed characteristics of the user.

An example of an adaptable feature in one of my systems (not an adaptable system) is a feature that we designed into a microcomputer-based simulation package. As the simulation is running and the numbers are changing on the screen for the manager to watch to determine trends and effects, the operator can change the speed of the presentation of the simulation. He or she can have the simulation proceed faster to get past slowly changing occurrences or slower to watch rapidly changing occurrences. He or she controls simulation speed by repeatedly using the greater-than or less-than key. Hitting the greater-than key speeds up the presentation of information; and hitting the less-than key slows down the presentation. He or she also can stop the simulation, change variables, or add processes and proceed.

Flexible Systems Have a Number of Options.
While we work toward these futuristic concepts we'll build *flexible* systems as illustrated with the lawn mower wheel adjustment in Figure 1.5.1.7.3. The lawn mower has several levels at which you can set the wheels. You get to pick one. Have you ever tried to set a wheel in between the choices? It doesn't work. Flexible systems provide a series (large or small) of fixed alternatives from which the user can choose. For instance, the user may select a graphical or a tabular format to view his or her information.

Custom-tailored Systems Fit A Specific Situation.
As in the tailor illustration in Figure 1.5.1.7.4., a system can fit the user if it is *custom-tailored* for him or her—even if he or she has a funny shape. The modern concept of windowing is a feature of a flexible system, not of an adaptive or adaptable one.

Responsive Systems Employ Artificial Intelligence.
The examples in the figures are responsive

systems not expert systems. A responsive system is an intelligent system, although not expert. We say a dog is intelligent if the dog performs routine tricks based on habit (See H.A. Simon.) learned though repetition. Responsiveness includes three characteristics: 1) timeliness—computer-based systems have shown this characteristic for years, 2) ability to observe and understand who manages and/or what is managed—the manager and the operation collectively constituting the user, and 3) the ability to interpret the user and eagerly and willingly carry out the user's wishes.

Responsiveness implies reasoning power and the ability to monitor and interpret; those are measures of artificial intelligence. The characteristic we call *user friendly* is the facade of responsiveness. The user *sees* user friendly but *experiences* responsiveness. Not only should people be responsive, but the decision support tools we use should be responsive to both who manages and what is managed.

All part of artificial intelligence, expert systems replicate an expert, responsive systems suit the user. Key features of expert systems include the abilities to make inferences and judgments, to deal with ambiguous and incomplete information, and to justify their conclusions with detailed explanation of their *reasoning* process. Expert systems won't do much for strategic-level endeavors, but responsive systems will. "The more expert the expert, the less logical his reasoning and the less he's able to describe how he works." (Robert Bernhard, Sr., Technology Editor, Systems and Software, July 1984).

Mike Sallada in my management systems engineering class connected the categories of systems to the endeavors framework described in Module 1.4.5.4.1. He wrote, "I think we are in a number of places on the evolutionary chain. Certainly we currently have and continue to develop custom-tailored systems. I think this is especially apparent in large main frame systems. I know the main frame was supposed to be dead, but just look at IBM's financial reports and it becomes apparent that the prognosticators were wrong. In smaller client server environments we frequently waiver between flexible and custom fit systems. Many times the off the shelf system has to be modified to fit the environment. Much of our development is currently in two areas, object oriented code and reusable code. Together they allow a system to be both flexible enough to be used in more than one organization, for more than one purpose; and the custom fit desired for each system. Adaptable systems seem to be the highest level we have currently approached for the mass market. With Windows 95 you can have the machine adapt to your preset requirements by your sign in password. I am not aware of any adaptive systems currently available, although we have discussed them in class. I think it is interesting that the further up the clerical-operational-tactical-strategic spectrum of the system; the lower we are in the range of custom-tailored, flexible, adaptable, adaptive spectrum."

In conclusion, we note that the cost of successful (adaptive, adaptable, flexible, or custom-fit) systems increases with sophistication and decreases as technology advances. Given the state of the art, the required sophistication, and short economic life caused by dynamic technology, right now we can't afford to be as successful as television commercials and computer zealots would have us believe.

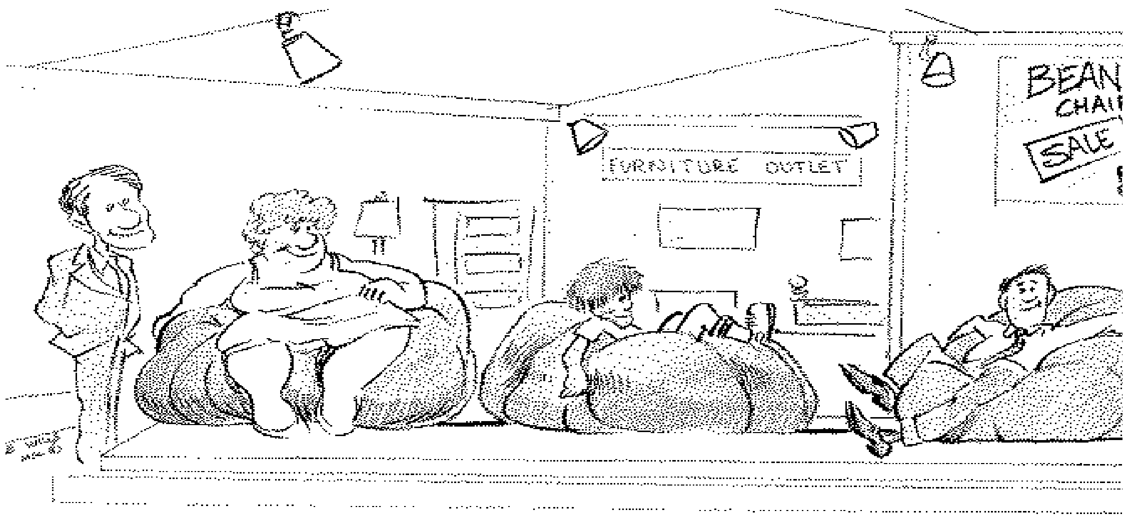

Figure 1.5.1.7.1. *The bean-bag chair inherently adapts to your body.*

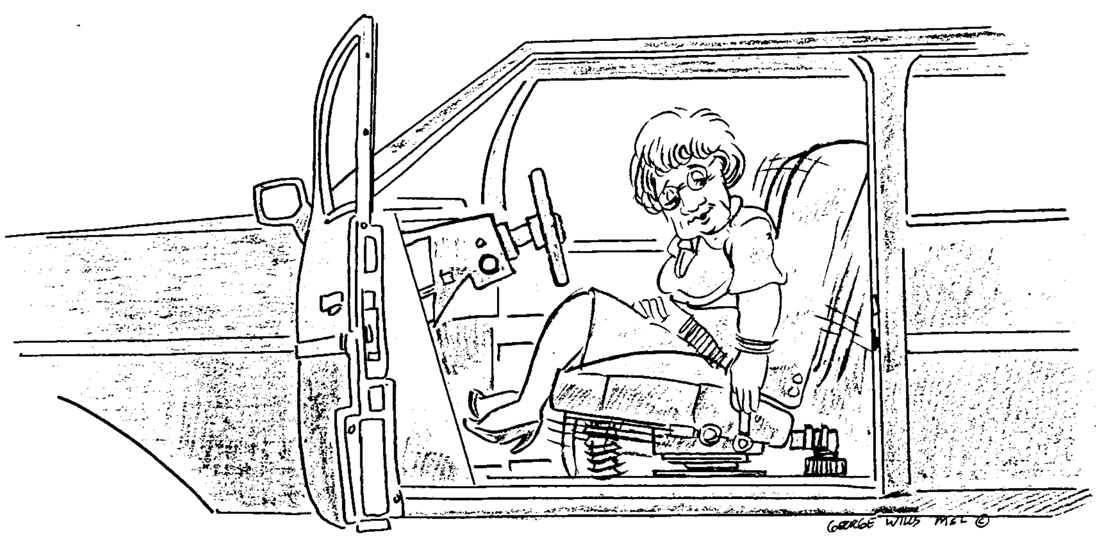

Figure 1.5.1.7.2. *The electric car-seat does exactly what you tell it to do.*

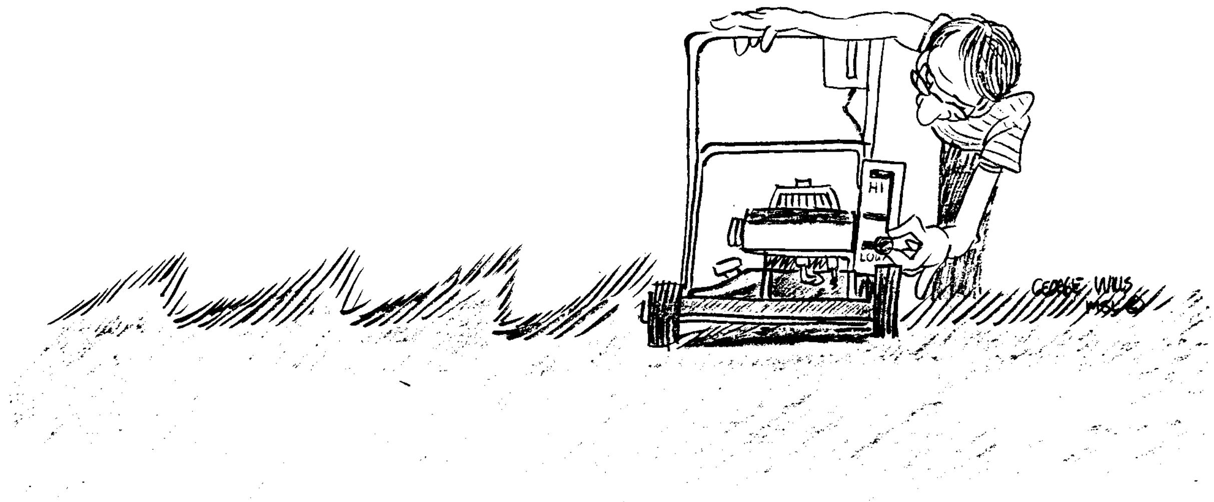

Figure 1.5.1.7.3. *You can choose from one of several options for lawn mower height.*

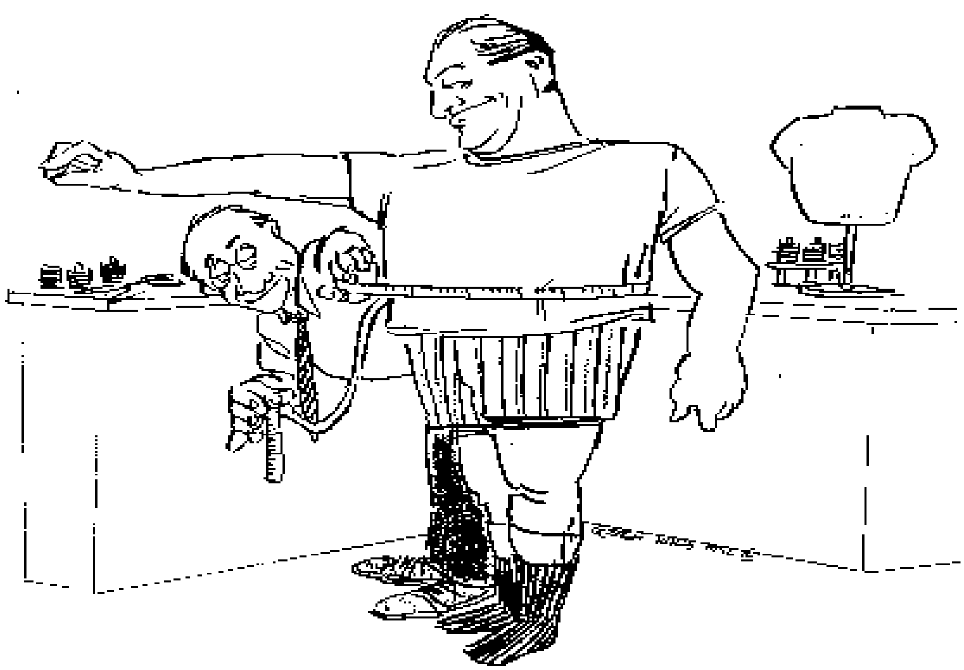

Figure 1.5.1.7.4. *The tailor finds the fit that's just right for you at the time he or she measures you.*

1.5.1.8. Exercise on Management Tools

To improve management tools, you must first figure out what tools are being used and how well they work; then you can determine what tools are needed.

Situation Description

Did you ever wonder where your electric bill went when you sent it to a post office box number? And why your check cleared your bank so fast? Your bill, with check, often goes directly to a bank. The bank sells a service called lockbox. The bank 1) receives the checks directly from the post office (picked up several times a day), 2) processes the checks (in just a few hours), and 3) deposits the money directly in their customer's account (in my example, the electric company). The bank charges the customer by the check processed and the information gathered (activity by date, financial information, etc.). The great advantage to the bank's customer (the electric company) is that lockbox gets your money into their account fast.

The lockbox unit in the bank employs 1) a management team; 2) many image scanning, letter opening, and other processing people; and 3) temporary workers for peak loads. Each worker has detailed procedures for operating their workstation. The management team sets yearly goals and objectives for the unit in an annual operating plan and for the individual in MBO (management by objectives) worksheets. Frequently during the day, at each station in the lockbox process, workers meticulously log work in, work out, and backlog. At the end of the day, the logs are collected and the data are transferred to a computer-based management information system (MIS). Management reviews the MIS first thing each morning with each worker individually. They recognize improved performance through monthly and annual celebrations. Once a year, management meets with each employee individually to appraise their performance against the MBO worksheets.

When a new management associate comes to work, he or she typically reviews and updates workflow charts and tally sheets that model the process. Based on these updates, they can verify they're measuring the right things at the right times and in the right ways.

Exercise

You've been hired to review the lockbox service with the objective of improving management tools. First, you must identify what the management tools are. Then, you must figure out how well the management tools are performing. Prepare a summary of what you expect to find and do.