A
DECISION SUPPORT SYSTEM
MODEL

by

Michael Wayne Farrell

B.S., Kansas State University, Manhattan, Kansas, 1975
M.S., Kansas State University, Manhattan, Kansas, 1979

A MASTER'S THESIS

submitted in partial fullfillment of the

requirements for the degree

MASTER OF BUSINESS ADMINISTRATION

College of Business Administration

KANSAS STATE UNIVERSITY
Manhattan, Kansas
1979

Approved by:

Raymond J. Coleman

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF DEFINITIONS

ACKNOWLEDGMENTS

I would like to thank the members of my thesis committee--Drs. Ray Coleman, Beth Unger, Richard Ruch, and Tom Brown--for their support and guidance over the past several months. I would like to express my gratitude to Mr. Gary Mundhenke of Farmland Industries for his time and cooperation used in verifying this work. A special thank you is extended to Lynn for her patience and understanding throughout this exercise. Last but definitely not least, Rhonda and Madi have my sincere appreciation of their talents and time used in completing the final copy.

I. INTRODUCTION

This thesis is based on the assumption that the new computer technologies will change the design and use of information systems in organizations.

> Definition 1.1 Information System
>
> An information system is a mathematical representation of the interaction of the components of an enterprise, i.e., inventory control, production, scheduling, shipping, accounting, etc.[26]

The increasing capabilities yet decreasing size and cost of computing processors and storage devices are relaxing previous constraints associated with information systems. Such constraints were the result of large physical size and high dollar costs. In the early applications of computers to information systems, a computer based information system could only be cost effective if a large number of users could share its cost. This philosophy of a central site computer system, shared among a number of the corporate divisions and functions, was carried into the corporate structure in the form of a data processing department with a centralized, management information system.

Definition 1.2 Management Information System

"A Management Information System (MIS) is a group of people, a set of manuals, computer programs, and data processing equipment to select, store, process, and retrieve data to reduce the uncertainty in decision making by yielding information for managers at the time they can most efficiently use it."[8]

This type of centralized system developed a trend of providing the manager with irrelevant information.[21] The irrelevant information was being provided in the form of routine management reports and computer print-out listings as a result of a structured, previously defined procedure.

In the early part of this decade, researchers were proposing that computer based information systems should do more than routine data processing and report generation.[2] Throughout the decade, some computer based information systems were designed to actually support the manager in the decision making process. Some of the systems have been studied and the recent results indicate that computer based decision support systems are achieving only limited success.[4,17]

Definition 1.3 Decision Support System

"A Decision Support System (DSS) implies the use of computers to: assist managers in their decision making processes in semistructured tasks; support, rather than replace, managerial judgement; and improve the effectiveness of decision making rather than its efficiency."[5]

There were also many technical developments in the computer industry during this decade. These developments related to both the computer hardware and software. The hardware developments included miniaturization of the processing elements and increased capacity of the storage devices. The software developments included systems to interconnect a large number of the processing elements and systems to manage large amounts of data.

The available literature and my own research indicate that new technologies of the computing discipline may provide the necessary freedom to design a true decision support system. The problem and/or challenge currently before us is how to apply the evolving computing technologies to the design of an information system to support decision making.

The remainder of this chapter presents the thesis organization. The thesis is organized around the following components: a literature search to determine previous work and to document the problem areas; a model that was developed to benefit the problem areas; and a test of the model's reasonability.

A documented literature search of a decade of developments is presented in chapter 2. Several problem areas are listed as a result of the documented developments. The second chapter concludes with suggested solution areas.

The third chapter expands upon the problem areas and solutions of the computing discipline. The solution concepts are then applied to the information systems discipline. A model of a decision support system concludes the third chapter by realizing a paper prototype that demonstrates the interaction of the model's components.

The fourth chapter provides a test of the model's reasonability through a series of information exchanges with Farmland Industries of Kansas City, Missouri. The model's components are evaluated in terms of their applicability to the organizational environment of Farmland Industries.

The final chapter provides a summary of the research work, model development, and reasonability test. The chapter concludes with suggestions for further research work and other MBA thesis topics.

As the lack of literature indicates, there is much work that can be done in this discipline to conceptualize, design, and implement an information system to support the decision making process.[6]

II. A DECADE OF DEVELOPMENTS

This chapter contains a summary of developments surrounding the information systems discipline for the past decade. The conclusion reached in this chapter is that some concepts associated with the computing discipline would prove beneficial in their application to the information systems discipline.

In 1971, two researchers at Massachusetts Institute of Technology, Gorry and Morton, recognized the interdependency between developments in two areas.[2] The two areas were conceptual developments to describe the decision making process and technical developments of the computing industry. The authors stated this interdependency would be beneficial to the design of systems to support the decision making process.

Gorry and Morton's work made contributions in the conceptual area by applying the techniques of the decision making process to the managerial activities of the business enterprise. The two authors began by explaining Robert Anthony's taxonomy for managerial activity. The activity was divided into three categories and is presented in the following definition.

Definition 2.1 Managerial Activity

Strategic Planning -- "the process of deciding on objectives of the organization, on changes in these objectives, and on the policies that are to govern the acquisition, use, and disposition of these resources;"[24]

Management Control -- "the process by which managers assure that resources are obtained and used effectively and efficiently in the accomplishment of the organization's objective (most often concerned with people);"[24]

Operational Control -- "the process of assuring that specific tasks are carried out effectively and efficiently (most often concerned with the manufacturing tasks)."[24]

The authors also expanded upon H.A. Simon's work in the area of management decision making. One of Simon's main themes was that human beings solve problems using both structured decision making techniques and unstructured decision making techniques. A definition of each decision making technique is presented.

Definition 2.2 Structured Decision Making

"Decisions are structured to the extent that they are repetitive and routine, to the extent that a definitie procedure has been worked out for handling them so that they don't have to be treated as new each time they occur."[9]

Definition 2.3 Unstructured Decision Making

> "Decisions are unstructured to the extent that they are novel and consequential. There is no cut-and-dried method of handling the problem because it hasn't arisen before, or because its precise nature and structure are elusive or complex, or because it is so important that it deserves a custom-tailored treatment."[9]

The result of Gorry and Morton's work was a matrix that combined Anthony's managerial activity taxonomy on one axis and Simon's structured and unstructured approach to decision making on the other axis. This provided a framework into which all decision making could be located and a framework around which an information system to support decision making could be designed. Figure 1 presents this matrix with specific examples of managerial problems inserted at different coordinates of the matrix. The researchers conclude with the observation that "almost all systems described in MIS literature only support structured decision making...the need exists to concentrate on unstructured decision support systems."[2]

Another researcher, Gerrity of Massachusetts Institute of Technology, indicated that very few computer based information systems were increasing the effectiveness of management decision making.[1] He went on to indicate that the primary reasons were methodological rather than technical. Some of those design methodologies were based upon a machine-centered, a data-centered, or a process-centered approach.

	OPERATIONAL CONTROL	MANAGEMENT CONTROL	STRATEGIC PLANNING
STRUCTURED	ACCOUNTS RECEIVABLE	BUDGET ANALYSIS ENGINEERED COSTS	TANKER FLEET MIX
	ORDER ENTRY	SHORT TERM FORECASTING	WAREHOUSE AND FACTORY LOCATION
	INVENTORY CONTROL		
SEMI-STRUCTURED	PRODUCTION SCHEDULING	VARIANCE ANALYSIS	MERGERS AND ACQUISITIONS
	CASH MANAGEMENT	BUDGET PREPARATION	NEW PRODUCT PLANNING
UNSTRUCTURED	PERT/COST SYSTEMS	SALES AND PRODUCTION	R AND D PLANNING

Figure 1

An Information Systems Framework

Definition 2.4 Machine-centered Methodology

"This is characterized by utmost concern with hardware system efficiency, throughput, and capacity utilization, and by a certain air of isolation (if not insulation) from user problems."[1]

Definition 2.5 Data-centered Methodology

"The philosophy of this approach is that if one starts by collecting all the operational control data in an organization and builds applications upward from that base, then one will eventually arrive at the integrated system which will serve the decision needs of management."[1]

Definition 2.6 Process-centered Methodology

"This approach is characterized by a factoring of an existing clerical process into small modules, which are then programmed. No real thought is given to overall system redesign. Although it may introduce some processing efficiency, the approach ignores the full potential offered by recent changes in information technology. It simply transfers a manual information system to a machine."[1]

He states that if we want to design systems to support and improve the decision making process, the system should use a decision-centered design methodology.

Definition 2.7 Decision-centered Methodology

"The philosophy of this approach is that information is valuable only to the extent that it affects decision. For this reason, the decision makers plus their decision process are explicity considered as part of the total system being designed."[1]

Gerrity concluded by outlining some of the elements required for a DSS based upon a decision-centerd methodology. Those elements include:

> 1) memory to store data structures relevant to the task; 2) operators to apply primitive functions to stored data structures; and 3) a set of rules to combine the operators into problem-solving procedures.

Several other research articles are available from this time period on information systems to support decision making.[18,19,21,22,23] They generally indicate the goal of an information system was to free the manager of routine processing and prevent information overload. This would allow the manager to be creative and innovative.

Armed with this new set of ideas, researchers, consultants, users, and manufactures spent the early part of the 70's designing systems to aid the manager in the decision making process.

By the middle of this decade, Alter had examined and classified 56 computer based information systems.[3] There was little organized knowledge available about Decision Support Systems at this time so Alter's work made a significant contribution. The systems were classified into those that were data oriented or those that were model oriented. The data oriented systems ranged from a retrieval system to an analysis type of system. The retrieval system was generally no more than a mechanized manual filing system. An analysis type of system was in the form of a structured system which usually performed some data reduction and predetermined computation. An example of an analysis type of system is one that projected a budget based upon line items from an income statement. The model oriented systems included linear programming models, the simulation models, and the optimization models.

There were several interesting results reported by Alter pertaining to the designers and users of the 56 systems. Alter's results indicate:

> 1) most of the systems were conceived by entrepreneurs, not the users; 2) the entrepreneur designers concentrated on technical characteristics; most of the innovative systems were not implemented because the user did not request them; and 3) most user initiated systems did little more than mechanize existing practices. [17]

These results support Gerrity's earlier recommendations that the information system user needs to be more involved in designing and operating an information system to support the user's decision making process.

A number of other researchers have indicated that management information systems as a discipline and that actual systems implemented by its disciples, still have a long way to go. "We do not yet have a mature academic MIS field of study which would stand as a 'discipline' in its own right... There are, for example, no academic journals of any stature that specifically concentrate on MIS...."[5] A "majority of existing computer based systems fall into structured transaction processing (accounts payable and payroll packages). It is easy to understand and automate information flows for these types of transaction systems and designers can demonstrate cost savings from these systems."[4] Bonini stated, "...there has been general success both for computers used in routine data processing and for Operations Research/Management Science (OR/MS)" types of "programmed decisions," but "the extension of these methodologies (OR/MS) to the less structured strategic management decision areas has generally met with resistance, frustration, or at best, marginal success."[15]

Another researcher questioned the ingredients of success. "Our formal information systems which are supposed to meet this need (i.e., timely and accurate information) are ineffective. They don't always help, sometimes they actually impede the decision making process. Information system failures are too numerous! Why do these systems fail? What are the ingredients of success? These are bothersome questions because the information problem looms more critical than ever."[16] These results show there are conceptual problems in designing systems to support managers in the decision making process.

There were also many technical developments during this time in the computer industry. Those developments included:

> 1) miniaturization of the processing elements; 2) increased density of semiconductor components which increases the capacity of storage devices; and 3) recognition that software tools and languages could improve system design and programming productivity by being more user-oriented and natural.

Advances in these three areas - processing devices, storage devices, and languages - led to advancements in new areas.

A mechanism was needed to interconnect the processing devices at geographically distinct locations so they could increase productivity reliability by sharing the processing load. This mechanism evolved into communication software.

Another mechanism was needed to manage the large amounts of data that users were beginning to have the capability to store. A users collection of data was considered their data bank or data base. The mechanism to manage this data evolved into data base management systems.

These technological advances will give users the capability to distribute processing elements throughout the corporation to collect and store data, local to the area in which it was produced. The user community is now asking for ways in which the technical advancements can be incorporated into systems to support the manager in the decision making process.

A more detailed description of advancements made in computer technology was presented by Withington.[14] He classified the time from 1966 to 1974 as the era of communicators. The corporation centralized the EDP function but collected data and supported batch processing through remote terminals. The next time period, the one in which we are currently existing, 1974 to 1982, Withington classified as information custodians. The results of this era will find corporate data files integrated and satellite networks of minicomputers to support full transactional processing. The era after 1982 has been classified as the era of action aids. Withington predicts greater involvement by people of all levels of the organization and computer networks of different organizations interconnected to share data or sophisticated processing hardware.

A summary of the preceeding developments would be beneficial before presenting the current problem areas. In the early part of the decade, researchers were proposing that systems should be designed to support the manager in the decision making process. By 1975, 56 systems had been built and studied. The results indicated most were used for little more than routine report generation.[10] Technical developments in the computing industry included miniaturization of processing elements and increased capacity of storage devices. These new forms of information technology are referred to as small-scale computing. "Constraints of past information technology have significantly influenced the fundamental nature of the organization," [11] the information flows within an organization, and the type of information system to support the decision making process. Small scale computing has relaxed these constraints. "The new forms of information technology will allow different organizational structures."[11] This new flexibility of designing organizations" [11] and the information systems to support the decision making in the new organizations, "is the challenge that small-scale computing technology has laid before us."[11] "'The move toward MIS is 10 years old at least, and I think most of the companies have been disappointed with the results of the first round', observes Herbert A. Simon, a Nobel Prize winning professor of computer science and psychology at Carnegie-Mellon University."[13]

The problem areas pertaining to decision support systems are presented in Table 1. These problem areas are based upon the past decade of developments. After Table 1, the problem areas are discussed and some solutions are proposed.

1) Requirement and Design Mismatch

"The main problem appears to be a mismatch between the requirements of decision makers or decision making and the DSS design or performance."[10] This problem area demonstrates the need for greater user involvement in the design, evaluation, and operation phases of the DSS. This type of user oriented approach had been suggested by Lucas in 1978.[4] The user should lead the design, specify the input and output formats that are meaningful to him, and

1) Mismatch between requirements of decision makers and DSS design and performance.

2) Existing DSSs do not help decision makers conceptualize the pieces necessary to make unstructured or semistructured decisions.

3) Existing DSSs cannot vary with styles and skills of decision makers in the organization.

4) Impact of small scale computing technology on the managerial task is a complex issue.

5) DSS design should be evolutionary and the system should flexible so that it can be easily and quickly be changed and extended.

6) How to understand the interaction of the decision process, the task, the user, and the organizational setting.

7) How to conceptualize and design a DSS independent of the technology on which it is to be physically implemented.

8) No formal theory of decision support as yet.

9) Information System field suffers from a lack of cummulative research in which an effort is made to derive new ideas from old experience.

Table 1. Decision Support System Problem Areas

evaluate the system based upon complexity, reliability, usefulness, and user appeal rather than technical elegance.

2) Conceptualize Unstructured Decisions

"Existing DSSs do not provide the representations which decision makers need for semistructured or unstructured decisions."[10] The DSS should help the decision maker conceptualize the pieces of the problem as opposed to requiring him to describe the decision making process a priori so that an algorithm can be structured, designed, or implemented.

3) Varying Styles of Decision Makers

"Existing DSSs require specification of the decision making process in advance and do not support a variety of styles, skills, and knowledge; thus they do not help decision makers exercise the personal control to which they are accustomed when making semistructured or unstructured decisions."[10] This problem area indicates the need for a DSS to be flexible and evolve with the decision making style of the organization's personnel and the dynamics of the environment from which the organization receives opportunities and encounters constraints.

4) Impact of Small Scale Computing

"The impacts of this technology on the nature of the managerial task is very complex and indirect."[11] A DSS should help the decision maker break up the complex managerial task into simple pieces. An information system should provide enough information for the manger to correctly order the pieces to solve the problem.

5) Flexible System Design

"The evolutionary nature of DSS has led to a design strategy that emphasizes...a flexible system that can easily and quickly be changed and extended."[12] Not only should a decision maker be able to solve problems in pieces but the DSS designer should also be able to connect a system composed of modular pieces to support the decision maker in a dynamic environment.

6) Interaction of Decision Components

"Support involves a very detailed understanding of the decision process, the task, the user, and the organizational setting in which the system is to be embedded."[12] The DSS needs the capability to represent parallel information flows between the task, the users, and not only the organizational environment but also the outside world.

7) Physically Independent System Design

"One major problem in defining support systems is that this area has been principally characterized by computer technology. It may be more meaningful to emphasize decision support and then evade the issue by stating a DSS should be based on available and suitable technology."[12] However the basic objective of applying computer technology to support decision making remains the same. The technological advancements of the computer industry now afford DSS designers and users the freedom to conceptualize and design support systems independent of actual physical implementation techniques.

8) Formal Decision Support Theory

"There is no formal theory of Decision Support as yet."[12]

9) Cummulative Research

"In general, the Information System field suffers from a lack of cumulative research in an effort to derive new ideas from old experiences."[12]

There are several solutions to the above problems. A specific solution for each problem area will be presented in a later part of this work, however, two general solutions are proposed. Because of the lack of theory and research in the information system discipline, I propose that the recognition of similiar problem areas in other disciplines can be made. The information system discipline can then benefit by transferring the concepts associated with solutions to the problem area of other disciplines. I also propose that a new language is needed that contains a vocabulary of concepts associated with these potential solutions. The language should be brief and the semantics of the concepts should be simple and precise. The vocabulary should easily encourage the use of the language to conceptualize and design information systems to support decision making.

The problem areas encountered in the computing discipline and those encountered in the information system discipline are similar. The computing discipline crisis related to a lack of knowledge on how to conceptualize, design, and implement a complex software system that was to manage scarce computing resources. The information system discipline today has a crisis that relates to a lack of knowledge on how to conceptualize, design, and implement a complex information system to support a decision maker in managing scarce corporate resources. The computing discipline invented solutions to their crisis by searching for abstract concepts to simplify the understanding of a software system that was to manage scarce computing resources. The most important innovation was to visualize the complex computing system as a system composed of parallel computing resources operating in a time-independent manner. Once this problem was understood and the concepts invented, they were incorporated into a new language. The new language allowed software systems to be developed that were simple, understandable, and reliable. The resulting systems could be conceptualized, designed, implemented, and documented in a number of man-months as opposed to the 200 to 5000 man-year efforts of the software systems that created the crisis. "As a result, the systems became more reliable than the hardware they ran on."[20]

I feel the information system discipline could benefit by transferring the concepts associated with the successful solutions derived in the computing discipline. These transferred computing concepts are presented in Table 2. The problem areas follow a parallel course. By transferring the concepts from one discipline to the other, the solutions should follow a parallel course.

Abstraction -- the use of names as abbreviations for more detailed explanations.[7]

User Defined Data Types -- a class of abstract objects which is completely characterized by the operations available on those objects.[27]

Hierarchical Decomposition -- a process by which an abstract component of a system is decomposed into sub-components. This process continues until all components are at their lowest level.[29]

Virtual Machine -- a mechanism by which users can concentrate on logical components and the flow of information between these components to solve the problem and be relieved of the technical details associated with the physical implementation devices. [30]

Concurrency -- a concept by which efficiency can be increased because the execution of multiple components can overlap in time.[29]

Data/Information Flow -- a concept that implies information is shared among a number of components of the system. A unit of information flows from one component to the next, with each component manipulating the data values.[28]

Access Rules -- a set of procedures that guarantees the integrity of information shared among a number of components.[29]

Pipelining -- a structuring mechanism by which system components are ordered to ensure logically correct information flows.[28]

Table 2. Concepts Used As Solutions in the Computing Discipline

III. A NEW APPROACH TO DESIGN INFORMATION SYSTEMS

This chapter is organized in the following manner. The solution concepts of the computing discipline are presented and explained in detail. The concepts are then applied as solutions to the problem areas of the information system's discipline as presented in the previous chapter. A paper prototype, which provides an integrated view of the solution concepts, concludes the chapter.

SOLUTION CONCEPTS

Abstraction

"Abstractions...are used in all intellectual disciplines to present knowledge in forms that are easily understood and remembered."[7] However, several information system design methodologies from current sources [6,8,25], fail to mention the use of abstraction to conceptualize or design an information system. Abstraction provides the capability to assign an abbreviated variable name to describe the concept rather than the details of the concept. As an example, operating system is used as an abbreviation of "a set of manual and automatic procedures that enable a group of people to share a computer installation efficiently."[7]

User Defined Data Types

This is the capability for a user to specify variable names for specific types of resource components instead of the physical device names or component details. As an example, an information system is required to have one or more data gathering components and one or more data display components. The most understandable way to represent the data gathering components, is to first describe an abstract type of data gathering component and the variables that compose this type.

```
Type datagathers = (terminalkeyboard,
                    electronicsurveillance,
                    punchtimemachine,
                    marketingdatabase).
```

A similiar representation can be used to describe an abstract type of data display component.

```
Type datadisplay = (terminalscreen,
                    lineprinter,
                    storagedatabase).
```

This concept encourages the use of abstract variable names to represent any one of the specific data gathering or data display devices.

Hierarchical Decomposition

This is a concept that provides the capability to decompose a system component into sub-components. As an example, the terminalkeyboard component, of the datagathers type presented previously, can be decomposed into 3 sub-components. Each sub-component provides another level of detail to describe the component's function. The 3 sub-components follow:

1) Wait for next keyboard message;
2) Interpret commands from keyboard message; and
3) Send commmands to destiniation resource.

This type of functional decomposition encourages conceptual understanding and consequently improved design and modification of system components.

Virtual Machine

This concept allows a system to be conceptualized and designed independent of the actual physical implementation. At this abstract level, it will be sufficient to describe a computing system as an input resource, a processing resource, and an output resource. The components of a virtual computing system are presented in Figure 2 without the details associated with the physical implementation devices.

Concurrency

An important concept of the computing discipline is to visualize the complex computing system as a system composed of parallel computing resources operating in a time-independent manner. Associated with each resource component of Figure 2 is a series of steps that are executed in a cyclical manner. This provides for continuous execution of functions unique to each resource component.

The terminalkeyboard component was decomposed into 3 sub-components in the section that presented an explanation

FIGURE 2

COMPUTING SYSTEM COMPONENTS

of hierarchical decomposition. For this example, the terminalkeyboard will be used to represent the input resource. The cyclic nature of the input resource can be implied by returning to the 1st sub-component state after receiving an acknowledgment from the message destination. This would allow the input resource to wait for another keyboard message. This cyclic nature can be described by adding a 4th sub-component state and the CYCLE,END control statements.

```
CYCLE
   I.1)  Wait for next keyboard message;
   I.2)  Interpret commands from keyboard message;
   I.3)  Send commands to destination resource;
   I.4)  Wait for destination acknowledgment;
END.
```

This type of sequential, yet cyclical processing can be carried into each component. The speed with which a component operates is proportional to the number of sequential steps in each component. This presents the time-independent manner of the parallel computing resources. A more detailed description of each component is presented in the explanation of the next solution concept.

Data/Information Flow

This concept is used to represent the flow of data between the computing resource components. Figure 3 presents the data flow from the input resource to the processing resource. The double lined arrow with the variable name DATA represents this data flow. DATA is hereafter used to represent the commands of statements I.2

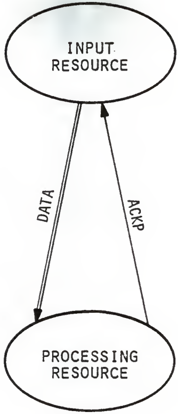

FIGURE 3

INPUT RESOURCE DATA FLOW

and I.3. The processing component returns a control signal to the input component to verify that it correctly received data from the input component. A control signal is represented by a single lined arrow with a variable name ACKP.

The output resource component also has a sequential, yet cyclical, series of steps to perform. The 4 sub-component states of the output resource are presented in their cyclical format.

```
CYCLE
  O.1  Wait for information from processing resource;
  O.2  Send ACKO control signal;
  O.3  Send information to lineprinter;
  O.4  Wait for lineprinter signal;
END.
```

For this example, a lineprinter device will be used to represent the output resource. Figure 4 presents this data flow.

The processing resource component also has a sequential, yet cyclical, series of steps to perform. The 5 sub-component states of the processing resource are presented in their cyclical format.

```
CYCLE
  P.1)  Wait for data from input resource;
  P.2)  Send acknowledgement signal (ACKP);
  P.3)  Transform data into information;
  P.4)  Send information to output resource;
  P.5)  Wait for acknowledgment signal (ACKO);
END.
```

The data transformation function of the processing component is provided by the system designer. Each processing component typically consists of a set of mathematical functions that can operate on the data as it flows from an

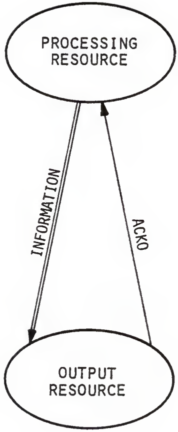

Figure 4
Output Resource Information Flow

input component to an output component. The processing resource also returns to the 1st sub-component state after it receives ACKO. Figure 5 presents this data flow.

The data/information flow that is presented in Figures 3, 4, and 5 implies that data or information flow directly from one computing resource component into another computing resource component. However a problem could arise if that specific data flow was to actually occur: Each resource component operates in a cyclical, time-independent manner, therefore both resource components could be accessing data values at the same time. An example of data shared between input resource and the processing resource demonstrates the problem. For this example, the processing resource has been changed to the following sub-component states:

```
CYCLE
  P.1)  Wait for data from input resource;
  P.2)  Send ACKP control signal;
  P.3)  Move data to a storeagedatabase;
  P.4)  Add data to a previous value of data;
  P.5)  Send information, i.e., previous value of data,
         to output resource;
  P.6)  Wait for ACKO control signal;
END.
```

If data had the value of 1 for statement P.3, the data value for P.4 should also be 1. However the input resource can operate at a different speed than the processing resource. Therefore the execution of statement I.3 could occur between the execution of statements P.3 and P.4. If statement I.3 was to occur between statements P.3 and P.4 and the function of statement I.3 was to send the data value 2, rather than 1, to the destination resource, statement P.4 would add 2

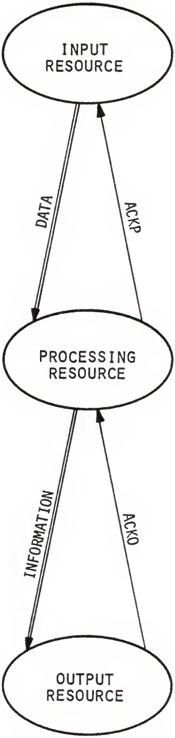

FIGURE 5

PROCESSING RESOURCE DATA/INFORMATION FLOW rather than 1 to the previous value of data. In this situation, an erroneous result would occur. A solution to prevent this type of problem was invented. The solution was another abstract concept that represents a temporary storage space for the data as it flows out of one computing resource component into another computing resource component. Figure 6 combines the data flows of Figures 3,4, and 5 and demonstrates use of the temporary storage spaces.

Access Rules

A mechanism is needed to ensure that only one computing resource at a time can access information in a temporary storage space. This mechanism is a set of access rules for each storage space. Access rules maintain the integrity of information and prevent contamination of the data contents in each storage space.

Definition 3.1 Integrity

> "Integrity of data implies the safeguarding of the data from malicious or erroneous tampering, faulty equipment, etc. The emphasis is on preserving all of the data and changes to the data so that the data base can be reconstructed in case of partial or total destruction...typical approaches include, journaling, checkpoints, and access controls."[31]

The temporary storage space located between the input resource and the processing resource, TSS1, has access rules that allow the input resource to SEND DATA and the processing resource to RECEIVE DATA. This storage space with its access rules are presented in Figure 7.

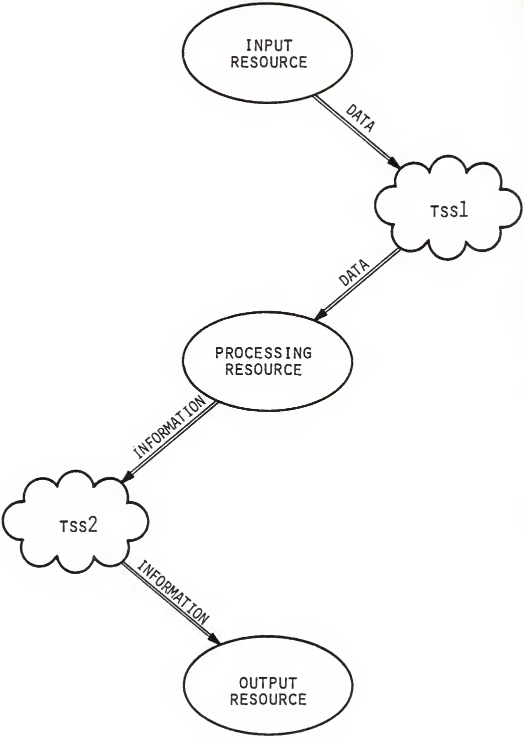

FIGURE 6

TOTAL COMPUTING SYSTEM
DATA/INFORMATION FLOW

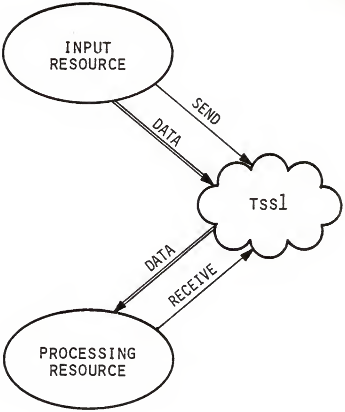

FIGURE 7

ACCESS RULES FOR TEMPORARY STORAGE SPACE 1

Statements I.3 and P.1 must be changed to reflect the temporary storage space. Statement I.3 becomes;

I.3) SEND DATA to TSS1.

The RECEIVE access rule of TSS1 delays the processing resource until DATA is available, as indicated by the SEND access rule of TSS1. TSS1 will then permit the data flow to the processing resource. Statement P.1 is changed to reflect this synchronization and data flow. Statement P.1 becomes;

P.1) RECEIVE DATA from TSS1.

A temporary storage space, TSS2, controls the synchronization and information flow between the processing resource and the output resource. TSS2 is presented in Figure 8.

Statement P.5 and O.1 must also be changed to reflect the tempoarary storage space. Statement P.5 becomes;

P.5) SEND INFORMATION, i.e., previous value of DATA, to TSS2.

Statement O.1 becomes;

O.1) RECEIVE INFORMATION from TSS2.

These concepts are beneficial to conceptualize and design a system that demonstrates simple and reliable information flows. The concepts also ensure the integrity of data as it flows among a number of computing resource components. Figure 9 presents another view of the system that depicts not only information flows but also access rules of each temporary storage space.

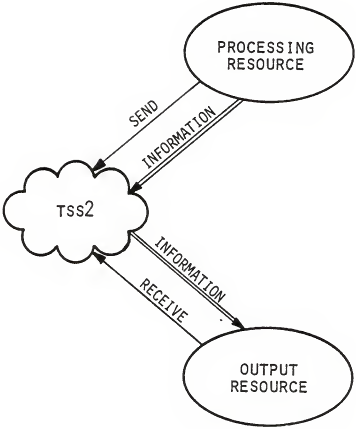

FIGURE 8

ACCESS RULES FOR TEMPORARY STORAGE SPACE 2

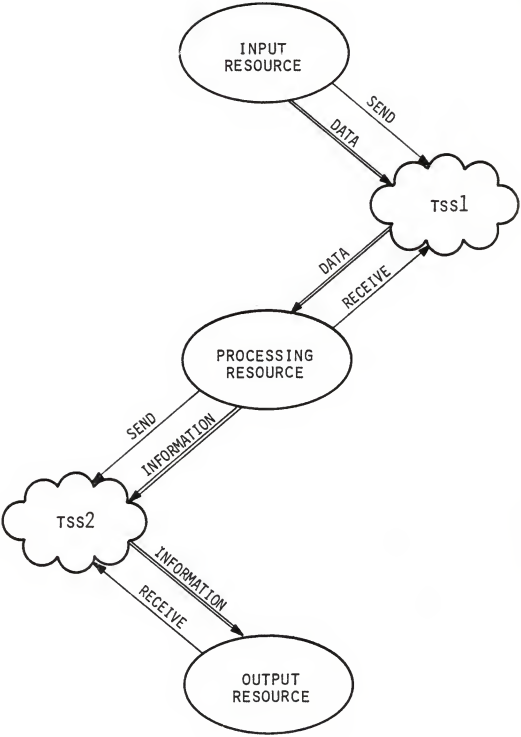

FIGURE 9

TOTAL COMPUTING SYSTEM DATA/INFORMATION FLOW INCLUDING ACCESS RULES

Pipelining

This concept represents a structuring mechanism by which the computing resource components are organized. By using this concept, a system designer can describe information flows between computing resource components to ensure a simple and reliable system design.

APPLICABILITY OF SOLUTION CONCEPTS TO PROBLEM AREAS

The purpose of this section is to present the problem areas and their corresponding solution concepts. The problems were presented in Table 1. The solution concepts were presented in Table 2. Each problem area may have more than one solution concept. Each solution concept may be applicable to more than one problem area. Table 3 presents the problem areas and their corresponding solution concepts.

1) Requirement and Design Mismatch

An applicable solution concept is the user defined data types. Underlying this concept is management's commitment to involve the user at all levels of the system; from conceptualization and design to actual implementation and usage. The abstract data types allow a user to conceptualize and design a DSS using a language of names in which the user is familiar. For example, a DSS should support different people in different organizational

Problem Areas	Solution Concepts
1) Requirement and Design Mismatch	User Defined Data Types
2) Conceptualize Unstructured Decisions	Abstraction, Hierarchical Decomposition, User Defined Data Types, Pipelining
3) Varying Styles of Decision Makers	Virtual Machine, Pipelining
4) Impact of Small Scale Computing	Abstraction, Hierarchical Decomposition, Concurrency, Pipelining
5) Flexible System Design	User Defined Data Types, Pipelining
6) Interaction of Decision Components	All Concepts
7) Physically Independent System Design	User Defined Data Types, Virtual Machine
8) Formal Decision Support Theory	No Specific Solution Concepts
9) Cummulative Research	Total Thesis Contents

Table 3. Problem Areas and Solution Concepts

components of the business enterprise. Therefore two different data types may be described to represent, at a user understandable level, the potential information flows between different people in different organizational components.

```
Type manager = (Coleman,
                Ruch,
                Brown,
                Unger).

Type department = (marketing,
                   deansoffice,
                   marketing,
                   computerscience).
```

2) <u>Conceptualize Unstructured Decisions</u>

Current techniques of the information system discipline do not provide the capability to represent semistructured or unstructured decision making. Underlying this problem area is the fact that semistructured or unstructured decision making requires an ad hoc or spur of the moment ordering of the decision making components. Simon has organized decision making, at the highest level, into 3 sub-components. They are:

> 1) search the environment; 2) establish alternatives; and 3) choose an alternative.[9]

Several solution concepts are applicable. Abstraction encourages a high level problem description. Hierarchical decomposition enables the solution components to be derived and described. User defined data types provide an understandable naming convention. Pipelining enables the components to be temporarily ordered in a correct solution sequence.

3) Varying Styles of Decision Makers

The varying skills and styles of the decision makers throughout the organization require a flexible DSS. The flexibility of a DSS is enhanced by the use of the virtual machine and the pipelining solution concepts: A virtual machine representation of an information system allows for easier modification because specific physical details do not need to be considered. Pipelining provides a structuring mechanism that can alter the ordering of the system components and consequently data flows. This allows a DSS to evolve with the changing characteristics of the decision makers in the organization.

4) Impact of Small Scale Computing

Several solution concepts help simplify the complex impact of small scale computing by encouraging an organization to view itself in terms of simple, interacting components. The solution concepts are abstraction, hierarchical decomposition, concurrency, and pipelining. A statement by Simon on the complexity of the problem solving process is applicable. "The complexity of the problem solving process that makes its outcome so impressive is a complexity assembled out of relatively simple interactions among a large number of extremely simple basic elements."[9]

All of the above concepts encourage this approach to reduce complexity into simpler parts and interactions.

5) Flexible System Design

The modularity associated with user defined data types and the consequential pipelining mechanism to structure the abstract data types, ensure an evolutionary and flexible system design. The ability to define data types that represent new or changing functions of the business enterprise, ensure an evolutionary DSS. The simplicity with which the data types can be ordered to represent a data flow, ensure the flexibility of a DSS.

6) Interaction of Decision Components

The solution concepts most applicable to this problem area encourage a DSS to be visualized as a system of people (organizational structure) doing things (business processes) to resources to transform them into products or services. Each of these abstract types - people, things, resources, and products or services - operate in a parallel time-independent manner. The role of a DSS is to provide information to support the decision maker to organize the processes to transform the resources into products or services in a productive and economical manner. All of the solution concepts provide the ability to express data flows in terms of the organizational structures and the business processes.

7) Physically Independent System Design

A logical view of a DSS and its corresponding data flows can be created with user defined data types and a virtual machine. This approach allows the logical, functional view of the system to remain static if the physical implementation devices are changed.

8) Formal Decision Support Theory

This thesis does not provide any specific solution concepts that would provide a formal theory of decision support. However, as previous experience in the computing discipline indicates, [20] the application of these solution concepts will lead to a more formal understanding of the problems and consequently some theoretical work.

9) Cummulative Research

This author has made an effort to derive new ideas from old experience, a solution to the ninth problem area.

PAPER PROTOTYPE

The purpose of the paper prototype is to provide an integrated view of the solution concepts. A DSS is realized by demonstrating potential information flows in an educational environment. The educational environment was chosen for this example to demonstrate the flexibility of the model approach based upon the solution concepts. The next chapter realizes a DSS by demonstrating potential information in an agricultural business environment. The applicability of the solution concepts, in terms of a flexible system design and a logical model approach, will be demonstrated by realizing a DSS for two different environments.

An educational environment can be represented by a system of people (faculty members), doing things (teach classes) to resources (students) to transform them into products or services (alumni). There are a number of people and things associated with the educational enterprise. They are described in more detail through each of the following data types.

The people of the enterprise can be described by declaring a type of educationmember.

```
Type educationmember = (administrator,
                        professor,
                        student,
                        coach,
                        alumnus).
```

There are a number of things that describe the processes of the enterprise. They can be described by a type of educationprocess.

```
Type educationprocess = (schedule,
                         teach,
                         learn,
                         recruit
                         donate).
```

The resources available to the enterprise can be described by declaring a student type of resource.

```
Type student = (highschoolgrad,
                transferstudent,
                returningstudent).
```

The remainder of this chapter presents a virtual DSS composed of the above abstract data types - administration, professor, and student.

The first information flow presented in the paper prototype is used to support decisions relating to the scheduling process. The scheduling process can be decomposed into a faculty scheduling process which assigns instructors to classes and a student scheduling process which assigns students to classes during an enrollment period. There is a requirement for three temporary storage structures to contain data that is shared among a number of the abstract data types. The following type declaration describes these storage structures.

```
Type TSS = (facultydatabase,
            coursedatabase,
            studentdatabase).
```

The faculty scheduling process consists of information flows between the facultydatabase, coursedatabase, and the administrator educationmember type. The administrator would RECEIVE from the facultydatabase, an instructor name and the course the instructor is qualified to teach. The administrator would then ASSIGN the instructor to the corresponding course in the coursedatabase. This information flow is presented in Figure 10.

The student scheduling process consists of information flows between the coursedatabase, the studentdatabase, and

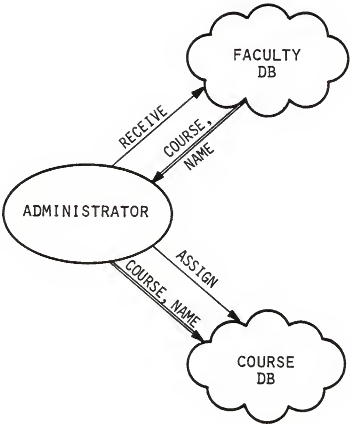

FIGURE 10

FACULTY SCHEDULING PROCESS

the student educationmember type. The student would RECEIVE from the coursedatabase a list that contained a course and instructor pair. The student would then ENROLL in a course by sending the student's name back to the coursedatabase and to the studentdatabase. This information flow is presented in Figure 11.

The second information flow presented in this prototype is a result of the scheduling process. The information flow enables a professor type of educationmember to obtain a class roster. The professor's name and course would be sent to the studentdatabase. A list of student names enrolled in the course would be returned to the professor. This information flow is presented in Figure 12.

The chapter has been organized into three parts. The first part provided an explanation for each solution concept. The second part related the solution concepts to the problem areas. The third part presented a paper prototype of a DSS model for an educational enterprise. The model demonstrated several information flows and provided an integrated view of the solution concepts to realize a simple, modular, virtual DSS. The next chapter provides the reasonability of the solution concepts and the model approach.

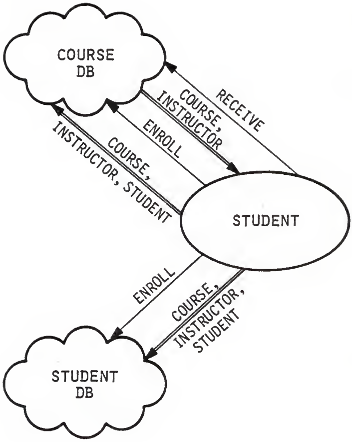

Figure 11

Student Scheduling Process

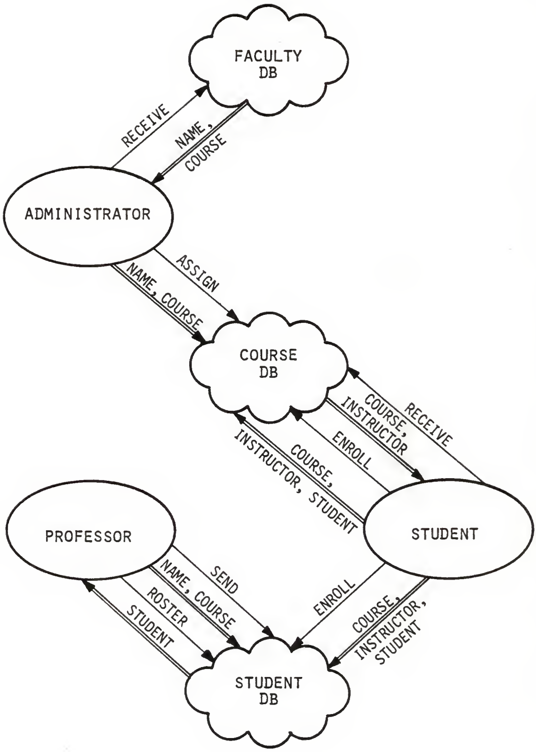

Figure 12
Professor Roster Process

IV. A TEST OF MODEL REASONABILITY

This chapter contains the results of a series of information exchanges with Mr. Gary Mundhenke, Vice President of Data Processing, at Farmland Industries of Kansas City, Missouri. Mr. Mundhenke is the manager for all computer related services and support provided by Farmland Industries. It was through Mr. Mundhenke's cooperation that an evaluation of the problem areas of chapter 2 was obtained and that a model of potential information flows within Farmland Industries was realized. Both the problem area evaluation and the Farmland model demonstrate the reasonability of the decision support system model.

Farmland Industries is a business enterprise that provides agriculturally oriented products and services. The products include pretroleum, fertilizer, agricultural chemicals, feed, and many other farm supply items. The services range from publications to data processing. Information about the products and services is provided through computerized communication links from two data processing sites to computerized terminals at local cooperatives. The data processing sites are located in Kansas City and Hutchinson. The local cooperatives, who use the computerized information system, are dispersed across a 16 state area from Illinois to Colorado and from North Dakota to Texas.

The data processing sites are characterized by 2 IBM computers (models 370-168 and 370-3033 in Kansas City) and 1 computer (model 370-168) in Hutchinson. The system can be characterized by 3 processing units, 3 printers, and 140 storage devices. There are 350 staff members required to maintain and support the data processing sites.

PROBLEM AREA EVALUATION

The problem areas of chapter 2 are confirmed by the results of a four hour interview session with Mr. Mundhenke. His comments and observations indicate that the problem areas presented in a previous chapter are representative of the past and present problem areas of Farmland Industries. Each problem area will be presented and followed by Mr. Mundhenke's comments.

1) Requirement and Design Mismatch

Mr. Mundhenke's comments indicate that current systems have helped decision makers solve problems at the operations level but that systems are needed to help decision makers solve problems in the strategic planning and unstructured area as shown in Figure 1.

2) Conceptualize Unstructured Decisions

An example of this problem area was provided in the Farmland environment. The first component of decision making, i.e., search the environment, was a problem area for structured decision making as well as semistructured or unstructured decision making. The problem was based upon the data available within Farmland. The different decision makers could not anticipate the kinds of data needed or determine the availability of data, its format, or how to retrieve the data. Consequently, all decision making was structured in terms of available data and its format.

3) Varying Styles of Decision Makers

This problem area was experienced at Farmland. Parts of the Farmland information system were made obsolete through recent management changes; through reorganization of geographic territories; and through changes to the entire accounting and sales reporting system.

4) Impact of Small Scale Computing

Mr. Mundhenke explained that current concerns with this problem area are based upon the proliferation of computer processing based typewriters and text processing machines throughout the organization. The primary need of this problem area is to provide an overall corporate guideline to ensure centralized control of data that is captured and used throughout the organization. This type of guideline will help apply the small and simple computer based tools to the complex managerial task.

5) Flexible System Design

The comments from Mr. Mundhenke indicate that evolutionary and flexible systems are easy in theory but from a practical and economical point of view, hard to implement. The problem is created by the need to design, test, and integrate new system components while maintaining current processing schedules and requirements.

6) Interaction of Decision Components

This type of problem does exist at Farmland Industries because the current system can not clearly express some of the business component interactions. Some attempts are being made at Farmland to visually express some of the decision component interactions in a simple and understandable manner. Coincidentally, some of the solution concepts of chapter 3 are being applied.

7) Physically Independent System Design

This is a problem area of Farmland Industries in the sense that user request and response times must be short, i.e., 1 to 5 seconds. In the view of Farmland Industries, this response time still requires an efficient system design based upon physical implementation technology. Mr. Mundhenke's comments indicate that real time performance specifications will always require a system to address the physical implementation environment.

8) Formal Decision Support Theory

This problem area was experienced by Farmland Industries during their recent attempt to analyze the information needs of their business environment. Farmland executives looked to external sources for a methodology to follow in the analysis process. One of only several available methodologies was chosen to guide Farmland through the analysis process. The chosen methodology enabled the information needs of the Farmland environment to be analyzed from a decision making viewpoint. The results of the analysis provided processes of the Farmland environment and the data necessary to support the processes. The key result of the analysis indicated that data should be managed as an asset, like cash or personnel. A reorganization of the information processing components is currently underway at Farmland to provide the necessary data flow between the processes.

9) Cummulative Research

Mr. Mundhenke experienced this problem during a recent management seminar at the Harvard Business School. Only one case presented and studied during the seminar was concerned with the impact of the new computer technologies on the design and use of information sytems.

FARMLAND DECISION SUPPORT SYSTEM MODEL

This section of chapter 4 provides a DSS model based upon the data flow between the business processes of Farmland Industries. Several of the Farmland processes are described and the solution concepts of chapter 3 are used to realize the DSS model.

The awareness and acceptance of the problem areas of chapter 2 led to a recent business systems planning study at Farmland: The study team was composed of management members at the Vice President level. They completed a comprehensive analysis of Farmland's business activities. The results of the analysis provided essential business processes and the types of information necessary to support those processes. The results are being used to establish an Information Systems Architecture that will support information flow among the processes of the business enterprise. It is this architecture which will be represented in the DSS model.

The Information System Architecture approach used by Farmland, coincides with the decision-centered methodology proposed by Gerrity.[1] Several benefits of this approach have been indicated by Farmland. One of the benefits is a system that represents logical information flows required to support business processes. This system is defined independent of any implementation techniques, i.e., manual or automated procedures are not specified at this point. A second benefit from this approach, is the flexibility of an information system to keep pace with the changing business environment. The logical view of the Information System can be maintained by the addition, deletion, and/or modification of business processes and the consequential information necessary to support any changes. Another benefit of this approach, is an Information System Architecture based upon information flows independent of organizational structure. This type of architecture allows organizational restructuring while minimizing the risk of obsolescence to the data processing resources and systems.

The business systems planning study and the Information Systems Architecture approach used so far by Farmland, has incorporated the following solution concepts identified in chapter 3:

abstraction;
user defined data types;
virtual machine;
concurrency; and
data/information flow.

These concepts and the remaining solution concepts will be used to realize a DSS model that represents potential information flows within Farmland Industries.

Hierarchical decomposition will be used to decompose some of the business processes into functional sub-components. A set of access rules will be provided for each temporary storage data base so that the information contained in the data bases can support and be shared among the Farmland business processes. The pipelining solution concept will be used to structure the processes and the storage data bases to realize a model of known information flows that were determined as a result of the business systems planning study.

The first step to realize a DSS is to describe a type of process that represents the components of the Farmland enterprise. The following type definition presents a subset of the process components contained in Appendix 1. This enables a simplified model to be presented.

```
Type process = (planning,
                externalenvironment,
                products,
                personnel,
                financial,
                control).
```

Only 3 of the process components - planning, products, and financial - will be represented in the DSS model.

The next step in realizing the DSS model is to describe the storage data bases that contain information to support the business processes. Again, only several data base types of Appendix 1 are presented and used in later examples.

```
Type planningdb = (corporatedb,
                   marketdb,
                   productdb,
                   localcoopdb).

Type externalenvironmentdb = (marketdb,
                              industrydb,
                              forecastdb,
                              agproducerdb).

Type productiondb = (inventorydb,
                     vendordb).
```

Only 5 of the storage data bases presented - planningdb, externalenvironmentdb, productiondb, inventorydb, and vendordb - will be represented in the DSS model.

In the previous chapter, each storage space contained a set of access rules. These access rules maintained the integrity of the data by coordinating the data flow that is shared among the business processes through the storage data bases. The results of the business systems planning study indicate that some of the processes create data and the remaining processes use data. A data creating process will send data to a storage data base. A data using process will receive data from a storage data base. Each storage data base of the DSS model will have a SEND and RECEIVE access rule.

Pipelining is the final solution concept that will be used to realize the DSS model. Structuring the Farmland processes, in a manner so that data is "piped" among the process components and the storage data bases, demonstrates the pipelining solution concept.

Several information flows are presented in the following figures. Each of the 3 process components are decomposed into functional sub-components to demonstrate the Farmland DSS model.

The function of the planning process is to determine the corporate mission, objectives, and policies based upon the goods and services that are to be sold or consumed. A high level description of this process implies that the process RECEIVEs data from the externalenvironment data bases and SENDs the objectives and policies to the planning data bases. The following sub-component states describe this process.

```
CYCLE
  Plan.1)  RECEIVE DATA from externalenvironmentdb;
  plan.2)  Transform DATA into Mission, Objectives,
            and Policies;
  Plan.3)  SEND Mission, Objectives, and Policies
            to planningdb;
END.
```

This data/information flow is presented in Figure 13.

The function of the products process is to obtain materials and supplies to develop specific goods or services. According to this function, the products process would RECEIVE data about materials and supplies from the vendor data base. The process would then develop goods or services based upon the materials and supplies and SEND information about the specific goods or services to the inventory data base. The following sub-component states describe this process.

```
CYCLE
  Prod.1)  RECEIVE DATA from vendordb;
  Prod.2)  Develop goods or services;
  Prod.3)  SEND INFORMATION to inventorydb;
END.
```

This data/information flow is presented in Figure 14.

The final data/information flow presented is the financial process whose function is to manage the allocation of expenditures to transform resources into goods and services. Expenditure decisions are based upon data about the external environment and data about the types of goods and services currently in production. This data is contained in their respective data bases. Information from

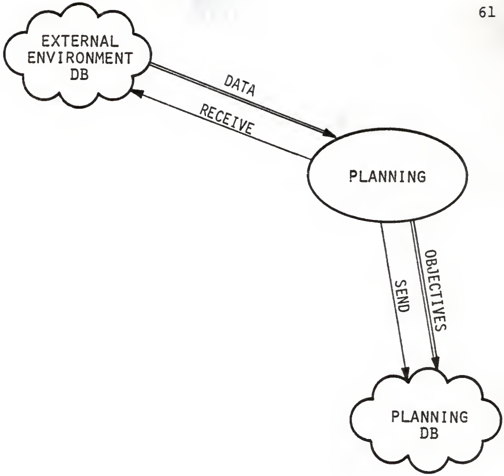

FIGURE 13
DATA/INFORMATION FLOW TO SUPPORT PLANNING DECISIONS

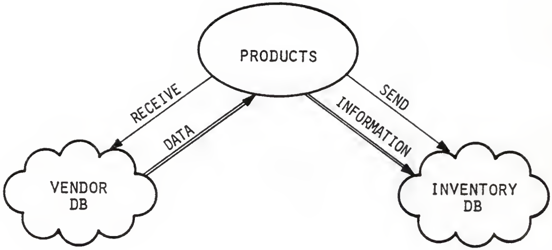

Figure 14

Data/Information Flow to Support Products Decisions the resulting decisions is sent to the planning data bases. The following sub-component states describe this process.

```
CYCLE
  Fin.1)  RECEIVE DATA from externalenvironmentdb
           and productiondb;
  Fin.2)  Determine expenditure types and amounts;
  Fin.3)  SEND INFORMATION to planningdb;
END.
```

This data/information flow is presented in Figure 15.

The data/information flows presented in Figures 13, 14, and 15, are combined in Figure 16 to represent the Farmland Decision Support System model. Each of the solution concepts discussed in chapter 3 - abstraction, user defined data types, hierarchical decomposition, virtual machine, concurrency, data/information flow, access rules, and pipelining - have been used to conceptualize, describe, visualize, and design the Farmland DSS model. Definition 1.3 implies that a DSS uses computers to assist decision making in semistructured and unstructured tasks. The 3 process components described in the Farmland DSS model - planning, products, and financial - represent semistructured and unstructured tasks. The planning process of the DSS model describes new product planning and R&D planning of Figure 1. The products process of the DSS model provides managment control of products as shown in Figure 1. Expenditure allocations in the financial process of the DSS model describe budget preparation tasks of Figure 1. All of the process components represented in the DSS have the potential to assist decision making in semistructured or unstructured tasks.

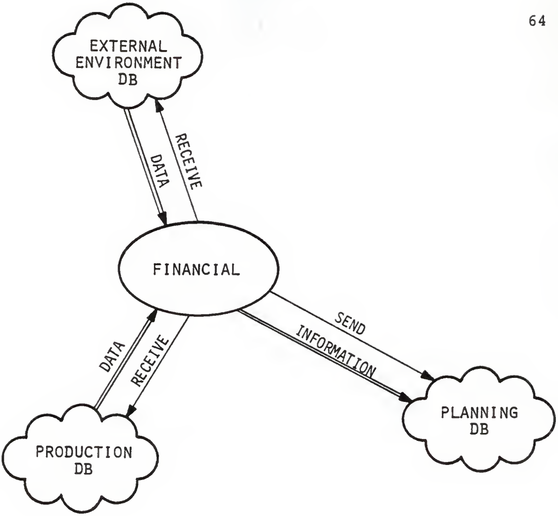

FIGURE 15

DATA/INFORMATION FLOW TO SUPPORT FINANCIAL DECISIONS

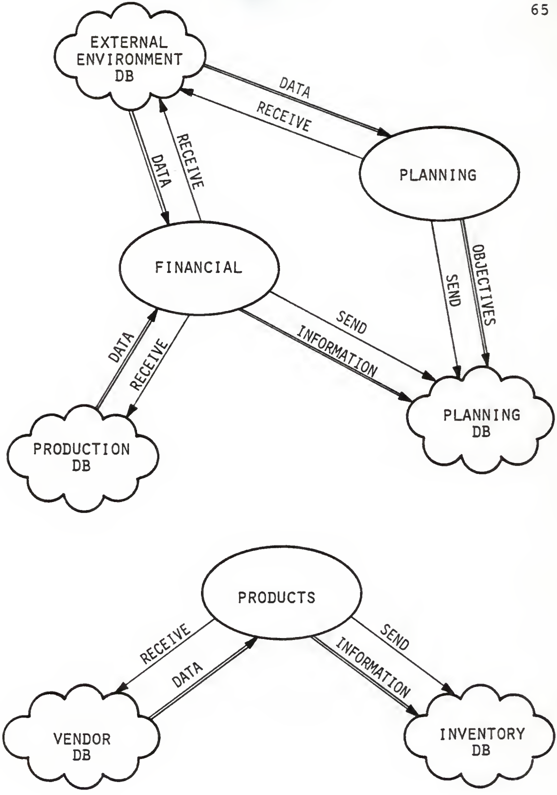

FIGURE 16

FARMLAND DECISION SUPPORT SYSTEM MODEL

A DSS, as implied in Definition 1.3, should support decision making rather than replace it. The types of information derived from the process components of the DSS model - mission, objectives, policies, inventory information, and product information - all support decision making rather than replace decision making.

The final part of the DSS definition implies that the effectiveness of decision making should be improved. Any improvements to the effectiveness of decision making that the DSS model provides can only be determined through a real time physical implementation.

The Farmland DSS model can satisfy two parts of the DSS definition. The third part cannot be satisfied from a paper prototype.

This chapter has been organized into two parts. The first part confirmed the problem areas of chapter 2. The concluding part presented a DSS model that incorporated the solution concepts of chapter 3. The information flows represented in the model have the potential to aid semistructured and unstructured decision making within the business processes of Farmland Industries. The confirmed problem areas and the Information System Architecture approach of Farmland Industries, which can be supported by the decision support system model, all demonstrate the reasonability of the decision support system model.

V. CONCLUSION

The goal of this thesis was to demonstrate the applicablity of new computer technologies to management information systems. Motivation for the work was twofold. Preliminary research indicated current information systems were not responding to the changing needs of decision makers and that the majority of information systems were based upon older computer technologies.

The beginning point was to determine the current state of the information system discipline. A literature search was conducted and a series of developments over the last ten years was presented. These developments indicated that information systems were being developed to support structured decision making and that problem areas existed in conceptualizing, designing, and using information systems to support unstructured decision making.

A similarity of problem areas between the information system discipline and the computing discipline was noted. Solution concepts of the computing discipline were explained. The solution concepts were applied to the problem areas of the information system discipline. A paper prototype model was developed to incorporate the solution concepts: The model demonstrated information flows necessary to support potential decisions within an educational enterprise. The application of the computing discipline solution concepts to the information system discipline problem areas has provided a mechanism to conceptualize, design, and use information systems to support unstructured decision making.

The reasonability of the solution concepts and the model approach was provided by the application of the model to the environment of Farmland Industries of Kansas City, Missouri.

The new computer technologies have not been physically applied to a management information system model. However the logical concepts associated with the new computer technologies have been used as solution concepts to realize two decision support system models: The first system was used to explain the concepts and the second system was used to demonstrate reasonability. The model approach and the logical concepts have demonstrated the applicability of new computer technologies to management information systems through a decision support system model. Final comments from Mr. Mundhenke indicate that the job before us is to make complex representations simple. The application of the solution concepts help provide a simplier representation.

As a result of the work accomplished in this thesis, several further study areas and MBA thesis topics are suggested below:

1) conduct a survey of MIS/DSS users to determine specific system functions, features, and implementation technology;

2) continue the description and development of a meta language to incorporate these and other solution concepts to be used for MIS/DSS design models;

3) develop a data model that describes data and its attributes in the context of a business environment;

4) determine a methodology to be followed to build MIS/DSS sub-components and prototypes;

5) define the translation process for the concepts described in the meta language to an actual implementation;

6) conceptualize, design, and implement some MIS/DSS prototypes;

7) enhance the decision support literature through publication of results from all of the above study areas.

REFERENCES

1) Gerrity, Thomas P. Jr., "Design of Man-Machine Decision Systems: An Application to Portfolio Management," Sloan Management Review, Vol 12, No 2, Winter 1971, pp 59-75.

2) Gorry, G. Anthony and Michael S. Scott Morton, "A Framework for Management Information Systems," Sloan Management Review, Vol 13, No 1, Fall 1971, pp. 55-70

3) Alter, Steven, "A Taxonomy of Decision Support Systems," Sloan Management Review, Vol 19, No 1, Fall 1977, pp 39-56.

4) Lucas, Henry C. Jr., "The Evolution of an Information System: From Key Man to Every Person," Sloan Management Review, Vol 19, No 2, Winter 1978, pp. 39-52.

5) Keen, Peter G.W. and Michael S. Scott Morton, "Decision Support Systems: An Organizational Perspective," Addison - Wesley Publishing Company, 1978.

6) Burch, John C. Jr., Felix R. Strater, and Gary Grudnitski, "Information Systems: Theory and Practice, Second Edition," John Wiley and Sons, 1979.

7) Hansen, Per Brinch, "Operating Systems Principles," Prentice-Hall, Inc, 1973.

8) Murdick, Robert G. and Joel E. Ross, "Information Systems for Modern Management," Prentice-Hall, Inc, 1975.

9) Simon, Herbert A., "The New Science of Management Decision," Harper & Brothers Publishers, 1960.

10) Carlson, Eric D., "An Approach for Designing Decision Support Systems," Database, Vol 10, No 3, Winter 1979, pp 3-15.

11) Dempster, Michael, Goran Fick, and Richard Hackathorn, "Impacts of Small-Scale Computing Technology on Managerial Tasks," Working Paper 79-03-09, Department of Decision Sciences, The Wharton School, Philadelphia, PA 19104, March 1979.

12) Keen, Peter G. W. and Richard D. Hachathorn, "Decision Support Systems and Personal Computing," Working Paper 79-01-03, Department of Decision Sciences, The Wharton School, Philadelphia, PA 19104, April 3, 1979.

13) "Solving a computer mismatch in management," Business Week, April 2, 1979, pp 73-76.

14) Withington, Frederic G., "Five Generations of Computers," Harvard Business Review, July-August 1974, Vol 52, No 4, pp 99-108.

15) Bonini, Charles P., "Computer, Modeling, and Management Education," California Management Review, Winter 1978, Vol XXI, No 2, pp 47-55.

16) Daft, Richard L. and Norman B. MacIntosh, "A New Approach to Design and Use of Management Information," California Management Review, Fall 1978, Vol XXI, No 1, pp 82-92.

17) Alter, Steven L., "How effective managers use information systems," Harvard Business Review, November-December 1976, Vol 54, No 6, pp 45-53.

18) Argyris, Chris, "Management Information Systems: The Challenge to Rationality and Emotionality," Management Science, February 1971, Vol 17, No 6, pp B-275.

19) McFarlan, F. Warren, "Problems Planning the Information System," Harvard Business Review, March-April 1971, Vol 49, No 2, pp 75-89.

20) Hansen, Per Brinch, "A Keynote Address on Concurrent Programming," Computer, Vol 12, No 5, May 1979, pp 50-56.

21) Ackoff, Russell, "Management MisInformation Systems," Management Science, Vol 14, No 4, December 1967, pp B-147.

22) Ziegler, James R., "Computerized Information Systems," reprinted in MIS: Management Dimensions, Holden-Day, Inc, 1973, pp 485-494.

23) Chervary, Norman L. and Gary W. Dickson, "Economic Evaluation of Management Information Systems: An Economic Framework," reprinted in MIS: Management Dimensions, Holden-Day, Inc, 1973, pp 529-540.

24) Anthony, R. N., "Planning and Control Systems: A Framework for Analysis," Boston, Harvard University Graduate School of Business Administration, 1965.

25) Putnam, Arnold O., "MANAGEMENT INFORMATION SYSTEMS: Planning, Developing, Managing," Herman Publishing, Inc, 1977.

26) Dearden, J., "How to Organize Information Systems," Harvard Business Review, Vol 43, No 2, March-April 1965, pp 65-73.

27) Liskov, Barbara and Stephen Zilles, "Programming With Abstract Data Types," Computation Structures Group Memo 99, Massachusetts Institute of Technology, March 1974.

28) Denning, Peter J., "Operating Systems Principles for Data Flow Networks," Computer, Vol 11, No 7, July 1978, pp 86-97.

29) Hansen, Per Brinch, "The Architecture of Concurrent Programming," Prentice-Hall, Inc., 1977.

30) Madnick, Stuart E., and John J. Donovan, "Operating Systems," McGraw Hill Book Company, 1974.

31) Tsichritzis, Dionysios C. and Frederick H. Lochovsky, "Data Base Management Systems," Academic Press, 1977.

APPENDIX A

FARMLAND INDUSTRIES, INC.
Business Systems Planning Study
Information Systems Architecture
MARCH 23, 1979

Legend:
- U Data User
- ■ Data Creator

PROCESSES RELATING TO →

PLANNING and **EXTERNAL ENVIRONMENT**

DATA CLASSES	#	1 Corporate Planning	2 Market Planning	3 Product Planning	4 Sales Planning	5 Local Coop Develop Planning	6 Production Planning	7 Procurement Planning	8 Distribution Planning	9 Personnel Planning	10 Financial Planning	11 Governmental & Reg. Affairs	12 Market Research	13 Local Coop Operations	14 Financing Local Coops	15 Membership Acquisition	16 Public Relations	17 Member Relations	18 Advising Local Coops	19 Local Coop Education
Corporate Plan Data	1	■	U	U	U	U	U	U	U	U	U	U	U				U			
Market Plan Data	2		■	U	U	U	U	U	U			U				U				
Product Plan Data	3			■	U	U	U	U	U			U								
Sales Plan Data	4				■	U	U	U	U	U	U	U	U							
Local Coop Development Plan Data	5			U	U	■	U	U	U	U	U	U		U	U		U	U	U	U
Production Plan Data	6						■	U	U	U	U	U								
Procurement Plan Data	7						U	■	U		U	U								
Distribution Plan Data	8						U	U	■	U	U	U								
Personnel Plan Data	9									■	U	U								
Financial Plan Data	10										■	U								
Regulatory Data	11	U		U	U	U	U	U	U	U	U	■		U		U			U	U
Market Data	12		U		U	U						U	■	U					U	
Industry Data	13	U			U		U	U	U	U		U	■	U	U		U			
Forecast Data	14	U	U	U	U		U	U	U	U	U		■	U						
Ag Producer Data	15		U			U						U	■					U		
Local Coop Data	16	U	U		U	U			U			U		■	■	U	U	U	U	U
Product and Material Data	17			U			U					U		U						U
Vendor Data	18							U				U								
Production Data	19						U					U								
Facilities Data	20	U	U				U		U	U		U								
Inventory Data	21								U			U								
Transportation Data	22								U			U	U	U						
Order Data	23				U				U				U	U						
Selling Price Data	24				U							U	U	U						
Non-Member Customer Data	25								U											
Personnel Data	26	U								U		U							U	
Sales History Data	27		U	U	U	U						U	U	U						
Cost Data	28		U	U	U	U	U	U	U	U	U	U	U	U						
Financial & Accounting Data	29	U									U	U		U	U		U	U		
Performance Data	30	U	U	U	U	U	U	U	U	U	U							U		

PRODUCTION, PROCUREMENT, DISTRIBUTION, & SALES

DATA CLASSES	#	20 Product Development	21 Engineering	22 Procurement	23 Production	24 Construction	25 Facilities Maintenance	26 Materials Management	27 Transportation	28 Order Processing	29 Warehousing	30 Inventory Management	31 Returns, Credits, Adjustments	32 Pricing	33 Selling	34 Advertising & Promotion	35 Sales Management	36 Sale of By-Products & Surplus
Corporate Plan Data	1																	
Market Plan Data	2																	
Product Plan Data	3	U																
Sales Plan Data	4					U								U	U	U	U	
Local Coop Development Plan Data	5																	
Production Plan Data	6		U		U	U	U	U										U
Procurement Plan Data	7																	
Distribution Plan Data	8		U			U	U		U		U	U						
Personnel Plan Data	9																	
Financial Plan Data	10													U				
Regulatory Data	11	U	U		U	U			U					U				
Market Data	12													U		U		
Industry Data	13													U		U		
Forecast Data	14													U				
Ag Producer Data	15															U		
Local Coop Data	16								U	U				U	U	U	U	
Product and Material Data	17	■	■	■		U		U	U	U	U	U	U		U	U		U
Vendor Data	18			■				U					U					
Production Data	19		U	■				U										
Facilities Data	20		U	■														
Inventory Data	21																	
Transportation Data	22			U						U			U					
Order Data	23				U	U			U		U	U						
Selling Price Data	24									U					U			U
Non-Member Customer Data	25									U								
Personnel Data	26				U				U		U						U	
Sales History Data	27																	U
Cost Data	28		U		U	U	U	U	U		U	U	U	U		U	U	U
Financial & Accounting Data	29																	U
Performance Data	30																	

PERSONNEL and **FINANCIAL RESOURCE**

DATA CLASSES	#	37 Hiring, Recruiting, Termination	38 Employee Relations	39 Compensation Administration	40 Benefits Administration	41 Labor Contract Administration	42 Safety & Security	43 Accounts Payable	44 Credit Administration	45 Funds Administration	46 Capital Asset Management	47 Payroll	48 Financial Reporting	49 Taxation	50 Member Equity Admin.	51 Accounting	52 Auditing
Corporate Plan Data	1															U	U
Market Plan Data	2																
Product Plan Data	3																
Sales Plan Data	4																
Local Coop Development Plan Data	5																
Production Plan Data	6																
Procurement Plan Data	7																
Distribution Plan Data	8																
Personnel Plan Data	9	U	U	U	U	U	U										
Financial Plan Data	10								U	U	U		U	U			
Regulatory Data	11	U	U	U	U	U	U		U	U		U	U	U	U	U	U
Market Data	12																
Industry Data	13					U			U	U							
Forecast Data	14																
Ag Producer Data	15																
Local Coop Data	16								U						U		U
Product and Material Data	17						U	U								U	U
Vendor Data	18							U	U							U	U
Production Data	19															U	U
Facilities Data	20						U				U			U		U	U
Inventory Data	21							U								U	U
Transportation Data	22															U	U
Order Data	23							U								U	U
Selling Price Data	24															U	U
Non-Member Customer Data	25								U							U	U
Personnel Data	26				U	U	U					U				U	U
Sales History Data	27	U		U										U	U		
Cost Data	28					U				U							
Financial & Accounting Data	29			U	U	U		U	U	U	U		U				
Performance Data	30		U	U													

CONTROL

DATA CLASSES	#	53 Coordin. & Eval. of Formal Plan	54 Corporate Control	55 Market Control	56 Product Control	57 Sales Control	58 Local Coop Development Control	59 Production Control	60 Procurement Control	61 Distribution Control	62 Personnel Control	63 Financial Control
Corporate Plan Data	1	U	U									
Market Plan Data	2	U		U								
Product Plan Data	3	U			U							
Sales Plan Data	4	U				U						
Local Coop Development Plan Data	5	U					U					
Production Plan Data	6	U						U				
Procurement Plan Data	7	U							U			
Distribution Plan Data	8	U								U		
Personnel Plan Data	9	U									U	
Financial Plan Data	10	U										U
Regulatory Data	11											
Market Data	12											
Industry Data	13											
Forecast Data	14											
Ag Producer Data	15											
Local Coop Data	16		U				U					
Product and Material Data	17											
Vendor Data	18											
Production Data	19											
Facilities Data	20											
Inventory Data	21											
Transportation Data	22											
Order Data	23											
Selling Price Data	24											
Non-Member Customer Data	25											
Personnel Data	26											
Sales History Data	27			U	U	U	U					
Cost Data	28			U	U	U	U	U	U	U	U	
Financial & Accounting Data	29		U	U	U	U	U	U	U	U	U	U
Performance Data	30											

A
DECISION SUPPORT SYSTEM
MODEL

by

Michael Wayne Farrell

B.S., Kansas State University, Manhattan, Kansas, 1975
M.S., Kansas State University, Manhattan, Kansas, 1979

AN ABSTRACT OF A MASTER'S THESIS

submitted in partial fullfillment of the

requirements for the degree

MASTER OF BUSINESS ADMINISTRATION

College of Business Administration

KANSAS STATE UNIVERSITY
Manhattan, Kansas
1979

The purpose of this thesis was to recognize problem areas and provide solution concepts to the information system discipline. Previous computing discipline constraints required an information system to be designed based upon a machine-centered or a data-centered approach. However recent developments in the computing discipline are relaxing the constraints so that an information system can be designed based upon a decision-centered approach.

The technical developments of the computing discipline were applied as solution concepts to the problem areas of the information system discipline. A decision support system model was realized to demonstrate the solution concepts. The reasonability of the model was demonstrated through its applicability to Farmland Industries of Kansas City, Missouri.