

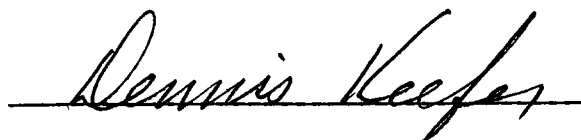
To the Graduate Council:

I am submitting herewith a thesis written by Martin H. Davis, Jr. entitled "A Geographical Information System for the Big South Fork Area." I have examined the final copy of this thesis for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Master of Science, with a major in Engineering Science.



F. Shahrokhi, Major Professor

We have read this thesis
and recommend its acceptance:



Accepted for the Council:



The Graduate School

A GEOGRAPHICAL INFORMATION SYSTEM
FOR THE BIG SOUTH FORK AREA

A Thesis
Presented for the
Master of Science
Degree
The University of Tennessee, Knoxville

Martin H. Davis, Jr.

August 1984

Copyright © Martin H. Davis, Jr., 1984
All rights reserved

ACKNOWLEDGEMENTS

In the course of performing this thesis research, I would like to thank a number of people. I thank Dr. F. Shahrokhi, who served as chairman of my committee, for his invaluable assistance, support, and comments. I also thank Dr. D. Keefer and Dr. K. Kimble, the other members of my committee, for their time and efforts. Thanks also go to Scott Milam, Jay Muthuswami, and Mike Luke for their constant friendship and encouragement. I am also indebted to my father for the kind use of his computer and word processor and to both my mother and father for their love and encouragement. I finally thank my loving and patient wife Carol for her understanding of and commitment to my life and goals.

ABSTRACT

The Big South Fork National River and Recreation Area has identified their need for a Geographical Information System (GIS). The planned use for the GIS is to reduce the time necessary to answer inquiries and make decisions concerning the natural resources in the Big South Fork area. A GIS has four generic components: data definition and acquisition, data input and storage, data retrieval and analysis, and report generation and display. A GIS can analyze data either in a polygonal or raster, or combination, format. Nine stages in the design of a GIS can be identified, the most important of which is the pilot study. Pilot programs were written to implement the latter three generic components of a GIS for the Big South Fork area. This was done using a combination of the BASIC and PASCAL languages with the dBASEII relational database software and an IBM Personal Computer with associated peripherals. The following was accomplished: geographical entities may be input and stored in a polygonal format (as points, lines, and polygons with associated descriptor data) with editing capabilities, single-theme and multi-theme analysis (the latter through the use of polygon to raster conversion schemes) may be performed, and tabular and graphical outputs may be produced. More work needs to be done to ensure the following for the final product: that the techniques for computing areas and lengths, storing boundary data points, and converting polygonal information to raster format are most efficient for the available hardware/software combination and that the developed software is user friendly.

TABLE OF CONTENTS

CHAPTER	PAGE
I. INTRODUCTION	1
II. DESCRIPTION AND CHARACTERISTICS OF GEOGRAPHICAL INFORMATION SYSTEMS.	2
Introduction	2
Differentiation Between GIS and Management Information System or Data Base Management System.	3
Generic Components of a GIS.	5
Data Formats	7
Coordinate Systems	12
Design Methodologies	13
III. DESIGN OF THE BIG SOUTH FORK GIS	18
IV. PILOT STUDY OF BIG SOUTH FORK GIS.	24
Data Input and Storage	24
Data Retrieval and Analysis.	34
Report Generation and Display.	40
Examples of Output	43
Summary.	47
V. ANALYSIS AND DISCUSSION.	50
Data Input and Storage	50
Data Retrieval and Analysis.	52
Report Generation and Display.	53
Summary.	54
LIST OF REFERENCES	56
VITA	60

LIST OF FIGURES

FIGURE	PAGE
1. Seven historical sites from the pilot study theme called HISTORICAL.	44
2. Two streams from the pilot study theme called STREAM	45
3. Three vegetation polygons from the pilot study theme called VEG	46
4. The grid cells which represent the intersection of the streams in Figure 2 and the vegetation polygons in Figure 3	48
5. A composite of Figures 1, 2, 3, and 4.	49

CHAPTER I

INTRODUCTION

Decision makers in today's world are often faced with two problems. First, too many times there appears to be an overwhelming glut of information which must be turned into a rational decision. Second, this glut must be stored in such a manner that it can be recalled in a relatively quick manner. As French and Taylor put it (1982), "[m]odern society imposes a large number of constraints on the planning and management of those goods and services which affect the general welfare."

Typically, stock market brokers, airline reservation clerks, or grocery store managers might be thought of as having such problems with a glut of data. However, there is a growing number of other types of people who have such problems but have not had the capability to do much about it in a relatively inexpensive way. This thesis has come from the attempt to provide a group of people working in an outdoors environment the capability to manage better their flow of information.

The Big South Fork National River and Recreation Area is being provided what is called generically a Geographical Information System (GIS). This system will help them solve the two problems mentioned earlier. This thesis will describe GISs and their components, the nature of the problem in the Big South Fork area, the design of the GIS for the Big South Fork, and a description of the pilot study programs.

CHAPTER II

DESCRIPTION AND CHARACTERISTICS OF GEOGRAPHICAL INFORMATION SYSTEMS

Introduction

What is a Geographical Information System (GIS)? The specific answer to that oftentimes depends upon who answers, but a GIS is probably best defined as a "computerized system designed to store, process, and analyze spatial data" (Brooks and Pease, 1978). Therefore, a GIS is a system employing a computer which the decision maker hopes will enable him to solve, or at least make easier to handle, the two problems mentioned in the Chapter I.

To what areas may a GIS be applied? GISs have been used for quite a variety of purposes including land use inventories, environmental impact studies, forest management, water resources management, agricultural surveys, socio-economic studies, transmission line siting, wildlife habitat monitoring, suburban growth management, ski resort design, nuclear power plant siting, and hydrological modeling (Brooks and Pease, 1978; Goldberg, 1982).

Why must a GIS be a computerized system? Certainly, before the advent of computers, these types of analyses were being performed. The answer to this goes back, in part, to the two problems mentioned in Chapter I. The use of GISs has automated the previous manual techniques of overlaying data for analysis and interpretation. When using the prior manual techniques, the analyst could pick only the more important themes because of time, accuracy, and human

limitations (Pavlidis, 1982). This will become perhaps more evident in later discussion of the characteristics of GISs. The other part of the answer for GISs being computerized is that the computerization enables analysis to be performed that might not otherwise be possible. Since, as Short (1982) put it, "[t]he eye becomes confused and the mind tends to 'boggle' when more than two or three maps are visually compared simultaneously," it is evident that use of a GIS overcomes this human limitation. Short (1982) goes on to say that in performing the analysis manually, ". . . the user simply glances back and forth among several maps and mentally establishes connections and relations." It is evident that such a manual process may be prone to error due to a simple cause such as fatigue. A computer should be able to perform better such analyses. Therefore, it is always important to realize that computers should not be used just to duplicate in an electronic fashion what has been done previously manually, but also that they should be used to accomplish tasks that would have been impossible or impractical manually.

A GIS is a computerized system enabling spatial data to be stored, processed, and analyzed. GISs are used in many different types of studies and applications. A GIS is used to automate and extend prior manual analyses in the decision making process.

Differentiation Between GIS and Management Information System or Data Base Management System

Cicone (1977) has stated there are two generic kinds of information systems: (a) object oriented and (b) spatially or

geographically oriented. He defines object oriented systems to be concerned with "scientific, statistical, or management" types of data whereas spatially oriented systems are concerned with "objects plus location." A number of authors have stated this difference similarly. "A GIS has its data geo-referenced, which distinguishes it from other information systems." (Brooks and Pease, 1978) "GISs differ from other information systems in the need for locational identifiers of the data and the use of those identifiers in manipulating the data." (Dueker and Horton, 1971) "Given a number of basic spatial entities and measures of their location in earth space, the geographic information system must also deal with a host of non-spatial attributes of these entities." (Marble and Peuquet, 1983). Similar statements have been made by Greenlee (1983), Short (1982), and Goldberg (1982).

Examples of object oriented systems, otherwise known as Management Information Systems (MISs) or Data Base Management Systems (DBMSs), are numerous. In reference to Chapter I, a MIS or DBMS could include the daily trading information for the past ten years for every stock on the New York Stock Exchange, an airline's scheduled flights and available seating, or the inventory of a grocery store. An example of the type of data in a GIS could be a vegetation map showing all the different classes of vegetation in a particular area and their geographical extent.

One can argue that a MIS or DBMS can contain geographical information. For example, in an airline's reservation system, the data recorded about each flight might include the departure and

destination cities. This certainly is geographical information. However, such geographical information is of a rather simplistic nature. The vegetation class information, on the other hand, is of a much more complicated nature. Complications arise due to the desire to store and analyze the **geographical extent** of the data. How would an ordinary MIS or DBMS go about storing geographical data which has both location and extent?

Thus, a GIS can be viewed as an extended MIS or DBMS that must deal not only with statistical or tabular data, but must also deal with the spatial location and extent of the data items. This is what differentiates a GIS from a MIS or DBMS, but this is also what makes the implementation of a GIS more difficult. Although, as will be shown, a GIS depends upon a MIS or DBMS as a component, there must be special ways of dealing with the spatial nature of the data that a MIS or DBMS is not normally equipped to handle.

Generic Components of a GIS

Just as the precise answer as to the definition of a GIS at times depends upon the answerer, so too the answer as to what are the components of a GIS may depend upon the person asked. In this section, what can be called the generic components of a GIS will be described. The delineation of the four components will be a synthesis of the ideas expressed by the following authors: Skinner and Gonzalez (1973); Brooks and Pease (1978); Salmen et al. (1977); Tomlinson et al. (1976); Cicone (1977); Alford (1978); Shelton and Estes (1979); Frank (1981); and Marble and Peuquet (1983). This

delineation will be referred to throughout the rest of the thesis as the generic components of a GIS. The four generic components of a GIS are defined as follows:

1. data definition and acquisition;
2. data input and storage;
3. data retrieval and analysis;
4. report generation and display.

Data definition and acquisition deal with what kinds of data are to be input into the system and whence the data shall be acquired. Data definition also includes the idea of how the data shall be represented before input to the GIS. Data acquisition is important in the sense of the GIS having hardware and software compatible with the form of the data.

Data input and storage deal with the how the data shall be input to the GIS and its manner of representation in the GIS. For example, data can be put into the GIS via manual or automatic means. An important consideration in the data input step is the editing procedure for correcting mistakes. In regard to storage, data can be stored in a polygonal or raster format (or combination of the two).

Data retrieval and analysis constitute the heart of the GIS. An important analysis is overlay analysis. That is, two or more types of data in a GIS can be put on top of each other and questions asked about the geographic intersections. For example, one might ask for the geographic areas that contain both historical sites of a certain importance and hardwood vegetation. Such overlay analysis

can become difficult manually when many different themes are overlaid.

Report generation and display deal with how the processed information is presented to the user. Reports could be in a tabular or statistical form. Another important output is a map. For example, a sub-surface minerals developer could be given a map showing the permissible regions in which he may explore.

Data Formats

This discussion falls into two parts. First, what kinds of data are stored in the GIS? Second, in what ways are those kinds of data represented?

As was mentioned before, Marble and Peuquet (1983) stated that the geographical data can be thought of as spatial entities with the addition of non-spatial types of information added on. Salmen et al. (1977) expressed this concept another way. They considered the geographical data to be "image" data and "descriptor" data. That is, image data was considered as graphic or map data. They gave examples ranging from uninterpreted aerial photographs to printed base maps. Descriptor data was considered to be in the form of tables, text, or labels. Both the image and descriptor data are tied together through the geographic or spatial location reference. These ideas go hand in hand with the concept of the GIS being an extension of the MIS or DBMS. The descriptor data is the part that the DBMS would normally be able to handle; it is the image data for which the DBMS must be modified.

The data can also be thought of in terms of being "qualitative" or "quantitative" (Brooks and Pease, 1978; Marble and Peuquet, 1983). Attributes such as land use type, soil type, or vegetation type can be considered to be qualitative in nature. Examples of quantitative data include vegetation density, population density, slope, and, importantly from a geographical standpoint, location, area, and shape. It would be expected that the DBMS would be able to handle the qualitative data and some of the quantitative data; however, modifications would probably be necessary for it to handle some of the geographical data such as location.

The next consideration is how the data is represented. For the descriptor data one need not delve into the details because the conventional DBMS will be handling that part. The only question that one might consider is how much storage the DBMS needs for a particular kind of datum. The answer might then dictate which DBMS to acquire or the answer to the question whether to store numeric data as numbers or characters. The major focus, however, for the designer of the GIS is how the image data should be represented.

Over the years two competing philosophies about storing image (or map) data have come into being. One is called the vector format and the other the raster format (Marble and Peuquet, 1983).

A good description of the vector format (otherwise known as the polygonal format) comes from Irish and Myers (1982):

Polygon-based GIS's employ vector data structures in representing space. Areal entities are geocoded as an aggregation of polygons, where each polygon represents a homogeneous area of a map. An area's boundaries are commonly encoded as a circuit of X-Y coordinates. Encoded polygons are accompanied by unique reference codes that

identify the polygon and serve as relation links to relational links to records in tabular files stored in the data base.

There are actually two types of polygonal systems: (a) the "true" system and (b) the "area boundary" system (Brooks and Pease, 1978). In the true system, each polygon is digitized and stored as a record; therefore, each line is actually digitized twice. In the area boundary system, the boundary between polygons is digitized only once with some additional datum stored to indicate what kind of polygon is to the "left" and the "right."

Another aspect of the polygonal system is that "[t]he basic types of entities considered in computer handling of spatial data are points, lines, and areas" (Marble and Peuquet, 1983). Therefore, a polygonal system must be able to handle point and line types of data in addition to the areal data. Examples of point type of data include utility poles, archaeological sites, manholes, and wells. Line type data could include stream beds, power transmission lines, and bus or subway lines. Areal data types include items already mentioned such as vegetation classes and zones of relative fire danger. To some extent, the scale of the data may play a role whether it is considered point, line, or areal. For example, on a large enough scale, an archaeological site might be considered an area rather than a point.

A description of the raster system (also known as the grid cell system) comes from Brooks and Pease (1978):

In the grid system a uniform grid is superimposed over the study area and the attributes of each grid cell are entered into the system. The grid coordinates may relate to one of the referencing systems previously described [meaning

what geographic coordinate system] or to an arbitrary xy matrix. Depending on the system used, it may be possible to enter one or several attributes or weighted values for each grid cell, but each cell can usually only represent one class for each attribute, i.e., one soil class, one species of tree, etc.

Of course, the same types of data, points, lines, and areas, are represented in the grid cell system, but not in explicit form. Rather, the grid cell system of representing graphical or map data is very analogous to how a digital image represents a photograph.

Naturally, there are various advantages and disadvantages associated with both kinds of data representation. A synthesis of the following authors is given: Wilson and Thomson (1981); Goldberg (1982); Junkin (1979); Greenlee (1983); and Short (1982).

An important advantage of the polygonal system is that an accurate representation of the original map data may be retained. Another way of expressing this is that the boundary detail between features will be retained. This is difficult to achieve with the grid cell system because the storage requirements go up as the square of an increase in the linear resolution. If it is important that the original map lines be reproduced accurately, then more than likely the polygonal system will be necessitated; however, the encoding and manipulation procedures of the polygonal system will probably be more complicated.

A prominent disadvantage of the polygonal method is the difficulty with which it handles overlay analysis. This results from the difficulties in concocting an algorithm which can compute the intersection of overlapping polygons. Such algorithms can easily be inefficient. In addition, another problem called the

"slivers" arises because of errors in the original digitization of the input into the GIS. For example, consider a map of vegetation class data overlaid with a map of soil classification data. Further consider that there is a common natural boundary such as a stream for some of the polygons in both maps. More than likely when the two maps were digitized into the GIS, that stream boundary was not represented precisely the same in the two separate digitization processes. Thus, when the two maps are overlaid in the computer, the stream boundary between the two maps will not exactly match, and a small polygon intersection called a "sliver" could result. Slivers will be particularly prone to result in the true polygonal system in which each boundary is digitized twice. On the other hand, once the data from different sources in the grid cell system is put into a common grid cell size, then overlay becomes very easy compared to the polygonal system. The cells are in effect "geometrically registered," a problem often encountered with multi-scene and multi-temporal analysis in remote sensing. Therefore, if possible, to maintain ease of implementation, the grid cell system should be used for overlay analysis.

If possible, a GIS would ideally combine the best of both the polygonal and grid cell systems. Such a hybrid GIS could use the polygonal system as much as possible for reproducing original map lines and feature boundaries, but it would use the grid cell system when performing the complex overlay analysis. Another possibility is for a GIS to incorporate the idea of "quad trees." Quad trees are interesting because they allow one both to represent data in a

compact form and to operate on the data while in the compact form (Woodwark, 1982; Oliver and Wiseman, 1983). Therefore, it is possible that future GISs will incorporate quad trees in their design.

Coordinate Systems

There are commonly four different georeference, or coordinate systems, in use in the United States (Brooks and Pease, 1978), each with its own advantages and disadvantages:

1. Universal Transverse Mercator (UTM);
2. State Plane Coordinates;
3. Latitude and Longitude;
4. U.S. Township and Range Coordinates.

The Universal Transverse Mercator coordinate system is used throughout the world. It is based upon the metric system for its units. Since it is a projection of three dimensions upon a flat surface, there can be distortion problems. The distortion is usually not important except near edges of what are called the "zones" in the polar regions.

The State Plane Coordinates combine the Mercator and Lambert projections in order to minimize the distortion within a given state. However, the coordinates differ from state to state, thereby making calculations across state boundaries difficult.

Latitude and Longitude, being a true three dimensional coordinate system, can give the most accurate representation of an object's location. But, since Latitude and Longitude are really

spherical coordinates, the manipulations can be tedious when performing transformations from three dimensions to two dimensions. Of course, Latitude and Longitude are used throughout the world also.

U.S. Township and Range Coordinates have been widely used in field surveys and resource records. Because the survey pattern is at times irregular, the manipulations can become a problem.

Outside the United States the metes and bounds method is widely used. This method uses landmarks, iron stakes, and compass directions for specifying location. (Paludan, 1984).

As Brooks and Pease (1978) state, the "choice of a particular geo-referencing system depends upon the user's requirements and objectives and the data sources." Since the U.S. Geological Survey has conversion routines developed between the various systems, in principle it should not be difficult to develop a GIS using any of the systems at will. Wilson and Thomson (1981) state that most of the requests from their users are for results in the UTM system. This author believes that usage will tend toward the UTM system more and more because of its simplicity.

Design Methodologies

Marble and Peuquet (1983) have asserted that the major reason GISs in the past have been underutilized and faulty has been due to poor design. They further state that not initially consulting with the potential users to determine their needs creates problems. This would seem to be common sense. As in the definition of GISs and

their generic components, there seems to be no universal agreement yet as to what elements should constitute the design stage of a GIS. Therefore, in the following shall be discussed several different authors' concepts of what is important in designing a GIS.

Dueker and Horton (1971) state that in a generic sense the design of a GIS can be approached in two parts. First, the conceptual framework for the data must be decided. That is, will the polygonal or the grid cell system be chosen? Second, alternative methods of inputting, retrieving, analyzing, and displaying the data, based on the chosen framework, must be evaluated. Factors such as cost and accuracy would fall under this part.

Shelton and Hardy (1974) state:

Fundamental to the design stage is the concept that an information system should be viewed as an assemblage of technologies to support analysis and decision-making, and the needs of these tasks ought to dictate selection of data and of specific components of remote sensing and computer technology.

A synopsis of their nine steps of design follows.

1. The objectives and purposes of the project should be defined, as well as the scope. For example, what are the immediate needs? Will this be a permanent project or just temporarily used? Will any other agency use the data? What are the constraints on time and cost?

2. Some basic design decisions should be made. What products will be needed (maps, photographs, tables, etc.)? Does a system already exist that fits within the allotted budget? Is it necessary

to use a computerized system? Again, how will the time and cost constraints be met?

3. This and the next two steps should be performed in parallel. An evaluation of the applications should be made. This evaluation should be focused on what types of data will be acquired and put into the system. Not only should present data be examined, but some thought as to future data should be given.

4. An evaluation should be made as to what technical alternatives exist for accomplishing the project. For example, the various current remote sensing technologies for acquiring and interpreting data could be examined. The various types of computer equipment and software and their prices and capabilities should be examined, as well as the availability and service provisions.

5. The organizational aspects of the project should also be considered. This falls into three categories. First, who is providing the leadership to the project? Will it be the initial agency that wanted the project, or will they turn it over to somebody else? Second, who performs the technical development of the project? Can the agency do it in-house, or will they have to contract it out? Third, after the project is complete, who will administer it? Where will the GIS be located, and who will be responsible for continued service and training?

6. Now tentative decisions can be made about the technical framework of the project. For example, the types of maps and scales, the sources of data, the necessary level of resolution or detail, and the coordinate reference system can all be considered.

Also, the types of analyses to be performed upon the data should be clearly defined. These decisions need not be final, but the results of Step 3 should be incorporated.

7. Selection of the components of the GIS can be made. Such items as whether the method of input should be manual or automated should be considered. The level of detail required to be kept about the data will play a role in determining the equipment. An evaluation of the types of analyses to be performed will determine whether commercially available software will suffice or whether special software must be developed. All these items must be kept within the time and cost constraints as well.

8. An implementation plan should be developed. This plan should summarize the time and cost estimates and how the decisions made in the prior steps meet the project's goals.

9. Lastly, a pilot project should be developed which implements on a small scale the decisions made about the project. This enables the implementation staff to find where problems exist in order that they may be corrected before too much effort has been expended. This is a very important stage in the project as it allows refinements of the ideas expressed in the prior steps before commitment to the final design.

Skinner and Gonzalez (1973) have two principles of design for GISs. First,

An earth resources information system should accept data from many different sources, dealing with many different subject areas; process the data and manage the resulting information to effectively answer a broad spectrum of queries relating to earth resources and the environment.

Their second principle says that a "comprehensive earth resources information system must provide for interaction with, or inclusion of, existing specialized systems with useful capability."

Finally, Marble and Peuquet (1983) state four parts to a GIS design methodology:

1. description of information system objectives and specifications;
2. definition of organizational resources and constraints toward system development;
3. generation and evaluation of alternative system designs;
4. overall evaluation of costs and benefits for system selection.

As can be seen, there is not always agreement on what constitutes good GIS design. However, as Shelton and Hardy (1974) say, "Our point is primarily that there should be a design."

CHAPTER III

DESIGN OF THE BIG SOUTH FORK GIS

Of the four different design methodologies or design principles mentioned in Chapter II, this author finds the nine design steps due to Shelton and Hardy to be the most straightforward, concise, and meaningful. Therefore, it was decided to apply their nine steps to the problem of the Big South Fork. As they state, "[t]here are very few firm principles in designing the kind of project discussed [in their paper]" (Shelton and Hardy, 1974). It is important to remember in reading the following discussion that it was not always possible to follow these nine steps precisely. Rather, what shall be pointed out is how the Big South Fork GIS design fits into these nine steps of Shelton's and Hardy's. Since every project is more than likely different from other projects, the design steps must be flexible to accomodate such differences.

In the Step 1, there are two questions to be asked. One, what are the objectives and purposes? Two, what is the scope?

The Big South Fork area comprises over 125,000 acres with only 15 - 20 people to manage it. These people represent a diversity of skills and are charged with not only seeing that the public can enjoy recreation but also for overseeing limited commercial development. Currently, in order to carry out their charge, the Big South Fork personnel must wade through dozens of maps, charts, reports, tables, and statistics.

Therefore, the major objective of the GIS is to reduce the time

needed to perform analyses of the data by an order of magnitude from several weeks to several days. The primary scope is to perform the project within the constraints of available money and the required software and hardware.

In Step 2 a few basic design decisions are to be made. What products are important? Is the accuracy of the boundary detail on maps important? What are the constraints?

The Park Service personnel felt that both tabular or statistical reports and map output are important. They also felt that boundary detail should be accurately represented. Constraints also existed in terms of the hardware and software to be used in the project.

Step 3 is a consideration of what types of data will be put into the GIS, both present data and future data. During discussions with the Big South Fork personnel (Duwe, 1983), approximately 25 different types of data were identified as being of current interest. The terminology for these different kinds of data is the word "themes." That is, a theme represents a particular kind of data.

The themes identified are:

1. vegetation maps;
2. elevation;
3. slope;
4. aspect;
5. fire hazard areas;
6. deferred areas;

7. transportation systems;
8. utilities;
9. geological formations;
10. soils maps;
11. above surface mineral rights;
12. below surface mineral rights;
13. political boundaries;
14. mineral resources;
15. waterfalls;
16. developed areas;
17. endangered species;
18. archaeological sites;
19. historical sites;
20. water quality monitoring stations;
21. kudzu sites;
22. rivers and streams;
23. exotic animals;
24. agricultural leasing areas;
25. erosion areas.

Many different example themes were identified, some of which represent data already acquired, others data to be acquired. Most importantly, it was found that the ability of the users to define their own themes in the future is a critical design goal for the success of the GIS.

In Step 4, the technical alternatives are to be examined. The examination of the alternatives should fall under two categories:

a) acquisition of data and b) the equipment for the GIS itself.

It is irrelevant to the design of the GIS how the data is acquired as long as it is in a format compatible with the GIS's modes of entry. Sophisticated remote sensing acquisition and analysis or field work, to use two extremes, may be employed. Consideration of selection of the hardware and software was not possible because of Park Service constraints.

In Step 5 organizational aspects of the project are to be considered. Who provides the project's leadership? Who provides the technical development? Who administers the project after it is finished?

The Park Service, in conjunction with the Corps of Engineers, provides the coordination of the project, UTSI provides the technical development, and the Big South Fork administers and uses the system after its completion.

In Step 6 decisions are to be made as to the technical framework of the project. In particular, the themes identified in Step 3 should be kept in mind when working on this part of the design. Several decisions have been made and are discussed below.

The UTM system of reference was chosen because of its simplicity and wide use. Maps of any scale, as long as they use UTM, can be entered. The polygonal system will represent the themes, and it is realized that the software must let the user define a theme as either a point, line, or polygon with the appropriate accompanying attributes. Single-theme and multi-theme analysis are identified as being desirable, with the realization

that multi-theme analysis will present problems since the polygonal format was chosen. Tabular and map products are identified as being necessary with a dot matrix printer and color pen plotter being seen as sufficient to produce such output.

In Step 7 selection of the components of the GIS is to be made. Because of internal Park Service decisions, the heart of the system is an IBM Personal Computer with 256K RAM, 8087/8088 math coprocessor pair, two floppy disk drives of 354K capacity each, color monitor, and keyboard. Using the concept of the four generic components of a GIS developed in Chapter II, the other hardware and software selected by the Park Service for this project will be listed by those categories.

1. Data definition and acquisition. Since this topic deals with defining the data and the means by which the data is acquired for input into the GIS, there is no equipment pertinent for this project.

2. Data input and storage:

- a. GTCO Digi-5 Digitizing Pad;
- b. Corvus 20 MB hard disk with mirror backup.

3. Data retrieval and analysis:

- a. dBASEII relational database management software by Ashton-Tate;
- b. PASCAL language by Microsoft;
- c. interpreter BASIC by Microsoft.

4. Report generation and display:

- a. Epson FX-100 Dot Matrix Printer;

b. HP 7470A Personal Pen Plotter.

In Step 8 an implementation plan is to be developed. The plan for this project is simple. The first part is to perform a literature search to ascertain what work had been done before. The literature search also serves to give a consistent set of generic definitions about GISs. The second part is to relate what had already been done on the project under the initial time constraints and what remained to be done in the way of a design methodology. The third part is to carry out initial verification programs on the techniques proposed in the design. The fourth part is to assess the results of those initial pilot programs. The fifth and last part of the plan will be to decide the next phase of the software design and prepare the pilot programs for translation into a user friendly system.

Chapters I and II are the result of the first part of this plan. Chapter III is the result of the second part. Chapter IV is the result of the third part, and Chapter V is the result of the fourth part.

In Step 9 of the design a pilot study of the techniques decided upon in the first eight steps is to be conducted. In the next chapter, the verification programs developed to test the equipment and techniques chosen are discussed.

CHAPTER IV

PILOT STUDY OF BIG SOUTH FORK GIS

Conducting a pilot study of techniques available in a project is an important step. The test of the techniques allows one to assess their actual utility, rather than their proposed utility. It is to be hoped that the weaknesses of the techniques will be found and recommendations made as to either corrections or where more study is needed. The pilot study also gives the developers of the project a model from which to judge at what points techniques such as "user friendliness" must be interjected. Since the pilot study is done on a small scale, changes can be made easily and their impacts viewed quickly.

As will be recalled, there are four generic components in a GIS. As was mentioned in Chapter III, this project did not involve the first component, which was the acquisition of the data to be input into the GIS. Therefore, the pilot study was only concerned with the other three components, the data input and storage, the data retrieval and analysis, and the report generation and display. The following discussion will divide the pilot study into these three areas.

Data Input and Storage

The steps developed to effect data input and storage are:

1. if it is a new data base, use dBASEII to define the data base and its structure;

2. if it is a new data base, run the STRUCTUR program (written in dBASEII) to translate the data base structure into a format that BASIC and PASCAL can utilize;

3. run the DIGITIZE program (written in BASIC) to gather information from the digitizing pad and supply the other information as defined in the data base structure;

4. run the TRANSLAT program (written in PASCAL) to compute lengths and areas, as applicable, of the geographic entities, store the boundary information, as applicable, and translate the other information into a form that dBASEII can utilize;

5. run the FILLBASE program (written in dBASEII) that puts the gathered information into the data base for later access and analysis.

It should be noted that there is no specific program to lead the user through Step 1. Instead, it must be assumed that the user is knowledgeable enough about dBASEII to create a new data base and its structure. As will be recalled in earlier discussion, it is an important goal that the Park Service users be freed as much as possible from any restrictions. The point of using dBASEII in this project is to give the users the freedom to define their data bases and to ask questions about relations among the data without hindrance. If a program had been written to perform dBASEII functions, such as defining a data base's structure, such a program would more than likely be more restrictive than the dBASEII itself. Also, because of time and money constraints it was felt the developers should spend the majority of their effort on developing

the specific capabilities needed to implement a GIS. Therefore, it is imperative that the Park Service personnel learn how to operate dBASEII in a comfortable manner. They must know how to define new data bases, manipulate existing data within the data base, and select the appropriate data for output.

Therefore, in Step 1, the user must know how to use the dBASEII "CREATE" command. With the CREATE command he will create an empty data base and define its structure. No matter what kind of geographical entity (theme) his data base is to represent, there are four required pieces of information. These four pieces represent the first four fields of each record in the data base. These four fields are TYPE, FILENAME, REF:EAST, and REF:NORTH.

TYPE is a character field defined to be of width 1. Its value will be either a "P", "L", or "G" for "point", "line", or "polygon" respectively. Thus, TYPE represents which kind of the three basic geographical entities is being stored in a record in the data base.

FILENAME is a character field of width 8. It represents the name of the disk file where the appropriate boundary information, if the entity is a line or polygon, is stored. Its width is only 8 characters because the IBM's Disk Operating System (DOS) limits filenames to only 8 characters.

REF:EAST is a numeric field of width 7 (including the decimal point) with three digits reserved to the right of the decimal point. It represents the "pseudo-easting" of the reference point of the entity, be it a point, line, or polygon. It is a pseudo easting because it represents the easting in terms of kilometers rather than

meters. This is acceptable and appropriate because most 7-1/2' quad sheets are marked in UTM with kilometers. As will be seen later, this is also convenient for the graphics output capability.

Eastings in the Big South Fork area are in the vicinity of 700 km.

REF:NORTH is a numeric field of width 8 (including the decimal point) with three digits reserved to the right of the decimal point. It represents the "pseudo-northing" of the reference point of the entity, be it a point, line, or polygon. It too is pseudo because it represents the northing in terms of kilometers rather than meters. The quad sheets also have the northings marked in kilometers. This representation is also convenient for the graphics output. Northings in the Big South Fork area are in the vicinity of 4000 km.

If the entity represented by the data base is the line type, the next attribute must be LENGTH. It is a numeric field of width 10 (including the decimal point) with 6 digits reserved to the right of the decimal point. It represents the length of the line entity in kilometers since all references to entities are in terms of kilometers. If the entity is the polygon type, the next attribute must be AREA. It too is a numeric field of width 10 (including the decimal point) with 6 digits reserved to the right of the decimal point. It represents the area of the polygon entity in square kilometers. If the entity is the point type, there is no next required attribute.

The other attributes, if any, defined in the data base structure by the user with the CREATE command are of his choice.

What they are is not of importance to the rest of the software. The user may have as many more as desired as long as the fundamental dBASEIII constraints of no more than 32 fields (attributes) per record and no more than 1000 characters per record (with no more than 255 in any one field) are not exceeded. In addition to the character and numeric field types described above, one other type of field is available to the user. It is called the logical field. Its width is always 1, and its value must be either "true" or "false."

In the pilot study, for example, three different themes exist. The first theme is named VEG for vegetation polygons, and it contains the following attributes: TYPE, FILENAME, REF:EAST, REF:NORTH, AREA, CLASS, COMP:NO, and QUAD:NAME. CLASS is an attribute describing the classification of the vegetation, COMP:NO describes the compartment number inside the park within which the entity was found, and QUAD:NAME is the name of the quad sheet on which the entity is located. The second theme is named STREAM for streams running through the Big South Fork and contains the following attributes: TYPE, FILENAME, REF:EAST, REF:NORTH, LENGTH, and CLASS. CLASS is an attribute describing whether the stream is wet or dry. The third theme created is called HISTORICAL for important historical sites located in the Big South Fork area. It contains the following attributes: TYPE, FILENAME, REF:EAST, REF:NORTH, and IMPORT. IMPORT contains a value corresponding to the importance attached to the site.

After exiting the CREATE command, the user does not put

directly into the data base any data yet. The user should check his data base structure for the accuracy of his definitions. If a mistake in the structure has been made, it should be corrected before any data is entered as it becomes somewhat complicated in dBASEII to change a data base structure after it contains data. This Step 1 is performed only when the data base is first created. It is not performed again unless changing the structure of the data base and then only if one understands how to manipulate dBASEII so that the original data, if desired to be kept, is not lost.

After assuring the accuracy of the data base structure, the user performs Step 2 by running the developed program STRUCTUR (written in dBASEII). This program will ask one question: What is the name of the theme (= name of data base)? This program takes the data base's structure and puts it into the "System Data Format," as dBASEII terms it, which is another name for standard ASCII format. A disk file is created containing the structure in the ASCII format. This is necessary because the BASIC and PASCAL programs will need to refer to the structure, and they cannot do so if the structure is left in dBASEII format.

After Steps 1 and 2 are finished, Steps 3, 4, and 5 may be performed as often as desired, but once Step 3 is started, one must continue all the way through Step 5 before going back to Step 3. Step 3 is the point at which the information on a map is digitized. The BASICA version of the Microsoft BASIC language on the IBM is entered and the developed program DIGITIZE run. The first step in DIGITIZE is, assuming the base map has been fastened to the

digitizing tablet, to select three reference points on the map, tell the computer their real world coordinates (in eastings and northings), and digitize the corresponding points on the map. Next, the name of the theme being digitized is asked for. After this preliminary information is entered one time, the program goes into a loop. The loop first asks the user to digitize the reference point of the entity. If the entity is a point, the user is directly prompted to supply the values of the other attributes as determined from the ASCII structure file. If the entity is a line or polygon, the user is asked to digitize the boundary points before being prompted for the values of the other attributes. This data is stored in a temporary data file, and the user is asked whether to continue the process.

When the user has finished a given digitizing session and discontinued the program DIGITIZE, the computer is taken out of the BASICA mode and put into the DOS mode so that PASCAL may be run. Step 4 is to run the TRANSLAT program, developed in PASCAL. TRANSLAT takes the raw data acquired through DIGITIZE, both from the digitizing pad and supplied as answers by the user, and processes it. First, the three reference points on the base map are used to develop the equations necessary to convert all succeeding digitizing pad coordinates into the real world coordinates. The program uses a technique due to Allendoerfer and Oakley (1969). According to this method, if there exists a point $P(x,y)$ in a plane in one coordinate system, it may be transformed into the point $P'(x',y')$ in a different coordinate system (which may have undergone translation,

dilatation, rotation, reflection, or a combination, with respect to the unprimed coordinate system) by the following equations:

$$x' = a_1x + b_1y + c_1$$

$$y' = a_2x + b_2y + c_2$$

Therefore, by digitizing three distinct points, there will be six equations with six unknowns (a_1 , b_1 , c_1 , a_2 , b_2 , and c_2). TRANSLAT then solves these six equations using Cramer's rule. The next step in TRANSLAT is to convert all the digitizer's coordinates describing the reference point and boundary points, as applicable, of the entities into the real world coordinates of eastings and northings.

The next step in TRANSLAT, depending on which kind of entity is being examined, is to determine the length or area (or do nothing in the case of a point). For a line entity, the length is determined using the distance formula for points in a plane. Successive distances between pairs of points in the digitized line are computed and added for the total. Since the boundary points have been converted to eastings and northings, the length comes out in kilometers as desired. For a polygonal entity, the area is computed via a technique due to Du Val (1966). For a plane polygon, if $P_i(x_i, y_i)$ for $i = 1$ to N represent the boundary points of the polygon, then the area A may be expressed as:

$$A = 0.5*(x_1y_2 - x_2y_1 + x_2y_3 - x_3y_2 + \dots + x_{N-1}y_N - x_Ny_{N-1} + x_Ny_1 - x_1y_N)$$

In this representation, computing the area requires for N points

2N+1 multiplications and N-1 additions. However, this formula may be written in a more computationally efficient manner:

$$A = 0.5*[x_1(y_2-y_N) + x_2(y_3-y_1) + \dots + x_{N-1}(y_N-y_{N-2}) + x_N(y_1-y_{N-1})]$$

In this representation, the computation requires only N+1 multiplications and still N-1 additions. This latter form is the one programmed into TRANSLAT.

After the length or area has been computed (or nothing done for a point entity), a boundary file on disk is created, using the name contained in the attribute FILENAME (except for a point entity which doesn't need a separate boundary file). Finally, the attribute information is written to a temporary data file in ASCII format with the spacing determined by the ASCII structure file. TRANSLAT must put its results for dBASEII into ASCII format because that is, again, the only way in which dBASEII can interact with other languages and processors.

After TRANSLAT has finished, dBASEII must be entered in order that the last step may be performed. Step 5 is to run the developed dBASEII program called FILLBASE. FILLBASE takes the temporary ASCII format data file resulting from TRANSLAT and appends the records into the proper data base. The unneeded data files are then erased in a housekeeping function. When FILLBASE has finished, the data base contains the information acquired from the DIGITIZE step and processed by the TRANSLAT step.

Steps 3, 4, and 5 may be repeated as often as necessary.

However, as mentioned before, when Step 3 is finished, Steps 4 and 5 must be performed; otherwise, what was done in Step 3 is lost. As much data as desired may be prepared for input into the appropriate dBASEII data base file with the only constraints being how much disk storage one has and the fundamental limitation of dBASEII being able to store only 64K records in one data base file (i.e., only 64K entities of any one theme).

As will be recalled, it was mentioned that editing of data is important. This may be done in conjunction with the graphical output programs described later. Two types of editing may be performed. One type pertains to the graphical data. Do the digitized features match closely with the source information? If not, the appropriate record in the data base file must be deleted and Steps 3, 4, and 5 repeated for the feature. The other type of editing pertains to the descriptor data contained in the data base (the attributes or fields). For example, the COMP:NO of a vegetation polygon might have been entered incorrectly. In this case, one may use the editing features of dBASEII itself since the error is contained solely within the data base file. There is a dBASEII command called "EDIT" which allows one to examine the contents of any record and modify the fields as necessary. Again, this capability assumes a working knowledge of dBASEII and its capabilities.

Data Retrieval and Analysis

The steps developed to effect data retrieval and analysis are:

1. for single-theme analysis use dBASEII commands only;
2. for multi-theme analysis:
 - a. use dBASEII commands;
 - b. run the GRIDCELL program (written in PASCAL);
 - c. run the FILLOVER program (written in dBASEII);
 - d. use dBASEII commands again.

This is the most important part for which the user must be familiar with dBASEII. The power of the data base language is in freeing the user to query the data base without having had to define the relations among the data beforehand. Therefore, if the users have command of the dBASEII language, they are not dependent upon anybody else's restricting the questions that may asked about the data. It cannot be emphasized too much how imperative it is that the end user be familiar with the data manipulation commands available to him in dBASEII.

Single-theme analysis entails asking questions of the data contained in just one theme. For example, take the VEG vegetation polygon theme created in the pilot study. A typical query might be to know what vegetation polygons of Class 5G with area greater than 0.001 km^2 but less than 0.102 km^2 in Compartment 63 exist. The dBASEII command to perform this would be:

DISPLAY ALL FOR CLASS='5G' .AND. AREA>0.001 .AND. AREA<0.102

Since two of the attributes for entities are the easting and northing for the reference point, a query could be based on that information:

```
DISPLAY ALL FOR REF:EAST>690.0 .AND. REF:NORTH<7052.4
```

The complexity of the relations among the data that may be expressed is limited only by the length of a command line in dBASEII, 255 characters. However, this is only an inconvenience as one could cascade the relational queries one after another until finally all but the desired records were weaned out. No programs need be developed for this kind of analysis as the dBASEII language has the capability already to accomplish this.

Multi-theme analysis is conceptually easy to understand, but it takes a bit of effort to accomplish. Conceptually, multi-theme analysis is like placing several single themes on top of each other and being able to look through the themes. One might ask, using the VEG and STREAM themes of the pilot study as examples, what areas contain wet streams running through vegetation polygons of Class 6P in Compartment 54. Conceptually, one would look through transparent maps representing the two themes and visually find the intersection of those wet streams with the desired vegetation polygons. For two or three themes this might not prove too difficult, depending on the nature of the query. As cited in earlier discussion, it would be difficult, if not impossible, to do such analysis visually with more themes.

As will also be recalled, it was stated that when the GIS data is stored in polygonal format to enhance the reproduction of boundaries, it also becomes more difficult to perform the multi-theme analysis. This is because one must worry about how to calculate the intersections of boundaries, both of line and polygon entities. Therefore, to perform more easily multi-theme analysis, it is desirable that the GIS data be manipulated in a raster form. The procedure described below shows a method by which this can be done for the Big South Fork GIS.

First, in multi-theme analysis, it is necessary that the user be able to perform some single-theme analysis using the appropriate dBASEII commands as discussed above. Refer back to the example of overlaying wet streams with vegetation polygons of Class 6P in Compartment 54. The analyst needs first to separate the wet streams from the dry streams with a dBASEII command like this:

```
DISPLAY ALL FOR CLASS='WET'
```

When he is satisfied that these indeed are the streams desired, he needs to put those records into a temporary data file in ASCII format so that the PASCAL program GRIDCELL can act upon them:

```
COPY TO STREAM.TMP ALL FIELDS
TYPE,FILENAME,REF:EAST,REF:NORTH SDF FOR CLASS='WET'
```

Next, a similar analysis must be done on the vegetation polygons. After calling up the VEG data base, the analyst might type:

```
DISPLAY ALL FOR CLASS='6P' .AND. COMP:NO='54'
```

It is good practice to view on the screen the results of the relation(s) among the data in order to see how much data that represents and to ensure the relation(s) was(were) expressed properly. Alternatively, dBASEII has a command which could be used to count the number of records fitting the specified criteria. Then, the records must be copied to the temporary data file:

```
COPY TO VEG.TMP ALL FIELDS TYPE,FILENAME,REF:EAST,REF:NORTH  
SDF FOR CLASS='6P' .AND. COMP:NO='54'
```

If there were more themes to be overlaid, a similar procedure would be followed for each one. It is important to note that the techniques of single-theme analysis must be used first in order to isolate the entities within a theme that are to be overlaid with entities from other themes.

The next step in multi-theme analysis is to run the developed program GRIDCELL (written in PASCAL). GRIDCELL prompts the user for each theme under consideration and assumes that the proper temporary ASCII format data file for each theme has been created as outlined above. Everything else in GRIDCELL is automatic in operation.

GRIDCELL creates a raster cell layer for each theme under consideration. As each theme is examined, the appropriate cells in the raster cell layer are marked as to the presence or not of the entity. GRIDCELL uses techniques developed in Foley and Van Dam (1982) to perform the conversion of the data being stored in

polygonal format to raster cell format. Converting points to raster cells is straightforward; the raster cell containing the point is marked as being true to the presence of the point. A basic incremental algorithm is used in converting polygonal format lines to their raster equivalents. Every raster cell through which a line entity passes is marked as being true to the presence of the line. For converting polygonal polygons to raster format, a recursive 4-connected region fill is used. The cells through which the boundary of the polygon passes are marked as being true; then, using as a starting point the reference point defined by REF:EAST and REF:NORTH, the cells above, below, to the right, and to the left are examined to see if they are already marked. If not, they are marked, and the cells above, below, to the right, and to the left of each of those four are examined in the same way, and the process is repeated recursively until the interior region is "flooded." PASCAL's ability for a procedure to call itself enables this procedure to be simply implemented.

After each raster cell layer for each theme under consideration has been created, GRIDCELL creates a master data base called OVERLAY. There are $N+4$ fields in each record of this data base, where N is the number of themes being overlaid. The number of records is equal to the number of cells in the raster cell layers, which depends upon the geographic area being covered and the size of each cell. The first N fields of each record are logical fields, i.e., they have a value of either true or false. These first N fields correspond to the presence or not of each theme in the record

(or cell). The last four fields in each record correspond to the minimum easting of the cell, the minimum northing of the cell, the maximum easting of the cell, and the maximum northing of the cell. In other words, the last four fields describe the real world coordinates of opposite corners of the cell. Therefore, in our STREAM and VEG overlaying example, there is a total of 6 fields in each record. The first field is whether a stream exists in that cell. The second field is whether a vegetation polygon exists in that cell. The third through the sixth fields describe the geographic coordinates of the cell. Again, it is important to note that whether a cell has the existence of a particular entity depends on the selection made in the first part of the multi-theme analysis. Only those entities copied to the temporary data files are converted into raster format. The master data base created by GRIDCELL, called OVERLAY, is put into ASCII format. This, again, is due to the manner in which dBASEII and other languages can communicate.

The names of the first N fields of OVERLAY correspond to the names of the themes being overlaid. The order in which they come depends upon the order in which they were specified to GRIDCELL (this order is independent of the order in which the temporary data files were created in the first step). The names of the last four fields are EAST:MIN, NORTH:MIN, EAST:MAX, NORTH:MAX. One might ask how many themes can be overlaid at one time. Since dBASEII has a maximum of 32 fields per record, the maximum N is 28.

The third step in multi-theme analysis is to run the developed program FILLOVER (written in dBASEII). FILLOVER takes the ASCII

format data base OVERLAY and copies it into a dBASEII format data base also called OVERLAY. This process is automatic and requires no intervention from the user.

Lastly, to complete the multi-theme analysis, the user must call up the data base OVERLAY. Now the analyst may ask the question, in our example of the STREAM and VEG overlay, what cells contain the presence of both the streams and vegetation polygons:

DISPLAY ALL FOR STREAM .AND. VEG

If there had been more than two themes overlaid, then the appropriate combination could have been asked. Again, it is important to note that in this step of the multi-theme analysis, only the presence or not of a theme in a raster cell can be ascertained. Selection criteria based upon the attributes of the theme, such as compartment number, area, or class, must have been chosen in the first step analogous to single-theme analysis.

Report Generation and Display

The steps developed for effecting report generation and display are:

1. for tabular, statistical reports use dBASEII commands;
2. for graphical output:
 - a. use dBASEII to select the proper records;
 - b. use POLYGRAF or OVERGRAF (written in PASCAL) to put the

proper records (data in polygonal format or multi-theme analysis cells respectively) to the HP plotter.

The first type of report generation and display requires, again, a working knowledge of dBASEII. dBASEII has the ability to compose reports based upon the user's selection of relations among the data. Those reports may be displayed upon the screen or sent to the printer. For example, one might ask for a summary of all vegetation polygons of Class 6P in Compartment 54 and their total area. Report forms may be stored in dBASEII and called upon at any time. Reports could also be generated with the OVERLAY data base. For example, how many cells exist containing historical sites of importance greater than 4 and vegetation polygons of Class 6P? No programs need be developed to lead the user through this kind of report generation as that capability exists within the dBASEII language.

The other type of report generation and display is of a graphical nature. Rather than seeing a statistical report of entities, it may be preferable to view them. The first step in doing this is to call up the appropriate data base while in dBASEII. Next the appropriate records should be selected and put into a temporary ASCII format data file. In the case of regular GIS data bases in polygonal format, the final command might look like:

```
COPY TO VEG.TMP ALL FIELDS TYPE,FILENAME,REF:EAST,REF:NORTH
SDF FOR CLASS='6P' .AND. AREA>0.010 .AND. COMP:NO='53'
```

Notice this is the same type of command and file name that had to be

done in the multi-theme analysis prior to running the program GRIDCELL. If the graphical display were to be performed upon the OVERLAY raster cell data base, the command might read:

```
COPY TO RASTER.TMP ALL FIELDS  
EAST:MIN,NORTH:MIN,EAST:MAX,NORTH:MAX SDF FOR STREAM .AND.  
VEG
```

In either case, an ASCII format temporary data file is formed so that the plotting programs, written in PASCAL, can access the desired entities.

If the entities to be graphed are polygonal, the developed program POLYGRAF is run next. It asks which theme is being graphed, and it asks some information on the geographical area's limits. If OVERLAY's raster cell entities are being graphed, OVERGRAF is run. OVERGRAF asks the appropriate questions about the limits of the geographical area being plotted. Both POLYGRAF and OVERGRAF rely heavily upon developed PASCAL software which translated the HP plotter's graphics language into PASCAL procedures. These developed procedures allow the user the convenience of communicating with the plotter in easy to use PASCAL procedures and allow the capability for more complex procedures to be built based upon the simple ones.

Because the same sheet of paper can be put back into the plotter with a high degree of accuracy, it is possible, through repeated applications of POLYGRAF, to overlay several themes in different colors if colored pens are available for the plotter. Alternatively, an overlay analysis may be performed using the

previously outlined procedure, then repeated applications of POLYGRAF and OVERGRAF on the same sheet of paper can show the themes in their original polygonal format and the results of the overlay analysis in the raster cell format.

It was mentioned earlier that the pseudo eastings and northings prove to be convenient for the graphics. This is because when specifying to the HP plotter one's real world coordinates to be assigned to the plotter coordinates, only real world coordinates between -32767 and +32767 may be used. Therefore, the simplest choice is to represent the eastings and northings in kilometers in order to make the readings within the plotter's acceptable range.

As was mentioned in the sub-section on data input and storage in this chapter, editing is performed in conjunction with the graphical display capability. Through the use of POLYGRAF, themes may be graphed in order to be checked for their boundary detail accuracy.

Examples of Output

The three different themes in the pilot study exist on a "fake" quad sheet. Simple maps were used so that the results would be easy to see.

Figure 1 shows the seven historical sites existing in the HISTORICAL theme. Figure 2 shows the two streams existing in the STREAM theme. Figure 3 shows the three vegetation polygons existing in the VEG theme. All three figures were produced with the POLYGRAF program.

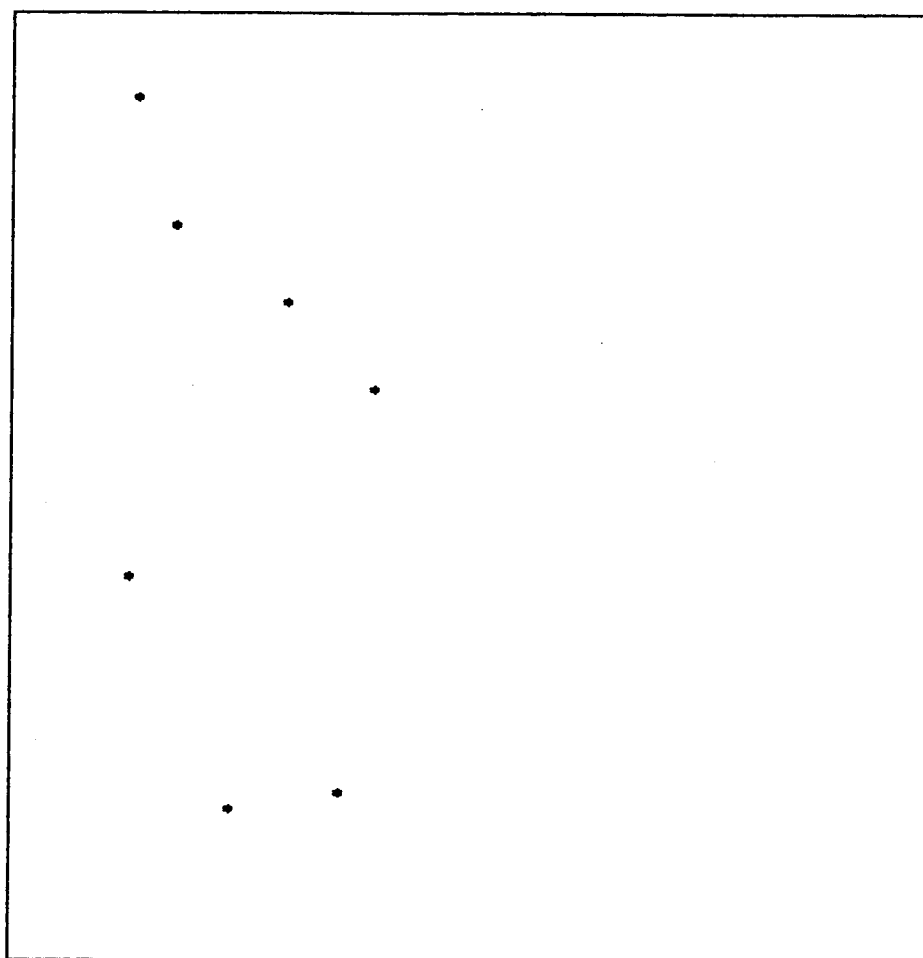


Figure 1. Seven historical sites from the pilot study theme called HISTORICAL.

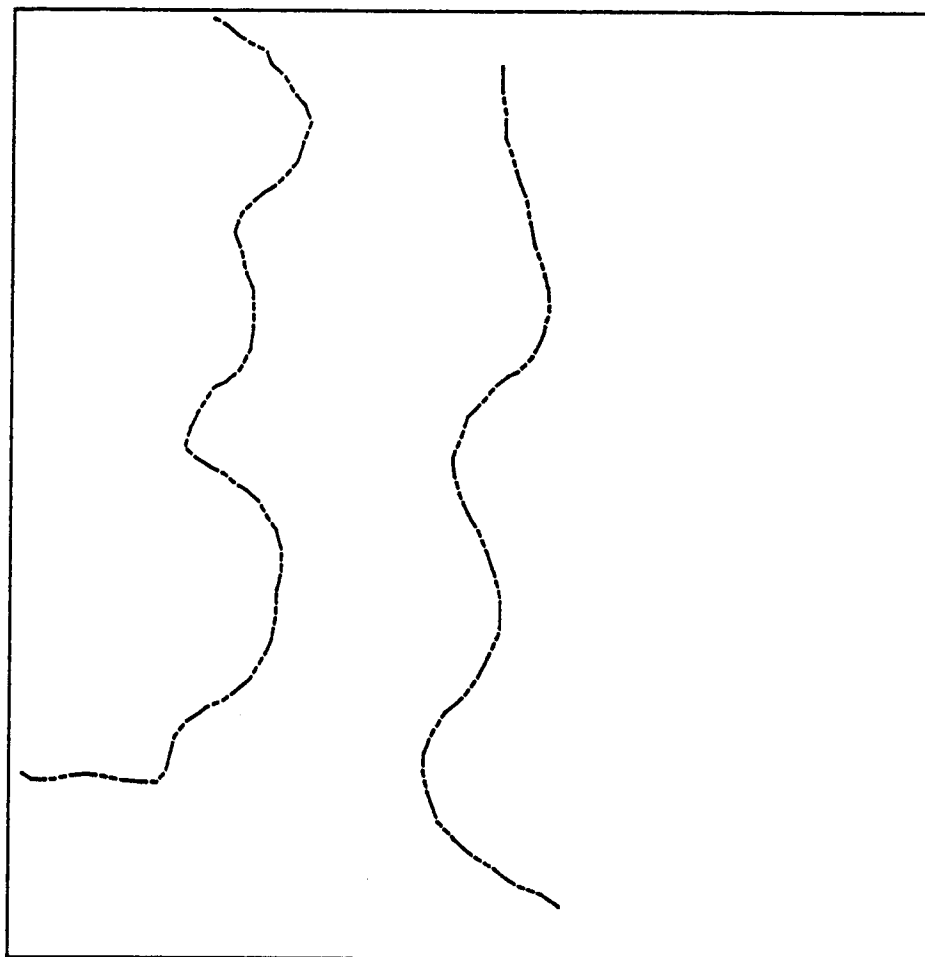


Figure 2. Two streams from the pilot study theme called STREAM.

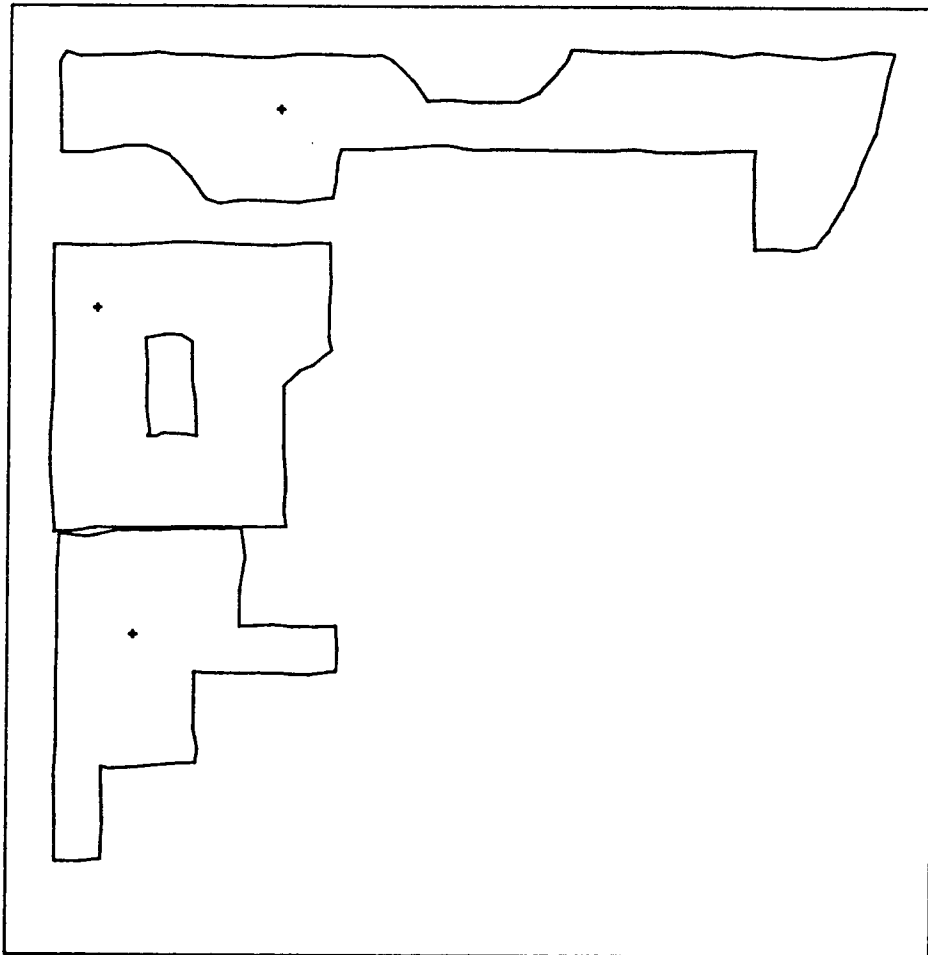


Figure 3. Three vegetation polygons from the pilot study theme called VEG.

The question was asked as to what areas contain both stream and vegetation data. Figure 4 shows the grid cells which are the answer to this question. Figure 4 was produced by the OVERGRAF program.

Figure 5 is a composite of Figures 1 - 4 produced by plotting all four figures on the same sheet of paper (which gives a good indication of how well the same sheet of paper can be repositioned in the plotter).

Summary

A set of procedures and programs has been developed to test the preliminary design of the Big South Fork GIS. These procedures and programs cover the areas of data input and storage, data retrieval and analysis, and report generation and display. A working knowledge of the dBASEII language is a necessity in order that the manual dBASEII procedures may be followed intelligently. BASIC and PASCAL programs accomplish the rest of the procedures. Therefore, the implementation of the GIS is a combination of the user's intelligence and discrimination and the machine's number crunching ability.

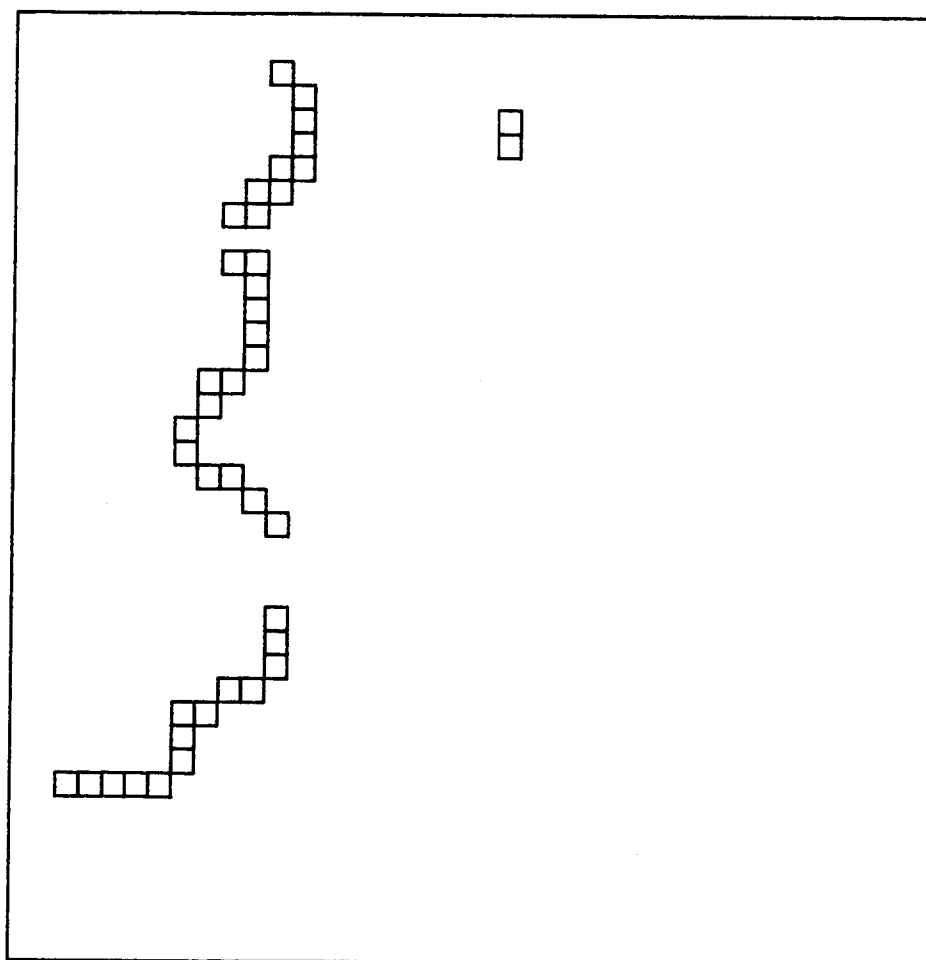


Figure 4. The grid cells which represent the intersection of the streams in Figure 2 and the vegetation polygons in Figure 3.

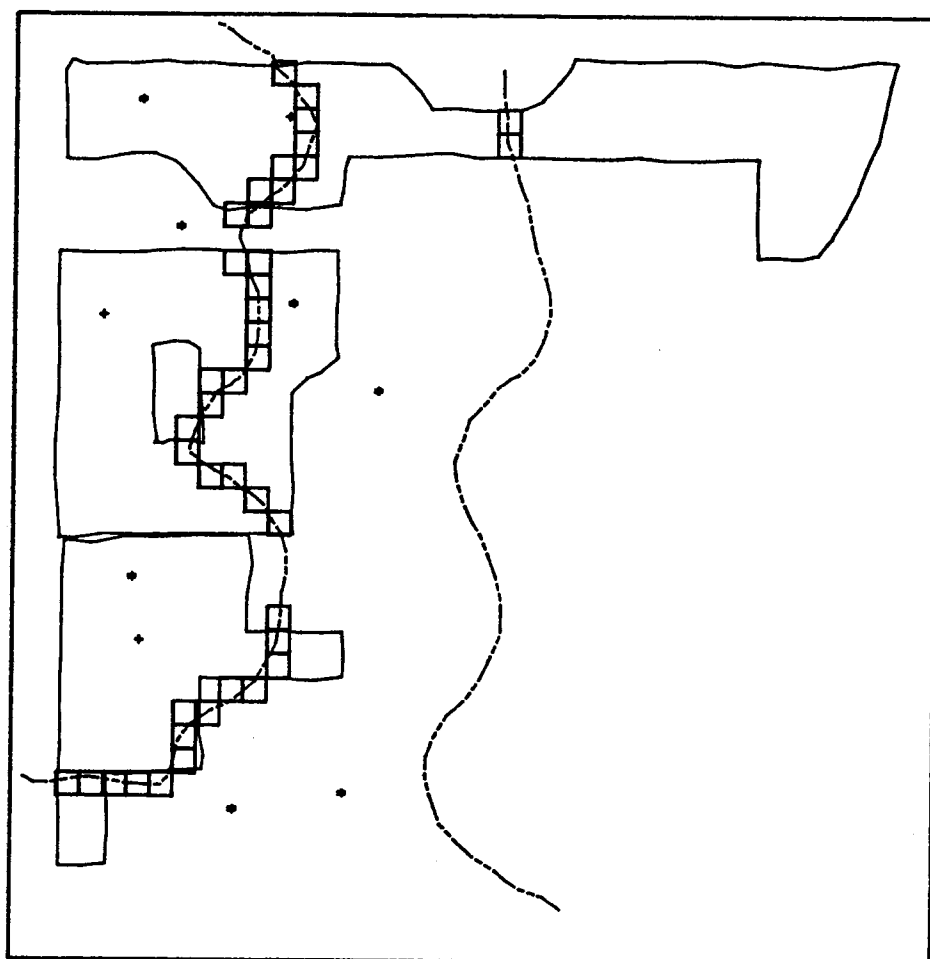


Figure 5. A composite of Figures 1, 2, 3, and 4.

CHAPTER V

ANALYSIS AND DISCUSSION

Since the description of the pilot study procedures and programs was broken into three areas, this discussion will follow suit.

Data Input and Storage

Because of the necessity of the user knowing dBASEII, it is imperative that he be trained well in its use. He must know how, in this part of the GIS, to create the structure of the data base.

A good technique in this part is the ability to define the data base structure in dBASEII and then communicate the structure to the appropriate BASIC and PASCAL programs through use of the STRUCTUR program. This enables the user to supply values for the attributes of the entities only one time and at the same time as the digitization. This is certainly preferable to separate and/or duplicate data entry processes.

DIGITIZE is only a data entry program. Problems were encountered with the PASCAL language in controlling the digitizing pad, but if these are solved, then it could be translated into PASCAL for consistency's sake. However, since no calculations are made in DIGITIZE this would not speed the process up. The human factor of accurately digitizing is the limiting speed factor in this process. A better degree of user friendliness should be put into DIGITIZE in order to reduce the possibility of mistakes in entering

the data. For example, when the attribute values are supplied (such as class, compartment number, importance, etc.), a checking and editing feature on those answers could be incorporated. If such sophistication were desired, a separate system of files could be set up to predefine acceptable attribute values for each kind of theme. Another user friendly technique that could be incorporated would be keeping track of file names for line and polygon boundary files and automatically assigning those instead of asking for the names from the user.

TRANSLAT has two major areas, each of which could be studied more. In one part, the area or length, as appropriate, is calculated from the boundary points. If it were possible to find more efficient algorithms (in terms of speed since the code for the existing algorithms is only approximately a dozen lines for each), then that part would be speeded up. The other part is in the representation of the boundary points of the entities. Currently, each entity requiring its boundary to be stored uses one disk file per boundary. This is probably not the most efficient way to store the boundaries. Probably some manner of storing multiple boundaries in one disk file would be more efficient. Such a technique might not make TRANSLAT run faster, but in the long run would save disk space.

Other than more user friendliness, FILLBASE is a simple program which cannot be changed since it is comprised of only a few dBASEIII commands. It is a good technique since it relieves the user of the

tedium of remembering what are basically housekeeping commands on transferring data from ASCII format files to dBASEII data bases.

Data Retrieval and Analysis

Even more important here than in the first part of the GIS is the ability of the user to handle in a capable manner dBASEII and its data manipulation and query commands. Again, the power of dBASEII is in the user's being able to specify his queries about relations among the data without having been constrained beforehand. To perform single-theme, and multi-theme, analyses the user must be able to express his queries in dBASEII format. He must also be able to understand how to copy selected fields of the appropriate records to a temporary data file in ASCII format.

Before commitment to the final software design, more study should be given to the techniques of converting polygonally formatted data to raster or grid cell format. The techniques chosen in the pilot study were chosen because of their simplicity and ease of implementation. Since such conversions tend to be computationally intensive, study should be made on how to speed the techniques up. It is also possible that if the IBM's core memory is expanded to its maximum of 1 MB, use could be made of utility programs which fool the operating system into thinking that extra core is a "disk." Such electronic disks are faster than physical disks and should make the use of temporary data files to be fast.

A fundamental limitation to this developed technique of letting each record in the OVERLAY data base represent a grid cell is the

maximum number of records dBASEII allows, 64K. This means the largest square grid available is only about 250 cells on a side. Therefore, though not possible in this project, consideration should be given to using a more powerful data base system. Naturally, using more grid cells would also mean being sure that the fastest algorithms possible were being used in the raster conversion process.

FILLOVER is another simple dBASEII program that relieves the user of the housekeeping of transferring ASCII format data into the dBASEII data base format. It does not need to be made more user friendly as everything is automatic in its operation.

Emphasis again needs to be placed upon the fact that to complete the multi-theme analysis, the user must be able to select the records showing the presence of the desired themes. However, if the user was able to get this far with the overlay analysis, he should be able to complete this step.

Report Generation and Display

Again, it must be noted that the user is responsible for learning how to use effectively the report generation capabilities of dBASEII. dBASEII's report facilities are already user friendly and guide one through the process of defining the elements of the report. Still, the user must also be responsible for selecting the appropriate records and fields and whatever summaries he desires to go into the report.

To perform the graphical display the user must be able, as in

the multi-theme analysis, to copy selected fields of the appropriate records to a temporary file in ASCII data format. After that, the POLYGRAF and OVERGRAF programs run smoothly after being given the geographic range to be mapped.

There is probably little that could be done to make them run faster as the plotter has inherent limitations in its plotting speed. The computer consistently can give commands to it faster than it can accomplish them. In the OVERGRAF program various techniques could be experimented with to see which ones give the best results for shading in the raster cells that are graphed, if such a shading effect is desired.

The graphing programs do need to be made more user friendly, particularly in the area of allowing the user to specify his geographic ranges and scales at which he wants his map to be produced. Because of the way in which real world coordinates are mapped to the plotter coordinates this may require some ingenuity, but it should be possible.

Summary

Techniques for accomplishing the Big South Fork GIS were developed and implemented into pilot software. The software does work and produce results. However, before commitment to the final software design, two more stages of design should be performed. First, the techniques for computing areas and lengths, storing boundary point data, and converting polygonal format data to raster cell format should be researched more to assure the developers that

the most efficient algorithms in terms of speed and storage are being used. Second, each step of the GIS should be examined closely for the degree of user friendliness to be implemented. The degree decided upon will be dependent upon the sophistication desired and the effort needed to implement it balanced against the utility and usefulness of being friendly.

LIST OF REFERENCES

LIST OF REFERENCES

- Alford, William L. "Digital Image Analysis of LANDSAT Data." Proceedings of the Twelfth International Symposium on Remote Sensing of Environment. Ann Arbor: University of Michigan, 1978, pp. 329-53.
- Allendoerfer, Carl B. and Oakley, Cletus O. Principles of Mathematics. Third Edition. New York: McGraw-Hill, 1969.
- Brooks, Kris and Pease, James R. Geographic Information Systems: A Review. Washington: USDA Extension Service, 1978.
- Cicone, Richard C. "Remote Sensing and Geographically Based Information Systems." Proceedings of the Eleventh International Symposium on Remote Sensing of Environment. Ann Arbor: University of Michigan, 1977, pp. 1127-36.
- Du Val, Patrick. Encyclopaedia Britannica, Vol. 18. Chicago: William Benton, Publisher, 1966, pp. 193-5.
- Dueker, Kenneth J. and Horton, Frank E. "Urban Change Detection Systems: Status and Prospects." Proceedings of the Seventh International Symposium on Remote Sensing of Environment. Ann Arbor: University of Michigan, 1971, pp. 1523-36.
- Duwe, Michael. Personal communication at Big South Fork area: 1983.
- Foley, J. D. and Van Dam, A. Fundamentals of Interactive Computer Graphics. Reading, Massachusetts: Addison-Wesley, 1982.
- Frank, Thomas D. "Comprehensive Geo-Data Base Control with an Electronic Coordinate Digitizer." Machine Processing of Remotely Sensed Data. West Lafayette, Indiana: Purdue University, 1981, pp. 497-504.
- French, P. and Taylor, M. "Interactive Management and Updating of Spatial Data Bases." NASA Conference Publication 2261, 1982, pp. 231-8.
- Goldberg, Michael. Geographic Information Systems at the Goddard Space Flight Center. NASA Technical Memorandum 83941: 1982.
- Greenlee, David D. Reference Notes: Spatial Analysis Concepts. EROS Data Center, 1983.
- Irish, R. R. and Myers, W. L. "An Interface for Remote Sensing Digital Image Systems and Geographic Information Systems." NASA Conference Publication 2261, 1982, pp. 203-11.

- Junkin, Bobby G. Procedure for Extraction of Disparate Data From Maps into Computerized Data Bases. NASA Reference Publication 1048: 1979.
- Marble, Duane F. and Peuquet, Donna J. "Geographic Information Systems and Remote Sensing." Manual of Remote Sensing. Second Edition. Ed. Robert N. Colwell. Falls Church, Virginia: American Society of Photogrammetry, 1983, pp. 923-58.
- Oliver, M. A. and Wiseman, N. E. "Operations on Quadtree Encoded Images." The Computer Journal. Vol. 26, No. 1, February, 1983, pp. 83-91.
- Paludan, C. T. N. Personal communication at University of Tennessee Space Institute: 1984.
- Pavrides, Michael G. "Data Base Management for Geographic Information Systems." NASA Conference Publication 2261, 1982, pp. 250-5.
- Salmen, Larry; Nez, George; Reed, Carl; Hamill, John; and Groppes, James. A General Design Schema for an Operational Geographic System: Region Six, U.S. Fish Wildlife Service. Fort Collins, Colorado: U.S. Fish Wildlife Service, 1977.
- Shelton, Ronald L. and Estes, John E. "Integration of Remote Sensing and Geographic Information Systems." Proceedings of the Thirteenth International Symposium on Remote Sensing of Environment. Ann Arbor: University of Michigan, 1979, pp. 463-83.
- Shelton, Ronald L. and Hardy, Ernest E. "Design Concepts for Land Use and Natural Resource Inventories and Information Systems." Proceedings of the Ninth International Symposium on Remote Sensing of Environment. Ann Arbor: University of Michigan, 1974, pp. 517-35.
- Short, Nicholas M. The LANDSAT Tutorial Workbook. NASA Reference Publication 1078: 1982.
- Skinner, C. W. and Gonzalez, R. W. "On the Management and Processing of Earth Resources Information." Conference on Machine Processing of Remotely Sensed Data. West Lafayette, Indiana: Purdue University, 1973, pp. 1A-1-11.
- Tomlinson, R. F.; Calkins, H.W.; and Marble, D. F. Computer Handling of Geographical Data. Paris: The Unesco Press, 1976.
- Wilson, C. L. and Thomson, F. J. "Integration and Manipulation of Remotely Sensed and Other Data in Geographic Information Systems." Pecora VII Symposium. Falls Church, Virginia: American Society of Photogrammetry, 1981, pp. 303-17.

Woodwark, J. R. "The Explicit Quad Tree as a Structure for
Computer Graphics." The Computer Journal. Vol. 25, No. 2, May,
1982, pp. 235-8.

VITA

Martin H. Davis, Jr. was born in Chattanooga, Tennessee in 1959. He attended elementary school in that city and was graduated from The McCallie School for Boys in Chattanooga in 1977. In September, 1977 he entered the Florida Institute of Technology in Melbourne. In June, 1981 he was graduated from F.I.T. with a Bachelor of Science degree in Physics and a Bachelor of Science degree in Space Science, both magna cum laude.

Until the fall of 1982 he worked at Technical Laboratories, Inc. in Chattanooga, designing and implementing custom software to handle their needs in laboratory sample data entry, tracking, reporting, and management. In September, 1982 he accepted a Graduate Research Assistantship at the University of Tennessee Space Institute and began study toward the Master's degree. This degree was awarded in August, 1984.

Mr. Davis is a member of Sigma Pi Sigma, Blue Key National Honor Fraternity, the American Institute of Aeronautics and Astronautics, and the American Association for the Advancement of Science.