A HYBRID STRUCTURE FOR THE STORAGE AND MANIPULATION OF VERY LARGE SPATIAL DATA SETS

By

Donna J. Peuquet
U.S. Geological Survey

U.S. Geological Survey Open-File Report 82-816

CONTENTS

ILLUSTRATIONS

ABSTRACT

The map data input and output problem for geographic information systems is rapidly diminishing with the increasing availability of mass digitizing, direct spatial data capture and graphics hardware based on raster technology. Although a large number of efficient raster-based algorithms exist for performing a wide variety of common tasks on these data, there are a number of procedures which are more efficiently performed in vector mode or for which raster mode equivalents of current vector-based techniques have not yet been developed. This paper presents a hybrid spatial data structure, named the "vaster" structure, which can utilize the advantages of both raster and vector structures while potentially eliminating, or greatly reducing, the need for raster-to-vector and vector-to-raster conversion. Other advantages of the vaster structure are also discussed.

INTRODUCTION

The current concept of digital data bases for spatial data came about as a natural extension of the concept of data bases for business applications. However, experience has demonstrated that the data structures used for business-oriented data base systems are generally unsuitable for spatial data applications. The data structures which have been developed specifically for the storage of digital map line data tend not to be very compact. This problem was eclipsed for many years by the mechanical problem of converting cartographic archives into digital form in a fast and reliable manner. Because general-purpose geographic information systems were often overwhelmed by the problems of graphic data collection, minimal effort was devoted to developing better means of storing the data after they had been digitized.

The map data capture problem is rapidly diminishing with the increasing availability and use of reliable mass digitizing systems based on flatbed or rotating-drum scanners as well as through the direct collection of spatial data in digital form (for example, Landsat data).

Some of the earlier geographic information systems have developed sizeable collections of spatial data as a result of a continuous data collection effort over time. The increasingly large data volumes encountered in these systems have, in the past, been accommodated by the even larger increases in computer capacity and speeds which have been provided by improvements in hardware technology. However, the very large spatial data volumes now being ge nerated are reaching a level of complexity and rate of growth for which hardware solutions alone are inadequate. Operational geographic information systems are encountering major performance problems, or are being deliberately restricted in data base size and range of application in order to avoid these problems. These problems occur because the existing raster and vector types of structures for cartographic line data do not provide the necessary combination of compactness, computational efficiency, or flexibility to satisfy the processing requirements of these large systems. This paper will examine these problems and offer a third type of data structure which is designed to combine the advantage of raster and vector data formats within a unified structure and thus provide greater potential efficiency and range of application.

BACKGROUND

Many government and non-government agencies are attempting to build very large, integrated spatial data bases for incorporation into geographic information systems which will, in turn, become the basic analytical tool and information source within their respective organizations. The U.S. Geological Survey envisions a cartographic data base containing all information from 57,000 map sheets covering the conterminous United States. If these sheets were scanned once at, for example, 250 pixels per map inch (which is not high precision by cartographic standards), the total data volume would be approximately 1.5 x 10**13 pixels. This would be only a small part of the entire USGS natural resource data base which will also include a wide variety of hydrologic, geologic, and natural resource information for various types of applications. In light of these data volumes, it becomes critical to develop spatial data structures which allow efficient data storage and retrieval as well as accommodation of a wide range of types of spatial data and applications.

The paper map is, in general terms, a two-dimensional model of geographic space. The area of the piece of paper represents a given area on the surface of the Earth. In addition to the data explicitly recorded as lines and other types of symbology, there is a large number of implicit spatial relationships recorded such as adjacency and relative position, which are also spatial data and which must be preserved in some way within the data structure. A critical problem underlying the transformation of map data into digital form is the preservation of these interrelationships in an efficient manner [1].

The numerous formats which have been developed for recording spatial data in digital form can be classified into the two basic types, vector and raster. As shown in figure 1, the basic logical unit of data in vector format is a single vector, or map line. In this sense, vector-type data structures are one-dimensional, consisting of strings of coordinates recorded along each individual line on the original map. Vector structures thus lend themselves to straightforward implementation on standard sequential, list-oriented computers. Spatial relationships, however, must either be explicitly recorded, thereby increasing the volume of stored data, or be computed each time they are required (for example, areal adjacencies).

The grid or matrix structure is an uncompacted raster structure. The basic logical unit is a single grid cell, or pixel, of the matrix. Alternatively, it may be viewed as an x-y grid vertex. In either scheme, each grid cell represents a location in two-dimensional space and the structure can be traversed in either the x or the y direction with equal ease. To eliminate the storage of locations with null or redundant data values, raster structures are usually compressed in one direction. This effectively produces a data structure in which the map area is divided into very thin strips, or scan lines. Along each scan line, only the x-locations with non-null values (that is, scan-line intersection with a map line) are recorded. This makes traversal of the data structure in the y-direction more difficult than in the x-direction because the exact position of a given x-value within a scan line record will vary from one scan line to the next. The spatial ordering of the data, however, is not altered. A universal characteristic of raster formats is that the order of the data in the digital file is determined by its location in space. Unlike vector structures, spatial relationships are implicit in the raster data structure itself. Raster-type structures subsequently lend themselves to a more natural and less complex representation of two-dimensional data. Another significant advantage to raster formats is that current hardware technology for graphic data input and output, particularly for large data volumes, is heavily raster-oriented. This means that map data in raster format are usually a "given" in present cartographic data base systems.

In contrast to their data storage problems, vector structures do have significant processing advantages since they lend themselves to a greater variety of analytical techniques. Most known algorithms for specific cartographic processes were developed for manual application and are vector oriented [1]. However, there are a number of procedures, such as coordinate transformation and contouring, which are more efficient in raster mode because they are parallel in nature or because they rely heavily on some type of spatial relationship information.

The two types of spatial data structures thus offer significant storage versus processing tradeoffs for map data. Neither is currently adequate by itself for efficient storage and manipulation of cartographic data in a large-volume geographic in formation systems.

ALTERNATIVE APPROACHES

One approach to the storage and processing tradeoffs between raster and vector data structures is to store the spatial data in raster format, perhaps with only minor modification from its raw scanner output form, and then to convert it to vector format when advantageous for performing a given analytic or manipulative process. This conversion approach is the most commonly adopted because it is conceptually so straightforward. What is soon discovered, however, is that raster-to-vector conversion, as well as the reverse process, can quickly become a bottleneck within a system because the volume of data and frequency of use increases [2,3]. Raster-to-vector conversion requires some type of intricate line-following procedure, because cartographic lines are characteristically both convoluted and topologically complex. These conversion procedures represent significant system overhead which must be avoided or minimized.

Another approach is to develop new raster-oriented algorithms for processes which currently have only vector-oriented solutions. Theoretically, this could ultimately eliminate the need for vector data structures and result in the development of exclusively raster-oriented geographic information systems. This would be particularly desirable in applications where raster-formatted areal data, such as Landsat imagery, is used in conjunction with map line data. A wide variety of efficient raster-oriented data analysis and manipulation procedures have already been developed, primarily within the field of image processing [1]. However, some spatial analytical processes seem to be intrinsically vector-oriented and cannot be restated in a parallel or raster-oriented manner [4]. These include some commonly performed analytical procedures, such as network shortest-path and optimal-routing problems. Other processes may prove to be so much more efficient when performed in vector mode that the additional time overhead for raster-to-vector conversion would be cancelled out. "Forced" vectorization of raster data could thus be greatly reduced but not totally eliminated in a raster-oriented geographic information system if any intrinsically vector-oriented procedures are required by the user.

A possible solution to this dilemma is the development of alternate types of spatial data structures for line data. The hybrid spatial data structure which is presented below utilizes the advantages of both raster and vector data structures while virtually eliminating large scale raster-to-vector and vector-to-raster conversions by incorporating characteristics of both structures. The term coined here for this hybrid structure is the "vaster data structure.

THE VASTER DATA STRUCTURE

Definition

The vector-to-raster conversion process is in essence a sorting operation [5]. A totally sorted file of spatial data arranged in both x and y order is, by definition, a raster-formatted file. It seems reasonable to assume that spatial data simply stored "frozen" part way through the vector-to-raster conversion process would require only part of the computational effort to be converted into either a completely raster or a completely vector format. To take this concept one step farther, it is possible to derive such a part-raster, part-vector structure which could be used as an intermediate format without the need for conversion to either of the two basic structures.

The convention used is to define a single scan line as the ordered set of all x values of recorded spatial entities occurring with the same y value. In an uncompacted raster (for example, grid) format there is one data value recorded for each x value within any given scan line. The basic logical unit of the raster structure is the scan line, whereas a map line segment is the basic logical unit of the vector structure. A map line segment is customarily defined to be a single uninterrupted map line. A line interruption is defined as the occurrence of the end of a map line, or the intersection with another line or the map boundary.

In contrast, the basic logical unit of the vaster structure is the swath. Each swath spans a constant, known range over y and would correspond to a group of contiguous scan lines if the data were organized in raster format. For the purpose of this discussion, a convention of swaths arranged in order of ascending y will be used. Each swath contains a raster component and a vector component, as depicted in figures 2a and 2b. Both components are recorded at the same grid resolution. The leading edge of each swath (minimum y value) is recorded in raster format as a single scan line and functions as the index record for the swath, containing an identifier and x-coordinate for each map-line intercept. The encoding used is similar to the structure developed by Merrill [6]. The raster encoding scheme used in the vaster structure, however, contains all map line intersections within the same record. This is done to allow for efficient linkages between the raster and vector portions of the swath and to allow types of line structures other than nested polygons. The data contained in the remainder of the swath are recorded in vector format. All vectors contained in each swath are arranged in scan line intercept sequence in order of ascending x; polygons internal to a swath are listed separately. Each line intercept noted in the index-record functions as the endpoint of each vector line segment within the swath, in sequence. Note that there is no scan line record at the end of the swath, since the next scan line record is functionally the leading edge of the next swath.

Figure 3 shows the digital form of the vaster-formatted data of figure 2b. Both the raster and vector portions of the swath are coded in forms which allow a substantial degree of data compaction without loss of precision. The raster record in each swath contains an ordered sequence of line identifier - x coordinate pairs. The line identifier is a unique but arbitrarily assigned primary key for each line segment. Each coordinate is the x-distance from the left edge of the scanned area measured in grid resolution units. The actual (that is, map) x-y coordinates for the edges of the digitized map are recorded separatelv Thus in figure 3, the index record for swath 3 starts with line "A" intersecting the raster at local x location 2, line "B" at location 4, etc. A straight line segment coincident with the scan line is noted by a length factor in addition to the x coordinate (for example, D6+2). A local inflection point coincident with the scan line is recorded as a line of zero length. This length factor is coded as a "+" followed by the x distance. A map-line junction falling on a scan line is noted by a "-" before the x value. These are the only two special line configuration notations occurring within an index record.

This vaster notation avoids the storage of null values, and the use of a local x-coordinate notation is an effective method for minimizing the number of bits required to store each coordinate. Both of these compaction mechanisms are particularly advantageous for map line data, since cartographic lines are normally sparse relative to the pixel resolution, and absolute map coordinates are not required for most analytical applications. Incremental coordinate notation (that is, delta-x) is not used except in the case of scan line intersection of a map line which must stand alone for use as end coordinates of the attached vectors.

Each map line in the remainder of the swath, as shown in figure 3, is recorded in a vector chain code format which has been modified to fit the special requirements of the hybrid vaster structure. These modifications will be explained below. Vector chain coding has been discussed at length by Freeman [7,8,9]. and on a comparative basis for cartographic applications by Boyle and Robertson [10].

Each continuous map vector within the swath is coded as a single chain. Chains are stored in ordered sequence so that there is a direct correspondence between a line segment's position in the scan line record and the location of its vector chain code representation within the remainder of the swath. There are only two situations in which a one-to-one correspondence does not hold. The first is the occurrence of a line segment which is totally contained within a single swath and does not intersect any swath boundary. The second is the occurrence of a line segment, local maximum or minimum, or line junction falling upon a scan line. Selected graphic examples of line configurations in this second cateogry are included in figure 4.

As shown in figure 3, the chain codes for line segments totally contained within the vector portion of a single swath are placed at the end of the swath record in order of occurrence, which is determined by the minimum y-coordinate of each such line segment. Internal line segments and the other special line configurations included in figure 3 are digitally encoded through the use of special chain codes which are listed in table 1. The structure of special codes is modeled after that given by Freeman [8]. Each of these codes begins with "04" as originally given by Freeman. This chain code sequence indicates that the line retraces itself, which is a meaningless occurrence for normally encountered map lines. It is therefore used as a flag to indicate that a special code sequence follows. The special code numbers assigned to specific line configurations, however, have been modified for the vaster structure. Additional special line configuration codes may be added as needed for specific applications. A line segment or local maximum coincident with the scan line record with both legs of the chain oriented upward (that is, a chain in the preceding swath tangent to the scan line of the current swath) from each end (see fig. 4) is indicated by the occurence of a swath chain consisting of only an "04" code with no additional chain codes, and corresponding to a line segment within the scan line record. The special code in this case thus simply acts as a place-keeper to maintain the one-to-one correspondence with map lines contained in the scan-line record. Note that, as figure 3 shows, a chain link connecting a chain to the scan line record in the next swath is not recorded, because this would be redundant.

Whenever map (that is, global) coordinates are required, they can be derived by first calculating the global x values of each data element along the scan line record within the swath by a simple addition of the global x-coordinate value recorded for the edge of the area to each local x value. These then become the global x values of the ends of the short chain-coded vector segments within the swath.

Given cartographic resolution standards, the length of a chain increment is generally small compared to the convolution of the cartographic line [10]. This results in a predominance of straight and gently curved segments relative to the unit increment length of the chain. The use of an abbreviated code which indicates distance in the last recorded direction instead of explicitly recording a repeated chain code, allows a high degree of compaction of the original map-line vector without losing any of the original map line accuracy [8]. This chain repeat-code is the only special code given in table 1 that is not needed to preserve the topological integrity of the original line data. This run-length chain code offers a space advantage for straight line segments exceeding five resolution units in length, a frequent occurrence in the representation of cartographic data.

The Generation of Vaster Data from Raster Data

The vaster data structure is derived from skeletonized scanner output through a raster sequential process consisting of three phases:

1) encoding the scan line index record
2) vector chain generation
3) resolution of vector connectivity

The first two phases generate the swaths and together require only one pass through the input file. The third phase is performed on the vasterized file to reconnect short-chain vector segments generated as a byproduct of phase 2 into continuous lines.

For each new swath, the next scan line remaining in the input data file is converted into the index record. All background (non-line) pixels are ignored. The local x-coordinate of each occupied cell is recorded along with a line identifier tag. For the first swath in the map, each line encountered is assumed to be the end point of a new vector chain and is assigned a new, unique and arbitrary tag. These identifiers are also inserted into a stack of active map lines which are currently being followed. This stack is maintained throughout the file generation process for the entire map. Identifiers are added and deleted, as appropriate, during both raster and vector-chain generation portions of the process so that the correct identifier can be assigned to each increment of each map line as they are followed through successive scan lines of the input file.

The vector-chain generation and connectivity resolution phases employ an algorithm described in detail by Chakravarty [11] for chain generation and junction detection. The basic process is to convolve a 3 x 3 window with the digitized area in raster fashion. Each newly coded chain link is checked for connectivity and determined to be one of four types;

1) continuation of an existing vector chain
2) the endpoint of a vector chain
3) the beginning of a new vector chain, or
4) a junction

The appropriate actions are then taken to update the active chain stack and the vector chain lists.

If the new chain link is determined to be a continuation of an existing vector chain, the new link is added to the appropriate chain within the current swath chain list and the pointer indicating the current identifier in the active chain list is updated. If the new chain link is determined to be the endpoint of an existing vector chain, the "end of line" code is added to the appropriate chain and the chain identifier is removed from the active chain list. If the new link is not connected to an existing vector, the link is appended to the end of the chain list as the start of a new vector chain, and a new chain identifier is inserted into the active chain stack. If a junction is detected, the "junction" code is added to the chain, the appropriate number of new (empty) vector chain records are inserted within the current swath chain list as the next records in sequence, and an equal number of identifiers are inserted in the active vector stack in place of the current vector identifier. If a line closure (that is, two lines being followed are found to be parts of the same line) is detected (always occurring at a local minimum), both vector identifiers are inserted into a second temporary list, consisting of of connected identifiers along with pointers to their respective chain records. At the end of this phase, the input data have been reformatted into vaster swaths, but the original map vectors have been artificially segmented and coded only in the forward scan direction due to the parallel nature of the vector-chain generation algorithm. The vector-chain codes at this point are therefore similar to the raster-chain codes described by Cederborg [12].

The connectivity resolution phase is most efficiently performed after the entire map has been coded into swaths. The resultant swath file is processed to assemble vector segments into continuous chain-coded, vector records by utilizing the list of connected identifiers. This involves correcting the directionality of some chain coded segments and replacing all line identifier "aliases" with the same identifier, as well as correcting some chain-code notation to reflect special map-line configurations.

At the end of the conversion process, each map line segment is recorded with a unique identifier, which can subsequently be replaced by a more meaningful identifier. The line segments are assembled to define additional higher levels of spatial entities, such as county boundaries, stream networks, or whatever the nature of the data and the application requires.

Figure 5 shows how vaster-formatted data might be organized as a stored digital file. The raster and vector portions of the data base are physically separate. This separation is dictated by application considerations (see below). A separate list of pointers to the beginning of each swath, in sequence, is maintained to provide fast random access to any individual swath, given its beginning y value. This list points to the location of the beginning of the swath index scan-line record. The raster and vector chain records are stored in the same sequence within their respective files, with a pointer at the beginning of each index scan-line record to the location of the first chain within the swath in the vector chain list.

Utilization of the Vaster Structure

Given the presence of raster-formatted data within the vaster data structure and with the remainder of the data locationally keyed to these raster-formatted positions, the spatial ordering and preservation of spatial relationships inherent in the raster structure is retained to a significant degree. In addition, the presence of both raster- and vector-formatted data in the vaster structure offers retrieval and processing advantages for cartographic data base applications. Functionally, the vaster format contains a two-level hierarchy. The raster portion of swaths may be used alone, without need to reference the more detailed vector file for many common processes or portions of processes, thereby speeding up performance considerably. A significant advantage of the use of this hybrid data structure for a very large data base is the ability to make use of only the raster file as a set of generalized data for quick browsing or sampling of the data base. This separation of use requires that the raster portion of the data base be stored as a separate file in order to maximize speed in accessing the raster data as well as for more efficient data retrieval based on spatial criteria.

Many queries could utilize only the coarser, raster structured portion of the data base where approximate solutions are desired, as may be the case in many centroid, area, perimeter, or arc-length calculations. Most spatial relation queries such as point inclusion and relative position could also be answered from raster-formatted data, using raster oriented algorithms which are generally more efficient for this type of task because they ignore the bulk of the data. These algorithms would operate in the conventional manner, as described by Merrill [6]. The margin of error between approximate measures derived from the raster data alone is a function of both the width of the swath (that is, the sampling frequency) and the sinuosity of the map lines themselves If this margin of error is unacceptable in the case of spatial relation queries, only a small portion of the vector-formatted data need be referenced. For example, to determine containment of a point within an enclosed polygon where the y-coordinate of the given point falls between two scan-line records (that is, within a swath), then only the chain code boundaries of the particular polygon in question within the pertinent swath is converted to raster format so that the Jacobsen Plumb-line algorithm can be performed using the same y value as the given point [13].

There are other processes for which raster data organization will increase the efficiency of vector algorithms given the preservation of spatial ordering in the vaster format. The two foremost examples are mosaicking and overlay of vaster-coded data when these procedures are desired at full resolution. The approach to both of these processes is to first perform mosaicking or overlay on the raster-formatted portion of the data in each swath. This serves to "align" the remaining chain-coded short line segments for easier vector-mode processing.

The vaster structure may also be utilized as a full-detail vector chain coded file. Any algorithms applicable to vector chain codes can be utilized in the normal manner after reconnecting the short chain segments of individual swaths by reference to the index (scan-line) records. Speed is thus maximized when a vaster-formatted spatial data base is used for orientation and approximate information over large areas and more detailed analyses, based on chain-coded vectors, are reserved for relatively small areas.

EXTENSIONS

Ancillary data

In many applications when dealing with a cartographic data base it is necessary to store data elements describing various attributes of the quantized spatial entities which are not derivable directly from the coordinate data. For example, if the quantized map lines represent a hydrologic network, it may be necessary to record data on stream depth, rate of flow, and amount of suspended sediments for each link in the stream network. These descriptor data are best stored physically separate from the quantized map lines utilizing a non-spatial structure better suited to utilization of standard data base management system (DBMS) techniques. The common link between the map, or geographic definition, portion of the data base and the attribute portion is the unique geographic entity names contained in both portions. These are used as a primary key to directly retrieve information from the attribute files. Subsequent retrieval of the map data for a named geographic entity in the data base may be achieved through the use of a locational key inserted into the attribute portion of the data base (for example, polygon centroid coordinates or minimum and maximum y-coordinates.) This simple locational index is then used as a primary key for the map data, similar to location of a street of a given name on a road map by use of the alphabetized street name table to look up the general locational code of the map grid cell containing the desired street.

The physical separation of spatial and ancillary, non-spatial, data provides important advantages for efficiency and logical simplicity. This may be carried one step farther through the creation of a separate table or "object dictionary" containing the locational indices for geographic entities, as shown in figure 6. The object dictionary is composed of a pointer to the spatial files, a pointer to the attribute file for each entity, as well as the entity name, thus providing a cross-reference table for quickly retrieving either spatial or attribute data. This may be particularly useful for retrieval of spatial data based upon entity name. In figure 6, two pointers for the spatial file (minimum and maximum y values) for each data entity are shown. This would facilitate retrieval of geographic areas based on one or more named geographic entities.

Different Types of Spatial Entities

The vaster data structure as described is geared toward recording map data represented primarily in the form of separate line segments with points recorded as lines of zero length. The vaster data structure is also suitable for closed areas.

A special problem arises, however, for adjacent polygon (polygon web) structures. Each line segment in this case is a boundary segment between two areas. However, it is also a characteristic of map data that these boundary segments frequently have additional attributes. For example, county boundaries may be made up of a combination of geographic line entities such as roads, streams, or shorelines. In this case, an additional entry in the object dictionary for "entity type" is needed. Any item in the item dictionary could belong to more than one "entity type".

CONCLUSIONS AND FUTURE WORK

The vaster data format would seem to provide a "best-of-both-worlds" solution to the efficient storage and handling of spatial data. Large volumes of map data can be digitized via scanners, converted only part-way to vector format, and then be utilized by a wide variety of both raster-oriented and vector-oriented algorithms in the conventional manner.

The problem of determining the optimum data storage resolution for vaster-formatted data has two related aspects. The first is a complex data sampling problem: How narrow must the swaths be to provide a satisfactory interval between raster-formatted records for the first-level data resolution? Each significant map line should be intersected at least once by an index scan line, thus avoiding occurrences of map entities which are completely contained within the vector portion of a single swath. Closely spaced index scan lines (that is, narrow swaths relative to map-line density and sinuosity), while avoiding such occurrences, cause multiple scan line intersections of the same continuous map line, which can inflate data volumes. It may be that any line less than a given length could be declared not significant and thereby ignored. However, the usual circumstance is that such a threshold cannot be determined, or that all lines must be assumed to be significant. Another alternative is to utilize variable-width swaths, dependent on the local density of data to be captured. This is related to many linear-traverse spatial-sampling problems and extensions of these techniques may provide viable solutions [14,15].

A second aspect of the resolution problem arises from the hybrid nature of the vaster structure: What is the raster-vector data volume ratio which would provide optimum overall performance for a given group of raster-and vector-oriented algorithms? It is the joint use of rasters and vectors which takes this task a level beyond that of a standard system optimization problem. The two aspects of this vaster resolution problem are highly interrelated, and few data are available upon which to evaluate the tradeoffs.

Therefore, The next logical step in investigating the vaster approach is to analyze the storage and processing characteristics of the vaster structure via empirical simulation. Observed comparative behavior, in turn, should provide a basis for theoretical generalization. Timing tests using the vaster data structure in conjunction with commonly used manipulative procedures need to be performed and then compared with the performance of equivalent operations utilizing standard raster or vector formats. This currently appears to be the only sure means of determining the relative operational characteristics of the three types of spatial data structures. This research is currently under way.

The vaster data structure does not represent the ultimate answer to all spatial data storage and processing problems. It does, however, show potential for simultaneously minimizing the disadvantages of the raster and vector-type data structures while capitalizing on the advantages of both.

REFERENCES

1. Peuquet, D., 1979, Raster processing - an alternative approach to automated cartographic data handling, American Cartographer, vol. 6, 129-139.

2. Peuquet, D., 1981, An examination of techniques for reformatting digital cartographic data, Part I: the raster-to-vector process, Cartographica, vol. 18, 34-38.

3. Peuquet, D., 1981, An examination of techniques for reformatting digital cartographic data, Part II: the raster-to-vector process, Cartographica, vol. 18, 21-33.

4. Lee, T., 1980, Two dimensional voronoi diagrams in the lp-metric, Journal of the Associaton for Computing Machinery, vol. 27, 604-618.

5. Franklin, R., 1979, Evaluation of algorithms to display vector plots on raster devices, Computer Graphics and Image Processing, vol. 11, 377-397.

6. Merrill, R., 1973, Representaton of contours and regions for efficient computer search, Communications of the Association for Computing Machinery, vol. 16, 69-82.

7. Freeman, H., 1979, Analysis and manipulation of line drawing data, Proceedings, NATO Advanced Study Institute on Map Data Processing.

8. Freeman, H., 1974, Computer processing of line-drawing images, Computer Surveys, vol. 6, 57-97

9. Freeman, H., 1961, On the encoding of arbitrary geometric configurations, IRE Transactions, EC-10, 260-268.

10. Boyle, H. R., and Robertson, P. K., 1978, Optimization of vector line format structures for digitized cartographic lines, unpublished report, Department of Electrical Engineering, University of Saskatchewan.

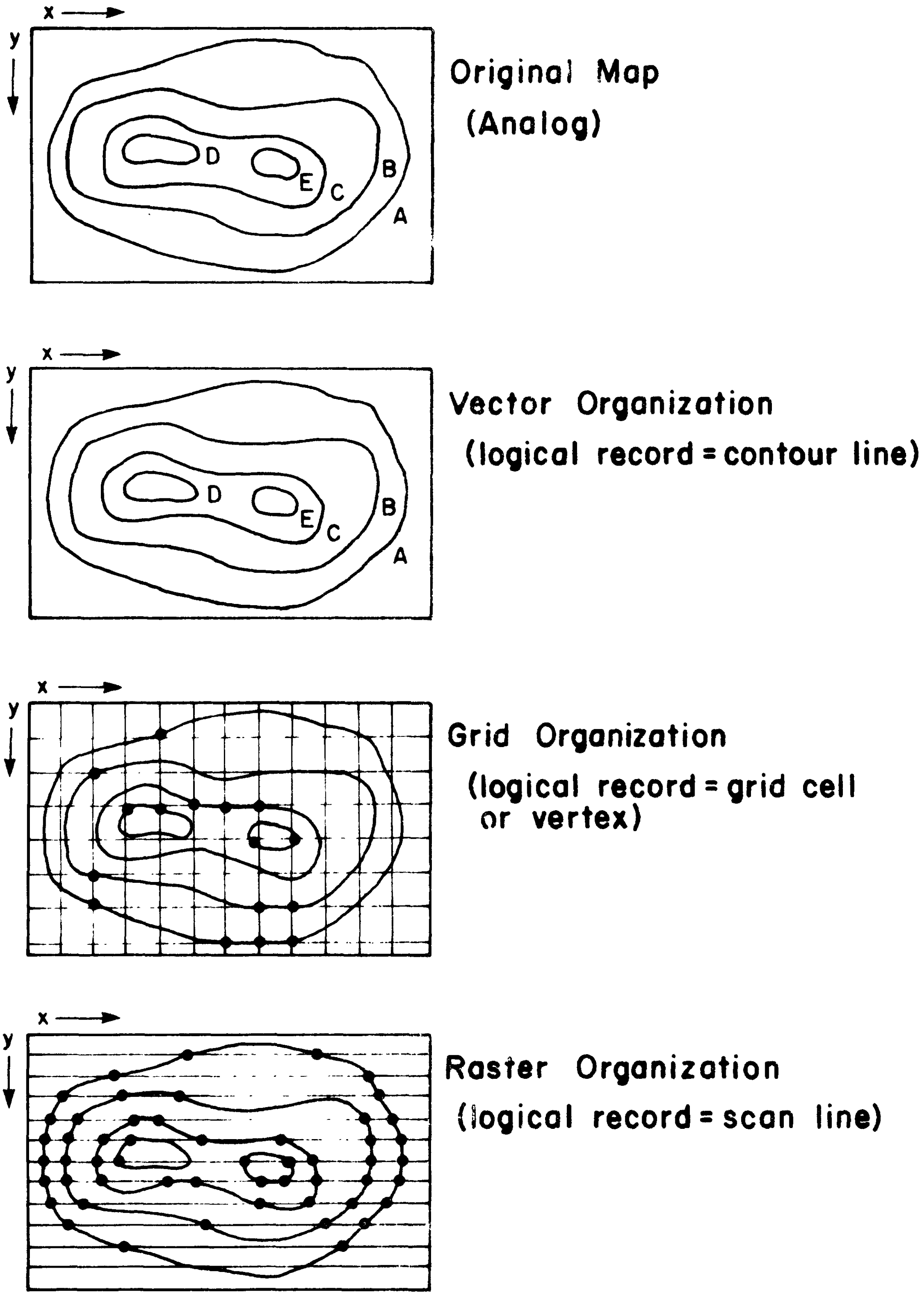
11. Chakravarty, I., 1981, A single-pass, chain generating algorithm for regin boundaries, Computer Graphics and Image Processing, vol. 11, 377-397.

12. Cederberg, R., 1979, Chain-link coding and segmentation of raster scan device, Computer Graphics and Image Processing, vol. 10, 224-234.

13. Jacobsen, J., 1968, Geometric relationships for retrieval of geographic informaton, IBM Systems Journal, vol. 7 331-341.

14. Matern, B., 1960, Spatial variation, Medd. Statens Bkogsforskningsinstitut, vol. 49, 144.

15. Robinson, J., 1975, Frequency analysis, sampling and errors in spatial data, in J. C. Davis & M. J. McCullough, Display and Analysis of Spatial Data.



Original Map
(Analog)

Vector Organization
(logical record = contour line)

Grid Organization
(logical record = grid cell or vertex)

Raster Organization
(logical record = scan line)

Figure I. Types of Data Organization

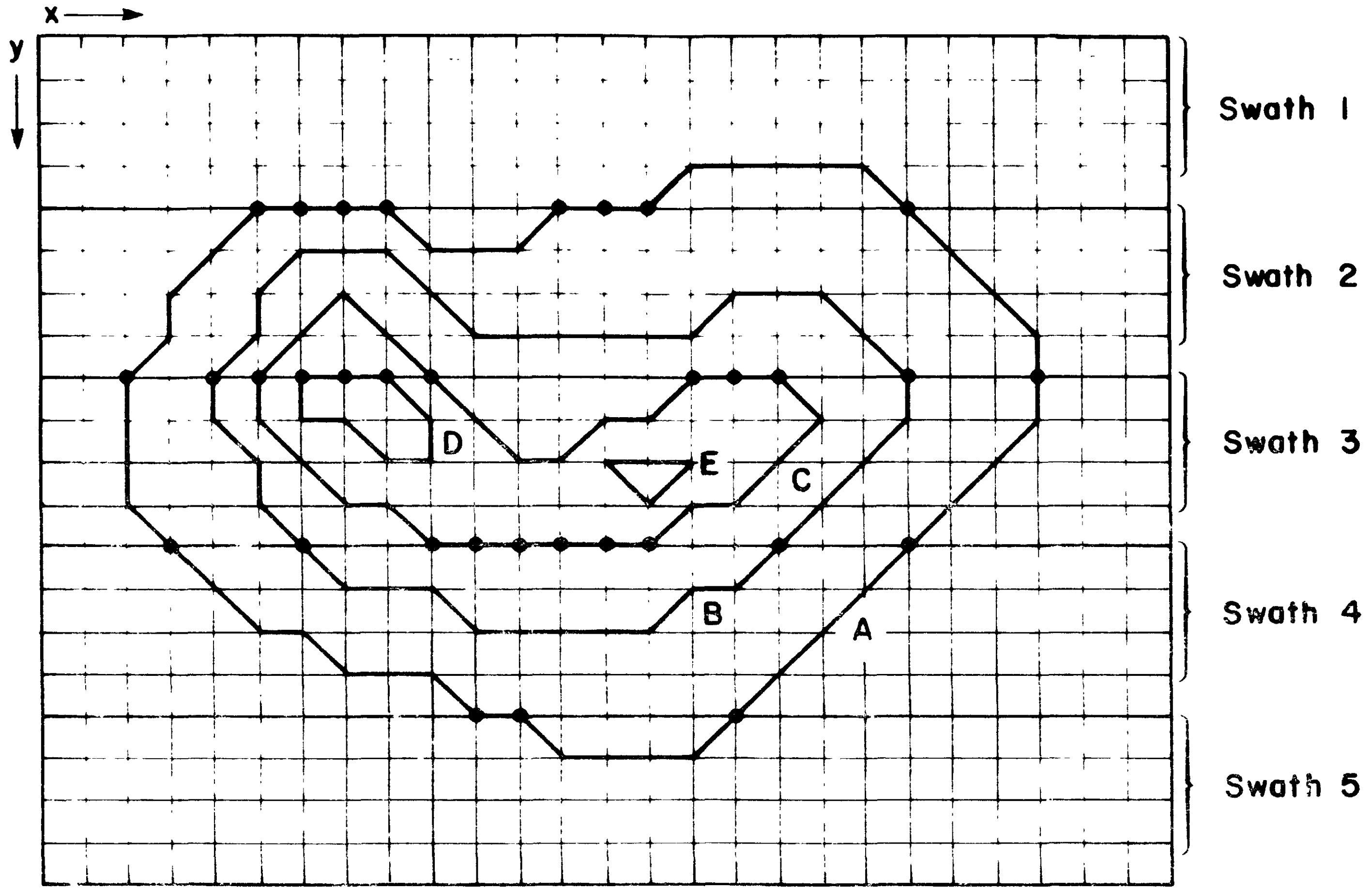
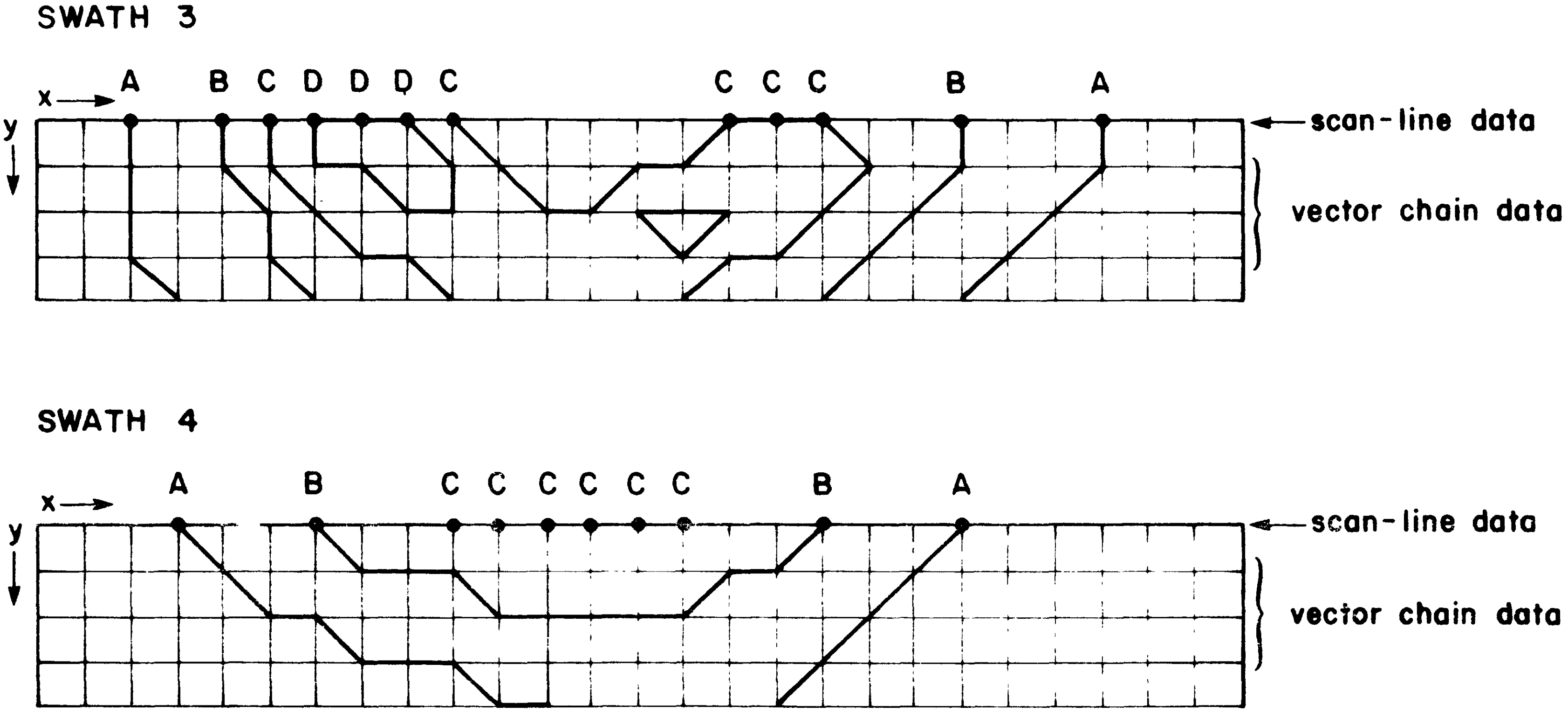
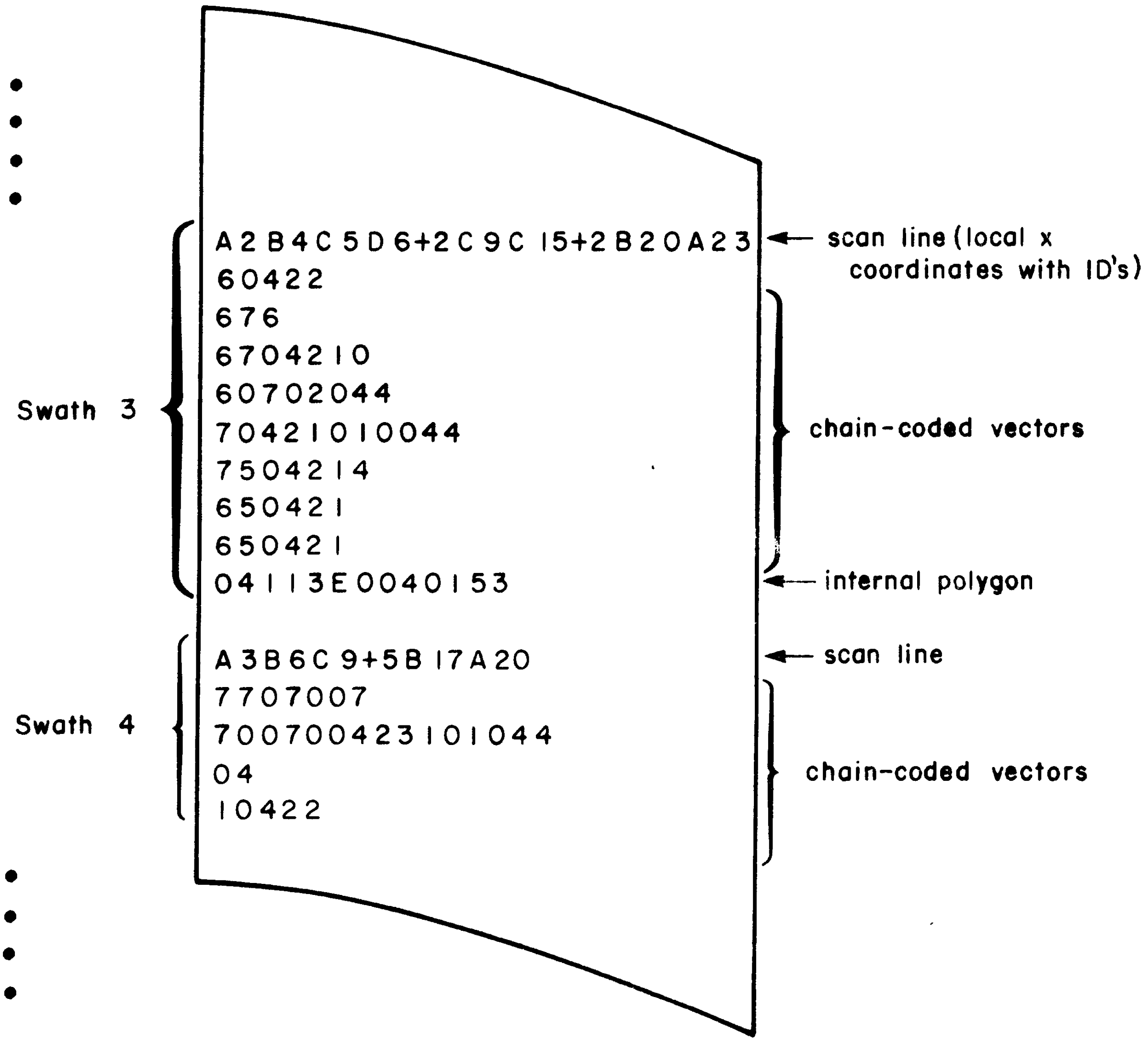


Figure 2a. Vaster Organization (logical record = swath)


SWATH 3

SWATH 4

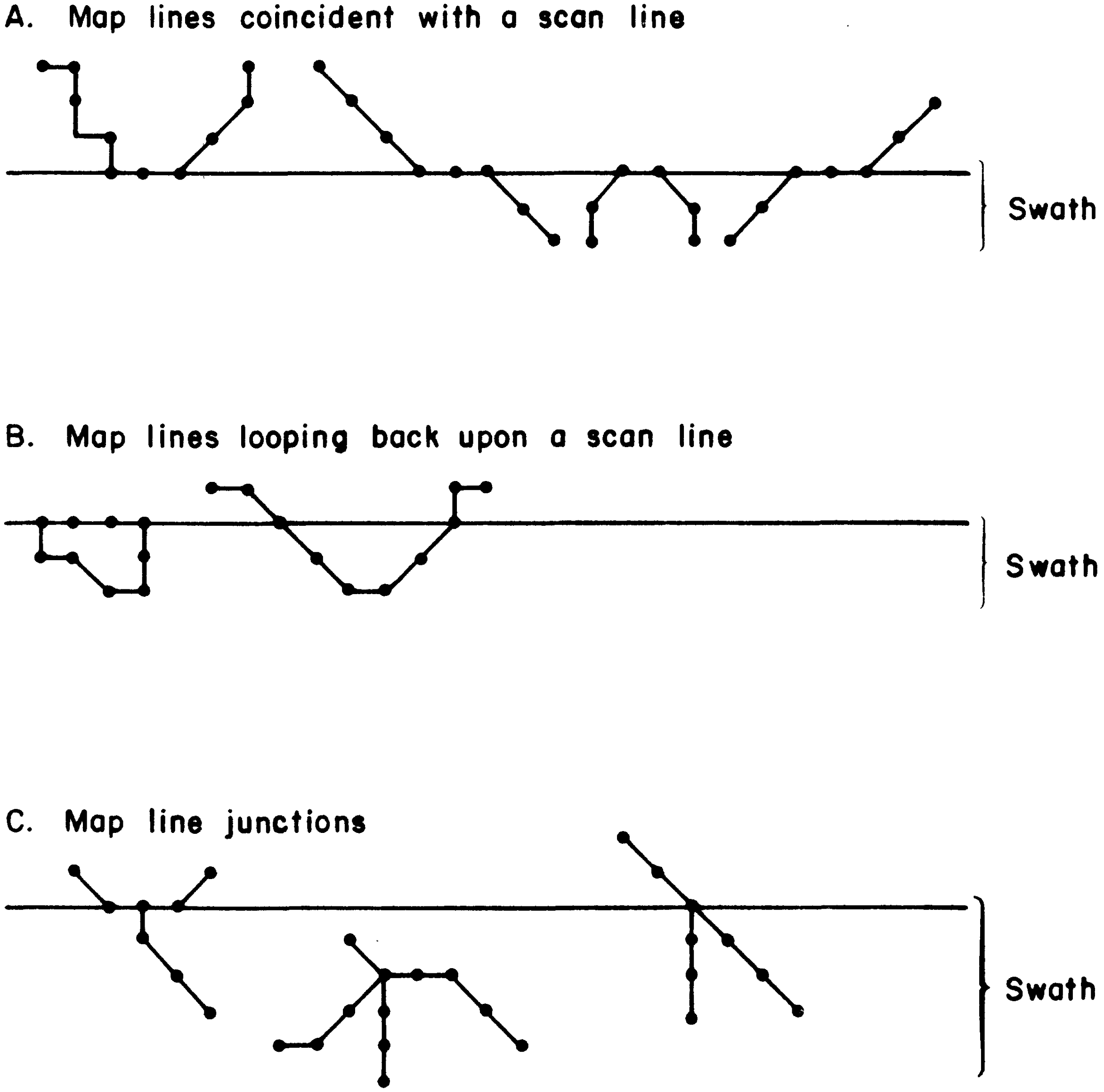
Figure 2b. Individual Swaths



```
A2B4C5D6+2C9C15+2B20A23
60422
676
6704210
60702044
70421010044
7504214
650421
650421
04113E0040153

A3B6C9+5B17A20
7707007
700700423101044
04
10422
```

Figure 3. Vaster-formatted data of figure 2b in digital form



A. Map lines coincident with a scan line

B. Map lines looping back upon a scan line

C. Map line junctions

Figure 4. Some Special Map Line Configurations

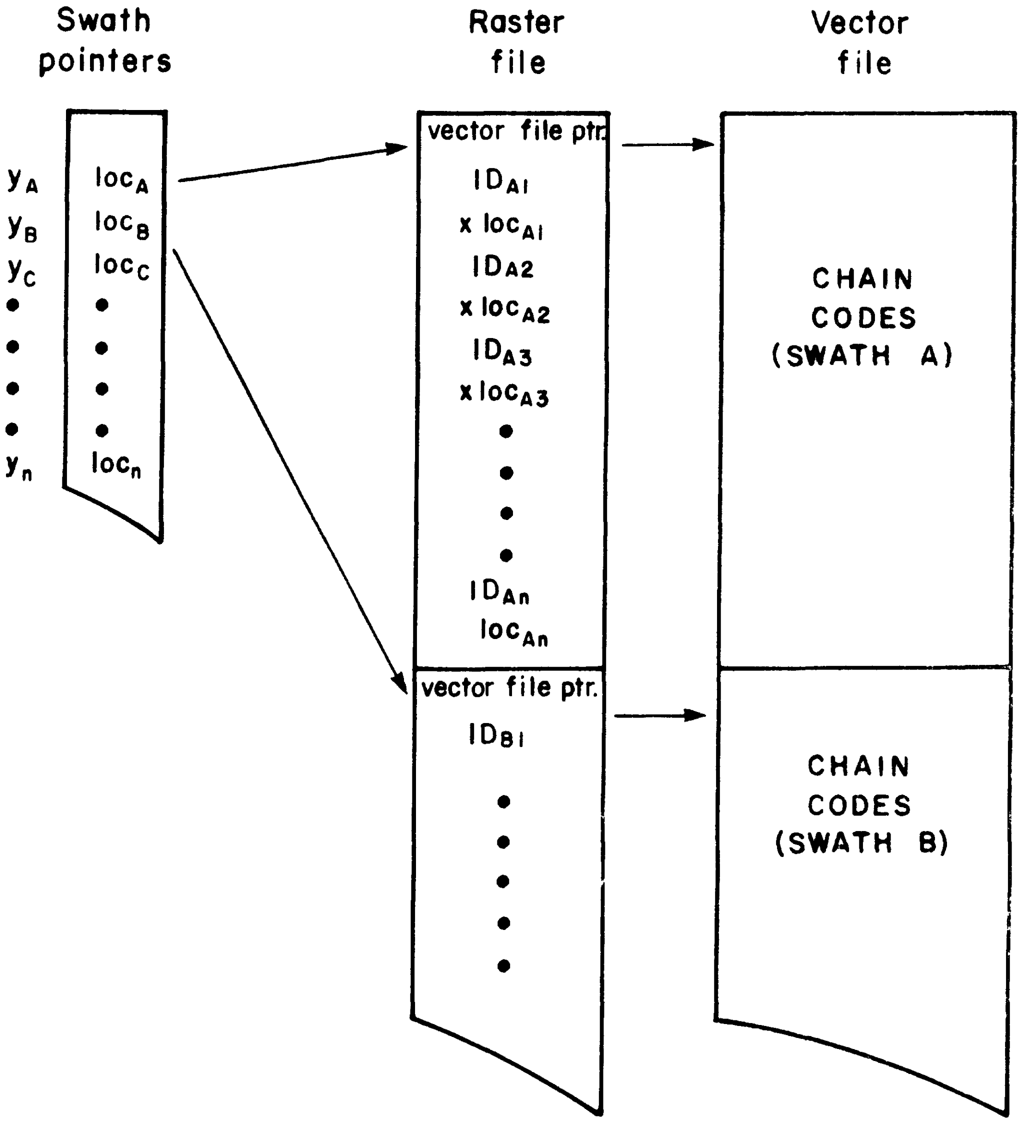


Figure 5. File Structure for Vaster-Formatted Data

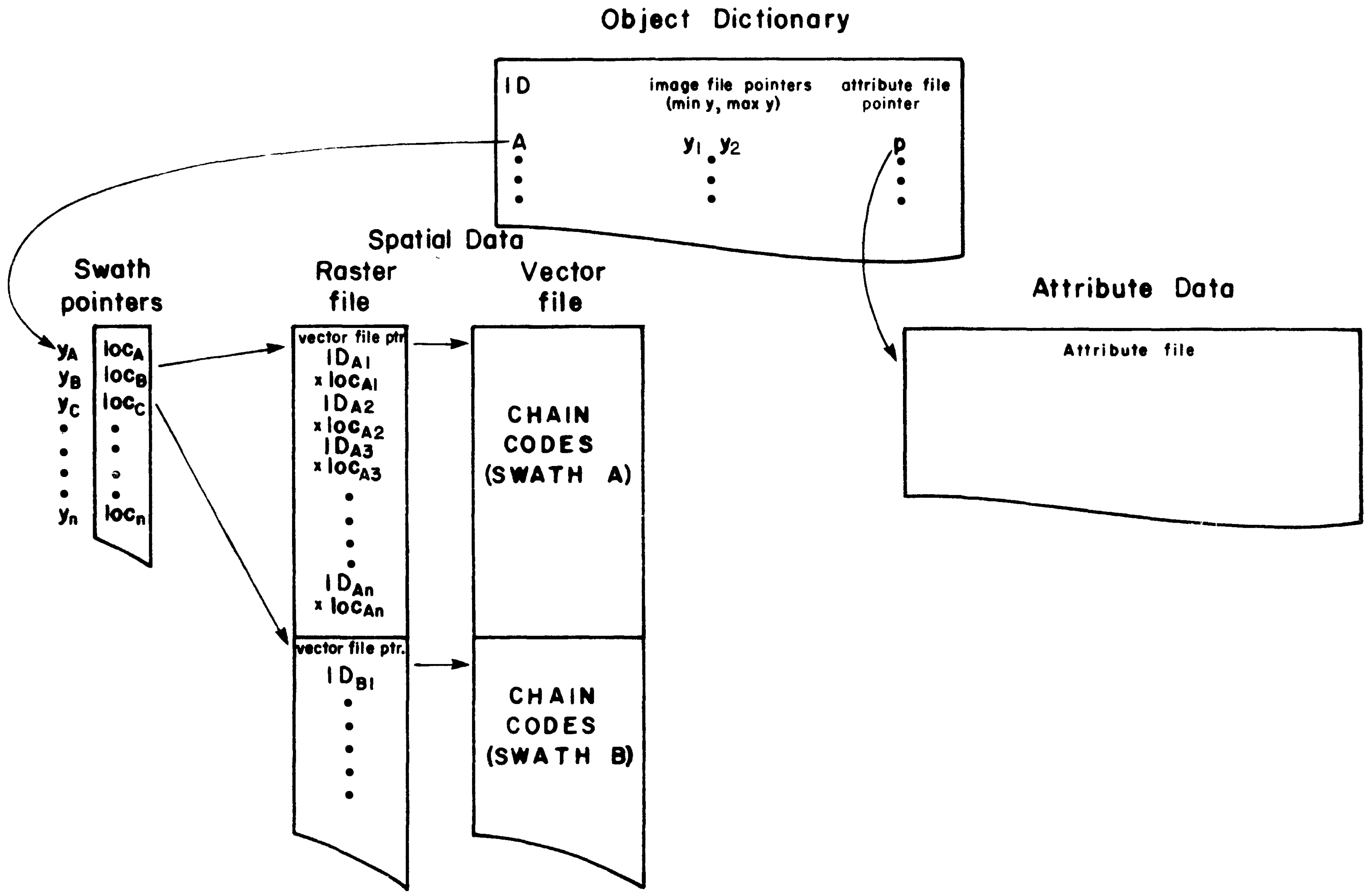


Figure 6. Database Schema Employing Vaster-Formatted Data