BUREAU OF THE CENSUS
STATISTICAL RESEARCH DIVISION REPORT SERIES
SRD Research Report Number: Census/SRD/RR-86-12

STORING, RETRIEVING AND MAINTAINING INFORMATION ON GEOGRAPHIC STRUCTURES: A GEOGRAPHIC TABULATION UNIT BASE (GTUB) APPROACH

by

David Meixler
Geography Division
Alan Saalfeld
Statistical Research Division
Bureau of the Census

This series contains research reports, written by or in cooperation with staff members of the Statistical Research Division, whose content may be of interest to the general statistical research community. The views reflected in these reports are not necessarily those of the Census Bureau nor do they necessarily represent Census Bureau statistical policy or practice. Inquiries may be addressed to the author(s) or the SRD Report Series Coordinator, Statistical Research Division, Bureau of the Census, Washington, D.C. 20233.

Recommended by: Paul Biemer

Report completed: April 21, 1986

Report issued: April 21, 1986

STORING, RETRIEVING AND MAINTAINING INFORMATION ON GEOGRAPHIC STRUCTURES: A GEOGRAPHIC TABULATION UNIT BASE (GTUB) APPROACH

David Meixler and Alan Saalfeld
Bureau of the Census
Washington, DC 20233

ABSTRACT

The various levels and subdivisions of geography and their interrelations may be organized for computer storage, retrieval, and update through the use of multilist structures. In this process several files are created to correspond to the different partitions of the whole space or universe. A directory is used to keep track of entities for each entity type—state, county, congressional district, etc. The "state" directory contains 57 entries, for example, one entry for each state (50) or territory (7).

A fundamental file of unnamed, elementary entities is the basis for storing information on the geography of the space. The records of the fundamental file correspond to elementary geographic areas which are minimal intersections of entities, and which are traditionally called GTUB's for "Geographic Tabulation Unit Base." Each GTUB record contains pointers for several lists. Each list type corresponds to an entity type, say "county". The county list type occupies a unique field in each record, and the value in that field is a pointer to the next GTUB with the same county code. Each list type corresponds to a partition of space and provides a method to store many specific lists of this type. The list elements are the subsequent records of the same entity.

The GTUB's are equivalent to the atomic elements in the complete geographic tabulation lattice. The multilist structure permits the reconstruction of the geographic lattice if desired—hence the properties of lattices may be utilized implicitly with the GTUB.

INTRODUCTION

A topologically sound geometric structure can guarantee mathematically consistent areal classification. The fundamental element of area in a topologically structured map file is the 2-cell; and in the United States, there are over ten million 2-cells. The surface of the country may be theoretically subdivided into those ten million 2-cells with all land area counted (covered) exactly once. All geographic areas are composed of sets of 2-cells. In that sense the 2-cell is the basic unit or fundamental building block for geographic regions. The 2-cells may be unnecessarily small and too numerous for the kind of "building" required to keep track of geographic relationships. Often contiguous 2-cells will behave as a group, remaining together under any geographic partition of the whole space. These groups which behave as single units make up the GTUB's. The 2-cells belonging to a single GTUB will all have the same geography; that is, they are contained in the same regions. GTUB's will be characterized by the property that their constituent 2-cells all have the same geography; and that any 2-cell not belonging to the GTUB must have different geography in at least one category.

A GTUB is an intersection of more familiar geographic regions. A computer data structure called a multilist will be used to represent GTUB's. The example in the next section is provided to illustrate the relations among 2-cells, GTUB's, higher-level geographic structures, and the multilist representation of the geographic relations.

AN ILLUSTRATED EXAMPLE OF GTUB/MULTILIST STRUCTURES

Consider the following subdivisions of the same rectangular area into 2-cells, regions, zones, and districts:

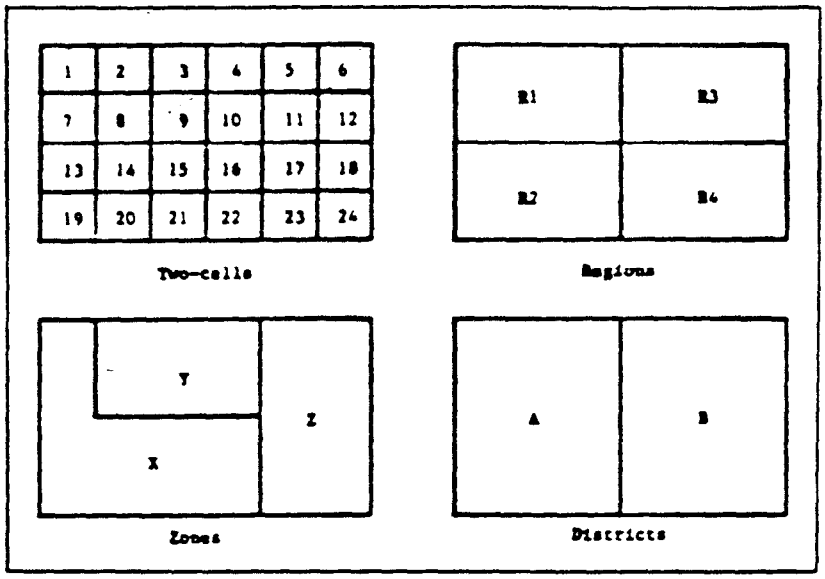

Figure 1. Different Subdivisions of a Rectangular Area.

The 2-cells are automatically a finer physical partition than any of the other subdivisions. All lines used to delineate geographic area boundaries **plus** all street and other linear feature lines go into determining the boundaries or outer limits of the 2-cells. The regions are also a finer partition than the districts in this example; zones are not comparable to regions or to districts in the sense that neither is finer than the other. A finer partition is called a **refinement.** A common refinement may always be obtained for any collection of partitions by taking intersections of areas. The common refinement for the zones, districts, and regions is shown below. This refinement has as its elements the GTUB's for zones, districts, and regions.

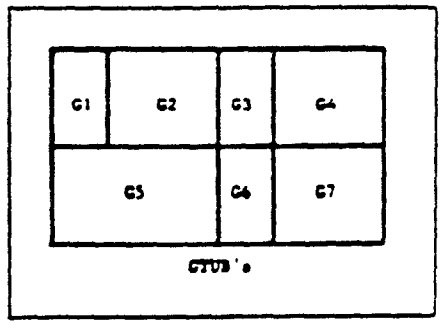

Figure 2. GTUB Sets for Zone-District-Region Geography.

The GTUB's depend on the collection of partitions of geographic areas. Adding other levels of geography will increase the number of GTUB's and make the new GTUB set a refinement of the earlier GTUB set. The GTUB set will always have fewer elements than the 2-cell set because every GTUB will still be composed of 2-cells. In the example above there are 24 2-cells and 7 GTUB's. If every 2-cell were subdivided into 100 2-cells in the example above, there would still be only 7 GTUB's. The number of GTUB's depends on the geography above the GTUB's, not the geometry below (i.e. the 2-cells, which originate from street and other linear feature patterns). The following figure is a Haase diagram showing the inclusion relations of the districts, regions, zones, and GTUB's. A line between entities indicates inclusion. The higher entity contains the lower entity.

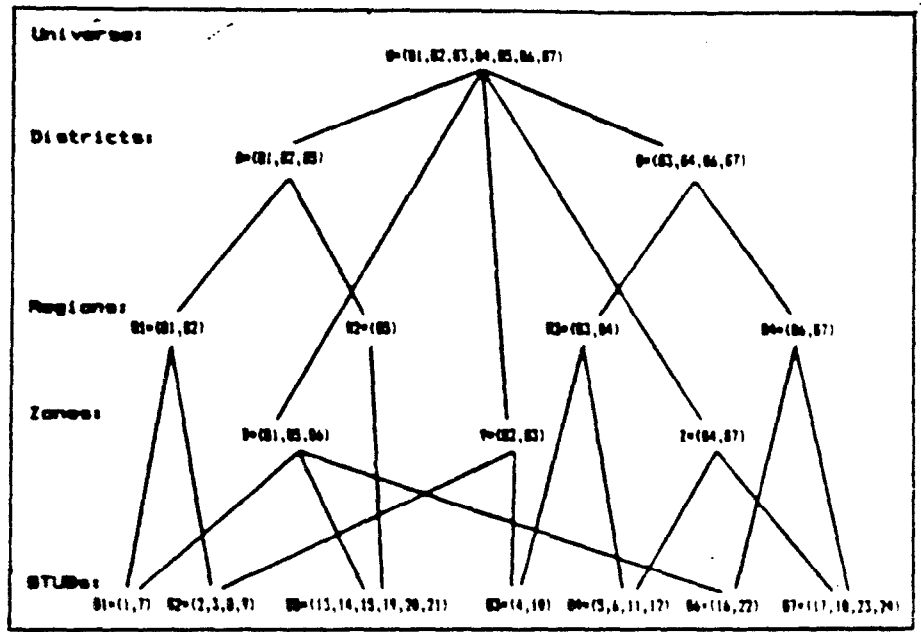

Figure 3. Haase Diagram for Set Inclusion Relations for GTUB's.

In terms of lattice theory, the GTUB's form the set of greatest lower bounds for sets of selected elements from every partitioning set. The GTUB's are also least upper bounds for sets of 2-cells having the same geography. The GTUB's provide a new level of geography between 2-cells and other entities which simplifies the relational scheme, as shown in the figures below.

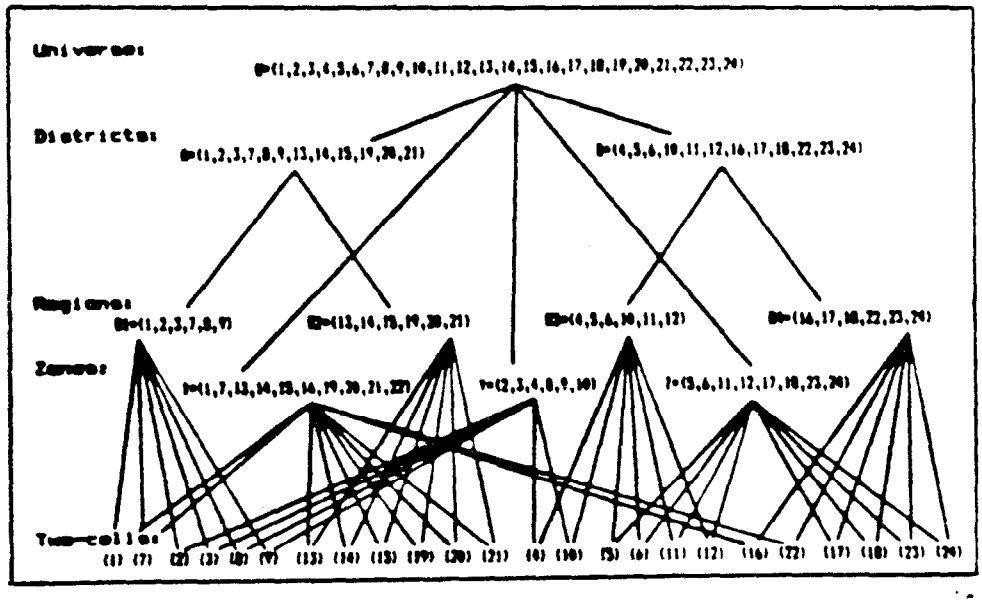

Figure 4. Set Inclusions of 2-cells Without GTUB's.

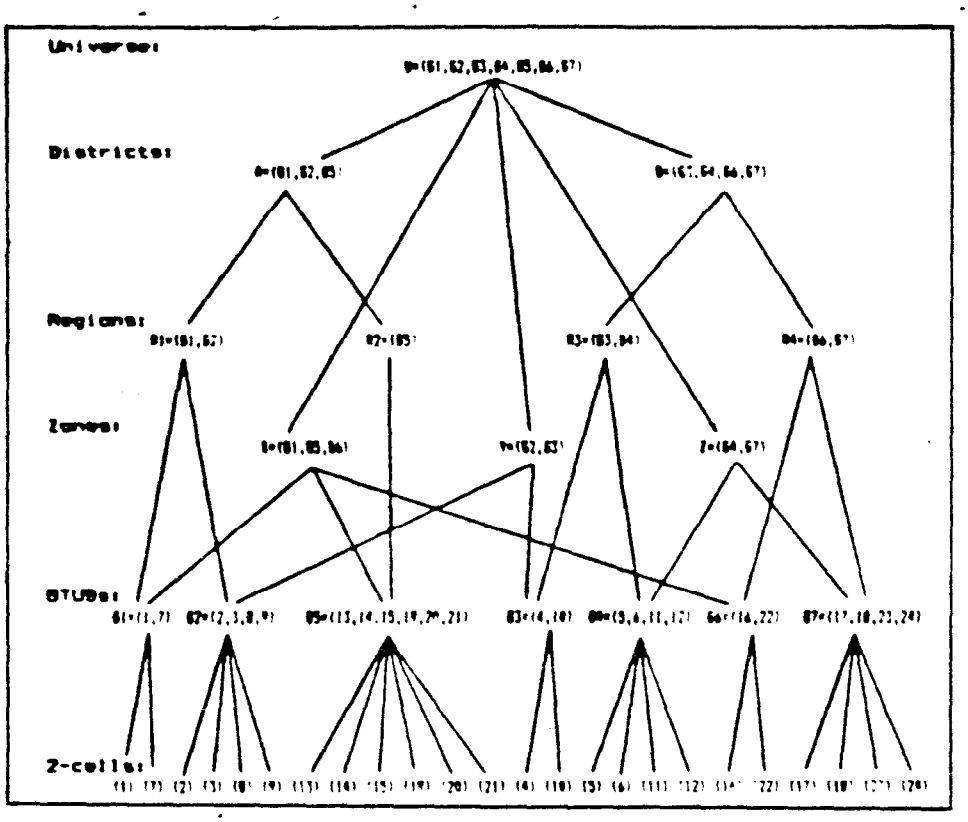

Figure 5. Set Inclusions of 2-cells with GTUB's.

Figure 4 illustrates the structure required to link 2-cells directly to the geography. Without GTUB's, every 2-cell must be joined to every minimal level of geographic entity type (in this case, zone and region). In this particular example there are only two minimal levels. In the more general case, for **m** minimal levels and **n** 2-cells, **mn** links are required. GTUB's provide a unique minimal level of geography on which to anchor all 2-cells.

Figure 5 reveals the clear separation of geography and 2-cells that the GTUB's provide. There are no fundamental inclusions of 2-cells in entities other than GTUB's. This permits the GTUB's themselves to function as the fundamental unit of geography for many applications which do not require the detail of 2-cell information. At the same time, however, the GTUB's are directly linked to the family of 2-cells to permit extracting information on those 2-cells when it is needed.

Figure 5 suggests that different data structures for GTUB's-and-above and for GTUB's-and-below are advisable. The GTUB's simply partition the set of 2-cells, whereas more complex interactions take place above the GTUB's. This will always be the case because the GTUB's are least upper bounds for 2-cell sets and greatest lower bounds for the other geography.

The next section describes the multilist representation for the geography with GTUB's and illustrates the description with the example given above.

Multilist Representation of Geography with GTUB's.

A multilist is a computer data structure capable of providing several simultaneous linkages of data. The data points are regarded as belonging to several lists—each data point belongs to one list for each list type or entity type. GTUB records are the data points, and unique fields on the records identify the containing lists and the list links. Directories support the multilist of GTUB's by keeping track of higher level geography. Higher level geography refers to entity types which have proper refinements between themselves and the GTUB's. An example of higher level geography in the illustration above is the district. The figures below present the collection of data structures required for multilist representation of the GTUB relations presented earlier.

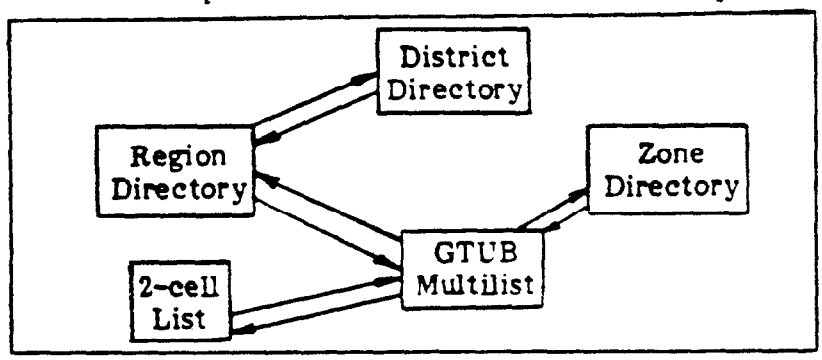

Figure 6. Files and File Links for Multilist GTUB Representation.

More specifically, the files for the example given above will consist of the following set of records:

District Directory

District ID	First Region
A	R1
b	R3

Two-cell List

2-cell ID	GTUB ID	Next 2-cell, Same GTUB
1	G1	7
2	G2	3
3	G2	8
4	G3	10
5	G4	6
6	G4	11
7	G1	--
8	G2	9
9	G2	--
10	G3	--
11	G4	12
12	G4	--
13	G5	14
14	G5	15
15	G5	19
16	G6	22
17	G7	18
18	G7	23
19	G8	20
20	G8	21
21	G5	--
22	G6	--
23	G7	24
24	G7	--

Region Directory

Region ID	First GTUB	District ID	Next Region, Same District
R1	G1	A	R2
R2	G5	A	--
R3	G3	b	R4
R4	G6	b	--

Zone Directory

Zone ID	First GTUB
X	G1
Y	G2
Z	G4

GTUB Multilist

GTUB ID	First 2-cell	Region ID	Next GTUB, Same Region	Zone ID	Next GTUB, Same Zone
G1	1	R1	G2	X	G8
G2	2	R1	--	Y	G3
G3	4	R3	G4	Y	--
G4	5	R3	--	Z	G7
G5	13	R2	--	X	G6
G6	16	R4	G7	X	--
G7	17	R4	--	Z	--

Figure 7. Contents of All Files in the Multilist Database.

Within each file, records correspond to entities or specific areal units. Information is grouped as follows on the records:

Entity identifier: links to lower levels: links to higher levels.

Not every record will contain all three groups of information, although two of the three will always appear. The 2-cells have no lower levels; and, in the example above, districts and zones have no higher levels.

Although the entire collection of geographic entities does not form a hierarchy, some subsets of entity types do form hierarchies. The entity type hierarchies result when one level of geography is a refinement of another level (counties within states, for example). The file organization will reflect this.

Within the GTUB records, the various fields corresponding to the "Next GTUB, same entity" link the records in individual lists that constitute the multilist. In the example above, the "Next GTUB, same entity" field of the GTUB record corresponding to entity type "zone" produces 3 linked lists of GTUB's: (G1, G5, G6,), (G2, G3), and (G4, G7) corresponding to the zones X, Y, and Z, respectively.

Notice that GTUB lists are not generated for higher level geography, such as district. GTUB lists may be built for higher level geography by aggregating lists of lower level entities which constitute the higher level entities. The aggregation procedure imitates the initial list formation routine: references to the "Next lower level entity, same higher level entity" generate lists of lower level entities. Each lowest level entity corresponds to a list of GTUB's. The lists of GTUB's for every lowest level entity in the higher level entity, can be concatenated to produce a list for the higher level entity.

Decomposition procedures of higher level entities of any type into lower level entities amount to list building and concatenation operations for the multilist structure described above.

MULTILIST FILE COMPLEXITY

The example presented in the previous section greatly simplifies the interactions of files in order to illustrate clearly the internal file structures. The next figure from a Census Bureau TIGER specification memo illustrates some, but not all, of the geographic areas which were used for tabulation geography on the 1980 Census of Housing and Population. The entities shown are those 1980 geographic classifications which will again be retained through the 1990 Census. The following diagram represents a subset of both the 1980 and the 1990 geographic areas.

The next figure, figure 8, although far more complex than the illustration of figure 6, represents approximately one-third of the total levels of geography that will be carried in the 1990 Census of Housing and Population. Currently there are close to twenty lowest level geographic entity types immediately above the GTUB level being considered for inclusion in the 1990 geographic database.

In addition to possessing complex interactions, the geographic data base is subject to change. Layers of geography may be added or subtracted. Entities within entity types may be redefined or reorganized. Boundaries may change, necessitating the creation of new units at every level, even down to the 2-cell.

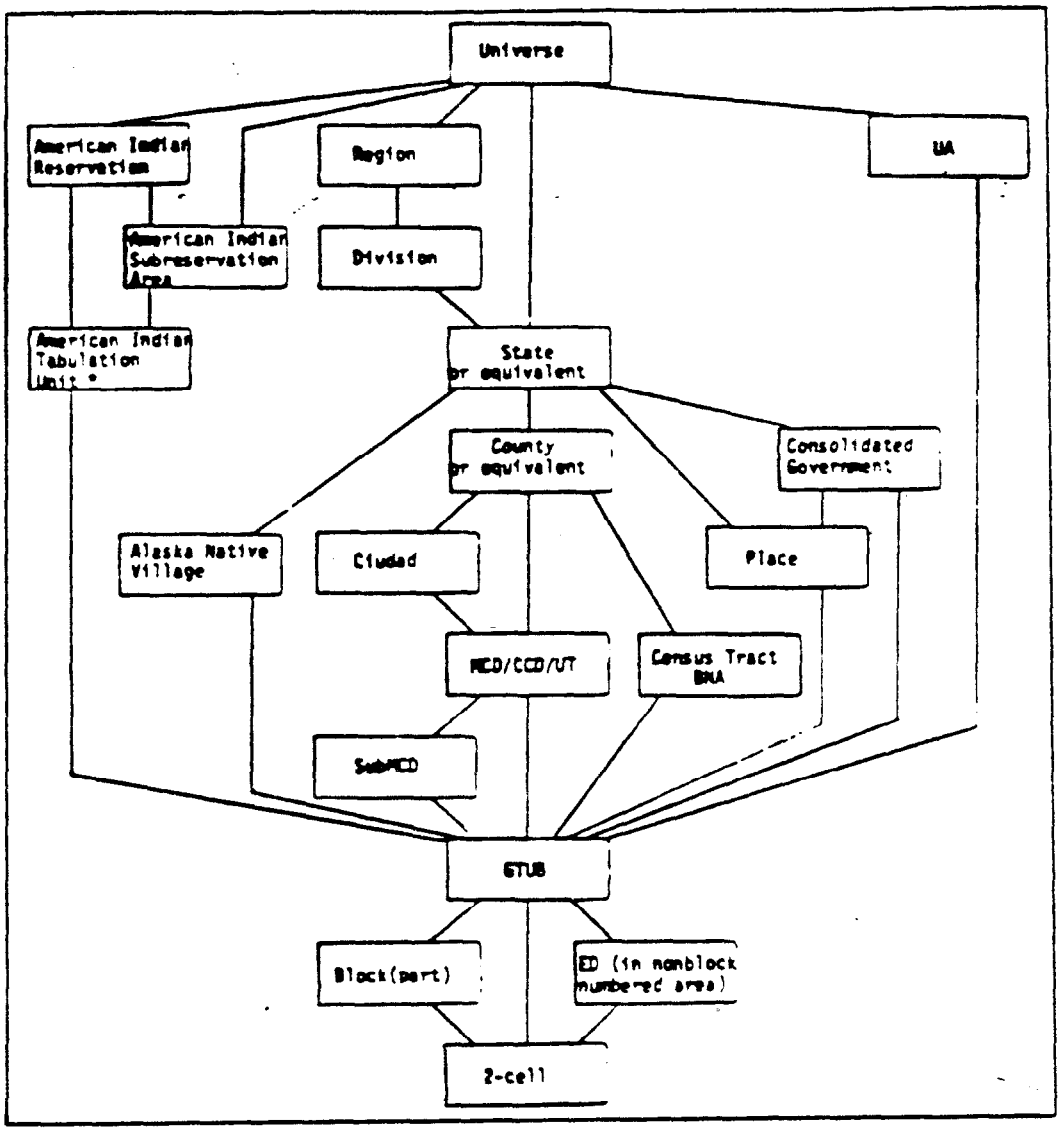

Figure 8. Some 1980 U.S. Geographic Areas to be Included in 1990

In order to organize large dynamic geographic structures such as the structure diagrammed above, the multilist database will require many file management capabilities. Some of those capabilities are summarized in the final section.

ONGOING RESEARCH AND DEVELOPMENT WITH GTUB'S

The Haase diagram in figure 5 illustrated the way that GTUB's can replace 2-cells as the fundamental unit of geography. The efficiency gained in working with GTUB's will depend to a great extent on the relative sizes of the GTUB file and the 2-cell file. As partitioning sets increase, the GTUB's get smaller and their number increases. In theory, the GTUB's could become as numerous as the 2-cells. Some investigation is underway to study the trade-offs and to determine the point at which the GTUB's become inefficient.

The GTUB's, even with their potential inefficiencies, nevertheless provide a minimal set of building blocks from which to build all of the geographic entities. This aspect of GTUB's was not emphasized in the earlier sections; and it is an important consideration for maintaining geographic consistency. Regions may be reconstructed from their GTUB's by means of multilists; and, moreover, the boundaries of those GTUB's may be added using a Boolean sum to reconstruct the boundary of the containing region. The integrated approach of the GTUB and the multilist totally eliminates the polygon overlay

inconsistencies that result from storing different types of entities on separate files. The overlay figure below illustrates the manner in which GTUB's behave as jigsaw puzzle pieces which can be used to build any geographic entity.

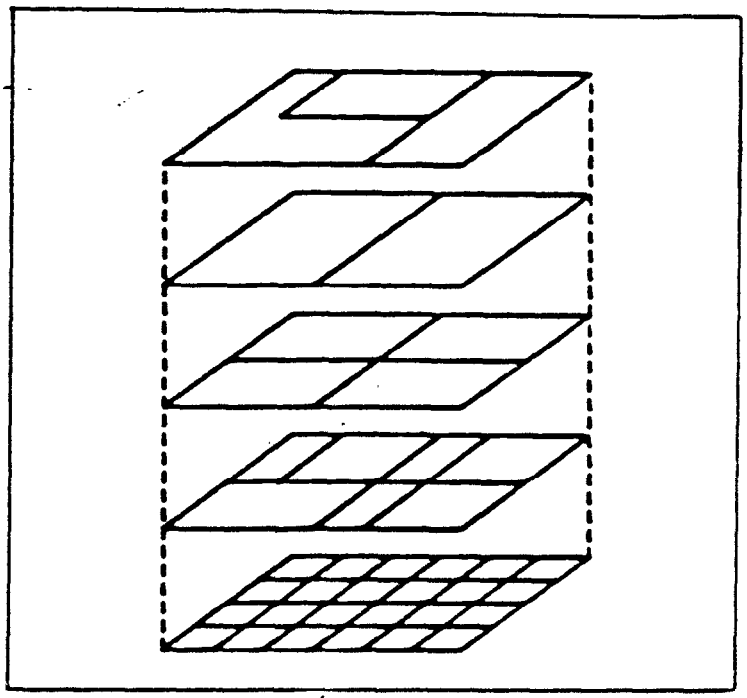

Figure 9. All Overlays are Made Up of GTUB's.

Although the directories in the example presented in this paper were given the same multilist structure as the GTUB file, in practice there may be shortcuts and savings gained by using other structures for higher level directories and files. Census coding of many entities embeds higher level identifiers in the codes of the lower level entities. For example, a Minor Civil Division code carries digits identifying the county and state. This additional structure will be utilized in implementing a multilist database for GTUB's, although it was not specifically acknowledged in the illustrative presentation above.

GTUB file building may be accomplished by adding the layers of geographic entity types one layer at a time. As layers are added, the new GTUB partition becomes a refinement of the old GTUB partition. GTUB elements are split according to way that the new entities partition the 2-cells within the old GTUB. Processing of the splitting of GTUB's by additional layers of geography may be automated to a high degree provided locational information on 2-cells can be retrieved readily, and provided that the new layer's entities are also defined geometrically.

Removing layers of geography results in the consolidation of GTUB's. This process may be accomplished without locational information about the 2-cells.

Several other areas of database management for the GTUB multilist are candidates for study for efficient algorithms. Because of the large file sizes, efficient management is critical.

BIBLIOGRAPHY

Bureau of the Census, 1984, **TIGER Specification Memorandum Series**, GEO-CAT—Phase I, Ch. 2, Doc. 5, Washington, DC.

Wiederhold, G., 1977, Database Design, McGraw-Hill, NY, NY.