A SPATIAL-OBJECT LEVEL ORGANIZATION OF TRANSFORMATIONS FOR CARTOGRAPHIC GENERALIZATION

Robert McMaster — Department of Geography
Leone Barnett — Department of Computer Science
University of Minnesota
Minneapolis, Minnesota 55455

ABSTRACT

A current research thrust in geographic information systems is the attempt to add generalization functionality to existing software. However, it is difficult to create user-friendly environments for non-experts attempting to generalize maps because the organization and naming (or description) of generalization operators is based on a mixture of human-focused functionality and computer-focused data structuring. The generalization operators have names such as simplification, smoothing, displacement, enhancement, typification, and exaggeration, which suggest their functions. However, they are commonly divided into raster and vector groups of operators, which specify their applicability to a data structure or their geometry (points, lines, and areas). Analysis of these operators, focused on an examination of their higher level, or human oriented, purposes, reveals that there is redundancy among them, and, at times, multiple purposes embedded within them. In this paper, a structure for generalization operations is created that addresses the different levels of abstraction inherent in understanding the operators. While most existing frameworks organize operators at the data structure or conceptual raster or vector level, the spatial-object level may be used for a higher-level organization. Such a structure offers an alternative to designing and implementing generalization operators based solely on the raster/vector dichotomy and geometry.

INTRODUCTION

Current methods of automated cartographic generalization, although useful and necessary for manipulation of geometric objects, fall short of providing a system with general capabilities that produce adequate results. There is a growing awareness that cartographic features are not simply geometric objects. According to Muller (Muller 1991), "more sophisticated methods will be required which take into account the phenomenal aspects of a line." This will include system use of rules that allow more intelligent decision making.

At this point, there is not even a clear set of concepts concerning generalization that establish standards for the design of a generally useful interface, allowing a human to interact with the system at a relatively high level. Currently, humans must be very tuned in to the details of the geometric representations of their data (i.e. vector, raster, point, line, etc.). This is not likely to be the aspect of their data that is of utmost importance to them, but they are forced to consider this aspect to use the system effectively.

The problem of how to incorporate real-world, or "phenomenal," information in a manner that can facilitate automated generalization can be approached from a perspective of "semantic information" analysis. In this context, semantic information is whatever information is necessary to understand the meaning (or meanings) of data stored in the computer. System support of this information means the system has the ability to attach appropriate meaning (via other data and operations) to the data it controls. Analysis, in this case, of generalization operators, is necessary because in order to codify and represent the meaning of the data the operators are applied to, we need a better understanding of what the operators actually do.

To facilitate this analysis, we offer a perspective on the relationship between geometric or spatial data, and real-world data. This perspective is summarized in Figure 1. Using this, we can understand and classify generalization operators at a spatial level of abstraction that removes the concern with vector and raster representations at the higher level, while maintaining them, appropriately, at the lower level. Before we discuss Figure 1 as the basis for a new, semantically oriented, approach for organizing generalization operators, we review previous approaches to operator classification.

PREVIOUS WORK

Some authors have attempted to formulate rules concerning generalization. In trying to understand what rules could guide the generalization process, authors have come up with different ways to think about and describe the different generalization tasks involved. They organize the tasks in terms of immediate purpose, high level function, focus of use, and/or constraints.

We begin with Brassel (1985) who developed one of earliest comprehensive listings of operations. Brassel organized generalization operations by geometry, and included classes of (1) point features, (2) line features, (3) area features, and (4) volume features. The point features category included operations such as expand, displace, classify, and display while several of those operations found in the area features category included select, eliminate, classify and displace. Brassel's structure included redundancy among the four categories, as well as operations that many authors would not include as part of generalization (such as the line features operation of "change topology in linear networks"). It was, nonetheless, the first attempt to organize what had been, up to that time, a rather haphazard development of generalization operations.

Beard examined generalization operators with respect to the formalization of generalization rules. Beard's view of generalization operations is based on the constraints that are pertinent to generalization. The four constraints include (1) graphic, (2) structural, (3) application, and (4) procedural. Procedural constraints, for instance, control the order and interaction of actions or operations as well as the order in which these constraints are satisfied (Beard, 1991). As shown below, Beard identifies seven operations for generalization. She categorizes them into three broad classes, including: (1) operations to reduce the number of objects, (2) operations for simplifying the spatial domain, and (3) operations for simplifying the attribute domain. Those operations designed to reduce the number of objects, for instance, include select, group, link, and aggregate. Those operations designed for simplifying the spatial domain include link, aggregate, collapse, and simplify. Classification is designed for simplifying the attribute domain. Each of these operators is then categorized in terms of graphic, structural, and application constraints.

Generalization operations: Beard

(1) select
(2) simplify
(3) collapse
(4) link
(5) aggregate
(6) classify
(7) group

Generalization techniques: Mackaness

(1) change symbols
(2) mask symbols
(3) increase size difference
(4) select
(5) omit
(6) simplify
(7) combine (reclassify)
(8) displace
(9) exaggerate

Mackaness (1991) provides an alternative approach. In identifying nine generalization techniques, he focuses on the process and the structure of the processes. Some of these techniques, such as select, omit, simplify, combine, displace, and exaggerate are similar to those of other investigators. Others, including change symbols, mask symbols, and increase size difference, relate more to the manipulation of symbols than to geometric generalization. Mackaness proposes that both rose diagrams and thermometers can be used to measure the interaction among and individual performance of the operations.

The structure proposed by McMaster and Monmonier, and reported in McMaster (1991), classifies the operators of generalization into those designed for manipulating the geometric description of geographical elements, and those designed for manipulating the attributes of (typically, simple cell) elements. Operators involving attribution include classification and symbolization. Many authors consider classification, symbolization, and simplification to be core concepts of map generalization. A further division of operations identifies those used for raster and vector data structures. For instance, generalization of raster data can involve structural, numerical, numerical categorization, and categorical classes of operations. Generalization of vector data involves operators that are designed for a specific vector data type--point, line, area, and volume. Commonly, those operators applied to linear features in vector format include simplify, smooth, displace, merge, enhance, and omit.

Despite the reasonably successful attempt to clearly distinguish between vector and raster operations, a structure such as the one described above, ignores the fact that redundancy exists both between raster and vector operators, and among operators for different vector data types. Additionally, such a structure would not facilitate the use of semantic-level information in improving the results of generalization. Most of the operations, and specific algorithms, have been designed at the data structure level or the conceptual raster or vector level.

It is our goal to understand and organize generalization operators, in terms of the functional tasks they actually perform, in a manner that makes clear when differences are important, and when similarities should be acknowledged. To this end, we have developed a new model, based on levels of meaning, that provides a context for understanding and specifying the meaning of operators.

OVERVIEW OF LEVELS

In Figure 1 we present a model of how different types of information may be organized into levels of "meaning." The "lowest" level, shown at the bottom, is called the "Data Structure or Implementation Level." The "highest" level, shown at the top, is the "Real-world Phenomena Model Level." This level might also be called the "Entity Level" by those familiar with the Spatial Data Transfer Standard (SDTS) terminology (National Institute of Standards and Technology 1992). Referring to a level as "high" or "low" is somewhat arbitrary. However it does reflect that concepts primarily of interest to humans are called "high level" relative to "low level" concepts primarily useful in computer programming. The levels that involve real-world information are also levels where many types of "semantic" information may be found.

The model was developed as a means of understanding the generalization operators, both existing and desired, in terms of functionality. For our purposes, functionality refers to specific types of data transformation and manipulation. A justification for modeling the different levels of meaning that are involved in generalization is, based on what level you assume an operator exists, the meaning of the operator (its function) is different. The following three assumptions concerning generalization are the basis for the structure of the model:

1. There are two traditional, and very different, classes of data transformations that occur as part of cartographic generalization. One is spatial-object based (i.e. based on geometry), and the other is real-world information based. (Often, real-world information is stored as attributes for raster data.)
2. The term "attribute" is over-loaded, and depending on what level of meaning is assumed, an attribute plays a different role and takes on a different meaning.
3. A conceptual spatial-object level subsumes lower level spatial models such as those known as vector and raster. There are limitations or constraints on operations that may be fundamental to a lower level model, such as vector or raster, but do not apply at the conceptual spatial-object level.

Note that the assumptions as stated go slightly beyond the commonly accepted view that generalization predominantly involves spatial transformations and attribute transformations. While this is true, it doesn't explicitly emphasize the important connection between attributes and real-world information. In addition, attributes may be any type of data, including spatial or geometric. Assumption number 1 above more clearly emphasizes that any spatial transformation is in a separate class from transformations requiring some sort of real-world data.

In the next section, each level is described in more detail. We identify the major classes of activity or function in each. We discuss how some existing generalization operators fit into these classes, and that some even fit into multiple classes because they serve multiple functions.

DESCRIPTION OF LEVELS

Data Structure or Implementation Level

The "Data Structure or Implementation Level" is of the least importance in this paper, but is included to illustrate the difference between a *conceptual* vector or raster structure and its actual *implementation*. This difference exists since a line may be represented many ways. For example, a line at the conceptual vector level, may be implemented at the lower level, as a sequence of x, y coordinate pairs, with each list stored in a separate record. Alternatively, the same conceptual line may be represented as a line object, where its line-id either points to or is related to a separate file containing points. This level may be broken down further, if desired, as was described by Peuquet (1990). However this is the lowest level we need to distinguish for the purposes of this paper. At this level, attributes are any values, irrespective of meaning, that are associated with the data structures used for either the vector or raster objects.

Conceptual Vector or Raster Level

This level ideally could include all lower level spatial models that manage various types or aspects of spatial data. For now, due to the types of operators developed for generalization, this level comprises two separate models that have received much attention in the literature. These separate models, often referred to as the vector and raster models, are sometimes described as the "dual" of each other (Peuquet 1988). The conceptual raster model is based on a regular partitioning or tessellation of space. The conceptual vector model is based on a point, line, and polygon description of the geometry of an object in space. Attributes at this level are, again, any values, irrespective of meaning, that are associated with the vector or raster primitive objects. These include points, lines, and polygons for the vector model, and raster cells for the raster model. This level is significant since many operators have been developed with this level in mind, as described previously.

In subsequent sections we examine transformations that relate specifically to either the conceptual raster model or the conceptual vector model. We identify general categories of activity for each of these, and we indicate which generalization operators exemplify an activity, at least as far as one of their functions is concerned. The conceptual vector model is discussed next. Then, just prior to discussing the conceptual raster model, some higher levels are introduced.

Conceptual Vector Level

Transformations that relate to the conceptual vector model are easily confused with those of the next higher level, which is called the "Conceptual Spatial Object Level." This is because the higher level subsumes the conceptual vector model. This means that many spatial transformations can occur at both of the levels involved. The major distinction between the conceptual spatial-object level and the conceptual vector (or raster) level centers around the definition of an area. At the conceptual vector level, the basic objects, or primitives, include the point, the line, and the polygon. The line is a

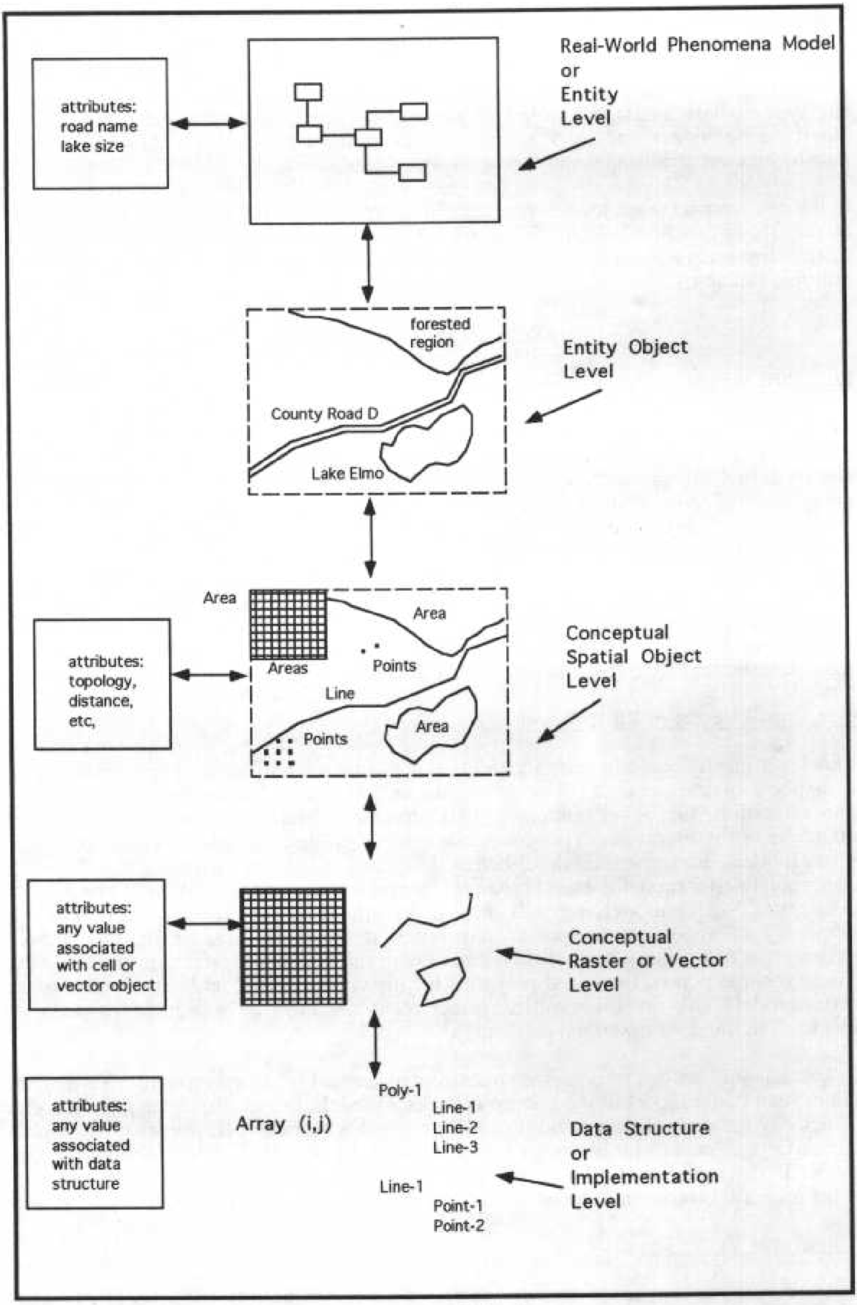

Figure 1: Levels of Meaning

sequence of points, and the polygon may be either a closed sequence of lines or points. The polygon defines the boundary of an area, but that is the only way in which an area may be represented. At the conceptual spatial-object level, the primitives are the point, the line, and the area. An area in this case is a more complicated notion. It includes the type of area commonly represented by a raster cell. It also includes the type of area that may be represented by a cluster of contiguous raster cells. Thus at the conceptual spatial-object level, the area concept includes, but is not limited by, the conceptual vector means of modeling area.

This is a subtle distinction, and often one that is either not explicitly recognized, or is ignored in the discussion of generalization operators. The result is a use of terminology that is not quite clear. For example, a discussion of the "smoothing" operation, without specification of the level of interest, may refer to vector line smoothing, or a raster smoothing operation called "erode smooth," or smoothing of any spatial object that happens to be a line. In fact, an author may have the conceptual spatial-object level "in mind," yet discuss the concepts as if the conceptual vector-level constraints must apply.

Given the conceptual vector primitives, the conceptual vector-level transformations mostly concern conversions from one type of primitive to another of the same or different type. Many of these conversions are identified and named (as generalization operators) by McMaster and Shea (1992). Their names of groups of vector generalization operators include "Point feature generalization," "Line feature generalization," and "Area feature generalization." Their first group includes "aggregation" and "displacement." Their second includes "simplification," "smoothing," "displacement," "merging," and "enhancement." The last one above includes "amalgamation," "collapse," and "displacement."

This is not a false categorization for the conceptual vector-level operators, but it does not serve to completely and precisely specify all such operators. For example, the same name is used to specify more than one operation (as in "displacement") making that operator imprecise at the vector-model level. "Aggregation," which operates on points, may indicate a points-to-area transformation or a points-to-point operation. If it specifically means the points-to-area operation, what operator handles points-to-point? An operator that is missing is one that involves the removal of a common boundary line between two areas (i.e. polygons). Although similar to "amalgamate" it is technically different, since one involves contiguous polygons with a common line boundary, and the other involves non-contiguous polygons.

Given the lack of fine distinctions of these operators with respect to the conceptual vector level, it may be that the operator names are more well suited to the conceptual spatial level instead. Indeed, that is the level some authors intend when they use these operator names. That may be reasonable to do, but there is clearly a difference in the understanding of what the operator does if the context is one level or the other.

A more complete organization of conceptual vector-level transformations would specify, within each of the broader categories of point, line, and polygon generalizations, exactly what geometric changes occur. The specifications would include what type the original object would change into (e.g. points-to-area), as well as how many of the original objects are to be generalized (e.g. one vs. more than one). In addition, the contiguous or non-contiguous arrangement of the objects should be factored in. As an example, an operation such as line simplification could be classified as a 1-line-to-1-line type of transformation, and aggregation as a many-points-to-1-polygon transformation. If a many-points-to-1-point operation is desired (which some may view as an aggregation operation also), the difference between the two kinds of aggregation can be stated clearly and better understood. Although at a higher conceptual level the distinction may not be of interest, at the lower levels, it is important. The design of algorithms to perform operations at, for example, the conceptual vector or raster level, depends on such distinctions. We are currently involved in researching a full specification and classification of the conceptual vector-level operations.

Real-world Phenomena Model Level

Just as there is a tendency to mix conceptual vector-level concepts with the spatial object level, there is a tendency to mix conceptual raster-level concepts with those that involve real-world information. Thus, we introduce the levels involving such information first.

Often in research on automated generalization techniques, real-world information is either ignored (as in vector based methods), or viewed as "attributes" of a location (as in raster based methods). This may be adequate when spatial-object manipulations are the primary focus, but, increasingly, the meaning of the spatial object is being viewed as an important factor in generalization. Once real-world information, as well as spatial-object descriptions, are available to the system, more powerful generalization techniques may be developed.

To this end we have included a "Real-world Phenomena Model Level," and distinguish it clearly from the "Conceptual Spatial Object Level." The real-world phenomena model level includes any and all information that is deemed important enough to remember or utilize in generalization, but that is not part of the geometric or spatial description of real-world objects. Thus, the modeling of the river that a line stands for occurs at the real-world phenomena-model level, but the modeling of a line may occur at the conceptual spatial-object level. Or, a model of land use that is based upon classification of reflectance values belongs at the real-world phenomena model level, but the raster cell that holds specific attribute values does not. A fundamental class of activity that occurs at the real-world phenomena model level, and is important in generalization, is classification.

The Entity Object Level

This level is introduced to show the relationship between real-world phenomena modeled at the real-world phenomena model level, and spatial objects representing them, which are modeled at the conceptual spatial-object level. The name "entity object" is borrowed from the SDTS. In there, it was used to distinguish between a spatial object per se and a spatial object that is used to represent a real-world entity (or feature). Thus, in our figure, transformations that belong at the entity-object level are those that involve both the spatial object and whatever it represents. An open research issue concerns the identification of the major categories of transformation that may occur at this level. However, we can begin thinking about transformations at this level by analyzing the functions that are typical of the lower levels.

In contrast with many conceptual vector-level generalization operators, operators developed for use with the raster model actually perform functions at the entity-object level. That is, since they use and/or transform both real-world attribute data as well as structural spatial data, they occur at the level where the two types of data are bound. Although we describe below how some existing raster generalization operators actually exemplify activities at this level, the most commonly discussed vector generalization operators do not. However, there are some operations in commercial vector-based geographic information systems (GIS) that possibly should be classified at the entity-object level, in that spatial object descriptions are changed based on real-world types of attribute values. For example, although not typically viewed as generalization, the set operations in Arc/Info allow overlay and union of polygons, wherein attributes are used in specifying the operations, and new polygons are produced as the result.

The Conceptual Raster Level

Having described the levels that relate to real-world information, we can now identify the basic functional activities that occur at the raster level. Existing raster generalization operators help illustrate the functional activities. Three basic categories of functions that occur at the raster level are: change structure of cells, change attributes of cells, and change structure via attributes. The change in structure of cells may affect either cell size or cell number. The change in attributes of cells may involve data from neighboring cells or not. Due to potential attribute changes, the change to cell structure via attributes may also involve data from neighboring cells or not.

Following are examples of existing types of raster generalization operators, which are described in the McMaster and Shea framework, that fall into each of these functional categories:

CHANGE STRUCTURE OF CELLS
- change cell size
  resampling
- change total cell number
  simple structural reduction

CHANGE ATTRIBUTES OF CELLS
- use data from neighboring cells
  low pass filters and high pass filters
  edge enhancement
  feature dissolve, erode smooth, gap bridge
- use no data from neighboring cells
  categorical merge
  numerical categorization (image classification)

CHANGE STRUCTURE VIA ATTRIBUTES
- use data from neighboring cells
  categorical aggregation
- use no data from neighboring cells

Some of the above listed operators may also be classified as entity-object level operations. That is, any operations that require both attribute data and structural data must operate on the level where the two are bound. The clearest examples are methods that use data from neighboring cells. In contrast, methods that only change the structure of cells would not be classified at the entity-object level, as they perform a function strictly at the raster level.

At the purely raster level these functional groups of activity appear to be complete. However, raster data may be used to implement concepts at the spatial-object or entity-object levels. In order for this to occur, the notion of a "raster object" must be supported (Schylberg 1993). By this we mean that the system must be able to recognize a collection of contiguous raster cells, forming any shape, as an areal object. Raster object identification and management, typically not supported in GIS's, might be viewed as a raster level functional activity, but its purpose would primarily be to serve higher levels of information management.

Spatial Object Level

The "Conceptual Spatial Object Level" supports descriptions and transformations of objects in space that are based on the spatial object primitives, the point, the line, and the area. Since general spatial descriptions often also include spatial relationships and topology, this type of information belongs at this level, regardless of whether they are implicit or stored directly, possibly as attributes. Note that while attributes may occur at this level, the meaning of a spatial-object attribute has some spatial connotation. This does not imply that spatial objects never have real-world attributes associated with them. However, if an attribute with real-world meaning is directly associated with a spatial object, it means that an entity object supporting the connection between the two has effectively been created.

The transformations that support generalization at the spatial-object level include, in the sense that they subsume, the spatial/structural transformations of lower levels, such as vector and raster. However, at the spatial-object level, two lower level operators performing the same conceptual spatial operation would be redundant. Thus identifying, naming, and classifying operators at the conceptual spatial level should be done in a way that groups similar functions, and provides one name for like functions.

Although we are in the process of researching a full specification of functional operations for this level, this is not yet completed. However, we have identified some basic categories of function or activity. These categories are based upon generalization goals that are achieved through operators that manipulate spatial or geometric objects.

Specific algorithms have been developed that attempt to mimic traditional generalization techniques, e.g. simplification for weeding unnecessary information, smoothing for improving the aesthetic qualities, and displacement for shifting features apart. One major objective of generalization is thus for the aesthetic graphic display of cartographic information. Operations such as displacement, smoothing, and enhancement are designed for display purposes and can be classified under a display category of activity. A second objective is to reduce spatial and aspatial information. The reduction may be controlled either geometrically, as with simplify, collapse and structural reduction operations, or by making use of attribute data, as with low-pass and high-pass filters. The third objective of generalization, fusion, which joins or combines, may also be controlled either geometrically (as with aggregate, merge, and amalgamate operations), or by use of attributes, as with parallelepiped and minimum-distance-to-means operators. At a conceptual spatial level, then, basic classes of generalization functions include aesthetic display, reduction of information, and fusing of spatial features.

Notice that at the higher conceptual-spatial level, these basic classes are relevant for both vector and raster data. In addition, similar activities are grouped under a single classification. For instance, in dealing with objects in vector mode, a user amalgamates areas (which possibly represent census tracts), merges two linear features together, (which may represent the banks of a river), or aggregates point features into one area. All three, of course, fuse spatial objects together, whether they are points, lines, or areas. Likewise, a polygon dissolve in raster-mode fuses features together. Below, we depict this simplified structure and give examples of how existing operators may be classified.

DISPLAY
- *displace*
- *smooth*
- *enhance*

REDUCE
- geometrically-controlled
  - *simplify*
  - *collapse*
  - *structural reduction*
- attribute-controlled
  - *low-pass filters*
  - *high-pass filters*

FUSE
- geometrically-controlled
  - *aggregate*
  - *merge*
- attribute-controlled
  - *amalgamate*
  - *minimum distance to means classification*
  - *parallelepiped classification*
  - *merge categories*

Looking at this categorization, it is obvious that, sometimes, the means of accomplishing a spatial-object transformation is through the use of attributes. That provides a clue that some of these operations may actually be working at the entity-object level. It also indicates that some of these operators have more than one function embedded in them. To put it another way, a specific kind of spatial-object transformation should be viewed as one function, and an attribute change or manipulation should be viewed as another, even if the two functions are accomplished with one operator at a lower level.

The conceptual spatial-object level should have transformations identified and defined for this level, that strictly deal with spatial-object changes. Once we have completed the identification and description of basic transformations for the spatial-object level, it will become easier to define transformations that are fundamental to the entity-object level.

This should be the basis for providing general support for integrating semantic information in with geometric descriptions of concepts.

CONCLUSION

In order to develop cartographic generalization operators that make greater use of semantic information, and to develop systems that can effectively support such operators, a better understanding of the relationship between real-world information and spatial-object descriptions for them is required. In addition, a better specification of what higher level operators ought to do with respect to various types of semantic information is needed.

To this end, we have begun a process of analysis. As an initial step we have identified the different conceptual levels of meaning that are currently, although usually implicitly, involved in the functions that generalization operators perform, or are envisioned to perform. The explicit separation and recognition of these levels allows us to understand the basic types of transformations and functional activities that occur at each. With this understanding, development of generalization operators can be appropriately focused on the level of interest to the developer. In addition, requirements for lower level operators needed to support higher level activities can be more easily identified.

REFERENCES

Armstrong, Marc P. 1991. "Knowledge Classification and Organization," in *Map Generalization: Making Rules for Knowledge Representation*. Barbara Buttenfield and Robert B. McMaster, (eds.). Longman, United Kingdom, pp. 103-118.

Beard, M. Kate. 1991. "Constraints on Rule Formation" in *Map Generalization: Making Rules for Knowledge Representation*. Barbara Buttenfield and Robert B. McMaster, (eds.). Longman, United Kingdom, pp. 121-135.

Brassel, K.E. 1985. "Strategies and Data Models for Computer-Aided Generalization" *International Yearbook of Cartography*, Vol. 25: pp. 11-29.

Buttenfield, Barbara P. and Robert B. McMaster. 1991. *Map Generalization: Making Rules for Knowledge Representation*. Longman, United Kingdom

Mackaness, William A. 1991. "Integration and Evaluation of Map Generalization" in *Map Generalization: Making Rules for Knowledge Representation*. Barbara Buttenfield and Robert B. McMaster, (eds.). Longman, United Kingdom, pp. 217-226.

McMaster, Robert B. 1991. "Conceptual Frameworks for Geographical Knowledge," in *Map Generalization: Making Rules for Knowledge Representation*. Barbara Buttenfield and Robert B. McMaster (eds.). Longman, United Kingdom, pp. 21-39.

McMaster, Robert B. and K. Stuart Shea. 1992. *Generalization in Digital Cartography*. Resource Publication of the Association of American Geographers.

National Institute of Standards and Technology. 1992. *Federal Information Processing Standard Publication 173 (Spatial Data Transfer Standard)*. U.S. Department of Commerce.

Peuquet, Donna J. 1990. "A Conceptual Framework and Comparison of Spatial Data Models," in *Introductory Readings in Geographic Information Systems*. Donna Peuquet and Duane Marble (eds.). Taylor & Francis, New York.

Peuquet, Donna J. 1988. "Representations of Geographic Space: Toward a Conceptual Synthesis." *Annals of the Association of American Cartographers*, 78(3). pp. 375-394.

Schylberg, Lars. 1993. *Computational Methods for Generalization of Cartographic Data in a Raster Environment*. Photogrammetric Reports No. 60, Royal Institute of Technology, Department of Geodesy and Photogrammetry, Stockholm, Sweden.