COMBINING SPACE AND TIME: NEW POTENTIAL FOR TEMPORAL GIS

INTRODUCTION: THE MAP METAPHOR

Humans have always needed to share information about their surroundings, whether as members of a band of hunter-gatherers or as European traders exploiting new routes to the Orient. From the earliest times, the preferred method of representing geographic information was the analog model, in which the true dimensions of the Earth's surface were scaled or proportioned so they could be fitted on an easily handled sheet of paper, or onto a globe. By the 19th Century, the technology of gathering, collating, interpreting, and printing maps had developed to a fine art, and detailed maps of much of the Earth's land surface were readily available from national mapping agencies and commercial firms.

Despite their elegance and efficiency, paper maps have always suffered from some fundamental inadequacies as media for storing what is known about the surface of the Earth. Most obviously, they require that the Earth be flattened, a trivial requirement in the case of local areas but increasingly problematic when maps must cover large proportions of the Earth's surface. The Earth is not flat, and in principle no flat map can ever show aspects of the Earth's surface in constant proportion, so over the years humans have become used to the inherent distortions of the paper sheets of atlases. Maps are also expensive to create and only become financially viable when they can be produced and used in large numbers; as a result, maps tend to show those features of the Earth's surface that are of the widest interest, such as coastlines, roads, and topography. For the same reason, they must also be valid for as long as possible, so maps tend to emphasize features that are fixed and constant, a point that is particularly relevant to this chapter. As two-dimensional models, maps cannot show information that is truly three-dimensional, such as the configuration of an underground mine or cave system. Finally, the devices used to make measurements of distance or area from maps—rulers, dividers, planimeters—are inherently imprecise and tedious to use [1].

The widespread adoption of digital technology has revolutionized the process of sharing geographic information during the past four decades. Instead of an analog model, in which the features of the Earth's surface were (imperfectly) proportioned, a digital computer stores information in coded form, allowing it to be copied, manipulated, transmitted, and edited at electronic speed. But the earliest computers were designed to solve problems in numerical analysis and cryptography. It was not obvious that one could store the contents of a map in a computer, or that doing so would have any valid purpose.

In the mid-1960s, a number of visionaries working under contract to the Government of Canada created what became known as the Canada Geographic Information System (CGIS) [2]. The primary driver for this project was the Canada Land Inventory, a major effort to assess the actual and potential uses of land in a portion of the Canadian landmass extending north from the U.S. border. Measures of area were essential but could be derived from maps only by a tedious manual process, so groundbreaking processes were devised for converting the content of maps to digital form, calculating area, and reporting the results. While maps were the primary input, there was no intention in the original design to produce them as output, in part because computer-driven plotting devices had not yet been invented.

Over time, the vision of a geographic information system evolved and grew, based primarily on the notion of a GIS as a computerized mapping system—as a container of maps. A number of methods were devised for creating efficient and economical representations of the contents of maps in digital form. The concept of layers—that geographic reality could be separated into a number of distinctly different themes—represented a natural extension of the practice of printing color maps from different plates using distinct inks and became one of the icons of GIS. By the late 1990s, geographic information systems were capable of capturing the important information from maps and performing virtually any conceivable operation on the results, including display, measurement, and advanced forms of analysis and modeling.

However, the map metaphor continued to dominate thinking. Various methods used to represent geographic data in GIS essentially began with the contents of maps. The raster option is analogous to the methods used to create digital images: a map is captured as a rectangular grid of cells and stored as an ordered sequence, typically row by row from the top left. The vector option first identifies all of the features on the map, classifies them as points, lines, or areas, and creates a digital representation using coordinates. Each point is captured as a single set of coordinates (normally two, but possibly three in some applications that involve elevation); each line is captured as a *polyline*, an ordered sequence of points that is assumed connected by straight segments; and each area is captured as a *polygon*, with a similar ordered sequence of points that returns to the first point to close the area. Not surprisingly, a GIS was explained as a digital map container. It inherited many of the conventions and practices of cartography, including the concept of scale or representative fraction. Like map-makers, GIS designers continued to struggle with concepts that were problematic for maps, including time and the third dimension.

One of the unfortunate consequences of this heritage concerns uncertainty. No map can perfectly replicate the real world, since it inevitably generalizes, abstracts, and approximates the complexity of reality. Thus a GIS database created from maps similarly leaves its user uncertain about the real world that the databases supposedly represent. While paper maps are obviously approximate, there is an unfortunate tendency to believe that because a computer works to seven or even fourteen significant digits, its outputs must be as accurate as its precision would suggest. GIS designers have tended to encourage rather than dispel this misconception by largely ignoring the basic scientific principle that results should be reported to a precision that matches their inherent accuracy rather than the internal precision of any calculations. Moreover, cartographers generally lack methods for displaying the uncertainty associated with maps. GIS tends to have inherited this legacy [3].

MAINSTREAM CONCEPTS: OBJECT ORIENTATION

As a concept, information is notoriously difficult to define. In practice, information takes an enormous variety of forms, from printed text to spoken instructions to the equations of mathematics. Nevertheless, any information processing system must make assumptions about the nature of information, and a GIS is no exception. As we have seen, early GIS assumed that the information to be captured and stored in its database would be contained on maps. Early GIS clearly had problems capturing anything that could not be represented using the paper and pen of the cartographer.

Since the 1960s, the computing industry has developed and produced a number of types of *database management* systems. Each system has attempted to provide a general solution to the task of storing data and has similarly made assumptions about the nature of data. By the 1970s, the *relational* database management system had become dominant [4]. Its basic assumption is that all information can be expressed in the form of tables, with common keys providing the links between tables. This model proved ideal for a wide range of applications, from airline reservations to banking, and the database management industry flourished accordingly. Moreover, several visionaries in the GIS industry recognized that this model could be usefully applied to the contents of maps. Figure 1 shows three maps, one of county boundaries, one of city streets, and one of soil classes. In each case, the map can be regarded as a collection of *nodes* (the points where lines come together or end), *arcs* (the lines connecting nodes and separating areas), and *faces* (the areas bounded by the arcs). Moreover, each arc is connected to two nodes, and each arc bounds two faces (for arcs representing dead-ends in a street network the two two faces will be the same, and one node will have no connections to other arcs). So these maps could be represented by three tables, of nodes, arcs, and faces. The common keys of the relational model could keep track of the relationships between the elements.

[Figure 1 about here]

One problem remained, however, in applying the relational model to map content. The arcs that connect nodes are generally curved and must be represented as polylines with multiple segments. But the number of segments per arc varies, whereas the tabular structure of the relational model requires a constant number of entries per row. If polylines were stored in a table, the number of columns would have to be sufficient to accommodate the polyline with the largest number of segments, and a large proportion of the table would be empty. Instead, GIS designers adopted what became known as the *hybrid* approach, in which polylines and their coordinates were stored outside the relational model. While awkward, this solution survived in the leading GIS products from 1980, when the relational model was first introduced, well into the 1990s. It meant that only part of the database could be stored in relational tables and required GIS developers to maintain in effect two databases, one for the tabular information on arcs, nodes, and faces, and one for polylines and their coordinates.

The solution preserved the map metaphor. Just as certain types of information could not be displayed on maps, so the same types of information were problematic for GIS. Many authors have commented on the inability of 1990s GIS to store information on change through time [5,6], or information about three-dimensional structures [7], or information about hierarchical relationships in geographic information.

This situation changed dramatically in the late 1990s after a new data model emerged to address many of the deficiencies of the relational approach. The *object-oriented* model begins with the assumption that any application is based on well-defined objects that can be organized into classes—everything in the world is an instance of a class. Thus geography consists of mountains, cities, buildings, trees, and so forth. Each class has certain attributes: mountains would have heights, names, and locations; trees would have heights, girths, and ages. In the object-oriented model, classes are organized into hierarchies through the concept of *inheritance*. For example, the class *male human* could be represented as a specialization of the more-general class *human*; every male human has all of the attributes associated with humans, plus specific attributes associated only with male humans. *Human* could be represented as a specialization of the class *primate*, which could itself be a specialization of the class *mammal*, and so forth.

This concept of inheritance turns out to be very useful in the design of GIS databases. A county, for example, could be modeled as a specialization of the more-general class *administrative unit*, which could in turn inherit all of the general properties of the class *polygon*, one of the fundamental concepts of any GIS and one that is defined and made operational by the GIS designer. An object-oriented design for a GIS database thus resembles a tree. At the top are the classes defined by the GIS designer and valid for all applications: the polygons, polylines, points, and other classes for which the designer has implemented various procedures for capturing, editing, displaying, storing, and analyzing each respective data type. Below in the tree are increasingly specialized classes relevant to the application. For example, Figure 2 shows a design that might be appropriate for a

GIS application in military history. Points are specialized to *engagement*, *camp*, *headquarters*, and polylines are specialized to *march* and *front*. In addition, polygons could be specialized to *country* and *terrain type* though these are not shown in the figure.

[Figure 2 about here]

The object-oriented model represents a substantial advance on earlier thinking. In addition, the two decades that had elapsed since the development of the earlier hybrid approach based on the relational model had seen massive improvement in processing speed and the costs of data storage. Developers of database management systems had vastly improved the flexibility of their products, particularly what could be stored in the cells of tables. While earlier versions had allowed only single numbers or fixed-length strings of characters to be stored, by the late 1990s it had become possible to store a wide range of information types, including images and variable-length strings. Today, it is straightforward to store an entire polyline or polygon in a cell, obviating the need for the separate storage system of the hybrid approach.

The earlier model had stored the relationships between the arcs, nodes, and faces of Figure 1, in part to enable faster access to such properties as adjacency and connectivity. With the increased computing power of the late 1990s, however, it became easy to compute such properties as and when needed. These so-called *topological* properties could now be omitted from databases, leading to much simpler designs. The county boundary map shown in Figure 1A, for example, could be represented simply as a single class of polygons, with each polygon’s coordinates stored as a single attribute termed *shape*, rather than the much more cumbersome hybrid approach of three tables (nodes, arcs, and faces) and a separate database of polyline coordinates.

All of these advances succeeded in moving substantially away from the map metaphor, allowing GIS databases to store things that were fundamentally not mappable. Classes could be time-dependent, with attributes and locations that changed through time. Areas could overlap, and lines could cross without forming nodes. The only requirement was that the geographic world be conceived as populated by discrete, identifiable, and countable objects in an otherwise empty space. This so-called *discrete-object* model coincides substantially with the way humans think about the geographic world, but it conflicts fundamentally with the approach often taken in science, in which the world is characterized by continuously varying *fields* of such properties as elevation, population density, or rainfall. Moreover, concepts of continuous variation may be the only way of dealing with uncertainty about the locations of objects or events, or about the locations of object boundaries (see [8] for a discussion of GIS objects with indeterminate boundaries). The GIS community is still struggling with how to achieve efficient representation of such continuous phenomena within the object-oriented approach that now dominates the industry.

Two more concepts of the object-oriented approach have particular value in GIS. Besides the generalization and specialization associated with inheritance, the object-oriented model allows classes to be related hierarchically through *aggregation* relationships, in which the instances of one class are aggregations of two or more component classes. Such hierarchical relationships are impossible in the relational model but are commonly found in geographic reality. For example, a military *campaign* might be modeled as an aggregation of the classes *engagement*, *troop movement*, *camp*, etc. A geographic region might be modeled as an aggregation of its components *city*, *village*, *road*, etc. These relationships allow the properties of aggregates to be related to the properties of the components through simple rules such as addition, counting, or averaging.

Second, the object-oriented approach recognizes *associations* between classes, much as the relational model supported the notion of common keys. These might be used to represent associations between cities and the nations that contain them, or between lakes and the rivers that flow into them, or between roads and the intersections where they join. But the object-oriented approach goes one step further than the relational model in this area. In addition to recording the existence of relationships between objects, it is possible to create *association classes* and to use them to give attributes to relationships. Thus two cities might be linked with data on the historic flow of migrants between them; or the number of newspaper stories that originated in one city and were published in the other; or the time taken to travel between them in historic periods.

Figure 3 shows the full set of these relationships and the symbols commonly used to represent them in graphic depictions of object-oriented database designs. The examples later in this chapter illustrate each of these types of relationships.

[Figure 3 about here]

In summary, the object-oriented model provides a much more flexible and general approach to the representation of geographic information which is much less limited than its predecessors by the map metaphor. In this new approach, it is easy to represent time-specific events, objects that move or change properties through time, and other aspects of geography that were similarly difficult to represent on maps. A single database has replaced the cumbersome hybrid design. Numerous tools are now available to assist in the design process, including so-called computer-assisted software engineering (CASE) tools. It is common to create designs for databases using the standard conventions of Unified Modeling Language (UML) [9] and graphics packages such as Microsoft's Visio. Modern GIS includes tools to bring such designs into the GIS and to create the necessary tables and relationships automatically, so that they are ready to accept data [10]. Object-oriented designs allow for inheritance, aggregation, and association classes, none of which were supported by previous data models. In essence, GIS has moved from a technology heavily dominated by the map metaphor to one that provides a comprehensive, efficient, and flexible approach to the representation of phenomena in space and time.

MOVING OBJECTS: LIFELINES AND TRACKS

One of the commonest examples of moving objects is the tracks formed by humans as they move through their daily activities, as they travel to work and on vacations, and as they relocate during a lifetime. At the daily scale, such tracks are of great use to transportation planners as they struggle to understand the patterns of trips made by individuals and their vehicles and to plan better for future demands on the transportation system. Figure 4 shows an example, the result of tracking a sample of individuals in Portland, Oregon during a short period. At the scale of years or an entire lifetime, such tracks or *lifelines* can be valuable contributions to the historical record, showing as they do the meetings that occur between individuals, the routes followed by explorers, and patterns of travel and migration from one place of residence to another. They can also be invaluable records of the exposure of individuals to environmental contaminants, particularly in relation to diseases with long latencies, such as cancer. Hägerstrand [11] formalized many of the basic concepts of such tracks, and there have been several recent efforts to extend his framework. Today, large volumes of tracking data are becoming available as a result of the widespread use of the Global Positioning System, and geographers and others are developing interesting ways of analyzing and modeling such data.

[Figure 4 about here]

Figure 5 shows an object-oriented database design for storing tracking data. The fundamental element of a track is the observation of an individual's position at some point in time. GPS data might produce such points as often as every second, whereas tracks compiled from diaries, such as those of the Lewis and Clark expedition, might record locations only a few times per day. A *track* is composed of many observations of position and the name of the individual who made the track, which is stored along with other attributes of the individual in the *individual* class. A track also is a type of polyline and also inherits from this basic GIS class. When a *meeting* occurs between individuals, their tracks coincide for a period of time equal to the duration of the meeting, and the members of the group are aggregated by the *meeting* class. Such meetings might be observed and recorded directly or inferred from the spatial and temporal coincidence of tracks.

[Figure 5 about here]

In his analysis of the visual display of information, Tufte [12] identifies the map made by Minard of Napoleon’s Moscow campaign, reproduced in Figure 6, as a particularly effective representation of the events and conditions of the campaign. The map shows the route taken by the army, major events along the route, the size of the army, and the temperature during the return from Moscow, all factors important in understanding this military and humanitarian catastrophe. Alan Glennon and I used this as a working example to develop a general data model for capturing information on flow-like phenomena—a *use case* in the terminology of database design. In principle, we looked for a design in the form of classes, attributes, and relationships, which would succeed in capturing the objects shown on the map and their characteristics and allow the map to be regenerated from the database using the cartographic rendering tools of GIS. The test of such a design is satisfied when some location exists within the database to store every fact portrayed on the map and when the full content of the map can be regenerated from the database. The cartographic styles used by Minard are not generally available in today's GIS, so the regenerated map will appear somewhat different and lack the grace of the historic styles.

[Figure 6 about here]

Figure 7 shows our design, which was informed by several additional examples or use cases of flow-like phenomena, and has been simplified to accommodate only the major features of the Minard map. Polylines provide the underlying geometry. Reaches are sections of polylines with consistent patterns of attributes, starting and ending at defined distances along the polyline. The volume of flow on a reach can vary along the reach, defined by a flow function. Polylines start and end at nodes, which are identified with placenames. A full representation of all of the content of the map would require a significantly more elaborate design, but this one serves to illustrate the basic principles involved.

[Figure 7 about here]

A generic design such as this can provide the basis for storing, analyzing, and displaying numerous examples of flow-like phenomena. In addition to the Minard map, we used use cases based on historic migration between U.S. states, and underground flows in a region of Central Kentucky, to develop the design. But now that it is established, the same basic design can serve to store geographic information on flow-like applications as diverse as the Lewis and Clark expedition, the movement of armies prior to the Battle of Waterloo, or the invasions of Italy during the collapse of the Roman Empire. All of these examples are difficult to display in map form because of their temporal complexity, but a generic object-oriented design can provide the basis for storing all of the information relevant to their study, and for extracting, studying, and visualizing different aspects. The necessary tools to implement this design as an extension to ESRI's ArcGIS are available for download at http://dynamicgeography.ou.edu/flow/index.html.

CHANGING BOUNDARIES

Agencies such as the US Bureau of the Census provide abundant data on the socioeconomic and demographic characteristics of the population, and similar agencies perform the same function in other countries. To maintain confidentiality, results are necessarily aggregated to standard units, such as counties, ZIP codes, or census tracts. Unfortunately, each of these standard units has a tendency to shift through time, making it notoriously difficult to obtain longitudinal perspectives on social patterns. In some historic cases, it may even be impossible to determine the exact boundaries of certain reporting zones. For example, the zone known as Portage County, Wisconsin entirely changed its location between two census years, the same name being assigned to two completely disjoint areas in two succeeding enumerations. In other cases, reporting zones have been split or merged, and boundaries have been shifted. Historians interested in geographic patterns of Chinese society face even more daunting problems, since there may be no maps whatever to define the boundaries of a certain named reporting zone in some past period.

Several attempts have been made to overcome these problems by making census data available for historically accurate sets of geographic units. Typically, such projects begin with the most recent and accurate census boundaries, then work backwards through time, adjusting boundaries to accord with their historical location for each census year. The largest and likely most successful of these for the United States is the National Historical Geographic Information System (NHGIS), funded by the National Science Foundation and housed at the University of Minnesota (www.nhgis.org). It aims to provide high-quality digital boundaries of census units for each enumeration decade, linked to aggregate population data for each census unit, from 1790 to 2000. The Atlas of Historical County Boundaries Project at the Newberry Library (www.newberry.org/ahcbp) records boundary changes from the colonial period to the present, enabling historically accurate mapping of census and other data at the county level for any point in time.

Early GIS database designs created great difficulty for this type of application, because the map metaphor implied that any layer would be a single snapshot, making it impossible to organize multiple snapshots into a coherent database. A common workaround employed a concept often termed the *region*, consisting of an aggregation of parts of a reporting zone, assembled to match the definition of the zone at a specific time. Consider, for example, all of the maps of census reporting zones since the first census year. Superimposing all of these maps creates a single map. Each face in this composite map represents an area that has never been divided by the reporting zones of any year. This map will contain many thousands of faces. Moreover, it will be impacted by a problem that is universally encountered in GIS. Each of the maps will have been digitized independently, such that if two maps contain the same boundary, it will appear slightly different because of random errors and distortions introduced by the digitizing process. Thus when the two versions of the boundary are overlaid, a large number of small slivers will be produced. The complete overlay of all maps will contain a vast number of such slivers. GIS software commonly attempts to remove such slivers based on a user-defined tolerance, but this solution is in many ways imperfect.

Having prepared the composite, it is possible to recreate each census year's boundary map by dissolving boundaries between faces in the composite that were part of the same reporting zone, or *region*. This is achieved by preparing a table indicating which faces belonged to which reporting zones in each year.

Modern GIS designs solve the problem in a much simpler way and are able to do so because of the enormous increases in computing power and storage capacity of the past decade. Figure 8 presents a simple design, in which the names or other identifying attributes of zones are linked to their polygonal shapes through associations. Any change in the boundary of a county can be recorded by storing both versions; and a change in name without change in boundary can be recorded by storing both names. Problems of slivers can be addressed using the powerful tools now available in modern GIS. Again, such designs are fundamentally incompatible with the map metaphor and enabled because of the adoption of object-oriented approaches.

[Figure 8 about here]

ASSESSMENT AND CONCLUSION

Early GIS was dominated by the problem of creating digital representations of the contents of maps, a valuable exercise as we saw in the case of CGIS and one that sustained the rapid growth of GIS applications that occurred prior to the 1990s. But the inability to deal with dynamics and other inadequacies of the early approach, including the need to create two databases—one to store the nodes, arcs, and faces of networks, and the other to store coordinates—eventually led to a dissatisfaction with the relational database approach. Since the 1990s, the field has seen enthusiastic adoption of the object-oriented model, because of its ability to deal with events in time and other aspects of geographic data that are incompatible with cartographic practice and with the early data models. This process has relied on massive increases in computing power and storage capacity, largely removing the constraints that in part necessitated the earlier approach. Today, GIS databases can readily deal with objects that move, objects that change shape, and other phenomena that cannot be portrayed on paper maps.

As with other innovations, this one is being adopted slowly in a community and industry that have grown used to earlier approaches, and are committed to them in many ways.

Thus a historian approaching GIS is still tempted to think of this as a technology of maps and to assume that one cost of adopting GIS will be an inability to deal with time—that GIS is the toolkit of geographers, who look at the world as essentially static, and focus on understanding cross-sectional rather than longitudinal variation. In reality, this view is by now no longer valid—GIS has been dealing successfully with space and time for many years. But other factors serve to preserve the legacy of a space-only approach. We still lack good tools for the analysis of dynamic data in GIS, because such data have only recently appeared. While researchers have made much progress in developing such tools, it will be some time before they are fully deployed. Demand drives the GIS software industry,and static applications still dominate the community, so tools inevitably will first appear among the research community and will be implemented only slowly in commercial products. Dynamic, three-dimensional data is also still comparatively scarce, particularly for historic periods, despite the efforts of such projects as NHGIS.

These comments lead to some fairly confident predictions about the future of GIS, at least as far as historians are concerned. First, tools for analysis of dynamic data will appear at a steadily increasing rate in the coming years. These will include methods for visualization, for analysis of real data against simple models and hypotheses, and for such basic operations as clustering similar tracks and preparing longitudinal series. The latter will require improvements in current methods of areal interpolation [13], which has become the key technique for estimating attributes of hypothetical reporting zones. Specialized data models for dynamic data will also begin to appear, and there is every reason to believe that a data model designed specifically for the needs of historians will be added to the list of existing specialized GIS data models [10]. In short, the transition to object-oriented data modeling in GIS has solved some basic problems, allowing a rapid expansion of interest in the use of GIS to improve our understanding of historical and other time-dependent phenomena. After many years, we are finally able to examine data in full spatio-temporal perspective.

ENDNOTES

1. D.H. Maling, *Measurements from Maps: Principles and Methods of Cartometry* (New York: Pergamon, 1989).

2. T.W. Foresman, editor, *The History of Geographic Information Systems: Perspectives from the Pioneers* (Upper Saddle River, NJ: Prentice Hall PTR, 1998).

3. J.-X. Zhang and M.F. Goodchild, *Uncertainty in Geographical Information* (New York: Taylor and Francis, 2001).

4. C.J. Date, *An Introduction to Database Systems* (Reading, MA: Addison-Wesley, 2000).

5. G. Langran, *Time in Geographic Information Systems* (London: Taylor and Francis, 1993).

6. D.J. Peuquet, *Representations of Space and Time* (New York: Guilford, 2002).

7. J. Raper, *Multidimensional Geographic Information Science* (New York: Taylor and Francis, 2000).

8. P.A. Burrough and A.U. Frank, editors, *Geographic Objects with Indeterminate Boundaries* (London: Taylor and Francis, 1996).

9. G. Booch, J. Rumbaugh, and I. Jacobson, *The Unified Modeling Language User Guide* (Upper Saddle River, NJ: Addison-Wesley, 2005).

10. D. Arctur and M. Zeiler, *Designing Geodatabases: Case Studies in GIS Data Modeling* (Redlands, CA: ESRI Press, 2004).

11. T. Hägerstrand, “What about people in regional science?” *Papers of the Regional Science Association* 24 (1970): 7.

12. E.R. Tufte, *The Visual Display of Quantitative Information* (Cheshire, CT: Graphics Press, 1986).

13. M.F. Goodchild, L. Anselin, and U. Deichmann, “A framework for the areal interpolation of socioeconomic data,” *Environment and Planning A* 25 (1993): 383.

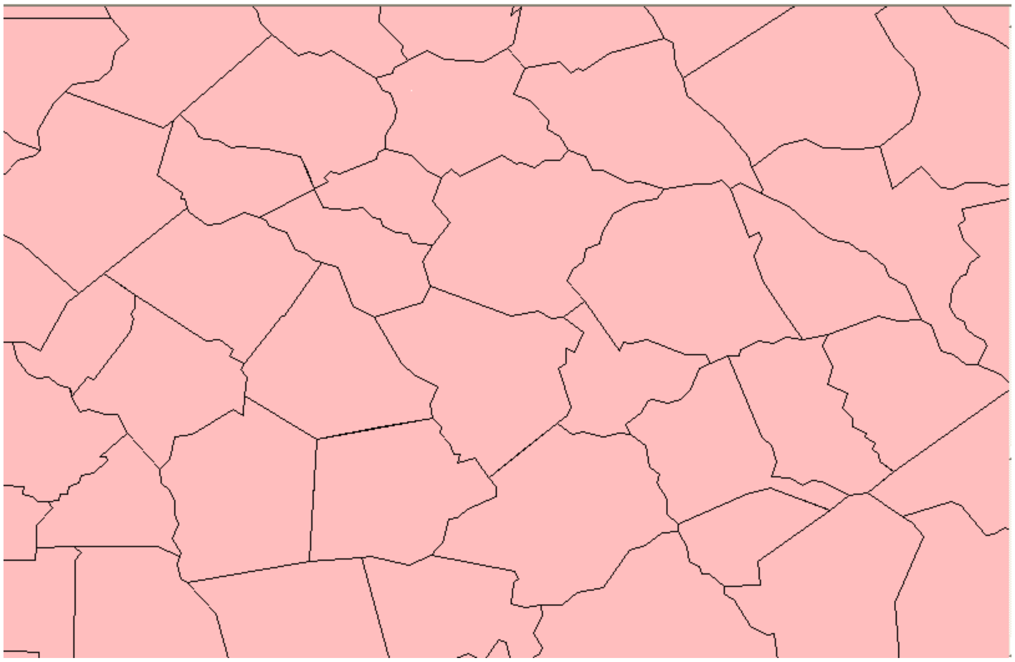
14. M.-P. Kwan and J. Lee, “Geo-visualization of human activity patterns using 3-D GIS: a time-geographic approach,” in M.F. Goodchild and D.G. Janelle, editors, *Spatially Integrated Social Science* (New York: Oxford University Press, 2004), 48.

FIGURE CAPTIONS

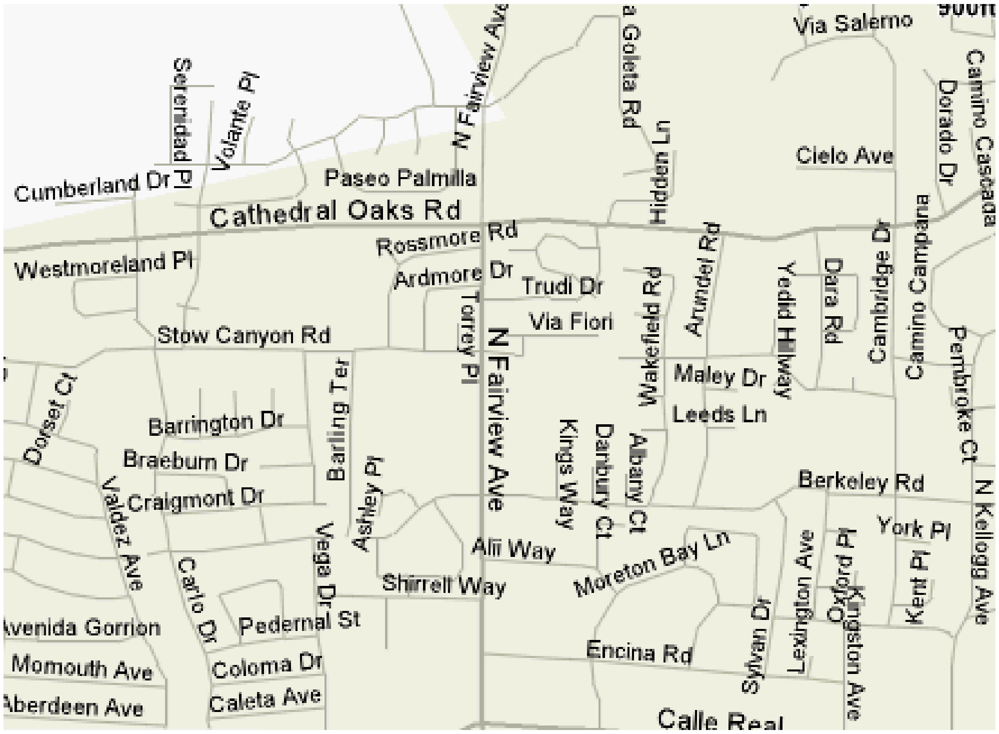
1. Three types of data readily accommodated in early hybrid GIS databases: (A) a county boundary map, (B) a street map, and (C) a soil map.
2. A simple design for a database representing military campaigns, showing the concept of specialization. The point and polyline classes are shown in italics, indicating that they are provided by the GIS, and will have no instances of their own in this application. The design could be simplified further by identification of Side and Leader classes associated with Front, March, Camp, and Engagement classes. Since these classes would not have locations, they would have no inheritance from the GIS classes.
3. The graphic symbols used to depict inheritance, aggregation, and association relationships between classes in UML. Each county is a kind of polygon; each person lives in a county; and each community is an aggregation of people.
4. A visualization of tracks of individuals in Portland, OR. The horizontal axes denote position, and the vertical axis time. Source: [14].
5. A simple design for storing tracking data, showing person, location, track, and meeting classes. Persons and meetings do not have locations, and thus do not inherit from GIS classes.
6. A reproduction of Minard’s original map of Napoleon’s Moscow campaign.
7. A simplified design for a database containing the primary contents of the Minard map (see text).

8. A simple design for a longitudinal database of census reporting zones. Each name may be associated with more than one boundary, and boundaries may exist for varying lengths of time.

9. Figure 1 (A)



(B)



(C)

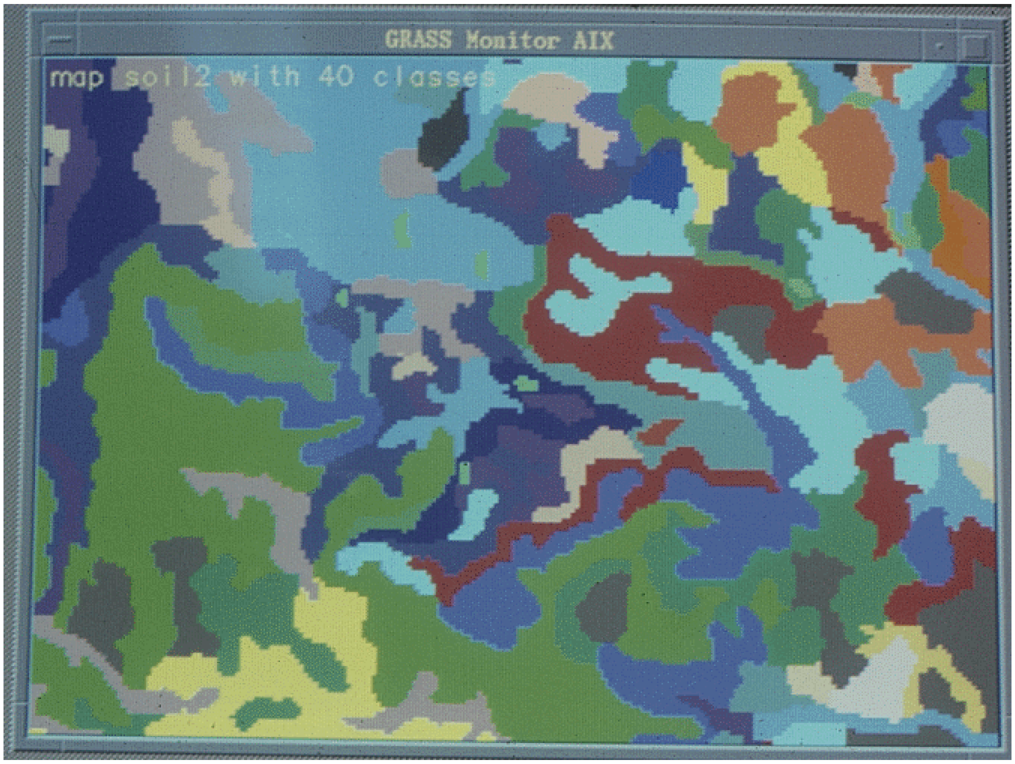

GRASS Monitor AIX
map soil2 with 40 classes

Figure 2:

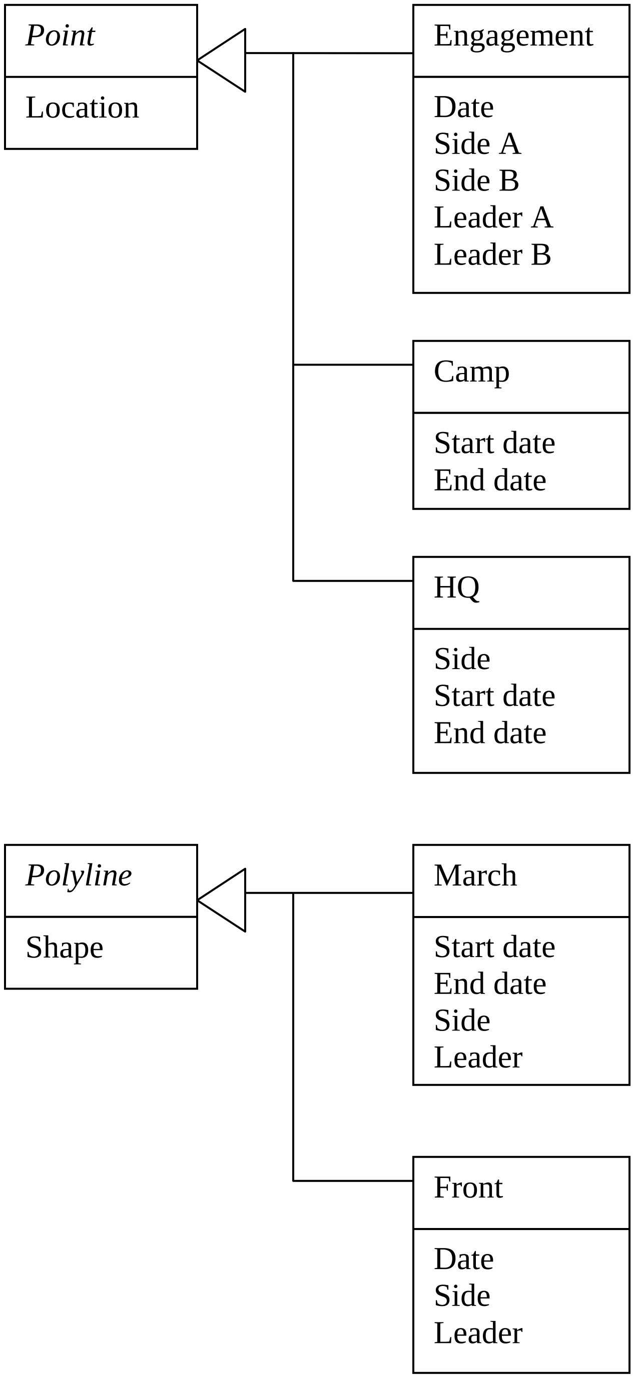

Point
Location
Engagement
Date
Side A
Side B
Leader A
Leader B
Camp
Start date
End date
HQ
Side
Start date
End date
Polyline
Shape
March
Start date
End date
Side
Leader
Front
Date
Side
Leader

Figure 3:

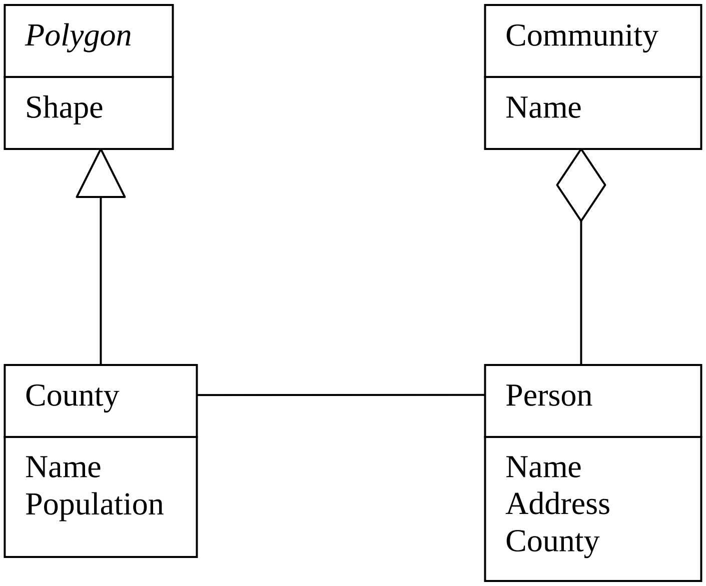

Polygon
Shape
Community
Name
County
Name
Population
Person
Name
Address
County

Figure 4:

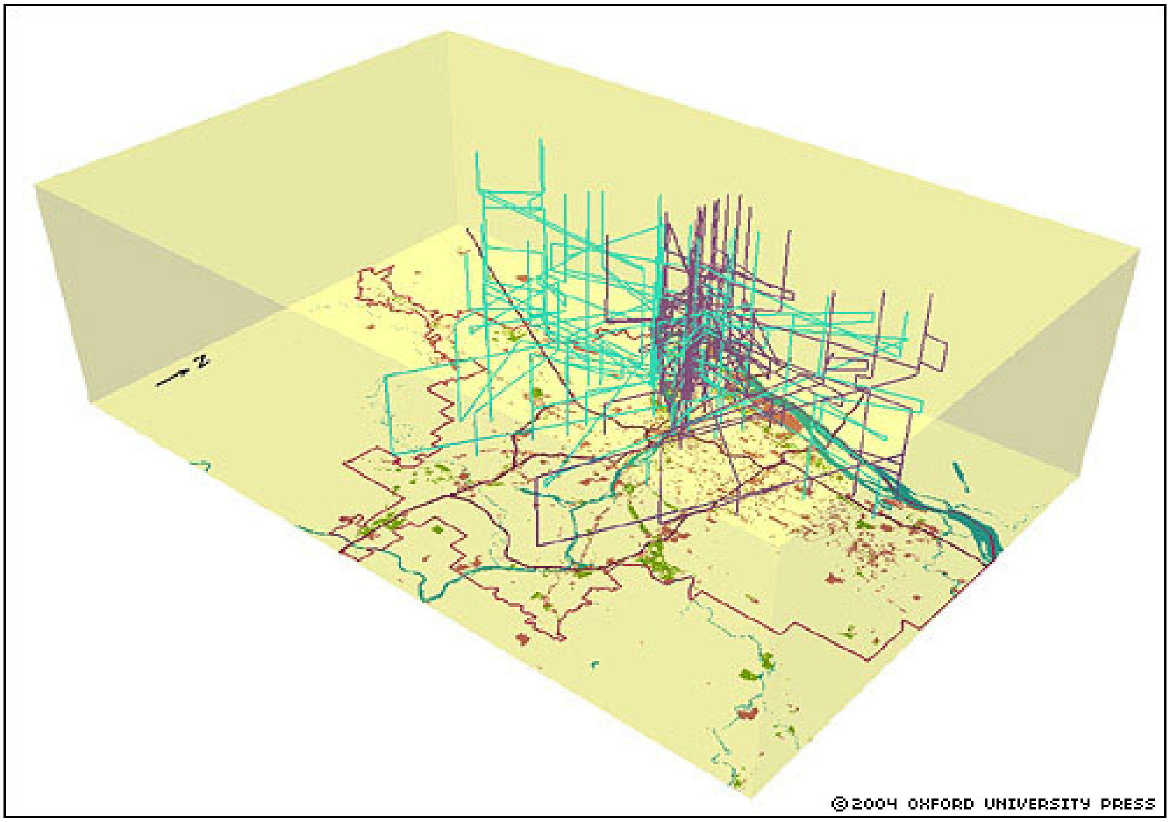


Figure 5:

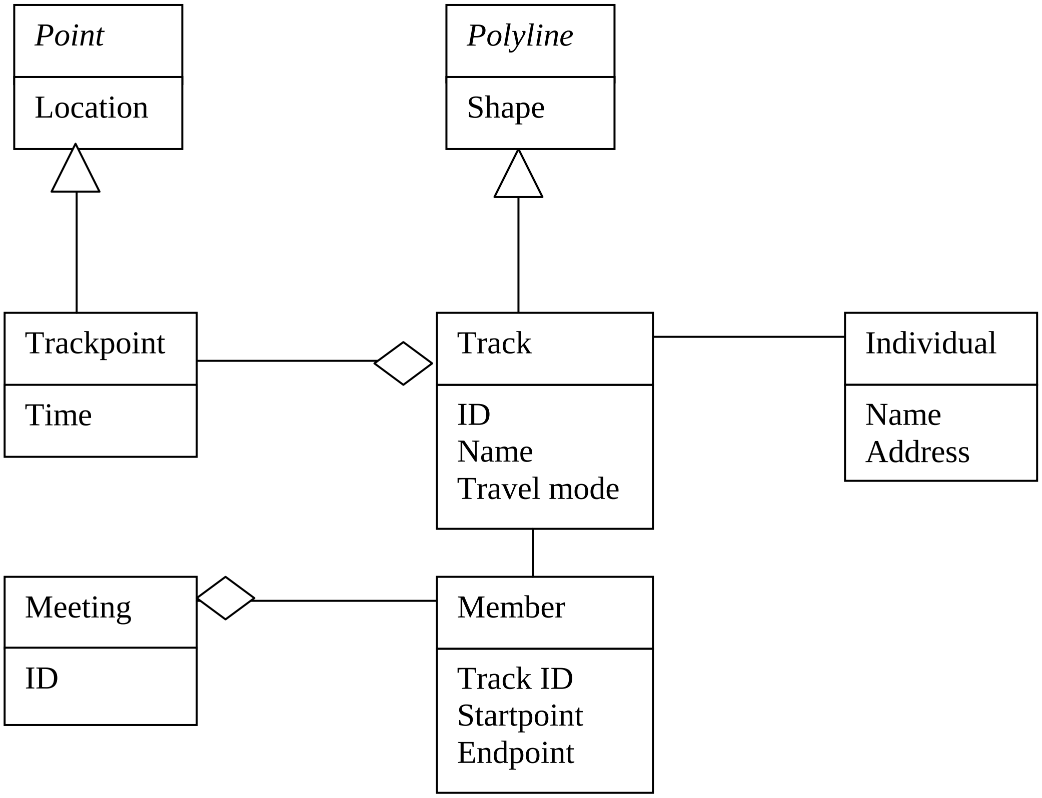

Point
Location
Polyline
Shape
Trackpoint
Time
Track
ID
Name
Travel mode
Individual
Name
Address
Meeting
ID
Member
Track ID
Startpoint
Endpoint

Figure 6:

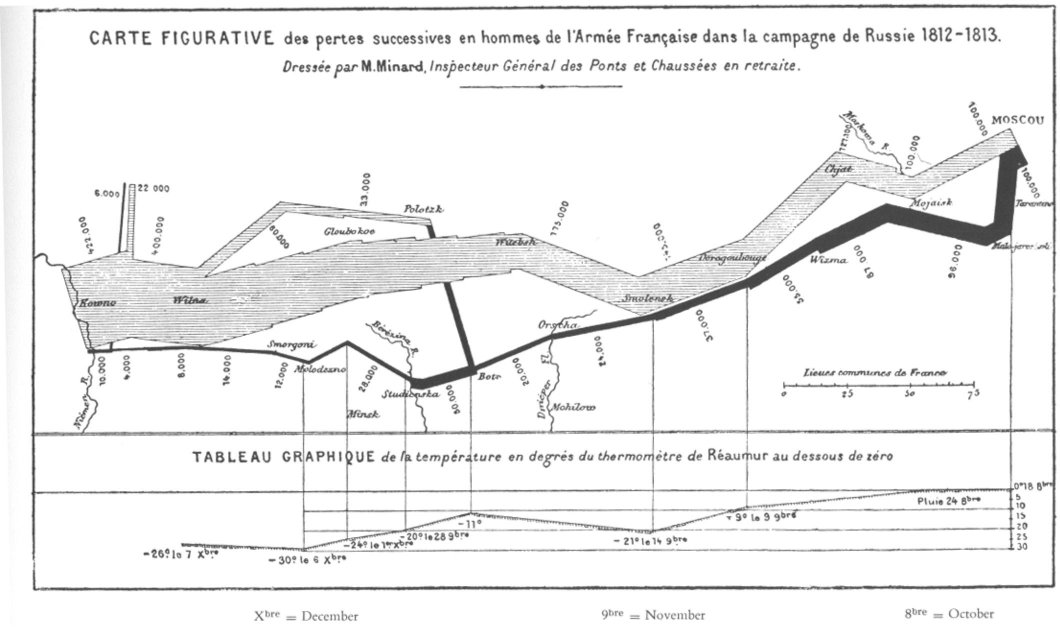

CARTE FIGURATIVE des pertes successives en hommes de l'Armée Française dans la campagne de Russie 1812–1813.
Dressée par M. Minard, Inspecteur Général des Ponts et Chaussées en retraite.
MOSCOU
Kowno
Wilna
Polotzk
Gloubokoe
Witebsk
Smolensk
Dorogobouge
Wizma
Chjat
Mojaisk
Tarantino
Malo-jarosewli
Orscha
Bobr
Studienska
Minsk
Molodezno
Smorgoni
Mohilow
Moskowa R.
Niemen R.
Bérézina R.
Dnieper R.
422.000
400.000
6.000
22 000
33.000
60.000
175.000
145.000
127.100
100.000
100.000
100.000
96.000
87.000
55.000
37.000
24.000
20.000
50.000
28.000
12.000
14.000
8.000
4.000
10.000
Lieues communes de France
0
25
50
75
TABLEAU GRAPHIQUE de la température en degrés du thermomètre de Réaumur au dessous de zéro
0° le 18 8bre
Pluie 24 8bre
– 9° le 9 9bre
– 21° le 14 9bre
– 11°
– 20° le 28 9bre
– 24° le 1er Xbre
– 30° le 6 Xbre
– 26° le 7 Xbre
5
10
15
20
25
30
Xbre = December
9bre = November
8bre = October

Figure 7:

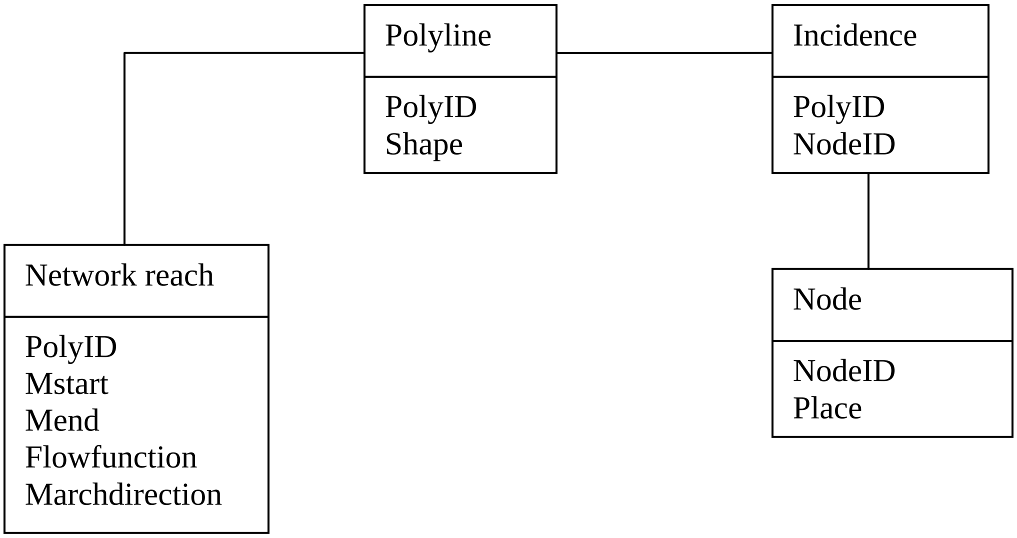
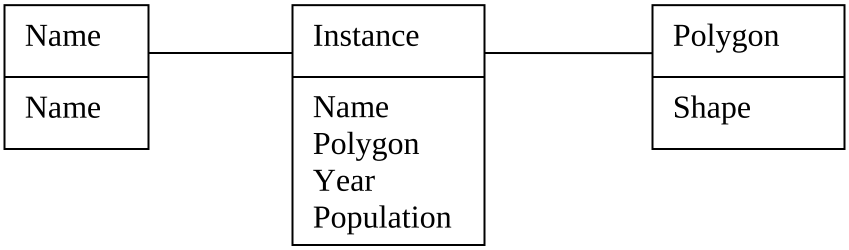

Polyline
PolyID
Shape
Incidence
PolyID
NodeID
Network reach
PolyID
Mstart
Mend
Flowfunction
Marchdirection
Node
NodeID
Place

Figure 8:


Name
Name
Instance
Name
Polygon
Year
Population
Polygon
Shape