The core of GIScience
a process-based approach

ITC Educational textbook series

UNIVERSITY OF TWENTE

FACULTY OF GEO-INFORMATION SCIENCE AND EARTH OBSERVATION

ITC

Chapter 8

Spatial data modelling, collection and management

Rolf de By
Otto Huisman
Richard Knippers
Menno-Jan Kraak
Alfred Stein

8.1 Geographic Information and spatial data types

8.1.1 Geographic phenomena

Defining geographic phenomena

A GIS operates under the assumption that the spatial phenomena involved occur in a two- or three-dimensional Euclidean space. Euclidean space can be informally defined as a model of space in which locations are represented by coordinates—(x, y) in 2D and (x, y, z) in 3D space—and distance and direction can defined with geometric formulas. In 2D, this is known as the Euclidean plane. To represent relevant aspects of real-world phenomena inside a GIS, we first need to define what it is we are referring to. We might define a geographic phenomenon as a manifestation of an entity or process of interest that:

Euclidean space

geographic phenomena

- can be named or described;
- can be georeferenced; and
- can be assigned a time (interval) at which it is/was present.

Relevance of phenomena for the use of a GIS depends entirely on the objectives of the study at hand. For instance, in water management, relevant objects can be river

objectives of the application

basins, agro-ecological units, measurements of actual evapotranspiration, meteorological data, groundwater levels, irrigation levels, water budgets and measurements of total water use. All of these can be named or described, georeferenced and provided with a time interval at which each exists. In multipurpose cadastral administration, the objects of study are different: houses, land parcels, streets of various types, land use forms, sewage canals and other forms of urban infrastructure may all play a role. Again, these can be named or described, georeferenced and assigned a time interval of existence.

Not all relevant information about phenomena has the form of a triplet (description, georeference, time interval). If the georeference is missing, then the object is not positioned in space: an example of this would be a legal document in a cadastral system. It is obviously somewhere, but its position in space is not considered relevant. If the time interval is missing, we might have a phenomenon of interest that exists permanently, i.e. the time interval is infinite. If the description is missing, then we have something that exists in space and time, yet cannot be described. Obviously this last issue limits the usefulness of the information.

Types of geographic phenomena

The definition of geographic phenomena attempted above is necessarily abstract and is, therefore, perhaps somewhat difficult to grasp. The main reason is that geographic phenomena come in different "flavours". Before categorizing such "flavours", there are two further observations to be made.

First, to represent a phenomenon in a GIS requires us to state what it is and where it is. We must provide a description—or at least a name—on the one hand, and a georeference on the other hand. We will ignore temporal issues for the moment and come back to these in Subsection 8.1.4, the reason being that current GISs do not provide much automatic support for time-dependent data. This topic must, therefore, be considered as an example of advanced GIS use. Second, some phenomena are manifest throughout a study area, while others only occur in specific localities. The first type of phenomena we call geographic fields; the second type we call objects.

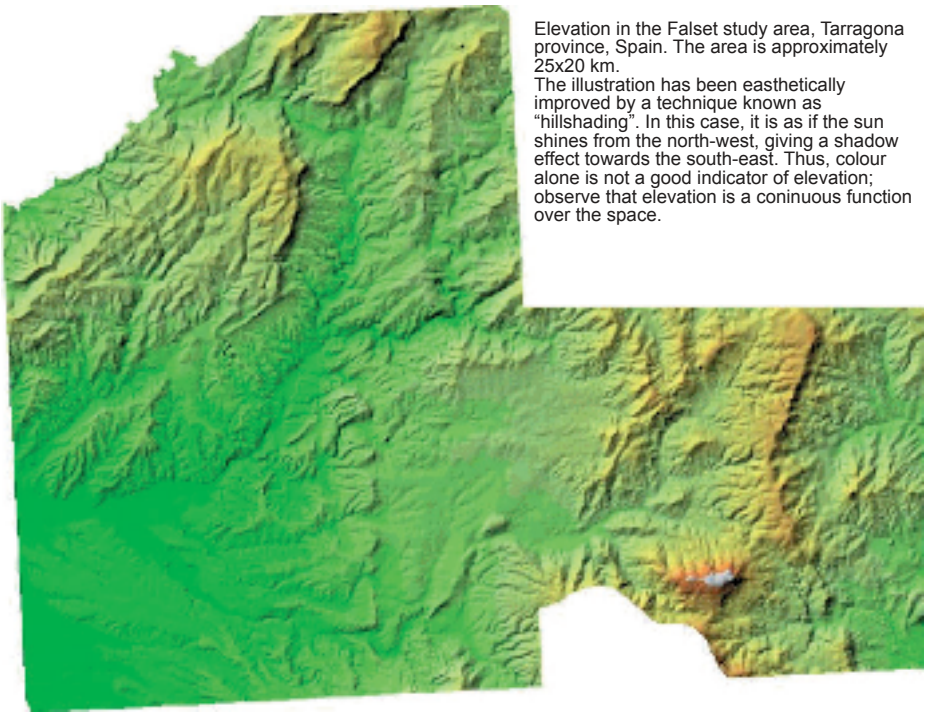
Geographic fields

A geographic field is a geographic phenomenon that has a value "everywhere" in the study area. We can therefore think of a field as a mathematical function f that associates a specific value with any position in the study area. Hence if (x, y) is a position in the study area, then $f(x, y)$ expresses the value of f at location (x, y). Fields can be discrete or continuous. In a continuous field, the underlying function is assumed to be "mathematically smooth", meaning that the field values along any path through the study area do not change abruptly, but only gradually. Good examples of continuous fields are air temperature, barometric pressure, soil salinity and elevation. A continuous field can even be differentiable, meaning that we can determine a measure of change in the field value per unit of distance anywhere and in any direction. For example, if the field is elevation, this measure would be slope, i.e. the change of elevation per metre distance; if the field is soil salinity, it would be salinity gradient, i.e. the change of salinity per metre distance.

discrete and continuous fields

Figure 8.1 illustrates the variation in elevation of a study area in Falset, Spain. A colour scheme has been chosen to depict that variation. This is a typical example of a continuous field. Discrete fields divide the study space in mutually exclusive, bounded parts, with all locations in one part having the same field value. Typical examples are land classifications, for instance, using either geological classes, soil type, land use type, crop type or natural vegetation type. An example of a discrete field—in this case iden-

tifying geological units in the Falset study area—is provided in Figure 8.2. Observe that locations on the boundary between two parts can be assigned the field value of the "left" or "right" part of that boundary.



Elevation in the Falset study area, Tarragona province, Spain. The area is approximately 25x20 km.
The illustration has been easthetically improved by a technique known as "hillshading". In this case, it is as if the sun shines from the north-west, giving a shadow effect towards the south-east. Thus, colour alone is not a good indicator of elevation; observe that elevation is a coninuous function over the space.

Figure 8.1
An example of a continuous field, namely the *elevation* in a study area in Falset, Spain.

Discrete fields are intermediate between continuous fields and geographic objects: discrete fields and objects both use "bounded" features. A discrete field, however, assigns a value to every location in the study area, which is not typically the case for geographic objects. These two types of fields differ in the type of cell values. A discrete field such as land use type will store cell values of the type "integer" and is therefore also called an integer raster. Discrete fields can be easily converted to polygons since it is relatively easy to draw a boundary line around a group of cells with the same value. A continuous raster is also called a "floating point" raster.

A field-based model consists of a finite collection of geographic fields: we may be interested in, for example, elevation, barometric pressure, mean annual rainfall and maximum daily evapotranspiration, and would therefore use four different fields to model the relevant phenomena within our study area.

Data types and values Different kinds of data values may represent spatial "phenomena". Some of these kinds of data limit the types of analyses that can be done on the data:

1. Nominal data values provide a name or identifier to discriminate between different values. Specifically, we cannot do true computations with these values. An example is the names of geological units. This kind of data value is called categorical when the values assigned are sorted according to a set of non-overlapping categories. For example, we might identify the soil type of a given area as belonging to a certain (pre-defined) category.

2. Ordinal data values are numerical data that can be put in a natural sequence but that do not allow any other type of computation. Household income, for instance, could be classified as being either "low", "average" or "high". Clearly this is their natural sequence, but this is all we can say—we can not say that a high income is twice as high as an average income.

3. Interval data values are numerical data that allow simple forms of computation like addition and subtraction. However, interval data have no arithmetic zero value, and do not support multiplication or division. For instance, a temperature of 20 °C is not twice as hot as 10 °C.

4. Ratio data values are numerical data that allow most, if not all, forms of arithmetic computation. Ratio data have a natural zero value and multiplication and division of values are possible operators (distances measured in metres are an example of ratio data). Continuous fields can be expected to have ratio data values, hence we can interpolate them.

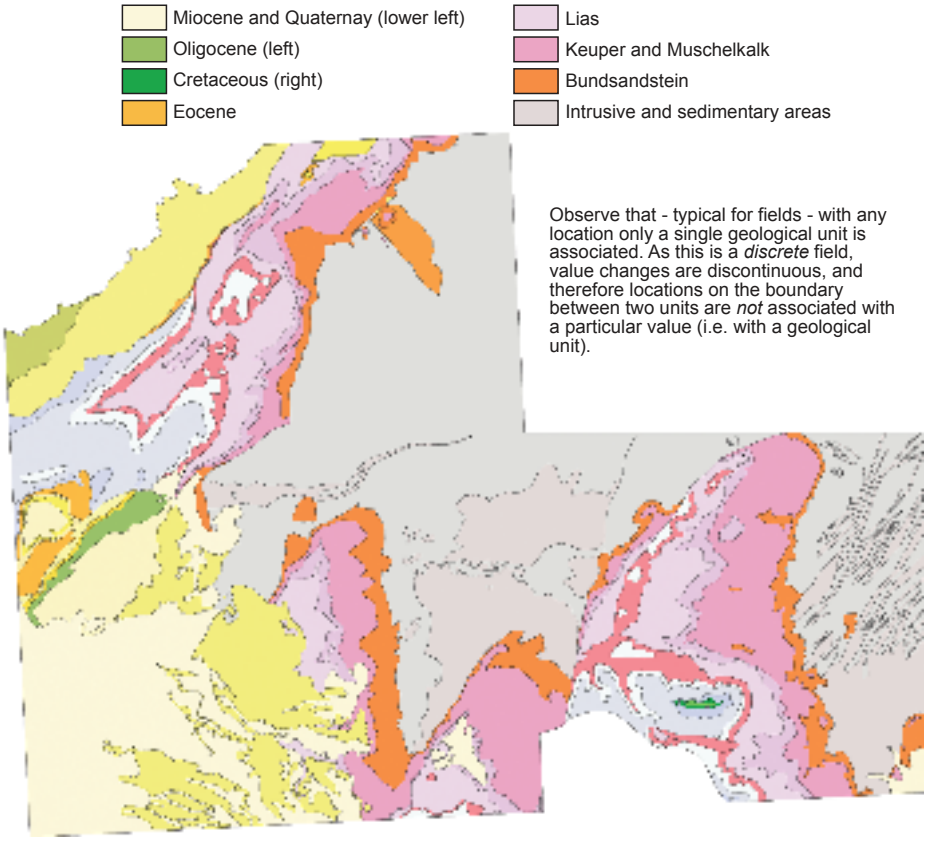


Figure 8.2
A discrete field indicating geological units as used in a foundation-engineering study for constructing buildings. Falset study area as in Figure 8.1.

qualitative and quantitative data

We usually refer to nominal or categorical data values as "qualitative" data because we are limited in terms of the computations we can do on this type of data. Interval and ratio data are known as *quantitative* data as they refer to quantities. Ordinal data, however, do not fit either of these data types. Often, ordinal data refer to a ranking scheme or some kind of hierarchical phenomena. Road networks, for example, are made up of motorways, main roads and residential streets. We might expect roads

classified as motorways to have more lanes and carry more traffic than a residential street.

Geographic objects

When a geographic phenomenon is not present everywhere in the study area, but somehow "sparsely" populates it, we look at it as a collection of geographic objects. Such objects are usually easily distinguished and named, and their position in space is determined by a combination of one or more of the following parameters: location (where is it?), shape (what form does it have?), size (how big is it?) and orientation (in which direction is it facing?). How we want to use the information determines which of these four parameters is required to represent the object. For instance, for geographic objects such as petrol stations all that matters in an in-car navigation system is *where* they are. Thus, in this particular context, location alone is enough, and shape, size and orientation are irrelevant. For roads, however, some notion of location (where does the road begin and end?), shape (how many lanes does it have?), size (how far can one travel on it?) and orientation (in which direction can one travel on it?) seem to be relevant components of information in the same system.

geographic objects

Shape is an important component because one of its factors is dimension. This relates to whether an object is perceived as a point feature or a linear, area or volume feature. In the above example, petrol stations are apparently zero-dimensional, i.e. they are perceived as points in space; roads are one-dimensional, as they are considered to be lines. In another use of road information—for instance, in multi-purpose cadastral systems, in which the precise location of sewers and manhole covers matters—roads might be considered as two-dimensional entities, i.e. as areas.

Figure 8.3 illustrates geological faults in the Falset study area, a typical example of a geographic phenomenon that is made up of objects. Each of the faults has a location, and the fault's shape is represented as a one-dimensional object. The size, which is length in the case of one-dimensional objects, is also indicated. Orientation does not play a role here.

Usually, we do not study geographic objects in isolation, but instead look at collections of objects, which we consider as a unit. These collections may have specific geographic characteristics. Most of the more interesting ones obey specific natural laws. The most common (and obvious) of these characteristics is that different objects do not occupy the same location. This, for instance, holds for the collection of petrol stations in an in-car navigation system, the collection of roads in that system, and the collection of land parcels in a cadastral system. We will see in Section 8.1.2 that this natural law of "mutual non-overlap" has been a guiding principle in the design of computer representations of geographic phenomena.

Collections of geographic objects can also be interesting phenomena at a higher level of aggregation: forest plots form forests, groups of parcels form suburbs, streams, brooks and rivers form a river drainage system, roads form a road network, and SST buoys form an SST sensor network. It is sometimes useful to view geographic phenomena at this more aggregated level and look at characteristics such as coverage, connectedness and capacity. For example:

geographic scale

- Which part of a particular road network is within 5 km of a petrol station (a coverage question)?
- What is the shortest route between two cities via the road network (a connectedness question)?
- How many cars can optimally travel from one city to another in an hour (a

capacity question)?

multi-scale

In this context, multi-scale approaches are sometimes used. Such approaches deal with maintaining, and operating on, multiple representations of the same geographic phenomenon, e.g. a point representation in some cases, and an area representation in others. To support these approaches, the database must store representations of geographic phenomena (spatial features) in a scaleless and seamless manner. Scaleless means that all coordinates are world coordinates, i.e. are given in units that are used to reference features in the real world (using a spatial reference system). From such values, calculations can be easily performed and an appropriate scale can be chosen for visualization. Other spatial relationships between the members of a collection of geographic objects may exist and can be relevant in GIS usage. Many of these fall under the category of topological relationships, discussed in Subsection 8.1.3.

Figure 8.3
A number of geological faults in the Falset (Spain) study area; see Figure 8.1. Faults are indicated as blue lines; the study area, with main geological eras, is indicated by the grey background only as a reference.

Boundaries

Where shape or size of areas matter, the notion of a boundary comes into play. This concerns geographic objects but also the constituents of a discrete geographic field, as clearly demonstrated in Figure 8.2. Location, shape and size are fully determined if we know an area's boundary, and thus the boundary is a good candidate for its representation. This especially applies to areas with naturally crisp boundaries. A crisp boundary is one that can be determined at an almost arbitrary level of precision, dependent only on the data-acquisition technique applied. Fuzzy boundaries contrast with crisp boundaries in that a fuzzy boundary is not a precise line, but is rather, itself an area of transition.

crisp and fuzzy boundaries

As a rule of thumb, crisp boundaries are more common in man-made phenomena, whereas fuzzy boundaries are more common in natural phenomena. In recent years, various research efforts have addressed the issue of explicit treatment of fuzzy bound-

aries, but there is still only limited support in existing GIS software. Typically, the areas identified in a geological classification, like that of Figure 8.2, are vaguely bounded in reality, but applications of this geological information probably do not require high positional accuracy of the boundaries involved. Therefore, an assumption that they are actually crisp boundaries will have little influence on the usefulness of the data

8.1.2 Computer representations of geographic information

Up to this point, we have not considered how geoinformation, such as fields and objects, is represented in a computer. Now that we have discussed the main characteristics of geographic phenomena (above), let us now examine representation in more detail.

Various geographic phenomena have the characteristics of continuous functions in space. Elevation, for instance, can be measured at many locations and each location may give a different value. To represent such a phenomenon in computer memories, we could either:

- try to store as many (location, elevation) observation pairs as possible, or
- try to find a symbolic representation of the elevation field function as a formula in terms of x and y—like $(3.0678x^2 + 20.08x - 7.34y)$ or some such—that can be evaluated to give us the elevation at any given $(x,\ y)$ location.

Both approaches have their drawbacks. A drawback of the first approach is that it is impossible to store all elevation values for all locations since there are infinitely many pairs. A drawback of the second approach is that it is impossible to know the shape of this function, and it would be extremely difficult to derive such a function. In GISs, usually a combination of both approaches is taken. We store a finite, but intelligently chosen set of (sample) locations together with their elevations. This gives us the elevations at the locations stored. An interpolation function allows us to infer an acceptable elevation value for locations that are not stored.

spatial autocorrelation
Tobler's first law of geography

Interpolation is made possible by a principle called spatial autocorrelation. This is a fundamental principle based on Tobler's first law of geography, which states that locations that are closer together are more likely to have similar values than locations that are farther apart. An example is sea-surface temperature, for which one might expect a high degree of correlation between measurements taken close together. In the case of elevations, a simplistic interpolation function takes the elevation value of the nearest stored location and assigns this to the location that is not stored. Smarter interpolation functions, involving more than a single stored value, should be preferred.

Line objects, either by themselves or in their role as region object boundaries, are continuous phenomena that must be finitely represented. In real life, these objects are usually not straight, and can be erratically curved. A famous paradoxical question is whether one can actually measure the length of Great Britain's coastline, i.e. can one measure around rocks, pebbles or even grains of sand? In a computer, such random, curvilinear features can never be fully represented: they require a degree of generalization. Phenomena with intrinsic continuous and/or infinite characteristics therefore have to be represented with finite means (computer memory) for computer manipulation, yet any finite representation scheme is open to errors of interpretation. To allow for this, fields are usually implemented with a tessellation approach, and objects with a (topological) vector approach. In the following subsections we discuss tessellations and vector-based representations and how these are applied to represent geographic fields and objects.

Regular tessellations

tessellation

Tessellation (or tiling) is a partitioning of space into mutually exclusive cells that together make up the complete study space. For each cell, a (thematic) value is assigned to characterize that part of space. Three types of regular tessellation are illustrated in Figure 8.4. In a regular tessellation, the cells have the same shape and size; a simple example of this is a rectangular raster of unit squares, represented in a computer in the 2D case as an array of $n \times m$ elements (see Figure 8.4 left).

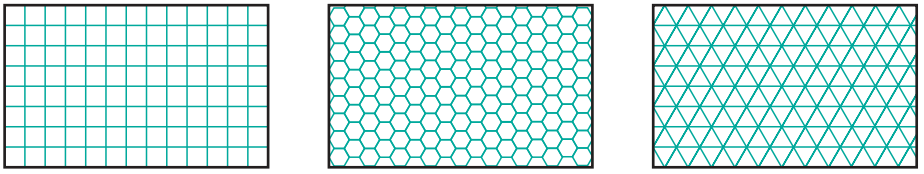


Figure 8.4
The three most common types of regular tessellation: from left to right, square cells, hexagonal cells and triangular cells.

All regular tessellations have in common that the cells have the same shape and size, and that the field attribute value assigned to a cell is associated with the entire area occupied by the cell. Square cell tessellation is commonly used, mainly because georeferencing of such a cell is straightforward. This type of tessellation is known under various names in different GIS packages: e.g. "raster" or "raster map".

grid

The size of the area that a single raster cell represents is called the raster's resolution. Sometimes, the term "grid" is used, but strictly speaking a grid is an equally spaced collection of points, all of which have some attribute value assigned. Grids are often used for discrete measurements that occur at regular intervals. Grid points are then considered synonymous to raster cells.

There are some issues related to cell-based partitioning of the study space. The field value of a cell can be interpreted as one for the complete tessellation cell, in which case the field is discrete, not continuous or even differentiable. Some convention is needed to state which value prevails on cell boundaries. With square cells, this convention states that lower and left boundaries belong to the cell. There are two approaches to refining the solution of this continuity issue: make the cell size smaller, so as to make the "continuity gaps" between the cells smaller; and/or assume that a cell value only represents elevation for one specific location in the cell, and to provide a good interpolation function for all other locations that have the continuity characteristic. If one wants to use rasters for continuous field representation, one usually uses the first approach but not the second, as the second technique is usually considered computationally too intensive for large rasters.

The location associated with a raster cell is fixed by convention: it may be the cell centroid (mid-point) or, for instance, its left lower corner. Values for other positions are computed using an interpolation function applied to one or more nearby field values. This allows us to represent continuous, even differentiable, functions. An important advantage of regular tessellations is that we know how they partition space, and that we can make our computations specific to this partitioning. This leads to fast algorithms. An obvious disadvantage is that they are not adaptive to the spatial phenomenon we want to represent. The cell boundaries are both artificial and fixed: they may or may not coincide with the boundaries of the phenomena of interest. If we use any of the above regular tessellations to represent an area with minor elevation differences, then, clearly we would need just as many cells as in a strongly undulating terrain: the data structure does not adapt to the lack of relief. We would, for instance, still use the $m \times n$ cells for the raster, even though variations in elevation are irrelevant.

Irregular tessellations

Regular tessellations provide simple structures with straightforward algorithms that are, however, not adaptive to the phenomena they represent. This means they might not represent the phenomena in the most efficient way. For this reason, substantial research effort has been put into irregular tessellation. Again, these are partitions of space into mutually distinct cells, but now the cells may vary in size and shape, allowing them to adapt to the spatial phenomena that they represent. Irregular tessellations are more complex than regular ones, but they are also more adaptive, which typically leads to a reduction in the amount of computer memory needed to store the data.

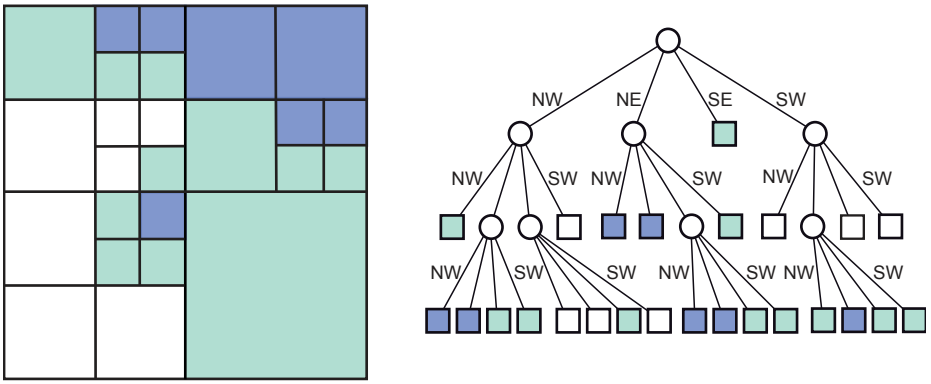


Figure 8.5
An 8×8, three-value raster (here, three colours) and its representation as a region quadtree. To construct a quadtree, the field is successively split into four quadrants until all parts have only a single value. After the first split, the southeast quadrant is entirely green, and this is indicated by a green square at level two of the tree. Other quadrants have to be split further.

quadtrees

A well-known data structure in this family—upon which many more variations have been based—is the region quadtree. It is based on a regular tessellation of square cells, but takes advantage of cases where neighbouring cells have the same field value, so that they can be represented together as one bigger cell. A simple illustration is provided in Figure 8.5. It shows a small 8×8 raster with three possible field values: white, green and blue. The quadtree that represents this raster is constructed by repeatedly splitting up the area into four quadrants, which are called NW, NE, SE, SW for obvious reasons. This procedure stops when all the cells in a quadrant have the same field value. The procedure produces an upside-down, tree-like structure, hence the name "quadtree". In the computer's main memory, the nodes of a quadtree (both circles and squares in Figure 8.5) are represented as records. The links between them are pointers, i.e. a programming technique to address (or to point to) other records. Quadtrees are adaptive because they apply Tobler's law (see Subsection 8.1.2). When a conglomerate of cells has the same value, they are represented together in the quadtree, provided their boundaries coincide with the predefined quadrant boundaries. Therefore, a quadtree provides a nested tessellation: quadrants are only split if they have two or more different values.

Vector representations

Tessellations do not explicitly store georeferences of the phenomena they represent. A georeference is a coordinate pair from some geographic space, and is also known as a vector. Instead, tesselations provide a georeference of the lower left corner of the raster, for instance, plus an indicator of the raster's resolution, thereby implicitly providing georeferences for all cells in the raster. In vector representations, georeferences are explicitly associated with the geographic phenomena. Examples and their vector representation are discussed below. To start, we will examine the TIN representation for geographic fields, which is a hybrid between tessellations and vector representations.

TIN

Triangulated Irregular Networks A commonly-used data structure in GIS software is the triangulated irregular network, or TINs. It is a standard implementation techniques for digital terrain models, but it can also be used to represent any continuous field. The principles on which a TIN is based are simple. It is built from a set of locations for which we have a measurement, for instance an elevation. The locations can be arbitrarily scattered in space and are not usually on a regular grid. Any location together with its elevation value can be viewed as a point in three-dimensional space (Figure 8.6). From these 3D points, we can construct an irregular tessellation made of triangles. Two such tessellations are illustrated in Figure 8.7.

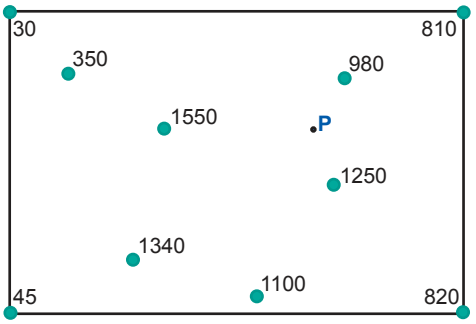


Figure 8.6
Input locations and their (elevation) values for a TIN construction. The location P is an arbitrary location that has no associated elevation measurement.

In three-dimensional space, three points uniquely determine a plane, as long as they are not collinear, i.e. they must not be positioned on the same line. A plane fitted through these points has a fixed aspect and gradient and can be used to compute an approximation of elevation of other locations. Since we can pick many triples of points, we can construct many such planes, and therefore we can have many elevation approximations for a single location, such as P (Figure 8.6).

By restricting the use of a plane to the triangular area "between" the three anchor points, we obtain a triangular tessellation of the complete study space. Unfortunately there are many different tessellations for a given input set of anchor points, as Figure 8.7 shows. Some tessellations are better than others, in the sense that they give smaller errors of elevation approximation. For instance, if we base our elevation computation for location P on the shaded triangle in the left-hand diagram, we will get a different value than that from the shaded triangle in the right-hand diagram. The latter will provide a better approximation because the average distance from P to the three triangle anchors is smaller.

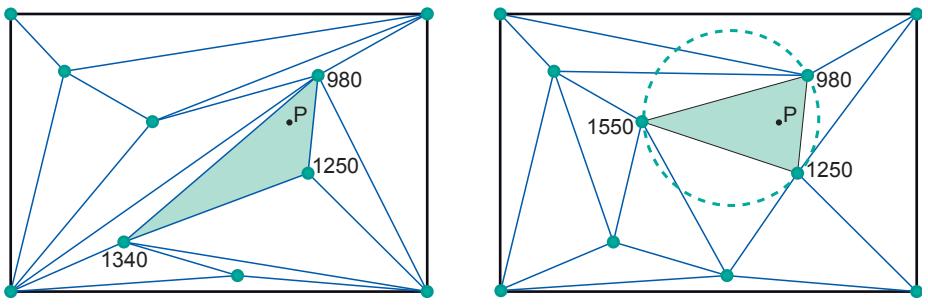


Figure 8.7
Two triangulations based on the input locations of Figure 8.6: (a) one with many "stretched" triangles; (b) the triangles are more "equilateral", known as *Delaunay triangulation*.

Delaunay triangulation

The triangulation of Figure 8.7b is a Delaunay triangulation, which in a sense is an optimal triangulation. There are several ways of defining such a triangulation (see [95]). Two important properties are, first, that the triangles are as equilateral ('equal-sided') as they can be, given the set of anchor points and, second, that for each triangle, the

circumcircle through its three anchor points does not contain any other anchor point. One such circumcircle is depicted on the right of Figure 8.7b. A TIN clearly is a vector representation: each anchor point has a stored georeference. Yet we might also call it an irregular tessellation, as the chosen triangulation provides a partitioning of the entire study space. However, in this case, the cells do not have an associated stored value as is typical of tessellations, but rather a simple interpolation function that uses the elevation values of its three anchor points.

Point representations Points are defined as single coordinate pairs (x, y) in 2D space, or coordinate triplets (x, y, z) in 3D space. Points are used to represent objects that are best described as shapeless, size-less, zero-dimensional features. Whether this is the case really depends on the purposes of the application and also on the spatial extent of the objects compared to the scale used in the application. For a tourist map of a city, a park would not usually be considered a point feature, but perhaps a museum would, and certainly a public phone booth might be represented as a point. In addition to the georeference, administrative or thematic data are usually stored for each point object that can capture relevant information about it. For phone-booth objects, for example, this may include the telephone company owning the booth, its phone number and the date it was last serviced.

Line representations Line data are used to represent one-dimensional objects such as roads, railroads, canals, rivers and power lines. Again, there is an issue of relevance for the application and the scale that the application requires. For the example of mapping tourist information, bus, subway and tram routes are likely to be relevant line features. Some cadastral systems, on the other hand, may consider roads to be two-dimensional features, i.e. having a width as well as length. Previously, we noted that arbitrary, continuous curvilinear features are as equally difficult to represent as continuous fields. GISs, therefore, approximate such features (finitely!) as lists of nodes: the two end nodes and zero or more internal nodes, or vertices, define a line. Other terms for "line" that are commonly used in some GISs are polyline, arc or edge. A node or vertex is like a point, but it only serves to define the line and provide shape in order to obtain a better approximation of the actual feature. The straight parts of a line between two consecutive vertices or end nodes are called line segments. Many GISs store a line as a sequence of coordinates of its end nodes and vertices, assuming that all its segments are straight. This is usually good enough, as cases in which a single straight line segment is considered an unsatisfactory representation can be dealt with by using multiple (smaller) line segments, instead of one.

arc, edge, node, vertex

Still, in some cases we would like to have the opportunity to use arbitrary curvilinear features to represent real-world phenomena. Think of a garden design with perfectly circular or elliptical lawns, or of detailed topographic maps showing roundabouts and the sidewalks. In principle all of this can be stored in a GIS, but currently many systems do not accommodate such shapes. A GIS function supporting curvilinear features uses parameterized mathematical descriptions, a discussion of which is beyond the scope of this textbook. Collections of (connected) lines may represent phenomena that are best viewed as networks. With networks, interesting questions arise that have to do with connectivity and network capacity. These relate to applications such as traffic monitoring and watershed management. With network elements—i.e. the lines that make up the network—extra values are commonly associated, such as distance, quality of the link or the carrying capacity.

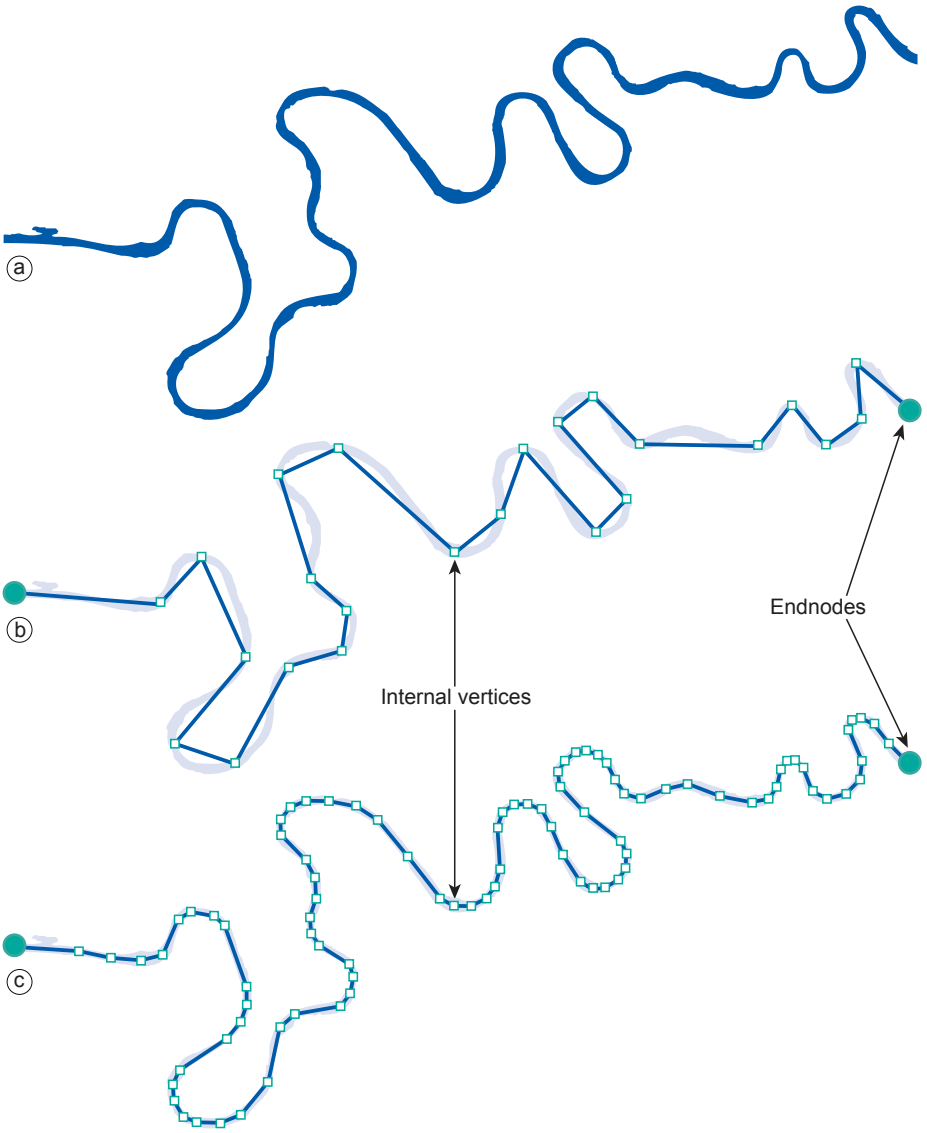


Figure 8.8
Examples of line representation: (a) the centerline of a river can be represented as a line feature; (b) this line feature consists of a start node, an end node and internal vertices; (c) by increasing the number of internal vertices, we can closer resemble the shape of the feature.

polygons

Area representations When area objects are stored using a vector approach, the usual technique is to apply a boundary model. This means that each area feature is represented by some arc/node structure that determines a polygon as the area's boundary. Common sense dictates that area features of the same kind are best stored in a single data layer, represented by mutually non-overlapping polygons. This results in an application-determined (i.e. adaptive) partition of space. A polygon representation for an area object is another example of a finite approximation of a phenomenon that may have a curvilinear boundary in reality. If the object has a fuzzy boundary, a polygon is an even worse approximation, even though potentially it may be the only one possible. Figure 8.9 illustrates a simple study with three area objects, each represented by polygon boundaries. Clearly, we expect additional data to accompany the area data. Such information could be stored in database tables.

A simple but naïve representation of area features would be to list for each polygon

the list of lines that describes its boundary. Each line in the list would, as before, be a sequence that starts with a node and ends with one, possibly with vertices in between. A closer look at the shared boundary between the bottom left and right polygons in Figure 8.9 shows why this approach is far from optimal. As the same line makes up the boundary from the two polygons, this line would be stored twice in the above representation, namely once for each polygon. This is a form of data duplication—known as data redundancy—which is (at least in theory) unnecessary, although it remains a feature of some systems. Another disadvantage of such polygon-by-polygon representations is that if we want to identify the polygons that border the bottom left polygon, we have to do a complicated and time-consuming search analysis comparing the vertex lists of all boundary lines with that of the bottom left polygon. For Figure 8.9, with just three polygons, this is fine, but in a data set with 5000 polygons, and perhaps a total of 25,000 boundary lines, this becomes a tedious task, even with the fastest of computers.

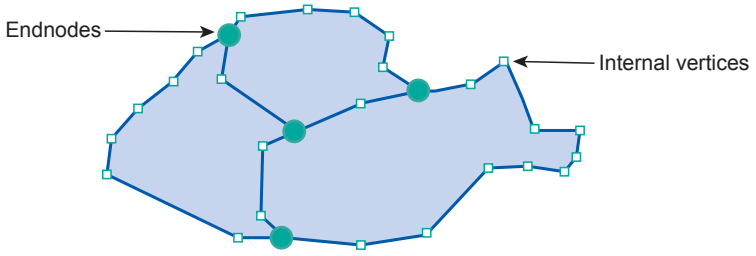


Figure 8.9
Areas as they are represented by their boundaries. Each boundary is a cyclic sequence of line features; each line, as before, is a sequence of two end nodes with zero or more vertices in between.

boundary model

The boundary model is an improved representation that deals with these disadvantages. It stores parts of a polygon's boundary as non-looping arcs and indicates which polygon is on the left and which is on the right of each arc. A simple example of the boundary model can be seen in Figure 8.10. It illustrates which additional information is stored about spatial relationships between lines and polygons. Obviously, real coordinates for nodes (and vertices) will also be stored in another table. The boundary model is also called the topological data model as it captures some topological information, such as polygon neighbourhood, for example. Observe that it is a matter of a simple query to find all the polygons that are the neighbour of a given polygon, unlike the case above.

line	from	to	left	right	vertexlist
b_1	4	1	W	A	...
b_2	1	2	B	A	...
b_3	1	3	W	B	...
b_4	2	4	C	A	...
b_5	3	4	W	C	...
b_6	3	2	C	B	...

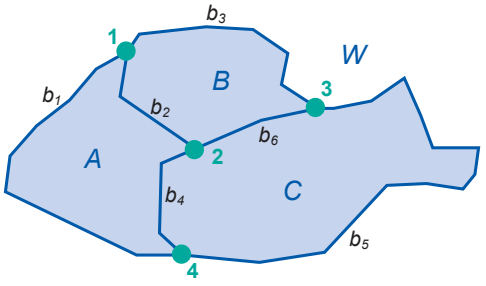


Figure 8.10
A simple boundary model for the polygons A, B and C. For each arc, we store the start and end node (as well as a vertex list, but this has been omitted from the table), its left and right polygon. The "polygon" W denotes the polygon of the outside world.

Topology and spatial relationships

General spatial topology Topology deals with spatial properties that do not change under specific transformations. Take features (as in Figure 8.11) drawn on a sheet of rubber. These features can be made to change in shape and size by stretching and pulling the sheet, yet some properties of these features will not change:

- area E is still inside area D;
- the neighbourhood relationships between A, B, C, D, and E stay intact, and their boundaries have the same start and end nodes;
- the areas are still bounded by the same boundaries, only the shapes and lengths of their perimeters have changed.

topological relationships

Topological relationships are built from simple elements into more complex elements: nodes define line segments, and line segments connect to define lines, which in turn define polygons. Issues relating to order, connectivity and adjacency of geographical elements form the basis of more sophisticated GIS analyses. These relationships (called topological properties) are invariant under a continuous transformation and are referred to as a topological mapping.

We will now consider topological aspects in two ways. Firstly, using simplices, we will look at how simple elements define more complex ones. Secondly, we will examine the logical aspects of topological relationships using set theory. The three-dimensional case is also briefly discussed.

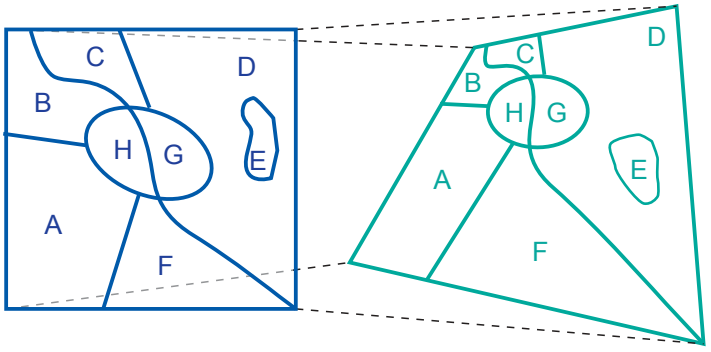


Figure 8.11
Rubber sheet transformation: the space is transformed, yet many relationships between the constituents remain unchanged.

Topological relationships The mathematical properties of the geometric space used for spatial data may be described as follows:

- The space is a 3D Euclidean space in which we can determine for every point its coordinates as a triple (x, y, z) of real numbers. In this space, we can define features such as points, lines, polygons and volumes as geometric primitives of the respective dimension. A point is a zero-dimensional, a line a one-dimensional, a polygon a two-dimensional, and a volume a three-dimensional primitive.
- The space is a metric space, which means that we can always compute the distance between two points according to a given distance function. Such a function is also known as a metric.
- The space is a topological space, the definition of which is a bit complicated. In essence, for every point in the space we can find a neighbourhood around it that fully belongs to that space as well.
- Interiors and boundaries are properties of spatial features that remain invariant under topological mappings. This means that, under any topological mapping, the interior and the boundary of a feature remains unbroken and intact.

A number of advantages exist when our computer representations of geographic phenomena have built-in sensitivity to topological issues. Questions related to the "neighbourhood" of an area are a case in point. To obtain some "topological sensitivity", simple building blocks have been proposed with which more complicated representations can be constructed:

- We can define features within the topological space that are easy to handle and that can be used as representations of geographic objects. These features are called simplices as they are the simplest geometric shapes of some dimension:
  - point (0-simplex),
  - line segment (1-simplex),
  - triangle (2-simplex),
  - and tetrahedron (3-simplex).
- When we combine various simplices into a single feature, we obtain a simplicial complex; see Figure 8.12 for examples.

As the topological characteristics of simplices are well-known, we can infer the topological characteristics of a simplicial complex from the way it was constructed.

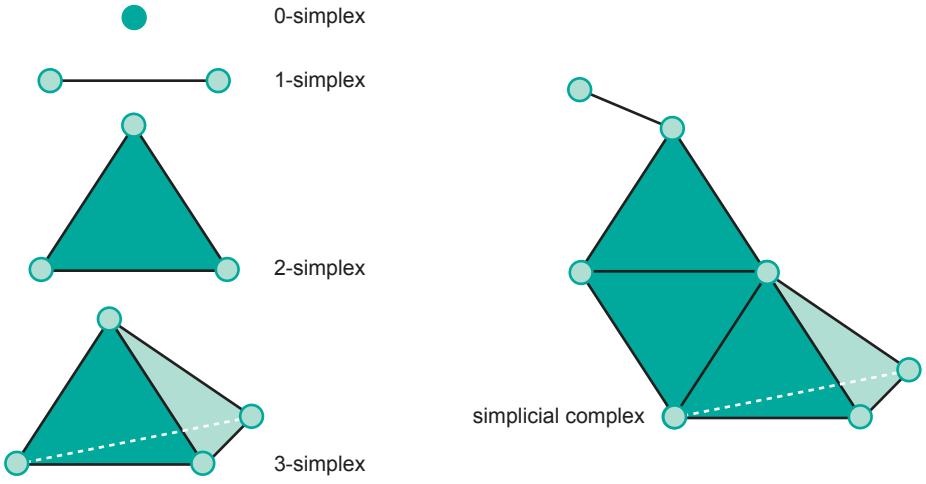


Figure 8.12
Simplices and a simplicial complex. Features are approximated by a set of points, line segments, triangles and tetrahedrons.

The topology of two dimensions We can use the topological properties of interiors and boundaries to define relationships between spatial features. Since the properties of interiors and boundaries do not change under topological mapping, we can investigate their possible relations between spatial features. We can define the *interior* of a region, R, as the largest set of points of R for which we can construct a disc-like environment around it (no matter how small) that also falls completely inside R. The boundary of R is the set of those points belonging to R that do not belong to the interior of R, i.e. one cannot construct a disc-like environment around such points that still belongs to R completely.

interior and exterior

Let us consider a spatial region A. It has a boundary and an interior, both seen as (infinite) sets of points, which are denoted by $boundary(A)$ and $interior(A)$, respectively. We consider all possible combinations of intersections ($\cap$) between the boundary and

set theory

the interior of A with those of another region, B, and test whether they are the empty set ($\emptyset$) or not. From these intersection patterns, we can derive eight (mutually exclusive) spatial relationships between two regions. If, for instance, the interiors of A and B do not intersect, but their boundaries do, yet the boundary of one does not intersect the interior of the other, we say that A and B meet. In mathematics, we can therefore define the "meets relationship" using set theory as:

$$\begin{aligned} A \; meets \; B \quad \stackrel{\text{def}}{=} \quad & interior(A) \cap interior(B) = \emptyset \wedge \\ & boundary(A) \cap boundary(B) \neq \emptyset \wedge \\ & interior(A) \cap boundary(B) = \emptyset \wedge \\ & boundary(A) \cap interior(B) = \emptyset. \end{aligned}$$

In the above formula, the symbol $\wedge$ expresses the logical connective "and". Thus, the formula states four properties that must all hold for the formula to be true.

Figure 8.13 shows all eight spatial relationships: *disjoint*, *meets*, *equals*, *inside*, *covered by*, *contains*, *covers* and *overlaps*. These relationships can be used, for instance, in queries on a spatial database. Rules of how simplices and simplicial complexes can be embedded in 2D and 3D space are quite different. Such a set of rules defines the topological consistency of that space. It can be proven that if the rules presented and illustrated in Figure 8.14 are satisfied for all features in 2D space, then the features define a topologically consistent configuration in 2D space.

topological consistency

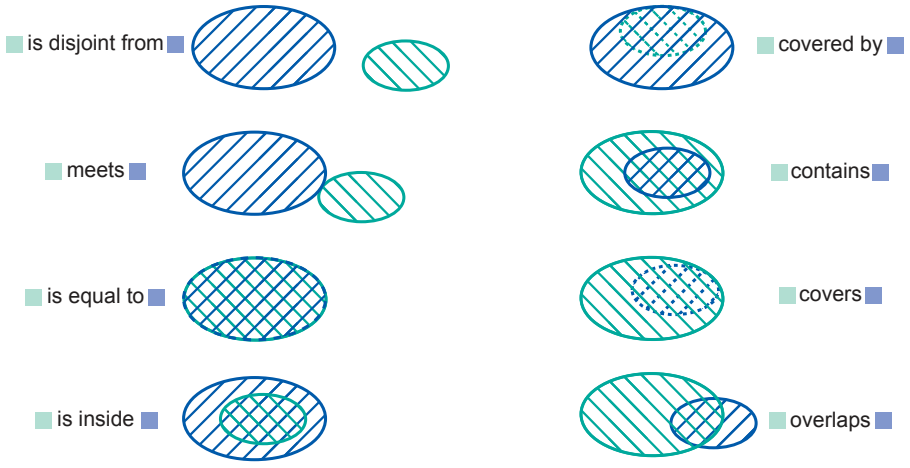


Figure 8.13
Spatial relationships between two regions derived from the topological invariants of intersections of boundary and interior.

The topology of three dimensions Our discussion of vector representations and spatial topology has so far focused on objects in 2D space. The history of spatial data handling is almost purely 2D, and this is remains the case for the majority of present-day GIS applications. Many application domains make use of elevational data, but these are usually accommodated for by what are known as 2.5D data structures. These 2.5D data structures are similar to the 2D data structures just discussed, using points, lines and areas. They also apply the rules of two-dimensional topology, as illustrated in Figure 8.14. This means that different lines cannot cross without intersecting nodes and that different areas cannot overlap. There is, however, one important aspect in

1. Every 1-simplex ('arc') must be bounded by two 0-simplices ('nodes', namely its begin and end node).
2. Every 1-simplex borders two 2-simplices ('polygons', namely its 'left' and 'right' polygons).
3. Every 2 -simplex has a closed boundary consisting of an alternating (and cyclic) sequence of 0- and 1-simplices.
4. Around every 0-simplex exists an alterning (and cyclic) sequence of 1- and 2-simplices.
5. 1-simplices only intersect at their (bounding) nodes.

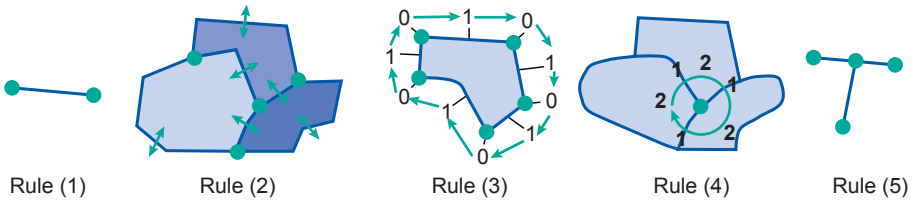


Figure 8.14
The five rules of topological consistency in two-dimensional space.

which 2.5D data do differ from standard 2D data and that is in their association of an additional $z-$value with each 0-simplex ('node'). Thus, nodes also have an elevation value associated with them. Essentially, this allows the GIS user to represent 1- and 2-simplices that are non-horizontal, so that a piece-wise planar, "wrinkled surface" can be constructed as well, much like a TIN. One cannot have two different nodes with identical x and y coordinates but different z values. Such nodes would constitute a perfectly vertical feature, which is not allowed. Consequently, true solids cannot be represented in a 2.5D GIS.

Solid representation is an important feature for some dedicated GIS application domains. Two examples of such applications are: mineral exploration, where solids represent ore bodies; and urban models, where solids represent various human constructions, such as buildings and sewers. The 3D characteristics of such objects are fundamentally important since their depth and volume may matter, or their real life visibility must be faithfully represented. A solid can be defined as a true 3D object. An important class of solids in 3D GISs is formed by polyhedra, which are the solids limited by their planar facets. A facet is polygon-shaped, flat side that is part of the boundary of a polyhedron. Any polyhedron has at least four facets, which happens to be the case for the 3-simplex. Most polyhedra have many more facets, e.g. a cube already has six.

Scale and resolution

large-scale and small-scale maps

In the practice of spatial data handling, one often comes across questions like "What is the resolution of the data?" or "At what scale is your data set?" Now that we have moved firmly into the digital age, these questions sometimes defy an easy answer. Map scale can be defined as the ratio between the distance on a printed map and the distance of the same stretch in the terrain. A 1:50,000 scale map means that 1 cm on the map represents 50,000 cm (i.e. 500 m) in the terrain. "Large-scale" means that the ratio is relatively large, so typically it means there is much detail to see, as on a 1:1000 printed map. "Small-scale", in contrast, means a small ratio, hence less detail, as on a 1:2,500,000 printed map. When applied to spatial data, the term resolution is commonly associated with the cell width of the tessellation applied.

Digital spatial data, as stored in a GIS, are essentially without scale: scale is a ratio notion associated with visual output, such as a map or on-screen display, not with the data that was used to produce the map or display. When digital spatial data sets have been collected with a specific map-making purpose in mind, and all maps have been designed to use one single map scale, for instance 1:25,000, we may assume that the

data carries the characteristic of "a 1:25,000 digital data set."

Representations of geographic fields

We have looked at various representation techniques in some detail. Now we can study which of them can be used to represent a geographic field. A geographic field can be represented by means of a tessellation, a TIN or a vector representation. The choice between them is determined by the requirements of the application in mind. It is more common to use tessellations, notably rasters, for field representation, but vector representations are in use too. We have already looked at TINs, so the following subsections only present examples of the other two representation techniques.

Raster representation of a field Figure 8.15 illustrates how a raster represents a continuous field, in this case elevation. Different shades of blue indicate different elevation values, with darker blue tones indicating higher elevations. The choice of a blue spectrum is only to make the illustration aesthetically pleasing; real elevation values are stored in the raster, so we could have printed a real number value in each cell instead. This would not have made the figure very legible, however. A raster can be thought of as a long list of field values: actually, there should be $m \times n$ such values present. The list is preceded with some extra information, such as a single georeference for the origin of the whole raster, a cell-size indicator, the integer values for m and n, and an indicator of data type for interpreting cell values. Rasters and quadtrees do not store the georeference of each cell, but infer it from the extra information about the raster. A TIN is a much "sparser" data structure: as compared to a regular raster, the amount of data stored is less for a structure of approximately equal interpolation error. The quality of the TIN depends on the choice of anchor points, as well as on the triangulation built from it. It is, for instance, wise to perform "ridge following" during the data acquisition process for a TIN. Anchor points on elevation ridges will assist in correctly representing peaks and faces of mountain slopes.

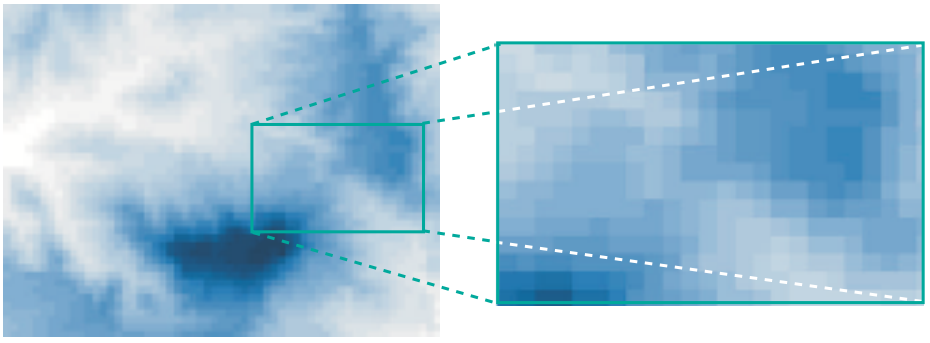


Figure 8.15
A raster representation (in part) of the elevation of the study area of Figure 8.1 (Falset, Spain). Actual elevation values are indicated in shades of blue. The depicted area is the northeast flank of the mountain in the southeastern part of the study area. The right-hand figure zooms in on a part of the left-hand figure.

Vector representation of a field We briefly mention the vector representation for fields such as elevation, which uses isolines of the field. An isoline is a linear feature that connects points with equal field values. When the field is elevation, we also speak of contour lines. Elevations in the Falset study area are represented by contour lines in Figure 8.16. Both TINs and isoline representations use vectors. Isolines as a representation mechanism are not common, however. They are used as a geoinformation visualization technique (in mapping, for instance), but usually it is better to choose a TIN for representing this type of field. Many GIS packages provide functions to generate an isoline visualization from a TIN.

Figure 8.16
A vector-based elevation field representation for the Falset study area; see Figure 8.1. Elevation isolines are indicated at a resolution of 25 m.

Representation of geographic objects

The representation of geographic objects is most naturally supported with vectors. After all, objects are identified by the parameters of location, shape, size and orientation (see Section 8.1.1), and many of these parameters can be expressed in terms of vectors. Tessellations are also commonly used for representing geographic objects.

Tessellations for representing geographic objects Remotely-sensed images are an important data source for GIS applications. Unprocessed digital images contain many pixels, each of which carrying a reflectance value. Various techniques exist to process digital images into classified images that can be stored in a GIS as a raster. Image classification characterizes each pixel into one of a finite list of classes, thereby obtaining an interpretation of the contents of the image. The recognized classes can be crop types, as in the case of Figure 8.17, or urban land use classes, as in the case of Figure 8.18. These figures illustrate the unprocessed images (a) and a classified version of the image (b). For the application at hand, perhaps only potato fields (Figure 8.17b, in yellow) or industrial complexes (Figure 8.18b, in orange) are of interest. This would mean that all other classes are considered unimportant and would probably be ignored in further analysis. If that further analysis can be carried out with raster data formats, then there is no need to consider vector representations.

Nonetheless, we must make a few observations regarding the representation of geographic objects in rasters. Line and point objects are more awkward to represent using rasters. Area objects, however, are conveniently represented in a raster, although area boundaries may appear as jagged edges. This is a typical by-product of raster resolution versus area size, and artificial cell boundaries. This may have consequences for area-size computations: the precision with which the raster defines the object's size is limited. After all, we could say that rasters are area based and that geographic objects

that are perceived as lines or points are considered to have zero area size. Standard classification techniques may, moreover, fail to recognize these objects as points or lines.

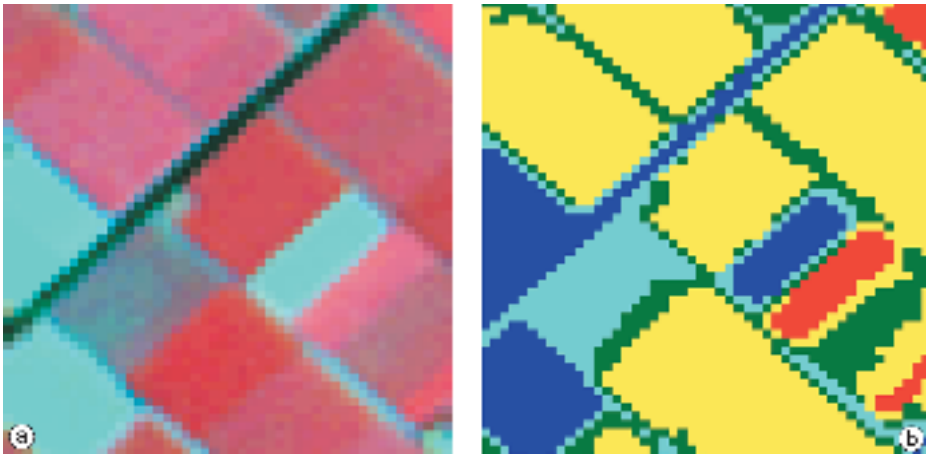


Figure 8.17
An unprocessed digital image (a) and a classified raster (b) of an agricultural area.

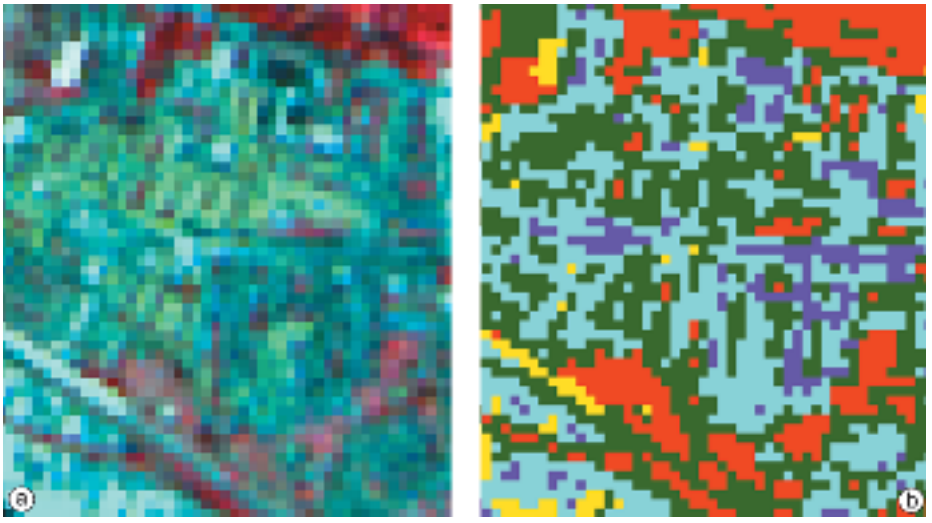


Figure 8.18
An unprocessed digital image (a) and a classified raster (b) of an urban area.

Many GISs support line representations in a raster, as well as operations on them. Lines can be represented as strings of neighbouring raster cells of equal value (Figure 8.19). Supported operations are connectivity operations and distance computations. Note that the issue of the precision of such computations needs to be addressed.

Vector representations of geographic objects A more natural way of depicting geographic objects is by means of vector representations. Most of the issues related to this have already been discussed in Subsection 8.1.2, so a small example should suffice here. Figure 8.20 shows a number of geographic objects in the vicinity of the ITC building. These objects are area representations in a boundary model. Nodes and vertices of the polylines that make up the objects' boundaries are not illustrated, though obviously they have been stored.

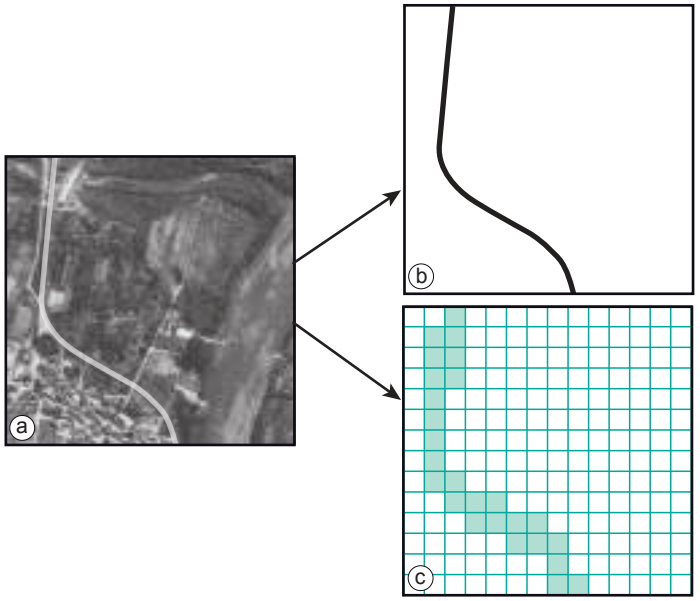


Figure 8.19
A linear feature (a) represented as a vector line feature (b) and in a raster layer (c).

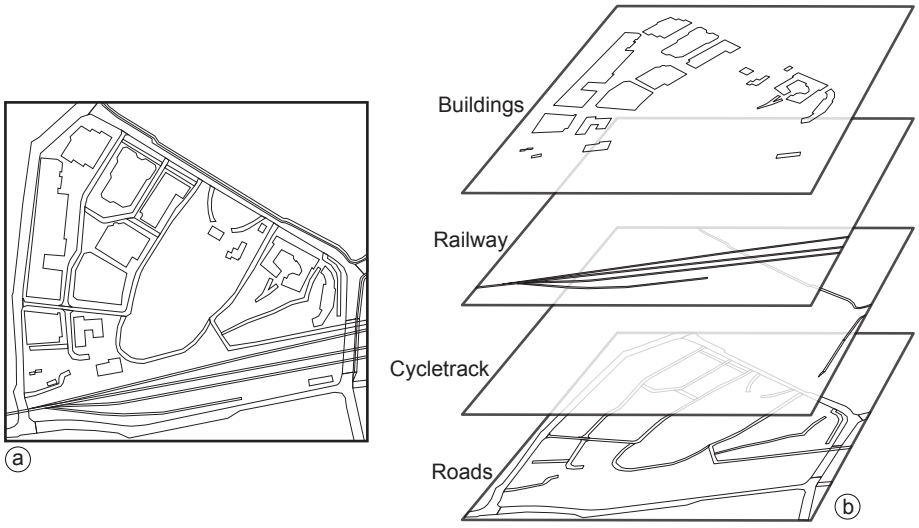


Figure 8.20
Various objects displayed as area objects in a vector representation. Similar data types are stored in the same single layer (e.g. Buildings). For each different type a new layer is used (b).

8.1.3 Organizing and managing spatial data

In Subsection 8.1.2 we discussed various types of geographic information and ways of representing it. We did not, however, pay much attention to how various sorts of spatial data can be combined in a single system.

The main principle of data organization applied in a GIS is that of spatial data layers. A spatial data layer is either a representation of a continuous or discrete field, or a collection of objects of the same kind. Usually, the data are organized such that similar elements are in a single data layer. For example, all telephone booth point objects would be in one layer, and all road line objects in another. A data layer contains spatial data—of any of the types discussed above—as well as attribute (i.e. thematic) data, which further describes the field or objects in the layer. Attribute data are quite

often arranged in tabular form, maintained in some kind of geo-database, as we will see in Section 8.4. An example of two field-data layers is provided in Figure 8.21.

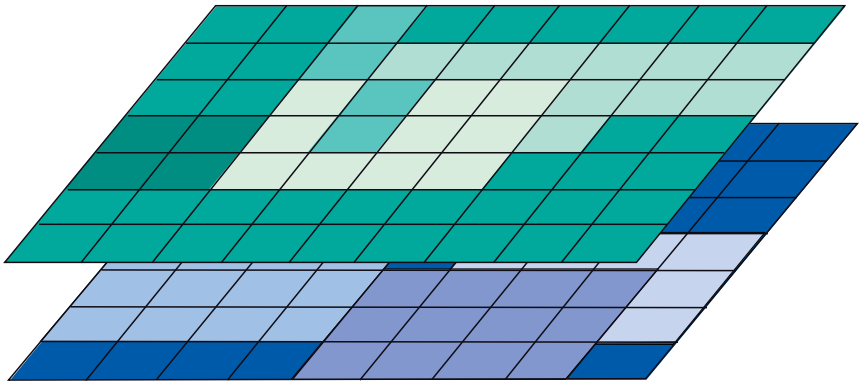


Figure 8.21
Different rasters can be overlaid to look for spatial correlations.

Data layers can be laid over each other, inside a GIS package, to study combinations of geographic phenomena. We shall see below that a GIS can be used to study the spatial correlation between different phenomena, albeit requiring computations that overlay one data layer with another. This is schematically depicted in Figure 8.22 for two different object layers. Field layers can also be involved in overlay operations. For a more detailed discussion of the functions offered for data management by GISs and database systems, refer to Chapter 9.

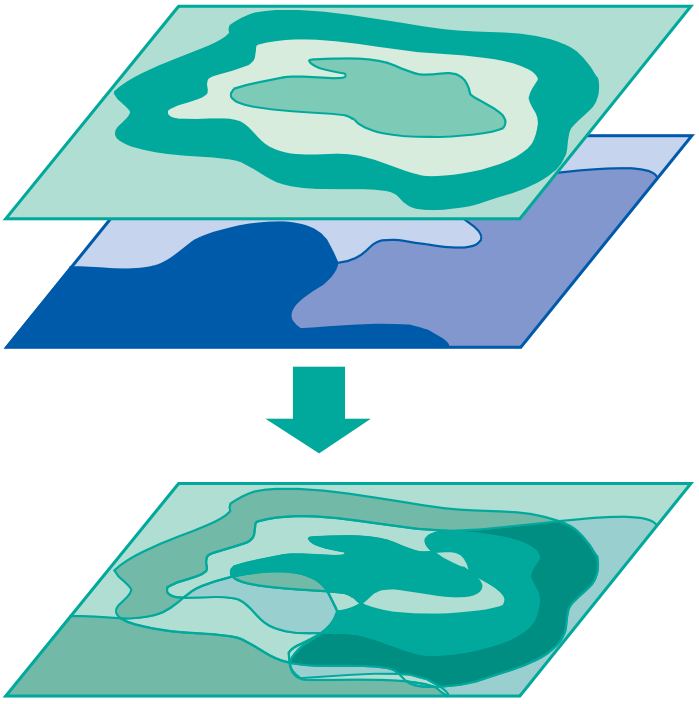


Figure 8.22
Two different object layers can be overlaid to look for spatial correlations; the result can be used as a separate (object) layer.

8.1.4 The temporal dimension

Besides having geometric, thematic and topological properties, geographic phenomena also change over time and are thus dynamic. Examples include identifying the

owners of a land parcel in 1972, or determining how land cover in a certain area changed from native forest to pasture land over a specific time period. Some features or phenomena change slowly, e.g. geological features or land cover, as in the example just given. Other phenomena change very rapidly, such as the movement of people or atmospheric conditions. Some examples are provided in Figure 8.23. For an increasing number of applications, these changes themselves are the key aspect of the phenomenon to be studied. For different applications, different scales of measurement will apply.

dynamic phenomena

- Where and when did something happen?
- How fast did this change occur?
- In which order did the changes occur?

The way we represent relevant components of the real world in our models determines the kinds of questions we can or cannot answer. In this chapter we have already discussed representation issues for spatial features, but so far we have ignored issues for incorporating time. The main reason is that GISs still offer limited support for doing so. As a result, most studies require substantial efforts from the GIS user in data preparation and data manipulation. Also, besides representing an object or field in 2D or 3D space, the temporal dimension is of a continuous nature. Therefore, in order to represent it in a GIS we have to discretize the time dimension.

time in GIS

Spatio-temporal data models are ways of organizing representations of space and time in a GIS. Several representation techniques have been proposed in the literature. Perhaps the most common of these is the "snapshot state", which represents a single moment in time of an ongoing natural or man-made process. We may store a series of these "snapshot states" to represent "change", but we must be aware that this is by no means a comprehensive representation of that process. Further treatment of spatio-temporal data models is outside the scope of this book and readers are referred to Langran [64] for a discussion of relevant concepts and issues.

As time is the central concept of the temporal dimension, a brief examination of the nature of time may clarify our thinking when we work with this dimension:

- Discrete and continuous time: Time can be measured along a discrete or continuous scale. Discrete time is composed of discrete elements (seconds, minutes, hours, days, months, or years). For continuous time, no such discrete elements exist: for any two moments in time there is always another moment in between. We can also structure time by events (moments) or periods (intervals). When we represent intervals by a start and an end event, we can derive temporal relationships between events and periods, such as "before", "overlap", and "after".
- Valid time and transaction time: Valid time (or world time) is the time when an event really happened, or a string of events took place. Transaction time (or database time) is the time when the event was stored in the database or GIS. Note that the time at which we store something in a database is typically (much) later than when the related event took place.
- Linear, branching and cyclic time: Time can be considered to be linear, extending from the past to the present ('now'), and into the future. This view gives a single time line. For some types of temporal analysis, branching time—in which different time lines from a certain point in time onwards are possible—and cyclic time—in which repeating cycles such as seasons or days of the week are recognized—make more sense and can be useful.

- Time granularity: When measuring time, we speak of granularity as the precision of a time value in a GIS or database (e.g. year, month, day, second). Different applications may obviously require different granularity. In cadastral applications, time granularity might well be a day, as the law requires deeds to be date-marked; in geological mapping applications, time granularity is more likely to be in the order of thousands or millions of years.

- Absolute and relative time: Time can be represented as absolute or relative. Absolute time marks a point on the time line where events happen (e.g. "6 July 1999 at 11:15 p.m."). Relative time is indicated relative to other points in time (e.g. "yesterday", "last year", "tomorrow", which are all relative to "now", or "two weeks later", which is relative to some other arbitrary point in time.).

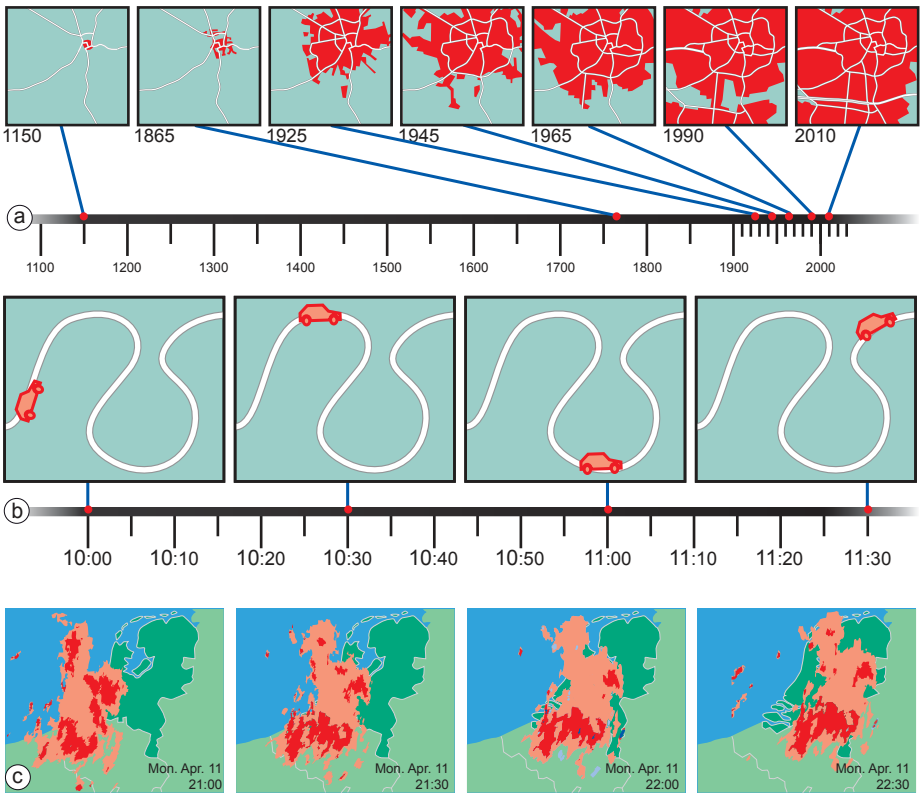


Figure 8.23
Examples of spatio-temporal phenomena: (a) expansion of a city, the area covered by the city grows over time, but the location of the city does not change; (b) a moving car will change position, but the object car does not change size or shape; (c) over time, the position of a cloud will change, but also the size and shape of the cloud can undergo changes over time.

spatio-temporal analysis

In spatio-temporal analysis we consider changes of spatial and thematic attributes over time. We can keep the spatial domain fixed and look only at the attribute changes over time for a given location in space. We might be interested how land cover has changed for a given location or how land use has changed for a given land parcel over time, provided its boundary has not changed. On the other hand, we can keep the attribute domain fixed and consider the spatial changes over time for a given thematic attribute. In this case, we might want to identify locations that were covered by forest over a given period of time.

Finally, we can assume both the spatial and attribute domains are variable and consider how fields or objects have changed over time. This may lead to notions of object motion—a subject receiving increasing attention in the literature. Applications

of moving object research include traffic control, mobile telephony, wildlife tracking, vector-borne disease control and weather forecasting. In these types of applications, the problem of object identity becomes apparent. When does a change or movement cause an object to disappear and become something new? With wildlife this is quite obvious; with weather systems less so. But this should no longer be a surprise: we have already seen that some geographic phenomena can be nicely described as objects, while others are better represented as fields.

8.2 Data entry

Spatial data can be obtained from various sources. It can be collected from scratch, using direct spatial-data acquisition techniques, or indirectly, by making use of existing spatial data collected by others. The first source could include field survey data and remotely sensed images. To the second source belongs printed maps and existing digital data sets. This section discusses the collection and use of data from both sources.

8.2.1 Spatial data input

One way to obtain spatial data is by direct observation of relevant geographic phenomena. This can be done through ground-based field surveys or by using remote sensors on satellites or aircraft. Many Earth science disciplines have developed specific survey techniques as ground-based approaches remain the most important source of reliable data in many cases.

primary data

Data that are captured directly from the environment are called *primary data*. With primary data, the core concern in knowing their properties is to know the process by which they were captured, the parameters of any instruments used, and the rigour with which quality requirements were observed.

In practice, it is not always feasible to obtain spatial data by direct capture. Factors of cost and available time may be a hindrance, and sometimes previous projects have acquired data that may fit a current project's purpose.

secondary data

In contrast to direct methods of data capture, spatial data can also be sourced indirectly. This includes data derived by scanning existing printed maps, data digitized from a satellite image, processed data purchased from data-capture firms or international agencies, and so on. This type of data is known as *secondary data*. Secondary data are derived from existing sources and have been collected for other purposes, often not connected with the investigation at hand.

Key sources of primary and secondary data, and several issues related to their use in analyses that users should be aware of, are discussed in the remainder of this section.

8.2.2 Aerial surveys and satellite remote sensing

Aerial photographs are a major source of digital data (see Section 4.6); soft-copy workstations are used to digitize features directly from stereo pairs of digital photographs. These systems allow data to be captured in two or three dimensions, with elevations measured directly from a stereo pair using the principles of photogrammetry. Analogue aerial photos are often scanned before being entered into a soft-copy system, but with the advance of high-quality digital cameras this step can now be skipped.

In general, the alignment of roads and railways, lakes and water, and shapes of buildings are easily interpreted on aerial photographs—assuming that the scale of the photographs is not too small. Also, constructions such as dikes, bridges, air fields and the main types of vegetation and cultivation are mostly clearly visible. Nevertheless, numerous attribute data related to terrain features cannot be interpreted on aerial photographs: e.g. the administrative qualification of roads, sea and lake depths, functions of buildings, street names, and administrative boundaries. We will have to collect this information in the field or from existing data sets and maps (e.g. road maps, navigational charts or town plans). Issues related to the process of visual image interpretation are discussed in greater detail in Section 6.1.

Satellite remote sensing is another important source of spatial data. For this, satellites use different sensor packages to passively measure reflectance of parts of the elec-

tromagnetic spectrum or radio waves that were emitted by an active sensor such as radar (see Sections 4.4). Remote sensing collects raster data that can be further processed using different wavelength bands to identify objects and classes of interest, e.g. land cover. Issues related to the pre-processing of satellite remote-sensing data are discussed in greater detail in Chapter 5.

Figure 8.24
Aerial surveys (a) and satellite remote sensing (b) are employed to map relatively large areas at comparably large scales, source : Shuttle Radar Topography Mission, U.S. Geological Survey Department of the Interior/USGS and NASA, JPL.

8.2.3 Terrestrial surveys

Terrestrial surveys are usually employed for details that must be measured accurately, e.g. survey control stations, property boundaries, buildings, and road construction works (Figure 8.25). The surveyed data are often used to supplement and update existing data and for verification of data collected from aerial surveys and by satellite remote sensing. A terrestrial survey records features showing their relative position both horizontally and vertically. Several surveying techniques are employed to do this.

In horizontal positioning, measured angles at, and at distances from, known points are used to determine the positions of other points. Traditionally, survey measurements were made with optical and mechanical surveying instruments, such as a theodolite to measure angles, and more accurate electronic and electro-optical devices such as lasers for measuring distances. A more modern instrument is a total station, which is a theodolite with an electronic distance measurement device. Since the introduction of total stations, there has been a technological shift from the use of optical-mechanical devices to fully electronic systems incorporating a computer and relevant software.

Though satellite receivers are used for terrestrial surveying, total stations are still used widely, along with other types of surveying instruments, because of their accuracy, and area of operation: satellite systems do not work well in areas with dense tree cover or a high density of buildings.

Vertical positioning is usually done by levelling, which is a technique for measuring differences in height between established points relative to a datum or base point. Over short distances, levelling telescopes are used to view a staff or pole and, with the aid of a bar code, the height is recorded in relation to the previous station (Figure 8.25).

Elevation heights can also be derived with satellite receivers, albeit usually with some-

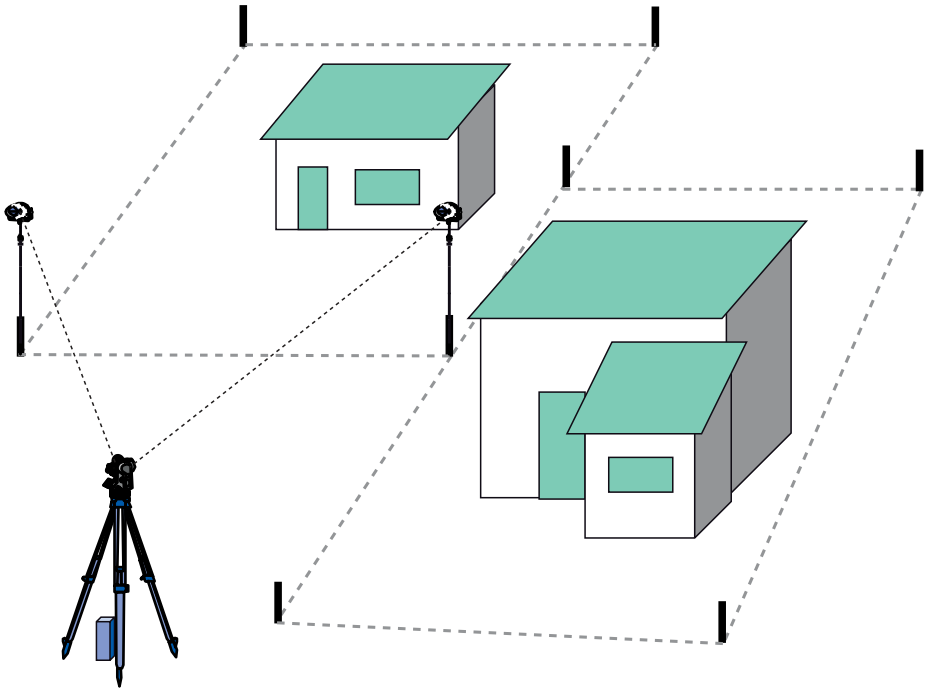


Figure 8.25
Terrestrial surveys are used to map details such as property boundaries.

Figure 8.26
Satellite-based surveys enable efficient data collection in open areas.

what less accuracy than for traditional precise levelling. However, the accuracy of satellite receivers may be similar if traditional levelling has to be used over a long distance. Laser altimetry (see Section 4.5) is employed for large areas, but its accuracy is not as good as levelling or GPS.

8.2.4 Field surveys

Every field science in natural resources, water resources, and urban and regional planning has a range of techniques for collecting data in the terrain. Full details of these techniques are given in various standard texts for the disciplines concerned.

Field surveys of natural and water resources are frequently carried out to check and supplement information derived from the interpretation of aerial photographs and satellite imagery (Figure 8.27). Often, sample areas are chosen within a study area for more detailed investigations. Socio-economic data, however, are often collected on the basis of administrative districts, with the result that their location is insufficiently precise to permit analysis of high quality.

Figure 8.27
Field workers checking and collecting supplementary information in the field.

Primary socio-economic data are collected by interviews and questionnaires. If the investigation is unofficial, the response will depend on the type of information required. In general, information of a financial or personal nature is difficult to come by and, even if given, may not be wholly reliable.

Fortunately a wealth of societal and economic data is available from official sources. Private individuals and commercial undertakings are usually required to provide government agencies with information via censuses, tax returns, etc. Since much of this data is confidential, it will usually be refined and generalized before it is released to others.

An example of a publicly available statistical data set is the International Data Base (IDB) provided by the US Census Bureau. It contains demographic and socio-economic statistics collected for 227 countries and areas of the world. The major types of data made available by the IDB are population by age and sex, birth and dead rates, migration, ethnicity, religion, language, literacy, labour force, employment, income and household composition.

Mobile GIS

Until recently, printed maps and forms were taken to the field and the information collected was sketched as notes on the map or written down on a form. This information was entered into a GIS database after returning to the office. This method of data collection is inefficient and prone to error. With a mobile GIS system and the support of a satellite receiver, we can take a GIS into the field with us on powerful, compact mobile computers and view, capture and update information, and then synchronize changes between the field and office (Figure 8.28).

Professional applications for mobile GISs are endless—utilities, forestry, environmental monitoring, field engineering, to mention a few. With the integration of systems, users are able to view each others' locations and, for example, share field data dynamically across their organization. Specifically, the data captured with mobile GISs can be instantly checked, updated and exchanged if necessary.

A simple task-driven mobile application begins in the office. GIS data are extracted from the main database and mapped onto the mobile device to be used in the field. The updated data are uploaded after returning to the office (Figure 8.29a).

A high-end mobile GIS application typically runs on a powerful laptop computer,

Figure 8.28
Mobile GIS provides the integration of mapping, GIS and positioning to field users via hand-held and mobile devices.

many of which provide a rich set of tools comparable to a desktop GIS application. A fast wireless connection enables direct access to maps and databases at the office, and synchronizes changes between the field and office through a web service (Figure 8.29b).

In cases where there is no connection to the main database in the office (e.g. a firewall makes access impossible), field edits can be synchronized later, when access to the main database is provided (Figure 8.29c). A versioned transaction may take care of the situation that the same feature (in the field) is updated several times: it can compare the updates (reconciling the version edits) before transferring the feature to the main, or parent, database.

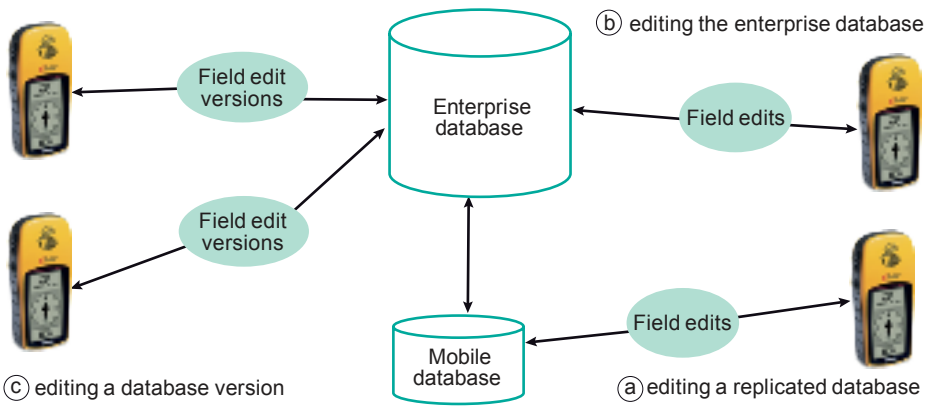


Figure 8.29
Mobile updating strategies: (a) Data are extracted from the main (enterprise) database and mapped onto the mobile device. Field edits are uploaded to the main database after returning to the office. (b) Wireless connection between field and office enables real-time updating. (c) Multiple versions of the database are used to enable updating by multiple mobile users, disconnected from the network.

8.2.5 Digitizing and scanning of analogue maps

A traditional method of obtaining spatial data is through digitizing existing printed maps. This can be done using various techniques. Before adopting this approach, one must be aware that positional errors already on the map will further accumulate and that one must be willing to accept these errors.

There are two forms of digitizing: on-tablet and on-screen manual digitizing (Figure 8.30). In on-tablet digitizing, the original map is fitted on a special surface (the tablet), while in on-screen digitizing, a scanned image of the map (or some other image) is shown on the computer screen. In both of these forms, an operator follows the map's features (mostly lines) with a mouse device, thereby tracing the lines, and storing location coordinates relative to a number of previously defined control points.

The function of these points is to "lock" a coordinate system onto the digitized data: the control points on the map have known coordinates, so by digitizing them we tell the system implicitly where all other digitized locations are. At least three control points are needed, but preferably more should be digitized, to allow checking for any positional errors.

control points

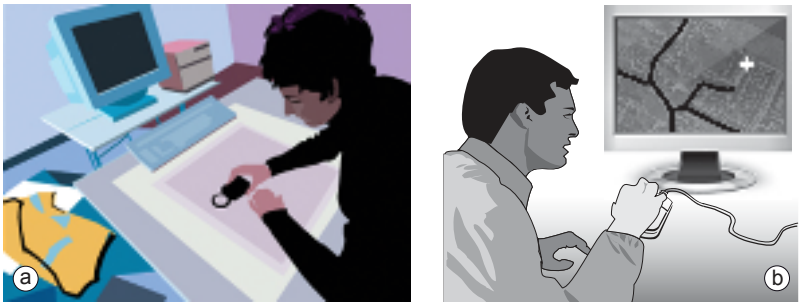


Figure 8.30
Manual digitizing techniques: (a) on-tablet digitizing; (b) on-screen digitizing.

Another set of techniques also works from a scanned image of the original map, but uses a GIS to find features in the image. These techniques are known as semi-automatic or automatic digitizing, depending on how much operator interaction is required. If vector data are to be distilled from this procedure, a process known as vectorization follows the scanning process. This procedure is less labour-intensive but can only be applied for relatively simple sources.

Scanning

A scanned image of the original map is needed for on-screen manual digitizing and semi-automatic/automatic digitizing. A range of scanners are available for obtaining a scanned image, starting from a small-format (A4) desktop scanner with resolutions of 200–800 dpi, through to high-end flatbed and drum scanners suitable for very accurate scanning of large-sized documents (A0) (Figure 8.31).

Figure 8.31
Main types of scanners: (a) a flatbed scanner. (b) a drum scanner.

A scanner illuminates a document and measures the intensity of the reflected light with a CCD array. The result is an image represented as a matrix of pixels, each of which holds an intensity value. Office scanners have a fixed maximum resolution, expressed as the highest number of pixels they can identify per inch; the unit is dots per inch (dpi). For manual on-screen digitizing of a printed map, a resolution of 200–300 dpi is usually sufficient, depending on the thickness of the thinnest lines. For man-

ual on-screen digitizing of aerial photographs, higher resolutions are recommended—typically, at least 800 dpi.

Semi-automatic/automatic digitizing requires a resolution that results in scanned lines of several pixels wide to enable the computer to trace the centre of the lines and thus avoid displacements. For printed maps, a resolution of 300–600 dpi is usually sufficient. Automatic or semi-automatic tracing from aerial photographs can only be done in a limited number of cases. Usually the information from aerial photos is obtained through visual interpretation.

After scanning, the resulting image can be improved by various image processing techniques. It is important to understand that scanning does not result in a structured data set of classified and coded objects. Additional work is required to recognize features and to associate categories and other thematic attributes with them.

Vectorization

The process of distilling points, lines and polygons from a scanned image is called vectorization. As scanned lines may be several pixels wide, they are often first thinned to retain only the centreline. The remaining centreline pixels are converted to series of (x, y) coordinate pairs, defining a polyline. Subsequently, features are formed and attributes are attached to them. This process may be entirely automated or performed semi-automatically, with the assistance of an operator. Pattern recognition methods—like Optical Character Recognition (OCR) for text—can be used for the automatic detection of graphic symbols and text.

Vectorization causes errors such as small spikes along lines, rounded corners, errors in T- and X-junctions, displaced lines or jagged curves. These errors are corrected in an automatic or interactive post-processing phase. The phases of the vectorization process are illustrated in Figure 8.32.

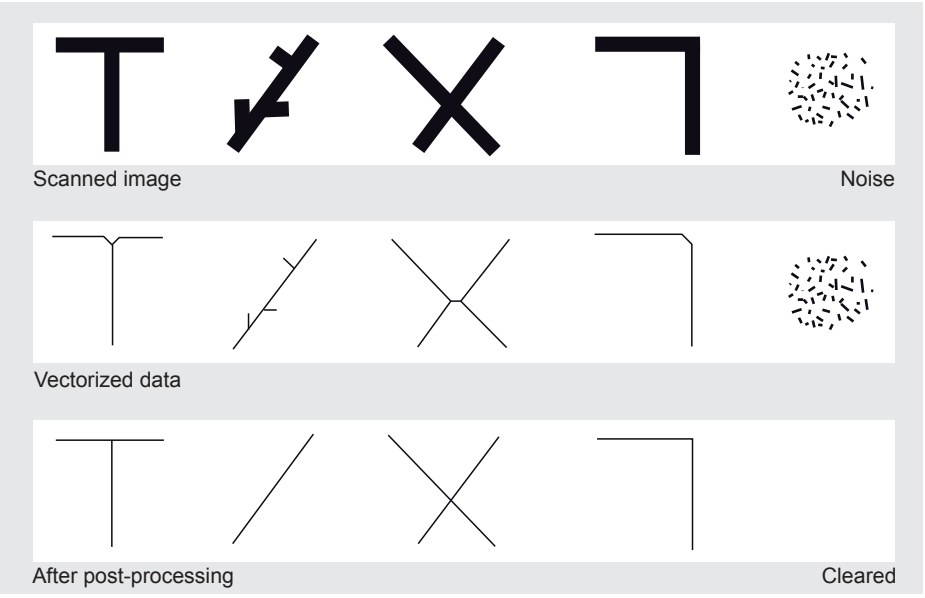


Figure 8.32
The phases of the vectorization process and various sorts of minor errors resulting from it. These are repaired in a post-processing phase.

Selecting a digitizing technique

The choice of digitizing technique depends on the quality, complexity and contents of the input document. Complex images are better manually digitized; simple images

are better automatically digitized. Images that are full of detail and symbols—such as topographic maps and aerial photographs—are therefore better digitized manually. Images that show only one type of information (e.g. elevation contours) are better automatically digitized.

In practice, the optimal choice may be a combination of methods. For example, contour-line film separations can be automatically digitized and used to produce a DEM. Existing topographic maps must be digitized manually, but new, geometrically corrected aerial photographs, with vector data from the topographic maps displayed directly over it, can be used for updating existing data files by means of manual on-screen digitizing.

8.2.6 Obtaining spatial data elsewhere

Over the past two decades, spatial data have been collected in digital form at an increasing rate and stored in various databases by the individual producers for their own use and for commercial purposes. More and more of these data are being shared among GIS users. There are several reasons for this. Some data are freely available, yet other data are only available commercially, as is the case for most satellite imagery. High quality data remain both costly and time consuming to collect and verify, as well as the fact that more and more GIS applications are looking at not just local, but national or even global, processes. As we will see in Section 8.4, new technologies have played a key role in the increasing availability of geospatial data. As a result of this increasing availability, we have to be more and more careful that the data we have acquired are of sufficient quality to be used in analyses and decision-making.

There are several related initiatives in the world to supply base data sets at national, regional and global levels, as well as those aiming to harmonize data models and definitions of existing data sets. Global initiatives include, for example, the Global Map, the USGS Global GIS database and the Second Administrative Level Boundaries (SALB) project. SALB, for instance, is a UN project aiming at improving the availability of information about administrative boundaries in developing countries.

Data formats and standards An important problem in any environment involved in digital data exchange is that of data formats and data standards. Different formats have been implemented by various GIS vendors, and different standards came about under different standardization committees. The phrase "data standard" refers to an agreed way, in terms of content, type and format, of representing data in a system. The good news about both formats and standards is that there are many to choose from; the bad news is that this can lead to a range of conversion problems. Several meta-data standards for digital spatial data exist, including those of the International Organization for Standardization (ISO) and the Open Geospatial Consortium (OGC).

8.3 Data preparation

Spatial data preparation aims to make acquired spatial data fit for use. Images may require enhancements and corrections of the classification scheme of the data. Vector data also may require editing, such as the trimming of line overshoots at intersections, deleting duplicate lines, closing gaps in lines, and generating polygons. Data may require conversion to either vector or raster formats to match other data sets that will be used in analyses. Additionally, the data preparation process includes associating attribute data with the spatial features through either manual input or reading digital attribute files into the GIS/DBMS.

The intended use of the acquired spatial data may require a less-detailed subset of the original data set, as only some of the features are relevant for subsequent analysis or subsequent map production. In these cases, data and/or cartographic generalization can be performed on the original data set.

This entire section treats a range of procedures for data checking, cleaning up, and integration to prepare vector data for analysis. Issues related to the preparation process of remote sensing data have already been discussed in Chapter 5.

8.3.1 Data checks and repairs

automatic and manual checking

Acquired data sets must be checked for quality in terms of the accuracy, consistency and completeness. Often, errors can be identified automatically, after which manual editing methods can be used to correct the errors. Alternatively, some software may identify and automatically correct certain types of errors. The geometric, topological, and attribute components of spatial data are discussed in the following subsections.

"Clean-up" operations are often performed in a standard sequence. For example, crossing lines are split before dangling lines are erased, and nodes are created at intersections before polygons are generated; see Figure 8.33.

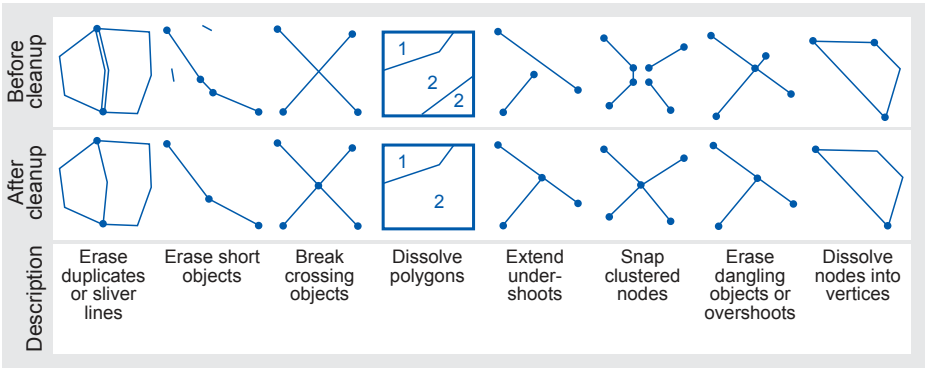


Figure 8.33
Clean-up operations for vector data.

With polygon data, one usually starts with many polylines (in an unwieldy format known as spaghetti data) that are combined and cleaned in the first step (Figure 8.34a–b). This results in fewer polylines with nodes being created at intersections. Then, polygons can be identified (Figure 8.34c). Sometimes, polylines that should connect to form closed boundaries do not, and must, therefore, be connected either manually or automatically. In a final step, the elementary topology of the polygons can be derived (Figure 8.34d).

Associating attributes Attributes may be automatically associated with features that have unique identifiers (Figure 8.35). In the case of vector data, attributes are

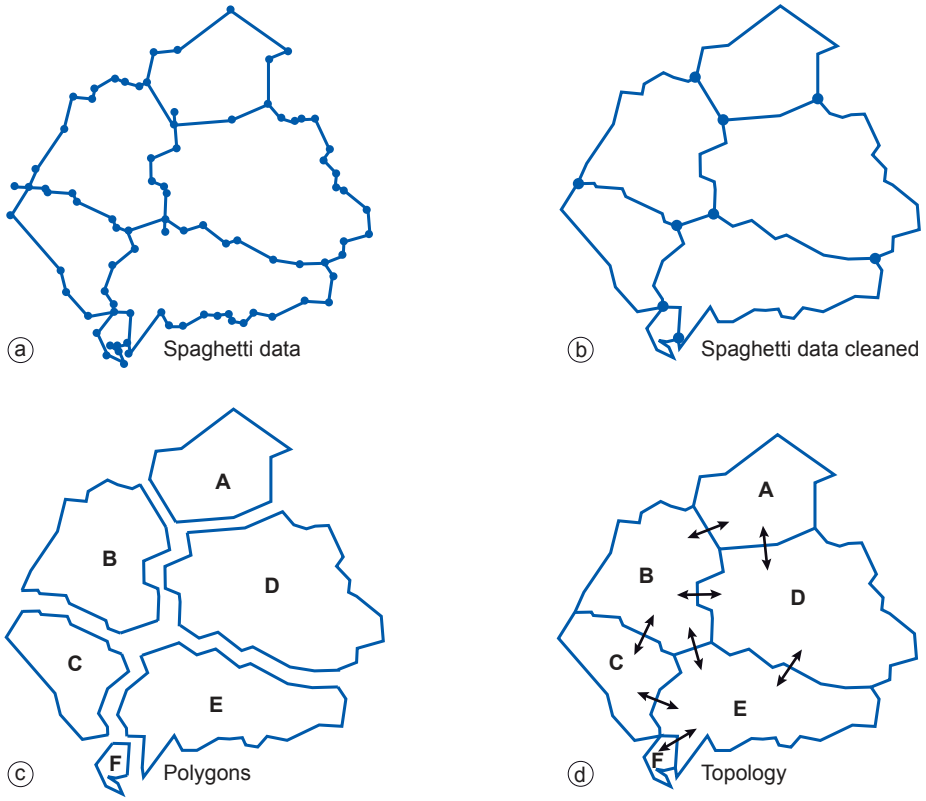


Figure 8.34
Successive clean-up operations for vector data, turning spaghetti data into topological structure.

assigned directly to the features, while in a raster the attributes are assigned to all cells that represent a feature.

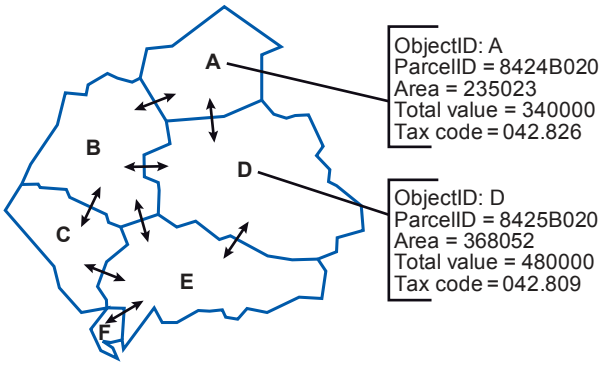


Figure 8.35
Attributes are associated with features that have unique identifiers.

It follows that, depending on the data type, assessment of attribute accuracy may range from a simple check on the labelling of features—for example, is a road classified as a metalled road actually surfaced or not?—to complex statistical procedures for assessing the accuracy of numerical data, such as the percentage of pollutants present in a soil.

Rasterization or vectorization Vectorization produces a vector data set from a raster. In some sense, we have looked at this already: namely in the production of

a vector set from a scanned image. Another form of vectorization is used when we want to identify features or patterns in remotely sensed images. The keywords here are feature extraction and pattern recognition, which are dealt with in Chapter 6.

If much or all of the subsequent spatial data analysis is to be carried out on raster data, one may want to convert vector data sets to raster data. This process, known as rasterization, involves assigning point, line and polygon attribute values to raster cells that overlap with the respective point, line or polygon. To avoid information loss, the raster resolution should be carefully chosen on the basis of the geometric resolution. A cell size that is too large may result in cells that cover parts of multiple vector features, and then ambiguity arises as to what value to assign to the cell. If, on the other hand, the cell size is too small, the file size of the raster may increase significantly.

Rasterization itself could be seen as a "backwards step": firstly, raster boundaries are only an approximation of the objects' original boundary. Secondly, the original "objects" can no longer be treated as such, as they have lost their topological properties. Rasterization is often done because it facilitates easier combination with other data sources that are also in raster formats, and/or because there are several analytical techniques that are easier to perform on raster data (see Chapter 9). An alternative to rasterization is to not perform it during the data preparation phase, but to use GIS rasterization functions "on the fly", i.e. when the computations call for it. This allows the vector data to be kept and raster data to be generated from them when needed. Obviously, the issue of performance trade-offs must be looked into.

Topology generation We have already discussed the derivation of elementary polygon topology starting from uncleaned polylines. However, more topological relations may sometimes be needed, as for instance in networks where questions of line connectivity, flow direction and which lines have overpasses and underpasses may need to be addressed. For polygons, questions that may arise involve polygon inclusion: is a polygon inside another one, or is the outer polygon simply around the inner polygon?

In addition to supporting a variety of analytical operations, topology can aid in data editing and in ensuring data quality. For example, adjacent polygons such as parcels have shared edges; they do not overlap, nor do they have gaps. Typically, topology rules are first defined (e.g. "there should be no gaps or overlap between polygons"), after which validation of the rules takes place. The topology errors can be identified automatically, to be followed by manual editing methods to correct the errors.

An alternative to storing topology together with features in the spatial database is to create topology on the fly, i.e. when the computations call for it. The created topology is temporary, only lasting for the duration of the editing session or analysis operation.

8.3.2 Combining data from multiple sources

A GIS project usually involves multiple data sets, so the next step addresses the issue of how these multiple sets relate to each other. The data sets may be of the same area but differ in accuracy, or they may be of adjacent areas, having been merged into a single data set, or the data sets may be of the same or adjacent areas but are referenced in different coordinate systems. Each of these situations is discussed in the following subsections.

Differences in accuracy Issues relating to positional error, attribute accuracy and temporal accuracy are clearly relevant in any combination of data sets, which may themselves have varying levels of accuracy.

Images come at a certain resolution, and printed maps at a certain scale. This typically

results in differences of resolution of acquired data sets, all the more since map features are sometimes intentionally displaced or in another way generalized to improve readability of the map. For instance, the course of a river will only be approximated roughly on a small-scale map, and a village on its northern bank should be depicted north of the river, even if this means it has to be displaced on the map a little bit. The small scale causes an accuracy error. If we want to combine a digitized version of that map with a digitized version of a large-scale map, we must be aware that features may not be where they seem to be. Analogous examples can be given for images of different resolutions.

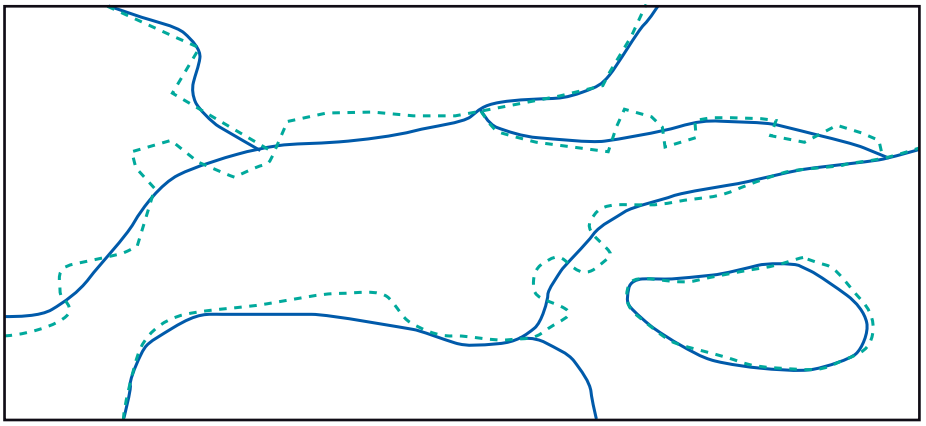


Figure 8.36
The integration of two vector data sets representing the same phenomenon may lead to sliver polygons.

In Figure 8.36, the polygons of two digitized maps at different scales are overlaid. Owing to scale differences in the sources, the resulting polygons do not perfectly coincide, and polygon boundaries cross each other. This causes small, artefact polygons in the overlay that are known as sliver polygons. If the map scales differ significantly, the polygon boundaries of the large-scale map should probably take priority, but when the differences are slight, we need interactive techniques to resolve any issues.

sliver polygons

There can be good reasons for having data sets at different scales. A good example is found in mapping organizations. European organizations maintain a single source database that contains the base data. This database is essentially scale-less and contains all data required for even the largest scale map to be produced. For each map scale that the mapping organization produces, they derive a separate database from the foundation data. Such a derived database may be called a cartographic database since the data stored are elements to be printed on a map, including, for instance, data on where to place name tags and what colour to give them. This may mean the organization has one database for the larger scale ranges (1:5000–1:10,000) and other databases for the smaller scale ranges; they maintain a multi-scale data environment.

More recent research has addressed the development of one database incorporating both larger and smaller scale ranges. Here we identify two main approaches: one approach to realize this is to store multiple representations of the same object in a multiple representation database. The database must keep track of links between different representations for the same object and must also provide support for decisions as to which representations to use in which situation. Another approach is to maintain one database for the larger scale ranges and derive representations for the smaller scale ranges on the fly. That means the data have to be generalized in real time. A combination of both approaches is to store multiple object representations for time-consuming generalization processes, which sometimes cannot be done fully automatically, and derive representations for the smaller scales in real time on the fly for rapid general-

ization processes.

Merging data sets of adjacent areas When individual data sets have been prepared as just described, they sometimes have to be integrated into a single, "seamless" data set, whilst ensuring that the appearance of the integrated geometry is as homogeneous as possible. Edge matching is the process of joining two or more map sheets, for instance, after they have been separately digitized.

edge matching

Merging adjacent data sets can be a major problem. Some GIS functions, such as line smoothing and data clean-up (removing duplicate lines) may have to be performed. Figure 8.37 illustrates a typical situation. Some GISs have merge or edge-matching functions to solve the problems arising from merging adjacent data. At map-sheet edges, feature representations have to be matched in order for them to be combined.

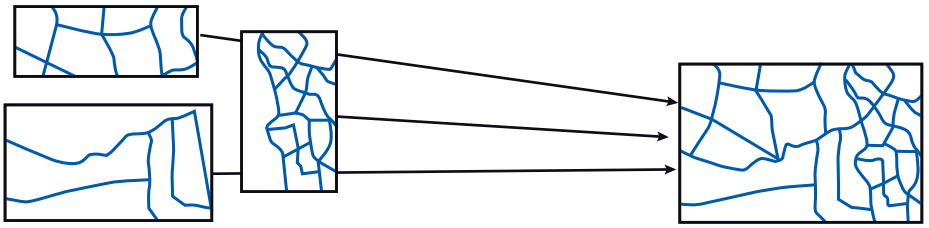


Figure 8.37
Multiple adjacent data sets, after cleaning, can be matched and merged into a single data set.

Coordinates of the objects along shared borders are adjusted to match those in the neighbouring data sets. Mismatches may still occur, so a visual check and interactive editing is likely to be required.

Differences in coordinate systems It may be the case that data layers that are to be combined or merged in some way are referenced in different coordinate systems, or are based upon different datums. As a result, data may need a coordinate transformation (Figure 3.22), or both a coordinate transformation and datum transformation (Figure 3.23). It may also be the case that data have been digitized from an existing map or data layer (Subsection 8.2.6). In this case, geometric transformations help to transform device coordinates (coordinates from digitizing tablets or screen coordinates) into world coordinates (geographic coordinates, metres, etc.).

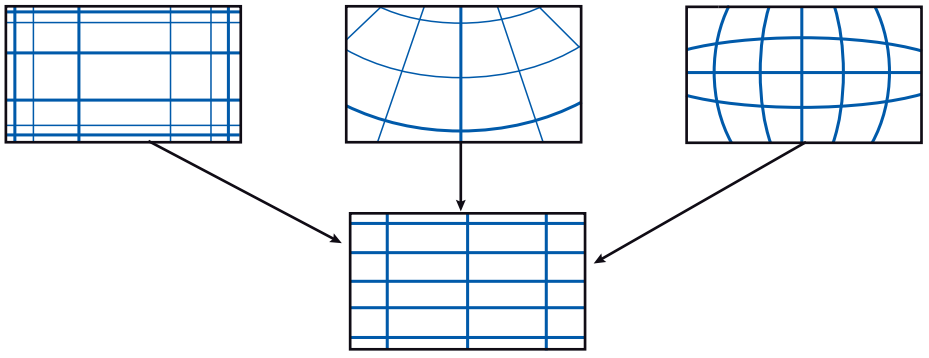


Figure 8.38
The integration of data sets into one common coordinate system.

Other data preparation functions A range of other data preparation functions exist that support conversion or adjustment of the acquired data to format requirements

that have been defined for data storage purposes. These include format transformation functions; functions to bring data sets in line with a common data-content model or a particular database schema; functions for removal of redundant vertices; data clipping operations; and so on. These functions and others discussed earlier are often performed in a standard sequence.

A good illustration of applying data preparation functions in a standard order is the SALB project. In this project, editing procedures were developed to clean the delimitation of the administrative units—to make them fit the international border standard—and prepare the attributes. These procedures are standardized to ensure the comparability between countries in terms of quality. The preparation functions for these procedures include changing formats and projections; adding missing delimitation of administrative boundaries; clipping the administrative boundaries to the international border; uniting multiple polygons; deleting redundant vertices; cleaning and checking attributes; building topology for the units; correcting gaps and overlaps between the units; and calculating attribute values such as areas and lengths.

8.4 Data management and processing systems

The ability to manage and process spatial data is a critical component for any functioning GIS. Simply put, data processing systems refer to hardware and software components that are able to process, store and transfer data. This section discusses the components of systems that facilitate the handling of spatial data and processing of geoinformation. To provide some context for the discussion, the section begins with a brief discussion of computer hardware and software trends over the past three decades.

In the subsections that follow, we then discuss database management systems (DBMSs) and illustrate some principles and methods of data extraction from a database. The final subsection of this section (Subsection 8.4.3) looks at the merging of GISs and DBMSs, and the emergence of spatial databases in recent years. It notes their key advantages, and briefly illustrates the use of a spatial database for data storage and processing. Before we deal with database aspects in detail, however, it is good to review the spatial data handling process, since it puts constraints on how we intend to organize our data and what kind of questions the system should be able to answer.

8.4.1 Stages of spatial data handling

Spatial data capture and preparation

Functions for capturing data are closely related to the disciplines of surveying engineering, photogrammetry, and remote sensing and the processes of digitizing. Remote sensing, in particular, is the field that provides photographs and images as the raw, base data from which spatial data sets are derived. Surveys of a study area often need to be conducted to collect data that cannot be obtained with remote sensing techniques, or to validate the data thus obtained. Traditional techniques for obtaining spatial data, typically from paper sources, included manual digitizing and scanning. In recent years there has been a significant increase in the availability and sharing of digital—geospatial—data. Various media and computer networks play an important role in the dissemination of these data, particularly the Internet.

The data, even though it has been obtained in a digital format, may still not be quite ready for use in the system. This may be because the format applied in the capturing process was not quite the format required for storage and further use; some type of data conversion is then required. In part, this problem may also arise when the captured data represent only raw, base data, from which the data objects of real interest to the system still need to be constructed. For example, semi-automatic digitizing may produce line segments, while the application requires non-overlapping polygons. A build and verification phase would then be needed to obtain these from the captured lines. Issues related to data acquisition and preparation are discussed in greater detail in the Sections 8.2 and 8.3.

Spatial data storage and maintenance

The way that data are stored plays a central role in their processing and, eventually, our understanding of them. In most available systems, spatial data are organized in layers by theme and/or scale. Examples are layers of thematic categories, such as land use, topography and administrative subdivisions, each according to their mapping scale. An important underlying principle is that a representation of the real world has to be designed such that it reflects phenomena and their relationships as naturally as possible. In a GIS, features are represented together with their attributes—geometric and non-geometric—and relationships. The geometry of features is represented with primitives of the respective dimension: a windmill probably as a point; an agricultural

field as a polygon. The primitives follow either the vector or the raster approach.

As discussed in Section 8.1, vector data types describe an object through its boundary, thus dividing the space into parts that are occupied by the respective objects. The raster approach subdivides space into (regular) cells, mostly as a square tessellation of two or three dimensions. These cells are called pixels in 2D and voxels in 3D. The data indicate for every cell which real-world feature is covered, provided the cell represents a discrete field. In the case of a continuous field, the cell holds a representative value for that field. Table 8.2 lists advantages and disadvantages of raster and vector representations.

Table 8.1
Raster and vector representations compared.

Raster representation	Vector representation
advantages	
• simple data structure • simple implementation of overlays • efficient for image processing	• efficient representation of topology • adapts well to scale changes • allows representing networks • allows easy association with attribute data
disadvantages	
• less compact data structure • difficulties in representing topology • cell boundaries independent of feature boundaries	• complex data structure • overlay more difficult to implement • inefficient for image processing • more update-intensive

The storage of a raster is, in principle, straightforward. It is stored in a file as a long list of values, one for each cell, preceded by a small list of extra data (the "file header"), which specifies how to interpret the long list. The order of the cell values in the list can, but need not necessarily, be left to right, top to bottom. This simple encoding scheme is known as row ordering. The header of the raster will typically specify how many rows and columns the raster has, which encoding scheme was used, and what sort of values are stored for each cell.

header file

Raster files can be large. For efficiency reasons, it is wise to organize the long list of cell values in such a way that spatially nearby-cells are also near to each other in the list. This is why other encoding schemes have been devised. The reader is referred to [65] for a more detailed discussion.

Low-level storage structures for vector data are much more complicated, and a discussion of this topic is beyond the scope of this textbook. The best intuitive understanding can be obtained from Figure 8.10, which illustrates a boundary model for polygon objects. Similar structures are in use for line objects. For further, advanced, reading see [102]. GIS packages support both spatial and attribute data, i.e. they accommodate spatial data storage using a vector approach and attribute data using tables. Historically, however, database management systems (DBMSs) have been based on the notion of tables for data storage.

GIS applications have been able to link to an external database to store attribute data and make use of its superior data management functions. Currently, all major GIS packages provide facilities to link with a DBMS and exchange attribute data with it. Spatial (vector) and attribute data are still sometimes stored in separate structures, although they can now be stored directly in a spatial database. More detail on these issues is provided in Subsection 8.4.2. Maintenance of data, spatial or otherwise, can best be defined as the combination of activities needed to keep the data set up to date

and as supportive as possible for the user community. It deals with obtaining new data and entering them into the system, as well as possibly replacing outdated data. The purpose is to have an up-to-date, stored data set available. After a major flood, for instance, we may have to update road-network data to reflect that roads have been washed away or have become otherwise impassable.

The need for updating spatial data originates from the requirements posed by the users, as well as the fact that many aspects of the real world change continuously. Data updates can take different forms. It may be that a completely new survey has been carried out, from which an entirely new data set will be derived, to replace the current set. This is typically the case if the spatial data originate from remote sensing, for example, from a new vegetation-cover set or from a new digital elevation model. Furthermore, local ground surveys may reveal local changes, such as new constructions or changes in land use or ownership. In such cases, local changes to a large spatial data set are typically required. Such local changes should take into account matters of data consistency, i.e. they should leave other spatial data within the same layer intact and correct.

Spatial query and analysis

The most characteristic parts of a GIS are its functions for spatial analysis, i.e. operators that use spatial data to derive new geoinformation. Spatial queries and process models play an important role in this functionality. One of the key uses of GISs has been to support spatial decision-making. Spatial decision-support systems (SDSSs) are a category of information systems composed of a database, GIS software, models, and a "knowledge engine" that allows users to deal specifically with location-related problems.

GIS functions are used for maintenance of the data and for analysing the data in order to infer spatial information. Analysis of spatial data can be defined as computing new information to provide new insights from existing spatial data. Consider an example from the domain of road construction. In mountainous areas, this is a complex engineering task with many cost factors, including the number of tunnels and bridges to be constructed, the total length of the tarmac, and the volume of rock and soil to be moved. GISs can help to compute such costs on the basis of an up-to-date digital elevation model and a soil map. Maintenance and analysis of attribute data is discussed further in Subsection 8.4.2. The exact nature of the analysis will depend on the application requirements, but computations and analytical functions can operate on both spatial and non-spatial data; Chapter 9 discusses these issues in more detail. For now, we will focus on the last stage of Figure 8.39, the presentation of spatial data.

Spatial data presentation

The presentation of spatial data, whether in print or on-screen, on maps or in tabular displays, or as "raw" data, is closely related to the discipline of cartography. The presentation may either be an end-product, for example a printed atlas, or an intermediate product, such as spatial data made available through the Internet. Table 8.2 lists several methods and devices used for the presentation of spatial data.

Cartography, information visualization and scientific visualization make use of these methods and devices in their products. Section 10.1 is devoted to visualization techniques for spatial data.

8.4.2 Database management systems

database

A database is a large, computerized collection of structured data. In the non-spatial domain, databases have been in use since the 1960s for various purposes, such as bank

Table 8.2
Spatial data presentation.

Method	Devices
Hardcopy	• printer • plotter (pen plotter, ink-jet printer, thermal transfer printer, electrostatic plotter) • film writer
Soft copy	• computer screen
Output of digital data sets	• magnetic tape • CD-ROM or DVD • the Internet

account administration, stock monitoring, salary administration, sales and purchasing administration and flight reservation systems. These applications have in common that the amount of data is quite large, but the data themselves have a simple and regular structure. Designing a database is not an easy task. First, one has to consider carefully what the purpose of the database is and who its users will be. Second, one needs to identify the available data sources and define the format in which the data will be organized within the database. This format is usually called the database structure. Only when all this is in place can data be entered into the database. Data must be kept up to date and it is, therefore, wise to set up the processes for doing this and to make someone responsible for regular maintenance. Documentation of the database design and set up is crucial for an extended database life (proprietary databases tend to outlive the professional careers of their original designers).

A database management system (DBMS) is a software package that allows the user to set up, use and maintain a database. Just as a GIS allows the set up of a GIS application, a DBMS offers generic functionality for database organization and data handling. In the next subsection we take a closer look at what type of functions are offered by DBMSs. Many standard PCs are equipped with a DBMS called Microsoft Access. This package offers a useful set of functions and the capacity to store terabytes of information.

DBMS

Reasons for using a DBMS

There are various reasons why one would want to use a DBMS for data storage and processing:

- A DBMS supports the storage and manipulation of very large data sets. Some data sets are so big that storing them in text files or spreadsheet files becomes too awkward for practical use. The result may be that finding simple facts takes minutes, and performing simple calculations perhaps even hours. A DBMS is specifically designed for these purposes.

- A DBMS can be instructed to guard data correctness. For instance, an important aspect of data correctness is data-entry checking: ensuring that the data that are entered into the database do not contain obvious errors. For instance, since we know in what study area we are working, we also know the range of possible geographic coordinates, so we can ensure the DBMS checks them upon entry. This is a simple example of the type of rules, generally known as integrity constraints, that can be defined in, and automatically checked by, a DBMS. More complex integrity constraints are certainly possible; their definition is is an aspect of the database design.

- A DBMS supports the concurrent use of the same data set by many users. Large data sets are often built up over time. As a result, substantial investments are required to create and maintain them, and probably many people are involved in the data collection, maintenance and processing. Such data sets are often considered to have high strategic value by their owner(s) and many people may want to use them within an organization. Moreover, different users may have different views about the data. As a consequence, users will be under the impression that they are operating on their personal database and not on one shared by many people. They may all be using the database at the same time without affecting each other's activities. This DBMS function is referred to as concurrency control.
- A DBMS provides users with a high-level, declarative query language, with as its most important use the formulation of queries.
- A DBMS supports the use of a data model, which is a language with which one can define a database structure and manipulate the data stored in it. The most prominent data model is the relational data model; this is discussed in full in Subsection 8.4.3. Its primitives are tuples (also known as records, or rows) with attribute values, and relations, which are sets of similarly formed tuples.
- A DBMS includes data backup and recovery functions, to ensure data availability at all times. As potentially many users rely on the availability of the data, the data must be safeguarded against possible calamities. Regular backups of the data set and automatic recovery schemes provide insurance against loss of data.
- A DBMS allows the control of data redundancy. A well-designed database takes care of storing single facts only once. Storing a fact several times—a phenomenon known as data redundancy—can lead to situations in which stored facts may contradict each other, causing reduced usefulness of the data. Redundancy is, however, not necessarily always problematic, as long as we specify where it occurs so that it can be controlled.

Alternatives for data management

The decision whether or not to use a DBMS will depend, among other things, on how much data there are or will be, what type of use will be made of it, and how many users might be involved. On the small-scale side of the spectrum—when the data set is small, its use is relatively simple, and there is just one user—we might use simple text files and a word processor. Think of a personal address book as an example or a small set of simple field observations. Text files offer no support for data analysis, except perhaps in alphabetical sorting. If our data set is still small and numeric in nature, and we have a single type of use in mind, a spreadsheet program might suffice. This might also be the case if we have a number of field observations with measurements that we want to prepare for statistical analysis.

If, however, we carry out region- or nation-wide censuses, with many observation stations and/or field observers and all sorts of different measurements, one quickly needs a database to keep track of all the data. Spreadsheet programs are generally not suitable for this, however, as they do not accommodate concurrent use of data sets well, although they do support some data analysis, especially when it comes to calculations for a single table, such as averages, sums, minimum and maximum values.

All such computations are usually restricted to a single table of data. When one wants to relate the values in the table with values of another nature in some other table, skilful expertise and significant amounts of time may be required to achieve this.

PrivatePerson	TaxId	Surname	BirthDate
	101-367	Garcia	10/05/1952
	134-788	Chen	26/01/1964
	101-490	Fakolo	14/09/1931

Parcel	PId	Location	AreaSize
	3421	2001	435
	8871	1462	550
	2109	2323	1040
	1515	2003	245

TitleDeed	Plot	Owner	DeedDate
2109	101-367	18/12/1996	
8871	101-490	10/01/1984	
1515	134-788	01/09/1991	
3421	101-367	25/09/1996	

Figure 8.39
An example of a small database consisting of three relations (tables), all with three attributes, and three, four and four tuples, respectively. PrivatePerson / Parcel / TitleDeed are the names of the three tables. Surname is an attribute of the PrivatePerson table; the Surname attribute value for person with TaxId '101-367' is 'Garcia'.

Relational data models

relational data model

For relational data models, the structures used to define the database are attributes, tuples and relations. Computer programs either perform data extraction from the database without altering it, in which case they are termed queries, or they change the database contents, in which case we speak of updates or transactions. The technical terms related to database technology are defined below. An extremely small database selected from a cadastral setting is illustrated in Figure 8.39. This database consists of three tables, one for storing people's details, one for storing land-parcel details and a third for storing details concerning title deeds. Various sources of information are kept in the database such as a taxation identifier (TaxId) for people, a parcel identifier (PId) for land parcels, and the date of a title deed (DeedDate).

PrivatePerson	(TaxId : string, Surname : string, Birthdate : date)
Parcel	(Pid : number, Location : polygon, AreaSize : number)
TitleDeed	(Plot : number, Owner : string, DeedDate : date)

Table 8.3
The relation schemas for the three tables of the database in Figure 8.39.

Relations, tuples and attributes In relational data models, a database is viewed as a collection of relations, also commonly referred to as tables.

relation

A data model is a language that allows the definition of:

- the structures that will be used to store the base data;
- the integrity constraints that the stored data have to obey at all moments in time;
- the computer programs used to manipulate the data.

A table or relation is itself a collection of tuples (or records). In fact, each table is a collection of tuples that are similarly shaped. By this, we mean that a tuple has a fixed number of named fields (also known as attributes). All tuples in the same relation have the same named fields. In a diagram, such as in Figure 8.39, relations can be displayed as data in tabular form, as the relations provided in the figure demonstrate. The PrivatePerson table has three tuples; the Surname attribute value for the first tuple shown is "Garcia."

tuple

attribute domain

The phrase "that are similarly shaped" takes this a bit further. It requires that all values for the same attribute come from a single domain of values. An attribute's domain is a (possibly infinite) set of atomic values such as, for example, the set of integer number values or the set of real number values. In our cadastral database example, the domain of the Surname attribute, for instance, is a string, so any surname is represented as a sequence of text characters, i.e. as a string. The availability of other domains depends on the DBMS, but usually integer (the whole numbers), real (all numbers), date, yes/no and a few more are included. When a relation is created, we need to indicate what type of tuples it will store. This means that we must:

1. provide a name for the relation;
2. indicate which attributes it will have;
3. set the domain of each attribute.

relation schema

attribute

A relation definition obtained in this way is known as the relation schema of that relation. The definition of a relation schema is an important part of a database. An attribute is a named field of a tuple, with which each tuple associates a value, the tuple's attribute value. Our example database has three relation schemas; one of which is TitleDeed. The relation schemas together make up the database schema. The relation schemas for the database of Figure 8.39are given in Table 8.3. Underlined attributes (and their domains) indicate the primary key of the relation, which will be defined and discussed below.

Relation schemas are stable and will rarely change over time. This is not true of the tuples stored in tables: typically, they are often changing, either because new tuples are added or others are removed, or still others will undergo changes in their attribute values. The set of tuples in a relation at some point in time is called the relation instance at that moment. This tuple set is always finite: you can count how many tuples there are. Figure 8.39 gives us a single database instance, i.e. one relation instance for each relation. One of the relation instances has three tuples, two of them have four. Any relation instance always contains only tuples that comply with the relation schema of the relation.

Finding tuples and building links between them We have already indicated that database systems are particularly good at storing large quantities of data. (Our example database is not even small, it is tiny!) The DBMS must support rapid searching among many tuples. This is why relational data models use the notion of a key. In other words, if we have a value for each of the key attributes we are guaranteed to find no more than one tuple in the table with that combination of values; it remains possible that there is no tuple for the given combination. In our example database, the set TaxId, Surname is a key of the relation PrivatePerson: if we know both a TaxId and a Surname value, we will find at most one tuple with that combination of values.

Every relation has a key, though possibly it is the combination of all attributes. Such a large key is, however, not handy because we must provide a value for each of its attributes when we search for tuples. Clearly, we want a key to have as few as possible attributes: the fewer, the better.

A key of a relation comprises one or more attributes. A value for these attributes uniquely identifies a tuple.

key

If a key has just one attribute, it obviously cannot have less attributes. Some keys have two attributes; an example is the key Plot, Owner of relation TitleDeed. We need

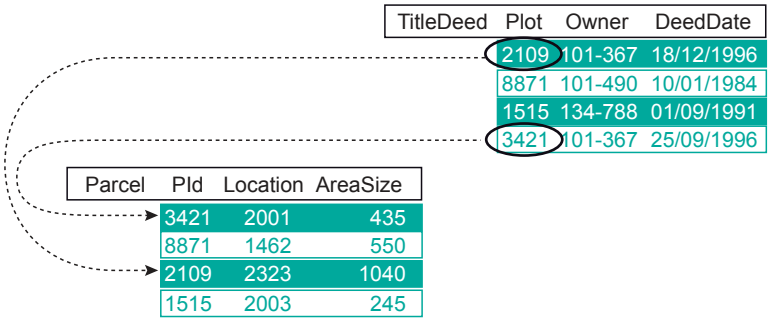


Figure 8.40
The table TitleDeed has a foreign key in its attribute Plot. This attribute refers to key values of the Parcel relation, as indicated for two TitleDeed tuples. The table TitleDeed actually has a second foreign key in the attribute Owner, which refers to PrivatePerson tuples.

both attributes because there can be many title deeds for a single plot (in the case of plots that are sold often), but also many title deeds for a single person (say, in the case of wealthy persons). When we provide a value for a key, we can look up the corresponding tuple in the table (if such a tuple exists). A tuple can refer to another tuple by storing that other tuple's key value. For instance, a TitleDeed tuple refers to a Parcel tuple by including that tuple's key value. The TitleDeed table has a special attribute Plot for storing such values. The Plot attribute is called a foreign key because it refers to the primary key (Pid) of another relation (Parcel). This is illustrated in Figure 8.41.

foreign key

Two tuples of the same relation instance can have identical foreign key values: for instance, two TitleDeed tuples may refer to the same Parcel tuple. A foreign key is, therefore, not a key of the relation in which it appears, despite its name! A foreign key must have as many attributes as the primary key that it refers to.

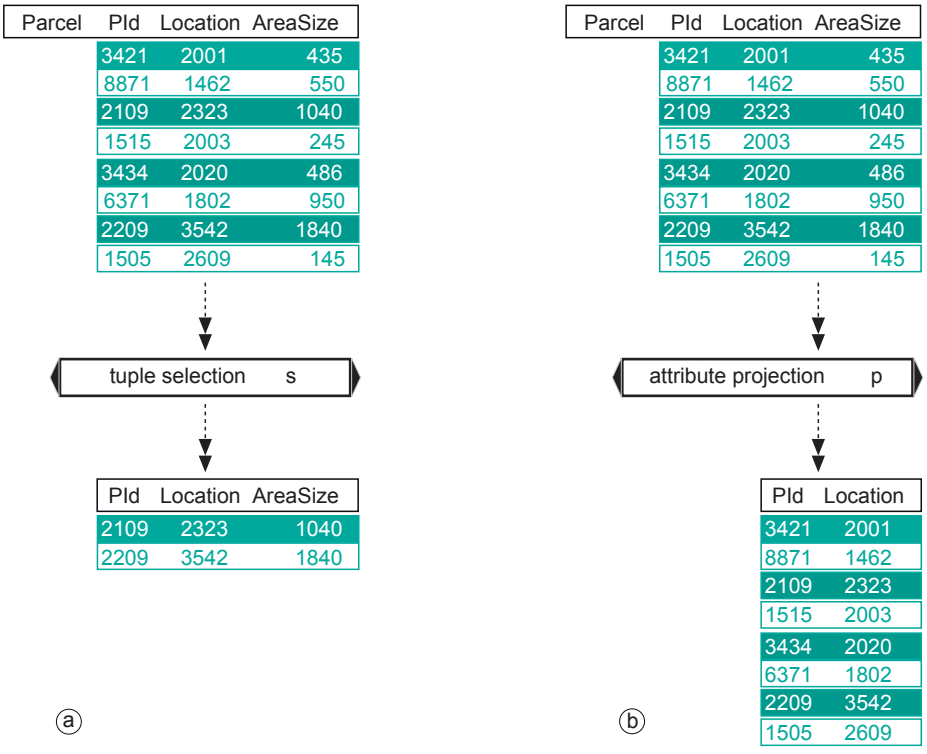


Figure 8.41
The two unary query operators: (a) tuple selection has a single table as input and produces another table with less tuples. Here, the condition was that AreaSize must be over 1000; (b) attribute projection has a single table as input and produces another table with fewer attributes. Here, the projection is onto the attributes PId and Location.

Querying a relational database

We will now look at the three most elementary query operators. These are quite powerful because they can be combined to define queries of higher complexity. The three query operators have some common traits. First, all of them require input and produce output, and both input and output are relations! This guarantees that the output of one query (a relation) can be the input of another query, which makes it possible to build more and more complex queries—if we want to.

The first query operator is called tuple selection. Tuple selection works like a filter: it allows tuples that meet the selection condition to pass and disallows tuples that do not meet the condition; see Figure 8.41a. The operator is given some input relation, as well as a selection condition about tuples in the input relation. A selection condition is a truth statement about a tuple's attribute values, such as AreaSize $>$ 1000. For some tuples in Parcel, this statement will be true and for others it will be false. Tuple selection on the Parcel relation with this condition will result in a set of Parcel tuples for which the condition is true.

A second operator, called attribute projection, is also illustrated in Figure 8.41. Besides an input relation, this operator requires a list of attributes, all of which should be attributes of the schema of the input relation. The output relation of this operator has as its schema only the list of attributes given, so we say that the operator projects onto these attributes. Contrary to the first operator, which produces fewer tuples, this operator produces fewer attributes compared to the input relation.

SQL

The most common operator for defining queries in a relational database is the language SQL, which stands for Structured Query Language. The two queries of Figure 8.41 would be written in SQL as follows:

```
SELECT  *
FROM    Parcel
WHERE   AreaSize > 1000
```

(a) tuple selection from the Parcel relation, using the condition AreaSize > 1000. The * indicates that we want to extract all attributes of the input relation.

```
SELECT  Pid, Location
FROM    Parcel
```

(b) attribute projection from the Parcel relation. The SELECT clause indicates that we only want to extract the two attributes Pid and Location. There is no WHERE-clause in this query.

Attribute projection works like a tuple formatter: it passes through all tuples of the input, and reshapes each of them in the same way.

Queries like the two above do not create stored tables in the database. This is why the result tables have no name: they are virtual tables. The result of a query is a table that is shown to the user who executed the query. Whenever the user closes her/his view on the query result, that result is lost. The SQL code for the query is, however, stored for future use. The user can re-execute the query again to obtain a view on the result once more.

SQL differs from the other two query languages in that it requires two input relations. The operator is called the join and is illustrated in Figure 8.42.

The output relation of this operator has as attributes those of the first and the second input relations. The number of attributes therefore increases. The output tuples are obtained by taking a tuple from the first input relation and "gluing" it to a tuple from the second input relation. The join operator uses a condition that expresses which tuples from the first relation are combined ('glued') with which tuples from the second. The example of Figure 8.42 combines TitleDeed tuples with Parcel tuples, but only

those for which the foreign key Plot matches with primary key PId.

The join operator takes two input relations and produces one output relation, gluing two tuples together (one from each input relation) to form a bigger tuple—provided they meet a specified condition.

join operator

The join query for our example is easily expressed in SQL as:

```
SELECT ?
FROM TitleDeed, Parcel
WHERE TitleDeed.Plot = Parcel.PId
```

The FROM clause identifies the two input relations; the WHERE clause states the join condition.

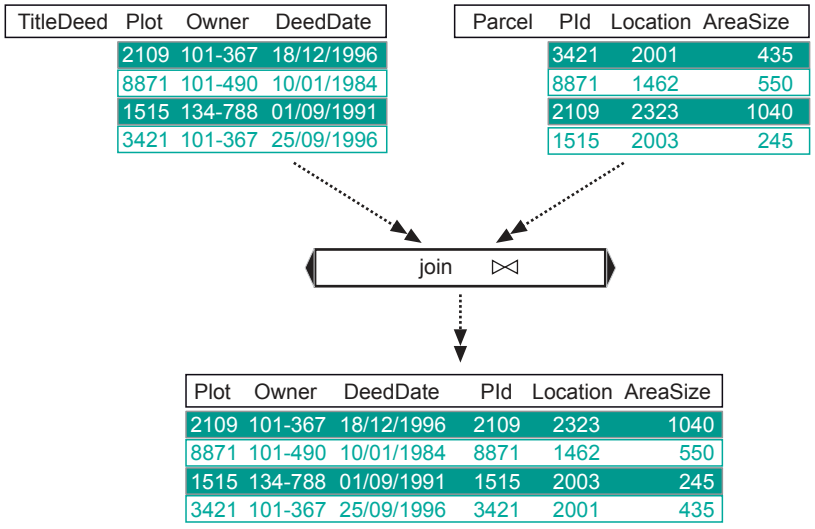


Figure 8.42
The essential binary query operator: join. The join condition for this example is TitleDeed.Plot = ParcelPid, which expresses a foreign key/key link between TitleDeed and Parcel. The result relation has $3 + 3 = 6$ attributes.

It is often not sufficient to use just one operator for extracting sensible information from a database. The strength of the above operators is hidden in the fact that they can be combined to produce more advanced and useful query definitions. A final example illustrates this. Take another look at the join of Figure 8.42. Suppose we really wanted to obtain combined TitleDeed/Parcel information, but only for parcels with a size over 1000, and we only wanted to see the owner identifier and deed date of such title deeds.

We can take the result of the join above and select the tuples that show a parcel size over 1000. The result of this tuple selection can then be taken as the input for an attribute selection that only leaves Owner and DeedDate. This is illustrated in Figure 8.43.

Finally, we may look at the SQL statement that would give us the query of Figure 8.43. It can be written as:

```
SELECT Owner
DeedDate FROM TitleDeed
Parcel WHERE TitleDeed.Plot = Parcel.PId AND AreaSize > 1000
```

8.4.3 GISs and spatial databases

Linking GISs and DBMSs

GIS software provides support for spatial data and thematic or attribute data. GISs have traditionally stored spatial data and attribute data separately. This required the GIS to provide a link between the spatial data (represented with rasters or vectors), and their non-spatial attribute data. The strength of GIS technology lies in its built-in "understanding" of geographic space and all functions that derive from this, for purposes such as storage, analysis and map production. GIS packages themselves can store tabular data, but they do not always provide a fully-fledged query language to operate on the tables.

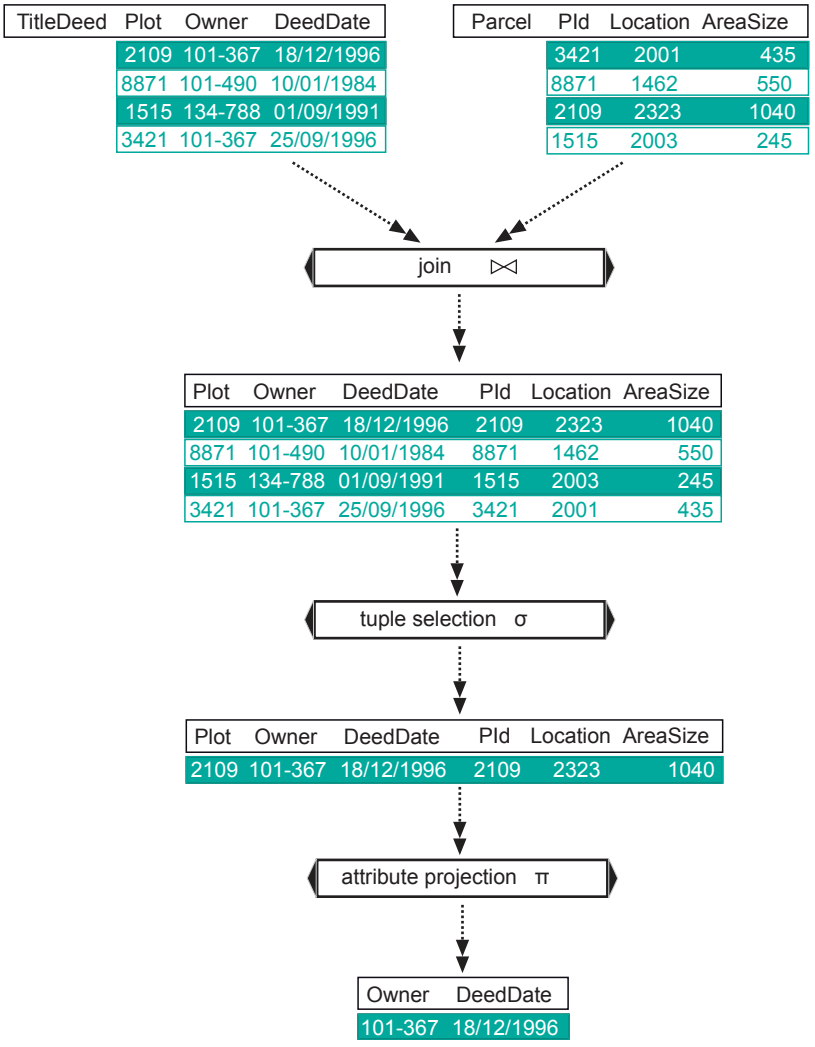


Figure 8.43
A combined selection/projection/join query for selecting owners and deed dates for parcels with a size larger than 1000. The join is done first, then follows a tuple selection on the resulting tuples of the join, which is completed by an attribute projection.

DBMSs have a long tradition in handling attribute (i.e. administrative, non-spatial, tabular, thematic) data in a secure way for multiple users at the same time. Arguably, DBMSs offer much better table functionality, since they are specifically designed for this purpose. Many data in GIS applications are attribute data, so it made sense to use a DBMS for it. For this reason, many GIS applications have made use of external

DBMSs for data support. In this role, the DBMS serves as a centralized data repository for all users, while each user runs her/his own GIS software, which obtains its data from the DBMS. This means that the GIS has to link the spatial data represented by rasters or vectors and the attribute data stored in an external DBMS.

With raster representations, each raster cell stores a characteristic value. This value can be used to look up attribute data in an accompanying database table. For instance, the land use raster of Figure 8.44 indicates the land use class for each of its cells, while an accompanying table provides full descriptions for all classes, perhaps including some statistical information for each of the types. Note the similarity with the key/foreign key concept in relational databases.

With vector representations, our spatial objects—whether they are points, lines or polygons—are automatically given a unique identifier by the system. This identifier is usually just called the *object ID* or feature ID and is used to link the spatial object (as represented by vectors) with its attribute data in an attribute table. The principle applied here is similar to that in raster settings, but in this case each object has its own identifier. The ID in the vector system functions as a key, and any reference to an ID value in the attribute database is a foreign key reference to the vector system. For example, in Figure 8.45, Parcel is a table with attributes, linked to the spatial objects stored in a GIS by the Location column. Obviously, several tables may make references to the vector system, but it is not uncommon to have some main table for which the ID is actually also the key.

object ID

linking objects and tables

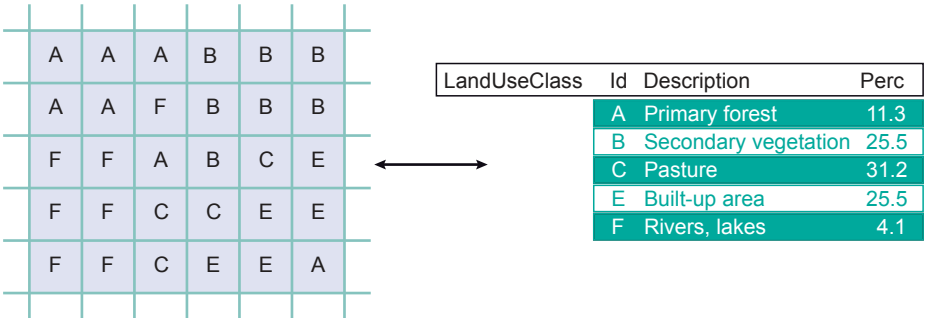


Figure 8.44
A raster representing land use and a related table providing full text descriptions (amongst other things) of each land use class.

Spatial database functionality

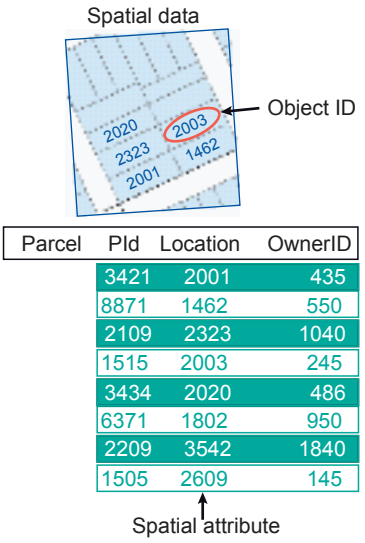
DBMS vendors have over the last 20 years recognized the need for storing more complex data, such as spatial data. The main problem was that additional functionality was needed by DBMSs in order to process and manage spatial data. As the capabilities of computer hardware to process information have increased, so too has the desire for better ways of representing and managing spatial data. During the 1990s, object-oriented and object-relational data models were developed for just this purpose. These extend standard relational models by providing support for objects, including "spatial" ones.

Currently, GIS software packages are able to store spatial data using a range of commercial and open-source DBMSs (e.g. Oracle, Informix, IBM DB2, Sybase, and PostGres) with the help of spatial extensions. Some GIS software have integrated database "engines" and therefore do not need these extensions. ESRI's ArcGIS, for example, has the main components of the Microsoft Access database software built in. This means that a designer of a GIS application can choose whether to store the application data in the GIS or in the DBMS.

integrated database "engines"

DBMS spatial extension

Spatial databases, also known as geo-databases, are implemented directly on existing



Parcel	PId	Location	OwnerID
3421	2001		435
8871	1462		550
2109	2323		1040
1515	2003		245
3434	2020		486
6371	1802		950
2209	3542		1840
1505	2609		145

Spatial attribute

Figure 8.45
Storage and linking of vector attribute data between a GIS and a DBMS.

DBMSs using extension software to allow them to handle spatial objects.

There are several advantages in doing this, as we will see below. Put simply, spatial data can be stored in a special database column, referred to as the "geometry" or "feature" or "shape data type", depending on the specific software package. This means GISs can rely fully on a DBMS support for spatial data, making use of a DBMS for data query and storage (and multi-user support), and a GIS for spatial functionality. Small-scale GIS applications may not require a multi-user capability; these applications could be supported by spatial data support from a personal database.

Parcel	PId	Geometry	OwnerID
3421	"MULTIPOLYGON(((257462.704979333 464780.750851061,257463.89798...)))"		435
8871	"MULTIPOLYGON(((257409.813950544 464789.91585049,257407.896903...)))"		550
2109	"MULTIPOLYGON(((257785.714911912 464796.839972167,257782.59794...)))"		1040
1515	"MULTIPOLYGON(((257790.672100448 464807.13792585,257788.608078...)))"		245
3434	"MULTIPOLYGON(((257435.527950478 464803.92887633,257428.254887...)))"		486
6371	"MULTIPOLYGON(((257432.476077854 464813.848852072,257433.147910...)))"		950
2209	"MULTIPOLYGON(((257444.888027332 464826.555046319,257446.43201...)))"		1840
1505	"MULTIPOLYGON(((256293.760107491 464935.203846095,256292. 00881...)))"		145

Figure 8.46
Geometry data stored directly in a spatial database table.

A spatial database allows a wide variety of users to access large data sets (both geographic and alphanumeric) and manage their relations, while guaranteeing their integrity. The Open Geospatial Consortium (OGC) has released a series of standards for geographic data formats that (among other things), define:

- which tables must be present in a geo-database (i.e. a geometry columns table and a spatial reference system table);
- the data formats, called "Simple Features" (i.e. point, line, polygon, etc.);
- a set of SQL-like instructions for geographic analysis.

The architecture of a spatial database differs from a standard relational DBMS not only because it can handle geometry data and manage projections, but also because

of the availability a larger set of commands that extend the standard SQL language (distance calculations, buffers, overlay, conversion between coordinate systems, etc.). A geo-database must provide a link between the spatial data represented by rasters or vectors and their non-spatial attribute data.

The capabilities of spatial databases will continue to evolve over time. Currently, ESRI's ArcGIS "Geodatabase" can store topological relationships directly in the database, providing support for different kinds of features (objects) and their behaviour (relations with other objects), as well as ways to validate these relations and behaviours. Effectively, this is the same type of functionality offered by traditional DBMSs, but with geospatial data. Currently, some spatial database packages, such as PostGIS, have full 3D support, as opposed to the 2D support offered by many.

storing topology

Querying a spatial database A spatial DBMS provides support for geographic coordinate systems and transformations. It will also provide storage of the relationships between features, including the creation and storage of topological relationships. As a result, one is able to use functions for "spatial query" (exploring spatial relationships). To illustrate, a spatial query using SQL to find all the Thai restaurants within 2 km of a given hotel would look like:

spatial query

```
SELECT R.Name
FROM Restaurants AS R,
Hotels as H
WHERE R.Type = Thai AND
H.name = Hilton AND
Intersect(R.Geometry, Buffer(H.Geometry, 2))
```

The Intersect command creates a spatial join between restaurants and hotels. The Geometry column carries the spatial data. It is likely that in the near future all spatial data will be stored directly in spatial databases.

8.5 GIS Working environment

8.5.1 Spatial Data Infrastructure (SDI)

The way in which spatial data are perceived, expected, and consumed by users in their applications depends, to a large extent, on the current context and shape of technology, projects and markets. Interactions between these three drivers form the basis for the requirements of geoinformation systems at any given time. At present, these interactions translate into systems having to operate in an interconnected environment.

As the systems that rely on spatial data have moved from single, separate working environments towards connected and cooperative environments, different needs, requirements and challenges have emerged. To address these changes, the spatial information community came up with the Spatial Data Infrastructure (SDI)initiative. In [80] an SDI is defined as *the relevant base collection of technologies, policies and institutional arrangements that facilitate the availability of and access to spatial data*. Several definitions of an SDI exist, however, each adjusted slightly to fit specific needs (see [80]). Regardless of the author or the context, the issue comes down to one objective: interoperability, i.e. the property of diverse systems and organizations that allows them to work together, to inter-operate. The targeted objective of an SDI is, therefore, seamless access to all the constituent elements of a geoinformation system: data, operations, and results. These three elements are collectively called "geo-resources". "Seamless" here means transparently over a network, regardless of computer platform, format or application. Central to this objective are standards.

An SDI is not an entity in itself; it is rather an approach for working efficiently and effectively in a distributed, cooperative environment. There is, therefore, no recipe for the implementation of an SDI. Through the years, experts have come up with different interpretations of the concept and, therefore, different SDI implementations have been created too. The most familiar approach for implementation is based on the notion of a clearinghouse: i.e.épository to store descriptions of existing spatial data. These descriptions, known as meta-data, are created and stored in a standardized format. A clearinghouse allows spatial data producers to publish and disseminate meta-data, which in turn can be queried by users to discover spatial data resources. This approach describes the first generation of SDIs, and it was the way to go about implementing them in the early nineties. This could be achieved given the standards available and the maturity of the geoinformation technology of the day. The latest generation of SDI implementations focuses on geo-services. It is based on sounder standards and more robust technology. It uses webservices as a mechanism to provide access to geo-resources. The following subsections describe the developments that are considered to be state of the art in the realm of SDI.

8.5.2 Standards

The underlying working principle of an SDI based on webservices is that it operates on the World Wide Web, also known as the Web. Terms like the Internet and the World Wide Web are often used interchangeably, however they are not one and the same. The Internet is a network, or rather a global system of interconnected *computer networks* that use the standardized Internet Protocol Suite (TCP/IP) to carry a wide range of information resources and services. The most well known application built on top of the Internet is the Web. The Web is a system of interlinked *documents* connected by means of hyperlinks and accessible via the Internet. Other internet-based applications include electronic mail, file transfer, social networking, and multi-player gaming.

In line with this working principle, developers of SDIs have to adhere to two sets of technical standards. The first is the set of technical specifications and guidelines on

which the Web is based. The second is the set of technical specifications that address interoperability issues among geo-resources. The standards for the Web are developed by the World Wide Web Consortium (W3C) [85]. The W3C is an international community, led by Web inventor Tim Berners-Lee, that develops the standards needed to ensure the long-term growth of the Web. Standards for interoperability of geo-resources address a multitude of issues ranging from data capture to presentation and are developed by different organizations, the most of which prominent are the International Organization for Standardization (ISO), through the technical committee ISO/TC 211 [48], and the Open Geospatial Consortium (OGC) [85].

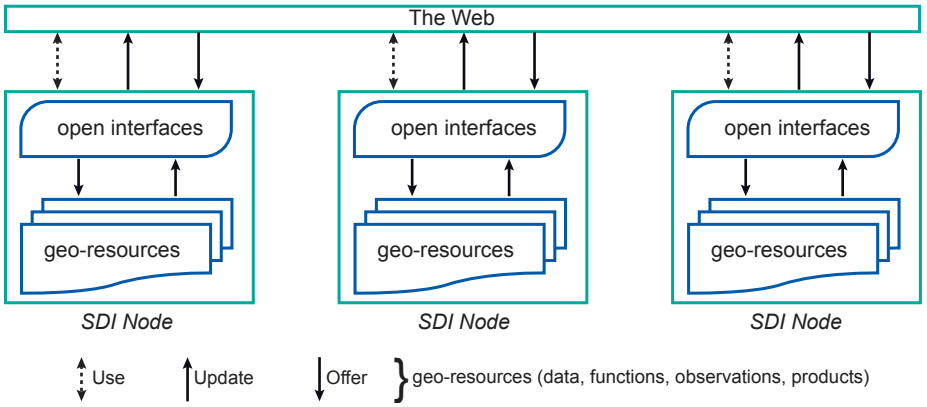


Figure 8.47
Schematic representation of an SDI.

The role of standards in an SDI should only be to support its main objective, i.e. interoperability. In this context, standards are deployed at the interface layer between available geo-resources and their users. Standards that go beyond this purpose, for example data models, should only be used as reference standards in SDI development. SDI participants need to understand the benefits and limitations of this distinction.

Following this paradigm, OGC's standardization efforts provide a comprehensive suite of open interface specifications. An interface is defined as a connection point where two separate system components interact to exchange parameters and instructions. An interface is presented as an ordered set of parameters (with specific names and data types) and instructions (with specific names and functions) for components to interact. An interface that allows any two arbitrary systems (system components) to interact is known as an open interface. Systems that use this type of interface are considered open systems. As a result, an open system is permeable to its environment and can produce a large (potentially infinite) set of services as a result of its interactions. In contrast, a closed system delivers a limited number of services and its sources are only those contained within its boundaries. The relevant OGC standards and how they are used within an SDI are explained in section 8.5.4.

8.5.3 SDI architecture

From an architectural point of view, an SDI is a collaborative network of disparate systems called SDI nodes. An SDI node is a moderately to highly complex information system, usually of long life expectancy, in which geo-resources feature rather prominently. SDI nodes are totally independent systems, which means that they also have to fulfill requirements that are different from those of the SDI itself.

We can argue that we have an SDI in place when we have SDI nodes that host geo-resources that can be found, used and maintained, or even created, by other SDI nodes. Figure 8.48 depicts an schematic view on the components of such an SDI. For the design of an SDI that fulfills the criteria mentioned above, three perspectives need to be

addressed. First, the vertical perspective that focuses on the design of SDI nodes as state-and-function systems that hold data and provide functions in the form of services for operating on that data. Second, the horizontal perspective that focuses on how to specify the communication patterns between SDI nodes, abstractly constructing a larger system, i.e. the SDI. And third, the introspective perspective that focuses on augmenting the specified SDI system with the means to query itself for service possibilities and to allow the dynamic creation of new services.

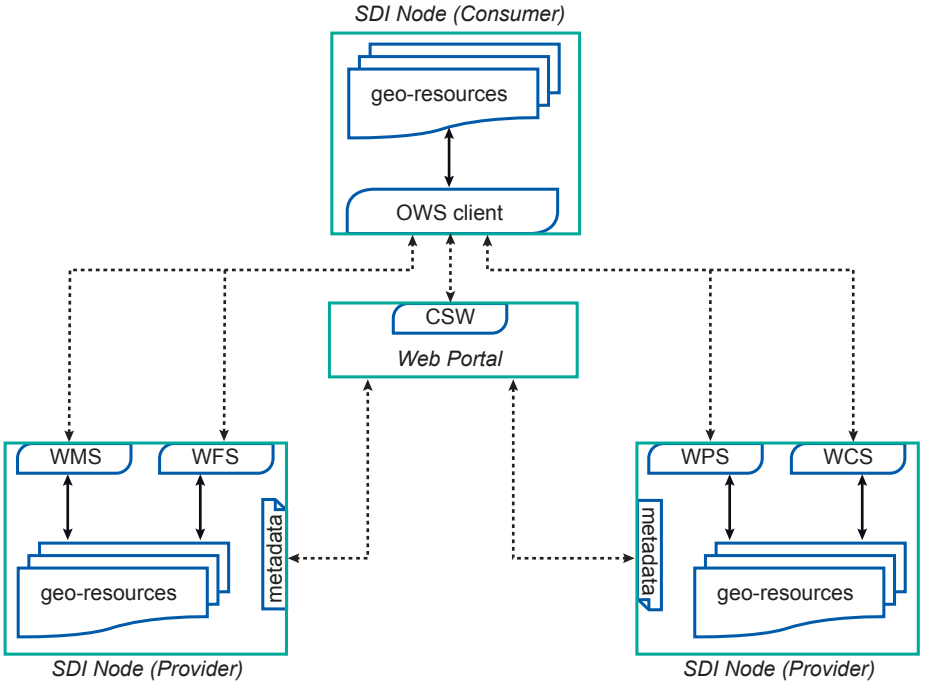


Figure 8.48
Webservices-based SDI architecture.

SDI nodes work based on the well-known client–server architecture, where a partition of responsibilities is enforced. The clients make requests to the server; the server processes the requests and returns the results to the client. In the context of SDI, these roles are interchangeable between SDI nodes. This means an SDI node can play both roles, i.e. that of client or that of server, depending on the circumstances.

8.5.4 Geo-webservices

Modern SDIs are based on webservices, which are defined as mechanisms designed to support interoperable machine-to-machine interaction over the Web. Webservices operate on the basis of standardized technologies and formats/protocols as defined by the W3C (see [126]). Webservices are self-contained, modular applications that can be described, published, located, and invoked over the Web. The working framework on which webservices operate is known as the publish-find-bind paradigm, whereby service providers make services available to service users, who consume resources by locating and binding to services. Interactions between service users and service providers are realized by exchanging messages. These messages are encoded using the eXtensible Markup Language (XML). XML is a general-purpose specification for encoding documents electronically.

Taking advantage of the webservice framework, OGC has developed a set of technical specifications known as OGC Web Services (OWS). OWS specifications are defined using open, non-proprietary Web standards. OWS specifications are platform-neutral

specifications for the implementation of interfaces. Some of the OGC Web Service standards that are relevant for an SDI include (see Figure 8.48):

- The Catalogue Service for the Web (CSW) defines common interfaces to discover, browse, and query meta-data about data, services and other geo-resources. Catalogue services consume and deliver meta-data according to the ISO standards for meta-data. This includes the ISO 19115:2003 *Geographic information–Meta-data*, which defines the schema for the identification, extent, quality, spatial and temporal schema, spatial reference, and distribution of spatial data; and also ISO 19119:2005 *Geographic information–Services*, which defines service meta-data in terms of the architecture patterns for service interfaces used and defines the standard relationship to the Open Systems Environment model. The ISO 19119 standard also presents a taxonomy for services and prescribes how to create a platform-neutral service specification.
- The Web Map Service (WMS) Implementation supports the creation and display of registered and superimposed map-like views of spatial data that come from one source or simultaneously from multiple remote and heterogeneous sources.
- The Web Feature Service (WFS) Implementation Specification allows a client to retrieve and update spatial data encoded in Geography Markup Language (GML) from one or multiple sources. The specification defines interfaces for spatial data access and manipulation operations. Through these interfaces, a Web user or service can combine, use and manage spatial data from different sources. In addition, the transactional version of the WFS specification includes the option to insert, update, or delete features from a vector data source.
- The Web Coverage Service (WCS) Implementation Specification allows clients to access parts of a grid coverage offered by a server. The data served by a WCS are grid data that are usually encoded in a binary image format. The output includes coverage meta-data.
- The Web Processing Service (WPS) Implementation Specification defines an interface that facilitates the publishing of processing functionality and also the discovery of and binding to those processes by clients. A WPS may offer calculations as simple as a buffer, or as complicated as a global climate change model. The data required by a WPS can be delivered across a network using OGC Web Services.
- The Sensor Web Enablement (SWE) set of specifications enable all types of Web and/or Internet-accessible sensors, instruments, and imaging devices to be accessible and, where applicable, controllable via the Web.

Besides the above mentioned interface specifications, an important OGC standard to achieve interoperability is the Geography Markup Language (GML). Its encoding standard is an XML grammar for expressing spatial features. GML serves as a modelling language for geographic systems, as well as an open interchange format for transactions on spatial data over the Web. In keeping with most XML-based grammars, there are two parts to a GML document: the schema that describes the structure of the data contained in the document; and the instance document that contains the actual data.

In addition to the open standards defined by OGC, there are other types of geo-webservices that can be exploited in an SDI environment. Companies like Google and Microsoft have created their own set of services to access geo-resources. These

resources include satellite data, routing operations, and so on. These services are commonly accessed via a web browser. However to properly embed those services within other applications, their developers provide what is known as an application programming interface, or API for short (see Figure 8.49). An API is a set of routines, data structures, object classes and/or protocols that can be used to build applications based on the associated services. An API for the Web is typically defined as a set of request messages along with a definition of the structure of response messages, usually expressed in XML. Most of these commercially-based geo-webservices used Web 2.0 implementation tools (see Subsection 8.5.6).

8.5.5 Meta-data

From a technical point of view, an SDI is a facility that liaises between producers and users of geographic data sets and services. For this arrangement to work, data sets and services have to be made discoverable, analysable and accessible. This is achieved by the creation of descriptions known as meta-data. Meta-data is the mechanism that producers have that enables potential users to find, analyse and evaluate their resources (data sets and services) and determine their fitness for use.

The term meta-data and the meta-data itself have become widely used over the last few years by the geo-community as if it was something new. In reality, however, its underlying concepts have been in use for generations. A map legend, for example, is one embodiment of meta-data, containing details about the publisher of the map, its publication date, the type of map, the spatial reference and the map's scale and accuracy, etc. This connection has become somewhat lost in the transition from analogue to digital data production processes.

Some authors define three categories of meta-data, based on how it is actually used: i.e. discovery, exploration and exploitation meta-data. Discovery meta-data simply enables users to find existing data and services. Discovery meta-data helps answering the question "who has what data/service and from where?", the where being an area of interest defined by means of coordinates, geographical names or administrative areas. Exploration meta-data enable users to determine whether some existing data/service is useful for their application. Exploration meta-data answer questions like "why, when and how was certain data collected". Exploitation meta-data enable users to access, transfer, load, interpret and use data/services in their applications. In addition to access, this type of meta-data also includes details about the price of the data/service and licensing and copyrights.

Meta-data is defined as a formalized and agreed upon set of properties that describe in a significant amount of detail the characteristics of a data set and/or service. ISO has therefore specified in its 19100 suite of standards the set of properties that properly describe a data set and a service.

The ISO 19115 standard defines the meta-data for vector data sets. It is applicable to a whole data set, aggregations of data sets, individual features, and the various classes of objects that compose a feature. This standard defines a large set of meta-data properties (400+), some of which are considered mandatory and some optional. To adhere to the standard, one should implement descriptions that incorporate the mandatory properties and a selection of the optional properties. The result is known as an ISO profile, a subset of the original standard.

As mentioned earlier, meta-data is mainly disseminated via meta-data catalogues that meet the OGC-defined Catalogue Service Web (CSW)-implementation specification. This specification defines the interfaces and binding mechanisms required to publish and access digital catalogues of meta-data for data, services, and related resources. Implementations of the CSW specification are known as Catalogue Services. A repository

of CSW services is known as an OGC Catalogue.

8.5.6 Web portals

Once services have been implemented for different SDI nodes, a common practice is to facilitate their access by building portals. Spatial web portals can be thought of as a gateway that provides access to geo-resources via geo-webservices on the Web. A web portal is simply a website that gives visitors organized access in a unified way, typically through catalog services, to geo-resources on the Web, and preferably also to the people and organizations offering those geo-resources (see Figure 8.49). A portal potentially offers access to many other sites. Consequently, a web portal can also be used to aggregate content.

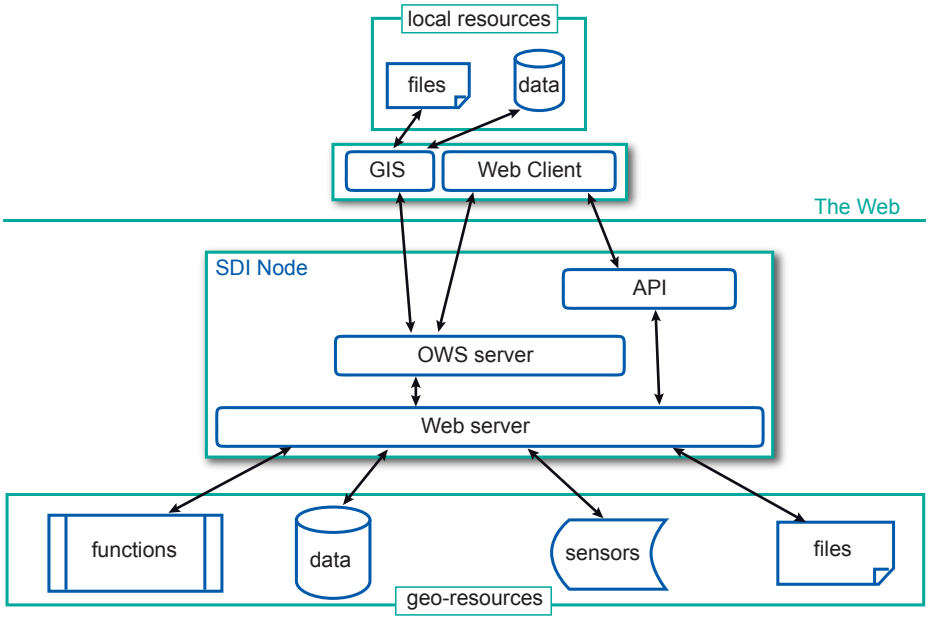


Figure 8.49
SDI node architecture and communication patterns.

8.5.7 Web 2.0

One of the latest developments for the Web, which is also available to SDI developers, is the Web 2.0 concept. Web 2.0 refers to what is perceived as a second generation of Web application development and Web design. Web 2.0 does not refer to any specific change in the technology of the Web, but rather the way in which developers implement websites and thus to the way in which people perceive and use the Web. Web 2.0 applications are characterized by their interactivity and user-centred design. Web 2.0 websites behave similarly to desktop applications that are familiar to computer users. These websites do, therefore, more than just retrieve and display information. Users can exercise control over the activity by using the interactive functions provided by the site. One principle behind Web 2.0 sites is asynchronous communication between the client application and the server. As a consequence, instead of having to reload a webpage whenever there is input from the user, the Web application makes background requests and, based on the response, dynamically updates the sections of the webpage that are affected.

This new way of working on the Web has been adopted by the SDI community and nowadays Web 2.0 tools are available that allow interactive use of spatial data over

the Web. Incidentally most of these tools have come from the open source community. Using these tools, which are available in the form of APIs, SDI developers can build highly-interactive web applications that enable users to perform all sorts of data manipulations over the Web. All the common functions for editing, analysing and processing spatial data that are conventionally only available in desktop GIS packages are now available to users in an SDI environment.

8.6 Data quality

With the advent of satellite remote sensing, GPS and GIS technology, and the increasing availability of digital spatial data, resource managers and others who formerly relied on the surveying and mapping profession to supply high quality map products are now in a position to produce maps themselves. At the same time, GISs are being increasingly used for *decision-support* applications, with increasing reliance on secondary data sourced through data providers or via the internet, from geo-webservices. The consequences of using low-quality data when making important decisions are potentially grave. There is also a danger that uninformed GIS users will introduce errors by incorrectly applying geometric and other transformations to the spatial data held in their database.

application requirements

Below we look at the main issues related to the data quality of spatial data. As outlined in Section 8.1, we will discuss positional, temporal and attribute accuracy, lineage, completeness, and logical consistency. We will begin with a brief discussion of the terms accuracy and precision, as these are often taken to mean the same thing. For a more detailed discussion and advanced topics relating to data quality, the reader is referred to [28].

8.6.1 Accuracy and precision

So far we have used the terms error, accuracy and precision without appropriately defining them. Accuracy should not be confused with *precision*, which is a statement of the smallest unit of measurement to which data can be recorded. In conventional surveying and mapping practice, accuracy and precision are closely related. Instruments with an appropriate precision are employed, and surveying methods chosen, to meet specified tolerances in accuracy. In GISs, however, the numerical precision of computer processing and storage usually exceeds the accuracy of the data. This can give rise to what is known as *spurious accuracy*, for example calculating area sizes to the nearest m^2 from coordinates obtained by digitizing a $1:50,000$ map.

accuracy tolerances

The relationship between accuracy and precision can be clarified using graphs that display the probability distribution (see below) of a measurement against the true value T. In Figure 8.50, we depict the cases of good/bad accuracy against good/bad precision.[1] An *accurate* measurement has a mean close to the true value; a *precise* measurement has a sufficiently small variance.

8.6.2 Positional accuracy

The surveying and mapping profession has a long tradition of determining and minimizing errors. This applies particularly to land surveying and photogrammetry, both of which tend to regard positional and height errors as undesirable. Cartographers also strive to reduce geometric and attribute errors in their products, and, in addition, define quality in specifically cartographic terms, for example quality of line work, layout, and clarity of text.

It must be stressed that all measurements made with surveying and photogrammetric instruments are subject to error. These include:

1. Human errors in measurement (e.g. reading errors) generally referred to as gross errors or *blunders*. These are usually large errors resulting from carelessness, which could have been avoided through careful observation, although it is never absolutely certain that all blunders could have been avoided or eliminated.

[1] Here we use the terms "good" and "bad" to illustrate the extremes of both accuracy and precision. In real world terms, we refer to whether data are "fit for use" for a given application.

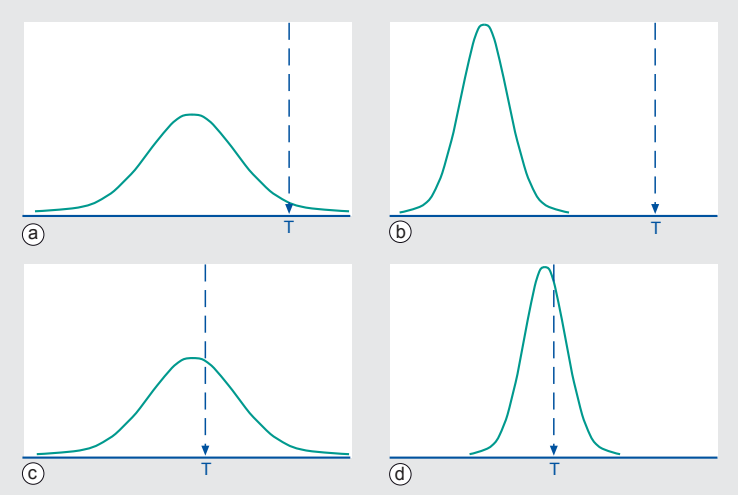


Figure 8.50
A measurement probability function and the underlying true value T: (a) bad accuracy and precision, (b) bad accuracy/good precision, (c) good accuracy/bad precision, and (d) good accuracy and precision.

error sources

2. Instrumental or *systematic* errors (e.g. due to maladjustment of instruments). This leads to errors that vary systematically in sign and/or magnitude, but can go undetected by repeating the measurement with the same instrument. Systematic errors are particularly dangerous because they tend to accumulate.

3. So–called *random* errors caused by natural variations in the quantity being measured. These are effectively the errors that remain after blunders and systematic errors have been removed. They are usually small, and dealt with in least–squares adjustment.

Section 3.2 discussed the errors inherent in various methods of spatial positioning. Below we will at more general ways of quantifying positional accuracy using *root mean square error* (*RMSE*).

Measurement errors are generally described in terms of *accuracy*. In the case of spatial data, accuracy may relate not only to the determination of coordinates (positional error) but also to the measurement of quantitative attribute data. The accuracy of a single measurement can be defined as:

> "*the closeness of observations, computations or estimates to the true values or the values perceived to be true*" [79].

In the case of surveying and mapping, the "truth" is usually taken to be a value obtained from a survey of higher accuracy, for example by comparing photogrammetric measurements with the coordinates and heights of a number of independent check points determined by field survey. Although it is useful for assessing the quality of definite objects, such as cadastral boundaries, this definition clearly has practical difficulties in the case of natural resource mapping where the "truth" itself is uncertain, or boundaries of phenomena become fuzzy. This type of uncertainty in natural resource data is elaborated upon on page 301.

relative and absolute accuracy

Prior to the availability of GPS, resource surveyors working in remote areas sometimes had to be content with ensuring an acceptable degree of *relative accuracy* among the measured positions of points within the surveyed area. If location and elevation are fixed with reference to a network of control points that are assumed to be free of error, then the *absolute accuracy* of the survey can be determined.

Root mean square error

Location accuracy is normally measured as a *root mean square error* (*RMSE*). The RMSE is similar to, but not to be confused with, the standard deviation of a statistical sample. The value of the RMSE is normally calculated from a set of check measurements (coordinate values from an independent source of higher accuracy for identical points). The differences at each point can be plotted as error vectors, as is done in Figure 8.51 for a single measurement. The error vector can be seen as having constituents in the x- and y-directions, which can be recombined by vector addition to give the error vector representing the locational error.

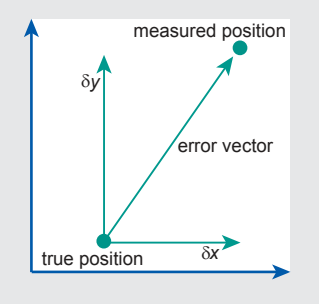


Figure 8.51
The positional error of a measurement can be expressed as a vector, which in turn can be viewed as the vector addition of its constituents in the x- and y-directions, respectively δx and δy.

For each checkpoint, the error vector has components δx and δy. The observed errors should be checked for a *systematic* error component, which may indicate a (possibly repairable) lapse in the measurement method. Systematic error has occurred when $\sum \delta x \neq 0$ or $\sum \delta y \neq 0$.

The systematic error $\delta\overline{x}$ in x is then defined as the average deviation from the true value:

$$\delta\overline{x} = \frac{1}{n}\sum_{i=1}^{n} \delta x_i.$$

Analogously to the calculation of the variance and standard deviation of a statistical sample, the root mean square errors m_x and m_y of a series of coordinate measurements are calculated as the square root of the average squared deviations:

$$m_x = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \delta x_i^2} \quad \text{and} \quad m_y = \sqrt{\frac{1}{n}\sum_{i=1}^{n} \delta y_i^2},$$

where δx^2 stands for $\delta x \cdot \delta x$. The total RMSE is obtained with the formula

$$m_{\text{total}} = \sqrt{m_x^2 + m_y^2},$$

which, by the Pythagorean rule, is the length of the average (root squared) vector.

Accuracy tolerances

Many kinds of measurement can be naturally represented by a bell-shaped probability density function p, as depicted in Figure 8.52(a). This function is known as the *normal (or Gaussian) distribution* of a continuous, random variable, in the figure indicated as Y. It shape is determined by two parameters: μ, which is the mean expected value for Y, and σ, which is the standard deviation of Y. A small σ leads to a more attenuated bell-shaped function.

distribution of errors

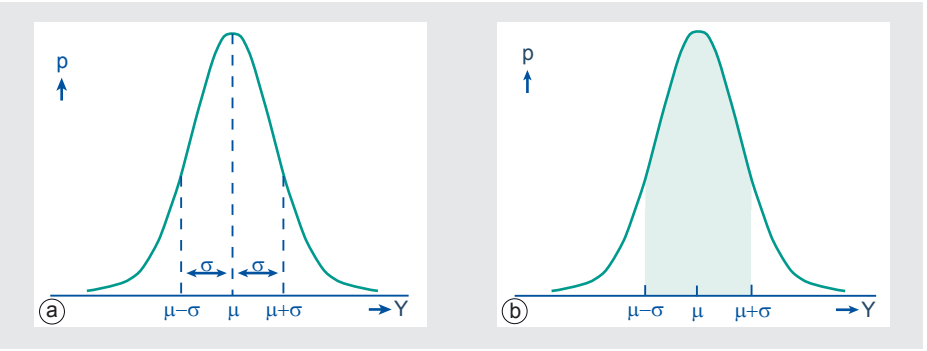


Figure 8.52
(a) Probability density function p of a variable Y, with its mean μ and standard deviation σ. (b) The probability that Y is in the range $[\mu - \sigma, \mu + \sigma]$.

Any probability density function p has the characteristic that the area between its curve and the horizontal axis is equal to 1. Probabilities P can be inferred from p as the area under p's curve. Figure 8.52(b), for instance, depicts $P(\mu - \sigma \leq Y \leq \mu - \sigma)$, i.e. the probability that the value for Y is within distance σ from μ. In a normal distribution this specific probability for Y is always 0.6826.

The *RMSE* can be used to assess the probability that a particular set of measurements does not deviate too much from, i.e. is within a certain range of, the "true" value. In the case of coordinates, the probability density function is often considered to be that of a two-dimensional normally distributed variable (see Figure 8.53). The three standard probability values associated with this distribution are:

- 0.50 for a circle with a radius of 1.1774 m_x around the mean (known as the *circular error probable*, CEP);
- 0.6321 for a circle with a radius of 1.412 m_x around the mean (known as the *root mean square error*, RMSE);
- 0.90 for a circle with a radius of 2.146 m_x around the mean (known as the *circular map accuracy standard*, CMAS).

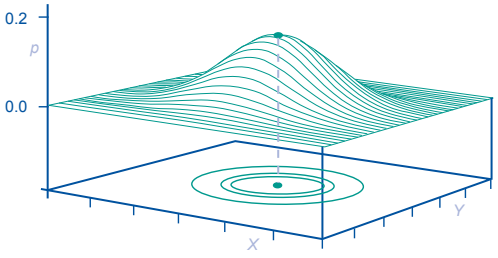


Figure 8.53
Probability density p of a normally distributed, two-dimensional variable (X, Y) (also known as a normal, bivariate distribution). In the ground plane, starting from the inside out, are the circles associated with CEP, RMSE and CMAS.

The RMSE provides an estimate of the spread of a series of measurements around their (assumed) "true" values. It is therefore commonly used to assess the quality of transformations such as the absolute orientation of photogrammetric models or the spatial referencing of satellite imagery. The RMSE also forms the basis of various statements for reporting and verifying compliance with defined map accuracy *tolerances*. An example is the American National Map Accuracy Standard, which states that:

> *"No more than 10% of well-defined points on maps of 1:20,000 scale or greater may be in error by more than 1/30 inch."*

Normally, compliance to this tolerance is based on at least 20 well-defined checkpoints.

The epsilon band

As a line is composed of an infinite number of points, confidence limits can be described by what is known as an epsilon (ε) or Perkal band at a fixed distance on either side of the line (Figure 8.54). The width of the band is based on an estimate of the probable location error of the line, for example to reflect the accuracy of manual digitizing. The epsilon band may be used as a simple means for assessing the likelihood that a point receives the correct attribute value (Figure 8.55).

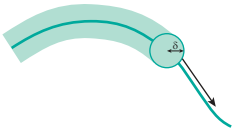


Figure 8.54
The ε or Perkal band is formed by rolling an imaginary circle of a given radius along a line.

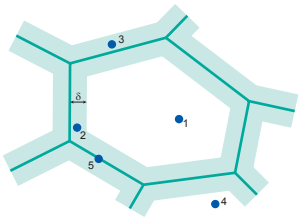


Figure 8.55
The ε band may be used to assess the likelihood that a point falls within a particular polygon (source: [86]). Point 3 is less likely part of the middle polygon than point 2.

Describing natural uncertainty in spatial data

There are many situations, particularly in surveys of natural resources, where, according to Burrough, "practical scientists, faced with the problem of dividing up undividable complex continua have often imposed their own crisp structures on the raw data" [13, p. 16]. In practice, the results of classification are normally combined with other categorical layers and continuous field data to identify, for example, areas suitable for a particular land use. In a GIS, this is normally achieved by overlaying the appropriate layers using logical operators.

classification

Particularly in the case of natural resource maps, the boundaries between units may not actually exist as lines but only as transition zones, across which one area continuously merges into another. In these circumstances, rigid measures of positional accuracy, such as RMSE (Figure 8.51), may be virtually insignificant in comparison to the uncertainty inherent in vegetation and soil boundaries, for example.

boundaries

In conventional applications of the error matrix to assess the quality of nominal (categorical) data such as land use, individual samples can be considered in terms of Boolean set theory. The Boolean *membership function* is binary, i.e. an element is either a member of the set (membership is `true`) or it is not a member of the set (membership is `false`). Such a membership notion is well-suited to the description of spatial features such as land parcels for which no ambiguity is involved and an individual ground truth sample can be judged to be either correct or incorrect. As Burrough notes, "increasingly, people are beginning to realize that the fundamental axioms of simple binary logic present limits to the way we think about the world. Not only in everyday situations, but also in formalized thought, it is necessary to be able to deal with concepts that are not necessarily `true` or `false`, but that operate somewhere in between."

membership functions

Since its original development by Zadeh [130], there has been considerable discussion of fuzzy, or continuous, set theory as an approach for handling imprecise spatial data. In GIS, fuzzy set theory appears to have two particular benefits:

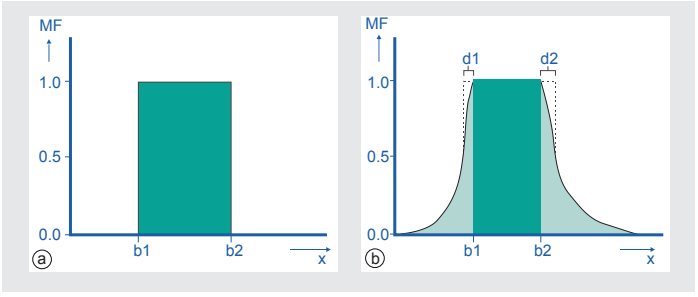
fuzzy set theory

1. the ability to handle logical modelling (map overlay) operations on inexact data; and
2. the possibility of using a variety of natural language expressions to qualify uncertainty.

Unlike Boolean sets, fuzzy or continuous sets have a membership function, which can assign to a member any value between 0 and 1 (see Figure 8.56). The membership function of the Boolean set of Figure 8.56a can be defined as MF^B, where:

$$\mathrm{MF}^B(x) = \begin{cases} 1 & \text{if } b_1 \leq x \leq b_2 \\ 0 & \text{otherwise} \end{cases}$$

The crisp and uncertain set membership functions of Figure 8.56 are illustrated for the one-dimensional case. Obviously, in spatial applications of fuzzy set techniques we typically would use two-dimensional sets (and membership functions).



(a) .

Figure 8.56
(a) Crisp (Boolean) and (b) uncertain (fuzzy) membership functions MF. After Heuvelink [44]

The continuous membership function of Figure 8.56b, in contrast to function MF^B above, can be defined according to Heuvelink [44] as a function MF^C :

$$\mathrm{MF}^C(x) = \begin{cases} \frac{1}{1+\left(\frac{x-b_1}{d_1}\right)^2} & \text{if } x < b_1 \\ 1 & \text{if } b_1 \leq x \leq b_2 \\ \frac{1}{1+\left(\frac{x-b_2}{d_2}\right)^2} & \text{if } x > b_2 \end{cases}$$

The parameters d_1 and d_2 denote the width of the transition zone around the kernel of the class such that $\mathrm{MF}^C(x) = 0.5$ at the thresholds $b_1 - d_1$ and $b_2 + d_2$, respectively. If d_1 and d_2 are both zero, the function MF^C reduces to MF^B.

An advantage of fuzzy set theory is that it permits the use of natural language to describe uncertainty, for example, "near," "east of" and "about 23 km from". Such natural language expressions can be more faithfully represented by appropriately chosen membership functions.

8.6.3 Attribute accuracy

We can identify two types of attribute accuracies. These relate to the type of data we are dealing with:

- For *nominal or categorical* data, the accuracy of labelling (for example the type of land cover, road surface, etc).
- For *numerical* data, numerical accuracy (such as the concentration of pollutants in a soil, height of trees in forests, etc).

It follows that depending on the data type, assessment of attribute accuracy may range from a simple check on the labelling of features—for example, is a road classified as a metalled road actually surfaced or not?—to complex statistical procedures for assessing the accuracy of numerical data, such as the percentage of pollutants present in a soil.

When spatial data are collected in the field, it is relatively easy to check on the appropriate feature labels. In the case of remotely sensed data, however, considerable effort may be required to assess the accuracy of the classification procedures. This is usually done by means of checks at a number of sample points. The field data are then used to construct an error matrix (also known as a confusion or misclassification matrix) that can be used to evaluate the accuracy of the classification. An example is provided in Table 8.4, where three land use types are identified. For 62 check points that are forest, the classified image identifies them as forest. However, two forest check points are classified in the image as agriculture. *Vice versa*, five agriculture points are classified as forest. Observe that correct classifications are found on the main diagonal of the matrix, which sums up to 92 correctly classified points out of 100 in total.

error matrix

Table 8.4
Example of a simple error matrix for assessing map attribute accuracy. The overall accuracy is $(62 + 18 + 12)/100 = 92\%$.

Classified image	**Reference data**			
	Forest	*Agriculture*	*Urban*	Total
Forest	62	5	0	67
Agriculture	2	18	0	20
Urban	0	1	12	13
Total	64	24	12	100

8.6.4 Temporal accuracy

As noted, the amount of spatial data sets and archived remotely-sensed data has increased enormously over the last decade. These data can provide useful temporal information, such as changes in land ownership and the monitoring of environmental processes such as deforestation. Analogous to its positional and attribute components, the quality of spatial data may also be assessed in terms of its *temporal accuracy*. For a static feature this refers to the difference in the values of its coordinates at two different times.

Temporal accuracy includes not only the accuracy and precision of time measurements (for example, the date of a survey) but also the temporal consistency of different data sets. Because the positional and attribute components of spatial data may change together or independently, it is also necessary to consider their temporal validity. For example, the boundaries of a land parcel may remain fixed over a period of many years whereas the ownership attribute may change more frequently.

consistency and validity

8.6.5 Lineage

Lineage describes the history of a data set. In the case of published maps, some lineage information may be provided as part of its meta-data, in the form of a note on the data sources and procedures used in the compilation of the data. Examples include the date and scale of aerial photography, and the date of field verification. Especially for digital data sets, however, lineage may be defined more formally as:

> "*that part of the data quality statement that contains information that describes the source of observations or materials, data acquisition and compilation methods, conversions, transformations, analyses and derivations that the data has been*

> *subjected to, and the assumptions and criteria applied at any stage of its life."*
> [22]

All of these aspects affect other aspects of quality, for example positional accuracy. Clearly, if no lineage information is available, it is not possible to adequately evaluate the quality of a data set in terms of "fitness for use".

8.6.6 Completeness

incomplete and overcomplete

Completeness refers to whether there are data lacking in the database compared to what exists in the real world. Essentially, it is important to be able to assess what does and what does not belong to a *complete* data set as intended by its producer. It might be incomplete (i.e. it is "missing" features which exist in the real world), or overcomplete (i.e. it contains "extra" features which do not belong within the scope of the data set as it is defined).

Completeness can relate to either spatial, temporal, or thematic aspects of a data set. For example, a data set of property boundaries might be spatially incomplete because it contains only 10 out of 12 suburbs; it might be temporally incomplete because it does not include recently subdivided properties; and it might be thematically overcomplete because it also includes building footprints.

8.6.7 Logical consistency

For any particular application, (predefined) logical rules concern:

- the *compatibility* of data with other data in a data set (e.g. in terms of data format);
- the absence of any *contradictions* within a data set;
- the *topological consistency* of the data set; and
- the allowed attribute *value ranges*, as well as combinations of attributes. For example, attribute values for population, area and population density must agree for all entities in the database.

The absence of any inconsistencies does not necessarily imply that the data are accurate.

8.7 Spatial variation and interpolation

A central activity in studies involving spatial variability is to get from point observations towards area-covering statements. In a GIS one may need a map of a spatial property. Variability (Oxford Dictionary: the quality of being variable in some respect) in space is defined as the phenomenon that a variable changes in space. Earlier, on page 238, we noted that at a certain location a variable may be observed. At a very small distance from this location, the variable may be observed again, and it is likely that it will deviate from the previous observation. The deviations may increase as the distances increase. Description of variability (how large are the deviations if the distance increases) is important for process-based interpretation. An example that will be analysed below concerns the issue of global change, for which weather data is crucial. At a single moment in time, temperature, rainfall and other meteorological data vary in space. Measurements can only be collected reliably at a limited number of locations, making the creation of a map a major undertaking. In another example, to better understand the spatial distribution of wealth and income at a city level, the prices of houses that are for sale can provide important information. Rarely, if ever, will all houses in a city be for sale at a single moment, nevertheless a map of house prices can and should be created. A quantitative approach for making such maps from a collection of point observations within a GIS is called geostatistics. Earth sciences was one of the first disciplines to develop this approach, initially in mining and geology, and later in the agricultural and environmental sciences.

In order to describe spatial variation and interpolation we first single out continuous data. Continuous data represent a continuous phenomenon. They can in principle be collected at any location. Such data are to be distinguished from vector data (roads, buildings, etc.) that one usually considers to be fixed. Notice that house prices are in fact related to a polygon (namely the parcel), but at the city level we can consider them as points. Continuous data are likely to vary throughout a region. Even when all data have been measured in precisely the same manner, i.e. without error and by the same surveyor, variation will still occur. House prices will vary in space, and also the percentage of clay will vary from place to place. Because such variation takes place in space, we speak of spatial variation. Naturally, these variables are allocated to its place in space.

A crucial step in the process is that of obtaining a value at a point where no measurement has been taken. There are several methods for achieving this:

Inverse distance For inverse distance interpolation, weights are assigned to observations. These weights are proportional to the inverse of the distance between a prediction point and an observation point. Distances can be squared although any other power of the distance may also be taken. Inverse distance routines result in a map showing islands, i.e. anomalies in the form of dark and light circles reflecting the values in the observation points.

TIN TIN procedures (discussed on page 246) combine observations by lines. For some data, such as elevation data, TIN procedures are applied successfully. However, TIN procedures fall short if the number of observations is relatively small and if the data (and their locations) are inaccurate.

Trend surfaces Trend surfaces give a global pattern of spatial interpolation. They might give a general picture of the variable in the region.

Geostatistics In essence, in geostatistics the spatial variation of a variable is modelled and then subjected to optimal interpolation. Our focus here is on geostatistics.

For several decades, the field of geostatistics has been exploring approaches for dealing quantitatively with spatial variation in data. Usually, two stages are distinguished. The first stage is an analysis of the (spatial) dependence, i.e. how large is the variation as a function of the distance between observation. The second stage is the joint production of a map of the variable and a map of its precision.

Statistical and geostatistical procedures may be helpful in a number of stages of the interpretation and evaluation of data with a spatial distribution. In many studies they have proven to be indispensable, especially if the amount of available data is great. Some aspects to keep in mind are:

- How can one quantify the *type* and the *amount* of the variation. Some examples: Which properties vary within a region? Does every property varying at the same scale? What is the relation between the spatial variation of a property and aspects of soils, such as sedimentation, and the effect of human activities in the past?
- To *predict* the value of a variable at an unvisited location. Some examples: What is the mean value of nitrate leaching in a parcel? What is the total amount of polluted soil? What is the uncertainty associated with the prediction?

A quantitative approach to spatial variability is of a crucial importance in many spatial studies.

Consider an area which is homogeneous (stationary) with regard to a particular variable being studied. The variogram $\gamma(h)$ is a function of the distance h between locations in the area. The variogram for distance h equals half the expectation of squared differences of variables located at this distance from each other.

8.7.1 The empirical variogram

Observations in space are linked to their coordinates and each observation has its own specific location in space. The value of the coordinate x is essentially linked with the variable Y. Such spatial variables are therefore expressed as $Y(x)$: the place dependence of the variable Y on the location x is given explicitly. They are termed *regionalized variables*. For $Y(x)$ one may read any spatially varying property. The variable $Y(x)$ is put in capitals to indicate that it is a stochastic variable: i.e. a variable that is influenced by unknown and sometimes unmeasurable factors outside our control; it is subject to random influences.

regionalized variables

As before, an important characteristic to be dealt with is that the data close to each other are more likely to be similar than data collected at larger distances from each other. This implies that the variables $Y(x_1)$ and $Y(x_2)$ in two locations x_1 and x_2 are probably more alike if the distance between the locations is small, than if the distance is large. The dependence between regionalized variables at different locations is the main, characteristic difference with traditional stochastic variables. The size and the functional form of the differences as a function of the distance will be studied.

An important aspect of regionalized variables is their *expectation* $E[Y(x)] = \mu$ and their *variance* $Var[Y(x)] = \sigma^2$. As the word says: the expectation is the value that would be expected if random influences were absent. For a range of observations, the expectation is estimated by the mean, or by the median (the 50th percentile). The variance is a measure for the noise around the mean: a noisy observation will have a large variance as compared to the expectation, whereas a precise observation would have a low variance. Situations exist, however, where μ does not exist; or σ^2 is not finite. Our examination of geostatistics will focus on the less restrictive requirement,

summarized in what is known as the intrinsic hypothesis [51]. Consider two points along the transect x and $x + h$, the latter point being located at a distance h from the first point x. The intrinsic hypothesis is:

1. $E[Y(x) - Y(x + h)] = 0$
2. $Var[Y(x) - Y(x + h)] < \infty$ and is independent of x.

The first part can be interpreted as follows: the expectation of the difference of a regionalized variable at location x and at a distance h from x equals zero. The second part of the hypothesis requires that the variance of the difference of a regionalized variable measured at location x and at a distance h from x exists, and is *independent of* x. The difference between the two variables $Y(x)$ and $Y(x + h)$ is called a pair difference. The precise form of the dependence of the variance of pair differences on h is often interesting for interpretive purposes, as we will see below.

The spatial dependence function of observations is defined by the second part of the intrinsic hypothesis. It is termed the *variogram* $\gamma(h)$. The variogram is defined as a function of the distance h between locations in the observation space:

variogram

$$\gamma(h) = \frac{1}{2} E\left[Y(x) - Y(x + h)\right]^2 \tag{8.1}$$

Because the expectation of $Y(x) - Y(x + h)$ is equal to zero (intrinsic hypothesis!), the variogram equals half the variance of pair differences at a distance h. Due to the assumption summarized in the intrinsic hypothesis, this variance of pair differences exists and is properly defined. Note that the inclusion of the factor $\frac{1}{2}$ in the expression allows a straightforward comparison with the covariance function: $\gamma(h) = C(0) - C(h)$. The variogram is *independent* of the place where the regionalized variables are located. The squared pair differences have the same expectation, regardless of whether they are measured at one part of the transect or another. Loosely speaking, we expect similar differences between observations independent of the part in the area. In many practical studies one observes an increase of the variogram with increasing distance between the observation locations. This implies that the dependence *decreases* with *increasing* distance h between locations. Loosely speaking: observations close to each other are more likely to be similar than observations at a larger distance from each other.

In order to determine the variogram for data collected in two dimensions, an approach similar to that defined above for the transect data can be applied. The only complication is that for each observation two coordinates are associated, instead of just one coordinate. An estimate, the empirical variogram, $\hat{\gamma}(h)$, may be obtained by taking for a fixed value of h all pairs of points with a separation distance approximately equal to h, squaring the differences between the measurements constituting such a pair, summing these squared differences and dividing the sum by 2 and by the total number of pairs $N(h)$ obtained for this distance. Let the ith pair consist of the two points $y(x_i)$ and $y(x_i + h)$, where we have now used a lower case symbol to distinguish it from the variable $Y(x)$ used earlier. We thus obtain the following equation:

$$\hat{\gamma}(h) = \frac{1}{2N(h)} \sum_{i=1}^{N(h)} \left(y(x_i) - y(x_i + h)\right)^2 \tag{8.2}$$

This is repeated for all values of h. A graph may display $\hat{\gamma}(h)$ as a function of h. As the distances between the observation locations are not equal to each other, distance

classes are created: all *pairs* of points of approximately the same sampling distance are grouped into one distance class. This distance defines the class to which the pair belongs.

The user usually has to decide upon a lag length. Sometimes this choice is rather obvious, as in the example of the equidistant observations. A choice for a lag length typically influences the number of distance classes. If the lag length is chosen to be large (larger than the largest occurring distance), only one distance class remains that contains all pair of observations. The estimated variogram value for this distance class equals the estimated variance of the variable: the spatial dependence between the observations is then neglected. At the other extreme, we may choose a very small lag length, resulting in a large number of distance classes, each containing only a few pairs of observations. Although this may be illustrative for some purposes, it is not usually very informative.

For practical purposes there are some general rules that have to be obeyed in order to obtain reliable variogram estimates:

1. The number of pairs of observation points in each class must exceed 30.
2. The maximum distance h between observation points for which the variogram may be determined should not exceed half the length of the area.

In some programs a lag tolerance also has to be specified: if the lag tolerance is less than half the lag length, pairs of observations may be excluded from the analysis; whereas if it exceeds half the lag length, pairs of observations may be allocated to different distance classes.

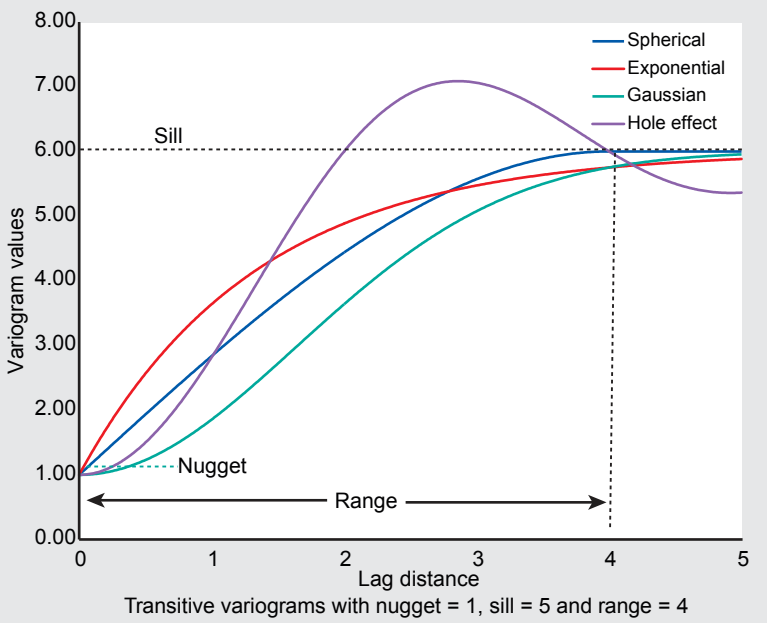


Figure 8.57
Four common variogram models, each with a nugget equal to 1, a sill equal to 6, and a range equal to 4. The sill is only reached by the spherical model, the other models are within 95% of the sill when the distance is equal to the range.

It is often necessary to fit a specific function through the variogram estimates. A practical way to do this is to estimate the parameters of such a function by a non-linear regression procedure. A distinction can be made between transitive variograms (which apply to fields that have a finite variance) and infinite variograms. Commonly

used transitive variograms are the spherical variogram, the exponential variogram, the Gaussian variogram and the hole effect (or wave) variogram, see Figure 8.57. These variograms all depend upon the distance h. They are characterized by two parameters, a range of influence a and the sill variance b. The *range* a is a measure for the distance up to which the spatial dependence extends. Between locations separated by a distance exceeding a, the regionalized variables are uncorrelated; between locations separated by a distance smaller than a the regionalized variables are dependent. The *sill value* (or the variance) is the value b that the variogram reaches if h tends to infinity, i.e. if the observations are growing to be uncorrelated. A special variogram is the Nugget variogram, which takes a constant value (C_0) for all distances h. The *nugget effect*, a term borrowed from gold mining, denotes the non-spatial variability, the variability at very small distances or the operator bias. If a sill value and a range are not observed then an infinite variogram is the most appropriate. The most common infinite variogram is the power variogram that is characterized by two parameters: the power m and a multiplication parameter k.

range

sill

nugget

Any sum of variograms can be made, in particular the nugget effect is often added to other variograms, resulting in a variogram with a discontinuity at the origin. By definition, $\gamma(0) = 0$. In all the equations, C_0, A and b, or k and m, are positive parameters that are to be determined from the original data. The exponential variogram never reaches the sill value (nor does the Gaussian variogram or the Hole effect variogram). The parameter a is therefore associated with the range but is not similar to the range. We define the effective range to be equal to $3{\cdot}a$, being the distance where the exponential model reaches 95% of the sill value. Similar values apply for the other variograms. The Gaussian variogram is characterized by its horizontal behaviour at the origin. This variogram is encountered, for example, when there is uncertainty with respect to the precise location of the observations. The hole effect (wave) variogram is regularly encountered in practice. It points to periodicities of the variable caused by human influences, sedimentation processes, etc. Interpretation of such periodicities is often important.

Whenever a sill value is reached it can be interesting to study the sill/nugget ratio, which gives an indication of the part of the variability to be assigned to spatial variability and of the part to be assigned to non-spatial variability. If the ratio is close to 1, the non-spatial variability is dominant, otherwise the spatial variability is.

8.7.2 Interpolation

Interpolation is used to create a GIS layer out of point observations on a continuous variable. The reason for doing this could be manifold: for visualization purposes, for making a proper reference with other data, or for making a combination of different layers. Consider the problem of obtaining $Y(x_0)$, i.e. the value of a variable $Y(x)$ at an unvisited location x_0. A basic fact is Tobler's law (see Subsection 8.1.2). This already implies intuitively that it is highly unlikely that all the predictions are equal to the mean value: deviations from the mean are likely to occur, especially in the neighbourhood of largely deviating observations. If an observation is above the regression line, then it is highly likely that observations in its neighbourhood will be above this line as well.

We will focus attention on linear combinations of the observations that we will call a predictor for $Y(x_0)$. Hence, each observation is assigned a *weight* such that the predictor is without bias (i.e. the predictor may yield too high a value or too low a value, but on the average it is just right). Carrying out predictions is never precise, and the difference between the value of the variable had we observed it and its interpolated value is called the prediction error. We are interested in predictions that have the lowest

variance of the prediction error.

Let the optimal predictor be denoted with $\hat{Y}(x_0)$. It is linear in the observations, therefore

$$\hat{Y}(x_0) = \sum_{i=1}^{n} \lambda_i Y(x_i) \tag{8.3}$$

with as yet unknown weight λ_i. For the prediction error $e = \hat{Y}(x_0) - Y(x_0)$ it is assumed that its expectation is zero ($E[e] = 0$) and that its variance is minimal among all linear unbiased predictors: $Var[\hat{Y}(x_0) - Y(x_0)]$ is minimal. The variance of the prediction error is of importance and is to be calculated below; see equation 8.5.

Predicting requires knowledge of the variogram. Suppose, therefore, that there are n observations, that a variogram has been determined, that a model has been fitted, and that its parameters have been estimated. The variogram values can be determined for all distances between all pairs of points consisting of observation points (which are $\frac{1}{2}n(n-1)$ in number). They are contained in the, symmetric, $n \times n$ matrix G. The elements of G, g_{ij}, are then filled with the values obtained from $g_{ij} = \gamma(|x_i - x_j|)$; g_{ii} contains the variogram for the distance between x_i and x_i, a pair of points with distance equal to zero, and hence $g_{ii} = 0$; g_{ij} for $i \neq j$ contains the variogram value for the distance between x_i and x_j and is equal to g_{ij}. The variogram can also be determined for all pairs of points consisting of an observation point and the prediction location (which are n in number). These are contained in the vector g_0. The ith element of g_0 contains the variogram value for the distance between x_i and the prediction location x_0. We first estimate the spatial mean $\hat{\mu}$ by means of the generalized least squares estimator

$$\hat{\mu} = \left(1_n^T G^{-1} 1_n\right)^{-1} 1_n^T G^{-1} y \tag{8.4}$$

where y is the vector of observations and 1_n is the vector of n elements, all equal to 1, and T denotes the transpose of a vector (or matrix). Notice that it is different from the average value, as it essentially includes the spatial dependence.

Equation 8.3 can be further modified as

$$\hat{Y}(x_0) = \sum_{i=1}^{n} \lambda_i Y(x_i) = \hat{\mu} + g_0^T G^{-1} (y - \hat{\mu} \cdot 1_n) \tag{8.5}$$

The equation consists of two terms: the spatial mean $\hat{\mu}$ and the term $g_0^T G^{-1}(y - \hat{\mu} \cdot 1_n)$. This second term expresses the influence on the predictor of the residuals $(y - \hat{\mu}1_n)$ of the observations y with respect to the mean value $\hat{\mu}1_n$. The residuals are transformed with $g_0^T G^{-1}$. Such a transformation is clearly based upon the variogram. Variogram values among the observation points are in cluded in the matrix G, variogram values between the observation points and the prediction location in the vector g_0. Predicting an observation in the presence of spatially dependent observations is termed Kriging, named after the first practitioner of these procedures, the South African mining engineer Daan Krige, who did much of his early empirical work in the Witwatersrand gold mines.

Kriging

Every prediction is associated with a prediction error. The prediction error itself cannot be determined, but an equation for the variance of the prediction error is given by

prediction error variance

$$Var(Y(x_0) - \hat{Y}(x_0)) = -g_0^T G^{-1} g_0 + \frac{x_a^2}{V} \tag{8.6}$$

where x_a equals $1 - g_0^T G^{-1} 1_n$ and $V = 1_n' G^{-1} 1_n$. All matrices and vectors can be filled on the basis of the data, the observation locations and the estimated variogram. We remark that the variance of the prediction error depends only *indirectly* upon the observations: the vector y is not included in the equation. However, the variogram is estimated on the basis of the observations, and hence the observations appear in an indirect way in the equation. The *configuration* of the n observation points and the one prediction location influences the prediction error variance as well.

If a map has to be constructed of a spatial property for which the observations are collected in a 2-dimensional space the following procedure may be used:

1. Determine the empirical variogram;
2. Fit a variogram to the empirical variogram;
3. Predict values at the nodes of a fine-meshed grid;
4. Present the results in a two- or a three-dimensional perspective by linking individual predictions with line elements.

In addition to the map itself, it may be desirable (and sometimes even necessary) to display the prediction error variance (or its square root), which is obtained at the same nodes of the fine-meshed grid as the predictions themselves. This map displays the spatial uncertainty of the map.

In the manner described above, it is possible to jointly make two layers in a GIS by spatial interpolation of point observations. Such interpolation has the property that it is driven by the spatial variability of the continuous variable and is, hence, specific for each variable.

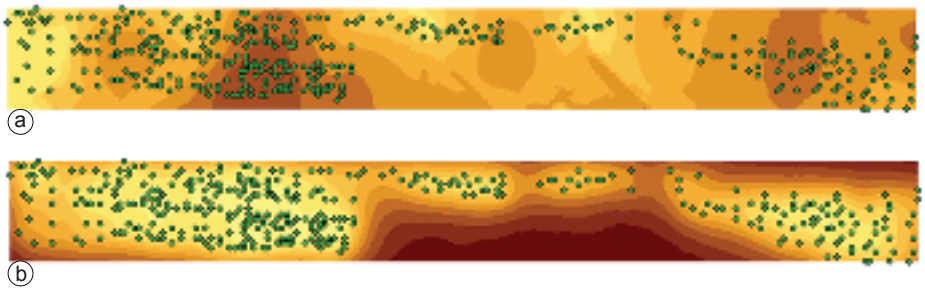


Figure 8.58
Ordinary kriging of lead (Pb) concentrations in soil within an area in The Hague. x-coordinate ranges over 2 km, y-coordinate ranges over 200 m. (a) predictions of the concentrations (values from 6 to 2900 ppm), (b) kriging standard deviations (values from 316 to 359).

To illustrate that we consider an example. In the city of The Hague a soil inventory was carried out within an area of 2 km by 200 m (Figure 8.58), bordering a railway. This inventory followed the closing of a cable factory in the area, and the decision making on the intended future use of the area. In total some 500 observations were provided on several chemical constituents of the possibly contaminated soil. One of the critical constituents is the amount of lead (Pb) that was sampled from the top 50 cm of the soil. In order to get a full overview of the spatial variation of this variable, a variogram was constructed and two maps were produced: the kriged map of the Pb contamination in the soil (Figure 8.58a), and a map of the standard deviation of the prediction error (Figure 8.58b). Maps were produced with ArcGIS. The top map showed a large spatial

variation, from peak values up to 2900 ppm (parts per million, equivalent to 2.9 g Pb per kg soil), and critical environmental thresholds were exceeded. Such heavily polluted soils have to be cleaned, in particular if the future use of the area would be residential. The kriging standard deviation showed less variation in absolute values, but it shows relatively low values close to observation locations, and increasing values at locations that are farther away from the sampling locations.