

Data Query and Buffering

Querying data

As with all database systems, one of the core parts of GIS functionality is the ability to query the data. With GIS software there are two basic forms of querying: spatial and attribute. Spatial querying asks the question ‘what is at this location’? This is usually done by simply clicking on a feature and listing its attributes. More complex spatial queries could select all the features within a box or a polygon, or ask ‘what is near to this feature’? Queries of this type often require the use of buffering or overlay techniques as described later in the chapter. Attribute querying asks the question ‘where does this occur’? If a user has a layer consisting of the locations of churches with some information about each church, an attribute query could select all the churches whose denomination is Catholic and then draw them with a certain symbol. The user could then query the database to select all Protestant churches and draw these with a different symbol to compare the patterns.

Manipulating and measuring spatial data

Most GIS software packages come with a suite of options that allow the user to manipulate spatial data. One of the most basic of these is simply to change the projection system used. This can significantly alter the appearance of maps, particularly maps of the world, and can also make it possible to integrate data from layers that use different projection systems. In Britain, this could be used to take a variety of early maps on different projections and re-project them onto the National Grid. This would allow comparisons between different early maps, as well as with modern ones. Putting adjacent map sheets onto the same projection allows their digital representations to be joined to form a single layer. Where there are distortions to the sheets the maps will have to be rubber-sheeted (or “edge-matched”) to ensure that the edges of the two sheets make a perfect join. This involves telling the software where certain key points are on the layer and where they actually should be. The entire layer will then be distorted using these references.

Most GIS software packages will also calculate basic statistics about their spatial features. Typical examples include calculating the length of lines, the area and perimeter of polygons, and the distances between points. There are many examples of why these basic measures can be useful. These include measuring distances along a transport network, the use of areas to calculate population densities, and calculating the distances between settlements.

Bringing data together to acquire knowledge

Manipulating the spatial component of a single layer of data is useful, but the full potential of GIS lies in its ability to integrate data from a variety of layers. At a basic level this merely involves combining layers on-screen to compare patterns. This might be as simple as taking a raster scan of a map and placing a vector layer over the top. The raster layer provides a spatial context for the features in the vector layer. Another option is to lay one vector layer on top of another; for example to compare the pattern of roads with the location of farms to see

which farms lie near the major roads. Field boundaries might be a third layer added to this. This approach goes beyond basic mapping, as querying the underlying attribute database allows a detailed understanding of a multifaceted study area to be developed. In this way an integrated understanding of the problem can be derived from many (possibly highly disparate) sources. Healey and Stamp provide an example of this in their study of regional economic growth in the north-eastern United States (Healey and Stamp 2000). They have created a large and comprehensive database that contains industrial plants, such as blast furnaces, foundries and coalmines, represented by points. This is combined with polygon data showing the boundaries of natural resource deposits and, in areas of more detailed study, land parcels. To study the economic development of these they also needed a database of the transport system, and this is reproduced by layers of lines that include the railroad and canal networks, the turnpike roads and the rivers. This can be combined with aggregate, background information, such as that extracted from both the population and industrial censuses.

Formally integrating data through overlay

An overlay operation combines not only the spatial data but also the attribute data. This has many potential uses. For example, a user has a polygon layer containing data about administrative units, such as Irish counties, and wants to find out what proportion of each county was covered by water using a polygon layer showing lakes. An overlay operation would produce a new polygon layer that combined the attributes of both polygon layers and thus each new polygon would have a combination of the county attributes and the attributes from the water layer. It is also possible to combine point or line layers with polygon layers using overlay. For example, as inputs we might have a point layer representing towns and a polygon layer representing counties and we want to determine which towns lay in which county. Overlaying the two layers would produce a point layer with the county polygon attributes added to each point, thus giving us the required information.

Combining buffering and overlay allows complex spatial queries and operations to be performed. For example, with a line layer showing the road network and a point layer containing farm locations, a user may want to calculate which farms lie within 1km of a major road. First, the user selects only the major roads from the road layer and copies these to a new line layer. A buffer is then placed around the new layer so that a polygon layer is created with polygons representing areas within 1km of a major road. If only farms within 1km of a major road are required, then a 'cookie cutter' overlay can be used. In this only farms on the input layer lying within a polygon on the buffer layer will be copied to the final layer. The final layer only contains five of the original farms and will contain all the attributes of both the farms and roads source layers.

Perhaps more than any other GIS operation, overlay tests the accuracy of the input layers. If a point layer is overlaid with a polygon layer then inaccurate polygon boundaries can easily lead to a point lying near a boundary being allocated to the wrong polygon. Accuracy is tested further where two polygon layers intersect. It is unlikely that any two operators will digitise a curve in exactly the same way even if the same source map is used. If this happens

then the overlay operation will lead to sliver polygons being formed. These are very small polygons formed in the manner. This may seem like a trivial problem but is in fact the bane of vector overlay operations. It is possible to attempt to remove sliver polygons automatically. They tend to be long and thin and thus have a small area compared to the length of their perimeter. While this can be used to identify slivers, deleting them can still be problematic as it requires a decision about which boundary should be deleted. The problems caused by sliver polygons will depend on the scale and accuracy of the two sources, and the accuracy of the digitising. If the two layers have both been digitised to a high standard of accuracy from high quality source maps of similar scales, then the problems are likely to be minimal and can usually be solved by automated procedures within the software. If any of these three criteria are not met there is likely to be a significant job tidying the resulting output layer. [The creation of sliver polygons] Overlay can also be performed on raster data providing they use the same pixel sizes. This is sometimes referred to as map algebra as two or more input layers are used to create an output layer whose cell values are calculated based on a mathematical operation between the input layers. An example of this is where cell values on the two input layers are added to calculate values on the output layer. Other mathematical operations such as subtraction and multiplication can also be used, as can a wide range of other logical operations such as Boolean algebra.

When two layers are combined using an overlay operation, the resulting layer will be at best as accurate as the less accurate layer. Unfortunately, the result is likely to be more inaccurate than this as error will be cumulatively added from both layers. This is termed error propagation and means that, as layers are combined, errors and uncertainty can multiply surprisingly quickly. This means that when multiple overlays are performed this must be done with the limitations of all the source layers being borne in mind.

Integrating incompatible polygon data through areal interpolation

A final issue to be discussed is areal interpolation. This commonly occurs where there are two or more polygon layers of socio-economic data and the polygons represent administrative units, which a user wants to integrate. Where the two sets of polygons nest perfectly because the administrative units used were identical, this is a simple operation. Where they do not, for example if we are comparing census data published using registration districts with election data published for constituencies, then overlay does not provide the complete answer as it is uncertain how to allocate data to the resulting polygons. The traditional response to this is aggregation, which results in a loss of spatial detail, something that a GIS approach should attempt to avoid.

Conclusions:

Information from spatially detailed, integrated databases GIS software provides extensive functionality that allows a user to approach his or her dataset in a way that combines the spatial and attribute components of their data. This functionality leads to added value being extracted from an existing dataset. All datasets have limitations, and the extra functionality provided by GIS software allows us to use the data in ways that their creator would never

have envisaged. As a result it is important to consider the limitations of all layers when manipulating them with the GIS. It is also important to consider the limitations of the techniques used on the data, particularly those that integrate data. As long as the results of spatial operations are understood within these limitations, GIS software provides new functionality that should allow new understanding to be derived from spatially referenced data. In this chapter we have started to see the usefulness of the combined spatial and attribute data model used by GIS. This allows data to be queried and integrated in ways that no other approach can manage. The key advantage of this is that it allows the complexity of the data to be handled without undesirable simplification.

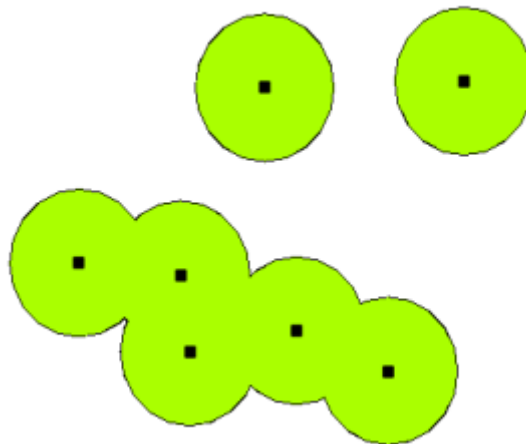
Buffering

Buffering usually creates two areas: one area that is **within** a specified distance to selected real world features and the other area that is **beyond**. The area that is within the specified distance is called the **buffer zone**.

A **buffer zone** is any area that serves the purpose of keeping real world features distant from one another. Buffer zones are often set up to protect the environment, protect residential and commercial zones from industrial accidents or natural disasters, or to prevent violence. Common types of buffer zones may be greenbelts between residential and commercial areas, border zones between countries eg. noise protection zones around airports, or pollution protection zones along rivers.

In a GIS Application, **buffer zones are** always represented as **vector polygons** enclosing other polygon, line or point features (see [figure point buffer](#), [figure line buffer](#),).

Figure Point Buffer 1:



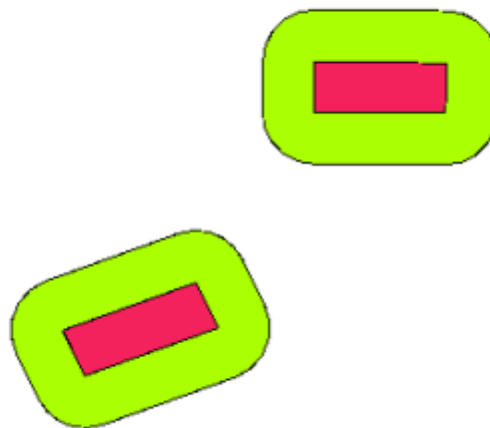
A buffer zone around vector points.

Figure Line Buffer 1:



A buffer zone around vector polylines.

Figure Polygon Buffer 1:



A buffer zone around vector polylines.

Variations in buffering

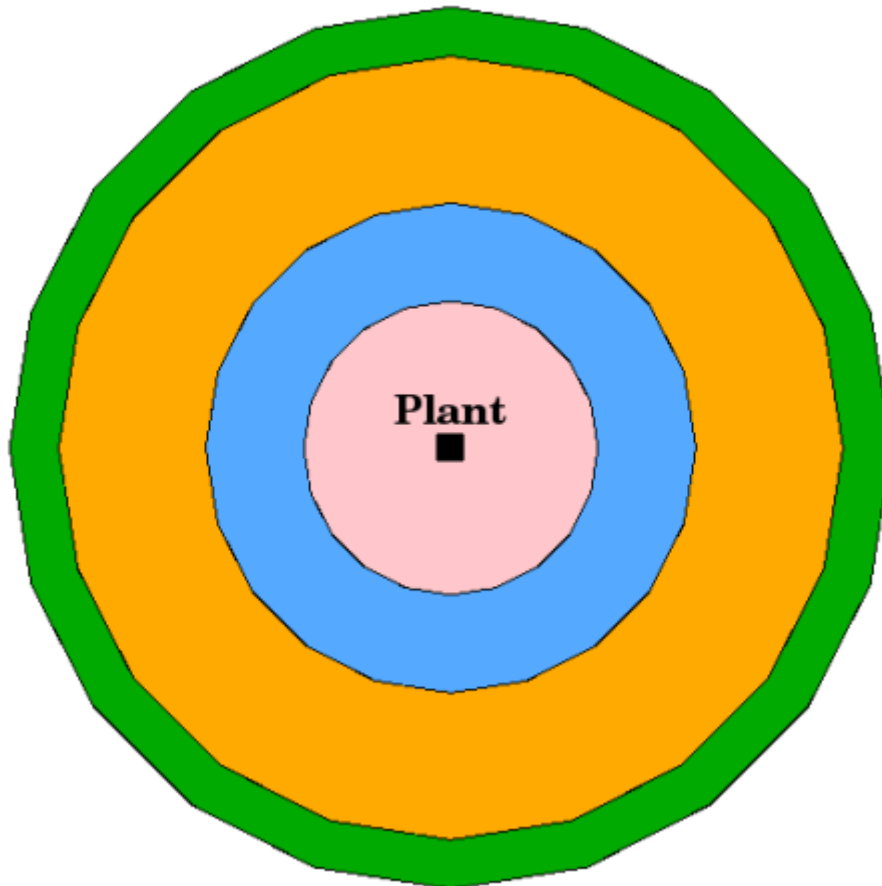
There are several variations in buffering. The **buffer distance** or buffer size **can vary** according to numerical values provided in the vector layer attribute table for each feature. The numerical values have to be defined in map units according to the Coordinate Reference System (CRS) used with the data. For example, the width of a buffer zone along the banks of a river can vary depending on the intensity of the adjacent land use. For intensive cultivation the buffer distance may be bigger than for organic farming. [figure variable buffertable buffer attributes.](#)

Buffers around polyline features, such as rivers or roads, do not have to be on both sides of the lines. They can be on either the left side or the right side of the line feature. In these cases the left or right side is determined by the direction from the starting point to the end point of line during digitising.

Multiple buffer zones

A feature can also have more than one buffer zone. A nuclear power plant may be buffered with distances of 10, 15, 25 and 30 km, thus forming multiple rings around the plant as part of an evacuation plan (see [figure multiple buffers](#)).

Figure Multiple Buffers 1:



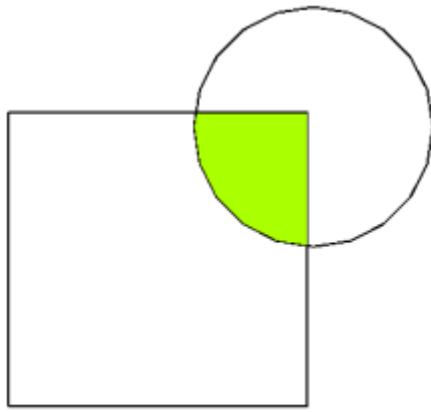
Buffering a point feature with distances of 10, 15, 25 and 30 km.

More spatial analysis tools

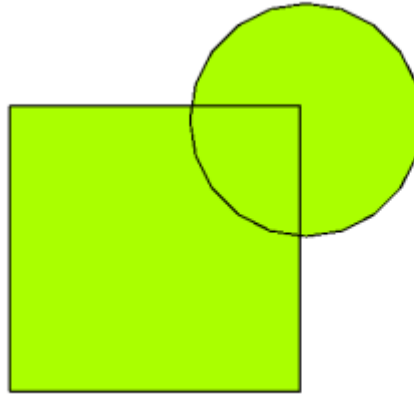
Buffering is an important and often used spatial analysis tool but there are many others that can be used in a GIS and explored by the user.

Spatial overlay is a process that allows you to identify the relationships between two polygon features that share all or part of the same area. The output vector layer is a combination of the input features information (see [figure overlay operations](#)).

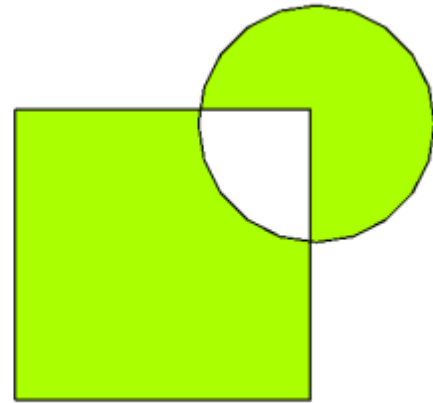
Figure Overlay Operations 1:



Intersection



Union



Symmetrical Difference

Spatial overlay with two input vector layers (a_input = rectangle, b_input = circle). The resulting vector layer is displayed green.

Typical spatial overlay examples are:

- **Intersection:** The output layer contains all areas where both layers overlap (intersect).
- **Union:** the output layer contains all areas of the two input layers combined.
- **Symmetrical difference:** The output layer contains all areas of the input layers except those areas where the two layers overlap (intersect).
- **Difference:** The output layer contains all areas of the first input layer that do not overlap (intersect) with the second input layer.