



---

## Feature: Representing Geographic Features

- *Robert Ischeil Petersen (PAL)*

When faced with storing geographic information one must determine how to represent that information in the storage system. There are a number of factors to consider such as the precision of the location, the coordinate system, the data storage mechanism, the types of features and the degree to which the location data will be specified. This is by no means an exhaustive list but to my mind represents some of the key factors.

The precision of the location information is dependant on the data that is to be referenced. Some data sets are sufficiently represented by location information with precision on the order of a kilometer, while other data require precision down to the centimeter scale. Another consideration is the precision to which a location can be established. This may depend on the equipment used: a legacy dataset many have navigation determined by sextant; GPS suitable for recreation may give a location that is accurate to 10 meters; and differential GPS systems can, with a sufficiently long site occupation, give a location accurate to the nearest centimeter. A storage system for recording this location information must therefore be capable of preserving the precision of the measurement.

While a location can be specified without a coordinate system, e.g. 'the end of the Scripps Institution of Oceanography pier' or 'in the refrigerator behind the ketchup', the efficacy of such references is dubious. More common is a set of coordinates representing the location. For the coordinates to make any sense they must be associated with a well-defined coordinate system. There are a wide variety of coordinate systems available. Some span the globe such as WGS84 or UTM. Other coordinate systems are more locally focused such as the State Plane coordinate system. Still other coordinate systems are unique to a specific application or area of study such as the California Cooperative Oceanic Fisheries

Investigations (CalCOFI) project sampling grid. Regardless of the system used a set of coordinates paired with a coordinate system specifies a well-defined point. As such one can convert coordinates between coordinate systems. The details of such transformations are frequently non-trivial; nonetheless, they are common and in many cases computer codes already exist to facilitate these transformations.

Given that sets of coordinates require a coordinate system to be meaningful, the representation of location must therefore include the coordinate system. This is sometimes not evident when a community uses the same reference system tacitly; that is, it is when faced with data recorded in two different coordinate systems that the differences become evident and can be resolved if the coordinate systems are clear. Some representations explicitly include the coordinate system in the location data and in other cases the coordinate system is fixed for a representation and that system is specified as metadata.

Up to this point I have used the term “location” without a strict specification. It is now time to rectify this oversight. A location can be represented by various geometries. I’ll restrict the discussion here to two-dimensional geometries, though one can readily imagine including additional spatial or temporal dimensions. The simplest geometry is a point. A latitude and longitude pair is an example of a point. While the physical reality of a point, something of infinitesimal extent, may be questionable a point may well be a suitable simplification. A polyline is a collection of many points and includes the intermediate points along a straight line between specified vertices. This type of geometry is suitable for representing features such as the course of a river or track of a vessel. A polygon is like a polyline with the added requirement that the end point connect to the beginning point. It is also possible to have collections of these geometries. These are but a few of the simplest geometries available and give an idea of possible representations.

With these general considerations in place, we may now move from the abstract to the concrete and look at a few of the available representations.

### **Custom Database Storage**

Storing a location in a database is often a convenient way to georeference data that is already in the database. The design of such a database is dependant on all of the above considerations and it is up to the designer to determine how to accommodate the geographic data. When storing a number in the database one must consider the precision requirements. Is an IEEE float sufficient or is a character field a better storage mechanism. Storing all locations as points can be

accommodated with two columns in the case of decimal latitude and longitude or if storing degrees, minutes and seconds, six columns. More sophisticated geometries can be accommodated by leveraging the relational features of a database. The specification of the coordinate system may be universal to the database, e.g. all points are WGS84 or may be stored with each location. The benefits of such a storage system are that one may make the system as simple or complex as is necessary. The proprietary nature of such a system may make interactions with other systems more difficult.

### **Simple Feature**

The “Simple Feature” specification is a standard published by the OpenGIS group. The Simple Feature specification defines an object model for geometries and can accommodate all of the geometries listed above. The standard defines two representations, “Well Known Text” (WKT) and “Well Known Binary” (WKB). While the specification object model accounts for a coordinate system or “Spatial Reference System” the WKT and WKB representations exist without an explicit reference to their coordinate system. Representing the feature as either WKT or WKB in a CLOB or BLOB column respectively could accommodate storage in an RDBMS. Using the WKT representation provides for arbitrary precision while the WKB representation suffers from the IEEE floating point limitations that one would imagine. As with the custom database representation the coordinate system would need to be specified either universally for all features or on a feature-by-feature basis.

### **KML**

KML is a file format defined by Google and used with Google Maps and Google Earth. The format is based on XML. KML can represent all of the geometries listed above. In addition to representing geographic features KML supports the inclusion of additional data such as images and URLs and one can specify the display parameters of the features. KML is frequently stored as stand-alone file and can be imported to Google Earth or displayed by Google Maps using a URL referencing the document. Because it is a text document points can be represented with arbitrary precision. KML only supports the WGS84 coordinate system.

### **ESRI Shapefile**

The Shapefile is the standard format used by ESRI GIS products. The term “Shapefile” is a bit of a misnomer as it is actually composed of several files. Shapefiles can accommodate all of the geometry types listed above. Shapefiles can also contain attributes associated with each feature. Each Shapefile references the

coordinate system used. There are several libraries available that allow one to create and modify Shapefiles.

There are several options for representing geographic features that range from do-it-yourself solution to full-featured representation that include additional data and display information. Of course there are more options than those presented here. The ultimate choice of representation will impact the systems ability to represent the information accurately and interface with other systems. Currently there is no one clear standard and designers need to weigh all factors before settling on a particular implementation.

### **Example Case: Two LTER Oceanographic Sites**

With Ocean Informatics work in conjunction with PAL and CCE LTER sites, where the data types, tool types, and logistics combine to result in a decision not to venture into the full realm of GIS, my work has involved evaluating how best to associate geographic information with biological oceanography datasets. In evaluating our requirements it was found that though the geographic information was recorded in proprietary coordinate systems storing the points according by latitude and longitude is preferred. The classes of features required the use of most of the geometries listed above. An additional consideration was to insure that the data could smoothly transition to a full GIS solution at some point in the future if necessary. Simple Features stored as WKT satisfied these requirements, could be stored in the database as a single column and provided a sufficiently rich representation without the overhead of other representations.

### **Further Reading**

The following links may be of use to readers interested in the more technical aspects of computer storage and georeferencing.

Floating Point Errors - [http://docs.sun.com/source/806-](http://docs.sun.com/source/806-3568/ncg_goldberg.html)

3568/ncg\_goldberg.html Simple Feature -

<http://www.opengeospatial.org/standards/sfa> ESRI Shapefile -

<http://www.esri.com/library/whitepapers/pdfs/shapefile.pdf> KML Documentation -

<http://code.google.com/apis/kml/documentation/>