

An Introduction to Open Source Geospatial Tools

by [Tyler Mitchell](#), author of [Web Mapping Illustrated](#)
GRSS would like to thank Mr. Mitchell for this tutorial.

Geospatial technologies come in many forms, from mapping to data analysis applications. Using them has traditionally required professionals trained in geospatial information management. While geospatial professionals have been the main users and developers of geospatial applications, the landscape has changed dramatically over the past few years.



The development of open source geospatial software is an exciting part of the new geospatial landscape. Open source project offerings cover the spectrum of tools: command-line data conversion, spatially aware enterprise databases, internet mapping applications, desktop Geographic Information System (GIS) applications, geoprocessing libraries, and more. Eager developers, companies and organizations are collaborating on the new generation of geospatial technologies, providing desktop and server-side applications, APIs, and development platforms that are changing the way we work and do business.

A quick glance at websites like [OpenSourceGIS.org](#) garners a massive list of tools that are freely available to you for a wide array of tasks. How do you get started? This article introduces a handful of the applications that are available, grouped according to task. These concepts and applications are discussed further in my book [Web Mapping Illustrated](#).

Data

When talking about geospatial technologies, we usually mean tools that interact with geospatial data. Interaction occurs on several fronts: creating, converting, manipulating, or visualizing data. These are the common categories of tools. Parallel to these are developer-focused tools in the form of programming libraries, allowing a deeper level of interaction.

Just as wood is needed for the carpenter to use his tools, data is the raw material underlying geospatial tools. The increasing availability of geospatial information is transforming business and management in many parts of the world. Companies, communities, and individuals are realizing that they must have geospatial information to understand their world.

How you store data is a critical consideration. ESRI shapefiles are in a file-based format, but for more advanced vector data management capabilities, the strongest option is the spatial extension to the PostgreSQL database, PostGIS (more on PostGIS under *Manipulation*, below).

Creation

When data doesn't already exist, it must be created. Some of the most traditional methods of data creation use a desktop mapping program to draw shapes on pre-existing base maps. For example, using an aerial photograph as a base, you can click on points of interest, or draw a line around a certain area. That data is then saved into a file, inserted as a database record, or transmitted to some other geospatial service. Source data can also be captured using a GPS receiver and exported into a format for another program, or accessed in real time by an application.

In its simplest form information can be made digital (that is, digitized) by scanning photographs or hard copy maps into digital image files. Or, for locational point data, coordinates can be saved into a text file and used by many open source geospatial applications.

Desktop mapping programs often include the ability to draw shapes on top of other maps. The focus of desktop mapping programs is not always data creation, so their ability to do so varies greatly.

[OpenEV](#)

"OpenEV is a library, and reference application for viewing and analyzing raster and vector geospatial data."

If you want to do a quick digitizing job, OpenEV can get you up and running easily. It doesn't have a lot of fancy editing tools, but reads in dozens of raster and vector data types, which you can then use as a base for drawing your own shapes. It has a Python scripting environment and includes many image enhancement tools.

[Quantum GIS](#)

"Quantum GIS (QGIS) is a Geographic Information System that runs on Linux, Unix, Mac OS X, and Windows. QGIS supports vector, raster, and database formats."

It can access PostGIS databases, in addition to dozens of other vector and raster formats. It supports feature labeling and has a great user community. Extensibility is provided through a plugin environment.

Conversion

Many geospatial projects require significant amounts of data conversion. It is not uncommon to spend as much as 80 percent of your time converting data between formats and fine-tuning the way the data is organized. De facto data format standards (for example, ESRI shapefiles for vector data; GeoTIFF for raster/image data) can help you choose a format to use if you are flexible, but depending on the programs used in a project, a particular format may be required.

GDAL/OGR

"GDAL/OGR is a translator library for raster geospatial data formats that is released under an X/MIT-style open source license. As a library, it presents a single abstract data model to the calling application for all supported formats. The related OGR library (which lives within the GDAL source tree) provides a similar capability for simple features vector data."

The most powerful tools for data conversion are part of the GDAL/OGR project. This project comes with several command-line utilities for converting and projecting raster and vector data. The GDAL utilities handle raster data and OGR utilities handle vector data. Both are bundled together under the GDAL project banner.

GDAL/OGR libraries can be used in your own applications that require conversion or data access. The pre-made command-line utilities can also be very helpful. The [FWTools package](#) makes it easy to install these tools. Try using the `gdal_translate` or `ogr2ogr` programs to convert between formats. For example:

To convert a JPEG image into a GeoTIFF file:
`gdal_translate input.jpg output.tif`

To convert an ESRI Shapefile into GML:
`ogr2ogr -f "GML" output.gml input.shp`

AVCE00

"AVCE00 is an ... ANSI-C library that makes Arc/Info (binary) Vector Coverages appear as E00! It allows you to read and write binary coverages just as if they were E00 files. ... For those who do not need a library but simply want to convert some coverages, the package includes the AVCIMPORT and AVCEXPRT conversion programs."

This is a handy tool because many people think they need to have Arc/Info to convert E00 export files into coverages. Arc/Info export files are one of the formats not supported by GDAL/OGR. You can use the `avcimport` program to import E00 files and create binary coverages with them. Then OGR utilities can be used to access or convert the coverages into other formats.

Manipulation

Of course, data is not always ready to use, even if it has been properly converted. Data often requires manipulation. There are many types of manipulation that may be needed, such as removing unwanted features, adding fields, changing attribute values, clipping features with other features, creating buffered polygons from a line, and so on.

There are two sets of tools that cover a large portion of geospatial data manipulation needs. Both can use the extended power of the GEOS libraries for advanced geometric operations.

[GDAL/OGR](#)

The GDAL/OGR command-line utilities are not just good for converting data, but can also manipulate raster and vector datasets. The `gdal_translate` and `ogr2ogr` utilities have several options for selecting subsets of information or making changes in the output files. They can also be used to project data into particular spatial reference systems (AKA projections). For example:

Use an SQL-like `SELECT` statement to only output certain features into a new file:
`ogr2ogr -select "county = 'Oakland'" output.shp input.shp.`

Clip an image to only output the portion of an image within a certain geographic area:

```
gdal_translate -projwin -122 45 -120 55 input.tif output.tif
```

[PostGIS](#)

Conversion tools like `ogr2ogr` can take vector data files and export them into a PostGIS-enabled database. Because PostGIS is a PostgreSQL database that can also store geometric (spatial) data types, it is an excellent way to bring tabular and spatial data together into a common management environment.

PostGIS has the ability to manipulate data as well as store it. This provides GIS-like abilities within an SQL database environment. The SQL functions include `buffer`, `intersection`, `within`, `distance`, and more. These functions take geometric data from columns in PostGIS tables and return new geometries or other information. For example, the `distance` function will compute the distance between spatial features, and the `buffer` function will return a new geometry that is a polygon buffered at a certain distance from the source feature.

Here is an example of an SQL query that selects a particular point feature and applies a radial buffer of 100 meters to create a polygon:

```
SELECT buffer(geo_data,100)
FROM city_points;
```

[Geometry Engine Open Source \(GEOS\)](#)

If you are writing your own applications, particularly in C++, you can use GEOS libraries to give you spatial manipulation capabilities. Both GDAL and PostGIS can use GEOS to allow advanced capabilities of feature manipulation. Some of the PostGIS functions mentioned above require the GEOS libraries. GDAL can use GEOS functions if you use GDAL in a programming environment.

Visualization

Projects that have a mapping component need some sort of visual output. The output could be a graphic file or paper printout.

All the applications mentioned under the *Creation* section above have the ability to draw and color map data. There are many other options, some of which also include editing functionality. Although there are desktop mapping environments available, interactive web mapping is certainly the topic of the day.

[MapServer](#)

This popular internet mapping application has both programming interfaces and a web CGI mode. Programming interfaces exist for several languages. It also includes command-line tools for creating static image files of maps, legends, scale bars, and so on.

There are a couple of *frameworks* for building MapServer applications using PHP. Consider using one of these to get started quickly:

- [MapBender](#)
- [Chameleon](#)

Other Applications

Many people are using other applications to meet their daily needs. Here are some more options.

[GRASS GIS](#)

"The Geographic Resources Analysis Support System, commonly referred to as GRASS GIS, is a Geographic Information System used for data management, image processing, graphics production, spatial modelling, and visualization of many types of data."

This full-fledged, desktop GIS has been used for many years by the government, academia, and industry to do sophisticated analysis and mapping.

[OpenMap](#)

"Using OpenMap, you can quickly build applications and applets that access data from legacy databases and applications. OpenMap provides the means to allow users to see and manipulate geospatial information."

This JavaBean-based toolkit includes many of the standard desktop GIS features, including visualization and editing.

[Generic Mapping Tools \(GMT\)](#)

"GMT is an open source collection of 60 tools for manipulating geographic and Cartesian data sets (including filtering, trend fitting, gridding, projecting, and so

on) and producing Encapsulated PostScript File (EPS) illustrations ranging from simple x-y plots via contour maps to artificially illuminated surfaces and 3-D perspective views."

These are command-line tools that use text configuration files, and are largely used by the oceanographic science community.

[uDig](#)

"User-friendly Desktop Internet GIS (uDig) is an open source spatial data viewer/editor, with special emphasis on the OpenGIS standards for internet GIS, the Web Map Server and Web Feature Server standards. uDig will provide a common Java platform for building spatial applications with open source components."

This is a Java (Eclipse)-based application with many features, and is highly extensible.

[JUMP](#)

"The JUMP Unified Mapping Platform (JUMP) is a GUI-based application for viewing and processing spatial data. It includes many functions common to other popular GIS products for the analysis and manipulation of geospatial data."

This feature-rich desktop GIS environment can create, edit, and manipulate data. It can do much more through additional plugins. JUMP is a Java application and has several levels of programming interfaces available.

Conclusion

Open source geospatial applications and programming environments can fill all of the standard components of a geospatial project. The geospatial landscape is becoming rich with choice.