Making Magnetic Maps

01/08/14

Introduction

Geologists and geophysicists make and use maps. These maps must have a common coordinate system. Otherwise, it becomes very difficult to compare information and/or collate observations. In the past, a common coordinate system for most maps simply was not easy to achieve. During the 1990's, the advent of widely available global positioning system (GPS) receivers made it possible to achieve commonality and cross-referencing among coordinate systems. The downside, if there is one, is that we have to be much more careful about the way we gather and use data that is destined to be displayed on maps.

Ellipsoids, Datums, and Geoids

Earth is a geometrically complex object. The easiest way to approximate the shape of Earth is with a sphere. A sphere has constant radius and therefore departs from the actual shape of Earth fairly dramatically. The equatorial radius of Earth is approximately 6378.139 km, the polar radius of Earth is approximately 6356.75 km, a difference of 21.38 km. Consequently, the figure of the Earth is more precisely approximated by an oblate ellipsoid of revolution. The planet is flattened along its axis of rotation, its equatorial bulge a result of rotation. The amount of flattening of the ellipsoid can be expressed in terms of equatorial and polar radii:

$$f = \frac{R_{eq} - R_{po}}{R_{eq}} \tag{1}$$

where R_{eq} is the radius of Earth at the equator and R_{po} is the radius at the pole. In many map projection formulas, the eccentricity of the ellipsoid is used, rather than flattening.

$$e^2 = 2f - f^2 \tag{2}$$

where e is the eccentricity. If Earth were a body of homogeneous density rotating in space, the ellipsoid of revolution should describe its shape perfectly. This is not the case. There are substantial regional departures from the ideal ellipsoidal shape. A best-fit ellipsoid can be used to approximate the shape of the Earth. Such an ellipsoid is called a reference ellipsoid. As discussed in the following, different reference ellipsoids are in common use.

Departures of the figure of the Earth from the reference ellipsoid occur primarily because of large scale density differences within Earth. These density differences cause undulations in the shape of Earth. So a further surface can be defined that accounts for these regional undulations. This surface is called the geoid. The two main characteristics of the geoid are:

- Earth's gravity field is perpendicular to the geoid everywhere - making the geoid an equipotential surface.
- The geoid coincides with the theoretical position of the surface of Earth's oceans at rest.

The geoid could be any equipotential surface. On Earth, it is simply convenient to refer the geoid to mean sea-level. The elevation of the geoid commonly deviates from the reference ellipsoid by up to 40 m. In some locations, such as off the coast of India, the geoid differs from the ellipsoid by more than 100 m. This means that if you sail on a ship from Madagascar to Bombay, you travel through a trough 100 m deep with respect to the reference ellipsoid. Actual topography varies with respect to both the ellipsoid and the geoid. Usually, elevation is referenced to the geoid, as in *meters above sea-level*, but it is possible to reference topography to the reference ellipsoid.

Any geographic location on the surface of Earth can be expressed as a latitude and longitude. While elevation is usually referenced to the geoid, latitude and longitude are referenced to the ellipsoid. Two things are needed to express a position on the surface of Earth in terms of latitude and longitude:

- An origin for the coordinate system
- An equation for the reference ellipsoid

The origin is agreed upon by convention. Zero latitude corresponds to the equator (the plane orthogonal to the axis of Earth's rotation). Values of latitude vary from 0° N at the equator to 90° N at the north pole and 90° S at the south pole. Often, especially in many computer programs, latitude is represented as positive in the northern hemisphere and negative in the southern hemisphere. That is latitude varies from $+90°$ at the north pole to $-90°$ at the south pole. Longitude is more arbitrary. Zero longitude is designated as the meridian of the astronomical observatory in Greenwich, England. Longitude increases to the east (just like on a familiar xy -plot with positive values increasing to the right). The eastern hemisphere extends to a longitude of 180° E. Three different ways are used to represent longitude in the western hemisphere. The simplest approach, and most rarely used, is to simply let longitude values increase eastward until reaching 360° as the meridian at Greenwich, England, is approached from the west. In this system, longitude is uniquely expressed as any number, $0° - 360°$. Alternatively, longitude in the western hemisphere can be assigned a negative number, in which case longitude varies from $-180°$ to $+180°$. The most common means of expressing longitude, and the least useful for computations and in computer programs, is to express longitude using E and W to designate eastern and western hemisphere, respectively. In this system, longitude varies from 180° W to 180° E. Many computational errors have been made by forgetting which system is being used to designate longitude.

Various equations for the reference ellipsoid are in use. These equations differ in the average radius of the Earth chosen and the eccentricity (or flattening) of the ellipse. In the context of map coordinates, these different reference ellipsoids are referred to as different map datums. Three common map datums are described in Table 1.

Table 1: Common map datums and their parameters

Ellipsoid (Datum)	R_{eq} (m)	$1/f$
Clarke 1866 (NAD27)	6378206.4	294.9786982
WGS84	6378137.0	298.257223563
GRS 1980 (NAD83)	6378137.0	298.257222101

NAD27 refers to the North American Datum of 1927; NAD83 refers to the North American Datum of 1983; WGS refers to the World Geodetic System; GRS refers to Geodetic Reference Systems. These are the most common ellipsoids (sometimes called spheroids) and map datums that you are likely to encounter, but there are very many more. Prior to the mid-1980s, map datums and reference ellipsoids were derived from regional surveys. These ellipsoids were optimized, naturally enough to fit the figure of Earth in the regional area of interest. So, for example, the Clarke 1866 ellipsoid was derived for North America and its datum referenced to a location in central Kansas (meaning that the fit is optimal at this location). Departures from the geoid using Clarke 1866 NAD27 are minimized for North America, but are quite large for other parts of the globe. In the mid-1980s a global best-fit ellipsoid could be calculated using satellite data. The datum for this ellipsoid is Earth-centered, in other words, the center of Earth is essentially the datum's *origin*, and departure of the ellipsoid from the geoid is minimized globally.

The WGS84 datum is now commonly used, worldwide. Nevertheless, a large number of US maps in print were constructed using the Clarke 1866 ellipsoid and the NAD27 datum. In other parts of the world, other local ellipsoids and datums were used. When different datums are in use, it is crucial to understand that *a given location at the surface of the Earth will have a different latitude and longitude depending on the datum used.* The difference between WGS84 and NAD27 is large in a state like Florida, that is far from the NAD27 origin. In Florida, the difference between NAD27 and WGS84 is about one second of latitude.

All this has two practical results for people who gather and use map data. First, the ellipsoid and datum used must be reported when map coordinates are given. Otherwise the data cannot be associated to a single specific location on the surface of Earth. Second, GPS receivers are very often used to determine position. Most GPS receivers will report position (*e.g.,* latitude and longitude) using one of any number of datums. If these data are to be plotted on a map, the datum used by the GPS receiver must match the datum used to create the map, otherwise the location of the data will be incorrectly plotted on the map. It is a very simple procedure to note the map datum - surprisingly this information often is not noted, resulting in errors that can be very difficult to identify and fix later.

Grids

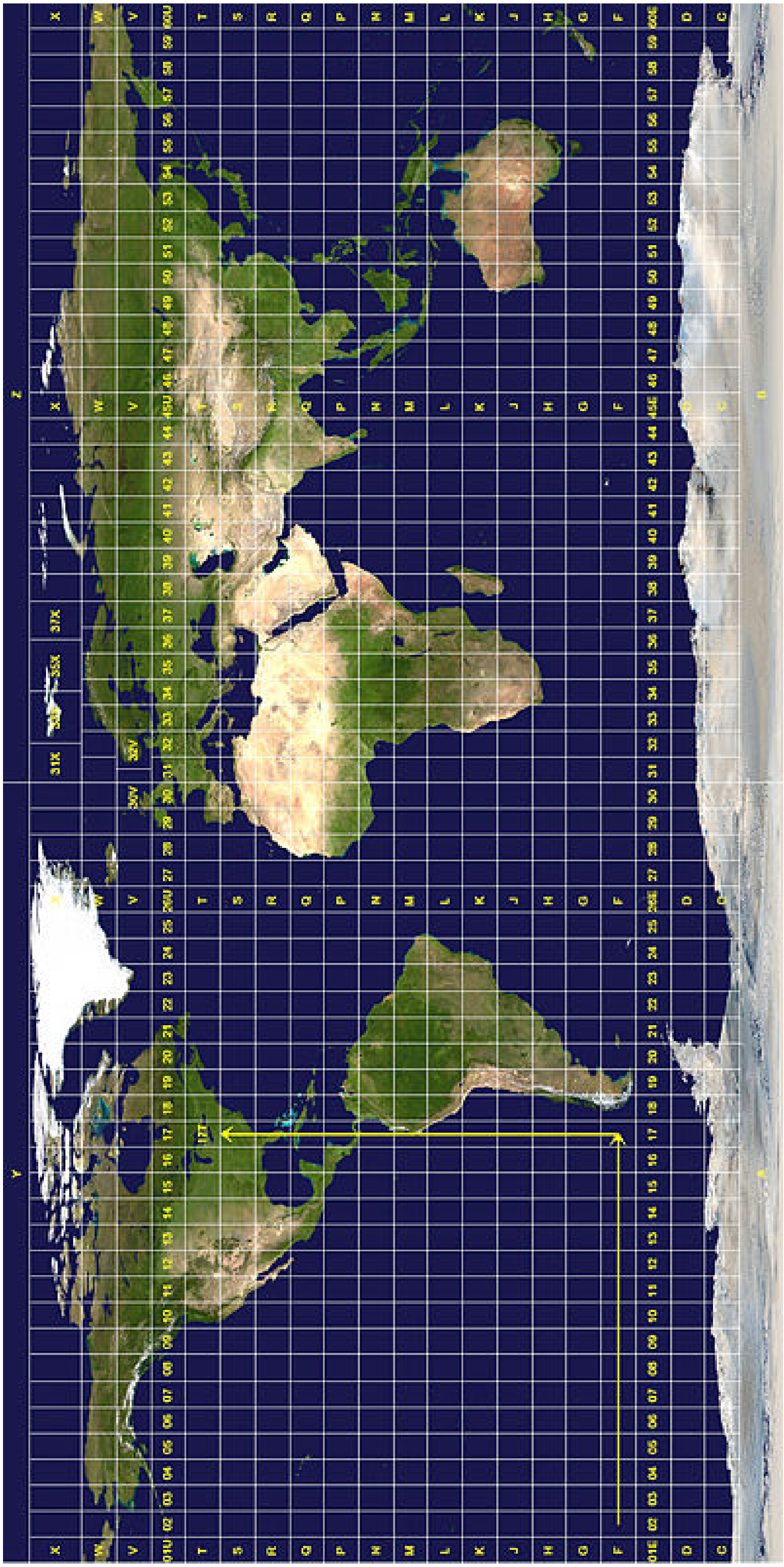
Latitude and longitude coordinates provide a reasonable way to report position on the reference ellipsoid. But this coordinate system has serious drawbacks. Just how far is it between 56°N, 47°W and 56.1°N, 46.9°W? In what direction would you walk to go from 56°N, 47°W toward 56.1°N, 46.9°W? Such distances and directions are difficult to calculate on a sphere, much less on an oblate ellipsoid.

Rectangular grid systems have been developed to simplify distance calculations on maps. These are Cartesian grids. The Y-axis is oriented N–S and Y-coordinates are usually referred to as the Northing, or sometimes as distance north. The X-axis is oriented E–W and X coordinates are referred to as the Easting or distance east. Usually the units of these grids are meters, or occasionally (unfortunately) feet.

The most important grid in use today is the Universal Transverse Mercator (UTM) grid - not to be confused with the Universal Transverse Mercator map projection. The Northing for the UTM grid in the northern hemisphere is given in meters north of the equator. Negative numbers are avoided at all costs in the UTM grid. So, in the southern hemisphere, the Northing is given in meters south of the equator (a negative number) + 10 000 000 m. For example, the Northing of a point located 2000 km north of the equator is 2 000 000 N. The Northing of a point located 2000 km south of the equator is 8 000 000 N. So in both the northern and southern hemispheres, UTM Northing is a positive number that increase as one walks north and decrease as one walks south.

The X-axis is perpendicular to the Y-axis at the origin (central meridian) of the coordinate system. Far from the central meridian, this results in considerable distortion of the grid. Consequently, the UTM grid breaks up Earth's surface into 60 zones, each with its own central meridian. Again, negative numbers are avoided in the UTM grid. Consequently the value of the Easting along the central meridian (the origin) of any zone is 500 000 E, rather than 0 E. West of the central meridian the Easting is less than 500 000 E; east of the central the Easting is more than 500 000 E. There are enough zones so that, using this scheme, no negative Easting coordinates occur on Earth. Note that there are exactly 60 places on the surface of Earth that have the UTM grid coordinate 2010222 N, 342343 E. When UTM coordinates are used, the zone must be reported. These zones are uniquely numbered 1-60 (see next page). As with latitude and longitude coordinates, the ellipsoid and datum must also be known. Like latitude and longitude, *the UTM coordinate of a given point on the surface of Earth is different for different ellipsoids and datums.*

The UTM grid is ideally suited for maps of relatively small areas (large scale maps). Large regions are more likely to cross UTM zones and this leads to problems. GPS receivers are normally capable of reporting position using the UTM grid or a variety of alternative grids. Different grids are sometimes reported on maps - particularly old maps. These alternative grids include the British Grid, and the State Plane Coordinate


Y
Z
A
B
X
W
V
T
S
R
Q
P
N
M
L
K
J
H
G
F
D
C
01U
01E
02
03
04
05
06
07
08
09
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26U
26E
27
28
29
30
30V
31
31X
32
32V
33
33X
34
35
35X
36
37
37X
38
39
40
41
42
43
44
45U
45E
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60U
60E
17T

system. These are all constructed with the same basic idea, but use different origins. For science, use a UTM grid.

Converting from Latitude/Longitude to UTM coordinates and the like

Given the differences in map datums and in coordinate systems, it is essential for geologists and geophysicists to easily convert coordinates from one datum to another, or from one grid to another. Fortunately there are tools available for doing this conversion. One of the most reliable tools was developed by staff at the US Geological Survey. This conversion tool is the Proj.4 Cartographic Projections Library. It is freely available, see the website:

`http://trac.osgeo.org/proj/`

for current versions and downloads of Proj.4. Once installed, Proj.4 is useful for solving coordinate system issues.

For example, suppose a colleague gives you the latitude and longitude of 82° W, 27° N (NAD27 datum) for a work site near Tampa, Florida. The *cs2cs* program can be run to convert this coordinate to the NAD83 datum:

```
type:     cs2cs +proj=latlong +datum=NAD27 +to +proj=latlong +datum=NAD83 -f "%0.6f"
type:     -82 27
```

Output of the *cs2cs* code using these parameters will be:

```
-81.999809      27.000336
```

That is, 82° W, 27° N (NAD27 datum) converts to 81.999809° W, 27.000336° N (NAD83 datum).

Proj.4 can also convert from latitude/longitude to the UTM grid system using the program, *proj*:

```
type:     proj +proj=utm +datum=WGS84 +zone=12 -f "%.0f"
type:     111d17'55"W 38d34'N
```

Note that the datum (in this case WGS84) and UTM zone (in this case zone 12) must be specified. Output of *proj* is:

```
473985      4268733
```

corresponding to 473985 E, and 4268733 N in the UTM grid for zone 12 (WGS84 datum). Similarly, to convert UTM coordinates to Latitude and Longitude, one can use:

```
type:     proj -I +proj=utm +ellips=WGS84 +zone=12  -f "%.6f"
type:     473985    4268733
```

yields:

```
-111.298620    38.566665
```

which corresponds to 111.298620° W, 38.566665° N (WGS84 datum). The Proj.4 Cartographic Projections Library contains vast resources for making such conversions, and can easily be applied to whole data files. Note, the *cs2cs* and *proj* programs can understand numbers in decimal degree format and in degrees/minutes/seconds format.

Map Projections

A map projection transfers the coordinates of points located on the reference ellipsoid to a flat surface - a map. A huge number of map projections are in frequent use and essentially an infinite number can be derived. Snyder (1982) discusses the mathematical basis of various map projections in detail. He broadly states that a map projection can be chosen on the bases of:

- Area: many types of map projection are equal area. A given area on one part of the map corresponds to an actual area on the surface of the Earth. This ratio is constant across the entire map. This is done at some expense - scale and angles are distorted on equal area map projections.
- Shape: Many map projections preserve shape and angle. These map projections are often called conformal
- Scale: No map projection preserves scale. That is, scale is not the same from one part of the map to another.

For large scale maps showing small areas, the differences between projections are fairly trivial. For large areas - like continents or ocean basin, shown on small scale maps, the differences are obvious and important to take into account. Snyder (1982) suggests appropriate map projections for specific uses. Here is a brief synopsis:

- To show the whole Earth in conformal projection - Mercator
- To show the whole Earth in equal area projection - Hammer or Eckert IV or VI
- To show a continent or ocean scale near the equator with predominant E-W extent in conformal projection - Mercator
- To show a continent or ocean scale along the equator with predominant E-W extent in equal-area projection - Cylindrical equal area
- To show a continent or ocean scale along the equator with predominant N-S extent in conformal projection - Transverse Mercator
- To show a continent or ocean scale along the equator with predominant N-S extent in equal area projection - Transverse Cylindrical Equal Area
- To show a continent or ocean scale away from the equator with predominant E-W extent in conformal projection - Lambert Conformal Conic
- To show a continent or ocean scale away from the equator with predominant E-W extent in equal-area projection - Albers Equal-Area Conic

...and there are many more!

Reference

Snyder, John Parr, Map Projections Used by the U. S. Geological Survey, 2nd edition, Geol. Survey Bulletin 1532, 313 p., U. S. Government Printing Office, Washington, D. C., 1982.

A GMT review

GMT stands for Generic Mapping Tools. GMT is a suite of programs developed by two graduate students at Columbia University - Paul Wessel and Walter Smith. Although they are long done with graduate school, their program is following them around. Lesson 1 - never write a computer program that other people will find useful! You will spend a great deal of time improving it for other people. Lesson 2 - if someone writes a useful program - USE IT! GMT continues to be an incredibly useful mapping tool for geologists and geophysicists. It is used to create location maps, plot data in numerous formats (xy plots, contour maps) and to do simple data analysis. Some of the geophysics tools in GMT are quite sophisticated. You can think of GMT as a geographic information system - like Grass, ArcView and similar products - only tailored to suit the needs of people who plot and present geological and geophysical data on maps. An important difference between GMT and GIS software is that the output from GMT is postscript. Postscript is a language that is used and interpreted by other programs (like Ghostview, Evince, and Adobe Acrobat Reader) to plot images on the computer screen and to print them. One important thing about postscript is that you will not be able to manipulate images easily, once the postscript is created. Also, unlike GIS software, you cannot query maps once the postscript is created. Occasionally this is a disadvantage, but for the great majority of applications in geology, this is not a serious drawback. Another important difference between GMT and many software products is that it is freely available to use and abuse. You can download it onto any computer you wish.

Why use GMT?

We will use GMT in this class to plot and view magnetic data on graphs and maps. See Figures 1 and 2 for two example maps created using GMT. GMT will be an indispensable tool for displaying complex maps, like contour plots, location data, etc.

Simple example using GMT (version 5)

The simplest way to use GMT is to just type the GMT command directly on the command line. Spaces act like characters so be careful when you use them. In this document the space character will be explicitly designated by this character: ␣ .If typed on the command line:

```
gmt␣pscoast␣-JM6i␣-R-129/-115/42/52␣-Df␣-W0.5p,black␣-G230␣-B2␣-P␣-V␣>␣map.eps
```

the GMT program `pscoast` will execute and output will be directed to a file `map.eps`. Try it! Now let's convert this image *eps* file to a *pdf* file so you can quickly and easily view your map:

```
gmt␣ps2raster␣map.eps␣-A␣-P␣-Tf
```

To view your map double click on it's icon or type:

```
evince␣map.pdf
```

`evince` is a *pdf* viewer found on most linux systems. You can use any *pdf* file viewer (*e.g.*, Adobe Reader) to look at the *pdf* version of your map. You can also edit (or make changes) to your map in a program that allows editing of encapsulated postscript (eps) files (*e.g.*, adobe illustrator). *eps* files use *vector* graphics and by opening the *map.eps* file, it is possible to label the x-axis and y-axis, to label features on your map, to adjust line thicknesses, to overlay additional map data, etc. The possibilities are only limited by your imagination. Remember, a picture is worth a thousand words, *but only if you design it well.*

Another GMT program that is useful when making maps is `psxy`. This program plots points or locations on a map. Let us add a useful location to our map.eps file. First create a text file, and name it *seattle.xy*, with the following two numbers typed on one line and separated by a space:

```
-122.333056␣47.609722
```

As you have probably guessed, these two numbers represent a location in units of degrees. Now, we need to execute our `pscoast` command again, with a slight change to the command line:

```
gmt␣pscoast␣-JM6i␣-R-129/-115/42/52␣-Df␣-W0.5p,black␣-G230␣-B2␣-P␣-V␣-K␣>␣map.eps
```

Now, execute the following line:

```
gmt␣psxy␣seattle.xy␣-J␣-R␣-St0.4c␣-Gred␣-O␣-V␣>>␣map.eps
```

and again convert your new *map.eps* file to a *pdf* file for viewing:

```
gmt␣ps2raster␣map.eps␣-A␣-P␣-Tf
```

and open the *pdf* file to view the results. GMT also has another command for labeling locations or points drawn on the map, `pstext`. First, modify the line in the *seattle.xy* file as shown:

```
-122.333056␣47.609722␣Seattle
```

Type all four GMT commands again. The `pscoast` command line stays the same, but note the change to the `psxy` command line:

```
gmt␣pscoast␣-JM6i␣-R-129/-115/42/52␣-Df␣-W0.5p,black␣-G230␣-B2␣-P␣-V␣-K␣>␣map.eps
gmt␣psxy␣seattle.xy␣-J␣-R␣-St0.4c␣-Gred␣-O␣-V␣-K␣>>␣map.eps
gmt␣pstext␣seattle.xy␣-F+a0+f12p+jLB␣-J␣-R␣-O␣-V␣>>␣map.eps
gmt␣ps2raster␣map.eps␣-A␣-P␣-Tf
```

A critical step in using GMT effectively is to understand the parameters that are passed to each GMT command. Some parameters are required, others are optional. Notice that the `-J` and `-R` parameters are used by each of the GMT commands. The -J parameter specifies the map projection and the map scale. In this example the `M` directs `pscoast` to draw the map using an equal area Mercator projection; the `6i` sizes the map to be 6 inches wide. The `-R` parameter specifies the region of interest or map boundaries. The order of the boundaries is always: `West/East/South/North`. These are the most important GMT parameters and used by almost all GMT commands.

Two other important parameters are the `-K` and the `-O`. Notice that these parameters must be included when your map involves more than one GMT command (in this example 3 commands were used). The `-K` parameter must be included with each command except the last one. The `-O` parameter must be included with each command after the first command. These parameters control the operation and synchronization of multiple commands. So if you can not view your map always check that the `-K` and the `-O` parameters are where they should be.

Each GMT command has additional parameters that adjust the output. Here is a brief listing of the parameters used in the examples:

-Df Use the full-resolution coastline database *(pscoast)*

-W0.5p,black Use a black, 0.5pt line thickness for outlining the coastlines

-G230 Use a 230 (grayscale) color for land masses *(pscoast)*

-B2 Annotate the x- and y-axes every 2 degrees

-P Plot the map in portrait mode

-St0.4c Plot map data using a triangle symbol that is 0.4 cm *(psxy)*

-Gred Use a symbol fill color of red

-F+a0+f12p+jLB Format text: 0 rotation, 12pt default font, attach LeftBottom edge of label to its location *(pstext)*

-A Adjust bounding box to minimum space around image *(ps2raster)*

-Tf Output image file type will be a single page PDF file

-V Print out messages, warnings, and errors to the screen when running this command

One other important point, notice the `>` and the `>>` symbols. The `>` symbol redirects output to a file. In this case the output is postscript code that defines the final map image. Try running the `gmt␣pscoast` command without the redirection symbol. You will see lines and lines of postscript code. This postscript code is interpreted by postscript viewers and certain printers to produce images. Now notice the `>>` symbol. This symbol is similar to but not the same as the `>` symbol. The `>>` symbol appends information to a file. So, for a series a GMT commands, the first command creates a postscript image file and each subsequent command adds additional information to this postscript image file. Guess what will happen if you use the `>` symbol instead of the `>>` symbol. Try it! This is a very common mistake, so be ready for to check for it if your output is not what you expect.

The best place to get information about the various GMT commands is from the GMT website: http://gmt.soest.hawaii.edu/ Here you can find Frequently Asked Questions (FAQ), documentation about all the commands and their parameters, and code snippets. Each GMT command has a manual page that can be accessed online or from your own command line. The online (html) version is more readable than the command line version but they both contain the same information. To access the command line manual page for `pscoast` just type `man␣pscoast`. To find online information just *google* the command you need information on.

What is GDAL?

GDAL or **Geospatial Data Abstraction Library** is a translator library for raster geospatial data formats. GDAL comes with a variety of useful command-line utilities for data translation and processing. You can use GDAL on all modern flavors of Unix: Linux, Solaris, Mac OS X, Solaris; and most versions of Microsoft Windows. Both, 32-bit and 64-bit architectures are supported. Most importantly, GDAL is freely available on the WEB. Check out the main GDAL website: http://www.gdal.org/ for more information and downloads.

Three useful GDAL programs for exploring map data

Data are supplied in a variety of flavors and image formats. Remembering and understanding all of the available formats can be a daunting task. The GDAL suite of programs will allow us to easily read image formats, re-project image data, and convert images to different image formats. Let's look at three useful GDAL programs: ***gdalinfo, gdalwarp, and gdal_translate***. These programs will assist us in exploring map data that is freely-available on the web.

gdalinfo The *gdalinfo* program lists various information about a GDAL supported raster dataset. This should be the first program you use after downloading some image data. Code execution is similar to GMT command-line programs:

```
gdalinfo␣-mm␣-proj4␣imagefile
```

Notice the two command-line parameters:

-mm Show the min/max values for each band in the dataset.

-proj4 Show the PROJ.4 string corresponding to the file's coordinate system.

`gdalinfo` has additional command-line parameters that you can read about on the GDAL website or using `man␣gdalinfo` (if GDAL is installed on your computer). Here is an example of some output:

```
gdalinfo ~/openev/utm.tif
Driver: GTiff/GeoTIFF
Size is 512, 512
Coordinate System is:
PROJCS["NAD27 / UTM zone 11N",
    GEOGCS["NAD27",
        DATUM["North_American_Datum_1927",
            SPHEROID["Clarke 1866",6378206.4,294.978698213901]],
        PRIMEM["Greenwich",0],
        UNIT["degree",0.0174532925199433]],
    PROJECTION["Transverse_Mercator"],
    PARAMETER["latitude_of_origin",0],
    PARAMETER["central_meridian",-117],
    PARAMETER["scale_factor",0.9996],
    PARAMETER["false_easting",500000],
    PARAMETER["false_northing",0],
    UNIT["metre",1]]
Origin = (440720.000000,3751320.000000)
Pixel Size = (60.000000,-60.000000)
Corner Coordinates:
Upper Left  (  440720.000, 3751320.000) (117d38'28.21"W, 33d54'8.47"N)
Lower Left  (  440720.000, 3720600.000) (117d38'20.79"W, 33d37'31.04"N)
Upper Right (  471440.000, 3751320.000) (117d18'32.07"W, 33d54'13.08"N)
Lower Right (  471440.000, 3720600.000) (117d18'28.50"W, 33d37'35.61"N)
Center      (  456080.000, 3735960.000) (117d28'27.39"W, 33d45'52.46"N)
Band 1 Block=512x16 Type=Byte, ColorInterp=Gray
```

gdalwarp The `gdalwarp` program warps or re-projects an image into a new coordinate system. The program can re-project map data to any GDAL-*supported* projection *(see the* `-of␣format` *option below)*. This program is useful for converting geodetic coordinates (latitude, longitude) to Cartesian coordinates (*e.g.*, UTM). Consider this command line example:

```
gdalwarp␣-t_srs␣"+proj=utm␣+zone=10␣+datum=NAD83␣+ellps=GRS80"\
-tr␣30␣30\
-r␣bilinear\
-of␣netCDF␣imagefile.tiff␣imagefile.grd
```

Notice the function of the command line parameters:

-t_srs `"+proj=utm␣+zone=10␣+datum=NAD83␣+ellps=GRS80"` The output file will be a UTM zone 10 projection using the North American Datum of 1983 (NAD83) and the Geodetic Reference System 1980 (GRS80). The parameters between the double quotes are the same parameters used by PROJ.4, a library of routines for performing coordinate conversions between cartographic projections. It is also possible to specify the output file type in other ways, but if you know the PROJ.4 command line parameters it is convenient to be able to use them directly.

-tr 30 30 Resample the map to a grid size of 30 meters by 30 meters.

-r bilinear Use a bilinear resampling method. Other resampling options are available and these are listed on the `gdalwarp` man page.

-of netCDF The output file format will be a netCDF *(.grd)* file. This map grid type is used by most GMT programs. A list of all possible output grid types known to the `gdalwarp` program can be found by typing: `gdalwarp␣--formats`. The default format is GeoTIFF.

gdal_translate The `gdal_translate` program converts raster data between different formats, potentially resampling and/or rescaling pixels in the process.

Plotting a magnetic map

As an example, let us use these tools to plot an aeromagnetic map of Seattle, WA. High-resolution aeromagnetic data were collected by the US Geological Survey (USGS) in order to evaluate seismic hazards for the region. These data have been gridded by Blakely and colleagues and are available for download at: `http://pubs.usgs.gov/of/1999/of99-514/`. The data are provided as 1:100,000-scale quadrangle maps in Geosoft Grid Exchange Format (GXF), a standard ASCII format. The goal is to replot the map of Seattle in GMT and include the US coastlines contained within the map area. The GMT and GDAL tools above will allow you to make this plot. In order to plot coastlines in GMT the data must be in longitude/latitude format. Follow these steps:

Step 1: Navigate to the above-mentioned webpage and obtain the digital data for the Seattle quadrangle. Download this data to a clean directory and gunzip the seattle.gxf.gz file. If you are not familiar with `gunzip` use its manpage: `man␣gunzip`.

Step 2: Use `gdalinfo␣-mm␣-proj4␣seattle.gxf` to list the specifics of the map datum and coordinate system and to get the minimum and maximum values of the data.

Step 3: Use `gdalwarp␣-t_srs␣WGS84␣-of␣netCDF␣seattle.gxf␣seattle.gxf.grd` to transform the gxf file into a WGS84 latitude longitude netCDF grid file. This type of grid file is readable by most GMT commands. Now use the `gdalinfo` command on this new file to obtain the parameters for plotting with GMT. Specifically note the grid spacing and the units, the map boundaries, and the minimum and maximum magnetic data values (nT).

Step 4: A GMT script has been provided that will help you make this plot. Download the script file and its configuration file from canvas: `plot_anomaly.gmt.pl` and `plot_anomaly.conf`. You only need to edit the configuration file (`plot_anomaly.conf`) with the parameters for plotting the Seattle aeromagnetic map. Then execute the GMT script as follows:

```
␣␣␣␣perl␣plot_anomaly.gmt.pl
```

Two output files are created: `seattle.gxf.grd.pdf` and `seattle.gxf.grd.overlay.png`. The *pdf* file contains your map and the *png* file is suitable for overlay in Google Earth.