

**NAME**

grdraster – extract subregion from a binary raster and write a grd file

**SYNOPSIS**

```
grdraster [ filename | "text pattern" ] -Rwest/east/south/north[r] [ -Ggrdfilename ] [
-Ixinc[m]lc[/yinc[m]lc] ] [ -Jparameters ] [ -V ] [ -bo[s]SID]ncol ]
```

**DESCRIPTION**

**grdraster** reads a file called *grdraster.info* from the directory pointed to by the environment parameter **\$GMT\_GRIDDIR** (if this parameter is not set it defaults to **\$GMTHOME**/share/dbase). The info file defines binary arrays of data stored in scan-line format in data files. Each file is given a *filename* in the info file. **grdraster** figures out how to load the raster data into a grd file spanning a region defined by **-R**. By default the grid spacing equals the raster spacing. The **-I** option may be used to sub-sample the raster data. No filtering or interpolating is done, however; the *x\_inc* and *y\_inc* of the grd file must be multiples of the increments of the raster file and **grdraster** simply takes every *n*'th point. The output of **grdraster** is either grid or pixel registered depending on the registration of the raster used. It is up to the **GMT** system person to maintain the *grdraster.info* file in accordance with the available rasters at each site. Raster data sets are not supplied with **GMT** but can be obtained by anonymous ftp and on CD-ROM (see README page in dbase directory). **grdraster** will list the available files if no arguments are given. Finally, **grdraster** will write xyz-triplets to stdout if no output gridfile name is given

*filename*

If an integer matching one of the files listed in the *grdraster.info* file is given we will use that data set, else we will match the given text pattern with the data set description in order to determine the data set.

**-R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*-*[*dd:mm:ss.xxx*][**W**|**E**|**S**|**N**] format. Append **r** if lower left and upper right map coordinates are given instead of wesn. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude). If **r** is appended, you may also specify a map projection to define the shape of your region. The output region will be rounded off to the nearest whole grid-step in both dimensions.

**OPTIONS**

- G** Name of output grid file. If not set, the grid will be written as ASCII (or binary; see **-bo** xyz-triplets to stdout instead.
- I** *x\_inc* [and optionally *y\_inc*] is the grid spacing. Append **m** to indicate minutes or **c** to indicate seconds. If one of the units **e**, **k**, **i**, or **n** is appended instead, the increment will be assumed to be in meter, km, miles, or nautical miles, respectively, and will be converted to the equivalent degrees longitude at the middle latitude of the region (the conversion depends on **ELLIPSOID**). If *y\_inc* is given but set to 0 it will be reset equal to *x\_inc*; otherwise it will be converted to degrees latitude. If **=** is appended then the corresponding max *x* (*east*) or *y* (*north*) may be slightly adjusted to fit exactly the given increment [by default the increment may be adjusted slightly to fit the given domain]. Finally, instead of giving an increment you may specify the *number of nodes* desired by appending **+** to the supplied increment; the increment is then recalculated from the number of nodes and the domain. The resulting increment value depends on whether you have selected a gridline-registered or pixel- registered grid; see Appendix B for details.
- J** Selects the map projection. Scale is UNIT/degree, 1:xxxxx, or width in UNIT (upper case modifier). UNIT is cm, inch, or m, depending on the **MEASURE\_UNIT** setting in .gmtdefaults4, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width value. For map height, max dimension, or min dimension, append **h**, **+**, or **-** to the width, respectively. More details can be found in the **psbasemap** man pages.

**CYLINDRICAL PROJECTIONS:**

**-J***lon0/lat0/scale* (Cassini)

- Jj***lon0/scale* (Miller)
- Jm***scale* (Mercator - Greenwich and Equator as origin)
- Jmlon0/lat0/scale** (Mercator - Give meridian and standard parallel)
- Joal***lon0/lat0/azimuth/scale* (Oblique Mercator - point and azimuth)
- Job***lon0/lat0/lon1/lat1/scale* (Oblique Mercator - two points)
- Joc***lon0/lat0/lonp/latp/scale* (Oblique Mercator - point and pole)
- Jql***lon0/scale* (Equidistant Cylindrical Projection (Plate Carree))
- Jt***lon0/scale* (TM - Transverse Mercator, with Equator as  $y = 0$ )
- Jtlon0/lat0/scale** (TM - Transverse Mercator, set origin)
- Ju***zone/scale* (UTM - Universal Transverse Mercator)
- Jy***lon0/lats/scale* (Basic Cylindrical Projection)

#### AZIMUTHAL PROJECTIONS:

- Ja***lon0/lat0/scale* (Lambert)
- Je***lon0/lat0/scale* (Equidistant)
- Jf***lon0/lat0/horizon/scale* (Gnomonic)
- Jg***lon0/lat0/scale* (Orthographic)
- Jslon0/lat0/[slat]/scale** (General Stereographic)

#### CONIC PROJECTIONS:

- Jb***lon0/lat0/lat1/lat2/scale* (Albers)
- Jd***lon0/lat0/lat1/lat2/scale* (Equidistant)
- Jl***lon0/lat0/lat1/lat2/scale* (Lambert)

#### MISCELLANEOUS PROJECTIONS:

- Jh***lon0/scale* (Hammer)
- Ji***lon0/scale* (Sinusoidal)
- Jk[fls]***lon0/scale* (Eckert IV (f) and VI (s))
- Jn***lon0/scale* (Robinson)
- Jr***lon0/scale* (Winkel Tripel)
- Jv***lon0/scale* (Van der Grinten)
- Jw***lon0/scale* (Mollweide)

#### NON-GEOGRAPHICAL PROJECTIONS:

- Jp[a]***scale[/origin][rlz]* (Polar coordinates (theta,r))
- Jxx-scale[dllppow|t|T][y-scale[dllppow|t|T]]** (Linear, log, and power scaling)
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- bo** Selects binary output. Append **s** for single precision [Default is **d** (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of desired columns in your binary output file. This option applies only if no **-G** option has been set.

#### EXAMPLES

To extract data from raster 1, taking one point every 30 minutes, in an area extended beyond 360 degrees to allow later filtering, run

```
grdraster 1 -R-4/364/-62/62 -I30m -Gdata.grd
```

To obtain data for an oblique Mercator projection we need to extract more data that is actually used. This is necessary because the output of **grdraster** has edges defined by parallels and meridians, while the oblique map in general does not. Hence, to get all the data from the ETOPO2 data needed to make a contour map

for the region defined by its lower left and upper right corners and the desired projection, use

```
grdramer ETOPO2 -R160/20/220/30r -Joc190/25.5/292/69/1 -Gdata.grd
```

To extract data from the 2 min Geoware relief blend and write it as binary double precision xyz-triplets to standard output:

```
grdramer "2 min Geoware" -R20/25/-10/5 -bo >! triplets.b
```

## SEE ALSO

*gmtdefaults(1)*, *GMT(1)*, *grdsample(1)*, *grdfilter(1)*

## REFERENCES

- Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.
- Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.
- Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

**NAME**

`img2grd` – Extract region of `img` in Mercator or geographic form

**SYNOPSIS**

```
img2grd imgfile -Ggrdfile -Rwest/east/south/north[r] -Ttype [ -Dminlat/maxlat ] [ -E ] [ -L ] [ -M ] [ -Nnavg ] [ -Sscale ] [ -V ] [ -Wmaxlon ] [ -mminutes ]
```

**DESCRIPTION**

**img2grd** is a front-end to **img2mercgrd** which reads an `img` format file and creates a `grdfile`. The **-M** option dictates whether or not the Spherical Mercator projection of the `img` file is preserved.

*imgfile* An `img` format file such as the marine gravity or seafloor topography fields estimated from satellite altimeter data by Sandwell and Smith. If the user has set an environment variable **\$GMT\_IMGDIR**, then **img2mercgrd** will try to find *imgfile* in **\$GMT\_IMGDIR**; else it will try to open *imgfile* directly.

**-G** *grdfile* is the name of the output `grdfile`.

**-R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in `[+-]dd:mm[:ss.xxx][W|E|S|N]` format. Append **r** if lower left and upper right map coordinates are given instead of `wesn`. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).

**-T** *type* handles the encoding of constraint information. *type* = 0 indicates that no such information is encoded in the `img` file (used for pre-1995 versions of the gravity data) and gets all data. *type* > 0 indicates that constraint information is encoded (1995 and later (current) versions of the `img` files) so that one may produce a `grd` file as follows: **-T1** gets data values at all points, **-T2** gets data values at constrained points and NaN at interpolated points; **-T3** gets 1 at constrained points and 0 at interpolated points.

**OPTIONS**

**-E** Can be used when **-M** is not set to force the final grid to have the exact same region as requested with **-R**. By default, the final region is a direct projection of the original Mercator region and will typically extend slightly beyond the requested latitude range, and furthermore the grid increment in latitude does not match the longitude increment. However, the extra resampling introduces small interpolation errors and should only be used if the output grid must match the requested region and have `x_inc = y_inc`.

**-L** With no other arguments, list all `*.img` files found in the directory pointed to by **\$GMT\_IMGDIR**, or the current directory if not defined. Ignored if other options are present on the command line.

**-M** Output a Spherical Mercator grid [Default is a geographic lon/lat grid].

**-N** Average the values in the input `img` pixels into *navg* by *navg* squares, and create one output pixel for each such square. If used with **-T3** it will report an average constraint between 0 and 1. If used with **-T2** the output will be average data value or NaN according to whether average constraint is > 0.5. *navg* must evenly divide into the dimensions of the `imgfile` in pixels. [Default 1 does no averaging].

**-S** Multiply the `img` file values by *scale* before storing in `grd` file. [Default is 1.0]. (`img` topo files are stored in (corrected) meters; gravity files in `mGal*10`; vertical deflection files in `microradians*10`, vertical gravity gradient files in `Eotvos*10`.)

**-V** Selects verbose mode, which will send progress reports to `stderr` [Default runs "silently"]. Particularly recommended here, as it is helpful to see how the coordinates are adjusted.

**-m** Indicate *minutes* as the width of an input `img` pixel in minutes of longitude. [Default is 2.0].

**-W** Indicate *maxlon* as the maximum longitude extent of the input `img` file. Versions since 1995 have had `maxlon = 360.0`, while some earlier files had `maxlon = 390.0`. [Default is 360.0].

- D** Use the extended latitude range -80.738/+80.738. Alternatively, append *minlat/maxlat* as the latitude extent of the input img file. [Default is -72.006/72.006].

### EXAMPLES

To extract data in the region **-R-40/40/-70/-30** from *world\_grav.img.7.2* and preserve the Mercator gridding:

```
img2grd world_grav.img.7.2 -Gmerc_grav.grd -R-40/40/-70/-30 -M -T1 -V
```

Without the **-M** option the same command will yield a geographic grid.

### SEE ALSO

*GMT(1)*, *img2mercgrd(1)*

**NAME**

`img2mercgrd` – Extract region of `img`, preserving Mercator, save as `grd`

**SYNOPSIS**

```
img2mercgrd imgfile -Ggrdfile -Rwest/east/south/north[r] -Ttype [ -Dminlat/maxlat ] [ -Nnavg ] [ -Sscale ] [ -V ] [ -Wmaxlon ] [ -mminutes ]
```

**DESCRIPTION**

**img2mercgrd** reads an `img` format file and creates a `grdfile`. The Spherical Mercator projection of the `img` file is preserved, so that the region **-R** set by the user is modified slightly; the modified region corresponds to the edges of pixels [or groups of *navg* pixels]. The `grdfile` header is set so that the x and y axis lengths represent distance from the west and south edges of the image, measured in user default units, with **-Jm1** and the adjusted **-R**. By setting the default **ELLIPSOID** = Sphere, the user can make overlays with the adjusted **-R** so that they match. See **EXAMPLES** below. The adjusted **-R** is also written in the `grdheader` remark, so it can be found later. The **-T***type* selects all data or only data at constrained pixels, and can be used to create a grid of 1s and 0s indicating constraint locations. The output `grd` file is pixel registered; it inherits this from the `img` file.

*imgfile* An `img` format file such as the marine gravity or seafloor topography fields estimated from satellite altimeter data by Sandwell and Smith. If the user has set an environment variable **\$GMT\_IMGDIR**, then **img2mercgrd** will try to find *imgfile* in **\$GMT\_IMGDIR**; else it will try to open *imgfile* directly.

**-G** *grdfile* is the name of the output `grdfile`.

**-R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd*:*mm*[:*ss.xxx*][**W**|**E**|**S**|**N**] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).

**-T** *type* handles the encoding of constraint information. *type* = 0 indicates that no such information is encoded in the `img` file (used for pre-1995 versions of the gravity data; all more recent files do not support this choice) and gets all data. *type* > 0 indicates that constraint information is encoded (1995 and later (current) versions of the `img` files) so that one may produce a `grd` file as follows: **-T1** gets data values at all points, **-T2** gets data values at constrained points and NaN at interpolated points; **-T3** gets 1 at constrained points and 0 at interpolated points.

**OPTIONS**

**-N** Average the values in the input `img` pixels into *navg* by *navg* squares, and create one output pixel for each such square. If used with **-T3** it will report an average constraint between 0 and 1. If used with **-T2** the output will be average data value or NaN according to whether average constraint is > 0.5. *navg* must evenly divide into the dimensions of the `imgfile` in pixels. [Default 1 does no averaging].

**-S** Multiply the `img` file values by *scale* before storing in `grd` file. [Default is 1.0]. (`img` topo files are stored in (corrected) meters; gravity files in mGal\*10; vertical deflection files in microradians\*10, vertical gravity gradient files in Eotvos\*10. Use **-S0.1** for those files.)

**-V** Selects verbose mode, which will send progress reports to `stderr` [Default runs "silently"]. Particularly recommended here, as it is helpful to see how the coordinates are adjusted.

**-m** Indicate *minutes* as the width of an input `img` pixel in minutes of longitude. [Default is 2.0].

**-W** Indicate *maxlon* as the maximum longitude extent of the input `img` file. Versions since 1995 have had *maxlon* = 360.0, while some earlier files had *maxlon* = 390.0. [Default is 360.0].

**-D** Use the extended latitude range -80.738/+80.738. Alternatively, append *minlat/maxlat* as the latitude extent of the input `img` file. [Default is -72.006/72.006].

**EXAMPLES**

To extract data in the region **-R-40/40/-70/-30** from `world_grav.img.7.2`, run

```
img2mercgrd world_grav.img.7.2 -Gmerc_grav.grd -R-40/40/-70/-30 -T1 -V
```

Note that the **-V** option tells us that the range was adjusted to **-R-40/40/-70.0004681551/-29.9945810754**. We can also use **grdinfo** to find that the grd file header shows its region to be **-R0/80/0/67.9666667**. This is the range of x,y we will get from a Spherical Mercator projection using **-R-40/40/-70.0004681551/-29.9945810754** and **-Jm1**. Thus, to take ship.lonlatgrav and use it to sample the merc\_grav.grd, we can do this:

```
gmtset ELLIPSOID Sphere  
mapproject -R-40/40/-70.0004681551/-29.9945810754 -Jm1 ship.lonlatgrav | grdtrack  
-Gmerc_grav.grd | mapproject -R-40/40/-70.0004681551/-29.9945810754 -Jm1 -I > ship.lonlatgravsat
```

It is recommended to use the above method of projecting and unprojecting the data in such an application, because then there is only one interpolation step (in **grdtrack**). If one first tries to convert the grd file to lon,lat and then sample it, there are two interpolation steps (in conversion and in sampling).

To make a lon,lat grid from the above grid we can use

```
grdproject merc_grav.grd -R-40/40/-70.0004681551/-29.9945810754 -Jm1 -I -F -D2m -Ggrav.grd
```

In some cases this will not be easy as the **-R** in the two coordinate systems may not align well. When this happens, we can also use (in fact, it may be always better to use)

```
grd2xyz merc_grav.grd | mapproject -R-40/40/-70.0004681551/-29.994581075 -Jm1 -I | surface  
-R-40/40/-70/70 -I2m -Ggrav.grd
```

To make a Mercator map of the above region, suppose our `gmtdefaults4 MEASURE_UNIT` is inch. Then since the above merc\_grav.grd file is projected with **-Jm1** it is 80 inches wide. We can make a map 8 inches wide by using **-Jx0.1** on any map programs applied to this grid (e.g., **grdcontour**, **grdimage**, **grdview**), and then for overlays which work in lon,lat (e.g., **psxy**, **pscoast**) we can use the above adjusted **-R** and **-Jm0.1** to get the two systems to match up.

However, we can be smarter than this. Realizing that the input img file had pixels 2.0 minutes wide (or checking the nx and ny with **grdinfo merc\_grav.grd**) we realize that merc\_grav.grd used the full resolution of the img file and it has 2400 by 2039 pixels, and at 8 inches wide this is 300 pixels per inch. We decide we don't need that many and we will be satisfied with 100 pixels per inch, so we want to average the data into 3 by 3 squares. (If we want a contour plot we will probably choose to average the data much more (e.g. 6 by 6) to get smooth contours.) Since 2039 isn't divisible by 3 we will get a different adjusted **-R** this time:

```
img2mercgrd world_grav.img.7.2 -Gmerc_grav_2.grd -R-40/40/-70/-30 -T1 -N3 -V
```

This time we find the adjusted region is **-R-40/40/-70.023256525/-29.9368261101** and the output is 800 by 601 pixels, a better size for us. Now we can create an artificial illumination file for this using **grdgradient**:

```
grdgradient merc_grav_2.grd -Gillum.grd -A0/270 -Ne0.6
```

and if we also have a cpt file called "grav.cpt" we can create a color shaded relief map like this:

```
grdimage merc_grav_2.grd -Iillum.grd -Cgrav.cpt -Jx0.1 -K > map.ps  
psbasemap -R-40/40/-70.023256525/-29.9368261101 -Jm0.1 -Ba10 -O >> map.ps
```

Suppose you want to obtain only the constrained data values from an img file, in lat/lon coordinates. Then run **img2mercgrd** with the **-T2** option, use **grd2xyz** to dump the values, pipe through **grep -v NaN** to

eliminate NaNs, and pipe through **mapproject** with the inverse projection as above.

**SEE ALSO**

*GMT(1)*, *grdproject(1)*, *mapproject(1)*

**NAME**

`pscoupe` – Plot cross-sections of focal mechanisms.

**SYNOPSIS**

```
pscoupe files -Jparameters -Rwest/east/south/north[r] -Aparameters [ -B[p|s]parameters ] [ -Ffill ] [ -Gfill ] [ -H[i]nrec ] [ -K ] [ -Lpen ] [ -M ] [ -N ] [ -O ] [ -P ] [ -S<symbol><scale>[/d] ] [ -s<symbol><size>[/d] ] [ -Tn ] [ -U[dx/dy][label] ] [ -V ] [ -Wpen ] [ -X[alc|r][x-shift[u] ] [ -X[alc|r][y-shift[u] ] [ -Zcpt ] [ -a[size/[P_symbol/[T_symbol]]] ] [ -gfill ] [ -efill ] [ -ppen ] [ -tpen ] [ -:[ilo] ] [ -ccopies ]
```

**DESCRIPTION**

`pscoupe` reads data values from *files* [or standard input] and generates *PostScript* code that will plot symbols, lines or polygons on a cross-section. Focal mechanisms may be specified and require additional columns of data. The *PostScript* code is written to standard output.

*files* list one or more file-names. If no files are given, `pscoupe` will read standard input.

A *new file* is created with the new coordinates (**x**, **y**) and the mechanism (from lower focal half-sphere for horizontal plane, to half-sphere behind a vertical plane). When the plane is not horizontal, - north direction becomes upwards steepest descent direction of the plane (**u**) - east direction becomes strike direction of the plane (**s**) - down direction (= north<sup>^</sup>east) becomes **u**<sup>^</sup>**s**

Axis angles are defined in the same way as in horizontal plane in the new system.

Moment tensor (initially in **r**, **t**, **f** system that is up, south, east) is defined in (-**u**<sup>^</sup>**s**, -**u**, **s**) system. A *file* is created with extracted events.

- J** Selects the map projection. Scale is UNIT/degree, 1:xxxxx, or width in UNIT (upper case modifier). UNIT is cm, inch, or m, depending on the **MEASURE\_UNIT** setting in `.gmtdefaults4`, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width value. For map height, max dimension, or min dimension, append **h**, **+**, or **-** to the width, respectively. More details can be found in the `psbasemap` man pages.

**CYLINDRICAL PROJECTIONS:**

- Jclon0/lat0/scale** (Cassini)
- Jjlon0/scale** (Miller)
- Jmscale** (Mercator - Greenwich and Equator as origin)
- Jmlon0/lat0/scale** (Mercator - Give meridian and standard parallel)
- Joalon0/lat0/azimuth/scale** (Oblique Mercator - point and azimuth)
- Joblon0/lat0/lon1/lat1/scale** (Oblique Mercator - two points)
- Joclon0/lat0/lonp/latp/scale** (Oblique Mercator - point and pole)
- Jqlon0/scale** (Equidistant Cylindrical Projection (Plate Carree))
- Jtlon0/scale** (TM - Transverse Mercator, with Equator as **y** = 0)
- Jtlon0/lat0/scale** (TM - Transverse Mercator, set origin)
- Juzone/scale** (UTM - Universal Transverse Mercator)
- Jylon0/lats/scale** (Basic Cylindrical Projection)

**AZIMUTHAL PROJECTIONS:**

- Jalon0/lat0/scale** (Lambert)
- Jelon0/lat0/scale** (Equidistant)
- Jflon0/lat0/horizon/scale** (Gnomonic)
- Jglon0/lat0/scale** (Orthographic)
- Jslon0/lat0/[slat]/scale** (General Stereographic)

**CONIC PROJECTIONS:**

- Jb***lon0/lat0/lat1/lat2/scale* (Albers)
- Jd***lon0/lat0/lat1/lat2/scale* (Equidistant)
- Jl***lon0/lat0/lat1/lat2/scale* (Lambert)

#### MISCELLANEOUS PROJECTIONS:

- Jh***lon0/scale* (Hammer)
- Ji***lon0/scale* (Sinusoidal)
- Jk**[**f**]*lon0/scale* (Eckert IV (f) and VI (s))
- Jn***lon0/scale* (Robinson)
- Jr***lon0/scale* (Winkel Tripel)
- Jv***lon0/scale* (Van der Grinten)
- Jw***lon0/scale* (Mollweide)

#### NON-GEOGRAPHICAL PROJECTIONS:

- Jp**[**a**]*scale[/origin][r]z* (Polar coordinates (theta,r))
- Jxx**-*scale*[**dlllppow**]**t**[**T**]/*y-scale*[**dlllppow**]**t**[**T**] (Linear, log, and power scaling)
- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd:mm[:ss.xxx]*[**W**|**E**|**S**|**N**] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude). If frame is defined from cross-section parameters (see **-A**) this option is not taken into account, but must be present.
- A** selects the cross-section.
  - Aa***lon1/lat1/lon2/lat2/dip/p\_width/dmin/dmax*[**f**]  
*lon* and *lat* are the longitude and latitude of points 1 and 2 limiting the length of the cross-section.  
*dip* is the dip of the plane on which the cross-section is made.  
*p\_width* is the width of the cross-section on each side of a vertical plane or above and under an oblique plane.  
*dmin* and *dmax* are the distances min and max from horizontal plane, along steepest descent direction.
  - Add **f** to get the frame from the cross-section parameters.
  - Ab***lon1/lat1/strike/p\_length/dip/p\_width/dmin/dmax*[**f**]  
*lon1* and *lat1* are the longitude and latitude of the beginning of the cross-section.  
*strike* is the azimuth of the direction of the cross-section.  
*p\_length* is the length along which the cross-section is made.  
 The other parameters are the same as for **-Aa** option.
  - Ac***x1/y1/x2/y2/dip/p\_width/dmin/dmax*[**f**]  
 The same as **-Aa** option with *x* and *y* cartesian coordinates.
  - Ad***x1/y1/strike/p\_length/dip/p\_width/dmin/dmax*[**f**]  
 The same as **-Ab** option with *x* and *y* cartesian coordinates.
- S** selects the meaning of the columns in the data file and the figure to be plotted.
- Sa***scale[/fontsize[/offset*[**u**]]]  
 Focal mechanisms in Aki and Richard convention. *scale* adjusts the scaling of the radius of the "beach ball", which will be proportional to the magnitude. The *scale* is the size for magnitude = 5 in **MEASURE\_UNIT** (unless **c**, **i**, **m**, or **p** is appended to indicate that the size information is in units of cm, inches, meters, or points, respectively). Use the **-T** option to render the beach ball transparent by drawing only the nodal planes and the circumference. The color or shade of the compressive quadrants can be specified with the **-G** option. The color or shade of the extensive quadrants can be specified with the **-E** option. Parameters are expected to be in the following columns:

- 1,2: longitude, latitude of event (-: option interchanges order)
- 3: depth of event in kilometers
- 4,5,6: strike, dip and rake
- 7: magnitude
- 8,9: not used; can be 0 0; allows use of the psmeca file format
- 10: text string to appear above the beach ball (default) or under (add **u**).

**-Scscale**

Focal mechanisms in Harvard CMT convention. *scale* adjusts the scaling of the radius of the "beach ball", which will be proportional to the magnitude. The *scale* is the size for magnitude = 5 (that is  $M_0 = 4E+23$  dynes-cm.) in **MEASURE\_UNIT** (unless **c**, **i**, **m**, or **p** is appended to indicate that the size information is in units of cm, inches, meters, or points, respectively). Use the **-T** option to render the beach ball transparent by drawing only the nodal planes and the circumference. The color or shade of the compressive quadrants can be specified with the **-G** option. The color or shade of the extensive quadrants can be specified with the **-E** option. Parameters are expected to be in the following columns:

- 1,2: longitude, latitude of event (-: option interchanges order)
- 3: depth of event in kilometers
- 4,5,6: strike, dip, and slip of plane 1
- 7,8,9: strike, dip, and slip of plane 2
- 10,11: mantissa and exponent of moment in dyne-cm (if magnitude is used instead of scalar moment, magnitude is in column 10 and 0 must be in column 11)
- 12,13: not used; can be 0 0; allows use of the psmeca file format
- 14: text string to appear above the beach ball (default) or under (add **u**).

**-Spscale[/fontsize[/offset[**u**]]]**

Focal mechanisms given with partial data on both planes. *scale* adjusts the scaling of the radius of the "beach ball", which will be proportional to the magnitude. The *scale* is the size for magnitude = 5 in **MEASURE\_UNIT** (unless **c**, **i**, **m**, or **p** is appended to indicate that the size information is in units of cm, inches, meters, or points, respectively). The color or shade of the compressive quadrants can be specified with the **-G** option. The color or shade of the extensive quadrants can be specified with the **-E** option. Parameters are expected to be in the following columns:

- 1,2: longitude, latitude of event (-: option interchanges order)
- 3: depth
- 4,5: strike, dip of plane 1
- 6: strike of plane 2
- 7: must be -1/+1 for a normal/inverse fault
- 8: magnitude
- 9,10: not used; can be 0 0; allows use of the psmeca file format
- 11: text string to appear above the beach ball (default) or under (add **u**).

**-Smscale[/fontsize[/offset[**u**]]]**

Seismic moment tensor. *scale* adjusts the scaling of the radius of the "beach ball", which will be proportional to the magnitude. The *scale* is the size for magnitude = 5 (that is seismic scalar moment =  $4E+23$  dynes-cm) in **MEASURE\_UNIT** (unless **c**, **i**, **m**, or **p** is appended to indicate

that the size information is in units of cm, inches, meters, or points, respectively). (**-T0** option overlays best double couple transparently.) Put **-Sd***scale*[/*fontsize*/*offset*[**u**]] to plot the only double couple part of moment tensor. Put **-Sz***scale*[/*fontsize*/*offset*[**u**]] to plot anisotropic part of moment tensor (zero trace). The color or shade of the compressive quadrants can be specified with the **-G** option. The color or shade of the extensive quadrants can be specified with the **-E** option. Parameters are expected to be in the following columns:

- 1,2:** longitude, latitude of event (**-:** option interchanges order)
- 3:** depth of event in kilometers
- 4,5,6,7,8,9:**  
mrr, mtt, mff, mrt, mrf, mtf in 10\*exponent dynes-cm
- 10:** exponent
- 11,12:** Not used; can be 0 0; allows use of the psmeca file format
- 13:** Text string to appear above the beach ball (default) or under (add **u**).

**-Sx***scale*[/*fontsize*/*offset*[**u**]]

Principal axis. *scale* adjusts the scaling of the radius of the "beach ball", which will be proportional to the magnitude. The *scale* is the size for magnitude = 5 (that is seismic scalar moment = 4\*10e+23 dynes-cm) in **MEASURE\_UNIT** (unless **c**, **i**, **m**, or **p** is appended to indicate that the size information is in units of cm, inches, meters, or points, respectively). (**-T0** option overlays best double couple transparently.) Put **-Sys***scale*[/*fontsize*/*offset*[**u**]] to plot the only double couple part of moment tensor. Put **-St***scale*[/*fontsize*/*offset*[**u**]] to plot anisotropic part of moment tensor (zero trace). The color or shade of the compressive quadrants can be specified with the **-G** option. The color or shade of the extensive quadrants can be specified with the **-E** option. Parameters are expected to be in the following columns:

- 1,2:** longitude, latitude of event (**-:** option interchanges order)
- 3:** depth of event in kilometers
- 4,5,6,7,8,9,10,11,12:**  
value (in 10\*exponent dynes-cm), azimuth, plunge of the T, N, and P axes.
- 13:** exponent
- 14,15:** longitude, latitude at which to place beach ball. Entries in these columns are necessary with the **-C** option. Using 0,0 in columns 9 and 10 will plot the beach ball at the longitude, latitude given in columns 1 and 2. The **-:** option will interchange the order of columns (1,2) and (9,10).
- 16:** Text string to appear above the beach ball (optional).

**-ssymbol**[*size*/*fontsize*/*offset*[**u**]]

selects a symbol instead of mechanism. Choose from the following: (**c**) circle, (**d**) diamond, (**i**) itriangle, (**s**) square, (**t**) triangle, (**x**) cross. *size* is the symbol size in **MEASURE\_UNIT** (unless **c**, **i**, **m**, or **p** is appended to indicate that the size information is in units of cm, inches, meters, or points, respectively). If *size* must be read, it must be in column 4 and the text string will start in column 5. Parameters are expected to be in the following columns:

- 1,2:** longitude, latitude of event (**-:** option interchanges order)
- 3:** depth of event in kilometers
- 4:** Text string to appear above the beach ball (default) or under (add **u**).

## OPTIONS

No space between the option flag and the associated arguments.

- B** Sets map boundary annotation and tickmark intervals; see the **psbasemap** man page for all the details.

- E***fill* Selects filling of extensive quadrants. Usually white. Set the shade (0–255) or color (r/g/b) [Default is 255/255/255].
- G***fill* Selects filling of focal mechanisms. By convention, the compressional quadrants are shaded. Set the shade (0–255) or color (r/g/b) [Default is 0/0/0]. Optionally, specify **-G***icon\_size/pattern*, where *pattern* gives the number of the image pattern (1-90) OR the name of a Sun rasterfile. *icon\_size* sets the unit size in inches. To invert black and white pixels, use **-GP** instead of **-Gp**. See Appendix E for information on individual patterns.
- H** Input file(s) has Header record(s). Number of header records can be changed by editing your `.gmtdefaults4` file. If used, **GMT** default is 1 header record. Use **-Hi** if only input data should have header records [Default will write out header records if the input data have them].
- K** More *PostScript* code will be appended later [Default terminates the plot system].
- L**[*pen*]  
Draws the "beach ball" outline using current pen (see **-W**) or sets pen attributes.
- M** Same size for any magnitude.
- N** Does **NOT** skip symbols that fall outside map border [Default plots points inside border only].
- O** Selects Overlay plot mode [Default initializes a new plot system].
- P** Selects Portrait plotting mode [**GMT** Default is Landscape, see **gmtdefaults** to change this].
- T**[*num\_of\_planes*]  
Plots the nodal planes and outlines the bubble which is transparent. If *num\_of\_planes* is  
  - 0: both nodal planes are plotted;
  - 1: only the first nodal plane is plotted;
  - 2: only the second nodal plane is plotted [Default: 0].
- U** Draw Unix System time stamp on plot. User may specify where the lower left corner of the stamp should fall on the page relative to lower left corner of plot. Optionally, append a label, or **c** (which will plot the command string.). The **GMT** parameters **UNIX\_TIME** and **UNIX\_TIME\_POS** can affect the appearance; see the **gmtdefaults** man page for details.
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- W** set pen attributes for text string or default pen attributes for fault plane edges. [Defaults: width = 1, color = 0/0/0, texture = *solid*].
- X -Y** Shift origin of plot by (*x-shift,y-shift*), and optionally append units (**c**, **i**, **m**, **p**). Prepend **a** for absolute coordinates; the default (**r**) will reset plot origin. Give **c** to center plot using current page size.
- Z***cptfile*  
Give a color palette file and let compressive part color be determined by the z-value in the third column.
- a**[*size*/[*P\_axis\_symbol*]/[*T\_axis\_symbol*]]  
Computes and plots P and T axes with symbols. Optionally specify *size* and (separate) P and T axis symbols from the following: (**c**) circle, (**d**) diamond, (**h**) hexagon, (**i**) inverse triangle, (**p**)point, (**s**) square, (**t**) triangle, (**x**) cross. [Defaults: 0.2**c/c/c** or 0.08**i/c/c**.]
- e***fill* Selects filling of T axis symbol. Set the shade (0–255) or color (r/g/b) [Default is color of extensive parts.]
- g***fill* Selects filling of P axis symbol. Set the shade (0–255) or color (r/g/b) [Default is color of compressive parts.]
- p**[*pen*]  
Draws the P axis outline using current pen (see **-W**), or sets pen attributes.
- t**[*pen*] Draws the T axis outline using current pen (see **-W**), or sets pen attributes.

- : Toggles between (longitude,latitude) and (latitude,longitude) input and/or output. [Default is (longitude,latitude)]. Append **i** to select input only or **o** to select output only. [Default affects both].
- c Specifies the number of plot copies. [Default is 1].

**SEE ALSO**

*GMT(1)*, *psbasemap(1)*, *psmeca(1)*, *psxy(1)*

**REFERENCES**

- Bomford, G., Geodesy, 4th ed., Oxford University Press, 1980.  
Aki, K. and P. Richards, Quantitative Seismology, Freeman, 1980.  
F. A. Dahlen and Jeoren Trom, Theoretical Seismology, Princeton, 1998, p.167. Definition of scalar moment.  
Cliff Frohlich, Cliff's Nodes Concerning Plotting Nodal Lines for P, Sh and Sv  
Seismological Research Letters, Volume 67, Number 1, January-February, 1996  
Thorne Lay, Terry C. Wallace, Modern Global Seismology, Academic Press, 1995, p.384.  
W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes in C, Cambridge University press (routine jacobi)

**AUTHOR**

Genevieve Patau  
CNRS UMR 7580  
Seismology Dept.  
Institut de Physique du Globe de Paris  
(patau@ipgp.jussieu.fr)

**NAME**

psmeca – Plot focal mechanisms on maps

**SYNOPSIS**

```
psmeca files -Jparameters -Rwest/east/south/north[r] [ -B[pls]parameters ] [ -Cpen ] [ -Ddepmin/dep-
max ] [ -Efill ] [ -Gfill ] [ -H[i][nrec] ] [ -K ] [ -L[pen] ] [ -M ] [ -N ] [ -O ] [ -P ] [ -S<sym-
bol><scale>[/d]] [ -Tnum_of_plane[pen] ] [ -U[/dx/dy]/[label] ] [ -V ] [ -Wpen ] [ -X[alcr][x-shift[u]] ]
[ -X[alcr][y-shift[u]] ] [ -Zcptfile ] [ -z ] [ -a[size[P_symbol[T_symbol]]] ] [ -gfill ] [ -efill ] [ -o ] [
-ppen ] [ -tpen ] [ -:[ilo] ] [ -ccopies ]
```

**DESCRIPTION**

**psmeca** reads data values from *files* [or standard input] and generates *PostScript* code that will plot focal mechanisms on a map. Most options are the same as for **psxy**. The *PostScript* code is written to standard output.

**ARGUMENTS**

*files* List one or more file-names. If no files are given, **psmeca** will read standard input.

- J Selects the map projection. Scale is UNIT/degree, 1:xxxxx, or width in UNIT (upper case modifier). UNIT is cm, inch, or m, depending on the **MEASURE\_UNIT** setting in .gmtdefaults4, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width value. For map height, max dimension, or min dimension, append **h**, **+**, or **-** to the width, respectively. More details can be found in the **psbasemap** man pages.

**CYLINDRICAL PROJECTIONS:**

- Jclon0/lat0/scale (Cassini)
- Jjlon0/scale (Miller)
- Jmscale (Mercator - Greenwich and Equator as origin)
- Jmlon0/lat0/scale (Mercator - Give meridian and standard parallel)
- Joalon0/lat0/azimuth/scale (Oblique Mercator - point and azimuth)
- Joblon0/lat0/lon1/lat1/scale (Oblique Mercator - two points)
- Joclon0/lat0/lonp/latp/scale (Oblique Mercator - point and pole)
- Jqlon0/scale (Equidistant Cylindrical Projection (Plate Carree))
- Jtlon0/scale (TM - Transverse Mercator, with Equator as y = 0)
- Jtlon0/lat0/scale (TM - Transverse Mercator, set origin)
- Juzone/scale (UTM - Universal Transverse Mercator)
- Jylon0/lats/scale (Basic Cylindrical Projection)

**AZIMUTHAL PROJECTIONS:**

- Jalon0/lat0/scale (Lambert)
- Jelon0/lat0/scale (Equidistant)
- Jflon0/lat0/horizon/scale (Gnomonic)
- Jglon0/lat0/scale (Orthographic)
- Jslon0/lat0/[slat]/scale (General Stereographic)

**CONIC PROJECTIONS:**

- Jblon0/lat0/lat1/lat2/scale (Albers)
- Jdlon0/lat0/lat1/lat2/scale (Equidistant)
- Jllon0/lat0/lat1/lat2/scale (Lambert)

**MISCELLANEOUS PROJECTIONS:**

- Jhlon0/scale (Hammer)

- Ji***lon0/scale* (Sinusoidal)
- Jk***[fls]lon0/scale* (Eckert IV (f) and VI (s))
- Jn***lon0/scale* (Robinson)
- Jr***lon0/scale* (Winkel Tripel)
- Jv***lon0/scale* (Van der Grinten)
- Jw***lon0/scale* (Mollweide)

#### NON-GEOGRAPHICAL PROJECTIONS:

- Jp***[a]scale[/origin][r]z* (Polar coordinates (theta,r))
- Jxx***-scale[dlllppowt|T][y-scale[dlllppowt|T]]* (Linear, log, and power scaling)
- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd:mm[:ss.xxx][W|E|S|N]* format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg -Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).
- S** Selects the meaning of the columns in the data file  
In order to use the same file to plot cross-sections, depth is in third column. Nevertheless, it is possible to use "old style" **psvelomeca** input files without depth in third column using the **-o** option.
- Sa***scale[c/i][/fontsize[/offset[u]]]*  
Focal mechanisms in Aki and Richard convention. *scale* adjusts the scaling of the radius of the "beach ball", which will be proportional to the magnitude. Scale is the size for magnitude = 5 in inch (unless **c**, **i**, **m**, or **p** is appended). Use the **-T** option to render the beach ball transparent by drawing only the nodal planes and the circumference. The color or shade of the compressive quadrants can be specified with the **-G** option. The color or shade of the extensive quadrants can be specified with the **-E** option. Parameters are expected to be in the following columns:
  - 1,2:** longitude, latitude of event (**-:** option interchanges order)
  - 3:** depth of event in kilometers
  - 4,5,6:** strike, dip and rake in degrees
  - 7:** magnitude
  - 8,9:** longitude, latitude at which to place beach ball. Entries in these columns are necessary with the **-C** option. Using 0,0 in columns 8 and 9 will plot the beach ball at the longitude, latitude given in columns 1 and 2. The **-:** option will interchange the order of columns (1,2) and (8,9).
  - 10:** Text string to appear above the beach ball (optional).
- Sc***scale[c/i][/fontsize[/offset[u]]]*  
Focal mechanisms in Harvard CMT convention. *scale* adjusts the scaling of the radius of the "beach ball", which will be proportional to the magnitude. Scale is the size for magnitude = 5 (that is  $M_0 = 4.0E23$  dynes-cm) in inch (unless **c**, **i**, **m**, or **p** is appended). Use the **-T** option to render the beach ball transparent by drawing only the nodal planes and the circumference. The color or shade of the compressive quadrants can be specified with the **-G** option. The color or shade of the extensive quadrants can be specified with the **-E** option. Parameters are expected to be in the following columns:
  - 1,2:** longitude, latitude of event (**-:** option interchanges order)
  - 3:** depth of event in kilometers
  - 4,5,6:** strike, dip, and rake of plane 1
  - 7,8,9:** strike, dip, and rake of plane 2

- 10,11:** mantissa and exponent of moment in dyne-cm
- 12,13:** longitude, latitude at which to place beach ball. Entries in these columns are necessary with the **-C** option. Using (0,0) in columns 11 and 12 will plot the beach ball at the longitude, latitude given in columns 1 and 2. The **-:** option will interchange the order of columns (1,2) and (12,13).
- 14:** Text string to appear above the beach ball (optional).

**-Smscale**[*c/i*][*/fontsize*][*/offset*[**u**]]

Seismic moment tensor (Harvard CMT, with zero trace). *scale* adjusts the scaling of the radius of the "beach ball", which will be proportional to the magnitude. Scale is the size for magnitude = 5 (that is scalar seismic moment = 4.0E23 dynes-cm) in inch (unless **c**, **i**, **m**, or **p** is appended). (**-T0** option overlays best double couple transparently.) Put **-Sdscale**[*/fontsize*][*/offset*[**u**]] to plot the only double couple part of moment tensor. Put **-Szscales**[*/fontsize*][*/offset*[**u**]] to plot anisotropic part of moment tensor (zero trace). The color or shade of the compressive quadrants can be specified with the **-G** option. The color or shade of the extensive quadrants can be specified with the **-E** option. Parameters are expected to be in the following columns

- 1,2:** longitude, latitude of event (**-:** option interchanges order)
- 3:** depth of event in kilometers
- 4,5,6,7,8,9:**  
mrr, mtt, mff, mrt, mrf, mtf in 10\*exponent dynes-cm
- 10:** exponent
- 11,12:** longitude, latitude at which to place beach ball. Entries in these columns are necessary with the **-C** option. Using 0,0 in columns 9 and 10 will plot the beach ball at the longitude, latitude given in columns 1 and 2. The **-:** option will interchange the order of columns (1,2) and (9,10).
- 13:** Text string to appear above the beach ball (optional).

**-Spscale**[*c/i*][*/fontsize*][*/offset*[**u**]]

Focal mechanisms given with partial data on both planes. *scale* adjusts the scaling of the radius of the "beach ball", which will be proportional to the magnitude. Scale is the size for magnitude = 5 in inch (unless **c**, **i**, **m**, or **p** is appended). The color or shade of the compressive quadrants can be specified with the **-G** option. The color or shade of the extensive quadrants can be specified with the **-E** option. Parameters are expected to be in the following columns

- 1,2:** longitude, latitude of event (**-:** option interchanges order)
- 3:** depth of event in kilometers
- 4,5:** strike, dip of plane 1
- 6:** strike of plane 2
- 7:** must be -1/+1 for a normal/inverse fault
- 8:** magnitude
- 9,10:** longitude, latitude at which to place beach ball. Entries in these columns are necessary with the **-C** option. Using 0,0 in columns 9 and 10 will plot the beach ball at the longitude, latitude given in columns 1 and 2. The **-:** option will interchange the order of columns (1,2) and (9,10).
- 11:** Text string to appear above the beach ball (optional).

**-Sxscales**[*c/i*][*/fontsize*][*/offset*[**u**]]

Principal axis. *scale* adjusts the scaling of the radius of the "beach ball", which will be proportional to the magnitude. Scale is the size for magnitude = 5 (that is seismic scalar moment = 4\*10e+23 dynes-cm) in inch (unless **c**, **i**, **m**, or **p** is appended). (**-T0** option overlays best double couple transparently.) Put **-Syscales**[*c/i*][*/fontsize*][*/offset*[**u**]] to plot the only double couple part of moment tensor. Put **-Stscales**[*c/i*][*/fontsize*][*/offset*[**u**]] to plot zero trace moment tensor. The

color or shade of the compressive quadrants can be specified with the **-G** option. The color or shade of the extensive quadrants can be specified with the **-E** option. Parameters are expected to be in the following columns

- 1,2:** longitude, latitude of event (**-:** option interchanges order)
- 3:** depth of event in kilometers
- 4,5,6,7,8,9,10,11,12:**  
value (in 10\*exponent dynes-cm), azimuth, plunge of T, N, P axis.
- 13:** exponent
- 14,15:** longitude, latitude at which to place beach ball. Entries in these columns are necessary with the **-C** option. Using 0,0 in columns 9 and 10 will plot the beach ball at the longitude, latitude given in columns 1 and 2. The **-:** option will interchange the order of columns (1,2) and (9,10).
- 16:** Text string to appear above the beach ball (optional).

## OPTIONS

No space between the option flag and the associated arguments.

- B** Sets map boundary annotation and tickmark intervals; see the **psbasemap** man page for all the details.
- C[pen][pointsize]**  
Offsets focal mechanisms to the longitude, latitude specified in the last two columns of the input file before the (optional) text string. A small circle is plotted at the initial location and a line connects the beachball to the circle. Specify *pen* and/or *pointsize* to change the line style and/or size of the circle. [Defaults: *pen* width = 1, color = 0/0/0, texture = solid; *pointsize* 0].
- D** *depmin/depmax* Plots events between *depmin* and *depmax*.
- Efill** Selects filling of extensive quadrants. Usually white. Set the shade (0–255) or color (r/g/b) [Default is 255/255/255].
- Gfill** Selects filling of focal mechanisms. By convention, the compressional quadrants of the focal mechanism beach balls are shaded. Set the shade (0–255) or color (r/g/b) [Default is 0/0/0].
- H** Input file(s) has Header record(s). Number of header records can be changed by editing your `.gmtdefaults4` file. If used, **GMT** default is 1 header record. Use **-Hi** if only input data should have header records [Default will write out header records if the input data have them].
- K** More *PostScript* code will be appended later [Default terminates the plot system].
- L[pen]**  
Draws the "beach ball" outline with *pen* attributes. [Defaults width = 1, color = 0/0/0, texture = solid].
- N** Does **NOT** skip symbols that fall outside frame boundary specified by **-R** [Default plots symbols inside frame only].
- O** Selects Overlay plot mode [Default initializes a new plot system].
- P** Selects Portrait plotting mode [**GMT** Default is Landscape, see **gmtdefaults** to change this].
- T[num\_of\_planes]**  
Plots the nodal planes and outlines the bubble which is transparent. If *num\_of\_planes* is  
  - 0: both nodal planes are plotted;
  - 1: only the first nodal plane is plotted;
  - 2: only the second nodal plane is plotted.
- U** Draw Unix System time stamp on plot. User may specify where the lower left corner of the stamp should fall on the page relative to lower left corner of plot. Optionally, append a label, or **c** (which will plot the command string.). The **GMT** parameters **UNIX\_TIME** and **UNIX\_TIME\_POS** can

affect the appearance; see the **gmtdefaults** man page for details.

- V Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- W *pen* is a comma delimited list of *width*, *color* and *texture*, each of which is optional. *width* can be indicated as a measure (points, centimeters, inches) or as **faint**, **thin[ner|nest]**, **thick[er|lest]**, **fat[ter|test]**, or **obese**. *color* specifies a grey shade (0–255) or color (r/g/b, each in range 0–255; h-s-v, ranges 0–360, 0–1, 0–1; or c/m/y/k, each in range 0–100%; or valid color name). *texture* is a combination of dashes ‘-’ and dots ‘.’.
- X -Y Shift origin of plot by (*x-shift*,*y-shift*), and optionally append units (**c**, **i**, **m**, **p**). Prepend **a** for absolute coordinates; the default (**r**) will reset plot origin. Give **c** to center plot using current page size.
- Zcptfile Give a color palette file and let compressive part color be determined by the z-value in the third column.
- z Overlay zero trace moment tensor.
- a[*size*/[*P\_axis\_symbol*[/[*T\_axis\_symbol*]]] Computes and plots P and T axes with symbols. Optionally specify *size* and (separate) P and T axis symbols from the following: (**c**) circle, (**d**) diamond, (**h**) hexagon, (**i**) inverse triangle, (**p**) point, (**s**) square, (**t**) triangle, (**x**) cross. [Defaults: 0.2**c/c/c** or 0.08**i/c/c**.]
- efill Selects filling of T axis symbol. Set the shade (0–255) or color (r/g/b). Default is white.
- gfill Selects filling of P axis symbol. Set the shade (0–255) or color (r/g/b). Default black.
- o Use the **psvelomeca** input format without depth in the third column.
- p[*pen*] Draws the P axis outline using default pen (see -W), or sets pen attributes.
- t[*pen*] Draws the T axis outline using default pen (see -W), or sets pen attributes.
- : Toggles between (longitude,latitude) and (latitude,longitude) input and/or output. [Default is (longitude,latitude)]. Append **i** to select input only or **o** to select output only. [Default affects both].
- c Specifies the number of plot copies. [Default is 1].

## EXAMPLES

The following file should give a normal-faulting CMT mechanism:

```
psmecca -R239/240/34/35.2 -Jm4 -Sc0.4 -H1 <<END>>! test.ps
lon lat depth str dip slip st dip slip mant exp plon plat
239.384 34.556 12. 180 18 -88 0 72 -90 5.5 0 0 0
END
```

## SEE ALSO

*GMT*(1), *psbasemap*(1), *psxy*(1)

## REFERENCES

- Bomford, G., Geodesy, 4th ed., Oxford University Press, 1980.
- Aki, K. and P. Richards, Quantitative Seismology, Freeman, 1980.
- F. A. Dahlen and Jeoren Trom, Theoretical Seismology, Princeton, 1998, p.167.
- Cliff Frohlich, Cliff's Nodes Concerning Plotting Nodal Lines for P, Sh and Sv Seismological Research Letters, Volume 67, Number 1, January-February, 1996
- Thorne Lay, Terry C. Wallace, Modern Global Seismology, Academic Press, 1995, p.384.
- W.H. Press, S.A. Teukolsky, W.T. Vetterling, B.P. Flannery, Numerical Recipes in C, Cambridge University press (routine jacobi)

## AUTHORS

Genevieve Patau  
CNRS UMR 7580

Seismology Dept.  
Institut de Physique du Globe de Paris  
(patau@ipgp.jussieu.fr)

**NAME**

**pspolar** - Plot polarities on the inferior focal half-sphere on maps

**SYNOPSIS**

```
pspolar files -Jparameters -Rwest/east/south/north[r] -Dlon/lat -Msize -S<symbol><size> [
-B[pls]parameters ] [ -Clon/lat[/dash_width/pointsize] ] [ -Fcolor ] [ -Gfill ] [ -gfill ] [ -H[i][nrec] ] [ -h
] [ -K ] [ -L ] [ -N ] [ -O ] [ -P ] [ -sHalf-size[v[[v_width/h_length/h_width/shape]][g[color]][l] ] [ -Tan-
gle/form/justify/fontsize ] [ -tpen ] [ -U[/dx/dy/][label] ] [ -V ] [ -Wpen ] [ -X[alcr][x-shift[u] ] [
-X[alcr][y-shift[u] ] [ -copies ]
```

**DESCRIPTION**

**pspolar** reads data values from *files* [or standard input] and generates *PostScript* code that will plot stations on focal mechanisms on a map. The *PostScript* code is written to standard output.

Parameters are expected to be in the following columns

**1,2,3,4** station\_code, azimuth, take-off angle, polarity

polarity:

- compression can be c,C,u,U,+
- rarefaction can be d,D,r,R,-
- not defined is anything else

**ARGUMENTS**

*files* List one or more file-names. If no files are given, **pspolar** will read standard input.

- J** Selects the map projection. Scale is UNIT/degree, 1:xxxxx, or width in UNIT (upper case modifier). UNIT is cm, inch, or m, depending on the **MEASURE\_UNIT** setting in `.gmtdefaults4`, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width value. For map height, max dimension, or min dimension, append **h**, **+**, or **-** to the width, respectively. More details can be found in the **psbasemap** man pages.

**CYLINDRICAL PROJECTIONS:**

- Jclon0/lat0/scale** (Cassini)
- Jjlon0/scale** (Miller)
- Jmscale** (Mercator - Greenwich and Equator as origin)
- Jmlon0/lat0/scale** (Mercator - Give meridian and standard parallel)
- Joalon0/lat0/azimuth/scale** (Oblique Mercator - point and azimuth)
- Joblon0/lat0/lon1/lat1/scale** (Oblique Mercator - two points)
- Joclon0/lat0/lonp/latp/scale** (Oblique Mercator - point and pole)
- Jqlon0/scale** (Equidistant Cylindrical Projection (Plate Carree))
- Jtlon0/scale** (TM - Transverse Mercator, with Equator as  $y = 0$ )
- Jtlon0/lat0/scale** (TM - Transverse Mercator, set origin)
- Juzone/scale** (UTM - Universal Transverse Mercator)
- Jylon0/lats/scale** (Basic Cylindrical Projection)

**AZIMUTHAL PROJECTIONS:**

- Jalon0/lat0/scale** (Lambert)
- Jelon0/lat0/scale** (Equidistant)
- Jflon0/lat0/horizon/scale** (Gnomonic)
- Jglon0/lat0/scale** (Orthographic)
- Jslon0/lat0[/slat]/scale** (General Stereographic)

**CONIC PROJECTIONS:**

- Jb***lon0/lat0/lat1/lat2/scale* (Albers)
- Jd***lon0/lat0/lat1/lat2/scale* (Equidistant)
- Jl***lon0/lat0/lat1/lat2/scale* (Lambert)

#### MISCELLANEOUS PROJECTIONS:

- Jh***lon0/scale* (Hammer)
- Ji***lon0/scale* (Sinusoidal)
- Jk[fls]***lon0/scale* (Eckert IV (f) and VI (s))
- Jn***lon0/scale* (Robinson)
- Jr***lon0/scale* (Winkel Tripel)
- Jv***lon0/scale* (Van der Grinten)
- Jw***lon0/scale* (Mollweide)

#### NON-GEOGRAPHICAL PROJECTIONS:

- Jp[a]***scale[/origin][r/z]* (Polar coordinates (theta,r))
- Jxx-scale[dllppow/t/T][y-scale[dllppow/t/T]]** (Linear, log, and power scaling)
- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd:mm:ss.xxx*[*W|E|S|N*] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).
- D***longitude/latitude*  
Maps the bubble at given longitude and latitude point.
- Msize** Sets the size of the beach ball to plot polarities in. *Size* is in inch (unless **c**, **i**, **m**, or **p** is appended).
- S**<*symbol\_type*><*size*>  
Selects *symbol\_type* and *symbol size*. *Size* is in inch (unless **c**, **i**, **m**, or **p** is appended). Choose symbol type from *st(a)r*, (*c*)ircle, (*d*)iamond, (*h*)exagon, (*i*)nverted triangle, (*p*)oint, (*s*)quare, (*t*)riangle, (*x*)cross.

#### OPTIONS

No space between the option flag and the associated arguments.

- B** Sets map boundary annotation and tickmark intervals; see the **psbasemap** man page for all the details.
- C** Offsets focal mechanisms to the latitude and longitude specified in the last two columns of the input file.
- Efill** Selects filling of symbols for stations in extensive quadrants. Set the shade (0–255) or color (*r/g/b*) [Default is 250/250/250]. If **-Efill** is the same as **-Ffill**, use **-e** to outline.
- e[pen]**  
Outline symbols in extensive quadrants using *pen* or the default pen (see **-W**).
- Ffill** Sets background color of the beach ball. Default is no fill.
- f[pen]**  
Outline the beach ball using *pen* or the default pen (see **-W**).
- Gfill** Selects filling of symbols for stations in compressional quadrants. Set the shade (0–255) or color (*r/g/b*) [Default is 0/0/0].
- g[pen]**  
Outline symbols in compressional quadrants using *pen* or the default pen (see **-W**).
- H** Input file(s) has Header record(s). Number of header records can be changed by editing your *.gmtdefaults4* file. If used, **GMT** default is 1 header record. Use **-Hi** if only input data should

- have header records [Default will write out header records if the input data have them].
- h** Use special format derived from HYPO71 output
  - K** More *PostScript* code will be appended later [Default terminates the plot system].
  - N** Does **NOT** skip symbols that fall outside map border [Default plots points inside border only].
  - O** Selects Overlay plot mode [Default initializes a new plot system].
  - P** Selects Portrait plotting mode [GMT Default is Landscape, see **gmtdefaults** to change this].
  - shalf-size/[V[v\_width/h\_length/h\_width/shape]][Gr/g/b][L]**  
Plots S polarity azimuth.  
S polarity is in last column. It may be a vector (**V** option) or a segment. Give half-size,v\_width,h\_length,h\_width in inch (unless **c**, **i**, **m**, or **p** is appended). [**L**] option is for outline.
  - Tangle/form/justify/fontsize in points**  
To write station code. [Default is 0.0/0/5/12].
  - tpen** Set pen color to write station code. Default uses the default pen (see **-W**).
  - U** Draw Unix System time stamp on plot. User may specify where the lower left corner of the stamp should fall on the page relative to lower left corner of plot. Optionally, append a label, or **c** (which will plot the command string.). The GMT parameters **UNIX\_TIME** and **UNIX\_TIME\_POS** can affect the appearance; see the **gmtdefaults** man page for details.
  - V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
  - W** Set current pen attributes  
[Defaults: width = 1, color = 0/0/0, texture = solid].
  - X -Y** Shift origin of plot by (*x-shift,y-shift*), and optionally append units (**c**, **i**, **m**, **p**). Prepend **a** for absolute coordinates; the default (**r**) will reset plot origin. Give **c** to center plot using current page size.
  - :** Toggles between (longitude,latitude) and (latitude,longitude) input and/or output. [Default is (longitude,latitude)]. Append **i** to select input only or **o** to select output only. [Default affects both].
  - c** Specifies the number of plot copies. [Default is 1].

## EXAMPLES

```

pspolar -R239/240/34/35.2 -JM8 -N -Sc0.4 -H1 -D239.5/34.5 -M5 <<END>! test.ps
stat azim ih pol
0481 11 147 c
6185 247 120 d
0485 288 114 +
0490 223 112 -
0487 212 109 .
END
or
pspolar -R239/240/34/35.2 -JM8 -N -Sc0.4 -H1 -D239.5/34.5 -M5 -h <<END>! test.ps
Date Or. time stat azim ih
910223 1 22 0481 11 147 ipu0
910223 1 22 6185 247 120 ipd0
910223 1 22 0485 288 114 epu0
910223 1 22 0490 223 112 epd0
910223 1 22 0487 212 109 epu0
END

```

**SEE ALSO**

*GMT(1)*, *psbasemap(1)*, *psxy(1)*

**REFERENCES**

Bomford, G., Geodesy, 4th ed., Oxford University Press, 1980.

Aki, K. and P. Richards, Quantitative Seismology, Freeman, 1980.

**AUTHORS**

Genevieve Patau  
Seismology Dept.  
Institut de Physique du Globe de Paris  
(patau@ipgp.jussieu.fr)

**NAME**

psvelo – Plot velocity vectors, crosses, and wedges on maps

**SYNOPSIS**

```
psvelo files -Jparameters -Rwest/east/south/north[r] [ -B[pls]parameters ] [ -Dsigma_scale ] [ -Fcolor ]
[ -Ecolor ] [ -Gfill ] [ -H[i][nrec] ] [ -K ] [ -L ] [ -N ] [ -O ] [ -P ] [ -Ssymbol/scale/conffont_size ] [
-U[dx/dy]/[label] ] [ -V ] [ -Wpen ] [ -X[alclr][x-shift[u]] ] [ -X[alclr][y-shift[u]] ] [ -:[ilo] ] [ -copies ]
```

**DESCRIPTION**

**psvelo** reads data values from *files* [or standard input] and generates *PostScript* code that will plot velocity arrows on a map. Most options are the same as for **psxy**, except **-S**. The *PostScript* code is written to standard output. The previous version (**psvelomeca**) is now obsolete. It has been replaced by **psvelo** and **psmeca**.

**ARGUMENTS**

*files* List one or more file-names. If no files are given, **psvelo** will read standard input.

- J** Selects the map projection. Scale is UNIT/degree, 1:xxxxx, or width in UNIT (upper case modifier). UNIT is cm, inch, or m, depending on the **MEASURE\_UNIT** setting in `.gmtdefaults4`, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width value. For map height, max dimension, or min dimension, append **h**, **+**, or **-** to the width, respectively. More details can be found in the **psbasemap** man pages.

**CYLINDRICAL PROJECTIONS:**

- Jclon0/lat0/scale** (Cassini)
- Jjlon0/scale** (Miller)
- Jmscale** (Mercator - Greenwich and Equator as origin)
- Jmlon0/lat0/scale** (Mercator - Give meridian and standard parallel)
- Joalon0/lat0/azimuth/scale** (Oblique Mercator - point and azimuth)
- Joblon0/lat0/lon1/lat1/scale** (Oblique Mercator - two points)
- Joclon0/lat0/lonp/latp/scale** (Oblique Mercator - point and pole)
- Jqlon0/scale** (Equidistant Cylindrical Projection (Plate Carree))
- Jtlon0/scale** (TM - Transverse Mercator, with Equator as  $y = 0$ )
- Jtlon0/lat0/scale** (TM - Transverse Mercator, set origin)
- Juzone/scale** (UTM - Universal Transverse Mercator)
- Jylon0/lats/scale** (Basic Cylindrical Projection)

**AZIMUTHAL PROJECTIONS:**

- Jalon0/lat0/scale** (Lambert)
- Jelon0/lat0/scale** (Equidistant)
- Jflon0/lat0/horizon/scale** (Gnomonic)
- Jglon0/lat0/scale** (Orthographic)
- Jslon0/lat0/[slat]/scale** (General Stereographic)

**CONIC PROJECTIONS:**

- Jblon0/lat0/lat1/lat2/scale** (Albers)
- Jdlon0/lat0/lat1/lat2/scale** (Equidistant)
- Jllon0/lat0/lat1/lat2/scale** (Lambert)

**MISCELLANEOUS PROJECTIONS:**

- Jhlon0/scale** (Hammer)
- Jilon0/scale** (Sinusoidal)

- Jk**[fls]lon0/scale (Eckert IV (f) and VI (s))
- Jn**lon0/scale (Robinson)
- Jr**lon0/scale (Winkel Tripel)
- Jv**lon0/scale (Van der Grinten)
- Jw**lon0/scale (Mollweide)

#### NON-GEOGRAPHICAL PROJECTIONS:

- Jp**[a]scale[/origin][r]z (Polar coordinates (theta,r))
- Jxx**-scale[dlllppow|t|T][y-scale[dlllppow|t|T]] (Linear, log, and power scaling)
- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd:mm[:ss.xxx]*[W|E|S|N] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).
- S**Selects the meaning of the columns in the data file and the figure to be plotted.
- Se**velscale/confidence/fontsize.  
Velocity ellipses in (N,E) convention. *Vscale* sets the scaling of the velocity arrows. This scaling gives inches (unless **c**, **i**, **m**, or **p** is appended). *Confidence* sets the 2-dimensional confidence limit for the ellipse, e.g., 0.95 for 95% confidence ellipse. *FontSize* sets the size of the text in points. The ellipse will be filled with the color or shade specified by the **-G** option [default transparent]. The arrow and the circumference of the ellipse will be drawn with the pen attributes specified by the **-W** option. Parameters are expected to be in the following columns:
  - 1,2 longitude, latitude of station (-: option interchanges order)
  - 3,4 eastward, northward velocity (-: option interchanges order)
  - 5,6 uncertainty of eastward, northward velocities (1-sigma) (-: option interchanges order)
  - 7 correlation between eastward and northward components
  - 8 name of station (optional).
- Sn**barscale.  
Anisotropy bars. *Barscale* sets the scaling of the bars This scaling gives inches (unless **c**, **i**, **m**, or **p** is appended). Parameters are expected to be in the following columns:
  - 1,2 longitude, latitude of station (-: option interchanges order)
  - 3,4 eastward, northward components of anisotropy vector (-: option interchanges order)
- Sr**velscale/confidence/fontsize  
Velocity ellipses in rotated convention. *Vscale* sets the scaling of the velocity arrows. This scaling gives inches (unless **c**, **i**, **m**, or **p** is appended). *Confidence* sets the 2-dimensional confidence limit for the ellipse, e.g., 0.95 for 95% confidence ellipse. *FontSize* sets the size of the text in points. The ellipse will be filled with the color or shade specified by the **-G** option [default transparent]. The arrow and the circumference of the ellipse will be drawn with the pen attributes specified by the **-W** option. Parameters are expected to be in the following columns:
  - 1,2 longitude, latitude, of station (-: option interchanges order)
  - 3,4 eastward, northward velocity (-: option interchanges order)
  - 5,6 semi-major, semi-minor axes
  - 7 counter-clockwise angle, in degrees, from horizontal axis to major axis of ellipse.

**8** name of station (optional)

**-Sw***wedge\_scale/wedge\_mag*.

Rotational wedges. *Wedge\_scale* sets the size of the wedges in inches (unless **c**, **i**, **m**, or **p** is appended). Values are multiplied by *Wedge\_mag* before plotting. For example, setting *Wedge\_mag* to 1.e7 works well for rotations of the order of 100 nanoradians/yr. Use **-G** to set the fill color or shade for the wedge, and **-E** to set the color or shade for the uncertainty. Parameters are expected to be in the following columns:

**1,2** longitude, latitude, of station (**-:** option interchanges order)  
**3** rotation in radians  
**4** rotation uncertainty in radians

**-Sx***cross\_scale*

gives Strain crosses. *Cross\_scale* sets the size of the cross in inches (unless **c**, **i**, **m**, or **p** is appended). Parameters are expected to be in the following columns:

**1,2** longitude, latitude, of station (**-:** option interchanges order)  
**3** eps1, the most extensional eigenvalue of strain tensor, with extension taken positive.  
**4** eps2, the most compressional eigenvalue of strain tensor, with extension taken positive.  
**5** azimuth of eps2 in degrees CW from North.

## OPTIONS

No space between the option flag and the associated arguments.

- A** *Arrow\_width/Head\_length/Head\_width* Size of arrow in inches. [Default is 0.03/0.12/0.09].
- B** Sets map boundary annotation and tickmark intervals; see the **psbasemap** man page for all the details.
- C** Offset focal mechanisms to the latitude and longitude specified in the last two columns of the input file.
- D** *Sigma\_scale* can be used to rescale the uncertainties of velocities (**-Se** and **-Sr**) and rotations (**-Sw**). Can be combined with the *confidence* variable.
- Ffill** Sets the color or shade used for frame and annotation. [Default is 0/0/0 (black)]
- Efill** Sets the color or shade used for filling uncertainty wedges (**-Sw**) or velocity error ellipses (**-Se** or **-Sr**). [If **-E** is not specified, the uncertainty regions will be transparent.]
- Gfill** Select filling of ellipses, wedges, and focal mechanisms. By convention, the compressional quadrants of the focal mechanism beach balls are shaded. Set the shade (0–255) or color (r/g/b) [Default is 0/0/0]. Optionally, specify **-Gp***icon\_size/pattern*, where *pattern* gives the number of the image pattern (1-90) OR the name of a icon-format file. *icon\_size* sets the unit size in inches. To invert black and white pixels, use **-GP** instead of **-Gp**. See **pspatterns** for information on individual patterns.
- H** Input file(s) has Header record(s). Number of header records can be changed by editing your *.gmtdefaults4* file. If used, **GMT** default is 1 header record. Use **-Hi** if only input data should have header records [Default will write out header records if the input data have them].
- K** More *PostScript* code will be appended later [Default terminates the plot system].
- L** Draw lines. Ellipses and fault planes will have their outlines drawn using current pen (see **-W**).
- N** Do **NOT** skip symbols that fall outside the frame boundary specified by **-R**. [Default plots symbols inside frame only].

- O** Selects Overlay plot mode [Default initializes a new plot system].
- P** Selects Portrait plotting mode [GMT Default is Landscape, see **gmtdefaults** to change this].
- U** Draw Unix System time stamp on plot. User may specify where the lower left corner of the stamp should fall on the page relative to lower left corner of plot. Optionally, append a label, or **c** (which will plot the command string.). The **GMT** parameters **UNIX\_TIME** and **UNIX\_TIME\_POS** can affect the appearance; see the **gmtdefaults** man page for details.
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- W** Set pen attributes for velocity arrows, ellipse circumference and fault plane edges. [Defaults: width = 1, color = 0/0/0, texture = solid].
- X -Y** Shift origin of plot by (*x-shift,y-shift*), and optionally append units (**c**, **i**, **m**, **p**). Prepend **a** for absolute coordinates; the default (**r**) will reset plot origin. Give **c** to center plot using current page size.
- :** Toggles between (longitude,latitude) and (latitude,longitude) input and/or output. [Default is (longitude,latitude)]. Append **i** to select input only or **o** to select output only. [Default affects both].
- c** Specifies the number of plot copies. [Default is 1].

### EXAMPLES

The following should make big red arrows with green ellipses, outlined in red. Note that the 39% confidence scaling will give an ellipse which fits inside a rectangle of dimension *Esig* by *Nsig*.

```
psvelo << END -H2 -R-10/10/-10/10 -W1/255/0/0 .br -G0/255/0 -L -Se0.2/0.39/18 -B1g1 -Jx0.4/0.4
.br -A0.1/0.3/0.3 -P -V >! test.ps
Long. Lat. Evel Nvel Esig Nsig CorEN SITE
(deg) (deg) (mm/yr) (mm/yr)
0. -8. 0.0 0.0 4.0 6.0 0.500 4x6
-8. 5. 3.0 3.0 0.0 0.0 0.500 3x3
0. 0. 4.0 6.0 4.0 6.0 0.500
-5. -5. 6.0 4.0 6.0 4.0 0.500 6x4
5. 0. -6.0 4.0 6.0 4.0 -0.500 -6x4
0. -5. 6.0 -4.0 6.0 4.0 -0.500 6x-4
END
```

This example should plot some residual rates of rotation in the Western Transverse Ranges, California. The wedges will be dark gray, with light gray wedges to represent the 2-sigma uncertainties.

```
psvelo <<END -Sw0.4/1.e7 -W3 -G60 -E210 -H1 -D2 .br -Jm2.2 -R240./243./32.5/34.75
-Bf10ma60m/WeSn -P >! test.ps
lon lat spin(rad/yr) spin_sigma (rad/yr)
241.4806 34.2073 5.65E-08 1.17E-08
241.6024 34.4468 -4.85E-08 1.85E-08
241.0952 34.4079 4.46E-09 3.07E-08
241.2542 34.2581 1.28E-07 1.59E-08
242.0593 34.0773 -6.62E-08 1.74E-08
241.0553 34.5369 -2.38E-07 4.27E-08
241.1993 33.1894 -2.99E-10 7.64E-09
241.1084 34.2565 2.17E-08 3.53E-08
END
```

### SEE ALSO

*GMT*(l), *psbasemap*(l), *psxy*(l)

**REFERENCES**

Bomford, G., Geodesy, 4th ed., Oxford University Press, 1980.

**AUTHORS**

Kurt Feigl

CNRS UMR 5562

Toulouse, France

(Kurt.Feigl@cnes.fr)

Genevieve Patau

CNRS UMR 7580

Seismology Dept.

Institut de Physique du Globe de Paris

(patau@ipgp.jussieu.fr)

**NAME**

`mgd77convert` – Translate between different formats of MGD77 files

**SYNOPSIS**

`mgd77convert` *NGDC-ids* **-F***albt* **-T**[+]*albt* [ **-L**[*w*][*e*][+ ] ] [ **-V** ] [ **-4** ]

**DESCRIPTION**

**mgd77convert** reads versions of MGD77 files and writes the same data in (probably) another format to a new file in the current directory.

*NGDC-ids*

Can be one or more of five kinds of specifiers:

- 1) 8-character NGDC IDs, e.g., 01010083, JA010010etc., etc.
- 2) 2-character <agency> codes which will return all cruises from each agency.
- 3) 4-character <agency><vessel> codes, which will return all cruises from those vessels.
- 4) =<list>, where <list> is a table with NGDC IDs, one per line.
- 5) If nothing is specified we return all cruises in the data base.

(See `mgd77info -F` for agency and vessel codes). The ".mgd77" or ".nc" extensions will automatically be appended, if needed (use **-I** to ignore certain file types). Cruise files will be looked for first in the current directory and second in all directories listed in `$MGD77_HOME/mgd77_paths.txt` [If `$MGD77_HOME` is not set it will default to `$GMTHOME/share/mgd77`].

NOTE: Under Windows, only (1) and (4) are currently supported.

- F** Specifies the format of the input (From) files. Choose from **a** for standard MGD77 ASCII table (with extension .mgd77), **b** for the new MGD77+ netCDF format (with extension .nc), and **t** for a plain ASCII tab-separated table dump (with extension .dat).
- T** Specifies the format of the output (To) files. Choose from **a** for standard MGD77 ASCII table (with extension .mgd77), **b** for the new MGD77+ netCDF format (with extension .nc), and **t** for a plain ASCII tab-separated table dump (with extension .dat). We will refuse to create the file(s) if they already exist in the current directory. Prepend **+** to override this policy.

**OPTIONS**

No space between the option flag and the associated arguments.

- L** Set the level of verification reporting [none] and where to send such reports [stderr]. Append a combination of **w** for warnings, **e** for errors, and **+** to send such log information to stdout.
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- 4** By default, the storage types used in a MGD77+ netCDF file greatly exceed the precision imposed by the ASCII MGD77 format. However, for the five items **faa**, **eot**, **mag**, **diur** and **msd** we use 2-byte integers with implied precisions of 0.1 mGal, 0.1 nTesla, and 1 m as in the MGD77 format. It is possible that at some point these items will need to be stored as 4-byte ints which would allow precisions of 1 fTesla, 1 nGal, and 0.01 mm, respectively. This option activates such storage [Default uses 2-byte integers].

**EXAMPLES**

To convert 01010047.mgd77 and 01010008.mgd77 to new netCDF .nc files, and capture all verification messages, try

```
mgd77convert 01010047 01010008 -Fa -Tc -V -Lew+ > log.lis
```

To convert 01010047.nc back to MGD77 ASCII and make sure it is identical to the original file, try (Bourne shell syntax)

```
orig='mgd77path 01010047 -Ic'
mgd77convert 01010047 -Fc -Ta -V
diff $orig 01010047.mgd77
```

To convert 01010047.nc to a plain ASCII table for manual editing, overwriting any existing table, try

```
mgd77convert 01010047 -Fc -T+t -V
```

## FILE FORMATS

**mgd77convert** handles three different formats. (1) The MGD77 ASCII tables are the established standard for distribution of underway geophysical data to and from the NGDC data center. Normally, only the ship-operations people and the cruise PI might be involved in *making* an MGD77 ASCII file for transmission to NGDC; users are more interested in *reading* such files. (2) The MGD77+ netCDF format was developed to facilitate the use of MGD77 data by scientists. It contains all the information of the original MGD77 file and if you convert back and forth you end up with the original. However, file sizes are typically ~30% of the original ASCII format and is much faster to operate on. (3) The plain ASCII tab-separated dump is available for users who need to manually edit the content of a MGD77 file. This is usually easier to do when the columns are tab-separated than when they are all crunched together in the MGD77 punch-card format.

## OTHER TOOLS

The MGD77+ netCDF files are CF-1.0 and COARDS compliant and can be examined with general-purpose tools such as ncBrowse and ncView.

## SEE ALSO

*mgd77manage(1)*, *mgd77list(1)*, *mgd77sample(1)*, *mgd77track(1)* *x2sys\_init(1)*

## REFERENCES

- Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.
- Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.
- Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.
- ncBrowse, see <http://www.epic.noaa.gov/java/ncBrowse/>
- ncView, see [http://meteora.ucsd.edu/~pierce/ncview\\_home\\_page.html](http://meteora.ucsd.edu/~pierce/ncview_home_page.html)
- The Marine Geophysical Data Exchange Format - "MGD77", see <http://www.ngdc.noaa.gov/mgg/dat/geodas/docs/mgd77.txt>

**NAME**

mgd77info – Get information about MGD77[+] files

**SYNOPSIS**

**mgd77info** *NGDC-ids* [ **-C**[*mle*] ] [ **-E**[*mle*] ] [ **-I***gnore* ] [ **-M**[*item*]**lr** ] [ **-L**[*v*] ] [ **-V** ]

**DESCRIPTION**

**mgd77info** reads <legid>.[mgd77]nc files and produces a single record of information about each cruise specified. The information includes beginning and end times, total track distances in km, longitude and latitude range, and the total number of geophysical observations. Optionally, choose instead to see the original MGD77 header meta-data section or its individual members.

If you need to know which tracks are crossing through a given region and what kinds of geophysical observations are available, consider using the x2sys tools to set up a tracks index data base (see **x2sys\_init** for more information).

*NGDC-ids*

Can be one or more of five kinds of specifiers:

- 1) 8-character NGDC IDs, e.g., 01010083, JA010010etc., etc.
- 2) 2-character <agency> codes which will return all cruises from each agency.
- 3) 4-character <agency><vessel> codes, which will return all cruises from those vessels.
- 4) =<list>, where <list> is a table with NGDC IDs, one per line.
- 5) If nothing is specified we return all cruises in the data base.

(See mgd77info **-F** for agency and vessel codes). The ".mgd77" or ".nc" extensions will automatically be appended, if needed (use **-I** to ignore certain file types). Cruise files will be looked for first in the current directory and second in all directories listed in **\$MGD77\_HOME/mgd77\_paths.txt** [If **\$MGD77\_HOME** is not set it will default to **\$GMTHOME/share/mgd77**].

NOTE: Under Windows, only (1) and (4) are currently supported.

**OPTIONS**

No space between the option flag and the associated arguments.

- C** List abbreviations for all columns present in the MGD77[+] files. Append **m** or **e** to limit the display to the MGD77 standard or MGD77+ extended set only.
- E** Give a one-line summary for each cruise listed.
- M** List the meta-data (header) and (if present) the MGD77+ history for each cruise. Append **f** for a formatted display. This will list individual parameters and their values, one entry per output line, in a format that can be searched using standard UNIX text tools. Alternatively, append the name of a particular parameter (you only need to give enough characters - starting at the beginning - to uniquely identify the item). Give **-** to display the list of all parameter names. You may also specify the number of a parameter. For the raw, punchcard-formatted MGD77 original header block, append **r** instead.
- I** Ignore certain data file formats from consideration. Append **alcft** to ignore MGD77 ASCII, MGD77+ netCDF, or plain tab-separated ASCII table files, respectively. The option may be repeated to ignore more than one format. [Default ignores none].
- L** No cruise information is listed. Instead, we just display a list of the GEODAS institution 2-character codes and their names. Optionally, append **v** to also display the vessels and their 4-character codes for each institution. The following is the list of institutions:  
 (01) LAMONT (LDEO), (02) WOODS HOLE O.I., (03) NOAA, (04) US ARMY, (05) NEW ZEALAND, (06) US GEOL. SURVEY, (07) OREGON ST. UNIV, (08) U.HAWAII SOEST, (09) US NAVY, (10) UNIV OF TEXAS, (11) RICE UNIV., (12) CANADA, (13) UNIV OF CONN., (14) U.MIAMI (RSMAS), (15) SCRIPPS INST.OC, (16) CHINA, (17) U RHODE ISLAND, (18) DUKE UNIVERSITY, (19) UNITED KINGDOM, (20) U.WASHINGTON, (22) WESTERN GEOPHY., (23) TEXAS A&M UNIV., (24) AUSTRALIA, (25) MONACO, (29) RUSSIA, (30) SPAIN, (35) NIMA, (58) NETHERLANDS, (60) MIN MGMT SVC, (63) ISRAEL, (67) FRANCE, (71) SOUTH AFRICA, (75) US COAST GUARD, (76) BRAZIL, (77) INT. GRAV.

BUR, (83) GERMANY, (84) ORSTOM NEW CAL, (86) CUBA, (87) ARGENTINA, (88) US NSF, (89) INDIA, (90) PORTUGAL, (92) FINLAND, (93) CHILE, (J1) HYDR DEPT JAPAN, (J2) GEOL SRVY JAPAN, (J4) UNIV TOKYO, (J5) KOBE UNIV, (J7) UNIV OF RYUKYUS, (J8) J.O.D.C. JAPAN, (J9) CHIBA UNIV, (JA) INST.POLAR RES., (ZZ) INST NOT CODED.

**-V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].

## EXAMPLES

To get one-line summary information about the cruises 01010047.mgd77 and 01010008.mgd77, try

```
mgd77info 01010047 01010008 -E > listing.lis
```

To see the original raw MGD77 header meta-data for cruise 01010047.mgd77, run

```
mgd77info 01010047 -Mr
```

To determine all the parameters related to Gravity during cruise 01010047.mgd77, run

```
mgd77info 01010047 -Mf | grep Gravity
```

To determine the Magnetic sampling rate used during cruise 01010047.mgd77, run

```
mgd77info 01010047 -MfMagnetics_Sampling_Rate
```

To see all the columns that the MGD77+ cruise 01010047.nc contains, run

```
mgd77info 01010047.nc -C
```

## SEE ALSO

*mgd77list*(l), *mgd77manage*(l), *mgd77path*(l), *mgd77track*(l) *x2sys\_init*(l)

## REFERENCES

Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.

Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.

Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

The Marine Geophysical Data Exchange Format - "MGD77", see <http://www.ngdc.noaa.gov/mgg/dat/geodas/docs/mgd77.txt>

**NAME**

mgd77list – A data-extractor for MGD77[+] files

**SYNOPSIS**

```
mgd77list NGDC-ids -Fcolumns[,logic][:bittests] [ -Acdlflmcode ] [ -Cflag ] [ -Dstartdate ] [ -Dstopdate ] [ -E ] [ -H[i][nrec] ] [ -Ignore ] [ -X[corrtable] ] [ -M[flag] ] [ -Ndlsunit ] [ -Rwest/east/south/north[r] ] [ -Sstartdist[unit] ] [ -Sstopdist[unit] ] [ -T[mle] ] [ -V ] [ -Wweight ] [ -Z+|- ] [ -b[s|S|d|D][ncol] ]
```

**DESCRIPTION**

**mgd77list** reads <NGDC-id>.[mgd77|nc] files and produces an ASCII [or binary] table. The <NGDC-id>.[mgd77|nc] files contain track information such as leg-id, time and position, geophysical observables such as gravity, magnetics, and bathymetry, and control codes and corrections such as Eotvos and diurnal corrections. The MGD77+ extended netCDF files may also contain additional user columns (for a listing of available columns, use **mgd77info -C**, and to learn how to add your own custom columns, see **mgd77manage**). The user may extract any combination of these parameters, any of six computed quantities (distance, heading, velocity, Carter correction, and gravity and magnetic global reference fields), calendar sub-units of time (year, month, day, hour, min, sec), the NGDC id, and finally a preset weight (see **-W**). A sub-section can be specified by passing time- or distance-intervals along track or by selecting a geographical region. Finally, each output record may be required to pass any number of logical tests involving data values or bit flags.

*NGDC-ids*

Can be one or more of five kinds of specifiers:

- 1) 8-character NGDC IDs, e.g., 01010083, JA010010etc., etc.
- 2) 2-character <agency> codes which will return all cruises from each agency.
- 3) 4-character <agency><vessel> codes, which will return all cruises from those vessels.
- 4) =<list>, where <list> is a table with NGDC IDs, one per line.
- 5) If nothing is specified we return all cruises in the data base.

(See **mgd77info -F** for agency and vessel codes). The ".mgd77" or ".nc" extensions will automatically be appended, if needed (use **-I** to ignore certain file types). Cruise files will be looked for first in the current directory and second in all directories listed in **\$MGD77\_HOME/mgd77\_paths.txt** [If **\$MGD77\_HOME** is not set it will default to **\$GMTHOME/share/mgd77**].

NOTE: Under Windows, only (1) and (4) are currently supported.

**-F** The required *columns* string must be a comma-separated list of parameter abbreviations given in the desired output order. Any parameters given in UPPER case must not be NaN in a record for output to occur. Unless specified separately, the output format (if ASCII) is controlled by the GMT parameter **D\_FORMAT**. The available abbreviations are:

**drt** The digital record type, usually 3 or 5 (for Y2K-compliant cruises).

**id** The survey ID string (leg name).

**ngdcid** The 8-character NGDC cruise ID string (usually the file prefix).

**time** Choose between Absolute calendar time (**atime**, the default) in the format dictated by the GMT parameters **OUTPUT\_DATE\_FORMAT** and **OUTPUT\_CLOCK\_FORMAT**, or Relative time (**rtime**) in the format dictated by the GMT parameters **D\_FORMAT** and **TIME\_SYSTEM** (or **TIME\_EPOCH** and **TIME\_UNIT**).

**lon** Longitude in the format dictated by the GMT parameter **OUTPUT\_DEGREE\_FORMAT**.

**lat** Longitude in the format dictated by the GMT parameter **OUTPUT\_DEGREE\_FORMAT**.

**twt** Two-Way Travel time (in s).

**depth** Corrected bathymetry (in m, positive below sealevel).

**mtf1** Magnetic Total Field intensity from sensor 1 (in nTesla).

<b>mtf2</b>	Magnetic Total Field intensity from sensor 2 (in nTesla).
<b>mag</b>	Residual magnetic anomaly (in nTesla).
<b>gobs</b>	Observed gravity (in mGal).
<b>faa</b>	Free-air gravity anomaly (in mGal).
<b>ptc</b>	Position Type Code (1 = fix, 3 = interpolated, 9 = unspecified).
<b>bcc</b>	Bathymetric Correction Code, indicating the procedure used to convert travel time to depth. (01-55 = Matthews' zone used to correct the depth, 59 = Matthews' corrections used but the zones is unspecified in the data record, 60 = S. Kuwahara formula for T-S, 61 = Wilson formula for T-S, 62 = Del Grosso formula for T-S, 63 = Carter's tables, 88 = Other, described in header sections, 99 = unspecified).
<b>btc</b>	Bathymetric Type Code, indicating how the bathymetry value was obtained (1 = observed, 3 = interpolated, 9 = unspecified).
<b>msens</b>	Magnetic sensor for used to evaluate the residual field (1 = 1st or leading sensor, 2 = 2nd or trailing sensor, 9 = unspecified).
<b>msd</b>	Depth (or altitude) of the magnetic sensor (in m, positive below sealevel).
<b>diur</b>	Magnetic diurnal correction (in nTesla).
<b>eot</b>	Eotvos correction (in mGal).
<b>sln</b>	Seismic Line Number string.
<b>sspn</b>	Seismic Shot Point Number string.
<b>nqc</b>	Navigation Quality Code (5 = suspected, by source institution, 6 = suspected, by NGDC, 9 = no problems identified).

In addition, the following derived quantities can be requested:

<b>year</b>	The year of each record.
<b>month</b>	The month of each record.
<b>day</b>	The day of the month of each record.
<b>hour</b>	The hour of each record.
<b>min</b>	The minutes of each record.
<b>sec</b>	The decimal seconds of each record.
<b>dist</b>	Along-track distance from start of leg. For method of calculation, see <b>-C</b> [spherical great circle distances], and for distance units, see <b>-N</b> [km].
<b>az</b>	Ship azimuth (heading) measured clockwise from north (in degrees).
<b>vel</b>	Ship speed; see <b>-N</b> for units [m/s].
<b>weight</b>	Weight assigned to this data set (see <b>-W</b> ).
<b>carter</b>	Carter depth correction, if <b>twt</b> is present in file (in m). Sign: Correction is to be subtracted from uncorrected depths to yield a corrected depth.
<b>igrf</b>	International geomagnetic reference field (total field) (in nTesla).
<b>ngrav</b>	International Gravity reference Field ("normal gravity") (in mGal). Field is selected based on the parameter Gravity Theoretical Formula Code in the cruise's MGD77 header. If this is not set or is invalid we default to the IGF 1980. Alternatively, specify the field directly using <b>-Af</b> (see that option for more details).

The following short-hand flags are also recognized:

- all** This returns all data columns in the file.
- mgd77** This results in all 27 MGD77 fields being written out in the official MGD77 order.
- geo** This limits the output to 10 fields (**time**, **lon**, **lat** plus the seven geophysical observations **twf**, **depth**, **mtf1**, **mtf2**, **mag**, **gobs**, and **faa**). By appending **+** to either of these set we will also append **dist**, **azim**, **vel**, and **weight** as listed above.

As an option, logical tests may be added for any of the parameters by appending *,logic*, which is itself composed of one or more comma-separated instructions of the form *parOPvalue*, where *par* is one of the parameters listed above, **OP** is a logical operator (<, <=, =, !=, >=, >, |), and *value* is a constant used in the comparison. Floating point parameters are compared numerically; character parameters are compared lexically (after leading and trailing blanks have been removed). The bit comparison (|) means that at least one of the bits in *value* must be turned on in *par*. At least one of the tests must be true for the record to be output, except for tests using UPPER case parameters which all must be true for output to occur. Note that specifying a test does not imply that the corresponding column will be included in the output stream; it must be present in *columns* for that to occur. Note: some of the operators are special UNIX characters and you are advised to place quotes around the entire argument to **-F**.

Finally, for MGD77+ files you may optionally append *:bittests* which is : (a colon) followed by one or more comma-separated *+col* terms. This compares specific bitflags only for each listed column. Here, **+** means the chosen bit must be 1 (ON) whereas **-** means it must be 0 (OFF). All bit tests given must be passed. By default, MGD77+ files that have the special **MGD77\_flags** column present will use those flags, and observations associated with ON-bits (meaning they are flagged as bad) will be set to NaN; append : with no trailing information to turn this behavior off (i.e., no bit flags will be consulted).

## OPTIONS

No space between the option flag and the associated arguments.

- A** By default, corrected depth (**depth**), magnetic residual anomaly (**mag**), free-air gravity anomaly (**faa**), and the derived quantity Carter depth correction (**carter**) are all output as is (if selected in **-F**); this option adjusts that behavior. For each of these columns there are 2–4 ways to adjust the data. Append **c**(arter), **d**(epth), **f**(aa), or **m**(ag) and select the *code* for the procedure you want applied. You may select more than one procedure for a data column by summing their numerical *codes* (1, 2, 4, and 8). E.g., **-Ac3** will first try method **-Ac1** to estimate a Carter correction but if **depth** is NaN we will next try **-Ac2** which only uses **twf**. In all cases, if any of the values required by an adjustment procedure is NaN then the result will be NaN.
- Ac** Determines how the **carter** correction term is calculated. Below, **C(twf)** stands for the Carter-corrected depth (it also depends on **lon**, **lat**), **U(twf, v)** is the uncorrected depth ( $= \text{twf} * v / 2$ ) using as *v* the "Assumed Sound Velocity" parameter in the MGD77 header (if it is a valid velocity, otherwise we default to 1500 m/s); alternatively, append your preferred velocity *v* in m/s, **TU(depth, v)** is the 2-way travel time estimated from the (presumably) uncorrected **depth**, and **TC(depth)** is the 2-way travel time obtained by inverting the (presumably) corrected **depth** using the Carter correction formula. Select from
- Ac1[,v]** returns difference between **U(twf, v)** and **depth** [Default].
  - Ac2[,v]** returns difference between **U(twf, v)** and **Carter (twf)**.
  - Ac4[,v]** returns difference between (assumed uncorrected) **depth** and **Carter (TU(depth))**.
  - Ac8[,v]** returns difference between **U(TC(depth), v)** and **depth**.
- Ad** Determines how the **depth** column output is obtained:
- Ad1** returns **depth** as stored in the data set [Default].
  - Ad2[,v]** returns calculated uncorrected depth **U(twf, v)**.
  - Ad4** returns calculated corrected depth **C(twf)**.
- Af** Determines how the **faa** column output is obtained. If **ngrav** (i.e., the International Gravity reference Field (IGF), or "normal gravity") is required it is selected based on the MGD77 header

- parameter "Theoretical Gravity Formula Code"; if this code is not present or is invalid we default to 4. Alternatively, append the preferred *field* (1–4) to select 1 (Heiskanen 1924), 2 (IGF 1930), 3 (IGF 1967) or 4 (IGF 1980). Select from
- Af1**[*field*] returns **faa** as stored in the data set [Default]. Optionally, sets the IGF *field* to use if you also have requested **ngrav** as an output column in –**F**.
  - Af2**[*field*] returns the difference between **gobs** and **ngrav** (with optional *field* directive).
  - Am** Determines how the **mag** column output is obtained. There may be one or two total field measurements in the file (**mtf1** and **mtf2**), and the column **msens** may state which one is the leading sensor (1 or 2; it may also be undefined). Select from
    - Am1** returns **mag** as stored in the data set [Default].
    - Am2** returns the difference between **mgfx** and **igrf**, where **x** is the leading sensor (**1** or **2**) indicated by the **msens** data field (defaults to **1** if unspecified).
    - Am4** returns the difference between **mgfx** and **igrf**, where **x** is the sensor (**2** or **1**) *not* indicated by the **msens** data field (defaults to **2** if unspecified).
  - C** Append a one-letter code to select the procedure for along-track distance calculation (see –**N** for selecting units):
    - f** Flat Earth distances.
    - g** Great circle distances [Default].
    - e** Geodesic distances on current GMT ellipsoid.
  - Da** Do not list data collected before *startdate* (yyyy-mm-ddT[hh:mm:ss]) [Default is start of cruise].
  - Db** Do not list data collected after *stopdate* (yyyy-mm-ddT[hh:mm:ss]). [Default is end of cruise].
  - E** Exact match: Only output records that match all the requested geophysical columns [Default outputs records that matches at least one of the observed columns].
  - H** Issue a header record with names for each data field.
  - I** Ignore certain data file formats from consideration. Append **alcft** to ignore MGD77 ASCII, MGD77+ netCDF, or plain tab-separated ASCII table files, respectively. The option may be repeated to ignore more than one format. [Default ignores none].
  - L** Apply optimal corrections to columns where such corrections are available. Append the correction table to use [Default uses the correction table *mgd77\_corrections.d* in the **\$MGD77\_HOME** directory]. For the format of this file, see **CORRECTIONS** below.
  - M** Issue a multi-segment header record with cruise ID for each cruise.
  - N** Append **d** for distance or **s** for speed, then give the desired *unit* as **e** (meter or m/s), **k** (km or km/hr), **m** (miles or miles/hr), or **n** (nautical miles or knots). [Default is –**Ndk** –**Nse** (km and m/s)].
  - R** *west*, *east*, *south*, and *north* specify the Region of interest, and you may specify them in decimal degrees or in [+–]dd:mm[ss.xxx][W|E|S|N] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands –**Rg** –**Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).
  - Sa** Do not list data that are less than *startdist* meter along track from port of departure. Append **k** for km, **m** for miles, or **n** for nautical miles [Default is 0 meters].
  - Sb** Do not list data that are more than *stopdist* meters along track from port of departure. Append **k** for km, **m** for miles, or **n** for nautical miles [Default is end of track].
  - T** Turns OFF the otherwise automatic adjustment of values based on correction terms that are stored in the MGD77+ file and used to counteract such things as wrong units used by the source institution when creating the original MGD77 file from which the MGD77+ file derives (the option has no effect on plain MGD77 ASCII files). Append **m** or **e** to limit the option to the MGD77 or extended columns set only [Default applies to both].

- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- W** Set the weight for these data. Weight output option must be set in **-F**. This is useful if the data are to be processed with the weighted averaging techniques offered by **blockmean**, **blockmedian**, and **blockmode** [1].
- Z** Append the sign you want for **depth**, **carter**, and **msd** values below sea level (**-Z-** gives negative bathymetry) [Default is positive down].
- bo** Selects binary output mode (single precision; append **d** for double precision, or use **SID** to swap bytes on output) [Default is ASCII]. **-H** is ignored if **-bo** is selected. Likewise, string-fields cannot be selected.

## EXAMPLES

To get a (distance, heading, gravity, bathymetry) listing from 01010047.mgd77, starting at June 3 1971 20:45 and ending at distance = 5000 km, use the following command:

```
mgd77list 01010047 -Da1971-06-03T20:45 -Sb5000 -Fdist,azim,faa,depth > myfile.d
```

To make input for **blockmean** and **surface** using free-air anomalies from all the cruises listed in the file cruises.lis, but only the data that are inside the specified area, and make the output binary:

```
mgd77list 'cat cruises.lis' -Flon,lat,faa -R-40/-30/25/35 -bo > allgrav.b
```

To extract the locations of depths exceeding 9000 meter that were not interpolated (**btc** != 1) from all the cruises listed in the file cruises.lis:

```
mgd77list 'cat cruises.lis' -F"depth,DEPTH>9000,BTC!=1" > really_deep.d
```

To extract dist, faa, and grav12\_2 from records whose depths are shallower than 3 km and where none of the requested fields are NaN, from all the MGD77+ netCDF files whose cruise ids are listed in the file cruises.lis, we try

```
mgd77list 'cat cruises.lis' -E -Ia -F"dist,faa,grav12_2,depth<3000" > shallow_grav.d
```

To extract dist, faa, and grav12\_2 from all the MGD77+ netCDF files whose cruise ids are listed in the file cruises.lis, but only retrieve records whose bitflag for faa indicates BAD values, we try

```
mgd77list 'cat cruises.lis' -E -Ia -F"dist,faa,grav12_2:+faa" > bad_grav.d
```

To output lon, lat, mag, and faa from all the cruises listed in the file cruises.lis, but recalculate the two residuals based on the latest reference fields, try:

```
mgd77list 'cat cruises.lis' -Flon,lat,mag,faa -Af2,4 -Am2 > data.d
```

## IGRF

The IGRF calculations are based on a Fortran program written by Susan Macmillan, British Geological Survey, translated to C via f2c by Joaquim Luis, U Algarve, and adapted to GMT-style by Paul Wessel.

## IGF

The equations used are reproduced here using coefficients extracted directly from the source code (let us know if you find errors):

$$(1) g = 978052.0 * [1 + 0.005285 * \sin^2(\text{lat}) - 7e-6 * \sin^2(2*\text{lat}) + 27e-6 * \cos^2(\text{lat}) * \cos^2(\text{lon}-18)]$$

$$(2) g = 978049.0 * [1 + 0.0052884 * \sin^2(\text{lat}) - 0.0000059 * \sin^2(2*\text{lat})]$$

$$(3) g = 978031.846 * [1 + 0.0053024 * \sin^2(\text{lat}) - 0.0000058 * \sin^2(2*\text{lat})]$$

$$(4) g = 978032.67714 * [(1 + 0.00193185138639 * \sin^2(\text{lat})) / \sqrt{1 - 0.00669437999013 * \sin^2(\text{lat})}]$$

## CORRECTIONS

The correction table is an ASCII file with coefficients and parameters needed to carry out corrections. Comment records beginning with # are allowed. All correction records are of the form

*cruiseID observation correction*

where *cruiseID* is a NGDC prefix, *observation* is one of the abbreviations for geophysical observations listed under **-F** above, and *correction* consists of one or more *terms* that will be summed up and then **subtracted** from the observation before output. Each *term* must have this exact syntax:

*factor*[\**[function]*]*[(scale)(abbrev[-origin])*]<sup>*power*</sup>

where terms in brackets are optional (the brackets themselves are not used but regular parentheses must be used as indicated). No spaces are allowed except between *terms*. The *factor* is the amplitude of the basis function, while the optional *function* can be one of sin, cos, or exp. The optional *scale* and *origin* can be used to translate the argument (before giving it to the optional function). The argument *abbrev* is one of the abbreviations for observations listed above. If *origin* is given as **T** it means that we should replace it with the value of *abbrev* for the very first record in the file (this is usually only done for *time*). If the first record entry is NaN we revert *origin* to zero. Optionally, raise the entire expression to the given *power*, before multiplying by the amplitude. The following is an example of fictitious corrections to the cruise 99999999, implying the **depth** should have the Carter correction removed, **faa** should have a linear trend removed, the magnetic anomaly (**mag**) should be corrected by a strange dependency on ship heading and latitude, and **gobs** needs to have 10 mGal added (hence given as -10):

```
99999999 depth 1.0*((carter))
99999999 faa 14.1 1e-5*((time-T))
99999999 mag 0.5*cos(0.5*(azim-19))^2 1.0*exp(-1e-3(lat))^1.5
99999999 gobs -10
```

## SEE ALSO

*mgd77convert*(1), *mgd77info*(1), *mgd77manage*(1), *mgd77track*(1)

## REFERENCES

- Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.
- Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.
- Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.
- The Marine Geophysical Data Exchange Format - "MGD77", see <http://www.ngdc.noaa.gov/mgg/dat/geodas/docs/mgd77.txt>
- IGRF, see <http://www.ngdc.noaa.gov/AGA/vmod/igrf.html>

**NAME**

`mgd77manage` – Manage extra columns in MGD77+ files

**SYNOPSIS**

```
mgd77manage NGDC-ids [ -A[+]a|c|d|D|e|g|i|n|t|Tfileinfo ] [ -Cfile ] [ -Dabbrev1,abbrev2,... ] [
-Eempty ] [ -Iabbrev/nameofunit/t/scale/offset/comment ] [ -Nk|l|m|n ] [ -Q[value] ] [ -V ] [
-b[s|S|d|D][ncol] ]
```

**DESCRIPTION**

**mgd77manage** deals with maintaining extra custom columns in MGD77+ netCDF files. You can either delete one or more columns, add a new column, update an existing column with new data, or supply error correction information (\*.e77 files). New data may come from a table (ASCII unless **-b** is used), be based on existing columns and certain theoretical expressions, or they may be obtained by sampling a grid (choose between GMT grid or a Sandwell/Smith Mercator \*.img grid) along track. The new data will be appended to the MGD77+ file in the form of an extra data column of specified type. The data file will be modified; no new file will be created. For the big issues, see the DISCUSSION section below.

*NGDC-ids*

Can be one or more of five kinds of specifiers:

- 1) 8-character NGDC IDs, e.g., 01010083, JA010010etc., etc.
- 2) 2-character <agency> codes which will return all cruises from each agency.
- 3) 4-character <agency><vessel> codes, which will return all cruises from those vessels.
- 4) =<list>, where <list> is a table with NGDC IDs, one per line.
- 5) If nothing is specified we return all cruises in the data base.

(See `mgd77info -F` for agency and vessel codes). The ".mgd77" or ".nc" extensions will automatically be appended, if needed (use **-I** to ignore certain file types). Cruise files will be looked for first in the current directory and second in all directories listed in `$MGD77_HOME/mgd77_paths.txt` [If `$MGD77_HOME` is not set it will default to `$GMTHOME/share/mgd77`].

NOTE: Under Windows, only (1) and (4) are currently supported.

**OPTIONS**

No space between the option flag and the associated arguments

**-A** Add a new data column. If an existing column with the same abbreviation already exists in the file we will cowardly refuse to update the file. Specifying **-A+** overcomes this reluctance (However, sometimes an existing column cannot be upgraded without first deleting it; if so you will be warned). Select a column source code among **a**, **c**, **d**, **D**, **e**, **g**, **i**, **n**, **t**, or **T**; detailed descriptions for each choice follow:

**a** Append filename of a single column table to add. File must have the same number of rows as the MGD77+ file. If no file is given we read from stdin instead.

**c** Create a new column that derives from existing data or formulas for corrections and reference fields. Append **c** for the Carter corrections subtracted from uncorrected depths, **g** for the IGF gravity reference field (a.k.a "normal gravity"), and **m** for the IGRF total field magnetic reference field. For gravity we choose the reference field based on the parameter Gravity Theoretical Formula Code in the cruise's MGD77 header. If this is not set or is invalid we default to the IGF 1980. You can override this behaviour by appending the desired code: 1 = Heiskanen 1924, 2 = International 1930, 3 = IGF1967, or 4 = IGF1980.

**d** Append filename of a two-column table with the first column holding distances along track and the second column holding data values. If no file is given we read from stdin instead. Records with matching distances in the MGD77+ file will be assigned the new values; at other distances we set them to NaN. Alternatively, give upper case **D** instead and we will interpolate the column at all record distances. See **-N** for choosing distance units and **-C** for choosing how distances are calculated.

**e** Expects to find an *e77* error/correction log from **mgd77sniffer** with the name *NGDC\_ID.e77* in the current directory or in  $\$MGD77\_HOME/E77$ ; this file will be examined and used to make modifications to the header values, specify a systematic correction for certain columns (such as scale and offset), and add or update the special column **flag** which may hold bitflags (0 = GOOD, 1 = BAD) for each data field in the standard MGD77 data set. Any fixed correction terms found (such as needing to scale a field by 0.1 or 10 because the source agency used incorrect units) will be written as attributes to the netCDF MGD77+ file and applied when the data are read by **mgd77list**. Ephemeral corrections such as those determined by crossover analysis are not kept in the data files but reside in correction tables (see **mgd77list** for details). By default, the first character of each line in the *e77* file (which is Y or N) will be consulted to see if the corresponding adjustment should be applied. You can override this behavior by appending one or more modifiers to the **-Ae** command: **H** will apply all header corrections regardless of Y/N, whereas **h** will ignore all header corrections, **F** will apply all fixed systematic trend corrections regardless of Y/N, whereas **f** will ignore them, **n**, **v**, **s**, and **g** will ignore bitflags pertaining to navigation, data values, data slopes, and grid comparisons, respectively. For the bitflags, **-Ae** means we will replace any existing bitflags in the file with the new values whereas **-A+e** means we will merge new flags and existing flags.

**g** Sample a GMT geographic (lon, lat) grid along the track given by the MGD77+ file using bicubic interpolation (however, see **-Q**). Append name of a GMT grid file.

**i** Sample a Sandwell/Smith Mercator \*.img grid along the track given by the MGD77+ file using bicubic interpolation (however, see **-Q**). Append the gridded data scale (typically 1 or 0.1), the IMG file mode (0-3), and finally the img grid filename. The modes stand for the following: (0) Img files with no constraint code, returns data at all points, (1) Img file with constraints coded, return data at all points, (2) Img file with constraints coded, return data only at constrained points and NaN elsewhere, and (3) Img file with constraints coded, return 1 at constraints and 0 elsewhere.

**n** Append filename of a two-column table with the first column holding the record number (0 to nrows - 1) and the second column holding data values. If no file is given we read from stdin instead. Records with matching record numbers in the MGD77+ file will be assigned the new values; at other records we set them to NaN.

**t** Append filename of a two-column table with the first column holding absolute times along track and the second column holding data values. If no file is given we read from stdin instead. Records with matching times in the MGD77+ file will be assigned the new values; at other times we set them to NaN. Alternatively, give upper case **T** instead and we will interpolate the column at all record times.

- C** Append a one-letter code to select the procedure for along-track distance calculation when using **-AdID** (see **-N** for selecting distance units):
  - f** Flat Earth distances.
  - g** Great circle distances [Default].
  - e** Geodesic distances on current GMT ellipsoid.
- D** Give a comma-separated list of column abbreviations that you want to delete from the MGD77+ files. Do NOT use this option to remove columns that you are replacing with new data (use **-A+** instead). Because we cannot remove variables from netCDF files we must create a new file without the columns to be deleted. Once the file is successfully created we temporarily rename the old file, change the new filename to the old filename, and finally remove the old, renamed file.
- E** Give a single character that will be repeated to fill empty string values, e.g., '9' will yield a string like "99999..." [9].
- I** In addition to file information we must specify additional information about the extra column. Specify a short (16 char or less, using letters, digits, or underscores only) abbreviation for the

selected data, its more descriptive name, the data unit, the data type 1-character code (**byte**, **short**, **float**, **int**, **double**, or **text**) you want used for storage in the netCDF file, any scale and offset we should apply to the data to make them fit inside the range implied by the chosen storage type, and a general comment (< 128 characters) regarding what these data represent. Note: If text data type is selected then the terms "values" in the **-A** discussion refer to your text data. Furthermore, the discussion on interpolation does not apply and the NaN value becomes a "no string" value (see **-E** for what this is). Place quotes around terms with more than one word (e.g., "Corrected Depth").

- N** Specify the distance unit used when using **-AdID** by appending **e** (meter), **k** (km), **m** (miles), or **n** (nautical miles). [Default is **-Nk** (km)].
- Q** Quick mode, use bilinear rather than bicubic interpolation [Default]. Optionally, append *value* in the  $0 \leq \text{value} \leq 1$  range. This parameter controls how close to nodes with NaN values the interpolation will go. E.g., a *value* of 0.5 will interpolate about 1/2-way from a non-NaN to a NaN node, whereas 0.1 will go about 90% of the way, etc. [Default is 1, which means none of the four nearby nodes may be NaN]. A *value* of 0 will just return the value of the nearest node instead of interpolating. Only relevant when **-Agli** is selected.
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- bi** Selects binary input. Append **s** for single precision [Default is **d** (double)]. Uppercase **S** (or **D**) will force byte-swapping. This applies to the input 1- or 2-column data files specified under some of the **-A** options. The binary input option is only available for numerical data columns.

## EXAMPLES

To append Geosat/ERS-1 gravity version 11.2 as an extra data column in the cruises 01010047.nc and 01010008.nc, storing the values as mGal\*10 in a 2-byte short integer, try

```
mgd77manage 01010047 01010008 -Ai10/1/grav.11.2.img -Isatgrav/"Geosat/ERS-1 grav-  
ity"/"mGal"/s/10/0/"Sandwell/Smith version 11.2" -V
```

To append a filtered version of magnetics as an extra data column of type float for the cruise 01010047.nc, and interpolate the filtered data at the times given in the MGD77+ file, try

```
mgd77manage 01010047 -ATmymag.tm -Ifiltmag/"Intermediate-wavelength magnetic residu-  
als"/"nTesla"/f/1/0/"Useful for looking for isochrons" -V
```

To delete the existing extra columns satfaa, coastdist, and satvgg from all MGD77+ files, try

```
mgd77manage `cat allmgd77.lis` -Dsatfaa,coastdist,satvgg -V
```

To create a 4-byte float column with the correct IGRF reference field in all MGD77+ files, try

```
mgd77manage `cat allmgd77.lis` -Acm -Iigrf/"IGRF reference field"/"nTesla"/f/1/0/"IGRF version 10 for  
1990-2010" -V
```

## DISCUSSION

### 1. Preamble

The mgd77 supplement is an attempt to (1) improve on the limited functionality of the existing mgg supplement, (2) incorporate some of the ideas from Scripps' gmt+ supplement by allowing extra data columns, and (3) add new capabilities for managing marine geophysical trackline data stored in an architecture-independent CF-1.0- and COARDS-compliant netCDF file format. Here are some of the underlying ideas and steps you need to take to maintain your files.

### 2. Introduction

Our starting point is the MGD77 ASCII data files distributed from NGDC on CD-ROMS, DVD-ROMS, and via FTP. Using Geodas to install the files locally we choose the "Carter corrected depth" option which

will fill in the depth column using the two-way traveltimes and the Carter tables if twt is present. This step yields ~5000 individual cruise files. Place these in one or more sub-directories of your choice, list these sub-directories (one per line) in the file `mgd77_paths.txt`, and place that file in the directory pointed to by `$MGD77_HOME`; if not set this variable defaults to `$GMTHOME/share/mgd77`.

### 3. Conversion

Convert the ASCII MGD77 files to the new netCDF MGD77+ format using `mgd77convert`. Typically, you will make a list of all the cruises to be converted (with or without extension), and you then run

```
mgd77convert -Fa -Tc -V -Lwe+ 'cat cruises.lis' > log.txt
```

The verbose settings will ensure that all problems found during conversion will be reported. The new \*.nc files may also be placed in one or more separate sub-directories and these should also be listed in the `mgd77_paths.txt` file. We suggest you place the directories with \*.nc files ahead of the \*.mgd77 directories. When you later want to limit a search to files of a certain extension you should use the `-I` option.

### 4. Adding new columns

`mgd77manage` will allow you to add additional data columns to your \*.nc files. These can be anything, including text strings, but most likely are numerical values sampled along the track from a supplied grid or an existing column that have been filtered or manipulated for a particular purpose. The format supports up to 32 such extra columns. See this man page for how to add columns. You may later decide to remove some of these columns or update the data associated with a certain column. Data extraction tools such as `mgd77list` can be used to extract a mix of standard MGD77 columns (navigation, time, and the usual geo-physical observations) and your custom columns.

### 5. Error sources

Before we discuss how to correct errors we will first list the different classes of errors associated with MGD77 data: (1) Header record errors occur when some of the information fields in the header do not comply with the MGD77 specification or required information is missing. `mgd77convert` will list these errors when the extended verbose setting is selected. These errors typically do not affect the data and are instead errors in the *meta-data* (2) Fixed systematic errors occur when a particular data column, despite the MGD77 specification, has been encoded incorrectly. This usually means the data will be off by a constant factor such as 10 or 0.1, or in some cases even 1.8288 which converts fathoms to meters. (3) Unknown systematic errors occur when the instrument that recorded the data or the processing that followed introduced signals that appear to be systematic functions of time along track, latitude, heading, or some other combination of terms that have a physical or logical explanation. These terms may sometimes be resolved by data analysis techniques such as along-track and across-track investigations, and will result in correction terms that when applied to the data will remove these unwanted signals in an optimal way. Because these correction terms may change when new data are considered in their determination, such corrections are considered to be ephemeral. (4) Individual data points or sequences of data may violate rules such as being outside of possible ranges or in other ways violate sanity. Furthermore, sequences of points that may be within valid ranges may give rise to data gradients that are unreasonable. The status of every point can therefore be determined and this gives rise to bitflags GOOD or BAD. Our policy is that error sources 1, 2, and 4 will be corrected by supplying the information as meta-data in the relevant \*.nc files, whereas the corrections for error source 3 (because they will constantly be improved) will be maintained in a separate list of corrections.

### 6. Finding errors

The `mgd77sniffer` is a tool that does a thorough along-track sanity check of the original MGD77 ASCII files and produces a corresponding \*.e77 error log. All problems found are encoded in the error log, and recommended fixed correction terms are given, if needed. An analyst may verify that the suggested corrections are indeed valid (we only want to correct truly obvious unit errors), edit these error logs and modify such correction terms and activate them by changing the relevant code key (see `mgd77sniffer` for more details). `mgd77manage` can ingest these error logs and (1) correct bad header records given the

suggestions in the log, (2) insert fixed correction terms to be used when reading certain columns, and (3) insert the bit-flags found. Rerun this step if you later find other problems as all flags will be recreated.

### 7. Error corrections

The extraction program **mgd77list** allows for corrections to be applied on-the-fly when data are requested. First, data with BAD bitflags are suppressed. Second, data with fixed systematic correction terms are corrected accordingly. Third, data with ephemeral correction terms will have those corrections applied (if a correction table is supplied). All of these steps require the presence of the relevant meta-data and all can be overruled by the user. In addition, users may add their own bitflags as separate data columns and use **mgd77list**'s logical tests to further dictate which data are suppressed from output.

### CREDITS

The IGRF calculations are based on a Fortran program written by Susan Macmillan, British Geological Survey, translated to C via f2c by Joaquim Luis, and adapted to GMT style by Paul Wessel.

### SEE ALSO

*mgd77convert(1), mgd77list(1), mgd77info(1), mgd77sniffer(1) mgd77track(1) x2sys\_init(1)*

### REFERENCES

- Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.
- Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.
- Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.
- The Marine Geophysical Data Exchange Format - "MGD77", see <http://www.ngdc.noaa.gov/mgg/dat/geodas/docs/mgd77.txt>
- IGRF, see <http://www.ngdc.noaa.gov/AGA/vmod/igrf.html>

**NAME**

`mgd77path` – Get full pathname for MGD77 files

**SYNOPSIS**

`mgd77path` *NGDC-ids* [ **-D** ] [ **-Iignore** ] [ **-V** ]

**DESCRIPTION**

`mgd77path` returns the full pathname to one or more MGD77 files. The pathname returned for a given cruise may change with time due to reshuffling of disks/subdirectories.

*NGDC-ids*

Can be one or more of five kinds of specifiers:

- 1) 8-character NGDC IDs, e.g., 01010083, JA010010etc., etc.
- 2) 2-character <agency> codes which will return all cruises from each agency.
- 3) 4-character <agency><vessel> codes, which will return all cruises from those vessels.
- 4) =<list>, where <list> is a table with NGDC IDs, one per line.
- 5) If nothing is specified we return all cruises in the data base.

(See `mgd77info -F` for agency and vessel codes). The ".mgd77" or ".nc" extensions will automatically be appended, if needed (use **-I** to ignore certain file types). Cruise files will be looked for first in the current directory and second in all directories listed in `$MGD77_HOME/mgd77_paths.txt` [If `$MGD77_HOME` is not set it will default to `$GMTHOME/share/mgd77`].

NOTE: Under Windows, only (1) and (4) are currently supported.

**OPTIONS**

No space between the option flag and the associated arguments.

- D** Instead of cruise listings, just show the directory paths currently used in the search.
- I** Ignore certain data file formats from consideration. Append `alcit` to ignore MGD77 ASCII, MGD77+ netCDF, or plain tab-separated ASCII table files, respectively. The option may be repeated to ignore more than one format. [Default ignores none].
- V** Report the total number of cruises found.

**EXAMPLES**

To obtain pathnames for cruises 01010008 and 01010007, run

```
mgd77path 01010008 01010007
```

To obtain pathnames for cruises 01010008 and 01010007, but only if there are MGD77+ version in netCDF, run

```
mgd77path 01010008 01010007 -Ia -It
```

To see the list of active directories where MGD77 files might be stored, run

```
mgd77path -D
```

**SEE ALSO**

`GMT(1)` `mgd77info(1)` `mgd77list(1)` `mgd77manage(1)` `mgd77track(1)`

**REFERENCES**

- Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.
- Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released,

[http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.  
Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.  
The Marine Geophysical Data Exchange Format - "MGD77", see <http://www.ngdc.noaa.gov/mgg/dat/geodas/docs/mgd77.txt>

**NAME**

mgd77sniffer – Scan for errors in MGD77 data

**SYNOPSIS**

**mgd77sniffer** *NGDC-ids* [ **-A***fieldabbrev, scale, offset* ] [ **-C***maxspd* ] [ **-D***fillm|sv* ] [ **-g***fieldabbrev, imggrid, scale, mode* ] [ **-G***fieldabbrev, grid* ] [ **-H***[i][nrec]* ] [ **-I***fieldabbrev, rec1, recN* ] [ **-N** ] [ **-L***custom limits file* ] [ **-Q***[value]* ] [ **-R***west/east/south/north[r]* ] [ **-S***l|t* ] [ **-T***gap* ] [ **-V** ] [ **-W***gl|sl|t|v|x* ] [ **-bo***[s|S|d|D][ncol]* ]

**DESCRIPTION**

**mgd77sniffer** scans old (pre-Y2K) and new format MGD77 files for errors using point-by-point sanity checking, along-track detection of excessive slopes, and optional comparison of cruise data with global gravity and predicted bathymetry grids. Detected data problems are output by default as verbose descriptions of each detected error, often resulting in multiple messages per scanned record. Data problems are optionally output (**-De** option) using a computer-parseable format (see E77 ERROR FORMAT description below). Default error thresholds are derived from histograms of all MGD77 geophysical data collected between 1952 and January, 2006. Thresholds are adjustable with the **-L** option.

*NGDC-ids*

Can be one or more of five kinds of specifiers:

- 1) 8-character NGDC IDs, e.g., 01010083, JA010010etc., etc.
- 2) 2-character <agency> codes which will return all cruises from each agency.
- 3) 4-character <agency><vessel> codes, which will return all cruises from those vessels.
- 4) =<list>, where <list> is a table with NGDC IDs, one per line.
- 5) If nothing is specified we return all cruises in the data base.

(See mgd77info **-F** for agency and vessel codes). The ".mgd77" or ".nc" extensions will automatically be appended, if needed (use **-I** to ignore certain file types). Cruise files will be looked for first in the current directory and second in all directories listed in **\$MGD77\_HOME/mgd77\_paths.txt** [If **\$MGD77\_HOME** is not set it will default to **\$GMTHOME/share/mgd77**].

NOTE: Under Windows, only (1) and (4) are currently supported.

**REQUIREMENTS**

The mgd77sniffer links with Generic Mapping Tools 4.0 or later along with the supplemental GMT packages x2sys and mgd77. See <http://gmt.soest.hawaii.edu> for GMT details. Grids for comparison with cruise data may be downloaded via the web. Among the many useful grids are NOAA's ETOPO2 global topography grid available from the National Geophysical Data Center's website as well as Sandwell and Smith's Raw Gravity grid available from Scripps Institution of Oceanography's website.

**OPTIONS**

- A** Apply scale factor and DC adjustment to specified data field. Allows adjustment of cruise data prior to along-track analysis. CAUTION: data must be thoroughly examined before applying these global data adjustments. May not be used for multiple cruises.
- C** Set maximum ship speed in m/s, or knots with **-N** option. Ship speeds exceeding 10 m/s (~20 knots) are flagged as excessive by default.
- D** Suppress default warning output and only dump cruise data row-by-row such as values, gradients, grid-cruise differences, E77 error summaries for each record, re-created MGD77 records or sniffer limits.

**-Dd** output differences between cruise and grid data. Requires **-G** option. Output columns include:

*lat lon dist cruiseZ gridZ diff [cruiseZ2 gridZ2 diff2 ...]*

Note: grid values are subtracted from cruise data so a positive difference implies cruise > grid. For multiple grid comparison, *cruiseZ gridZ diff* are repeated for each grid comparison in command line order.

**-De** output E77 error classification format. Error output is divided into (1) a header containing information globally applicable to the cruise and (2) individual error records summarizing all errors encountered in each cruise record. `mgd77sniffer` writes E77 directly to `<ngdc_id.e77>` file handle. See **E77 ERROR FORMAT** below for additional details.

**-Df** output delta Z (change in geophysical field) column and delta S (change in distance) for each geophysical field. Distance between observations often differ for different fields depending on instrument sampling rate, so `ds` is included for each geophysical observation. Output columns include:

*d[twt] ds d[depth] ds d[mtf1] ds d[mtf2] ds d[mag] ds d[diur] ds d[msd] ds d[gobs] ds d[eot] ds d[faa] ds*

**-DI** display `mgd77sniffer` limits. Customize this output to create a custom limits file for the **-L** option. No additional arguments are required. Output columns include:

*fieldabbrev min max maxSlope maxArea*

**-Dm** output MGD77 format records in Y2K-compliant MGD77 format

**-Ds** output calculated gradients for speed and geophysical fields. Gradients correspond to the gradient type selected in the **-S** option (spatial derivatives by default). Output columns include:

*speed d[twt] d[depth] d[mtf1] d[mtf2] d[mag] d[diur] d[msd] d[gobs] d[eot] d[faa]*

See **MGD77 FIELD INFO** below for field and abbreviations descriptions.

**-Dv** display values for the twelve position and geophysical fields for each MGD77 data record (in this order):

*lat lon twt depth mtf1 mtf2 mag diur msens gobs eot faa*

See below for **MGD77 FIELD INFO**.

- g** Compare cruise data to the specified grid in Sandwell/Smith Mercator format. Requires a valid MGD77 field abbreviation (see **MGD77 FIELD INFO** below) followed by a comma, the path (if not in current directory) and grid filename, a scale to multiply the data (1 or 0.1), and mode which stand for the following: (0) Img files with no constraint code, returns data at all points, (1) Img file with constraints coded, return data at all points, (2) Img file with constraints coded, return data only at constrained points and NaN elsewhere, and (3) Img file with constraints coded, return 1 at constraints and 0 elsewhere.
- G** Compare cruise data to the specified grid. Requires a valid MGD77 field abbreviation (see **MGD77 FIELD INFO** below) followed by a comma, then the path (if not in current directory) and grid filename. Multiple grid comparison is supported by using separate **-g** or **-G** calls for each grid. See **GRID FILE INFO** below.

Grid comparison activates several additional error checks. (1) Re-weighted Least Squares Regression of ship versus grid data determines slope and DC shift, which when differing from expected 1 and 0, respectively, may indicate incorrectly scaled ship data, including incorrect units or instrument drift as well as erroneous gravity tie-in. (2) Accumulated ship grid offsets are computed along-track and excessive offsets are flagged according to `maxArea` threshold (use **-L** option to adjust `maxArea`). Warning: predicted bathymetry grids are constrained by cruise data so grids and cruise data are not always independent. Comparison of cruise bathymetry with predicted bathymetry grids also activates a "navigation crossing over land" check.

- H** (with **-Glg** only) disable data decimation during RLS regression.
- I** Append a field abbreviation and the first and last record in a range of records that should be flagged as bad (and set to NaN prior to the analysis). Repeat as many times as needed. May not be used for multiple cruises.
- L** Override mgd77sniffer default error detection limits. Supply path and filename to the custom limits file. Rows not beginning with a valid MGD77 field abbreviation are ignored. Field abbreviations are listed below in exact form under MGD77 FIELD INFO. Multiple field limits may be modified using one default file, one field per line. Field min, max, max slope and max area may be changed for each field. Max slope pertains to the gradient type selected using the **-S** option. Max area is used by the **-G** option as the threshold for flagging excessive offsets from the specified grid. Dump defaults **-DI** to view syntax or to quickly create an editable custom limits file.

Example custom default file contents (see below for units):

```
# abbrev min max maxSlope maxArea
twt 0 15 1 0
depth 0 11000 500 5000
mag -800 800 - -
faa -300 300 100 2500
```

Use a dash '-' to retain a default limit. Hint: to test your custom limits, try: mgd77sniffer **-DI -L**<yourlimitsfile>

- N** Use nautical units.
- Q** Quick mode, use bilinear rather than bicubic interpolation [Default]. Optionally, append *value* in the  $0 \leq \text{value} \leq 1$  range. This parameter controls how close to nodes with NaN values the interpolation will go. E.g., a *value* of 0.5 will interpolate about 1/2-way from a non-NaN to a NaN node, whereas 0.1 will go about 90% of the way, etc. [Default is 1, which means none of the four nearby nodes may be NaN]. A *value* of 0 will just return the value of the nearest node instead of interpolating.
- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+ -]dd:mm[:ss.xxx][W|E|S|N] format. Append **r** if lower left and upper right map coordinates are given instead of wesn. The two shorthands **-Rg -Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).
- S** Specify gradient type for along-track excessive slope checking.
  - Sd** Calculate change in z values along track (dz). Output is given in geophysical units, e.g. mGal.
  - Ss** Calculate spatial gradients (dz/ds). Output is given in geophysical units per km along the survey track, e.g. mGal/km.
  - St** Calculate time gradients (dz/dt) [default]. Output is given in geophysical units per second along the survey track, e.g. mGal/sec.
- T** Adjusts mgd77sniffer gap handling. By default, data gaps greater than 5 km are skipped. Set to zero to de-activate gap skipping.
- W** Print out only certain warning types for verbose error messages. Comma delimit any combination of **cgloislvlx**: where (c) type code warnings, (g)radient out of range, (o)ffsets from grid (requires **-Glg**), (s)peed out of range, (t)ime warnings, (v)alue out of range, (x) warning summaries. By default ALL warning messages are printed. Not compatible with any **-D** options.
- V** runs in verbose mode.
- b** output binary data for **-Ddflslv** option. Append **s** for single and **d** for **double precision [Default is double]**.

#### MGD77 FIELD INFO

<i>Field</i>	<i>Abbreviation</i>	<i>Units</i>
Two-way Travel Time	twt	sec

Corrected Depth	depth	m
Mag Total Field1	mtf1	nT
Mag Total Field2	mtf2	nT
Residual Magnetic	mag	nT
Diurnal Correction	diur	nT
Mag Sensor Depth/Alt	msens	m
Observed Gravity	gobs	mGal
Eotvos Correction	eot	mGal
Free Air Anomaly	faa	mGal

## GRID FILE INFO

For **-g** the grids must be in the format used by Sandwell & Smith, which is a spherical Mercator 2-byte grid with no header. For **-G** the grid files can be of any grid type supported by GMT and therefore must contain a GMT header. A correctly formatted \*.i2 grid file can be generated using grdraster as shown below.

```
gmtset GRIDFILE_SHORTHAND TRUE
```

Create/edit .gmt\_io file to include the following rows:

```
# GMT I/O shorthand file
# suffix  format_id  scale  offset  NaN
grd      0      -      -      -
i2       2      -      -      32767
```

```
grdraster 1 -R0/359:55/-90/90 -Getopo5_hdr.i2
```

The new grid, etopo5\_hdr.i2 in this example, contains a GMT header and can be used in the **-G** option to compare cruise depth with grid values.

## E77 ERROR FORMAT

### Header

Information pertaining to an entire cruise, such as NGDC and survey institution identification codes, cruise examination time, two-way travel time corrector information, data precision warnings, as well as systematic scales, DC shifts and correlation coefficients from global grid comparisons are reported as E77 header information.

### Sample

```
# Cruise 06050010 ID L476WG MGD77 FILE VERSION: 19870415 N_RECS: 27268
# Examined: Tue Feb 21 16:33:34 2006
# Examiner: mtchandl
# Arguments: -De -Gfaa,/data/GRIDS/sandwell_smith11_hdr.i2
# Errata: Header
N-faa-E-01: Regression against grid gave scale 0.108643 which is statistically different from 1.0.
Recommended scale: (10.0)
N-faa-E-02: Regression against grid gave DC offset -1.319411 which is statistically different from
0.0
I-twt-W-03: More recent bathymetry correction table available
I-depth-W-04: Integer precision in depth
```

### Error Record

Individual error records have strict format. Included is a time or distance column followed by record number, a formatted error code string, and finally a verbose description of errors detected in the record. Four error classes are encoded into the error code string with different alphabetic characters representing unique error types. See below for error code format description.

### Format

<time/distance> <record number> <error code string> <description>

**Sample**

# Errata: Data  
 1971-04-28T18:27:30 345 0-0-OQ-0 gradient: mtf1, mag excessive  
 1971-05-14T18:01:30 5413 0-0-OQ-0 gradient: mtf1, mag excessive  
 1971-05-15T03:41:30 5634 0-Q-0-0 value: mag invalid

**Error Code Description**

Each of the four error classes is separated by a dash '-' and described by a combination of alphabetic characters or 0 signifying no detected problems.

Error classes: NAV-VAL-GRAD-GRID

**Error Class Descriptions**

NAV (navigation):

- 0 - fine
- A - time out of range
- B - time decreasing
- C - excessive speed
- D - above sea level
- E - lat undefined
- F - lon undefined

VAL (value):

- 0 - fine
- K - twt invalid
- L - depth invalid
- O - mtf1 invalid
- etc.

GRAD (gradient):

- 0 - fine
- K - d[twt] excessive
- L - d[depth] excessive
- O - d[mtf1] excessive
- etc.

GRID (grid comparison):

- 0 - fine
- L - depth offset
- W - faa offset
- etc.

The NAV error class has unique cases while VAL, GRAD and GRID classes are described by alphabetic characters for each of the 24 numeric fields in MGD77 format order.

MGD77 bit-pattern w/ E77 alpha characters

```
|-----|-----|
|XWVUTSRQPONMLKJIHGFE DCBA|E77 Code|
|-----|-----|
|nfe g m d m m m m b b d t p l l m h d m y t d | F I |
|q a o o s i s a t t t c e w t o a i o a o e z r l i D |
|c a t b d u e g f f c c p t c n t n u y n a t l e |
```



**NAME**

`mgd77track` – A shiptrack plotting program

**SYNOPSIS**

```
mgd77track NGDC-ids -Rwest/east/south/north[r] -Jparameters [ -A[c][size] ] [ -B[pls]parameters  ] [ -C[gle]  ] [ -D[startdate]  ] [ -D[stopdate]  ] [ -I[gnore]  ] [ -K  ] [ -L[trackticks]  ] [ -O  ] [ -P  ] [ -S[startdist][u]  ] [ -S[stopdist][u]  ] [ -T[t][dms,mc,mfs,mf,mfc]  ] [ -U[dx/dy][label]  ] [ -V  ] [ -W[pen]  ] [ -X[alclr][x-shift][u]  ] [ -X[alclr][y-shift][u]  ] [ -c[opies]  ]
```

**DESCRIPTION**

**mgd77track** reads NGDC MGD77 cruises and creates *PostScript* code that will plot one or more ship tracks on a map using the specified projection. The *PostScript* code is written to standard output.

*NGDC-ids*

Can be one or more of five kinds of specifiers:

- 1) 8-character NGDC IDs, e.g., 01010083, JA010010etc., etc.
- 2) 2-character <agency> codes which will return all cruises from each agency.
- 3) 4-character <agency><vessel> codes, which will return all cruises from those vessels.
- 4) =<list>, where <list> is a table with NGDC IDs, one per line.
- 5) If nothing is specified we return all cruises in the data base.

(See `mgd77info -F` for agency and vessel codes). The ".mgd77" or ".nc" extensions will automatically be appended, if needed (use **-I** to ignore certain file types). Cruise files will be looked for first in the current directory and second in all directories listed in `$MGD77_HOME/mgd77_paths.txt` [If `$MGD77_HOME` is not set it will default to `$GMTHOME/share/mgd77`].

NOTE: Under Windows, only (1) and (4) are currently supported.

- J** Selects the map projection. Scale is UNIT/degree, 1:xxxxx, or width in UNIT (upper case modifier). UNIT is cm, inch, or m, depending on the `MEASURE_UNIT` setting in `.gmtdefaults4`, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width value. For map height, max dimension, or min dimension, append **h**, **+**, or **-** to the width, respectively. More details can be found in the `psbasemap` man pages.

**CYLINDRICAL PROJECTIONS:**

- J***lon0/lat0/scale* (Cassini)
- J***jlon0/scale* (Miller)
- J***mscale* (Mercator - Greenwich and Equator as origin)
- J***mlon0/lat0/scale* (Mercator - Give meridian and standard parallel)
- J***oalon0/lat0/azimuth/scale* (Oblique Mercator - point and azimuth)
- J***oblon0/lat0/lon1/lat1/scale* (Oblique Mercator - two points)
- J***oclon0/lat0/lonp/latp/scale* (Oblique Mercator - point and pole)
- J***qlon0/scale* (Equidistant Cylindrical Projection (Plate Carree))
- J***tlon0/scale* (TM - Transverse Mercator, with Equator as  $y = 0$ )
- J***tlon0/lat0/scale* (TM - Transverse Mercator, set origin)
- J***uzone/scale* (UTM - Universal Transverse Mercator)
- J***ylon0/lats/scale* (Basic Cylindrical Projection)

**AZIMUTHAL PROJECTIONS:**

- J***alon0/lat0/scale* (Lambert)
- J***elon0/lat0/scale* (Equidistant)
- J***flon0/lat0/horizon/scale* (Gnomonic)
- J***glon0/lat0/scale* (Orthographic)
- J***slon0/lat0/[slat]/scale* (General Stereographic)

**CONIC PROJECTIONS:**

- Jb***lon0/lat0/lat1/lat2/scale* (Albers)
- Jd***lon0/lat0/lat1/lat2/scale* (Equidistant)
- Jl***lon0/lat0/lat1/lat2/scale* (Lambert)

#### MISCELLANEOUS PROJECTIONS:

- Jh***lon0/scale* (Hammer)
- Ji***lon0/scale* (Sinusoidal)
- Jk**[**fls**]*lon0/scale* (Eckert IV (f) and VI (s))
- Jn***lon0/scale* (Robinson)
- Jr***lon0/scale* (Winkel Tripel)
- Jv***lon0/scale* (Van der Grinten)
- Jw***lon0/scale* (Mollweide)

#### NON-GEOGRAPHICAL PROJECTIONS:

- Jp**[**a**]*scale[/origin][r|z]* (Polar coordinates (theta,r))
- Jxx**-*scale*[**dllppow**l**t**l**T**][*y-scale*[**dllppow**l**t**l**T**]] (Linear, log, and power scaling)
- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*-dd:mm[:ss.xxx]*[**W**|**E**|**S**|**N**] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).

#### OPTIONS

No space between the option flag and the associated arguments.

- A** Annotate each leg every time it enters the plot. Append **c** to annotate using the MGD77 cruise ID [Default uses the filename prefix]. Optional *size* is the font size in points. The leg annotation font is controlled by **LABEL\_FONT**.
- B** Sets map boundary annotation and tickmark intervals; see the **psbasemap** man page for all the details.
- C** Select procedure for along-track distance calculation:
  - f** Flat Earth distances.
  - g** Great circle distances [Default].
  - e** Geodesic distances on current GMT ellipsoid.
- Da** Do not plot data collected before *startdate* (yyyy-mm-dd**T**[hh:mm:ss]) [Default is first day].
- Db** Do not plot data collected after *stopdate* (yyyy-mm-dd**T**[hh:mm:ss]). [Default is last day].
- I** Ignore certain data file formats from consideration. Append **alc**l**t** to ignore MGD77 ASCII, MGD77+ netCDF, or plain table files, respectively. The option may be repeated to ignore more than one format. [Default ignores none].
- K** More *PostScript* code will be appended later [Default terminates the plot system].
- L** To put time/distance log-marks on the track. E.g. **a500ka24ht6h** means (**a**)nnotate every 500 km (**k**) and 24 **h**(ours), with (**t**)ickmarks every 500 km and 6 hours. Alternatively you may use the modifiers **d** (days) and **n** (nautical miles).
- O** Selects Overlay plot mode [Default initializes a new plot system].
- P** Selects Portrait plotting mode [GMT Default is Landscape, see **gmtdefaults** to change this].
- Sa** Do not plot data that are less than *startdist* meter along track from port of departure. Append **k** for km, **m** for miles, or **n** for nautical miles [Default is 0 meters].
- Sb** Do not plot data that are more than *stopdist* meter along track from port of departure. Append **k** for km, **m** for miles, or **n** for nautical miles [Default is end of track].

- T      Controls the attributes of the three kinds of markers (**T** for the first time marker in a new day, **t** for additional time markers in the same day, and **d** for distance markers). For each of these you can specify the 5 comma-separated attributes *markersize*, *markercolor*, *markerfontsize*, *markerfont*, and *markerfontcolor*. Repeat the **-T** option for each marker type.
- U      Draw Unix System time stamp on plot. User may specify where the lower left corner of the stamp should fall on the page relative to lower left corner of plot. Optionally, append a label, or **c** (which will plot the command string.). The **GMT** parameters **UNIX\_TIME** and **UNIX\_TIME\_POS** can affect the appearance; see the **gmtdefaults** man page for details.
- W      Append *pen* used for the trackline. [Default is 0.25p,black]. [Default is solid].
- X -Y   Shift origin of plot by (*x-shift,y-shift*), and optionally append units (**c**, **i**, **m**, **p**). Prepend **a** for absolute coordinates; the default (**r**) will reset plot origin. Give **c** to center plot using current page size.
- V      Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- c      Specifies the number of plot copies. [Default is 1].

### EXAMPLES

To generate a Mercator plot of the track of the cruise 01010007 in the area 70W to 20E, 40S to 20N, using a Mercator scale of 0.1inch/degree, label the tracks with 10 points characters, annotate the boundaries every 10 degrees, draw gridlines every 5 degrees, and mark the track every day and 1000 km, with ticks every 6 hours and 250 km, and send the plot to the default printer, enter the following command:

```
mgd77track 01010007 -R70W/20E/40S/20N -Jm0.1 -B10g5 -A10 -La1da1000kf6hf250k | lpr
```

### SEE ALSO

*mgd77info(1)*, *psbasemap(1)* *mgd77list(1)*

### REFERENCES

- Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.
- Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.
- Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.
- The Marine Geophysical Data Exchange Format - "MGD77", see <http://www.ngdc.noaa.gov/mgg/dat/geodas/docs/mgd77.txt>

**NAME**

binlegs – Maintain the **GMT** index files for mgg supplement

**SYNOPSIS**

**binlegs** *leglistfile* [ **-R** ] [ **-D** ] [ **-P***path* ] [ **-V** ]

**DESCRIPTION**

**binlegs** will make changes to the **mgg** supplement data base files *gmt\_legs.d* and *gmt\_index.b* for the legs listed in the *leglistfile*. Normally, only the data archivist will need to be concerned with these operations.

*leglistfile*

Contains the list of legs to be worked on.

**OPTIONS**

No space between the option flag and the associated arguments

- R** Replace the current information if a leg already is in the system. [Default is append].
- D** Delete information for these legs from the system index files.
- P** Set the path to where the *gmt\_legs.d* and *gmt\_index.b* files reside. [Default is the **\$GMTHOME/lib.mgg** directory].
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].

**EXAMPLES**

To add new cruises to the system:

```
binlegs newlegs.lis -V -P/usr/lib/gmt
```

To remove bad legs from the system:

```
binlegs badlegs.lis -D -V -P/usr/lib/gmt
```

**SEE ALSO**

*GMT*(1), *gmtlegs*(1), *gmt2bin*(1)

**REFERENCES**

- Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.
- Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.
- Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

**NAME**

`dat2gmt` – Convert an ASCII file to a binary gmt file

**SYNOPSIS**

`dat2gmt gmtfile`

**DESCRIPTION**

`dat2gmt` reads an ASCII version of a gmt file (made by `gmt2dat`) from standard input and creates a binary gmt file.

**EXAMPLES**

To create a gmt file from the ASCII file `c2104_fixed.ascii`, run

```
dat2gmt c2104.gmt < c2104_fixed.ascii
```

**SEE ALSO**

`gmt2dat(1)`, `GMT(1)`

**REFERENCES**

Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.

Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.

Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

**NAME**

gmt2bin – Create bin-index files from gmt files

**SYNOPSIS**

**gmt2bin** *listfile* [ **-P***path* ] [ **-V** ]

**DESCRIPTION**

**gmt2bin** reads a file with a list of leg ids, opens each gmt file, and creates a bin-index file for each leg called <legid>.bix. These are used by **binlegs** to update the system files read by **gmtlegs**. Normally, only the data archivist will need to be concerned with these operations.

*listfile* This is a one column ASCII file with one leg id pr line.

**OPTIONS**

- P** Sets full path to directory with gmtfiles. [Default is current directory].
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].

**SEE ALSO**

*binlegs*(1), *dat2gmt*(1), *GMT*(1)

**REFERENCES**

- Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.
- Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.
- Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

**NAME**

gmt2dat – Convert a binary gmt file to an ASCII file

**SYNOPSIS**

**gmt2dat** *gmtfile*

**DESCRIPTION**

**gmt2dat** reads a <legid>.gmt file and converts it to a regular ASCII text file that can be edited using the text editor. The ASCII text is sent to standard output and can be redirected to a file/program/printer, etc.

**EXAMPLES**

To convert c2104.gmt to ASCII:

```
gmt2dat c2104.gmt > c2104.ascii
```

**SEE ALSO**

*dat2gmt(1)*, *GMT(1)*

**REFERENCES**

Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.

Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.

Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

**NAME**

`gmtinfo` – Get information about individual cruises

**SYNOPSIS**

`gmtinfo` *leg-ids*

**DESCRIPTION**

`gmtinfo` reports the minimum/maximum coordinates, year of the cruise, leg id, number of records, the number of gravity, magnetics, and bathymetry points found, and the start/stop dates for the leg. All the information is written to standard output on one line so that it is easy to use `awk` to add up numbers etc.

*leg-ids* Can be one or more `gmtleg`-names, like `c2610`. Use ‘`cat list_of_legs`’ to pass all the legnames in the file `list_of_legs`.

**EXAMPLES**

To obtain information about cruises `c2603` and `v2819`, run

```
gmtinfo c2603 v2819 > info_file
```

**SEE ALSO**

*GMT*(1)

**REFERENCES**

Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.

Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.

Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

**NAME**

gmtlegs – Find cruises in a given region

**SYNOPSIS**

**gmtlegs** **-R***west/east/south/north***[r]** **[-G]** **[-M]** **[-T]** **[-L]** **[-V]**

**DESCRIPTION**

**gmtlegs** will report the name of all the cruises that happened to be wholly or partially inside the specified region. As an option, only those cruises that collected a certain type of data can be reported.

**-R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd:mm[:ss.xxx]***[W|E|S|N]** format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).

**OPTIONS**

No space between the option flag and the associated arguments

**-G** Report cruises that collected Gravity data. [Default is any data].

**-M** Report cruises that collected Magnetics. [Default is any data].

**-T** Report cruises that collected Topography. [Default is any data].

**-L** Long output, i.e. cruisenames and data types available. [Default is cruisenames only].

**-V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].

**EXAMPLES**

To find all cruises around Hawaii that collected gravity and topography:

```
gmtlegs -R-162/-152/18/25 -G -T > hawaii_gt.legs
```

To find all cruises with magnetics:

```
gmtlegs -R0/360/-90/90 -M > all_m.legs
```

**SEE ALSO**

*GMT*(1), *gmt2bin*(1), *binlegs*(1)

**REFERENCES**

Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.

Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.

Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

**NAME**

gmtlist – A data-extractor for <legid>.gmt files

**SYNOPSIS**

```
gmtlist leg-ids [ -Ccorrectionfile ] [ -Dstartdate ] [ -Dbstopdate ] [ -Fflags ] [ -G ] [ -Hi[nrec] ] [ -Rwest/east/south/north[r] ] [ -Sstartdist ] [ -Sbstopdist ] [ -V ] [ -Wweight ] [ -bo[s|S|d|D][ncol] ]
```

**DESCRIPTION**

**gmtlist** reads <legid>.gmt files and produces an ASCII [or binary] table. The <legid>.gmt files contain time(**s**), latitude(**y**), longitude(**x**), gravity(**g**/**G**), magnetics(**m**/**M**), and bathymetry(**t**/**T**), and the user may extract any combination of these 6 parameters + distance(**d**), heading (**h**), velocity (**v**), and weight (**w**, see **-W**). A sub-section can be specified by passing time- or distance-intervals along track or by selecting a region.

*leg-ids* Can be one or more cruisenames. To give a list of names, use 'cat list\_of\_legs'.

**OPTIONS**

No space between the option flag and the associated arguments

- C** Apply crossover Corrections to the data. If no correction file is given, gmtlist reads the default correction file. (See **XSYSTEM** for how to set up your own correction file).
- Da** Do not list data collected before *startdate* (mm/dd/yyyy/hh:mm) [Default is first day].
- Db** Do not list data collected after *stopdate* (mm/dd/yyyy/hh:mm). [Default is last day].
- F** Can be any combination of **syxgmtdhv** to indicate the desired output data. The data will appear in the order indicated by flags. If **G**, **M**, or **T** is substituted for **g**, **m**, **t**, only records that have that combination of data are written out. If **s** is followed by **c** (calendar), then yyyy/mm/dd/hh/mm/ss is written out, if **s** is followed by **j** (julian), then yyyy/jd/hh/mm/ss is written out. Default is seconds from start of year.
- G** Force Geographical longitudes (-180/+180) rather than geodetic (0-360) [Default].
- H** Issue a header record with names for each data field.
- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd*:*mm*[*:ss.xxx*][*W*|*E*|*S*|*N*] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).
- Sa** Do not list data that are less than *startdist* km along track from port of departure. [Default is 0].
- Sb** Do not list data that are more than *stopdist* km along track from port of departure. [Default is length of track].
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- W** Set the weight for these data. Weight output option must be set in **-F**. This is useful if the data are to be processed with the weighted averaging techniques offered by **blockmean**, **blockmedian**, and **blockmode**.
- b** Selects binary output mode (in double precision) [Default is ASCII]. **-H** is ignored if **-b** is selected. Julian and calendar dates are not supported for binary time output (i.e., you get seconds from start of year).

**EXAMPLES**

To get a (distance, heading, crossover-corrected gravity, bathymetry) listing from c2104.gmt, starting at June 3 1971 20:45 and ending at distance = 5000 km, use the following command:

```
gmtlist c2104 -Da6/3/1971/20:45 -Sb5000 -Fdhgt -C > myfile
```

To make input for **blockmean** and **surface** using all the cruises listed in the file gmt.list, but only the data that are inside the specified area, and make output binary:

```
gmtlist 'cat gmt.list' -Fxyg -R-40/-30/25/35 -b > allgrav.xyg
```

**SEE ALSO**

*GMT(1)*, *x\_system(1)*

**REFERENCES**

Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.

Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.

Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

**NAME**

`gmtpath` – Get full pathname for gmt files

**SYNOPSIS**

`gmtpath` *leg-ids*

**DESCRIPTION**

`gmtpath` returns the full pathname to one or more gmt-files. The pathname returned for a given cruise may change with time due to reshuffling of disks/subdirectories.

*leg-ids* Can be one or more gmtleg-names, like c2610. Use ‘cat list\_of\_legs’ to pass all the legnames in the file list\_of\_legs.

**EXAMPLES**

To obtain pathnames for cruises c2603 and v2819, run

```
gmtpath c2603 v2819
```

**SEE ALSO**

*GMT*(1)

**REFERENCES**

Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.

Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.

Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

**NAME**

gmttrack – A shiptrack plotting program

**SYNOPSIS**

**gmttrack** *leg-ids* **-R***west/east/south/north***[r]** **-J***parameters* [ **-B****[pls]***parameters* ] [ **-M***trackticks* ] [ **-W***pen***[red/green/blue]****[OA]** ] [ **-A***size* ] [ **-K** ] [ **-O** ] [ **-P** ] [ **-U****[dx/dy]****[label]** ] [ **-V** ] [ **-X****[alclr]****[x-shift****[u]]** ] [ **-X****[alclr]****[y-shift****[u]]** ] [ **-ccopies** ]

**DESCRIPTION**

**gmttrack** reads gmt cruises and creates *PostScript* code that will plot one or more ship tracks on a map using the specified projection. The *PostScript* code is written to standard output.

*leg-ids* Can be one or more gmtleg-names, like c2104 v3206 etc.

**-J** Selects the map projection. Scale is UNIT/degree, 1:xxxxx, or width in UNIT (upper case modifier). UNIT is cm, inch, or m, depending on the **MEASURE\_UNIT** setting in .gmtdefaults4, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width value. For map height, max dimension, or min dimension, append **h**, **+**, or **-** to the width, respectively. More details can be found in the **psbasemap** man pages.

**CYLINDRICAL PROJECTIONS:**

- Jclon0/lat0/scale** (Cassini)
- Jjlon0/scale** (Miller)
- Jmscale** (Mercator - Greenwich and Equator as origin)
- Jmlon0/lat0/scale** (Mercator - Give meridian and standard parallel)
- Joalon0/lat0/azimuth/scale** (Oblique Mercator - point and azimuth)
- Joblon0/lat0/lon1/lat1/scale** (Oblique Mercator - two points)
- Joclon0/lat0/lonp/latp/scale** (Oblique Mercator - point and pole)
- Jqlon0/scale** (Equidistant Cylindrical Projection (Plate Carree))
- Jtlon0/scale** (TM - Transverse Mercator, with Equator as  $y = 0$ )
- Jtlon0/lat0/scale** (TM - Transverse Mercator, set origin)
- Juzone/scale** (UTM - Universal Transverse Mercator)
- Jylon0/lats/scale** (Basic Cylindrical Projection)

**AZIMUTHAL PROJECTIONS:**

- Jalon0/lat0/scale** (Lambert)
- Jelon0/lat0/scale** (Equidistant)
- Jflon0/lat0/horizon/scale** (Gnomonic)
- Jglon0/lat0/scale** (Orthographic)
- Jslon0/lat0/[slat]/scale** (General Stereographic)

**CONIC PROJECTIONS:**

- Jblon0/lat0/lat1/lat2/scale** (Albers)
- Jdlon0/lat0/lat1/lat2/scale** (Equidistant)
- Jllon0/lat0/lat1/lat2/scale** (Lambert)

**MISCELLANEOUS PROJECTIONS:**

- Jhlon0/scale** (Hammer)
- Jilon0/scale** (Sinusoidal)
- Jk[fls]lon0/scale** (Eckert IV (f) and VI (s))
- Jnlon0/scale** (Robinson)
- Jrlnon0/scale** (Winkel Tripel)
- Jvlon0/scale** (Van der Grinten)

**-Jwlon0/scale** (Mollweide)

### NON-GEOGRAPHICAL PROJECTIONS:

**-Jp[a]scale[/origin][r|z]** (Polar coordinates (theta,r))

**-Jxx-scale[dlllppow|t|T][y-scale[dlllppow|t|T]]** (Linear, log, and power scaling)

- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd*:*mm*[:*ss.xxx*][*W|E|S|N*] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).

### OPTIONS

No space between the option flag and the associated arguments.

- B** Sets map boundary annotation and tickmark intervals; see the **psbasemap** man page for all the details.
- A** Annotate each leg every time it enters the plot. Optional *size* is character size in points.
- Da** Do not plot data collected before *startdate* (mm/dd/yyyy/hh:mm) [Default is first day].
- Db** Do not plot data collected after *stopdate* (mm/dd/yyyy/hh:mm). [Default is last day].
- K** More *PostScript* code will be appended later [Default terminates the plot system].
- M** To put time/distance Marks on the track. E.g. **a500ka24ht6h** means (**a**)nnotate every 500 km (**k**) and 24 **h**(ours), with (**t**)ickmarks every 500 km and 6 hours.
- O** Selects Overlay plot mode [Default initializes a new plot system].
- P** Selects Portrait plotting mode [GMT Default is Landscape, see **gmtdefaults** to change this].
- Sa** Do not plot data that are less than *startdist* km along track from port of departure. [Default is 0].
- Sb** Do not plot data that are more than *stopdist* km along track from port of departure. [Default is length of track].
- U** Draw Unix System time stamp on plot. User may specify where the lower left corner of the stamp should fall on the page relative to lower left corner of plot. Optionally, append a label, or **c** (which will plot the command string.). The GMT parameters **UNIX\_TIME** and **UNIX\_TIME\_POS** can affect the appearance; see the **gmtdefaults** man page for details.
- W** *pen* is thickness of the trackline. [Default is 1]. Optionally, specify the rgb combination to obtain a colored trackline [Default is black]. Append **o** for dotted line, **a** for dashed. [Default is solid].
- X -Y** Shift origin of plot by (*x-shift,y-shift*), and optionally append units (**c**, **i**, **m**, **p**). Prepend **a** for absolute coordinates; the default (**r**) will reset plot origin. Give **c** to center plot using current page size.
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- c** Specifies the number of plot copies. [Default is 1].

### EXAMPLES

To generate a Mercator plot of the track of cruises C2610 and V1512 in the area 150E to 154E, 18N to 23N, using a Mercator scale of 1.5inch/degree, label the tracks with 10 points characters, annotate the boundaries every degree, and draw gridlines every 30 minutes, and send the plot to the default printer, enter the following command:

```
gmttrack c2610 v1512 -R150/154/18/23 -Jm1.5 -B1g30m -A10 | lpr
```

### SEE ALSO

*GMT*(1), *psbasemap*(1)

**REFERENCES**

- Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.
- Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.
- Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.
- Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

**NAME**

`mgd77togmt` – Convert an MGD-77 ASCII file to a binary gmt file

**SYNOPSIS**

`mgd77togmt` [ **-Llist** ] [ *mgd77file* ] **-Flegid** **-Ystart\_year** [ **-Ainformation** ] [ **-Itime\_increment** ] [ **-V** ]

**DESCRIPTION**

**mgd77togmt** reads an ASCII MGD-77 format file [or standard input] and creates a binary gmt file. The 16 header records specified in the MGD-77 documentation may or may not be present. If two-way travel times are available, **mgd77togmt** will convert those to corrected depths using the Carter tables. Both old and new (Y2K-compliant) MGD-77 files are supported.

**-L** Name of file with several records of the format *mgd77file legid start\_year*. For multiple files, use this option rather than the **-F -Y** combination.

**-F** Leg id that will be used in file name (*legid.gmt*).

**-Y** The year of the first data point in the file.

**OPTIONS**

No space between the option flag and the associated arguments.

**-A** Set an optional 10 character information string to be stored in the header [Default is blank].

**-I** Used for files where time is missing. The increment is used to calculate fake times.

**-V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].

**EXAMPLES**

To create a gmt file from the ASCII file *c2104.mgd77*, run

```
mgd77togmt c2104.mgd77 -Y1977 -Fc2104
```

To create new gmt files from all the *mgd77* files listed in *list.d*:

```
mgd77togmt -Llist.d -V
```

**SEE ALSO**

*dat2gmt*(1), *GMT*(1)

**REFERENCES**

Wessel, P., and W. H. F. Smith, 2005, The Generic Mapping Tools (GMT) version 4.1 Technical Reference & Cookbook, SOEST/NOAA.

Wessel, P., and W. H. F. Smith, 1998, New, Improved Version of Generic Mapping Tools Released, EOS Trans., AGU, 79 (47), p. 579.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, EOS Trans., AGU, 76 (33), p. 329.

Wessel, P., and W. H. F. Smith, 1995, New Version of the Generic Mapping Tools Released, [http://www.agu.org/eos\\_elec/95154e.html](http://www.agu.org/eos_elec/95154e.html), Copyright 1995 by the American Geophysical Union.

Wessel, P., and W. H. F. Smith, 1991, Free Software Helps Map and Display Data, EOS Trans., AGU, 72 (41), p. 441.

MGD77 Task Group, 1977, The Marine Geophysical Data Exchange Format - MGD77, Key to Geophysical Records documentation No. 10, National Geophysical and Solar-Terrestrial Data Center, Boulder, CO.

Carter, D. J. T., *Echo Sounding Correction Tables: Formerly Matthews' Tables*, 150 pp, Hydrographic Department, Ministry of Defense, Taunton, Somerset, England, 1980.

**NAME**

gmtdigitize – Digitizing and Inverse map transformation of map x/y coordinates

**SYNOPSIS**

**gmtdigitize** *-Jparameters* *-Rwest/east/south/north[r]* [ *-A* ] [ *-Cdevice* ] [ *-Dlimit* ] [ *-F* ] [ *-H[i][nrec]* ] [ *-Llpi* ] [ *-M[flag]* ] [ *-Nnamestem* ] [ *-S* ] [ *-V* ] [ *-Zk|v* ] [ *-bo[s|Std|D][ncol]* ] [ *>* output.d ]

**DESCRIPTION**

**gmtdigitize** digitizes points from a digitizer via a serial line connection and computes map coordinates using the specified map projection. The program is interactive and will take you through the setup procedure and how you will digitize points. The program will determine the actual map scale as well as rotation of the paper that is taped to the digitizer table. By default the output will go to stdout.

No space between the option flag and the associated arguments. Use upper case for the option flags and lower case for modifiers.

**-J** Selects the map projection. The following character determines the projection. If the character is upper case then the argument(s) supplied as scale(s) is interpreted to be the map width (or axis lengths), else the scale argument(s) is the map scale (see its definition for each projection). UNIT is cm, inch, or m, depending on the **MEASURE\_UNIT** setting in .gmtdefaults4, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width values. Append **h**, **+**, or **-** to the given *width* if you instead want to set map height, the maximum dimension, or the minimum dimension, respectively [Default is **w** for width].

The ellipsoid used in the map projections is user-definable by editing the .gmtdefaults4 file in your home directory. 63 commonly used ellipsoids and a spheroid are currently supported, and users may also specify their own ellipsoid parameters (see man gmtdefaults for more details). **GMT** default is WGS-84. Several **GMT** parameters can affect the projection: **ELLIPSOID**, **INTERPOLANT**, **MAP\_SCALE\_FACTOR**, and **MEASURE\_UNIT**; see the **gmtdefaults** man page for details.

Choose one of the following projections (The **E** or **C** after projection names stands for Equal-Area and Conformal, respectively):

**CYLINDRICAL PROJECTIONS:**

**-Jclon0/lat0/scale** or **-JClon0/lat0/width** (Cassini).

Give projection center and scale (1:xxxx or UNIT/degree).

**-Jclon0/lat0/scale** or **-JClon0/lat0/width** (Cassini).

Give projection center and scale (1:xxxx or UNIT/degree).

**-Jjlon0/scale** or **-JJlon0/width** (Miller Cylindrical Projection).

Give the central meridian and scale (1:xxxx or UNIT/degree).

**-Jmparameters** (Mercator [**C**]).

Specify one of:

**-Jmscale** or **-JMwidth**

Give scale along equator (1:xxxx or UNIT/degree).

**-Jmlon0/lat0/scale** or **-JMlon0/lat0/width**

Give central meridian, standard latitude and scale along parallel (1:xxxx or UNIT/degree).

**-Joparameters** (Oblique Mercator [**C**]).

Specify one of:

**-Joalon0/lat0/azimuth/scale** or **-JOalon0/lat0/azimuth/width**

Set projection center, azimuth of oblique equator, and scale.

**-Joblon0/lat0/lon1/lat1/scale** or **-JOBlon0/lat0/lon1/lat1/scale**

Set projection center, another point on the oblique equator, and scale.

**-Joclon0/lat0/lonp/latp/scale** or **-JOclon0/lat0/lonp/latp/scale**

Set projection center, pole of oblique projection, and scale.

Give scale along oblique equator (1:xxxx or UNIT/degree).

**-Jqlon0/scale** or **-JQlon0/width** (Equidistant Cylindrical Projection (Plate Carree)).

Give the central meridian and scale (1:xxxx or UNIT/degree).

**-Jtparameters** (Transverse Mercator [C]).

Specify one of:

**-Jtlon0/scale** or **-JTlon0/width**

Give the central meridian and scale (1:xxxx or UNIT/degree).

**-Jtlon0/lat0/scale** or **-JTlon0/lat0/width**

Give projection center and scale (1:xxxx or UNIT/degree).

**-Juzone/scale** or **-JUzone/width** (UTM - Universal Transverse Mercator [C]).

Give the zone number (1-60) and scale (1:xxxx or UNIT/degree).

Zones: prepend - or + to enforce southern or northern hemisphere conventions [northern if south > 0].

**-Jylon0/lats/scale** or **-JYlon0/lats/width** (Basic Cylindrical Projections [E]).

Give the central meridian, standard parallel, and scale (1:xxxx or UNIT/degree). The standard parallel is typically one of these (but can be any value):

45 - The Peters projection

37.4 - The Trystan Edwards projection

30 - The Behrman projection

0 - The Lambert projection

#### AZIMUTHAL PROJECTIONS:

Except for polar aspects, **-Rw/e/s/n** will be reset to **-Rg**. Use **-R<...>r** for smaller regions.

**-Jalon0/lat0/scale** or **-JAlon0/lat0/width** (Lambert [E]).

*lon0/lat0* specifies the projection center. Give scale as 1:xxxx or *radius/lat*, where *radius* is distance in UNIT from origin to the oblique latitude *lat*.

**-Jelon0/lat0/scale** or **-JElon0/lat0/width** (Equidistant).

*lon0/lat0* specifies the projection center. Give scale as 1:xxxx or *radius/lat*, where *radius* is distance in UNIT from origin to the oblique latitude *lat*.

**-Jflon0/lat0/horizon/scale** or **-JFlon0/lat0/horizon/width** (Gnomonic).

*lon0/lat0* specifies the projection center. *horizon* specifies the max distance from projection center (in degrees, < 90). Give scale as 1:xxxx or *radius/lat*, where *radius* is distance in UNIT from origin to the oblique latitude *lat*.

**-Jglon0/lat0/scale** or **-JGlon0/lat0/width** (Orthographic).

*lon0/lat0* specifies the projection center. Give scale as 1:xxxx or *radius/lat*, where *radius* is distance in UNIT from origin to the oblique latitude *lat*.

**-Jslon0/lat0/scale** or **-JSlon0/lat0/width** (General Stereographic [C]).

*lon0/lat0* specifies the projection center. Give scale as 1:xxxx (true at pole) or *slat/1:xxxx* (true at standard parallel *slat*) or *radius/lat* (*radius* in UNIT from origin to the oblique latitude *lat*).

#### CONIC PROJECTIONS:

**-Jblon0/lat0/lat1/lat2/scale** or **-JBlon0/lat0/lat1/lat2/width** (Albers [E]).

Give projection center, two standard parallels, and scale (1:xxxx or UNIT/degree).

**-Jdlon0/lat0/lat1/lat2/scale** or **-JDlon0/lat0/lat1/lat2/width** (Equidistant)

Give projection center, two standard parallels, and scale (1:xxxx or UNIT/degree).

**-Jl***lon0/lat0/lat1/lat2/scale* or **-JL***lon0/lat0/lat1/lat2/width* (Lambert [C])  
Give origin, 2 standard parallels, and scale along these (1:xxxx or UNIT/degree).

#### MISCELLANEOUS PROJECTIONS:

**-Jh***lon0/scale* or **-JH***lon0/width* (Hammer [E]).  
Give the central meridian and scale along equator (1:xxxx or UNIT/degree).

**-Ji***lon0/scale* or **-JI***lon0/width* (Sinusoidal [E]).  
Give the central meridian and scale along equator (1:xxxx or UNIT/degree).

**-Jk***[f|s]lon0/scale* or **-JK***[f|s]lon0/width* (Eckert IV (f) and VI (s) [E]).  
Give the central meridian and scale along equator (1:xxxx or UNIT/degree).

**-Jn***lon0/scale* or **-JN***lon0/width* (Robinson).  
Give the central meridian and scale along equator (1:xxxx or UNIT/degree).

**-Jr***lon0/scale* **-JR***lon0/width* (Winkel Tripel).  
Give the central meridian and scale along equator (1:xxxx or UNIT/degree).

**-Jv***lon0/scale* or **-JV***lon0/width* (Van der Grinten).  
Give the central meridian and scale along equator (1:xxxx or UNIT/degree).

**-Jw***lon0/scale* or **-JW***lon0/width* (Mollweide [E]).  
Give the central meridian and scale along equator (1:xxxx or UNIT/degree).

#### NON-GEOGRAPHICAL PROJECTIONS:

**-Jp***[a]scale[/origin][rlz]* or **-JP***[a]width[/origin][rlz]* (Polar coordinates (theta,r))  
Optionally insert **a** after **-Jp** [ or **-JP** ] for azimuths CW from North instead of directions CCW from East [default]. Optionally append */origin* in degrees to indicate an angular offset [0]. Finally, append **r** if r is elevations in degrees (requires s >= 0 and n <= 90) or **z** if you want to annotate depth rather than radius [Default]. Give scale in UNIT/r-unit.

**-Jx***-scale[/y-scale]* or **-JX***width[/height]* (Linear, log, and power scaling)  
Give *x-scale* in UNIT/x-unit and/or *y-scale* in UNIT/y-unit; or specify *width* and/or *height* in UNIT (*y-scale=x-scale* if not specified separately). Use negative scale(s) to reverse the direction of an axis (e.g., to have y be positive down). Optionally, append to *x-scale*, *y-scale*, *width* or *height* one of the following:

**d** Data are geographical coordinates (in degrees).

**l** Take log10 of values before scaling.

**ppower** Raise values to *power* before scaling.

**t** Input coordinates are time relative to **TIME\_EPOCH**.

**T** Input coordinates are absolute time.

Default axis lengths (see gmtdefaults) can be invoked using **-JXh** (for landscape); **-JXv** (for portrait) will swap the x- and y-axis lengths. The **GMT** default unit for this installation is UNIT. However, you may change this by editing your .gmtdefaults4 file(s) (run gmtdefaults to create one if you do not have it).

For geographic projections you can give 1 as the scale will be solved for anyway.

**-R** *xmin, xmax, ymin, and ymax* specify the Region of interest. For geographic regions, these limits correspond to *west, east, south, and north* and you may specify them in decimal degrees or in [+]*-dd:mm[:ss.xxx][W|E|S|N]* format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude). For calendar time coordinates you may either give relative time (relative to the selected **TIME\_EPOCH** and in the selected **TIME\_UNIT**; append **t** to **-JXlx**), or absolute time of the form [*date*]**T**[*clock*] (append **T** to **-JXlx**). At least one of *date* and *clock* must be present; the **T** is always required. The *date* string must be of the form [-]*yyyy[-mm[-dd]]* (Gregorian calendar) or *yyyy[-Www[-d]]* (ISO week calendar), while the *clock*

string must be of the form hh:mm:ss[.xxx]. The use of delimiters and their type and positions must be as indicated (however, input/output and plotting formats are flexible).

## OPTIONS

- A** Give an audible signal each time the digitizer mouse/puck is clicked [Default is silent].
- C** Specify the device (port) to read from [Default is /dev/ttyS0].
- D** Only output a point if it is further than *limit* units from the previous point. Append **c**, **i**, **m**, **p** for cm, inch, meter, or point, respectively [Default is no limit].
- F** Force the program to ask for 4 arbitrary calibration points [Default is to use the 4 corners of the map, if possible].
- H** This option allows you to write out any number of header records to the beginning of the output file. Each record will automatically start with a #-character to indicate comment. Headers are not written if multiple output files are selected with **-N -M**.
- L** Set the digitizer table resolution in lines per inch [2540].
- M** Multiple segment file(s). Segments are separated by a special record. For ASCII files the first character must be *flag* [Default is '>']. For binary files all fields must be NaN and **-b** must set the number of output columns explicitly. By default the **-M** setting applies to both input and output. Use **-Mi** and **-Mo** to give separate settings.
- N** Set name for output file(s). If a regular filename is given, then all digitized data will be written to that file. If the file contains a C-format for an integer (i.e., %d) then the file is used as a format statement to create unique filenames based on the current segment number (e.g., line\_%d.d will yield files line\_0.d, line\_1.d, etc). By default, all output is written to stdout. Multiple segment files requires specifying the **-M** option.
- S** Suppress points that fall outside the specified map region [Default outputs all points].
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"]. The program will also duplicate data output to stderr for monitoring.
- Z** Append **v** to prompt for a z-value and output it as a third data column. Append **k** to output the button key as the final data column. Both **-Zk** and **-Zv** can be specified. [Default is just 2 column x,y output].
- O** Selects Overlay plot mode [Default initializes a new plot system].

## EXAMPLES

To digitize lines from a mercator map made for a given region, and save each line segment in individual files called segment\_000.xy, segment\_001.xy etc, try

```
gmtdigitize -R20/50/12/25 -Jm1:1 -M -Nsegment_%3.3d.xy
```

To digitize seismically defined interfaces from a multichannel seismic section, with horizontal distances from 130 to 970, and vertical times from 0 to 10 seconds, write out the button code, and save all line segment to a single multisegment file, and beep at each click, try

```
gmtdigitize -R130/970/0/10 -Jx1/-1 -M -A -Z > interfaces.d
```

## SYSTEM SETUP

This applies to the Calcomp DrawingBoard III hooked up to a RedHat Linux workstation. We use /dev/ttyS0 as the serial port and change permissions so that it is world read/write-able. Then, stty -F /dev/ttyS0 evenp will set the terminal settings, which can be checked with stty -F /dev/ttyS0 -a. Setup of digitizer: We use the CalComp 2000 ASCII (Save 3) setup, which has:

```
Mode: Point
Baud Rate: 9600
Data Bits: 7
Parity: Even
```

Data Rate: 125 pps

Resolution: 200 lpi

Output Format: Format 0

Emulation: CalComp 2000 ASCII

(A) We need to make a slight modification to the Preset No 3 settings: (1) 2450 LPI instead of 200, and (2) None instead of yes for added CR. These modifications can be changed and saved to Preset 3 on the digitizer but a power outage may reset in back to the factory defaults, necessitating a manual reset of those two settings. (B) Setup tty port. `stty -F /dev/ttyS0 evenp` (C) Run `gmtdigitize`. Map scale doesn't matter; it is computed from the region and plot size.

**SEE ALSO**

*gmtdefaults(1), gmt(1), gmtstitch(1), mapproject(1), project(1)*

**NAME**

gmtstitch – Join line segments whose end points match within tolerance

**SYNOPSIS**

```
gmtstitch infiles [ -Dtemplate ] [ -Hi[nrec] ] [ -Milo[flag] ] [ -Tcutoff[m|c|k|K] ] [ -V ] [ -:ilo ] [ -bilo[s|S|d|D][ncol] ] [ -bos|S|d|D[ncol] ]
```

**DESCRIPTION**

**gmtstitch** reads one or more data files (which may be multisegment files; see **-M**) and examines the coordinates of the end points of all line segments. If a pair of end points are identical or closer to each other than the specified separation tolerance then the two line segments are joined into a single segment. The process repeats until all the remaining endpoints no longer pass the tolerance test; the resulting segments are then written out to standard output.

*file(s)* One or more data files. If none are supplied then we read standard input.

**OPTIONS**

- D** For multiple segment data, dump each segment to a separate output file [Default writes a multiple segment file to stdout]. Append a format template for the individual file names; this template **must** contain a C format specifier that can format an integer argument (the segment number); this is usually %d but could be %8.8d which gives leading zeros, etc. [Default is gmtstitch\_segment\_%d.d].
- H** Input file(s) has Header record(s). Number of header records can be changed by editing your .gmtdefaults4 file. If used, **GMT** default is 1 header record. Use **-Hi** if only input data should have header records [Default will write out header records if the input data have them].
- M** Multiple segment file(s). Segments are separated by a special record. For ASCII files the first character must be *flag* [Default is '>']. For binary files all fields must be NaN and **-b** must set the number of output columns explicitly. By default the **-M** setting applies to both input and output. Use **-Mi** and **-Mo** to give separate settings.
- T** Specifies the separation tolerance in the data coordinate units [0]. Append **m** or **c** for minutes or seconds, **k** for km (implies **-fg** and use flat Earth approximation, or **K** for km (implies -fg, and use exact geodesic distances. If the current ELLIPSOID is Sphere then spherical great circle distances are used. If two lines has endpoints that are closer than this cutoff they will be joined.
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- :** Toggles between (longitude,latitude) and (latitude,longitude) input and/or output. [Default is (longitude,latitude)]. Append **i** to select input only or **o** to select output only. [Default affects both].
- bi** Selects binary input. Append **s** for single precision [Default is **d** (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of columns in your binary file if it exceeds the columns needed by the program. [Default is 2 input columns].
- bo** Selects binary output. Append **s** for single precision [Default is **d** (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of desired columns in your binary output file. [Default is same as input].
- f** Special formatting of input and output columns (time or geographical data). Specify **i**(nput) or **o**(utput) [Default is both input and output]. Give one or more columns (or column ranges) separated by commas. Append **T** (Absolute calendar time), **t** (time relative to chosen **TIME\_EPOCH**), **x** (longitude), **y** (latitude), or **f** (floating point) to each column or column range item. Shorthand **-f~~ilo~~g** means **-f~~ilo~~0x,1y** (geographic coordinates).

**ASCII FORMAT PRECISION**

The ASCII output formats of numerical data are controlled by parameters in your .gmtdefaults4 file. Longitude and latitude are formatted according to **OUTPUT\_DEGREE\_FORMAT**, whereas other values are formatted according to **D\_FORMAT**. Be aware that the format in effect can lead to loss of precision in the output, which can lead to various problems downstream. If you find the output is not written with enough precision, consider switching to binary output (**-bo** if available) or specify more decimals using the

**D\_FORMAT** setting.

## EXAMPLES

To combine the digitized multisegment lines `segment_*.d` (whose coordinates are in cm) into as few complete lines as possible, assuming the end points slop could be up to 0.1 mm, run

```
gmtstitch segment_*.d -Tf0.1 -M > new_segments.d
```

To combine the digitized segments in the multisegment file `my_lines.d` (whose coordinates are in lon,lat) into as few complete lines as possible, assuming the end points slop could be up to 150 m, and write the complete segments to separate files called `Map_segment_0001.dat`, `Map_segment_0002.dat`, etc., run

```
gmtstitch my_lines.d -Tf0.15k -M -DMap_segment_%4.4d.dat
```

## SEE ALSO

*GMT(1)*, *mapproject(1)*

**NAME**

makepattern – make GMT color pattern from b/w pattern or icon

**SYNOPSIS**

**makepattern** *lbit.ras* | *iconfile* **-Cfr/g/b** **-Cbr/g/b**

**DESCRIPTION**

**makepattern** reads either a Sun 1-bit rasterfile OR a Sun icon file. It then accepts color choices for the foreground and background pixels and colorizes the pattern, writing it as a 8-bit Sun rasterfile on stdout. These patterns may then be used in **GMT** (3.1 or later) **-Gp** settings or by **psimage**.

*lbit.ras* | *iconfile*

Either a 1-bit Sun rasterfile (standard format, no RLE) or a Sun icon file (as used in **GMT** 3.0).

**-Cf** Sets the color for the foreground pixels (the ones) [black].

**-Cb** Sets the color for the background pixels (the zeros) [white].

**WIN32 fix**

Since binary redirection under WIN32 does not work, you must specify the output file with **-Gnewfile.ras**. This option is only available under WIN32.

**EXAMPLES**

To create a colored red/blue 8-bit Sun rasterfile pattern from the old 3.0-style iconpattern stored in file custom.icon, run

```
makepattern custom.icon -Cf255/0/0 -Cb0/0/255 > custom.ras
```

To create a green/blue 8-bit Sun rasterfile pattern from a 1-bit Sun raster called bits.ras, run

```
makepattern bits.ras -Cf0/255/0 -Cb0/0/255 > colorbits.ras
```

**SEE ALSO**

*GMT*(1), *psimage*(1)

**NAME**

`nc2xy` – Converting netCDF column file(s) to ASCII xy data

**SYNOPSIS**

`nc2xy files [ -Fvar1/var2/... ] [ -S[r] ] [ -V ]`

**DESCRIPTION**

`nc2xy` reads one or more netCDF files with column data and writes out those columns in ASCII format to standard output, so that they can be used by `psxy`, `psxyz`, or `xyz2grd`. Modify the precision of the ASCII output format by editing the `D_FORMAT` parameter in your `.gmtdefaults4` file or use `--D_FORMAT=format` on the command line.

*files* Names of netCDF files to be converted.

**OPTIONS**

- `-F` Specify up to 10 names of the variables (separated by slashes) to be printed out. All variables need to have only one dimension of equal length. When omitted, the first two variables in the netCDF file will be printed.
- `-S` Suppress output for records with one or more NaN values [Default outputs all nodes]. Append `r` to reverse the suppression, i.e., only output the records with at least one NaN value.
- `-V` Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].

**EXAMPLES**

To print out latitude, longitude and height in a netCDF file as ASCII records, while suppressing all NaN values:

```
nc2xy -Flat/lon/height -S trackfile.nc > trackfile.xy
```

**SEE ALSO**

`gmtdefaults(1)`, `GMT(1)`, `grdedit(1)`, `psxy(1)`, `psxyz(1)`, `xyz2grd(1)`

**NAME**

`ps2raster` – Converts one or several *PostScript* file(s) to raster formats using GhostScript

**SYNOPSIS**

`ps2raster psfile(s) [ -A ] [ -Eresolution ] [ -Gghost_path ] [ -Llistfile ] [ -N ] [ -Tfjglmt ]`

**DESCRIPTION**

`ps2raster` reads one or more *PostScript* files (or a file with a list of *PostScript* filenames) and modifies the BoundingBox and PageSize in order that the raster image will have a size determined by the BoundingBox. As an option, a tight BoundingBox may be computed.

*psfiles* Names of *PostScript* files to be converted. The output files will have the same name but with the conventional extension name associated to the raster format (e.g. .jpg for the jpeg format).

**OPTIONS**

- A Adjust the BoundingBox to the minimum required by the image content.
- E Set raster resolution in dpi [default = 300].
- G Full path to your ghostscript executable. NOTE: For Unix systems this is generally not necessary. However, under Windows, GhostScript is not added to the system's path. So either you do it yourself, or give the full path here. (e.g., -Gc:\programs\gs\gs7.05\bin\gswin32c).
- L The *listfile* is an ASCII file with the names of the *PostScript* files to be converted.
- N Do *not* remove auxiliary files (by default it does). Auxiliary files are built using as base the input *PostScript* files and consist of: *psfile\_tmpxx.eps* -> *PostScript* with a modified BoundingBox  
*psTO raster.sclbat* -> script (or batch) with the ghostscript command that does the job. Use this option to save the script and run it later with different settings.
- T Sets the output raster format, where **f** means PDF, **j** means JPEG, **g** means PNG, **m** means PPM, and **t** means TIF [default is JPEG].

**EXAMPLE**

To convert a the file *psfile.ps* to png using a tight BoundingBox and assuming that *gs* can be found in your system's path:

```
ps2raster psfile.ps -A -Tg
```

**SEE ALSO**

*GMT*(1)

**NAME**

`psmegaplot` – To create poster-size *PostScript* plots from page-size plot

**SYNOPSIS**

`psmegaplot plotfile -Sscale [ -C ]`

**DESCRIPTION**

`psmegaplot` reads a **GMT**-produced *PostScript* file and 'blows it up' by an arbitrary scale. The result is several plots that make up the pieces in a jigsaw puzzle. Cut and paste these pieces to make a 'megaplot'.

*plotfile* *PostScript* file as created by the **GMT** programs.

`-S` Multiply the x- and y-dimension by *scale*. *scale* must be larger than 1.0.

**OPTIONS**

No space between the option flag and the associated arguments.

`-C` Plot crop marks in each corner of every output page. [Default is no marks]

**EXAMPLES**

To magnify a contour-map *PostScript* file 3 times, giving a total of 9 output pages to be pasted together, use

```
psmegaplot hawaii_contour.ps -S3 -C > hawaii_contour_megaplot.ps
```

**BUGS**

This program is unlikely to work on non-**GMT**-produced *PostScript* files since it anticipates the structure of the file. It also assumes that the output media is Letter size.

**SEE ALSO**

*GMT*(1)

**NAME**

pssegy – Create imagemasked postscript from SEGY file

**SYNOPSIS**

```
pssegy SEGYfile -Jparameters -Rwest/east/south/north[r] -Ddeviation -F[rgb|gray]|-W [ -Bbias ] [
-Cclip ] [ -Eerror ] [ -I ] [ -K ] [ -Lnsamp ] [ -Mntrace ] [ -N ] [ -O ] [ -P ] [ -Sheader ] [ -Tfilename
] [ -Uredvel ] [ -V ] [ -Xscale ] [ -Ysample int ] [ -Z ]
```

**DESCRIPTION**

**pssegy** reads a native (IEEE) format SEGY file and produces a *PostScript* image of the seismic data. The *imagemask* operator is used so that the seismic data are plotted as a 1-bit deep bitmap in a single (user-specified) color or gray shade, with a transparent background. The bitmap resolution is taken from the current **GMT** defaults. The seismic traces may be plotted at their true locations using information in the trace headers (in which case order of the traces in the file is not significant). Standard **GMT** geometry routines are used so that in principle any map projection may be used, however it is likely that the geographic projections will lead to unexpected results. Beware also that some parameters have non-standard meanings. Note that the order of operations before the seismic data are plotted is deviation\*[clip]([bias]+[normalize])(sample value)). Deviation determines how far *in the plot coordinates* a [normalized][biased][clipped] sample value of 1 plots from the trace location. The SEGY file should be a disk image of the tape format (ie 3200 byte text header, which is ignored, 400 byte binary reel header, and 240 byte header for each trace) with samples as native real\*4 (IEEE real on all the platforms to which I have access)

*SEGYfile*

Seismic data set to be imaged

- J** Selects the map projection. Scale is UNIT/degree, 1:xxxxx, or width in UNIT (upper case modifier). UNIT is cm, inch, or m, depending on the **MEASURE\_UNIT** setting in .gmtdefaults4, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width value. For map height, max dimension, or min dimension, append **h**, **+**, or **-** to the width, respectively. More details can be found in the **psbasemap** man pages.

**CYLINDRICAL PROJECTIONS:**

- Jclon0/lat0/scale** (Cassini)
- Jjlon0/scale** (Miller)
- Jmscale** (Mercator - Greenwich and Equator as origin)
- Jmlon0/lat0/scale** (Mercator - Give meridian and standard parallel)
- Joalon0/lat0/azimuth/scale** (Oblique Mercator - point and azimuth)
- Joblon0/lat0/lon1/lat1/scale** (Oblique Mercator - two points)
- Joclon0/lat0/lonp/latp/scale** (Oblique Mercator - point and pole)
- Jqlon0/scale** (Equidistant Cylindrical Projection (Plate Carree))
- Jtlon0/scale** (TM - Transverse Mercator, with Equator as y = 0)
- Jtlon0/lat0/scale** (TM - Transverse Mercator, set origin)
- Juzone/scale** (UTM - Universal Transverse Mercator)
- Jylon0/lats/scale** (Basic Cylindrical Projection)

**AZIMUTHAL PROJECTIONS:**

- Jalon0/lat0/scale** (Lambert)
- Jelon0/lat0/scale** (Equidistant)
- Jflon0/lat0/horizon/scale** (Gnomonic)
- Jglon0/lat0/scale** (Orthographic)
- Jslon0/lat0/[slat]/scale** (General Stereographic)

**CONIC PROJECTIONS:**

- Jblon0/lat0/lat1/lat2/scale** (Albers)

- Jd***lon0/lat0/lat1/lat2/scale* (Equidistant)
- Jl***lon0/lat0/lat1/lat2/scale* (Lambert)

#### MISCELLANEOUS PROJECTIONS:

- Jh***lon0/scale* (Hammer)
- Ji***lon0/scale* (Sinusoidal)
- Jk[fls]***lon0/scale* (Eckert IV (f) and VI (s))
- Jn***lon0/scale* (Robinson)
- Jr***lon0/scale* (Winkel Tripel)
- Jv***lon0/scale* (Van der Grinten)
- Jw***lon0/scale* (Mollweide)

#### NON-GEOGRAPHICAL PROJECTIONS:

- Jp[a]***scale[/origin][r|z]* (Polar coordinates (theta,r))
  - Jxx-scale[dlllppow|t|T][y-scale[dlllppow|t|T]]** (Linear, log, and power scaling)
  - R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd:mm[:ss.xxx][W|E|S|N]* format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).
  - D** gives the deviation in X units of the plot for 1.0 on the scaled trace.
  - F[rgb|gray]**  
Fill trace (variable area, defaults to filling positive). *rgb* or *gray* gives the color with which the *imagemask* is filled.
  - W** Draw wiggle trace.
- You *must* specify at least one of **-W** and **-F**.

#### OPTIONS

No space between the option flag and the associated arguments.

- B** Bias to apply to data (added to sample values).
- C** Sample value at which to clip data (clipping is applied to both positive and negative values).
- E** Allow *error* difference between requested and actual trace locations when using **-T** option.
- I** Fill negative rather than positive excursions.
- K** More *PostScript* code will be appended later [Default terminates the plot system].
- L** Override number of samples per trace in reel header (program attempts to determine number of samples from each trace header if possible to allow for variable length traces).
- M** Override number of traces specified in reel header. Program detects end of file (relatively) gracefully, but this parameter limits number of traces that the program attempts to read.
- N** Normalize trace by dividing by rms amplitude over full trace length.
- O** Selects Overlay plot mode [Default initializes a new plot system].
- P** Selects Portrait plotting mode [GMT Default is Landscape, see **gmtdefaults** to change this].
- S** Read trace locations from trace headers: header is either *c* for CDP, *o* for offset, or *b<num>* to read a long starting at byte *<num>* in the header (first byte corresponds to *num=0*). Default has location given by trace number.
- T** Plot only traces whose location corresponds to a list given in *filename*. Order in which traces are listed is not significant - the entire space is checked for each trace.

- U Apply reduction velocity by shifting traces *upwards* by redvel/loffsetl. Negative velocity removes existing reduction. Units should be consistent with offset in trace header and sample interval.
- V Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- X Multiply trace locations by scale before plotting.
- Y Override sample interval in reel header.
- Z Do not plot traces with zero rms amplitude.

**EXAMPLES**

To plot the SEG Y file wal.segy with normalized traces plotted at true offset locations, clipped at +-3 and with wiggle trace and positive variable area shading in black, use

```
pssegy wal.segy -JX5/-5 -R0/100/0/10 -D1 -C3 -N -So -W -F0 > segy.ps
```

To plot the SEG Y file wal.segy with traces plotted at true cdp\*0.1, clipped at +-3, with bias -1 and negative variable area shaded red, use

```
pssegy wal.segy -JX5/-5 -R0/100/0/10 -D1 -C3 -Sc -X0.1 -F255/0/0 -B-1 -I > segy.ps
```

**SEE ALSO**

*GMT(1)*, *pssegyz(1)*

**NAME**

pssegyz – Create imagemasked postscript from SEG Y file

**SYNOPSIS**

**pssegyz** *SEG Yfile* **-J***parameters* **-Jz/Z***parameters* **-R***west/east/south/north*[*zmin/zmax*][**r**] **-D***deviation*  
**-F**[*rgb/gray*]|**-W** [**-B***bias*] [**-C***clip*] [**-E***azimuth/elevation*] [**-I**] [**-K**] [**-L***nsamp*] [**-M***ntrace*] [**-N**] [**-O**] [**-P**] [**-S***header\_x/header\_y*] [**-U***redvel*] [**-V**] [**-X***scale*] [**-Y***sample\_int*] [**-Z**]

**DESCRIPTION**

**pssegyz** reads a native (IEEE) format SEG Y file and produces a *PostScript* image of the seismic data. The *imagemask* operator is used so that the seismic data are plotted as a 1-bit deep bitmap in a single (user-specified) color or gray shade, with a transparent background. The bitmap resolution is taken from the current **GMT** defaults. The seismic traces may be plotted at their true locations using information in the trace headers (in which case order of the traces in the file is not significant). Standard **GMT** geometry routines are used so that in principle any map projection may be used, however it is likely that the geographic projections will lead to unexpected results. Beware also that some parameters have non-standard meanings, and a couple of the options for **pssegy** are not available in **pssegyz**. Note that the order of operations before the seismic data are plotted is *deviation\*[clip]([bias]+[normalize](sample value))*. Deviation determines how far *in the plot coordinates* a [normalized][biased][clipped] sample value of 1 plots from the trace location. The SEG Y file should be a disk image of the tape format (ie 3200 byte text header, which is ignored, 400 byte binary reel header, and 240 byte header for each trace) with samples as native real\*4 (IEEE real on all the platforms to which I have access)

*SEG Yfile*

Seismic data set to be imaged

- J** Selects the map projection. Scale is UNIT/degree, 1:xxxxx, or width in UNIT (upper case modifier). UNIT is cm, inch, or m, depending on the **MEASURE\_UNIT** setting in *.gmtdefaults4*, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width value. For map height, max dimension, or min dimension, append **h**, **+**, or **-** to the width, respectively. More details can be found in the **psbasemap** man pages.

**CYLINDRICAL PROJECTIONS:**

- Jclon0/lat0/scale** (Cassini)
- Jjlon0/scale** (Miller)
- Jmscale** (Mercator - Greenwich and Equator as origin)
- Jmlon0/lat0/scale** (Mercator - Give meridian and standard parallel)
- Joalon0/lat0/azimuth/scale** (Oblique Mercator - point and azimuth)
- Joblon0/lat0/lon1/lat1/scale** (Oblique Mercator - two points)
- Joclon0/lat0/lonp/latp/scale** (Oblique Mercator - point and pole)
- Jqlon0/scale** (Equidistant Cylindrical Projection (Plate Carree))
- Jtlon0/scale** (TM - Transverse Mercator, with Equator as  $y = 0$ )
- Jtlon0/lat0/scale** (TM - Transverse Mercator, set origin)
- Juzone/scale** (UTM - Universal Transverse Mercator)
- Jylon0/lats/scale** (Basic Cylindrical Projection)

**AZIMUTHAL PROJECTIONS:**

- Jalon0/lat0/scale** (Lambert)
- Jelon0/lat0/scale** (Equidistant)
- Jflon0/lat0/horizon/scale** (Gnomonic)
- Jglon0/lat0/scale** (Orthographic)
- Jslon0/lat0/[slat]/scale** (General Stereographic)

**CONIC PROJECTIONS:**

- Jb***lon0/lat0/lat1/lat2/scale* (Albers)
- Jd***lon0/lat0/lat1/lat2/scale* (Equidistant)
- Jl***lon0/lat0/lat1/lat2/scale* (Lambert)

#### MISCELLANEOUS PROJECTIONS:

- Jh***lon0/scale* (Hammer)
- Ji***lon0/scale* (Sinusoidal)
- Jk**[*fls*]*lon0/scale* (Eckert IV (f) and VI (s))
- Jn***lon0/scale* (Robinson)
- Jr***lon0/scale* (Winkel Tripel)
- Jv***lon0/scale* (Van der Grinten)
- Jw***lon0/scale* (Mollweide)

#### NON-GEOGRAPHICAL PROJECTIONS:

- Jp**[*a*]*scale*[*origin*][*r*][*z*] (Polar coordinates (theta,r))
- Jxx**-*scale*[*dlllppow*][*t*][*T*][*y-scale*][*dlllppow*][*t*][*T*] (Linear, log, and power scaling)
- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*-dd:mm[:ss.xxx]*[*W|E|S|N*] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).
- D** gives the deviation in X units of the plot for 1.0 on the scaled trace, This may be a single number (applied equally in X and Y directions) or *devX/devY*
- E** Azimuth and elevation for the 3-D projection.
- F**[*rgb*]*gray*  
Fill trace (variable area, defaults to filling positive). *rgb* or *gray* gives the color with which the *imagemask* is filled.
- W** Draw wiggle trace.

You *must* specify at least one of **-W** and **-F**.

#### OPTIONS

No space between the option flag and the associated arguments.

- B** Bias to apply to data (added to sample values).
- C** Sample value at which to clip data (clipping is applied to both positive and negative values).
- I** Fill negative rather than positive excursions.
- K** More *PostScript* code will be appended later [Default terminates the plot system].
- L** Override number of samples per trace in reel header (program attempts to determine number of samples from each trace header if possible to allow for variable length traces).
- M** Override number of traces specified in reel header. Program detects end of file (relatively) gracefully, but this parameter limits number of traces that the program attempts to read.
- N** Normalize trace by dividing by rms amplitude over full trace length.
- O** Selects Overlay plot mode [Default initializes a new plot system].
- P** Selects Portrait plotting mode [GMT Default is Landscape, see **gmtdefaults** to change this].
- S** Read trace locations from trace headers: header is either *c* for CDP, *o* for offset, *b*<num> to read a long starting at byte <num> in the header (first byte corresponds to num=0), or a number to fix the location.  
First parameter for *x*, second for *y*. Default has *X* and *Y* given by trace number.

- U Apply reduction velocity by shifting traces *upwards* by redvel/loffsetl. Negative velocity removes existing reduction. Units should be consistent with offset in trace header and sample interval.
- V Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- X Multiply trace locations by scale before plotting.
- Y Override sample interval in reel header.
- Z Do not plot traces with zero rms amplitude.

### EXAMPLES

To plot the SEGY file wa1.segy with normalized traces plotted at true offset locations, clipped at +-3 and with wiggle trace and positive variable area shading in black, use

```
pssegyz wa1.segy -JX5/-5 -D1 -Jz0.05 -E180/5 -R0/100/0/10/0/10 -C3 -N -So -W -F0 > segy.ps
```

### BUGS

Variable area involves filling four-sided figures of distressing generality. I *know* that some of the more complex degenerate cases are not dealt with correctly or at all; the incidence of such cases increases as viewing angles become more oblique, and particularly as the viewing elevation increases. Wiggle-trace plotting is not affected.

### SEE ALSO

*GMT*(1), *pssegy*(1)

**NAME**

`segy2grd` – Converting SEGYZ file to grd file format

**SYNOPSIS**

```
segy2grd segyfile -Ggrdfile -Ixinc[unit][=|+][/yinc[unit][=|+]] -Rwest/east/south/north[r] [ -A[n|z] ] [
-Dxname/yname/zname/scale/offset/title/remark ] [ -F ] [ -Nnodata ] [ -S[zfile] ] [ -V ] [ -Z[flags] ] [
-:[ilo] ] [ -bi[s|S|d|D][ncol] ]
```

**DESCRIPTION**

**segy2grd** reads an IEEE SEGYZ file and creates a binary grdfile. Either a simple mapping (equivalent to `xyz2grd -Z`) or a more complicated averaging where a particular grid cell includes values from more than one sample in the SEGYZ file can be done. **segy2grd** will report if some of the nodes are not filled in with data. Such unconstrained nodes are set to a value specified by the user [Default is NaN]. Nodes with more than one value will be set to the average value.

*segyfile* ASCII [or binary] file holding *z* or (*x,y,z*) values. *xyz* triplets do not have to be sorted (for binary triplets, see **-b**). 1-column *z* tables must be sorted and the **-Z** must be set).

**-G** *grdfile* is the name of the binary output grdfile.

**-I** *x\_inc* [and optionally *y\_inc*] is the grid spacing. Append **m** to indicate minutes or **c** to indicate seconds.

**-R** *west, east, south,* and *north* specify the Region of interest, and you may specify them in decimal degrees or in [*+|-*]*dd:mm[:ss.xxx]*[*W|E|S|N*] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).

**OPTIONS**

**-A** Add up multiple values that belong to the same node (same as **-Az**). Append **n** to simply count the number of data points that were assigned to each node. [Default (no **-A** option) will calculate mean value]. Not used for simple mapping.

**-D** Give values for *xname, yname, zname, scale, offset, title,* and *remark*. To leave some of these values untouched, specify = as the value.

**-F** Force pixel registration [Default is grid registration].

**-N** No data. Set nodes with no input sample to this value [Default is NaN].

**-S** set variable spacing *header* is **c** for cdp, **o** for offset, **b**<number> for 4-byte float starting at byte number. If **-S** not set, assumes even spacing of samples at the *dx, dy* supplied with **-I**

**-L** Override number of samples in each trace

**-X** applies scalar *x-scale* to coordinates in trace header to match the coordinates specified in **-R**

**-Y** Specifies sample interval as *s\_int* if incorrect in the SEGYZ file

**-M** Fix number of traces to read in. Default tries to read 10000 traces. **-M0** will read number in binary header, **-Mn** will attempt to read only *n* traces.

**-V** Selects verbose mode, which will send progress reports to `stderr` [Default runs "silently"].

**EXAMPLES**

To create a grdfile from an even spaced SEGYZ file `test.segy`, try

```
segy2grd test.segy -I0.1/0.1 -Gtest.grd -R198/208/18/25 -V
```

Note that this will read in 18-25s (or km) on each trace, but the first trace will be assumed to be at X=198

To create a grdfile from the SEGYZ file `test.segy`, locating traces according to the CDP number, where there are 10 CDPs per km and the sample interval is 0.1, try

```
seg y2grd test.seg y -Gtest.grd -R0/100/0/10 -I0.5/0.2 -V -X0.1 -Y0.1
```

Because the grid interval is larger than the SEG Y file sampling, the individual samples will be averaged in bins

**SEE ALSO**

*GMT(1)*, *grd2xyz(1)*, *grdedit(1)*, *psseg y(1)*

**NAME**

backtracker – Forward and Backward flowlines and hotspot tracks

**SYNOPSIS**

```
backtracker [infile(s)] -Estage_file [ -A[young/old] ] [ -C ] [ -Dfb ] [ -H[i][nrec] ] [ -Lfbstep ] [
-M[i][flag] ] [ -Nupper_age ] [ -Qfixed_age ] [ -Sfilestem ] [ -Tzero_age ] [ -V ] [ -W[alt] ] [ -:[i][lo] ]
[ -b[i][sl]SldID][ncol] ]
```

**DESCRIPTION**

**backtracker** reads (longitude, latitude, age) positions from *infiles* [or standard input] and computes projected (x,y,t) coordinates using the specified rotation parameters. It can either calculate final positions [Default] or create a sampled track between the initial and final positions. The former mode allows additional data fields after the first 3 columns which must have (longitude,latitude,age). See option **-:** on how to read (latitude,longitude,age) files.

No space between the option flag and the associated arguments. Use upper case for the option flags and lower case for modifiers.

*infile(s)* Data file(s) to be projected. If not given, standard input is read.

**-E** Give file with rotation poles and opening angles. This file must contain one record for each rotation; each record must be of the following format:

```
lon lat tstart [tstop] ccw-angle [k a b c d e f g]
```

where tstart [and tstop] are in Ma, with lon, lat, and angle in degrees. For stage rotations, the oldest stage must be listed first, whereas for total reconstruction rotations the youngest rotation is listed first. tstart (tstop) is the age of the old (young) end of the stage. A covariance matrix, if present, may be given in the given format. Blank lines and records whose first column contains # will be ignored.

**OPTIONS**

- A** Used in conjunction with **-Lbif** to limit the track output to those sections whose predicted ages fit between the *young* and *old* limits. If **-LBIF** is used instead then the limits apply to the stage ids. If no limits are supplied then individual limits for each seamount are expected in columns 4 and 5 of the input file.
- C** Expect Total Reconstruction Poles (Finite Poles) rather than Backward Stage Poles [Default]. File format is similar to the stage pole format except the youngest rotation must be listed first and that the tstart column is optional (assumed to be 0 Ma).
- D** Set the direction to go: **-Df** will go forward in time (from younger to older positions), while **-Db** will go backward in time (from older to younger positions) [Default].
- H** Input file(s) has Header record(s). Number of header records can be changed by editing your *.gmtdefaults4* file. If used, **GMT** default is 1 header record. Use **-Hi** if only input data should have header records [Default will write out header records if the input data have them].
- L** Specify a sampled path between initial and final position: **-Lf** will draw flowlines, while **-Lb** will draw backtrack (hotspot track) paths. Append sampling interval in km. If *step* < 0 then only the bend-times (knots) will be returned. When **-LF** or **-LB** is used, the third output column will contain the stage id (1 is youngest) [Default is along-track ages].
- M** Multiple segment file(s). Segments are separated by a special record. For ASCII files the first character must be *flag* [Default is '>']. For binary files all fields must be NaN and **-b** must set the number of output columns explicitly. By default the **-M** setting applies to both input and output. Use **-Mi** and **-Mo** to give separate settings.
- N** Set the maximum age to extend the oldest stage back in time [Default is no extension].
- Q** Assign a fixed age to all positions. Only lon, lat input is expected.

- S** When **-L** is set, the tracks are normally written to stdout as a multisegment file. Specify a *filestem* to have each track written to *filestem.#*, where *#* is the track number. The track number is also copied to the 4th output column.
- T** Set the current time [Default is 0 Ma].
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- W** Rotates the given input (lon,lat,t) and calculates the confidence ellipse for the projected point. The input point *must* have a time coordinate that matches a particular finite rotation time, otherwise the point will be skipped. Append **t** or **a** to output time or angle, respectively, after the projected lon, lat. After these 2-3 items, we write azimuth, major, minor (in km) for the 95% confidence ellipse.
- :** Toggles between (longitude,latitude) and (latitude,longitude) input and/or output. [Default is (longitude,latitude)]. Append **i** to select input only or **o** to select output only. [Default affects both].
- bi** Selects binary input. Append **s** for single precision [Default is **d** (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of columns in your binary file if it exceeds the columns needed by the program. [Default is 3 input columns].
- bo** Selects binary output. Append **s** for single precision [Default is **d** (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of desired columns in your binary output file. [Default depends on settings].

### EXAMPLES

To backtrack the (x,y,t) points in the file seamounts.d to their origin (presumably the hotspot), using the DC85.d Euler poles, run

```
backtracker seamounts.d -Db -EDC85.d > newpos.d
```

To project flowlines forward from the (x,y,t) points stored in several 3-column, binary, double precision files, run

```
backtracker points.* -Df -EDC85.d -Lf25 -bo -bi3 > lines.b
```

This file can then be plotted with **psxy -M**.

### SEE ALSO

*GMT*(1), *project*(1), *mapproject*(1), *hotspotter*(1), *originator*(1)

### REFERENCES

Wessel, P., 1999, "Hotspotting" tools released, EOS Trans. AGU, 80 (29), p. 319.

**NAME**

grdrotater – Rotate a grid using a finite rotation

**SYNOPSIS**

```
grdrotate ingrdfile -Goutgrdfile -Tplon/plat/omega [ -Fpolygonfile ] [ -H[nrec] ] [ -M[flag] ] [ -N ] [ -Q[value] ] [ -Rwest/east/south/north[r] ] [ -S ] [ -V ] [ -:[ilo] ] [ -b[ilo][s|S|d|D][ncol] ]
```

**DESCRIPTION**

**grdrotater** reads a geographical grid and reconstructs it given a total reconstruction rotation. Optionally, the user may supply a clipping polygon in multiple-segment format; then, only the part of the grid inside the polygon is used to determine the return grid region. The outline of the projected region is returned on stdout.

No space between the option flag and the associated arguments. Use upper case for the option flags and lower case for modifiers.

*ingrdfile*

Name of a grid file in geographical (lon, lat) coordinates.

**-G** Name of output grid. This is the grid with the data reconstructed according to the specified rotation.

**-T** Finite rotation. Specify the longitude and latitude of the rotation pole and the opening angle, all in degrees.

**OPTIONS**

**-F** Specify a multi-segment closed polygon file that describes the inside area of the grid that should be projected [Default projects entire grid].

**-H** Input file(s) has Header record(s). Number of header records can be changed by editing your `.gmtdefaults4` file. If used, **GMT** default is 1 header record. Use **-Hi** if only input data should have header records [Default will write out header records if the input data have them].

**-M** Multiple segment file. Segment separator is a record beginning with *flag*. [Default is '>'].

**-N** Do Not output the rotated polygon outline [Default will write it to stdout].

**-Q** Quick mode, use bilinear rather than bicubic interpolation [Default]. Optionally, append *value* in the  $0 \leq value \leq 1$  range. This parameter controls how close to nodes with NaN values the interpolation will go. E.g., a *value* of 0.5 will interpolate about 1/2-way from a non-NaN to a NaN node, whereas 0.1 will go about 90% of the way, etc. [Default is 1, which means none of the four nearby nodes may be NaN]. A *value* of 0 will just return the value of the nearest node instead of interpolating.

**-R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in `[+]dd:mm[:ss.xxx][W|E|S|N]` format. Append *r* if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).

**-S** Skip the rotation of the grid, just rotate the polygon outline (requires **-F**).

**-V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].

**-:** Toggles between (longitude,latitude) and (latitude,longitude) input/output. [Default is (longitude,latitude)].

**-bi** Selects binary input. Append *s* for single precision [Default is **d** (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of columns in your binary file if it exceeds the columns needed by the program. [Default is 2 input columns].

**-bo** Selects binary output. Append *s* for single precision [Default is **d** (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of desired columns in your binary output file. [Default is same as input].

**EXAMPLES**

To rotate the data defined by grid `topo.grd` and the polygon outline `clip_path.d`, using a finite rotation with pole at (135.5, -33.0) and a rotation angle of 37.3 degrees and bicubic interpolation, try

```
grdrotater topo.grd -T135.5/-33/37.3 -V -Fclip_path.d -Grot_topo.grd > rot_clip_path.d
```

To rotate the entire grid `faa.grd` using a finite rotation pole at (67:45W, 22:35S) and a rotation angle of 19.6 degrees using a bilinear interpolation, try

```
grdrotater faa.grd -T67:45W/22:35S/19.6 -V -Q -Grot_faa.grd > rot_faa_path.d
```

To just see how the outline of the grid `large.grd` will plot after the same rotation, try

```
grdrotater large.grd -T67:45W/22:35S/19.6 -V -S | psxy -Rg -JH180/6i -B30 -M -W0.5p | gv -
```

Let say you have rotated `gridA.grd` and `gridB.grd`, restricting each rotation to nodes inside polygons `polyA.d` and `polyB.d`, respectively, using rotation A = (123W,22S,16,4) and rotation B = (108W, 16S, -14.5), yielding rotated grids `rot_gridA.grd` and `rot_gridB.grd`. To determine the region of overlap between the rotated grids, we use `grdmath`:

```
grdmath 1 rot_gridA.grd ISNAN SUB 1 rot_gridB.grd ISNAN SUB 2 EQ = overlap.grd
```

The grid `overlap.grd` now has 1s in the regions of overlap and 0 elsewhere. You can use it as a mask or use `grdcontour` to extract a polygon (contour).

**SEE ALSO**

*backtracker(1), hotspotter(1), originator(1) rotconverter(1)*

**NAME**

hotspotter – Create CVA image from seamount flowlines

**SYNOPSIS**

```
hotspotter [infile(s)] -Estage_file -GCVAGrid -Lxinc[unit][=+][/yinc[unit][=+]]
-Rwest/east/south/north[r] [-C] [-Dfactor] [-F] [-H[i][nrec]] [-Nupper_age] [-T] [-V] [-:[:iio]]
] [-bi[s|S|d|D][ncol]]
```

**DESCRIPTION**

**hotspotter** reads (longitude, latitude, amplitude, radius, age) records from *infile(s)* [or standard input] and calculates flowlines using the specified stage pole (Euler) rotations. These flowlines are convolved with the shape of the seamount (using a Gaussian shape given amplitude and radius = 6 sigma) and added up to give a Cumulative Volcano Amplitude grid (CVA). See option **-:** on how to read (latitude,longitude,...) files.

No space between the option flag and the associated arguments. Use upper case for the option flags and lower case for modifiers.

*infile(s)* Data file(s) to be processed. If not given, standard input is read.

**-E** Give file with stage poles and opening angles. This file must contain one record for each stage; each record must be of the following format:

```
lon lat tstart tstop ccw-angle
```

where tstart and tstop are in Ma and the rest is in degrees. The oldest stage must be listed first. tstart (tstop) is the age of the old (young) end of the stage. Blank lines and records whose first column contains # will be ignored.

**-G** Specify name for output grid file.

**-I** *x\_inc* [and optionally *y\_inc*] is the grid spacing. Append **m** to indicate minutes or **c** to indicate seconds. If one of the units **e**, **k**, **i**, or **n** is appended instead, the increment will be assumed to be in meter, km, miles, or nautical miles, respectively, and will be converted to the equivalent degrees longitude at the middle latitude of the region (the conversion depends on **ELLIPSOID**). If *y\_inc* is given but set to 0 it will be reset equal to *x\_inc*; otherwise it will be converted to degrees latitude. If = is appended then the corresponding max *x* (*east*) or *y* (*north*) may be slightly adjusted to fit exactly the given increment [by default the increment may be adjusted slightly to fit the given domain]. Finally, instead of giving an increment you may specify the *number of nodes* desired by appending **+** to the supplied increment; the increment is then recalculated from the number of nodes and the domain. The resulting increment value depends on whether you have selected a gridline-registered or pixel-registered grid; see Appendix B for details.

**-R** *west*, *east*, *south*, and *north* specify the Region of interest, and you may specify them in decimal degrees or in [+]-dd:mm[:ss.xxx][W|E|S|N] format. Append **r** if lower left and upper right map coordinates are given instead of wesn. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).

**OPTIONS**

**-C** Expect Total Reconstruction Poles (Finite Poles) rather than Backward Stage Poles [Default] File format is similar to the stage pole format except the youngest rotation must be listed first and that the tstart column is optional (assumed to be 0 Ma).

**-D** Modify the sampling interval along flowlines. Default [0.5] gives approximately 2 points within each grid box. Smaller factors gives higher resolutions at the expense of longer processing time.

**-F** Force pixel registration [Default is grid registration].

**-H** Input file(s) has Header record(s). Number of header records can be changed by editing your .gmtdefaults4 file. If used, **GMT** default is 1 header record. Use **-Hi** if only input data should have header records [Default will write out header records if the input data have them].

**-N** Set the upper age to assign seamounts whose crustal age is unknown (i.e. NaN) [no upper age].

- T** Truncate seamount ages exceeding the upper age set with **-N** [no truncation].
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- :** Toggles between (longitude,latitude) and (latitude,longitude) input and/or output. [Default is (longitude,latitude)]. Append **i** to select input only or **o** to select output only. [Default affects both].
- bi** Selects binary input. Append **s** for single precision [Default is **d** (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of columns in your binary file if it exceeds the columns needed by the program. [Default is 5 input columns].

### EXAMPLES

To create a CVA image from the Pacific (x,y,z,r,t) data in the file *seamounts.d*, using the DC85.d Euler poles, run

```
hotspotter seamounts.d -EDC85.d -GCVA.grd -R130/260/-66/60 -I10m -N145 -T -V
```

This file can then be plotted with **grdimage**.

### SEE ALSO

*GMT(1)*, *grdimage(1)*, *project(1)*, *mapproject(1)*, *backtracker(1)*, *originator(1)*

### REFERENCES

Wessel, P., 1999, "Hotspotting" tools released, EOS Trans. AGU, 80 (29), p. 319.

**NAME**

originator – Associate seamounts with hotspot point sources

**SYNOPSIS**

**originator** [*infile(s)*] **-E***stage\_file* **-F***hs\_file* [ **-C** ] [ **-D***d\_km* ] [ **-H***i*[*nrec*] ] [ **-N***upper\_age* ] [ **-S**[*n\_hs*] ] [ **-T** ] [ **-V** ] **-W***maxdist* [ **-:**[*ilo*] ] [ **-b**[*s*]**I**[*D*][*ncol*] ]

**DESCRIPTION**

**originator** reads (longitude, latitude, height, radius, crustal\_age) records from *infiles* [or standard input] and uses the given Absolute Plate Motion (APM) stage poles and the list of hotspot locations to determine the most likely origin (hotspot) for each seamount. It does so by calculating flowlines back in time and determining the closest approach to all hotspots. The output consists of the input records with four additional fields added for each of the *n\_hs* closest hotspots. The four fields are the hotspot id (e.g., HWI), the stage id of the flowline segment that came closest, the pseudo-age of the seamount, and the closest distance to the hotspot (in km). See option **-:** on how to read (latitude, longitude, height, radius, crustal\_age) files.

No space between the option flag and the associated arguments. Use upper case for the option flags and lower case for modifiers.

*infile(s)* Seamount data file(s) to be analyzed. If not given, standard input is read.

**-E** Give file with stage poles and opening angles. This file must contain one record for each stage; each record must be of the following format:

```
lon lat tstart tstop ccw-angle
```

where tstart and tstop are in Ma and the rest is in degrees. The oldest stage must be listed first. tstart (tstop) is the age of the old (young) end of the stage. Blank lines and records whose first column contains # will be ignored.

**-F** Give file with hotspot locations. This file must contain one record for each hotspot to be considered; each record must be of the following format:

```
lon lat hs_id hs_name
```

E.g., for Hawaii this may look like

```
205    20    HWI    Hawaii
```

Blank lines and records whose first column contains # will be ignored.

**OPTIONS**

- C** Expect Total Reconstruction Poles (Finite Poles) rather than Backward Stage Poles [Default] File format is similar to the stage pole format except the youngest rotation must be listed first and that the tstart column is optional (assumed to be 0 Ma).
- D** Sets the flowline sampling interval in km. [Default is 5].
- H** Input file(s) has Header record(s). Number of header records can be changed by editing your .gmtdefaults4 file. If used, **GMT** default is 1 header record. Use **-Hi** if only input data should have header records [Default will write out header records if the input data have them].
- N** Set the maximum age to extend the oldest stage back in time [no extension].
- S** Set the number of closest hotspots to report [Default is 1].
- T** Truncate seamount ages exceeding the upper age set with **-N** [no truncation].
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- W** Only report those seamounts whose flowlines came within *maxdist* to any hotspot [Default reports all seamounts].

- : Toggles between (longitude,latitude) and (latitude,longitude) input and/or output. [Default is (longitude,latitude)]. Append **i** to select input only or **o** to select output only. [Default affects both].
- bi** Selects binary input. Append **s** for single precision [Default is **d** (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of columns in your binary file if it exceeds the columns needed by the program. [Default is 5 input columns].

### EXAMPLES

To find the likely (hotspot) origins of the seamounts represented by the (x,y,z,r,tc) points in the file seamounts.d, using the DC85.d Euler poles and the pac\_hs.d list of possible hotspots, and report the 2 most likely hotspot candidates for each seamount, run

```
originator seamounts.d -S2 -EDC85.d -Fpac_hs.d > origins.d
```

### SEE ALSO

*GMT(1)*, *project(1)*, *mapproject(1)*, *backtracker(1)* *hotspotter(1)*

### REFERENCES

Wessel, P., 1999, "Hotspotting" tools released, EOS Trans. AGU, 80 (29), p. 319.

**NAME**

rotconverter – Manipulate finite and stage rotations

**SYNOPSIS**

**rotconverter** [ +/- ] *rotA* [ +/- *rotB* ] [ +/- *rotC* ] ... [ **-C[alt]** ] [ **-D** ] [ **-Fin/out** ] [ **-N** ] [ **-S** ] [ **-T** ] [ **-V** ]

**DESCRIPTION**

**rotconverter** reads one or more plate motion models (rotations) stored in the given files. If more than one plate motion model is given we will add or subtract them in the order they were listed. The minus sign means we should first transpose the rotation and then add it to the previous rotation. The input files must all be of the same type (stage poles or finite rotations) which may differ from the desired output format; see **-F**. If a file cannot be opened we will attempt to decode the file name as a single rotation whose parameters are separated by slashes.

No space between the option flag and the associated arguments. Use upper case for the option flags and lower case for modifiers.

*rotX* Name of a file with a plate motion model. Separate several files with desired operator (+ or -). The very first file may also have a leading minus to imply a transpose. If any of the specified rotation models cannot be opened as a file, we will try to decode the file name as *lon/lat/tstart/tstop/angle* for a single rotation given on the command line. The *tstop* argument is required for stage poles only. For a finite rotation without any time information, give *lon/lat/angle* only.

**OPTIONS**

- C** Write out a column header record indentifying the various columns [Default is no header record]. Append **a** to indicate opening angles and **t** to indicate opening rates [Default].
- D** Report longitudes use the -180/+180 range [Default is 0/360].
- F** Specify both the input and output format for rotations. The *in* and *out* flags must be either **f** or **s** for finite or stage rotations, respectively. Note that both must be specified if **-F** is set [Default is **-Fff** (both input and output are finite rotations)].
- N** Place all output poles in the northern hemisphere [Default reports positive rotation angles].
- S** Place all output poles in the southern hemisphere [Default reports positive rotation angles].
- T** Transpose the final result, i.e., change the sign of the rotation angles.
- V** Verbose option: Report statistics of extracted rotations [Default is silent].

**EXAMPLES**

To convert the finite rotations in the file *model\_finite.APM* to stage poles, run

```
rotconverter model_finite.APM -Ffs > model_stages.APM
```

To obtain Nazca motion relative to Pacific hotspots by adding the motion of Nazca relative to a fixed Pacific to the Pacific-Hotspot reference model *DC85\_stages.d*, and report the result as finite reconstruction poles in the northern hemisphere, try

```
rotconverter DC85_stages.APM + Pac_Naz_stages.RPM -N -Fsf > Naz_HS_finite.APM
```

To add the final rotations ROT(150.1, 70.5, -20.3) and ROT (145.0, 40.0, 11.4), try

```
rotconverter 150.1/70.5/-20.3 + 145/40/11.4
```

which prints out 157.32, -80.44, 11.97.

**SEE ALSO**

*backtracker*(1), *hotspotter*(1), *originator*(1)

**NAME**

`x2sys_binlist` – Create bin-index files from track data files

**SYNOPSIS**

`x2sys_binlist track(s) [ -TTAG ] [ -Cclflag ] [ -D ] [ -Nunit ] [ -V ]`

**DESCRIPTION**

`x2sys_binlist` reads one or more track data files and produces a multi-segment ASCII track bin-index file (tbf) with the track id in the header and one data record per bin crossed; these records contain *lon*, *lat*, *index*, *flags*[, *dist*], where *lon*, *lat* is the coordinates of the center of the bin, the *index* is the 1-D number of the bin, and *flags* is a bitflag that describes which data fields were available in each bin. The optional *dist* requires `-D`. The input files can be of any format, which must be described and passed with the `-T` option. The bin-index listing is a crude representation of where the track goes and is used by the data archivist to build a track data base for track queries such as to determine which tracks need to be compared for crossover analysis. You must run `x2ys_init` to initialize the tag before you can run the indexing.

*tracks* Can be one or more binary or ASCII data files. To give a list of names, use ‘cat tracks.lis’. If the names are missing the suffix we will append the suffix that goes with this *TAG*. Track files will be looked for first in the current directory and second in all directories listed in `$X2SYS_HOME/TAG_paths.txt` (if it exists). [If `$X2SYS_HOME` is not set it will default to `$GMTHOME/share/x2sys`].

**OPTIONS**

No space between the option flag and the associated arguments.

- `-C` Append a one-letter code to select the procedure for along-track distance calculation selected with `-D` (see `-N` for selecting units):
  - `c` Cartesian distances in user units [Default].
  - `f` Flat Earth distances.
  - `g` Great circle distances.
  - `e` Geodesic distances on current **GMT** ellipsoid.
- `-D` Calculate the length of trackline segments per bin [Default skips this step]. See `-C` for method and `-N` for units. The length fragments are given as the 5th output column (after the *flags*).
- `-N` Append the desired distance *unit* as `e` (meter), `k` (km), `m` (miles), or `n` (nautical miles). [Default is `-Nk` (km)]. Requires `-D`.
- `-T` Sets the data set *TAG* which among other things sets the definition file.
- `-V` Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].

**EXAMPLES**

To create a bin index table from the MGD77 file 01030061.mgd77 using a global, 1x1 degree bin system, do

```
x2sys_binlist 01030061.mgd77 -TMGD77 > 01030061.tbf
```

To create a track bin index file of all MGD77+ files residing in the current directory at a bin resolution of 30 x 30 minutes, calculating the geodesic distance lengths for each bin in km, run

```
x2sys_binlist *.nc -TMGD77+ -Ce -D > all.tbf
```

**SEE ALSO**

`gmtdefaults(1)` `x2sys_init(1)`

**NAME**

**x2sys\_cross** – Find and compute Cross-Over Errors

**SYNOPSIS**

```
x2sys_cross track(s) -TTAG [ -Celfile ] [ -Flags ] [ -Ilalc ] [ -Jparameters ] [ -Kcombi.lis ] [ -O ] [ -Ndlunit ] [ -Qeli ] [ -Slluspeed ] [ -V ] [ -Wtdlnsize ] [ -2 ] [ -bo[s|S|d|D][ncol] ]
```

**DESCRIPTION**

**x2sys\_cross** is used to determine all intersections between ("external cross-overs") or within ("internal cross-overs") tracks, and report the time, position, distance along track, heading and speed along each tracksegment, and the crossover and mean values for all observables. The names of the tracks are passed on the command line. By default, **x2sys\_cross** will look for both external and internal cross-overs. As an option, you may choose to project all data using one of the map-projections prior to calculating the crossovers.

*tracks* Can be one or more binary or ASCII data files. To give a list of names, use 'cat tracks.lis'. If the names are missing the suffix we will append the suffix that goes with this *TAG*. Track files will be looked for first in the current directory and second in all directories listed in **\$X2SYS\_HOME/TAG\_paths.txt** (if it exists). [If **\$X2SYS\_HOME** is not set it will default to **\$GMTHOME/share/x2sys**].

**-T** Sets the data set *TAG* which, among other things, sets the definition file.

**OPTIONS**

No space between the option flag and the associated arguments.

- C** Select procedure for along-track distance calculation:  
**c** Cartesian distances [Default].  
**f** Flat Earth distances.  
**g** Great circle distances.  
**e** Geodesic distances on current **GMT** ellipsoid.
- F** Sets which columns to use. Give a comma-separated list of column names; this list must include **x**, **y** or **lon**, **lat** and ideally **time** [Default selects all columns].
- I** Sets the interpolation mode. Choose among:  
**I** Linear interpolation [Default].  
**a** Akima spline interpolation.  
**c** Cubic spline interpolation.
- J** Selects the map projection. Scale is UNIT/degree, 1:xxxxx, or width in UNIT (upper case modifier). UNIT is cm, inch, or m, depending on the **MEASURE\_UNIT** setting in .gmtdefaults4, but this can be overridden on the command line by appending **c**, **i**, or **m** to the scale/width value. For map height, max dimension, or min dimension, append **h**, **+**, or **-** to the width, respectively. More details can be found in the **psbasemap** man pages.

**CYLINDRICAL PROJECTIONS:**

- Jclon0/lat0/scale** (Cassini)
- Jjlon0/scale** (Miller)
- Jmscale** (Mercator - Greenwich and Equator as origin)
- Jmlon0/lat0/scale** (Mercator - Give meridian and standard parallel)
- Joalon0/lat0/azimuth/scale** (Oblique Mercator - point and azimuth)
- Joblon0/lat0/lon1/lat1/scale** (Oblique Mercator - two points)
- Joclon0/lat0/lonp/latp/scale** (Oblique Mercator - point and pole)
- Jqlon0/scale** (Equidistant Cylindrical Projection (Plate Carree))
- Jtlon0/scale** (TM - Transverse Mercator, with Equator as y = 0)
- Jtlon0/lat0/scale** (TM - Transverse Mercator, set origin)
- Juzone/scale** (UTM - Universal Transverse Mercator)
- Jylon0/lats/scale** (Basic Cylindrical Projection)

**AZIMUTHAL PROJECTIONS:**

- Jal***lon0/lat0/scale* (Lambert)
- Jel***lon0/lat0/scale* (Equidistant)
- Jfl***lon0/lat0/horizon/scale* (Gnomonic)
- Jgl***lon0/lat0/scale* (Orthographic)
- Jsl***lon0/lat0/[slat]/scale* (General Stereographic)

**CONIC PROJECTIONS:**

- Jbl***lon0/lat0/lat1/lat2/scale* (Albers)
- Jdl***lon0/lat0/lat1/lat2/scale* (Equidistant)
- Jll***lon0/lat0/lat1/lat2/scale* (Lambert)

**MISCELLANEOUS PROJECTIONS:**

- Jhl***lon0/scale* (Hammer)
- Jil***lon0/scale* (Sinusoidal)
- Jk[fls]***lon0/scale* (Eckert IV (f) and VI (s))
- Jnl***lon0/scale* (Robinson)
- Jrl***lon0/scale* (Winkel Tripel)
- Jvl***lon0/scale* (Van der Grinten)
- Jwl***lon0/scale* (Mollweide)

**NON-GEOGRAPHICAL PROJECTIONS:**

- Jp[a]***scale[/origin][r|z]* (Polar coordinates (theta,r))
- Jxx-scale[dlllppow|t|T][y-scale[dlllppow|t|T]]** (Linear, log, and power scaling)
- K** Only process the pair-combinations found in the file *combi.lis* [Default process all possible combinations among the specified files].
- L** Output results using the old XOVER format [Default is X2SYS format]. This option should only be used with \*.gmt-formatted MGD77 files.
- N** Append **d** for distance or **s** for speed, then give the desired *unit* as **e** (meter or m/s), **k** (km or km/hr), **m** (miles or miles/hr), or **n** (nautical miles or knots). [Default is **-Ndk -Nse** (km and m/s)].
- Q** Append **e** for external crossovers only, and **i** for internal crossovers only [Default is all crossovers].
- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+ -]dd:mm[:ss.xxx][W|E|S|N] format. Append **r** if lower left and upper right map coordinates are given instead of wesn. The two shorthands **-Rg -Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).
- S** Defines window of track speeds. If speeds are outside this window we do not calculate a crossover. Specify
  - Sl** sets lower speed [Default is 0].
  - Su** sets upper speed [Default is Infinity].
- V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].
- W** Give **t**, **d**, or **n** and append the corresponding maximum time gap [Infinity], or distance gap [Infinity] (both in user units) allowed between the two points immediately on either side of a crossover. If the gap exceeds this maximum gap we skip the crossover calculation. For **-Wn**, give the maximum number of points on either side to use in the interpolation [3].
- 2** Report the values of each track at the crossover [Default reports the crossover value and the mean value].

- bo** Selects binary output. Append *s* for single precision [Default is *d* (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of desired columns in your binary output file.

## REMARKS

The COEs found are printed out to standard output in ASCII format (unless **-bo** is set). When ASCII is chosen, the output format depends on whether or not old-style XOVER output (**-L**) has been selected. If so, then the first record contains the file names and their start year, whereas subsequent records have the data for each COE encountered. The fields written out are lat, lon, time along track #1 and #2, *x\_gravity*, *x\_magnetics*, *x\_bathymetry*, average gravity, average magnetics, average bathymetry, heading along track #1 and #2. If the default format is chosen then the header record starts with "> ", and the output format for crossovers are first 10 columns that contain *x*, *y*, time along track #1 and #2, distance along track #1 and #2, heading along track #1 and #2, velocity along track #1 and #2, and then pairs of (COE, average value) for each data type (or track-values; see **-2**). It is recommended that the Akima spline is used instead of the natural cubic spline, since it is less sensitive to outliers that tend to introduce wild oscillations in the interpolation.

## SIGN CONVENTION

If *lega* and *legb* are passed on the command line, then the COE value is Value (*lega*) - Value (*legb*).

## EXAMPLES

To compute all internal crossovers in the gmt-formatted file *c2104.gmt*, and output in the old XOVER format, using the tag *MGG*, try

```
x2sys_cross c2104.gmt -L -TMGG > c2104.d
```

To find the crossover locations with bathymetry between the two *MGD77* files *A13232.mgd77* and *A99938.mgd77*, using the *MGD77* tag, try

```
x2sys_cross A13232.mgd77 A99938.mgd77 -Qe -TNGD77 -Flon,lat,depth > crossovers.d
```

## SEE ALSO

*GMT*(1), *x2sys\_init*(1) *x2sys\_datalist*(1) *x2sys\_get*(1) *x2sys\_put*(1)

**NAME**

`x2sys_datalist` – A generic data-extractor for ASCII or binary files

**SYNOPSIS**

```
x2sys_datalist track(s) -TTAG [ -Fname1,name2,... ] [ -H[i][nrec] ] [ -Rwest/east/south/north[r] ] [ -S ]
[ -V ] [ -bo[s|l|d|D][ncol] ]
```

**DESCRIPTION**

**x2sys\_datalist** reads one or more files and produces a single ASCII [or binary] table. The files can be of any format, which must be described and passed with the **-T** option. You may limit the output to a geographic region, and insist that the output from several files be separated by a multiple segment header. Only the named data fields will be output [Default selects all columns].

*tracks* Can be one or more binary or ASCII data files. To give a list of names, use ‘cat tracks.lis’. If the names are missing the suffix we will append the suffix that goes with this *TAG*. Track files will be looked for first in the current directory and second in all directories listed in `$X2SYS_HOME/TAG_paths.txt` (if it exists). [If `$X2SYS_HOME` is not set it will default to `$GMTHOME/share/x2sys`].

**-T** Sets the data set *TAG* which among other things sets the definition file.

**OPTIONS**

No space between the option flag and the associated arguments.

**-F** Give a comma-separated sub-set list of column names defined in the definition file. [Default selects all data columns].

**-H** Input file(s) has Header record(s). Number of header records can be changed by editing your `.gmtdefaults4` file. If used, **GMT** default is 1 header record. Use **-Hi** if only input data should have header records [Default will write out header records if the input data have them].

**-M** Output a multisegment header between data from each track. Note this option does not imply anything about the input file; that information is conveyed via the system tag (**-T**).

**-S** Suppress output records where all the data columns are NaN [Default will output all records].

**-V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].

**-bo** Selects binary output. Append *s* for single precision [Default is *d* (double)]. Uppercase **S** (or **D**) will force byte-swapping. Optionally, append *ncol*, the number of desired columns in your binary output file.

**EXAMPLES**

To extract all data from the old-style MGG supplement file `c2104.gmt`, recognized by the tag `MGG`:

```
x2sys_datalist c2104.gmt -Tmgg > myfile
```

To make lon,lat, and depth input for **blockmean** and **surface** using all the files listed in the file `tracks.lis` and define by the tag `TRK`, but only the data that are inside the specified area, and make output binary, run

```
x2sys_datalist ‘cat tracks.lis’ -TTRK -Flon,lat,depth -R-40/-30/25/35 -bo > alltopo.xyz
```

**SEE ALSO**

*blockmean(1)*, *GMT(1)*, *surface(1)*, *x2sys\_cross(1)*

**NAME**

`x2sys_get` – Get track listing from track index databases

**SYNOPSIS**

`x2sys_get -TTAG [ -C ] [ -Fflags ] [ -L[list] [ -Nflags ] [ -Rwest/east/south/north[r] ] [ -V ] ]`

**DESCRIPTION**

`x2sys_get` will return the names of the track data files that match the given requirements. You may choose a geographic region and optionally ask only for tracks that meet certain data criteria. Finally, you may select an option to list all possible pairs that might generate crossovers.

*TAG* The unique tag name of this data type system.

**OPTIONS**

No space between the option flag and the associated arguments.

- C** Instead of reporting the track names, just output the coordinates of the center of each bin that has at least one track with the specified data.
- F** Give a comma-separated list of column names (as described in the definition file) that should be present. [Default selects all data columns].
- L** Crossover mode. Return a list of track pairs that should be checked for possible crossovers. The list is determined from the bin-index data base on the assumption that tracks occupying the same bin are very likely to intersect. By default we return all possible pairs in the data base. Append the name of a file with a list of tracks if you want to limit the output to those pairs that involve at least one of the track names in your list. The output is suitable for the **-K** option in `x2sys_cross`.
- N** Give a comma-separated list of column names (as described in the definition file) that must be absent.
- R** *west, east, south, and north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd:mm[:ss.xxx]*[W|E|S|N] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg -Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).
- V** Selects verbose mode, which will send progress reports to `stderr` [Default runs "silently"].

**EXAMPLES**

To find all the tracks associated with the tag `mgd77`, restricted to occupy a certain region in the south Pacific, and have at least free air anomalies and bathymetry, try

```
x2sys_get -V -Tmgd77 -R180/240/-60/-30 -Ffaa,depth
```

To find all the tracks associated with the tag `mgd77` that have depth but not `tw`, try

```
x2sys_get -V -Tmgd77 -Fdepth -Ntw
```

To find all the pairs associated with the tag `mgd77` that might intersect each other, but only those pairs which involves tracks in your list `new.lis`, try

```
x2sys_get -V -Tmgd77 -Lnew.lis > xpairs.lis
```

**SEE ALSO**

`x2sys_binlist(1)` `x2sys_cross(1)` `x2sys_datalist(1)` `x2sys_init(1)` `x2sys_put(1)`

**NAME**

`x2sys_init` – Initialize data base for track data files

**SYNOPSIS**

```
x2sys_init TAG -Ddefile [ -Esuffix ] [ -F ] [ -Gdlg ] [ -Idx[/dy] ] [ -M[ilo][flag] ] [
-Rwest/east/south/north[r] ] [ -V ] [ -Wtdgap ]
```

**DESCRIPTION**

`x2sys_init` initializes a set of data bases that are particular to one kind of track data. These data, their associated data bases, and key parameters are given a short-hand notation called a system TAG. The TAG remembers settings such as file format, whether the data are geographic or not, and the binning resolution for track indexes. Running `x2sys_init` is a prerequisite to running `x2sys_binlist` which will create a crude representation of where data track go and which observations are available; this information serves as input to `x2sys_put` which updates the track data base. Then, `x2sys_get` can be used to find which tracks and data are available inside a given region.

*TAG* The unique name of this data type system.

**-D** Definition file for this data set.

**OPTIONS**

No space between the option flag and the associated arguments.

**-E** Specifies the file extension (suffix) for these data files. If not given we use the definition file prefix as the suffix (see **-D**).

**-F** Force creating new files if old ones are present [Default will abort if old files are found].

**-G** Selects geographical coordinates. Append **d** for discontinuity at the Dateline (makes longitude go from -180 to + 180) or **g** for discontinuity at Greenwich (makes longitude go from 0 to 360 [Default]).

**-I** *x\_inc* [and optionally *y\_inc*] is the grid spacing. Append **m** to indicate minutes or **c** to indicate seconds.

**-M** Multiple segment file(s). Segments are separated by a special record. For ASCII files the first character must be *flag* [Default is '>']. For binary files all fields must be NaN.

**-R** *west*, *east*, *south*, and *north* specify the Region of interest, and you may specify them in decimal degrees or in [+]*dd*:*mm*[:*ss*.*xxx*][*W*|*E*|*S*|*N*] format. Append **r** if lower left and upper right map coordinates are given instead of *wesn*. The two shorthands **-Rg** **-Rd** stand for global domain (0/360 or -180/+180 in longitude respectively, with -90/+90 in latitude).

**-V** Selects verbose mode, which will send progress reports to *stderr* [Default runs "silently"].

**-W** Give **t** or **d** and append the corresponding maximum time gap (in user units; this is typically seconds [Infinity]), or distance gap (in user units; this is typically km [Infinity] ) allowed between the two points immediately on either side of a crossover.

**DEFINITION FILES**

The \*.def files contain information about the data file format. The \*.def files have two sections: (1) header information and (2) column information. All header information starts with the character # in the first column, immediately followed by an upper-case directive. If the directive takes an argument it is separated by white-space. You may append a trailing # comments. Five directives are recognized:

**ASCII** states that the data files are in ASCII format.

**BINARY** states that the data files are binary files.

**SKIP** takes an integer argument which is either the number of lines to skip (when reading ASCII files) or the number of bytes to skip (when reading binary files).

**GEO** indicates that these are geographic data set, with periodicities in the *x*-coordinate (longitudes).

**MULTISEG** means each track consists of multiple segments separated by a **GMT** multisegment header (alternatively, use **-M** when defining the system TAG).

The column information consists of one line per column in the order the columns appear in the data file. For each column you must provide seven attributes:

*name type NaN NaN-proxy scale offset oformat*

*name* is the name of the column variable. It is expected that you will use the special names *lon* (or *x* if Cartesian) and *lat* (or *y*) for the two required coordinate columns, and *time* when optional time data are present.

*type* is **a** for ASCII representations of numbers, whereas for binary files you may choose from **c** for signed 1-byte character (-127,+128), **u** for unsigned byte (0-255), **h** for signed 2-byte integers (-32768,+32767), **i** for signed 4-byte integers (-2,147,483,648,+2,147,483,647), **f** for 4-byte floating points and **d** for 8-byte double precision floating points.

*NaN* is **Y** if certain values are to be replaced by NAN, and **N** otherwise.

*NaN-proxy* is that special value.

*scale* is used to multiply the data after reading.

*offset* is used to add to the scaled data.

*oformat* is a C-style format string used to print values from this column.

For ASCII data the *type* format is always **a**. *NaN* should be **Y** if there is a certain value (e.g., -9999) that is used to represent missing data (NaNs). If so, the *NaN-proxy* value should be set accordingly (otherwise it is ignored). The *scale* and *offset* are applied to the data values after they are read (first scaled, then offset), whereas the *oformat* is used by some programs for formatted output. If you give - as the *oformat* then GMT's formatting machinery will be used instead (i.e., **PLOT\_DEGREE\_FORMAT**, **PLOT\_DATE\_FORMAT**, **PLOT\_CLOCK\_FORMAT**). Some file formats already have definition files premade. These include *mgd77* (for plain ASCII MGD77 data files), *mgd77+* (for enhanced MGD77+ netCDF files), *gmt* (for old *mgg* supplement binary files), *xy* (for plain ASCII *x*, *y* tables), *xyz* (same, with *z*-column), *geo* (for plain ASCII longitude, latitude files), and *geoz* (same, with *z*-column).

## EXAMPLES

If you have a large set of track data files you can organize them using the *x2sys* tools. Here we will outline the steps. Let us assume that your track data file format consist of 2 header records with text information followed by any number of identically formatted data records with 6 columns (*lat*, *lon*, *time*, *obs1*, *obs2*, *obs3*) and that files are called \*.trk. We will call this the "line" format. First, we create the *line.def* file:

```
# Define file for the line format
#ASCII          # File is ASCII
#SKIP 2        # Skip 2 header records
#name  type    NaN   NaN-proxy  scale offset oformat
lon    a       N     0         1     0      %10.5f
lat    a       N     0         1     0      %9.5f
time   a       N     0         1     0      %7.1f
obs1   a       N     0         1     0      %7.2f
obs2   a       N     0         1     0      %7.2f
obs3   a       N     0         1     0      %7.2f
```

Next we create the TAG and the databases for these line track files. Assuming these contain geographic data and that we want to keep track of the data distribution at a 1 x 1 degree resolution, we may run

```
x2sys_init LINE -V -G -Dline -R0/360/-90/90 -I1/1 -Etrk
```

where we have used *LINE* as our system tag. When *x2sys* tools try to read your line data files they will first look in the current directory and second look in the file *TAG\_paths.txt* for a list of additional directories to examine. Therefore, create such a file (here *LINE\_paths.txt*) and stick the full paths to your data directories there. All TAG-related files (definition files, tag files, and track data bases built) will be expected to be in the directory pointed to by **\$X2SYS\_HOME**.

**Create tbf file(s):**

Once the (empty) TAG database have been initialized we go through a two-step process to populate them. First we run **x2sys\_binlist** on all our track files to create one (or more) multi-segment track bin-index files (tbf). These contain information on which 1 x 1 degree bins (or any other blocksize; see **-I**) each track has visited and which observations (in your case obs1, obs2, obs3) were actually observed (not all tracks may have all three kinds of observations everywhere). For instance, if your tracks are listed in the file tracks.lis we may run this command:

```
x2sys_binlist -V -TLINE 'cat tracks.lis' > tracks.tbf
```

**Update index data base:**

Next, the track bin-index files are fed to **x2sys\_put** which will insert the information into the TAG databases:

```
x2sys_put -V -TLINE tracks.tbf
```

**Search for data:**

You may now use **x2sys\_get** to find all the tracks within a certain sub-region, and optionally limit the search to those tracks that have a particular combination of observables. E.g., to find all the tracks which has both obs1 and obs3 inside the specified region, run

```
x2sys_get -V -TLINE -R20/40/-40/-20 -Fobs1,obs3 > tracks.tbf
```

**MGD77[+] or GMT:**

Definition files already exist for MGD77 files (both standard ASCII and enhanced netCDF-based MGD77+ files), the old \*.gmt files manipulated by the mgg supplements, and plain x,y[,z] and lon,lat[,z] tracks. To initiate new track databases to be used with MGD77 data from NGDC, try

```
x2sys_init MGD77 -V -Dmgd77 -Emgd77 -Rd -Gd -I1/1 -Wt900 -Wd5
```

where we have chosen a 15 minute (900 sec) or 5 km threshold to indicate a data gap; the other steps are similar.

**Binary files:**

Let us pretend that your line files actually are binary files with a 128-byte header structure (to be skipped) followed by the data records and where *lon*, *lat*, *time* are double precision numbers while the three observations are 2-byte integers which must be multiplied by 0.1. Finally, the first two observations may be -32768 which means there is no data available. All that is needed is a different line.def file:

```
# Define file for the binary line format
#BINARY      # File is now binary
#SKIP 128    # Skip 128 bytes
#name  type   NaN?  NaN-proxy  scale offset offormat
lon    d      N      0          1          0          %10.5f
lat    d      N      0          1          0          %9.5f
time   d      N      0          1          0          %7.1f
obs1   h      Y     -32768    0.1        0          %6.1f
obs2   h      Y     -32768    0.1        0          %6.1f
obs3   h      N      0          0.1        0          %6.1f
```

The rest of the steps are identical.

**SEE ALSO**

*x2sys\_binlist(1)* *x2sys\_get(1)* *x2sys\_put(1)*

**NAME**

`x2sys_put` – Update track data bases

**SYNOPSIS**

`x2sys_put` [ *info.tbf* ] **-TTAG** [ **-V** ]

**DESCRIPTION**

`x2sys_put` accepts a track bin-index file created by `x2sys_binlist` and adds this information about the data tracks to the relevant data base. The system *TAG* must match the tag encoded in the *info.tbf* file. To inquire about tracks in the data base, use `x2sys_get`.

**-T** Give *TAG*, the unique name of this data type system.

**OPTIONS**

No space between the option flag and the associated arguments.

*info.tbf* Name of a single track bin file. If not given, stdin will be read.

**-V** Selects verbose mode, which will send progress reports to stderr [Default runs "silently"].

**EXAMPLES**

To add the information stored in the track bin-index file *latest.tbf* to the track data bases associated with the tag *MGD77*, try

```
x2sys_put latest.tbf -V -TMGD77
```

**SEE ALSO**

`x2sys_binlist(1)` `x2sys_get(1)`

**NAME**

**x\_edit** – convert between binary and ASCII crossover correction tables.

**SYNOPSIS**

**x\_edit** **-A**[*X*]*[name]* **-O**[*outname*]

**DESCRIPTION**

The purpose of **x\_edit** is to convert between ASCII and binary versions of the crossover correction tables.

**-A** Name of ASCII correction table [stdin], write binary table to file given by **-O** [stdout].

**-X** Name of binary correction table [stdin], write ASCII table to file given by **-O** [stdout].

**-O** Name of output correction table [stdout].

**EXAMPLES**

To convert the binary file `crossover_corr.b` to ASCII and write to stdout, run

```
x_edit -Xcrossover_corr.b > corrections.d
```

**SEE ALSO**

*GMT*(1), *x\_system*(1)

**REFERENCES**

Wessel, P. XOVER: A Cross-over Error Detector for Track Data, *Computers & Geosciences*, 15, 333-346.

**NAME**

**x\_init** – Initialization of Cross-Over Data Bases.

**SYNOPSIS**

**x\_init -I**

**DESCRIPTION**

The purpose of **x\_init** is to create two key files that will be holding all the cross-over information for all the cruises tested. These files are called *xx\_base.b* and *xx\_legs.b*, and they will contain data for each cross-over and totals for each leg, respectively. **x\_init** will create these two files in the current directory, and write out the fileheader with a 1 as the next record number in the case of *xx\_base.b*. NB! Any previous files with the same names in the current directory will be erased, so use with caution.

**-I** Must be present form initialization to take effect.

**BEWARE**

Both files are binary and have fixed record lengths. The formats of the files are as follows:

**xx\_base.b:**

Record length = 40 bytes.  
Rec # 1: fileheader  
Rec # 2: pairheader  
Next n recs: crossover-structure  
Rec # n+3: pairheader etc.

The fileheader is a 40 byte character string where the last 10 bytes contains the record number of the last record in the file. The pairheader is a 40 byte character string which contains the names of the two legs in question and the number of COEs generated between them. The crossover-structure contains all the information for one COE, that is lat, lon, time along leg values of gravity, magnetics, and bathymetry at the cross-over point, the heading of track #1, and the heading of track #2. See program listing for more details on formats. The file is sequential in that new data are appended at the end of the file.

**xx\_legs.b:**

Record length = 204 bytes.

Each record contains the complete information for each leg that has been checked. The leg-structure contains fields like year of cruise, number of internal/external COEs for gravity, magnetics, and bathymetry, their means and standard deviations, and the best fitting regression lines, i.e. the dc-shifts and drift-rates computed from the time/error data points. See program listing for complete description of record format.

**SEE ALSO**

*GMT(1)*, *x\_system(1)*

**REFERENCES**

Wessel, P. XOVER: A Cross-over Error Detector for Track Data, *Computers & Geosciences*, 15, 333-346.

**NAME**

**x\_list** – Extract Cross-Over Information from the Data Base.

**SYNOPSIS**

**x\_list** [ *leg1* ] [ *leg2* ] [ *-dataflags* ] [ *-Rwest/east/south/north* ] [ *-Xxbase* ] [ *-H* ] [ *-I* ] [ *-V* ] [ *-Ccorrfile* ] [ *-Zignorefile* ]

**DESCRIPTION**

**x\_list** will allow the user to extract information from the *xx\_base.b* and *xx\_legs.b* files. Several options can be passed on the command line in order to specify which parameters to extract. They are:

*leg1 leg2*

If two legs are specified, the cross-overs they have in common are returned. If one leg is specified, all the external cross-overs involving this leg are returned, OR only its internal cross-overs if *-I* is selected. If no legs are given, all the external (or internal with *-I*) cross-overs are returned. This program is useful if one wants to look at, say, gravity COEs versus time from port for a particular leg, etc.

*dataflags*

This is a string of any combination of the following:

**l** means list both legnames.

**t** means list time.

**x** means list longitude.

**y** means list latitude.

**g** means list gravity COEs.

**m** means list magnetics COEs.

**b** means list bathymetry COEs.

**G** means list average gravity at cross-over.

**M** means list average magnetic anomaly at cross-over.

**B** means list average bathymetry at cross-over.

**h** means list heading of ship at cross-over.

The components are written out in the order they appear in *dataflags*. The default output is **-txygmBGMbhl**. When internal COEs are desired, the time reported is the elapsed time since the ship first occupied the cross-over point. For external COEs the time means time from the start of the year in seconds.

- R** Only return cross-overs inside the specified region *west*, *east*, *south*, and *north*. [Default is world].
- X** Indicate alternate *xx\_base.b* file.
- H** Issue one header record on output.
- I** Report on internal COEs [Default is external COEs].
- C** Apply cross-over correction to the data (i.e. reports the cross-over value after the best-fitting corrections have been applied to both legs). If no *corrfile* is given, the default correction file is assumed.
- Z** Ignore those legs that appear in the *ignorefile*.
- V** Operate in verbose mode.

**SEE ALSO**

*GMT*(1), *x\_system*(1)

**REFERENCES**

Wessel, P. XOVER: A Cross-over Error Detector for Track Data, *Computers & Geosciences*, 15, 333-346.

**NAME**

**x\_over** – Find and compute Cross-Over Errors

**SYNOPSIS**

**x\_over** *leg\_1* [ *leg\_2* ] [ **-A** ] [ **-C** ] [ **-L** ] [ **-V** ] [ **-W***timegap* ] [ **-G***fact* ] [ **-M***fact* ] [ **-T***fact* ] [ **-N***np\_int* ]

**DESCRIPTION**

**x\_over** is used to inspect two cruises to see if they intersect, and if so report the time, position, discrepancies in gravity/magnetics/bathymetry, heading for each track segment, and the average values of the geophysical observables at the cross-over point. The names of the legs are passed on the command line. If they are identical or only one name is passed, then **x\_over** looks for internal cross-overs. The optional parameters are:

- A** Use an Akima spline to interpolate the geophysical field at the cross-over point.
- C** Use a Natural Cubic spline function instead.
- L** Use a linear interpolant [Default].
- W** Do not compute cross-overs if the 2 nearest points are more than *timegap* minutes apart.
- G** Scale gravity by *fact* [Default is 0.1 since gmt-files store gravity in g.u.]
- M** Scale magnetic anomaly by *fact* [1.0].
- T** Scale bathymetry by *fact* [1.0].
- N** Specify how many points to use in the interpolation [Default is 6].
- V** Verbose, report the number of cross-overs for this pair of legs.

**BEWARE**

The COEs found are printed out to standard output in ASCII format. The first record contains the leg names and their start year, whereas subsequent records have the data for each COE encountered. The fields written out are lat, lon, time along track #1, time along track #2, x\_gravity, x\_magnetics, x\_bathymetry, average gravity, average magnetics, average bathymetry, heading along track #1, and heading along track #2. Sign convention: If *lega* and *legb* are passed on the command line, then the COE value is Value (*lega*) - Value (*legb*). It is recommended that the Akima spline is used instead of the natural cubic spline, since it is less sensitive to outliers that tend to introduce wild oscillations in the interpolation.

**SEE ALSO**

*GMT*(1), *x\_system*(1)

**REFERENCES**

Wessel, P. XOVER: A Cross-over Error Detector for Track Data, *Computers & Geosciences*, 15, 333-346.

**NAME**

**x\_remove** – Remove COEs for Selected Cruises from the Data Base

**SYNOPSIS**

**x\_remove** *removefile* [ **-X***xbase* ] [ **-L***legbase* ] [ **-V** ]

**DESCRIPTION**

**x\_remove** will read a list of 'bad' cruises, read the **XSYSTEM** data base files, and write out new **XSYSTEM** files that do not include any COEs generated by the bad cruises. This is useful if certain cruises turn out to have systematic errors (e.g. the topography values were inadvertently given as fathoms instead of meters), and we would like to re-run **x\_over/x\_update** after the systematic errors have been rectified. The new file names will consist of the old file names with the suffix *\_new* appended.

**-X** Indicates alternate xbase file [Default is *xx\_base.b*].

**-L** Indicates alternate legbase file [Default is *xx\_legs.b*].

**-V** Means operate in verbose mode.

**SEE ALSO**

*GMT*(1), *x\_system*(1)

**REFERENCES**

Wessel, P. XOVER: A Cross-over Error Detector for Track Data, *Computers & Geosciences*, 15, 333-346.

**NAME**

**x\_report** – Report Cross-Over Error Statistics for Cruises

**SYNOPSIS**

**x\_report** [ *leg* ] [ **-X***legbase* ] [ **-G -M -T** ]

**DESCRIPTION**

**x\_report** reads the *x\_system* data base file and prints out statistics for each leg. The statistics are the number of, means, and standard deviations of COEs for both the internal and external cases. One or more data types (GMT) may be specified.

*leg* Report on *leg* only. [Default is all legs].

**-X** Specify alternate leg data base [Default is *xx\_legs.b*].

**-G** Give information for Gravity cross-overs only. [Default is G, M, and T].

**-M** Give information for Magnetics cross-overs only. [Default is G, M, and T].

**-T** Give information for Topography cross-overs only. [Default is G, M, and T].

**SEE ALSO**

*GMT*(1), *x\_system*(1)

**REFERENCES**

Wessel, P. XOVER: A Cross-over Error Detector for Track Data, *Computers & Geosciences*, 15, 333-346.

**NAME**

**x\_setup** – Determine Pairs of Legs that need Cross-Over Checking

**SYNOPSIS**

**x\_setup** [ **-L***legfile* ] [ **-R***west/east/south/north* ]

**DESCRIPTION**

**x\_setup** scans the *gmtindex.b* file that contains information on which legs occupy each 1 by 1 degree bin and reports all the pairs of legs that may overlap. Each pair is only reported once, even if the two legs involved are found in other bins also.

**-L** Only report pairs where at least one of the legs are present in the *legfile*.

**-R** The region of interest. [Default is world]

**BEWARE**

The **-L** option is useful when one wants to check out a handful of new cruises. Obviously we only need to compute COEs for the pairs that involve one or two of the new cruises, since all the other combinations have already been done.

**SEE ALSO**

*GMT(1)*, *x\_system(1)*

**REFERENCES**

Wessel, P. XOVER: A Cross-over Error Detector for Track Data, *Computers & Geosciences*, 15, 333-346.

**NAME**

**x\_solve\_dc\_drift** – Find Best-fitting D.C.-shift and Drift Estimates

**SYNOPSIS**

**x\_solve\_dc\_drift** [ **-X***ibase* ] [ **-L***ibase* ] [ **-I***ignorefile* ] [ **-S** ] [ **-C***iteration* ] [ **-M***min\_nx* ] [ **-R***west/east/south/north* ] [ **-U***uselegsfile* ] [ **-V** ] [ **-B***binfile* ] [ **-A***ASCIIfile* ]

**DESCRIPTION**

**x\_solve\_dc\_drift** will read the database files and, by iterating, find the best-fitting regression line to the <time,COE> points for each leg that minimizes the overall standard deviation of the data set in a least squares sense. Finally, correction file(s) are created.

- X** Indicate an alternate database to read. [Default is *xx\_base.b*].
- L** Indicate an alternate legbase to read. [Default is *xx\_legs.b*].
- I** Ignore information for certain legs (that might be bad, etc).
- S** Reset the old d.c.-shift and drift estimates to zero before iterating.
- C** Specify how many iterations to do. Default is interactive session.
- M** Solve for drift only if the leg has more than *min\_nx* cross-overs.
- R** Only take COEs inside this region into account. [Default is world].
- U** Solve for corrections involving COEs from legs in the *uselegsfile* only.
- B** Create a binary correction file (which can be read by **x\_list** and **gmtlist**).
- A** Create an ASCII correction table. At least one of **-A** and **-B** must be specified.
- V** Operate in verbose mode.

**BEWARE**

The reason for the **-M** option is that if the COEs are few, then a stable estimate of the slope may not be obtained (e.g. when most of the COEs occur midway between ports). However, in most cases the bulk of the COEs do occur near the ports so that a drift estimate can be computed. Conventional wisdom recommends plotting the time-series and the computed regression line to see if it makes sense.

**SEE ALSO**

*GMT*(1), *x\_system*(1)

**REFERENCES**

Wessel, P. XOVER: A Cross-over Error Detector for Track Data, *Computers & Geosciences*, 15, 333-346.

**NAME**

**x\_system** – A Cross-Over Error Analysis Tool

**INTRODUCTION**

The **x\_system** was developed to aid in the task of gridding geophysical track data, e.g. gravity, magnetics, or bathymetry. It has long been recognized that although the data quality along track may be quite good, one usually finds discrepancies at the points where two tracks intersect. These cross-over errors (COE) can be large enough to cause artificial features in the final gridded dataset, which would render geological interpretations of such a map questionable. Also, notoriously bad cruises will generate high COEs along their tracks, and should ideally be removed from the data base before gridding is attempted. The reasons why COEs arise are many and will not be dealt with here. Although originally intended to be used for marine gravity data only, **x\_system** has been designed to handle magnetics and bathymetry as well. (For an overview of gravity COEs, see Wessel and Watts [1988]). In most cases, marine gravity COEs can be explained by a simple model having only 2 parameters. These are a d.c.-shift and a drift-rate that apply for the duration of the cruise. The goal of the COE analysis is thus to determine the dc-shifts and drift-rates for each leg that will minimize the COEs in a least squares sense, and at the same time flag cruises that exhibit unreasonably high COEs (even after correction for d.c.-shift/drift). Furthermore, we can also assign a 'quality index' for each cruise by looking at the standard deviation of the COEs. The d.c.-shift/drift rate model may not be as meaningful for magnetics and bathymetry as it is for gravity. However, looking for high COEs is still one of the best ways of identifying systematic errors in the magnetic/bathymetric data sets.

**x\_system PHILOSOPHY**

Since the d.c.-shift/drift corrections for a given cruise depend entirely on the values of the COEs generated at intersections with other cruises, there is no such thing as a 'final correction' as long as we keep on adding data to the data base. This means that the system must be able to incorporate new data and compute a new set of d.c.-shifts/drift-rates that takes the new COEs into account. **x\_system** is made modular so that one program computes the actual COEs, one program archives the COE information, and the remaining programs do various tasks like reporting statistics (to flag bad cruises), extracting a subset of the COE database, and solving for the best fitting d.c.-shift/drift corrections. This way only the new COEs generated need to be computed and added to the database before a new correction solution is sought.

All the 8 programs that make up the **x\_system** package have been written in the C programming language and are intended to be run on a UNIX machine. Thus, it is assumed that the user has access to UNIX tools like *awk*, *grep*, and *sort*, and that the operating system provides a means for redirecting input/output. Likewise, it is assumed that all the geophysical data are stored in the GMT-format as outlined in the **GMT MGG** supplements man pages, and that the 1 by 1 degree bin information files (*gmtindex.b* and *gmtlegs.b*) have been created and are being maintained by the database librarian.

**HOW TO DO IT**

To illustrate how one would set things up, we will go through the necessary steps and point out usage, useful tricks, and pitfalls. (A more complete description of what exactly each program does can be found in the man pages for each program). We will assume that we initially have N cruises in our GMT data bank, and that we just have received the **x\_system** package. The first thing to do is to run **x\_init** which will create an empty data base system. This will normally be done only once. With N cruises on our hands we will in the worst case have to compare the  $N*(N+1)/2$  possible pairs. This is where **x\_setup** comes in handy. It will read the 1 by 1 degree bin information files and print out a list of pairs that need to be checked. The two cruises that make up a pair will at least once occupy the same 1 by 1 degree bin, and may thus intersect. Those combinations which do not have any bins in common obviously don't have to be checked. Let's call this list of pairs *xpairs.lis*.

**x\_over** is the main program in the package as it is responsible for locating and computing the COEs. For details on algorithm, see Wessel [1989]. It takes two cruise names as arguments and writes out all the COEs generated between them (if any). Since *xpairs.lis* may contain quite a few pairs, the most efficient way of running **x\_over** is to create an executable command (batch) file that starts **x\_over** for each pair. Using *awk* to do this, we would say:

```
pratt% awk '{ printf "x_over -<options> %s %s\n", $1, $2}' xpairs.lis > xjob
pratt% chmod +x xjob      (make it executable)
```

```
pratt% xjob > xjob.d &
```

and relax while xjob is crunching the numbers. This is the time-consuming part of the COE analysis, and on a SUN-3 computer with Floating Point Accelerator installed we average about 10,000 pairs of cruises/day. It may pay off to split a huge xjob file into smaller parts, and call the output files xjob.d1, xjob.d2 etc. Most of the run-time is taken up by reading the GMT files; when in memory the actual computations are remarkably fast. The output file xjob.d will now have all the COE information in ASCII form. For each pair of legs there will be a header record stating the names of the cruises and their starting years. The following records up to the next header record (or End-Of-File) will contain lat, lon, time, value, etc. for each COE found. This is a temporary file, but it is wise to back it up to tape just in case.

When the **x\_over** part is done, time has come to archive the data more efficiently than ASCII files. This is done by **x\_update** which rearranges the data and updates the binary data base system. After this step the xjob.d files can be deleted (presuming they have been backed up to tape). At this stage we have several options available. We can list some of the COEs by running **x\_list**, which will extract COEs that match the options we pass, e.g. we might ask for all the internal COEs for cruise c2104, and only print out time and gravity COE. See the man pages for more details. **x\_report** can be run, and will output statistics for separate cruises, i.e. mean and standard deviation of the COEs for different data sets (gravity/magnetics/bathymetry). To solve for the best fitting corrections we would run **x\_solve\_dc\_drift**. This program will solve for the d.c.-shift/drift-rates for all cruises, update that information in the data base system, and create correction tables (ASCII and/or binary). We have now completed the COE analysis for our initial GMT data bank.

At some later time, however, we will get a new batch of cruises. We will then follow the the same recipe and go back and run **x\_setup**, but this time we will use the **-L** option so that only the pairs involving new cruises are returned. Then we would run the remaining programs exactly as described above.

## SEE ALSO

*GMT(1)*,

## AUTHOR

Paul Wessel, Lamont-Doherty Geological Observatory, Palisades, NY 10964. Wessel, P. XOVER: A Cross-over Error Detector for Track Data, *Computers & Geosciences*, 15, 333-346.

Wessel, P. and A. B. Watts, On the Accuracy of Marine Gravity Measurements, *J. Geophys. Res.*, 93, 393-413, 1988.

**NAME**

**x\_update** – Archiving of Cross-Over Information

**SYNOPSIS**

**x\_update** *cross-overfile* [ **-X***xbase* ] [ **-L***legbase* ] [ **-V** ] [ **-W***max* ]

**DESCRIPTION**

**x\_update** reads the ASCII file produced by **x\_over** and writes the information to the database files *xx\_base.b* and *xx\_legs.b* (unless other filenames are specified on the command line) using a compact, binary format. The options recognized are:

- X** Indicate alternate *xx\_base.b* file.
- L** Indicate alternate *xx\_legs.b* file.
- V** Verbose. Reports the pair currently being processed.
- W** Prints a warning if the number of COEs generated by any pair of legs exceed *max*.

**BEWARE**

For security reasons the *xx\_legs.b* are first renamed to *xx\_legs.b\_old*, to prevent loss of information if a black-out or system crash should occur during writing of the new *xx\_legs.b* file. It is recommended that the old file is left on the disk in case of other disasters. Note that **x\_update** does not check if a pair of legs already exist in the *xx\_base.b* file. It is the user's responsibility to ensure that duplication of information does not occur. Should some pairs already in the database need to be re-examined, then run **x\_remove** which will wipe out all traces of the specified cruises from the *x\_system* data base files. Now we can pass the pairs that need to be redone through **x\_over/x\_update** again.

**SEE ALSO**

*GMT*(1), *x\_system*(1)

**REFERENCES**

Wessel, P. XOVER: A Cross-over Error Detector for Track Data, *Computers & Geosciences*, 15, 333-346.