

Example:

```

2  0 -4.8416902576341E-04  0.00000000000000E+00
2  1 -2.6064871543930E-10  1.4150696890028E-09
2  2  2.4393057664421E-06 -1.4002827713200E-06
3  0  9.5709933625445E-07  0.00000000000000E+00

```

The binary format of the potential coefficients speeds up the reading. To convert a file with spherical harmonic coefficient from or to binary format use the support program "potbin".

**5.2 Point data format**

Data on individual points, e.g. a free-air anomaly value and its standard deviation, is stored in list format as free format, with lines

*id,  $\phi$ ,  $\lambda$  (degrees), *h*, data1, data2, ...*

Example:

```

10180  4.21974  113.77524  1578.14  16.61  2.0
10200  4.21127  113.76625  1578.17  17.61  2.0
10220  4.20248  113.75748  1578.18  17.51  2.0
10240  4.19376  113.74859  1578.20  16.71  2.0

```

The first column must always be integer identifier, the follows lat, lon (degrees) and height and one or more data. The height is sometimes understood as data number zero. Some programs may also operate with latitude and longitude replaced by UTM Northing and Easting. Unknown data values may be signalled by "9999" (or higher) – but not all programs are treating the unknown data option.

**5.3 Grid data (text format)**

GRAVSOFT grid data are stored rowwise from north to south, like you would read the values if they were printed in an atlas. The grid values are initiated with label of latitude ( $\phi$ ) and longitude ( $\lambda$ ) limits and spacing, the follows the data in free format:

$\phi_1$  ,  $\phi_2$  ,  $\lambda_1$  ,  $\lambda_2$  ,  $\Delta\phi$  ,  $\Delta\lambda$

$d_{n1}$   $d_{n2}$  .....  $d_{nm}$

.....

.....

$d_{11}$   $d_{12}$  .....  $d_{1m}$

Example (a grid in Borneo with 1' spacing in lat and lon:

```

0.000000  9.000000  106.000000  121.000000  0.01666667  0.01666667

-11.250  -11.060  -10.662  -10.053  -9.219  -8.151  -6.546
-5.083  -3.253  -1.366  0.450  2.060  3.403  4.041
 4.368  4.405  4.255  4.016  4.051  3.850  3.770
 3.694  3.576  3.358  3.169  2.371  1.417  0.373
-0.671  -1.641  -2.321  -3.065  -3.582  -3.927  -4.125
....

```

Each east-west row must be starting on a new line. Unknown data may be signalled by "9999". The grid label defines the exact latitude and longitude of the grid points, irrespectively whether the grids point values or average values over grid cells. The first data

value in a grid file is thus the NW-corner ( $\phi_2, \lambda_1$ ) and the last the SE-corner ( $\phi_1, \lambda_2$ ). The number of points in a grid file is thus

$$\begin{aligned} nn &= (\phi_2 - \phi_1) / \Delta\phi + 1 \\ ne &= (\lambda_2 - \lambda_1) / \Delta\lambda + 1 \end{aligned}$$

Note that in fortran integers are truncated, not rounded, so statements in programs typically will have a half-unit added (+ 1.5) to secure correct rounding of integers nn and ne.

Generally it is recommended to use longitudes in range  $-180$  to  $180$ , but  $0$  to  $360$  will also be ok in most cases.  $\lambda_2$  should always be greater than  $\lambda_1$ .

For some applications (e.g. 3-D “sandwich” interpolation or grids of vertical deflections) more than one grid may be in a grid file.

Grids may be in UTM projection. In this case latitude and longitude must be replaced by Northing and Easting (in meter), followed by an ellipsoid number defining the semi-major axis and flattening (1: WGS84; 2: Hayford-ED50, 3: Clarke-NAD27, 4: Bessel ellipsoid ..) and UTM zone number. Note that the programs accepting UTM grids will check by the magnitude of the latitude and longitude label information, if values larger than 360 is read, an UTM grid is assumed to be in the file.

Example of a UTM grid in zone 32, wgs84 ellipsoid used

```
6305000 6345000 505000 545000 10000 10000
1 32

60 200 110 200 200
101 201 40 30 10
40 20 10 20 10
30 10 0 0 0
15 0 0 0 0
```

## 5.4 Grid data (binary)

Binary grid data are stored in exactly the same way as the grid text files, but in real\*4 binary format, thus saving roughly half the space. The binary grid formats allows the direct-access to specific rows in the grid data, and are thus much faster to use as well.

The binary grid data are stored in internal records of 16 data values, with the first record containing the grid header and a special code (777) allowing programs to check if grids are in ascii or binary formats (some programs like GEOIP and GCOMB will work with either binary or text grids). Binary grids can generally not be moved between different operating systems or even compilers.

To convert between text or binary grids, use the program “GBIN”:

Example of grid conversion(interactive)

```
gbin
borneo.gri
borneo.bin
1
```