

The “.b” format

The .b format was invented by Dennis Milbert in the 1990’s for the purpose of standardizing grid manipulation within the geoid computational software of NGS. It is highly generic to any gridded data with geographic coordinates, and so has been adopted for use in other NGS software which uses grids, such as GEOCON.

The .b format is binary, and created using FORTRAN software. As such, it carries the FORTRAN artifact that each record (vector of multiple binary values created with a single “write” statement) has both a leading and trailing 4-byte buffer. The contents of that buffer are not terribly important, and relate solely to the length of the record itself. As such, any non-FORTRAN program which reads a “.b” file should be prepared to skip 4 bytes at the beginning and end of each record.

Overview:

The .b format begins with one header record, containing basic information about the grid. After that, the rest of the file is filled with records representing actual data in the grid. The first record after the header will contain all of the gridded values on the southernmost row of the grid, with the values in that record arranged from west to east. Each subsequent record contains data for one row northward from the previous row until the final record contains the gridded data (still west to east) of the northernmost row of the grid. This layout was chosen so that the column index would increase with increasing (east) longitude and the row index would increase with increasing latitude.

The header record is laid out as such:

(buffer)	<u>xlatsw</u>	<u>xlonsw</u>	<u>dlat</u>	<u>dlon</u>	<u>nlat</u>	<u>nlon</u>	<u>ikind</u>	(buffer)
-----------------	----------------------	----------------------	--------------------	--------------------	--------------------	--------------------	---------------------	-----------------

These values are described as such:

Name	Type	Bytes	Units	Description
<i>(buffer)</i>	Real	4	N/A	FORTRAN header/footer for each record
<i>xlatsw</i>	Real	8	Decimal degrees	Latitude of SW corner of grid (-90 to 90)
<i>xlonsw</i>	Real	8	Decimal degrees	East longitude of SW corner of grid (0 to 360)
<i>dlat</i>	Real	8	Decimal degrees	Spacing between rows

<i>dlon</i>	Real	8	Decimal degrees	Spacing between columns
<i>nlat</i>	Integer	4	N/A	Number of rows
<i>nlon</i>	Integer	4	N/A	Number of columns
<i>ikind</i>	Integer	4	N/A	Code for type of data in grid

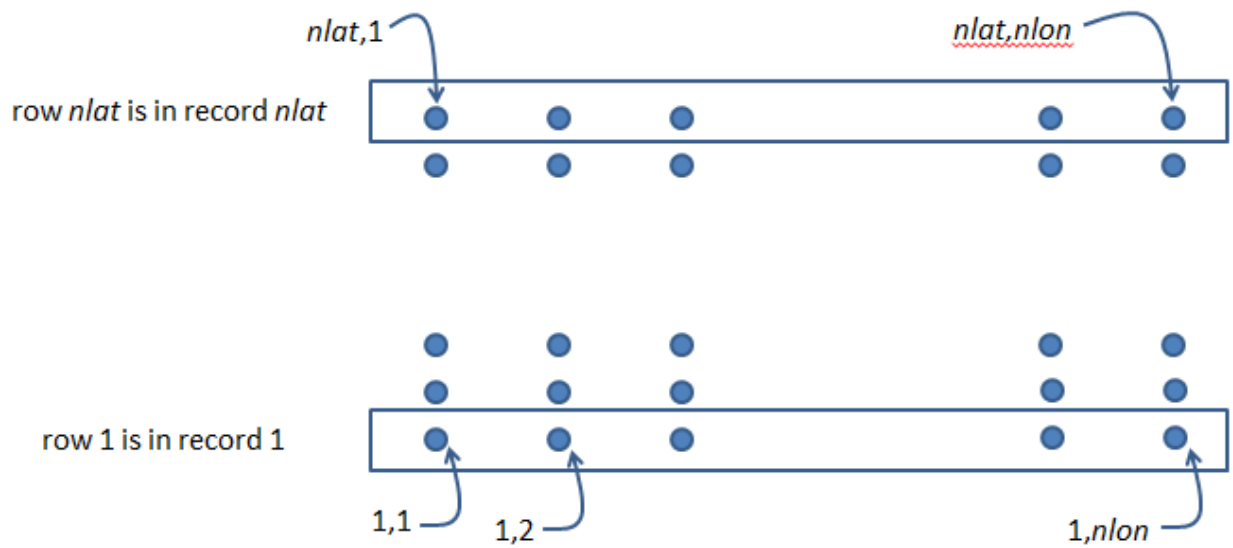
The possible codes for *ikind* are:

ikind value	Data in grid
-1	2 byte integer (special encryption for heights in the USA)
0	4 byte integer
1	4 byte real
2	2 byte integer

Following the header record, each row of the grid fills subsequent records. The type of data (real, integer, etc) in the grid (the “d” values) are determined by “*ikind*” from the header. The records look like this:

(buffer)	d(1,1)	d(1,2)	...	d(1,nlon)	(buffer)
(buffer)	d(2,1)	d(2,2)	...	d(2,nlon)	(buffer)
	
(buffer)	d(nlat,1)	d(nlat,2)	...	d(nlat,nlon)	(buffer)

Do not confuse the layout in the above figure (showing records from top to bottom) with how the data itself actually falls geographically (which is south to north). The first record contains data in the southernmost row:



The latitude of row 1 (record 1) is x_{latsw} . The latitude of row 2 (record 2) is $x_{latsw} + dlat$. The latitude of row 3 (record 3) is $x_{latsw} + 2 * dlat$, etc. The latitude of the final row ($nlat$) is $x_{latsw} + (nlat - 1) * dlat$.

Similarly, the longitude of the first grid point in any record is x_{lonsw} . The next grid point to the east has longitude $x_{lonsw} + dlon$, etc. The final grid point (easternmost) in any record will have longitude of $x_{lonsw} + (nlon - 1) * dlon$.