

Search

# Surfer 7 Grid File Format

**Surfer 7** grid files .GRD use a tag-based binary format to allow for future enhancements. Each section is preceded by a tag structure, which indicates the type and size of the following data. If a program does not understand or want a particular type of data, it can read the associated tag and quickly skip to the next section. In general, sections can appear in any order except for the first, which must be a Header section.

Data types used in **Surfer 7** grid files:

Type	Description
long	32 bit signed integer
double	64 bit double precision floating point value

Each section is preceded by a tag structure with the following format:

Element	Type	Description
Id	long	The type of data in the following section. See the next table for a list of valid values.
Size	long	The number of bytes in the section (not including this tag). Skipping this many bytes after reading the tag will align the file pointer on the next tag.

## Tag ID Values

*Tag Id* values. The 0x prefix indicates a hexadecimal value:

Id	Description
0x42525344	<i>Header</i> section – must be the first section within the file.
0x44495247	<i>Grid</i> section – describes a 2D matrix of Z values.
0x41544144	<i>Data</i> section – contains a variable amount of data. The size of the data section is given by the <i>Size</i> field in the tag structure.

0x49544c46	<i>Fault Info</i> section – describes the fault traces used when creating the grid.
------------	---

## Header Section

The *Header* section must be the first section in the file and has the following format:

Element	Type	Description
Version	long	Version number of the file format. Can be set to 1 or 2

If the version field is 1, then any value  $\geq$  [BlankValue](#) will be assigned the NoData value using Surfer's NoData value, 1.70141e+038.

If the version field is 2, then any value  $=$  [BlankValue](#) will be assigned the NoData value using Surfer's NoData value, 1.70141e+038.

## Grid Section

The *Grid* section consists of a header that describes a 2D matrix of values, followed by the matrix itself. This section encapsulates all of the data that was traditionally referred to as a grid:

Element	Type	Description
nRow	long	number of rows in the grid
nCol	long	number of columns in the grid
xLL	double	X coordinate of the lower left corner of the grid
yLL	double	Y coordinate of the lower left corner of the grid
xSize	double	spacing between adjacent nodes in the X direction (between columns)
ySize	double	spacing between adjacent nodes in the Y direction (between rows)
zlo	double	minimum Z value of the grid. NoData nodes are not included in the minimum.
zhi	double	maximum Z value of the grid. NoData nodes are not included in the maximum. Note that if all nodes are NoData nodes, the $zlo=zhi=1.71041e38$ .

BlankValue	double	nodes are assigned NoData if greater than or equal to this value. NoData values appear as 1.71041e38.
------------	--------	---

## Data Section

A *Data* section containing the 2D matrix of values (doubles) must immediately follow a *Grid* section. Within the *Data* section, the grid is stored in row-major order, with the lowest row (minimum Y) first

## Fault Info Section

A *Fault Info* section describes the fault geometry used to create the grid. *Fault Info* sections have the following format:

Element	Type	Description
nTraces	long	number of fault traces (polylines)
nVertices	long	total number of vertices in all the traces
data section		variable-sized data block consisting of an array of Trace structures immediately followed by the array of vertices

## Data Section

A *Data* section containing an array of *Trace* structures and an array of *Vertex* structures must immediately follow a *Fault Info* section. The number of *Trace* structures in the array is nTraces, and the number of *Vertex* structures is

## Trace Structure

Element	Type	Description
iFirst	long	0-based index into the vertex array for the first vertex of this trace
nPts	long	number of vertices in this trace

## Vertex Structure

Element	Type	Description
x	double	X coordinate of the vertex

y	double	Y coordinate of the vertex
---	--------	----------------------------

## Example

The following example illustrates the layout for a 5 row by 10 column grid:

Element	Type	Description
0x42525344	long	Tag: Id for Header section
4	long	Tag: Size of Header section
1	long	Header Section: Version
0x44495247	long	Tag: ID indicating a grid section
72	long	Tag: Length in bytes of the grid section
5	long	Grid Section: nRow
10	long	Grid Section: nCol
0.0	double	Grid Section: xLL
0.0	double	Grid Section: yLL
1.0	double	Grid Section: xSize
1.75	double	Grid Section: ySize
25.0	double	Grid Section: zMin
101.6	double	Grid Section: zMax
0.0	double	Grid Section: Rotation
1.70141e38	double	Grid Section: BlankValue
0x41544144	long	Tag: ID indicating a data section
400	long	Tag: Length in bytes of the data section (5 rows x 10 columns x 8 bytes per double)
Z11, Z12, ...	double	Data Section: First (lowest) row of matrix. 10 doubles

Z21, Z22, ...	double	Data Section: Second row of matrix. 10 doubles
Z31, Z32, ...	double	Data Section: Third row of matrix. 10 doubles
Z41, Z42, ...	double	Data Section: Fourth row of matrix. 10 doubles
Z51, Z52, ...	double	Data Section: Fifth row of matrix. 10 doubles

---

## See Also

[Grid Files](#)

[Surfer 6 Text Grid Format](#)

[Surfer 6 Grid File Format](#)

[File Format Chart](#)