NAME

mbneptune2esf – Tool to import beam flags from Simrad's Neptune system.

VERSION

Version 5.0

SYNOPSIS

mbneptune2esf [**–F***format* **–I***infile* **–O***outfile* **–M***mode* **–R***rules_file* **–V –H]**

DESCRIPTION

mbneptune2esf imports manual data flagging performed using Simrad's NEPTUNE software and flags those beams in the corresponding MB-System files.

The **binstat** tool in Simrad's NEPTUNE software has a number of advantages over mbedit as a tool for flagging bad data, including access to a variety of statistics from gridding the data. MB-System is generally more flexible and powerful in a number of other areas. **mbneptune2esf** tries to give you the best of both worlds.

mbneptune2esf reads a binstat_?.rules file generated by manually cleaning data using the correlation plot in the binstat tool of the neptune sofware written by Kongsberg Simrad and writes these edits in a mbsystem esf file. The purpose of this tool is to allow data cleaning done using neptune to be carried into the mbsystem environment – thus making neptune a more useful tool.

mbneptune2esf applies all the manual edits in the rules file given to those of the input files it recognises as matching the original lines. A line is considered to match if the line name contained in the rules file (e.g. 0002_20000408_221415) is part of the file name of the data file (e.g. 0002_20000408_221415.mb57).

Swath line files should be converted to format 57 using mbcopy before using this program, as format 56 does not have the ability to flag data (although data can be zeroed – but zero may be a valid depth).

Data flagged by rules generated in binstat is not flagged by this program – only manually flagged data is flagged.

mbneptune2esf can also be used to transfer processing done in neptune to other surveys. Neptune breaks a survey up into a number of blocks and records all the processing done in these blocks. Changing a block results in a loss of all the existing processing information. When adding data to an existing survey that falls outside any existing block, all of the existing blocks must be removed, and therefore all of the processing is lost. **mbneptune2esf** and **mbprocess** can be used to zero all the beams flagged during neptune processing and the processed files can then be used as the basis for a new survey, preserving the manual flagging (but losing the flagged data). The zero data can then be easily flagged in neptune.

```
cd $RAWDATAHOME
ls *raw.all | while (read file) ; do
                echo $file 56 1 >> datalist.mb-1
        done
cat $PROCHOME/old_survey/binstat_*.rules | mbneptune2esf –m3
mbprocess
cat *p.mb56 | handleEmX new_survey op
```

This program flags beams by outputting the flags as edit events to an "edit save file", like that produced by **mbedit**. If an "edit save file" (named by adding a ".esf" suffix to the input swath filename) already exists, the edits are read in and applied before the **mbneptune2esf** flagging algorithms are used. Once generated, the edit events can be applied to the data using the program **mbprocess**, which outputs a processed swath data file. The **mbprocess** program is also used to merge edited navigation, recalculate bathymetry, and apply other corrections to swath bathymetry data.

AUTHORSHIP

Gordon Keith (gordon.keith@csiro.au)
CSIRO Australia
David W. Caress (caress@mbari.org)
Monterey Bay Aquarium Research Institute
Dale N. Chayes (dale@ldeo.columbia.edu)
Lamont-Doherty Earth Observatory

OPTIONS

–F *format*
Sets the data format used if the input is read from stdin or from a file. If *format* < 0, then the input file specified with the **–I** option will actually contain a list of input swath sonar data files. This program uses the **MBIO** library and will read or write any swath sonar format supported by **MBIO**. A list of the swath sonar data formats currently supported by **MBIO** and their identifier values is given in the **MBIO** manual page. Default: *format* = 11. Only formats 56 and 57 files are affected by this program. Other files are ignored.

–H This "help" flag cause the program to print out a description of its operation and then exit immediately.

–I *infile*
Sets the input filename. If *format* > 0 (set with the **–F** option) then the swath sonar data contained in *infile* is read and processed. If *format* < 0, then *infile* is assumed to be an ascii file containing a list of the input swath sonar data files to be processed and their formats. The program will read and process the data in each one of these files. Each input file will have an associated output file with the ".esf" suffix. In the *infile* file, each data file should be followed by a data format identifier, e.g.:

datafile1 11
datafile2 24

This program uses the **MBIO** library and will read or write any swath sonar format supported by **MBIO**. A list of the swath sonar data formats currently supported by **MBIO** and their identifier values is given in the **MBIO** manual page. Only formats 56 and 57 files are affected by this program. Other files are ignored. Default: *infile* = "datalist.mb-1".

–M *mode*
Sets the manner in which flagged by neptune are handled.

mode = 1: Flags the flagged beam.
mode = 3: Zeros the flagged beam. Default: *mode* = 1.

–O *outfile*
Specifies a file to write a copy of the beam information extracted from the rules file. Mostly for debugging purposes.

–R *rules_file*
The binstat rules file containing the manual flagging rules to be applied. Rules files are normally $PROCHOME/*survey*/binstat_*block*.rules. The LINES_PING_BEAM_RULE rules from this file will be applied to the matching line(s) in the input file. Default: *rules_file* = stdin.

–V Normally, **mbneptune2esf** works "silently" without outputting anything to the stderr stream. If the **–V** flag is given, then **mbneptune2esf** works in a "verbose" mode and outputs the program version being used, all error status messages, and the number of beams flagged as bad.

EXAMPLES

The normal use of **mbneptune2esf** would be to apply all the manual edits in a Neptune survey to the corresponding MB-System survey represented by a datalist:

```
cat $PROCHOME/Broken_Reef/binstat_*.rules | \
        mbneptune2esf –I/data/mbproc/Broken_Reef/datalist.mb-1
```

Since the default input file is the datalist in the current directory, the following is equivalent if the Neptune environment variable PROCHOME evaluates to /data/proc:

```
cd /data/mbproc/Broken_Reef
cat /data/proc/Broken_Reef/binstat_*.rules | mbneptune2esf
```

To apply the manual edits from a particular block of a Neptune survey to only one of the related data files:

```
mbneptune2esf –R $PROCHOME/Broken_Reef/binstat_1.rules \
        -F57 –I0032_20000408_221201.mb57
```

To use the manual edits from a particular block of a Neptune survey to zero the beams of one of the related data files:

```
mbneptune2esf –R $PROCHOME/Broken_Reef/binstat_2.rules
        -M3  –F56 –I0002_20000408_153940_raw.all
```

SEE ALSO

mbsystem(1), **mbclean**(1), **mbedit**(1), **mbprocess**(1)

BUGS

mbneptune2esf does not flag beams flagged by rules in binstat, only beams that are flagged manually using the correlation plot.

Probably plenty of others.