NAME

mbsystem - A set of utilities for manipulating and processing swath sonar bathymetry, amplitude, and sidescan data.

VERSION

Version 5.1

AUTHORSHIP

David W. Caress (caress@mbari.org)
Monterey Bay Aquarium Research Institute
Dale N. Chayes (dale@ldeo.columbia.edu)
Lamont-Doherty Earth Observatory

INTRODUCTION

MB-System is a software package consisting of programs which manipulate, process, list, or display swath sonar bathymetry, amplitude, and sidescan data. This software is distributed freely (and for free) in the form of source code for Unix platforms. The heart of the system is an input/output library called **MBIO** which allows programs to work transparently with any of a number of supported swath sonar data formats. This approach has allowed the creation of "generic" utilities which can be applied in a uniform manner to sonar data from a variety of sources. Most of the programs are command-line tools, but the package does include graphical tools for editing swath bathymetry, editing navigation, modeling bathymetry calculation, and adjusting survey navigation.

The acronym MB stands for MultiBeam; this reflects the fact that the early development of **MB-System** focused on processing deep sea data from multibeam sonars. However, **MB-System** now supports several data formats from sidescan and interferometric sonars and works with data from all depth ranges.

The National Science Foundation (NSF) has been the primary supporter of **MB-System** development. NSF initially provided grants to the authors at the Lamont-Doherty Earth Observatory (L-DEO) of Columbia University in 1993 (two years), and 1995 (five years). SeaBeam Instruments also contributed significant effort from 1994 to 1999. Since Caress moved to the Monterey Bay Aquarium Research Institute (MBARI) in 1998, NSF has continued support through two five year grants (1999-2005 & 2006-2011) to both Caress at MBARI and Chayes at L-DEO. The Packard Foundation matches the NSF support to Caress at MBARI.

The original NSF mandate was to create a set of generic tools which would work with all of the sonar data used in NSF-supported projects. Not surprisingly, the core of the **MB-System** user community consists of NSF-supported researchers at U.S. institutions such as the Lamont-Doherty Earth Observatory, the Scripps Institution of Oceanography, and the Woods Hole Oceanographic Institution. However, **MB-System** has also come to be used widely in the international marine geological community.

The **MB-System** source code is distributed under the GNU General Public License as formulated by the GNU Project. MB-System installation depends on some other software packages (e.g. GMT and netCDF), all of which are freely available as source code.

COPYRIGHT AND LICENSING

All Wrongs Remembered

The MB-System source code is distributed under the GNU General Public License as formulated by the GNU Project. Previous MB-System distributions have been described as "public domain", which meant there was no restrictions whatsoever on the use of the code. We have now adopted the more restrictive GNU GPL license in order to insure that anyone who distributes software based in whole or in part on MB-System also distributes the modified MB-System source code and any additional source code.

The GNU GPL also prohibits the distribution of proprietary executables linked with MB-System libraries unless the source code is also distributed. We waive this restriction on distributing proprietary compiled programs for specific software products if and only if those software products meet the following two conditions:

1) The software product was created, sold, and delivered to customers using source code derived from MB-System release 4.6 distributions.

2) The software product was sold and delivered to customers prior to January 1, 2001.

HISTORY

The development of **MB-System** began in 1990 as part of ongoing research at L-DEO involving swath bathymetry data collected with SeaBeam multibeam sonars. Development was accelerated in 1991 as part of the effort to support the STN-Atlas Hydrosweep DS multibeam sonar on L-DEO's ship, the R/V Maurice Ewing. The National Science Foundation provided support in 1993 and 1994 to improve and extend **MB-System**. The intent of this initial grant was to provide a standard "generic" set of tools for processing and display of swath sonar data that could be used by the U.S. academic community. The first generally released version of **MB-System** (3.0) was made available in the Spring of 1993. This was followed by versions 3.1 and 3.2 in July, 1993, version 3.3 in November, 1993, and version 3.4 in December 1993. All of these early releases supported only SeaBeam and Hydrosweep data.

SeaBeam Instruments and Antarctic Support Associates provided additional support in 1994 for the development of **MB-System**, with particular emphasis on capabilities related to the new SeaBeam 2100 series of sonars. A considerably enhanced **MB-System** version 4.0 was released on October 22, 1994; this release followed an almost complete rewrite of the underlying source code. The new capabilities included support for sidescan as well as bathymetry data and support for data from a number of very different sonars.

The National Science Foundation funded a five year effort begun in 1995 to maintain and further develop **MB-System**. From 1994 to 1997, SeaBeam Instruments (a major multibeam sonar manufacturer and, at the time, the principal employer of David W. Caress) provided significant support for **MB-System** development and maintenance. Similarly, the Newport, RI office of the Science Applications International Corporation (SAIC) supported some **MB-System** development during 1997-1998, when David W. Caress worked there. Version 4.1 was released in November, 1994, followed by 4.2 in February 1995, 4.3 on March 12, 1996, 4.4 on August 27, 1996, and 4.5 on September 23, 1997.

David W. Caress joined the Monterey Bay Aquarium Research Institute (MBARI) in September, 1998. Version 4.6 was released on April 16, 1999. The final update to version 4.6 (4.6.10) was announced on March 8, 2000. The primary innovations during this period included support for the new generation of Simrad multibeam sonars and tools for generating data products that could be imported directly into GIS software packages.

The National Science Foundation has continued to support **MB-System** development through additional five year grants (2001-2006 & 2006-2011) to MBARI and L-DEO. The Packard Foundation matches the NSF support to Caress at MBARI. The version 5.0 release, of which this documentation is a part, incorporates yet another substantial rewrite of the underlying code as well as providing significant new capabilities.

WHAT'S NEW ABOUT VERSION 5

The version 5.0 release of **MB-System** includes a number of changes and improvements relative to the version 4 releases. The most significant changes include:

A new approach to managing data processing.

- Many tools - one output file. In previous versions of **MB-System**, each processing program read an input swath data file and produced an output swath data file. This "serial" processing scheme generally produced a large number of intermediate data files. **MB-System** version 5.0 features the integration of the editing and analysis tools with a single program, mbprocess, that outputs processed data files. The new "parallel" processing scheme covers bathymetry data processing, but does not yet incorporate the sidescan processing capabilities. All of the old tools and capabilities are still part of the distribution.
- Recursive datalists. The lists of data files used by gridding and plotting programs can now be recursive, making it simpler to manage data from many different surveys.
- Automatic format identification. **MB-System** programs will now attempt to automatically identify the swath data format based on the filename suffix.
- Ancillary data files. Many common **MB-System** tasks (e.g. swath plotting and gridding) can be accomplished more rapidly using ancillary data files containing file statistics (".inf" files), quickly read bathymetry (".fbt" files), and (".fnv") files. Each of these files is named by adding the indicated four character suffix to the original swath data filename. The ".inf" files are created by directing the output of **mbinfo** to a file. The ".fbt" files are creating by using **mbcopy** to extract the bathymetry into a format 71 file. The ".fnv" files are created using **mblist** to create a text navigation list. These ancillary files are automatically created together using the program **mbdatalist**.

New tools.

- **mbnavadjust**. This new tool allows users to adjust poorly navigated surveys by matching features in overlapping swathes. It is particularly useful for processing surveys conducted from submerged platforms.
- **mbprocess**. This new tool performs a variety of processing tasks and produces a single output processed swath data file. The program mbprocess can apply bathymetry edits from mbedit and mbclean, navigation edits from mbnavedit, sound velocity profile changes from mbvelocitytool, and a variety of other corrections.
- **mbset**. This new tool allows users to create and modify the parameter files used to control the operation of mbprocess.
- **mbdatalist**. This new tool allows users to list the files referenced by a recursive datalist structure. It can also be used to create the ancillary ".inf", ".fbt", and ".fnv" files for all of the data files referenced in a recursive datalist structure.
- **mbsvplist**. This new tool lists water sound velocity profiles embedded in swath data files, creating secondary files that can be read into MBvelocitytool.
- **mbareaclean**. This new tool identifies and flags artifacts in swath sonar bathymetry data within a specified area of interest. The area is divided into a grid with square cells or bins, and the data are grouped according to these bins. Once all of data are read, statistical tests are applied to the soundings within each bin.

Support for Projected Coordinate Systems.

- **MB-System** now incorporates the source code for the PROJ.4 Cartographic Projections library, providing support for (apparently) all commonly used geodetic coordinate systems. **PROJ.4** was developed by Gerald Evenden (then of the USGS), and was obtained from the www.remotesensing.org website.
- A large number of commonly used projected coordinate systems (e.g. UTM) are defined in a file (mbsystem/share/projections.dat) distributed with **MB-System**. These include all of the standard UTM

zones, all of the standard state plate coordinate systems, and most of the European Petroleum Survey Group (EPSG) coordinate systems (also including UTM).

- **MB-System** can now handle swath data that is navigated in a supported projected coordinate system. In particular, data files that are navigated with UTM eastings and northings instead of longitude and latitude can now be plotted and processed with **MB-System**.
- The programs **mbgrid** and **mbmosaic** can now output grids and mosaics in any of the projected coordinate systems specified in mbsystem/share/projections.dat.
- The TIFF images generated with **mbm_grdtiff** and **mbgrdtiff** now fully conform to the GeoTIFF standard, providing that the source grids or mosaics were generated using **mbgrid** or **mbmosaic** in either Geographic coordinates, UTM coordinates, or any of the EPSG coordinate systems specified in the projections.dat file. This means, for instance, that GeoTIFF images generated with **mbgrdtiff** will be properly georeferenced when they are imported into ESRI **ArcGIS** or other GIS packages.

Restructuring the code.

- All of the C code now conforms to the ANSI C standard.
- The underlying input/output library (**MBIO**) has been substantially rewritten. The structure has been streamlined, simplifying both future development and support of the existing code. The **MBIO** API has been greatly modified.

Handling of old Simrad multibeam data.

- Vendor format data from the old Simrad multibeams (pre-1997 sonars) are now supported by a single format id (51) rather than a separate format id for each sonar model. The old format id's are automatically aliased to 51, so existing shellscripts will continue to work.
- MB-System no longer supports beam flagging in format 51 data. The use of mbedit and mbclean on format 51 data will cause the flagged beams to be irrevocably nulled. Previous versions of MB-System used the highest bit in the depth values to represent beam flags because no Simrad data seemed to use that bit. We have not obtained data with depth values using the full bit-range, conflicting with the old beam flagging scheme. We recommend that old Simrad data be translated to the extended processing format (57) which contains proper beam flags and supports all processing functions. Format 57 is also used for processing data from the current Simrad multibeam sonars.
- Sidescan data from old Simrad multibeams (pre-1997 sonars) are now handled in the same manner as data from the newer sonars (e.g. EM3000, EM3000, EM120). The raw samples in the vendor data format are binned, averaged, and interpolated into a 1024 pixel sidescan swath. This binned sidescan is not saved in the vendor format, so (as above) it is recommended that the data be translated to an extended format (57) that stores both bathymetry beam flags and processed sidescan.

Streamlining of MB-System Default Parameters.

- Prior to version 5.0, the **MB-System** defaults set by **mbdefaults** included the format id, a control for ping averaging, longitude and latitude bounds for windowing by area, and begin and end times for windowing in time. These values are no longer set in the .mbio_defaults file or controlled by **mbdefaults**. As noted above, the format id is automatically identified from the filename when possible. When filenames do not match one of the recognized structures, users must specify the format using the relevant programs **-F***format* option. The controls for ping averaging and windowing in time and space are rarely used, and must now be explicitly set in command line arguments.

New Data Formats.

- Furuno HS10 multibeam bathymetry is supported as format 171.
- SeaBeam 2120 multibeam data in the L3 Communications XSE format are supported as format 94 (already used to support Elac Bottomchart MkII XSE data).

- Raw STN Atlas multibeam data generated by the upgraded Hydrosweep DS2 multibeam on the R/V Ewing are supported by read-only format 182. Processing is supported using the augmented read-write format 183.
- The IFREMER netCDF multibeam archiving data format is supported as format 75. Similarly, the IFREMER netCDF navigation archiving data format is supported as format 167.
- The STN Atlas processing data format SURF is supported as format 181. At present, SURF is supported as a read-only format. This allows plotting and gridding of the SURF data, but not processing. Writing or translating the SURF data to allow processing will be supported in a later version.
- The Hawaii Mapping Research Group's new MR1 format is supported as format 64. This format is used to disseminate data from both the HMRG interferometric sonars (e.g. MR1) and the WHOI DSL 120 deep-towed inteferometric sonar. This format has been supported by including the code for the HMRG library libmr1pr in the **MB-System MBIO** library. Thanks to Roger Davis and HMRG for making the code available under the GPL.
- The Edgetech Jstar format for sidescan and subbottom data is now supported as formats 132 and 133.

THE NEW VERSION 5 DATA PROCESSING STRUCTURE

Previous versions of **MB-System** have emphasized processing utilities that operate serially, that is, that read in an input swath data file, modify it, and then output a second swath data file. Serial processing utilities do allow for great flexibility in processing because one uses only the programs required and, in general, the order in which the programs are used does matter. However, one consequence of serial processing has been that processing data frequently results in a large number of intermediate data files. For instance, if an EM300 data file called "mbari_1998_524.mb57" has spikes in the bathymetry, outliers in the navigation, and the bathymetry has been calculated using an incorrect model for the water sound speed structure, users of **MB-System** would previously have processed it with a sequence something like this:

1) Run **mbclean** to automatically flag
the obvious bathymetric artifacts.
Input: mbari_1998_524.mb57
Output: mbari_1998_524_c.mb57

2) Run **mbedit** to interactively flag
bathymetric artifacts.
Input: mbari_1998_524_c.mb57
Output: mbari_1998_524_ce.mb57

3) Run **mbnavedit** to interactively clean
the navigation.
Input: mbari_1998_524_ce.mb57
Output: mbari_1998_524_cen.mb57

4) Run **mbvelocitytool** to generate an
appropriate sound velocity profile (SVP) for
recalculating the bathymetry.
Input: mbari_1998_524_cen.mb57
Output: good.svp

5) Run **mbbath** to recalculate the
bathymetry using the SVP file generated
using **mbvelocitytool**.
Input: mbari_1998_524_cen.mb57
Input: good.svp
Output: mbari_1998_524_cenb.mb57

6) Run **mbsimradmakess** to recalculate the
sidescan while ignoring sidescan samples
from beams now flagged as bad.
Input: mbari_1998_524_cenb.mb57
Output: mbari_1998_524_cenbs.mb57

7) Run **mbbackangle** to calculate an amplitude
vs grazing angle table for correcting
sidescan.
Input: mbari_1998_524_cenbs.mb57
Output: ampvsga.dat

8) Run **mbanglecorrect** to correct the sidescan.
Input: mbari_1998_524_cenbs.mb57
Input: ampvsga.dat
Output: mbari_1998_524_cenbsc.mb57

The result of this processing includes the input swath data file, the final swath data file, and five intermediate swath data files. The impact of this approach on data storage requirements is obvious (though some users have ameliorated the issue by working with scripts that automatically delete unneeded data files or by piping data from one non-graphical process to another to avoid making intermediate files). The serial processing approach also presents data management problems because data files frequently have different processing requirements, even within a single survey. We have found that working with very large datasets can be awkward when only a minority of data files require navigation editing or bathymetric recalculation, or when some data files require multiple episodes of bathymetry and navigation editing.

MB-System version 5 includes new utilities implementing a parallel processing scheme that simplifies the processing of most swath data. This scheme is centered around the program **mbprocess**, which can accomplish the following processing tasks in a single step:

- Merge edited navigation generated by **mbnavedit**.
- Apply bathymetry edit flags from **mbedit** and **mbclean**
- Recalculate bathymetry from raw travel time and angle data by raytracing through water sound speed models from **mbvelocitytool** or **mbsvplist**.
- Apply changes to roll bias, pitch bias, heading bias, and draft values.
- Recalculate sidescan from raw backscatter samples (Simrad multibeam data only).
- Recalculate sidescan from raw backscatter samples (Simrad multibeam data only).
- Correct sidescan for amplitude vs grazing angle patterns.
- Apply tides to bathymetry.

The actions of **mbprocess** are controlled by text parameter files. Each **mbprocess** parameter file is named by adding a ".par" suffix to the associated input swath data file and contains single line commands that set processing modes and parameters. Tools such as **mbedit**, **mbnavedit**, and **mbclean** all generate and/or modify parameter files in addition to generating parallel files used by **mbprocess**. The program **mbset** can also be used to create and modify **mbprocess** parameter files.

The same processing steps described above as a serial processing scheme can be accomplished with the new parallel processing scheme as follows:

1) Run **mbdatalist** to create ancillary
data files containing statistics
(".inf"), quickly read bathymetry
(".fbt"), and quickly read navigation

(".fnv"). These files are used to
speed common operations such as swath
plotting and gridding.
  Input: mbari_1998_524.mb57
  Output: mbari_1998_524.mb57.inf
          mbari_1998_524.mb57.fbt
          mbari_1998_524.mb57.fnv

2) Run **mbclean** to identify
the obvious bathymetric artifacts
and output a list of the edit events.
The parameter file is created and
set to apply bathymetry flags from
the ".esf" file.
  Input: mbari_1998_524.mb57
  Output: mbari_1998_524.mb57.esf
          mbari_1998_524.mb57.par

3) Run **mbedit** to interactively
identify bathymetric artifacts
and output a list of the edit events.
The existing edits from **mbclean**
are loaded and applied prior to editing.
The parameter file is updated and
set to apply bathymetry flags from
the ".esf" file.
  Input: mbari_1998_524.mb57
          mbari_1998_524.mb57.esf
          mbari_1998_524.mb57.par
  Output: mbari_1998_524.mb57.esf
          mbari_1998_524.mb57.par

4) Run **mbnavedit** to interactively
clean the navigation. The edited
navigation is output to the ".nve" file.
The parameter file is updated and
set to merge the navigation from
the ".nve" file.
  Input: mbari_1998_524.mb57
          mbari_1998_524.mb57.par
  Output: mbari_1998_524.mb57.nve
          mbari_1998_524.mb57.par

5) Run **mbvelocitytool** to generate an
an appropriate sound velocity profile
(SVP) for recalculating the bathymetry.
The SVP is output to the ".svp" file.
The parameter file is updated and
set to recalculate the bathymetry by
raytracing throught SVP model from
the ".svp" file.
  Input: mbari_1998_524.mb57
          mbari_1998_524.mb57.par
  Output: mbari_1998_524.mb57.svp

```
          mbari_1998_524.mb57.par

6) Run mbbackangle to generate an
  a set of amplitude vs grazing angle
  tables at regular intervals in the
  data. These tables are placed into
  a single ".sga" file. The parameter
  file is updated and set to correct
  the sidescan by interpolating the
  amplitude vs grazing angle table for
  each ping.
     Input:  mbari_1998_524.mb57
             mbari_1998_524.mb57.par
     Output: mbari_1998_524.mb57.sga
             mbari_1998_524.mb57.par

7) Run mbset to set the parameter file
  so that mbprocess will recalculate
  the sidescan (this is for Simrad
  multibeam data only) while ignoring
  sidescan samples from beams now flagged
  as bad.
     Input:  mbari_1998_524.mb57.par
     Output: mbari_1998_524.mb57.par

8) Run mbprocess to apply the bathymetric
  edits, merge the cleaned navigation,
  recalculate the bathymetry, recalculate
  the sidescan, and correct the sidescan.
  The processed swath data is written to
  an output swath data file. The usual
  ancillary data files containing statistics,
  quickly read bathymetry, and quickly
  read navigation are also created.
     Input:  mbari_1998_524.mb57
             mbari_1998_524.mb57.esf
             mbari_1998_524.mb57.nve
             mbari_1998_524.mb57.svp
             mbari_1998_524.mb57.aga
             mbari_1998_524.mb57.par
     Output: mbari_1998_524p.mb57
             mbari_1998_524p.mb57.inf
             mbari_1998_524p.mb57.fbt
             mbari_1998_524p.mb57.fnv
```

The result of this processing is a single output swath data file. Moreover, the processed output can be easily updated if, for example, additional bathymetry editing is required. If the **mbedit** program is used again, it will load the existing edit events from the ".esf" file and then update the ".esf" file. To incorporate the updated bathymetry edits, one just runs **mbprocess** again. One can similarly change the SVP file without impacting on the bathymetry editing or navigation editing components of the processing.

All of the old, serial processing utilities are still distributed with **MB-System**. However, some of the serial tools have been replaced by new versions of the same name (e.g. **mbedit**, **mbclean**, **mbbackangle**, and **mbnavedit**). In these cases, the old versions are preserved as programs with "old" added to their names (e.g. **mbeditold**, **mbcleanold**, **mbbackangleold**, and **mbnaveditold**).

THE NEW VERSION 5 DATALIST FILES

Previous versions of **MB-System** have used lists of data files, or datalists, as input to several programs (e.g. **mbgrid**, **mbmosaic**, **mbinfo**, **mblist**, and **mbm_plot**). The basic datalist entry has consisted of a swath data name (often including the entire path) followed by a space and then the **MB-System** format id for that file. Datalist entry lines starting with the character '#' are considered comments. Version 5 extends the definition and usage of datalists in several significant ways. First, datalists may now be recursive. A datalist entry may be another datalist file, as indicated by a format id of -1. Second, datalists entries may contain a third column which is interpreted as a gridding weight value by **mbgird**. This third value may be used to weight some data higher than other data. For example, one might weight SeaBeam 2112 data with a value of 1.0 and lower quality SeaBeam classic data with a value of 0.001. The result would be that the newer 2112 data effectively overlies the less good data wherever overlap occurs. The third new feature of datalists works with data processed using the new parallel processing scheme. In the parallel processing scheme the raw data files are often, but not always, accompanied by processed files produced by **mbprocess**. It is awkward to maintain datalists that directly refer to the current best datafiles. If one inserts the text $PROCESSED into the first line of a datalist containing raw files, then programs like **mbgrid** will read the processed file if it exists, but otherwise will read the raw file. Similarly, a first line of $RAW will force the programs to only read the raw files directly referenced in the datalist. These options also work recursively. The first instance of a $PROCESSED or $RAW tag will prevail over all others encountered through a hierarchy of recursive datalists. The gridding weight values will also be applied recursively, so one can specify the gridding weight for a large number of data files by applying to a datalist entry which is itself a datalist referencing those files. However, gridding weights are by default overridden by any values applied to the file entries themselves (this behavior can be reversed using the datalist tag $NOLOCALWEIGHT).

VERSION 5 FILE NAMING CONVENTIONS

The version 5 **MB-System** programs make extensive use of standardized filename suffixes. These suffixes allow **MB-System** programs to know what kinds of files it is working with, and in particular to determine swath data formats without the user having to specify them. Although the processing will generally work even if a user does not use the standard filenaming convention, we can guarentee that the user's work will be much easier if the convention is followed. Quite simply, the convention is for swath files to end with a suffix of the format ".mbXX" or ".mbXXX", where XX or XXX is the two digit or three digit **MB-System** format id, respectively. For instance, a Simrad EM3000 file in the processing format 57 might have a name like:

    0053_20020518_205816.mb57

and a Reson 8101 file in the GSF format 121 might have a name like:

    039_2106.mb121

MB-System programs are able to recognize many standard filename conventions used by sonar vendors or data logging package vendors. For instance, filenames with a "_raw.all" suffix, like:

    0053_20020518_205816_raw.all

are recognized as either old (format 51) or new (format 56) Simrad multibeam data (and the programs also determine which of the two formats apply). Filenames ending with ".rec" are recognized as SeaBeam 2100 multibeam data. Filenames ending with ".xse" are known to be Elac Bottomchart or SeaBeam 2100 multibeam data in the XSE format 94. In these instances the program **mbprocess** will automatically replace the vendor suffix with the **MB-System** convention suffix when it creates a processed output file.

LIST OF MB-SYSTEM PROGRAMS AND MACROS

See the individual manual pages for detailed information about specific programs. See the manual page for **MBIO** for information about the i/o library and the swath sonar data formats supported by **MBIO**.

These are the MB-system programs which are used in the version 5 parallel processing scheme. Those programs that are also relevant to the serial processing scheme are marked with an *:
mb7k2jstar: Extract Jstar format (format 132) sidescan

and subbottom data from Reson 7k (format 88)
data files.
mb7kpreprocess: Preprocess Reson 7k data (format 88),
including applying time lag and biases to
attitude and navigation data.
mbbackangle: Generates tables of the average
amplitude or sidescan values as a
function of the grazing angle with
the seafloor at regular intervals in the data.
mbclean: Automatically identifies and
flags bad beams in swath sonar
bathymetry data.
mbcontour*: Generate GMT compatible Postscript
color swath contour plots.
mbcopy*: Copy swath sonar data files.
mbdatalist: Parses recursive datalist files
and outputs the complete list of
data files, formats, and file weights.
mbdefaults*: Set or list default mbio parameters
for reading and writing swath sonar data
mbedit: Interactive editor used to flag bad
beams in swath sonar bathymetry data.
mbextractsegy: Extract subbottom profiler or
center beam reflection data to segy files.
mbformat*: List information about swath sonar
data formats supported by the **MBIO** library.
mbgetesf*: Extract list of flagging or unflagging
beam edit events from a swath sonar data
file in the edit save file (".esf") format
used by **mbedit**, **mbclean**, and
mbprocess.
mbgrdtiff: Generate TIFF image from gridded data
mbgrdviz: Vizualize GMT grids.
mbgrid*: Grid bathymetry, amplitude, and sidescan
data from swath sonar data files.
mbhistogram*: Obtain histogram of bathymetry,
amplitude, or sidescan data from
swath sonar data files.
mbinfo*: Output some basic statistics of
swath sonar data files.
mblevitus*: Create a water velocity profile
which is representative of the mean
annual water column for a specified
1 degree by 1 degree region.
mblist*: List data in swath sonar data files.
mbmosaic*: Mosaic sidescan and amplitude data.
mbnavadjust: Interactive navigation adjustment
package that adjusts navigation so
that swath bathymetry matches where
swathes overlap or cross.
mbnavedit: Interactive editor used to fix
problems with navigation in
swath sonar data files.
mbnavlist*: List navigation data in swath

sonar data files.
mbneptune2esf: Extract bathymetry edits from
Simrad Neptune software into edit save file format.
mbprocess: Performs a variety of processing
tasks in a single step, including
merging navigation, applying
bathymetry edits, recalculating
bathymetry, and recalculating
sidescan.
mbps*: Simple perspective views of swath
bathymetry in Postscript.
mbrolltimelag: Estimate attitude time lag by cross
correlation of apparent bottom slope with the
roll time series:
mbsegygrid: Generate time vs. trace number
grids of seismic data from segy files.
mbsegyinfo: Output some basic statistics of
segy seismic data files.
mbsegylist: List selected header values in
segy seismic data files.
mbset: Create and modify **mbprocess**
parameter files.
mbsvplist*: List water sound velocity profiles in swath
sonar data files.
mbswath*: Generate GMT compatible Postscript
color and color shaded relief swath plots.
mbvelocitytool*: Interactive program for
modeling the affect of the water
velocity profile on swath sonar
bathymetry calculations.

The following are MB-system programs which are not used in the version 5 parallel processing scheme. These programs are included in the version 5 releases for backward compatability with the old serial processing scheme:
hsdump: Lists contents of the various sorts of
data records in Hydrosweep DS data.
mbanglecorrect: Apply a grazing angle correction
to beam amplitude or sidescan data.
mbbackangle: Generates a table of the average
amplitude or sidescan values as a
function of the grazing angle with
the seafloor.
mbbath: Generates bathymetry from travel times
in swath sonar data.
mbcleanold: Old tool that utomatically
identifies and flags bad beams
in swath sonar bathymetry data.
mbcut: Removes data from portions of swath
as specified by the user.
mbeditold: Old interactive editor used to flag bad
beams in swath sonar bathymetry data.
mbfilter: Apply some simple filter functions
to sidescan, beam amplitude, or
bathymetry data.

mbgetmask: Extract list of flagged or edited
beams from a swath sonar data file.
mbmask: Apply editing information obtained
from one file with **mbgetmask**
to another file.
mbmerge: Merge swath sonar data with new
navigation.
mbnaveditold: Old interactive editor used to fix
problems with navigation in
swath sonar data files.
mbrollbias: Evaluate the roll bias of a
swath sonar system using two pieces
of coincident bathymetry data
collected with opposing ship headings.
mbsimradmakess: Regenerate sidescan imagery
from the raw amplitude samples contained
in data from Simrad EM300 and EM3000 sonars.
mbtide: Corrects swath sonar bathymetry
for tide data.
mbunclean: Unflags edited beams in swath sonar
bathymetry data.

Macros are programs or shellscripts which make use of programs from the MB-system and other software packages to accomplish common tasks easily. These are the current MB-system macros:
mbm_arc2grd: Convert an ESRI ArcView ASCII grid file
to a GMT grid file.
mbm_copy: Translate groups of swath data files
between formats
mbm_dslnavfix: Reads a WHOI DSL AMS-120 processed
navigation file containing UTM northings
and eastings and outputs a navigation file
containing longitude and latitude which is
suitable for use with **mbmerge**.
mbm_fmtvel: Scans a Hydrosweep DS data file
and outputs a formatted table of
the mean water velocity and surface
water velocity values used in
processing that data.
mbm_grd2arc: Converts a GMT grid file to an ESRI ArcView
ASCII grid file.
mbm_grd2geovrml: Create and execute commands which
generate a TerraVision tileset and GeoVRML
set of files that can be combined with
other data and viewed in a web browser.
mbm_grd3dplot: Reads a GMT GRD grid file and
writes a shellscript which will
generate a GMT 3D perspective view
of the data.
mbm_grdcut: Extracts a subarea of a GMT grid file.
mbm_grdplot: Reads a GMT GRD grid file and
writes a shellscript which will
generate a GMT map of the data.
mbm_grdtiff: Reads a GMT GRD grid file and writes a
shellscript which will generate a TIFF image

of the data.

mbm_grid: Reads a swath sonar data file and writes a shellscript which will grid bathymetry data or mosaic sidescan (or amplitude) data using reasonable guesses at the appropriate grid bounds and bin size.

mbm_makedatalist: Generates an MB-System datalist file referencing all identifiable swath files in the specified target directory.

mbm_plot: Reads a swath sonar data file and writes a shellscript which will generate a swath and/or contour plot of the data.

mbm_rollerror: Reads a swath sonar data file, calculates the noise in the vertical reference used by the sonar, and generates a file containing roll corrections which can be applied to the data.

mbm_route2mission: Translate an mbgrdviz survey route file into an MBARI AUV mission script.

mbm_stat: Runs **mbinfo** on a swath sonar data file and extracts beam statistics from the output of **mbinfo**.

mbm_utm: Performs forward and inverse UTM projections of ASCII data triples.

mbm_vrefcheck: Generates a plot of high pass filtered apparent crosstrack seafloor slope.

mbm_xbt: Processes a Sparton XBT data file and outputs a sound velocity profile file which can be used to process swath sonar data.

mbm_xyplot: Reads one or more ASCII "X-Y" data files and writes a shellscript which will generate an XY plot of the data.

EXAMPLE PROCESSING APPROACH

An example processing stream for swath sonar data which uses the new, parallel processing scheme is provided here. Note that a ’´ character at the end of a line indicates that the command line should actually continue with the text on the next line. Refer to individual program manual pages for detailed information on the command arguments and functionality of the various programs.

PROCESSING SEABEAM 2100 DATA

The following data processing stream is recommended for data obtained with SeaBeam 2100 series multibeam sonars. SeaBeam 2112 sonars are currently in use on the R/V Knorr (operated by the Woods Hole Oceanographic Institution), the R/V Ronald Brown (operated by NOAA), among other vessels. This same approach is appropriate for data from all other multibeam sonars, with small variations. The issues and differences associated with data from certain other types of sonars are discussed in the following sections.

Consider a data file "sb199411211212.rec" containing one hour’s worth of SeaBeam 2112 data in the vendor format (format 41). This file contains bathymetry, beam amplitude, and sidescan data. The following commands are typical for processing such data and generating preliminary maps.

Step 1: What’s in the data file?

First we run **mbinfo** to obtain statistics about the contents of the data file:

```
mbinfo -I sb199411211212.mb41
```

Seeing reasonable output assures us that we in fact know what kind of data we are processing.

Step 2: Generate ancillary files.

Next, we run **mbdatalist** to generate the statistics (".inf"), quickly read bathymetry (".fbt"), and quickly read navigation (".fnv") files that make many tasks run faster:

```
mbdatalist -I sb199411211212.mb41 \
        -N -V
```

Running this program generates three output files, which we call ancillary files:

```
sb199411211212.mb41.inf
sb199411211212.mb41.fbt
sb199411211212.mb41.fnv
```

Step 3: Generate first cut plota.

We are now set up to process the data. However, first we visually check the data by generating a swath plot of color filled bathymetry overlaid with contours and navigation. This is easily accomplished with **mbm_plot**:

```
mbm_plot -I sb199411211212.mb41 \
        -G1 -C -N -V \
        -O ZSwathBathCont        ZSwathBathCont.cmd
```

Here the ZSwathBathCont represents the plot name, and multiple files will be generated with names constructed by adding different suffixes to this name. The **mbm_plot** command generates a shellscript called ZSwathBathCont.cmd. Running this shellscript in turn generates a Postscript plot called ZSwathBathCont.ps, and then displays the plot to the screen using the Postscript viewer previously specified by the user with **mbdefaults**. The name ZSwathBathCont is descriptive but arbitrary. Users may specify any name they wish.

The types of swath plots produced by **mbm_plot** include:

- color fill bathymetry with contours
- shaded relief color fill bathymetry
- color fill bathymetry overlaid with amplitude
- grayscale amplitude
- grayscale sidescan

We have already generated the first type of plot. We now generate the other four as well:

```
mbm_plot -I sb199411211212.mb41 \
        -G2 -N -V \
        -O ZSwathBathShade
ZSwathBathShade.cmd
mbm_plot -I sb199411211212.mb41 \
        -G3 -S0/1 -N -V \
        -O ZSwathBathAmp
ZSwathBathAmp.cmd
mbm_plot -I sb199411211212.mb41 \
        -G4 -S -N -V \
        -O ZSwathAmp
ZSwathAmp.cmd
mbm_plot -I sb199411211212.mb41 \
```

-G5 -S -N -V \
-O ZSwathSS
ZSwathSS.cmd

We use the -S option to apply histogram equalization to sidescan and amplitude data; note that for the bathymetry overlaid with amplitude map we use -S0/1 so that the the amplitude data used for shading is histogram equalized but the bathymetry is not.

Step 3: Apply Analysis Tools

We now have a reasonable idea of the data quality. There are several data analysis and editing tools that may be used to fix problems (or just to further investigate the data). These may be used in any order, at any time.

Analysis Option A: Automatic Bathymetry Editing.

The program **mbclean** applies some simple artifact detection algorithms to the bathymetry, effectively providing a means of automatically editing the bathymetry. We generally recommend that users edit the bathymetry interactively (see option B below) because no automated filter yet approaches (in our opinion) the ability of the human eye and brain to discern the difference between interesting seafloor morphology and sonar artifact. In particular, none of the filters available in **mbclean** come remotely close to performing satisfactorily in general. However, many users do find it useful to preprocess the data with **mbclean** before editing in the hope that many or most of the artifacts can flagged automatically. Again, we emphasize the importance of not depending solely on automatic filters. **If you care about your data, look at it.**

When we apply **mbclean**, we usually use a filter that flags all soundings that deviate more than a specified fraction of the local median depth from that median depth (the **-G** option). The choice of the filter or filters and the filter parameters used depends very much on the nature of the bathymetry data being processed:

mbclean -I sb199411211212.mb41 -G0.9/1.1 -V

If an "edit save file" named sb199411211212.mb41.esf already exists, **mbclean** reads this file and applies the pre-existing edits prior to beginning its filtering. All pre-existing edit events and all newly generated flags by **mbclean** are output to a new "edit save file", again called sb199411211212.mb41.esf. Since this is the first of the analyis programs to be run on this data file, **mbclean** also creates an **mbprocess** parameter file called sb199411211212.mb41.par which contains all of the parameters and settings to be used by **mbprocess** in generating a processed swath file. If the parameter file already existed, **mbclean** would modify it so that the bathymetry edits would be applied when **mbprocess** is run.

Analysis Option B: Interactive Bathymetry Editing.

We use the interactive graphical tool **mbedit** to check the quality of the bathymetry and to flag artifacts as necessary. We can start **mbedit** with the simple command:

mbedit

and then open a swath file using the pull down menus and dialogs. Alternatively, we can specify the swath file on the command line:

mbedit -I sb199411211212.mb41

While we are editing the bathymetry, all edit events, both flag and unflag, are written to an "edit save file" called sb199411211212.mb41.esf. This file is closed when the <Done> or <Quit> button is clicked. In turn, this file of edit events will be read and these events applied if we run **mbclean**, or **mbprocess**, or if we choose to run **mbedit** again. If this is the first of the analyis programs to be run on this data file, **mbedit** also creates an **mbprocess** parameter file called sb199411211212.mb41.par which contains all of the parameters and settings to be used by **mbprocess** in generating a processed swath file. If the parameter file already existed, **mbedit** would modify it so that the bathymetry edits would be applied when **mbprocess** is run.

Analysis Option C: Editing the navigation.

We use the interactive graphical tool **mbnavedit** to check the quality of the navigation and to fix problems as necessary. We can start **mbnavedit** with the simple command:

mbnavedit

and then open a swath file using the pull down menus and dialogs. Alternatively, we can specify the swath file on the command line:

mbnavedit -I sb199411211212.mb41

When we have completed editing the navigation, we click the <Done> or <Quit> button. The program then writes the final navigation to an "edited navigation" file called sb199411211212.mb41.nve. The program **mbnavedit** also modifies (or creates if needed) an **mbprocess** parameter file and sets it so that the edited navigation is read and merged with the swath data when **mbprocess** is run.

Analysis Option D: Modeling sound velocity profiles.

We use the interactive graphical tool **mbvelocitytool** to model the effect of altering the sound velocity profile (SVP) used to calculate bathymetry from the raw travel times and angles stored in the swath data. We can start **mbvelocitytool** with the simple command:

mbvelocitytool

and then open a swath file using the pull down menus and dialogs. Alternatively, we can specify the swath file on the command line:

mbvelocitytool -I sb199411211212.mb41

If the **Levitus** database has been installed, **mbvelocitytool** will attempt to run **mblevitus** to extract a reference sound velocity profile for the approximate location of the swath data file. This SVP will be displayed along with the editable SVP used for modeling. See the **mbvelocitytool** manual page for details on its operation. If we conclude that the bathymetry data include artifacts associated with having been calculated using an incorrect SVP, and we arrive through modeling at an SVP which is more likely correct, we save this SVP using the <File->Save swath svp file> menu button. Then **mbvelocitytool** saves the edited SVP in a file named sb199411211212.mb41.svp, and also sets (or creates if needed) the parameter file sb199411211212.mb41.par so that **mbprocess** will recalculate the bathymetry by raytracing through this SVP.

Analysis Option E: Correcting sidescan data.

We often find that sidescan imagery, despite the best efforts of sonar manufacturers, is dominated by a systematic variation in amplitude across the swath. Most commonly, the center or nadir region of the swath is characterized by high amplitudes, and the outer swath exhibits much lower amplitudes. This effect can be corrected by **mbprocess** provided that an appropriate model for the variation in amplitude with grazing angle is available. Since the amplitude vs grazing angle function varies with the type of seafloor, we need to construct separate amplitude vs grazing angle correction tables at regular intervals through each swath data file. We use the program **mbbackangle** to construct the amplitude vs. grazing angle tables:

```
mbbackangle -I sb199411211212.mb41 \
        -P25 -N161/80 -V
```

Here a new table is constructed every 25 pings, and the tables will consist of 161 angle bins ranging from -80 degrees to +80 degress grazing angles. The tables are written by **mbbackangle** to an amplitude vs. grazing angle file called sb199411211212.mb41.sga. Of course, the program also sets (or creates if needed) the parameter file sb199411211212.mb41.par so that **mbprocess** will correct the sidescan.

Step 4: Process the data.

The program that actually takes the input, raw swath data and produces processed swath data is **mbprocess**. This program operates using the parameters listed in the parameter file sb199411211212.mb41.par. To process the data, we run

mbprocess -I sb199411211212.mb41

The program produces an output processed swath file called sb199411211212p.mb41 (a 'p' character is inserted in the filename just before the **MB-System** suffix. The program **mbprocess** also automatically generates the three basic ancillary data files for the processed swath file:

```
sb199411211212.mb41.inf
sb199411211212.mb41.fbt
sb199411211212.mb41.fnv
```

Step 5: Grid the bathymetry.

Now use **mbgrid** to grid the bathymetry. The greatest depth in the file is 4502 meters (from the mbinfo output). The 120 degree swath is 3.4 times the water depth wide, or 15.3 km wide. This translates to an average acrosstrack spacing of 15300 m / 120 = 127.5 m. If a region of a grid has more than one data point in each grid cell or bin, we say that this region is "oversampled". If some bins in a region have no data points, we say that this region is "undersampled". We choose a grid cell spacing of 150 m, which will cause the grid to be oversampled towards the center of the swath, but undersampled towards the edges of the swath.

The program **mbgrid** takes a datalist as input, so we first construct that file:

```
echo sb199411211212p.mb41 41 > datalist_grid
```

Now run **mbgrid** using the gaussian weighted mean algorithm (-F1) and longitude and latitude grid cell spacings of 150 m (-E150/150/m) to grid bathymetry (-A1). We use spline interpolation to fill in small gaps in the data no larger than two grid cells (-C2), and we set regions with no data to Nan values for compatibility with GMT programs (-N). We also specify -M so that grids of data density and data standard deviation will be generated:

```
mbgrid -Idatalist_grid -E150/150/m \
        -R114.2208/114.4209/-31.9001/-31.6377 \
        -OZGridBath -A1 -N -C2 -M -V
```

The program **mbgrid** creates a shellscript which, when executed, will generate a color fill plot overlaid with contours of the gridded bathymetry. Now run that shellscript:

```
ZGridBath.cmd
```

Shellscripts have also been created to generate plots of the data density and standard deviation grids:

```
ZGridBath_bath_num.cmd
ZGridBath_bath_sd.cmd
```

Step 6: Mosaic the corrected sidescan data.

The program **mbmosaic** operates in a fashion similar to **mbgrid**, but has special capabilities allowing users to prioritize which parts of the swath are used in the mosaic that are useful for sidescan data. In particular, it is possible to create mosaics which do not use the nadir region of the swath except where no other data is available, or to create mosaics of sidescan data with particular look azimuths. The sidescan data has a higher resolution than the bathymetry so a smaller grid size can be appropriate. However, we wish to overlay the sidescan on the bathymetry, so we use the same grid cell spacing of 150 m.

In order to give the nadir region a lower priority than the outer parts of the sidescan swath, create a file of data priorities (priorities range from 0 to 1) as a function of apparent grazing angle (negative angles are on the port side of the swath, zero is at nadir, and positive angles are on the starboard side of the swath):

```
-60.0 0.2
-45.0 1.0
-15.0 0.8
-14.9 0.1
14.9 0.1
15.0 0.8
45.0 1.0
60.0 0.2
```

Here the nadir region has been set to a low priority and the highest priority has been given to data from an apparent grazing angle of 45 degrees.

Now run **mbmosaic** to mosaic the bathymetry corrected sidescan data (-A4):

```
mbmosaic -Idatalist_grid -E150/150/m \
        -R114.2208/114.4209/-31.9001/-31.6377 \
        -Wangle_priority.dat -F0.10 \
        -OZMosaicSS -A4 -N -C2 -M -V
```

Now run the plot shellscripts to view the gridded sidescan data in grayscale and the ancillary data density and standard deviation plots:

```
ZMosaicSS.cmd
ZMosaicSS_num.cmd
ZMosaicSS_sd.cmd
```

Step 7: Generate additional maps of the gridded data.

First, we use **mbm_grdplot** to generate a color shaded relief view of the bathymetry. We choose to illuminate the bathymetry from the northeast (azimuth of 45 degrees) and to use a shading magnitude of 0.4 (-A0.4/45). Because the data has been gridded as bathymetry (positive down) rather than as topography (positive up), the default plot will have "hot" colors for deep regions and "cold" colors for shallow regions; this is the opposite of the convention we usually use. In order to fix the colors, we have to either rescale the data by multiplying the bathymetry by -1 (accomplished with -MGS-1), or flip the color pallette (accomplished with -D). We use the latter approach. Finally, because this grid is so small, the default shaded relief image is likely to be grainy. To fix this problem, we specify a dots per inch resolution of 72 (-MGQ72); this will take longer and generate a larger plotfile, but the plot will look better. We also use the -L option to specify the title and color scale label for the plot. Here is the command:

```
mbm_grdplot -IZGridBath.grd \
        -G2 -A0.4/45 -D -MGQ72 -V \
        -L"Shaded Relief Bathymetry":"Depth (meters)" \
        -Osb199411211212_bathshade
ZGridBathShade.cmd
```

Second, we use **mbm_grdplot** to generate a color fill view of the bathymetry overlaid with the gridded sidescan. The sidescan overlay is specified using the -K option. We want the colors for the bathymetry to be chosen without histogram equalization, but we also want histogram equalization to be applied to the sidescan data used for shading. To do this, we use -S0/1, where the first number (0) specifies no histogram equalization of the color scale and the second number (1) causes histogram equalization of the shading sidescan data to be implemented. In order to maintain the convention that high sidescan amplitudes are black, we flip both the color pallette (as in the previous example) and the shading scale with -D1/1. We could also flip the shading by specifing a negative shading magnitude (-A-0.4). In this case, we forgo specifying the image resolution, resulting in a grainy plot:

```
mbm_grdplot -IZGridBath.grd \
        -G3 -KZMosaicSS.grd \
        -S0/1 -D1/1 -A0.4 -V \
        -L"Bathymetry Overlaid With Sidescan":"Depth (meters)" \
        -OZGridBathSS
ZGridBathSS.cmd
```

Step 8: Generate 3D perspective views of the gridded data.

Now, generate a 3D perspective view of the gridded bathymetry with shading through synthetic illumination (-G2) using the macro **mbm_grd3dplot**. The grid file is in bathymetry (positive down) rather in topography (positive up), so the bathymetry needs to be rescaled by multiplying by -1 (-MGS-1). We choose an illumination magnitude of 0.4 and an illumination azimuth of 45 degrees (-A0.4/45). We also choose a perspective azimuth of 250 degrees and an elevation of 30 degrees (-E240/30):

```
mbm_grd3dplot -IZGridBath.grd \
        -G2 -A0.4/45 -E250/30 -MGS-1 -V \
        -OZGridBath3D
ZGridBath3D.cmd
```

Now, generate a 3D perspective view of the gridded bathymetry shaded using the gridded sidescan data (-KZMosaicSS.grd). We want the sidescan data to be histogram equalized, so we use -S0/1. We also want the shading to be more prominent than the default shading magnitude of 0.2 would produce, so we use -A0.5:

```
mbm_grd3dplot -IZGridBath.grd \
        -KZMosaicSS.grd \
        -G3 -A0.5 -E250/30 -D0/1 -S0/1 -MGS-1 -V \
        -Osb199411211212_bathss3d
ZGridBathSS3D.cmd
```

PROCESSING SIMRAD MULTIBEAM DATA

The older series of Simrad multibeam sonars (EM100, EM950, EM1000, EM12, EM12D, EM121A) record raw data in files typically named with an "_raw.all" suffix. These older data are supported in **MB-System** using format 51. The data from the newer Simrad multibeams (EM3000, EM2000, EM1002, EM300, EM120) also come in files with the "_raw.all" suffix, but the format is different, and is supported by **MB-System** as format 56. **MB-System** programs can automatically discern the difference between the two formats, and so the initial identification of the data type is easy with **mbinfo**.

However, both formats 51 and 56 lack places to store some information that is important to processing. In particular, the raw Simrad formats do not store beam flags, so the only to apply bathymetry edits is to null the flagged beams. This is not recommended, as it precludes unflagging flagged beams later determined to be good.

Also, the raw Simrad formats store the intial sidescan samples derived from subsampling the bottom returns in each of the formed beams. The locations of these samples on the seafloor are irregular, and the number of samples varies greatly from ping to ping. This raw sidescan is not well suited for mapmaking or processing because of the irregular locations. **MB-System** automatically bins and averages the available sidescan samples into a regularly spaced 1024 pixels with pixel sizes that vary only slightly from ping to ping. This resampled sidescan is what appears in swath plots and in the statistics reported by **mbinfo**. The raw formats have no space to store the rebinned sidescan, so any corrections or other processing applied to the sidescan in formats 51 and 56 is lost.

We have defined a processing format (57) for Simrad multibeam that stores beam flags and rebinned sidescan in addition to all of the information in either of the raw file formats (51 and 56). We strongly recommend that Simrad multibeam data from both old and newer sonars be translated into format 57 before processing. This is accomplished with mbcopy. For example, given a raw EM1000 file called 0021_19960714_123418_raw.all, we translate it to format 57 using:

```
mbcopy -F51/57 -I 0021_19960714_123418_raw.all \
        -O 0021_19960714_123418_raw.mb57
```

Similarly, given a raw EM300 file called 0005_20020425_034057_raw.all, we translate it to format 57 using:

```
mbcopy -F56/57 -I 0005_20020425_034057_raw.all \
        -O 0005_20020425_034057.mb57
```

One further feature of Simrad multibeam data processing should be noted. Given that the raw sidescan samples are derived directly from the bottom returns identified by the sonar for each beam, it follows that whenever a bottom return pick is erroneous, the sidescan samples associated with that beam are also erroneous. Consequently, the binned sidescan shoulde be reprocessed following the application of beam flags to exclude the incorrect sidescan samples from the final imagery. Recalculating the Simrad sidescan is an **mbprocess** option which can set using **mbset**. To turn on sidescan calculation, we use:

```
mbset -I 0005_20020425_034057.mb57 \
        =PSSRECALCMOD:1 -V
```

We can also turn off sidescan recalculation using:

```
mbset -I 0005_20020425_034057.mb57 \
        =PSSRECALCMOD:0 -V
```

In practice, **mbedit** and **mbclean** automatically set the parameter file to turn on sidescan recalculation when writing an "edit save file" for format 57 data, so users should not have to set this parameter manually.

PROCESSING HYDROSWEEP DS DATA

Hydrosweep DS sonars were used on the R/V Maurice Ewing (operated by the Lamont-Doherty Earth Observatory and the R/V Thomas Thompson (operated by the University of Washington) from the early 1990's until 2001. The raw data was logged in a text format (21) that is slow to read and write. We recommend that users confronted with format 21 data translate it to format 24 before processing. Format 24 is a binary data format that contains all of the information in the raw file, but is on the order of 15 times faster to read and write. We accomplish the translation using **mbcopy**:

```
mbcopy -F21/24 -Ihs_ew9204_134.mb21 \
        -Ohsih_ew9204_134.mb24
```

We also strongly recommend that users of Hydrosweep DS data recalculate the bathymetry using raytracing through an appropriate SVP, whether obtained using **mblevitus**, **mbvelocitytool**, or from other sources. The sonar calculates the raw bathymetry using a homogeneous water velocity model, and the results are almost always inferior to those obtained by doing the full raytracing calculations with a correct SVP.

Hydrosweep DS data do not contain sidescan, but they do contain per beam amplitude data. The amplitude data may be corrected in the same fashion as the sidescan. We run **mbbackangle** with the **-A***1* option to operate on the amplitude data:

```
mbbackangle -I sb199411211212.mb41 \
        -A1 -P25 -N161/80 -V
```

and then run **mbprocess** as usual.

PROCESSING HYDROSWEEP MD DATA

Processing Hydrosweep MD data is similar to processing Hydrosweep DS data. In particular, recalculating bathymetry from the travel times is necessary because the sonar uses a homogeneous water velocity model. The difference is simply in the data formats used. The raw Hydrosweep MD data files (typically named with a ".R" suffix) contain only the travel times; the sonar calculated bathymetry is contained in parallel files (typically named with a ".P" suffix). The ".R" files are supported as format 101. Format 102 data files contain bathymetry in addition to the travel times. To translate the data from format 101 to 102 we use:

```
mbcopy -F101/102 -Iys9409040607.R \
        -Oys9409040607.mb102
```

Bathymetry equivalent to that generated by the sonar will automatically be calculated in the copy process (the data stream includes the mean water velocity used by the sonar).

Hydrosweep MD data do not contain sidescan, but they do contain per beam amplitude data. The amplitude data may be corrected in the same fashion as the sidescan. We run **mbbackangle** with the **-A***1* option to operate on the amplitude data:

```
mbbackangle -I sb199411211212.mb41 \
        -A1 -P25 -N161/80 -V
```

and then run **mbprocess** as usual.

SEE ALSO

mbio(l)

BUGS

It doesn't do everything we want it to yet, it doesn't work with every kind of swath data ever collected, and sometimes it breaks.