## NAME

**mbedit** - Interactive editor used to flag bad bathymetry values in swath sonar bathymetry data.

## VERSION

Version 5.0

## SYNOPSIS

**mbedit** [**-B***yr/mo/da/hr/mn/sc* **-D** **-E***yr/mo/da/hr/mn/sc* **-F***format* **-G** **-I***infile* **-X** **-V** **-H]**

## DESCRIPTION

**MBedit** is an interactive editor used to identify and flag artifacts in swath sonar bathymetry data. Once a file has been read in, **MBedit** displays the bathymetry profiles from several pings, allowing the user to identify and flag anomalous beams. In extreme circumstances, entire pings may be nulled (see "KEYBOARD ACTIONS").

In the default mode the edit events are output to an "edit save" file. The program can also be operated in a "browse" mode where no edit events are output. If saved, 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. Users may cause **mbprocess** to be run automatically by specifying the **-X** option when starting **mbedit**.

In previous versions of **MB-System** (version 4.6.10 or earlier), **mbedit** produced output swath data files directly by reading and storing all of the swath data information in memory (including non-survey data records). This approach limited the amount of data that could be handled at once, frequently requiring users to process files in small pieces. The current version only stores the bathymetry data and only outputs changes to the beam flag status (good data or bad data) of bathymetry soundings.

The default **mbedit** display consists of several bathymetry acrosstrack profiles plotted with vertical offsets between each profile (in waterfall fashion). The profiles are plotted within a box with annotation showing the scaling of the x (acrosstrack distance) and y (depth) axes. Alternatively, the display can show the profiles without the waterfall offsets, with the view being either in the alongtrack direction, or in the acrosstrack dimension.

The number of pings displayed, and the width, vertical exageration, and annotation of the plot are all set by the user. Unflagged bathymetry points are shown as small filled black squares, and flagged bathymtetry points are shown as small unfilled red or blue squares. Soundings flagged manually are shown in red, and those flagged by filter algorithms are shown in green. When a sounding is selected in the info mode, it is displayed as a large, filled, blue square. The unflagged or good bathymetry points are connected by black line segments to show the acrosstrack bathymetry profiles for each ping. If the **Show Flagged Profile** toggle is set on, then red line segments connect the flagged depth values in the acrosstrack bathymetry profile. Each of the displayed pings has a label giving the record number in the data file, the ping time, and the center beam depth.

The editing is driven by the left mouse button and involves four basic edit modes and one information mode. In toggle mode each mouse pick toggles the nearest bathymetry point between flagged and unflagged states. In pick mode each mouse pick flags the nearest bathymetry point. In erase mode the left mouse button is held down as the cursor is dragged over the data; all bathymetry points touched by the mouse are flagged. Restore mode is just like erase mode, except that the affected bathymetry values are unflagged. In grab mode, a red rectangle is drawn as the mouse is dragged with the left button down. When the button is released, all unflagged soundings within the rectangle are flagged. Finally, info mode acts like pick mode, except that instead of flagging the nearest sounding, the program displays information about the selected ping and sounding above the bathymetry profile box. A few keyboard macros described below add additional flexibility to the editing process.

**MBedit** has some limited automatic flagging and unflagging options. The **Unflag View** button allows users to unflag all of the currently visible soundings, and the **Unflag Forward** button unflags all soundings from the current position forward to the end of the data held in memory. The **Controls** menu includes a **Filters...** button that brings up a dialog which sets the automatic flagging functions. At present, these include a median filter algorithm, a "wrong side filter", "data cutting" by beam number, and "data cutting" by acrosstrack distance. The "wrong side" filter flags soundings positioned on the "wrong" side of the swath (port soundings incorrectly located on the starboard side, or vice versa). The two "data cutting" algorithms flag beams inside or outside of specified beam number ranges or acrosstrack distance ranges, respectively. Additional automatic filters will be added in later versions.

If the ping includes flagged depths outside the plotting box, the ping label is underlain by a green box. If the ping includes unflagged depths outside the plotting box, the label is underlain by a red box, and a small black box appears to the left of the label. These colored labels help users identify bad bathymetry points which lie outside the box defined by the current vertical exageration and plot width values. In the case where unflagged depths lie outside the plotting box (red label), clicking on the small black box to the left of the ping label automatically flags all of the depth values outside the plotting box. Alternatively, users can decrease the vertical exageration and/or increase the plot width to bring the offending bathymetry points into view. Users should be aware that extreme bathymetric slopes or the use of high vertical exagerations may cause good depth values to lie outside the plotting box for the first and last pings in view.

The middle and right mouse buttons allow the user to step forward and backward, respectively, through the data file. The **Forward** and **Reverse** buttons provide a duplicate stepping capability. Left-handed users may reverse the right and left mouse button functions by pulling down the **Reverse Mouse Buttons** menu item from the **Controls** button.

**MBedit** can hold up to 25000 swath bathymetry profiles at a time in memory. In the case that a data file contains more survey data records than can be held in memory, users will find it necessary to step through multiple buffers of data. The **Next Buffer** button will cause **MBedit** to dump the current buffer contents and read in a new set of data. The handling of buffered data is set using the **Buffer Controls** dialog accessed by pulling down the **Buffer Controls...** menu item from the **Controls** button. The **Buffer Controls** dialog includes two slider controls, one entitled **Data Buffer Size** and the other **Buffer Retain Size**. Users on memory limited machines may find it necessary to set the maximum buffer size to a smaller number using the **Data Buffer Size** slider. The **Buffer Retain Size** slider sets the number of data records retained from the old buffer when more data are loaded.

The user can bring up a **Go To Specified Time** dialog by pulling down the **Go to a specified time...** menu item from the **Controls** button. This dialog allows the user to specify the time of a particular ping to be viewed. The first ping with a time tag later than or equal to the specified time is then displayed, providing such a ping is available. If an appropriate ping is not available in the current buffer, **MBedit** will dump and load buffers of data until such a ping is found or the end of the file is reached. Thus, specifying an incorrect "go to" time may cause **MBedit** to close the file. Caution is advised in using this feature.

**MBedit** creates "edit save" files containing a list of each edit command executed during an editing session. These files are given names consisting of the input filename appended with ".esf". The program also modifies the **mbprocess** parameter file for the input file by specifying the edit save file and setting the **mbprocess** mode to apply the edits when generating a processed swath output file. If no parameter file named "ifile.par" exists, **mbedit** creates one.

If a user seeks to open a swath sonar data file and an associated edit save file already exists, the user is given an option to apply the saved edits to the data as it is loaded. If the user chooses not to apply the saved edits, those edits will be lost as a new edit save file is created. If (as is usually the case) the saved edits are to be applied, **MBedit** copies the edit save file to a file named "mbedit_tmp.esf" and reads the saved edits from that second file. In this way, the use of edit save files and **mbprocess** allows users to edit a swath data file multiple times without generating multiple output files. Subsequent editing sessions serve to update the processed data rather than generate additional edited data files.

## AUTHORSHIP

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

## OPTIONS

**–B** *yr/mo/da/hr/mn/sc*
Sets the starting time for data allowed in the input data; pings with times before the starting time will be ignored. Default: *yr/mo/da/hr/mn/sc* = 1962/2/21/10/30/0.

**–D** Starts up the program in "browse" mode. If a file is opened in browse mode (either at startup or later), none of the edited data will be output to a file. The default is to output the edited data to a file.

**–E** *yr/mo/da/hr/mn/sc*
Sets the ending time for data allowed in the input data; pings with times after the ending time will be ignored. Default: *yr/mo/da/hr/mn/sc* = 2062/2/21/10/30/0.

**–F** *format*
Sets the format at startup for the input and output swath sonar data using **MBIO** integer format identifiers. This value can also be set interactively when specifying the input file. This program uses the **MBIO** library and will read 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.

**–G** This flag causes the program to treat the **Done** button as equivalent to the **Quit** button. This option is used when **MBedit** is started automatically by some other process and only a single file is to edited.

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

**–I** *infile*
Sets the data file from which the input data will be read at startup. This option is usually used only when **MBedit** is started automatically from some other process. The **-F** option should also be used to set the data format id. If the **-B** option is not used to specify browse mode, then the edit save output file is automatically set as *infile* with ".esf" appended.

**–X** This option causes **mbprocess** to run automatically on an input swath data file when the **mbedit** editing session is completed. The **-X** option effectively means that the corresponding processed swath data is generated or updated immediately. The program **mbprocess** will not be invoked if the editing session is in the browse mode.

**–V** Normally, **MBedit** outputs information to the stderr stream regarding the number of records loaded and dumped. If the **-V** flag is given, then **MBedit** works in a "verbose" mode and outputs the program version being used, all error status messages, and a large amount of other information including all of the beams flagged or zeroed.

## INTERACTIVE CONTROLS

**File** This button accesses a pulldown menu with two push buttons: **Open** and **File Selection List**,

**File->Open**
This button brings up a popup window which allows the user to specify an input swath sonar bathymetry data file, its **MBIO** format id, and the output mode. 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. If the swath sonar data file is named using the **MB-System** suffix convention (format 11 files end with ".mb11", format 41 files end with ".mb41", etc.), then the program will automatically use the appropriate format id; otherwise the format must be set by the user. The popup window also allows the output mode to be set to "browse" so that no edit events are output. When a valid file is specified and the **OK** button is clicked, file will be added to an internal list of swath files available for editing, and then that file will be loaded into memory for editing (if another file was already loaded, that file is closed out gracefully before the new file is loaded). If the specified input is a datalist (format id = -1), then all of the files referenced through that datalist will be added to the internal list, and the first of those files loaded. When a file is loaded for editing, **MBedit** reads as much data as will fit into the data buffer (typically 25000 records) and several pings are displayed as stacked bathymetry profiles.

**File->File Selection List**

This button brings up a popup window displaying the internal list of swath files available for editing. The list has four columns. The first (left-most) column is either blank or shows "<locked>" or "<loaded>". If a file is loaded for editing by this program, it shows as "<loaded>". If a file is being edited or processed by another program, it shows as "<locked>". Locked files cannot be opened for editing, and any file loaded for editing by **MBedit** will show as "locked" to other programs. The second column is either blank or shows "<esf>". This indicates if a file has been previously edited so that an edit save file already exists. The third and fourth columns show the filename path and the **MBIO** format id, respectively. Users can select one of the files in the available list. If the "Edit Selected File" button is clicked, the selected file will be opened for editing (and any file already loaded will be closed first). If the "Remove Selected File" button is selected, then the selected file will be removed from the available list.

**View** This button accesses a pulldown menu with several toggle buttons: **Waterfall View**, **Alongtrack View**, **Acrosstrack View**, **Show Flagged Profile**, **Show Bottom Detect Algorithms**, **Wide Bathymetry Profiles**, **Plot Time Stamps**, **Plot Ping Interval**, **Plot Longitude**, **Plot Latitude**, **Plot Heading**, **Plot Speed**, **Plot Center Beam Depth**, **Plot Sonar Altitude**, **Plot Sonar Depth**, **Plot Roll**, **Plot Pitch**, and **Plot Heave**.

The first three toggle buttons set the view mode, which may be a waterfall view, an alongtrack view, or an acrosstrack view. The next two toggle buttons set simple display options and may be set or unset individually. The remaining toggle buttons control the display plot modes; only one mode and therefore one of these toggle buttons may be set at any time.

**View->Waterfall View**

Sets the **mbedit** display so that the bathymetry acrosstrack profiles are plotted with vertical offsets between each profile (in waterfall fashion).

**View->Alongtrack View**

Sets the **mbedit** display so that the bathymetry acrosstrack profiles are viewed in the alongtrack direction without vertical offsets between the pings.

**View->Acrosstrack View**

Sets the **mbedit** display so that the bathymetry acrosstrack profiles are viewed in the acrosstrack direction without vertical offsets between the pings.

**View->Show Flagged Profile**

This toggle allows the user to specify whether the acrosstrack bathymetry profile includes only the unflagged or "good" bathymetry (toggle set to "Off") or whether the profile also includes the flagged or "bad" bathymetry (toggle set to "On"). In the latter case, red line segments show the portion of the profile associated with the flagged depth points.

**View->Show Bottom Detect Algorithms**

This toggle allows the user to specify whether the beams are colored according to their flagging status (toggle set to "Off" or according to the bottom detect algorithm used by the sonar (toggle set to "On"). The default is to color each sounding according to its flagging status - unflagged beams are black, manually flagged beams are red, and filter flagged beams are green. If the beams are colored according to bottom detect algorithm, then amplitude detects are shown in black, phase detects are shown in red, and beams for which the bottom detection algorithm is unknown are shown in green. Users should be aware that many swath data formats do not include bottom detection algorithm information.

**View->Wide Bathymetry Profiles**

This toggle, when set, causes **mbedit** to display the widest possible plots of the bathymetry profiles. All of the other options discussed immediately below narrow the primary plot in order to present additional information on the left side of the window.

**View->Plot Time Stamps**

This toggle, when set, causes **mbedit** to display time stamp information to the left of the bathymetry profiles.

**View->Plot Ping Interval**

This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the time between pings.
This plot appears to the left of the bathymetry profiles.

**View->Plot Longitude**

This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the navigation longitude associated with each ping.
This plot appears to the left of the bathymetry profiles.

**View->Plot Latitude**

This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the navigation latitude associated with each ping.
This plot appears to the left of the bathymetry profiles.

**View->Plot Heading**

This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the heading associated with each ping.
This plot appears to the left of the bathymetry profiles.

**View->Plot Speed**

This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the speed associated with each ping.
This plot appears to the left of the bathymetry profiles.

**View->Plot Center Beam Depth**

This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the speed associated with each ping.
This plot appears to the left of the bathymetry profiles.

**View->Plot Sonar Depth**

This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the sonar depth associated with each ping.
This plot appears to the left of the bathymetry profiles.

**View->Plot Sonar Altitude**

This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the sonar altitude associated with each ping.
This plot appears to the left of the bathymetry profiles.

**View->Plot Roll**

This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the roll associated with each ping. In addition to the roll time series shown in black, an estimate of the acrosstrack seafloor slope is shown in red (calculated by linear regression of the unflagged soundings for each ping), and the difference between the roll and apparent slope is shown in blue. If the sonar roll has been correctly applied, the roll and seafloor slope should be uncorrelated. If there is a timing error in the roll correction, then the apparent seafloor slope may correlate strongly with the roll.

This plot appears to the left of the bathymetry profiles.

**View->Plot Pitch**
This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the pitch associated with each ping.
This plot appears to the left of the bathymetry profiles.

**View->Plot Heave**
This toggle, when set, causes **mbedit** to display an automatically scaled vertical time series plot of the heave associated with each ping.
This plot appears to the left of the bathymetry profiles.

**Controls**
This button accesses a pulldown menu with six items: **Go To Specified Time...**, **Buffer Controls...**, **Annotation...**, **Filters...**, **Reverse Right/Left Key Macros**, and **Reverse Mouse Buttons**. The first (top) four items bring up a dialog of the same name. These dialogs are discussed below. The last two items are toggle buttons which set mouse button and key macro behaviors.

**Controls->Go To Specified Time...**
This menu item brings up a dialog which allows the user to specify the time of a particular ping to be displayed. Once the year, month, day, hour, minute, and second values are entered, clicking the **Apply** button causes **mbedit** to seek the specified target time. If the current data buffer begins after the target time, an error is returned. If the target time is later than the end of the current data buffer, then **mbedit** will dump and load buffers until the target time is reached or the data file ends. If the end of the file is reached during the search, the file will be closed.

**Controls->Buffer Controls...**
This menu item brings up a dialog which allows the user to set the data buffer handling through two sliders discussed immediately below.

**Controls->Buffer Controls->Data Buffer Size**
This slider on the **Buffer Controls** dialog sets the number of data records which can be held in the data buffer. Any change becomes effective the next time that a data file is read in.

**Controls->Buffer Controls->Buffer Retain Size**
This slider on the **Buffer Controls** dialog sets the number of data records which are held over in the buffer each time the old buffer is written out.

**Controls->Annotation...**
This menu item brings up a dialog which allows the user to set the annotation intervals for the across track distance and depth axes through the two sliders discussed immediately below.

**Controls->Annotation->X Axis Tick Interval**
This slider on the **Annotation** dialog sets the tick interval in m for the across track distance axis. If a particular value is desired which cannot be obtained by dragging the slider, the slider can be changed by increments of 1 by clicking with the left button inside the slider area, but not on the slider itself.

**Controls->Annotation->Y Axis Tick Interval**
This slider on the **Annotation** dialog sets the tick interval in m for the depth axis. If a particular value is desired which cannot be obtained by dragging the slider, the slider can be changed by increments of 1 by clicking with the left button inside the slider area, but not on the slider itself.

**Controls->Filters**
This menu item brings up a dialog which allows the user to turn automatic bathymetry filtering algorithms on and off and to set the filter parameters through the widgets discussed immediately below. Clicking on the **Apply** button first removes any previous automatic filter flags and then applies the current filter settings to the swath bathymetry data. These actions are applied only from the current position to the end of the data currently in memory. The filtering is not applied (or unapplied) to pings earlier in the data file than the current position. The **Reset** button restores the filter settings that were applied last.

**Controls->Filters->Median Spike FIlter**
This toggle button turns automatic median filtering of bathymetry profiles on and off. When median filtering is enabled, the median depth is calculated for each bathymetry ping. Each sounding in the ping is compared with the median value; any sounding that differs from the median by a percentage greater than a threshold percentage is flagged as bad data.

**Controls->Filters->% Median Depth Threshold**
This slider sets the threshold of the median depth filter in units of percentage of the median depth.

**Controls->Filters->Wrong Side FIlter**
This toggle button turns automatic wrong side filtering of bathymetry profiles on and off. This filter expects bathymetry beams to be numbered from port to starboard. Any beam numbered less than the center beam with a positive (starboard) acrosstrack distance is considered to be on the "wrong side", and any beam numbered higher than the center beam with a negative (port) acrosstrack distance is similarly regarded. For most swath data, the center beam occurs near the midway point in the beam list. This algorithm ignores apparent wrong side beams close to the midway beam, where close is defined as being within a threshold number of the midway beam.

**Controls->Filters->Beams from Center Threshold**
This slider sets the threshold of beam locations checked by the wrong side filter.

**Controls->Filters->Flag by Beam Number**
This toggle button turns automatic beam flagging by beam number on and off. The flagging is controlled by start and end beam number values (set by the two sliders discussed immediately below). If the start beam number is less than or equal to the end beam number, then all beams between and including the start and end are flagged, producing a flagged zone within the swath. If the start beam number is greater than the end beam number, then all beams before and including the end beam are flagged, and all beams including and after the start beam are flagged. This produces flagging of both swath edges.

**Controls->Filters->Start Flagging Beam Number**
This slider sets the beam number at which flagging by beam number starts.

**Controls->Filters->End Flagging Beam Number**
This slider sets the beam number at which flagging by beam number ends.

**Controls->Filters->Flag by Distance**
This toggle button turns automatic beam flagging by acrosstrack distance on and off. The flagging is controlled by start and end distance values (set by the two sliders discussed immediately below). If the start distance is less than or equal to the end distance, then all beams with acrosstrack distances between the start and end distances are flagged, producing a flagged zone within the swath. If the start distance is greater than the end distance, then all beams with acrosstrack distance less than the end distance are flagged, and all beams with acrosstrack distance greater than the start distance are flagged. This produces flagging of both swath edges.

**Controls->Filters->Start Flagging Distance**
This slider sets the beam number at which flagging by acrosstrack distance starts.

**Controls->Filters->End Flagging Distance**
This slider sets the beam number at which flagging by acrosstrack distance ends.

**Controls->Reverse Right/Left Key Macros**
This toggle button, when set, reverses the key macros associated with flagging all beams to the left or right of the last picked beam. This means that the ’A’, ’a’, ’J’, and ’j’ keys will flag to the right rather than to the left. Similarly, the ’D’, ’d’, ’L’, and ’l’ keys will flag to the left rather than to the right. This option allows users to conveniently handle swath bathymetry in which the beams are mistakenly ordered starboard to port instead of the usual port to starboard.

**Controls->Reverse Mouse Buttons**
This toggle button, when set, reverses the meaning of the right and left mouse buttons. This option is for the convenience of left-handed users.

**Start** This button causes the set of displayed pings to step backward to the beginning of the current buffer. The middle mouse button causes the same action if the 'G' or 'g' key is down.

**Reverse**
This button causes the set of displayed pings to step *nstep* pings backward in the current buffer. The middle mouse button causes the same action.

**Forward**
This button causes the set of displayed pings to step *nstep* pings forward in the current buffer. The right mouse button causes the same action.

**End** This button causes the set of displayed pings to step forward to the end of the current buffer. The right mouse button causes the same action if the 'G' or 'g' key is down.

**Next Buffer**
This button causes the program to write out the data from the current buffer and then read in and display the next buffer. If there is no more data to be read in after the current buffer has been written out, then the input and output files are closed.

**Done | Next File**
When the last file in the list of files available for editing has been loaded, this button shows as "Done". Otherwise, it shows as "Next File". In either case, this button causes the program to write out all of the edited navigation data and then close the current file. If the current file is not the last one in the available list, then the next unlocked file will be loaded for editing.

**Quit** This button causes the program to exit gracefully. If a data file has been loaded, all of the edited navigation will be written to the output file before exiting.

**About** This button causes the program to bring up a dialog showing the program's name, version, and authors.

**Acrosstrack Width**
This slider sets the width of the plot in meters; in general this value should be slightly larger than the swath width of the data being edited. If a particular value is desired which cannot be obtained by dragging the slider (e.g., the user wants a plot width of 10 meters but the slider jumps from 1 to 47), the slider can be changed by increments of 1 by clicking with the left button inside the slider area, but not on the slider itself.

**Vertical Exageration**
This slider sets the depth scale in terms of vertical exageration. The depth scale will change as the cross track distance scale is changed to maintain the same vertical exageration. If a particular value is desired which cannot be obtained by dragging the slider, the slider can be changed by increments of 0.01 by clicking with the left button inside the slider area, but not on the slider itself.

**Mode** This set of radio buttons allows the user to specify the edit mode. If mode is set to *Toggle*, then clicking the left mouse button will cause the nearest beam to toggle between flagged and unflagged. If mode is set to *Pick*, then clicking the left mouse button will cause the nearest unflagged beam to be flagged. If mode is set to *Erase*, then the cursor will change to an erasor and any beam with the cursor while the left mouse button is held down will be flagged. If mode is set to *Restore*, the behavior will be the same as for *Erase* except that the affected beams will be unflagged instead of flagged. In *Grab* mode, a red rectangle is drawn as the mouse is dragged with the left button down. When the button is released, all unflagged soundings within the rectangle are flagged. *Info* mode acts like pick mode, except that instead of flagging the nearest sounding, the program displays information about the selected ping and sounding above the bathymetry profile box. The edit mode can also be set using key macros (see the keyboard action section):

Toggle: 'Q', 'q', 'U', 'u'
Pick: 'W', 'w', 'I', 'i'
Erase: 'E', 'e', 'O', 'o'
Restore: 'R', 'r', 'P', 'p'
Grab: 'Y', 'y', '}, ']'

Info: 'T', 't', '{', '['

**Unflag View**

This button flags all unflagged beams among the currently displayed pings. Pings in the buffer before or after the current display are unaffected.

**Unflag View**

This button unflags all flagged beams among the currently displayed pings. Pings in the buffer before or after the current display are unaffected.

**Unflag Forward**

This button unflags all flagged beams among all pings from the start of the current display to the end of the current data buffer. Pings before the start of the current display are unaffected.

**Number of pings shown**

This slider sets the number of pings shown at a time.

**Number of pings to step**

This slider sets the number of pings to step when the **Forward** or **Reverse** buttons are pushed.

## MOUSE ACTIONS

**Left Mouse Button**

The left mouse button is used to pick beams. Good beams are shown as filled black squares and bad (flagged) beams as unfilled red or green squares. The result of picking a particular beam depends on the current edit mode, as set by the **Mode** button or keyboard macros defined below. The last picked beam (and ping) is remembered for use with some of the keyboard actions described below.

**Middle Mouse Button**

The middle mouse button causes the set of displayed pings to step *nstep* pings backward in the current buffer. The control button **Reverse** causes the same action. If the 'G' or 'g' key is depressed, the display will jump to the beginning of the current buffer.

**Right Mouse Button**

The right mouse button causes the set of displayed pings to step *nstep* pings forward in the current buffer. The control button **Forward** causes the same action. If the 'G' or 'g' key is depressed, the display will jump to the end of the current buffer.

## KEYBOARD ACTIONS

**'G' or 'g'**

**Big Jump:** Scolling forward while pressing one of these keys jumps to the end of the current buffer, and scrolling backward jumps to the beginning of the buffer.

**'Z', 'z', 'M', or 'm'**

**Bad Ping:** Pressing one of these keys causes all of the beams in the last picked ping to be flagged as bad.

**'X', 'x', '<', or ','**

**Right:** Pressing one of these keys causes all of the unflagged beams in the current display to to be flagged as bad. This is equivalent to the CFlag View **button.**

**'C', 'c', '>', or '.'**

**Right:** Pressing one of these keys causes all of the flagged beams in the current display to to be unflagged. This is equivalent to the **Unflag View** button.

**'S', 's', 'K', or 'k'**

**Good Ping:** Pressing one of these keys causes all of the beams in the last picked ping to be unflagged as good.

**’A’, ’a’, ’J’, or ’j’**
: **Left:** Pressing one of these keys causes all of the beams including and to the left of the last picked beam to be flagged as bad.

**’D’, ’d’, ’L’, or ’l’**
: **Right:** Pressing one of these keys causes all of the beams including and to the right of the last picked beam to be flagged as bad.

**’!’** **Zero Ping:** Pressing this key causes all of the beams in the ping associated with the last picked beam to be zeroed. This should be used only for completely ridiculous values, as the values are not recoverable.

**’Q’, ’q’, ’U’, or ’u’**
: **Toggle Mode:** Pressing one of these keys sets the edit mode to "toggle" so that clicking the left mouse button will cause the nearest beam to toggle between flagged and unflagged. The edit mode can also be set using the **Mode** toggle buttons.

**’W’, ’w’, ’I’, or ’i’**
: **Pick Mode:** Pressing one of these keys sets the edit mode to "pick" so that clicking the left mouse button will cause the nearest unflagged beam to be flagged. The edit mode can also be set using the **Mode** toggle buttons.

**’E’, ’e’, ’O’, or ’o’**
: **Erase Mode:** Pressing one of these keys sets the edit mode to "erase" so that clicking the left mouse button will cause any beam under the cursor while the left mouse button is held down to be flagged. The edit mode can also be set using the **Mode** toggle buttons.

**’R’, ’r’, ’P’, or ’p’**
: **Restore Mode:** Pressing one of these keys sets the edit mode to "restore" so that clicking the left mouse button will cause any beam under the cursor while the left mouse button is held down to be unflagged. The edit mode can also be set using the **Mode** toggle buttons.

**’T’, ’t’, ’{’, or ’[’**
: **Grab Mode:** Pressing one of these keys sets the edit mode to "grab" so that dragging the left mouse button will draw a red rectangle. Any beams inside the rectangle when the left mouse button is released will be flagged. The edit mode can also be set using the **Mode** toggle buttons.

**’Y’, ’y’, ’}’, or ’]’**
: **Info Mode:** Pressing one of these keys sets the edit mode to "info" so that clicking the left mouse button will cause information about the nearest beam to be displayed above the bathymetry profile box. The edit mode can also be set using the **Mode** toggle buttons.

**’2’** Sets the **mbedit** display so that the bathymetry acrosstrack profiles are plotted with vertical offsets between each profile (in waterfall fashion).

**’3’** Sets the **mbedit** display so that the bathymetry acrosstrack profiles are viewed in the alongtrack direction without vertical offsets between the pings.

**’4’** Sets the **mbedit** display so that the bathymetry acrosstrack profiles are viewed in the acrosstrack direction without vertical offsets between the pings.

## SEE ALSO

**mbsystem**(l), **mbprocess**(l), **mbset**(l), **mbclean**(l), **mbeditviz**(l), **mbinfo**(l)

## BUGS

This program is only fun when the data are good. The main window for **mbedit** is not resizeable.